Renji Tao

Finite Automata and Application to Cryptography

TSINGHUA UNIVERSITY PRESS

Renji Tao

Finite Automata and Application to Cryptography

Foreword

The important summarizing work of RENJI TAO appears now in book form. It is a great pleasure for me to see this happen, especially because I have known Professor Tao as one of the very early contributors to public-key cryptography. The research community has missed a book such as the present one now published by Tsinghua University Press and Springer. The book will be of special interest for students and researchers in the theories of finite automata, cryptography and error correcting codes.

One of the phenomena characterizing the second half of the last century is the rapid growth of computer science and informatics in general. The theory of finite automata, models of computing devices with a finite non-extensible memory, was initiated in the 1940s and 1950s, mainly by McCulloch, Pitts and Kleene. It has found numerous applications in most diverse areas, as exemplified by the series of yearly international conferences in implementation and applications of finite automata. The present work by Professor Tao develops a theory and contains strong results concerning invertible finite automata: the input sequence can be recovered from the output sequence. This is a desirable feature both in cryptography and error correcting codes. The book considers various types of invertibility and, for instance, the effect of bounded delay to invertibility.

Cryptography, secret writing, has grown enormously both in extent and importance and quality during the past few decades. This is obvious in view of the fact that so many transactions and so much confidential information are nowadays sent over the Internet. After the introduction of public-key cryptography by Diffie and Hellman in the 1970s, many devices were tried and applied for the construction of public-key cryptosystems. Professor Tao was one of such initiators in applying invertible finite automata. Although mostly in Chinese, his work was known also in the West. I referred to it already some twenty years ago. Later on, for instance, a PhD thesis was written about this topic in my university.

Many of the results in this book appear now for the first time in book form. The book systematizes important and essential results, as well as gives a comprehensive list of references. It can be used also as a starting point for further study. Different parts of the book are of varying importance for students and researchers, depending on their particular interests. Professor Tao gives useful guidelines about this in his Preface.

Much of the material in this book has not been previously available for western researchers. As a consequence, some of the results obtained by Professor Tao and his group already in the late 1970s have been independently rediscovered later. This concerns especially shift register sequences, for instance, the decimation sequence and the linear complexity of the product sequence.

I feel grateful and honored that Professor Tao has asked me to write this preface. I wish success for the book.

Turku, Finland, January 2008 *Arto Salomaa*

Preface

Automata theory is a mathematical theory to investigate behavior, structure and their relationship to discrete and digital systems such as algorithms, nerve nets, digital circuits, and so on. The first investigation of automata theory goes back to A. M. Turing in 1936 for the formulation of the informal idea of algorithms. Finite automata model the discrete and digital systems with finite "memory", for example, digital circuits. The theory of finite automata has received considerable attention and found applications in areas of computer, communication, automatic control, and biology, since the pioneering works of Kleene, Huffman, and Moore in the 1950s. Among others, autonomous finite automata including shift registers are used to generate pseudo-random sequences, and finite automata with invertibility are used to model encoders and decoders for error correcting and cipher as well as to solve topics in pure mathematics such as the Burnside problem for torsion groups. This book is devoted to the invertibility theory of finite automata and its application to cryptography. The book also focuses on autonomous finite automata and Latin arrays which are relative to the canonical form for one key cryptosystems based on finite automata.

After reviewing some basic concepts and notations on relation, function and graph, Chap. 1 gives the concept of finite automata, three types of "invertibility" for finite automata, and proves the equivalence between "feedforward invertibility" and "boundedness of decoding error propagation" which is the starting point of studying one key cryptosystems based on finite automata; a tool using labelled trees to represent states of finite automata is also given. In addition, some results on linear finite automata over finite fields are reviewed, in preparation for Chap. 7. Chapter 2 analyzes finite automata from the aspects of minimal output weight and input set. Results for weakly invertible finite automata are in return applied to establish the mutual invertibility for finite automata, and to evaluate complexity of searching an input given an output and an initial state for a kind of weakly invertible finite automata. In Chap. 3 the R_a R_b transformation method is presented for generating a kind of weakly invertible finite automata and correspondent weak inverse finite automata which are used in key generation in Chap. 9; this method is also used to solve the structure problem for quasi-linear finite automata over finite fields. Chapter 4 first discusses the relation between two linear R_a R_b transformation sequences and "composition" of R_a R_b trans-

formation sequences, then the relation of inversion by R_a R_b transformation method with inversion by reduced echelon matrix method and by canonical diagonal matrix polynomial method. Chapter 5 deals with the structure problem of feedforward inverse finite automata. Explicit expressions of feedforward inverse finite automata with delay $\leqslant 2$ are given. The result for delay 0 lays a foundation for the canonical form of one key cryptosystems based on finite automata in Chap. 8. In Chap. 6, for any given finite automaton which is invertible (weakly invertible, feedforward invertible, an inverse, or a weak inverse, respectively), the structure of all its inverses (weak inverses, weak inverses with bounded error propagation, original inverses, or original weak inverses, respectively) is characterized. Chapter 7 deals with autonomous linear finite automata over finite fields. Main topics contain representation of output sequences, translation, period, linearization, and decimation. The final two chapters discuss the application to cryptography. A canonical form for one key cryptosystems which can be implemented by finite automata without plaintext expansion and with bounded decoding error propagation is given in Chap. 8. As a component of the canonical form, the theory of Latin array is also dealt with. Chapter 9 gives a public key cryptosystem based finite automata and discusses its security. Some generalized cryptosystems are also given.

The material of this book is mainly taken from the works of our research group since the 1970s, except some basic results, for example, on linear finite automata and on partial finite automata. Of course, this book does not contain all important topics on invertibility of finite automata which our research group have investigated such as decomposition of finite automata and linear finite automata over finite rings. Results presented here other than Chaps. 1 and 7 are appearing for the first time in book form; Chapter 7 is appearing for the first time in English which is originally published in [97] and in Chap. 3 of the monograph [98]. This book is nearly self-contained, but algebra is required as a mathematical background in topics on linear finite automata, linear R_a R_b transformation, and Latin array; the reader is referred to, for example, [16], or [42] for matrix theory, [142] for finite group.

This book pursues precision in logic, which is extremely important for a mathematical theory. For automata theorists and other mathematicians interested merely in the invertibility theory of finite automata, the readers may read Chap. 1 to Chap. 6 and propose easily open problems on topics concerned. For an algebraist interested in the theory of shift register sequences, taking a glance at Chap. 7 is, at least to avoid overlap of research, harmless. A mathematician majoring in combinatory theory may be interested in Sects. 8.2 and 8.3 of Chap. 8.

For readers interested merely in one key cryptography, it is enough to read Chap. 1 (except Subsect. 1.2.3 and Sect. 1.6), the first two sections of Chap. 5, Chap. 7, and Chap. 8.

For readers interested merely in public key cryptography, they may read Chap. 1 (except Subsect. 1.2.4 and Sects. 1.3 and 1.5), Chap. 2 (except Sect. 2.2), Chap. 3 (except Sect. 3.3), Chap. 4, Sects. 6.1 and 6.5 of Chap. 6, and Chap. 9. They may skip over all proofs if they believe them to be correct; but a generation algorithm of finite automata satisfying the condition PI is directly obtained from several proofs in the first two sections of Chap. 3.

I would like to thank Zuliang Huang for his continuous encouragement and suggestions about the investigation on finite automata since the 1960s. Thanks also go to Peilin Yan, the late first director of Institute of Computing Technology, Chinese Academy of Sciences, and to Kongshi Xu, the first director of Institute of Software, Chinese Academy of Sciences, for their support and providing a suitable environment for me to do theoretical research since the 1970s. I am also grateful to many of my colleagues and students for various helpful discussions and valuable suggestions. My thanks go to Hongji Wang for his careful reading and commenting on the manuscript. Naturally, I have to take responsibility for any errors that may occur in this book. My special thanks go to Hui Xue for her continuing thorough and helpful editorial commentary, and careful polishing the manuscript. Finally, I thank my wife Shihua Chen and my daughter Xuemei Chen for their patience and continuous encouragement.

Beijing, May 2007 *Renji Tao*

Contents

1. Introduction

Renji Tao

Institute of Software, Chinese Academy of Sciences
Beijing 100080, China trj@ios.ac.cn

Summary.

Finite automata are a mathematical abstraction of discrete and digital systems with finite "memory". From a behavior viewpoint, such a system is a transducer which transforms an input sequence to an output sequence with the same length. Whenever the input sequence can be retrieved by the output sequence (and initial internal state), the system is with invertibility and may be used as an encoder in application to cipher or error correcting.

The invertibility theory of finite automata is dealt within the first six chapters of this book. In the first chapter, the basic concepts on finite automata are introduced. The existence of (weak) inverse finite automata and boundedness of delay for (weakly) invertible finite automata are proven in Sect. 1.4, and the coincidence between feedforward invertibility and bounded error propagation is presented in Sect. 1.5. In Sect. 1.7, we characterize the structure of (weakly) invertible finite automata by means of their state tree. In addition, there is a section that reviews some basic results of linear finite automata, as an introduction to Chap. 7.

Key words: *finite automata, invertible, weakly invertible, feedforward invertible, inverse, weak inverse, feedforward inverse, error propagation, state tree*

Finite automata are a mathematical abstraction of discrete and digital systems with finite "memory". From a structural viewpoint, such a system has an input and an output as well as an "internal state". Its time system is discrete (say, moments $0, 1, \ldots$). Only finite possible values can be taken by the input (output and internal state, respectively) at each moment. And, the output at the current moment and the internal state at the next moment can be uniquely determined by the input and the internal state at the current moment. From a behavior viewpoint, such a system is a transducer which transforms an input sequence to an output sequence with the same length.

Whenever the input sequence can be retrieved by the output sequence (and the initial internal state), the system is with invertibility and may be used as an encoder in application to cipher or error correcting.

The invertibility theory of finite automata is dealt within the first six chapters. In the first chapter, the basic concepts on finite automata are introduced. The existence of (weak) inverse finite automata and boundedness of delay for (weakly) invertible finite automata are proven in Sect. 1.4, and the coincidence between feedforward invertibility and bounded error propagation is presented in Sect. 1.5. In Sect. 1.6, we characterize the structure of (weakly) invertible finite automata by means of their state tree. In addition, there is a section that reviews some basic results of linear finite automata, as an introduction to Chap. 7.

1.1 Preliminaries

We begin with a brief excursion through some fundamental concepts. A reader acquainted with the notation used may skip this section. We will assume a familiarity with the most basic notions of set theory, such as membership $\in$, set-builder notation $\{\cdots \mid \cdots\}$ or $\{\cdots : \cdots\}$, empty set $\varnothing$, subset $\subseteq$, union $\cup$, intersection $\cap$, difference $\setminus$.

1.1.1 Relations and Functions

For any sets $A_1, A_2, \ldots, A_n$, the *Cartesian product* of $A_1, A_2, \ldots, A_n$ is the set

$$\{(a_1, a_2, \ldots, a_n) \mid a_i \in A_i, i = 1, 2, \ldots, n\},$$

denoted by $A_1 \times A_2 \times \cdots \times A_n$ (sometimes $(a_1, a_2, \ldots, a_n)$ is replaced by $\langle a_1, a_2, \ldots, a_n \rangle$). In the case of $A_i = A, i = 1, 2, \ldots, n$, $A_1 \times A_2 \times \cdots \times A_n$ is called the n-fold Cartesian product of A and is abbreviated to A^n. For any $i, 1 \leqslant i \leqslant n$, the i-th component of an element $(a_1, a_2, \ldots, a_n)$ in $A_1 \times A_2 \times \cdots \times A_n$ means a_i.

Let A and B be two sets. A *relation* R from A to B is a subset R of $A \times B$. If (a, b) is in the relation R, it is written as aRb. If (a, b) is not in the relation R, it is written as $a \not R\, b$. In the case of $A = B$, R is also called a *relation* on A.

A relation R on a set A is an *equivalence relation*, if the following conditions hold: (a) R is reflexive, i.e., $(a, a) \in R$ for any a in A; (b) R is symmetric, i.e., $(a, b) \in R$ implies $(b, a) \in R$ for any a and b in A; and (c) R is transitive, i.e., $(a, b) \in R$ and $(b, c) \in R$ imply $(a, c) \in R$ for any a, b and c in A.

Let R be an equivalence relation on A. For any a in A, the set $[a]_R = \{b \mid b \in A, (a,b) \in R\}$ is called the *equivalence class* containing a. The set $\{[a]_R \mid a \in A\}$ is called the *equivalence classes* of R.

Let A be a set and $\pi = \{H_i \mid i \in I\}$ be a family of subsets of A. If (a) $\cup_{i \in I} H_i = A$ and (b) $H_i \cap H_j = \varnothing$ for any different i and j in I, π is called a *partition* of A, and H_i, $i \in I$ are called *blocks* of the partition π.

Clearly, the equivalence classes of an equivalence relation on A define a partition. Conversely, a partition $\{H_i \mid i \in I\}$ of A determines an equivalence relation R on A in the following way:

$$(a,b) \in R \Leftrightarrow \exists i \in I(a \in H_i \ \& \ b \in H_i),$$
$$a, b \in A.$$

It is convenient to identify an equivalence relation with its partition.

Let R be a relation from A to B. The subset

$$\{a \in A \mid \exists b \in B((a,b) \in R)\}$$

of A is called the *domain* of R, and the subset

$$\{b \in B \mid \exists a \in A((a,b) \in R)\}$$

of B is called the *range* of R.

Suppose that R is a relation from A to B. Define a relation R^{-1} from B to A as follows:

$$(a,b) \in R^{-1} \Leftrightarrow (b,a) \in R,$$
$$a \in A, b \in B.$$

R^{-1} is called the *inverse relation* of R. Clearly, the domain of R and the range of R^{-1} are the same; the domain of R^{-1} and the range of R are the same.

Let R be a relation from A to B. If, for any a in A, any b and b' in B, $(a,b) \in R$ and $(a,b') \in R$ imply $b = b'$, R is called a *partial function* from A to B.

A single-valued *function* (*mapping*) from A to B is a partial function R from A to B such that the domain of R is A. A single-valued function from a set to itself is also called a *function* or a *transformation* on the set.

Let f be a single-valued mapping or partial function from A to B. For any a in the domain of f, the unique element in B, say b, satisfying $(a,b) \in f$ is written as $f(a)$, and is called the *value* of f at (the point) a. For any a not in the domain of f, we say that the value of f at (the point) a is *undefined*. For any relation R from A to B and any a in A, we also use $R(a)$ to denote the set $\{b \in B \mid (a,b) \in R\}$. Clearly, $R^{-1}(b) = \{a \in A \mid (b,a) \in R^{-1}\} = \{a \in A \mid (a,b) \in R\}$ for any b in B.

Let f be a single-valued mapping from A to B. If the range of f is B, f is called a *surjection*, or to be *surjective*, or a single-valued mapping from A *onto* B. If $f(a) \neq f(a')$ holds for any different elements a and a' in A, f is called an *injection*, or to be *injective*, or to be *one-to-one*. If f is injective and surjective, f is called a *bijection*, or to be *bijective*, or a *one-to-one* mapping from A *onto* B. If there exists a one-to-one mapping from A onto B, A is said to be *one-to-one correspondent* with B. A bijection from a finite set to itself is also called a *permutation* on the set, or of its elements.

If f is a partial or single-valued function from A to B, the inverse relation f^{-1} is also called *the inverse function* of f. Thus $f^{-1}(b) = \{a \in A \mid (a, b) \in f\} = \{a \in A \mid f(a) = b\}$. For any b in B, whenever $|f^{-1}(b)| = 1$, we also use $f^{-1}(b)$ to denote the unique element, say a, in $f^{-1}(b)$, where $f(a) = b$; from the context, the reader can easily understand the meaning of the notation without ambiguity. It is easy to see that if f is a bijection from A to B, then f^{-1} is a bijection from B to A and $f^{-1}(f(a)) = a$ for any $a \in A$, $f(f^{-1}(b)) = b$ for any $b \in B$.

An injection f from A to B is also called an *invertible* function, or an *invertible* transformation in the case of $A = B$; a partial or single-valued function g from B to A is called *an inverse function*, or *an inverse transformation* in the case of $A = B$, of f, if $g(f(a)) = a$ holds for any $a \in A$. For any partial or single-valued function f_i from A_i to B_i, $i = 1, 2$, if $A_2 \subseteq A_1$, $B_2 \subseteq B_1$ and $f_1(a) = f_2(a)$ for any $a \in A_2$, f_2 is called a *restriction* of f_1 (on A_2). We use $f_1|_{A_2}$ to denote a restriction of f_1 on A_2. Clearly, if g is an inverse function of f, then the inverse function f^{-1} of f is a restriction of g. We also use f^{-1} to denote an inverse function of f; from the context, the reader can easily understand the meaning of the notation without ambiguity.

A *vector function* of dimension n in s variables over F means a single-valued function from the s-fold Cartesian product of F (respectively an s-dimensional vector space over F) to the n-fold Cartesian product of F (respectively an n-dimensional vector space over F). For a vector function φ of dimension n in s variables over F, its value at the point $(x_1, \ldots, x_s)$ is usually expressed as $\varphi(x_1, \ldots, x_s)$; for any i, $1 \leqslant i \leqslant n$, the i-th component function of φ is a single-valued function from the s-fold Cartesian product of F (respectively an s-dimensional vector space over F) to F of which the value at each point $(x_1, \ldots, x_s)$ is the i-th component of $\varphi(x_1, \ldots, x_s)$. A vector function over $\{0, 1\}$ is called a *Boolean vector function*. A Boolean function means a Boolean vector function of dimension 1. A Boolean function $\varphi(x_1, \ldots, x_s)$ in s variables can be expressed by a polynomial of $x_1, \ldots, x_s$; if the degree of the polynomial is greater than 1, φ is said to be *nonlinear*. The Boolean function in s variables of which all values are 0 is called the *zero Boolean*

function in s variables. The function on A of which the value at each a in A equals a is called the *identity function* on A.

1.1.2 Graphs

We will discuss some fundamental concepts of graph. (V, Γ) is called a (*directed*) *graph*, if $\Gamma \subseteq V \times V$ for a nonempty set V. V is called the *vertex set*, and elements in V are called *vertices*. Γ is called the *arc set* or the *directed edge set*, and elements in Γ are called *arcs* or *directed edges*. For an arc $u = (a, b) \in \Gamma$, a is called the *initial vertex* of u, and b the *terminal vertex* of u.

Let $w = u_1 u_2 \dots u_i \dots$ be a finite or infinite sequence of arcs, where $u_i \in \Gamma$, $i = 1, 2, \dots$ If the terminal vertex of u_i is the initial vertex of u_{i+1} for any u_i, u_{i+1} in w, w is called a *path* of the graph (V, Γ). The number of arcs in w is called the *length* of the path w. The initial vertex of u_1 is called the *initial vertex* of the path w; and the terminal vertex of u_n is called the *terminal vertex* of the path w if the length of the path w is n.

If $w = u_1 u_2 \dots u_n$ is a path of the graph (V, Γ) and the terminal vertex of the arc u_n is the initial vertex of the arc u_1, the path w is called a *circuit* of the graph (V, Γ). Evidently, if there exists a circuit, then there exists a path of infinite length.

For any vertex a, the set $\{b|(b, a) \in \Gamma, b \in V\}$ is called the *incoming vertex set* of a, and the set $\{b|(a, b) \in \Gamma, b \in V\}$ is called the *outgoing vertex set* of a.

A vertex of which both the incoming vertex set and the outgoing vertex set are empty is called an *isolated vertex*.

We define recurrently the levels of vertices as follows. For any vertex a in V, if the incoming vertex set of a is empty, the *level* of a is defined to be 0. For any vertex a in V, if the levels of all vertices in the incoming vertex set of a have been defined and the maximum is h, the *level* of a is defined to be $h + 1$.

For any arc $u = (a, b)$, if levels of a and b have been defined, the *level* of the arc u is defined to be the level of the vertex a.

If the level of each vertex of (V, Γ) is defined and the maximum is h, we say that the graph *has level*, and the *level* of the graph is defined to be $h - 1$.

Clearly, if each vertex of (V, Γ) is an isolated vertex, then the level of the graph is -1.

If V is finite, the graph (V, Γ) is said to be *finite*.

Notice that for a finite graph, it has no circuit if and only if it has level, and the maximum of its path-lengths equals its level plus 1 if it has level.

It is convenient for some applications to introduce the *empty graph*. The vertex set and the arc set of the empty graph can be regarded as the empty set. The level of the empty graph is defined to be -2.

Two graphs (V, Γ) and (V', Γ') are said to be *isomorphic*, if there exists a one-to-one mapping φ from V onto V' such that (a, b) is an arc of (V, Γ) if and only if $(\varphi(a), \varphi(b))$ is an arc of (V', Γ'). Any such mapping φ is called an *isomorphism* from (V, Γ) to (V', Γ'). An isomorphism from a graph to itself is called an *automorphism* of the graph.

A graph (V', Γ') is called a *subgraph* of a graph (V, Γ), if $V' \subseteq V$ and $\Gamma' \subseteq \Gamma$.

A graph (V, Γ) is called a *tree* with root v, if the following conditions hold: (a) each vertex $(\neq v)$ is a terminal vertex of a unique arc; (b) v is not a terminal vertex of any arc; and (c) (V, Γ) has no circuit.

A vertex of a tree is called a *leaf*, if no arc emits from the vertex, i.e., the outgoing vertex set of the vertex is empty.

Let (V, Γ) and (V', Γ') be two trees. If (V', Γ') is a subgraph of (V, Γ), (V', Γ') is called a *subtree* of (V, Γ).

Let G be a (directed) graph (respectively tree). If an element of some set is assigned to each arc of G, or if an element of some set is assigned to each arc of G and an element of some set is assigned to each vertex of G, G is called a *labelled graph* (respectively *labelled tree*). The element assigned to an arc (respectively a vertex) is referred to as the arc (respectively vertex) label of the arc (respectively vertex).

1.2 Definitions of Finite Automata

1.2.1 Finite Automata as Transducers

For any set A, the concatenation of elements in A, say $a_0 a_1 \dots a_{l-1}$, is called a *word* (or a *finite sequence*) over A, and l its *length*, where $a_0, a_1, \dots, a_{l-1}$ are elements in A. In the case of $l = 0$, $a_0 a_1 \dots a_{l-1}$ is a void sequence which contains no element. The void sequence is called the *empty word* and its length is 0. We use ε to denote the empty word (void sequence), and $|\alpha|$ the length of a word α. The set of all the words over A including the empty word is denoted by A^*. If $a_0, a_1, \dots, a_n, \dots$ are elements in A, the concatenation of the infinite elements $a_0 a_1 \dots a_n \dots$ is called an *infinite-length word* or an *ω-word* (or an *infinite sequence*) over A. We use A^ω to denote the set of all infinite-length words over A. We also use A^n to denote the set of all words over A of length n for any nonnegative integer n.

Let $\alpha = a_0 a_1 \dots a_{m-1}$ and $\beta = b_0 b_1 \dots b_{n-1}$ be two words in A^*. The concatenation of α and β is $a_0 a_1 \dots a_{m-1} b_0 b_1 \dots b_{n-1}$, which is also a word in

A^* of length $m+n$, and is denoted by $\alpha\cdot\beta$, or $\alpha\beta$ for short. Clearly, $\alpha\cdot\varepsilon = \varepsilon\cdot\alpha = \alpha$. Similarly, if $\alpha = a_0a_1\dots a_{m-1}$ is in A^* and $\beta = b_0b_1\dots b_{n-1}\dots$ in A^ω, then the concatenation of α and β is the element $a_0a_1\dots a_{m-1}b_0b_1\dots b_{n-1}\dots$ in A^ω which is also denoted by $\alpha\cdot\beta$, or $\alpha\beta$ for short. Clearly, $\varepsilon\cdot\beta = \beta$. β is called a *prefix* of α, if there exists γ such that $\alpha = \beta\gamma$. β is called a *suffix* of α, if there exists γ such that $\alpha = \gamma\beta$. For any $U, V \subseteq A^*$, the concatenation of U and V is the set $\{\alpha\beta \mid \alpha \in U, \beta \in V\}$, denoted by UV.

A *finite automaton* is a quintuple $\langle X, Y, S, \delta, \lambda\rangle$, where X, Y and S are nonempty finite sets, δ is a single-valued mapping from $S \times X$ to S, and λ is a single-valued mapping from $S \times X$ to Y. X, Y and S are called the *input alphabet*, the *output alphabet* and the *state alphabet* of the finite automaton, respectively; and δ and λ are called the *next state function* and the *output function* of the finite automaton, respectively.

Expand the domain of δ to $S \times X^*$ as follows. For any state s_0 in S and any $l(>0)$ input letters $x_0, x_1, \dots, x_{l-1}$ in X, we compute recurrently states $s_1, \dots, s_l$ in S by

$$s_{i+1} = \delta(s_i, x_i), \quad i = 0, 1, \dots, l-1,$$

and define

$$\delta(s_0, x_0x_1\dots x_{l-1}) = s_l.$$

In the case of $l = 0$, we define

$$\delta(s_0, \varepsilon) = s_0.$$

Expand the domain of λ to $S \times (X^* \cup X^\omega)$ and the range of λ to $Y^* \cup Y^\omega$ as follows. For any state s_0 in S and any $l(>0)$ input letters $x_0, x_1, \dots, x_{l-1}$ in X, we define

$$\lambda(s_0, x_0x_1\dots x_{l-1}) = y_0y_1\dots y_{l-1},$$

where

$$y_i = \lambda(\delta(s_0, x_0x_1\dots x_{i-1}), x_i), \quad i = 0, 1, \dots, l-1.$$

In the case of $l = 0$, we define

$$\lambda(s_0, \varepsilon) = \varepsilon.$$

For any state s_0 in S and any infinite input letters $x_0, x_1, \dots$ in X, we define

$$\lambda(s_0, x_0x_1\dots) = y_0y_1\dots,$$

where

$$y_i = \lambda(\delta(s_0, x_0x_1 \ldots x_{i-1}), x_i), \quad i = 0, 1, \ldots$$

From the definitions, it is easy to see that

$$\delta(s, \alpha\beta) = \delta(\delta(s, \alpha), \beta), \quad s \in S,\ \alpha, \beta \in X^*$$

and

$$\lambda(s, \alpha\beta) = \lambda(s, \alpha)\lambda(\delta(s, \alpha), \beta), \quad s \in S,\ \alpha \in X^*,\ \beta \in X^* \cup X^\omega. \tag{1.1}$$

Notice that each state s of the finite automaton determines a single-valued mapping λ_s from $X^* \cup X^\omega$ to $Y^* \cup Y^\omega$, where

$$\lambda_s(\alpha) = \lambda(s, \alpha), \quad \alpha \in X^* \cup X^\omega.$$

λ_s is called the *automaton mapping* of s of which the restriction $\lambda_s|_{X^*}$ is a single-valued mapping from X^* to Y^* and the restriction $\lambda_s|_{X^\omega}$ is a single-valued mapping from X^ω to Y^ω. From (1.1), it is evident that $\lambda_s|_{X^*}$ and $\lambda_s|_{X^\omega}$ can be determined by each other. A single-valued mapping φ from X^* to Y^* is said to be *sequential*, if $|\varphi(\alpha)| = |\alpha|$ for any $\alpha \in X^*$ and $\varphi(\beta)$ is a prefix of $\varphi(\alpha)$ for any $\alpha \in X^*$ and any prefix β of α. From the definition of λ and (1.1), $\lambda_s|_{X^*}$ is sequential.

Example 1.2.1. Let $X = Y = \{0, 1\}$ and $S = \{0, 1\}^n = \{\langle a_1, \ldots, a_n\rangle | a_1, \ldots, a_n = 0, 1\}$. Define

$$\begin{aligned}
&\delta(\langle a_1, \ldots, a_n\rangle, x) = \langle a_2, \ldots, a_n, f(a_1, \ldots, a_n, x)\rangle,\\
&\lambda(\langle a_1, \ldots, a_n\rangle, x) = g(a_1, \ldots, a_n, x),\\
&\qquad a_1, \ldots, a_n, x = 0, 1,
\end{aligned}$$

where f and g are two single-valued mappings from $\{0, 1\}^{n+1}$ to $\{0, 1\}$. Then $\langle X, Y, S, \delta, \lambda\rangle$ is a finite automaton. We use $BSR_{f,g}$ to denote the finite automaton. The name is an abbreviation of the phrase "Binary Shift Register with feedback function f and mixer g ". Given $s_0 = \langle a_{-n}, \ldots, a_{-1}\rangle \in S$ and $x_i \in X$, $i = 0, 1, \ldots$, let

$$\lambda(s_0, x_0x_1 \ldots x_i \ldots) = y_0y_1 \ldots y_i \ldots$$

for some $y_i \in Y$, $i = 0, 1, \ldots$ Then

$$y_i = g(a_{i-n}, \ldots, a_{i-1}, x_i), \quad i = 0, 1, \ldots$$

and

$$\delta(s_0, x_0x_1 \ldots x_{i-1}) = \langle a_{i-n}, \ldots, a_{i-1}\rangle, \quad i = 0, 1, \ldots,$$

where

$$a_i = f(a_{i-n}, \ldots, a_{i-1}, x_i), \quad i = 0, 1, \ldots$$

A special case is of interest, where $f(a_1, \ldots, a_n, x)$ does not depend on x, that is, f is a single-valued mapping from $\{0,1\}^n$ to $\{0,1\}$, and g can be expressed as

$$g(a_1, \ldots, a_n, x) = g'(a_1, \ldots, a_n) \oplus x,$$

$\oplus$ standing for the exclusive-or operation, that is, the addition modulo 2 operation. This automaton is called the binary steam cipher in cryptology community. The key-sequence is generated by a shift register with feedback function f and output logic g'; the mixer of the key and the input is the addition modulo 2. Given $s_0 = \langle a_{-n}, \ldots, a_{-1} \rangle \in S$ and $x_i \in X$, $i = 0, 1, \ldots$, let

$$\lambda(s_0, x_0 x_1 \ldots x_i \ldots) = y_0 y_1 \ldots y_i \ldots$$

for some $y_i \in Y$, $i = 0, 1, \ldots$ Then

$$y_i = g'(a_{i-n}, \ldots, a_{i-1}) \oplus x_i, \quad i = 0, 1, \ldots$$

and

$$\delta(s_0, x_0 x_1 \ldots x_{i-1}) = \langle a_{i-n}, \ldots, a_{i-1} \rangle, \quad i = 0, 1, \ldots,$$

where

$$a_i = f(a_{i-n}, \ldots, a_{i-1}), \quad i = 0, 1, \ldots$$

Let $M_i = \langle X_i, Y_i, S_i, \delta_i, \lambda_i \rangle$, $i = 1, 2$ be two finite automata. For any $s_i \in S_i$, $i = 1, 2$, s_1 and s_2 are said to be *equivalent*, denoted by $s_1 \sim s_2$, if $X_1 = X_2$ and for any $\alpha \in X_1^*$, $\lambda_1(s_1, \alpha) = \lambda_2(s_2, \alpha)$ holds.

Let λ_{i,s_i} be the automaton mapping of s_i, $i = 1, 2$. Consider $\lambda_{i,s_i}|_{X_i^*}$ as a mapping from X_i^* to $(Y_1 \cup Y_2)^*$. Then $s_1 \sim s_2$ if and only if $\lambda_{1,s_1}|_{X_1^*} = \lambda_{2,s_2}|_{X_2^*}$. Consider $\lambda_{i,s_i}|_{X_i^\omega}$ as a mapping from X_i^ω to $(Y_1 \cup Y_2)^\omega$. Since $\lambda_{i,s_i}|_{X_i^*}$ and $\lambda_{i,s_i}|_{X_i^\omega}$ are determined by each other, we have that $s_1 \sim s_2$ if and only if $\lambda_{1,s_1}|_{X_1^\omega} = \lambda_{2,s_2}|_{X_2^\omega}$. Therefore, $s_1 \sim s_2$ if and only if $\lambda_{1,s_1} = \lambda_{2,s_2}$. In other words, $s_1 \sim s_2$ if and only if for any $\alpha \in X_1^\omega (= X_2^\omega)$, $\lambda_1(s_1, \alpha) = \lambda_2(s_2, \alpha)$ holds, if and only if for any $\alpha \in X_1^* \cup X_1^\omega (= X_2^* \cup X_2^\omega)$, $\lambda_1(s_1, \alpha) = \lambda_2(s_2, \alpha)$ holds.

From the definition, it is easy to show that the relation $\sim$ is reflexive, symmetric and transitive.

If $s_1 \sim s_2$ and $\alpha \in X_1^*$, then $\delta(s_1, \alpha) \sim \delta(s_2, \alpha)$. In fact, since $s_1 \sim s_2$, for any $\beta \in X_1^*$, we have $\lambda_1(s_1, \alpha) = \lambda_2(s_2, \alpha)$ and $\lambda_1(s_1, \alpha\beta) = \lambda_2(s_2, \alpha\beta)$. From

$$\lambda_i(s_i, \alpha\beta) = \lambda_i(s_i, \alpha)\lambda_i(\delta_i(s_i, \alpha), \beta), \ i = 1, 2,$$

it follows that

$$\lambda_1(s_1,\alpha)\lambda_1(\delta_1(s_1,\alpha),\beta) = \lambda_2(s_2,\alpha)\lambda_2(\delta_2(s_2,\alpha),\beta).$$

Thus

$$\lambda_1(\delta_1(s_1,\alpha),\beta) = \lambda_2(\delta_2(s_2,\alpha),\beta).$$

We conclude that $\delta(s_1,\alpha) \sim \delta(s_2,\alpha)$.

M_2 is said to be *stronger* than M_1, denoted by $M_1 \prec M_2$, if for any state s_1 in S_1, there exists a state s_2 in S_2 such that $s_1 \sim s_2$. M_1 and M_2 are said to be *equivalent*, denoted by $M_1 \sim M_2$, if $M_1 \prec M_2$ and $M_2 \prec M_1$. Clearly, the relation $\prec$ is reflexive and transitive, and the relation $\sim$ on finite automata is reflexive, symmetric and transitive.

A finite automaton is said to be *minimal*, if any different states of it are not equivalent.

M_1 and M_2 are said to be *isomorphic*, if $X_1 = X_2$, $Y_1 = Y_2$ and there exists a one-to-one mapping φ from S_1 onto S_2 such that

$$\varphi(\delta_1(s_1,x)) = \delta_2(\varphi(s_1),x),\ \lambda_1(s_1,x) = \lambda_2(\varphi(s_1),x),\quad s_1 \in S_1,\ x \in X_1.$$

φ is called an *isomorphism* from M_1 to M_2.

If M_1 and M_2 are isomorphic, then M_1 and M_2 are equivalent. In fact, since M_1 and M_2 are isomorphic, there exists an isomorphism φ from M_1 to M_2. We prove by induction on the length of α that

$$\lambda_1(s,\alpha) = \lambda_2(\varphi(s),\alpha) \tag{1.2}$$

holds for any s in S_1 and any α in X_1^*. *Basis* : $|\alpha| = 0$, i.e., $\alpha = \varepsilon$. Since

$$\lambda_1(s,\alpha) = \varepsilon = \lambda_2(\varphi(s),\alpha)$$

holds for any s in S_1, (1.2) holds for any s in S_1 and $\alpha = \varepsilon$. *Induction step* : Suppose that we have proven that (1.2) holds for any s in S_1 and any α in X_1^* of length n. Given $\alpha \in X_1^*$ of length $n+1$, let $\alpha = x\alpha'$, where $x \in X_1$. Then the length of α' is n. Since M_1 and M_2 are isomorphic, we have

$$\varphi(\delta_1(s,x)) = \delta_2(\varphi(s),x)$$

and

$$\lambda_1(s,x) = \lambda_2(\varphi(s),x).$$

From the induction hypothesis, we have

$$\lambda_1(\delta_1(s,x),\alpha') = \lambda_2(\varphi(\delta_1(s,x)),\alpha').$$

Thus

$$\begin{aligned}\lambda_1(s,\alpha) &= \lambda_1(s,x\alpha') = \lambda_1(s,x)\lambda_1(\delta_1(s,x),\alpha') \\ &= \lambda_2(\varphi(s),x)\lambda_2(\varphi(\delta_1(s,x)),\alpha') \\ &= \lambda_2(\varphi(s),x)\lambda_2(\delta_2(\varphi(s),x),\alpha') \\ &= \lambda_2(\varphi(s),x\alpha') = \lambda_2(\varphi(s),\alpha).\end{aligned}$$

Therefore, (1.2) holds for any s in S_1 and any α in X_1^* of length $n+1$. We conclude that (1.2) holds for any s in S_1 and any α in X_1^*. Thus $s \sim \varphi(s)$, $s \in S_1$. Since φ is surjective, M_1 and M_2 are equivalent.

Conversely, if M_1 and M_2 are minimal and equivalent, and $Y_1 = Y_2$, then M_1 and M_2 are isomorphic. In fact, since M_1 and M_2 are equivalent, we have $X_1 = X_2$ and for any state s in S_1 we can find a state s' in S_2 with $s \sim s'$. Let φ be the relation $\sim$ from S_1 to S_2. Since M_2 is minimal, φ is single-valued. Since M_1 and M_2 are equivalent, φ is a mapping from S_1 onto S_2. To prove φ is one-to-one, suppose that $\varphi(s_1) = \varphi(s_2)$. Since $s_i \sim \varphi(s_i)$, $i = 1, 2$, we have $s_1 \sim s_2$. Since M_1 is minimal, this yields $s_1 = s_2$. Thus φ is one-to-one. We conclude that φ is a one-to-one mapping from S_1 onto S_2. To prove that M_1 and M_2 are isomorphic, it is sufficient to prove that φ is an isomorphism. Since $s \sim \varphi(s)$ holds for any s in S_1, for any x in X_1, we have

$$\lambda_1(s,x) = \lambda_2(\varphi(s),x)$$

and $\delta_1(s_1,x) \sim \delta_2(\varphi(s),x)$. The latter yields

$$\varphi(\delta_1(s_1,x)) = \delta_2(\varphi(s),x).$$

Therefore, φ is an isomorphism from M_1 to M_2.

M_1 is called a *finite subautomaton* of M_2, denoted by $M_1 \leqslant M_2$, if $X_1 \subseteq X_2$, $Y_1 \subseteq Y_2$, $S_1 \subseteq S_2$, and

$$\begin{aligned}&\delta_1(s,x) = \delta_2(s,x),\ \ \lambda_1(s,x) = \lambda_2(s,x), \\ &\qquad s \in S_1,\ \ x \in X_1.\end{aligned}$$

For any finite automaton $M = \langle X, Y, S, \delta, \lambda\rangle$, any nonempty subset X' of X and any nonempty subset S' of S, if $\delta(S', X') = \{\delta(s,x) \mid s \in S', x \in X'\} \subseteq S'$, S' is said to be *closed* with respect to X' in M. Clearly, given M_2, for any nonempty subset X_1 of X_2, any nonempty subset S_1 of S_2, and any nonempty subset Y_1 of Y_2, if S_1 is closed with respect to X_1 in M_2 and $\lambda_2(S_1, X_1) = \{\lambda_2(s,x) \mid s \in S_1, x \in X_1\} \subseteq Y_1$, then

$$\langle X_1, Y_1, S_1, \delta_2|_{S_1\times X_1}, \lambda_2|_{S_1\times X_1}\rangle$$

is a finite subautomaton of M_2.

For any states s and s' of a finite automaton $\langle X, Y, S, \delta, \lambda\rangle$, if there exists $x \in X$ such that $s' = \delta(s,x)$, s' is called a *successor state* of s; if s' is a successor state of s, s is called a *predecessor state* of s'.

1.2.2 Special Finite Automata

We give definitions of several special finite automata.

Let $M = \langle X, Y, S, \delta, \lambda\rangle$ be a finite automaton. If for any s in S, $\delta(s, x)$ and $\lambda(s, x)$ do not depend on x, M is said to be *autonomous*. We abbreviate the autonomous finite automaton to a quadruple $\langle Y, S, \delta, \lambda\rangle$, where δ is a single-valued mapping from S to S, and λ is a single-valued mapping from S to Y.

Example 1.2.2. Let $Y = \{0, 1\}$ and $S = \{0, 1\}^n = \{\langle a_1, \ldots, a_n\rangle | a_1, \ldots, a_n = 0, 1\}$. Define

$$\begin{aligned}&\delta(\langle a_1, \ldots, a_n\rangle) = \langle a_2, \ldots, a_n, f(a_1, \ldots, a_n)\rangle,\\ &\lambda(\langle a_1, \ldots, a_n\rangle) = g(a_1, \ldots, a_n),\\ &\quad a_1, \ldots, a_n = 0, 1,\end{aligned}$$

where f and g are two single-valued mappings from $\{0, 1\}^n$ to $\{0, 1\}$. Then $\langle Y, S, \delta, \lambda\rangle$ is an autonomous finite automaton. We use $BASR_{f,g}$ to denote the autonomous finite automaton. The name is an abbreviation of the phrase "Binary Autonomous Shift Register with feedback function f and output function g".

Let $M_i = \langle Y_i, S_i, \delta_i, \lambda_i\rangle$, $i = 1, 2$ be two autonomous finite automata. M_1 is called an (*autonomous*) *finite subautomaton* of M_2, denoted by $M_1 \leqslant M_2$, if $Y_1 \subseteq Y_2$, $S_1 \subseteq S_2$ and

$$\begin{aligned}&\delta_1(s) = \delta_2(s),\ \lambda_1(s) = \lambda_2(s),\\ &\quad s \in S_1.\end{aligned}$$

For any autonomous finite automaton $M = \langle Y, S, \delta, \lambda\rangle$ and any nonempty subset S' of S, if $\delta(S') = \{\delta(s) \mid s \in S'\} \subseteq S$, S' is said to be *closed* in M. Clearly, given M_2, for any nonempty subset S_1 of S_2 and any nonempty subset Y_1 of Y_2, if S_1 is closed in M_2 and $\lambda_2(S_1) = \{\lambda_2(s) \mid s \in S_1\} \subseteq Y_1$, then $\langle Y_1, S_1, \delta_2|_{S_1}, \lambda_2|_{S_1}\rangle$ is an autonomous finite subautomaton of M_2.

For any single-valued mapping f from $Y^k \times X^{h+1}$ to Y, where h and k are nonnegative integers, and X and Y are two nonempty finite sets, we use M_f to denote a finite automaton defined by

$$y_i = f(y_{i-1}, \ldots, y_{i-k}, x_i, \ldots, x_{i-h}), \quad i = 0, 1, \ldots$$

More precisely, $M_f = \langle X, Y, Y^k \times X^h, \delta, \lambda\rangle$, where

$$\begin{aligned}&\delta(\langle y_{-1}, \ldots, y_{-k}, x_{-1}, \ldots, x_{-h}\rangle, x_0)\\ &\quad = \langle y_0, y_{-1}, \ldots, y_{-k+1}, x_0, x_{-1}, \ldots, x_{-h+1}\rangle,\end{aligned}$$

$$\begin{aligned}&\lambda(\langle y_{-1},\dots,y_{-k},x_{-1},\dots,x_{-h}\rangle,x_0)=y_0,\\&\quad y_0=f(y_{-1},\dots,y_{-k},x_0,x_{-1},\dots,x_{-h}),\\&\quad y_{-1},\dots,y_{-k}\in Y,\ x_0,x_{-1},\dots,x_{-h}\in X.\end{aligned}$$

M_f is called the (h,k)-*order memory finite automaton* determined by f. In the case of $k=0$, M_f is called the h-*order input-memory finite automaton* determined by f.

Let f and g be single-valued mappings from $Y^k\times U^{p+1}\times X^{h+1}$ to Y and U, respectively, where h and k are nonnegative integers, $p\geqslant -1$ is an integer, X, Y and U are nonempty finite sets. We use $M_{f,g}$ to denote a finite automaton defined by

$$\begin{aligned}&y_i=f(y_{i-1},\dots,y_{i-k},u_i,\dots,u_{i-p},x_i,\dots,x_{i-h}),\\&u_{i+1}=g(y_{i-1},\dots,y_{i-k},u_i,\dots,u_{i-p},x_i,\dots,x_{i-h}),\\&\quad i=0,1,\dots\end{aligned}$$

More precisely, $M_{f,g}=\langle X,Y,Y^k\times U^{p+1}\times X^h,\delta,\lambda\rangle$, where

$$\begin{aligned}&\delta(s,x_0)=\langle y_0,\dots,y_{-k+1},u_1,\dots,u_{-p+1},x_0,\dots,x_{-h+1}\rangle,\\&\lambda(s,x_0)=y_0,\\&\quad y_0=f(y_{-1},\dots,y_{-k},u_0,\dots,u_{-p},x_0,\dots,x_{-h}),\\&\quad u_1=g(y_{-1},\dots,y_{-k},u_0,\dots,u_{-p},x_0,\dots,x_{-h}),\\&\quad s=\langle y_{-1},\dots,y_{-k},u_0,\dots,u_{-p},x_{-1},\dots,x_{-h}\rangle\in Y^k\times U^{p+1}\times X^h,\\&\quad x_0\in X.\end{aligned}$$

$M_{f,g}$ is called the (h,k,p)-*order pseudo-memory finite automaton* determined by f and g. Clearly, in the case of $p=-1$, $M_{f,g}$ degenerates to M_f.

Let $M_a=\langle Y_a,S_a,\delta_a,\lambda_a\rangle$ be an autonomous finite automaton, f a single-valued mapping from $X^{c+1}\times\lambda_a(S_a)$ to Y. We use $\mathcal{SIM}(M_a,f)$ to denote a finite automaton $\langle X,Y,X^c\times S_a,\delta,\lambda\rangle$, where

$$\begin{aligned}&\delta(\langle x_{-1},\dots,x_{-c},s_a\rangle,x_0)=\langle x_0,x_{-1},\dots,x_{-c+1},\delta_a(s_a)\rangle,\\&\lambda(\langle x_{-1},\dots,x_{-c},s_a\rangle,x_0)=f(x_0,x_{-1},\dots,x_{-c},\lambda_a(s_a)),\\&\quad x_0,x_{-1},\dots,x_{-c}\in X,\ s_a\in S_a.\end{aligned}$$

$\mathcal{SIM}(M_a,f)$ is referred to as a *c-order semi-input-memory finite automaton* determined by M_a and f.

Clearly, if $f(x_0,\dots,x_{-c},t)$ does not depend on t, then $\mathcal{SIM}(M_a,f)$ degenerates to the input-memory finite automaton $M_{f'}$, where $f'(x_0,\dots,x_{-c})=f(x_0,\dots,x_{-c},t)$, t being an arbitrarily given element in $\lambda_a(S_a)$.

A finite automaton $\langle X, Y, S, \delta, \lambda \rangle$ is said to be *linear* over the finite field $GF(q)$, if X, Y and S are vector spaces over $GF(q)$ of dimensions l, m and n, respectively, δ is a linear mapping from $S \times X$ to S, and λ is a linear mapping from $S \times X$ to Y, where l, m and n are some nonnegative integers.

In the case where X, Y and S consist of all column vectors over $GF(q)$ of dimensions l, m and n, respectively, δ may be given by an $n \times n$ matrix A and an $n \times l$ matrix B over $GF(q)$, and λ may be given by an $m \times n$ matrix C and an $m \times l$ matrix D over $GF(q)$, that is,

$$\begin{aligned} \delta(s, x) &= As + Bx, \\ \lambda(s, x) &= Cs + Dx, \\ s &\in S,\ x \in X. \end{aligned}$$

The matrices A, B, C, D are called *structure matrices* of the finite automaton and l, m, n *structure parameters* of the finite automaton. A is referred to as the *state transition matrix* of the finite automaton. In the autonomous case, its structure matrices are A, C and its structure parameters are m, n.

1.2.3 Compound Finite Automata

For any two finite automata $M_i = \langle X_i, Y_i, S_i, \delta_i, \lambda_i \rangle$, $i = 1, 2$ with $Y_1 = X_2$, we use $C(M_1, M_2)$ to denote the superposition of M_1 and M_2, i.e., the finite automaton $\langle X_1, Y_2, S_1 \times S_2, \delta, \lambda \rangle$, where

$$\begin{aligned} \delta(\langle s_1, s_2 \rangle, x) &= \langle \delta_1(s_1, x), \delta_2(s_2, \lambda_1(s_1, x)) \rangle, \\ \lambda(\langle s_1, s_2 \rangle, x) &= \lambda_2(s_2, \lambda_1(s_1, x)), \\ s_1 &\in S_1, s_2 \in S_2, x \in X_1. \end{aligned}$$

Another kind of combination of finite automata may be defined as follows. Let g be a single-valued mapping from $U^r \times V^{p+1}$ to U, and f a single-valued mapping from W^{t+1} to V. $C'(M_f, M_g) = \langle W, U, U^r \times W^{p+t}, \delta, \lambda \rangle$ is a $(p+t, r)$-order memory finite automaton defined by

$$\begin{aligned} u_i &= g(u_{i-1}, \ldots, u_{i-r}, f(w_i, \ldots, w_{i-t}), \ldots, f(w_{i-p}, \ldots, w_{i-p-t})), \\ &\quad i = 0, 1, \ldots, \end{aligned}$$

that is,

$$\begin{aligned} &\delta(\langle u_{-1}, \ldots, u_{-r}, w_{-1}, \ldots, w_{-p-t} \rangle, w_0) \\ &\quad = \langle u_0, \ldots, u_{-r+1}, w_0, \ldots, w_{-p-t+1} \rangle, \\ &\lambda(\langle u_{-1}, \ldots, u_{-r}, w_{-1}, \ldots, w_{-p-t} \rangle, w_0) = u_0, \\ &\quad u_0 = g(u_{-1}, \ldots, u_{-r}, f(w_0, \ldots, w_{-t}), \ldots, f(w_{-p}, \ldots, w_{-p-t})), \\ &\quad w_0, w_{-1}, \ldots, w_{-p-t} \in W,\ u_{-1}, \ldots, u_{-r} \in U. \end{aligned}$$

Theorem 1.2.1. *Let* $s = \langle u_{-1}, \ldots, u_{-r}, w_{-1}, \ldots, w_{-p-t}\rangle$ *be a state of* $C'(M_f, M_g)$. *Let* $s_f = \langle w_{-1}, \ldots, w_{-t}\rangle$ *and* $s_g = \langle u_{-1}, \ldots, u_{-r}, v_{-1}, \ldots, v_{-p}\rangle$, *where* $v_i = f(w_i, \ldots, w_{i-t})$, $i = -1, \ldots, -p$. *Then the state* $\langle s_f, s_g\rangle$ *of* $C(M_f, M_g)$ *and* s *are equivalent. Moreover, if*

$$v_{-p} \ldots v_{-1} v_0 v_1 \ldots = \lambda_f(\langle w_{-p-1}, \ldots, w_{-p-t}\rangle, w_{-p} \ldots w_{-1} w_0 w_1 \ldots) \quad (1.3)$$

and

$$u_0 u_1 \ldots = \lambda_g(\langle u_{-1}, \ldots, u_{-r}, v_{-1}, \ldots, v_{-p}\rangle, v_0 v_1 \ldots), \quad (1.4)$$

then

$$u_0 u_1 \ldots = \lambda(\langle u_{-1}, \ldots, u_{-r}, w_{-1}, \ldots, w_{-p-t}\rangle, w_0 w_1 \ldots), \quad (1.5)$$

where λ_f, λ_g *and* λ *are output functions of* M_f, M_g *and* $C'(M_f, M_g)$, *respectively.*

Proof. Suppose that (1.3) and (1.4) hold. Then

$$v_i = f(w_i, \ldots, w_{i-t}),\ i = -p, \ldots, -1, 0, 1, \ldots,$$

and

$$u_i = g(u_{i-1}, \ldots, u_{i-r}, v_i, \ldots, v_{i-p}),\ i = 0, 1, \ldots$$

It immediately follows that

$$u_i = g(u_{i-1}, \ldots, u_{i-r}, f(w_i, \ldots, w_{i-t}), \ldots, f(w_{i-p}, \ldots, w_{i-p-t})),$$
$$i = 0, 1, \ldots$$

Thus (1.5) holds.

For any $w_0, w_1, \ldots$ in W, suppose that

$$u_0 u_1 \ldots = \lambda_g(s_g, \lambda_f(s_f, w_0 w_1 \ldots)).$$

Then there exist $v_0, v_1, \ldots$ in V such that

$$v_0 v_1 \ldots = \lambda_f(s_f, w_0 w_1 \ldots).$$

It follows that

$$u_0 u_1 \ldots = \lambda_g(s_g, v_0 v_1 \ldots).$$

Therefore, (1.3) and (1.4) hold. From the result proven in the preceding paragraph, (1.5) holds. Thus the state $\langle s_f, s_g\rangle$ of $C(M_f, M_g)$ and the state s of $C'(M_f, M_g)$ are equivalent. □

For $n \geqslant 1$, we use $C'(M_0, M_1, \ldots, M_n)$ to denote $C'(C'(M_0, M_1, \ldots, M_{n-1}), M_n)$, and $C(M_0, M_1, \ldots, M_n)$ to denote $C(C(M_0, M_1, \ldots, M_{n-1}), M_n)$. For $n = 0$, $C'(M_0, \ldots, M_n)$ and $C(M_0, \ldots, M_n)$ mean M_0. Clearly, $C'(C'(M_0, M_1, \ldots, M_{n-1}), M_n) = C'(M_0, C'(M_1, \ldots, M_n))$, $C(C(M_0, M_1, \ldots, M_{n-1}), M_n) = C(M_0, C(M_1, \ldots, M_n))$; that is, the associative law holds.

1.2.4 Finite Automata as Recognizers

We have dealt with a finite automaton as a transducer which transforms an input sequence to an output sequence of the same length.

We also defined a special kind of finite automata, i.e., autonomous finite automata. Behavior of an autonomous finite automaton is producing periodic sequences.

In history, a finite automaton is considered as an event (sequence) recognizer in the early 1950s by S. C. Kleene who introduced the concept of regular events and proved the equivalence between regular events and acceptable events by finite automata, an important result in formal language theory. We review some basic concepts.

For a finite automaton $\langle X, Y, S, \delta, \lambda\rangle$, if $|Y| = 1$, it is called a *finite automaton without output*, abbreviated to $\langle X, S, \delta\rangle$.

If $\langle X, S, \delta\rangle$ is a finite automaton without output, $s_0 \in S$ and $F \subseteq S$, the quintuple $\langle X, S, \delta, s_0, F\rangle$ is called a *finite automaton recognizer*, or *finite recognizer* for short. s_0 is called its *initial state*, and F its *final state set*.

Let $M = \langle X, S, \delta, s_0, F\rangle$ be a finite automaton recognizer. The set $R(M) = \{\alpha \mid \alpha \in X^*, \delta(s_0, \alpha) \in F\}$ is called the *recognizing set* of M.

For any finite automaton recognizer $M = \langle X, S, \delta, s_0, F\rangle$ and any set $A \subseteq X^*$, if $R(M) = A$, we say that M *recognizes* A.

1.3 Linear Finite Automata

We review some properties of linear finite automata and the definition of the z-transformation for linear finite automata.

Theorem 1.3.1. *Let M be a linear finite automaton over $GF(q)$ with structure matrices A, B, C, D. For any s_0 in S and any $x_0, x_1, \ldots$ in X, let $s_{i+1} = \delta(s_i, x_i)$, $y_i = \lambda(s_i, x_i)$, $i = 0, 1, \ldots$ Then*

$$s_i = A^i s_0 + \sum_{j=0}^{i-1} A^{i-j-1} B x_j,$$

$$y_i = C A^i s_0 + \sum_{j=0}^{i} H_{i-j} x_j,$$

$$i = 0, 1, \ldots,$$

where $H_0 = D$, $H_j = CA^{j-1}B$, $j > 0$.

Proof. We prove by induction on i that $s_i = A^i s_0 + \sum_{j=0}^{i-1} A^{i-j-1} B x_j$ holds for any $i \geqslant 0$. *Basis* : $i = 0$. It is trivial that $s_0 = A^0 s_0 + \sum_{j=0}^{0-1} A^{0-j-1} B x_j$

holds. *Induction step* : Suppose that $s_i = A^i s_0 + \sum_{j=0}^{i-1} A^{i-j-1} B x_j$ holds. Then

$$s_{i+1} = As_i + Bx_i = A\big(A^i s_0 + \sum_{j=0}^{i-1} A^{i-j-1} B x_j\big) + Bx_i$$

$$= A^{i+1} s_0 + \sum_{j=0}^{i} A^{i-j} B x_j.$$

That is, $s_{i+1} = A^{i+1} s_0 + \sum_{j=0}^{i} A^{i-j} B x_j$ holds. Therefore, $s_i = A^i s_0 + \sum_{j=0}^{i-1} A^{i-j-1} B x_j$ holds for any $i \geqslant 0$.

Using the proven result, for any $i \geqslant 0$, we have

$$y_i = Cs_i + Dx_i = C\big(A^i s_0 + \sum_{j=0}^{i-1} A^{i-j-1} B x_j\big) + Dx_i$$

$$= CA^i s_0 + \sum_{j=0}^{i-1} CA^{i-j-1} B x_j + Dx_i = CA^i s_0 + \sum_{j=0}^{i} H_{i-j} x_j.$$

That is, $y_i = CA^i s_0 + \sum_{j=0}^{i} H_{i-j} x_j$ holds for any $i \geqslant 0$. □

Theorem 1.3.2. *Let $M = \langle X, Y, S, \delta, \lambda\rangle$ be a linear finite automaton over $GF(q)$ with structure matrices A, B, C, D. For any $s_i \in S$, any $\alpha_i \in X^\omega$, any $c_i \in GF(q)$, $i = 1, 2$, we have*

$$\lambda(c_1 s_1 + c_2 s_2, c_1\alpha_1 + c_2\alpha_2) = c_1\lambda(s_1, \alpha_1) + c_2\lambda(s_2, \alpha_2).$$

Proof. For any infinite sequence (ω-word) β, denote $\beta = (\beta)_0(\beta)_1 \ldots$, where $|(\beta)_i| = 1$, $i = 0, 1, \ldots$ From Theorem 1.3.1, we have

$$(\lambda(c_1 s_1 + c_2 s_2, c_1\alpha_1 + c_2\alpha_2))_i$$

$$= CA^i(c_1 s_1 + c_2 s_2) + \sum_{j=0}^{i} H_{i-j}(c_1\alpha_1 + c_2\alpha_2)_j$$

$$= CA^i(c_1 s_1 + c_2 s_2) + \sum_{j=0}^{i} H_{i-j}((c_1\alpha_1)_j + (c_2\alpha_2)_j)$$

$$= c_1 CA^i s_1 + c_2 CA^i s_2 + c_1 \sum_{j=0}^{i} H_{i-j}(\alpha_1)_j + c_2 \sum_{j=0}^{i} H_{i-j}(\alpha_2)_j$$

$$= c_1(\lambda(s_1, \alpha_1))_i + c_2(\lambda(s_2, \alpha_2))_i,$$

$$i = 0, 1, \ldots$$

Thus $\lambda(c_1 s_1 + c_2 s_2, c_1\alpha_1 + c_2\alpha_2) = c_1\lambda(s_1, \alpha_1) + c_2\lambda(s_2, \alpha_2)$. □

From the proof of Theorem 1.3.2, we have the following corollaries.

Corollary 1.3.1. *Let $M = \langle X, Y, S, \delta, \lambda\rangle$ be a linear finite automaton over $GF(q)$. For any positive integer k, any $s_i \in S$, any $\alpha_i \in X^*$ with $|\alpha_i| = k$, any $c_i \in GF(q)$, $i = 1, 2$, we have*

$$\lambda(c_1 s_1 + c_2 s_2, c_1\alpha_1 + c_2\alpha_2) = c_1\lambda(s_1, \alpha_1) + c_2\lambda(s_2, \alpha_2).$$

Corollary 1.3.2. *Let $M = \langle X, Y, S, \delta, \lambda\rangle$ be a linear finite automaton over $GF(q)$. For any $s \in S$ and any $\alpha \in X^\omega$, we have*

$$\lambda(s, \alpha) = \lambda(s, 0^\omega) + \lambda(0, \alpha),$$

where 0 stands for the zero vector in S, and 0^ω stands for an infinite sequence consisting of zero vectors in X.

$\lambda(s, 0^\omega)$ is called the *free response* of the state s, and $\lambda(0, \alpha)$ the *force response* of the input α. Corollary 1.3.2 means that any output sequence of any linear finite automaton can be decomposed into the free response of the initial state and the force response of the input sequence.

Clearly, all the free responses of a linear finite automaton over $GF(q)$ is a vector space over $GF(q)$.

Theorem 1.3.3. *Let $M_i = \langle X_i, Y_i, S_i, \delta_i, \lambda_i\rangle$ be a linear finite automaton over $GF(q)$, $i = 1, 2$, and $X_1 = X_2$. For any $s_i \in S_i$, $i = 1, 2$, $s_1 \sim s_2$ if and only if the zero state of M_1 and the zero state of M_2 are equivalent and the free responses of s_1 and s_2 are the same.*

Proof. Suppose that $s_1 \sim s_2$. For any $\alpha \in X_1^\omega$, we then have $\lambda_1(s_1, \alpha) = \lambda_2(s_2, \alpha)$. Especially, taking $\alpha = 0^\omega$, we obtain $\lambda_1(s_1, 0^\omega) = \lambda_2(s_2, 0^\omega)$, that is, the free responses of s_1 and s_2 are the same. Since $\lambda_i(s_i, \alpha) = \lambda_i(s_i, 0^\omega) + \lambda_i(0, \alpha)$, $i = 1, 2$, $\lambda_1(s_1, \alpha) = \lambda_2(s_2, \alpha)$ yields $\lambda_1(0, \alpha) = \lambda_2(0, \alpha)$. It follows that the zero state of M_1 and the zero state of M_2 are equivalent.

Conversely, suppose that the zero state of M_1 and the zero state of M_2 are equivalent and the free responses of s_1 and s_2 are the same. Then for any $\alpha \in X_1^\omega$, $\lambda_1(0, \alpha) = \lambda_2(0, \alpha)$, and $\lambda_1(s_1, 0^\omega) = \lambda_2(s_2, 0^\omega)$. Thus for any $\alpha \in X_1^\omega$, $\lambda_1(s_1, \alpha) = \lambda_1(s_1, 0^\omega) + \lambda_1(0, \alpha) = \lambda_2(s_2, 0^\omega) + \lambda_2(0, \alpha) = \lambda_2(s_2, \alpha)$. Therefore, $s_1 \sim s_2$. □

Corollary 1.3.3. *Let M be a linear finite automaton over $GF(q)$. For any states s_1 and s_2 of M, $s_1 \sim s_2$ if and only if the free responses of s_1 and s_2 are the same.*

Corollary 1.3.4. *Let M_1 and M_2 be two linear finite automata over $GF(q)$. Then $M_1 \sim M_2$ if and only if the zero state of M_1 and the zero state of M_2 are equivalent and the free response spaces of M_1 and M_2 are the same.*

Corollary 1.3.5. *Let M and M' be two linear finite automata over $GF(q)$. If the state s_i of M and the state s'_i of M' are equivalent, $i = 1, \ldots, k$, then for any c_i in $GF(q)$, $i = 1, \ldots, k$, the state $c_1 s_1 + \cdots + c_k s_k$ of M and the state $c_1 s'_1 + \cdots + c_k s'_k$ of M' are equivalent.*

Let M be a linear finite automaton over $GF(q)$ with structure matrices A, B, C, D and structure parameters l, m, n. The matrix

$$K_n = \begin{bmatrix} C \\ CA \\ \vdots \\ CA^{n-1} \end{bmatrix}$$

is called the *diagnostic matrix* of M.

For any states s_1 and s_2 of M, since $s_1 \sim s_2$ if and only if their free responses are the same, $s_1 \sim s_2$ if and only if $CA^i s_1 = CA^i s_2$, $i = 0, 1, \ldots$ Since the degree of the minimal polynomial of A is at most n, $s_1 \sim s_2$ if and only if $CA^i s_1 = CA^i s_2$, $i = 0, 1, \ldots, n-1$. Thus $s_1 \sim s_2$ if and only if $K_n s_1 = K_n s_2$.

Let T be a matrix consisting of some maximal independent rows of K_n. Then $s_1 \sim s_2$ if and only if $Ts_1 = Ts_2$.

Theorem 1.3.4. *Let M be a linear finite automaton over $GF(q)$ with structure matrices A, B, C, D. Assume that a matrix T over $GF(q)$ satisfies conditions: rows of T are linear independent and for any states s_1 and s_2 of M $s_1 \sim s_2$ if and only if $Ts_1 = Ts_2$. Let M' be a linear finite automaton over $GF(q)$ with structure matrices A', B', C', D', where $A' = TAR$, $B' = TB$, $C' = CR$, $D' = D$, R is a right inverse matrix of T. Then M' is minimal and equivalent to M.*

Proof. Clearly, for any state s of M, $Ts = T(RTs)$; therefore, $s \sim RTs$.

We prove $TART = TA$. For any state s of M, the input 0 carries states s and RTs to states As and $ARTs$, respectively. From $RTs \sim s$, we have $ARTs \sim As$. It follows that $TARTs = TAs$. From arbitrariness of s, we have $TART = TA$.

We prove $CRT = C$. For any state s of M, since $s \sim RTs$, the output of the input 0 on the state s and the output of the input 0 on the state RTs are the same, namely, $Cs = CRTs$. From arbitrariness of s, we have $C = CRT$.

For any state s of M, Ts is a state of M'. We prove $s \sim Ts$. For any input sequence $x_0 x_1 \ldots$, let the output sequences on s and on Ts be $y_0 y_1 \ldots$ and $y'_0 y'_1 \ldots$, respectively. From Theorem 1.3.1, we have

$$y_i = CA^i s + Dx_0 + \sum_{j=1}^{i} CA^{i-j-1} B x_j,$$

$$y'_i = C'A'^i Ts + D'x_0 + \sum_{j=1}^{i} C'A'^{i-j-1}B'x_j,$$
$$i = 0, 1, \ldots$$

Using $TART = TA$ and $CRT = C$, this yields

$$\begin{aligned} y'_i &= (CR)(TAR)^i Ts + Dx_0 + \sum_{j=1}^{i}(CR)(TAR)^{i-j-1}(TB)x_j \\ &= (CR)TA^i s + Dx_0 + \sum_{j=1}^{i}(CR)TA^{i-j-1}Bx_j \\ &= CA^i s + Dx_0 + \sum_{j=1}^{i} CA^{i-j-1}Bx_j = y_i, \\ &i = 0, 1, \ldots \end{aligned}$$

Thus $s \sim Ts$.

Since for any state s' of M' there exists a state s of M such that $s' = Ts$, we have $s \sim s'$. On the other hand, for any state s of M, the state Ts of M' is equivalent to s. Thus M' and M are equivalent.

We prove that M' is minimal. Suppose that s'_1 and s'_2 are two equivalent states of M'. Let $s_i = Rs'_i$, $i = 1, 2$. Then s_i and Ts_i $(=s'_i)$ are equivalent, $i = 1, 2$. From $s'_1 \sim s'_2$, we have $s_1 \sim s_2$. It follows that $Ts_1 = Ts_2$. That is, $s'_1 = s'_2$. Thus M' is minimal. □

Let $M_i = \langle X_i, Y_i, S_i, \delta_i, \lambda_i \rangle$, $i = 1, 2$ be two linear finite automata over $GF(q)$. M_1 and M_2 are said to be *similar*, if there exists a linear isomorphism from M_1 to M_2. From the definition, it is easy to show that the similar relation is reflexive, symmetric and transitive.

If M_1 and M_2 are minimal and equivalent, and $Y_1 = Y_2$, then M_1 and M_2 are similar. In fact, it is proven in Subsect. 1.2.1 that the relation $\sim$ is an isomorphism from M_1 to M_2. Let φ be the relation $\sim$ from S_1 to S_2. We prove that φ is linear. Let $c_i \in GF(q)$, $s_i \in S_1$, $i = 1, 2$. Since $s_i \sim \varphi(s_i)$, $i = 1, 2$, from Corollary 1.3.5, we have $c_1 s_1 + c_2 s_2 \sim c_1\varphi(s_1) + c_2\varphi(s_2)$. Thus $\varphi(c_1 s_1 + c_2 s_2) = c_1\varphi(s_1) + c_2\varphi(s_2)$. We conclude that M_1 and M_2 are similar.

Let $f(z) = z^k + a_{k-1}z^{k-1} + \cdots + a_1 z + a_0$ be a polynomial over $GF(q)$. We use $P_{f(z)}$ to denote the matrix

$$P_{f(z)} = \begin{bmatrix} 0 & 1 & \cdots & 0 & 0 \\ 0 & 0 & \cdots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \cdots & 0 & 1 \\ -a_0 & -a_1 & \cdots & -a_{k-2} & -a_{k-1} \end{bmatrix}. \tag{1.6}$$

A linear finite automaton is called a *linear shift register*, if its state transition matrix is $P_{f(z)}$ for some $f(z)$.

Let $M_i = \langle X, Y, S_i, \delta_i, \lambda_i \rangle$ be a linear finite automaton over $GF(q)$ with structure matrices A_i, B_i, C_i, D_i and structure parameters l, m, n_i, $i = 1, \ldots, h$. The linear finite automaton with structure matrices A, B, C, D is called the *union* of M_1, ..., M_h, where

$$A = \begin{bmatrix} A_1 & & & \\ & A_2 & & \\ & & \ddots & \\ & & & A_h \end{bmatrix}, \quad B = \begin{bmatrix} B_1 \\ B_2 \\ \vdots \\ B_h \end{bmatrix},$$
$$C = [C_1\ C_2\ \ldots\ C_h], \quad D = D_1 + D_2 + \cdots + D_h.$$

In the definition, some M_i may be autonomous, where B_i and D_i are zero matrices.

Theorem 1.3.5. *Let M be a linear finite automaton over $GF(q)$. Then there exist linear shift registers M_1, ..., M_h over $GF(q)$ such that M is similar to the union of M_1, ..., M_h.*

Proof. Let A, B, C, D be structure matrices of M. It is known that there exists a nonsingular matrix P over $GF(q)$ such that

$$PAP^{-1} = \begin{bmatrix} P_{f_1(z)} & & & \\ & P_{f_2(z)} & & \\ & & \ddots & \\ & & & P_{f_h(z)} \end{bmatrix},$$

where $f_1(z)$, ..., $f_h(z)$ are the elementary divisors of A. Let M' be a linear finite automaton with structure matrices A', B', C', D, where $A' = PAP^{-1}$, $B' = PB$, $C' = CP^{-1}$. Clearly, M and M' are similar. Let M_i be a linear shift register with structure matrices $P_{f_i(z)}$, B_i, C_i, D_i, $i = 1, \ldots, h$, where

$$PB = \begin{bmatrix} B_1 \\ B_2 \\ \vdots \\ B_h \end{bmatrix}, \quad CP^{-1} = [C_1\ C_2\ \ldots\ C_h], \quad D = D_1 + D_2 + \cdots + D_h.$$

Clearly, M' is the union of M_1, ..., M_h. □

Let $M = \langle X, Y, S, \delta, \lambda \rangle$ be a linear finite automata over $GF(q)$ with structure matrices A, B, C, D and structure parameters l, m, n. Let $S_0 = \{\delta(0, \alpha), \alpha \in X^*\}$. Denote the restrictions of δ and λ on $S_0 \times X$ by δ_0 and λ_0, respectively. It is easy to verify that δ_0 and λ_0 are single-valued mappings

from $S_0 \times X$ to S_0 and Y, respectively. Thus $M_0 = \langle X, Y, S_0, \delta_0, \lambda_0 \rangle$ is a finite subautomaton of M. It is easy to see that S_0 is a subspace of S. Denote the dimension of S_0 by n_0. Let R be an $n \times n_0$ matrix so that columns of R form a basis of S_0. Let T be a left inverse matrix of R. Define $A_0 = TAR$, $B_0 = TB$, $C_0 = CR$, and $D_0 = D$. Let $M_0' = \langle X, Y, S_0', \delta_0', \lambda_0' \rangle$ be a linear finite automaton over $GF(q)$ with structure matrices A_0, B_0, C_0, D_0, where S_0' is the vector space of dimension n_0 over $GF(q)$. Then M_0' and M_0 are isomorphic. In fact, for any s_1 and s_2 in S_0, there exist s_1' and s_2' in S_0' such that $s_i = Rs_i'$, $i = 1, 2$. Thus $Ts_1 = Ts_2$ if and only if $s_1' = s_2'$. It follows that $s_1 = s_2$ if and only if $Ts_1 = Ts_2$. Let φ be a single-valued mapping from S_0' to S_0 defined by $\varphi(s') = Rs'$ for any s' in S_0'. Clearly, φ is bijective. For any s' in S_0' and for any x in X, since Rs' is in S_0, from the definitions of S_0 and δ_0, $\delta_0(\varphi(s'), x)$ is in S_0. Clearly, $\varphi(\delta_0'(s', x))$ is in S_0. Since

$$T\delta_0(\varphi(s'), x) = TARs' + TBx = A_0 s' + B_0 x$$

$$= T(R(A_0 s' + B_0 x)) = T\varphi(\delta_0'(s', x)),$$

we have $\delta_0(\varphi(s'), x) = \varphi(\delta_0'(s', x))$. We also have

$$\lambda_0(\varphi(s'), x) = CRs' + Dx = C_0 s' + D_0 x = \lambda_0'(s', x).$$

Thus φ is an isomorphism from M_0' to M_0. Therefore, M_0' and M_0 are isomorphic. M_0' is referred to as a *minimal linear finite subautomaton* of M. Since each of minimal linear finite subautomata of M is isomorphic to M_0, they are isomorphic. Since M_0' and M_0 are isomorphic, we have $S_0' = \{\delta'(0, \alpha), \alpha \in X^*\}$. Let M_a be the linear autonomous finite automaton with structure matrices A, C. M_a is referred to as the *maximal linear autonomous finite subautomaton* of M. From Corollary 1.3.2, for any state $\langle s_a, s_0' \rangle$ of the union of M_a and M_0', the state $s_a + Rs_0'$ of M and $\langle s_a, s_0' \rangle$ are equivalent. Conversely, for any state s of M, the state $\langle s, 0 \rangle$ of the union of M_a and M_0' and s are equivalent. Thus M and the union of M_a and M_0' are equivalent.

Take a formal symbol z. Let

$$\mathcal{F} = \Big\{ \sum_{\tau=-\infty}^{\infty} a_\tau z^\tau \mid a_\tau \in GF(q),\ \tau = 0, \pm 1, \pm 2, \ldots, \text{ and the number of nonzero } a_\tau \text{ for negative subscript } \tau \text{ is finite} \Big\}.$$

Any $a(z) = \sum_{\tau=-\infty}^{\infty} a_\tau z^\tau$ in $\mathcal{F}$, $\max\ n(a_\tau = 0 \text{ if } \tau < n)$ is called the *low degree* of $a(z)$. We also denote $a(z)$ by $\sum_{\tau=k}^{\infty} a_\tau z^\tau$ for any integer $k \leqslant$ the low degree of $a(z)$. In the case of $a_\tau = 0$ for any integer τ, we use 0 to denote $a(z)$. Notice that the low degree of 0 is $-\infty$. For any two elements

$a(z) = \sum_{\tau=-\infty}^{\infty} a_\tau z^\tau$ and $b(z) = \sum_{\tau=-\infty}^{\infty} b_\tau z^\tau$ in $\mathcal{F}$, we define the sum of $a(z)$ and $b(z)$ as an element $\sum_{\tau=-\infty}^{\infty}(a_\tau + b_\tau)z^\tau$ in $\mathcal{F}$, denoted by $a(z)+b(z)$. We define the product of $a(z)$ and $b(z)$ as an element $\sum_{\tau=-\infty}^{\infty} c_\tau z^\tau$ of $\mathcal{F}$, denoted by $a(z)b(z)$, where

$$c_\tau = \sum_{i+j=\tau, i\geqslant k, j\geqslant h} a_i b_j,$$

k and h are lower degrees of $a(z)$ and $b(z)$, respectively. It is easy to verify that $\mathcal{F}$ is a field and $GF(q)$ is its subfield in the isomorphic sense (a_0 in $GF(q)$ corresponds to $a_0 z^0$ in $\mathcal{F}$).

Let $\Omega = [\omega_0, \omega_1, \ldots]$ be an infinite sequence over $GF(q)$. The element $\sum_{\tau=0}^{\infty} \omega_\tau z^\tau$ in $\mathcal{F}$ is called the *generating function* or *z-transformation* of Ω. For any Ω, we use $\Omega(z)$ to denote its z-transformation.

Let $\Phi = [\varphi_0, \varphi_1, \ldots]$ be an infinite sequence over the column vector space of dimension k over $GF(q)$. Let $\varphi_\tau = [\varphi_{1\tau}, \ldots, \varphi_{k\tau}]^{\mathrm{T}}$, for any nonnegative integer τ. We use Φ_i to denote $[\varphi_{i0}, \varphi_{i1}, \ldots]$, that is, the i-th component sequence of Φ. $[\Phi_1(z), \ldots, \Phi_k(z)]^{\mathrm{T}}$ is called the *generating function* or *z-transformation* of Φ, denoted by $\Phi(z)$. It is easy to verify that the z-transformation of the linear combination of sequences, say $a_1\Phi_1 + \cdots + a_r\Phi_r$, is the linear combination of z-transformations, namely, $a_1\Phi_1(z) + \cdots + a_r\Phi_r(z)$,where $a_1, \ldots, a_r \in GF(q)$, $\Phi_1, \ldots, \Phi_r$ are r infinite sequences over $GF(q)$ or r infinite sequences over a column vector space over $GF(q)$.

Let $M = \langle X, Y, S, \delta, \lambda \rangle$ be a linear finite automaton over $GF(q)$ with structure matrices A, B, C, D and structure parameters l, m, n. For any s_0 in S and any $x_0, x_1, \ldots$ in X, let $s_{i+1} = \delta(s_i, x_i)$, $y_i = \lambda(s_i, x_i)$, $i = 0, 1, \ldots$ Then

$$\begin{aligned} s_{\tau+1} &= As_\tau + Bx_\tau, \\ y_\tau &= Cs_\tau + Dx_\tau, \\ \tau &= 0, 1, \ldots \end{aligned} \tag{1.7}$$

We use $X(z)$, $Y(z)$, and $S(z)$ to denote z-transformations of the input sequence $[x_0, x_1, \ldots]$, the output sequence $[y_0, y_1, \ldots]$, and the state sequence $[s_0, s_1, \ldots]$, respectively. We use $X_i(z)$, $Y_i(z)$, $S_i(z)$, $x_{i\tau}$, $y_{i\tau}$, and $s_{i\tau}$ to denote the i-th components of $X(z)$, $Y(z)$, $S(z)$, x_τ, y_τ, and s_τ, respectively. From (1.7), we have

$$S(z) = \begin{bmatrix} S_1(z) \\ \vdots \\ S_n(z) \end{bmatrix} = \begin{bmatrix} \sum_{\tau=0}^{\infty} s_{1\tau} z^{\tau} \\ \vdots \\ \sum_{\tau=0}^{\infty} s_{n\tau} z^{\tau} \end{bmatrix}$$

$$= \begin{bmatrix} s_{10} + \sum_{\tau=1}^{\infty} \big(\sum_{j=1}^{n} a_{1j} s_{j,\tau-1} + \sum_{j=1}^{l} b_{1j} x_{j,\tau-1} \big) z^{\tau} \\ \vdots \\ s_{n0} + \sum_{\tau=1}^{\infty} \big(\sum_{j=1}^{n} a_{nj} s_{j,\tau-1} + \sum_{j=1}^{l} b_{nj} x_{j,\tau-1} \big) z^{\tau} \end{bmatrix}$$

$$= \begin{bmatrix} s_{10} + z \sum_{j=1}^{n} a_{1j} \sum_{\tau=1}^{\infty} s_{j,\tau-1} z^{\tau-1} + z \sum_{j=1}^{l} b_{1j} \sum_{\tau=1}^{\infty} x_{j,\tau-1} z^{\tau-1} \\ \vdots \\ s_{n0} + z \sum_{j=1}^{n} a_{nj} \sum_{\tau=1}^{\infty} s_{j,\tau-1} z^{\tau-1} + z \sum_{j=1}^{l} b_{nj} \sum_{\tau=1}^{\infty} x_{j,\tau-1} z^{\tau-1} \end{bmatrix}$$

$$= \begin{bmatrix} s_{10} + z \sum_{j=1}^{n} a_{1j} S_j(z) + z \sum_{j=1}^{l} b_{1j} X_j(z) \\ \vdots \\ s_{n0} + z \sum_{j=1}^{n} a_{nj} S_j(z) + z \sum_{j=1}^{l} b_{nj} X_j(z) \end{bmatrix}$$

$$= s_0 + zAS(z) + zBX(z),$$

where a_{ij} and b_{ij} are elements at row i and column j of A and B, respectively. It follows that

$$(E - zA)S(z) = s_0 + zBX(z),$$

where E stands for the $n \times n$ identity matrix over $GF(q)$. Since the constant term of the determinant $|E - zA|$ is 1, we have

$$(E - zA)^{-1} = (E - zA)^{*} / |E - zA|,$$

where $(E - zA)^{*}$ is the adjoint matrix of $E - zA$, i.e., the matrix of which the element at row i and column j is the cofactor at row j and column i of $E - zA$. Thus

$$S(z) = (E - zA)^{-1}s_0 + z(E - zA)^{-1}BX(z)$$
$$= \frac{(E - zA)^*}{|E - zA|}s_0 + \frac{z(E - zA)^*B}{|E - zA|}X(z). \tag{1.8}$$

Since

$$y_{i\tau} = \sum_{j=1}^{n} c_{ij}s_{j\tau} + \sum_{j=1}^{l} d_{ij}x_{j\tau}, \quad i = 1, \ldots, m, \ \tau = 0, 1, \ldots,$$

we have

$$Y_i(z) = \sum_{j=1}^{n} c_{ij}S_j(z) + \sum_{j=1}^{l} d_{ij}X_j(z), \quad i = 1, \ldots, m,$$

where c_{ij} and d_{ij} are elements at row i and column j of C and D, respectively. It follows that $Y(z) = CS(z) + DX(z)$. From (1.8), we have

$$Y(z) = C(E - zA)^{-1}s_0 + (zC(E - zA)^{-1}B + D)X(z)$$
$$= \frac{C(E - zA)^*}{|E - zA|}s_0 + \left(\frac{zC(E - zA)^*B}{|E - zA|} + D\right)X(z). \tag{1.9}$$

Let

$$G(z) = C(E - zA)^{-1} = C(E - zA)^*/|E - zA|, \tag{1.10}$$
$$H(z) = zC(E - zA)^{-1}B + D = zC(E - zA)^*B/|E - zA| + D.$$

Then (1.9) is

$$Y(z) = G(z)s_0 + H(z)X(z). \tag{1.11}$$

In (1.11), $G(z)s_0$ is the z-transformation of the free response of the initial state s_0, and $H(z)X(z)$ is the z-transformation of the force response of the input sequence $x_0x_1\ldots$ $G(z)$ is called the *free response matrix* of M, and $H(z)$ is called the *transfer function matrix* of M. Clearly, if matrices $G(z)$ and $H(z)$ over the field $\mathcal{F}$ satisfy the condition that (1.11) holds for any s_0 and any $X(z)$, then $G(z)$ and $H(z)$ are the free response matrix and the transfer function matrix of M, respectively. In other words, the free response matrix and the transfer function matrix of M are uniquely determined by (1.11). Notice that from (1.10), each element of $G(z)$ and $H(z)$ may be expressed as a rational fraction of z with a nonzero constant term of the denominator.

In the case of

$$A = \begin{bmatrix} 0 & 1 & \cdots & 0 & 0 \\ 0 & 0 & \cdots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \cdots & 0 & 1 \\ -a_0 & -a_1 & \cdots & -a_{n-2} & -a_{n-1} \end{bmatrix}, \tag{1.12}$$

we have

$$|E - zA| = 1 + a_{n-1}z + \cdots + a_1 z^{n-1} + a_0 z^n, \tag{1.13}$$

$$(E - zA)^* = \begin{bmatrix} z^0\varphi_{n-1}(z) & z^1\varphi_{n-2}(z) & \cdots & z^{n-2}\varphi_1(z) & z^{n-1} \\ z^{-1}\psi_{n-1}(z) & z^0\varphi_{n-2}(z) & \cdots & z^{n-3}\varphi_1(z) & z^{n-2} \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ z^{-(n-2)}\psi_{n-1}(z) & z^{-(n-3)}\psi_{n-2}(z) & \cdots & z^0\varphi_1(z) & z^1 \\ z^{-(n-1)}\psi_{n-1}(z) & z^{-(n-2)}\psi_{n-2}(z) & \cdots & z^{-1}\psi_1(z) & z^0 \end{bmatrix},$$

where

$$\varphi_k(z) = 1 + \sum_{i=1}^{k} a_{n-i}z^i, \quad \psi_k(z) = \sum_{i=k+1}^{n} -a_{n-i}z^i, \tag{1.14}$$
$$k = 1, \ldots, n-1.$$

1.4 Concepts on Invertibility

A finite automaton $M = \langle X, Y, S, \delta, \lambda \rangle$ is said to be *invertible*, if for any s, s' in S, and any α, α' in X^ω, $\lambda(s, \alpha) = \lambda(s', \alpha')$ yields $\alpha = \alpha'$. In other words, M is invertible, if and only if for any s in S and any α in X^ω, α can be uniquely determined by $\lambda(s, \alpha)$.

Evidently, if M is invertible, then $M'' = \langle X, Y, S'', \delta'', \lambda'' \rangle$ is invertible in the case where $M'' \prec M$ or $M'' \leqslant M$.

M is invertible, if and only if for any s, s' in S, any x, x' in X, and any α, α' in X^ω, $\lambda(s, x\alpha) = \lambda(s', x'\alpha')$ implies $x = x'$, that is, for any s in S, any x in X, and any α in X^ω, x can be uniquely determined by $\lambda(s, x\alpha)$. In fact, the *only if* part is trivial. To prove the *if* part, suppose that $\lambda(s, \alpha) = \lambda(s', \alpha')$ for s, s' in S and α, α' in X^ω. We prove $\alpha = \alpha'$. Denote $\alpha = x_0 x_1 \ldots$, $\alpha' = x'_0 x'_1 \ldots$, for some x_i, x'_i in X, $i = 0, 1, \ldots$ To prove $x_i = x'_i$ for $i \geqslant 0$, let $s_i = \delta(s, x_0 \ldots x_{i-1})$ and $s'_i = \delta(s', x'_0 \ldots x'_{i-1})$. From $\lambda(s, x_0 x_1 \ldots) = \lambda(s, x'_0 x'_1 \ldots)$, we have $\lambda(s_i, x_i x_{i+1} \ldots) = \lambda(s_i, x'_i x'_{i+1} \ldots)$. Since for any t, t' in S, any x, x' in X, and any β, β' in X^ω, $\lambda(t, x\beta) = \lambda(t', x'\beta)$ yields $x = x'$, we have $x_i = x'_i$. Thus $\alpha = \alpha'$.

A finite automaton $M = \langle X, Y, S, \delta, \lambda \rangle$ is said to be *invertible with delay* τ, τ being a nonnegative integer, if for any s in S and any x_i in X, $i = 0, 1, \ldots, \tau$, x_0 can be uniquely determined by $\lambda(s, x_0 \ldots x_\tau)$, that is, for any s, s' in S and any x_i, x'_i in X, $i = 0, 1, \ldots, \tau$, $\lambda(s, x_0 \ldots x_\tau) = \lambda(s', x'_0 \ldots x'_\tau)$ yields $x_0 = x'_0$.

Evidently, if M is invertible with delay τ, then $M'' = \langle X, Y, S'', \delta'', \lambda'' \rangle$ is invertible with delay τ in the case where $M'' \prec M$ or $M'' \leqslant M$.

Clearly, if M is invertible with delay τ, then M is invertible. Below we prove that its converse proposition also holds.

Let $M = \langle X, Y, S, \delta, \lambda\rangle$ be a finite automaton. Construct a graph $G_M = (V, \Gamma)$ as follows. Let

$$R = \{(\delta(s,x), \delta(s',x')) \mid x \neq x', \lambda(s,x) = \lambda(s',x'), x, x' \in X, s, s' \in S\}.$$

In the case of $R \neq \varnothing$, let the vertex set V of G_M be the minimal subset of $S \times S$ satisfying the following conditions: (a) $R \subseteq V$, (b) if $(s,s') \in V$ and $\lambda(s,x) = \lambda(s',x')$ for $x, x' \in X$, then $(\delta(s,x), \delta(s',x')) \in V$. Let the arc set Γ of G_M be the set of all arcs $((s,s'), (\delta(s,x), \delta(s',x')))$ satisfying the following conditions: $(s,s') \in V$, $x, x' \in X$, and $\lambda(s,x) = \lambda(s',x')$. In the case of $R = \varnothing$, let G_M be the empty graph.

Theorem 1.4.1. *M is invertible if and only if G_M has no circuit. Moreover, if G_M has no circuit and the level of G_M is $\rho - 1$, then M is invertible with delay $\rho + 1$ and not invertible with delay τ for any $\tau \leqslant \rho$.*

Proof. Suppose that G_M has a circuit, say w. From the construction of G_M, for any vertex on w, there exists a path of which the terminal vertex is the vertex on w and the initial vertex is in R. Thus there exists a path $u_1u_2\ldots u_k$ such that the initial vertex of u_1 is in R and $u_ru_{r+1}\ldots u_k$ is a circuit for some r, $1 \leqslant r \leqslant k$. Let (s_i, s_i') be the initial vertex of u_i, $i = 1, 2, \ldots, k$. Then the terminal vertex of u_k is (s_r, s_r'). From the construction of G_M, there exist x_i, $x_i' \in X$, $i = 1, 2, \ldots, k$, such that $\delta(s_i, x_i) = s_{i+1}$, $\delta(s_i', x_i') = s_{i+1}'$, $i = 1, 2, \ldots, k-1$, $\delta(s_k, x_k) = s_r$, $\delta(s_k', x_k') = s_r'$, and $\lambda(s_i, x_i) = \lambda(s_i', x_i')$, $i = 1, 2, \ldots, k$. From $(s_1, s_1') \in R$, there exist s_0, $s_0' \in S$, x_0, $x_0' \in X$, such that $\delta(s_0, x_0) = s_1$, $\delta(s_0', x_0') = s_1'$, $\lambda(s_0, x_0) = \lambda(s_0', x_0')$ and $x_0 \neq x_0'$. Taking

$$\alpha = x_0x_1\ldots x_{r-1}x_r\ldots x_kx_r\ldots x_k\ldots,$$
$$\alpha' = x_0'x_1'\ldots x_{r-1}'x_r'\ldots x_k'x_r'\ldots x_k'\ldots,$$

we then have $\lambda(s_0, \alpha) = \lambda(s_0', \alpha')$. Since $x_0 \neq x_0'$, M is not invertible.

Conversely, suppose that G_M has no circuit. Then the level of G_M is an integer, say $\rho - 1$. In the case of $R = \varnothing$, it is evident that $\rho = -1$ and M is invertible with delay 0 $(= \rho + 1)$. In the case of $R \neq \varnothing$, for any states s_0 and s_0' of M, and any input sequences $\alpha = x_0x_1\ldots x_{\rho+1}$ and $\alpha' = x_0'x_1'\ldots x_{\rho+1}'$ of length $\rho + 2$, x_i, $x_i' \in X$, $i = 0, 1, \ldots, \rho + 1$, we prove by reduction to absurdity that $\lambda(s, \alpha) = \lambda(s, \alpha')$ implies $x_0 = x_0'$. Suppose to the contrary that $\lambda(s_0, x_0x_1\ldots x_{\rho+1}) = \lambda(s_0', x_0'x_1'\ldots x_{\rho+1}')$ and $x_0 \neq x_0'$, for some states s_0, s_0' in S and some input letters x_i, x_i', $i = 0, 1, \ldots, \rho + 1$ in X. Denote $s_i = \delta(s_{i-1}, x_{i-1})$, $s_i' = \delta(s_{i-1}', x_{i-1}')$, $i = 1, 2, \ldots, \rho + 2$. Since

$\lambda(s_0, x_0x_1 \dots x_{\rho+1}) = \lambda(s'_0, x'_0x'_1 \dots x'_{\rho+1})$, we have $\lambda(s_i, x_i) = \lambda(s'_i, x'_i)$, $i = 0, 1, \dots, \rho+1$. From $x_0 \neq x'_0$, we have $(s_1, s'_1) \in R$, and for any i, $1 \leqslant i \leqslant \rho+1$, there exists an arc, say u_i, such that the initial vertex of u_i is (s_i, s'_i) and the terminal vertex of u_i is (s_{i+1}, s'_{i+1}). Thus $u_1u_2 \dots u_{\rho+1}$ is a path of G_M. Since the length of the path is $\rho+1$, the level of G_M is at least ρ. This contradicts that the level of G_M is $\rho-1$. We conclude that M is invertible with delay $\rho+1$.

Let $0 \leqslant \tau \leqslant \rho$. Since the level of G_M is $\rho-1$, there exists a path of length τ of G_M, say $u_1u_2 \dots u_\tau$. Denote the initial vertex of u_i by (s_i, s'_i) and the terminal vertex of u_i by (s_{i+1}, s'_{i+1}), $i = 1, 2, \dots, \tau$. From the construction of G_M, without loss of generality, suppose that (s_1, s'_1) is in R. From the definition of R, there exist x_0, x'_0 in X and s_0, s'_0 in S, such that $\delta(s_0, x_0) = s_1$, $\delta(s'_0, x'_0) = s'_1$, $\lambda(s_0, x_0) = \lambda(s'_0, x'_0)$ and $x_0 \neq x'_0$. From the construction of G_M, there exist x_i, x'_i in X, $i = 1, 2, \dots, \tau$, such that $\delta(s_i, x_i) = s_{i+1}$, $\delta(s'_i, x'_i) = s'_{i+1}$, and $\lambda(s_i, x_i) = \lambda(s'_i, x'_i)$, $i = 1, 2, \dots, \tau$. Thus we have $\lambda(s_0, x_0x_1 \dots x_\tau) = \lambda(s'_0, x'_0x'_1 \dots x'_\tau)$. From $x_0 \neq x'_0$, M is not invertible with delay τ. □

Corollary 1.4.1. *If M is invertible, then there exists $\tau \leqslant n(n-1)/2$ such that M is invertible with delay τ, where n is the element number in the state alphabet of M.*

Proof. Suppose that M is invertible. Whenever G_M is the empty graph, M is invertible with delay 0 and $0 \leqslant n(n-1)/2$. Whenever G_M is not the empty graph, then $G_M = \langle V, \Gamma \rangle$ has no circuit. This yields that $s_1 \neq s_2$ for any (s_1, s_2) in V. Thus $|V| \leqslant n(n-1)$. It is evident that $(s_1, s_2) \in R$ if and only if $(s_2, s_1) \in R$. From the construction of G_M, this yields that $(s_1, s_2) \in V$ if and only if $(s_2, s_1) \in V$, and that $((s_1, s_2), (s_3, s_4)) \in \Gamma$ if and only if $((s_2, s_1), (s_4, s_3)) \in \Gamma$. Therefore, the number of vertices with level i is at least 2, for any i, $0 \leqslant i \leqslant \rho$, where $\rho-1$ is the level of G_M. Then we have $2(\rho+1) \leqslant n(n-1)$. Take $\tau = \rho+1$. Then $\tau \leqslant n(n-1)/2$. From Theorem 1.4.1, M is invertible with delay τ. □

Let $M = \langle X, Y, S, \delta, \lambda \rangle$ and $M' = \langle Y, X, S', \delta', \lambda' \rangle$ be two finite automata. For any states s in S and s' in S', if

$$(\forall \alpha)_{X^\omega} (\exists \alpha_0)_{X^*} [\, \lambda'(s', \lambda(s, \alpha)) = \alpha_0\alpha \;\&\; |\alpha_0| = \tau \,],$$

i.e., for any $\alpha \in X^\omega$ there exists $\alpha_0 \in X^*$ such that $\lambda'(s', \lambda(s, \alpha)) = \alpha_0\alpha$ and $|\alpha_0| = \tau$, (s', s) is called a *match pair* with delay τ or say that s' *τ-matches* s. Clearly, if s' τ-matches s and $\beta = \lambda(s, \alpha)$ for some α in X^*, then $\delta'(s', \beta)$ τ-matches $\delta(s, \alpha)$.

M' is called an *inverse* with delay τ of M, if for any s in S and any s' in S', (s', s) is a match pair with delay τ. M is called an *original inverse* with

delay τ of M', if M' is an inverse with delay τ of M. M' is called an *inverse* with delay τ, if M' is an inverse with delay τ of some finite automaton. M' is called an *inverse*, if M' is an inverse with delay τ for some τ.

Clearly, if M' is an inverse with delay τ of M, then M' is an inverse with delay τ of $M'' = \langle X, Y, S'', \delta'', \lambda'' \rangle$ in the case where $M'' \prec M$ or $M'' \leqslant M$. If M' is an inverse with delay τ of M, then $M'' = \langle Y, X, S'', \delta'', \lambda'' \rangle$ is an inverse with delay τ of M in the case where $M'' \prec M'$ or $M'' \leqslant M'$. Therefore, if M' is an inverse with delay τ, then $M'' = \langle Y, X, S'', \delta'', \lambda'' \rangle$ is an inverse with delay τ in the case where $M'' \prec M'$ or $M'' \leqslant M'$.

Theorem 1.4.2. *If M is invertible with delay τ, then there exists a τ-order input-memory finite automaton M' such that M' is an inverse with delay τ of M.*

Proof. Since M is invertible with delay τ, for any $s \in S$ and any x_0, x_1, ..., $x_\tau \in X$, x_0 can be uniquely determined by $\lambda(s, x_0 x_1 \dots x_\tau)$. Thus we can construct a single-valued mapping f from $Y^{\tau+1}$ to X satisfying the condition: if $y_0 y_1 \dots y_\tau = \lambda(s, x_0 x_1 \dots x_\tau)$, $s \in S$ and $x_0, x_1, \dots, x_\tau \in X$, then $f(y_\tau, \dots, y_1, y_0) = x_0$. Let $M'\langle Y, X, S', \delta', \lambda' \rangle$ be the τ-order input-memory finite automaton M_f. From the definition of f and the construction of M_f, it is easy to verify that for any $s \in S$, any $s' \in S'$ and any $x_0, x_1, \dots \in X$,

$$\lambda'(s', \lambda(s, x_0 x_1 \dots)) = x_{-\tau} \dots x_{-1} x_0 x_1 \dots$$

holds for some $x_{-\tau}, \dots, x_{-1} \in X$. Therefore, M' is an inverse with delay τ of M. □

Corollary 1.4.2. *M is invertible with delay τ if and only if there exists a finite automaton M' such that M' is an inverse with delay τ of M.*

A finite automaton $M = \langle X, Y, S, \delta, \lambda \rangle$ is said to be *weakly invertible*, if for any s in S, and any α, α' in X^ω, $\lambda(s, \alpha) = \lambda(s, \alpha')$ implies $\alpha = \alpha'$. In other words, M is weakly invertible, if and only if for any s in S and any α in X^ω, α can be uniquely determined by s and $\lambda(s, \alpha)$, if and only if for any s in S, $\lambda_s|_{X^\omega}$ is injective.

Evidently, if M is weakly invertible, then $M'' = \langle X, Y, S'', \delta'', \lambda'' \rangle$ is weakly invertible in the case where $M'' \prec M$ or $M'' \leqslant M$.

M is weakly invertible, if and only if for any s in S, any x, x' in X, and any α, α' in X^ω, $\lambda(s, x\alpha) = \lambda(s, x'\alpha')$ implies $x = x'$, that is, for any s in S, any x in X, and any α in X^ω, x can be uniquely determined by s and $\lambda(s, x\alpha)$. In fact, the *only if* part is trivial. To prove the *if* part, suppose that $\lambda(s, \alpha) = \lambda(s, \alpha')$ for s in S and α, α' in X^ω. We prove $\alpha = \alpha'$. Denote $\alpha = x_0 x_1 \dots$, $\alpha' = x'_0 x'_1 \dots$, for some x_i, x'_i in X, $i = 0, 1, \dots$ We prove

$x_i = x'_i$ by induction on i. *Basis* : $i = 0$. Since for any t in S, any x, x' in X, and any β, β' in X^ω, $\lambda(t, x\beta) = \lambda(t, x'\beta')$ implies $x = x'$, we have $x_0 = x'_0$. *Induction step* : Suppose that $x_i = x'_i$ holds for $i = 0, 1, \ldots, n$. Denote $s'' = \delta(s, x_0 \ldots x_n)$. Then $s'' = \delta(s, x'_0 \ldots x'_n)$. From $\lambda(s, x_0 x_1 \ldots) = \lambda(s, x'_0 x'_1 \ldots)$, we have $\lambda(s'', x_{n+1} x_{n+2} \ldots) = \lambda(s'', x'_{n+1} x'_{n+2} \ldots)$. Since for any t in S, any x, x' in X, and any β, β' in X^ω, $\lambda(t, x\beta) = \lambda(t, x'\beta')$ implies $x = x'$, we have $x_{n+1} = x'_{n+1}$. We conclude that $x_i = x'_i$ holds for any $i \geqslant 0$, that is, $\alpha = \alpha'$.

A finite automaton $M = \langle X, Y, S, \delta, \lambda \rangle$ is said to be *weakly invertible with delay* τ, τ being a nonnegative integer, if for any s in S and any x_i in X, $i = 0, 1, \ldots, \tau$, x_0 can be uniquely determined by s and $\lambda(s, x_0 \ldots x_\tau)$, that is, for any s in S and any x_i, x'_i in X, $i = 0, 1, \ldots, \tau$, $\lambda(s, x_0 \ldots x_\tau) = \lambda(s, x'_0 \ldots x'_\tau)$ implies $x_0 = x'_0$.

Evidently, if M is weakly invertible with delay τ, then $M'' = \langle X, Y, S'', \delta'', \lambda'' \rangle$ is weakly invertible with delay τ in the case where $M'' \prec M$ or $M'' \leqslant M$.

Clearly, if M is weakly invertible with delay τ, then M is weakly invertible. Below we prove that its converse proposition also holds.

Let $M = \langle X, Y, S, \delta, \lambda \rangle$ be a finite automaton. Construct a graph $G'_M = (V, \Gamma)$ as follows. Let

$$R' = \{(\delta(s, x), \delta(s, x')) \mid x \neq x', \lambda(s, x) = \lambda(s, x'), x, x' \in X, s \in S\}.$$

In the case of $R' \neq \varnothing$, let the vertex set V of G'_M be the minimal subset of $S \times S$ satisfying the following conditions: (a) $R' \subseteq V$, (b) if $(s, s') \in V$ and $\lambda(s, x) = \lambda(s', x')$ for $x, x' \in X$, then $(\delta(s, x), \delta(s', x')) \in V$. Let the arc set Γ of G'_M be the set of all arcs $((s, s'), (\delta(s, x), \delta(s', x')))$ satisfying the condition: $(s, s') \in V$, $x, x' \in X$, and $\lambda(s, x) = \lambda(s', x')$. In the case of $R' = \varnothing$, let G'_M be the empty graph.

Theorem 1.4.3. *M is weakly invertible if and only if G'_M has no circuit. Moreover, if G'_M has no circuit and the level of G'_M is $\rho - 1$, then M is weakly invertible with delay $\rho + 1$ and not weakly invertible with delay τ for any $\tau \leqslant \rho$.*

Proof. Replacing s'_0, R, G_M, "invertible" in the proof of Theorem 1.4.1 by s_0, R', G'_M, "weakly invertible", respectively, we obtain a proof of the theorem. Below we give the details.

Suppose that G'_M has a circuit, say w. From the construction of G'_M, for any vertex on w, there exists a path of which the terminal vertex is the vertex on w and the initial vertex is in R'. Thus there exists a path $u_1 u_2 \ldots u_k$ such that the initial vertex of u_1 is in R' and $u_r u_{r+1} \ldots u_k$ is a circuit for some r, $1 \leqslant r \leqslant k$. Let (s_i, s'_i) be the initial vertex of u_i, $i = 1, 2, \ldots, k$. Then

the terminal vertex of u_k is (s_r, s'_r). From the construction of G'_M, there exist x_i, $x'_i \in X$, $i = 1, 2, \ldots, k$, such that $\delta(s_i, x_i) = s_{i+1}$, $\delta(s'_i, x'_i) = s'_{i+1}$, $i = 1, 2, \ldots, k-1$, $\delta(s_k, x_k) = s_r$, $\delta(s'_k, x'_k) = s'_r$, and $\lambda(s_i, x_i) = \lambda(s'_i, x'_i)$, $i = 1, 2, \ldots, k$. From $(s_1, s'_1) \in R'$, there exist $s_0 \in S$, x_0, $x'_0 \in X$, such that $\delta(s_0, x_0) = s_1$, $\delta(s_0, x'_0) = s'_1$, $\lambda(s_0, x_0) = \lambda(s_0, x'_0)$ and $x_0 \neq x'_0$. Taking

$$\alpha = x_0 x_1 \ldots x_{r-1} x_r \ldots x_k x_r \ldots x_k \ldots,$$
$$\alpha' = x'_0 x'_1 \ldots x'_{r-1} x'_r \ldots x'_k x'_r \ldots x'_k \ldots,$$

we then have $\lambda(s_0, \alpha) = \lambda(s_0, \alpha')$. Since $x_0 \neq x'_0$, M is not weakly invertible.

Conversely, suppose that G'_M has no circuit. Then the level of G'_M is an integer, say $\rho - 1$. In the case of $R' = \varnothing$, it is evident that $\rho = -1$ and M is weakly invertible with delay 0 $(= \rho + 1)$. In the case of $R \neq \varnothing$, for any state s_0 of M, and any input sequences $\alpha = x_0 x_1 \ldots x_{\rho+1}$ and $\alpha' = x'_0 x'_1 \ldots x'_{\rho+1}$ of length $\rho + 2$, x_i, $x'_i \in X$, $i = 0, 1, \ldots, \rho + 1$, we prove by reduction to absurdity that $\lambda(s, \alpha) = \lambda(s, \alpha')$ implies $x_0 = x'_0$. Suppose to the contrary that $\lambda(s_0, x_0 x_1 \ldots x_{\rho+1}) = \lambda(s_0, x'_0 x'_1 \ldots x'_{\rho+1})$ and $x_0 \neq x'_0$, for some state s_0 in S, and some input letters x_i, x'_i, $i = 0, 1, \ldots, \rho + 1$ in X. Denote $s_i = \delta(s_{i-1}, x_{i-1})$, $s'_i = \delta(s'_{i-1}, x'_{i-1})$, $i = 1, 2, \ldots, \rho + 2$, where $s'_0 = s_0$. Since $\lambda(s_0, x_0 x_1 \ldots x_{\rho+1}) = \lambda(s_0, x'_0 x'_1 \ldots x'_{\rho+1})$, we have $\lambda(s_i, x_i) = \lambda(s'_i, x'_i)$, $i = 0, 1, \ldots, \rho+1$, where $s'_0 = s_0$. From $x_0 \neq x'_0$, we have $(s_1, s'_1) \in R'$, and for any i, $1 \leqslant i \leqslant \rho+1$, there exists an arc, say u_i, such that the initial vertex of u_i is (s_i, s'_i) and the terminal vertex of u_i is (s_{i+1}, s'_{i+1}). Thus $u_1 u_2 \ldots u_{\rho+1}$ is a path of G'_M. Since the length of the path is $\rho + 1$, the level of G'_M is at least ρ. This contradicts that the level of G'_M is $\rho - 1$. We conclude that M is weakly invertible with delay $\rho + 1$.

Let $0 \leqslant \tau \leqslant \rho$. Since the level of G'_M is $\rho - 1$, there exists a path of length τ of G'_M, say $u_1 u_2 \ldots u_\tau$. Denote the initial vertex of u_i by (s_i, s'_i) and the terminal vertex of u_i by (s_{i+1}, s'_{i+1}), $i = 1, 2, \ldots, \tau$. From the construction of G'_M, without loss of generality, suppose that (s_1, s'_1) is in R'. From the definition of R', there exist x_0, x'_0 in X and s_0 in S, such that $\delta(s_0, x_0) = s_1$, $\delta(s_0, x'_0) = s'_1$, $\lambda(s_0, x_0) = \lambda(s_0, x'_0)$ and $x_0 \neq x'_0$. From the construction of G'_M, there exist x_i, x'_i in X, $i = 1, 2, \ldots, \tau$, such that $\delta(s_i, x_i) = s_{i+1}$, $\delta(s'_i, x'_i) = s'_{i+1}$, and $\lambda(s_i, x_i) = \lambda(s'_i, x'_i)$, $i = 1, 2, \ldots, \tau$. Thus we have $\lambda(s_0, x_0 x_1 \ldots x_\tau) = \lambda(s_0, x'_0 x'_1 \ldots x'_\tau)$. From $x_0 \neq x'_0$, M is not weakly invertible with delay τ. □

Corollary 1.4.3. *If M is weakly invertible, then there exists $\tau \leqslant n(n-1)/2$ such that M is weakly invertible with delay τ, where n is the element number in the state alphabet of M.*

Proof. Replacing R, G_M, "invertible", "Theorem 1.4.1" in the proof of Corollary 1.4.1 by R', G'_M, "weakly invertible", "Theorem 1.4.3", respectively, we obtain a proof of the corollary. Below we give the details.

Suppose that M is weakly invertible. Whenever G'_M is the empty graph, M is invertible with delay 0 and $0 \leqslant n(n-1)/2$. Whenever G'_M is not the empty graph, then $G'_M = \langle V, \Gamma \rangle$ has no circuit. This yields that $s_1 \neq s_2$ for any (s_1, s_2) in V. Thus $|V| \leqslant n(n-1)$. It is evident that $(s_1, s_2) \in R'$ if and only if $(s_2, s_1) \in R'$. From the construction of G'_M, this yields that $(s_1, s_2) \in V$ if and only if $(s_2, s_1) \in V$, and that $((s_1, s_2), (s_3, s_4)) \in \Gamma$ if and only if $((s_2, s_1), (s_4, s_3)) \in \Gamma$. Therefore, the number of vertices with level i is at least 2, for any i, $0 \leqslant i \leqslant \rho$, where $\rho - 1$ is the level of G'_M. Then we have $2(\rho + 1) \leqslant n(n-1)$. Take $\tau = \rho + 1$. Then $\tau \leqslant n(n-1)/2$. From Theorem 1.4.3, M is weakly invertible with delay τ. □

Let $M = \langle X, Y, S, \delta, \lambda \rangle$ and $M' = \langle Y, X, S', \delta', \lambda' \rangle$ be two finite automata. M' is called a *weak inverse* with delay τ of M, if for any s in S there exists s' in S' such that (s', s) is a match pair with delay τ. M is called an *original weak inverse* with delay τ of M', if M' is a weak inverse with delay τ of M. M' is called a *weak inverse* with delay τ, if M' is a weak inverse with delay τ of some finite automaton. M' is called a *weak inverse*, if M' is a weak inverse with delay τ for some τ.

Clearly, if M' is a weak inverse with delay τ of M, then M' is a weak inverse with delay τ of $M'' = \langle X, Y, S'', \delta'', \lambda'' \rangle$ in the case where $M'' \prec M$ or $M'' \leqslant M$. If M' is a weak inverse with delay τ of M, then $M'' = \langle Y, X, S'', \delta'', \lambda'' \rangle$ is a weak inverse with delay τ of M in the case where $M' \prec M''$ or $M' \leqslant M''$. Therefore, if M' is a weak inverse with delay τ, then $M'' = \langle Y, X, S'', \delta'', \lambda'' \rangle$ is a weak inverse with delay τ in the case where $M' \prec M''$ or $M' \leqslant M''$.

Theorem 1.4.4. *If M is weakly invertible with delay τ, then there exists a finite automaton M' such that M' is a weak inverse with delay τ of M.*

Proof. Let $M = \langle X, Y, S, \delta, \lambda \rangle$. Since M is weakly invertible with delay τ, for any $s \in S$ and any $x_0, x_1, \ldots, x_\tau \in X$, x_0 can be uniquely determined by s and $\lambda(s, x_0 x_1 \ldots x_\tau)$. Thus we can construct a single-valued mapping f from $S \times Y^{\tau+1}$ to X satisfying the condition: if $y_0 y_1 \ldots y_\tau = \lambda(s, x_0 x_1 \ldots x_\tau)$, $s \in S$ and $x_0, x_1, \ldots, x_\tau \in X$, then $f(s, y_\tau, \ldots, y_1, y_0) = x_0$. Let $M' = \langle Y, X, S', \delta', \lambda' \rangle$ be a finite automaton, where

$$S' = \{\langle c, s, y_{-1}, \ldots, y_{-\tau} \rangle \mid c = 0, 1, \ldots, \tau,\ s \in S,\ y_{-1}, \ldots, y_{-\tau} \in Y\},$$

$$\delta'(\langle c, s, y_{-1}, \ldots, y_{-\tau} \rangle, y_0) = \begin{cases} \langle c+1, s, y_0, y_{-1}, \ldots, y_{-\tau+1} \rangle, & \text{if } 0 \leqslant c < \tau, \\ \langle c, \delta(s, f(s, y_0, y_{-1}, \ldots, y_{-\tau})), y_0, \ldots, y_{-\tau+1} \rangle, & \text{if } c = \tau, \end{cases}$$

$$\lambda'(\langle c, s, y_{-1}, \ldots, y_{-\tau}\rangle, y_0) = f(s, y_0, y_{-1}, \ldots, y_{-\tau}),$$
$$c = 0, 1, \ldots, \tau,\ s \in S,\ y_0, y_{-1}, \ldots, y_{-\tau} \in Y.$$

We prove that M' is a weak inverse with delay τ of M. Given any state s_0 in S, take any τ elements $y_{-1}, \ldots, y_{-\tau}$ in Y and let s'_0 be the state $\langle 0, s_0, y_{-1}, \ldots, y_{-\tau}\rangle$ in S'. To prove that s'_0 τ-matches s_0, for any $x_0, x_1, \ldots \in X$, let

$$\lambda(s_0, x_0 x_1 \ldots) = y_0 y_1 \ldots$$

for some $y_0, y_1, \ldots$ in Y. Denote

$$\delta(s_0, x_0 x_1 \ldots x_i) = s_{i+1},\ i = 0, 1, \ldots$$

From the definition of f, we have

$$f(s_i, y_{i+\tau}, \ldots, y_{i+1}, y_i) = x_i,\ i = 0, 1, \ldots$$

Let

$$\delta'(s'_0, y_0 y_1 \ldots y_i) = s'_{i+1},\ i = 0, 1, \ldots$$

Clearly,

$$s'_i = \langle i, s_0, y_{i-1}, \ldots, y_{i-\tau}\rangle,\ i = 0, 1, \ldots, \tau.$$

We prove by induction on i that $s'_i = \langle \tau, s_{i-\tau}, y_{i-1}, \ldots, y_{i-\tau}\rangle$ holds for any $i \geqslant \tau$. *Basis* : $i = \tau$. The result has proven above. *Induction step* : Suppose that $s'_i = \langle \tau, s_{i-\tau}, y_{i-1}, \ldots, y_{i-\tau}\rangle$ holds. We prove that $s'_{i+1} = \langle \tau, s_{i+1-\tau}, y_i, \ldots, y_{i+1-\tau}\rangle$ holds. Since $s'_{i+1} = \delta'(s'_i, y_i)$, from the definition of δ' and the induction hypothesis, we have

$$\begin{aligned} s'_{i+1} &= \langle \tau, \delta(s_{i-\tau}, f(s_{i-\tau}, y_i, y_{i-1}, \ldots, y_{i-\tau})), y_i, \ldots, y_{i+1-\tau}\rangle \\ &= \langle \tau, \delta(s_{i-\tau}, x_{i-\tau}), y_i, \ldots, y_{i+1-\tau}\rangle \\ &= \langle \tau, s_{i-\tau+1}, y_i, \ldots, y_{i+1-\tau}\rangle. \end{aligned}$$

We conclude that $s'_i = \langle \tau, s_{i-\tau}, y_{i-1}, \ldots, y_{i-\tau}\rangle$ holds for any $i \geqslant \tau$. Let

$$\lambda'(s'_i, y_i) = x'_i,\ i = 0, 1, \ldots$$

Then for any $i \geqslant \tau$, from the definition of λ', we have

$$x'_i = f(s_{i-\tau}, y_i, y_{i-1}, \ldots, y_{i-\tau}) = x_{i-\tau}.$$

It follows that $\lambda'(s'_0, \lambda(s_0, x_0 x_1 \ldots)) = x'_0 x'_1 \ldots x'_{\tau-1} x_0 x_1 \ldots$ Thus s'_0 τ-matches s_0. Therefore, M' is a weak inverse with delay τ of M. □

Corollary 1.4.4. *M is weakly invertible with delay τ if and only if there exists a finite automaton M' such that M' is a weak inverse with delay τ of M.*

Theorem 1.4.5. *Let $M = \langle X, Y, S, \delta, \lambda\rangle$ be a finite automaton. If M is weakly invertible with delay τ, then $|X| \leqslant |Y|$.*

Proof. Denote $l = |X|$ and $m = |Y|$. We prove by reduction to absurdity that $|X| \leqslant |Y|$. Suppose to the contrary that $|X| > |Y|$. Take any s in S. Let $V_h = \{\lambda(s, x_0 \dots x_{\tau-1+h}) \mid x_i \in X, i = 0, \dots, \tau-1+h\}$. Clearly, $|V_h| \leqslant m^{\tau+h}$. On the other hand, since M is weakly invertible with delay τ, $x_0 \dots x_{h-1}$ can be determined by s and $\lambda(s, x_0 \dots x_{\tau-1+h})$. Thus we have $l^h \leqslant |V_h|$. Therefore, $m^{\tau+h} \geqslant l^h$. It follows that $m^\tau (m/l)^h \geqslant 1$. From $|X| > |Y|$, we have $m/l < 1$. Thus $\lim_{h\to+\infty} m^\tau (m/l)^h = 0$. From $m^\tau (m/l)^h \geqslant 1$, we have $0 \geqslant 1$. This is a contradiction. We conclude that $|X| \leqslant |Y|$. □

Theorem 1.4.6. *Let $M = \langle X, Y, S, \delta, \lambda\rangle$ be a finite automaton. If $|X| = |Y|$ and M is weakly invertible with delay τ, then $\lambda(S, X^r)$(i.e., $\{\lambda(s, \alpha) \mid s \in S, \alpha \in X^r\}$) $= Y^r$ holds for any nonnegative integer r.*

Proof. In the case of $r = 0$, $\lambda(S, X^r) = \{\varepsilon\} = Y^r$. For any positive integer r, we prove by reduction to absurdity that $\lambda(S, X^r) = Y^r$. Suppose to the contrary that $\lambda(S, X^r) \neq Y^r$. Clearly, $\lambda(S, X^r) \subseteq Y^r$. It follows that $|\lambda(S, X^r)| \leqslant m^r - 1$, where $m = |Y|$. Take any s in S. Let $V_{r,h} = \{\lambda(s, x_0 \dots x_{\tau-1+hr}) \mid x_i \in X, i = 0, \dots, \tau-1+hr\}$. It is easy to see that for any $y_0 \dots y_{\tau-1+hr} \in V_{r,h}$, $y_{\tau+ir} \dots y_{\tau-1+(i+1)r}$ is in $\lambda(S, X^r)$, $i = 0, \dots, h-1$. Thus we have $|V_{r,h}| \leqslant m^\tau (m^r - 1)^h$. On the other hand, since M is weakly invertible with delay τ, we have $|V_{r,h}| \geqslant l^{hr}$, where $l = |X|$. Therefore, $m^\tau (m^r - 1)^h \geqslant l^{hr}$. From $l = m$, we have $m^\tau ((m^r - 1)/m^r)^h \geqslant 1$. It follows that $\lim_{h\to+\infty} m^\tau ((m^r-1)/m^r)^h \geqslant 1$. That is, $0 \geqslant 1$. This is a contradiction. We conclude that $\lambda(S, X^r) = Y^r$. □

1.5 Error Propagation and Feedforward Invertibility

Let $M' = \langle Y, X, S', \delta', \lambda'\rangle$ be an inverse finite automaton with delay τ of $M = \langle X, Y, S, \delta, \lambda\rangle$. For any infinite input sequence $\alpha \in X^\omega$ and any state $s \in S$, let $\beta = \lambda(s, \alpha)$. Then for any state $s' \in S$ of M', there exists $\alpha_0 \in X^*$ of length τ such that $\lambda'(s', \beta) = \alpha_0\alpha$. Suppose that $\beta = \beta_1\beta_2$ and $\beta' = \beta'_1\beta_2$ with $|\beta_1| = |\beta'_1|$. Let $\alpha = \alpha_1\alpha_2$ with $|\alpha_1| = |\beta'_1|$. Then

$$\begin{aligned}\lambda'(s', \beta') &= \lambda'(s', \beta'_1)\lambda'(\delta'(s', \beta'_1), \beta_2)\\ &= \lambda'(s', \beta'_1)\lambda'(\delta'(s', \beta'_1), \lambda(\delta(s, \alpha_1), \alpha_2))\\ &= \lambda'(s', \beta'_1)\alpha'_0\alpha_2\end{aligned}$$

for some sequence α_0' of length τ in X^*. Since $|\alpha_0\alpha_1| = |\lambda'(s', \beta_1')\alpha_0'|$, the i-th letter of $\lambda'(s', \beta)$ equals the i-th letter of $\lambda'(s', \beta')$ whenever $i > |\beta_1'| + \tau$. This means that propagation of decoding errors of the inverse M' to M is at most τ letters.

For the weak inverse case, one letter error could cause infinite decoding errors. For example, let X and Y be $\{0, 1\}$, and M a 1-order input-memory finite automaton M_f, where f is a single-valued mapping from X^2 to Y

$$f(x_0, x_{-1}) = x_0 \oplus x_{-1},$$

where $\oplus$ stands for addition modulo 2. Let M' be a (0,1)-order memory finite automaton M_g, where g is a single-valued mapping from $X \times Y$ to X

$$g(x_{-1}, y_0) = x_{-1} \oplus y_0.$$

It is easy to verify that any state x_{-1} of M' 0-matches the state x_{-1} of M. Thus M' is a weak inverse of M with delay 0. For the zero input sequence, i.e., $00\ldots0\ldots$, and the initial state 0 of M, the output sequence of M is the zero sequence $00\ldots0\ldots$ Since (0,0) is a match pair with delay 0, for the zero input sequence $00\ldots0\ldots$ and the initial state 0 of M', the output sequence of M' is the zero sequence $00\ldots0\ldots$ On the other hand, we can verify that for the input sequence $10\ldots0\ldots$, all zero but the first letter 1, and the initial state 0 of M', the output sequence of M' is the 1 sequence, i.e., $11\ldots1\ldots$ Thus propagation of decoding errors of the weak inverse M' to M is infinite.

We use $R(-n, \alpha)$ to denote the suffix of α of length $|\alpha| - n$ in the case of $|\alpha| \geqslant n$ or the empty word in the case of $|\alpha| < n$. For any α, β in Y^* with $|\alpha| = |\beta|$ and any nonnegative integer k, we use $\alpha =_k \beta$ to denote $R(-k, \alpha) = R(-k, \beta)$.

Let $M' = \langle Y, X, S', \delta', \lambda' \rangle$ be a weak inverse finite automaton with delay τ of $M = \langle X, Y, S, \delta, \lambda \rangle$. For any s in S and any s' in S', if s' τ-matches s, and if for any α in X^* and any β in Y^* with $|\alpha| = |\beta|$, and any k, $0 \leqslant k \leqslant |\beta| - c$, $\lambda(s, \alpha) =_k \beta$ implies $\lambda'(s', \lambda(s, \alpha)) =_{k+c} \lambda'(s', \beta)$, (s, s') is called a (τ, c)-*match pair*. If for any s in S there exists s' in S' such that (s, s') is a (τ, c)-match pair, we say that *propagation of weakly decoding errors of M' to M is bounded with length of error propagation $\leqslant c$*. The minimal nonnegative integer c satisfying the above condition is called the *length of error propagation*.

Theorem 1.5.1. *Let $M' = \langle Y, X, S', \delta', \lambda' \rangle$ be a weak inverse finite automaton with delay τ of $M = \langle X, Y, S, \delta, \lambda \rangle$. Assume that propagation of weakly decoding errors of M' to M is bounded with length of error propagation $\leqslant c$, where $c \geqslant \tau$. Then we can construct a c-order semi-input-memory finite automaton $\mathcal{SIM}(M'', f)$ such that $\mathcal{SIM}(M'', f)$ is a weak inverse finite automaton with delay τ of M.*

Proof. Since M' is a weak inverse finite automaton with delay τ of M and propagation of weakly decoding errors of M' to M is bounded with length of error propagation $\leqslant c$, for each s in S, we can choose a state of M', say $\varphi(s)$, such that $(s, \varphi(s))$ is a (τ, c)-match pair.

Given s in S, let $T_{s,0} = \{s\}$, $T_{s,i+1} = \{\delta(s_i, x) \mid x \in X, s_i \in T_{s,i}\}$, $T'_{s,0} = \{\varphi(s)\}$, and $T'_{s,i+1} = \{\delta'(s'_i, y) \mid y \in Y, s'_i \in T'_{s,i}\}$, $i = 0, 1, \ldots$ Since X, Y, S and S' are finite, it is easy to see that the sequence

$$(T_{s,0}, T'_{s,0}), (T_{s,1}, T'_{s,1}), \ldots, (T_{s,i}, T'_{s,i}), \ldots$$

is ultimately periodic, i.e., there exist $t_s \geqslant 0$ and $e_s \geqslant 1$ such that $T_{s,i+e_s} = T_{s,i}$ and $T'_{s,i+e_s} = T'_{s,i}$, $i = t_s, t_{s+1}, \ldots$ Let t_s and e_s be the least integers satisfying the above condition. Let $M_s = \langle Y_s, S_s, \delta_s, \lambda_s \rangle$ be an autonomous finite automaton, where

$$Y_s = S_s = \{w_{s,i}, i = 0, 1, \ldots, t_s + c + e_s - 1\},$$
$$\delta_s(w_{s,i}) = \begin{cases} w_{s,i+1}, & \text{if } 0 \leqslant i < t_s + c + e_s - 1, \\ w_{s,t_s+c}, & \text{if } i = t_s + c + e_s - 1, \end{cases}$$
$$\lambda_s(w) = w, \ w \in S_s.$$

Take a subset I of S with $\cup_{i \geqslant 0, s \in I} T_{s,i} = S$, i.e., $\{\delta(s, \alpha) \mid s \in I, \alpha \in X^*\} = S$. Let the autonomous finite automaton M'' be the juxtaposition of all M_s, s ranging over I, i.e., $M'' = \langle Y'', S'', \delta'', \lambda'' \rangle$, where $Y'' = \cup_{s \in I} Y_s$, $S'' = \cup_{s \in I} S_s$, and $\delta''(w) = \delta_s(w)$, $\lambda''(w) = \lambda_s(w)$, for any w in S_s and any s in I.

For any subset T of S, let $R(T) = \{\lambda(t, \alpha) \mid t \in T, \alpha \in X^*\}$. To define the single-valued mapping f from $Y^{c+1} \times S''$ to X, we need the following result.

For any s in S and any nonnegative n, states in $T'_{s,n}$ are c-carelessly-equivalent regarding $R(T_{s,n})$, i.e., for any s', s'' in $T'_{s,n}$ and any β in $R(T_{s,n})$, $\lambda'(s', \beta) =_c \lambda'(s'', \beta)$ holds.

To prove this result, let β be in $R(T_{s,n})$. From the definition of $R(T_{s,n})$, there exist s_n in $T_{s,n}$ and α in X^* such that $\lambda(s_n, \alpha) = \beta$. From the definition of $T_{s,n}$, there exists α_n in X^* of length n such that $\delta(s, \alpha_n) = s_n$. Let $\beta_n = \lambda(s, \alpha_n)$, $s'_n = \delta'(\varphi(s), \beta_n)$, $\alpha'_n = \lambda'(\varphi(s), \beta_n)$ and $\alpha' = \lambda'(s'_n, \beta)$. Suppose that s''_n is a state in $T'_{s,n}$. Then there exists β''_n in Y^* of length n such that $\delta'(\varphi(s), \beta''_n) = s''_n$. Let $\alpha''_n = \lambda'(\varphi(s), \beta''_n)$ and $\alpha'' = \lambda'(s''_n, \beta)$. Since $(s, \varphi(s))$ is a (τ, c)-match pair, we have

$$\alpha''_n \alpha'' =_{n+c} \alpha'_n \alpha'.$$

It follows that $\alpha'' =_c \alpha'$. Since $R(-c, \alpha')$ is independent of the choice of s''_n, states in $T'_{s,n}$ are c-carelessly-equivalent regarding $R(T_{s,n})$.

Given a state $w_{s,i}$ of M'' and $y_i, \ldots, y_{i-c}$ in Y, we define $f(y_i, \ldots, y_{i-c}, w_{s,i})$ as follows. When $i \geqslant c$ and $y_{i-c} \ldots y_i \in R(T_{s,i-c})$, we define

$f(y_i, \ldots, y_{i-c}, w_{s,i}) = \lambda'(\delta'(s'_{i-c}, y_{i-c} \ldots y_{i-1}), y_i)$, s'_{i-c} being a state in $T'_{s,i-c}$. From the result proven above, states in $T'_{s,i-c}$ are c-carelessly-equivalent regarding $R(T_{s,i-c})$. Thus the above value of f is independent of the choice of s'_{i-c}. When $\tau \leqslant i < c$ and $y_0 \ldots y_i \in R(T_{s,0})$, for any $x_0, \ldots, x_i \in X$, $\lambda(s, x_0 \ldots x_i) = y_0 \ldots y_i$ implies $\lambda'(\varphi(s), y_0 \ldots y_i) = x_{-\tau} \ldots x_{-1} x_0 \ldots x_{i-\tau}$ for some $x_{-\tau}$, …, x_{-1} in X. Evidently, $x_{i-\tau}$ is uniquely determined by i and y_0, …, y_i. Since $y_0 \ldots y_i \in R(T_{s,0})$, such $x_0, \ldots, x_i$ are existent. We define $f(y_i, \ldots, y_{i-c}, w_{s,i}) = x_{i-\tau}$. Otherwise, the value of $f(y_i, \ldots, y_{i-c}, w_{s,i})$ may be arbitrarily chosen from X.

To prove that $\mathcal{SIM}(M'', f)$ is a weak inverse with delay τ of M, we first prove that for any s in I, there exists a state s''' of $\mathcal{SIM}(M'', f)$ such that s''' τ-matches s. Let s be in I. Take $s''' = \langle y_{-c}, \ldots, y_{-1}, w_{s,0} \rangle$, where y_{-1}, …, y_{-c} are arbitrary elements in Y. For any $x_0, \ldots, x_j$ in X, let $y_0 \ldots y_j = \lambda(s, x_0 \ldots x_j)$ and $z_0 \ldots z_j = \lambda'''(s''', y_0 \ldots y_j)$, where λ''' is the output function of $\mathcal{SIM}(M'', f)$. We prove that $z_i = x_{i-\tau}$ holds for any i, $\tau \leqslant i \leqslant j$. In the case of $\tau \leqslant i < c$, since $y_0 \ldots y_i \in R(T_{s,0})$, from the construction of $\mathcal{SIM}(M'', f)$ and the definition of f, it immediately follows that $z_i = x_{i-\tau}$. In the case of $i \geqslant c$, take $h = i$ if $i < t_s + c + e_s$, and take $h = t_s + c + d$ if $i = t_s + c + d + ke_s$ for some $k > 0$ and d, $0 \leqslant d < e_s$. Since $h - c \geqslant t_s$ and $h = i \pmod{e_s}$, or $h = i$, we have $(T_{s,i-c}, T'_{s,i-c}) = (T_{s,h-c}, T'_{s,h-c})$. Let $s_{i-c} = \delta(s, x_0 \ldots x_{i-c-1})$ and $s'_{i-c} = \delta'(\varphi(s), y_0 \ldots y_{i-c-1})$. Then we have $s_{i-c} \in T_{s,h-c}$, $s'_{i-c} \in T'_{s,h-c}$, and $y_{i-c} \ldots y_i \in R(T_{s,h-c})$. From the construction of $\mathcal{SIM}(M'', f)$ and the definition of f, it is easy to show that

$$\begin{aligned}
&\delta'''(s''', y_0 \ldots y_{i-1}) = \langle y_{i-1}, \ldots, y_{i-c}, w_{s,h} \rangle, \\
&z_i = \lambda'''(\langle y_{i-1}, \ldots, y_{i-c}, w_{s,h} \rangle, y_i) \\
&\quad = f(y_i, y_{i-1}, \ldots, y_{i-c}, w_{s,h}) \\
&\quad = \lambda'(\delta'(s'_{i-c}, y_{i-c} \ldots y_{i-1}), y_i),
\end{aligned}$$

where δ''' is the next state function of $\mathcal{SIM}(M'', f)$. Since $(s, \varphi(s))$ is a (τ, c)-match pair, we have

$$\lambda'(\delta'(s'_{i-c}, y_{i-c} \ldots y_{i-1}), y_i) = \lambda'(\delta'(\varphi(s), y_0 \ldots y_{i-1}), y_i) = x_{i-\tau}.$$

It follows that $z_i = x_{i-\tau}$. Therefore, for any s in I, there exists a state s''' of $\mathcal{SIM}(M'', f)$ such that s''' τ-matches s. It follows that $\delta'''(s''', \beta)$ τ-matches $\delta(s, \alpha)$, if $\beta = \lambda(s, \alpha)$. From $S = \{\delta(s, \alpha) \mid s \in I, \alpha \in X^*\}$, for any s in S, there exists a state s''' of $\mathcal{SIM}(M'', f)$ such that s''' τ-matches s. Thus $\mathcal{SIM}(M'', f)$ is a weak inverse finite automaton with delay τ of M. □

Corollary 1.5.1. *In the above theorem, if M is a linear finite automaton over $GF(q)$ and $S = \{\delta(0, \alpha) \mid \alpha \in X^*\}$, then $\mathcal{SIM}(M'', f)$ can be equivalent to a linear c-order input-memory finite automaton.*

Proof. In the proof of the above theorem, take $I = \{0\}$. We use $R_j(T_{0,i})$ to denote the set of all elements in $R(T_{0,i})$ of length j. Since M is linear over $GF(q)$, for any j, $R_j(T_{0,0})$ is a vector space over $GF(q)$. Evidently, there exists a linear mapping from $R_{i+j}(T_{0,0})$ onto $R_j(T_{0,i})$. It follows that $R_j(T_{0,i})$ is a vector space over $GF(q)$. Clearly, $T_{0,i} \subseteq T_{0,i+1}$. This yields that $R_j(T_{0,i}) \subseteq R_j(T_{0,i+1})$. Denote $n = t_0 + c + e_0 - 1$. In the case of $y_{i-c} \dots y_i \in R(T_{0,i-c})$ & $c \leqslant i \leqslant n$, or in the case of $y_0 \dots y_i \in R(T_{0,0})$ & $\tau \leqslant i < c$ & $y_{i-c} = \dots = y_{-1} = 0$, we define the value of $f(y_i, \dots, y_{i-c}, w_{0,i})$ as identical as that in the proof of the above theorem. Taking

$$s''' = \langle y_{-c}, \dots, y_{-c+\tau-1}, 0, \dots, 0, w_{0,0} \rangle,$$

where $y_{-c}, \dots, y_{-c+\tau-1}$ are arbitrarily chosen from Y, from the proof of the above theorem, $y_0 \dots y_i = \lambda(0, x_0 \dots x_i)$ and $z_0 \dots z_i = \lambda'''(s''', y_0 \dots y_i)$ imply $z_\tau \dots z_i = x_0 \dots x_{i-\tau}$. In the case of $y_{i-c} \dots y_i \in R(T_{0,i-c})$ & $c \leqslant i \leqslant n$, there exist $\bar{s} \in T_{0,i-c}$ and $x_{i-c}, \dots, x_i$ in X such that $\lambda(\bar{s}, x_{i-c} \dots x_i) = y_{i-c} \dots y_i$. Therefore, for any α in X^*, $\delta(0, \alpha) = \bar{s}$ and $\lambda(0, \alpha) = \beta$ yield $x_{i-\tau} = \lambda'''(\delta'''(s''', \beta y_{i-c} \dots y_{i-1}), y_i)$. Since $\bar{s} \in T_{0,i-c} \subseteq T_{0,n-c}$, we may take α such that $|\alpha| = i - c$ or $n - c$. Then $\delta'''(s''', \beta y_{i-c} \dots y_{i-1})$ equals $\langle y_{i-c}, \dots, y_{i-1}, w_{0,i} \rangle$ or $\langle y_{i-c}, \dots, y_{i-1}, w_{0,n} \rangle$. It follows that $f(y_i, \dots, y_{i-c}, w_{0,i}) = x_{i-\tau} = f(y_i, \dots, y_{i-c}, w_{0,n})$. In the case of $y_0 \dots y_i \in R(T_{0,0})$ & $\tau \leqslant i < c$ & $y_{i-c} = \dots = y_{-1} = 0$, there exist $x_0, \dots, x_i$ in X such that $\lambda(0, x_0 \dots x_i) = y_0 \dots y_i$. From the definition of f, $f(y_i, \dots, y_{i-c}, w_{0,i}) = x_{i-\tau}$. Let $\alpha' = 0 \dots 0 x_0 \dots x_i$ and $\beta' = y'_0 \dots y'_n = 0 \dots 0 y_0 \dots y_i$ with $|\alpha'| = |\beta'| = n$. Since $\lambda(0, x_0 \dots x_i) = y_0 \dots y_i$, we have $\lambda(0, \alpha') = \beta'$. It follows that $\lambda(0, 0^{c-i} x_0 \dots x_i) = y'_{n-c} \dots y'_n \in R(T_{0,n-c})$. Thus $f(y_i, \dots, y_{i-c}, w_{0,n}) = f(y'_n, \dots, y'_{n-c}, w_{0,n}) = x_{i-\tau}$. Therefore, $f(y_i, \dots, y_{i-c}, w_{0,i}) = f(y_i, \dots, y_{i-c}, w_{0,n})$. Noticing that in the proof of the above theorem, $y_{-1}, \dots, y_{-c}$ in s''' are arbitrarily chosen, we can choose values of f at the other points such that $f(y_i, \dots, y_{i-c}, w_{0,i}) = f(y_i, \dots, y_{i-c}, w_{0,n})$ holds for any $y_i, \dots, y_{i-c}$ in Y and any i, $0 \leqslant i \leqslant n$. That is, the value $f(y'_c, \dots, y'_0, w)$ does not depend on w; therefore, f can be regarded as a function f' from Y^{c+1} to X, where $f'(y'_c, \dots, y'_0) = f(y'_c, \dots, y'_0, w_{0,n})$, $y'_c, \dots, y'_0 \in Y$. Thus $\mathcal{SIM}(M'', f)$ is equivalent to the c-order input-memory finite automaton $M_{f'}$.

We prove that f' is linear on $R_{c+1}(T_{0,n-c})$. For any $y_{n-c} \dots y_n$ and $y'_{n-c} \dots y'_n$ in $R_{c+1}(T_{0,n-c})$, from the definitions, there exist $x_0, \dots, x_n$, $x'_0, \dots, x'_n$ in X and $y_0, \dots, y_{n-c-1}$, $y'_0, \dots, y'_{n-c-1}$ in Y such that $y_0 \dots y_n = \lambda(0, x_0 \dots x_n)$ and $y'_0 \dots y'_n = \lambda(0, x'_0 \dots x'_n)$. For any a, a' in $GF(q)$, let $x''_i = a x_i + a' x'_i$, $y''_i = a y_i + a' y'_i$, $i = 0, 1, \dots, n$. Since M is linear, we have $y''_0 \dots y''_n = \lambda(0, x''_0 \dots x''_n)$. Taking $s''' = \langle 0, \dots, 0, w_{0,0} \rangle$, from the proof of the above theorem, s''' τ-matches the state 0 of M. It follows that

$$f'(y''_n, \dots, y''_{n-c}) = f(y''_n, \dots, y''_{n-c}, w_{0,n})$$

$$
\begin{aligned}
&= \lambda'''(\delta'''(s''', y''_0 \dots y''_{n-1}), y''_n) \\
&= x''_{n-\tau} = a x_{n-\tau} + a' x'_{n-\tau} \\
&= a\lambda'''(\delta'''(s''', y_0 \dots y_{n-1}), y_n) + a'\lambda'''(\delta'''(s''', y'_0 \dots y'_{n-1}), y'_n) \\
&= a f(y_n, \dots, y_{n-c}, w_{0,n}) + a' f(y'_n, \dots, y'_{n-c}, w_{0,n}) \\
&= a f'(y_n, \dots, y_{n-c}) + a' f'(y'_n, \dots, y'_{n-c}).
\end{aligned}
$$

Since $R_{c+1}(T_{0,n-c})$ is a subspace of Y^{c+1}, we can expand f such that f' is linear. Therefore, $M_{f'}$ is linear. □

Corollary 1.5.2. *Let M' be a weak inverse finite automaton with delay τ of a linear finite automaton M over $GF(q)$. Assume that propagation of weakly decoding errors of M' to M is bounded with length of error propagation $\leqslant c$, where $c \geqslant \tau$. Then we can construct a linear c-order semi-input-memory finite automaton $\mathcal{SIM}(M'', f)$ over $GF(q)$ such that $\mathcal{SIM}(M'', f)$ is a weak inverse finite automaton with delay τ of M.*

Proof. Since M is linear, M is equivalent to the union of M_a and M_0, where M_a is the maximal linear autonomous finite subautomaton of M and M_0 is a minimal linear finite subautomaton of M. Using Corollary 1.5.1, there exists a linear c-order input-memory finite automaton M_f which is a weak inverse of M_0 with delay τ. From automata $M_a = \langle Y, S_a, \delta_a, \lambda_a \rangle$ and $M_f = \langle Y, X, S_f, \delta_f, \lambda_f \rangle$ construct a finite automaton, say $M'' = \langle Y, X, S_a \times S_f, \delta'', \lambda'' \rangle$, where

$$
\begin{aligned}
&\delta''(\langle s_a, s_f \rangle, y) = \langle \delta_a(s_a), \delta_f(s_f, y + \lambda_a(s_a)) \rangle, \\
&\lambda''(\langle s_a, s_f \rangle, y) = \lambda_f(s_f, y + \lambda_a(s_a)), \\
&\qquad s_a \in S_a,\ s_f \in S_f,\ y \in Y.
\end{aligned}
$$

For any state s of M, s and the state $\langle s, 0 \rangle$ of the union of M_a and M_0 are equivalent. Let the state s_f of M_f τ-matches the state 0 of M_0. It is easy to see that the state $\langle -s, s_f \rangle$ of M'' τ-matches the state $\langle s, 0 \rangle$. Thus $\langle -s, s_f \rangle$ τ-matches s. Therefore, M'' is a weak inverse with delay τ of M.

From M_a construct a linear autonomous finite automaton $M'_a = \langle Y^{c+1}, S_a \times Y^c, \delta'_a, \lambda'_a \rangle$, where

$$
\begin{aligned}
&\delta'_a(\langle s_a, y'_{-1}, \dots, y'_{-c} \rangle) = \langle \delta_a(s_a), \lambda_a(s_a), y'_{-1}, \dots, y'_{-c+1} \rangle, \\
&\lambda'_a(\langle s_a, y'_{-1}, \dots, y'_{-c} \rangle) = \langle \lambda_a(s_a), y'_{-1}, \dots, y'_{-c} \rangle, \\
&\qquad s_a \in S_a,\ y'_{-1}, \dots, y'_{-c} \in Y.
\end{aligned}
$$

Let f' be a function from Y^{2c+2} to X defined by

$$
\begin{aligned}
&f'(y_0, y_{-1}, \dots, y_{-c}, y'_0, y'_{-1}, \dots, y'_{-c}) \\
&\qquad = f(y_0 + y'_0, y_{-1} + y'_{-1}, \dots, y_{-c} + y'_{-c}), \\
&\qquad y_i, y'_i \in Y,\ i = 0, -1, \dots, -c.
\end{aligned}
$$

It is easy to see that for any state $s'' = \langle s_a, s_f \rangle = \langle s_a, y''_{-1}, \ldots, y''_{-c} \rangle$ of M'' and any state $s' = \langle y_{-1}, \ldots, y_{-c}, \langle s_a, y'_{-1}, \ldots, y'_{-c} \rangle \rangle$ of $\mathcal{SIM}(M'_a, f')$, if $y_i + y'_i = y''_i, i = -1, \ldots, -c$, then s'' and s' are equivalent. Thus M'' and $\mathcal{SIM}(M'_a, f')$ are equivalent. It follows that $\mathcal{SIM}(M'_a, f')$ is a weak inverse with delay τ of M. Clearly, $\mathcal{SIM}(M'_a, f')$ is linear. Thus the corollary holds. □

A finite automaton M is said to be *feedforward invertible with delay* τ, if there exists a finite order semi-input-memory finite automaton which is a weak inverse with delay τ of M. A finite automaton M' is said to be a *feedforward inverse with delay* τ, if M' is a finite order semi-input-memory finite automaton which is a weak inverse with delay τ of some finite automaton.

Theorem 1.5.2. *A finite automaton M is feedforward invertible with delay τ, if and only if there exists a finite automaton M' such that M' is a weak inverse with delay τ of M and propagation of weakly decoding errors of M' to M is bounded.*

Proof. if : Assume that M' is a weak inverse with delay τ of M and propagation of weakly decoding errors of M' to M is bounded. Let c' be the length of error propagation. Denote $c = \max(c', \tau)$. Then propagation of weakly decoding errors of M' to M is bounded with length of error propagation $\leqslant c$. From Theorem 1.5.1, there exists a c-order semi-input-memory finite automaton M''' such that M''' is a weak inverse finite automaton with delay τ of M. Thus M is feedforward invertible with delay τ.

only if : Assume that M is feedforward invertible with delay τ. Then there exist a nonnegative integer c and a c-order semi-input-memory finite automaton $\mathcal{SIM}(M'', f)$ such that $\mathcal{SIM}(M'', f)$ is a weak inverse finite automaton with delay τ of M. Thus for each state s of M, we can choose a state of $\mathcal{SIM}(M'', f)$, say $\varphi(s)$, such that $\varphi(s)$ τ-matches s. We prove the following result: for any α in X^* and any β in Y^* with $|\alpha| = |\beta|$, any k, $0 \leqslant k \leqslant |\beta| - c$, $\lambda(s, \alpha) =_k \beta$ implies $\lambda'''(\varphi(s), \lambda(s, \alpha)) =_{k+c} \lambda'''(\varphi(s), \beta)$, where λ''' is the output function of $\mathcal{SIM}(M'', f)$. Denote $|\alpha| = l$. In the case of $l \leqslant k + c$, $R(-k - c, \lambda'''(\varphi(s), \lambda(s, \alpha))) = \varepsilon = R(-k - c, \lambda'''(\varphi(s), \beta))$. In the case of $l > k + c$, let $\varphi(s) = \langle y_{-c}, \ldots, y_{-1}, s''_0 \rangle$ and $\lambda(s, \alpha) = y_0 y_1 \ldots y_{l-1}$, where $y_{-c}, \ldots, y_{-1}, y_0, y_1, \ldots, y_{l-1} \in Y$. Then $\beta = y'_0 \ldots y'_{k-1}\, y_k \ldots y_{l-1}$, for some $y'_0, \ldots, y'_{k-1} \in Y$. Since $\mathcal{SIM}(M'', f)$ is a c-order semi-input-memory finite automaton, we have

$$\begin{aligned} \delta'''(\varphi(s), y_0 \ldots y_{k+c-1}) &= \langle y_{k+c-1}, \ldots, y_k, s''_{k+c} \rangle \\ &= \delta'''(\varphi(s), y'_0 \ldots y'_{k-1} y_k \ldots y_{k+c-1}), \end{aligned}$$

where $s''_{i+1} = \delta''(s''_i)$, $i = 0, 1, \ldots$, δ'' and δ''' are the next state functions of M'' and $\mathcal{SIM}(M'', f)$, respectively. Thus

$$R(-k-c, \lambda'''(\varphi(s), \lambda(s,\alpha))) = \lambda'''(\langle y_{k+c-1}, \ldots, y_k, s''_{k+c}\rangle, y_{k+c} \ldots y_{l-1})$$
$$= R(-k-c, \lambda'''(\varphi(s), \beta)).$$

We conclude that propagation of weakly decoding errors of $\mathcal{SIM}(M'', f)$ to M is bounded. □

Using Corollary 1.5.2, we have the following equivalent definition for linear finite automata.

Corollary 1.5.3. *Let M be a linear finite automaton. Then M is feedforward invertible with delay τ if and only if there exists a linear finite order semi-input-memory finite automaton which is a weak inverse with delay τ of M.*

1.6 Labelled Trees as States of Finite Automata

Let X and Y be two finite nonempty sets and τ a nonnegative integer. Define a kind of labelled trees with level τ as follows. Each vertex with level $\leqslant \tau$ emits $|X|$ arcs and each arc has a label of the form (x, y), where $x \in X$ and $y \in Y$. x and y are called the *input label* and the *output label* of the arc, respectively. Input labels of arcs with the same initial vertex are distinct from each other, they constitute the set X. Output labels of arcs with the same initial vertex are not restricted. We use $\mathcal{T}(X, Y, \tau)$ to denote the set of all such trees. For any vertex with level $i+1$, if the labels of arcs in the path from the root to the vertex are (x_0, y_0), (x_1, y_1), …, (x_i, y_i), $x_0 \ldots x_i$ is called the *input label sequence* of the path or of the vertex, and $y_0 \ldots y_i$ is called the *output label sequence* of the path or of the vertex.

Let T be a tree in $\mathcal{T}(X, Y, \tau)$. If for any two paths w_i of length $\tau+1$ in T, $i = 1, 2$, that the output label sequence of w_1 and of w_2 are the same implies that arcs of w_1 and of w_2 are joint, T is said to be *compatible*. Noticing that the initial vertex of w_i is the root, $i = 1, 2$, the condition that arcs of w_1 and of w_2 are joint is equivalent to the condition: the first arc of w_1 and of w_2 are the same. This condition is also equivalent to a condition: the first letter of the input label sequence of w_1 and of w_2 are the same.

Let T_1 and T_2 be two trees in $\mathcal{T}(X, Y, \tau)$. If for any path w_i of length $\tau+1$ in T_i, $i = 1, 2$, that the output label sequence of w_1 and of w_2 are the same implies that the first letter of the input label sequence of w_1 and of w_2 are the same, T_1 and T_2 are said to be *strongly compatible*.

For any $\mathcal{F} \subseteq \mathcal{T}(X, Y, \tau)$, if each tree in $\mathcal{F}$ is compatible, $\mathcal{F}$ is said to be *compatible*; if any two trees in $\mathcal{F}$ are strongly compatible, $\mathcal{F}$ is said to be *strongly compatible*.

For any T in $\mathcal{T}(X,Y,\tau)$, in the case of $\tau > 0$, we use $T_{\_}$ to denote the set of $|X|$ subtrees of T of which roots are the terminal vertices of arcs emitted from the root of T and arcs contain all arcs of T with level $\geqslant 1$. We use $T^{\text{-}}$ to denote the subtree of T which is obtained from T by deleting all vertices with level $\tau+1$ and all arcs with level τ. Clearly, $T_{\_} \subseteq \mathcal{T}(X,Y,\tau-1)$ and $T^{\text{-}} \in \mathcal{T}(X,Y,\tau-1)$. For any $\mathcal{F} \subseteq \mathcal{T}(X,Y,\tau)$, let $\mathcal{F}_{\_} = \cup_{T\in\mathcal{F}} T_{\_}$ and $\mathcal{F}^{\text{-}} = \{T^{\text{-}} \mid T \in \mathcal{F}\}$.

If $\tau = 0$ or $\mathcal{F}_{\_} \subseteq \mathcal{F}^{\text{-}}$, $\mathcal{F}$ is said to be *closed*. For any T_i in $\mathcal{T}(X,Y,\tau)$, $i = 1,2$, if $T_2^{\text{-}}$ and the subtree of T_1 in T_{1-}, of which the root is the terminal vertex of an arc emitted from the root of T_1 with input label x, are the same, T_2 is called an *x-successor* of T_1. Clearly, if $\mathcal{F} \subseteq \mathcal{T}(X,Y,\tau)$ is closed, then for any T_1 in $\mathcal{F}$ and any x in X, there exists T_2 in $\mathcal{F}$ such that T_2 is an x-successor of T_1.

Let $\mathcal{F}$ be a closed nonempty subset of $\mathcal{T}(X,Y,\tau)$, and ν a single-valued mapping from $\mathcal{F}$ to the set of positive integers. We construct a finite automaton $M = \langle X,Y,S,\delta,\lambda\rangle$, where

$$S = \{\langle T,i\rangle \mid T \in \mathcal{F}, i = 1,\ldots,\nu(T)\},$$

and δ and λ are defined as follows. For any T in $\mathcal{F}$ and any x in X, define

$$\delta(\langle T,i\rangle,x) = \langle T',j\rangle,$$

$$\lambda(\langle T,i\rangle,x) = y,$$

where T' is an x-successor of T, j is an integer with $1 \leqslant j \leqslant \nu(T')$, and (x,y) is a label of an arc emitted from the root of T. Notice that given T and x, from the construction of T, the value of y is unique, and from the closedness of $\mathcal{F}$, values of T' and j are existent but not necessary to be unique. Since M is determined by $\mathcal{F}$, ν and δ, we use $M(\mathcal{F},\nu,\delta)$ to denote the finite automaton M.

For any finite automaton $M = \langle X,Y,S,\delta,\lambda\rangle$ and any state s of M, construct a labelled tree with level τ, denoted by $T_\tau^M(s)$, as follows. The root of $T_\tau^M(s)$ is temporarily labelled by s. For each vertex v with level $\leqslant \tau$ of $T_\tau^M(s)$ and any x in X, an arc with label $(x,\lambda(s',x))$ is emitted from v, and we use $\delta(s',x)$ to label the terminal vertex of the arc temporarily, where s' is the label of v. Finally, deleting all labels of vertices, we obtain the tree $T_\tau^M(s)$.

Clearly, $T_\tau^M(s) \in \mathcal{T}(X,Y,\tau)$. It is easy to see that for any s in S and any x in X, $T_\tau^M(\delta(s,x))$ is an x-successor of $T_\tau^M(s)$. And for any path of length $\tau+1$ of $T_\tau^M(s)$, if the input label sequence and the output label sequence of the path are $x_0\ldots x_\tau$ and $y_0\ldots y_\tau$, respectively, then we have $\lambda(s,x_0\ldots x_\tau) = y_0\ldots y_\tau$.

From the construction of $M(\mathcal{F},\nu,\delta)$, we have the following lemma.

Lemma 1.6.1. *For any state $\langle T, i\rangle$ of $M(\mathcal{F}, \nu, \delta)$, $T_\tau^{M(\mathcal{F},\nu,\delta)}(\langle T, i\rangle)$ and T are the same.*

Lemma 1.6.2. *Let $M = \langle X, Y, S, \delta, \lambda\rangle$ be a finite automaton and $\mathcal{F} = \{T_\tau^M(s) \mid s \in S\}$. Then M is weakly invertible with delay τ if and only if $\mathcal{F}$ is compatible, and M is invertible with delay τ if and only if $\mathcal{F}$ is strongly compatible.*

Proof. Suppose that M is weakly invertible with delay τ. For any s in S and any path w_i of length $\tau + 1$ in $T_\tau^M(s)$, $i = 1, 2$, suppose that the output label sequence of w_1 and of w_2 are the same, say $y_0 \dots y_\tau$. Let $x_0 \dots x_\tau$ and $x'_0 \dots x'_\tau$ be the input label sequence of w_1 and of w_2, respectively. Then we have $\lambda(s, x_0 \dots x_\tau) = y_0 \dots y_\tau$ and $\lambda(s, x'_0 \dots x'_\tau) = y_0 \dots y_\tau$. It follows that $x_0 = x'_0$. Thus $T_\tau^M(s)$ is compatible. It follows that $\mathcal{F}$ is compatible.

Conversely, suppose that $\mathcal{F}$ is compatible. Let $\lambda(s, x_0 \dots x_\tau) = \lambda(s, x'_0 \dots x'_\tau)$ for s in S and $x_i, x'_i \in X$, $i = 0, 1, \dots, \tau$. From the construction of $T_\tau^M(s)$, there exist two paths w_1 and w_2 of length $\tau+1$ in $T_\tau^M(s)$ such that the input label sequence of w_1 and of w_2 are $x_0 \dots x_\tau$ and $x'_0 \dots x'_\tau$, respectively, and the output label sequence of w_1 and of w_2 are $\lambda(s, x_0 \dots x_\tau)$ and $\lambda(s, x'_0 \dots x'_\tau)$, respectively. Since $T_\tau^M(s)$ is compatible and $\lambda(s, x_0 \dots x_\tau) = \lambda(s, x'_0 \dots x'_\tau)$, we have $x_0 = x'_0$. Thus M is weakly invertible with delay τ.

Suppose that M is invertible with delay τ. For any s_1 and s_2 in S and any path w_i of length $\tau + 1$ in $T_\tau^M(s_i)$, $i = 1, 2$, suppose that the output label sequence of w_1 and of w_2 are the same, say $y_0 \dots y_\tau$. Let $x_0 \dots x_\tau$ and $x'_0 \dots x'_\tau$ be the input label sequence of w_1 and of w_2, respectively. Then we have $\lambda(s_1, x_0 \dots x_\tau) = y_0 \dots y_\tau$ and $\lambda(s_2, x'_0 \dots x'_\tau) = y_0 \dots y_\tau$. It follows that $x_0 = x'_0$. Thus $T_\tau^M(s_1)$ and $T_\tau^M(s_2)$ are strongly compatible. And it follows that $\mathcal{F}$ is strongly compatible.

Conversely, suppose that $\mathcal{F}$ is strongly compatible. Let $\lambda(s_1, x_0 \dots x_\tau) = \lambda(s_2, x'_0 \dots x'_\tau)$ for s_1, s_2 in S and x_i, $x'_i \in X$, $i = 0, 1, \dots, \tau$. From the construction of $T_\tau^M(s_i)$, there exist two paths w_1 in $T_\tau^M(s_1)$ and w_2 in $T_\tau^M(s_2)$ of length $\tau+1$ such that the input label sequence of w_1 and of w_2 are $x_0 \dots x_\tau$ and $x'_0 \dots x'_\tau$, respectively, and the output label sequence of w_1 and of w_2 are $\lambda(s_1, x_0 \dots x_\tau)$ and $\lambda(s_2, x'_0 \dots x'_\tau)$, respectively. Since $T_\tau^M(s_1)$ and $T_\tau^M(s_2)$ are strongly compatible and $\lambda(s_1, x_0 \dots x_\tau) = \lambda(s_2, x'_0 \dots x'_\tau)$, we have $x_0 = x'_0$. Thus M is invertible with delay τ. □

Theorem 1.6.1. *For any finite automaton $M(\mathcal{F}, \nu, \delta)$, $M(\mathcal{F}, \nu, \delta)$ is weakly invertible with delay τ if and only if $\mathcal{F}$ is compatible, and $M(\mathcal{F}, \nu, \delta)$ is invertible with delay τ if and only if $\mathcal{F}$ is strongly compatible.*

Proof. From Lemma 1.6.1, $\mathcal{F} = \{T_\tau^{M(\mathcal{F},\nu,\delta)}(s') |\ s' \text{ is a state of } M(\mathcal{F}, \nu, \delta)\}$. Using Lemma 1.6.2, the theorem follows. □

Lemma 1.6.3. *Let $M = \langle X, Y, S, \delta, \lambda\rangle$ be a finite automaton. Then there exists a finite automaton $M(\mathcal{F}, \nu, \delta')$ such that M and $M(\mathcal{F}, \nu, \delta')$ are isomorphic.*

Proof. Take $\mathcal{F} = \{T_\tau^M(s) \mid s \in S\}$. Partition S into blocks so that s_1 and s_2 belong to the same block if and only if $T_\tau^M(s_1) = T_\tau^M(s_2)$. Let $S_1, \ldots, S_r$ be the blocks of the partition. Define $\nu(T_\tau^M(s)) = |S_i|$ for any s in S_i. Let

$$S' = \{\langle T_\tau^M(s), i\rangle \mid s \in S, i = 1, \ldots, \nu(T_\tau^M(s))\}.$$

For any j, $1 \leqslant j \leqslant r$, let $s_1^j, \ldots, s_{|S_j|}^j$ be a permutation of elements in S_j. Define a single-valued mapping η from S to S':

$$\eta(s_k^j) = \langle T_\tau^M(s_k^j), k\rangle, \quad j = 1, \ldots, r, \ k = 1, \ldots, |S_j|.$$

It is easy to verify that η is bijective. Define

$$\delta'(s', x) = \eta(\delta(\eta^{-1}(s'), x)), \quad s' \in S', \ x \in X.$$

It is easy to show that η is an isomorphism from M to $M(\mathcal{F}, \nu, \delta')$. Therefore, M and $M(\mathcal{F}, \nu, \delta')$ are isomorphic. □

Theorem 1.6.2. *Let $M = \langle X, Y, S, \delta, \lambda\rangle$ be a finite automaton. Then there exists a finite automaton $M(\mathcal{F}, \nu, \delta')$ such that M and $M(\mathcal{F}, \nu, \delta')$ are isomorphic and M is weakly invertible (respectively invertible) with delay τ if and only if $\mathcal{F}$ is compatible (respectively strongly compatible).*

Proof. From Lemma 1.6.3, there exists a finite automaton $M(\mathcal{F}, \nu, \delta')$ such that M and $M(\mathcal{F}, \nu, \delta')$ are isomorphic. Thus M is weakly invertible with delay τ if and only if $M(\mathcal{F}, \nu, \delta')$ is weakly invertible with delay τ, and M is invertible with delay τ if and only if $M(\mathcal{F}, \nu, \delta')$ is invertible with delay τ. From Theorem 1.6.1, the theorem follows. □

In Sect. 6.5 of Chap. 6, we will deal with trees with arc label and vertex label. Let X, Y and S' be three finite nonempty sets and τ a nonnegative integer. Let T be a tree in $\mathcal{T}(X, Y, \tau)$. We assign an element in S' to each vertex of T, the element is referred to as the *label* of the vertex. We use $\mathcal{T}'(X, Y, S', \tau)$ to denote the set of all such trees with arc label and vertex label.

The concept of closedness for $\mathcal{T}(X, Y, \tau)$ may be naturally generalized to $\mathcal{T}'(X, Y, S', \tau)$. For any T in $\mathcal{T}'(X, Y, S', \tau)$, in the case of $\tau > 0$, we use $T_{\_}$ to denote the set of $|X|$ subtrees of T of which roots are the terminal vertices of arcs emitted from the root of T and arcs contain all arcs of T with level $\geqslant$ 1. We use $T^{\text{-}}$ to denote the subtree of T which is obtained from T by deleting all vertices with level $\tau + 1$ and all arcs with level τ. Clearly, $T_{\_} \subseteq$

$\mathcal{T}'(X, Y, S', \tau - 1)$ and $T^- \in \mathcal{T}'(X, Y, S', \tau - 1)$. For any $\mathcal{F} \subseteq \mathcal{T}'(X, Y, S', \tau)$, let $\mathcal{F}_- = \cup_{T \in \mathcal{F}} T_-$ and $\mathcal{F}^- = \{T^- \mid T \in \mathcal{F}\}$.

For any $\mathcal{F} \subseteq \mathcal{T}'(X, Y, S', \tau)$, if $\tau = 0$ or $\mathcal{F}_- \subseteq \mathcal{F}^-$, $\mathcal{F}$ is said to be *closed*. For any T_i in $\mathcal{T}'(X, Y, S', \tau)$, $i = 1, 2$, if T_2^- and the subtree of T_1 in T_{1-}, of which the root is the terminal vertex of an arc emitted from the root of T_1 with input label x, are the same, T_2 is called an *x-successor* of T_1.

Let $\mathcal{F}$ be a closed nonempty subset of $\mathcal{T}'(X, Y, S', \tau)$. Let ν be a single-valued mapping from $\mathcal{F}$ to the set of positive integers. Similar to the case of $\mathcal{T}(X, Y, \tau)$, we construct a finite automaton $M = \langle X, Y, S, \delta, \lambda \rangle$, where

$$S = \{\langle T, i \rangle \mid T \in \mathcal{F}, i = 1, \ldots, \nu(T)\},$$

and δ and λ are defined as follows. For any T in $\mathcal{F}$ and any x in X, define

$$\delta(\langle T, i \rangle, x) = \langle T', j \rangle,$$
$$\lambda(\langle T, i \rangle, x) = y,$$

where T' is an x-successor of T, j is an integer with $1 \leqslant j \leqslant \nu(T')$, and (x, y) is a label of an arc emitted from the root of T. Notice that given T and x, from the construction of T, the value of y is unique, and from the closedness of $\mathcal{F}$, values of T' and j are existent but not necessary to be unique. Since M is determined by $\mathcal{F}$, ν and δ, we still use $M(\mathcal{F}, \nu, \delta)$ to denote the finite automaton M.

Historical Notes

The original development of finite automata is found in [62, 58, 78]. In [78], the output function of a finite automaton is independent of the input. In this book we adopt the definition in [73]. In [62], finite automata are regarded as recognizers and their equivalence with regular sets is first proven. The compound finite automaton $C'(M, M')$ of finite automata M and M' is introduced in [112] for application to cryptography. References [59, 40, 24, 25, 47] deal with linear finite automata. Section 1.3 is in part based on [25], Theorem 1.3.5 is due to [35], and the material of z-transformation is taken from [98].

Finite order information lossless finite automata, that is, weakly invertible finite automata with finite delay in this book, are first defined in [60]. Finite order invertible finite automata are first defined in [96]. Most of Sect. 1.4 are taken from Sect. 2.1 of [98], where the decision method is based on [36]. In [71], feedforward invertible linear finite automata are defined by means of transfer function matrix. Section 1.5 is based on [99], in which semi-input-memory finite automata and feedforward invertible finite automata in general case are first defined. Section 1.6 is based on Sects. 2.7 and 2.8 of [98].

2. Mutual Invertibility and Search

Renji Tao

Institute of Software, Chinese Academy of Sciences
Beijing 100080, China trj@ios.ac.cn

Summary.

In this chapter, we first discuss the weights of output sets and input sets of finite automata and prove that for any weakly invertible finite automaton and its states with minimal output weight, the distribution of input sets is uniform. Then, using a result on output set, mutual invertibility of finite automata is proven. Finally, for a kind of compound finite automata, we give weights of output sets and input sets explicitly, and a characterization of their input-trees; this leads to an evaluation of search amounts of an exhausting search algorithm in average case and in worse case, and successful probabilities of a stochastic search algorithm.

The search problem is proposed in cryptanalysis for a public key cryptosystem based on finite automata in Chap. 9.

Key words: *minimal output weight, input tree, exhausting search, stochastic search, mutual invertibility*

According to the definition of weakly invertible finite automata with delay τ, from the initial state and the output sequence we can uniquely determine the input sequence except the last τ letters by means of exhausting search. The exhausting search method is effective, but not feasible in general. How to evaluate the complexity of the search amount? In parallel, for the stochastic search, what is the successful probability? These problems are proposed in cryptanalysis for a public key cryptosystem based on finite automata (see Subsect. 9.5.4).

In this chapter, we deal with these problems by studying the weights of output sets and input sets of finite automata. It is proved that for any weakly invertible finite automaton and its states with minimal output weight, the distribution of input sets is uniform. Then for a kind of alternatively compound finite automata of weakly invertible finite automata with delay 0

and "delayers", we give weights of output sets and input sets explicitly, and a characterization of their input-trees. This leads to an evaluation of search amounts of an exhausting search algorithm in average case and in worse case, and successful probabilities of a stochastic search algorithm.

In addition, mutual invertibility of finite automata is also proven using a result on output set.

2.1 Minimal Output Weight and Input Set

Let $M = \langle X, Y, S, \delta, \lambda \rangle$ be a finite automaton. For any $s \in S$ and any positive integer r, the set

$$\{\lambda(s, x_0 \dots x_{r-1}) \mid x_0, \dots, x_{r-1} \in X\}$$

is called the *r-output set* of s in M, denoted by $W^M_{r,s}$. $|W^M_{r,s}|$, the number of elements in $W^M_{r,s}$, is called the *r-output weight* of s in M. And $\min_{s \in S} |W^M_{r,s}|$ is called the *minimal r-output weight* in M, denoted by $w_{r,M}$. In the case of $r > 1$, for any $s \in S$ and $x \in X$, $\lambda(s, x) W^M_{r-1,\delta(s,x)}$ is called the *x-branch* of $W^M_{r,s}$. If for any two states s and s' of M there exists $\alpha \in X^*$ such that $s' = \delta(s, \alpha)$, M is said to be *strongly connected*.

Theorem 2.1.1. *Let $M = \langle X, Y, S, \delta, \lambda \rangle$ be weakly invertible with delay $r - 1$ and $|X| = |Y| = q$.*

(a) *$q | w_{r,M}$.*

(b) *For any $s \in S$, if $w_{r,M} = |W^M_{r,s}|$, then the number of elements in any x-branch of $W^M_{r,s}$ is $w_{r,M}/q$.*

(c) *For any $s \in S$, if $w_{r,M} = |W^M_{r,s}|$ and s' is a successor of s, then $w_{r,M} = |W^M_{r,s'}|$.*

(d) *For any $s \in S$, if $w_{r,M} = |W^M_{r,s}|$, s' is a successor of s, and $\beta_{r-1} \in W^M_{r-1,s'}$, then $\beta_{r-1} y \in W^M_{r,s'}$ for each $y \in Y$.*

(e) *For any $s \in S$, if $w_{r,M} = |W^M_{r,s}|$, then $W^M_{r+n,s} = W^M_{r,s} Y^n$ for any $n \geqslant 0$.*

(f) *For any $s \in S$, if $w_{r,M} = |W^M_{r,s}|$ and s' is a successor of s, then $W^M_{r-1+n,s'} = W^M_{r-1,s'} Y^n$ for any $n \geqslant 0$.*

(g) *If M is strongly connected, then $w_{r,M} = |W^M_{r,s}|$ holds for any $s \in S$.*

Proof. (b) Assume that $w_{r,M} = |W^M_{r,s}|$. First of all, since M is weakly invertible with delay $r - 1$, it is easy to see that for any different elements x and x' in X, the x-branch of $W^M_{r,s}$ and the x'-branch of $W^M_{r,s}$ are disjoint. Thus all distinct x-branches of $W^M_{r,s}$ constitute a partition of $W^M_{r,s}$. It follows that

$$w_{r,M} = |W^M_{r,s}| = \sum_{x \in X} |W^M_{r-1,\delta(s,x)}|. \tag{2.1}$$

We prove $|W^M_{r-1,\delta(s,x)}| \geqslant w_{r,M}/q$ for any $x \in X$ by reduction to absurdity. Suppose to the contrary that there is an element $x \in X$ satisfying $|W^M_{r-1,\delta(s,x)}| < w_{r,M}/q$. It is evident that $|W^M_{r,\delta(s,x)}| \leqslant q|W^M_{r-1,\delta(s,x)}|$. Thus $|W^M_{r,\delta(s,x)}| < q(w_{r,M}/q) = w_{r,M}$. This contradicts the definition of the minimal r-output weight. Therefore, $|W^M_{r-1,\delta(s,x)}| \geqslant w_{r,M}/q$.

Next, we prove $|W^M_{r-1,\delta(s,x)}| = w_{r,M}/q$ for any $x \in X$. Since $|W^M_{r-1,\delta(s,x)}| \geqslant w_{r,M}/q$ for any $x \in X$, it is sufficient to prove that $|W^M_{r-1,\delta(s,x)}| \not> w_{r,M}/q$ for any $x \in X$. We prove this fact by reduction to absurdity. Suppose to the contrary that there is an element $x' \in X$ satisfying $|W^M_{r-1,\delta(s,x')}| > w_{r,M}/q$. From (2.1), we have

$$w_{r,M} = \sum_{x\in X} |W^M_{r-1,\delta(s,x)}| > \sum_{x\in X} w_{r,M}/q = q(w_{r,M}/q) = w_{r,M}.$$

Thus $w_{r,M} > w_{r,M}$; this is a contradiction. Therefore, $|W^M_{r-1,\delta(s,x)}| \not> w_{r,M}/q$ holds for any $x \in X$.

(a) Let s be a state of M and $|W^M_{r,s}| = w_{r,M}$. From (b), $w_{r,M}/q$ is the number in elements of x-branch of $W^M_{r,s}$. Since the number of elements in x-branch of $W^M_{r,s}$ is an integer, we have $q|w_{r,M}$.

(c) Let s' be a successor of s and $w_{r,M} = |W^M_{r,s}|$. Then we have $s' = \delta(s, x)$ for some $x \in X$. From (b), the number of elements in x-branch of $W^M_{r,s}$ is $w_{r,M}/q$. This yields that $|W^M_{r-1,s'}| = w_{r,M}/q$. Thus $|W^M_{r,s'}| \leqslant q|W^M_{r-1,s'}| = q(w_{r,M}/q) = w_{r,M}$, that is, $|W^M_{r,s'}| \leqslant w_{r,M}$. On the other hand, from the definition of the minimal r-output weight, we have $|W^M_{r,s'}| \geqslant w_{r,M}$. Thus $|W^M_{r,s'}| = w_{r,M}$.

(d) Let s' be a successor of s, $w_{r,M} = |W^M_{r,s}|$, and $\beta_{r-1} \in W^M_{r-1,s'}$. From (c), we obtain $w_{r,M} = |W^M_{r,s'}|$. Since s' is a successor of s, we have $s' = \delta(s, x)$ for some $x \in X$. Then the x-branch of $W^M_{r,s}$ is $\lambda(s, x)W^M_{r-1,s'}$. From (b), we have $|W^M_{r-1,s'}| = w_{r,M}/q$. This yields that $|W^M_{r,s'}| < q(w_{r,M}/q)$ in case of $\beta_{r-1}y \notin W^M_{r,s'}$ for some $y \in Y$. Since $|W^M_{r,s'}| = w_{r,M} = q(w_{r,M}/q)$, we have $\beta_{r-1}y \in W^M_{r,s'}$ for each $y \in Y$.

(e) Let $w_{r,M} = |W^M_{r,s}|$. We prove by induction on n that $W^M_{r+n,s} = W^M_{r,s}Y^n$ for any $n \geqslant 0$. *Basis* : $n = 0$. It is trivial. *Induction step* : Suppose that $W^M_{r+n,s} = W^M_{r,s}Y^n$. Let $y_0 \dots y_{r+n}$ be an arbitrary element in $W^M_{r,s}Y^{n+1}$. Clearly, $y_0 \dots y_{r+n-1} \in W^M_{r,s}Y^n$. From the induction hypothesis, it follows that $y_0 \dots y_{r+n-1} \in W^M_{r+n,s}$. Thus there are $x_0, \dots, x_{r+n-1} \in X$ such that $y_0 \dots y_{r+n-1} = \lambda(s, x_0 \dots x_{r+n-1})$. Denote $s_{i+1} = \delta(s, x_0 \dots x_i)$ for $i = 0, 1, \dots, r + n - 1$. From $w_{r,M} = |W^M_{r,s}|$, using (c), we have $w_{r,M} = |W^M_{r,s_i}|$, $i = 1, \dots, r + n$. Since $y_{n+1} \dots y_{r+n-1} \in W^M_{r-1,s_{n+1}}$ and $w_{r,M} = |W^M_{r,s_{n+1}}|$, from (d), $y_{n+1} \dots y_{r+n} \in W^M_{r,s_{n+1}}$. It follows that there are $x'_{n+1}, \dots, x'_{r+n} \in X$ such that $y_{n+1} \dots y_{r+n} = \lambda(s_{n+1}, x'_{n+1} \dots x'_{r+n})$. Thus

$$y_0 \dots y_{r+n} = \lambda(s, x_0 \dots x_n)\lambda(s_{n+1}, x'_{n+1} \dots x'_{r+n})$$
$$= \lambda(s, x_0 \dots x_n x'_{n+1} \dots x'_{r+n}).$$

This yields $y_0 \dots y_{r+n} \in W^M_{r+n+1,s}$. Thus $W^M_{r,s}Y^{n+1} \subseteq W^M_{r+n+1,s}$. On the other hand, it is evident that $W^M_{r,s}Y^{n+1} \supseteq W^M_{r+n+1,s}$. We conclude that $W^M_{r,s}Y^{n+1} = W^M_{r+n+1,s}$.

(f) Let s' be a successor of s and $w_{r,M} = |W^M_{r,s}|$. To prove $W^M_{r-1+n,s'} \supseteq W^M_{r-1,s'}Y^n$, let $y_1 \dots y_{r-1+n}$ be an arbitrary element in $W^M_{r-1,s'}Y^n$. It follows that $y_1 \dots y_{r-1} \in W^M_{r-1,s'}$. Thus there exist $x_1, \dots, x_{r-1} \in X$ such that $y_1 \dots y_{r-1} = \lambda(s', x_1 \dots x_{r-1})$. Since s' is a successor of s, there exists $x_0 \in X$ such that $s' = \delta(s, x_0)$. Denoting $y_0 = \lambda(s, x_0)$, then $y_0 y_1 \dots y_{r-1} = \lambda(s, x_0 x_1 \dots x_{r-1})$. It follows that $y_0 y_1 \dots y_{r-1} \in W^M_{r,s}$. From (e), $y_0\ y_1\ \dots\ y_{r-1+n} \in W^M_{r+n,s}$. Thus there exist $x'_0, x'_1, \dots, x'_{r-1+n} \in X$ such that $y_0 y_1 \dots y_{r-1+n} = \lambda(s, x'_0 x'_1 \dots x'_{r-1+n})$. This yields $y_0 \dots y_{r-1} = \lambda(s,\ x'_0\ \dots\ x'_{r-1})$. Since $y_0 \dots y_{r-1} = \lambda(s, x_0 \dots x_{r-1})$ and M is weakly invertible with delay $r-1$, we obtain $x'_0 = x_0$. It follows immediately that $y_1 \dots y_{r-1+n} = \lambda(s', x'_1 \dots x'_{r-1+n})$. That is, $y_1 \dots y_{r-1+n} \in W^M_{r-1+n,s'}$. We conclude that $W^M_{r-1+n,s'} \supseteq W^M_{r-1,s'}Y^n$. Clearly, $W^M_{r-1+n,s'} \subseteq W^M_{r-1,s'}Y^n$. Thus $W^M_{r-1+n,s'} = W^M_{r-1,s'}Y^n$.

(g) Assume that M is strongly connected. Let $\bar{s}$ in S satisfy $w_{r,M} = |W^M_{r,\bar{s}}|$. For any $s \in S$, since M is strongly connected, there are x_0, …, x_n in X such that $s = \delta(\bar{s}, x_0 \dots x_n)$. Denote $s_{i+1} = \delta(\bar{s}, x_0 \dots x_i)$, for $0 \leqslant i \leqslant n$. From (c), we have $w_{r,M} = |W^M_{r,s_i}|$, $i = 1, \dots, n+1$. From $s_{n+1} = s$, it follows that $w_{r,M} = |W^M_{r,s}|$. □

For any $\beta \in Y^r$, $I^M_{\beta,s} = \{\alpha \mid \alpha \in X^*, \lambda(s,\alpha) = \beta\}$ is called the *β-input set* of s in M.

Let

$$w'_{r,M} = \min\{\, |I^M_{\beta,s}| \,:\, s \in S, |W^M_{r,s}| = w_{r,M}, \beta \in W^M_{r,s}\}.$$

$w'_{r,M}$ is called the *minimal r-input weight* in M.

Lemma 2.1.1. *Let $M = \langle X, Y, S, \delta, \lambda\rangle$ be weakly invertible with delay $r-1$ and $|X| = |Y|$. Let $w_{r,M} = |W^M_{r,s}|$. For any $x_0, \dots, x_{r-1} \in X$, if $|I^M_{y_0 \dots y_{r-1},s}| = w'_{r,M}$, then for any $y \in Y$, $|I^M_{y_1 \dots y_{r-1} y, \delta(s,x_0)}| = w'_{r,M}$, where $y_0 \dots y_{r-1} = \lambda(s, x_0 \dots x_{r-1})$.*

Proof. Assume that $|I^M_{y_0 \dots y_{r-1},s}| = w'_{r,M}$. Let $s' = \delta(s, x_0)$. Since M is weakly invertible with delay $r-1$, x_0 is uniquely determined by s and $y_0 \dots y_{r-1}$. Thus $|I^M_{y_0 y_1 \dots y_{r-1},s}| = |I^M_{y_1 \dots y_{r-1},s'}|$. Since $|I^M_{y_0 \dots y_{r-1},s}| = w'_{r,M}$, we have $|I^M_{y_1 \dots y_{r-1},s'}| = w'_{r,M}$. Denoting $|X| = q$, it follows that

$$\sum_{y \in Y} |I^M_{y_1 \dots y_{r-1} y, s'}| = q w'_{r,M}. \tag{2.2}$$

We prove $|I^M_{y_1\dots y_{r-1}y,s'}| = w'_{r,M}$ for each $y \in Y$. From Theorem 2.1.1 (d), for any $y \in Y$, $y_1 \dots y_{r-1}y \in W^M_{r,s'}$. From $w_{r,M} = |W^M_{r,s}|$, using Theorem 2.1.1 (c), we have $w_{r,M} = |W^M_{r,s'}|$. From the definition of $w'_{r,M}$, we then obtain $|I^M_{y_1\dots y_{r-1}y,s'}| \geqslant w'_{r,M}$ for each $y \in Y$. We prove $|I^M_{y_1\dots y_{r-1}y,s'}| \leqslant w'_{r,M}$ for each $y \in Y$ by reduction to absurdity. Suppose to the contrary that $|I^M_{y_1\dots y_{r-1}y',s'}| > w'_{r,M}$ for some $y' \in Y$. From $|I^M_{y_1\dots y_{r-1}y,s'}| \geqslant w'_{r,M}$ for each $y \in Y$, it follows that $\sum_{y\in Y} |I^M_{y_1\dots y_{r-1}y,s'}| > qw'_{r,M}$, this contradicts (2.2). We conclude that $|I^M_{y_1\dots y_{r-1}y,s'}| \leqslant w'_{r,M}$ for each $y \in Y$. Therefore, $|I^M_{y_1\dots y_{r-1}y,s'}| = w'_{r,M}$ for each $y \in Y$. □

Lemma 2.1.2. *Let $M = \langle X, Y, S, \delta, \lambda\rangle$ be weakly invertible with delay $r-1$ and $|X| = |Y|$. Let $w_{r,M} = |W^M_{r,s}|$. For any $x_0, \dots, x_{r-1} \in X$, if $|I^M_{y_0\dots y_{r-1},s}| = w'_{r,M}$, then for any $\beta \in W^M_{r,\delta(s,x_0\dots x_{r-1})}$, $|I^M_{\beta,\delta(s,x_0\dots x_{r-1})}| = w'_{r,M}$, where $y_0 \dots y_{r-1} = \lambda(s, x_0 \dots x_{r-1})$.*

Proof. Applying repeatedly Theorem 2.1.1 (c) r times, we obtain

$$|W^M_{r,\delta(s,x_0\dots x_i)}| = w_{r,M}$$

for any i, $0 \leqslant i \leqslant r-1$. Let $\beta = y_r \dots y_{2r-1} \in W^M_{r,\delta(s,x_0\dots x_{r-1})}$. Then there are $x_r, \dots, x_{2r-1} \in X$ such that $\beta = \lambda(\delta(s, x_0 \dots x_{r-1}), x_r \dots x_{2r-1})$. It follows that $\lambda(s, x_0 \dots x_{2r-1}) = y_0 \dots y_{2r-1}$. Applying repeatedly Lemma 2.1.1 r times, we obtain $|I^M_{y_{i+1}\dots y_{r+i},\delta(s,x_0\dots x_i)}| = w'_{r,M}$ for any i, $0 \leqslant i \leqslant r-1$. The case $i = r-1$ gives $|I^M_{\beta,\delta(s,x_0\dots x_{r-1})}| = w'_{r,M}$. □

Lemma 2.1.3. *Let $M = \langle X, Y, S, \delta, \lambda\rangle$ be weakly invertible with delay $r-1$ and $|X| = |Y| = q$. Then $w'_{r,M} = q^r/w_{r,M}$.*

Proof. Let s in S and $x_0, \dots, x_{r-1}$ in X satisfy $w_{r,M} = |W^M_{r,s}|$ and $|I^M_{\lambda(s,x_0\dots x_{r-1}),s}| = w'_{r,M}$. Since

$$\bigcup_{\beta\in W^M_{r,\delta(s,x_0\dots x_{r-1})}} I^M_{\beta,\delta(s,x_0\dots x_{r-1})} = X^r,$$

we have

$$\sum_{\beta\in W^M_{r,\delta(s,x_0\dots x_{r-1})}} |I^M_{\beta,\delta(s,x_0\dots x_{r-1})}| = q^r.$$

From Lemma 2.1.2, $|I^M_{\beta,\delta(s,x_0\dots x_{r-1})}| = w'_{r,M}$ for any $\beta \in W^M_{r,\delta(s,x_0\dots x_{r-1})}$. It follows that

$$\begin{aligned} |W^M_{r,\delta(s,x_0\dots x_{r-1})}| w'_{r,M} &= \sum_{\beta\in W^M_{r,\delta(s,x_0\dots x_{r-1})}} w'_{r,M} \\ &= \sum_{\beta\in W^M_{r,\delta(s,x_0\dots x_{r-1})}} |I^M_{\beta,\delta(s,x_0\dots x_{r-1})}| = q^r. \end{aligned}$$

Since $|W^M_{r,s}| = w_{r,M}$, from Theorem 2.1.1 (c), we have $|W^M_{r,\delta(s,x_0\dots x_{r-1})}| = w_{r,M}$. It follows immediately that $w_{r,M}w'_{r,M} = q^r$. Therefore, $w'_{r,M} = q^r/w_{r,M}$. □

Let

$$w''_{r,M} = \max\{\, |I^M_{\beta,s}| \,:\, s \in S, |W^M_{r,s}| = w_{r,M}, \beta \in W^M_{r,s}\}.$$

$w''_{r,M}$ is called the *maximal r-input weight* in M.

Lemma 2.1.4. *Let $M = \langle X, Y, S, \delta, \lambda\rangle$ be weakly invertible with delay $r-1$ and $|X| = |Y|$. Let $w_{r,M} = |W^M_{r,s}|$. For any $x_0, \dots, x_{r-1} \in X$, if $|I^M_{y_0\dots y_{r-1},s}| = w''_{r,M}$, then for any $y \in Y$, $|I^M_{y_1\dots y_{r-1}y,\delta(s,x_0)}| = w''_{r,M}$, where $y_0 \dots y_{r-1} = \lambda(s, x_0 \dots x_{r-1})$.*

Proof. The proof of this lemma is similar to Lemma 2.1.1 but replacing $w'_{r,M}$, $\geqslant, \leqslant$ and $>$ by $w''_{r,M}$, $\leqslant, \geqslant$ and $<$, respectively. □

Lemma 2.1.5. *Let $M = \langle X, Y, S, \delta, \lambda\rangle$ be weakly invertible with delay $r-1$ and $|X| = |Y|$. Let $w_{r,M} = |W^M_{r,s}|$. For any $x_0, \dots, x_{r-1} \in X$, if $|I^M_{y_0\dots y_{r-1},s}| = w''_{r,M}$, then for any $\beta \in W^M_{r,\delta(s,x_0\dots x_{r-1})}$, $|I^M_{\beta,\delta(s,x_0\dots x_{r-1})}| = w''_{r,M}$, where $y_0 \dots y_{r-1} = \lambda(s, x_0 \dots x_{r-1})$.*

Proof. The proof of this lemma is similar to Lemma 2.1.2 but replacing Lemma 2.1.1 and $w'_{r,M}$ by Lemma 2.1.4 and $w''_{r,M}$, respectively. □

Lemma 2.1.6. *Let $M = \langle X, Y, S, \delta, \lambda\rangle$ be weakly invertible with delay $r-1$ and $|X| = |Y| = q$. Then $w''_{r,M} = q^r/w_{r,M}$.*

Proof. The proof of this lemma is similar to Lemma 2.1.3 but replacing Lemma 2.1.2 and $w'_{r,M}$ by Lemma 2.1.5 and $w''_{r,M}$, respectively. □

Theorem 2.1.2. *Let $M = \langle X, Y, S, \delta, \lambda\rangle$ be weakly invertible with delay $r-1$ and $|X| = |Y| = q$. Then $w_{r,M} \mid q^r$ and for any $s \in S$ and any $\beta \in W^M_{r,s}$, if $w_{r,M} = |W^M_{r,s}|$, then $|I^M_{\beta,s}| = q^r/w_{r,M}$.*

Proof. From Lemma 2.1.3, $w'_{r,M} = q^r/w_{r,M}$. Since $w'_{r,M}$ is an integer, we have $w_{r,M} \mid q^r$.

Let $s \in S$, $\beta \in W^M_{r,s}$ and $w_{r,M} = |W^M_{r,s}|$. By definitions of $w'_{r,M}$ and $w''_{r,M}$, $w'_{r,M} \leqslant |I^M_{\beta,s}| \leqslant w''_{r,M}$. From Lemma 2.1.3 and Lemma 2.1.6, we have $w'_{r,M} = q^r/w_{r,M} = w''_{r,M}$. It follows that $w'_{r,M} = |I^M_{\beta,s}| = w''_{r,M}$. Therefore, $|I^M_{\beta,s}| = q^r/w_{r,M}$. □

Corollary 2.1.1. *Let $M = \langle X, Y, S, \delta, \lambda\rangle$ be weakly invertible with delay $r-1$ and $|X| = |Y| = q$. Let $w_{r,M} = |W^M_{r,s}|$. Then for any $x_0, \dots, x_{r-1} \in X$ and any $y \in Y$, $|I^M_{y_1\dots y_{r-1}y,\delta(s,x_0)}| = q^r/w_{r,M}$, where $y_0 \dots y_{r-1} = \lambda(s, x_0 \dots x_{r-1})$.*

Proof. From Theorem 2.1.2 and Lemmas 2.1.3, $|I^M_{y_0\ldots y_{r-1},s}| = q^r/w_{r,M} = w'_{r,M}$. Using Lemmas 2.1.1 and 2.1.3, $|I^M_{y_1\ldots y_{r-1}y,\delta(s,x_0)}| = w'_{r,M} = q^r/\omega_{r,M}$. □

Theorem 2.1.3. *Let $M = \langle X, Y, S, \delta, \lambda\rangle$ be weakly invertible with delay τ and $|X| = |Y| = q$.*

(a) *$w_{\tau+1,M} = qw_{\tau,M}$ and $w_{\tau,M}|q^\tau$.*

(b) *For any s in S, if $|W^M_{\tau,s}| = w_{\tau,M}$ and s' is a successor of s, then $|W^M_{\tau,s'}| = w_{\tau,M}$.*

(c) *For any s in S, if $|W^M_{\tau,s}| = w_{\tau,M}$, then $|W^M_{\tau+1,s}| = w_{\tau+1,M}$.*

(d) *For any s in S, if s is a successor state of $\bar{s}$ and $|W^M_{\tau+1,\bar{s}}| = w_{\tau+1,M}$, then $|W^M_{\tau,s}| = w_{\tau,M}$.*

(e) *For any s in S, if $|W^M_{\tau,s}| = w_{\tau,M}$ and $\beta \in W^M_{\tau,s}$, then $|I^M_{\beta,s}| = q^\tau/w_{\tau,M}$.*

(f) *If M is strongly connected, then $|W^M_{\tau,s'}| = w_{\tau,M}$ holds for any s' in S.*

(g) *If M is strongly connected and $\beta \in W^M_{\tau,s'}$, then $|I^M_{\beta,s'}| = q^\tau/w_{\tau,M}$.*

Proof. (a) Let s' be a state of M satisfying the condition: there exist $s \in S$ and $x \in X$ such that $s' = \delta(s,x)$ and $|W^M_{\tau+1,s}| = w_{\tau+1,M}$. From Theorem 2.1.1 (c), we have $|W^M_{\tau+1,s'}| = w_{\tau+1,M}$. From Theorem 2.1.1 (f), for any $n \geqslant 0$, $W^M_{\tau+n,s'} = W^M_{\tau,s'}Y^n$ holds. Taking $n = 1$, it follows that $w_{\tau+1,M} = |W^M_{\tau+1,s'}| = |W^M_{\tau,s'}|q$. Thus we obtain $w_{\tau,M} \leqslant |W^M_{\tau,s'}| = w_{\tau+1,M}/q$. We prove $w_{\tau,M} = w_{\tau+1,M}/q$ by reduction to absurdity. Suppose to the contrary that $w_{\tau,M} \neq w_{\tau+1,M}/q$. Since $w_{\tau,M} \leqslant w_{\tau+1,M}/q$, we have $w_{\tau,M} < w_{\tau+1,M}/q$. Therefore, there is $\bar{s} \in S$ such that $|W^M_{\tau,\bar{s}}| < w_{\tau+1,M}/q$. Clearly, $|W^M_{\tau+1,\bar{s}}| \leqslant |W^M_{\tau,\bar{s}}|q$. It follows that $|W^M_{\tau+1,\bar{s}}| < w_{\tau+1,M}$; this is a contradiction. Thus the hypothesis $w_{\tau,M} \neq w_{\tau+1,M}/q$ does not hold. That is, $w_{\tau,M} = w_{\tau+1,M}/q$.

From Theorem 2.1.2, we have $w_{\tau+1,M}|q^{\tau+1}$. Using $w_{\tau+1,M} = qw_{\tau,M}$, it follows that $w_{\tau,M}|q^\tau$.

(b) Let s' be a successor of s and $|W^M_{\tau,s}| = w_{\tau,M}$. Clearly, $|W^M_{\tau+1,s}| \leqslant q|W^M_{\tau,s}| = qw_{\tau,M}$. On the other hand, from (a), $|W^M_{\tau+1,s}| \geqslant w_{\tau+1,M} = qw_{\tau,M}$. Thus we have $|W^M_{\tau+1,s}| = qw_{\tau,M} = w_{\tau+1,M}$. From Theorem 2.1.1 (f) (for the case of $n = 1$ and $r = \tau + 1$), it follows that $W^M_{\tau+1,s'} = W^M_{\tau,s'}Y$. Using Theorem 2.1.1 (c), $|W^M_{\tau+1,s'}| = w_{\tau+1,M}$. Therefore, $w_{\tau+1,M} = q|W^M_{\tau,s'}|$. From (a), we obtain $|W^M_{\tau,s'}| = w_{\tau+1,M}/q = w_{\tau,M}$.

(c) From the proof of (b).

(d) Let s be a successor state of $\bar{s}$ and $|W^M_{\tau+1,\bar{s}}| = w_{\tau+1,M}$. From Theorem 2.1.1 (f) (for the case of $n = 1$ and $r = \tau + 1$), we have $W^M_{\tau+1,s} = W^M_{\tau,s}Y$. It follows immediately that $|W^M_{\tau+1,s}| = |W^M_{\tau,s}|q$. Next, we have $|W^M_{\tau+1,s}| = w_{\tau+1,M}$ from Theorem 2.1.1 (c), and $w_{\tau+1,M} = qw_{\tau,M}$ from (a). Therefore, $|W^M_{\tau,s}| = |W^M_{\tau+1,s}|/q = w_{\tau+1,M}/q = w_{\tau,M}$.

(e) Let $|W^M_{\tau,s}| = w_{\tau,M}$ and $\beta \in W^M_{\tau,s}$. From (c), $|W^M_{\tau+1,s}| = w_{\tau+1,M}$ holds. From Theorem 2.1.2, for any $y \in Y$, if $\beta y \in W^M_{\tau+1,s}$, then $|I^M_{\beta y,s}| = q^{\tau+1}/w_{\tau+1,M}$. Using (a), it immediately follows that $|I^M_{\beta y,s}| = q^\tau/w_{\tau,M}$.

We prove $|I^M_{\beta,s}| = q^\tau/w_{\tau,M}$ by reduction to absurdity. Suppose to the contrary that $|I^M_{\beta,s}| \neq q^\tau/w_{\tau,M}$. There are two cases to consider. In the case of $|I^M_{\beta,s}| > q^\tau/w_{\tau,M}$, it is evident that $\sum_{y\in Y} |I^M_{\beta y,s}| = q|I^M_{\beta,s}| > q^{\tau+1}/w_{\tau,M}$. Therefore, there is $y \in Y$ such that $\beta y \in W^M_{\tau+1,s}$ and $|I^M_{\beta y,s}| > q^\tau/w_{\tau,M}$. This contradicts $|I^M_{\beta y,s}| = q^\tau/w_{\tau,M}$ proven in the preceding paragraph. In the case of $|I^M_{\beta,s}| < q^\tau/w_{\tau,M}$, we have $\sum_{y\in Y} |I^M_{\beta y,s}| = q|I^M_{\beta,s}| < q^{\tau+1}/w_{\tau,M}$. Therefore, there is $y \in Y$ such that $\beta y \notin W^M_{\tau+1,s}$, or $\beta y \in W^M_{\tau+1,s}$ and $|I^M_{\beta y,s}| < q^\tau/w_{\tau,M}$. In the preceding paragraph, we have proven that $|I^M_{\beta y,s}| = q^\tau/w_{\tau,M}$ if $\beta y \in W^M_{\tau+1,s}$. Thus there is $y \in Y$ such that $\beta y \notin W^M_{\tau+1,s}$. On the other hand, from (a), $w_{\tau+1,M} = qw_{\tau,M}$. It follows that $|W^M_{\tau+1,s}| = w_{\tau+1,M} = qw_{\tau,M} = q|W^M_{\tau,s}|$. Thus we have $W^M_{\tau+1,s} = W^M_{\tau,s}Y$. Since $\beta \in W^M_{\tau,s}$, it follows that $\beta y \in W^M_{\tau+1,s}$ holds for any $y \in Y$. This is a contradiction. Therefore, the hypothesis $|I^M_{\beta,s}| \neq q^\tau/w_{\tau,M}$ does not hold, that is, $|I^M_{\beta,s}| = q^\tau/w_{\tau,M}$.

(f) Assume that M is strongly connected. For any s' in S, since M is strongly connected, there is s in S such that s' is a successor of s. From Theorem 2.1.1 (g), we have $w_{\tau+1,M} = |W^M_{\tau+1,s}|$. Using (d), it follows that $w_{\tau,M} = |W^M_{\tau,s'}|$.

(g) This is immediate from (e) and (f). □

2.2 Mutual Invertibility of Finite Automata

Lemma 2.2.1. *Let $M = \langle X, Y, S, \delta, \lambda\rangle$ and $M' = \langle Y, X, S', \delta', \lambda'\rangle$ be two finite automata. Assume that s_0 in S and $y_0, \ldots, y_{\tau-1}$ in Y satisfy the condition $y_0 \ldots y_{\tau+n} \in W^M_{\tau+n+1,s_0}$ for any $n \geqslant 0$ and any $y_\tau, \ldots, y_{\tau+n}$ in Y. If $s'_0 \in S'$ τ-matches s_0 and $s'_\tau = \delta'(s'_0, y_0 \ldots y_{\tau-1})$, then s_0 τ-matches s'_τ.*

Proof. Assume that s'_0 τ-matches s_0 and $s'_\tau = \delta'(s'_0, y_0 \ldots y_{\tau-1})$. Let $y_\tau, \ldots, y_{\tau+n}$ be arbitrary elements in Y, $n \geqslant 0$, and

$$x''_0 \ldots x''_n = \lambda'(s'_\tau, y_\tau \ldots y_{\tau+n}). \tag{2.3}$$

From the assumption, $y_0 \ldots y_{\tau+n} \in W^M_{\tau+n+1,s_0}$. Thus there exist $x'_0, \ldots, x'_{\tau+n} \in X$ such that

$$y_0 \ldots y_{\tau+n} = \lambda(s_0, x'_0 \ldots x'_{\tau+n}). \tag{2.4}$$

Since s'_0 τ-matches s_0, from (2.4), there exist $x'_{-\tau}, \ldots, x'_{-1} \in X$ such that

$$x'_{-\tau} \ldots x'_{-1} x'_0 \ldots x'_n = \lambda'(s'_0, y_0 \ldots y_{\tau+n}).$$

Noticing $s'_\tau = \delta'(s'_0, y_0 \ldots y_{\tau-1})$, it follows that

$$x'_0 \ldots x'_n = \lambda'(s'_\tau, y_\tau \ldots y_{\tau+n}). \tag{2.5}$$

From (2.5) and (2.3), we obtain $x'_0 \dots x'_n = x''_0 \dots x''_n$. Using this result, (2.4) yields

$$y_0 \dots y_{\tau+n} = \lambda(s_0, x''_0 \dots x''_n x'_{n+1} \dots x'_{\tau+n}).$$

It immediately follows that

$$y_0 \dots y_n = \lambda(s_0, x''_0 \dots x''_n). \tag{2.6}$$

From (2.3) and (2.6), we conclude that s_0 τ-matches s'_τ. □

Notice that $\lambda(s_0, x''_0 \dots x''_{\tau-1})$ equals $y_0 \dots y_{\tau-1}$ which is independent of $y_\tau, \dots, y_{\tau+n}$.

Theorem 2.2.1. *Assume that $M' = \langle Y, X, S', \delta', \lambda' \rangle$ is a weak inverse with delay τ of $M = \langle X, Y, S, \delta, \lambda \rangle$ and $|X| = |Y|$. Let*

$$S'_s = \{\delta'(s', \beta) \mid s' \in S', \beta \in W^M_{\tau,s} Y^*,\ s' \ \tau\text{-matches } s\} \tag{2.7}$$

for each $s \in S$, and $S'' = \bigcup_{s \in S_\tau} S'_s$, where

$$S_\tau = \{\delta(s, x) \mid s \in S, x \in X, w_{\tau+1,M} = |W^M_{\tau+1,s}|\}.$$

Then $M'' = \langle Y, X, S'', \delta'', \lambda'' \rangle$ is a finite subautomaton of M' of which M is a weak inverse with delay τ, where δ'' and λ'' are the restrictions of δ' and λ' on $S'' \times Y$, respectively.

Proof. Clearly, S_τ is nonempty. It follows that S'' is nonempty. For any state s'' in S'', from the definitions, there exist s in S_τ, s' in S', and β in $W^M_{\tau,s} Y^*$ such that s' τ-matches s and $s'' = \delta'(s', \beta)$. For any β' in Y^*, from $\beta \in W^M_{\tau,s} Y^*$, we have $\beta\beta' \in W^M_{\tau,s} Y^*$. Thus $\delta'(s'', \beta') = \delta'(s', \beta\beta')$ is in S'_s. It follows that $\delta'(s'', \beta')$ is in S''. Thus S'' is closed with respect to Y. Therefore, M'' is a finite subautomaton of M'.

For any state $s'' \in S''$, from the definition, there exists s in S_τ such that $s'' \in S'_s$. From the definition of S'_s, there exist s' in S' and β in $W^M_{\tau,s} Y^*$ such that s' τ-matches s and $s'' = \delta'(s', \beta)$. Denoting $\beta = y_0 \dots y_{\tau-1} y'_\tau \dots y'_l$, from the definition of S_τ, using Theorem 2.1.1 (f), we have $y_0 \dots y_{\tau+n} \in W^M_{\tau+n+1,s}$ for any $n \geqslant 0$ and any $y_\tau, \dots, y_{\tau+n}$ in Y. Denoting $s'_\tau = \delta'(s', y_0 \dots y_{\tau-1})$, from Lemma 2.2.1, s τ-matches s'_τ. Letting $\alpha = \lambda'(s'_\tau, y'_\tau \dots y'_l)$, it follows that $\delta(s, \alpha)$ τ-matches $\delta'(s'_\tau, y'_\tau \dots y'_l) = s''$. We conclude that M is a weak inverse of M'' with delay τ. □

Corollary 2.2.1. *If M is weakly invertible with delay τ and the input alphabet and the output alphabet of M have the same size, then M is a weak inverse with delay τ.*

Corollary 2.2.2. *If M is a weak inverse with delay τ and the input alphabet and the output alphabet of M have the same size, then there exists a finite subautomaton of M which is weakly invertible with delay τ and has the same input alphabet and the same output alphabet with M.*

Lemma 2.2.2. *Let $M' = \langle Y, X, S', \delta', \lambda' \rangle$ be a weak inverse with delay τ of $M = \langle X, Y, S, \delta, \lambda \rangle$ and $|X| = |Y|$. If M' is strongly connected, then M is a weak inverse with delay τ of M'.*

Proof. From Theorem 2.2.1 and its proof, it is enough to prove $S'' = S'$.

Take $s \in S$ with $|W^M_{\tau+1,s}| = w_{\tau+1,M}$. Since M' is a weak inverse with delay τ of M, we can find $s' \in S'$ such that s' τ-matches s. Take arbitrary $\alpha_0 \in X^*$ of length τ. Let $\beta_0 = \lambda(s, \alpha_0)$ and $s'_1 = \delta'(s', \beta_0)$. For any $s'_2 \in S'$, since M' is strongly connected, there exists $\beta_1 \in Y^*$ such that $s'_2 = \delta'(s'_1, \beta_1)$. It follows that $s'_2 = \delta'(s', \beta_0\beta_1)$. Since $\beta_0 \in W^M_{\tau,s}$, we have $\beta_0\beta_1 \in W^M_{\tau,s}Y^*$. From (2.7), it follows that $s'_2 \in S'_s$. From the arbitrariness of s'_2, we obtain $S' \subseteq S'_s$. This deduces $S' \subseteq S''$. On the other hand, it is evident that $S'' \subseteq S'$. Therefore, $S'' = S'$. □

Theorem 2.2.2. *Let $M' = \langle Y, X, S', \delta', \lambda' \rangle$ be strongly connected and $|X| = |Y|$. Then M' is weakly invertible with delay τ if and only if M' is a weak inverse with delay τ.*

Proof. only if : This is immediate from Corollary 2.2.1.

if : Suppose that M' is a weak inverse with delay τ. Then there exists $M = \langle X, Y, S, \delta, \lambda \rangle$ such that M' is a weak inverse with delay τ of M. Using Lemma 2.2.2, M is a weak inverse with delay τ of M'. Therefore, M' is weakly invertible with delay τ. □

2.3 Find Input by Search

2.3.1 On Output Set and Input Tree

For a weakly invertible finite automaton M with delay τ, an approach to finding $x_0 \dots x_{l-\tau}$ from s and $\lambda(s, x_0 \dots x_l)$ is guessing a value $x'_0 \dots x'_l$ and comparing $\lambda(s, x'_0 \dots x'_l)$ with $\lambda(s, x_0 \dots x_l)$. As soon as $\lambda(s, x'_0 \dots x'_l) = \lambda(s, x_0 \dots x_l)$, we obtain $x_0 \dots x_{l-\tau} = x'_0 \dots x'_{l-\tau}$. This is so-called "search". To evaluate the complexity of exhausting search or the successful probability of stochastic search, we need to know input-trees or input sets of M.

In this section, we suppose $|X| = |Y| = |Z| = q$, $|F| = p$ and $q = p^m$. Let $M = \langle X, Y, S, \delta, \lambda \rangle$ be a finite automaton.

In this section, we also use F^n to denote the set of all column vectors of dimension n over F for any set F and any nonnegative integer n. In the case

of $Y = F^m$, we use $D_{Y,r}$, or D_r for short, to denote the finite automaton $\langle Y, Y, F^r, \delta_D, \lambda_D\rangle$, for any r, $0 \leqslant r \leqslant m$, where

$$\delta_D([s_1, \ldots, s_r]^{\mathrm{T}}, [y_1, \ldots, y_m]^{\mathrm{T}}) = [y_{m-r+1}, \ldots, y_m]^{\mathrm{T}},$$
$$\lambda_D([s_1, \ldots, s_r]^{\mathrm{T}}, [y_1, \ldots, y_m]^{\mathrm{T}}) = [y_1, \ldots, y_{m-r}, s_1, \ldots, s_r]^{\mathrm{T}},$$
$$s_1, \ldots, s_r, y_1, \ldots, y_m \in F.$$

Let $M = \langle X, Y, S, \delta, \lambda\rangle$ be a finite automaton. For any $s \in S$ and any $n \geqslant 0$, we use $P1(M, s, n)$ to denote the following condition:

$$\forall \beta \in W^M_{n,s}(|I^M_{\beta,s}| = q^n/|W^M_{n,s}|).$$

And for any $t < n$, we use $P2(M, s, n, t)$ to denote the following condition:

$$\forall y_0 \cdots \forall y_{n-1}\ (\, y_0 \ldots y_{n-1} \in W^M_{n,s} \rightarrow \forall y'_t \cdots \forall y'_{n-1}(y_0 \ldots y_{t-1} y'_t \ldots y'_{n-1} \in W^M_{n,s}\,)\,).$$

Lemma 2.3.1. *Let $M = \langle X, Y, S, \delta, \lambda\rangle$, $M' = C(M, D_r)$, and $s' = \langle s, s_D\rangle$ be a state of M'.*

(a) *If $z_0 \ldots z_{n-1} \in W^{M'}_{n,s'}$, then $z_0 = \begin{bmatrix} z'_0 \\ s_D \end{bmatrix}$ for some $z'_0 \in F^{m-r}$.*

(b) *For any $\gamma = \begin{bmatrix} z'_0 \\ s_D \end{bmatrix}\begin{bmatrix} z'_1 \\ z''_1 \end{bmatrix} \ldots \begin{bmatrix} z'_{n-1} \\ z''_{n-1} \end{bmatrix}$, we have*

$$I^{M'}_{\gamma,s'} = \bigcup_{z''_n \in F^r} I^M_{\gamma',s}, \tag{2.8}$$

where $\gamma' = \begin{bmatrix} z'_0 \\ z''_1 \end{bmatrix}\begin{bmatrix} z'_1 \\ z''_2 \end{bmatrix} \ldots \begin{bmatrix} z'_{n-1} \\ z''_n \end{bmatrix}$.

(c) *If M satisfies conditions $P1(M, s, n)$ and $P2(M, s, n, t)$, then $|W^{M'}_{n,s'}| = |W^M_{n,s}|/p^r$ holds and M' satisfies conditions $P1(M', s', n)$ and $P2(M', s', n, t+1)$ in the case of $t + 1 < n$.*

Proof. (a) It is evident from the definition of D_r.

(b) Let $\gamma = \begin{bmatrix} z'_0 \\ s_D \end{bmatrix}\begin{bmatrix} z'_1 \\ z''_1 \end{bmatrix} \ldots \begin{bmatrix} z'_{n-1} \\ z''_{n-1} \end{bmatrix}$. Suppose that $x_0 \ldots x_{n-1} \in \bigcup_{z''_n \in F^r} I^M_{\gamma',s}$, where $\gamma' = \begin{bmatrix} z'_0 \\ z''_1 \end{bmatrix}\begin{bmatrix} z'_1 \\ z''_2 \end{bmatrix} \ldots \begin{bmatrix} z'_{n-1} \\ z''_n \end{bmatrix}$. Then there is $z''_n \in F^r$ such that $x_0 \ldots x_{n-1} \in I^M_{\gamma',s}$. Thus $\gamma' = \lambda(s, x_0 \ldots x_{n-1})$. From the definition of D_r, we have $\gamma = \lambda_D(s_D, \gamma')$. It follows immediately that $x_0 \ldots x_{n-1} \in I^{M'}_{\gamma,s'}$. Therefore, $I^{M'}_{\gamma,s'} \supseteq \bigcup_{z''_n \in F^r} I^M_{\gamma',s}$. On the other hand, suppose that $x_0 \ldots x_{n-1} \in I^{M'}_{\gamma,s'}$. Then $\gamma = \lambda'(s', x_0 \ldots x_{n-1})$, where λ' is the output function of M'. Since $s' = \langle s, s_D\rangle$, denoting

$$y_0 \ldots y_{n-1} = \lambda(s, x_0 \ldots x_{n-1}), \tag{2.9}$$

we have $\gamma = \lambda_D(s_D, y_0 \dots y_{n-1})$. From the definition of D_r, it follows that $y_0 \dots y_{n-1} = \gamma'$ for some $z''_n \in F^r$, where $\gamma' = \begin{bmatrix} z'_0 \\ z''_1 \end{bmatrix} \begin{bmatrix} z'_1 \\ z''_2 \end{bmatrix} \dots \begin{bmatrix} z'_{n-1} \\ z''_n \end{bmatrix}$. Thus (2.9) yields $x_0 \dots x_{n-1} \in \bigcup_{z''_n \in F^r} I^M_{\gamma',s}$. It follows immediately that $I^{M'}_{\gamma,s'} \subseteq \bigcup_{z''_n \in F^r} I^M_{\gamma',s}$. Therefore, (2.8) holds.

(c) Suppose that $P1(M, s, n)$ and $P2(M, s, n, t)$ hold.

For any element $y \in Y = F^m$, we use y' and y'' to denote the first $m - r$ components and the last r components of y, respectively, that is, $y = [y', y'']^{\mathrm{T}}$. To prove $|W^{M'}_{n,s'}| = |W^M_{n,s}|/p^r$, partition the set $W^M_{n,s}$ into blocks so that $y_0 \dots y_{n-1}$ and $z_0 \dots z_{n-1}$ belong to the same block if and only if $y_0 \dots y_{n-2} = z_0 \dots z_{n-2}$ and $y'_{n-1} = z'_{n-1}$. Since $P2(M, s, n, t)$ holds, the number of elements in each block is p^r. It follows that the number of blocks is $|W^M_{n,s}|/p^r$. From the definition of D_r, for any $\beta, \beta' \in W^M_{n,s}$, $\lambda_D(s_D, \beta) = \lambda_D(s_D, \beta')$ if and only if β and β' belong to the same block. Thus the number of elements in $W^{M'}_{n,s'}$ equals the number of blocks. Therefore, $|W^{M'}_{n,s'}| = |W^M_{n,s}|/p^r$ holds.

To prove $P1(M', s', n)$, let $s' = \langle s, s_D \rangle$ be a state of M' and $\gamma \in W^{M'}_{n,s'}$. Clearly, $I^M_{\beta,s} \cap I^M_{\beta',s} = \varnothing$ if $\beta \neq \beta'$. Using (b) and $P1(M, s, n)$, we have

$$|I^{M'}_{\gamma,s'}| = \sum_{z''_n \in F^r} |I^M_{\gamma',s}| = \sum_{z''_n \in F^r} q^n/|W^M_{n,s}| = p^r q^n/|W^M_{n,s}|, \tag{2.10}$$

where $\gamma = \begin{bmatrix} z'_0 \\ s_D \end{bmatrix} \begin{bmatrix} z'_1 \\ z''_1 \end{bmatrix} \dots \begin{bmatrix} z'_{n-1} \\ z''_{n-1} \end{bmatrix}$ and $\gamma' = \begin{bmatrix} z'_0 \\ z''_1 \end{bmatrix} \begin{bmatrix} z'_1 \\ z''_2 \end{bmatrix} \dots \begin{bmatrix} z'_{n-1} \\ z''_n \end{bmatrix}$. Using the result $|W^{M'}_{n,s'}| = |W^M_{n,s}|/p^r$ proven in the preceding paragraph, (2.10) yields

$$|I^{M'}_{\gamma,s'}| = q^n/(|W^M_{n,s}|/p^r) = q^n/|W^{M'}_{n,s'}|.$$

Therefore, $P1(M', s', n)$ holds.

We prove $P2(M', s', n, t+1)$ if $t+1 < n$. Suppose further that $t+1 < n$. Let $\gamma = \begin{bmatrix} z'_0 \\ s_D \end{bmatrix} \begin{bmatrix} z'_1 \\ z''_1 \end{bmatrix} \dots \begin{bmatrix} z'_{n-1} \\ z''_{n-1} \end{bmatrix} \in W^{M'}_{n,s'}$. Since $M' = C(M, D_r)$, there is $z''_n \in F^r$ such that $\gamma' \in W^M_{n,s}$, where $\gamma' = \begin{bmatrix} z'_0 \\ z''_1 \end{bmatrix} \begin{bmatrix} z'_1 \\ z''_2 \end{bmatrix} \dots \begin{bmatrix} z'_{n-1} \\ z''_n \end{bmatrix}$. From $P2(M, s, n, t)$, it follows immediately that

$$\begin{bmatrix} z'_0 \\ z''_1 \end{bmatrix} \begin{bmatrix} z'_1 \\ z''_2 \end{bmatrix} \dots \begin{bmatrix} z'_{t-1} \\ z''_t \end{bmatrix} \begin{bmatrix} z'_t \\ v''_{t+1} \end{bmatrix} \begin{bmatrix} v'_{t+1} \\ v''_{t+2} \end{bmatrix} \dots \begin{bmatrix} v'_{n-1} \\ v''_n \end{bmatrix} \in W^M_{n,s} \tag{2.11}$$

for any $v'_{t+1}, \dots, v'_{n-1} \in F^{m-r}$, $v''_{t+1}, \dots, v''_n \in F^r$. Since $M' = C(M, D_r)$, from the definition of D_r, (2.11) yields

$$\begin{bmatrix} z'_0 \\ s_D \end{bmatrix} \begin{bmatrix} z'_1 \\ z''_1 \end{bmatrix} \dots \begin{bmatrix} z'_t \\ z''_t \end{bmatrix} \begin{bmatrix} v'_{t+1} \\ v''_{t+1} \end{bmatrix} \dots \begin{bmatrix} v'_{n-1} \\ v''_{n-1} \end{bmatrix} \in W^{M'}_{n,s'}$$

for any v'_{t+1}, …, $v'_{n-1} \in F^{m-r}$, v''_{t+1}, …, $v''_{n-1} \in F^r$. Therefore, $P2(M', s, n, t+1)$ holds. □

A single-valued mapping φ from Y^* to Z^* is said to be *sequential*, if $|\varphi(\beta)| = |\beta|$ for any $\beta \in Y^*$ and $\varphi(\beta)$ is a prefix of $\varphi(\beta')$ for any $\beta' \in Y^*$ and any prefix β of β'.

Lemma 2.3.2. *Let $\bar{M} = \langle Y, Z, \bar{S}, \bar{\delta}, \bar{\lambda}\rangle$ be weakly invertible with delay 0, $M = \langle X, Y, S, \delta, \lambda\rangle$ and $M' = \langle X, Z, S', \delta', \lambda'\rangle = C(M, \bar{M})$. Suppose $|Y| = |Z|$.*

(a) *For any state $\bar{s}$ of $\bar{M}$, there is a sequential bijection $\varphi_{\bar{s}}$ from Y^* to Z^* such that for any state s of M and any $n > 0$, we have $W^{M'}_{n,\langle s,\bar{s}\rangle} = \varphi_{\bar{s}}(W^M_{n,s})$, $|W^{M'}_{n,\langle s,\bar{s}\rangle}| = |W^M_{n,s}|$ and $I^{M'}_{\varphi_{\bar{s}}(\beta),\langle s,\bar{s}\rangle} = I^M_{\beta,s}$ for any $\beta \in W^M_{n,s}$.*

(b) *For any state $s' = \langle s, \bar{s}\rangle$ of M', if M satisfies the condition $P1(M, s, n)$, then M' satisfies the condition $P1(M', s', n)$.*

(c) *For any state $s' = \langle s, \bar{s}\rangle$ of M', if M satisfies the condition $P2(M, s, n, t)$, then M' satisfies the condition $P2(M', s', n, t)$.*

Proof. (a) Let $\varphi_{\bar{s}}(\beta) = \bar{\lambda}(\bar{s}, \beta)$, for any $\beta \in Y^*$. Clearly, $\varphi_{\bar{s}}$ is sequential. Since $\bar{M}$ is weakly invertible with delay 0 and $|Y| = |Z|$, $\varphi_{\bar{s}}$ is bijective.

To prove $W^{M'}_{n,\langle s,\bar{s}\rangle} = \varphi_{\bar{s}}(W^M_{n,s})$, let $\beta \in W^M_{n,s}$. Then there is $\alpha \in X^*$ such that $\beta = \lambda(s, \alpha)$. Thus

$$\varphi_{\bar{s}}(\beta) = \bar{\lambda}(\bar{s}, \lambda(s, \alpha)) = \lambda'(\langle s, \bar{s}\rangle, \alpha) \in W^{M'}_{n,\langle s,\bar{s}\rangle}.$$

It immediately follows that $\varphi_{\bar{s}}(W^M_{n,s}) \subseteq W^{M'}_{n,\langle s,\bar{s}\rangle}$. On the other hand, suppose that $\gamma \in W^{M'}_{n,\langle s,\bar{s}\rangle}$. Then there is $\alpha \in X^*$ such that $\gamma = \lambda'(\langle s, \bar{s}\rangle, \alpha)$. Denoting $\beta = \lambda(s, \alpha)$, we have $\beta \in W^M_{n,s}$ and

$$\gamma = \bar{\lambda}(\bar{s}, \lambda(s, \alpha)) = \bar{\lambda}(\bar{s}, \beta) = \varphi_{\bar{s}}(\beta) \in \varphi_{\bar{s}}(W^M_{n,s}).$$

Thus $W^{M'}_{n,\langle s,\bar{s}\rangle} \subseteq \varphi_{\bar{s}}(W^M_{n,s})$. We conclude that $W^{M'}_{n,\langle s,\bar{s}\rangle} = \varphi_{\bar{s}}(W^M_{n,s})$.

Since $W^{M'}_{n,\langle s,\bar{s}\rangle} = \varphi_{\bar{s}}(W^M_{n,s})$ and $\varphi_{\bar{s}}$ is bijective, we have $|W^{M'}_{n,\langle s,\bar{s}\rangle}| = |W^M_{n,s}|$.

For any $\beta \in W^M_{n,s}$, we prove $I^{M'}_{\varphi_{\bar{s}}(\beta),\langle s,\bar{s}\rangle} = I^M_{\beta,s}$. Suppose that $\alpha \in I^M_{\beta,s}$. Then $\beta = \lambda(s, \alpha)$. It follows that

$$\lambda'(\langle s, \bar{s}\rangle, \alpha) = \bar{\lambda}(\bar{s}, \lambda(s, \alpha)) = \bar{\lambda}(\bar{s}, \beta) = \varphi_{\bar{s}}(\beta).$$

Thus $\alpha \in I^{M'}_{\varphi_{\bar{s}}(\beta),\langle s,\bar{s}\rangle}$. Therefore, $I^M_{\beta,s} \subseteq I^{M'}_{\varphi_{\bar{s}}(\beta),\langle s,\bar{s}\rangle}$. On the other hand, suppose that $\alpha \in I^{M'}_{\varphi_{\bar{s}}(\beta),\langle s,\bar{s}\rangle}$. Then $\varphi_{\bar{s}}(\beta) = \lambda'(\langle s, \bar{s}\rangle, \alpha)$. This yields

$$\varphi_{\bar{s}}(\beta) = \bar{\lambda}(\bar{s}, \lambda(s, \alpha)) = \varphi_{\bar{s}}(\lambda(s, \alpha)).$$

Since $\varphi_{\bar{s}}$ is bijective, it follows that $\beta = \lambda(s, \alpha)$. Thus $\alpha \in I^M_{\beta,s}$. Therefore, $I^{M'}_{\varphi_{\bar{s}}(\beta),\langle s,\bar{s}\rangle} \subseteq I^M_{\beta,s}$. We conclude $I^{M'}_{\varphi_{\bar{s}}(\beta),\langle s,\bar{s}\rangle} = I^M_{\beta,s}$.

(b) Suppose that $P1(M,s,n)$ holds, that is, $\forall\beta \in W^M_{n,s}(|I^M_{\beta,s}| = q^n/|W^M_{n,s}|)$. To prove $P1(M',s',n)$, let $\gamma \in W^{M'}_{n,s'}$. From (a), there is a bijection $\varphi_{\bar{s}}$ such that $W^{M'}_{n,s'} = \varphi_{\bar{s}}(W^M_{n,s})$, $|W^{M'}_{n,s'}| = |W^M_{n,s}|$ and $I^{M'}_{\varphi_{\bar{s}}(\beta),s'} = I^M_{\beta,s}$ for any $\beta \in W^M_{n,s}$. Taking $\beta = \varphi^{-1}_{\bar{s}}(\gamma)$, we have $\beta \in W^M_{n,s}$ and $I^{M'}_{\gamma,s'} = I^M_{\beta,s}$. Thus

$$|I^{M'}_{\gamma,s'}| = |I^M_{\beta,s}| = q^n/|W^M_{n,s}| = q^n/|W^{M'}_{n,s'}|.$$

We conclude that $P1(M',s',n)$ holds.

(c) Suppose that $P2(M,s,n,t)$ holds, that is, $t < n$ and $\forall y_0 \dots \forall y_{n-1}$ $(y_0 \dots y_{n-1} \in W^M_{n,s} \to \forall y'_t \dots \forall y'_{n-1}\ (y_0 \dots y_{t-1}y'_t \dots y'_{n-1} \in W^M_{n,s}))$. To prove $P2(M',s',n,t)$, let $z_0,\dots,z_{n-1}$ be in Z with $z_0 \dots z_{n-1} \in W^{M'}_{n,s'}$. Consider any elements $z'_t,\dots,z'_{n-1}$ in Z. We prove $z_0 \dots z_{t-1}\ z'_t\ \dots\ z'_{n-1} \in W^{M'}_{n,s'}$. From (a), there is a bijection $\varphi_{\bar{s}}$ such that $W^{M'}_{n,s'} = \varphi_{\bar{s}}(W^M_{n,s})$. Since $z_0 \dots z_{n-1} \in W^{M'}_{n,s'}$ and $\varphi_{\bar{s}}$ is surjective, there are $y_0,\dots,y_{n-1} \in Y$ such that $y_0 \dots y_{n-1} \in W^M_{n,s}$ and $z_0 \dots z_{n-1} = \varphi_{\bar{s}}(y_0 \dots y_{n-1})$. Denoting $y'_0 \dots y'_{n-1} = \varphi^{-1}_{\bar{s}}(z_0 \dots z_{t-1}z'_t \dots z'_{n-1})$, since $\varphi_{\bar{s}}$ is sequential and bijective, we have $y'_0 \dots y'_{t-1} = y_0 \dots y_{t-1}$. Since $P2(M,s,n,t)$ holds, this yields $y'_0 \dots y'_{n-1} \in W^M_{n,s}$. Using $W^{M'}_{n,s'} = \varphi_{\bar{s}}(W^M_{n,s})$, we obtain

$$z_0 \dots z_{t-1}z'_t \dots z'_{n-1} = \varphi_{\bar{s}}(y'_0 \dots y'_{n-1}) \in \varphi_{\bar{s}}(W^M_{n,s}) = W^{M'}_{n,s'}.$$

Therefore, $P2(M',s',n,t)$ holds. □

Lemma 2.3.3. *Let M_i, $i = 0,1,\dots,h$ be weakly invertible finite automata with delay 0 of which input alphabets and output alphabets are $X = F^m$. Let*

$$M = C(M_0, D_{X,r_1}, M_1, D_{X,r_2}, M_2, \dots, D_{X,r_{h-1}}, M_{h-1}, D_{X,r_h}, M_h).$$

Then for any $n \geqslant h$, any state s of M, and any $\beta \in W^M_{n,s}$, we have (a) *$P1(M,s,n)$ and $P2(M,s,n,t+h)$ for $t < n-h$ hold*; (b) *$|W^M_{n,s}| = |F|^{nm-r_1-\dots-r_h}$*; *and* (c) *$|I^M_{\beta,s}| = |F|^{r_1+\dots+r_h}$.*

Proof. We prove the lemma by induction on h. *Basis* : $h = 0$, that is, $M = M_0$. Since M is weakly invertible with delay 0, it is easy to see that for any $n \geqslant 0$ and any state s of M we have $|W^M_{n,s}| = |X|^n = |F|^{nm}$, and that for any $\beta \in W^M_{n,s}$ we have $|I^M_{\beta,s}| = 1 = |F|^0$. It follows immediately that $P1(M,s,n)$ holds. Notice that $W^M_{n,s}$ contains all elements of length n in X^*. Thus for any $t < n$, $P2(M,s,n,t)$ is evident. *Induction step* : Suppose that the lemma holds for the case of h. To prove the case of $h+1$, assume that M_i, $i = 0,1,\dots,h+1$ are weakly invertible finite automata with delay 0 and their input alphabets and output alphabets are $X = F^m$. Let

$$M' = C(M_0, D_{X,r_1}, M_1, D_{X,r_2}, M_2, \dots, D_{X,r_h}, M_h, D_{X,r_{h+1}}, M_{h+1}).$$

Denote

$$M = C(M_0, D_{X,r_1}, M_1, D_{X,r_2}, M_2, \ldots, D_{X,r_{h-1}}, M_{h-1}, D_{X,r_h}, M_h).$$

Then $M' = C(M, D_{X,r_{h+1}}, M_{h+1})$. Let s' be a state of M' and $n \geqslant h+1$. Denote $s' = \langle s, s_D, s_{h+1}\rangle$, where s, s_D and s_{h+1} are states of $M, D_{X,r_{h+1}}$ and M_{h+1}, respectively. From the induction hypothesis, $P1(M, s, n)$ and $P2(M, s, n, t+h)$ for $t < n-h$ hold. From Lemma 2.3.1 (c), $P1(C(M, D_{X,r_{h+1}}), \langle s, s_D\rangle, n)$ holds, and $P2(C(M, D_{X,r_{h+1}}), \langle s, s_D\rangle, n, t+h+1)$ holds for $t+h+1 < n$ (i.e., $t < n-(h+1)$). From Lemma 2.3.2 (b), $P1(C(M, D_{X,r_{h+1}}, M_{h+1}), \langle s, s_D, s_{h+1}\rangle, n)$ (i.e., $P1(M', s', n)$) holds. From Lemma 2.3.2 (c), $P2(C(M, D_{X,r_{h+1}}, M_{h+1}), \langle s, s_D, s_{h+1}\rangle, n, t+h+1)$ (i.e., $P2(M', s', n, t+h+1)$) holds for $t < n-(h+1)$. We conclude that (a) holds for the case $h+1$. We prove (b) for the case of $h+1$. From Lemma 2.3.1 (c), $|W^{C(M,D_{X,r_{h+1}})}_{n,\langle s,s_D\rangle}| = |W^M_{n,s}|/|F|^{r_{h+1}}$. Using Lemma 2.3.2 (a), it follows that

$$|W^{M'}_{n,s'}| = |W^{C(M,D_{X,r_{h+1}},M_{h+1})}_{n,\langle s,s_D,s_{h+1}\rangle}| = |W^{C(M,D_{X,r_{h+1}})}_{n,\langle s,s_D\rangle}| = |W^M_{n,s}|/|F|^{r_{h+1}}.$$

From the induction hypothesis,

$$|W^{M'}_{n,s'}| = |W^M_{n,s}|/|F|^{r_{h+1}} = |F|^{nm-r_1-\cdots-r_h}/|F|^{r_{h+1}} = |F|^{nm-r_1-\cdots-r_{h+1}}.$$

We prove (c) for the case of $h+1$. Using the result $P1(M', s', n)$ proven above, for any $\beta \in W^{M'}_{n,s'}$ we obtain $|I^{M'}_{\beta,s'}| = |X|^n/|W^{M'}_{n,s'}|$. Since $|W^{M'}_{n,s'}| = |F|^{nm-r_1-\cdots-r_{h+1}}$, it follows that $|I^{M'}_{\beta,s'}| = |F|^{mn}/|F|^{nm-r_1-\cdots-r_{h+1}} = |F|^{r_1+\cdots+r_{h+1}}$. Thus (c) holds for the case of $h+1$. □

For any finite automaton $M = \langle X, Y, S, \delta, \lambda\rangle$, if $X = Y = F^m$ and for any $s \in S$ and any $x \in X$ the first t components of $\lambda(s, x)$ are coincided with the corresponding components of x, M is said to be *t-preservable*.

Lemma 2.3.4. *Let $M' = C(M, D_r)$. If $M = \langle X, Y, S, \delta, \lambda\rangle$ is $(m-r)$-preservable, then for any state $s' = \langle s, w_0\rangle$ of M' and any input $\alpha = \begin{bmatrix} u_0 \\ v_0 \end{bmatrix} \ldots \begin{bmatrix} u_l \\ v_l \end{bmatrix}$ with $(m-r)$-dimensional u_i, the output $\lambda'(s', \alpha)$ of M' is in the form $\begin{bmatrix} u_0 \\ w_0 \end{bmatrix} \ldots \begin{bmatrix} u_l \\ w_l \end{bmatrix}$ and w_i is determined by s and $\begin{bmatrix} u_0 \\ v_0 \end{bmatrix} \ldots \begin{bmatrix} u_{i-1} \\ v_{i-1} \end{bmatrix}$ for any i, $0 < i \leqslant l$.*

Proof. Denote $\lambda(s, \alpha) = \begin{bmatrix} u'_0 \\ w_1 \end{bmatrix} \ldots \begin{bmatrix} u'_l \\ w_{l+1} \end{bmatrix}$ with $(m-r)$-dimensional u'_i. Clearly, w_i is determined by s and $\begin{bmatrix} u_0 \\ v_0 \end{bmatrix} \ldots \begin{bmatrix} u_{i-1} \\ v_{i-1} \end{bmatrix}$ for any i, $0 < i \leqslant l+1$. Since M is $(m-r)$-preservable, we have $u'_0 \ldots u'_l = u_0 \ldots u_l$. From the definition of D_r, the output $\lambda'(s', \alpha)$ of M' is $\begin{bmatrix} u'_0 \\ w_0 \end{bmatrix} \begin{bmatrix} u'_1 \\ w_1 \end{bmatrix} \ldots \begin{bmatrix} u'_l \\ w_l \end{bmatrix}$, which equals $\begin{bmatrix} u_0 \\ w_0 \end{bmatrix} \ldots \begin{bmatrix} u_l \\ w_l \end{bmatrix}$. □

Lemma 2.3.5. *Let M_i, $i = 0, 1, \ldots, h$ be finite automata of which input alphabets and output alphabets are $X = F^m$. Let*

$$M = C(M_0, D_{X,r_1}, M_1, D_{X,r_2}, M_2, \ldots, D_{X,r_{h-1}}, M_{h-1}, D_{X,r_h}, M_h, D_{X,r_{h+1}})$$

with $r_1 \leqslant r_2 \leqslant \cdots \leqslant r_{h+1}$, $h \geqslant 0$. If M_i is $(m - r_{i+1})$-preservable, $i = 0, 1, \ldots, h$, then for any state $s = \langle s_0, d_1, s_1, d_2, \ldots, s_h, d_{h+1}\rangle$ of M and any two inputs $\alpha = u_0 \ldots u_l$ and $\alpha' = u'_0 \ldots u'_l$, if $l \geqslant h$, $u_0 \ldots u_{l-h-1} = u'_0 \ldots u'_{l-h-1}$, and the first $m - r_i$ components of $u_{i+l-h-1}$ are coincided with the corresponding components of $u'_{i+l-h-1}$, $i = 1, \ldots, h+1$, then outputs on s for α and α' are the same.

Proof. Denote $\lambda(s, \alpha) = w_0 \ldots w_l$ and $\lambda(s, \alpha') = w'_0 \ldots w'_l$, where λ is the output function of M. We prove $w_0 \ldots w_l = w'_0 \ldots w'_l$ by induction on h. *Basis* : $h = 0$. From the hypothesis of the theorem, $u_0 \ldots u_{l-1} = u'_0 \ldots u'_{l-1}$ and the first $m - r_1$ components of u_l are coincided with the corresponding components of u'_l. Using Lemma 2.3.4, we have $w_0 \ldots w_l = w'_0 \ldots w'_l$. *Induction step* : Suppose that the lemma holds for the case of $h - 1$ with $h - 1 \geqslant 0$. (That is, replacing h by $h - 1$ in the lemma, the result holds.) To prove the case of h, let $\alpha = u_0 \ldots u_l$ and $\alpha' = u'_0 \ldots u'_l$ be two inputs with $l \geqslant h$. Assume that $u_0 \ldots u_{l-h-1} = u'_0 \ldots u'_{l-h-1}$ and the first $m - r_i$ components of $u_{i+l-h-1}$ are coincided with the corresponding components of $u'_{i+l-h-1}$, $i = 1, \ldots, h+1$. Denote the outputs of $C(M_0, D_{r_1})$ on the state $\langle s_0, d_1\rangle$ for the inputs α and α' by $v_0 \ldots v_l$ and $v'_0 \ldots v'_l$, respectively. Since $u_0 \ldots u_{l-h-1} = u'_0 \ldots u'_{l-h-1}$ and the first $m - r_1$ components of u_{l-h} are coincided with the corresponding components of u'_{l-h}, applying Lemma 2.3.4 to $C(M_0, D_{r_1})$, we obtain $v_0 \ldots v_{l-h} = v'_0 \ldots v'_{l-h}$. Applying Lemma 2.3.4 to $C(M_0, D_{r_1})$ again, the first $m - r_1$ components of v_{i+l-h} are coincided with the corresponding components of u_{i+l-h}, and the first $m - r_1$ components of v'_{i+l-h} are coincided with the corresponding components of u'_{i+l-h}, $i = 1, \ldots, h$. Since the first $m - r_{i+1}$ components of u_{i+l-h} are coincided with the corresponding components of u'_{i+l-h}, $i = 0, 1, \ldots, h$, using $r_1 \leqslant r_2 \leqslant \cdots \leqslant r_{h+1}$, it follows that the first $m - r_{i+1}$ components of v_{i+l-h} are coincided with the corresponding components of v'_{i+l-h}, $i = 1, \ldots, h$. Noticing $h - 1 \geqslant 0$, from the induction hypothesis on $\bar{M} = C(M_1, D_{X,r_2}, M_2, \ldots, D_{X,r_{h-1}}, M_{h-1}, D_{X,r_h}, M_h, D_{X,r_{h+1}})$ and its state $\bar{s} = \langle s_1, d_2, \ldots, s_h, d_{h+1}\rangle$, outputs of $\bar{M}$ on $\bar{s}$ for inputs $v_0 \ldots v_l$ and $v'_0 \ldots v'_l$ are the same, that is, $w_0 \ldots w_l = w'_0 \ldots w'_l$. □

For any $x_0, \ldots, x_{n+h} \in X$ and any $0 \leqslant r_1 \leqslant \cdots \leqslant r_{h+1} \leqslant m$, we use $E_{x_0 \ldots x_{n+h}, r_1, \ldots, r_{h+1}}$ to denote the set of all $x_0 \ldots x_{n-1} x'_n \ldots x'_{n+h}$ in X^* of length $n+h+1$ such that the first $m - r_{i+1}$ components of x'_{n+i} are coincided with the corresponding components of x_{n+i}, $i = 0, 1, \ldots, h$.

Lemma 2.3.6. *Let M_i, $i = 0, 1, \ldots, h$ be weakly invertible finite automata with delay 0 of which input alphabets and output alphabets are $X = F^m$. Let*

$$M = C(M_0, D_{X,r_1}, M_1, D_{X,r_2}, M_2, \ldots, D_{X,r_{h-1}}, M_{h-1}, D_{X,r_h}, M_h, D_{X,r_{h+1}})$$

with $r_1 \leqslant r_2 \leqslant \cdots \leqslant r_{h+1}$, $h \geqslant 0$. If M_i is $(m - r_{i+1})$-preservable, $i = 0, 1, \ldots, h$, then for any state s of M, any $\beta \in W^M_{n+h+1,s}$ and any $x_0 \ldots x_{n+h} \in I^M_{\beta,s}$, we have $I^M_{\beta,s} = E_{x_0 \ldots x_{n+h}, r_1, \ldots, r_{h+1}}$.

Proof. From Lemma 2.3.5, we have $I^M_{\beta,s} \supseteq E_{x_0 \ldots x_{n+h}, r_1, \ldots, r_{h+1}}$. Clearly, the number of elements in $E_{x_0 \ldots x_{n+h}, r_1, \ldots, r_{h+1}}$ is $p^{r_1 + \cdots + r_{h+1}}$ which is the number of elements in $I^M_{\beta,s}$ from Lemma 2.3.3 (c) (M_{h+1} implements the identical transformation). Thus $I^M_{\beta,s} = E_{x_0 \ldots x_{n+h}, r_1, \ldots, r_{h+1}}$. □

We use $II^M_{\beta,s}$ to denote the set $\{x \in X | \exists \alpha (x\alpha \in I^M_{\beta,s})\}$.

Lemma 2.3.7. *Let M_i, $i = 0, 1, \ldots, h$ be weakly invertible finite automata with delay 0 of which input alphabets and output alphabets are $X = F^m$. Let*

$$M = C(M_0, D_{X,r_1}, M_1, D_{X,r_2}, M_2, \ldots, D_{X,r_h}, M_h, D_{X,r_{h+1}})$$

with $r_1 \leqslant r_2 \leqslant \cdots \leqslant r_{h+1}$, $h \geqslant 0$. If M_i is $(m - r_{i+1})$-preservable, $i = 0, 1, \ldots, h$, then for any state s of M, any $\beta \in W^M_{n+h+1,s}$ and any $x_0 \ldots x_{n+h} \in I^M_{\beta,s}$, we have $II^M_{\beta,s} = E_{x_0, r_1}$ if $n = 0$, $\{x_0\}$ if $n > 0$.

Proof. This is immediate from Lemma 2.3.6. □

From the definitions, it is easy to show the following lemma.

Lemma 2.3.8. *Let $\bar{M} = \langle X, X, \bar{S}, \bar{\delta}, \bar{\lambda} \rangle$ and $M = \langle X, Y, S, \delta, \lambda \rangle$ be two finite automata, and $\bar{M}$ be weakly invertible with delay 0. Let $M' = \langle X, Y, S', \delta', \lambda' \rangle = C(\bar{M}, M)$. Then for any state $s' = \langle \bar{s}, s \rangle$ of M' and any $n \geqslant 0$, we have $W^{M'}_{n,s'} = W^M_{n,s}$, $|I^{M'}_{\beta,s'}| = |I^M_{\beta,s}|$ and $|II^{M'}_{\beta,s'}| = |II^M_{\beta,s}|$ for any $\beta \in W^M_{n,s}$.*

Using Lemma 2.3.2 (a) and Lemma 2.3.8, Lemma 2.3.6 and Lemma 2.3.7 yield the following.

Theorem 2.3.1. *Let M_i, $i = 0, 1, \ldots, h + 1$ be weakly invertible finite automata with delay 0 of which input alphabets and output alphabets are $X = F^m$. Let*

$$M = C(M_0, D_{X,r_1}, M_1, D_{X,r_2}, M_2, \ldots, D_{X,r_h}, M_h, D_{X,r_{h+1}}, M_{h+1})$$

with $r_1 \leqslant r_2 \leqslant \cdots \leqslant r_{h+1}$, $h \geqslant 0$. If M_i is $(m - r_{i+1})$-preservable, $i = 1, \ldots, h$, then for any state s of M and any $\beta \in W^M_{n+h+1,s}$, we have $|I^M_{\beta,s}| = p^{r_1 + \cdots + r_{h+1}}$, and $|II^M_{\beta,s}| = p^{r_1}$ if $n = 0$, 1 if $n > 0$.

Lemma 2.3.9. *Let M_i, $i = 0, 1, \ldots, h$ be weakly invertible finite automata with delay 0 of which input alphabets and output alphabets are $X = F^m$. Let*

$$M = C(M_0, D_{X,r_1}, M_1, D_{X,r_2}, M_2, \ldots, D_{X,r_h}, M_h, D_{X,r_{h+1}})$$

with $r_1 \leqslant r_2 \leqslant \cdots \leqslant r_{h+1}$, $h \geqslant 0$. If M_i is $(m - r_{i+1})$-preservable, $i = 0, 1, \ldots, h$, then for any state s of M and any $\beta \in W^M_{l+1,s}$, $l < h$, $I^M_{\beta,s} = E_{x_0 \ldots x_l, r_{h+1-l}, \ldots, r_{h+1}}$ and $II^M_{\beta,s} = E_{x_0, r_{h+1-l}}$ hold for any $x_0 \ldots x_l \in I^M_{\beta,s}$.

Proof. Denote $M' = C(M_0, D_{X,r_1}, \ldots, M_{h-l-1}, D_{X,r_{h-l}})$ and $M'' = C(M_{h-l}, D_{X,r_{h-l+1}}, \ldots, M_h, D_{X,r_{h+1}})$. Then $M = C(M', M'')$. Let $s = \langle s', s'' \rangle$, where s' and s'' are states of M' and M'', respectively. Below, we use λ, λ' and λ'' to denote output functions of M, M' and M'', respectively. Let $x_0 \ldots x_l \in I^M_{\beta,s}$ and

$$E' = \{\lambda'(s', \alpha) | \alpha \in E_{x_0 \ldots x_l, r_{h+1-l}, \ldots, r_{h+1}}\}.$$

Since $r_1 \leqslant r_2 \leqslant \cdots \leqslant r_{h-l} \leqslant r_{h+1-l}$ and M_i is $(m - r_{i+1})$-preservable, $i = 0, 1, \ldots, h - l$, we obtain $E' \subseteq E_{x_0 \ldots x_l, r_{h+1-l}, \ldots, r_{h+1}}$. It follows that $\lambda'(s', x_0 \ldots x_l)$, denoted by $x'_0 \ldots x'_l$, is in $E_{x_0 \ldots x_l, r_{h+1-l}, \ldots, r_{h+1}}$. Thus

$$E_{x'_0 \ldots x'_l, r_{h+1-l}, \ldots, r_{h+1}} = E_{x_0 \ldots x_l, r_{h+1-l}, \ldots, r_{h+1}}.$$

Since $\lambda''(s'', x'_0 \ldots x'_l) = \beta$, using Lemma 2.3.6 to M'', we have

$$I^{M''}_{\beta,s''} = E_{x'_0 \ldots x'_l, r_{h+1-l}, \ldots, r_{h+1}} = E_{x_0 \ldots x_l, r_{h+1-l}, \ldots, r_{h+1}}.$$

This yields $E' \subseteq I^{M''}_{\beta,s''}$. It follows that $E_{x_0 \ldots x_l, r_{h+1-l}, \ldots, r_{h+1}} \subseteq I^M_{\beta,s}$. On the other hand, suppose that $\alpha \notin E_{x_0 \ldots x_l, r_{h+1-l}, \ldots, r_{h+1}}$. Since $r_1 \leqslant r_2 \leqslant \cdots \leqslant r_{h-l} \leqslant r_{h+1-l}$ and M_i is $(m - r_{i+1})$-preservable, $i = 0, 1, \ldots, h - l$, we have $\lambda'(s', \alpha) \notin E_{x_0 \ldots x_l, r_{h+1-l}, \ldots, r_{h+1}}$. From $I^{M''}_{\beta,s''} = E_{x_0 \ldots x_l, r_{h+1-l}, \ldots, r_{h+1}}$, we have $\lambda''(s'', \lambda'(s', \alpha)) \neq \beta$, namely, $\lambda(s, \alpha) \neq \beta$. Thus $\alpha \notin I^M_{\beta,s}$. Therefore, $E_{x_0 \ldots x_l, r_{h+1-l}, \ldots, r_{h+1}} = I^M_{\beta,s}$. Moreover, from the definitions of $II^M_{\beta,s}$ and E, this equation yields $E_{x_0, r_{h+1-l}} = II^M_{\beta,s}$. □

Using Lemma 2.3.2 (a) and Lemma 2.3.8, this lemma yields the following result.

Theorem 2.3.2. *Let M_i, $i = 0, 1, \ldots, h + 1$ be weakly invertible finite automata with delay 0 of which input alphabets and output alphabets are $X = F^m$. Let*

$$M = C(M_0, D_{X,r_1}, M_1, D_{X,r_2}, M_2, \ldots, D_{X,r_h}, M_h, D_{X,r_{h+1}}, M_{h+1})$$

with $r_1 \leqslant r_2 \leqslant \cdots \leqslant r_{h+1}$, $h \geqslant 0$. If M_i is $(m - r_{i+1})$-preservable, $i = 1, \ldots, h$, then for any state s of M and any $\beta \in W^M_{l+1,s}$, $l < h$, we have $|I^M_{\beta,s}| = p^{r_{h+1-l} + \cdots + r_{h+1}}$ and $|II^M_{\beta,s}| = p^{r_{h+1-l}}$.

Let $M = \langle X, Y, S, \delta, \lambda \rangle$ be a finite automaton. For any $s \in S$ and any $\alpha \in Y^*$ of length $h+1$, we use $T_{M,s,\alpha}$ to denote a labelled tree defined as follows. We assign to the root of $T_{M,s,\alpha}$ a label s. Denoting $\alpha = y_0 \dots y_h$, for any i, $0 \leqslant i \leqslant h$, and any vertex v with level i and label s', if $x \in X$ and $\lambda(s', x) = y_i$, then an arc labelled x is emitted from v and we assign to its terminal vertex a label $\delta(s', x)$. Clearly, for any i, $0 \leqslant i \leqslant h$, $\lambda(s, x_0 \dots x_i) = y_0 \dots y_i$ if and only if there is a vertex with level $i+1$ to which from the root the unique path has the arc label sequence $x_0 \dots x_i$.

For a vertex v, the maximal length of paths with initial vertex v in $T_{M,s,\alpha}$ is called the *depth* of v. The depth of an arc means the depth of its terminal vertex. Clearly, if the level and label of v are i and s', respectively, then the depth of v is $\max j \, (\, j > 0 \rightarrow \exists x_i \dots \exists x_{i+j-1} \, (\, \lambda(s', x_i \dots x_{i+j-1}) = y_i \dots y_{i+j-1} \,) \,)$.

Theorem 2.3.3. *Let M_i, $i = 0, 1, \dots, h+1$ be weakly invertible finite automata with delay 0 of which input alphabets and output alphabets are $X = F^m$. Let*

$$M = C(M_0, D_{X,r_1}, M_1, D_{X,r_2}, M_2, \dots, D_{X,r_h}, M_h, D_{X,r_{h+1}}, M_{h+1})$$

with $r_1 \leqslant r_2 \leqslant \cdots \leqslant r_{h+1}$, $h \geqslant 0$. If M_i is $(m - r_{i+1})$-preservable, $i = 1, \dots, h$, then for any state s of M and any $\alpha \in W^M_{h+1,s}$, the number of arcs emitted from the root of $T_{M,s,\alpha}$ with depth $\geqslant l$ is $p^{r_{h+1-l}}$ for $0 \leqslant l \leqslant h$, therefore, the number of arcs emitted from the root of $T_{M,s,\alpha}$ with depth l is $p^{r_{h+1-l}} - p^{r_{h-l}}$ for $0 \leqslant l \leqslant h-1$, and the number of arcs emitted from the root of $T_{M,s,\alpha}$ with depth h is p^{r_1}.

Proof. Let $\alpha = y_0 \dots y_h$ and $0 \leqslant l \leqslant h$. From Theorem 2.3.1 and Theorem 2.3.2, $|II^M_{y_0 \dots y_l, s}| = p^{r_{h+1-l}}$. It is easy to see that the depth of an arc emitted from the root of $T_{M,s,\alpha}$ with label x is at least l if and only if $x \in II^M_{y_0 \dots y_l, s}$. Thus the number of arcs emitted from the root of $T_{M,s,\alpha}$ with depth $\geqslant l$ is $p^{r_{h+1-l}}$ □

Corollary 2.3.1. *Let M_i, $i = 0, 1, \dots, h+1$ be weakly invertible finite automata with delay 0 of which input alphabets and output alphabets are $X = F^m$. Let*

$$M = C(M_0, D_{X,r_1}, M_1, D_{X,r_2}, M_2, \dots, D_{X,r_h}, M_h, D_{X,r_{h+1}}, M_{h+1})$$

with $r_1 \leqslant r_2 \leqslant \cdots \leqslant r_{h+1}$, $h \geqslant 0$. If M_i is $(m - r_{i+1})$-preservable, $i = 1, \dots, h$, then for any state s of M and any $\alpha \in W^M_{h'+1,s}$, $h' < h$, the number of arcs emitted from the root of $T_{M,s,\alpha}$ with depth $\geqslant l$ is $p^{r_{h+1-l}}$ for $0 \leqslant l \leqslant h'$, therefore, the number of arcs emitted from the root of $T_{M,s,\alpha}$ with depth l is $p^{r_{h+1-l}} - p^{r_{h-l}}$ for $0 \leqslant l \leqslant h'-1$ and the number of arcs emitted from the root of $T_{M,s,\alpha}$ with depth h' is $p^{r_{h+1-h'}}$.

Proof. Take $\alpha' \in W^M_{h+1,s}$ of which α is a prefix. Applying Theorem 2.3.3 to α', the number of arcs emitted from the root of $T_{M,s,\alpha'}$ with depth $\geqslant l$ is $p^{r_{h+1-l}}$ for $0 \leqslant l \leqslant h'$. Since $T_{M,s,\alpha}$ is the first $h'+1$ levels of $T_{M,s,\alpha'}$, the number of arcs emitted from the root of $T_{M,s,\alpha}$ with depth $\geqslant l$ is $p^{r_{h+1-l}}$ for $0 \leqslant l \leqslant h'$. It follows that the number of arcs emitted from the root of $T_{M,s,\alpha}$ with depth l is $p^{r_{h+1-l}} - p^{r_{h-l}}$ for $0 \leqslant l \leqslant h'-1$ and the number of arcs emitted from the root of $T_{M,s,\alpha}$ with depth h' is $p^{r_{h+1-h'}}$. □

For any $x \in X$, the *x-branch* of $T_{M,s,\alpha}$, denoted by $T^x_{M,s,\alpha}$, means the subtree of $T_{M,s,\alpha}$ with the same root obtained by deleting all arcs emitted from the root but one with label x. Clearly, the level of a tree is equal to the depth of its root minus one. From Theorem 2.3.3 and Corollary 2.3.1, we have the following result.

Corollary 2.3.2. *Let M_i, $i = 0, 1, \ldots, h+1$ be weakly invertible finite automata with delay 0 of which input alphabets and output alphabets are $X = F^m$. Let*

$$M = C(M_0, D_{X,r_1}, M_1, D_{X,r_2}, M_2, \ldots, D_{X,r_h}, M_h, D_{X,r_{h+1}}, M_{h+1})$$

with $r_1 \leqslant r_2 \leqslant \cdots \leqslant r_{h+1}$, $h \geqslant 0$. If M_i is $(m-r_{i+1})$-preservable, $i = 1, \ldots, h$, then for any state s of M and any $\alpha \in W^M_{h'+1,s}$, $h' \leqslant h$, the number of branches of $T_{M,s,\alpha}$ with level l is $p^{r_{h+1-l}} - p^{r_{h-l}}$ for $0 \leqslant l \leqslant h'-1$, and the number of branches of $T_{M,s,\alpha}$ with level h' is $p^{r_{h+1-h'}}$.

Theorem 2.3.4. *Let M_i, $i = 0, 1, \ldots, h+1$ be weakly invertible finite automata with delay 0 of which input alphabets and output alphabets are $X = F^m$. Assume that M_i is $(m - r_{i+1})$-preservable, $i = 1, \ldots, h$. Let*

$$M = C(M_0, D_{X,r_1}, M_1, D_{X,r_2}, M_2, \ldots, D_{X,r_h}, M_h, D_{X,r_{h+1}}, M_{h+1})$$

with $r_1 \leqslant r_2 \leqslant \cdots \leqslant r_{h+1}$, $h \geqslant 0$.

(a) *For any state s of M and any $\alpha \in W^M_{n+h+1,s}$, the number of arcs in $T_{M,s,\alpha}$ is $\sum_{j=2}^{h+1} p^{\sum_{i=j}^{h+1} r_i} + (n+1)p^{\sum_{i=1}^{h+1} r_i}$.*

(b) *For any state s of M and any $\alpha \in W^M_{l+1,s}$, $l < h$, the number of arcs in $T_{M,s,\alpha}$ is $\sum_{j=h+1-l}^{h+1} p^{\sum_{i=j}^{h+1} r_i}$.*

Proof. (a) Let $\alpha = y_0 \ldots y_{n+h}$. From Theorem 2.3.1 and Theorem 2.3.2, we have $|I^M_{y_0 \ldots y_{j+h},s}| = p^{r_1+\cdots+r_{h+1}}$ for $0 \leqslant j \leqslant n$, and $|I^M_{y_0 \ldots y_j,s}| = p^{r_{h+1-j}+\cdots+r_{h+1}}$ for $0 \leqslant j \leqslant h-1$. Since the number of vertices with level $j+1$ of $T_{M,s,\alpha}$ is equal to $|I^M_{y_0 \ldots y_j,s}|$ for $0 \leqslant j \leqslant n+h$, the number of arcs of $T_{M,s,\alpha}$ is equal to $\sum_{j=0}^{n+h} |I^M_{y_0 \ldots y_j,s}|$ which equals $\sum_{j=2}^{h+1} p^{\sum_{i=j}^{h+1} r_i} + (n+1)p^{\sum_{i=1}^{h+1} r_i}$.

(b) Similar to (a), let $\alpha = y_0 \ldots y_l$, $l < h$. From Theorem 2.3.2, we have $|I^M_{y_0 \ldots y_j,s}| = p^{r_{h+1-j}+\cdots+r_{h+1}}$ for $0 \leqslant j \leqslant l$. Since the number of vertices with

level $j+1$ of $T_{M,s,\alpha}$ is equal to $|I^M_{y_0\dots y_j,s}|$ for $0 \leqslant j \leqslant l$, the number of arcs of $T_{M,s,\alpha}$ is equal to $\sum_{j=0}^{l} |I^M_{y_0\dots y_j,s}|$ which equals $\sum_{j=h+1-l}^{h+1} p^{\sum_{i=j}^{h+1} r_i}$. □

Corollary 2.3.3. *Let M_i, $i = 0, 1, \dots, h+1$ be weakly invertible finite automata with delay 0 of which input alphabets and output alphabets are $X = F^m$. Let*

$$M = C(M_0, D_{X,r_1}, M_1, D_{X,r_2}, M_2, \dots, D_{X,r_h}, M_h, D_{X,r_{h+1}}, M_{h+1})$$

with $r_1 \leqslant r_2 \leqslant \cdots \leqslant r_{h+1}$, $h \geqslant 0$. If M_i is $(m-r_{i+1})$-preservable, $i = 1, \dots, h$, then for any state s of M and any $\alpha \in W^M_{h'+1,s}$, $h' \leqslant h$, the number of the arcs of an x-branch with level $l+1$ of $T_{M,s,\alpha}$, $0 \leqslant l < h'$, is $1 + \sum_{j=h+1-l}^{h+1} p^{\sum_{i=j}^{h+1} r_i}$.

Proof. Let $T^x_{M,s,\alpha}$ be a branch with level $l+1$ of $T_{M,s,\alpha}$ and $l < h'$. Let T be a tree obtained by deleting the root of $T^x_{M,s,\alpha}$. It is easy to see that $T = T_{M,\delta(s,x),y_1\dots y_{l+1}}$, where $\alpha = y_0 \dots y_{h'}$. From Theorem 2.3.4 (b), the number of the arcs of $T_{M,\delta(s,x),y_1\dots y_{l+1}}$ is $\sum_{j=h+1-l}^{h+1} p^{\sum_{i=j}^{h+1} r_i}$. It follows that the number of the arcs of a $T^x_{M,s,\alpha}$ is $1 + \sum_{j=h+1-l}^{h+1} p^{\sum_{i=j}^{h+1} r_i}$. □

We point out that Lemmas 2.3.5 – 2.3.7, 2.3.9, Theorems 2.3.1 – 2.3.4 and Corollaries 2.3.1 – 2.3.3 still hold if we change the definition of preservation as follows. For a finite automaton $M = \langle X, Y, S, \delta, \lambda\rangle$, if $X = Y = F^m$ and for any $s \in S$ and any k, $1 \leqslant k \leqslant t \leqslant m$, the first k components of $\lambda(s,x)$ are independent of s and the last $m-k$ components of x, and λ_k is a bijection, where $\lambda_k(x')$ is the first k components of $\lambda(s,x)$, x' is the first k components of x, M is said to be t-preservable.

2.3.2 Exhausting Search

Let $M = \langle X, Y, S, \delta, \lambda\rangle$ be a finite automaton. An exhausting search algorithm to find an input sequence from an output sequence is the like of the following.

Algorithm 1 (exhausting search algorithm)
Input : a state s of M, an output sequence $y_0 y_1 \dots y_l \in W^M_{l+1,s}$.
Output : an input sequence $x_0 x_1 \dots x_l \in I^M_{y_0 y_1 \dots y_l, s}$.
Procedure :
1. Set $i = 0$.
2. Set $X_{s,x'_0x'_1\dots x'_{i-1}} = \{x | x \in X, y_i = \lambda(\delta(s, x'_0x'_1 \dots x'_{i-1}), x)\}$ in the case of $i > 0$, or $\{x | x \in X, y_i = \lambda(s,x)\}$ otherwise.
3. If $X_{s,x'_0\dots x'_{i-1}} \neq \varnothing$, then choose an element in it as x'_i, delete this element from it, increase i by 1 and go to Step 4; otherwise, decrease i by 1 and go to Step 5.

4. If $i > l$, then output $x'_0 \dots x'_l$ as the input $x_0 \dots x_l$ and stop; otherwise, go to Step 2.
5. If $i \geqslant 0$, go to Step 3; otherwise, prompt a failure information and stop.

Algorithm 1 is a so-called backtracking. It is convenient to understand the execution of this algorithm for an input, say s and $y_0 \dots y_l$, by means of the tree $T_{M,s,y_0\dots y_l}$; the output $x_0 \dots x_l$ is the arc label sequence of a longest path in $T_{M,s,y_0\dots y_l}$. To find one of the longest path in $T_{M,s,y_0\dots y_l}$, the algorithm attempts to exhaust all possible paths from the root to leaves. Whenever the level of a searched leaf is less than $l+1$, i.e., the path is not one of the longest, the search process comes back until an arc which is not searched yet is met, then the search process goes forward again. Whenever the level of a searched leaf is $l+1$, i.e., the path from the root to this leaf is the longest, the search process finishes and the arc label sequence of this path is the output.

How do we evaluate search amounts or accurate lower bounds in worse case or in average case of this algorithm? This is a rather difficult problem, as the structure of the input-trees for general finite automata has not been investigated yet except the case of finite automata discussed in the preceding subsection and the case of $C(M_1, M_0)$, where M_0 is linear and M_1 is weakly invertible with delay 0. We point out that the quantity $|I^M_{\beta,s}|$ may be used to evaluate a loose lower bound in worse case of the search amount. We use the number of arcs in $T_{M,s,y_0\dots y_l}$ passed in an execution of Algorithm 1 to express the search amount of that execution. Let M be weakly invertible with delay τ and $\tau \leqslant l$. Although the track of an execution of Algorithm 1 for an instance $y_0 \dots y_l$ is not clear for us, we may tentatively omit all parts but the part corresponding to $I^M_{y_0\dots y_{l-\tau},s}$ in $T_{M,s,y_0\dots y_l}$. It is evident that the minimal search amount is $l+1$ which can be reached by guessing $x'_0, \dots, x'_l$ as $x_0, \dots, x_l$, respectively. Meanwhile, in worse cases, the last guessed values of $x'_0, \dots, x'_{l-\tau}$ are $x_0, \dots, x_{l-\tau}$, respectively. The search amount in worse case is at least $l+|I^M_{y_0\dots y_{l-\tau},s}|$, as the search amount between two distinct elements of $I^M_{y_0\dots y_{l-\tau},s}$ is at least 1. Therefore, if $|I^M_{y_0\dots y_{l-\tau},s}|$ is large enough, then the exhausting search is very difficult.

Below we confine X and Y to F^m and the automaton M in Algorithm 1 to the form

$$M = C(M_0, D_{X,r_1}, M_1, D_{X,r_2}, M_2, \dots, M_{\tau-1}, D_{X,r_\tau}, M_\tau)$$

with $0 \leqslant r_1 \leqslant r_2 \leqslant \cdots \leqslant r_\tau \leqslant m$, and assume that $M_i = \langle X, X, S_i, \delta_i, \lambda_i \rangle$, $i = 0, 1, \dots, \tau$ are weakly invertible finite automata with delay 0 and that M_i is $(m - r_{i+1})$-preservable, $i = 1, \dots, \tau - 1$.

We first discuss the case of $l \geqslant \tau$. Since $y_0 y_1 \dots y_l \in W^M_{l+1,s}$, the level of $T_{M,s,y_0\dots y_l}$ is l. Since M is weakly invertible with delay τ, it is easy to see

that there exists a unique branch of $T_{M,s,y_0\ldots y_l}$ with level $\geqslant \tau$. This branch is $T^{x_0}_{M,s,y_0\ldots y_l}$. Notice that any other branch, say $T^{x_0'}_{M,s,y_0\ldots y_l}$, coincides with $T^{x_0'}_{M,s,y_0\ldots y_{\tau-1}}$. If the level of a branch of $T_{M,s,y_0\ldots y_l}$, say $h-1$, is greater than 0 and less than l, then $2 \leqslant h \leqslant \tau$ and, from Corollary 2.3.3 (with values $\tau-1$, $\tau-1$, $h-2$ for parameters h, h', l, respectively), the number of its arcs is

$$t_h = 1 + \sum_{j=\tau-h+2}^{\tau} p^{\sum_{i=j}^{\tau} r_i}.$$

Denoting $t_1 = 1$, it is trivial to see that a branch with level 0 has t_1 arcs. Consider the search process when Algorithm 1 is executing. Arcs in the tree $T_{M,s,y_0\ldots y_l}$ are searched branch by branch. Let levels of the first $i-1$ searched branches in $T_{M,s,y_0\ldots y_l}$ be less than τ and the i-th searched branch be $T^{x_0}_{M,s,y_0\ldots y_l}$. Let v_j be the number of branches with level $j-1$ in the first $i-1$ searched branches, $j = 1,\ldots,\tau$. Then the search amount of searching such $i-1$ branches is $\sum_{j=1}^{\tau} v_j t_j$. Denote

$$N = p^{r_\tau},\ m_j = p^{r_{\tau-j+1}} - p^{r_{\tau-j}},\ j = 1,\ldots,\tau, \tag{2.12}$$

where $r_0 = 0$. From Corollary 2.3.2 (with values $\tau-1$, $\tau-1$, $j-1$ for parameters h, h', l, respectively), the number of branches of $T_{M,s,y_0\ldots y_{\tau-1}}$ with level $j-1$ is m_j if $1 \leqslant j < \tau$, $m_j + 1$ if $j = \tau$. Since the level of the branch $T^{x_0}_{M,s,y_0\ldots y_l}$ is at least τ and for any $x_0' \neq x_0$, the branch $T^{x_0'}_{M,s,y_0\ldots y_l}$ coincides with the branch $T^{x_0'}_{M,s,y_0\ldots y_{\tau-1}}$, it follows that the number of branches of $T_{M,s,y_0\ldots y_l}$ with level $j-1$ is m_j, $j = 1,\ldots,\tau$; therefore, the number of branches of $T_{M,s,y_0\ldots y_l}$ is N. Fixing i and $v_j, j = 1,\ldots,\tau$, then

$$(i-1)!(N-i)!\tbinom{m_1}{v_1}\ldots\tbinom{m_\tau}{v_\tau}$$

is the number of all different permutations of N branches of $T_{M,s,y_0\ldots y_l}$ in which the i-th branch is $T^{x_0}_{M,s,y_0\ldots y_l}$ and the first $i-1$ branches consist of v_j branches with level $j-1$, $j = 1,\ldots,\tau$. Thus the average search amount for searching branches of $T_{M,s,y_0\ldots y_l}$ with level $\leqslant l-1$ is

$$c''_\tau = (N!)^{-1} \sum_{i=2}^{N} \sum_{d(i,v_1,\ldots,v_\tau)} (i-1)!(N-i)!\tbinom{m_1}{v_1}\ldots\tbinom{m_\tau}{v_\tau} \sum_{j=1}^{\tau} v_j t_j,$$

where $d(i,v_1,\ldots,v_\tau)$ represents the condition $v_1+\cdots+v_\tau = i-1\ \&\ 0 \leqslant v_1 \leqslant m_1\ \&\ \cdots\ \&\ 0 \leqslant v_\tau \leqslant m_\tau$. Letting $m_j = 0$ for $l \geqslant j > \tau$ and $\bar{m}_{l+1} = 1$, we have $c''_\tau = c''_l$, where

$$c''_l = (N!)^{-1} \sum_{i=2}^{N-\bar{m}_{l+1}+1} \sum_{d(i,v_1,\ldots,v_l)} (i-1)!(N-i)!\bar{m}_{l+1}\tbinom{m_1}{v_1}\ldots\tbinom{m_l}{v_l} \sum_{j=1}^{l} v_j t_j,$$

$\binom{0}{0} = 1$, and $d(i, v_1, \ldots, v_l) \Leftrightarrow v_1 + \cdots + v_l = i - 1 \;\&\; 0 \leqslant v_1 \leqslant m_1 \;\&\; \cdots \;\&\; 0 \leqslant v_l \leqslant m_l$.

The search process for the branch $T^{x_0}_{M,s,y_0\ldots y_l}$ with level l is reduced to the search process for $T_{M,\delta(s,x_0),y_1\ldots y_l}$. In the case of $l-1 \geqslant \tau$, we can analogously discuss as above. Repeat this discussion, until we reach a tree with level $\tau - 1$.

Now we consider the search process for $T_{M,s,y_0\ldots y_l}$ with $l < \tau$. Similar to the case of $T_{M,s,y_0\ldots y_l}$ with $l \geqslant \tau$, its arcs are searched branch by branch. Let the first $i-1$ searched branches in $T_{M,s,y_0\ldots y_l}$ be with level $\leqslant l-1$ and the i-th searched branch be with level l. If the level of a branch of $T_{M,s,y_0\ldots y_l}$, say $h-1$, satisfies $2 \leqslant h \leqslant l$, from Corollary 2.3.3 (with values $\tau - 1$, l, $h-2$ for parameters h, h', l, respectively), then the number of its arcs is

$$t_h = 1 + \sum_{j=\tau-h+2}^{\tau} p^{\sum_{i=j}^{\tau} r_i}.$$

Clearly, the number of arcs of a branch with level 0 is $t_1 (= 1)$. Let v_j be the number of branches with level $j-1$ in the first $i-1$ searched branches, $j = 1, \ldots, l$. Then the search amount of searching such $i-1$ branches is $\sum_{j=1}^{l} v_j t_j$. From Corollary 2.3.2 (with values $\tau - 1$, l, $j-1$ for parameters h, h', l, respectively), the number of branches of $T_{M,s,y_0\ldots y_l}$ with level $j-1$ is m_j, $j = 1, \ldots, l$, and the number of branches with level l is $\bar{m}_{l+1}$, where m_j is defined in (2.12), and $\bar{m}_{l+1} = p^{r_{\tau-l}}$. Fixing i and $v_j, j = 1, \ldots, l$, then

$$(i-1)!\tbinom{m_1}{v_1} \ldots \tbinom{m_l}{v_l} \tbinom{\bar{m}_{l+1}}{1} (N-i)!$$

is the number of all different permutations of N branches of $T_{M,s,y_0\ldots y_l}$ of which the i-th branch is $T^{x_0}_{M,s,y_0\ldots y_l}$ and the first $i-1$ branches consist of v_j branches with level $j-1$, $j = 1, \ldots, l$. Thus the average search amount for searching branches with level $\leqslant l-1$ is

$$c''_l = (N!)^{-1} \sum_{i=2}^{N-\bar{m}_{l+1}+1} \sum_{d(i,v_1,\ldots,v_l)} (i-1)!(N-i)!\bar{m}_{l+1}\tbinom{m_1}{v_1} \ldots \tbinom{m_l}{v_l} \sum_{j=1}^{l} v_j t_j,$$

where $d(i, v_1, \ldots, v_l)$ represents the condition $v_1 + \cdots + v_l = i - 1 \;\&\; 0 \leqslant v_1 \leqslant m_1 \;\&\; \cdots \;\&\; 0 \leqslant v_l \leqslant m_l$.

The average search amount for executing Algorithm 1 is the sum of the average search amounts for $T_{M,\delta(s,x_0\ldots x_{k-1}),y_k\ldots y_l}$ before searching its x_k-branch with level $l-k$, $k = 0, 1, \ldots, l-1$, because these search processes are independent. Therefore, the average search amount for executing Algorithm 1 is

$$\sum_{k=1}^{l} c''_k + l$$

$$= (N!)^{-1} \sum_{k=1}^{l} \sum_{i=2}^{N-\bar{m}_{k+1}+1} \sum_{d(i,v_1,\ldots,v_k)} (i-1)!\,(N-i)!\,\bar{m}_{k+1} \binom{m_1}{v_1} \ldots \binom{m_k}{v_k} \sum_{j=1}^{k} v_j t_j + l,$$

where $d(i, v_1, \ldots, v_k)$ represents the condition $v_1 + \cdots + v_k = i - 1$ & $0 \leqslant v_1 \leqslant m_1$ & $\cdots$ & $0 \leqslant v_k \leqslant m_k$. We state this result as a theorem.

Theorem 2.3.5. *Let M_i, $i = 0, 1, \ldots, \tau$ be weakly invertible finite automata with delay 0 of which input alphabets and output alphabets are $X = F^m$. Let*

$$M = C(M_0, D_{X,r_1}, M_1, D_{X,r_2}, M_2, \ldots, M_{\tau-1}, D_{X,r_\tau}, M_\tau)$$

with $0 \leqslant r_1 \leqslant r_2 \leqslant \cdots \leqslant r_\tau \leqslant m$, and M_i be $(m - r_{i+1})$-preservable, $i = 1, \ldots, \tau - 1$. If $y_0 \ldots y_l \in W^M_{l+1,s}$, then the average search amount for executing Algorithm 1 is

$$(N!)^{-1} \sum_{k=1}^{l} \sum_{i=2}^{N-\bar{m}_{k+1}+1} \sum_{d(i,v_1,\ldots,v_k)} (i-1)!\,(N-i)!\,\bar{m}_{k+1} \binom{m_1}{v_1} \ldots \binom{m_k}{v_k} \sum_{j=1}^{k} v_j t_j + l,$$

where $N = p^{r_\tau}$, $m_j = p^{r_{\tau-j+1}} - p^{r_{\tau-j}}$ for $j = 1, \ldots, l$, $\bar{m}_k = p^{r_{\tau-k+1}}$ for $k = 1, \ldots, l+1$, $r_j = 0$ if $j \leqslant 0$, $t_h = 1 + \sum_{j=\tau-h+2}^{\tau} p^{\sum_{i=j}^{\tau} r_i}$ for $h = 1, \ldots, \min(\tau, l)$, and $d(i, v_1, \ldots, v_k) \Leftrightarrow v_1 + \cdots + v_k = i - 1$ & $0 \leqslant v_1 \leqslant m_1$ & $\cdots$ & $0 \leqslant v_k \leqslant m_k$.

Consider $T_{M,s,y_0\ldots y_l}$ with level $l \geqslant \tau$. Let

$$j_{\max} = \max j\,(\,m_j \neq 0 \;\&\; 1 \leqslant j \leqslant l\,)$$

and

$$t_{\max} = t_{j_{\max}}.$$

The *main branch* means the unique branch of $T_{M,s,y_0\ldots y_l}$ with level l. A *next maximal branch* means a branch of $T_{M,s,y_0\ldots y_l}$ with level $j_{\max} - 1$. Fixing a next maximal branch, let $P_{\max}$ be the set of all permutations of N branches of $T_{M,s,y_0\ldots y_l}$ in which the next maximal branch is before the main branch. It is easy to see that

$$|P_{\max}| = \sum_{i=2}^{N} (N-2)! \binom{i-1}{1} = N!/2.$$

Corresponding to a permutation in $P_{\max}$, the search amount for executing Algorithm 1 is greater than $t_{\max}$. Notice that the average search amount for searching branches of $T_{M,\delta(s,x_0\ldots x_{k-1}),y_k\ldots y_l}$ before searching its x_k-branch with level $\leqslant l - k$ is c''_τ, for $k = 0, 1, \ldots, l - \tau$. It follows that

$$c''_\tau > (N!)^{-1}(N!/2)t_{\max} = t_{\max}/2.$$

In the case of $r_1 \geqslant 1$, we have $j_{\max} = \tau$. Therefore,

$$c''_\tau > t_\tau/2 = \big(1 + \sum_{j=2}^{\tau} p^{r_j+\cdots+r_\tau}\big)/2, \quad \text{if } r_1 \geqslant 1.$$

Since the average search amount for executing Algorithm 1 on $T_{M,s,y_0\ldots y_l}$ is greater than $(l+1-\tau)c''_\tau$, it is greater than $(l+1-\tau)\big(1+\sum_{j=2}^{\tau} p^{r_j+\cdots+r_\tau}\big)/2$ in the case of $r_1 \geqslant 1$. We state this result as a theorem.

Theorem 2.3.6. *Let M_i, $i = 0, 1, \ldots, \tau$ be weakly invertible finite automata with delay 0 of which input alphabets and output alphabets are $X = F^m$. Let*

$$M = C(M_0, D_{X,r_1}, M_1, D_{X,r_2}, M_2, \ldots, M_{\tau-1}, D_{X,r_\tau}, M_\tau)$$

with $1 \leqslant r_1 \leqslant r_2 \leqslant \cdots \leqslant r_\tau \leqslant m$, and M_i be $(m - r_{i+1})$-preservable, $i = 1, \ldots, \tau - 1$. If $y_0 \ldots y_l \in W^M_{l+1,s}$ and $l \geqslant \tau$, then the average search amount for executing Algorithm 1 is greater than

$$(l + 1 - \tau)\big(1 + \sum_{j=2}^{\tau} p^{r_j+\cdots+r_\tau}\big)/2.$$

Finally, we evaluate the search amount in worse case. Consider the search process when Algorithm 1 is executing. For searching the tree $T_{M,s,y_0\ldots y_l}$ with $y_0 \ldots y_l \in W^M_{l+1,s}$ and $l \geqslant \tau$, in worse cases, all arcs in $T_{M,s,y_0\ldots y_l}$ except some arcs mentioned below are searched. According to Corollary 2.3.2 (with values $\tau - 1$, $\tau - 1$ for parameters h, h', respectively), there are p^{r_1} branches of $T_{M,\delta(s,x_0\ldots x_{l-\tau}),y_{l-\tau+1}\ldots y_l}$ with level $\tau - 1$; only one in such branches, say $T^{x_{l-\tau+1}}_{M,\delta(s,x_0\ldots x_{l-\tau}),y_{l-\tau+1}\ldots y_l}$, is searched, because only one path of length $l + 1$ in $T_{M,s,y_0\ldots y_l}$ is searched. Next, according to Corollary 2.3.2 (with values $\tau - 1$, $\tau - 2$ for parameters h, h', respectively), there are p^{r_2} branches of $T_{M,\delta(s,x_0\ldots x_{l-\tau+1}),y_{l-\tau+2}\ldots y_l}$ with level $\tau - 2$; only one in such branches, say $T^{x_{l-\tau+2}}_{M,\delta(s,x_0\ldots x_{l-\tau+1}),y_{l-\tau+2}\ldots y_l}$, is searched, because only one path of length $l + 1$ in $T_{M,s,y_0\ldots y_l}$ is searched; and so on. According to Corollary 2.3.2 (with values $\tau - 1$, 2 for parameters h, h', respectively), there are $p^{r_{\tau-2}}$ branches of $T_{M,\delta(s,x_0\ldots x_{l-3}),y_{l-2}y_{l-1}y_l}$ with level 2; only one in such branches, say $T^{x_{l-2}}_{M,\delta(s,x_0\ldots x_{l-3}),y_{l-2}y_{l-1}y_l}$ is searched, because only one path of length $l + 1$ in $T_{M,s,y_0\ldots y_l}$ is searched. Finally, according to Corollary 2.3.2 (with values $\tau - 1$, 1 for parameters h, h', respectively), there are $p^{r_{\tau-1}}$ branches of $T_{M,\delta(s,x_0\ldots x_{l-2}),y_{l-1}y_l}$ with level 1; only two arcs in such a branch are searched, because only one path of length $l + 1$ in $T_{M,s,y_0\ldots y_l}$ is searched. For any k, $2 \leqslant k \leqslant \tau$, from Corollary 2.3.3 (with value $\tau - 1, k - 1, k - 2$ of the parameter h, h', l), the numbers of arcs of a branch of $T_{M,\delta(s,x_0\ldots x_{l-k}),y_{l-k+1}\ldots y_l}$ with level $k - 1$ is $1 + \sum_{j=\tau-k+2}^{\tau} p^{\sum_{i=j}^{\tau} r_i}$. Thus the number of arcs which are not searched is

$$\sum_{k=3}^{\tau}(p^{r_{\tau-k+1}}-1)(1+\sum_{j=\tau-k+2}^{\tau}p^{r_j+\cdots+r_\tau})+(p^{r_{\tau-1}}(1+p^{r_\tau})-2)$$

$$=\sum_{k=1}^{\tau-2}(p^{r_k}-1)(1+\sum_{j=k+1}^{\tau}p^{r_j+\cdots+r_\tau})+p^{r_{\tau-1}}(1+p^{r_\tau})-2$$

$$=\sum_{k=1}^{\tau}(p^{r_k}-1)+\sum_{k=1}^{\tau-1}(p^{r_k}-1)\sum_{j=k+1}^{\tau}p^{r_j+\cdots+r_\tau}$$

$$=\sum_{k=1}^{\tau}(p^{r_k}-1)+\sum_{j=2}^{\tau}(\sum_{k=1}^{j-1}p^{r_k}-(j-1))p^{r_j+\cdots+r_\tau}$$

$$=\sum_{k=1}^{\tau}(p^{r_k}-1)+\sum_{k=1}^{\tau-1}(\sum_{j=1}^{k}p^{r_j}-k)p^{r_{k+1}+\cdots+r_\tau}$$

$$=\sum_{k=1}^{\tau}(\sum_{j=1}^{k}p^{r_j}-k)p^{r_{k+1}+\cdots+r_\tau}.$$

From Theorem 2.3.4 (a) (with values $\tau-1$, $l+1-\tau$ for parameters h, n, respectively), the number of arcs in $T_{M,s,y_0\ldots y_l}$ is

$$\sum_{j=2}^{\tau}p^{\sum_{i=j}^{\tau}r_i}+(l-\tau+2)p^{\sum_{i=1}^{\tau}r_i}.$$

Then the search amount in worse case for executing Algorithm 1 is

$$\sum_{j=2}^{\tau}p^{r_j+\cdots+r_\tau}+(l-\tau+2)p^{r_1+\cdots+r_\tau}-\sum_{k=1}^{\tau}(\sum_{j=1}^{k}p^{r_j}-k)p^{r_{k+1}+\cdots+r_\tau}$$

$$=(l-\tau+2)p^{r_1+\cdots+r_\tau}-\sum_{k=1}^{\tau}(\sum_{j=1}^{k}p^{r_j}-k-1)p^{r_{k+1}+\cdots+r_\tau}-1$$

$$=(l-\tau+1)p^{r_1+\cdots+r_\tau}+\sum_{k=0}^{\tau}(k+1)p^{r_{k+1}+\cdots+r_\tau}-\sum_{k=1}^{\tau}(\sum_{j=1}^{k}p^{r_j})p^{r_{k+1}+\cdots+r_\tau}-1$$

$$=(l-\tau+1)p^{r_1+\cdots+r_\tau}+\sum_{k=1}^{\tau+1}kp^{r_k+\cdots+r_\tau}-\sum_{k=1}^{\tau}(\sum_{j=1}^{k}p^{r_j})p^{r_{k+1}+\cdots+r_\tau}-1$$

$$=(l-\tau+1)p^{r_1+\cdots+r_\tau}+\sum_{k=1}^{\tau}(kp^{r_k}-\sum_{j=1}^{k}p^{r_j})p^{r_{k+1}+\cdots+r_\tau}+\tau.$$

We state this result as a theorem.

Theorem 2.3.7. *Let M_i, $i=0,1,\ldots,\tau$ be weakly invertible finite automata with delay 0 of which input alphabets and output alphabets are $X=F^m$. Let*

$$M = C(M_0, D_{X,r_1}, M_1, D_{X,r_2}, M_2, \ldots, M_{\tau-1}, D_{X,r_\tau}, M_\tau)$$

with $0 \leqslant r_1 \leqslant r_2 \leqslant \cdots \leqslant r_\tau \leqslant m$, *and* M_i *be* $(m - r_{i+1})$*-preservable,* $i = 1, \ldots, \tau - 1$. *If* $y_0 \ldots y_l \in W^M_{l+1,s}$ *and* $l \geqslant \tau$, *then the search amount in worse case for executing Algorithm* 1 *is*

$$(l - \tau + 1)p^{r_1+\cdots+r_\tau} + \sum_{k=1}^{\tau} \big(kp^{r_k} - \sum_{j=1}^{k} p^{r_j}\big)p^{r_{k+1}+\cdots+r_\tau} + \tau.$$

Using Theorem 2.3.1, this theorem yields the following.

Corollary 2.3.4. *Let* M_i, $i = 0, 1, \ldots, \tau$ *be weakly invertible finite automata with delay* 0 *of which input alphabets and output alphabets are* $X = F^m$. *Let*

$$M = C(M_0, D_{X,r_1}, M_1, D_{X,r_2}, M_2, \ldots, M_{\tau-1}, D_{X,r_\tau}, M_\tau)$$

with $0 \leqslant r_1 \leqslant r_2 \leqslant \cdots \leqslant r_\tau \leqslant m$, *and* M_i *be* $(m - r_{i+1})$*-preservable,* $i = 1, \ldots, \tau - 1$. *If* $y_0 \ldots y_l \in W^M_{l+1,s}$ *and* $l \geqslant \tau$, *then* $(l - \tau + 1)p^{r_1+\cdots+r_\tau} + \tau$, *i.e.,* $(l - \tau + 1)|I^M_{y_0 \ldots y_l, s}| + \tau$ *is a positive lower bound of the search amount in worse case for executing Algorithm* 1 *and this bound can be reached if and only if* $r_1 = \cdots = r_\tau$.

2.3.3 Stochastic Search

Algorithm 2 (stochastic search algorithm)
Input : a state s of $M = \langle X, Y, S, \delta, \lambda\rangle$, an output sequence $y_0 y_1 \ldots y_l \in W^M_{l+1,s}$.
Output : an input sequence $x_0 x_1 \ldots x_l \in I^M_{y_0 y_1 \ldots y_l, s}$.
Procedure :
1. Set $i = 0$.
2. Set $X_{s,x'_0 x'_1 \ldots x'_{i-1}} = \{x | x \in X, y_i = \lambda(\delta(s, x'_0 x'_1 \ldots x'_{i-1}), x)\}$ in the case of $i > 0$, or $\{x | x \in X, y_i = \lambda(s, x)\}$ otherwise.
3. If $X_{s,x'_0 \ldots x'_{i-1}} \neq \varnothing$, then choose an element in it as x'_i, increase i by 1 and go to Step 4; otherwise, prompt a failure information and stop.
4. If $i > l$, then output $x'_0 \ldots x'_l$ as the input $x_0 \ldots x_l$ and stop; otherwise, go to Step 2.

Let $p_i^{y_0 \cdots y_l}$ be the probability of successfully choosing $x'_0, \ldots, x'_i$ in Algorithm 2.

Let $pr(x'_i | x'_0 \ldots x'_{i-1}, s, y_0 \ldots y_l)$ be the conditional probability of successfully choosing x'_i in Algorithm 2, $0 \leqslant i \leqslant l$. It is easy to see that

$$p_i^{y_0 \cdots y_l} = p_{i-1}^{y_0 \cdots y_l} pr(x'_i | x'_0 \ldots x'_{i-1}, s, y_0 \ldots y_l),\ i = 0, 1, \ldots, l,$$

where $p_{-1}^{y_0 \cdots y_l} = 1$. Thus

$$p_l^{y_0 \cdots y_l} = \prod_{i=0}^{l} pr(x_i' | x_0' \ldots x_{i-1}', s, y_0 \ldots y_l).$$

Below we confine X and Y to F^m and the automaton M in Algorithm 2 to the form

$$M = C(M_0, D_{X,r_1}, M_1, D_{X,r_2}, M_2, \ldots, M_{\tau-1}, D_{X,r_\tau}, M_\tau)$$

with $0 \leqslant r_1 \leqslant r_2 \leqslant \cdots \leqslant r_\tau \leqslant m$, and assume that $M_i = \langle X, X, S_i, \delta_i, \lambda_i \rangle$, $i = 0, 1, \ldots, \tau$ are weakly invertible finite automata with delay 0 and that M_i is $(m - r_{i+1})$-preservable, $i = 1, \ldots, \tau - 1$. Notice that $X_{s,x_0' x_1' \ldots x_{i-1}'} = I^M_{y_i, \delta(s, x_0' \ldots x_{i-1}')}$ and

$$pr(x_i' | x_0' \ldots x_{i-1}', s, y_0 \ldots y_l) = |II^M_{y_i \ldots y_l, \delta(s, x_0' \ldots x_{i-1}')}| / |I^M_{y_i, \delta(s, x_0' \ldots x_{i-1}')}|.$$

Using Theorem 2.3.1 (with values $\tau - 1$, $l - \tau - i + 1$ for parameters h, n, respectively), we have $|II^M_{y_i \ldots y_l, \delta(s, x_0' \ldots x_{i-1}')}| = 1$ if $\tau - l + i \leqslant 0$, or p^{r_1} if $\tau - l + i = 1$. Using Theorem 2.3.2 (with values $\tau - 1$, $l - i$ for parameters h, l, respectively), we have $|II^M_{y_i \ldots y_l, \delta(s, x_0' \ldots x_{i-1}')}| = p^{r_{\tau - l + i}}$ if $\tau - l + i > 1$. Using Theorem 2.3.2 (with values $\tau - 1$, 0 for parameters h, l, respectively), we have $|I^M_{y_i, \delta(s, x_0' \ldots x_{i-1}')}| = p^{r_\tau}$, for $i = 0, 1, \ldots, l$. It follows that

$$pr(x_i' | x_0' \ldots x_{i-1}', s, y_0 \ldots y_l) = p^{r_{\tau - l + i}} / p^{r_\tau} = p^{r_{\tau - l + i} - r_\tau}, \ i = 0, 1, \ldots, l,$$

where $r_j = 0$, for $j \leqslant 0$. Therefore,

$$\begin{aligned} p_l^{y_0 \cdots y_l} &= \prod_{i=0}^{l} pr(x_i' | x_0' \ldots x_{i-1}', s, y_0 \ldots y_l) \\ &= \prod_{i=0}^{l} p^{r_{\tau - l + i} - r_\tau} \\ &= p^{\sum_{i=0}^{l} (r_{\tau - l + i} - r_\tau)} \\ &= p^{\sum_{i=0}^{l} r_{\tau - l + i} - (l+1) r_\tau} \\ &= p^{\sum_{i=0}^{\min(l, \tau - 1)} r_{\tau - i} - (l+1) r_\tau}. \end{aligned}$$

We obtain the following.

Theorem 2.3.8. *Let M_i, $i = 0, 1, \ldots, \tau$ be weakly invertible finite automata with delay 0 of which input alphabets and output alphabets are $X = F^m$. Let*

$$M = C(M_0, D_{X,r_1}, M_1, D_{X,r_2}, M_2, \ldots, M_{\tau-1}, D_{X,r_\tau}, M_\tau)$$

with $0 \leqslant r_1 \leqslant r_2 \leqslant \cdots \leqslant r_\tau \leqslant m$, *and* M_i *be* $(m - r_{i+1})$*-preservable,* $i = 1, \ldots, \tau - 1$. *If* $y_0 \ldots y_l \in W^M_{l+1,s}$, *then the probability of successfully choosing* $x'_0, \ldots, x'_l$ *of Algorithm* 2 *is*

$$p_l^{y_0 \cdots y_l} = p^{\sum_{i=0}^{\min(l,\tau-1)} r_{\tau-i} - (l+1) r_\tau}.$$

Historical Notes

The concepts of the r-output set, the r-output weight and β-input set are first defined in [4] for $r = |\beta| = 1$ and in [8] for the general case, and the minimal 1-output weight is also defined in [4]. The minimal r-output weight for general r, the minimal r-input weight and the maximal r-input weight are introduced in [128]. And the input-tree $T_{M,s,\alpha}$ is introduced in [120]. The material of this chapter is based on [128].

3. R_a R_b Transformation Method

Renji Tao

Institute of Software, Chinese Academy of Sciences
Beijing 100080, China trj@ios.ac.cn

Summary.

For characterization of the structure of weakly invertible finite automata, the state tree method is presented in Chap. 1. However, from an algorithmic viewpoint, it is rather hard to manipulate such state trees for large state alphabets and delay steps. In this chapter, the R_a R_b transformation is presented and used to generate a kind of weakly invertible finite automata and their weak inverses. This result paves the way for the key generation of a public key cryptosystem based on finite automata in Chap. 9. For weakly invertible quasi-linear finite automata over finite fields, the structure problem is also solved by means of the R_a R_b transformation method.

This chapter may be regarded as an introduction to Chap. 9.

Key words: *R_a R_b transformation, inversion method, quasi-linear finite automata*

For characterization of the structure of weakly invertible finite automata, the state tree method is presented in Chap. 1. However, from an algorithmic viewpoint, it is rather hard to manipulate such state trees for large state alphabets and delay steps. In this chapter, the R_a R_b transformation is presented and used to generate a kind of weakly invertible finite automata and their weak inverses. This result paves the way for the key generation of a public key cryptosystem based on finite automata in Chap. 9. For weakly invertible quasi-linear finite automata over finite fields, the structure problem is also solved by means of the R_a R_b transformation method.

3.1 Sufficient Conditions and Inversion

Throughout this chapter, for any integer i, any nonnegative integer k and any symbol string z, we use $z(i,k)$ to denote the symbol string $z_i, z_{i-1}, \ldots, z_{i-k+1}$ (void string in the case of $k=0$). Let X and U be two finite nonempty sets. Let Y be a column vector space of dimension m over a finite commutative ring R with identity, where m is a positive integer. For any integer i, we use x_i (x'_i), y_i (y'_i, y''_i) and u_i to denote elements in X, Y and U, respectively.

Let r and t be two nonnegative integers, and p an integer with $p \geqslant -1$. For any nonnegative integer k, let f_k and f'_k be two single-valued mappings from $X^{r+1} \times U^{p+1} \times Y^{k+t+1}$ to Y.

Rule R_a : Let $eq_k(i)$ be an equation in the form

$$f_k(x(i,r+1), u(i,p+1), y(i+k,k+t+1)) = 0.$$

Let φ_k be a transformation on $eq_k(i)$, and $eq'_k(i)$ the transformational result in the form

$$f'_k(x(i,r+1), u(i,p+1), y(i+k,k+t+1)) = 0.$$

If $eq_k(i)$ and $eq'_k(i)$ are equivalent (viz. their solutions are the same), $eq'_k(i)$ is said to be *obtained from* $eq_k(i)$ *by Rule* R_a *using* φ_k, denoted by

$$eq_k(i) \overset{R_a[\varphi_k]}{\longrightarrow} eq'_k(i).$$

Rule R_b : Assume that $eq'_k(i)$ is an equation in the form

$$f'_k(x(i,r+1), u(i,p+1), y(i+k,k+t+1)) = 0$$

and that the last $m - r_{k+1}$ components of the left side of $eq'_k(i)$ do not depend on u_i and x_i. Let $eq_{k+1}(i)$ be the equation

$$\begin{bmatrix} E'_k f'_k(x(i,r+1), u(i,p+1), y(i+k,k+t+1)) \\ E''_k f'_k(x(i+1,r+1), u(i+1,p+1), y(i+1+k,k+t+1)) \end{bmatrix} = 0,$$

where E'_k and E''_k are the submatrix of the first r_{k+1} rows and the submatrix of the last $m - r_{k+1}$ rows of the $m \times m$ identity matrix, respectively. $eq_{k+1}(i)$ is said to be *obtained from* $eq'_k(i)$ *by Rule* R_b *with respect to variables* x *and* u, denoted by

$$eq'_k(i) \overset{R_b[r_{k+1}]}{\longrightarrow} eq_{k+1}(i).$$

Notice that the result equation $eq_{k+1}(i)$ of applying Rule R_b to $eq'_k(i)$ is still in the form

$$f_{k+1}(x(i,r+1), u(i,p+1), y(i+k+1,k+1+t+1)) = 0$$

on which Rule R_a should be applied.

Assume that

$$eq_k(i) \xrightarrow{R_a[\varphi_k]} eq'_k(i),\ eq'_k(i) \xrightarrow{R_b[r_{k+1}]} eq_{k+1}(i),\quad k = 0, 1, \ldots, \tau - 1.$$

It is easy to prove the following properties.

Property (a) For any $k, 0 \leqslant k < \tau$, $eq_k(i)$ if and only if $eq'_k(i)$.

Property (b) For any $k, 0 \leqslant k < \tau$, if $eq'_k(i), i = b, b+1, \ldots$, then $eq_{k+1}(i), i = b, b+1, \ldots$

Property (c) For any $k, 0 \leqslant k < \tau$, if $eq_{k+1}(i), i = b, b+1, \ldots$, then $eq'_k(i), i = b+1, b+2, \ldots$

From Property (a) and Property (b), we have the following.

Property (d) If $eq_0(i),\ i = b, b+1, \ldots$, then $eq_\tau(i),\ i = b, b+1, \ldots$

From Property (a) and Property (c), we have the following.

Property (e) If $eq_\tau(i),\ i = b, b+1, \ldots$, then $eq_0(i),\ i = b+\tau, b+\tau+1, \ldots$

Letting C be a matrix with m columns, we use $Ceq'_k(i)$ to denote the equation obtained by multiplying two sides of $eq'_k(i)$ on the left by C. Using Property (a), it is easy to show the following property.

Property (f) For any $k, 0 \leqslant k < \tau$, and any $b < e$, $eq_k(i),\ i = b, b+1, \ldots, e$ if and only if

$$\begin{aligned}
&eq_{k+1}(i),\ i = b, b+1, \ldots, e-1,\\
&E'_k eq'_k(e),\\
&E''_k eq'_k(b).
\end{aligned}$$

Applying Property (f) repeatedly, we have the following.

Property (g) $eq_0(i),\ i = 0, 1, \ldots, \tau$ if and only if

$$\begin{aligned}
&eq_\tau(0),\\
&E'_0 eq'_0(\tau), E'_1 eq'_1(\tau-1), \ldots, E'_{\tau-1} eq'_{\tau-1}(1),\\
&E''_0 eq'_0(0), E''_1 eq'_1(0), \ldots, E''_{\tau-1} eq'_{\tau-1}(0).
\end{aligned}$$

Let $M = \langle X, Y, Y^t \times U^{p+1} \times X^r, \delta, \lambda\rangle$ be a finite automaton defined by

$$\begin{aligned}
&\delta(\langle y(i-1,t), u(i,p+1), x(i-1,r)\rangle, x_i)\\
&\quad = \langle y(i,t), u(i+1,p+1), x(i,r)\rangle,\\
&\lambda(\langle y(i-1,t), u(i,p+1), x(i-1,r)\rangle, x_i) = y_i,
\end{aligned}$$

where

$$\begin{aligned}
y_i &= f(y(i-1,t), u(i,p+1), x(i,r+1)),\\
u_{i+1} &= g(y(i-1,t), u(i,p+1), x(i,r+1)),
\end{aligned} \tag{3.1}$$

f and g are two single-valued mappings from $Y^t \times U^{p+1} \times X^{r+1}$ to Y and U, respectively. Assume that $eq_0(i)$ is the equation

$$-y_i + f(y(i-1,t), u(i,p+1), x(i,r+1)) = 0 \tag{3.2}$$

and that

$$eq_k(i) \overset{R_a[\varphi_k]}{\longrightarrow} eq'_k(i),\ eq'_k(i) \overset{R_b[r_{k+1}]}{\longrightarrow} eq_{k+1}(i),\quad k = 0, 1, \ldots, \tau - 1$$

is an R_a R_b transformation sequence.

Let f^*_τ be a single-valued mapping from $X^r \times U^{p+1} \times Y^{\tau+t+1}$ to X. From f^*_τ and g in (3.1), construct a finite automaton $M^* = \langle Y, X, X^r \times U^{p+1} \times Y^{\tau+t}, \delta^*, \lambda^* \rangle$ by

$$\begin{aligned}
&\delta^*(\langle x(i-1,r), u(i,p+1), y'(i-1,\tau+t)\rangle, y'_i) \\
&\quad = \langle x(i,r), u(i+1,p+1), y'(i,\tau+t)\rangle, \\
&\lambda^*(\langle x(i-1,r), u(i,p+1), y'(i-1,\tau+t)\rangle, y'_i) = x_i,
\end{aligned}$$

where

$$\begin{aligned}
x_i &= f^*_\tau(x(i-1,r), u(i,p+1), y'(i,\tau+t+1)), \\
u_{i+1} &= g(y'(i-\tau-1,t), u(i,p+1), x(i,r+1)).
\end{aligned}$$

Lemma 3.1.1. *Assume that for any parameters x_{i-1}, …, x_{i-r}, u_i, …, u_{i-p}, $y_{i+\tau}$, …, y_{i-t}, $eq_\tau(i)$ has a solution x_i*

$$x_i = f^*_\tau(x(i-1,r), u(i,p+1), y(i+\tau, \tau+t+1)).$$

Let

$$s^*_0 = \langle x(-1,r), u(0,p+1), y'(-1,\tau+t)\rangle$$

be a state of M^*. For any $y'_0, y'_1, \ldots \in Y$, if

$$x_0 x_1 \ldots = \lambda^*(s^*_0, y'_0 y'_1 \ldots),$$

then

$$y'_0 y'_1 \ldots = \lambda(s_\tau, x_\tau x_{\tau+1} \ldots),$$

where

$$\begin{aligned}
&s_\tau = \langle y'(-1,t), u(\tau,p+1), x(\tau-1,r)\rangle, \\
&u_{i+1} = g(y'(i-\tau-1,t), u(i,p+1), x(i,r+1)),\ i = 0, 1, \ldots, \tau-1.
\end{aligned}$$

Proof. Denoting

$$\begin{aligned}
u_{i+1} &= g(y'(i-\tau-1,t), u(i,p+1), x(i,r+1)), \\
&\quad i = \tau, \tau+1, \ldots,
\end{aligned} \tag{3.3}$$

since $x_0x_1\ldots = \lambda^*(s_0^*, y_0'y_1'\ldots)$, we have

$$x_i = f_\tau^*(x(i-1,r), u(i,p+1), y'(i,\tau+t+1)),\ i=0,1,\ldots$$

Denoting $y_{i+\tau} = y_i'$ for any integer i, this yields that

$$x_i = f_\tau^*(x(i-1,r), u(i,p+1), y(i+\tau,\tau+t+1)),\ i=0,1,\ldots$$

From the hypothesis of the lemma, for any parameters $x_{i-1}, \ldots, x_{i-r}, u_i, \ldots, u_{i-p}, y_{i+\tau}, \ldots, y_{i-t}$, such a value of x_i is a solution of $eq_\tau(i)$, i.e.,

$$f_\tau(x(i,r+1), u(i,p+1), y(i+\tau,\tau+t+1)) = 0,$$

for $i = 0,1,\ldots$ Thus $eq_\tau(i),\ i=0,1,\ldots$ hold. From Property (e), we have $eq_0(i),\ i=\tau,\tau+1,\ldots$ It immediately follows that

$$y_i = f(y(i-1,t), u(i,p+1), x(i,r+1)),\ i=\tau,\tau+1,\ldots$$

Using (3.3), we then have

$$\begin{aligned} y'_{i-\tau} &= f(y'(i-\tau-1,t), u(i,p+1), x(i,r+1)),\\ u_{i+1} &= g(y'(i-\tau-1,t), u(i,p+1), x(i,r+1)),\\ &\quad i=\tau,\tau+1,\ldots \end{aligned}$$

From the definition of M, this yields $y_0'y_1'\ldots = \lambda(s_\tau, x_\tau x_{\tau+1}\ldots)$. □

Theorem 3.1.1. *Assume that for any parameters $x_{i-1}, \ldots, x_{i-r}, u_i, \ldots, u_{i-p}, y_{i+\tau}, \ldots, y_{i-t}$, $eq_\tau(i)$ has a solution x_i*

$$x_i = f_\tau^*(x(i-1,r), u(i,p+1), y(i+\tau,\tau+t+1)).$$

Let $M^{**} = \langle Y, X, S^{**}, \delta^*|_{S^{**}\times Y}, \lambda^*|_{S^{**}\times Y}\rangle$ be a finite subautomaton of M^*, where

$$S^{**} = \{\delta^*(s^*, y_0\ldots y_{\tau-1}) \mid s^* \in X^r \times U^{p+1} \times Y^{\tau+t}, y_0, \ldots, y_{\tau-1} \in Y\}.$$

For any state

$$s_0^* = \langle x(-1,r), u(0,p+1), y'(-1,\tau+t)\rangle$$

of M^{**}, if

$$s_0 = \langle y'(-\tau-1,t), u(0,p+1), x(-1,r)\rangle,$$

then the state s_0 of M matches s_0^* with delay τ and $\lambda(s_0,\alpha) = y'_{-\tau}\ldots y'_{-1}$ for any $\alpha \in \lambda^*(s_0^*, Y^\tau)$. Therefore, M is a weak inverse with delay τ of M^{**}.

Proof. For any $y'_0, y'_1 \ldots \in Y$, let

$$x_0 x_1 \ldots = \lambda^*(s_0^*, y'_0 y'_1 \ldots).$$

From the definition of S^{**}, since $s^* \in S^{**}$, there are $x_{-r-1}, \ldots, x_{-r-\tau} \in X$, $u_{-p-1}, \ldots, u_{-p-\tau} \in U$, and $y'_{-\tau-t-1}, \ldots, y'_{-2\tau-t} \in Y$ such that

$$\delta^*(s^*_{-\tau}, y'_{-\tau} \ldots y'_{-1}) = s_0^*,$$

where

$$s^*_{-\tau} = \langle x(-\tau-1, r), u(-\tau, p+1), y'(-\tau-1, \tau+t)\rangle.$$

It follows that

$$\begin{aligned} x_{-\tau} \ldots x_{-1} x_0 x_1 \ldots &= \lambda^*(s^*_{-\tau}, y'_{-\tau} \ldots y'_{-1} y'_0 y'_1 \ldots), \\ u_{i+1} &= g(y'(i-\tau-1, t), u(i, p+1), x(i, r+1)), \\ i &= -\tau, \ldots, -1, \end{aligned}$$

where $x_{-\tau} \ldots x_{-r-1} = \lambda^*(s^*_{-\tau}, y'_{-\tau} \ldots y'_{-r-1})$ in the case of $\tau > r$. From Lemma 3.1.1, we obtain

$$y'_{-\tau} \ldots y'_{-1} y'_0 y'_1 \ldots = \lambda(s_0, x_0 x_1 \ldots).$$

Thus, s_0 matches the state s_0^* with delay τ and $\lambda(s_0, \alpha) = y'_{-\tau} \ldots y'_{-1}$ for any $\alpha \in \lambda^*(s_0^*, Y^\tau)$. □

Corollary 3.1.1. *If for any parameters x_{i-1}, ..., x_{i-r}, u_i, ..., u_{i-p}, $y_{i+\tau}, \ldots, y_{i-t}$, $eq_\tau(i)$ has a solution x_i, then M is a weak inverse with delay τ.*

Theorem 3.1.2. *Assume that $t = 0$ and that for any parameters x_{i-1}, ..., x_{i-r}, u_i, ..., u_{i-p}, $y_{i+\tau}$, ..., y_{i-t}, $eq_\tau(i)$ has a solution x_i*

$$x_i = f^*_\tau(x(i-1, r), u(i, p+1), y(i+\tau, \tau+t+1)).$$

For any state

$$s_0^* = \langle x(-1, r), u(0, p+1), y'(-1, \tau+t)\rangle$$

of M^*, if

$$s_0 = \langle u(0, p+1), x(-1, r)\rangle,$$

then the state s_0 of M matches s_0^* with delay τ. Therefore, M is a weak inverse with delay τ of M^*.

Proof. For any $y'_0, y'_1 \ldots \in Y$, let

$$x_0 x_1 \ldots = \lambda^*(s_0^*, y'_0 y'_1 \ldots).$$

Let

$$s_\tau = \langle u(\tau, p+1), x(\tau - 1, r) \rangle,$$

where

$$u_{i+1} = g(u(i, p+1), x(i, r+1)),\ i = 0, 1, \ldots, \tau - 1.$$

From Lemma 3.1.1, we obtain

$$y'_0 y'_1 \ldots = \lambda(s_\tau, x_\tau x_{\tau+1} \ldots).$$

Since $t = 0$, it is easy to see that $\delta(s_0, x_0 \ldots x_{\tau-1}) = s_\tau$. Thus

$$y''_0 \ldots y''_{\tau-1} y'_0 y'_1 \ldots = \lambda(s_0, x_0 x_1 \ldots)$$

for some $y''_0, \ldots, y''_{\tau-1} \in Y$. Therefore, s_0 matches s_0^* with delay τ. □

Theorem 3.1.3. *If for any parameters $x_{i-1}, \ldots, x_{i-r}, u_i, \ldots, u_{i-p}, y_{i+\tau}, \ldots, y_{i-t}$, $eq_\tau(i)$ has at most one solution x_i, then M is weakly invertible with delay τ.*

Proof. Assume that for any parameters x_{i-1}, …, x_{i-r}, u_i, …, u_{i-p}, $y_{i+\tau}$, …, y_{i-t}, $eq_\tau(i)$ has at most one solution x_i. For any initial state $s_0 = \langle y(-1, t),\ u(0, p+1),\ x(-1, r)\ \rangle$, and any input $x_0 x_1 \ldots$ of M, let

$$y_0 y_1 \ldots = \lambda(s_0, x_0 x_1 \ldots).$$

From the definition of M, (3.1) holds for $i = 0, 1, \ldots$ Since $eq_0(i)$ is defined by (3.2), from (3.1), $eq_0(i)$ holds for $i = 0, 1, \ldots$ Using Property (d), $eq_\tau(i)$ holds for $i = 0, 1, \ldots$ It immediately follows that for such values of x_{i-1}, …, x_{i-r}, u_i, …, u_{i-p}, $y_{i+\tau}$, …, y_{i-t}, $eq_\tau(i)$ has a unique solution x_i, $i = 0, 1, \ldots$ Thus x_0 is uniquely determined by the initial state s_0 and the output sequence $y_0 \ldots y_\tau$. Therefore, M is weakly invertible with delay τ. □

Lemma 3.1.2. *Assume that for any x_i, …, x_{i-r}, u_i, …, u_{i-p}, $y_{i+\tau}$, …, y_{i-t}, if they satisfy the equation $eq_\tau(i)$ then $x_i = f_\tau^*$ ($x(i-1, r)$, $u(i, p+1)$, $y(i+\tau, \tau+t+1)$). For any state $s_0 = \langle y(-1, t), u(0, p+1), x(-1, r) \rangle$ and any input $x_0 x_1 \ldots$ of M, if*

$$y_0 y_1 \ldots = \lambda(s_0, x_0 x_1 \ldots),$$

then

$$\lambda^*(\langle x(-1, r), u(0, p+1), y(\tau - 1, \tau + t) \rangle, y_\tau y_{\tau+1} \ldots) = x_0 x_1 \ldots$$

Proof. Suppose that $y_0y_1\ldots = \lambda(s_0, x_0x_1\ldots)$. From the definition of M, we have

$$\begin{aligned} y_i &= f(y(i-1,t), u(i,p+1), x(i,r+1)),\\ u_{i+1} &= g(y(i-1,t), u(i,p+1), x(i,r+1)),\\ &\quad i = 0,1,\ldots \end{aligned} \tag{3.4}$$

From the proof of Theorem 3.1.3, $eq_\tau(i)$ holds for $i = 0,1,\ldots$ Using the hypothesis of the lemma, for any x_i, …, x_{i-r}, u_i, …, u_{i-p}, $y_{i+\tau}$, …, y_{i-t}, if they satisfy the equation $eq_\tau(i)$ then $x_i = f_\tau^*$ ($x(i-1,r)$, $u(i,p+1)$, $y(i+\tau, \tau+t+1)$). Thus for such values of x_i,…, x_{i-r}, u_i, …, u_{i-p}, $y_{i+\tau}$, …, y_{i-t}, we have

$$x_i = f_\tau^*(x(i-1,r), u(i,p+1), y(i+\tau, \tau+t+1)),\ i = 0,1,\ldots \tag{3.5}$$

Denote $y'_{j-\tau} = y_j$ for any j. Using (3.5) and (3.4), it immediately follows that

$$\begin{aligned} x_i &= f_\tau^*(x(i-1,r), u(i,p+1), y'(i,\tau+t+1)),\\ u_{i+1} &= g(y'(i-\tau-1,t), u(i,p+1), x(i,r+1)),\\ &\quad i = 0,1,\ldots \end{aligned}$$

From the definition of M^*, we have

$$\begin{aligned} x_0x_1\ldots &= \lambda^*(\langle x(-1,r), u(0,p+1), y'(-1,\tau+t)\rangle, y'_0y'_1\ldots)\\ &= \lambda^*(\langle x(-1,r), u(0,p+1), y(\tau-1,\tau+t)\rangle, y_\tau y_{\tau+1}\ldots). \end{aligned}$$ □

Theorem 3.1.4. *Assume that each state of M has a predecessor state and that for any x_i, …, x_{i-r}, u_i, …, u_{i-p}, $y_{i+\tau}$, …, y_{i-t}, if they satisfy the equation $eq_\tau(i)$ then $x_i = f_\tau^*(x(i-1,r), u(i,p+1), y(i+\tau,\tau+t+1))$. Then M^* is a weak inverse with delay τ of M.*

Proof. For any state $s_0 = \langle y(-1,t), u(0,p+1), x(-1,r)\rangle$ of M, since any state of M has a predecessor state, there exist $x_{-r-1},\ldots,x_{-r-\tau} \in X$, $u_{-p-1},\ldots,u_{-p-\tau} \in U$, $y_{-t-1},\ldots,y_{-t-\tau} \in Y$ such that

$$\begin{aligned} \lambda(s_{-\tau}, x_{-\tau}\ldots x_{-1}) &= y_{-\tau}\ldots y_{-1},\\ \delta(s_{-\tau}, x_{-\tau}\ldots x_{-1}) &= s_0, \end{aligned}$$

where

$$s_{-\tau} = \langle y(-\tau-1,t), u(-\tau,p+1), x(-\tau-1,r)\rangle.$$

For any input $x_0x_1\ldots$ of M, let

$$y_0 y_1 \ldots = \lambda(s_0, x_0 x_1 \ldots).$$

Then

$$y_{-\tau} \ldots y_{-1} y_0 y_1 \ldots = \lambda(s_{-\tau}, x_{-\tau} \ldots x_{-1} x_0 x_1 \ldots).$$

From Lemma 3.1.2, we have

$$\lambda^*(s_0^*, y_0 y_1 \ldots) = x_{-\tau} \ldots x_{-1} x_0 x_1 \ldots,$$

where

$$s_0^* = \langle x(-\tau - 1, r), u(-\tau, p+1), y(-1, \tau + t) \rangle.$$

It immediately follows that s_0^* matches s_0 with delay τ. Thus M^* is a weak inverse with delay τ of M. □

Corollary 3.1.2. *Assume that $t = 0$ and that for any parameters x_0, …, x_{-r+1}, u_0, …, u_{-p+1}, $g(u_0, \ldots, u_{-p}, x_0, \ldots, x_{-r})$ as a function of the variables u_{-p} and x_{-r} is surjective.*[1] *Assume that for any x_i, …, x_{i-r}, u_i, …, u_{i-p}, $y_{i+\tau}$, …, y_{i-t}, if they satisfy the equation $eq_\tau(i)$ then $x_i = f_\tau^*(x(i-1, r), u(i, p+1), y(i+\tau, \tau+t+1))$. Then M^* is a weak inverse with delay τ of M.*

Proof. Since $t = 0$ and for any parameters x_0, …, x_{-r+1}, u_0, …, u_{-p+1}, $g(u_0, \ldots, u_{-p}, x_0, \ldots, x_{-r})$ as a function of the variables u_{-p} and x_{-r} is surjective, it is easy to show that each state of M has a predecessor state. From Theorem 3.1.4, M^* is a weak inverse with delay τ of M. □

For any finite automaton $M' = \langle Y, X, S', \delta', \lambda' \rangle$ so that $S' = \bar{S} \times Y^k$ for some $k \geqslant 0$ and $\delta'(\langle s, y_{-1}, \ldots, y_{-k} \rangle, y_0)$ is in the form $\langle s', y_0, \ldots, y_{-k+1} \rangle$, the finite automaton $M'' = \langle Y, X, \bar{S} \times Y^k \times N_\tau, \delta'', \lambda'' \rangle$ is called the *τ-stay* of M', where

$$N_\tau = \{0, 1, \ldots, \tau\},$$

$$\delta''(\langle s, y_{-1}, \ldots, y_{-k}, c \rangle,\ y_0) = \begin{cases} \langle s, y_0, \ldots, y_{-k+1}, c+1 \rangle, & \text{if } c < \tau, \\ \langle \delta'(\langle s, y_{-1}, \ldots, y_{-k} \rangle, y_0), c \rangle, & \text{if } c = \tau, \end{cases}$$

$$\lambda''(\langle s, y_{-1}, \ldots, y_{-k}, c \rangle,\ y_0) = \lambda'(\langle s, y_{-1}, \ldots, y_{-k} \rangle, y_0).$$

From the definition of τ-stay, it is easy to verify the following lemma.

Lemma 3.1.3. *Assume that M'' is the τ-stay of M'. For any state s' of M', the state $\langle s', \tau \rangle$ of M'' and s' are equivalent.*

[1] Precisely speaking, $g(u_0, \ldots, u_{-p}, x_0, \ldots, x_{-r})$ as a function of the variables u_{-p} and x_{-r} means the restriction of g on the set $\{ (u_0, \ldots, u_{-p}, x_0, \ldots, x_{-r}) : u_{-p} \in U, x_{-r} \in X \}$.

From Lemma 3.1.3 and the definition of τ-stay, it is easy to prove the following lemma.

Lemma 3.1.4. *Assume that M'' is the τ-stay of M'. For any state $s'' = \langle s, y_{-1}, \ldots, y_{-k}, 0\rangle$ of M'' and any $y_0, y_1, \ldots \in Y$, there are $x_0, \ldots, x_{\tau-1} \in X$ such that*

$$\lambda''(s'', y_0y_1\ldots) = x_0\ldots x_{\tau-1}\lambda'(s', y_\tau y_{\tau+1}\ldots),$$

where $s' = \langle s, y_{\tau-1}, \ldots, y_{\tau-k}\rangle$.

Theorem 3.1.5. *Assume that for any $x_i, \ldots, x_{i-r}, u_i, \ldots, u_{i-p}, y_{i+\tau}, \ldots, y_{i-t}$, if they satisfy the equation $eq_\tau(i)$ then $x_i = f^*_\tau(x(i-1,r), u(i,p+1), y(i+\tau, \tau+t+1))$. Let M'' be the τ-stay of M^*. Then M'' is a weak inverse with delay τ of M. Moreover, for any state $s_0 = \langle y(-1,t), u(0,p+1), x(-1,r)\rangle$ of M, the state $s'' = \langle x(-1,r), u(0,p+1), y(-1,\tau+t), 0\rangle$ of M'' matches s_0 with delay τ, for any $y_{-t-1}, \ldots, y_{-\tau-t}$ in Y.*

Proof. For the state $s_0 = \langle y(-1,t), u(0,p+1), x(-1,r)\rangle$ and any input $x_0x_1\ldots$ of M, let

$$y_0y_1\ldots = \lambda(s_0, x_0x_1\ldots).$$

From Lemma 3.1.2,

$$\lambda^*(s^*_\tau, y_\tau y_{\tau+1}\ldots) = x_0x_1\ldots,$$

where

$$s^*_\tau = \langle x(-1,r), u(0,p+1), y(\tau-1, \tau+t)\rangle.$$

Using Lemma 3.1.4, there are $x'_0, \ldots, x'_{\tau-1} \in X$ such that

$$\lambda''(s'', y_0y_1\ldots) = x'_0\ldots x'_{\tau-1}\lambda^*(s^*_\tau, y_\tau y_{\tau+1}\ldots).$$

It follows that

$$\lambda''(s'', y_0y_1\ldots) = x'_0\ldots x'_{\tau-1}x_0x_1\ldots$$

Thus, s'' matches s_0 with delay τ. □

3.2 Generation of Finite Automata with Invertibility

Let X and U be two finite nonempty sets. Let m be a positive integer, and Y a column vector space of dimension m over a finite commutative ring R with identity.

For any integer i, we use x_i, u_i and y_i to denote elements in X, U and Y, respectively. We use $R[y_{i+k}, \ldots, y_{i-t}]$ to denote the polynomial ring consisting of all polynomials of components of $y_{i+k}, \ldots, y_{i-t}$ with coefficients in R.

Let r, t and τ be nonnegative integers with $\tau \leqslant r$, and p an integer with $p \geqslant -1$. Let f_k and f'_k be two single-valued mappings from $X^{r+1} \times U^{p+1} \times Y^{k+t+1}$ to Y for any nonnegative integer k. We use $eq_k(i)$ to denote the equation

$$f_k(x(i, r+1), u(i, p+1), y(i+k, k+t+1)) = 0$$

and use $eq'_k(i)$ to denote the equation

$$f'_k(x(i, r+1), u(i, p+1), y(i+k, k+t+1)) = 0.$$

Let $\psi^l_{\mu\nu}$ be a column vector of dimension l of which each component is a single-valued mapping from $U^{\mu+1} \times X^{\nu+1}$ to R, for integers $\mu \geqslant -1$, $\nu \geqslant 0$ and $l \geqslant 1$. For any integers $h \geqslant 0$ and i, let

$$\psi^{lh}_{\mu\nu}(u, x, i) = \begin{bmatrix} \psi^l_{\mu\nu}(u(i, \mu+1), x(i, \nu+1)) \\ \vdots \\ \psi^l_{\mu\nu}(u(i-h, \mu+1), x(i-h, \nu+1)) \end{bmatrix}.$$

Assume that f_k can be expressed in the following form

$$\begin{aligned} &f_k(x(i, r+1), u(i, p+1), y(i+k, k+t+1)) \\ &\quad = \sum_{j=0}^{r} G_{jk}(y(i+k, k+t+1)) \psi^l_{\mu\nu}(u(i-j, \mu+1), x(i-j, \nu+1)), \end{aligned} \tag{3.6}$$

where $G_{jk}(y(i+k, k+t+1))$ is an $m \times l$ matrix over $R[y_{i+k}, \ldots, y_{i-t}]$, $0 \leqslant j \leqslant r$. Let

$$G_k(i) = [G_{0k}(y(i+k, k+t+1)), \ldots, G_{rk}(y(i+k, k+t+1))].$$

Then (3.6) can be rewritten as follows

$$f_k(x(i, r+1), u(i, p+1), y(i+k, k+t+1)) = G_k(i) \psi^{lr}_{\mu\nu}(u, x, i).$$

Notice that the right side of the above equation does not depend on x_{i-j} for $j > r$ and u_{i-j} for $j > p$. The matrix $G_k(i)$ in such an expression is not unique for general $\psi^l_{\mu\nu}$. $G_k(i)$ is referred to as a *coefficient matrix* of f_k or $eq_k(i)$. Similarly, assume that f'_k can be expressed in the form

$$\begin{aligned} &f'_k(x(i, r+1), u(i, p+1), y(i+k, k+t+1)) \\ &\quad = \sum_{j=0}^{r} G'_{jk}(y(i+k, k+t+1)) \psi^l_{\mu\nu}(u(i-j, \mu+1), x(i-j, \nu+1)) \end{aligned}$$

or

$$f'_k(x(i,r+1),u(i,p+1),y(i+k,k+t+1)) = G'_k(i)\psi^{lr}_{\mu\nu}(u,x,i),$$

where $G'_{jk}(y(i+k,k+t+1))$ is an $m \times l$ matrix over $R[y_{i+k},\ldots,y_{i-t}]$, $0 \leqslant j \leqslant r$, and

$$G'_k(i) = [G'_{0k}(y(i+k,k+t+1)),\ldots,G'_{rk}(y(i+k,k+t+1))];$$

$G'_k(i)$ is referred to as a *coefficient matrix* of f'_k or $eq'_k(i)$.

In the case where the transformation φ_k on $eq_k(i)$ is multiplying two sides of $eq_k(i)$ on the left by a matrix polynomial $P_k(y(i+k,k+t+1))$, we also denote $eq_k(i) \overset{R_a[P_k]}{\longrightarrow} eq'_k(i)$ instead of $eq_k(i) \overset{R_a[\varphi_k]}{\longrightarrow} eq'_k(i)$. In this case, the rule R_a can be restated as follows.

Rule R_a : Let $G_k(i)$ be an $m \times l(r+1)$ matrix over $R[y_{i+k},\ldots,y_{i-t}]$. Let $P_k(y(i+k,k+t+1))$ be an invertible $m \times m$ matrix over $R[y_{i+k},\ldots,y_{i-t}]$. Assume that

$$G'_k(i) = P_k(y(i+k,k+t+1))G_k(i).$$

$G'_k(i)$ is said to be *obtained from* $G_k(i)$ *by Rule* R_a *using* P_k, denoted by

$$G_k(i) \overset{R_a[P_k]}{\longrightarrow} G'_k(i).$$

Clearly, if $G_k(i)$ and $G'_k(i)$ are coefficient matrices of $eq_k(i)$ and $eq'_k(i)$, respectively, then $eq_k(i) \overset{R_a[P_k]}{\longrightarrow} eq'_k(i)$ and $G_k(i) \overset{R_a[P_k]}{\longrightarrow} G'_k(i)$ are the same.

The Rule R_b can be restated as follows.

Rule R_b : Let $G'_k(i)$ be an $m \times l(r+1)$ matrix over $R[y_{i+k},\ldots,y_{i-t}]$. Assume that the submatrix of the last $m - r_{k+1}$ rows and the first l columns of $G'_k(i)$ has zeros whenever $r_{k+1} < m$. Let $G_{k+1}(i)$ be the matrix obtained by shifting the last $m - r_{k+1}$ rows of $G'_k(i)$ l columns to the left entering zeros to the right and replacing each variable y_j of elements in the last $m - r_{k+1}$ rows by y_{j+1} for any j. $G_{k+1}(i)$ is said to be *obtained from* $G'_k(i)$ *by Rule* R_b, denoted by

$$G'_k(i) \overset{R_b[r_{k+1}]}{\longrightarrow} G_{k+1}(i).$$

Clearly, if $G_{k+1}(i)$ and $G'_k(i)$ are coefficient matrices of $eq_{k+1}(i)$ and $eq'_k(i)$, respectively, then $eq'_k(i) \overset{R_b[r_{k+1}]}{\longrightarrow} eq_{k+1}(i)$ and $G'_k(i) \overset{R_b[r_{k+1}]}{\longrightarrow} G_{k+1}(i)$ are the same.

It is easy to see that the condition that for any parameters x_{i-1}, …, $x_{i-\nu}$, u_i, …, $u_{i-\mu}$, $y_{i+\tau}$, …, y_{i-t},

$$G_{0\tau}(y(i+\tau,\tau+t+1))\psi^{l}_{\mu\nu}(u(i,\mu+1),x(i,\nu+1))$$

as a function of the variable x_i is a surjection yields the condition in Theorem 3.1.1 that for any parameters x_{i-1}, …, x_{i-r}, u_i, …, u_{i-p}, $y_{i+\tau}$, …, y_{i-t},

$eq_\tau(i)$ has a solution x_i(the reverse proposition is also true in some case, see [135]). [1] And for any parameters $x_{i-1}, \ldots, x_{i-\nu}, u_i, \ldots, u_{i-\mu}, y_{i+\tau}, \ldots, y_{i-t}$,

$$G_{0\tau}(y(i+\tau, \tau+t+1))\psi^l_{\mu\nu}(u(i, \mu+1), x(i, \nu+1))$$

as a function of the variable x_i is an injection, if and only if the condition in Theorem 3.1.3 holds, that is, for any parameters $x_{i-1}, \ldots, x_{i-r}, u_i, \ldots, u_{i-p}$, $y_{i+\tau}, \ldots, y_{i-t}$, $eq_\tau(i)$ has at most one solution x_i.

We now modify the above rules R_a and R_b to deal with incomplete specified matrices.

Let G be an $m \times l(\tau+1)$ incomplete specified matrix. Let $0 \leqslant k \leqslant \tau$. If there exist $r_0, r_1 \ldots, r_k$, $0 = r_0 \leqslant r_1 \leqslant \cdots \leqslant r_k \leqslant m$, such that whenever $r_i < r_{i+1}$ in row $r_i + 1$ to row r_{i+1} of G elements of the first $l(\tau+1-i)$ columns are defined and elements of the last li columns are undefined for $i = 0, 1, \ldots, k$, where $r_{k+1} = m$, G is called an (l, k)-*echelon matrix*. It is easy to see that $r_0, r_1, \ldots, r_k$ satisfying the above condition are unique. r_i is referred to as the *i-height* of G, $0 \leqslant i \leqslant k$.

Example 3.2.1. $\tau = 5$, $k = 3$. The matrix G_3

$$G_3 = \begin{bmatrix} G_{10} & G_{11} & G_{12} & G_{13} & G_{14} & G_{15} \\ G_{20} & G_{21} & G_{22} & G_{23} & G_{24} & * \\ G_{30} & G_{31} & G_{32} & G_{33} & * & * \\ G_{40} & G_{41} & G_{42} & * & * & * \end{bmatrix}$$

is an $(l, 3)$-echelon matrix, where G_{ij} is an $(r_i - r_{i-1}) \times l$ complete specified matrix, $i = 1, 2, 3, 4$, $j = 0, 1, \ldots, 6-i$, $*$ stands for "undefined", and $0 = r_0 \leqslant r_1 \leqslant r_2 \leqslant r_3 \leqslant r_4 = m$. Clearly, r_i is the i-height of G_3, $0 \leqslant i \leqslant 3$.

Notice that if G is an (l, k)-echelon matrix with i-height r_i for $0 \leqslant i \leqslant k$, then G is an $(l, k+1)$-echelon matrix with i-height r_i for $0 \leqslant i \leqslant k$ and with $(k+1)$-height m.

Rule R_a(modified): Let $G_k(i)$ be an $m \times l(\tau+1)$ (l, k)-echelon matrix over $R[y_{i+k}, \ldots, y_{i-t}]$, and $0 \leqslant k \leqslant \tau$. Let $P_k(y(i+k, k+t+1))$ be an invertible $m \times m$ matrix over $R[y_{i+k}, \ldots, y_{i-t}]$ in the form

$$P_k(y(i+k, k+t+1)) = \begin{bmatrix} E_{r_k} & 0 \\ P_{k1}(y(i+k, k+t+1)) & P_{k2}(y(i+k, k+t+1)) \end{bmatrix},$$

where E_{r_k} is the $r_k \times r_k$ identity matrix, r_k is the k-height of $G_k(i)$. Assume that

$$G'_k(i) = P_k(y(i+k, k+t+1))G_k(i).$$

[1] Precisely speaking, $G_{0\tau}(y_{i+\tau}, \ldots, y_{i-t})\psi^l_{\mu\nu}(u_i, \ldots, u_{i-\mu}, x_i, \ldots, x_{i-\nu})$ as a function of the variable x_i means its restriction on the set $\{(y_{i+\tau}, \ldots, y_{i-t}, u_i, \ldots, u_{i-\mu}, x_i, \ldots, x_{i-\nu}) \mid x_i \in X\}$.

$G'_k(i)$ is said to be *obtained from* $G_k(i)$ *by Rule* R_a *using* P_k, denoted by

$$G_k(i) \overset{R_a[P_k]}{\longrightarrow} G'_k(i).$$

In computing elements of $P_k(y(i+k, k+t+1))G_k(i)$, we define $u \cdot u = u+u = u$, $x \cdot u = u \cdot x = u$, $0 \cdot u = u \cdot 0 = 0$ and $y+u = u+y = u$, where u stands for undefined symbol, $x(\neq 0)$ and y are any elements in $R[y_{i+k}, \ldots, y_{i-t}]$.

Let $G_k(i)$ be an $m \times l(\tau+1)$ (l,k)-echelon matrix over $R[y_{i+k}, \ldots, y_{i-t}]$, and $0 \leqslant k \leqslant \tau$. If $G_k(i) \overset{R_a[P_k]}{\longrightarrow} G'_k(i)$, then $G'_k(i)$ is also an $m \times l(\tau+1)$ (l,k)-echelon matrix over $R[y_{i+k}, \ldots, y_{i-t}]$, and the j-height of $G'_k(i)$ is the same as the j-height of $G_k(i)$ for any j, $0 \leqslant j \leqslant k$.

Rule R_b(modified): Let $G'_k(i)$ be an $m \times l(\tau+1)$ (l,k)-echelon matrix over $R[y_{i+k}, \ldots, y_{i-t}]$, and $0 \leqslant k < \tau$. Assume that for some r_{k+1}, $m \geqslant r_{k+1} \geqslant r_k$, the submatrix of the last $m - r_{k+1}$ rows and the first l columns of $G'_k(i)$ has zeros whenever $r_{k+1} < m$. Let $G_{k+1}(i)$ be the matrix obtained by shifting the last $m - r_{k+1}$ rows of $G'_k(i)$ l columns to the left entering "undefined" to the right and replacing each variable y_j of elements in the last $m - r_{k+1}$ rows by y_{j+1} for any j. $G_{k+1}(i)$ is said to be *obtained from* $G'_k(i)$ *by Rule* R_b, denoted by

$$G'_k(i) \overset{R_b[r_{k+1}]}{\longrightarrow} G_{k+1}(i).$$

Let $G'_k(i)$ be an $m \times l(\tau+1)$ (l,k)-echelon matrix over $R[y_{i+k}, \ldots, y_{i-t}]$. If $G'_k(i) \overset{R_b[r_{k+1}]}{\longrightarrow} G_{k+1}(i)$, then $G_{k+1}(i)$ is an $m \times l(\tau+1)$ $(l, k+1)$-echelon matrix over $R[y_{i+k+1}, \ldots, y_{i-t}]$ of which the j-height is the same as the j-height of $G'_k(i)$ for any j, $0 \leqslant j \leqslant k$ and the $(k+1)$-height is r_{k+1}.

Lemma 3.2.1. *Let* $eq_0(i)$ *be*

$$\sum_{j=0}^{r} G_{j0}(y(i, t+1))\psi^l_{\mu\nu}(u(i-j, \mu+1), x(i-j, \nu+1)) = 0,$$

and $G_0(i)$ *the* $m \times l(\tau+1)$ $(l,0)$*-echelon matrix*

$$[G_{00}(y(i,t+1)), \ldots, G_{\tau 0}(y(i,t+1))],$$

$\tau \leqslant r$. *If*

$$G_k(i) \overset{R_a[P_k]}{\longrightarrow} G'_k(i),\ G'_k(i) \overset{R_b[r_{k+1}]}{\longrightarrow} G_{k+1}(i), \quad k = 0, 1, \ldots, \tau-1$$

is a modified R_a R_b *transformation sequence, then*

$$eq_k(i) \overset{R_a[P_k]}{\longrightarrow} eq'_k(i),\ eq'_k(i) \overset{R_b[r_{k+1}]}{\longrightarrow} eq_{k+1}(i), \quad k = 0, 1, \ldots, \tau-1$$

is an R_a R_b *transformation sequence, and the first* l *columns of* $G_\tau(i)$ *and the first* l *columns of the coefficient matrix of* $eq_\tau(i)$ *are the same.*

Proof. We use $\bar{G}_k(i)$ and $\bar{G}'_k(i)$ to denote the coefficient matrices of $eq_k(i)$ and $eq'_k(i)$, respectively. It is sufficient to show that

$$\bar{G}_k(i) \overset{R_a[P_k]}{\longrightarrow} \bar{G}'_k(i),\ \bar{G}'_k(i) \overset{R_b[r_{k+1}]}{\longrightarrow} \bar{G}_{k+1}(i),\quad k = 0, 1, \ldots, \tau - 1$$

is an R_a R_b transformation sequence in the original sense and that corresponding matrices (between G_k and $\bar{G}_k$, and between G'_k and $\bar{G}'_k$) are compatible.[1] This can be proved by simple induction as follows. It is evident that $G_0(i)$ and $\bar{G}_0(i)$ are compatible. Suppose that $G_k(i)$ and $\bar{G}_k(i)$ are compatible and $G_k(i) \overset{R_a[P_k]}{\longrightarrow} G'_k(i)$ in the modified sense for $k < \tau$. Letting $\bar{G}'_k(i) = P_k\bar{G}_k(i)$, we have $\bar{G}_k(i) \overset{R_a[P_k]}{\longrightarrow} \bar{G}'_k(i)$ in the original sense. Since $G_k(i)$ is an (l, k)-echelon matrix and P_k has special shape, it is easy to verify that $G'_k(i)$ and $\bar{G}'_k(i)$ are compatible. Suppose that $G'_k(i)$ and $\bar{G}'_k(i)$ are compatible and $G'_k(i) \overset{R_b[r_{k+1}]}{\longrightarrow} G_{k+1}(i)$ in the modified sense for $k < \tau$. From $G'_k(i) \overset{R_b[r_{k+1}]}{\longrightarrow} G_{k+1}(i)$, the submatrix of the last $m - r_{k+1}$ rows and the first l columns of $G'_k(i)$ has zeros whenever $r_{k+1} < m$. Since $G'_k(i)$ and $\bar{G}'_k(i)$ are compatible, it follows that the submatrix of the last $m - r_{k+1}$ rows and the first l columns of $\bar{G}'_k(i)$ has zeros whenever $r_{k+1} < m$. Thus we have $\bar{G}'_k(i) \overset{R_b[r_{k+1}]}{\longrightarrow} \bar{G}_{k+1}(i)$ in the original sense. Clearly, $G_{k+1}(i)$ and $\bar{G}_{k+1}(i)$ are compatible. □

Rule R_a^{-1} : Let $G'_k(i)$ be an $m \times l(\tau + 1)$ (l, k)-echelon matrix over $R[y_{i+k}, \ldots, y_{i-t}]$, and $0 \leqslant k \leqslant \tau$. Let $P'_k(y(i + k, k + t + 1))$ be an invertible $m \times m$ matrix over $R[y_{i+k}, \ldots, y_{i-t}]$ in the form

$$P'_k(y(i + k, k + t + 1)) = \begin{bmatrix} E_{r_k} & 0 \\ P'_{k1}(y(i + k, k + t + 1)) & P'_{k2}(y(i + k, k + t + 1)) \end{bmatrix},$$

where E_{r_k} is the $r_k \times r_k$ identity matrix, r_k is the k-height of $G'_k(i)$. Let

$$G_k(i) = P'_k(y(i + k, k + t + 1))G'_k(i).$$

$G_k(i)$ is said to be *obtained from* $G'_k(i)$ *by Rule* R_a^{-1} *using* P'_k, denoted by

$$G'_k(i) \overset{R_a^{-1}[P'_k]}{\longrightarrow} G_k(i).$$

In computing elements of $P'_k(y(i + k, k + t + 1))G'_k(i)$, we define $u \cdot u = u + u = u$, $x \cdot u = u \cdot x = u$, $0 \cdot u = u \cdot 0 = 0$ and $y + u = u + y = u$, where u stands for undefined symbol, $x(\neq 0)$ and y are any elements in $R[y_{i+k}, \ldots, y_{i-t}]$.

Let $G'_k(i)$ be an $m \times l(\tau + 1)$ (l, k)-echelon matrix over $R[y_{i+k}, \ldots, y_{i-t}]$, and $0 \leqslant k \leqslant \tau$. If $G'_k(i) \overset{R_a^{-1}[P'_k]}{\longrightarrow} G_k(i)$, then $G_k(i)$ is also an $m \times l(\tau + 1)$

[1] Two matrices are compatible, if for any position (i, j), elements of the two matrices are the same whenever they are defined.

(l,k)-echelon matrix over $R[y_{i+k},\ldots,y_{i-t}]$, and the j-height of $G_k(i)$ is the same as the j-height of $G'_k(i)$ for any j, $0 \leqslant j \leqslant k$.

Rule R_b^{-1} : Let $G_{k+1}(i)$ be an $m \times l(\tau+1)$ $(l,k+1)$-echelon matrix over $R[y_{i+k+1},\ldots,y_{i-t}]$, and $0 \leqslant k < \tau$. Assume that elements in the first r_{k+1} rows of $G_{k+1}(i)$ do not depend on y_{i+k+1} and that elements in the last $m - r_{k+1}$ rows of $G_{k+1}(i)$ do not depend on y_{i-t}, where r_{k+1} is the $(k+1)$-height of $G_{k+1}(i)$. Let $G'_k(i)$ be the matrix obtained by shifting the last $m-r_{k+1}$ rows of $G_{k+1}(i)$ l columns to the right entering zeros to the left and replacing each variable y_j of elements in the last $m-r_{k+1}$ rows by y_{j-1} for any j. $G'_k(i)$ is said to be *obtained from* $G_{k+1}(i)$ *by Rule* R_b^{-1}, denoted by

$$G_{k+1}(i) \overset{R_b^{-1}[r_{k+1}]}{\longrightarrow} G'_k(i).$$

Let $G_{k+1}(i)$ be an $m\times l(\tau+1)$ $(l,k+1)$-echelon matrix over $R[y_{i+k+1},\ldots,y_{i-t}]$, and $0 \leqslant k < \tau$. If $G_{k+1}(i) \overset{R_b^{-1}[r_{k+1}]}{\longrightarrow} G'_k(i)$, then $G'_k(i)$ is an $m\times l(\tau+1)$ (l,k)-echelon matrix over $R[y_{i+k},\ldots,y_{i-t}]$, and the j-height of $G'_k(i)$ is the same as the j-height of $G_{k+1}(i)$ for any j, $0 \leqslant j < k$, and the k-height of $G'_k(i)$ is the sum of the k-height and the $(k+1)$-height of $G_{k+1}(i)$.

Lemma 3.2.2.

$$G_k(i) \overset{R_a[P_k]}{\longrightarrow} G'_k(i),\ G'_k(i) \overset{R_b[r_{k+1}]}{\longrightarrow} G_{k+1}(i),\quad k=0,1,\ldots,\tau-1 \tag{3.7}$$

is a modified R_a R_b *transformation sequence if and only if*

$$G_{k+1}(i) \overset{R_b^{-1}[r_{k+1}]}{\longrightarrow} G'_k(i),\ G'_k(i) \overset{R_a^{-1}[P_k^{-1}]}{\longrightarrow} G_k(i),\ k=\tau-1,\ldots,1,0 \tag{3.8}$$

is an R_a^{-1} R_b^{-1} *transformation sequence.*

Proof. For any k, $0 \leqslant k < \tau$ and any invertible matrix $P_k(y(i+k,k+t+1))$ over $R[y_{i+k},\ldots,y_{i-t}]$, it is easy to see that $P_k(y(i+k,k+t+1))$ is in the form

$$P_k(y(i+k,k+t+1)) = \begin{bmatrix} E_{r_k} & 0 \\ P_{k1}(y(i+k,k+t+1)) & P_{k2}(y(i+k,k+t+1)) \end{bmatrix}$$

if and only if $P_k^{-1}(y(i+k,k+t+1))$ is in the form

$$P_k^{-1}(y(i+k,k+t+1)) = \begin{bmatrix} E_{r_k} & 0 \\ P'_{k1}(y(i+k,k+t+1)) & P'_{k2}(y(i+k,k+t+1)) \end{bmatrix}.$$

From the definitions of R_a and R_a^{-1}, it follows that $G_k(i) \overset{R_a[P_k]}{\longrightarrow} G'_k(i)$ if and only if $G'_k(i) \overset{R_a[P_k^{-1}]}{\longrightarrow} G_k(i)$. Similarly, from the definitions of R_b and R_b^{-1}, it is easy to verify that $G'_k(i) \overset{R_b[r_{k+1}]}{\longrightarrow} G_{k+1}(i)$ if and only if $G_{k+1}(i) \overset{R_b^{-1}[r_{k+1}]}{\longrightarrow} G'_k(i)$. Therefore, (3.7) is a modified R_a R_b transformation sequence if and only if (3.8) is an R_a^{-1} R_b^{-1} transformation sequence. □

Lemma 3.2.3. *Assume that* $eq_0(i)$ *is*

$$\sum_{j=0}^{r} G_{j0}(y(i,t+1))\psi^{l}_{\mu\nu}(u(i-j,\mu+1),x(i-j,\nu+1))=0$$

and $G_0(i)=[G_{00}(y(i,t+1)),\ldots,G_{\tau 0}(y(i,t+1))]$, $\tau\leqslant r$. *For any* $m\times l(\tau+1)$ (l,τ)*-echelon matrix* $G_\tau(i)$ *over* $R[y_{i+\tau},\ldots,y_{i-t}]$, *if*

$$G_{k+1}(i)\overset{R_b^{-1}[r_{k+1}]}{\longrightarrow}G'_k(i),\ G'_k(i)\overset{R_a^{-1}[P'_k]}{\longrightarrow}G_k(i),\ k=\tau-1,\ldots,1,0 \tag{3.9}$$

is an R_a^{-1} R_b^{-1} *transformation sequence, then*

$$eq_k(i)\overset{R_a[P_k]}{\longrightarrow}eq'_k(i),\ eq'_k(i)\overset{R_b[r_{k+1}]}{\longrightarrow}eq_{k+1}(i),\quad k=0,1,\ldots,\tau-1 \tag{3.10}$$

is an R_a R_b *transformation sequence and the first* l *columns of* $G_\tau(i)$ *and the first* l *columns of the coefficient matrix of* $eq_\tau(i)$ *are the same, where* $P_k(y(i+k,k+t+1))=(P'_k(y(i+k,k+t+1)))^{-1}$, $0\leqslant k<\tau$.

Proof. Assume that (3.9) is an R_a^{-1} R_b^{-1} transformation sequence. From Lemma 3.2.2,

$$G_k(i)\overset{R_a[P_k]}{\longrightarrow}G'_k(i),\ G'_k(i)\overset{R_b[r_{k+1}]}{\longrightarrow}G_{k+1}(i),\quad k=0,1,\ldots,\tau-1$$

is a modified R_a R_b transformation sequence. From Lemma 3.2.1, (3.10) is an R_a R_b transformation sequence, and the first l columns of $G_\tau(i)$ and the first l columns of the coefficient matrix of $eq_\tau(i)$ are the same. □

Theorem 3.2.1. *Let* f *and* g *be two single-valued mappings from* $Y^t\times U^{p+1}\times X^{r+1}$ *to* Y *and* U, *respectively. Let* $M=\langle X,Y,Y^t\times U^{p+1}\times X^r,\delta,\lambda\rangle$ *be a finite automaton defined by*

$$\begin{aligned}&\delta(\langle y(i-1,t),u(i,p+1),x(i-1,r)\rangle,x_i)\\&\quad=\langle y(i,t),u(i+1,p+1),x(i,r)\rangle,\\&\lambda(\langle y(i-1,t),u(i,p+1),x(i-1,r)\rangle,x_i)=y_i,\end{aligned}$$

where

$$\begin{aligned}y_i&=f(y(i-1,t),u(i,p+1),x(i,r+1)),\\u_{i+1}&=g(y(i-1,t),u(i,p+1),x(i,r+1)).\end{aligned}$$

Let $eq_0(i)$ *be the equation*

$$-y_i+f(y(i-1,t),u(i,p+1),x(i,r+1))=0.$$

Assume that

$$-y_i + f(y(i-1,t), u(i,p+1), x(i,r+1)) = \sum_{j=0}^{r} G_{j0}(y(i,t+1))\psi^l_{\mu\nu}(u(i-j,\mu+1), x(i-j,\nu+1))$$

and $G_0(i) = [G_{00}(y(i,t+1)), \ldots, G_{\tau 0}(y(i,t+1))]$, $\tau \leqslant r$.

(a) *If there exist an* $m\times l(\tau+1)$ (l,τ)*-echelon matrix* $G_\tau(i)$ *over* $R[y_{i+\tau}, \ldots, y_{i-t}]$ *and an* R_a^{-1} R_b^{-1} *transformation sequence* (3.9) *such that for any parameters* x_{i-1}, …, $x_{i-\nu}$, u_i, …, $u_{i-\mu}$, $y_{i+\tau}$, …, y_{i-t},

$$G_{0\tau}(y(i+\tau, \tau+t+1))\psi^l_{\mu\nu}(u(i,\mu+1), x(i,\nu+1)) \tag{3.11}$$

as a function of the variable x_i *is a surjection, then* M *is a weak inverse with delay* τ.

(b) *If there exist an* $m\times l(\tau+1)$ (l,τ)*-echelon matrix* $G_\tau(i)$ *over* $R[y_{i+\tau}, \ldots, y_{i-t}]$ *and an* R_a^{-1} R_b^{-1} *transformation sequence* (3.9) *such that for any parameters* x_{i-1}, …, $x_{i-\nu}$, u_i, …, $u_{i-\mu}$, $y_{i+\tau}$, …, y_{i-t}, (3.11) *as a function of the variable* x_i *is an injection, then* M *is weakly invertible with delay* τ.

Proof. (a) Assume that (3.9) is an R_a^{-1} R_b^{-1} transformation sequence and that for any parameters x_{i-1}, …, $x_{i-\nu}$, u_i, …, $u_{i-\mu}$, $y_{i+\tau}$, …, y_{i-t}, (3.11) as a function of the variable x_i is a surjection. From Lemma 3.2.3, (3.10) is an R_a R_b transformation sequence, and the first l columns of $G_\tau(i)$ and the first l columns of the coefficient matrix of $eq_\tau(i)$ are the same, where $P_k(y(i+k, k+t+1)) = (P'_k(y(i+k, k+t+1)))^{-1}$, $0 \leqslant k < \tau$. Clearly, the condition that for any parameters x_{i-1}, …, $x_{i-\nu}$, u_i, …, $u_{i-\mu}$, $y_{i+\tau}$, …, y_{i-t}, (3.11) as a function of the variable x_i is a surjection yields the condition that for any parameters x_{i-1}, …, x_{i-r}, u_i, …, u_{i-p}, $y_{i+\tau}$, …, y_{i-t}, $eq_\tau(i)$ has a solution x_i. From Corollary 3.1.1, M is a weak inverse with delay τ.

(b) This part is similar to part (a) but using Theorem 3.1.3 instead of Corollary 3.1.1. □

We point out that the matrices $P_k(y(i+k, k+t+1))$ in the definition of R_a and $P'_k(y(i+k, k+t+1))$ in the definition of R_a^{-1} could be extended by replacing E_k in $P_k(y(i+k, k+t+1))$ and $P'_k(y(i+k, k+t+1))$ as an invertible quasi-lower-triangular matrix over $R[y_{i+k}, \ldots, y_{i-t}]$ of which the block at position (i,j) is an $(r_i - r_{i-1}) \times (r_j - r_{j-1})$ matrix. In this case, results in this section still hold.

3.3 Invertibility of Quasi-Linear Finite Automata

3.3.1 Decision Criteria

Assume that X and Y are column vector spaces over $GF(q)$ of dimension l and m, respectively. For any integer i, we use x_i (x'_i) and y_i (y'_i) to denote column vectors over $GF(q)$ of dimensions l and m, respectively. Assume that r and τ are two nonnegative integers with $\tau \leqslant r$.

Let $M = \langle X, Y, Y^t \times X^r, \delta, \lambda \rangle$ be an (r,t)-order memory finite automaton over $GF(q)$. If M is defined by

$$y_i = \sum_{j=0}^{\tau} B_j x_{i-j} + g(x(i-\tau-1, r-\tau), y(i-1,t)),$$
$$i = 0, 1, \ldots, \tag{3.12}$$

where B_j is an $m \times l$ matrix over $GF(q)$, $j = 0, 1, \ldots, \tau$, g is a single-valued mapping from $X^{r-\tau} \times Y^t$ to Y, M is said to be *τ-quasi-linear* over $GF(q)$.

In this section, for any k, $0 \leqslant k \leqslant \tau$, we use $eq_k(i)$ to denote an equation

$$\sum_{j=0}^{\tau-k} B_{jk} x_{i-j} - A_{-k} y_{i+k}$$
$$+ g_k(x(i-\tau+k-1, k+r-\tau), y(i+k-1, k+t)) = 0 \tag{3.13}$$

and $eq'_k(i)$ to denote an equation

$$\sum_{j=0}^{\tau-k} B'_{jk} x_{i-j} - A'_{-k} y_{i+k}$$
$$+ g'_k(x(i-\tau+k-1, k+r-\tau), y(i+k-1, k+t)) = 0, \tag{3.14}$$

where A_{-k}, A'_{-k}, B_{jk} and B'_{jk} are $m \times m$, $m \times m$, $m \times l$ and $m \times l$ matrices over $GF(q)$, respectively, and g_k and g'_k are two single-valued mappings from $X^{k+r-\tau} \times Y^{k+t}$ to Y.

$eq_k(i) \xrightarrow{R_a[P_k]} eq'_k(i)$ is said to be *linear over* $GF(q)$, if P_k is a matrix over $GF(q)$ and for some $r_{k+1} \geqslant 0$ the first r_{k+1} rows of B'_{0k} is linearly independent over $GF(q)$ and B'_{0k} has zeros in the last $m - r_{k+1}$ rows whenever $r_{k+1} < m$. $eq'_k(i) \xrightarrow{R_b[r_{k+1}]} eq_{k+1}(i)$ is said to be *linear over* $GF(q)$, if the first r_{k+1} rows of B'_{0k} is linearly independent over $GF(q)$. An R_a R_b transformation sequence

$$eq_k(i) \xrightarrow{R_a[P_k]} eq'_k(i),\ eq'_k(i) \xrightarrow{R_b[r_{k+1}]} eq_{k+1}(i), \quad k = 0, 1, \ldots, h \tag{3.15}$$

is said to be *linear over* $GF(q)$, if $eq_k(i) \xrightarrow{R_a[P_k]} eq'_k(i)$ and $eq'_k(i) \xrightarrow{R_b[r_{k+1}]} eq_{k+1}(i)$ are linear over $GF(q)$ for $k = 0, 1, \ldots, h$.

Lemma 3.3.1. *Let*

$$eq_k(i) \overset{R_a[P_k]}{\longrightarrow} eq'_k(i),\ eq'_k(i) \overset{R_b[r_{k+1}]}{\longrightarrow} eq_{k+1}(i),\quad k = 0, 1, \ldots, \tau - 1$$

be linear over $GF(q)$. If the rank of the matrix $[B_{00}, A_{-0}]$ is m, then the rank of the matrix $[B_{0\tau}, A_{-\tau}]$ is m.

Proof. For any k, $0 \leqslant k < \tau$, since $eq_k(i) \overset{R_a[P_k]}{\longrightarrow} eq'_k(i)$, the rank of $[B_{0k}, A_{-k}]$ and the rank of $[B'_{0k}, A'_{-k}]$ are the same. Since $eq'_k(i) \overset{R_b[r_{k+1}]}{\longrightarrow} eq_{k+1}(i)$ is linear over $GF(q)$, the first r_{k+1} rows of B'_{0k} is linearly independent and B'_{0k} has zeros in the last $m - r_{k+1}$ rows whenever $r_{k+1} < m$. Therefore, if the rank of $[B'_{0k}, A'_{-k}]$ is m, then the rank of the last $m - r_{k+1}$ rows of A'_{-k} is $m - r_{k+1}$ and the rank of $[B_{0,k+1}, A_{-k-1}]$ is m. By simple induction, the rank of $[B_{0\tau}, A_{-\tau}]$ is m. □

Let $eq_0(i)$ be the equation

$$\sum_{j=0}^{\tau} B_j x_{i-j} - y_i + g(x(i - \tau - 1, r - \tau), y(i - 1, t)) = 0, \tag{3.16}$$

that is, the equation (3.13) with $k = 0$, where $B_{j0} = B_j, j = 0, 1, \ldots, \tau$, A_{-0} is the identity matrix, and $g_0 = g$.

Theorem 3.3.1. *Let $eq_0(i)$ be* (3.16). *Let*

$$eq_k(i) \overset{R_a[P_k]}{\longrightarrow} eq'_k(i),\ eq'_k(i) \overset{R_b[r_{k+1}]}{\longrightarrow} eq_{k+1}(i),\quad k = 0, 1, \ldots, \tau - 1$$

be linear over $GF(q)$. Then M is a weak inverse with delay τ if and only if the rank of $B_{0\tau}$ is m.

Proof. if : Suppose that the rank of $B_{0\tau}$ is m. Then there is a right inverse matrix of $B_{0\tau}$, say $B_{0\tau}^{-1}$. Thus for any parameters x_{i-1}, ..., x_{i-r}, $y_{i+\tau}$, ..., y_{i-t}, $eq_\tau(i)$ has a solution x_i

$$x_i = B_{0\tau}^{-1} A_{-\tau} y_{i+\tau} - B_{0\tau}^{-1} g_\tau(x(i-1, r), y(i + \tau - 1, \tau + t)).$$

From Corollary 3.1.1, M is a weak inverse with delay τ.

only if : Suppose that M is a weak inverse with delay τ. Then there is a finite automaton M_1 such that M is a weak inverse with delay τ of M_1. Therefore, for any state s_1 of M_1 there is a state $s = \langle y(-1, t), x(-1, r)\rangle$ of M such that s matches s_1 with delay τ. For any input $y'_0\ y'_1\ \ldots$ of M_1, let

$$x_0 x_1 \ldots = \lambda_1(s_1, y'_0 y'_1 \ldots)$$

and

$$y_0 y_1 \ldots = \lambda(s, x_0 x_1 \ldots), \tag{3.17}$$

where λ_1 is the output function of M_1. Since s matches s_1 with delay τ, we have $y_{j+\tau} = y'_j, j = 0, 1, \ldots$ Meanwhile, from (3.17), $eq_0(i)$ holds for $i = 0, 1, \ldots$ Using Property (d), it follows that $eq_\tau(i)$ holds for $i = 0, 1, \ldots$

We prove by reduction to absurdity that the rank $r_{\tau+1}$ of $B_{0\tau}$ is m. Suppose to the contrary that $r_{\tau+1} < m$. Since $eq'_{\tau-1}(i) \xrightarrow{R_b[r_\tau]} eq_\tau(i)$ is linear over $GF(q)$, the first r_τ rows of $B_{0\tau}$ are linearly independent and $r_\tau \leqslant r_{\tau+1}$. Then a linear R_a transformation can be applied to $eq_\tau(i)$, say $eq_\tau(i) \xrightarrow{R_a[P_\tau]} eq'_\tau(i)$. Thus $eq'_\tau(i)$ holds for $i = 0, 1, \ldots$ It follows that $E''_\tau eq'_\tau(i)$ holds for $i = 0, 1, \ldots$, that is,

$$E''_\tau P_\tau B_{0\tau} x_i - E''_\tau P_\tau A_{-\tau} y_{i+\tau} + E''_\tau P_\tau g_\tau(x(i-1, r), y(i+\tau-1, \tau+t)) = 0,$$
$$i = 0, 1, \ldots, \tag{3.18}$$

where E''_τ is the submatrix of the last $m - r_{\tau+1}$ rows of the $m \times m$ identity matrix. Noticing that $E''_\tau P_\tau B_{0\tau} = 0$ and $y_{j+\tau} = y'_j$ for $j \geqslant 0$, from (3.18), we have

$$E''_\tau P_\tau A_{-\tau} y'_i - E''_\tau P_\tau g_\tau(x(i-1, r), y'(i-1, \tau+t)) = 0,$$
$$i = \tau + t, \tau + t + 1, \ldots \tag{3.19}$$

From Lemma 3.3.1, the rank of $[B_{0\tau}, A_{-\tau}]$ is m. Since $m > r_{\tau+1}$ and $E''_\tau P_\tau B_{0\tau} = 0$, rows of $E''_\tau P_\tau A_{-\tau}$ are linearly independent. Thus the formula (3.19) gives constraint equations, that is, when $i \geqslant \tau + t$, the input y'_i of M_1 depends on the past inputs $y'_{i-1}, \ldots, y'_{i-\tau-t}$ and the past outputs $x_{i-1}, \ldots, x_{i-r}$. This is a contradiction. Therefore, the hypothesis $r_{\tau+1} < m$ does not hold. We conclude that $r_{\tau+1} = m$. □

Notice that if $eq_0(i)$ is defined by (3.16), then a linear R_a R_b transformation sequence of length 2τ beginning at $eq_0(i)$ is existent.

Let $\bar{M}$ be a τ-order input-memory finite automaton defined by

$$y_i = \sum_{j=0}^{\tau} B_j x_{i-j}, \ i = 0, 1, \ldots, \tag{3.20}$$

where B_j is an $m \times l$ matrix over $GF(q)$, $j = 0, 1, \ldots, \tau$.

Corollary 3.3.1. *Let $eq_0(i)$ be the equation*

$$\sum_{j=0}^{\tau} B_j x_{i-j} - y_i = 0, \tag{3.21}$$

that is, the equation (3.13) *with $k = t = 0$ and $r = \tau$, where $B_{j0} = B_j, j = 0, 1, \ldots, \tau$, A_{-0} is the identity matrix, and $g_0 = 0$. Let*

$$eq_k(i) \overset{R_a[P_k]}{\longrightarrow} eq'_k(i),\ eq'_k(i) \overset{R_b[r_{k+1}]}{\longrightarrow} eq_{k+1}(i), \quad k = 0, 1, \ldots, \tau - 1$$

be linear over $GF(q)$. Then $\bar{M}$ is a weak inverse with delay τ if and only if the rank of $B_{0\tau}$ is m.

Corollary 3.3.2. *Let $eq_0(i)$ be* (3.21)*. Let*

$$eq_k(i) \overset{R_a[P_k]}{\longrightarrow} eq'_k(i),\ eq'_k(i) \overset{R_b[r_{k+1}]}{\longrightarrow} eq_{k+1}(i), \quad k = 0, 1, \ldots, \tau - 1$$

be linear over $GF(q)$. Then $\bar{M}$ is an inverse with delay τ if and only if the rank of $B_{0\tau}$ is m.

Corollary 3.3.3. *Assume that B_j in* (3.12) *is the same as B_j in* (3.20) *for $0 \leqslant j \leqslant \tau$. Then M is a weak inverse with delay τ if and only if $\bar{M}$ is a weak inverse with delay τ, if and only if $\bar{M}$ is an inverse with delay τ.*

Proof. Notice that in Theorem 3.3.1 $B_{0\tau}$ depends only on $B_0, \ldots, B_\tau$. Since linear R_a R_b transformation sequence is existent, from Theorem 3.3.1, Corollary 3.3.1 and Corollary 3.3.2, the corollary follows. □

Theorem 3.3.2. *Let $eq_0(i)$ be* (3.16)*. Let*

$$eq_k(i) \overset{R_a[P_k]}{\longrightarrow} eq'_k(i),\ eq'_k(i) \overset{R_b[r_{k+1}]}{\longrightarrow} eq_{k+1}(i), \quad k = 0, 1, \ldots, \tau - 1$$

be linear over $GF(q)$. Then M is weakly invertible with delay τ if and only if the rank of $B_{0\tau}$ is l.

Proof. if : Suppose that the rank of $B_{0\tau}$ is l. Then there is a left inverse matrix of $B_{0\tau}$, say $B_{0\tau}^{-1}$. Thus for any parameters x_{i-1}, ..., x_{i-r}, $y_{i+\tau}$, ..., y_{i-t}, $eq_\tau(i)$ has at most one solution x_i

$$x_i = B_{0\tau}^{-1} A_{-\tau} y_{i+\tau} - B_{0\tau}^{-1} g_\tau(x(i-1, r), y(i+\tau-1, \tau+t)).$$

From Theorem 3.1.3, M is weakly invertible with delay τ.

only if : Suppose that the rank of $B_{0\tau}$ is less than l. We prove that M is not weakly invertible with delay τ.

For any state $s_0 = \langle y(-1, t), x(-1, r)\rangle$ of M, and any input sequence $x_0 \ldots x_\tau$ of length $\tau + 1$ of M, let

$$y_0 \ldots y_\tau = \lambda(s_0, x_0 \ldots x_\tau). \tag{3.22}$$

Since $eq_0(i)$ is (3.16), from (3.12), (3.22) is equivalent to the system of equations

$$eq_0(i), \quad i = 0, 1, \ldots, \tau. \tag{3.23}$$

Using Property (g), the system (3.23) is equivalent to the system of equations

$$\begin{aligned} &eq_\tau(0),\\ &E'_0eq'_0(\tau), E'_1eq'_1(\tau-1),\ldots, E'_{\tau-1}eq'_{\tau-1}(1),\\ &E''_0eq'_0(0), E''_1eq'_1(0),\ldots, E''_{\tau-1}eq'_{\tau-1}(0), \end{aligned} \tag{3.24}$$

where E'_k and E''_k are submatrices of the first r_{k+1} rows and the last $m-r_{k+1}$ rows of the $m \times m$ identity matrix, respectively. Therefore, M is not weakly invertible with delay τ if and only if there are two solutions of (3.24) in which the corresponding values of $x_{-1},\ldots,x_{-r}$, $y_\tau,\ldots,y_{-t}$ are the same and the corresponding values of x_0 are different.

Given an arbitrary initial state $s'_0 = \langle y'(-1,t), x'(-1,r)\rangle$ of M, and any input sequence $x'_0\ldots x'_\tau$ of length $\tau+1$ of M, let

$$y'_0\ldots y'_\tau = \lambda(s'_0, x'_0\ldots x'_\tau).$$

Then $y_\tau = y'_\tau,\ldots,y_{-t} = y'_{-t}$, $x_\tau = x'_\tau,\ldots,x_{-r} = x'_{-r}$ is a solution of (3.24). Notice that the equation $eq_\tau(0)$ depends only on $x_0,\ldots,x_{-r}$ and $y_\tau,\ldots,y_{-t}$, and in the polynomial expression of $eq_\tau(0)$ the variable x_0 only occurs in linear term with coefficient $B_{0\tau}$. Since the rank of $B_{0\tau}$ is less than l, there is a column vector $\Delta \neq 0$ such that $B_{0\tau}\Delta = 0$. To find another solution of (3.24), take $y_\tau = y'_\tau,\ldots,y_{-t} = y'_{-t}$, $x_{-1} = x'_{-1},\ldots,x_{-r} = x'_{-r}$, and $x_0 = x'_0 + \Delta$. Clearly, such new values satisfy the equation $eq_\tau(0)$. Notice that the equation $E''_keq'_k(0)$ depends only on $x_{-1},\ldots,x_{-r}$ and $y_k,\ldots,y_{-t}$, $k = 0,1,\ldots,\tau-1$. Thus $y_\tau = y'_\tau,\ldots,y_{-t} = y'_{-t}$, $x_{-1} = x'_{-1},\ldots,x_{-r} = x'_{-r}$ satisfy the system of equations $E''_keq'_k(0)$, $k = 0,1,\ldots,\tau-1$. We seek new values of $x_1,\ldots,x_\tau$ step by step. Suppose that we have sought the new values of $x_0,\ldots,x_{\tau-k-1}$ such that the equations $eq_\tau(0)$, $E'_{\tau-1}eq'_{\tau-1}(1)$, $E'_{\tau-2}eq'_{\tau-2}(2),\ldots$, $E'_{k+1}eq'_{k+1}(\tau-k-1)$ hold, and $0 \leqslant k \leqslant \tau-1$. Since $E'_keq'_k(\tau-k)$ depends only on $x_{\tau-k},\ldots,x_{\tau-k-r}$, $y_\tau,\ldots,y_{\tau-k-t}$ and in the polynomial expression of $E'_keq'_k(\tau-k)$ the variable $x_{\tau-k}$ only occurs in linear term with a coefficient matrix $E'_kP_kB_{0k}$ of which rows are linearly independent, we can seek a value of $x_{\tau-k}$ from the new values of $x_{\tau-k-1},\ldots,x_{-r}$, $y_\tau,\ldots,y_{-t}$ by solving the equation $E'_keq'_k(\tau-k)$. (If solutions for $x_{\tau-k}$ are not unique, then take arbitrarily such a solution as the new value of $x_{\tau-k}$. If the number of rows of E'_k is 0, then the new value of $x_{\tau-k}$ can be arbitrarily taken.) Since equations $eq_\tau(0)$, $E'_{\tau-1}eq'_{\tau-1}(1)$, $E'_{\tau-2}eq'_{\tau-2}(2)$, …, $E'_{k+1}eq'_{k+1}(\tau-k-1)$ do not depend on $x_{\tau-k}$, from the hypothesis that new values $x_0,\ldots,x_{\tau-k-1}$ satisfy the system of equations $eq_\tau(0)$, $E'_{\tau-1}eq'_{\tau-1}(1)$, …, $E'_{k+1}eq'_{k+1}(\tau-k-1)$, the new values of $x_0,\ldots,x_{\tau-k-1},x_{\tau-k}$ satisfy the system of equations $eq_\tau(0)$, $E'_{\tau-1}eq'_{\tau-1}(1)$, $E'_{\tau-2}eq'_{\tau-2}(2)$, …, $E'_{k+1}eq'_{k+1}(\tau-k-1)$, $E'_keq'_k(\tau-k)$. Repeating the above

process for k from $\tau-1$ to 0, we can obtain new values of $x_0, \ldots, x_\tau$ such that the equations $eq_\tau(0)$, $E'_{\tau-1}eq'_{\tau-1}(1)$, $E'_{\tau-2}eq'_{\tau-2}(2)$, …, $E'_0eq'_0(\tau)$ hold. Thus we obtain a new solution of the system (3.24) in which $y_\tau = y'_\tau, \ldots, y_{-t} = y'_{-t}$, $x_{-1} = x'_{-1}, \ldots, x_{-r} = x'_{-r}$, and $x_0 = x'_0 + \Delta$. Therefore, there are two solutions of (3.24) in which the corresponding values of $x_{-1}, \ldots, x_{-r}$, $y_\tau, \ldots, y_{-t}$ are the same and the corresponding values of x_0 are different. We conclude that M is not weakly invertible with delay τ. □

Corollary 3.3.4. *Let $eq_0(i)$ be (3.21). Let*

$$eq_k(i) \overset{R_a[P_k]}{\longrightarrow} eq'_k(i),\ eq'_k(i) \overset{R_b[r_{k+1}]}{\longrightarrow} eq_{k+1}(i), \quad k = 0, 1, \ldots, \tau - 1$$

be linear over $GF(q)$. Then $\bar{M}$ is weakly invertible with delay τ if and only if the rank of $B_{0\tau}$ is l.

Corollary 3.3.5. *Assume that B_j in (3.12) is the same as B_j in (3.20) for any j, $0 \leqslant j \leqslant \tau$. Then M is weakly invertible with delay τ if and only if $\bar{M}$ is weakly invertible with delay τ.*

Proof. Notice that in Theorem 3.3.2 $B_{0\tau}$ depends only on $B_0, \ldots, B_\tau$. Since linear R_a R_b transformation sequence is existent, from Theorem 3.3.2 and Corollary 3.3.4, the corollary follows. □

3.3.2 Structure Problem

In this subsection, unless otherwise stated, $G_k(i)$ and $G'_k(i)$ in modified Rules R_a, R_b and Rules R_a^{-1}, R_b^{-1} defined in Sect. 3.2 do not depend on $y_{i+k}, \ldots, y_{i-t}$ and are abbreviated to G_k and G'_k, respectively. Let R be a finite field $GF(q)$.

$G_k \overset{R_a[P_k]}{\longrightarrow} G'_k$ is said to be *linear* over $GF(q)$, if P_k is a matrix over $GF(q)$ and for some $r_{k+1} \geqslant$ the k-height of G_k, the first r_{k+1} rows of B'_{0k} is linearly independent over $GF(q)$ and B'_{0k} has zeros in the last $m - r_{k+1}$ rows whenever $r_{k+1} < m$, where B'_{0k} is the submatrix of the first l columns of G'_k. $G'_k \overset{R_b[r_{k+1}]}{\longrightarrow} G_{k+1}$ is said to be *linear* over $GF(q)$, if the first r_{k+1} rows of B'_{0k} is linearly independent over $GF(q)$. An R_a R_b transformation sequence

$$G_k \overset{R_a[P_k]}{\longrightarrow} G'_k,\ G'_k \overset{R_b[r_{k+1}]}{\longrightarrow} G_{k+1}, \quad k = 0, 1, \ldots, h \tag{3.25}$$

is said to be *linear* over $GF(q)$, if $G_k \overset{R_a[P_k]}{\longrightarrow} G'_k$ and $G'_k \overset{R_b[r_{k+1}]}{\longrightarrow} G_{k+1}$ are linear over $GF(q)$ for $k = 0, 1, \ldots, h$.

Let

$$G_0 = [B_0, \ldots, B_\tau]$$

be an $m \times l(\tau+1)$ matrix over $GF(q)$ determined by M which is defined by (3.12). Clearly, G_0 is an $(l,0)$-echelon matrix and there exists a linear modified R_a R_b transformation sequence

$$G_k \overset{R_a[P_k]}{\longrightarrow} G'_k,\ G'_k \overset{R_b[r_{k+1}]}{\longrightarrow} G_{k+1},\quad k=0,1,\ldots,\tau-1.$$

Lemma 3.3.2. *Let* $G_0=[B_0,\ldots,B_\tau]$. *If*

$$G_k \overset{R_a[P_k]}{\longrightarrow} G'_k,\ G'_k \overset{R_b[r_{k+1}]}{\longrightarrow} G_{k+1},\quad k=0,1,\ldots,\tau-1$$

is a linear modified R_a R_b *transformation sequence, then*

$$eq_k(i) \overset{R_a[P_k]}{\longrightarrow} eq'_k(i),\ eq'_k(i) \overset{R_b[r_{k+1}]}{\longrightarrow} eq_{k+1}(i),\quad k=0,1,\ldots,\tau-1$$

is a linear R_a R_b *transformation sequence and the submatrix of the first* l *columns of* G_τ *is the same as the coefficient matrix* $B_{0\tau}$ *of* x_i *in* $eq_\tau(i)$, *where* $eq_0(i)$ *is* (3.16).

Proof. Suppose that $G_k \overset{R_a[P_k]}{\longrightarrow} G'_k$, $G'_k \overset{R_b[r_{k+1}]}{\longrightarrow} G_{k+1}$ are linear over $GF(q)$, $0 \leqslant k < \tau$ and that $[B_{0k},\ldots,B_{\tau-k,k}]$, determined by the coefficients of $eq_k(i)$, is the leftmost $l(\tau-k+1)$ columns of G_k, where $eq_k(i)$ is expressed as (3.13). Then the first r_{k+1} rows of B'_{0k} are linearly independent over $GF(q)$, and B'_{0k} has zeros in the last $m-r_{k+1}$ rows whenever $r_{k+1}<m$, where $B'_{0k}=P_kB_{0k}$ is the leftmost l columns of G'_k. Thus $eq_k(i) \overset{R_a[P_k]}{\longrightarrow} eq'_k(i)$ is linear over $GF(q)$, where $eq'_k(i)$ is $P_keq_k(i)$ and is expressed as (3.14). Clearly, $[B'_{0k},\ldots,B'_{\tau-k,k}]$, determined by the coefficients of $eq'_k(i)$, is the leftmost $l(\tau-k+1)$ columns of G'_k. It is easy to see that $eq'_k(i) \overset{R_b[r_{k+1}]}{\longrightarrow} eq_{k+1}(i)$ is linear over $GF(q)$ and $[B_{0,k+1},\ldots,B_{\tau-k-1,k+1}]$, determined by the coefficients of $eq_{k+1}(i)$, is the leftmost $l(\tau-(k+1)+1)$ columns of G_{k+1}. By simple induction, the lemma follows. □

Theorem 3.3.3. *Let* $G_0=[B_0,\ldots,B_\tau]$, *and*

$$G_k \overset{R_a[P_k]}{\longrightarrow} G'_k,\ G'_k \overset{R_b[r_{k+1}]}{\longrightarrow} G_{k+1},\quad k=0,1,\ldots,\tau-1$$

be a linear modified R_a R_b *transformation sequence. Then* M *is a weak inverse with delay* τ *if and only if the rank of* $B_{0\tau}$ *is* m, *and* M *is weakly invertible with delay* τ *if and only if the rank of* $B_{0\tau}$ *is* l, *where* $B_{0\tau}$ *is the submatrix of the first* l *columns of* G_τ.

Proof. This theorem immediately follows from Theorem 3.3.1, Theorem 3.3.2 and Lemma 3.3.2. □

$G'_k \overset{R_a^{-1}[P'_k]}{\longrightarrow} G_k$ is said to be *linear* over $GF(q)$, if P'_k is a matrix over $GF(q)$ and for some $r_{k+1} \geqslant$ the k-height of G'_k, the first r_{k+1} rows of B'_{0k} is linearly independent over $GF(q)$ and B'_{0k} has zeros in the last $m - r_{k+1}$ rows whenever $r_{k+1} < m$, where B'_{0k} is the submatrix of the first l columns of G'_k. $G_{k+1} \overset{R_b^{-1}[r_{k+1}]}{\longrightarrow} G'_k$ is said to be *linear* over $GF(q)$, if the first r_{k+1} rows of B'_{0k} is linearly independent over $GF(q)$. An R_a^{-1} R_b^{-1} transformation sequence

$$G_{k+1} \overset{R_b^{-1}[r_{k+1}]}{\longrightarrow} G'_k,\ G'_k \overset{R_a^{-1}[P'_k]}{\longrightarrow} G_k,\ k = h, \ldots, 1, 0 \tag{3.26}$$

is said to be *linear* over $GF(q)$, if $G_{k+1} \overset{R_b^{-1}[r_{k+1}]}{\longrightarrow} G'_k$ and $G'_k \overset{R_a^{-1}[P'_k]}{\longrightarrow} G_k$ are linear over $GF(q)$ for $k = h, \ldots, 1, 0$.

Lemma 3.3.3.

$$G_k \overset{R_a[P_k]}{\longrightarrow} G'_k,\ G'_k \overset{R_b[r_{k+1}]}{\longrightarrow} G_{k+1},\quad k = 0, 1, \ldots, \tau - 1$$

is a linear modified R_a R_b transformation sequence if and only if

$$G_{k+1} \overset{R_b^{-1}[r_{k+1}]}{\longrightarrow} G'_k,\ G'_k \overset{R_a^{-1}[P_k^{-1}]}{\longrightarrow} G_k,\ k = \tau - 1, \ldots, 1, 0$$

is a linear R_a^{-1} R_b^{-1} transformation sequence.

Proof. From Lemma 3.2.2,

$$G_k \overset{R_a[P_k]}{\longrightarrow} G'_k,\ G'_k \overset{R_b[r_{k+1}]}{\longrightarrow} G_{k+1},\quad k = 0, 1, \ldots, \tau - 1$$

is a modified R_a R_b transformation sequence if and only if

$$G_{k+1} \overset{R_b^{-1}[r_{k+1}]}{\longrightarrow} G'_k,\ G'_k \overset{R_a^{-1}[P_k^{-1}]}{\longrightarrow} G_k,\ k = \tau - 1, \ldots, 1, 0$$

is an R_a^{-1} R_b^{-1} transformation sequence. It is easy to see that $G_k \overset{R_a[P_k]}{\longrightarrow} G'_k$ is linear if and only if $G'_k \overset{R_a[P_k^{-1}]}{\longrightarrow} G_k$ is linear and that $G'_k \overset{R_b[r_{k+1}]}{\longrightarrow} G_{k+1}$ is linear if and only if $G_{k+1} \overset{R_b^{-1}[r_{k+1}]}{\longrightarrow} G'_k$ is linear. From the definition of linear transformation sequence, the lemma follows. □

Theorem 3.3.4. *Let M be a finite automaton defined by* (3.12) *and* $G_0 = [B_0, \ldots, B_\tau]$, $\tau \leqslant r$.

(a) *M is a weak inverse finite automaton with delay τ if and only if there exist an $m \times l(\tau + 1)$ (l, τ)-echelon matrix G_τ over $GF(q)$ and a linear R_a^{-1} R_b^{-1} transformation sequence*

$$G_{k+1} \overset{R_b^{-1}[r_{k+1}]}{\longrightarrow} G'_k,\ G'_k \overset{R_a^{-1}[P'_k]}{\longrightarrow} G_k,\ k = \tau - 1, \ldots, 1, 0 \tag{3.27}$$

such that the rank of the submatrix $B_{0\tau}$ of the first l columns of G_τ is m.

(b) *M is weakly invertible finite automaton with delay τ if and only if there exist an $m \times l(\tau+1)$ (l,τ)-echelon matrix G_τ over $GF(q)$ and a linear R_a^{-1} R_b^{-1} transformation sequence* (3.27) *such that the rank of the submatrix $B_{0\tau}$ of the first l columns of G_τ is l.*

Proof. (a) Suppose that M is a weak inverse with delay τ. Clearly, there exists a linear modified R_a R_b transformation sequence

$$G_k \overset{R_a[P_k]}{\longrightarrow} G'_k,\ G'_k \overset{R_b[r_{k+1}]}{\longrightarrow} G_{k+1},\quad k = 0, 1, \ldots, \tau - 1.$$

From Theorem 3.3.3, the rank of $B_{0\tau}$ is m, where $B_{0\tau}$ is the submatrix of the first l columns of G_τ. From Lemma 3.3.3, taking $P'_k = P_k^{-1}$ for $k = 0, 1, \ldots, \tau - 1$, (3.27) is a linear R_a^{-1} R_b^{-1} transformation sequence.

Conversely, suppose that (3.27) is a linear R_a^{-1} R_b^{-1} transformation sequence and the rank of $B_{0\tau}$ is m. Let $P_k = (P'_k)^{-1}$, $k = 0, 1, \ldots, \tau - 1$. From Lemma 3.3.3,

$$G_k \overset{R_a[P_k]}{\longrightarrow} G'_k,\ G'_k \overset{R_b[r_{k+1}]}{\longrightarrow} G_{k+1},\quad k = 0, 1, \ldots, \tau - 1$$

is a linear modified R_a R_b transformation sequence. Using Theorem 3.3.3, M is a weak inverse with delay τ.

(b) This part is similar to part (a). □

We use R_c to denote an operator: shift the last c rows l columns to the right entering zeros to the left.

Clearly, (3.27) is a linear R_a^{-1} R_b^{-1} transformation sequence if and only if the following conditions hold:

$$G'_k = R_{m-r_{k+1}} G_{k+1},\quad G_k = P'_k G'_k,$$
$$k = \tau - 1, \ldots, 1, 0,$$

P'_k is an $m \times m$ invertible matrix over $GF(q)$ in the form

$$P'_k = \begin{bmatrix} E_{r_k} & 0 \\ P'_{k1} & P'_{k2} \end{bmatrix},$$

r_k is the k-height of G'_k, G_{k+1} is an $(l, k+1)$-echelon matrix over $GF(q)$ with $(k+1)$-height $r_{k+1} \geqslant r_k$ and the first r_{k+1} rows of the submatrix of the first l columns of G_{k+1} are linear independent over $GF(q)$, $k = \tau - 1, \ldots, 1, 0$, where E_{r_k} is the $r_k \times r_k$ identity matrix. In computing elements of $P'_k G'_k$, we define $u \cdot u = u + u = u$, $v \cdot u = u \cdot v = u$, $0 \cdot u = u \cdot 0 = 0$ and $w + u = u + w = u$, where u stands for undefined symbol, $v (\neq 0)$ and w are any elements in $GF(q)$. Notice that the k-height of G_k is the same as the k-height of G_τ and the submatrices consisting of the first l columns and the first r_k rows of G_k and G_τ are the same, for $k < \tau$. From Theorem 3.3.4, we obtain the following.

Theorem 3.3.5. *Let M be a finite automaton defined by* (3.12).

(a) *M is a weak inverse finite automaton with delay τ if and only if there exist an $m\times l(\tau+1)$ (l,τ)-echelon matrix G_τ and $m\times m$ nonsingular matrices*

$$P'_k = \begin{bmatrix} E_{r_k} & 0 \\ P'_{k1} & P'_{k2} \end{bmatrix}, \quad k = 0, 1, \ldots, \tau - 1$$

such that the rank of the submatrix $B_{0\tau}$ of the first l columns of G_τ is m, and

$$[B_0, \ldots, B_\tau] = P'_0 R_{m-r_1}(P'_1 R_{m-r_2}(\ldots(P'_{\tau-1} R_{m-r_\tau}(G_\tau))\ldots))$$

if $\tau > 0$ and $[B_0, \ldots, B_\tau] = G_\tau$ otherwise, where E_{r_k} is the $r_k \times r_k$ identity matrix, and r_k is the k-height of G_τ.

(b) *M is a weakly invertible finite automaton with delay τ if and only if there exist an $m \times l(\tau + 1)$ (l,τ)-echelon matrix G_τ and $m \times m$ nonsingular matrices*

$$P'_k = \begin{bmatrix} E_{r_k} & 0 \\ P'_{k1} & P'_{k2} \end{bmatrix}, \quad k = 0, 1, \ldots, \tau - 1$$

such that the rank of the submatrix $B_{0\tau}$ of the first l columns of G_τ is l, the first r_τ rows of $B_{0\tau}$ are linearly independent over $GF(q)$, and

$$[B_0, \ldots, B_\tau] = P'_0 R_{m-r_1}(P'_1 R_{m-r_2}(\ldots(P'_{\tau-1} R_{m-r_\tau}(G_\tau))\ldots))$$

if $\tau > 0$ and $[B_0, \ldots, B_\tau] = G_\tau$ otherwise, where E_{r_k} is the $r_k \times r_k$ identity matrix, and r_k is the k-height of G_τ.

Denote $r_{\tau+1} = m$. Let $h = \min\ k\{\, r_k = m, 1 \leqslant k \leqslant \tau + 1 \,\}$. Notice that in the case of $r_k = m$, R_{m-r_k} is the identity operator and P'_k is the identity matrix. In the case of $h \leqslant \tau$, we have

$$\begin{aligned} &P'_0 R_{m-r_1}(P'_1 R_{m-r_2}(\ldots(P'_{\tau-1} R_{m-r_\tau}(G_\tau))\ldots)) \\ &\quad = P'_0 R_{m-r_1}(P'_1 R_{m-r_2}(\ldots(P'_{h-2} R_{m-r_{h-1}}(P'_{h-1} G_\tau))\ldots)). \end{aligned}$$

Since P'_{h-1} is nonsingular, the rank of the submatrix of the first l columns of $P'_{h-1}G_\tau$ is the same as the rank of the submatrix of the first l columns of G_τ. Since $h \leqslant \tau$, we have $r_\tau = m$. Therefore, the first r_τ rows of the submatrix of the first l columns of $P'_{h-1}G_\tau$ are linearly independent over $GF(q)$ if and only if the first r_τ rows of the submatrix of the first l columns of G_τ are linearly independent over $GF(q)$. Clearly, $P'_{h-1}G_\tau$ is also an $m\times l(\tau+1)$ (l,τ)-echelon matrix over $GF(q)$ of which the i-height is the same as the i-height of G_τ for any i, $0 \leqslant i \leqslant \tau$. From Theorem 3.3.5, taking $G = P'_{h-1}G_\tau$, we then have the following theorem.

Theorem 3.3.6. *Let M be a finite automaton defined by* (3.12).

(a) *M is a weak inverse finite automaton with delay τ if and only if there exist an $m \times l(\tau+1)$ (l,τ)-echelon matrix G and $m \times m$ nonsingular matrices*

$$P'_k = \begin{bmatrix} E_{r_k} & 0 \\ P'_{k1} & P'_{k2} \end{bmatrix}, \quad k = 0, 1, \ldots, h-2$$

such that the rank of the submatrix of the first l columns of G is m, and

$$[B_0, \ldots, B_\tau] = P'_0 R_{m-r_1}(P'_1 R_{m-r_2}(\ldots(P'_{h-2} R_{m-r_{h-1}}(G))\ldots))$$

if $h > 1$ and $[B_0, \ldots, B_\tau] = G$ otherwise, where E_{r_k} is the $r_k \times r_k$ identity matrix, r_k is the k-height of G, and $h = \min\ k\{\, r_k = m, 1 \leqslant k \leqslant \tau+1 \,\}$ $(r_{\tau+1} = m$ by convention$)$.

(b) *M is a weakly invertible finite automaton with delay τ if and only if there exist an $m \times l(\tau+1)$ (l,τ)-echelon matrix G and $m \times m$ nonsingular matrices*

$$P'_k = \begin{bmatrix} E_{r_k} & 0 \\ P'_{k1} & P'_{k2} \end{bmatrix}, \quad k = 0, 1, \ldots, h-2$$

such that the rank of the submatrix G_0^l of the first l columns of G is l, the first r_τ rows of G_0^l are linearly independent over $GF(q)$, and

$$[B_0, \ldots, B_\tau] = P'_0 R_{m-r_1}(P'_1 R_{m-r_2}(\ldots(P'_{h-2} R_{m-r_{h-1}}(G))\ldots))$$

if $h > 1$ and $[B_0, \ldots, B_\tau] = G$ otherwise, where E_{r_k} is the $r_k \times r_k$ identity matrix, r_k is the k-height of G, and $h = \min\ k\{\, r_k = m, 1 \leqslant k \leqslant \tau+1 \,\}$ $(r_{\tau+1} = m$ by convention$)$.

Let G be an $m \times l(\tau+1)$ incompletely specified matrix over $GF(q)$. If there are $k_m, \ldots, k_1$ such that $0 \leqslant k_m \leqslant \cdots \leqslant k_1 \leqslant \tau$, and for each j, $1 \leqslant j \leqslant m$, in row j of G elements in the first $l(1+k_j)$ columns are defined and in the last $l(\tau - k_j)$ columns are undefined, that is, G is in the form

$$\begin{bmatrix} G_{10} & \ldots & G_{1,k_m} & G_{1,k_m+1} & \ldots & G_{1,k_j} & G_{1,k_j+1} & \ldots & G_{1,k_1} & * & \ldots & * \\ \ldots & \ldots & \ldots & \ldots & \ldots & \ldots & \ldots & \ldots & \ldots & \ldots & & \\ G_{j0} & \ldots & G_{j,k_m} & G_{j,k_m+1} & \ldots & G_{j,k_j} & * & \ldots & * & * & \ldots & * \\ \ldots & \ldots & \ldots & \ldots & \ldots & \ldots & \ldots & \ldots & \ldots & \ldots & & \\ G_{m0} & \ldots & G_{m,k_m} & * & \ldots & * & * & \ldots & * & * & \ldots & * \end{bmatrix},$$

where $*$ stands for the $1 \times l$ "undefined matrix", G_{ij} is a $1 \times l$ completely specified matrix, G is called an *echelon matrix* and k_j is called the *j-length* of G, for $j = 1, \ldots, m$.

Let G be an $m \times l(\tau+1)$ (l,τ)-echelon matrix over $GF(q)$ with i-height r_i, $i = 0, 1, \ldots, \tau$. Let

$$k_j = \tau - \min\ i\{\, 0 \leqslant i \leqslant \tau,\ r_i < j \leqslant r_{i+1}\,\},\quad j = 1, \ldots, m, \tag{3.28}$$

where $r_{\tau+1} = m$. Then G is an echelon matrix with j-length k_j, $j = 1, \ldots, m$. In fact, for any j, $1 \leqslant j \leqslant m$, when $r_i < j \leqslant r_{i+1}$ for some i, $0 \leqslant i \leqslant \tau$, from the definition of i-height, we have that in row j of G elements in the first $l(\tau + 1 - i)$ columns are defined and elements in the last li columns are undefined. From (3.28), $k_j = \tau - i$. Then in row j of G elements in the first $l(1 + k_j)$ columns are defined and elements in the last $l(\tau - k_j)$ columns are undefined. Since $r_0 \leqslant r_1 \leqslant \cdots \leqslant r_\tau$, it is easy to verify that $0 \leqslant k_m \leqslant \cdots \leqslant k_1 \leqslant \tau$.

Conversely, let G be an $m \times l(\tau + 1)$ echelon matrix over $GF(q)$ with j-length k_j, $j = 1, \ldots, m$. Let

$$r_i = \min\ c\{\, 0 \leqslant c \leqslant m,\ k_j < \tau + 1 - i \quad \text{if } c < j \leqslant m\,\},\ i = 1, \ldots, \tau.$$

Then G is an (l, τ)-echelon matrix with i-height r_i, $i = 1, \ldots, \tau$.

Since $0 = r_0 \leqslant r_1 \leqslant \cdots \leqslant r_\tau \leqslant r_{\tau+1} = m$, using $k_m = \tau - \min\ i\{\, 0 \leqslant i \leqslant \tau,\ r_i < m \leqslant r_{i+1}\,\}$, we have $k_m = \tau - \min\ i\{\, 0 \leqslant i \leqslant \tau, m = r_{i+1}\,\}$. It follows immediately that

$$\begin{aligned}\min\ k\{\, 1 \leqslant k \leqslant \tau + 1, m = r_k\,\} - 1 &= \min\ i\{\, 0 \leqslant i \leqslant \tau, m = r_{i+1}\,\} \\ &= \tau - k_m.\end{aligned}$$

From the above discussion, Theorem 3.3.6 can be restated as follows.

Theorem 3.3.7. *Let M be a finite automaton defined by* (3.12).

(a) *M is a weak inverse finite automaton with delay τ if and only if there exist an $m \times l(\tau + 1)$ echelon matrix G with j-length k_j, $j = 1, \ldots, m$, and $m \times m$ nonsingular matrices*

$$P'_k = \begin{bmatrix} E_{r_{k-1}} & 0 \\ P'_{k1} & P'_{k2} \end{bmatrix},\quad k = 1, 2, \ldots, \tau - k_m$$

such that the rank of the submatrix of the first l columns of G is m, and

$$[B_0, \ldots, B_\tau] = P'_1 R_{m-r_1}(P'_2 R_{m-r_2}(\ldots(P'_{\tau-k_m} R_{m-r_{\tau-k_m}}(G))\ldots))$$

if $k_m < \tau$ and $[B_0, \ldots, B_\tau] = G$ otherwise, where $E_{r_{k-1}}$ is the $r_{k-1} \times r_{k-1}$ identity matrix, $r_0 = 0$,

$$r_i = \min\ c\{\, 0 \leqslant c \leqslant m,\ k_j < \tau + 1 - i \text{ if } c < j \leqslant m\,\},\ i = 1, \ldots, \tau.$$

(b) *M is a weakly invertible finite automaton with delay τ if and only if there exist an $m \times l(\tau + 1)$ echelon matrix G with j-length $k_j, j = 1, \ldots, m$, and $m \times m$ nonsingular matrices*

$$P'_k = \begin{bmatrix} E_{r_{k-1}} & 0 \\ P'_{k1} & P'_{k2} \end{bmatrix}, \quad k = 1, 2, \ldots, \tau - k_m$$

such that the rank of the submatrix G_0^l *of the first* l *columns of* G *is* l*, the first* r_τ *rows of* G_0^l *are linearly independent over* $GF(q)$*, and*

$$[B_0, \ldots, B_\tau] = P'_1 R_{m-r_1}(P'_2 R_{m-r_2}(\ldots(P'_{\tau-k_m} R_{m-r_{\tau-k_m}}(G))\ldots))$$

if $k_m < \tau$ *and* $[B_0, \ldots, B_\tau] = G$ *otherwise, where* $E_{r_{k-1}}$ *is the* $r_{k-1} \times r_{k-1}$ *identity matrix,* $r_0 = 0$*,*

$$r_i = \min\ c\{\, 0 \leqslant c \leqslant m,\ k_j < \tau + 1 - i \text{ if } c < j \leqslant m \,\},\ i = 1, \ldots, \tau.$$

Historical Notes

The R_a R_b transformation method is first proposed in [96, 98] to deal with the invertibility problem of linear finite automata over finite fields. The method is then used to quasi-linear finite automata over finite fields in [21]. References [135, 136] use the R_a R_b transformation method to finite automata over rings, and [22] to quadratic finite automata. The R_a R_b transformation method is also generalized to generate a kind of weakly invertible finite automata and their weak inverses in [118, 127]. Sections 3.1 and 3.2 are based on [127] and Sect. 3.3 is based on [21].

4. Relations Between Transformations

Renji Tao

Institute of Software, Chinese Academy of Sciences
Beijing 100080, China trj@ios.ac.cn

Summary.

In application to public key cryptosystems, for finite automata in public keys no feasible inversion algorithm had been found. In Sect. 3.1 of the preceding chapter, an inversion method by R_a R_b transformation was given implicitly. In this chapter, a relation between two R_a R_b transformation sequences beginning at the same equation is derived. It means that in the inversion process it is enough to choose any one of the linear R_a R_b transformation sequences. Then, by exploring properties of "composition" of two R_a R_b transformation sequences, it is shown that the inversion method by R_a R_b transformation works for some special compound finite automata.

Other two inversion methods are by reduced echelon matrices, and by canonical diagonal matrix polynomials. Results in the last two sections show that the two inversion methods are "equivalent" to the inversion method by R_a R_b transformation.

This chapter provides a foundation for assertions on the weak key of the public key cryptosystem based on finite automata in Sect. 9.4.

Key words: *R_a R_b transformation, reduced echelon matrix, canonical diagonal matrix polynomial*

In application to public key cryptosystems, for finite automata in public keys no feasible inversion algorithm had been found. In Sect. 3.1 of the preceding chapter, an inversion method by R_a R_b transformation was given implicitly. That is, from a given finite automaton M make an equation $eq_0(i)$, choose an R_a R_b transformation sequence of length 2τ beginning at $eq_0(i)$, check whether the variable x_i in the equation $eq_\tau(i)$ has at most one solution (respectively a solution), if so, then an weak inverse (respectively original weak inverse) finite automaton with delay τ of M can be feasibly constructed from

$eq_\tau(i)$. In the first section of this chapter, a relation between two R_a R_b transformation sequences beginning at the same equation is derived. It means that in the inversion process it is enough to choose any one of the linear R_a R_b transformation sequences.

By exploring properties of "composition" of two R_a R_b transformation sequences, it is shown that the inversion method by R_a R_b transformation works for some special compound finite automata, for example, one component belongs to quasi-linear finite automata, and another component is a weakly invertible or weak inverse finite automaton generated by linear R_a R_b transformation or with delay 0.

Another inversion method is to solve equations corresponding to an output sequence of length $\tau + 1$ by means of finding the reduced echelon matrix of the coefficient matrix of the equations. Section 4.3 proves that for a weakly invertible finite automaton with delay τ, its weak inverse can be found by the reduced echelon matrix method if and only if it can be found by the R_a R_b transformation method.

Using a "time shift" operator z, coefficient matrices of pseudo-memory finite automata may be expressed by matrix polynomials of z. By means of reducing to canonical diagonal matrix polynomials, an inversion method was derived. In Sect. 4.4, relations between terminating and elementary R_a R_b transformation sequences and canonical diagonal matrix polynomials and the existence of such R_a R_b transformation sequence are investigated. From presented results, it is easy to see that the inversion method by canonical diagonal matrix polynomial works if and only if the inversion method by R_a R_b transformation works.

This chapter provides a foundation for assertions on the weak key of the public key cryptosystem based on finite automata in Sect. 9.4.

4.1 Relations Between R_a R_b Transformations

Throughout this chapter, for any integer i, any nonnegative integer k and any symbol string z, we use $z(i,k)$ to denote the symbol string $z_i, z_{i-1}, \ldots, z_{i-k+1}$. Let X and U be two finite nonempty sets. Let Y be a column vector space of dimension m over a finite commutative ring R with identity, where m is a positive integer. In this section, for any integer i, we use x_i, u_i and y_i to denote elements in X, U and Y, respectively.

Let $\psi^l_{\mu\nu}$ be a column vector of dimension l of which each component is a single-valued mapping from $U^{\mu+1} \times X^{\nu+1}$ to Y for some integers $\mu \geqslant -1$, $\nu \geqslant 0$ and $l \geqslant 1$. For any integers $h \geqslant 0$ and i, let

$$\psi_{\mu\nu}^{lh}(u,x,i)=\begin{bmatrix}\psi_{\mu\nu}^{l}(u(i,\mu+1),x(i,\nu+1))\\ \vdots\\ \psi_{\mu\nu}^{l}(u(i-h,\mu+1),x(i-h,\nu+1))\end{bmatrix}.$$

In the case of $\mu=-1$, $\psi_{\mu\nu}^{l}(u,x)$ and $\psi_{\mu\nu}^{lh}(u,x,i)$ are abbreviated to $\psi_{\nu}^{l}(x)$ and $\psi_{\nu}^{lh}(x,i)$, respectively.

In this section, for any nonnegative integer c, let $eq_c(i)$ be an equation

$$\varphi_c(y(i+c,c+t+1))+[B_{0c},\ldots,B_{hc}]\psi_{\mu\nu}^{lh}(u,x,i)=0,$$

and let $eq'_c(i)$ be an equation

$$\varphi'_c(y(i+c,c+t+1))+[B'_{0c},\ldots,B'_{hc}]\psi_{\mu\nu}^{lh}(u,x,i)=0,$$

where φ_c and φ'_c are two single-valued mappings from Y^{c+t+1} to Y, B_{jc} and B'_{jc} are $m\times l$ matrices over R, $j=0,1,\ldots,h$. Similarly, for any nonnegative integer c, let $\bar{eq}_c(i)$ be an equation

$$\bar{\varphi}_c(y(i+c,c+t+1))+[\bar{B}_{0c},\ldots,\bar{B}_{hc}]\psi_{\mu\nu}^{lh}(u,x,i)=0,$$

and let $\bar{eq}'_c(i)$ be an equation

$$\bar{\varphi}'_c(y(i+c,c+t+1))+[\bar{B}'_{0c},\ldots,\bar{B}'_{hc}]\psi_{\mu\nu}^{lh}(u,x,i)=0,$$

where $\bar{\varphi}_c$ and $\bar{\varphi}'_c$ are two single-valued mappings from $Y^{c+t+1}$to Y, $\bar{B}_{jc}$ and $\bar{B}'_{jc}$ are $m\times l$ matrices over R, $j=0,1,\ldots,h$.

For such expressions, $eq_c(i)\xrightarrow{R_a[P_c]}eq'_c(i)$ is said to be *linear* over R, if P_c is a matrix over R, and for some $r_{c+1}\geqslant 0$ the first r_{c+1} rows of B'_{0c} is linearly independent over R and B'_{0c} has zeros in the last $m-r_{c+1}$ rows whenever $r_{c+1}<m$. $eq'_c(i)\xrightarrow{R_b[r_{c+1}]}eq_{c+1}(i)$ is said to be *linear* over R, if the first r_{c+1} rows of B'_{0c} is linearly independent over R. An R_a R_b transformation sequence

$$eq_c(i)\xrightarrow{R_a[P_c]}eq'_c(i),\ eq'_c(i)\xrightarrow{R_b[r_{c+1}]}eq_{c+1}(i),\quad c=0,1,\ldots,n \tag{4.1}$$

is said to be *linear* over R, if $eq_c(i)\xrightarrow{R_a[P_c]}eq'_c(i)$ and $eq'_c(i)\xrightarrow{R_b[r_{c+1}]}eq_{c+1}(i)$ are linear over R for $c=0,1,\ldots,n$. The R_a R_b transformation sequence (4.1) is said to be *elementary over* R, if (4.1) is linear over R and P_c is in the form

$$P_c=\begin{bmatrix}E_{r_c} & 0\\ P_{c1} & P_{c2}\end{bmatrix},$$

$c=0,1,\ldots,n$, where E_r stands for the $r\times r$ identity matrix for any r, and $r_0=0$.

For any integers n and r with $0 \leqslant r \leqslant n$, we use $E_{n,r}$ to denote the $n \times n$ matrix $\begin{bmatrix} 0 & E_r \\ E_{n-r} & 0 \end{bmatrix}$, and $E_{n,-r}$ to denote $E_{n,n-r}$. Denote the $n \times r$ zero matrix by $0_{n,r}$. Also we use $DIA_{P,n}$ to denote the quasi-diagonal matrix $\begin{bmatrix} P & & \\ & \ddots & \\ & & P \end{bmatrix}$ with n occurrences of P.

Below the ring R is restricted to a finite field $GF(q)$, and the R_a transformation means multiplying two sides of an equation on the left by a matrix over $GF(q)$.

Lemma 4.1.1. *Let $c > 0$, and let*

$$eq_{c-1}(i) \overset{R_a[P_{c-1}]}{\longrightarrow} eq'_{c-1}(i), eq'_{c-1}(i) \overset{R_b[r_c]}{\longrightarrow} eq_c(i) \tag{4.2}$$

be linear over $GF(q)$ and

$$\bar{eq}_{c-1}(i) \overset{R_a[\bar{P}_{c-1}]}{\longrightarrow} \bar{eq}'_{c-1}(i), \bar{eq}'_{c-1}(i) \overset{R_b[\bar{r}_c]}{\longrightarrow} \bar{eq}_c(i). \tag{4.3}$$

If

$$\sum_{j=0}^{h} \bar{B}_{j,c-1} z^j = \big(\sum_{j=0}^{c-1} Q_{j,c-1} z^j\big)\big(\sum_{j=0}^{h} B_{j,c-1} z^j\big) \tag{4.4}$$

for some $m \times m$ matrices $Q_{j,c-1}$ over $GF(q)$, $j = 0, 1, \ldots, c-1$, then there exist $m \times m$ matrices Q_{jc} over $GF(q)$, $j = 0, 1, \ldots, c$ such that

$$\sum_{j=0}^{h} \bar{B}_{jc} z^j = \big(\sum_{j=0}^{c} Q_{jc} z^j\big)\big(\sum_{j=0}^{h} B_{jc} z^j\big). \tag{4.5}$$

Moreover, if $Q_{0,c-1}$ is nonsingular and (4.3) is linear over $GF(q)$, then Q_{0c} is nonsingular and $\bar{r}_c = r_c$.

Proof. Let $R_1 = E_{m,\bar{r}_c}$, $R_2 = [0_{m,\bar{r}_c} E_m 0_{m,m-\bar{r}_c}]$, $R_3 = DIA_{\bar{P}_{c-1},2}$,

$$R_4 = \begin{bmatrix} Q_{0,c-1} & Q_{1,c-1} & \ldots & Q_{c-1,c-1} & 0 & 0 \\ 0 & Q_{0,c-1} & \ldots & Q_{c-2,c-1} & Q_{c-1,c-1} & 0 \end{bmatrix},$$

$R_5 = DIA_{P_{c-1}^{-1},c+2}$, $R_6 = E_{m(c+2),r_c}$, $R_7 = DIA_{E_{m,-r_c},c+2}$. Then we have $R_5^{-1} = DIA_{P_{c-1},c+2}$, $R_6^{-1} = E_{m(c+2),-r_c}$, $R_7^{-1} = DIA_{E_{m,r_c},c+2}$.

Let $Q = R_1 R_2 R_3 R_4 R_5 R_6 R_7$. Partition $Q = [Q_{0c}, \ldots, Q_{c+1,c}]$, where Q_{jc} has m columns, $j = 0, 1, \ldots, c+1$. We prove that $Q_{c+1,c}$ is 0. For any j, $0 \leqslant j \leqslant c-1$, let $Q'_{j,c-1} = \bar{P}_{c-1} Q_{j,c-1} P_{c-1}^{-1}$ and $Q'_{j,c-1} = \begin{bmatrix} Q^1_{j,c-1} & Q^2_{j,c-1} \\ Q^3_{j,c-1} & Q^4_{j,c-1} \end{bmatrix}$,

or $\begin{bmatrix} Q_j^1 & Q_j^2 \\ Q_j^3 & Q_j^4 \end{bmatrix}$ for short, where $Q_{j,c-1}^1$ and $Q_{j,c-1}^4$ are $\bar{r}_c \times r_c$ and $(m- \bar{r}_c) \times (m - r_c)$ matrices, respectively. Then we have

$$R_3R_4R_5 = \begin{bmatrix} Q'_{0,c-1} & Q'_{1,c-1} & \cdots & Q'_{c-1,c-1} & 0 & 0 \\ 0 & Q'_{0,c-1} & \cdots & Q'_{c-2,c-1} & Q'_{c-1,c-1} & 0 \end{bmatrix}.$$

It follows that

$$R_2R_3R_4R_5R_6 = \begin{bmatrix} Q_0^4 & Q_1^3 & Q_1^4 & Q_2^3 & \ldots & Q_{c-1}^4 & 0 & 0 & 0 & 0 & Q_0^3 \\ 0 & Q_0^1 & Q_0^2 & Q_1^1 & \ldots & Q_{c-2}^2 & Q_{c-1}^1 & Q_{c-1}^2 & 0 & 0 & 0 \end{bmatrix}.$$

Let $R_2R_3R_4R_5R_6 = [\bar{Q}_{0c}, \ldots, \bar{Q}_{c+1,c}]$, where $\bar{Q}_{jc}$ has m columns, $j = 0, 1, \ldots, c+1$. To prove $Q_{0,c-1}^3 = 0$, from (4.4), we have $\bar{B}_{0,c-1} = Q_{0,c-1}B_{0,c-1}$. Thus

$$\bar{P}_{c-1}\bar{B}_{0,c-1} = \bar{P}_{c-1}Q_{0,c-1}P_{c-1}^{-1}(P_{c-1}B_{0,c-1}) = Q'_{0,c-1}(P_{c-1}B_{0,c-1}). \quad (4.6)$$

Since (4.2) is linear over $GF(q)$, the first r_c rows of $P_{c-1}B_{0,c-1}$ are linearly independent over $GF(q)$ and the last $m - r_c$ rows of $P_{c-1}B_{0,c-1}$ are 0. From (4.3), the last $m - \bar{r}_c$ rows of $\bar{P}_{c-1}\bar{B}_{0,c-1}$ are 0. Using (4.6), it follows that $Q_{0,c-1}^3$, the submatrix of the first r_c columns and the last $m - \bar{r}_c$ rows of $Q'_{0,c-1}$, is 0. Since $Q_{0,c-1}^3 = 0$ yields $\bar{Q}_{c+1,c} = 0$, we obtain $\bar{Q}_{c+1,c} = 0$; therefore, $Q_{c+1,c} = R_1\bar{Q}_{c+1,c}E_{m,-r_c} = 0$.

Let $\bar{B}_{j,c-1} = \bar{B}'_{j,c-1} = 0_{m,m}$, $j = h+1, \ldots, h+c-1$, and

$$\Omega_k^r = \begin{bmatrix} B_{0r} & B_{1r} & \ldots & \ldots & B_{hr} & & \\ & B_{0r} & B_{1r} & \ldots & \ldots & B_{hr} & \\ & & \ddots & \ddots & \ddots & \ddots & \ddots \\ & & & B_{0r} & B_{1r} & \ldots & \ldots B_{hr} \end{bmatrix}$$

with mk rows, for any r and any k. From (4.4), we have

$$\sum_{j=0}^{h+c-1} \bar{B}_{j,c-1}z^j = \Big(\sum_{j=0}^{c-1} Q_{j,c-1}z^j\Big)\Big(\sum_{j=0}^{h} B_{j,c-1}z^j\Big).$$

It follows immediately that

$$\begin{bmatrix} \bar{B}_{0,c-1} & \bar{B}_{1,c-1} & \ldots & \bar{B}_{h+c-1,c-1} & 0 & 0 \\ 0 & \bar{B}_{0,c-1} & \ldots & \bar{B}_{h+c-2,c-1} & \bar{B}_{h+c-1,c-1} & 0 \end{bmatrix} = R_4\Omega_{c+2}^{c-1}. \quad (4.7)$$

Let $\bar{B}_{jc} = 0_{m,m}$, $j = h+1, \ldots, h+c$. From (4.3), we have

$$R_1R_2R_3\begin{bmatrix}\bar{B}_{0,c-1} & \bar{B}_{1,c-1} & \dots & \bar{B}_{h+c-1,c-1} & 0 & 0\\ 0 & \bar{B}_{0,c-1} & \dots & \bar{B}_{h+c-2,c-1} & \bar{B}_{h+c-1,c-1} & 0\end{bmatrix}$$
$$= R_1R_2\begin{bmatrix}\bar{B}'_{0,c-1} & \bar{B}'_{1,c-1} & \dots & \bar{B}'_{h+c-1,c-1} & 0 & 0\\ 0 & \bar{B}'_{0,c-1} & \dots & \bar{B}'_{h+c-2,c-1} & \bar{B}'_{h+c-1,c-1} & 0\end{bmatrix}$$
$$= \begin{bmatrix}0 & \bar{B}_{0c} & \dots & \bar{B}_{h+c-2,c} & \bar{B}_{h+c-1,c} & \bar{B}_{h+c,c}\end{bmatrix}.$$

Using (4.7), we have

$$\begin{bmatrix}0 & \bar{B}_{0c} & \dots & \bar{B}_{h+c-2,c} & \bar{B}_{h+c-1,c} & \bar{B}_{h+c,c}\end{bmatrix} = R_1R_2R_3R_4\Omega_{c+2}^{c-1}$$
$$= R_1R_2R_3R_4R_5R_6R_7R_7^{-1}R_6^{-1}R_5^{-1}\Omega_{c+2}^{c-1} = Q(R_7^{-1}R_6^{-1}R_5^{-1}\Omega_{c+2}^{c-1}).$$

From (4.2), it follows that

$$\begin{bmatrix}0 & \bar{B}_{0c} & \dots & \bar{B}_{h+c-2,c} & \bar{B}_{h+c-1,c} & \bar{B}_{h+c,c}\end{bmatrix} = Q\begin{bmatrix}0 & \Omega_{c+1}^{c}\\ G' & G''\end{bmatrix} \tag{4.8}$$

for some matrices G' and G'' with m rows. Since $Q = [Q_{0c}, \dots, Q_{c+1,c}]$ and $Q_{c+1,c} = 0$, (4.8) yields

$$[\bar{B}_{0c}, \dots, \bar{B}_{h+c,c}] = [Q_{0c}, \dots, Q_{cc}]\Omega_{c+1}^{c},$$

which is equivalent to (4.5) because $\bar{B}_{jc} = 0$ for $h+1 \leqslant j \leqslant h+c$. That is, (4.5) holds.

Now we suppose that $Q_{0,c-1}$ is nonsingular and (4.3) is linear over $GF(q)$. Since (4.3) and (4.2) are linear over $GF(q)$, $\bar{r}_c$ and r_c are the ranks of $\bar{B}_{0,c-1}$ and $B_{0,c-1}$, respectively. Since $\bar{B}_{0,c-1} = Q_{0,c-1}B_{0,c-1}$ and $Q_{0,c-1}$ is nonsingular, we have $\bar{r}_c = r_c$.

Since $\bar{P}_{c-1}$, P_{c-1} and $Q_{0,c-1}$ are nonsingular, $Q'_{0,c-1} = \bar{P}_{c-1}Q_{0,c-1}P_{c-1}^{-1}$ is nonsingular. Noticing $\bar{r}_c = r_c$ and $Q^3_{0,c-1} = 0$, it follows that $Q^4_{0,c-1}$ and $Q^1_{0,c-1}$ are nonsingular. Thus $\bar{Q}_{0c} = \begin{bmatrix}Q^4_{0,c-1} & Q^3_{1,c-1}\\ 0 & Q^1_{0,c-1}\end{bmatrix}$ is nonsingular. Therefore, $Q_{0c} = R_1\bar{Q}_{0c}E_{m,-r_c}$ is nonsingular. □

Theorem 4.1.1. *Assume that $eq_0(i)$ and $\bar{eq}_0(i)$ are the same. Assume that*

$$eq_c(i) \overset{R_a[P_c]}{\longrightarrow} eq'_c(i),\ eq'_c(i) \overset{R_b[r_{c+1}]}{\longrightarrow} eq_{c+1}(i),\quad c = 0, 1, \dots, \tau-1 \tag{4.9}$$

is a linear R_a R_b transformation sequence and

$$\bar{eq}_c(i) \overset{R_a[\bar{P}_c]}{\longrightarrow} \bar{eq}'_c(i),\ \bar{eq}'_c(i) \overset{R_b[\bar{r}_{c+1}]}{\longrightarrow} \bar{eq}_{c+1}(i),\quad c = 0, 1, \dots, \tau-1 \tag{4.10}$$

is an R_a R_b transformation sequence. Then there exist $m \times m$ matrices $Q_{j\tau}$, $j = 0, 1, \dots, \tau$ over $GF(q)$ such that

$$\sum_{j=0}^{h} \bar{B}_{j\tau} z^j = \Big(\sum_{j=0}^{\tau} Q_{j\tau} z^j\Big)\Big(\sum_{j=0}^{h} B_{j\tau} z^j\Big).$$

Moreover, if (4.10) *is linear over* $GF(q)$*, then* $\bar{r}_j = r_j$ *holds for any* j*,* $1 \leqslant j \leqslant \tau$ *and* $Q_{0\tau}$ *is nonsingular.*

Proof. Since $\bar{eq}_0(i)$ and $eq_0(i)$ are the same, (4.4) holds in the case of $c = 1$, where Q_{00} is the identity matrix. Applying Lemma 4.1.1 τ times, c from 1 to τ, we obtain the theorem. □

Theorem 4.1.2. *Let* $eq_0(i)$ *and* $\bar{eq}_0(i)$ *be the same and equivalent to the equation*

$$\varphi_0(y(i, t+1)) + [B_{00}, \ldots, B_{h0}]\psi^{lh}_{\mu\nu}(u, x, i) = 0.$$

(a) *If* (4.10) *is an* R_a R_b *transformation sequence and* $\bar{B}_{0\tau}\psi^{l}_{\mu\nu}(u(i, \mu+1), x(i, \nu+1))$ *as a function of the variable* x_i *is an injection, then for any linear* R_a R_b *transformation sequence* (4.9), $B_{0\tau}\psi^{l}_{\mu\nu}(u(i, \mu+1), x(i, \nu+1))$ *as a function of the variable* x_i *is an injection.*

(b) *If* (4.10) *is an* R_a R_b *transformation sequence and* $\bar{B}_{0\tau}\psi^{l}_{\mu\nu}(u(i, \mu+1), x(i, \nu+1))$ *as a function of the variable* x_i *is a surjection, then for any linear* R_a R_b *transformation sequence* (4.9), $B_{0\tau}\psi^{l}_{\mu\nu}(u(i, \mu+1), x(i, \nu+1))$ *as a function of the variable* x_i *is a surjection.*

Proof. (a) Suppose that (4.10) is an R_a R_b transformation sequence and $\bar{B}_{0\tau}\psi^{l}_{\mu\nu}(u(i, \mu+1), x(i, \nu+1))$ as a function of the variable x_i is an injection. For any linear (4.9), from Theorem 4.1.1, there exists an $m \times m$ matrix $Q_{0\tau}$ such that $\bar{B}_{0\tau} = Q_{0\tau}B_{0\tau}$. It follows immediately that for any parameters x_{i-1}, …, $x_{i-\nu}$, u_i, …, $u_{i-\mu}$, $B_{0\tau}\psi^{l}_{\mu\nu}(u(i, \mu+1), x(i, \nu+1))$ as a function of the variable x_i is an injection.

(b) The proof of part (b) is similar to part (a), just by replacing "injection" by "surjection". □

Notice that if for any linear R_a R_b transformation sequence (4.9), $B_{0\tau}\psi^{l}_{\mu\nu}(u(i, \mu+1), x(i, \nu+1))$ as a function of the variable x_i is a surjection, then for any parameters x_{i-1}, …, $x_{i-h-\nu}$, u_i, …, $u_{i-h-\mu}$, $y_{i+\tau}$, …, y_{i-t}, the equation $eq_\tau(i)$ has a solution x_i. If for any linear R_a R_b transformation sequence (4.9), $B_{0\tau}\psi^{l}_{\mu\nu}(u(i, \mu+1), x(i, \nu+1))$ as a function of the variable x_i is an injection, then for any parameters x_{i-1}, …, $x_{i-h-\nu}$, u_i, …, $u_{i-h-\mu}$, $y_{i+\tau}$, …, y_{i-t}, the equation $eq_\tau(i)$ has at most one solution x_i.

4.2 Composition of R_a R_b Transformations

Let X and U be two finite nonempty sets. Let Y' and Y be two column vector spaces over a finite field $GF(q)$ of dimensions m' and m, respectively, where m' and m are two positive integers.

For any nonnegative integer c, let $1eq_c(i)$ be an equation

$$\xi_c(y(i+c,c+1)) + [F_{0c},\ldots,F_{h_1c}]\psi_{\mu\nu}^{lh_1}(u,x,i) = 0$$

and let $1eq'_c(i)$ be an equation

$$\xi'_c(y(i+c,c+1)) + [F'_{0c},\ldots,F'_{h_1c}]\psi_{\mu\nu}^{lh_1}(u,x,i) = 0,$$

where ξ_c and ξ'_c are two single-valued mappings from Y'^{c+1} to Y', F_{jc} and F'_{jc} are $m' \times l$ matrices over $GF(q)$, $j = 0,1,\ldots,h_1$.

For any nonnegative integer c, let $0eq_c(i)$ be an equation

$$\eta_c(y(i+c,c+k+1)) + [B_{0c},\ldots,B_{h_0c}]\begin{bmatrix} x_i \\ \vdots \\ x_{i-h_0} \end{bmatrix} = 0$$

and let $0eq'_c(i)$ be an equation

$$\eta'_c(y(i+c,c+k+1)) + [B'_{0c},\ldots,B'_{h_0c}]\begin{bmatrix} x_i \\ \vdots \\ x_{i-h_0} \end{bmatrix} = 0,$$

where η_c and η'_c are two single-valued mappings from Y^{c+k+1} to Y, B_{jc} and B'_{jc} are $m \times m'$ matrices over $GF(q)$, $j = 0,1,\ldots,h_0$.

Let $h = h_0 + h_1$. In this section, for any nonnegative integer c, let $eq_c(i)$ be an equation

$$\varphi_c(y(i+c,c+k+1)) + [C_{0c},\ldots,C_{hc}]\psi_{\mu\nu}^{lh}(u,x,i) = 0$$

and let $eq'_c(i)$ be an equation

$$\varphi'_c(y(i+c,c+k+1)) + [C'_{0c},\ldots,C'_{hc}]\psi_{\mu\nu}^{lh}(u,x,i) = 0,$$

where φ_c and φ'_c are two single-valued mappings from $Y^{c+k+1}$to Y, C_{jc} and C'_{jc} are $m \times l$ matrices over $GF(q)$, $j = 0,1,\ldots,h$.

For any nonnegative integer r, let

$$\Gamma_{h+r} = \begin{bmatrix} F_0 & F_1 & \ldots & \ldots & F_{h_1} & & & \\ & F_0 & F_1 & \ldots & \ldots & F_{h_1} & & \\ & & \ddots & \ddots & \ddots & \ddots & \ddots & \\ & & & F_0 & F_1 & \ldots & \ldots & F_{h_1} \end{bmatrix} \tag{4.11}$$

be an $(h_0+r)m' \times (h+r)l$ matrix, where F_j, $j = 0,1,\ldots,h_1$ are $m' \times l$ matrices over $GF(q)$.

Lemma 4.2.1. *Let $c > 0$ and*

$$[C_{0,c-1}, \ldots, C_{h,c-1}] = [B_{0,c-1}, \ldots, B_{h_0,c-1}]\Gamma_{h+1}. \tag{4.12}$$

If

$$0eq_{c-1}(i) \overset{R_a[P_{c-1}]}{\longrightarrow} 0eq'_{c-1}(i),\ 0eq'_{c-1}(i) \overset{R_b[r_c]}{\longrightarrow} 0eq_c(i), \tag{4.13}$$

then

$$eq_{c-1}(i) \overset{R_a[P_{c-1}]}{\longrightarrow} eq'_{c-1}(i),\ eq'_{c-1}(i) \overset{R_b[r_c]}{\longrightarrow} eq_c(i) \tag{4.14}$$

and

$$[C_{0c}, \ldots, C_{hc}] = [B_{0c}, \ldots, B_{h_0c}]\Gamma_{h+1}. \tag{4.15}$$

Proof. Let $R_1 = E_{m,r_c}$, $R_2 = [0_{m,r_c} E_m\ 0_{m,m-r_c}]$, $R_3 = DIA_{P_{c-1},2}$. By (4.13), we have

$$\begin{aligned} & R_1R_2R_3 \begin{bmatrix} B_{0,c-1} & B_{1,c-1} & \ldots & B_{h_0,c-1} & 0 \\ 0 & & B_{0,c-1} & \ldots & B_{h_0-1,c-1} & B_{h_0,c-1} \end{bmatrix} \\ & = R_1R_2 \begin{bmatrix} B'_{0,c-1} & B'_{1,c-1} & \ldots & B'_{h_0,c-1} & 0 \\ 0 & & B'_{0,c-1} & \ldots & B'_{h_0-1,c-1} & B'_{h_0,c-1} \end{bmatrix} \\ & = \begin{bmatrix} 0 & B_{0c} & \ldots & B_{h_0-1,c} & B_{h_0c} \end{bmatrix}. \end{aligned} \tag{4.16}$$

We prove that (4.14) is an R_a R_b transformation sequence. It is sufficient to prove that the last $m - r_c$ rows of $P_{c-1}C_{0,c-1}$ are 0. Let $C'_{0,c-1} = P_{c-1}C_{0,c-1}$. Since (4.12) yields $C_{0,c-1} = B_{0,c-1}F_0$, we have $C'_{0,c-1} = B'_{0,c-1}F_0$. From (4.13), it is easy to see that the last $m - r_c$ rows of $B'_{0,c-1}$ are 0. It follows immediately that the last $m - r_c$ rows of $C'_{0,c-1}$ are 0.

Since (4.14) holds, we have

$$\begin{aligned} & R_1R_2R_3 \begin{bmatrix} C_{0,c-1} & C_{1,c-1} & \ldots & C_{h,c-1} & 0 \\ 0 & & C_{0,c-1} & \ldots & C_{h-1,c-1} & C_{h,c-1} \end{bmatrix} \\ & = R_1R_2 \begin{bmatrix} C'_{0,c-1} & C'_{1,c-1} & \ldots & C'_{h,c-1} & 0 \\ 0 & & C'_{0,c-1} & \ldots & C'_{h-1,c-1} & C'_{h,c-1} \end{bmatrix} \\ & = \begin{bmatrix} 0 & C_{0c} & \ldots & C_{h-1,c} & C_{hc} \end{bmatrix}. \end{aligned}$$

From (4.12) and (4.16), it follows that

$$\begin{aligned} & \begin{bmatrix} 0 & C_{0c} & \ldots & C_{h-1,c} & C_{hc} \end{bmatrix} \\ & = R_1R_2R_3 \begin{bmatrix} B_{0,c-1} & B_{1,c-1} & \ldots & B_{h_0,c-1} & 0 \\ 0 & & B_{0,c-1} & \ldots & B_{h_0-1,c-1} & B_{h_0,c-1} \end{bmatrix} \Gamma_{h+2} \\ & = \begin{bmatrix} 0 & B_{0c} & \ldots & B_{h_0-1,c} & B_{h_0c} \end{bmatrix} \Gamma_{h+2}. \end{aligned}$$

Therefore, (4.15) holds. □

For any nonnegative integers c and r, let Γ^c_{h+r} be an $(h_0+r)m' \times (h+r)l$ matrix

$$\Gamma^c_{h+r} = \begin{bmatrix} F_{0c} & F_{1c} & \dots & \dots & F_{h_1c} & & \\ & F_{0c} & F_{1c} & \dots & \dots & F_{h_1c} & \\ & & \ddots & \ddots & \ddots & \ddots & \ddots \\ & & & F_{0c} & F_{1c} & \dots & \dots F_{h_1c} \end{bmatrix}. \tag{4.17}$$

Lemma 4.2.2. *Assume that $\tau_0 \geqslant 0$, $c > 0$,*

$$[C_{0,\tau_0+c-1}, \dots, C_{h+c-1,\tau_0+c-1}] = [Q_{0,c-1}, \dots, Q_{h_0+c-1,c-1}]\Gamma^{c-1}_{h+c}, \tag{4.18}$$

$C_{h+j,\tau_0+c-1} = 0$ for $j > 0$, where $Q_{j,c-1}$ is an $m \times m'$ matrix over $GF(q)$, $j = 0, 1, \dots, h_0 + c - 1$. If

$$1eq_{c-1}(i) \xrightarrow{R_a[P_{c-1}]} 1eq'_{c-1}(i),\ 1eq'_{c-1}(i) \xrightarrow{R_b[r_c]} 1eq_c(i), \tag{4.19}$$

then there exist an $m \times m$ invertible matrix $\bar{P}_{\tau_0+c-1}$ over $GF(q)$ and a nonnegative integer $\bar{r}_{\tau_0+c}$ such that

$$eq_{\tau_0+c-1}(i) \xrightarrow{R_a[\bar{P}_{\tau_0+c-1}]} eq'_{\tau_0+c-1}(i),\ eq'_{\tau_0+c-1}(i) \xrightarrow{R_b[\bar{r}_{\tau_0+c}]} eq_{\tau_0+c}(i) \tag{4.20}$$

and there exist $m \times m'$ matrices Q_{0c}, …, $Q_{h_0+c,c}$ over $GF(q)$ such that

$$[C_{0,\tau_0+c}, \dots, C_{h+c,\tau_0+c}] = [Q_{0c}, \dots, Q_{h_0+c,c}]\Gamma^c_{h+c+1}$$

and the rank of Q_{0c} is not less than the rank of $Q_{0,c-1}$, where $C_{h+j,\tau_0+c} = 0$ for $j > 0$.

Proof. Let $Q'_{0,c-1}$ be the reduced echelon matrix of $Q_{0,c-1}P^{-1}_{c-1}$, and $\bar{P}_{\tau_0+c-1}$ an $m \times m$ invertible matrix with $Q'_{0,c-1} = \bar{P}_{\tau_0+c-1}Q_{0,c-1}P^{-1}_{c-1}$. Let $\bar{r}_{\tau_0+c}$ be the rank of the submatrix of the first r_c columns of $Q'_{0,c-1}$.

Let $R_1 = E_{m,\bar{r}_{\tau_0+c}}$, $R_2 = [0_{m,\bar{r}_{\tau_0+c}} E_m 0_{m,m-\bar{r}_{\tau_0+c}}]$, $R_3 = DIA_{\bar{P}_{\tau_0+c-1},2}$,

$$R_4 = \begin{bmatrix} Q_{0,c-1} & Q_{1,c-1} & \dots & Q_{h_0+c-1,c-1} & 0 & & 0 \\ 0 & & Q_{0,c-1} & \dots & Q_{h_0+c-2,c-1} & Q_{h_0+c-1,c-1} & 0 \end{bmatrix},$$

$R_5 = DIA_{P^{-1}_{c-1},h_0+c+2}$, $R_6 = E_{m'(h_0+c+2),r_c}$, $R_7 = DIA_{E_{m',-r_c},h_0+c+2}$. It follows that $R_5^{-1} = DIA_{P_{c-1},h_0+c+2}$, $R_6^{-1} = E_{m'(h_0+c+2),-r_c}$, and $R_7^{-1} = DIA_{E_{m',r_c},h_0+c+2}$. Let $Q = R_1R_2R_3R_4R_5R_6R_7$. Partition $Q = [Q_{0c}, \dots, Q_{h_0+c+1,c}]$, where Q_{jc} has m' columns, $j = 0, 1, \dots, h_0 + c + 1$.

For any j, $1 \leqslant j \leqslant h_0 + c - 1$, let $Q'_{j,c-1} = \bar{P}_{\tau_0+c-1}Q_{j,c-1}P^{-1}_{c-1}$. For any j, $0 \leqslant j \leqslant h_0 + c - 1$, let $Q'_{j,c-1} = \begin{bmatrix} Q^1_{j,c-1} & Q^2_{j,c-1} \\ Q^3_{j,c-1} & Q^4_{j,c-1} \end{bmatrix}$, or $\begin{bmatrix} Q^1_j & Q^2_j \\ Q^3_j & Q^4_j \end{bmatrix}$ for short, where

$Q^1_{j,c-1}$ and $Q^4_{j,c-1}$ are $\bar{r}_{\tau_0+c} \times r_c$ and $(m- \bar{r}_{\tau_0+c}) \times (m'- r_c)$ matrices, respectively. We prove $Q^3_{0,c-1} = 0$ and $Q_{h_0+c+1,c} = 0$. Since $Q'_{0,c-1}$ is the reduced echelon matrix of $Q_{0,c-1}P^{-1}_{c-1}$, from the definition of $\bar{r}_{\tau_0+c}$, we have $Q^3_{0,c-1} = 0$. Clearly,

$$R_3R_4R_5 = \begin{bmatrix} Q'_{0,c-1} & Q'_{1,c-1} \cdots Q'_{h_0+c-1,c-1} & 0 & 0 \\ 0 & Q'_{0,c-1} \cdots Q'_{h_0+c-2,c-1} & Q'_{h_0+c-1,c-1} & 0 \end{bmatrix}.$$

Therefore, we have

$$R_2R_3R_4R_5R_6 = \begin{bmatrix} Q^4_0 & Q^3_1 & Q^4_1 & Q^3_2 \cdots Q^4_{h_0+c-1} & 0 & 0 & 0 & 0 & Q^3_0 \\ 0 & Q^1_0 & Q^2_0 & Q^1_1 \cdots Q^2_{h_0+c-2} & Q^1_{h_0+c-1} & Q^2_{h_0+c-1} & 0 & 0 & 0 \end{bmatrix}.$$

Let $R_2R_3R_4R_5R_6 = [\bar{Q}_{0c}, \ldots, \bar{Q}_{h_0+c+1,c}]$, where $\bar{Q}_{jc}$ has m' columns, $j = 0, 1, \ldots, h_0+c+1$. Since $Q^3_{0,c-1} = 0$ yields $\bar{Q}_{h_0+c+1,c} = 0$, we obtain $\bar{Q}_{h_0+c+1,c} = 0$ and $Q_{h_0+c+1,c} = R_1\bar{Q}_{h_0+c+1,c}E_{m',-r_c} = 0$.

We prove that (4.20) is an R_a R_b transformation sequence. It is sufficient to prove that the last $m - \bar{r}_{\tau_0+c}$ rows of $\bar{P}_{\tau_0+c-1}C_{0,\tau_0+c-1}$ are 0. Let $C'_{0,\tau_0+c-1} = \bar{P}_{\tau_0+c-1}C_{0,\tau_0+c-1}$. Since (4.18) yields $C_{0,\tau_0+c-1} = Q_{0,c-1}F_{0,c-1}$, we have

$$\begin{aligned} C'_{0,\tau_0+c-1} &= \bar{P}_{\tau_0+c-1}Q_{0,c-1}F_{0,c-1} \\ &= (\bar{P}_{\tau_0+c-1}Q_{0,c-1}P^{-1}_{c-1})(P_{c-1}F_{0,c-1}) \\ &= Q'_{0,c-1}F'_{0,c-1}. \end{aligned}$$

Since $Q^3_{0,c-1} = 0$ and the last $m'- r_c$ rows of $F'_{0,c-1}$ are 0, the last $m- \bar{r}_{\tau_0+c}$ rows of C'_{0,τ_0+c-1} are 0.

To prove $[C_{0,\tau_0+c}, \ldots, C_{h+c,\tau_0+c}] = [Q_{0c}, \ldots, Q_{h_0+c,c}]\Gamma^c_{h+c+1}$, using (4.19), it is easy to verify that $R^{-1}_5\Gamma^{c-1}_{h+c+2}$ equals

$$\begin{bmatrix} F'_{0,c-1} & F'_{1,c-1} \cdots & \cdots & F'_{h_1,c-1} & & \\ & F'_{0,c-1} & F'_{1,c-1} \cdots & \cdots & F'_{h_1,c-1} & \\ & & \ddots \quad \ddots & \ddots \quad \ddots & \ddots \quad \ddots & \ddots \\ & & & F'_{0,c-1} & F'_{1,c-1} \cdots & \cdots & F'_{h_1,c-1} \end{bmatrix}$$

and $R^{-1}_7R^{-1}_6R^{-1}_5\Gamma^{c-1}_{h+c+2} = \begin{bmatrix} 0 & \Gamma^c_{h+c+1} \\ G' & G'' \end{bmatrix}$ for some matrices G' and G'' with m' rows. Since $Q = [Q_{0c}, \ldots, Q_{h_0+c+1,c}]$ and $Q_{h_0+c+1,c} = 0$, we have

$$QR^{-1}_7R^{-1}_6R^{-1}_5\Gamma^{c-1}_{h+c+2} = [Q_{0c}, \ldots, Q_{h_0+c,c}]\,[0, \Gamma^c_{h+c+1}]. \tag{4.21}$$

From (4.20) and $C_{h+j,\tau_0+c-1} = 0$ for $j > 0$, we have

$$R_1R_2R_3\begin{bmatrix} C_{0,\tau_0+c-1} & C_{1,\tau_0+c-1} & \dots & C_{h+c-1,\tau_0+c-1} & 0 & 0 \\ 0 & C_{0,\tau_0+c-1} & \dots & C_{h+c-2,\tau_0+c-1} & C_{h+c-1,\tau_0+c-1} & 0 \end{bmatrix}$$
$$= R_1R_2\begin{bmatrix} C'_{0,\tau_0+c-1} & C'_{1,\tau_0+c-1} & \dots & C'_{h+c-1,\tau_0+c-1} & 0 & 0 \\ 0 & C'_{0,\tau_0+c-1} & \dots & C'_{h+c-2,\tau_0+c-1} & C'_{h+c-1,\tau_0+c-1} & 0 \end{bmatrix}$$
$$= \begin{bmatrix} 0 & C_{0,\tau_0+c} & \dots & C_{h+c-2,\tau_0+c} & C_{h+c-1,\tau_0+c} & 0 \end{bmatrix} \tag{4.22}$$

and $C_{h+j,\tau_0+c} = 0$ for $j > 0$. On the other hand, from (4.18), we obtain

$$R_1R_2R_3\begin{bmatrix} C_{0,\tau_0+c-1} & C_{1,\tau_0+c-1} & \dots & C_{h+c-1,\tau_0+c-1} & 0 & 0 \\ 0 & C_{0,\tau_0+c-1} & \dots & C_{h+c-2,\tau_0+c-1} & C_{h+c-1,\tau_0+c-1} & 0 \end{bmatrix}$$
$$= R_1R_2R_3R_4\Gamma^{c-1}_{h+c+2} = R_1R_2R_3R_4R_5R_6R_7R_7^{-1}R_6^{-1}R_5^{-1}\Gamma^{c-1}_{h+c+2}$$
$$= QR_7^{-1}R_6^{-1}R_5^{-1}\Gamma^{c-1}_{h+c+2}.$$

Using (4.22) and (4.21), it follows immediately that

$$[0, C_{0,\tau_0+c}, \dots, C_{h+c,\tau_0+c}] = QR_7^{-1}R_6^{-1}R_5^{-1}\Gamma^{c-1}_{h+c+2}$$
$$= [Q_{0c}, \dots, Q_{h_0+c,c}]\,[0, \Gamma^c_{h+c+1}],$$

where $C_{h+c,\tau_0+c} = 0$. Therefore,

$$[C_{0,\tau_0+c}, \dots, C_{h+c,\tau_0+c}] = [Q_{0c}, \dots, Q_{h_0+c,c}]\Gamma^c_{h+c+1}.$$

Finally, we prove that the rank of Q_{0c} is not less than the rank of $Q_{0,c-1}$. It is easy to see that the rank of $Q_{0,c-1}$ is equal to the rank of $Q'_{0,c-1}$. From $Q_{0c} = R_1\bar{Q}_{0c}E_{m',-r_c}$, the rank of Q_{0c} is equal to the rank of $\bar{Q}_{0c}$. Since $Q'_{0,c-1} = \begin{bmatrix} Q^1_{0,c-1} & Q^2_{0,c-1} \\ 0 & Q^4_{0,c-1} \end{bmatrix}$ and $\bar{r}_{\tau_0+c}$ is the rank of $\begin{bmatrix} Q^1_{0,c-1} \\ 0 \end{bmatrix}$, the rows of $Q^1_{0,c-1}$ are linearly independent. It follows that the rank of $Q'_{0,c-1}$ is equal to the sum of the rank of $Q^1_{0,c-1}$ and the rank of $Q^4_{0,c-1}$. On the other hand, $\bar{Q}_{0c} = \begin{bmatrix} Q^4_{0,c-1} & Q^3_{1,c-1} \\ 0 & Q^1_{0,c-1} \end{bmatrix}$ deduces that the rank of $\bar{Q}_{0c}$ is equal to or greater than the sum of the rank of $Q^1_{0,c-1}$ and the rank of $Q^4_{0,c-1}$. Therefore, the rank of $\bar{Q}_{0c}$ is equal to or greater than the rank of $Q'_{0,c-1}$. It follows that the rank of Q_{0c} is equal to or greater than the rank of $Q_{0,c-1}$. □

Theorem 4.2.1. *Let*

$$[C_{00}, \dots, C_{h0}] = [B_{00}, \dots, B_{h_0 0}]\Gamma^0_{h+1}. \tag{4.23}$$

Assume that

$$0eq_c(i) \xrightarrow{R_a[P_c]} 0eq'_c(i),\ 0eq'_c(i) \xrightarrow{R_b[r_{c+1}]} 0eq_{c+1}(i),\quad c = 0, 1, \dots, \tau_0 - 1 \tag{4.24}$$

and

$$1eq_c(i) \overset{R_a[P'_c]}{\longrightarrow} 1eq'_c(i),\ 1eq'_c(i) \overset{R_b[r'_{c+1}]}{\longrightarrow} 1eq_{c+1}(i),\quad c = 0, 1, \ldots, \tau_1 - 1 \quad (4.25)$$

are two R_a R_b transformation sequences. Then there exist an R_a R_b transformation sequence

$$eq_c(i) \overset{R_a[\bar{P}_c]}{\longrightarrow} eq'_c(i),\ eq'_c(i) \overset{R_b[\bar{r}_{c+1}]}{\longrightarrow} eq_{c+1}(i),\quad c = 0, 1, \ldots, \tau - 1$$

and $m \times m'$ matrices $Q_{0\tau_1}$, ..., $Q_{h_0+\tau_1,\tau_1}$ over $GF(q)$ such that

$$[C_{0\tau}, \ldots, C_{h+\tau_1,\tau}] = [Q_{0\tau_1}, \ldots, Q_{h_0+\tau_1,\tau_1}]\Gamma^{\tau_1}_{h+\tau_1+1} \quad (4.26)$$

and the rank of $Q_{0\tau_1}$ is not less than the rank of $B_{0\tau_0}$ in $0eq_{\tau_0}(i)$, where $\tau = \tau_0+\tau_1$, $C_{h+j,\tau} = 0$ for $j > 0$. Therefore, if the rank of $B_{0\tau_0}$ is $\min(m, m')$, then the rank of $Q_{0\tau_1}$ is $\min(m, m')$.

Proof. From (4.23), (4.12) for $c = 1$ in Lemma 4.2.1 holds, where $F_j = F_{j0}$, $j = 0, 1, \ldots, h_1$. From (4.24), using Lemma 4.2.1 τ_0 times, c from 1 to τ_0, we obtain

$$eq_c(i) \overset{R_a[P_c]}{\longrightarrow} eq'_c(i),\ eq'_c(i) \overset{R_b[r_{c+1}]}{\longrightarrow} eq_{c+1}(i),\quad c = 0, 1, \ldots, \tau_0 - 1 \quad (4.27)$$

and

$$[C_{0\tau_0}, \ldots, C_{h\tau_0}] = [B_{0\tau_0}, \ldots, B_{h_0\tau_0}]\Gamma^0_{h+1}. \quad (4.28)$$

Let $Q_{j0} = B_{j\tau_0}$ for any j, $0 \leqslant j \leqslant h_0$. Then the rank of $B_{0\tau_0}$ is the rank of Q_{00}. Clearly, (4.28) yields (4.18) for $c = 1$ in Lemma 4.2.2. From (4.25), using Lemma 4.2.2 τ_1 times, c from 1 to τ_1, we obtain that there exist

$$eq_c(i) \overset{R_a[\bar{P}_c]}{\longrightarrow} eq'_c(i),\ eq'_c(i) \overset{R_b[\bar{r}_{c+1}]}{\longrightarrow} eq_{c+1}(i),\quad c = \tau_0, \tau_0 + 1, \ldots, \tau - 1 \quad (4.29)$$

and $Q_{0\tau_1}$, ..., $Q_{h_0+\tau_1,\tau_1}$ such that (4.26) holds and the rank of $Q_{0\tau_1}$ is not less than the rank of Q_{00}, where $\tau = \tau_0 + \tau_1$, $C_{h+j,\tau} = 0$ for $j > 0$. Thus the rank of $Q_{0\tau_1}$ is not less than the rank of $B_{0\tau_0}$. Letting $\bar{P}_j = P_j$ and $\bar{r}_j = r_j$ for any j, $0 \leqslant j \leqslant \tau_0 - 1$, from (4.27) and (4.29),

$$eq_c(i) \overset{R_a[\bar{P}_c]}{\longrightarrow} eq'_c(i),\ eq'_c(i) \overset{R_b[\bar{r}_{c+1}]}{\longrightarrow} eq_{c+1}(i),\quad c = 0, 1, \ldots, \tau - 1$$

is an R_a R_b transformation sequence. This completes the proof of the theorem. □

Let $M_0 = \langle Y', Y, Y^k \times Y'^{h_0}, \delta_0, \lambda_0 \rangle$ be a special h_0-quasi-linear finite automaton defined by

$$y_i = \varphi_{out}(y(i-1,k)) + [B_0,\ldots,B_{h_0}]\begin{bmatrix} y'_i \\ \vdots \\ y'_{i-h_0}\end{bmatrix},\ i=0,1,\ldots \tag{4.30}$$

Let $M_1 = \langle X, Y', U^{p+1}\times X^{h_1}, \delta_1, \lambda_1\rangle$ be a pseudo-memory finite automaton defined by

$$\begin{aligned} y'_i &= f(u(i,p+1), x(i,h_1+1)),\\ u_{i+1} &= g(u(i,p+1), x(i,h_1+1)),\\ &\quad i=0,1,\ldots, \end{aligned} \tag{4.31}$$

and suppose that f can be expressed in the form

$$f(u(i,p+1), x(i,h_1+1)) = [F_0,\ldots,F_{h_1}]\psi^{lh_1}_{\mu\nu}(u,x,i).$$

Let $h = h_0 + h_1$ and

$$[C_0, C_1, \ldots, C_h] = [B_0, B_1, \ldots, B_{h_0}]\Gamma_{h+1}, \tag{4.32}$$

where Γ_{h+1} is defined by (4.11).

From M_0 and M_1, a finite automaton $\langle X, Y, Y^k \times U^{p+1} \times X^h, \delta, \lambda\rangle$ is defined by

$$\begin{aligned} y_i &= \varphi_{out}(y(i-1,k)) + [C_0,\ldots,C_h]\psi^{lh}_{\mu\nu}(u,x,i),\\ u_{i+1} &= g(u(i,p+1), x(i,h_1+1)),\\ &\quad i=0,1,\ldots \end{aligned}$$

Theorem 4.2.2. *Let M_0 and M_1 be finite automata defined by* (4.30) *and* (4.31)*, respectively. Let $1eq_0(i)$ be equivalent to the equation*

$$-y_i + [F_0,\ldots,F_{h_1}]\psi^{lh_1}_{\mu\nu}(u,x,i) = 0,$$

and $eq_0(i)$ be equivalent to the equation

$$-y_i + \varphi_{out}(y(i-1,k)) + [C_0,\ldots,C_h]\psi^{lh}_{\mu\nu}(u,x,i) = 0. \tag{4.33}$$

Assume that (4.25) *is an R_a R_b transformation sequence.*

(a) *If $F_{0\tau_1}\psi^l_{\mu\nu}(u(i,\mu+1), x(i,\nu+1))$ in $1eq_{\tau_1}(i)$ as a function of the variable x_i is an injection and M_0 is weakly invertible with delay τ_0, then for any linear R_a R_b transformation sequence*

$$eq_c(i) \overset{R_a[P_c]}{\longrightarrow} eq'_c(i),\ eq'_c(i) \overset{R_b[r_{c+1}]}{\longrightarrow} eq_{c+1}(i),\quad c=0,1,\ldots,\tau-1 \tag{4.34}$$

with $\tau = \tau_0 + \tau_1$, $C_{0\tau}\psi^l_{\mu\nu}(u(i,\mu+1), x(i,\nu+1))$ in $eq_\tau(i)$ as a function of the variable x_i is an injection.

(b) *If* $F_{0\tau_1}\psi^l_{\mu\nu}(u(i,\mu+1),x(i,\nu+1))$ *in* $1eq_{\tau_1}(i)$ *as a function of the variable* x_i *is a surjection and* M_0 *is a weak inverse with delay* τ_0*, then for any linear* R_a R_b *transformation sequence* (4.34) *with* $\tau = \tau_0 + \tau_1$, $C_{0\tau}\psi^l_{\mu\nu}(u(i,\mu+1),x(i,\nu+1))$ *in* $eq_\tau(i)$ *as a function of the variable* x_i *is a surjection.*

Proof. (a) Suppose that M_0 is weakly invertible with delay τ_0. Then $m \geqslant m'$. Let $0eq_0(i)$ be equivalent to the equation

$$-y_i + \varphi_{out}(y(i-1,k)) + [B_0,\ldots,B_{h_0}]\begin{bmatrix} x_i \\ \vdots \\ x_{i-h_0}\end{bmatrix} = 0.$$

Let (4.24) be a linear R_a R_b transformation sequence. From Theorem 3.3.2 in Chap. 3, the rank of $B_{0\tau_0}$ in $0eq_{\tau_0}(i)$ is m'. Using (4.32), (4.33), (4.11) and (4.17), it is easy to see that (4.23) holds. From Theorem 4.2.1, there exist an R_a R_b transformation sequence

$$\bar{eq}_c(i) \overset{R_a[\bar{P}_c]}{\longrightarrow} \bar{eq}'_c(i),\ \bar{eq}'_c(i) \overset{R_b[\bar{r}_{c+1}]}{\longrightarrow} \bar{eq}_{c+1}(i),\quad c = 0,1,\ldots,\tau-1$$

with $\tau = \tau_0 + \tau_1$ and $m \times m'$ matrices $Q_{0\tau_1}, \ldots, Q_{h_0+\tau_1,\tau_1}$ over $GF(q)$ such that

$$[\bar{C}_{0\tau},\ldots,\bar{C}_{h+\tau_1,\tau}] = [Q_{0\tau_1},\ldots,Q_{h_0+\tau_1,\tau_1}]\Gamma^{\tau_1}_{h+\tau_1+1} \tag{4.35}$$

holds and the rank of $Q_{0\tau_1}$ is m', where $\bar{eq}_0(i)$ is $eq_0(i)$, $\bar{eq}_\tau(i)$ is in the form

$$\bar{\varphi}_\tau(y(i+\tau,\tau+k+1)) + [\bar{C}_{0\tau},\ldots,\bar{C}_{h\tau}]\psi^{lh}_{\mu\nu}(u,x,i) = 0,$$

and $\bar{C}_{h+j,\tau} = 0$ for $j > 0$. Clearly, (4.35) yields $\bar{C}_{0\tau} = Q_{0\tau_1}F_{0\tau_1}$. Since $F_{0\tau_1}\psi^l_{\mu\nu}$ $(u(i,\mu+1),x(i,\nu+1))$ in $1eq_{\tau_1}(i)$ as a function of the variable x_i is an injection and the rank of $Q_{0\tau_1}$ is m', $\bar{C}_{0\tau}\psi^l_{\mu\nu}(u(i,\mu+1),x(i,\nu+1))$ in $\bar{eq}_\tau(i)$ as a function of the variable x_i is an injection. From Theorem 4.1.2, for any linear R_a R_b transformation sequence (4.34), $C_{0\tau}\psi^l_{\mu\nu}(u(i,\mu+1),x(i,\nu+1))$ in $eq_\tau(i)$ as a function of the variable x_i is an injection.

(b) The proof of part (b) is similar to part (a). What we need to do is to replace the phrases "weakly invertible", "is m'", "Theorem 3.3.2" and "injection" in the proof of part (a) by "a weak inverse", "is m", "Theorem 3.3.1" and "surjection", respectively. □

Below M_1 is restricted to an h_1-order input-memory finite automaton.

Lemma 4.2.3. *For any* (h_0,k)*-order memory finite automaton* $M_0 = \langle Y', Y,\ S_0,\delta_0,\lambda_0\rangle$ *and any* h_1*-order input-memory finite automaton* $M_1 = \langle X, Y',S_1,\ \delta_1,\lambda_1\rangle$ *with* $|X| = |Y'|$*, if* M_1 *is weakly invertible with delay* 0*, then for any state* s_0 *of* M_0 *there exist a state* s *of* $C'(M_1,M_0)$ *and a state* s_1 *of* M_1 *such that* s *and the state* $\langle s_1,s_0\rangle$ *of* $C(M_1,M_0)$ *are equivalent.*

Proof. Denote $s_0 = \langle y(-1,k), y'(-1,h_0)\rangle$. From Theorem 1.4.6, since M_1 is weakly invertible with delay 0, there exist x_{-1}, ..., $x_{-h_0-h_1} \in X$, such that

$$\lambda_1(\langle x(-h_0-1,h_1)\rangle, x_{-h_0}\dots x_{-1}) = y'_{-h_0}\dots y'_{-1}.$$

Let $s = \langle y(-1,k), x(-1,h_0+h_1)\rangle$ and $s_1 = \langle x(-1,h_1)\rangle$. From Theorem 1.2.1, s and $\langle s_1, s_0\rangle$ are equivalent. □

Lemma 4.2.4. *Let $M_0 = \langle Y', Y, S_0, \delta_0, \lambda_0\rangle$ be an (h_0,k)-order memory finite automaton, and $M_1 = \langle X, Y', S_1, \delta_1, \lambda_1\rangle$ an h_1-order input-memory finite automaton with $|X| = |Y'|$. Assume that M_1 is weakly invertible with delay 0.*

(a) *$C'(M_1, M_0)$ is weakly invertible with delay τ if and only if M_0 is weakly invertible with delay τ.*

(b) *$C'(M_1, M_0)$ is a weak inverse with delay τ if and only if M_0 is a weak inverse with delay τ.*

Proof. (a) Suppose that $C'(M_1, M_0)$ is weakly invertible with delay τ. For any state s_0 of M_0, from Lemma 4.2.3, there exist a state s of $C'(M_1, M_0)$ and a state s_1 of M_1 such that s and the state $\langle s_1, s_0\rangle$ of $C(M_1, M_0)$ are equivalent. Since M_1 is weakly invertible with delay 0, there exists a finite automaton $M'_1 = \langle Y', X, S'_1, \delta'_1, \lambda'_1\rangle$ such that M'_1 is a weak inverse with delay 0 of M_1. Since $|X| = |Y'|$, for any state s'' of M_1 and any state s' of M'_1, s'' 0-matches s' if and only if s' 0-matches s''. Thus there exists a state s'_1 of M'_1 such that s_1 0-matches s'_1. It follows that the state s_0 of M_0 and the state $\langle s'_1, \langle s_1, s_0\rangle\rangle$ of $C(M'_1, C(M_1, M_0))$ are equivalent. Clearly, the state $\langle s'_1, \langle s_1, s_0\rangle\rangle$ of $C(M'_1, C(M_1, M_0))$ is equivalent to the state $\langle s'_1, s\rangle$ of $C(M'_1, C'(M_1, M_0))$. It follows that the state s_0 of M_0 is equivalent to the state $\langle s'_1, s\rangle$ of $C(M'_1, C'(M_1, M_0))$. Suppose that M' is a weak inverse of $C'(M_1, M_0)$ with delay τ and the state s' of M' τ-matches s. Let $\bar{M}_1$ be the τ-stay of M_1 and $\bar{s}_1 = \langle s_1, 0\rangle$. It is easy to see that the state $\langle s', \bar{s}_1\rangle$ of $C(M', \bar{M}_1)$ τ-matches the state $\langle s'_1, s\rangle$ of $C(M'_1, C'(M_1, M_0))$. Therefore, the state $\langle s', \bar{s}_1\rangle$ of $C(M', \bar{M}_1)$ τ-matches the state s_0 of M_0. Thus $C(M', \bar{M}_1)$ is a weak inverse of M_0 with delay τ. We conclude that M_0 is weakly invertible with delay τ.

Conversely, suppose that M_0 is weakly invertible with delay τ. Let M'_0 be a weak inverse of M_0 with delay τ. Since M_1 is weakly invertible with delay 0, there exists M'_1 such that M'_1 is a weak inverse with delay 0 of M_1. Let $\bar{M}'_1$ be the τ-stay of M'_1. We prove that $C(M'_0, \bar{M}'_1)$ is a weak inverse with delay τ of $C'(M_1, M_0)$. Let s be a state of $C'(M_1, M_0)$. Clearly, there is a state $\langle s_1, s_0\rangle$ of $C(M_1, M_0)$ such that s and $\langle s_1, s_0\rangle$ are equivalent. Since M'_0 is a weak inverse of M_0 with delay τ, there is a state s'_0 of M'_0 such that s'_0 τ-matches s_0. Since M'_1 is a weak inverse of M_1 with delay 0, there is a state s'_1

of M_1' such that s_1' 0-matches s_1. Letting $\bar{s}_1' = \langle s_1', 0\rangle$, it follows that the state $\langle s_0', \bar{s}_1'\rangle$ of $C(M_0', \bar{M}_1')$ τ-matches the state $\langle s_1, s_0\rangle$ of $C(M_1, M_0)$. Therefore, the state $\langle s_0', \bar{s}_1'\rangle$ of $C(M_0', \bar{M}_1')$ τ-matches the state s of $C'(M_1, M_0)$. Thus $C(M_0', \bar{M}_1')$ is a weak inverse of $C'(M_1, M_0)$ with delay τ. We conclude that $C'(M_1, M_0)$ is weakly invertible with delay τ.

(b) Suppose that $C'(M_1, M_0)$ is a weak inverse with delay τ. Then there exists M' such that $C'(M_1, M_0)$ is a weak inverse with delay τ of M'. It follows that $C(M_1, M_0)$ is a weak inverse with delay τ of M'. For any state s' of M', choose a state $\langle \varphi(s'), s_0\rangle$ of $C(M_1, M_0)$ such that $\langle \varphi(s'), s_0\rangle$ τ-matches s'. Let M_0' be the finite subautomaton of $C(M', M_1)$ of which the state alphabet is the set $\{\delta''(\langle s', \varphi(s')\rangle, \beta) | s' \in S', \beta \in Y^*\}$ and the input alphabet and the output alphabet are Y and Y', respectively, where S' is the state alphabet of M', δ'' is the next state function of $C(M', M_1)$. Since for any state s' of M', there exists a state s_0 of M_0 such that the state $\langle \varphi(s'), s_0\rangle$ of $C(M_1, M_0)$ τ-matches s', it is easy to prove that s_0 τ-matches the state $\langle s', \varphi(s')\rangle$ of M_0'. From the construction of M_0', for any state of M_0' there exists a state of M_0 τ-matching it. Therefore, M_0 is a weak inverse with delay τ of M_0'. We conclude that M_0 is a weak inverse with delay τ.

Conversely, suppose that M_0 is a weak inverse with delay τ. Then there exists M_0' such that M_0 is a weak inverse with delay τ of M_0'. Since M_1 is weakly invertible with delay 0, there exists M_1' such that M_1' is a weak inverse with delay 0 of M_1. Since $|X| = |Y'|$, for any state s'' of M_1 and any state s' of M_1', s'' 0-matches s' if and only if s' 0-matches s''. For any state s_0' of M_0', let s_0 be a state of M_0 such that s_0 τ-matches s_0'. From Lemma 4.2.3, there exist a state s of $C'(M_1, M_0)$ and a state s_1 of M_1 such that s and the state $\langle s_1, s_0\rangle$ of $C(M_1, M_0)$ are equivalent. Let s_1' be a state of M_1' such that s_1 matches s_1' with delay 0. For each s_0' fix such an s_1', denoted by $\varphi'(s_0')$. Let M' be the finite subautomaton of $C(M_0', M_1')$ of which the state alphabet is the set $\{\delta''(s'', \beta) \mid s'' \in S'', \beta \in Y^*\}$ and the input alphabet and the output alphabet are Y and X, respectively, where $S'' = \{\langle s_0', \varphi'(s_0')\rangle \mid s_0' \in S_0'\}$, S_0' is the state alphabet of M_0', δ'' is the next state function of $C(M_0', M_1')$. From the above discussion, for each state $\langle s_0', \varphi'(s_0')\rangle$ in S'' there exists a state $\langle s_1, s_0\rangle$ of $C(M_1, M_0)$ such that $\langle s_1, s_0\rangle$ τ-matches $\langle s_0', \varphi'(s_0')\rangle$ and $\langle s_1, s_0\rangle$ is equivalent to some state s of $C'(M_1, M_0)$. It follows that s τ-matches $\langle s_0', \varphi'(s_0')\rangle$. From the construction of M', for any state s' of M' there is a state $\bar{s}$ of $C'(M_1, M_0)$ such that $\bar{s}$ τ-matches s'. Thus $C'(M_1, M_0)$ is a weak inverse with delay τ of M'. We conclude that $C'(M_1, M_0)$ is a weak inverse with delay τ. □

Since M_1 is an h_1-order input-memory finite automaton, that is, $p = -1$ in (4.31), $M_1 = \langle X, Y', X^{h_1}, \delta_1, \lambda_1\rangle$ is defined by

$$y'_i = f(x(i, h_1 + 1)), \quad i = 0, 1, \ldots, \tag{4.36}$$

and f is expressed in the form

$$f(x(i, h_1 + 1)) = [F_0, \ldots, F_{h_1}]\psi_\nu^{lh_1}(x, i).$$

From the definition of compound finite automata, $C'(M_1, M_0) = \langle X, Y, Y^k \times X^h, \delta, \lambda\rangle$ is a finite automaton defined by

$$y_i = \varphi_{out}(y(i-1, k)) + [C_0, \ldots, C_h]\psi_\nu^{lh}(x, i), \quad i = 0, 1, \ldots,$$

where C_j, $j = 0, 1, \ldots, h$ are defined by (4.32), and $h = h_0 + h_1$.

Theorem 4.2.3. *Let M_0 be an (h_0, k)-order memory finite automaton defined by* (4.30), *and M_1 an h_1-order memory finite automaton defined by* (4.36) *with $|X| = |Y'|$. Let $eq_0(i)$ be equivalent to the equation*

$$-y_i + \varphi_{out}(y(i-1, k)) + [C_0, \ldots, C_h]\psi_\nu^{lh}(x, i) = 0,$$

and

$$eq_c(i) \overset{R_a[P_c]}{\longrightarrow} eq'_c(i), \; eq'_c(i) \overset{R_b[r_{c+1}]}{\longrightarrow} eq_{c+1}(i), \quad c = 0, 1, \ldots, \tau - 1$$

a linear R_a R_b transformation sequence, where C_j, $j = 0, 1, \ldots, h$ are defined by (4.32). *Assume that M_1 is weakly invertible with delay* 0.

(a) *$C'(M_1, M_0)$ is a weak inverse with delay τ if and only if for any parameters x_{i-1}, …, x_{i-h}, $y_{i+\tau}$, …, y_{i-k}, $eq_\tau(i)$ has a solution x_i.*

(b) *$C'(M_1, M_0)$ is weakly invertible with delay τ if and only if for any parameters x_{i-1}, …, x_{i-h}, $y_{i+\tau}$, …, y_{i-k}, $eq_\tau(i)$ has at most one solution x_i.*

Proof. (a) The *if* part is a special case of Corollary 3.1.1. To prove the *only if* part we suppose that $C'(M_1, M_0)$ is a weak inverse with delay τ. It follows that $q^m = |Y| \leqslant |X|$. From Lemma 4.2.4, M_0 is a weak inverse with delay τ. Let

$$0eq_c(i) \overset{R_a[\bar{P}_c]}{\longrightarrow} 0eq'_c(i), \; 0eq'_c(i) \overset{R_b[\bar{r}_{c+1}]}{\longrightarrow} 0eq'_{c+1}(i), \quad c = 0, 1, \ldots, \tau - 1 \tag{4.37}$$

be a linear R_a R_b transformation sequence, where $0eq_(i)$ is equivalent to the equation

$$-y_i + \varphi_{out}(y(i-1, k)) + [B_0, \ldots, B_{h_0}] \begin{bmatrix} x_i \\ \vdots \\ x_{i-h_0} \end{bmatrix} = 0,$$

where $x_i, \ldots, x_{i-h_0}$ take values in Y'. Applying Lemma 4.2.1 τ times, c from 1 to τ, we obtain an R_a R_b transformation sequence

$$\bar{eq}_c(i) \overset{R_a[\bar{P}_c]}{\longrightarrow} \bar{eq}'_c(i),\ \bar{eq}'_c(i) \overset{R_b[\bar{r}_{c+1}]}{\longrightarrow} \bar{eq}_{c+1}(i),\quad c = 0, 1, \ldots, \tau - 1 \tag{4.38}$$

satisfying

$$[\bar{C}_{0c}, \ldots, \bar{C}_{hc}] = [B_{0c}, \ldots, B_{h_0c}]\Gamma_{h+1},\ c = 0, 1, \ldots, \tau,$$

where $\bar{eq}_0(i)$ is $eq_0(i)$, $\bar{eq}_c(i)$ is in the form

$$\varphi_c(y(i + c, c + k + 1)) + [\bar{C}_{0c}, \ldots, \bar{C}_{hc}]\psi^{lh}_\nu(x, i) = 0,$$

$\bar{eq}'_c(i)$ is in the form

$$\varphi'_c(y(i + c, c + k + 1)) + [\bar{C}'_{0c}, \ldots, \bar{C}'_{hc}]\psi^{lh}_\nu(x, i) = 0,$$

φ_c and φ'_c are two single-valued mappings from $Y^{c+k+1}$to Y, $\bar{C}_{jc}$ and $\bar{C}'_{jc}$ are $m \times l$ matrices over $GF(q)$. It follows that

$$\bar{C}_{0c} = B_{0c}F_0,\ c = 0, 1, \ldots, \tau.$$

Thus

$$\bar{C}'_{0c} = B'_{0c}F_0,\ c = 0, 1, \ldots, \tau - 1.$$

Since M_1 is weakly invertible with delay 0, the rank of F_0 is m'. Noticing that (4.37) is linear over $GF(q)$, from the definition (on p.111), using these facts, it is easy to see that (4.38) is linear over $GF(q)$. Since $eq_0(i)$ and $\bar{eq}_0(i)$ are the same, using Theorem 4.1.1, there exists an $m \times m$ nonsingular matrix $Q_{0\tau}$ such that $\bar{C}_{0\tau} = Q_{0\tau}C_{0\tau}$. Thus we have $C_{0\tau} = Q^{-1}_{0\tau}B_{0\tau}F_0$. Notice that (4.37) is also linear in the sense of Sect. 3.3 (see p.95). Using Theorem 3.3.1, since M_0 is a weak inverse with delay τ, the rank of $B_{0\tau}$ is m. It follows that the rank of $Q^{-1}_{0\tau}B_{0\tau}$ is m. Since M_1 is weakly invertible with delay 0, for any parameters x_{i-1}, ..., $x_{i-\nu}$, $F_0\psi^l_\nu(x(i, \nu + 1))$ as a function of the variable x_i is injective. From $|X| = |Y'|$, this function is bijective. Since the rank of $Q^{-1}_{0\tau}B_{0\tau}$ is m and $q^m \leqslant |X| = |Y'|$, for any parameters x_{i-1}, ..., $x_{i-\nu}$, $Q^{-1}_{0\tau}B_{0\tau}F_0\psi^l_\nu(x(i, \nu+1))$, i.e., $C_{0\tau}\psi^l_\nu(x(i, \nu+1))$, as a function of the variable x_i is surjective. It follows that for any parameters x_{i-1}, ..., x_{i-h}, $y_{i+\tau}$, ..., y_{i-k}, the equation $eq_\tau(i)$ has a solution x_i.

(b) The *if* part is a special case of Theorem 3.1.3. To prove the *only if* part we suppose that $C'(M_1, M_0)$ is weakly invertible with delay τ. From Lemma 4.2.4, M_0 is weakly invertible with delay τ. Let (4.37) be a linear R_a R_b transformation sequence. Applying Lemma 4.2.1 τ times, c from 1 to τ, we obtain an R_a R_b transformation sequence (4.38) satisfying

$$[\bar{C}_{0c}, \ldots, \bar{C}_{hc}] = [B_{0c}, \ldots, B_{h_0c}]\Gamma_{h+1},\ c = 0, 1, \ldots, \tau.$$

It follows that

$$\bar{C}_{0c} = B_{0c}F_0,\ c = 0, 1, \ldots, \tau.$$

Similar to the proof of (a), (4.38) is linear and there exists an $m \times m$ nonsingular matrix $Q_{0\tau}$ such that $\bar{C}_{0\tau} = Q_{0\tau}C_{0\tau}$. Thus we have $C_{0\tau} = Q_{0\tau}^{-1}B_{0\tau}F_0$. Using Theorem 3.3.2, since M_0 is weakly invertible with delay τ, the rank of $B_{0\tau}$ is m'. It follows that the rank of $Q_{0\tau}^{-1}B_{0\tau}$ is m'. Since M_1 is weakly invertible with delay 0, for any parameters x_{i-1}, ..., $x_{i-\nu}$, $F_0\psi_\nu^l(x(i, \nu+1))$ as a function of the variable x_i is injective. From $|X| = |Y'|$, this function is bijective. Since the rank of the $m \times m'$ matrix $Q_{0\tau}^{-1}B_{0\tau}$ is m', for any parameters x_{i-1}, ..., $x_{i-\nu}$, $Q_{0\tau}^{-1}B_{0\tau}F_0\psi_\nu^l(x(i, \nu+1))$, i.e., $C_{0\tau}\psi_\nu^l(x(i, \nu+1))$, as a function of the variable x_i is injective. It follows that for any parameters x_{i-1}, ..., x_{i-h}, $y_{i+\tau}$, ..., y_{i-k}, the equation $eq_\tau(i)$ has at most one solution x_i. □

From the proof of the above theorem, we have the following.

Corollary 4.2.1. *Let M_0 be an (h_0, k)-order memory finite automaton defined by* (4.30), *and M_1 an h_1-order memory finite automaton defined by* (4.36) *with $|X| = |Y'|$. Let $eq_0(i)$ be equivalent to the equation*

$$-y_i + \varphi_{out}(y(i-1, k)) + [C_0, \ldots, C_h]\psi_\nu^{lh}(x, i) = 0,$$

and

$$eq_c(i) \overset{R_a[P_c]}{\longrightarrow} eq'_c(i),\ eq'_c(i) \overset{R_b[r_{c+1}]}{\longrightarrow} eq_{c+1}(i),\quad c = 0, 1, \ldots, \tau-1$$

a linear R_a R_b transformation sequence, where C_j, $j = 0, 1, \ldots, h$ are defined by (4.32). *Assume that M_1 is weakly invertible with delay* 0.

(a) *$C'(M_1, M_0)$ is a weak inverse with delay τ if and only if $C_{0\tau}\psi_\nu^l(x(i, \nu+1))$ as a function of the variable x_i is surjective.*

(b) *$C'(M_1, M_0)$ is weakly invertible with delay τ if and only if $C_{0\tau}\psi_\nu^l(x(i, \nu+1))$ as a function of the variable x_i is injective.*

4.3 Reduced Echelon Matrix

For any nonnegative integer c, let $eq_c(i)$ be an equation

$$\varphi_c(y(i+c, c+k+1)) + [C_{0c}, \ldots, C_{hc}]\psi_{\mu\nu}^{lh}(u, x, i) = 0$$

and let $eq'_c(i)$ be an equation

$$\varphi'_c(y(i+c, c+k+1)) + [C'_{0c}, \ldots, C'_{hc}]\psi_{\mu\nu}^{lh}(u, x, i) = 0,$$

where φ_c and φ'_c are two single-valued mappings from Y^{c+k+1} to Y, C_{jc} and C'_{jc} are $m \times l$ matrices over a finite field $GF(q)$, $j = 0, 1, \ldots, h$.

Theorem 4.3.1. *Let $eq_0(i)$ be equivalent to the equation*

$$-y_i + \varphi_{out}(y(i-1,k)) + [C_0,\ldots,C_h]\psi^{lh}_{\mu\nu}(u,x,i) = 0,$$

and

$$\Gamma = \begin{bmatrix} C_0 & C_1 & \ldots & \ldots & C_h & & \\ & C_0 & C_1 & \ldots & \ldots & C_h & \\ & & \ddots & \ddots & \ddots & \ddots & \ddots \\ & & & C_0 & C_1 & \ldots & \ldots & C_h \end{bmatrix}$$

be an $m(\tau+1)\times l(\tau+h+1)$ matrix, where φ_{out} is a single-valued mapping from Y^k to Y. Let

$$eq_c(i) \overset{R_a[P_c]}{\longrightarrow} eq'_c(i),\ eq'_c(i) \overset{R_b[r_{c+1}]}{\longrightarrow} eq_{c+1}(i),\quad c=0,1,\ldots,\tau \tag{4.39}$$

be a linear R_a R_b transformation sequence. Assume that the reduced echelon matrix of Γ is expressed in the form

$$\begin{bmatrix} D_{11} & D_{12} & D_{13} \\ 0 & D_{22} & D_{23} \\ 0 & 0 & D_{33} \end{bmatrix},$$

where D_{11} and D_{22} are row independent and have $l\tau$ and l columns, respectively. Then D_{22} and the submatrix of the first $r_{\tau+1}$ rows of $C'_{0\tau}$ in $eq'_\tau(i)$ are row equivalent.[1]

Proof. Using Properties (g) and (a) of R_a R_b transformations in Sect. 3.1, from (4.39), the system of equations $eq_0(i)$, $i=0,1,\ldots,\tau$ is equivalent to the system of equations $E'_0eq'_0(\tau)$, $E'_1eq'_1(\tau-1)$, …, $E'_{\tau-1}eq'_{\tau-1}(1)$, $E'_\tau eq'_\tau(0)$, $E''_0eq'_0(0)$, $E''_1eq'_1(0)$, …, $E''_\tau eq'_\tau(0)$, where E'_j and E''_j are the submatrices of the first r_{j+1} rows and the last $m-r_{j+1}$ rows of the $m\times m$ identity matrix, respectively, $j=0,1,\ldots,\tau$. Since, for any $\psi^l_{\mu\nu}$, Γ is the coefficient matrix with respect to $\psi^{l,\tau+h}_{\mu\nu}(u,x,\tau)$ in the system of equations $eq_0(i)$, $i=\tau,\tau-1,\ldots,0$, it follows that Γ is row equivalent to a matrix Γ':

$$\begin{bmatrix} E'_0C'_{00} & E'_0C'_{10} & & & & \ldots & E'_0C'_{h0} & & & \\ & E'_1C'_{01} & E'_1C'_{11} & & & \ldots & & E'_1C'_{h1} & & \\ & & \ddots & \ddots & & \ddots & & \ddots & & \\ & & & E'_\tau C'_{0\tau} & E'_\tau C'_{1\tau} & \ldots & & & E'_\tau C'_{h\tau} \\ & & & & E''_0C'_{10} & \ldots & & & E''_0C'_{h0} \\ & & & & E''_1C'_{11} & \ldots & & & E''_1C'_{h1} \\ & & & & \ldots & \ldots & & & \ldots \\ & & & & E''_\tau C'_{1\tau} & \ldots & & & E''_\tau C'_{h\tau} \end{bmatrix}.$$

[1] Two matrices A and B are row equivalent, if there exists a nonsingular matrix T such that $B=TA$.

Since (4.39) is linear over $GF(q)$, $E'_jC'_{0j}$ is row independent for any j, $0 \leqslant j \leqslant \tau$. Thus there is a nonsingular matrix

$$P = \begin{bmatrix} P_{00} & P_{01} & \dots & P_{0,\tau-1} & P_{0\tau} & P_{0,\tau+1} \\ 0 & P_{11} & \dots & P_{1,\tau-1} & P_{1\tau} & P_{1,\tau+1} \\ \dots & \dots & \dots\dots & & \dots & \dots \\ 0 & 0 & \dots & P_{\tau-1,\tau-1} & P_{\tau-1,\tau} & P_{\tau-1,\tau+1} \\ 0 & 0 & \dots & 0 & P_{\tau\tau} & P_{\tau,\tau+1} \\ 0 & 0 & \dots & 0 & 0 & P_{\tau+1,\tau+1} \end{bmatrix}$$

such that $P\Gamma'$ is the reduced echelon matrix of Γ, where P_{cc} is an $r_{c+1} \times r_{c+1}$ matrix and $P_{cc}E'_cC'_{0c}$ is the reduced echelon matrix of $E'_cC'_{0c}$, for $c = 0, 1, \dots, \tau$. Let

$$P\Gamma' = \begin{bmatrix} D_{11} & D_{12} & D_{13} \\ D_{21} & D_{22} & D_{23} \\ D_{31} & D_{32} & D_{33} \end{bmatrix},$$

where D_{11} and D_{22} are $(r_1 + \dots + r_\tau) \times l\tau$ and $r_{\tau+1} \times l$ matrices, respectively. It is easy to see that D_{11} is row independent, $D_{21} = 0$, $D_{22} = P_{\tau\tau}E'_\tau C'_{0\tau}$, $D_{31} = 0$ and $D_{32} = 0$. Noticing that $P_{\tau\tau}$ is nonsingular, $E'_\tau C'_{0\tau}$, the submatrix of the first $r_{\tau+1}$ rows of $C'_{0\tau}$, is row equivalent to D_{22}. Since the reduced echelon matrix is unique, the theorem holds. □

Corollary 4.3.1. *Under the hypothesis of Theorem* 4.3.1, *for any parameters* $x_{i-1}, \dots, x_{i-\nu}, u_i, \dots, u_{i-\mu}$, $C_{0\tau}\psi^l_{\mu\nu}(u(i,\mu+1), x(i,\nu+1))$ *as a function of the variable* x_i *is an injection if and only if* $D_{22}\psi^l_{\mu\nu}(u(i,\mu+1), x(i,\nu+1))$ *as a function of the variable* x_i *is an injection, and* $C_{0\tau}\psi^l_{\mu\nu}(u(i,\mu+1), x(i,\nu+1))$ *as a function of the variable* x_i *is a surjection if and only if* $D_{22}\psi^l_{\mu\nu}(u(i, \mu+1), x(i,\nu+1))$ *as a function of the variable* x_i *is a surjection and* $r_{\tau+1} = m$.

Proof. Since $eq_\tau(i) \xrightarrow{R_a[P_\tau]} eq'_\tau(i)$ and $eq'_\tau(i) \xrightarrow{R_b[r_{\tau+1}]} eq_{\tau+1}(i)$ are linear, $C'_{0\tau} = P_\tau C_{0\tau} = \begin{bmatrix} E'_\tau C'_{0\tau} \\ 0 \end{bmatrix}$ has rank $r_{\tau+1}$. Therefore, $C_{0\tau}\psi^l_{\mu\nu}(u(i,\mu+1), x(i, \nu+1))$ as a function of the variable x_i is an injection if and only if $C'_{0\tau}\psi^l_{\mu\nu}$ $(u(i,\mu+1), x(i,\nu+1))$ as a function of the variable x_i is an injection, if and only if $E'_\tau C'_{0\tau}\psi^l_{\mu\nu}(u(i,\mu+1),\ x(i,\nu+1))$ as a function of the variable x_i is an injection. From Theorem 4.3.1, D_{22} and $E'_\tau C'_{0\tau}$ are row equivalent. It follows that $C_{0\tau}\psi^l_{\mu\nu}(u(i,\mu+1), x(i,\nu+1))$ as a function of the variable x_i is an injection if and only if $D_{22}\ \psi^l_{\mu\nu}\ (u(i,\mu+1),\ x(i,\nu+1))$ as a function of the variable x_i is an injection. Similarly, $C_{0\tau}\psi^l_{\mu\nu}(u(i,\mu+1),\ x(i,\nu+1))$ as a function of the variable x_i is a surjection if and only if $C'_{0\tau}\psi^l_{\mu\nu}\ (u(i,\mu+1),$ $x(i,\nu+1))$ as a function of the variable x_i is a surjection, if and only if $E'_\tau C'_{0\tau}\psi^l_{\mu\nu}(u(i,\mu+1), x(i,\nu+1))$ as a function of the variable x_i is a surjection and $r_{\tau+1} = m$. Since $E'_\tau C'_{0\tau}\psi^l_{\mu\nu}(u(i,\mu+1), x(i,\nu+1))$ as a function of the variable x_i is a surjection if and only if $D_{22}\psi^l_{\mu\nu}(u(i,\mu+1), x(i,\nu+1))$ as a

function of the variable x_i is a surjection, we obtain that $C_{0\tau}\psi^l_{\mu\nu}(u(i,\mu+1), x(i,\nu+1))$ as a function of the variable x_i is a surjection if and only if $D_{22}\psi^l_{\mu\nu}(u(i,\mu+1), x(i,\nu+1))$ as a function of the variable x_i is a surjection and $r_{\tau+1} = m$. □

From this observation, if some inversion method by reduced echelon matrix based on injectiveness or surjectiveness of $D_{22}\psi^l_{\mu\nu}(u(i,\mu+1), x(i,\nu+1))$ is applicable to a finite automaton M, so is the R_a R_b transformation method described in Sect. 3.1. But the method of reduced echelon matrix for finding weak inverse of M is a bit more complex than the R_a R_b transformation method, $\tau+1$ equations v. one equation.

As to finding weak inverse of a finite automaton by reduced echelon matrix, assume that $M = \langle X, Y, Y^k \times U^{h+\mu+1} \times X^{h+\nu}, \delta, \lambda\rangle$ is defined by

$$\begin{aligned} y_i &= \varphi_{out}(y(i-1,k)) + [C_0, \ldots, C_h]\psi^{lh}_{\mu\nu}(u,x,i),\\ u_{i+1} &= g(u(i,h+\mu+1), x(i,h+\nu+1)),\\ &\quad i = 0,1,\ldots \end{aligned}$$

We can multiply Γ on the left by a nonsingular matrix $\bar{P}$ to obtain its reduced echelon matrix. Because the reduced echelon matrix of a matrix is unique, we obtain D_{22} and D_{23} in Theorem 4.3.1. Let $\bar{P} = \begin{bmatrix} \bar{P}_1 \\ \bar{P}_2 \\ \bar{P}_3 \end{bmatrix}$, where the numbers of rows of $\bar{P}_1$ and D_{11} are the same, and the numbers of rows of $\bar{P}_2$ and D_{22} are the same. Denote $\bar{P}_2 \begin{bmatrix} -y_{i+\tau}+\varphi_{out}(y(i+\tau-1,k)) \\ \cdots \\ -y_i+\varphi_{out}(y(i-1,k)) \end{bmatrix}$ by $\bar{\varphi}(y(i+\tau, \tau+k+1))$. If for any parameters $x_{i-1}, \ldots, x_{i-\nu}, u_i, \ldots, u_{i-\mu}$, $D_{22}\psi^l_{\mu\nu}(u(i,\mu+1), x(i,\nu+1))$ as a function of the variable x_i is an injection, then for any parameters $x_{i-1}, \ldots, x_{i-h-\nu}, u_i, \ldots, u_{i-h-\mu}, y_{i+\tau}, \ldots, y_{i-k}$, the equation

$$\bar{\varphi}(y(i+\tau, \tau+k+1)) + [D_{22}, D_{23}]\psi^{lh}_{\mu\nu}(u,x,i) = 0$$

has at most one solution x_i. Let f^* be a single-valued mapping from $X^{h+\nu} \times U^{h+\mu+1} \times Y^{\tau+k+1}$ to X so that if such a solution x_i exists, then

$$x_i = f^*(x(i-1,h+\nu), u(i,h+\mu+1), y(i+\tau,\tau+k+1)).$$

Construct a finite automaton $M^* = \langle Y, X, X^{h+\nu} \times U^{h+\mu+1} \times Y^{\tau+k}, \delta^*, \lambda^*\rangle$, where

$$\begin{aligned} &\delta^*(\langle x(i-1,h+\nu), u(i,h+\mu+1), y'(i-1,\tau+k)\rangle, y'_i)\\ &\quad = \langle x(i,h+\nu), u(i+1,h+\mu+1), y'(i,\tau+k)\rangle,\\ &\lambda^*(\langle x(i-1,h+\nu), u(i,h+\mu+1), y'(i-1,\tau+k)\rangle, y'_i) = x_i, \end{aligned}$$

$$x_i = f^*(x(i-1, h+\nu), u(i, h+\mu+1), y'(i, \tau+k+1)),$$
$$u_{i+1} = g(u(i, h+\mu+1), x(i, h+\nu+1)).$$

Similar to the discussions in Sect. 3.1, the τ-stay of M^* is a weak inverse with delay τ of M.

4.4 Canonical Diagonal Matrix Polynomial

4.4.1 R_a R_b Transformations over Matrix Polynomial

Let F be a field, z an indeterminate, and $F[z]$ the polynomial ring over F. A matrix, of which elements are polynomials of z, is called a *matrix polynomial*. Let $M_{m,n}(F[z])$ be the set of all $m \times n$ matrix polynomials, and $GL_k(F[z])$ the set of all $k \times k$ invertible matrix polynomials. [1]

We use $DIA_{m,n}(g_1(z), \ldots, g_{\min(m,n)}(z))$ to denote the $m \times n$ matrix of which main diagonal elements are $g_1(z)$, ..., $g_{\min(m,n)}(z)$ in turn and zero elsewhere. If $m \times n$ matrices are partitioned into $r \times s$ blocks, we also use $DIA_{m,n}(A_1(z), \ldots, A_{\min(r,s)}(z))$ to denote the matrix of which main diagonal blocks are $A_1(z), \ldots, A_{\min(r,s)}(z)$ in turn and zero elsewhere. Denote the $n \times n$ identity matrix by E_n. For any $m \geqslant n \geqslant 0$, denote $DIA_{m,m}(E_n, zE_{m-n})$ by $I_{m,n}$. For any matrix A and any matrix polynomial $A(z)$, we use $A(i_1, \ldots, i_r; j_1, \ldots, j_r)$ and $A(i_1, \ldots, i_r; j_1, \ldots, j_r; z)$ to denote their r-order minors of rows $i_1, \ldots, i_r$ and columns $j_1, \ldots, j_r$, and call them $(i_1, \ldots, i_r; j_1, \ldots, j_r)$ minors of A and $A(z)$, respectively.

A matrix polynomial can be transformed into the canonical diagonal form by elementary transformations. That is, for any $C(z) \in M_{m,n}(F[z])$ with rank r, there exist $P(z) \in GL_m(F[z])$, $Q(z) \in GL_n(F[z])$, r nonnegative integers $a_1, \ldots, a_r$ and r polynomials $f_1(z), \ldots, f_r(z)$ such that

$$C(z) = P(z) DIA_{m,n}(z^{a_1} f_1(z), \ldots, z^{a_r} f_r(z), 0, \ldots, 0) Q(z),$$

$0 \leqslant a_1 \leqslant \cdots \leqslant a_r$, $f_j(z) \mid f_{j+1}(z)$ for $j = 1, \ldots, r-1$, and $f_j(0) \neq 0$ for $j = 1, \ldots, r$.

Let $C(z)$ in $M_{m,n}(F[z])$ be $\sum_{j=0}^{h} C_{j0} z^j$. We can expand the definitions of R_a R_b transformations on matrices over $GF(q)$ to the matrix $[C_{00}, \ldots, C_{h0}]$ over the field F. In parallel, we define R_a R_b transformations for matrix polynomial as follows.

Rule R_a : Let $k \geqslant 0$, $C_k(z) \in M_{m,n}(F[z])$ and $C_k(z) = \sum_{j=0}^{h} C_{jk} z^j$. Let P_k be a nonsingular matrix over F, and

[1] A matrix polynomial is said to be *invertible*, if its determinant is a nonzero constant.

$$C'_{jk} = P_k C_{jk},\ j = 0, 1, \ldots, h.$$

$C'_k(z) = \sum_{j=0}^{h} C'_{jk} z^j$ is said to be *obtained from* $C_k(z)$ *by Rule* R_a *using* P_k, denoted by

$$C_k(z) \overset{R_a[P_k]}{\longrightarrow} C'_k(z).$$

Rule R_b : Let $k \geqslant 0$, $C'_k(z) \in M_{m,n}(F[z])$ and $C'_k(z) = \sum_{j=0}^{h} C'_{jk} z^j$. If the last $m - r_{k+1}$ rows of C'_{0k} are 0 in the case of $r_{k+1} < m$, $C_{k+1}(z) = I^{-1}_{m,r_{k+1}} C'_k(z) \in M_{m,n}(F[z])$ is said to be *obtained from* $C'_k(z)$ *by Rule* R_b, denoted by

$$C'_k(z) \overset{R_b[r_{k+1}]}{\longrightarrow} C_{k+1}(z).$$

An R_a R_b transformation sequence

$$C_k(z) \overset{R_a[P_k]}{\longrightarrow} C'_k(z),\ C'_k(z) \overset{R_b[r_{k+1}]}{\longrightarrow} C_{k+1}(z),\quad k = 0, 1, \ldots, t-1 \tag{4.40}$$

is said to be *elementary*, if for any k, $0 \leqslant k \leqslant t-1$, P_k is in the form

$$P_k = \begin{bmatrix} E_{r_k} & 0 \\ P_{k1} & P_{k2} \end{bmatrix},$$

and the first r_{k+1} rows of C'_{0k} is linearly independent over F, where $r_0 = 0$.

Notice that if (4.40) is an elementary R_a R_b transformation sequence, then $r_j \geqslant r_{j-1}$ for $j = 2, \ldots, t$.

The R_a R_b transformation sequence (4.40) is said to be *terminating*, if the last $m - r_t$ rows of $C_t(z)$ are 0 in the case of $r_t < m$ and the first r_t rows of C_{0t} are linearly independent over F.

From the definitions of R_a and R_b transformations, we have the following.

Lemma 4.4.1. *If* (4.40) *is an* R_a R_b *transformation sequence, then*

$$C_t(z) = I^{-1}_{m,r_t} P_{t-1} I^{-1}_{m,r_{t-1}} P_{t-2} \ldots I^{-1}_{m,r_1} P_0 C_0(z).$$

Lemma 4.4.2. *Let* (4.40) *be an elementary* R_a R_b *transformation sequence. Let* $m_1 = r_1$, $m_j = r_j - r_{j-1}$, $j = 2, \ldots, t$. *Then there exists an* $m \times m$ *invertible matrix polynomial* $\bar{P}(z)$ *such that degrees of elements in columns* $r'_k + 1$ *to* r'_{k+1} *of* $\bar{P}(z)$ *are at most* $t - k - 1$ *for* $k = 0, 1, \ldots, t-1$, *and*

$$\begin{aligned} &P_0^{-1} I_{m,r_1} P_1^{-1} I_{m,r_2} \ldots P_{t-2}^{-1} I_{m,r_{t-1}} P_{t-1}^{-1} I_{m,r_t} \\ &\quad = \bar{P}(z) D I A_{m,m}(E_{m_1}, zE_{m_2}, \ldots, z^{t-1} E_{m_t}, z^t E_{m-r_t}), \end{aligned}$$

where $r'_0 = 0$, $r'_i = r_i$, $i = 1, \ldots, t-1$, *and* $r'_t = m$.

Proof. It is easy to verify that for any k, $1 \leqslant k \leqslant t$,

$$I_{m,r_1} I_{m,r_2} \dots I_{m,r_k} = DIA_{m,m}(E_{m_1}, zE_{m_2}, \dots, z^{k-1}E_{m_k}, z^k E_{m-r_k}).$$

Denote

$$P_j = \begin{bmatrix} E_{m_1} & & & \\ & \ddots & & \\ & & E_{m_j} & \\ P_{j1} & \cdots & P_{jj} & P_{j,j+1} \end{bmatrix}, \ j = 1, \dots, t-1.$$

Clearly, P_j^{-1} is in the form

$$P_j^{-1} = \begin{bmatrix} E_{m_1} & & & \\ & \ddots & & \\ & & E_{m_j} & \\ P'_{j1} & \cdots & P'_{jj} & P'_{j,j+1} \end{bmatrix}, \ j = 1, \dots, t-1.$$

Let

$$P'_j(z) = \begin{bmatrix} E_{m_1} & & & \\ & \ddots & & \\ & & E_{m_j} & \\ z^j P'_{j1} & \cdots & zP'_{jj} & P'_{j,j+1} \end{bmatrix}, \ j = 1, \dots, t-1. \tag{4.41}$$

It is easy to see that

$$I_{m,r_1} I_{m,r_2} \dots I_{m,r_k} P_k^{-1} = P'_k(z) I_{m,r_1} I_{m,r_2} \dots I_{m,r_k}, \ k = 1, \dots, t-1.$$

Using this observation, it is easy to prove, by induction on k, the following proposition

$$I_{m,r_1} P_1^{-1} I_{m,r_2} P_2^{-1} \dots I_{m,r_k} P_k^{-1} = P'_1(z) P'_2(z) \dots P'_k(z) I_{m,r_1} I_{m,r_2} \dots I_{m,r_k}$$

holds for $k = 1, 2, \dots, t-1$. This yields

$$\begin{aligned} &P_0^{-1} I_{m,r_1} P_1^{-1} I_{m,r_2} P_2^{-1} \dots I_{m,r_{t-1}} P_{t-1}^{-1} I_{m,r_t} \\ &\quad = P_0^{-1} P'_1(z) P'_2(z) \dots P'_{t-1}(z) I_{m,r_1} I_{m,r_2} \dots I_{m,r_{t-1}} I_{m,r_t} \\ &\quad = \bar{P}(z) DIA_{m,m}(E_{m_1}, zE_{m_2}, \dots, z^{t-1}E_{m_t}, z^t E_{m-r_t}), \end{aligned}$$

where $\bar{P}(z) = P_0^{-1} P'_1(z) P'_2(z) \dots P'_{t-1}(z)$.

From (4.41), the determinant of $P'_j(z)$ is a nonzero constant for any j, $1 \leqslant j \leqslant t-1$. It follows immediately that the determinant of $P_0^{-1} P'_1(z) P'_2(z) \dots P'_{t-1}(z)$ is a nonzero constant. Therefore, $\bar{P}(z)$ is invertible.

Partition $P'_1(z) \ \dots \ P'_j(z)$ into $[P''_{j1}(z), \ \dots, \ P''_{j,j+1}(z)]$, where $P''_{jk}(z)$ has m_k columns for $k = 1, \dots, j$. From (4.41), it is easy to prove by induction on j that degrees of elements of $P''_{jk}(z)$ are at most $j - k + 1$ for any k, j,

$1 \leqslant k \leqslant j+1$ and $1 \leqslant j \leqslant t-1$. It follows immediately that degrees of elements in columns $r'_k + 1$ to r'_{k+1} of $\bar{P}(z)$ are at most $t-k-1$ for $k = 0, 1, \ldots, t-1$. □

Theorem 4.4.1. *Let* (4.40) *be an elementary* R_a R_b *transformation sequence. Let* $m_1 = r_1$, $m_j = r_j - r_{j-1}$, $j = 2, \ldots, t$. *Then there exists an* $m \times m$ *invertible matrix polynomial* $\bar{P}(z)$ *such that degrees of elements in columns* $r'_k + 1$ *to* r'_{k+1} *of* $\bar{P}(z)$ *are at most* $t-k-1$ *for* $k = 0, 1, \ldots, t-1$, *and*

$$C_0(z) = \bar{P}(z) DIA_{m,m}(E_{m_1}, zE_{m_2}, \ldots, z^{t-1}E_{m_t}, z^t E_{m-r_t}) C_t(z),$$

where $r'_0 = 0$, $r'_i = r_i$, $i = 1, \ldots, t-1$, *and* $r'_t = m$.

Proof. From Lemma 4.4.1,

$$C_t(z) = I^{-1}_{m,r_t} P_{t-1} I^{-1}_{m,r_{t-1}} P_{t-2} \ldots I^{-1}_{m,r_1} P_0 C_0(z).$$

Thus

$$C_0(z) = P_0^{-1} I_{m,r_1} P_1^{-1} I_{m,r_2} \ldots P_{t-2}^{-1} I_{m,r_{t-1}} P_{t-1}^{-1} I_{m,r_t} C_t(z).$$

Using Lemma 4.4.2, it follows immediately that there exists an $m \times m$ invertible matrix polynomial $\bar{P}(z)$ such that degrees of elements in columns $r'_k + 1$ to r'_{k+1} of $\bar{P}(z)$ are at most $t-k-1$ for $k = 0, 1, \ldots, t-1$, and

$$C_0(z) = \bar{P}(z) DIA_{m,m}(E_{m_1}, zE_{m_2}, \ldots, z^{t-1}E_{m_t}, z^t E_{m-r_t}) C_t(z). \qquad \square$$

Corollary 4.4.1. *Let* (4.40) *be a terminating and elementary* R_a R_b *transformation sequence. Let* $m_1 = r_1$, $m_j = r_j - r_{j-1}$, $j = 2, \ldots, t$. *Then there exists an* $m \times m$ *invertible matrix polynomial* $\bar{P}(z)$ *such that degrees of elements in columns* $r'_k + 1$ *to* r'_{k+1} *of* $\bar{P}(z)$ *are at most* $t-k-1$ *for* $k = 0, 1, \ldots, t-1$, *and*

$$C_0(z) = \bar{P}(z) DIA_{m,r_t}(E_{m_1}, zE_{m_2}, \ldots, z^{t-1}E_{m_t}) \bar{Q}(z), \tag{4.42}$$

where $r'_0 = 0$, $r'_i = r_i$, $i = 1, \ldots, t-1$, $r'_t = m$, *and* $\bar{Q}(z)$ *is the first* r_t *rows of* $C_t(z)$.

Corollary 4.4.2. *Let* (4.40) *be a terminating and elementary* R_a R_b *transformation sequence. Then the rank of* $C_0(z)$ *is* r_t.

Proof. Let r be the rank of $C_0(z)$. Since $\bar{P}(z)$ is invertible, from (4.42), the rank of $C'(z) = DIA_{m,r_t}(E_{m_1}, zE_{m_2}, \ldots, z^{t-1}E_{m_t})\ \bar{Q}(z)$ is r. It is easy to see that any k-order minor of $C'(z)$ equals 0 if $k > r_t$. On the other hand, since $\bar{Q}(0)$ is row independent, there exists a nonzero r_t-order minor $\bar{Q}(1, \ldots, r_t; j_1, \ldots, j_{r_t}; z)$ of $\bar{Q}(z)$. It follows that the r_t-order minor $C'(1, \ldots, r_t; j_1, \ldots, j_{r_t}; z)$ is nonzero. Thus r_t is the rank of $C'(z)$. It follows that $r = r_t$. □

4.4.2 Relations Between R_a R_b Transformation and Canonical Diagonal Form

Given $C_0(z)$ in $M_{m,n}(F[z])$ with rank r, there exist $P(z) \in GL_m(F[z])$, $Q(z) \in GL_n(F[z])$, r nonnegative integers $a_1, \ldots, a_r$, and r polynomials $f_1(z), \ldots, f_r(z)$ such that

$$C_0(z) = P(z)DIA_{m,n}(z^{a_1}f_1(z), \ldots, z^{a_r}f_r(z), 0, \ldots, 0)Q(z), \tag{4.43}$$

$0 \leqslant a_1 \leqslant \cdots \leqslant a_r$, $f_j(z) \mid f_{j+1}(z)$ for $j = 1, \ldots, r-1$, and $f_j(0) \neq 0$ for $j = 1, \ldots, r$.

Let (4.40) be a terminating and elementary R_a R_b transformation sequence. From Corollaries 4.4.1 and 4.4.2, we can construct an $m \times m$ invertible matrix polynomial $\bar{P}(z)$ such that (4.42) holds, that is,

$$C_0(z) = \bar{P}(z)DIA_{m,r}(E_{m_1}, zE_{m_2}, \ldots, z^{t-1}E_{m_t})\bar{Q}(z),$$

where $m_1 = r_1$, $m_j = r_j - r_{j-1}$, $j = 2, \ldots, t$, and $\bar{Q}(z)$ is the first r rows of $C_t(z)$. Denote $b_i = k - 1$ for $r_{k-1} < i \leqslant r_k$, $1 \leqslant k \leqslant t$, $i = 1, \ldots, r$. Then $DIA_{m,r}(E_{m_1}, zE_{m_2}, \ldots, z^{t-1}E_{m_t}) = DIA_{m,r}(z^{b_1}, z^{b_2}, \ldots, z^{b_r})$. It follows that

$$C_0(z) = \bar{P}(z)DIA_{m,r}(z^{b_1}, z^{b_2}, \ldots, z^{b_r})\bar{Q}(z). \tag{4.44}$$

Notice that $\bar{Q}(z)$ has r rows and rank r and $\bar{Q}(0)$ is row independent. Thus there exist $\bar{M}(z) \in GL_r(F[z])$, $\bar{R}(z) \in GL_n(F[z])$, and r polynomials $g_1(z), \ldots, g_r(z)$ such that

$$\bar{Q}(z) = \bar{M}(z)DIA_{r,n}(g_1(z), \ldots, g_r(z))\bar{R}(z),$$

$g_j(z) \mid g_{j+1}(z)$ for $j = 1, \ldots, r-1$, and $g_j(0) \neq 0$ for $j = 1, \ldots, r$. From (4.44), it follows that

$$C_0(z) = \bar{P}(z)DIA_{m,r}(z^{b_1}, z^{b_2}, \ldots, z^{b_r})\bar{M}(z)DIA_{r,n}(g_1(z), \ldots, g_r(z))\bar{R}(z). \tag{4.45}$$

Since P, Q, $\bar{P}$ and $\bar{R}$ are invertible, from (4.43) and (4.45), D_{af} and $D_b\bar{M}(z)D_g$ are equivalent [1] and their determinant factors are the same, where

$$\begin{aligned} D_{af} &= DIA_{m,n}(z^{a_1}f_1(z), \ldots, z^{a_r}f_r(z), 0, \ldots, 0), \\ D_b &= DIA_{m,r}(z^{b_1}, z^{b_2}, \ldots, z^{b_r}), \\ D_g &= DIA_{r,n}(g_1(z), \ldots, g_r(z)). \end{aligned}$$

[1] $A(z)$ and $B(z)$ in $M_{m,n}(F[z])$ are equivalent if and only if there exist $P'(z)$ in $GL_m(F[z])$ and $Q'(z)$ in $GL_n(F[z])$ such that $A(z) = P'(z)B(z)Q'(z)$.

Lemma 4.4.3. *For any $A(z) \in GL_n(F[z])$ and any $r \leqslant n$, let*

$$d_{i_1,\ldots,i_r} = \gcd\{A(i_1,\ldots,i_r;j_1,\ldots,j_r;z),\ 1 \leqslant j_1 < \cdots < j_r \leqslant n\},$$
$$d'_{j_1,\ldots,j_r} = \gcd\{A(i_1,\ldots,i_r;j_1,\ldots,j_r;z),\ 1 \leqslant i_1 < \cdots < i_r \leqslant n\}.$$

Then $d_{i_1,\ldots,i_r}$ and $d'_{j_1,\ldots,j_r}$ are nonzero constants.

Proof. Using Laplace expansion theorem, for any $t < n$, we have

$$\begin{aligned} &A(i_1,\ldots,i_{t+1};j_1,\ldots,j_{t+1};z) \\ &\quad = \sum_{k=1}^{t+1}(-1)^{t+1+k}a_{i_{t+1}j_k}(z)A(i_1,\ldots,i_t;j_1,\ldots,j_{k-1},j_{k+1},\ldots,j_{t+1};z) \\ &\quad = a(z)d_{i_1,\ldots,i_t} \end{aligned}$$

for some $a(z) \in F[z]$, where $a_{ij}(z)$ is the element at row i and column j of $A(z)$. Thus $d_{i_1,\ldots,i_{t+1}} = a(z)d_{i_1,\ldots,i_t}$ for some $a(z) \in F[z]$. It follows that $d_{i_1,\ldots,i_n} = a(z)d_{i_1,\ldots,i_r}$ for some $a(z) \in F[z]$. Therefore, $|A(z)| = a(z)d_{i_1,\ldots,i_r}$ for some $a(z) \in F[z]$. Since $A(z) \in GL_n(F[z])$, $|A(z)|$ is a nonzero element in F. Thus $d_{i_1,\ldots,i_r}$ is a nonzero element in F.

Similarly, expanding $A(i_1,\ldots,i_{t+1};j_1,\ldots,j_{t+1};z)$ by the $(t+1)$-th column, we can prove that $d'_{j_1,\ldots,j_r}$ is a nonzero element in F. □

Lemma 4.4.4. *For any i, $1 \leqslant i \leqslant r$, we have $a_1 + \cdots + a_i = b_1 + \cdots + b_i$.*

Proof. Consider the $(1,\ldots,i;j_1,\ldots,j_i)$ minor of $D_b\bar{M}(z)D_g$, $j_i \leqslant r$. Noticing the shape of matrices, this minor is equal to $z^{b_1+\cdots+b_i}\ \bar{M}(1,\ldots,i;j_1,\ldots,j_i;z)\ g_{j_1}(z)\ldots g_{j_i}(z)$. Consider the set

$$S = \{\bar{M}(1,\ldots,i;j_1,\ldots,j_i;z),\ 1 \leqslant j_1 < \cdots < j_i \leqslant r\}.$$

Denote the greatest common divisor of polynomials in S by $d(z)$. From Lemma 4.4.3, $d(z)$ is a nonzero constant. By $\varphi_{1,\ldots,i}(z)$ we denote the greatest common divisor of all $(1,\ldots,i;j_1,\ldots,j_i)$ minors of $D_b\bar{M}(z)D_g$ for $1 \leqslant j_1 < \cdots < j_i \leqslant r$. It follows that

$$\varphi_{1,\ldots,i}(z) = z^{b_1+\cdots+b_i}\varphi'_{1,\ldots,i}(z),$$

for some $\varphi'_{1,\ldots,i}$ with $\varphi'_{1,\ldots,i}(0) \neq 0$.

On the other hand, for any $1 \leqslant k_1 < \cdots < k_i \leqslant m$ and $1 \leqslant j_1 < \cdots < j_i \leqslant n$, $k_i > r$ or $j_i > r$, the $(k_1,\ldots,k_i;j_1,\ldots,j_i)$ minor of $D_b\bar{M}(z)D_g$ is 0. For any $1 \leqslant k_1 < \cdots < k_i \leqslant r$ and $1 \leqslant j_1 < \cdots < j_i \leqslant r$, the $(k_1,\ldots,k_i;j_1,\ldots,j_i)$ minor of $D_b\bar{M}(z)D_g$ has a factor $z^{b_{k_1}+\cdots+b_{k_i}}$. Clearly, $b_{k_1} + \cdots + b_{k_i} \geqslant b_1 + \cdots + b_i$ holds. Thus the multiplicity of z in the i-order determinant factor of $D_b\bar{M}(z)D_g$ is $b_1 + \cdots + b_i$. Notice that the determinant

factors of D_{af} and of $D_b\bar{M}(z)D_g$ are the same. Since the i-order determinant factor of D_{af} is $z^{a_1+\cdots+a_i}\ f_1(z)\dots f_i(z)$ and $f_j(0)\neq 0$ for any j, $1\leqslant j\leqslant i$, we have $z^{a_1+\cdots+a_i}=z^{b_1+\cdots+b_i}$. □

Lemma 4.4.5. *For any i, $1\leqslant i\leqslant r$, we have $f_1(z)\dots f_i(z)=g_1(z)\dots g_i(z)$.*

Proof. Similar to the discussion in the proof of Lemma 4.4.4, the $(j_1,\dots, j_i;\ 1,\dots, i)$ minor of $D_b\ \bar{M}(z)D_g$ equals $z^{b_{j_1}+\cdots+b_{j_i}}\ \bar{M}(j_1,\dots,j_i;1,\dots,i;z)$ $g_1(z)\dots g_i(z)$ if $j_i\leqslant r$. Consider the set

$$S'=\{\bar{M}(j_1,\dots,j_i;1,\dots,i;z),\ 1\leqslant j_1<\cdots<j_i\leqslant r\},$$

and denote the greatest common divisor of polynomials in S' by $d'(z)$. From Lemma 4.4.3, $d'(z)$ is a nonzero constant. By $\psi_{1,\dots,i}(z)$ we denote the greatest common divisor of all $(j_1,\dots,j_i;1,\dots,i)$ minors of $D_b\bar{M}(z)D_g$ for $1\leqslant j_1<\cdots<j_i\leqslant r$. It follows that

$$\psi_{1,\dots,i}(z)=z^b g_1(z)\dots g_i(z)$$

for some nonnegative integer b.

On the other hand, for any $1\leqslant k_1<\cdots<k_i\leqslant n$ and $1\leqslant j_1<\cdots<j_i\leqslant m$, $k_i>r$ or $j_i>r$, the $(j_1,\dots,j_i;k_1,\dots,k_i)$ minor of $D_b\bar{M}(z)D_g$ is 0. For any $1\leqslant k_1<\cdots<k_i\leqslant r$ and $1\leqslant j_1<\cdots<j_i\leqslant r$, the $(j_1,\dots,j_i;k_1,\dots,k_i)$ minor of $D_b\bar{M}(z)D_g$ has a factor $g_{k_1}(z)\dots g_{k_i}(z)$ which has the factor $g_1(z)\dots g_i(z)$. Thus the non z factor in the i-order determinant factor of $D_b\bar{M}(z)D_g$ is $g_1(z)\dots g_i(z)$. Notice that the determinant factors of D_{af} and of $D_b\bar{M}(z)D_g$ are the same. Since the i-order determinant factor of D_{af} is $z^{a_1+\cdots+a_i}\ f_1(z)\dots f_i(z)$ and $f_j(0)\neq 0$ for any j, $1\leqslant j\leqslant i$, we have $f_1(z)\dots f_i(z)=g_1(z)\dots g_i(z)$. □

Lemma 4.4.6. *$a_j=b_j$ and $f_j(z)=g_j(z)$ for $j=1,\dots,r$.*

Proof. From Lemmas 4.4.4 and 4.4.5. □

Theorem 4.4.2. *Let $DIA_{m,n}(z^{a_1}f_1(z),\dots,z^{a_r}f_r(z),0,\dots,0)$ be the canonical diagonal form of $C_0(z)\in M_{m,n}\ (F\ [z])$, where $f_j(0)\neq 0$, $j=1,\dots,r$. Let (4.40) be a terminating and elementary R_a R_b transformation sequence, and $\bar{Q}(z)$ the first r rows of $C_t(z)$.*

(a) *There exists $\bar{P}(z)\in GL_m(F[z])$ such that degrees of elements in columns r'_k+1 to r'_{k+1} of $\bar{P}(z)$ are at most $t-k-1$ for $k=0,1,\dots,t-1$, and*

$$C_0(z)=\bar{P}(z)DIA_{m,r}(z^{a_1},\dots,z^{a_r})\bar{Q}(z),$$

where $r'_0=0$, $r'_i=r_i$, $i=1,\dots,t-1$, and $r'_t=m$.

(b) *$DIA_{r,n}(f_1(z),\dots,f_r(z))$ is the canonical diagonal form of $\bar{Q}(z)$.*

Proof. (a) From Corollaries 4.4.1 and 4.4.2, and Lemma 4.4.6.

(b) From Lemma 4.4.6 and Corollary 4.4.2. □

4.4.3 Relations of Right-Parts

From now on, we denote

$$D(z) = DIA_{r,r}(z^{a_1}, \ldots, z^{a_r})$$

with $0 \leqslant a_1 \leqslant \cdots \leqslant a_r = t-1$. Let $m_i = \max j\ [\exists k(1 \leqslant k \leqslant r\ \&\ a_k = a_{k+1} = \cdots = a_{k+j-1} = i-1)]$, $i = 1, \ldots, t$. Then

$$D(z) = DIA_{r,r}(E_{m_1}, zE_{m_2}, \ldots, z^{t-1}E_{m_t}).$$

Lemma 4.4.7. *For any $L(z) \in M_{r,r}(F[z])$, $\bar{Q}(z)$, $Q^*(z) \in M_{r,n}(F[z])$, assume that*

$$L(z)D(z)Q^*(z) = D(z)\bar{Q}(z) \tag{4.46}$$

and $Q^*(0)$ is row independent. If $L(z) = L_0 + zL_1 + z^2L_2 + \cdots + z^{t-1}L_{t-1} + z^tL_t(z)$ and for any h, $0 \leqslant h < t$, L_h is partitioned into blocks $L_h = [L_{hij}]_{1 \leqslant i,j \leqslant t}$ with $m_i \times m_j$ L_{hij}, then $L_{hij} = 0$ whenever $i - j > h$.

Proof. Denote $D(z)\bar{Q}(z) = A_0 + zA_1 + z^2A_2 + \cdots + z^{t-1}A_{t-1} + z^tA_t(z)$ and $D(z)Q^*(z) = A_0^* + zA_1^* + z^2A_2^* + \cdots + z^{t-1}A_{t-1}^* + z^tA_t^*(z)$. For any k, $0 \leqslant k < t$, partition A_k and A_k^* into t blocks

$$A_k = \begin{bmatrix} A_{1,k+1} \\ \vdots \\ A_{t,k+1} \end{bmatrix}, \quad A_k^* = \begin{bmatrix} A_{1,k+1}^* \\ \vdots \\ A_{t,k+1}^* \end{bmatrix},$$

where $A_{i,k+1}$ and $A_{i,k+1}^*$ have m_i rows, $i = 1, \ldots, t$. Since $D(z) = DIA_{r,r}$ $(E_{m_1}, zE_{m_2}, \ldots, z^{t-1}E_{m_t})$, we have $A_{ij} = A_{ij}^* = 0$ for any i, j, $1 \leqslant j < i \leqslant t$. Since $Q^*(0)$ is row independent, rows of $A_{11}^*, \ldots, A_{tt}^*$ are linearly independent.

For any k, $0 \leqslant k < t$, we prove, by induction on k, the proposition $P(k)$: $L_{hij} = 0$ if $t \geqslant i \geqslant j + h + 1$, $1 \leqslant j \leqslant k - h + 1$ and $0 \leqslant h \leqslant k$. *Basis* : $k = 0$. Comparing constant terms in two sides of (4.46), we have

$$L_0 \begin{bmatrix} A_{11}^* \\ 0 \\ \vdots \\ 0 \end{bmatrix} = \begin{bmatrix} A_{11} \\ 0 \\ \vdots \\ 0 \end{bmatrix}.$$

It follows immediately that

$$\begin{bmatrix} L_{021} \\ \vdots \\ L_{0t1} \end{bmatrix} A_{11}^* = 0.$$

Noticing that A_{11}^* is row independent, we have $L_{0i1} = 0$ for any i, $2 \leqslant i \leqslant t$. Thus $P(0)$ holds. *Induction step* : Suppose that $P(k-1)$ holds and $k < t$. That is, $L_{hij} = 0$ for $t \geqslant i \geqslant j + h + 1$, $1 \leqslant j \leqslant k - h$ and $0 \leqslant h \leqslant k - 1$. Comparing coefficients of z^k in two sides of (4.46), we have

$$L_k \begin{bmatrix} A_{11}^* \\ 0 \\ 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix} + L_{k-1} \begin{bmatrix} A_{12}^* \\ A_{22}^* \\ 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix} + \cdots + L_1 \begin{bmatrix} A_{1k}^* \\ \vdots \\ A_{kk}^* \\ 0 \\ \vdots \\ 0 \end{bmatrix} + L_0 \begin{bmatrix} A_{1,k+1}^* \\ \vdots \\ A_{k+1,k+1}^* \\ 0 \\ \vdots \\ 0 \end{bmatrix} = \begin{bmatrix} A_{1,k+1} \\ \vdots \\ A_{k+1,k+1} \\ 0 \\ \vdots \\ 0 \end{bmatrix}.$$

It follows immediately that

$$\begin{bmatrix} A_{k+1,k+1} \\ 0 \\ \vdots \\ 0 \end{bmatrix}$$

$$= L_k^{(k)} \begin{bmatrix} A_{11}^* \\ 0 \\ 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix} + L_{k-1}^{(k)} \begin{bmatrix} A_{12}^* \\ A_{22}^* \\ 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix} + \cdots + L_1^{(k)} \begin{bmatrix} A_{1k}^* \\ \vdots \\ A_{kk}^* \\ 0 \\ \vdots \\ 0 \end{bmatrix} + L_0^{(k)} \begin{bmatrix} A_{1,k+1}^* \\ \vdots \\ A_{k+1,k+1}^* \\ 0 \\ \vdots \\ 0 \end{bmatrix},$$

where

$$L_h^{(k)} = \begin{bmatrix} L_{h,k+1,1} & \cdots & L_{h,k+1,t} \\ L_{h,k+2,1} & \cdots & L_{h,k+2,t} \\ \cdots & \cdots & \cdots \\ L_{ht1} & \cdots & L_{htt} \end{bmatrix}, \ h = 0, 1, \ldots, k.$$

From the induction hypothesis, this yields

$$\begin{bmatrix} A_{k+1,k+1} \\ 0 \\ \vdots \\ 0 \end{bmatrix}$$

$$= L_k^{(k)} \begin{bmatrix} A_{11}^* \\ 0 \\ 0 \\ 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix} + L_{k-1}^{(k)} \begin{bmatrix} 0 \\ A_{22}^* \\ 0 \\ 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix} + \cdots + L_1^{(k)} \begin{bmatrix} 0 \\ \vdots \\ A_{kk}^* \\ 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix} + L_0^{(k)} \begin{bmatrix} 0 \\ \vdots \\ 0 \\ A_{k+1,k+1}^* \\ 0 \\ \vdots \\ 0 \end{bmatrix}.$$

Since rows of $A_{11}^*, \ldots, A_{k+1,k+1}^*$ are linear independent, we have $L_{h,i,k-h+1} = 0$ for $t \geqslant i \geqslant k+2$ and $0 \leqslant h \leqslant k$. From $P(k-1)$, this yields $P(k)$.

From $P(t-1)$, $L_{hij} = 0$ for any $0 \leqslant h < t$ and $i - j > h$. □

Lemma 4.4.8. *For any $L(z) \in M_{r,r}(F[z])$, $\bar{Q}(z)$, $Q^*(z) \in M_{r,n}(F[z])$, assume that* (4.46) *holds and $Q^*(0)$ is row independent. Then there exists $R(z) \in M_{r,r}(F[z])$ such that*

$$\bar{Q}(z) = R(z)Q^*(z).$$

Proof. Denote $L(z) = L_0 + zL_1 + z^2L_2 + \cdots + z^{t-1}L_{t-1} + z^tL_t(z)$. Partition L_k into blocks $(L_{kij})_{1\leqslant i,j\leqslant t}$ with $m_i \times m_j$ L_{kij}, $k = 0, 1, \ldots, t-1$. Partition $z^kL_kD(z)$ into blocks $(L'_{kij}(z))_{1\leqslant i,j\leqslant t}$ with $m_i \times m_j$ $L'_{kij}(z)$, $k = 0, 1, \ldots, t-1$. From Lemma 4.4.7, it is easy to see that

$$L'_{kij}(z) = \begin{cases} z^{k+j-1}L_{kij}, & \text{if } i - j \leqslant k, \\ 0, & \text{otherwise.} \end{cases}$$

It follows that for any k, $0 \leqslant k \leqslant t-1$,

$$z^kL_kD(z) = D(z)R_k(z),$$

where

$$R_k(z) = [R_{kij}(z)]_{1\leqslant i,j\leqslant t},$$
$$R_{kij}(z) = \begin{cases} z^{k+j-i}L_{kij}, & \text{if } i - j \leqslant k, \\ 0, & \text{otherwise.} \end{cases}$$

Clearly, there exists a matrix polynomial $R_t(z)$ such that

$$z^tL_t(z)D(z) = D(z)R_t(z).$$

Let

$$R(z) = \sum_{k=0}^{t-1} R_k(z) + R_t(z).$$

We then have

$$L(z)D(z) = D(z)R(z).$$

Using (4.46), this yields

$$D(z)R(z)Q^*(z) = D(z)\bar{Q}(z).$$

It follows that $R(z)Q^*(z) = \bar{Q}(z)$. □

Lemma 4.4.9. *For any* $R(z), R'(z) \in M_{r,r}(F[z])$ *and* $\bar{Q}(z), Q^*(z) \in M_{r,n}(F[z])$, *if*

$$Q^*(z) = R(z)\bar{Q}(z),\ \bar{Q}(z) = R'(z)Q^*(z) \tag{4.47}$$

hold and $\bar{Q}(0)$ *or* $Q^*(0)$ *are row independent, then*

$$R'(z)R(z) = R(z)R'(z) = E_r,$$

therefore, $R(z)$ *and* $R'(z)$ *are invertible.*

Proof. Suppose that $\bar{Q}(0)$ is row independent. From (4.47), we have

$$\bar{Q}(z) = R'(z)R(z)\bar{Q}(z).$$

Let $T(z) = E_r - R'(z)R(z)$. Denote $T(z) = T_0 + zT_1 + \cdots + z^tT_t$. Then we have

$$(T_0 + zT_1 + \cdots + z^tT_t)\bar{Q}(z) = 0. \tag{4.48}$$

We prove $T_i = 0$ for any i, $0 \leqslant i \leqslant t$ by induction on i. Denote $\bar{Q}(z) = Q_0 + zQ_1 + \cdots + z^sQ_s$. In the case of $i = 0$, from (4.48), we have $T_0Q_0 = 0$. Since $Q_0 = \bar{Q}(0)$ is row independent, we obtain $T_0 = 0$. Suppose that we have proven $T_j = 0$ for $0 \leqslant j \leqslant i-1 \leqslant t-1$. Using (4.48), we have $T_iQ_0 = 0$. Since Q_0 is row independent, we obtain $T_i = 0$. We conclude $T(z) = 0$. It immediately follows that $R'(z)R(z) = E_r$. Therefore, $R(z)R'(z) = E_r$.

The proof is similar in the case where $Q^*(0)$ is row independent. □

For any $C(z) \in M_{m,n}(F[z])$ with rank r, $P(z) \in GL_m(F[z])$ and $Q(z) \in GL_n(F[z])$, if $P^{-1}(z)\ C(z)\ Q^{-1}(z)$ is the canonical diagonal form of $C(z)$, say $DIA_{m,n}(z^{a_1}f_1(z), \ldots, z^{a_r}f_r(z), 0, \ldots, 0)$, with $f_j(0) \neq 0$, $j = 1, \ldots, r$, $DIA_{r,n}(f_1(z), \ldots, f_r(z))\ Q(z)$ is called a *right-part* of $C(z)$.

Notice that the constant term of a right-part of $C(z)$ is row independent.

Theorem 4.4.3. *Given* $C_0(z)$ *in* $M_{m,n}(F[z])$ *with rank* r, *assume that* $Q^*(z)$ *is a right-part of* $C_0(z)$ *and* $DIA_{m,n}(z^{a_1}f_1(z), \ldots, z^{a_r}f_r(z), 0, \ldots, 0)$ *is the canonical diagonal form of* $C_0(z)$, *where* $0 \leqslant a_1 \leqslant \cdots \leqslant a_r$, $f_j(z) \mid f_{j+1}(z)$ *for* $j = 1, \ldots, r-1$, *and* $f_j(0) \neq 0$ *for* $j = 1, \ldots, r$. *Let* (4.40), *i.e.*,

$$C_k(z) \overset{R_a[P_k]}{\longrightarrow} C'_k(z),\ C'_k(z) \overset{R_b[r_{k+1}]}{\longrightarrow} C_{k+1}(z),\quad k = 0, 1, \ldots, t-1$$

be a terminating and elementary R_a R_b transformation sequence, and $\bar{Q}(z)$ the submatrix of the first r rows of $C_t(z)$. Then there exists $R(z) \in GL_r(F[z])$ such that

$$Q^*(z) = R(z)\bar{Q}(z).$$

Proof. Assume that

$$C_0(z) = P(z)DIA_{m,n}(z^{a_1}f_1(z), \ldots, z^{a_r}f_r(z), 0 \ldots, 0)Q(z)$$

and

$$Q^*(z) = DIA_{r,n}(f_1(z), \ldots, f_r(z))Q(z)$$

for some $P(z) \in GL_m(F[z])$ and $Q(z) \in GL_n(F[z])$. Then we have

$$C_0(z) = P(z)DIA_{m,r}(z^{a_1}, \ldots, z^{a_r})Q^*(z).$$

From Theorem 4.4.2 (a), there exists $\bar{P}(z) \in GL_m(F[z])$ such that

$$C_0(z) = \bar{P}(z)DIA_{m,r}(z^{a_1}, \ldots, z^{a_r})\bar{Q}(z).$$

It follows that

$$P(z)DIA_{m,r}(z^{a_1}, \ldots, z^{a_r})Q^*(z) = \bar{P}(z)DIA_{m,r}(z^{a_1}, \ldots, z^{a_r})\bar{Q}(z).$$

Thus there exists $P'(z) \in GL_m(F[z])$ such that

$$P'(z)DIA_{m,r}(z^{a_1}, \ldots, z^{a_r})Q^*(z) = DIA_{m,r}(z^{a_1}, \ldots, z^{a_r})\bar{Q}(z).$$

Let $L(z)$ be the submatrix of the first r rows and the first r columns of $P'(z)$. Then we have

$$L(z)DIA_{r,r}(z^{a_1}, \ldots, z^{a_r})Q^*(z) = DIA_{r,r}(z^{a_1}, \ldots, z^{a_r})\bar{Q}(z),$$

that is,

$$L(z)D(z)Q^*(z) = D(z)\bar{Q}(z).$$

Symmetrically, there exists $L'(z) \in M_{r,r}(F[z])$ such that

$$L'(z)D(z)\bar{Q}(z) = D(z)Q^*(z).$$

From Corollary 4.4.2, we have $r = r_t$. It follows that $\bar{Q}(0)$ is row independent. Since $Q(0)$ is row independent, $Q^*(0)$ is row independent. From Lemmas 4.4.8, there exist $R(z)$ and $R'(z)$ in $M_{r,r}(F[z])$ such that

$$Q^*(z) = R(z)\bar{Q}(z),\ \bar{Q}(z) = R'(z)Q^*(z).$$

Using Lemmas 4.4.9, $R(z)$ is invertible. This completes the proof of the theorem. □

Corollary 4.4.3. *Under the hypothesis of Theorem* 4.4.3, $Q^*(0)$ *and* $\bar{Q}(0)$ *are row equivalent.*

Corollary 4.4.4. *Let* $\psi(x_1, \ldots, x_s)$ *be a vector function of dimension* n *in* s *variables over* F. *Under the hypothesis of Theorem* 4.4.3, *the following results hold.*

(a) *For any parameters* $x_{l+1}, \ldots, x_s$, $Q^*(0)\psi(x_1, \ldots, x_s)$ *is injective if and only if for any parameters* $x_{l+1}, \ldots, x_s$, $\bar{Q}(0)\psi(x_1, \ldots, x_s)$ *is injective.*

(b) *For any parameters* $x_{l+1}, \ldots, x_s$, $Q^*(0)\psi(x_1, \ldots, x_s)$ *is surjective if and only if for any parameters* $x_{l+1}, \ldots, x_s$, $\bar{Q}(0)\psi(x_1, \ldots, x_s)$ *is surjective.*

Corollary 4.4.5. *For any* $C_0(z)$ *in* $M_{m,n}(F[z])$ *with rank* r. *Assume that* $P(z)^{-1}\, C_0(z)\, Q(z)^{-1}$ *is the canonical diagonal form of* $C_0(z)$ *for some* $P(z) \in GL_m(F[z])$ *and* $Q(z) \in GL_n(F[z])$ *and that*

$$C_k(z) \overset{R_a[P_k]}{\longrightarrow} C'_k(z),\ C'_k(z) \overset{R_b[r_{k+1}]}{\longrightarrow} C_{k+1}(z), \quad k = 0, 1, \ldots, t-1$$

is a terminating and elementary R_a R_b *transformation sequence. Let* $\tilde{Q}(z)$ *and* $\bar{Q}(z)$ *be the submatrix of the first* r *rows of* $Q(z)$ *and* $C_t(z)$, *respectively. Let* $\psi(x_1, \ldots, x_s)$ *be a vector function of dimension* n *in* s *variables over* F.

(a) *For any parameters* $x_{l+1}, \ldots, x_s$, $\tilde{Q}(0)\psi(x_1, \ldots, x_s)$ *is injective if and only if for any parameters* $x_{l+1}, \ldots, x_s$, $\bar{Q}(0)\psi(x_1, \ldots, x_s)$ *is injective.*

(b) *For any parameters* $x_{l+1}, \ldots, x_s$, $\tilde{Q}(0)\psi(x_1, \ldots, x_s)$ *is surjective if and only if for any parameters* $x_{l+1}, \ldots, x_s$, $\bar{Q}(0)\psi(x_1, \ldots, x_s)$ *is surjective.*

4.4.4 Existence of Terminating R_a R_b Transformation Sequence

Recall some notations in Sect. 4.1, but R is restricted to a finite field $GF(q)$. Let U and X be two finite nonempty sets. Let Y be a column vector space of dimension m over $GF(q)$, where m is a positive integer. Let $l_X = \log_q |X|$. For any integer i, we use x_i (x'_i), u_i and y_i (y'_i) to denote elements in X, U and Y, respectively.

Let $\psi^l_{\mu\nu}$ be a column vector of dimension l of which each component is a single-valued mapping from $U^{\mu+1} \times X^{\nu+1}$ to Y for some integers $\mu \geqslant -1$, $\nu \geqslant 0$ and $l \geqslant 1$. For any integers $h \geqslant 0$ and i, let

$$\psi^{lh}_{\mu\nu}(u, x, i) = \begin{bmatrix} \psi^l_{\mu\nu}(u(i, \mu+1), x(i, \nu+1)) \\ \vdots \\ \psi^l_{\mu\nu}(u(i-h, \mu+1), x(i-h, \nu+1)) \end{bmatrix}.$$

For any nonnegative integer c, let $eq_c(i)$ be an equation

$$\varphi_c(y(i+c, c+k+1)) + [B_{0c}, \ldots, B_{hc}]\psi^{lh}_{\mu\nu}(u, x, i) = 0$$

and let $eq'_c(i)$ be an equation

$$\varphi'_c(y(i+c, c+k+1)) + [B'_{0c}, \ldots, B'_{hc}]\psi^{lh}_{\mu\nu}(u, x, i) = 0,$$

where φ_c and φ'_c are two single-valued mappings from Y^{c+k+1} to Y, B_{jc} and B'_{jc} are $m \times l$ matrices over $GF(q)$, $j = 0, 1, \ldots, h$.

An R_a R_b transformation sequence

$$eq_c(i) \overset{R_a[P_c]}{\longrightarrow} eq'_c(i),\ eq'_c(i) \overset{R_b[r_{c+1}]}{\longrightarrow} eq_{c+1}(i),\quad c = 0, 1, \ldots, e \tag{4.49}$$

is said to be (t, e) *circular*, if $0 \leqslant t \leqslant e$ and $B_{j,e+1} = B_{jt}$, $j = 0, 1, \ldots, h$. (4.49) is said to be *circular*, if it is (t, e) circular, for some t. The R_a R_b transformation sequence (4.49) is said to be *terminating*, if the last $m - r_{e+1}$ rows of $B_{j,e+1}$ are 0 in the case of $r_t < m$, $j = 0, 1, \ldots, h$, and the first r_{e+1} rows of $B_{0,e+1}$ are linearly independent over $GF(q)$.

Let $M = \langle X, Y, Y^k \times U^{h+\mu+1} \times X^{h+\nu}, \delta, \lambda\rangle$ be a finite automaton over $GF(q)$ defined by

$$\begin{aligned}
&\delta(\langle y(i-1,k), u(i, h+\mu+1), x(i-1, h+\nu)\rangle, x_i)\\
&\quad = \langle y(i,k), u(i+1, h+\mu+1), x(i, h+\nu)\rangle,\\
&\lambda(\langle y(i-1,k), u(i, h+\mu+1), x(i-1, h+\nu)\rangle, x_i) = y_i,
\end{aligned}$$

where

$$\begin{aligned}
y_i &= \varphi(y(i-1,k)) + [B_0, \ldots, B_h]\psi^{lh}_{\mu\nu}(u, x, i),\\
u_{i+1} &= g(u(i, h+\mu+1), x(i, h+\nu+1)),
\end{aligned}$$

φ is a single-valued mapping from Y^k to Y, B_0, …, B_h are $m \times l$ matrices over $GF(q)$, and g is a single-valued mapping from $U^{h+\mu+1} \times X^{h+\nu+1}$ to U.

Let $eq_0(i)$ be the equation

$$-y_i + \varphi(y(i-1,k)) + [B_0, \ldots, B_h]\psi^{lh}_{\mu\nu}(u, x, i) = 0. \tag{4.50}$$

For any state $s = \langle y(-1,k), u(0, h+\mu+1), x(-1, h+\nu)\rangle$ of M and any nonnegative integer n, let

$$\begin{aligned}
&Y^s_n = \{\lambda(s, x_0 \ldots x_n) \mid x_0, \ldots, x_n \in X\},\\
&W^s_n = \{w_0 \ldots w_n \mid w_i = y_i - \varphi(y_{i-1}, \ldots, y_{i-k}), i = 0, 1, \ldots, n,\ y_0 \ldots y_n \in Y^s_n\}.
\end{aligned}$$

Lemma 4.4.10. $|W^s_n| = |Y^s_n|$.

Proof. From the definition of W^s_n, we have $|W^s_n| \leqslant |Y^s_n|$. Let $M_w = \langle Y, Y, Y^k, \delta_w, \lambda_w\rangle$ be a k-order input-memory finite automaton defined by

$$w_i = y_i - \varphi(y_{i-1}, \ldots, y_{i-k}),\ i = 0, 1, \ldots$$

Clearly, $w_0 \ldots w_n \in W^s_n$ if and only if $w_0 \ldots w_n = \lambda_w(\langle y_{-1}, \ldots, y_{-k}\rangle, y_0 \ldots y_n)$ for some $y_0 \ldots y_n \in Y^s_n$. Since M_w is weakly invertible with delay 0, we have $|W^s_n| \not< |Y^s_n|$. Thus $|W^s_n| = |Y^s_n|$. □

Lemma 4.4.11. (a) *If M is weakly invertible with delay τ, then $|Y_n^s| \geqslant q^{l_X(n-\tau+1)}$.*

(b) *If M is a weak inverse with delay τ and a state s of M τ-matches some state, then $|Y_n^s| \geqslant q^{m(n-\tau+1)}$.*

Proof. (a) We prove $|Y_n^s| \geqslant q^{l_X(n-\tau+1)}$ by reduction to absurdity. Suppose to the contrary that $|Y_n^s| < q^{l_X(n-\tau+1)}$. Then there exist $x_0, \ldots, x_n, x'_0, \ldots, x'_n \in X$ such that

$$x_0 \ldots x_{n-\tau} \neq x'_0 \ldots x'_{n-\tau} \tag{4.51}$$

and

$$\lambda(s, x_0 \ldots x_n) = \lambda(s, x'_0 \ldots x'_n). \tag{4.52}$$

Since M is weakly invertible with delay τ, (4.52) yields $x_0 \ldots x_{n-\tau} = x'_0 \ldots x'_{n-\tau}$. This contradicts (4.51). Thus $|Y_n^s| \geqslant q^{l_X(n-\tau+1)}$.

(b) Assume that M is a weak inverse with delay τ of $M' = \langle Y, X, S', \delta', \lambda' \rangle$ and a state s of M τ-matches some state s' of M'. Then for any $y'_0, \ldots, y'_n \in Y$ there exist $y_0, \ldots, y_{\tau-1} \in Y$ such that

$$\lambda(s, \lambda'(s', y'_0 \ldots y'_n)) = y_0 \ldots y_{\tau-1} y'_0 \ldots y'_{n-\tau}.$$

Since $y'_0 \ldots y'_{n-\tau}$ may take $q^{m(n-\tau+1)}$ values, we have $|Y_n^s| \geqslant q^{m(n-\tau+1)}$. □

Lemma 4.4.12. (a) *If M is weakly invertible with delay τ, then $|W_n^s| \geqslant q^{l_X(n-\tau+1)}$.*

(b) *If M is a weak inverse with delay τ and a state s of M τ-matches some state, then $|W_n^s| \geqslant q^{m(n-\tau+1)}$.*

Proof. From Lemmas 4.4.10 and 4.4.11, the lemma holds. □

Consider the R_a R_b transformation sequence (4.49), where $eq_0(i)$ is (4.50). Let $s = \langle y(-1,k), u(0,h+\mu+1), x(-1,h+\nu) \rangle$ be a state of M. For any $n \geqslant 0$ and any c, $0 \leqslant c \leqslant e+1$, let

$$\begin{aligned} W_{nc}^s = \{ w_{0c} \ldots w_{nc} \mid\ & w_{ic} = [B_{0c}, \ldots, B_{hc}] \psi_{\mu\nu}^{lh}(u, x, i),\ i = 0, 1, \ldots, n, \\ & u_{j+1} = g(u(j, h+\mu+1), x(j, h+\nu+1)), \\ & j = 0, 1, \ldots, n-1,\ x_0, x_1, \ldots, x_n \in X \}, \end{aligned}$$

and for any $n \geqslant 0$ and any c, $0 \leqslant c \leqslant e$, let

$$\begin{aligned} W_{nc}'^s = \{ w'_{0c} \ldots w'_{nc} \mid\ & w'_{ic} = [B'_{0c}, \ldots, B'_{hc}] \psi_{\mu\nu}^{lh}(u, x, i),\ i = 0, 1, \ldots, n, \\ & u_{j+1} = g(u(j, h+\mu+1), x(j, h+\nu+1)), \\ & j = 0, 1, \ldots, n-1,\ x_0, x_1, \ldots, x_n \in X \}. \end{aligned}$$

Lemma 4.4.13. $W_{n0}^s = W_n^s$.

Lemma 4.4.14. $|W^s_{nc}| = |W'^s_{nc}|$.

Proof. From the definition of R_a, we have

$$W'^s_{nc} = \{w'_{0c} \dots w'_{nc} \mid w'_{ic} = P_c w_{ic}, i = 0, 1, \dots, n,\ w_{0c} \dots w_{nc} \in W^s_{nc}\}.$$

Since P_c is nonsingular, we obtain $|W^s_{nc}| = |W'^s_{nc}|$. □

Lemma 4.4.15. $|W^s_{n-1,c+1}| \geqslant |W^s_{nc}| q^{-m}$.

Proof. For any element $w'_{0c} \dots w'_{nc}$ in W'^s_{nc}, denoting the first r_{c+1} rows and the last $m - r_{c+1}$ rows of w'_{ic} by w'^u_{ic} and w'^b_{ic}, respectively, $i = 0, 1, \dots, n$, from the definition of R_b, we have $w_{0,c+1} \dots w_{n-1,c+1} \in W^s_{n-1,c+1}$, where the first r_{c+1} rows and the last $m - r_{c+1}$ rows of $w_{i,c+1}$ are w'^u_{ic} and $w'^b_{i+1,c}$, respectively, $i = 0, 1, \dots, n-1$. Since the number of elements in W'^s_{nc} which have the same value of $w'^u_{0c} \dots w'^u_{n-1,c}\ w'^b_{1c} \dots w'^b_{nc}$ is at most q^m (the number of values of $w'^b_{0c}\ w'^u_{nc}$), we have $|W^s_{n-1,c+1}| \geqslant |W'^s_{nc}| q^{-m}$. Using Lemma 4.4.14, it immediately follows that $|W^s_{n-1,c+1}| \geqslant |W^s_{nc}| q^{-m}$. □

Lemma 4.4.16. $|W^s_{n-j,c+j}| \geqslant |W^s_{nc}| q^{-jm}$, *for any* j, $1 \leqslant j \leqslant n, e - c + 1$.

Proof. We prove the lemma by induction on j. *Basis* : $j = 1$. The result holds from Lemma 4.4.15. *Induction step* : suppose that $|W^s_{n-j+1,c+j-1}| \geqslant |W^s_{nc}| q^{-(j-1)m}$. From Lemma 4.4.15, we have $|W^s_{n-j,c+j}| \geqslant |W^s_{n-j+1,c+j-1}|$ q^{-m}. It follows from induction hypothesis that $|W^s_{n-j,c+j}| \geqslant |W^s_{nc}| q^{-jm}$. □

Lemma 4.4.17. (a) *If M is weakly invertible with delay τ, then for any j,* $1 \leqslant j \leqslant n, e + 1$,

$$|W^s_{n-j,j}| \geqslant q^{l_X(n-\tau+1)-jm}.$$

(b) *If M is a weak inverse with delay τ and a state s of M τ-matches some state, then for any j,* $1 \leqslant j \leqslant n, e + 1$,

$$|W^s_{n-j,j}| \geqslant q^{m(n-\tau-j+1)}.$$

Proof. From Lemma 4.4.16 for $c = 0$, we have $|W^s_{n-j,j}| \geqslant |W^s_{n0}| q^{-jm}$ for any j, $1 \leqslant j \leqslant n, e + 1$. Using Lemma 4.4.13, it follows that $|W^s_{n-j,j}| \geqslant |W^s_n| q^{-jm}$. In the case where M is weakly invertible with delay τ, using Lemma 4.4.12 (a), we have $|W^s_{n-j,j}| \geqslant q^{l_X(n-\tau+1)} q^{-jm} = q^{l_X(n-\tau+1)-jm}$. In the case where M is a weak inverse with delay τ, using Lemma 4.4.12 (b), we have $|W^s_{n-j,j}| \geqslant q^{m(n-\tau+1)} q^{-jm} = q^{m(n-\tau-j+1)}$. □

Lemma 4.4.18. *Assume that the R_a R_b transformation sequence* (4.49) *is elementary and (t, e) circular. Then we have $r_{t+1} = r_{t+2} = \cdots = r_{e+1}$ and $w_{it} = w_{i,e+1}$, $i \geqslant 0$, where*

$$w_{ic} = [B_{0c}, \dots, B_{hc}] \psi^{lh}_{\mu\nu}(u, x, i),\ i = 0, 1, \dots \tag{4.53}$$

Proof. Since (4.49) is elementary, we have $r_{t+1} \leqslant r_{t+2} \leqslant \cdots \leqslant r_{e+1}$. Since (4.49) is (t, e) circular, we have $B_{j,e+1} = B_{jt}$, $j = 0, 1, \ldots, h$. Thus there exists an elementary R_a R_b transformation $eq_{e+1}(i) \overset{R_a[P_t]}{\longrightarrow} eq'_{e+1}(i)$, $eq'_{e+1}(i) \overset{R_b[r_{t+1}]}{\longrightarrow} eq_{e+2}(i)$. It follows that $r_{e+1} \leqslant r_{t+1}$. Therefore, $r_{t+1} = r_{t+2} = \cdots = r_{e+1}$. From the definition, using $B_{j,e+1} = B_{jt}$ for $j = 0, 1, \ldots, h$, we have $w_{it} = w_{i,e+1}$ for $i \geqslant 0$. □

Let (4.49) be an elementary and (t, e) circular R_a R_b transformation sequence. Taking

$$P_{t+c(e-t+1)+j} = P_{t+j}, r_{t+c(e-t+1)+j+1} = r_{t+j+1}, c = 1, 2, \ldots, j = 0, 1, \ldots, e-t,$$

it is evident that for any $n \geqslant e$,

$$eq_c(i) \overset{R_a[P_c]}{\longrightarrow} eq'_c(i),\ eq'_c(i) \overset{R_b[r_{c+1}]}{\longrightarrow} eq_{c+1}(i),\quad c = 0, 1, \ldots, n \tag{4.54}$$

is an elementary R_a R_b transformation sequence. Such an R_a R_b transformation sequence (4.54) is called a *natural expansion* of (4.49).

Let w_{ic} be defined by (4.53) for any nonnegative integer c. From the definition of the natural expansion, it is easy to see that $w_{i,t+c(e-t+1)+j} = w_{i,t+j}$, for any $c \geqslant 1$ and any j, $0 \leqslant j \leqslant e-t$.

Lemma 4.4.19. *Assume that the R_a R_b transformation sequence* (4.49) *is elementary and (t, e) circular. Then for any $c, 1 \leqslant c \leqslant e-t+1$, there exist a single-valued mapping f_c from $(GF(q)^{r_{t+1}})^c$ to $GF(q)^{m-r_{t+1}}$ and an $(m-r_{t+1}) \times (m-r_{t+1})$ nonsingular matrix $\bar{P}_c$ over $GF(q)$ such that*

$$\begin{aligned} &w^u_{i,t+c} = w^u_{it}, \\ &w^b_{i,t+c} = f_c(w^u_{i+1,t}, \ldots, w^u_{i+c,t}) + \bar{P}_c w^b_{i+c,t}, \\ &\quad i = 0, 1, \ldots, \end{aligned} \tag{4.55}$$

where $w^u_{p,t+j}$ and $w^b_{p,t+j}$ are the first r_{t+1} rows and the last $m - r_{t+1}$ rows of $w_{p,t+j}$ which is defined by (4.53)*, respectively.*

Proof. We prove the result for the case of $r_t = r_{t+1}$. First at all, we have the following proposition: for any j, $0 \leqslant j \leqslant e-t$,

$$\begin{aligned} &w^u_{i,t+j+1} = w^u_{i,t+j}, \\ &w^b_{i,t+j+1} = P'_{t+j} w^u_{i+1,t+j} + P''_{t+j} w^b_{i+1,t+j}, \\ &\quad i = 0, 1, \ldots, \end{aligned} \tag{4.56}$$

where

$$P_{t+j} = \begin{bmatrix} E_{r_{t+j}} & 0 \\ P'_{t+j} & P''_{t+j} \end{bmatrix}$$

in (4.49). In fact, denoting

$$w'_{ic} = [B'_{0c}, \ldots, B'_{hc}]\psi^{lh}_{\mu\nu}(u, x, i),\ i = 0, 1, \ldots,$$

from Lemma 4.4.18 and the definitions of R_a and R_b, we have

$$w'_{i,t+j} = P_{t+j}w_{i,t+j} = \begin{bmatrix} w^u_{i,t+j} \\ P'_{t+j}w^u_{i,t+j} + P''_{t+j}w^b_{i,t+j} \end{bmatrix}$$

and

$$w_{i,t+j+1} = \begin{bmatrix} w^u_{i,t+j} \\ P'_{t+j}w^u_{i+1,t+j} + P''_{t+j}w^b_{i+1,t+j} \end{bmatrix}.$$

Thus (4.56) holds. We now prove the lemma by induction on c. *Basis* : $c = 1$. From (4.56) with $j = 0$, (4.55) holds in the case of $c = 1$. *Induction step* : suppose that (4.55) holds in the case of c and $1 \leqslant c \leqslant e - t$. From (4.56) with $j = c$ and the induction hypothesis, we have

$$\begin{aligned} w^u_{i,t+c+1} &= w^u_{i,t+c} = w^u_{it}, \\ w^b_{i,t+c+1} &= P'_{t+c}w^u_{i+1,t+c} + P''_{t+c}w^b_{i+1,t+c} \\ &= P'_{t+c}w^u_{i+1,t} + P''_{t+c}(f_c(w^u_{i+2,t}, \ldots, w^u_{i+1+c,t}) + \bar{P}_c w^b_{i+1+c,t}). \end{aligned}$$

Taking

$$f_{c+1}(w^u_{i+1,t}, \ldots, w^u_{i+(c+1),t}) = P'_{t+c}w^u_{i+1,t} + P''_{t+c}f_c(w^u_{i+2,t}, \ldots, w^u_{i+1+c,t})$$

and

$$\bar{P}_{c+1} = P''_{t+c}\bar{P}_c,$$

it follows that

$$\begin{aligned} w^u_{i,t+(c+1)} &= w^u_{it}, \\ w^b_{i,t+(c+1)} &= f_{c+1}(w^u_{i+1,t}, \ldots, w^u_{i+(c+1),t}) + \bar{P}_{c+1}w^b_{i+(c+1),t}. \end{aligned}$$

Thus (4.55) holds in the case of $c + 1$.

Below we prove the lemma for the case of $r_t < r_{t+1}$. Take a natural expansion of (4.49), say (4.54), with $n = e + (e - t + 1)$. Denote $e' = n$ and $t' = e + 1$. Clearly, (4.54) is elementary and (t', e') circular. Noticing that (4.54) is also (t, e') circular, from Lemma 4.4.18, we have $r_{i+1} = r_{t+1}$ for any i, $t < i \leqslant e'$. It follows that $r_{t'} = r_{t'+1}$. Since the lemma for the case of $r_t = r_{t+1}$ is true, replacing (4.49) by (4.54), we obtain that for any c, $1 \leqslant c \leqslant e' - t' + 1$, there exist a single-valued mapping f_c from $(GF(q)^{r_{t'+1}})^c$ to $GF(q)^{m-r_{t'+1}}$ and an $(m - r_{t'+1}) \times (m - r_{t'+1})$ nonsingular matrix $\bar{P}_c$ over $GF(q)$ such that

$$w^u_{i,t'+c} = w^u_{it'},\ w^b_{i,t'+c} = f_c(w^u_{i+1,t'}, \ldots, w^u_{i+c,t'}) + \bar{P}_c w^b_{i+c,t'},\ i = 0, 1, \ldots$$

Notice that $e' - t' + 1 = e - t + 1$, $r_{t+1} = r_{t'+1}$, and $w_{i,t+j} = w_{i,t'+j}$ for any j, $0 \leqslant j \leqslant e - t + 1$. Thus for any c, $1 \leqslant c \leqslant e - t + 1$, f_c is a single-valued mapping from $(GF(q)^{r_{t+1}})^c$ to $GF(q)^{m-r_{t+1}}$, $\bar{P}_c$ is an $(m-r_{t+1}) \times (m-r_{t+1})$ nonsingular matrix over $GF(q)$, and

$$w^u_{i,t+c} = w^u_{it},\ w^b_{i,t+c} = f_c(w^u_{i+1,t}, \ldots, w^u_{i+c,t}) + \bar{P}_c w^b_{i+c,t},\ i = 0, 1, \ldots$$

That is, (4.55) holds. □

Lemma 4.4.20. *Let* (4.49) *be an elementary and* (t, e) *circular* R_a R_b *transformation sequence. Then there exist a single-valued mapping* f *from* $(GF(q)^{r_{t+1}})^{e-t+1}$ *to* $GF(q)^{m-r_{t+1}}$ *and an* $(m - r_{t+1}) \times (m - r_{t+1})$ *nonsingular matrix* P *over* $GF(q)$ *such that*

$$w^b_{i+e-t+1,t} = f(w^u_{i+1,t}, \ldots, w^u_{i+e-t+1,t}) + P w^b_{it},\ i = 0, 1, \ldots,$$

where w^u_{jt} *and* w^b_{jt} *are the first* r_{t+1} *rows and the last* $m - r_{t+1}$ *rows of* w_{jt} *which is defined by* (4.53)*, respectively.*

Proof. From Lemma 4.4.18, $w_{it} = w_{i,e+1}$, $i = 0, 1, \ldots$ Using Lemma 4.4.19 with $c = e - t + 1$, there exist a single-valued mapping f_{e-t+1} from $(GF(q)^{r_{t+1}})^{e-t+1}$ to $GF(q)^{m-r_{t+1}}$ and an $(m-r_{t+1}) \times (m-r_{t+1})$ nonsingular matrix $\bar{P}_{e-t+1}$ over $GF(q)$ such that

$$w^b_{it} = w^b_{i,e+1} = f_{e-t+1}(w^u_{i+1,t}, \ldots, w^u_{i+e-t+1,t}) + \bar{P}_{e-t+1} w^b_{i+e-t+1,t},$$
$$i = 0, 1, \ldots$$

It follows that

$$w^b_{i+e-t+1,t} = -\bar{P}^{-1}_{e-t+1} f_{e-t+1}(w^u_{i+1,t}, \ldots, w^u_{i+e-t+1,t}) + \bar{P}^{-1}_{e-t+1} w^b_{it},$$
$$i = 0, 1, \ldots$$ □

Lemma 4.4.21. *Assume that the* R_a R_b *transformation sequence* (4.49) *is elementary and* (t, e) *circular. Let* (4.54) *be a natural expansion of* (4.49)*. Then we have* $|W^s_{nt}| \leqslant q^{m(e+1)+r_{t+1}(n-e)}$.

Proof. Assume that w_{ic} is defined by (4.53) for any nonnegative integer c. Then $w_{i,t+c(e-t+1)+j} = w_{i,t+j}$, for any $c \geqslant 1$ and any j, $0 \leqslant j \leqslant e - t$. Denote the first r_{t+1} rows and the last $m - r_{t+1}$ rows of w_{ij} by w^u_{ij} and w^b_{ij}, respectively. From Lemma 4.4.20, $w^b_{i+e-t+1,t}$ can be uniquely determined by w^b_{it} and $w^u_{i+1,t}, \ldots, w^u_{i+e-t+1,t}$, for $i = t, t+1, \ldots, n-(e-t+1)$. It follows that $w^b_{e+1,t} \ \ldots \ w^b_{n,t}$ can be uniquely determined by $w^b_{tt}, \ldots, w^b_{et}$ and $w^u_{t+1,t}, \ldots, w^u_{nt}$. Since the number of values of $w_{0t} \ \ldots \ w_{et} \ w^u_{e+1,t} \ \ldots \ w^u_{nt}$ is at most $(q^m)^{e+1}(q^{r_{t+1}})^{n-e}$, we have

$$|W^s_{nt}| \leqslant (q^m)^{e+1}(q^{r_{t+1}})^{n-e} = q^{m(e+1)+r_{t+1}(n-e)}.$$ □

Theorem 4.4.4. *Assume that the R_a R_b transformation sequence* (4.49) *is elementary and (t, e) circular.*

(a) *If M is weakly invertible, then $l_X \leqslant r_{e+1}$.*

(b) *If M is a weak inverse, then $r_{e+1} = m$.*

Proof. (a) Assume that M is weakly invertible with delay τ. From Lemma 4.4.17 (a) (with values t, $n + t$ for parameters j, n, respectively) and Lemma 4.4.21, we have

$$q^{l_X(n+t-\tau+1)-tm} \leqslant |W^s_{nt}| \leqslant q^{m(e+1)+r_{t+1}(n-e)},$$

whenever n is large enough. It follows that

$$q^{n(l_X-r_{t+1})} \leqslant q^{l_X(\tau-t-1)+m(t+e+1)-r_{t+1}e},$$

whenever n is large enough. We prove $l_X \leqslant r_{t+1}$ by reduction to absurdity. Suppose to the contrary that $r_{t+1} < l_X$. Then we have

$$\infty = \lim_{n\to\infty} q^{n(l_X-r_{t+1})} \leqslant q^{l_X(\tau-t-1)+m(t+e+1)-r_{t+1}e}.$$

This is a contradiction. We conclude $r_{t+1} \geqslant l_X$. From Lemma 4.4.18, $r_{e+1} = r_{t+1}$. It follows that $r_{e+1} \geqslant l_X$.

(b) Assume that M is a weak inverse with delay τ. From Lemma 4.4.17 (b) (with values t, $n+t$ for parameters j, n, respectively) and Lemma 4.4.21, we have

$$q^{m(n-\tau+1)} \leqslant |W^s_{nt}| \leqslant q^{m(e+1)+r_{t+1}(n-e)},$$

whenever n is large enough. It follows that

$$q^{n(m-r_{t+1})} \leqslant q^{m(\tau+e)-r_{t+1}e},$$

whenever n is large enough. We prove $m \leqslant r_{t+1}$ by reduction to absurdity. Suppose to the contrary that $r_{t+1} < m$. Then we have

$$\infty = \lim_{n\to\infty} q^{n(m-r_{t+1})} \leqslant q^{m(\tau+e)-r_{t+1}e}.$$

This is a contradiction. We conclude $r_{t+1} \geqslant m$. From $m \geqslant r_{e+1} = r_{t+1}$, it follows that $r_{e+1} = m$. □

Corollary 4.4.6. *Assume that the elementary R_a R_b transformation sequence* (4.49) *is (t, e) circular.*

(a) *If M is invertible, then $l_X \leqslant r_{e+1}$;*

(b) *If M is an inverse, then $r_{e+1} = m$.*

Theorem 4.4.5. (a) *If $m = l_X$ and M is invertible or weakly invertible, then there exists an elementary and terminating R_a R_b transformation sequence of which $eq_0(i)$ is* (4.50).

(b) *If M is an inverse or a weak inverse, then there exists an elementary and terminating R_a R_b transformation sequence of which $eq_0(i)$ is* (4.50).

Proof. Clearly, there exists an elementary and circular R_a R_b transformation sequence (4.49) of which $eq_0(i)$ is (4.50). Since $r_{e+1} \leqslant m$, from Corollary 4.4.6 or Theorem 4.4.4, we have $m = r_{e+1}$. It follows that (4.49) is terminating. □

Historical Notes

References [107, 108] give a feasible inversion method using linear R_a R_b transformation for some kind of finite automata, and [108] derives a relation between linear R_a R_b transformation sequences. Section 4.1 is based on [108] but extends the scope of objects. Section 4.2 is based on [108] (but extends the scope of objects) and [107], where Lemma 4.2.2 is enhanced according to Lemmas 5.5 and 5.6 in [135]. From the viewpoint of automata, [83] proposes an inversion method by reduced echelon matrix and [28] proposes an inversion method by canonical diagonal matrix polynomial; these methods are feasible for some kind of finite automata. The equivalence between the inversion method by reduced echelon matrix and the linear R_a R_b transformation method is given in [108, 135, 137]. Section 4.3 is based on [108]. The equivalence between the inversion method by canonical diagonal matrix polynomial and the linear R_a R_b transformation method is given in [132], and Sect. 4.4 is based on [132, 121].

5. Structure of Feedforward Inverses

Renji Tao

Institute of Software, Chinese Academy of Sciences
Beijing 100080, China trj@ios.ac.cn

Summary.

In Chaps. 1 and 3, we have adopted two methods, the state tree method and the R_a R_b transformation method, to deal with the structure problem. The former is suitable for general finite automata but not easy to manipulate for large parameters. Contrarily, the latter is easy to manipulate but only suitable for very special finite automata — linear or quasi-linear ones. For nonlinear finite automata, the investigation meets with difficulties.

A feedforward invertible finite automaton is more complex in structure as compared with its feedforward inverse. We first explore the structure problem for the simple. This chapter presents two approaches to the investigation for small delay cases. A decision criterion for feedforward inverse finite automata with delay τ is proven and used to derive an explicit expression for ones of delay 0 which lays a foundation of a canonical form for one key cryptosystems in Chap. 8. In another approach based on mutual invertibility of finite automata, we give an explicit expression for feedforward inverse finite automata with delay 1 and for binary feedforward inverse finite automata with delay 2.

Key words: *semi-input-memory finite automata, feedforward inverse, weakly invertible*

In Chaps. 1 and 3, we have adopted two methods, the state tree method and the R_a R_b transformation method, to deal with the structure problem. The former is suitable for general finite automata but not easy to manipulate for large parameters. Contrarily, the latter is easy to manipulate but only suitable for very special finite automata — linear or quasi-linear ones over finite fields. In general, for nonlinear finite automata, the investigation meets up with difficulties for lack of mathematical tools.

Although semi-input-memory finite automata are nonlinear, they have simpler structure as compared with general finite automata. So a feedforward

invertible finite automaton is more complex in structure as compared with its feedforward inverse. We first explore the structure problem for the simple. In this chapter, we present two approaches to the investigation for small delay cases. A decision criterion for feedforward inverse finite automata with delay τ is proven and used to derive an explicit expression for ones of delay 0 which lays a foundation of a canonical form for one key cryptosystems in Chap. 8. In another approach based on mutual invertibility of finite automata, we give an explicit expression for feedforward inverse finite automata with delay 1 and for binary feedforward inverse finite automata with delay 2.

5.1 A Decision Criterion

Let $M' = \langle Y, X, S', \delta', \lambda' \rangle$ be a c-order semi-input-memory finite automaton $\mathcal{SIM}(M_a, f)$, where $S' = Y^c \times S_a$, $M_a = \langle Y_a, S_a, \delta_a, \lambda_a \rangle$ is an autonomous finite automaton, and f is a single-valued mapping from $Y^{c+1} \times \lambda_a(S_a)$ to X. The restriction of f on a subset of $Y^{c+1} \times \lambda_a(S_a)$ is still denoted by f.

Let $u_1, \ldots, u_n$ be $n\,(\geqslant 1)$ different states of $M_a = \langle Y_a, S_a, \delta_a, \lambda_a \rangle$. If $\delta_a(u_i) = u_{i+1}$, $i = 1, \ldots, n-1$, $\delta_a(u_n) = u_1$, $\{u_1, \ldots, u_n\}$ is called a *cycle* of M_a.

$M_a = \langle Y_a, S_a, \delta_a, \lambda_a \rangle$ is said to be *cyclic* , if S_a is a cycle of M_a. M_a is said to be *strongly cyclic* , if $Y_a = S_a$, $\lambda_a(s_a) = s_a$ holds for any $s_a \in S_a$, and M_a is cyclic.

Theorem 5.1.1. *M' is a feedforward inverse with delay τ if and only if there exists a finite subautomaton $\bar{M}_a$ of M_a such that $\bar{M}_a$ is cyclic and $\mathcal{SIM}(\bar{M}_a, f)$ is a feedforward inverse with delay τ.*

Proof. Since $\mathcal{SIM}(\bar{M}_a, f)$ is a finite subautomaton of M', the *if* part is evident. To prove the *only if* part, suppose that M' is a feedforward inverse with delay τ. Then there exists a finite automaton $M = \langle X, Y, S, \delta, \lambda \rangle$ such that M' is a weak inverse with delay τ of M. Let s be in S. Then there exists $s' = \langle y_{-1}, \ldots, y_{-c}, t \rangle$ in S' such that s' τ-matches s. Let $\delta_a^0(t) = t$ and $\delta_a^{i+1}(t) = \delta_a(\delta_a^i(t))$ for any $i \geqslant 0$. Consider the infinite sequence t, $\delta_a(t)$, $\delta_a^2(t)$, ... Since S_a is finite, some states occur repetitively in the sequence. Let the earliest repetitive states be $\delta_a^p(t)$ and $\delta_a^{p+r}(t)$. Take $\bar{S}_a = \{\delta_a^p(t), \delta_a^{p+1}(t), \ldots, \delta_a^{p+r-1}(t)\}$. Clearly, $\bar{S}_a$ is closed in M_a. Thus there exists a finite subautomaton $\bar{M}_a$ of M_a such that the state alphabet of $\bar{M}_a$ is $\bar{S}_a$. On the other hand, let $\bar{S} = \{\delta(s, \alpha) \mid \alpha \in X^*, |\alpha| \geqslant p\}$. It is evident that $\bar{S}$ is closed with respect to X in M. Thus there exists a finite subautomaton $\bar{M}$ of M such that the input, output and state alphabets of $\bar{M}$ are X, Y and $\bar{S}$, respectively. We prove $\mathcal{SIM}(\bar{M}_a, f)$ is a weak inverse with delay τ

of $\bar{M}$. Let $\bar{s}$ be in $\bar{S}$. Then there exists α in X^* such that $|\alpha| \geqslant p$ and $\bar{s} = \delta(s, \alpha)$. Let $\lambda(s, \alpha) = \beta = y_0 \dots y_{k-1}$, where $k = |\alpha|$. Clearly, $\delta'(s', \beta) = \langle y_{k-1}, \dots, y_{k-c}, \delta_a^k(t)\rangle$. From $k \geqslant p$, $\delta_a^k(t)$ is in $\bar{S}_a$. It follows that $\delta'(s', \beta)$ is also a state of the finite subautomaton $\mathcal{SIM}(\bar{M}_a, f)$ of M'. Since the state s' of M' τ-matches the state s of M and $\beta = \lambda(s, \alpha)$, the state $\delta'(s', \beta)$ of M' τ-matches the state $\delta(s, \alpha)$ $(= \bar{s})$ of M. Thus the state $\delta'(s', \beta)$ of $\mathcal{SIM}(\bar{M}_a, f)$ τ-matches the state $\bar{s}$ of $\bar{M}$. We conclude that $\mathcal{SIM}(\bar{M}_a, f)$ is a feedforward inverse with delay τ of $\bar{M}$. Therefore, $\mathcal{SIM}(\bar{M}_a, f)$ is a weak inverse with delay τ. □

For any t in S_a, we use f_t to denote a single-valued mapping from Y^{c+1} to X, defined by $f_t(y_c, \dots, y_0) = f(y_c, \dots, y_0, \lambda_a(t))$, $y_0, \dots, y_c \in Y$. For any τ, $0 \leqslant \tau \leqslant c$, and any $t \in S_a$, let

$$F_{f_t}^{(\tau)} = \{x_0 \dots x_c y_0 \dots y_c | x_0, \dots, x_c \in X, y_0, \dots, y_c \in Y, f_t(y_c, \dots, y_0) = x_{c-\tau}\}.$$

Algorithm of $F_t^{(\tau)}, t \in S_a$

Input : An autonomous finite automaton $M_a = \langle Y_a, S_a, \delta_a, \lambda_a\rangle$, a single-valued mapping f from $Y^{c+1} \times \lambda_a(S_a)$ to X.

Output : Sets $F_t^{(\tau)}$, $t \in S_a$.

Procedure :

1. Take $F_t = F_{f_t}^{(\tau)}$, $t \in S_a$.
2. For each $t \in S_a$, each $x_i \in X$, each $y_i \in Y$, $i = 1, \dots, c$, if there exists $x_{c+1} \in X$ such that $x_1 \dots x_{c+1} y_1 \dots y_{c+1} \notin F_t$ holds for any $y_{c+1} \in Y$, then delete elements $x_0 \dots x_c y_0 \dots y_c$, $x_0 \in X$, $y_0 \in Y$ from $F_{f_{t'}}$, for any $t' \in \delta_a^{-1}(t)$ $(= \{t_0 \in S_a \mid \delta_a(t_0) = t\})$.
3. Repeat Step 2 until no element can be deleted.
4. Output $F_t^{(\tau)} = F_t$, $t \in S_a$, and stop.

From the algorithm of $F_t^{(\tau)}, t \in S_a$, it is easy to show the following lemma.

Lemma 5.1.1. (a) $F_t^{(\tau)} \subseteq F_{f_t}^{(\tau)}$, $t \in S_a$.

(b) *If* $t = \delta_a(u)$ *and* $F_t^{(\tau)} = \varnothing$, *then* $F_u^{(\tau)} = \varnothing$.

(c) *For any* $t \in S_a$, *any* $x_0 \dots x_c y_0 \dots y_c \in F_t^{(\tau)}$ *and any* $x_{c+1} \in X$, *there exists* $y_{c+1} \in Y$ *such that* $x_1 \dots x_{c+1} y_1 \dots y_{c+1}$ *is in* $F_v^{(\tau)}$, *where* $v = \delta_a(t)$.

Theorem 5.1.2. *M' is a feedforward inverse with delay τ if and only if there exists t in S_a such that $F_t^{(\tau)} \neq \varnothing$.*

Proof. only if : Suppose that M' is a feedforward inverse with delay τ. Then there exists a finite automaton $M = \langle X, Y, S, \delta, \lambda\rangle$ such that M' is a weak inverse with delay τ of M. Let s be a state of M. Then there exists a state $s' = \langle y_{-1}, \dots, y_{-c}, t\rangle$ of M' such that s' τ-matches s. Thus for any $x_0, x_1, \dots$ in X and any $y_0, y_1, \dots$ in Y, if $y_0 y_1 \dots = \lambda(s, x_0 x_1 \dots)$, then there

exist $x_{-\tau}, \ldots, x_{-1}$ in X such that $\lambda'(s', y_0 y_1 \ldots) = x_{-\tau} \ldots x_{-1} x_0 x_1 \ldots$ From the construction of M', it is easy to see that $x_{i-\tau} = f(y_i, \ldots, y_{i-t}, \lambda_a(\delta_a^i(t)))$, $i = 0, 1, \ldots$ Thus for any $i \geqslant c$, $x_{i-c} \ldots x_i y_{i-c} \ldots y_i$ is in $F_{f_u}^{(\tau)}$, where $u = \delta_a^i(t)$. For any $u \in \{\delta_a^i(t), i = c, c+1, \ldots\}$, define $F'_u = \{x_{i-c} \ldots x_i y_{i-c} \ldots y_i \mid i \geqslant c$, $x_{i-c}, \ldots, x_i \in X$, $u = \delta_a^i(t)$, there exist $x_0, \ldots, x_{i-c-1} \in X$ such that $\lambda(\delta(s, x_0 \ldots x_{i-c-1}), x_{i-c} \ldots x_i) = y_{i-c} \ldots y_i \}$. Then we have $F'_u \subseteq F_{f_u}^{(\tau)}$ for any $u \in \{\delta_a^i(t), i = c, c+1, \ldots\}$. We prove the proposition: for any $u \in \{\delta_a^i(t), i = c, c+1, \ldots\}$, any $a_0 \ldots a_c b_0 \ldots b_c \in F'_u$, and any $a_{c+1} \in X$, there exists $b_{c+1} \in Y$ such that $a_1 \ldots a_{c+1} b_1 \ldots b_{c+1} \in F'_v$, where $v = \delta_a(u)$. In fact, from the definition of F'_u, there exist $i \geqslant c$, $x_0, \ldots, x_i \in X$, and $y_0, \ldots, y_i \in Y$ such that $\lambda(s, x_0 \ldots x_i) = y_0 \ldots y_i$, $x_{i-c} \ldots x_i y_{i-c} \ldots y_i = a_0 \ldots a_c b_0 \ldots b_c$, and $u = \delta_a^i(t)$. Let $b_{c+1} = \lambda(\delta(s, x_0 \ldots x_i), a_{c+1})$. Clearly, $\lambda(s, x_0 \ldots x_i\ a_{c+1}) = y_0 \ldots y_i b_{c+1}$. Since $v = \delta_a(u) = \delta_a^{i+1}(t)$, from the definition of F'_v, we have $x_{i-c+1} \ldots x_i a_{c+1} y_{i-c+1} \ldots y_i b_{c+1} = a_1 \ldots a_{c+1} b_1 \ldots b_{c+1} \in F'_v$.

Since $\{\delta_a^i(t), i = c, c+1, \ldots\}$ is finite, there exist different elements $u_1, \ldots, u_n$ in $\{\delta_a^i(t), i = c, c+1, \ldots\}$ such that $\delta_a(u_i) = u_{i+1}$, $i = 1, \ldots, n-1$ and $\delta_a(u_n) = u_1$. Since $F'_u \subseteq F_{f_u}^{(\tau)}$ holds for any $u \in \{u_1, \ldots, u_n\}$, using the above proposition, by induction on steps of the algorithm of $F_t^{(\tau)}, t \in S_a$, it is easy to prove that $F'_u \subseteq F_u^{(\tau)}$ holds for any $u \in \{u_1, \ldots, u_n\}$. For any $u \in \{u_1, \ldots, u_n\}$, from $F'_u \neq \varnothing$, we have $F_u^{(\tau)} \neq \varnothing$.

if : Suppose that $F_t^{(\tau)} \neq \varnothing$ for some t in S_a. From Lemma 5.1.1 (b), it is easy to see that $F_u^{(\tau)} \neq \varnothing$ for any u in $\{\delta_a^i(t), i = 0, 1, \ldots\}$. Since $\{\delta_a^i(t), i = 0, 1, \ldots\}$ is finite, there exist different elements $u_1, \ldots, u_n$ in $\{\delta_a^i(t), i = 0, 1, \ldots\}$ such that $\delta_a(u_i) = u_{i+1}$, $i = 1, \ldots, n-1$ and $\delta_a(u_n) = u_1$. We construct $M = \langle X, Y, S, \delta, \lambda \rangle$ as follows. Take $S = \{\langle x_0 \ldots x_c y_0 \ldots y_c, u \rangle \mid x_0 \ldots x_c y_0 \ldots y_c \in F_u^{(\tau)}$, $u = u_1, \ldots, u_n\}$. Let $\langle x_0 \ldots x_c y_0 \ldots y_c, u \rangle$ be in S. Then $x_0 \ldots x_c y_0 \ldots y_c \in F_u^{(\tau)}$ and $u \in \{u_1, \ldots, u_n\}$. From Lemma 5.1.1 (c), for any $x_{c+1} \in X$, there exists $y_{c+1} \in Y$ such that $x_1 \ldots x_{c+1} y_1 \ldots y_{c+1} \in F_v^{(\tau)}$, where $v = \delta_a(u)$. Clearly, $v \in \{u_1, \ldots, u_n\}$. It follows that $\langle x_1 \ldots x_{c+1} y_1 \ldots y_{c+1}, v \rangle$ is in S. Choose arbitrarily such a y_{c+1}, and define $\delta(\langle x_0 \ldots x_c y_0 \ldots y_c, u \rangle, x_{c+1}) = \langle x_1 \ldots x_{c+1} y_1 \ldots y_{c+1}, \delta_a(u) \rangle$ and $\lambda(\langle x_0 \ldots x_c y_0 \ldots y_c, u \rangle, x_{c+1}) = y_{c+1}$. We prove that M' is a weak inverse with delay τ of M. For any state $s = \langle x_{-c-1} \ldots x_{-1}\ y_{-c-1} \ldots y_{-1}, u \rangle$ of M, let $s' = \langle y_{-1}, \ldots, y_{-c}, \delta_a(u) \rangle$ which is a state of M'. For any $x_0, x_1, \ldots$ in X, let $\lambda(s, x_0 x_1 \ldots) = y_0 y_1 \ldots$, where $y_0, y_1, \ldots \in Y$. From the construction of M, it is easy to see that $\delta(s, x_0 \ldots x_i) = \langle x_{i-c} \ldots x_i y_{i-c} \ldots y_i, \delta_a^{i+1}(u) \rangle$, $i = 0, 1, \ldots$ Therefore, for any $i \geqslant 0$, we have $x_{i-c} \ldots x_i y_{i-c} \ldots y_i \in F_v^{(\tau)}$, where $v = \delta_a^{i+1}(u)$. From $F_v^{(\tau)} \subseteq F_{f_v}^{(\tau)}$, it follows that $f(y_i, \ldots, y_{i-c}, \lambda_a(\delta_a^{i+1}(u))) = x_{i-\tau}$, $i = 0, 1, \ldots$ Let $\delta'(s', y_0 \ldots y_{i-1}) = s'_i$, $i = 0, 1, \ldots$ Then $s'_i = \langle y_{i-1}, \ldots, y_{i-c}, \delta_a^{i+1}(u) \rangle$, $i = 0, 1, \ldots$ Thus $\lambda'(s'_i, y_i) = f(y_i, \ldots, y_{i-c}, \lambda_a(\delta_a^{i+1}(u))) = x_{i-\tau}$, $i = 0, 1, \ldots$

It follows that $\lambda'(s', y_0y_1\ldots) = x_{-\tau}\ldots x_{-1}x_0x_1\ldots$ Thus s' τ-matches s. We conclude that M' is a weak inverse with delay τ of M. Therefore, M' is a feedforward inverse with delay τ. □

Corollary 5.1.1. *If there exists a cycle C of M_a such that for any $u \in C$ and any $y_{-1}, \ldots, y_{-c} \in Y, |f(Y, y_{-1}, \ldots, y_{-c}, \lambda_a(u))| = |X|$ holds, then for any $\tau, 0 \leqslant \tau \leqslant c, M'$ is a feedforward inverse with delay τ.*

Proof. It is easy to verify that for any $u \in C$, any $x_0\ldots x_cy_0\ldots y_c \in F^{(\tau)}_{f_u}$ and any $x_{c+1} \in X$, there exists $y_{c+1} \in Y$ such that $f(y_{c+1}, \ldots, y_1, \lambda_a(\delta_a(u))) = x_{c-\tau+1}$. It follows that $x_1\ldots x_{c+1}y_1\ldots y_{c+1} \in F^{(\tau)}_{f_v}$, where $v = \delta_a(u) \in C$. For any $u \in C$, from the algorithm of $F^{(\tau)}_t, t \in S_a$, we have $F^{(\tau)}_u = F^{(\tau)}_{f_u} \neq \varnothing$. From Theorem 5.1.2, M' is a feedforward inverse with delay τ. □

5.2 Delay Free

Lemma 5.2.1. *Let $M = \langle X, Y, S, \delta, \lambda\rangle$, $M' = \langle Y, X, S', \delta', \lambda'\rangle$ and $M'' = \langle Y, X, S', \delta'', \lambda''\rangle$ be finite automata, and $s' \in S'$. Assume that $|\lambda'(s', Y)| < |X|$ and that $\lambda'(s'', y) = \lambda''(s'', y)$ holds for any $s'' \in S' \setminus \{s'\}$ and any $y \in Y$. If M' is a weak inverse with delay 0 of M, then M'' is a weak inverse with delay 0 of M.*

Proof. Suppose that M' is a weak inverse with delay 0 of M. Let s be a state of M. Then there exists a state $\varphi(s)$ of M' such that $\varphi(s)$ 0-matches s. Clearly, $\varphi(s)$ is also a state of M''. We prove that the state $\varphi(s)$ of M'' 0-matches the state s of M. For any x_0, x_1, $\ldots \in X$, let $y_0y_1\ldots = \lambda(s, x_0x_1\ldots)$, where y_0, y_1, $\ldots \in Y$. We prove by reduction to absurdity a proposition: $\delta'(\varphi(s), y_0\ldots y_j) \neq s'$ holds for $j = -1, 0, 1, \ldots$ Suppose to the contrary that $\delta'(\varphi(s), y_0\ldots y_j) = s'$ holds for some $j \geqslant -1$. From $y_0y_1\ldots y_j = \lambda(s, x_0x_1\ldots x_j)$, for any $x \in X$ there exists $y_x \in Y$ such that $y_0y_1\ldots y_jy_x = \lambda(s, x_0x_1\ldots x_jx)$. Since the state $\varphi(s)$ of M' 0-matches the state s of M, we have $\lambda'(\varphi(s), y_0y_1\ldots y_jy_x) = x_0x_1\ldots x_jx$. It follows that $\lambda'(s', y_x) = \lambda'(\delta'(\varphi(s), y_0y_1\ldots y_j), y_x) = x$. Since x may take any element in X, we have $|\lambda'(s', Y)| = |X|$. This contradicts the assumption of the theorem. Thus the proposition holds. Since $\lambda'(s'', y) = \lambda''(s'', y)$ holds for any $s'' \in S' \setminus \{s'\}$ and $y \in Y$, using the above proposition, we have $\lambda'(\varphi(s), y_0y_1\ldots) = \lambda''(\varphi(s), y_0y_1\ldots)$. Since $\lambda'(\varphi(s), y_0y_1\ldots) = x_0x_1\ldots$, we have $\lambda''(\varphi(s), y_0y_1\ldots) = x_0x_1\ldots$ Thus the state $\varphi(s)$ of M'' 0-matches the state s of M. Therefore, M'' is a weak inverse with delay 0 of M. □

Theorem 5.2.1. *M is a feedforward invertible with delay 0 if and only if there exists $\mathcal{SIM}(M_a, f)$ such that $\mathcal{SIM}(M_a, f)$ is a weak inverse with delay*

0 of M and $|f(Y, y_{-1}, \ldots, y_{-c}, \lambda_a(t))| = |X|$ holds for any state t of M_a and any $y_{-1}, \ldots, y_{-c}$ in Y.

Proof. The *if* part is trivial. The *only if* part can be obtained by applying repeatedly Lemma 5.2.1. □

Let $M' = \langle Y, X, S', \delta', \lambda' \rangle$ be a c-order semi-input-memory finite automaton $\mathcal{SIM}(M_a, f)$, where $S' = Y^c \times S_a$, $M_a = \langle Y_a, S_a, \delta_a, \lambda_a \rangle$ is an autonomous finite automaton, and f is a single-valued mapping from $Y^{c+1} \times \lambda_a(S_a)$ to X.

From Corollary 5.1.1, if $|f(Y, y_{-1}, \ldots, y_{-c}, \lambda_a(t))| = |X|$ holds for any state t of M_a and any $y_{-1}, \ldots, y_{-c}$ in Y, then M' is a feedforward inverse with delay 0. Below we prove that the sufficient condition is also necessary in the case of $|X| = |Y|$.

Lemma 5.2.2. *Let M_a be cyclic, and $|X| = |Y|$. If there exist v in S_a and $y_0, \ldots, y_{c-1}$ in Y such that $|f(Y, y_{c-1}, \ldots, y_0, \lambda_a(v))| < |X|$, then M' is not a feedforward inverse with delay 0.*

Proof. Let $v \in S_a$ and $y_0, \ldots, y_{c-1} \in Y$. Suppose that $|f(Y, y_0, \ldots, y_{c-1}, \lambda_a(v))| < |X|$. Then $X \setminus f(Y, y_{c-1}, \ldots, y_0, \lambda_a(v)) \neq \varnothing$ holds and for any $x_0, \ldots, x_{c-1}$ in X, any x_c in $X \setminus f(Y, y_{c-1}, \ldots, y_0, \lambda_a(v))$ and any y_c in Y, $x_0 \ldots x_c y_0 \ldots y_c$ is not in $F^{(0)}_{f_v}$. Therefore, for any $x_0, \ldots, x_{c-1}$ in X, any x_c in $X \setminus f(Y, y_{c-1}, \ldots, y_0, \lambda_a(v))$ and any y_c in Y, $x_0 \ldots x_c y_0 \ldots y_c$ is not in $F^{(0)}_v$. Let $u \in S_a$ with $\delta_a(u) = v$. From the algorithm of $F^{(\tau)}_t, t \in S_a$, for any $x_{-1}, x_0, \ldots, x_{c-1}$ in X and any y_{-1} in Y, $x_{-1} x_0 \ldots x_{c-1} y_{-1} y_0 \ldots y_{c-1}$ is not in $F^{(0)}_u$.

We prove a proposition: for any j, $1 \leqslant j \leqslant c$, and any $p, q \in S_a$, if $\delta_a(p) = q$ and $x_{-j} \ldots x_{c-j} y_{-j} \ldots y_{c-j} \notin F^{(0)}_q$ holds for any $x_{-j}, \ldots, x_{c-j}$ in X and any $y_{-j}, \ldots, y_{-1}$ in Y, then $x_{-j-1} \ldots x_{c-j-1} y_{-j-1} \ldots y_{c-j-1} \notin F^{(0)}_p$ holds for any $x_{-j-1}, \ldots, x_{c-j-1}$ in X and any $y_{-j-1}, \ldots, y_{-1}$ in Y. In fact, from the algorithm of $F^{(\tau)}_t, t \in S_a$, it is sufficient to prove that for any $x_{-j}, \ldots, x_{c-j-1}$ in X and any $y_{-j}, \ldots, y_{-1}$ in Y, there exists x_{c-j} in X such that $x_{-j} \ldots x_{c-j}$ $y_{-j} \ldots y_{c-j-1} y \notin F^{(0)}_q$ holds for any y in Y. There are two cases to consider. In the case of $|f(Y, y_{c-j-1}, \ldots, y_{-j}, \lambda_a(q))| < |X|$, we choose an element in $X \setminus f(Y, y_{c-j-1}, \ldots, y_{-j}, \lambda_a(q))$ as x_{c-j}. Then $x_{-j} \ldots x_{c-j} y_{-j} \ldots y_{c-j-1} y \notin F^{(0)}_{f_q}$ holds for any y in Y. Therefore, $x_{-j} \ldots x_{c-j} y_{-j} \ldots y_{c-j-1} y \notin F^{(0)}_q$ holds for any y in Y. In the case of $|f(Y, y_{c-j-1}, \ldots, y_{-j}, \lambda_a(q))| = |X|$, from $|X| = |Y|$, it is easy to see that for any x in X there exists uniquely y in Y such that $f(y, y_{c-j-1}, \ldots, y_{-j}, \lambda_a(q)) = x$. Let $x_{c-j} = f(y_{c-j}, \ldots, y_{-j}, \lambda_a(q))$. Then for any y in $Y \setminus \{y_{c-j}\}$, we have $f(y, y_{c-j-1}, \ldots, y_{-j}, \lambda_a(q)) \neq x_{c-j}$. Thus $x_{-j} \ldots x_{c-j} y_{-j} \ldots y_{c-j-1} y \notin F^{(0)}_{f_q}$ holds for any y in $Y \setminus \{y_{c-j}\}$. Therefore, $x_{-j} \ldots x_{c-j} y_{-j} \ldots y_{c-j-1} y \notin F^{(0)}_q$ holds for any y in $Y \setminus \{y_{c-j}\}$. Since

$x_{-j} \ldots x_{c-j} y_{-j} \ldots y_{c-j} \notin F_q^{(0)}$ holds, $x_{-j} \ldots x_{c-j} y_{-j} \ldots y_{c-j-1} y \notin F_q^{(0)}$ holds for any y in Y.

We have proven in the first paragraph of the proof that $x_{-1}\ x_0\ \ldots\ x_{c-1}\ y_{-1}\ y_0\ \ldots\ y_{c-1} \notin F_u^{(0)}$ holds for any x_{-1}, x_0, ..., x_{c-1} in X and any y_{-1} in Y. Using the above proposition c times, we have that there exists $w \in S_a$ such that $x_{-c-1} \ldots x_{-1} y_{-c-1} \ldots y_{-1} \notin F_w^{(0)}$ holds for any $x_{-1}, \ldots, x_{-c-1}$ in X and any $y_{-1}, \ldots, y_{-c-1}$ in Y. It follows that $F_w^{(0)} = \varnothing$ holds. Since M_a is cyclic, from Lemma 5.1.1 (b), we have $F_t^{(0} = \varnothing$ holds for any $t \in S_a$. From Theorem 5.1.2, M' is not a feedforward inverse with delay 0. □

Theorem 5.2.2. *If $|X| = |Y|$, then M' is a feedforward inverse with delay 0 if and only if there exists a cycle C of M_a such that $|f(Y, y_{-1}, \ldots, y_{-c}, \lambda_a(u))| = |X|$ holds for any $u \in C$ and any $y_{-1}, \ldots, y_{-c} \in Y$.*

Proof. From Corollary 5.1.1, the *if* part holds.

To prove the *only if* part, suppose that M' is a feedforward inverse with delay 0. From Theorem 5.1.1, there exists a cycle C of M_a such that $\mathcal{SIM}(\bar{M}_a, f)$ is a feedforward inverse with delay 0, where $\bar{M}_a = \langle Y_a, C, \delta_a|_C, \lambda_a|_C \rangle$. From Lemma 5.2.2, $|f(Y, y_{-1}, \ldots, y_{-c}, \lambda_a(u))| = |X|$ holds for any $u \in C$ and any $y_{-1}, \ldots, y_{-c} \in Y$. □

Theorem 5.2.2 can be proven using mutual invertibility (Theorem 2.2.2) as follows.

It is easy to verify the following proposition.

Proposition 5.2.1. *For any finite automaton $M = \langle X, Y, S, \delta, \lambda \rangle$, M is weakly invertible with delay 0 if and only if for any state s of M, $\lambda_s|_X$ is an injection from X to Y.*

Let $M' = \langle Y, X, S', \delta', \lambda' \rangle$ be a c-order semi-input-memory finite automaton $\mathcal{SIM}(M_a, f)$ with $|X| = |Y|$. From Theorem 5.1.1, M' is a feedforward inverse with delay 0 if and only if there exists a finite subautomaton $\bar{M}_a$ of M_a such that $\bar{M}_a$ is cyclic and $\mathcal{SIM}(\bar{M}_a, f)$ is a feedforward inverse with delay 0. Since $\mathcal{SIM}(\bar{M}_a, f)$ is strongly connected, using Theorem 2.2.2, $\mathcal{SIM}(\bar{M}_a, f)$ is a feedforward inverse with delay 0 if and only if $\mathcal{SIM}(\bar{M}_a, f)$ is weakly invertible with delay 0. From $|X| = |Y|$, using Proposition 5.2.1, $\mathcal{SIM}(\bar{M}_a, f)$ is a feedforward inverse with delay 0 if and only if $|f(Y, y_{-1}, \ldots, y_{-c}, \lambda_a(u))| = |X|$ holds for any state u of $\bar{M}_a$ and any $y_{-1}, \ldots, y_{-c}$ in Y. Let C be the state alphabet of $\bar{M}_a$. Clearly, C is a cycle of M_a. Thus M' is a feedforward inverse with delay 0 if and only if $|f(Y, y_{-1}, \ldots, y_{-c}, \lambda_a(u))| = |X|$ holds for any $u \in C$ and any $y_{-1}, \ldots, y_{-c} \in Y$. This completes another proof of Theorem 5.2.2.

5.3 One Step Delay

Let $M = \langle X, Y, X^c \times S_a, \delta, \lambda \rangle$ be a c-order semi-input-memory finite automaton $\mathcal{SIM}(M_a, f)$, where $M_a = \langle Y_a, S_a, \delta_a, \lambda_a \rangle$ is an autonomous finite automaton, and f is a single-valued mapping from $X^{c+1} \times \lambda_a(S_a)$ to Y.

Lemma 5.3.1. *Let $M = \langle X, Y, S, \delta, \lambda \rangle$ be a semi-input-memory finite automaton $\mathcal{SIM}(M_a, f)$.*

(a) *M is strongly connected if and only if M_a is cyclic.*

(b) *If M_a is cyclic, $|X| = |Y|$ and M is weakly invertible with delay τ, then $|W^M_{\tau+1,s}| = w_{\tau+1,M}$ and $|W^M_{\tau,s}| = w_{\tau,M}$ hold for any $s \in S$.*

Proof. (a) It is trivial from definitions.

(b) From (a), M is strongly connected. From Theorem 2.1.3 (f), $|W^M_{\tau,s}| = w_{\tau,M}$ holds for any $s \in S$. Using Theorem 2.1.3 (c), it immediately follows that $|W^M_{\tau+1,s}| = w_{\tau+1,M}$ holds. □

Lemma 5.3.2. *Let $M = \langle X, Y, S, \delta, \lambda \rangle$ be weakly invertible with delay 1, and $|X| = |Y| = q$. Let $|W^M_{1,s}| = w_{1,M}$. Divide q successors of s into blocks such that $\delta(s, x_i)$ and $\delta(s, x_j)$ belong to the same block if and only if $\lambda(s, x_i) = \lambda(s, x_j)$. Then the number of blocks is $w_{1,M}$, each block consists of $q/w_{1,M}$ different states, the set of the outputs of length 1 on each state in a block has $w_{1,M}$ elements and such $q/w_{1,M}$ sets for the block constitute a partition of Y.*

Proof. Denote the successors $\delta(s, x)$, $x \in X$ of s by $s_1, \ldots, s_q$ (they are not necessary to be different from each other). From Theorem 2.1.3 (b), we have $|W^M_{1,s_j}| = w_{1,M}$, $j = 1, \ldots, q$. Divide $s_1, \ldots, s_q$ into blocks such that $s_i = \delta(s, x_i)$ and $s_j = \delta(s, x_j)$ belong to the same block if and only if $\lambda(s, x_i) = \lambda(s, x_j)$. Since $|W^M_{1,s}| = w_{1,M}$, the number of blocks is $w_{1,M}$. From Theorem 2.1.3 (e), $|I^M_{y,s}| = q/w_{1,M}$ holds for any $y \in W^M_{1,s}$. Thus each block consists of $q/w_{1,M}$ elements. Since M is weakly invertible with delay 1, the sets of the outputs of length 1 on any two elements in a block are disjoint. It follows that any two elements in a block are different states. For any block T, from $|W^M_{1,s_j}| = w_{1,M}$, $j = 1, \ldots, q$, we have $|W^M_{1,t}| = w_{1,M}$ for any $t \in T$. From $|T| = q/w_{1,M}$ and $W^M_{1,t} \cap W^M_{1,t'} = \varnothing$ for any different t and t' in T, it follows that the sets $W^M_{1,t}$, $t \in T$ constitute a partition of Y. □

We use $Y^{\langle w \rangle}$ to denote the set of all subsets with w elements of Y, that is, $Y^{\langle w \rangle} = \{T | T \subseteq Y, |T| = w\}$.

Let φ be a single-valued mapping from X to $Y^{\langle w \rangle}$ and $|X| = |Y|$. Let ψ be a uniform mapping from X to $\{1, \ldots, w\}$, that is, w is a divisor of $|X|$ and $|\psi^{-1}(j)| = |X|/w$ for any j, $1 \leqslant j \leqslant w$. If $\bigcup_{x \in \psi^{-1}(j)} \varphi(x) = Y$ holds for

any j, $1 \leqslant j \leqslant w$, ψ is called a *valid partition* of φ. We use P_φ to denote the set of all valid partitions of φ. Denote $n_\varphi = |P_\varphi|$; $n_\varphi = 0$ means no valid partition of φ.

We make an order of elements in Y. It leads to an ordering of elements in each subset of Y. We use $m(j, T)$ to denote the j-th element of T in the ordering.

Lemma 5.3.3. *Let $M = \langle X, Y, S, \delta, \lambda\rangle$ be a c-order semi-input-memory finite automaton $\mathcal{SIM}(M_a, f)$, and M_a be strongly cyclic. Let M be weakly invertible with delay 1, and $|X| = |Y|$. Then f can be expressed as*

$$f(x_0, x_{-1}, \ldots, x_{-c}, s_a) = m(\psi_{x_{-c},\ldots,x_{-1},s_a}(x_0), \varphi_{x_{-c},\ldots,x_{-2},s_a}(x_{-1})),$$

where $\psi_{x_{-c},\ldots,x_{-1},s_a} \in P_{\varphi_{x_{-c+1},\ldots,x_{-1},\delta_a(s_a)}}$, and $\varphi_{x_{-c},\ldots,x_{-2},s_a}$ is a single-valued mapping from X to $Y^{\langle w_{1,M}\rangle}$.

Proof. Define $\varphi_{x_{-c},\ldots,x_{-2},s_a}(x_{-1}) = \lambda(\langle x_{-1}, \ldots, x_{-c}, s_a\rangle, X)$. From Lemma 5.3.1 (a), M is strongly connected. From Lemma 5.3.1 (b), we have $|W^M_{1,s}| = w_{1,M}$ for any $s \in S$. Thus for any $x_{-c}, \ldots, x_{-2} \in X$ and $s_a \in S_a$, $\varphi_{x_{-c},\ldots,x_{-2},s_a}$ is a single-valued mapping from X to $Y^{\langle w_{1,M}\rangle}$. For any state $s = \langle x_{-1}, \ldots, x_{-c}, s_a\rangle$ of M, define $\psi_{x_{-c},\ldots,x_{-1},s_a}(x_0) = j$, where $\lambda(s, x_0)$ is the j-th element of $\lambda(s, X)$. Since $|\lambda(s, X)| = |W^M_{1,s}| = w_{1,M}$, $\psi_{x_{-c},\ldots,x_{-1},s_a}$ is a single-valued mapping from X to $\{1, \ldots, w_{1,M}\}$. Noticing that $\psi^{-1}_{x_{-c},\ldots,x_{-1},s_a}(j) = I^M_{y_j,s}$ when the j-th element of $W^M_{1,s}$ is y_j, from Theorem 2.1.3 (g), we have $|\psi^{-1}_{x_{-c},\ldots,x_{-1},s_a}(j)| = |I^M_{y_j,s}| = q/w_{1,M}$. Therefore, $\psi_{x_{-c},\ldots,x_{-1},s_a}$ is uniform. From the definitions, it is easy to verify that

$$f(x_0, x_{-1}, \ldots, x_{-c}, s_a) = m(\psi_{x_{-c},\ldots,x_{-1},s_a}(x_0), \varphi_{x_{-c},\ldots,x_{-2},s_a}(x_{-1})).$$

To prove $\psi_{x_{-c},\ldots,x_{-1},s_a} \in P_{\varphi_{x_{-c+1},\ldots,x_{-1},\delta_a(s_a)}}$, take arbitrarily an integer j, $1 \leqslant j \leqslant w_{1,M}$. From the definition, $x \in \psi^{-1}_{x_{-c},\ldots,x_{-1},s_a}(j)$ if and only if $\lambda(s, x) = y_j$, where $s = \langle x_{-1}, \ldots, x_{-c}, s_a\rangle$, y_j is the j-th element of $W^M_{1,s}$. Since M is weakly invertible with delay 1, $\lambda(\langle x_0, x_{-1}, \ldots, x_{-c+1}, \delta_a(s_a)\rangle, X)$ and $\lambda(\langle x'_0, x_{-1}, \ldots, x_{-c+1}, \delta_a(s_a)\rangle, X)$ are disjoint if $\lambda(s, x_0) = \lambda(s, x'_0) = y_j$ and $x_0 \neq x'_0$. That is, $\varphi_{x_{-c+1},\ldots,x_{-1},\delta_a(s_a)}(x_0)$ and $\varphi_{x_{-c+1},\ldots,x_{-1},\delta_a(s_a)}(x'_0)$ are disjoint, if $x_0, x'_0 \in \psi^{-1}_{x_{-c},\ldots,x_{-1},s_a}(j)$ and $x_0 \neq x'_0$. Since $|\psi^{-1}_{x_{-c},\ldots,x_{-1},s_a}(j)| = |I^M_{y_j,s}| = q/w_{1,M}$, $|\varphi_{x_{-c+1},\ldots,x_{-1},\delta_a(s_a)}(x)| = w_{1,M}$ and $|Y| = q$, we have

$$\bigcup_{x \in \psi^{-1}_{x_{-c},\ldots,x_{-1},s_a}(j)} \varphi_{x_{-c+1},\ldots,x_{-1},\delta_a(s_a)}(x) = Y.$$

Thus $\psi_{x_{-c},\ldots,x_{-1},s_a}$ is a valid partition of $\varphi_{x_{-c+1},\ldots,x_{-1},\delta_a(s_a)}$. □

Lemma 5.3.4. *Let $M = \langle X, Y, S, \delta, \lambda\rangle$ be a c-order semi-input-memory finite automaton $\mathcal{SIM}(M_a, f)$, M_a be strongly cyclic, and $|X| = |Y|$. If f can be expressed as*

$$f(x_0, x_{-1}, \ldots, x_{-c}, s_a) = m(\psi_{x_{-c},\ldots,x_{-1},s_a}(x_0), \varphi_{x_{-c},\ldots,x_{-2},s_a}(x_{-1})),$$

where $\psi_{x_{-c},\ldots,x_{-1},s_a} \in P_{\varphi_{x_{-c+1},\ldots,x_{-1},\delta_a(s_a)}}$, and $\varphi_{x_{-c},\ldots,x_{-2},s_a}$ is a single-valued mapping from X to $Y^{\langle w\rangle}$, then M is weakly invertible with delay 1 and $w_{1,M} = w$.

Proof. For any state $s = \langle x_{-1}, \ldots, x_{-c}, s_a\rangle$ of M, any $x_0, x_1, x_0', x_1' \in X$, let $y_0y_1 = \lambda(s, x_0x_1)$ and $y_0'y_1' = \lambda(s, x_0'x_1')$, where $y_0, y_1, y_0', y_1' \in Y$. Suppose that $y_0y_1 = y_0'y_1'$. To prove that M is weakly invertible with delay 1, it is sufficient to prove $x_0 = x_0'$. Suppose to the contrary that $x_0 \neq x_0'$. Since $y_0 = y_0'$, we have $f(x_0, x_{-1}, \ldots, x_{-c}, s_a) = f(x_0', x_{-1}, \ldots, x_{-c}, s_a)$. Therefore, the values of $\psi_{x_{-c},\ldots,x_{-1},s_a}(x_0)$ and $\psi_{x_{-c},\ldots,x_{-1},s_a}(x_0')$ are the same; we denote the value by j. On the other hand, since $\psi_{x_{-c},\ldots,x_{-1},s_a}$ is a valid partition of $\varphi_{x_{-c+1},\ldots,x_{-1},\delta_a(s_a)}$, we have

$$\bigcup_{x\in\psi^{-1}_{x_{-c},\ldots,x_{-1},s_a}(j)} \varphi_{x_{-c+1},\ldots,x_{-1},\delta_a(s_a)}(x) = Y.$$

From $|\psi^{-1}_{x_{-c},\ldots,x_{-1},s_a}(j)| = q/w$, $|\varphi_{x_{-c+1},\ldots,x_{-1},\delta_a(s_a)}(x)| = w$ and $|Y| = q$, it follows that $\varphi_{x_{-c+1},\ldots,x_{-1},\delta_a(s_a)}(x)$, x ranging over elements in $\psi^{-1}_{x_{-c},\ldots,x_{-1},s_a}(j)$, constitute a partition of Y. Since $x_0 \neq x_0'$, $\varphi_{x_{-c+1},\ldots,x_{-1},\delta_a(s_a)}(x_0)$ and $\varphi_{x_{-c+1},\ldots,x_{-1},\delta_a(s_a)}(x_0')$ are disjoint. It follows that

$$\begin{aligned} &m(\psi_{x_{-c+1},\ldots,x_{-1},x_0,\delta_a(s_a)}(x_1), \varphi_{x_{-c+1},\ldots,x_{-1},\delta_a(s_a)}(x_0)) \\ &\quad \neq m(\psi_{x_{-c+1},\ldots,x_{-1},x_0',\delta_a(s_a)}(x_1'), \varphi_{x_{-c+1},\ldots,x_{-1},\delta_a(s_a)}(x_0')). \end{aligned}$$

Therefore,

$$f(x_1, x_0, x_{-1}, \ldots, x_{-c+1}, \delta_a(s_a)) \neq f(x_1', x_0', x_{-1}, \ldots, x_{-c+1}, \delta_a(s_a)),$$

that is, $y_1 \neq y_1'$. This contradicts $y_0y_1 = y_0'y_1'$. Thus the hypothesis $x_0 \neq x_0'$ does not hold. We conclude that $x_0 = x_0'$.

From the definition of the valid partition, it is easy to see that for any state $s = \langle x_{-1}, \ldots, x_{-c}, s_a\rangle$ of M, the number of the elements in $f(X, x_{-1}, \ldots, x_{-c}, s_a)$ is w, that is, $|\lambda(s, X)| = w$. From Lemma 5.3.1 (a), M is strongly connected. Using Lemma 5.3.1 (b), we have $w_{1,M} = |\lambda(s, X)| = w$. □

Theorem 5.3.1. *Let $M = \langle X, Y, S, \delta, \lambda\rangle$ be a c-order semi-input-memory finite automaton $\mathcal{SIM}(M_a, f)$, $M_a = \langle Y_a, S_a, \delta_a, \lambda_a\rangle$ be strongly cyclic, and*

$|X| = |Y|$. *Then M is weakly invertible with delay* 1 *if and only if there exist single-valued mappings $\varphi_{x_{-c},\ldots,x_{-2},s_a}$ from X to $Y^{\langle w\rangle}$, $x_{-c},\ldots,x_{-2} \in X$, $s_a \in S_a$ such that* (a) *for any $x_{-c},\ldots,x_{-2} \in X$ and any $s_a \in S_a$, $P_{\varphi_{x_{-c},\ldots,x_{-2},s_a}}$ is non-empty, and* (b) *for any $x_{-c},\ldots,x_{-1} \in X$ and any $s_a \in S_a$, there exists $\psi_{x_{-c},\ldots,x_{-1},s_a} \in P_{\varphi_{x_{-c+1},\ldots,x_{-1},\delta_a(s_a)}}$ such that*

$$f(x_0, x_{-1}, \ldots, x_{-c}, s_a) = m(\psi_{x_{-c},\ldots,x_{-1},s_a}(x_0), \varphi_{x_{-c},\ldots,x_{-2},s_a}(x_{-1})) \quad (5.1)$$

holds for any $x_0 \in X$. Moreover, whenever the above condition holds, w in the condition is equal to $w_{1,M}$.

Proof. only if : from Lemma 5.3.3.
if and $w = w_{1,M}$: from Lemma 5.3.4. □

Remark In the case of $w_{1,M} = 1$, in the condition (b), we have $\psi_{x_{-c},\ldots,x_{-1},s_a}(x_0) = 1$. Therefore, the equation (5.1) can be simplified into

$$f(x_0, x_{-1}, \ldots, x_{-c}, s_a) = \varphi_{x_{-c},\ldots,x_{-2},s_a}(x_{-1}).$$

Meanwhile, the condition (a) is equivalent to the condition: for any $x_{-c},\ldots,x_{-2} \in X$ and any $s_a \in S_a$, $\varphi_{x_{-c},\ldots,x_{-2},s_a}$ is a surjection (or an injection, or a bijection).

In the case of $w_{1,M} = |X|$, $\varphi_{x_{-c},\ldots,x_{-2},s_a}(x_{-1}) = Y$ and $\psi_{x_{-c},\ldots,x_{-1},s_a}$ is bijective. It follows that the right-side of the equation (5.1) as a function of x_0 defines a bijection from X to Y. In this case, the condition in Theorem 5.3.1 is equivalent to the condition: for any $x_{-c},\ldots,x_{-1} \in X$ and any $s_a \in S_a$, $f(x_0, x_{-1}, \ldots, x_{-c}, s_a)$ as a function of x_0 is a bijection from X to Y. Therefore, this case degenerates to the case of weakly invertible with delay 0.

To sum up, for the cases of $w_{1,M} = 1, |X|$, the expressions of semi-input-memory finite automata with strongly cyclic autonomous finite automata are very succinct, and their synthesization is clear. Below we present synthesizing method for the case where $w_{1,M}$ is a proper divisor of $|X|$ other than 1.

Synthesizing method Given $|X| = |Y| = q$, $c \geqslant 1$ and a strongly cyclic autonomous finite automaton $M_a = \langle Y_a, S_a, \delta_a, \lambda_a\rangle$. Suppose that $w|q$ and $w \neq 1, q$. Find f, a single-valued mapping from $X^{c+1} \times S_a$ to Y, such that $\mathcal{SIM}(M_a, f)$ is weakly invertible with delay 1.

Step 1. For any $x_{-c},\ldots,x_{-2} \in X$ and any $s_a \in S_a$, choose arbitrarily a single-valued mapping $\varphi_{x_{-c},\ldots,x_{-2},s_a}$ from X to $Y^{\langle w\rangle}$, of which a valid partition is existent, that is, there exists a single-valued mapping ψ from X to $\{1,\ldots,w\}$ such that for any j, $1 \leqslant j \leqslant w$, $|\psi^{-1}(j)| = |X|/w$ and $\bigcup_{x\in\psi^{-1}(j)} \varphi_{x_{-c},\ldots,x_{-2},s_a}(x) = Y$. Denote the set of all valid partitions of $\varphi_{x_{-c},\ldots,x_{-2},s_a}$ by $P_{\varphi_{x_{-c},\ldots,x_{-2},s_a}}$.

Step 2. For any $x_{-c}, \ldots, x_{-1} \in X$ and any $s_a \in S_a$, choose arbitrarily a valid partition, say $\psi_{x_{-c},\ldots,x_{-1},s_a}$, in $P_{\varphi_{x_{-c+1},\ldots,x_{-1},\delta_a(s_a)}}$.

Step 3. Define

$$f(x_0, x_{-1}, \ldots, x_{-c}, s_a) = m(\psi_{x_{-c},\ldots,x_{-1},s_a}(x_0), \varphi_{x_{-c},\ldots,x_{-2},s_a}(x_{-1}))$$

for $x_{-c}, \ldots, x_1, x_0 \in X$, $s_a \in S_a$, where $m(j, T)$ denotes the j-th element in T. (The order of elements of $T \subseteq Y$ is naturally induced by a given order of elements in Y.)

According to Theorem 5.3.1, each $\mathcal{SIM}(M_a, f)$ obtained by the above synthesizing method is weakly invertible with delay 1, and all c-order semi-input-memory finite automata with strongly cyclic autonomous finite automata which are weakly invertible with delay 1 can be found by the above synthesizing method.

Example 5.3.1. Let $X = Y = \{0, 1, 2, 3\}$. Take $c \geqslant 1$, $w = 2$. Suppose that $M_a = \langle Y_a, S_a, \delta_a, \lambda_a \rangle$ is a strongly cyclic autonomous finite automaton. Find a single-valued mapping from $X^{c+1} \times S_a$ to Y, say f, such that $\mathcal{SIM}(M_a, f)$ is weakly invertible with delay 1.

According to the synthesizing method mentioned above, for any $x_{-c}, \ldots, x_{-2} \in X$ and any $s_a \in S_a$, we define $\varphi_{x_{-c},\ldots,x_{-2},s_a}(x) = \varphi(x)$, where $\varphi(x) = \{0, 1\}$ if $x = 0, 2$, $\varphi(x) = \{2, 3\}$ if $x = 1, 3$. Define ψ_i as in Table 5.3.1.

Table 5.3.1 Definition of ψ_i

x	$\psi_1(x)$	$\psi_2(x)$	$\psi_3(x)$	$\psi_4(x)$
0	1	1	2	2
1	1	2	2	1
2	2	2	1	1
3	2	1	1	2

It is easy to verify that ψ_1, ψ_2, ψ_3 and ψ_4 are all valid partitions of φ. For any $x_{-c}, \ldots, x_{-1} \in X$ and any $s_a \in S_a$, take $\psi_{x_{-c},\ldots,x_{-1},s_a} = \psi_1$ if $x_{-c} = 0$, or ψ_2 if $x_{-c} \neq 0$. Take an order for elements in Y: the first to the fourth elements are 0, 1, 2, 3. We then have

$$f(x_0, x_{-1}, \ldots, x_{-c}, s_a) = \begin{cases} m(\psi_1(x_0), \varphi(x_{-1})), & \text{if } x_{-c} = 0, \\ m(\psi_2(x_0), \varphi(x_{-1})), & \text{if } x_{-c} \neq 0. \end{cases}$$

It follows that

$$f(x_0, x_{-1}, \ldots, x_{-c}, s_a) = \begin{cases} 0, & \text{if } x_{-c} = 0,\ x_{-1} = 0, 2,\ x_0 = 0, 1, \\ 1, & \text{if } x_{-c} = 0,\ x_{-1} = 0, 2,\ x_0 = 2, 3, \\ 2, & \text{if } x_{-c} = 0,\ x_{-1} = 1, 3,\ x_0 = 0, 1, \\ 3, & \text{if } x_{-c} = 0,\ x_{-1} = 1, 3,\ x_0 = 2, 3, \\ 0, & \text{if } x_{-c} \neq 0,\ x_{-1} = 0, 2,\ x_0 = 0, 3, \\ 1, & \text{if } x_{-c} \neq 0,\ x_{-1} = 0, 2,\ x_0 = 1, 2, \\ 2, & \text{if } x_{-c} \neq 0,\ x_{-1} = 1, 3,\ x_0 = 0, 3, \\ 3, & \text{if } x_{-c} \neq 0,\ x_{-1} = 1, 3,\ x_0 = 1, 2. \end{cases}$$

Theorem 5.3.2. *Let $M = \langle X, Y, S, \delta, \lambda\rangle$ be a c-order semi-input-memory finite automaton $\mathcal{SIM}(M_a, f)$, $M_a = \langle Y_a, S_a, \delta_a, \lambda_a\rangle$ be strongly cyclic, and $|X| = |Y|$. Then M is a feedforward inverse with delay 1 if and only if there exist single-valued mappings $\varphi_{x_{-c},\ldots,x_{-2},s_a}$, $x_{-c}, \ldots, x_{-2} \in X$, $s_a \in S_a$ from X to $Y^{\langle w\rangle}$ such that* (a) *for any $x_{-c}, \ldots, x_{-2} \in X$ and any $s_a \in S_a$, $P_{\varphi_{x_{-c},\ldots,x_{-2},s_a}}$ is non-empty, and* (b) *for any $x_{-c}, \ldots, x_{-1} \in X$ and any $s_a \in S_a$, there exists $\psi_{x_{-c},\ldots,x_{-1},s_a} \in P_{\varphi_{x_{-c+1},\ldots,x_{-1},\delta_a(s_a)}}$ such that* (5.1) *holds for any $x_0 \in X$. Moreover, whenever the above condition holds, w in the condition is equal to $w_{1,M}$.*

Proof. Since M, i.e. $\mathcal{SIM}(M_a, f)$, is a semi-input-memory finite automaton and M_a is strongly cyclic, M is strongly connected. Using Theorem 2.2.2, M is a feedforward inverse with delay 1 if and only if M is weakly invertible with delay 1. From Theorem 5.3.1, the theorem holds. □

It should be pointed out that any $\mathcal{SIM}(M_a, f)$ can be expressed as $\mathcal{SIM}(\bar{M}_a, \bar{f})$ such that the output function of $\bar{M}_a$ is the identity function. In fact, let $M_a = \langle Y_a, S_a, \delta_a, \lambda_a\rangle$. Take $\bar{M}_a = \langle S_a, S_a, \delta_a, \bar{\lambda}_a\rangle$,where $\bar{\lambda}_a(s_a) = s_a$ for any $s_a \in S_a$. Take $\bar{f}(x_0, x_{-1}, \ldots, x_{-c}, s_a) = f(x_0, x_{-1}, \ldots, x_{-c}, \lambda_a(s_a))$. Then $\mathcal{SIM}(\bar{M}_a, \bar{f}) = \mathcal{SIM}(\bar{M}_a, \bar{f})$.

5.4 Two Step Delay

For any finite automaton $M = \langle X, Y, S, \delta, \lambda\rangle$ and any state s of M, if for any $\alpha = x_0 \ldots x_l$ of length $l + 1$ in X^*, x_0 can be uniquely determined by s and $\lambda(s, \alpha)$, s is called a $\leqslant l$-*step state*; for $l > 0$, if s is a $\leqslant l$-step state and not a $\leqslant (l-1)$-step state, s is called an l-*step state*; if s is a $\leqslant 0$-step state, s is called a 0-*step state*. Clearly, if M is weakly invertible with delay τ and s is a state of M, then s is an l-step state for some l, $0 \leqslant l \leqslant \tau$.

Lemma 5.4.1. *Let $M = \langle X, Y, S, \delta, \lambda\rangle$ be a finite automaton, and $|X| = 2$. Let $|W^M_{2,s}| = 2$. If s is a 0-step state and s' a successor state of s, then s' is not a 0-step state.*

Proof. Assume that s is a 0-step state and s' a successor state of s. We prove by reduction to absurdity that s' is not a 0-step state. Suppose to the contrary that s' is a 0-step state. Since s and s' are 0-step states, we have $|\lambda(s, X)| = |\lambda(s', X)| = 2$. Since s' is a successor state of s, it is easy to see that $|W^M_{2,s}| \geqslant 3$. This contradicts $|W^M_{2,s}| = 2$. We conclude that s' is not a 0-step state. □

Lemma 5.4.2. *Let $M = \langle X, Y, S, \delta, \lambda\rangle$ be a finite automaton, and $|X| = 2$.*

(a) *If s in S is a 1-step state and s' a successor state of s, then s' is not a 0-step state.*

(b) *Let s' and s'' be two different successor states of $s \in S$. If s, s' and s'' are not 0-step states and $|W^M_{2,s}| = 2$, then $|\lambda(s', X)| = |\lambda(s'', X)| = 1$ and $\lambda(s', X) \neq \lambda(s'', X)$, therefore, s is a 1-step state.*

Proof. (a) Assume that s in S is a 1-step state and s' a successor state of s. We prove by reduction to absurdity that s' is not a 0-step state. Suppose to the contrary that s' is a 0-step state. Since s is a 1-step state and s' a 0-step state, we have $|\lambda(s, X)| = 1$ and $|\lambda(s', X)| = 2$. Since s' is a successor state of s, it is easy to see that there exist $x_0, x'_0, x_1, x'_1 \in X$ such that $x_0 \neq x'_0$ and $\lambda(s, x_0x_1) = \lambda(s, x'_0x'_1)$. This contradicts that s is a 1-step state. We conclude that s' is not a 0-step state.

(b) Since s, s' and s'' are not 0-step states, we have $|\lambda(s, X)| = |\lambda(s', X)| = |\lambda(s'', X)| = 1$. From $|W^M_{2,s}| = 2$, it follows that $\lambda(s', X) \neq \lambda(s'', X)$. Therefore, s is a 1-step state. □

Lemma 5.4.3. *Let $M = \langle X, Y, S, \delta, \lambda\rangle$ be weakly invertible with delay 2, and $w_{2,M} = |X| = |Y| = 2$. Let $S_0 = \{s | s \in S, |W^M_{2,s}| = w_{2,M}\}$.*

(a) *If s' and s'' are two different successor states of $s \in S_0$, then s' is a 0-step state if and only if s'' is a 0-step state.*

(b) *If s in S_0 is a 2-step state and s' a successor state of s, then s' is a 0-step state.*

Proof. (a) Assume that s' and s'' are two different successor states of $s \in S_0$. We prove by reduction to absurdity that s' is a 0-step state if and only if s'' is a 0-step state. Suppose to the contrary that one state in $\{s', s''\}$ is a 0-step state and the other is not. Without loss of generality, suppose that s' is a 0-step state and s'' is not. Then we have $|\lambda(s', X)| = 2$ and $|\lambda(s'', X)| = 1$. Since s' and s'' are two different successor states of s and $|w^M_{2,s}| = 2$, we have $|\lambda(s, X)| = 1$. (Otherwise, we obtain $|w^M_{2,s}| = 3$, which contradicts $s \in S_0$.) Since s is in S_0, from Theorem 2.1.3 (b), s'' is in S_0. Thus we have $\lambda(s_3, X) \cup \lambda(s_4, X) = Y$, where s_3 and s_4 are two different successors of s''. Let x_1 in X satisfy $\lambda(s', x_1) = \lambda(s'', x_1)$. Since s' is a 0-step state, such an x_1 is existent. Denote $s_1 = \delta(s', x_1)$. From $\lambda(s_3, X) \cup \lambda(s_4, X) = Y$, we

can find x_2 and x_2' in X and s''' in $\{s_3, s_4\}$ such that $\lambda(s_1, x_2) = \lambda(s''', x_2')$. Let x_0, x_0', x_1' in X satisfy $s' = \delta(s, x_0)$, $s'' = \delta(s, x_0')$ and $s''' = \delta(s'', x_1')$. Then $x_0 \neq x_0'$ and $\lambda(s, x_0x_1x_2) = \lambda(s, x_0'x_1'x_2')$. This contradicts that M is weakly invertible with delay 2. We conclude that s' is a 0-step state if and only if s'' is a 0-step state.

(b) Assume that s in S_0 is a 2-step state and s' a successor state of s. We prove by reduction to absurdity that s' is a 0-step state. Suppose to the contrary that s' is not a 0-step state. Then $|\lambda(s', X)| = 1$. Since s is a 2-step state, we have $|\lambda(s, X)| = 1$. Let s'' be a successor state of s other than s'. From (a), s'' is not a 0-step state. Thus $|\lambda(s'', X)| = 1$. From $s \in S_0$, we obtain $\lambda(s', X) \cap \lambda(s'', X) = \varnothing$. It immediately follows that s is a 1-step state. This contradicts that s is a 2-step state. We conclude that s' is a 0-step state. □

Lemma 5.4.4. *Let $M = \langle X, Y, S, \delta, \lambda\rangle$ be a weakly invertible finite automaton with delay 2. Let $w_{2,M} = |X| = |Y| = 2$ and $S_0 = \{s | s \in S, |W_{2,s}^M| = w_{2,M}\}$.*

(a) *Let s and s' be two different successor states of $s_{-1} \in S_0$. Let s_1 and s_2 be two different successor states of s, and s_1' and s_2' be two different successor states of s'. Assume that s_1 and s_2 are not 0-step states and s is a 0-step state. Then s_1' and s_2' are not 0-step states and s' is a 0-step state. Moreover, $\lambda(s_1, X) = \lambda(s_2, X)$ if and only if $\lambda(s_1', X) = \lambda(s_2', X)$; if $\lambda(s_1, X) = \lambda(s_2, X)$, then $\lambda(s_1', X) = \lambda(s_2', X) \neq \lambda(s_1, X)$; and if $\lambda(s_1, X) \neq \lambda(s_2, X)$, then $\lambda(s_1, X) = \lambda(s_1', X)$ if and only if $\lambda(s, x_1) \neq \lambda(s', x_1')$, where x_1 and x_1' in X satisfy $\delta(s, x_1) = s_1$ and $\delta(s', x_1') = s_1'$.*

(b) *Let s_1 and s_2 be two different successor states of s, and s_1' and s_2' be two different successor states of s', where s and s' are two different successor states of a state in S_0. If $\lambda(s_1, X) = \lambda(s_2, X)$ and $|\lambda(s_1, X)| = 1$, then $|\lambda(s_1', X)| = 1$ and $\lambda(s_1', X) = \lambda(s_2', X) \neq \lambda(s_1, X)$.*

Proof. (a) Since s is a 0-step state, using Lemma 5.4.3 (a), s' is a 0-step state. Since s is a 0-step state, using Lemma 5.4.1 and Lemma 5.4.2 (a), s_{-1} is a 2-step state. We prove that s_1' and s_2' are not 0-step states. For any $i \in \{1, 2\}$, since s' a successor state of s_{-1} and s_i' a successor state of s', there exist x_0', $x_1' \in X$ such that $\delta(s_{-1}, x_0') = s'$ and $\delta(s_{-1}, x_0'x_1') = s_i'$. Since s_{-1} is a 2-step state and s a 0-step state, there exist x_0, $x_1 \in X$ such that $x_0' \neq x_0$ and $\lambda(s_{-1}, x_0x_1) = \lambda(s_{-1}, x_0'x_1')$. Clearly, $\delta(s_{-1}, x_0x_1) = s_j$ for some $j \in \{1, 2\}$. Since M is weakly invertible with delay 2, $\lambda(s_j, X) \cap \lambda(s_i', X) = \varnothing$. It immediately follows that $|\lambda(s_i', X)| = 1$, that is, s_i' is not a 0-step state.

Denote $\lambda(s_i, X) = \{e_i\}$ and $\lambda(s_i', X) = \{e_i'\}$ for $i = 1, 2$. Suppose $e_1 = e_2$. We prove by reduction to absurdity that $e_1' = e_2' \neq e_1$, that is, $e_i' \neq e_1$ for $i = 1, 2$. Suppose to the contrary that $e_i' = e_1$ for some $i \in \{1, 2\}$. Since s_{-1} is

a 2-step state and s is a 0-step state, there exist x_0, x_1, x_0', $x_1' \in X$ such that $x_0' \neq x_0$, $\lambda(s_{-1}, x_0x_1) = \lambda(s_{-1}, x_0'x_1')$, $\delta(s_{-1}, x_0) = s$ and $\delta(s_{-1}, x_0'x_1') = s_i'$. Let $\delta(s_{-1}, x_0x_1) = s_j$. Noticing $e_1 = e_2$, we have $e_j = e_i'$, that is, $\lambda(s_j, x_2) = \lambda(s_i', x_2)$ for any $x_2 \in X$. It follows that $\lambda(s_{-1}, x_0x_1x_2) = \lambda(s_{-1}, x_0'x_1'x_2)$. This contradicts that s_{-1} is a 2-step state. We conclude that $e_1' = e_2' \neq e_1$. Using this result, we obtain that $e_1 = e_2$ implies $e_1' = e_2'$. From symmetry, $e_1' = e_2'$ implies $e_1 = e_2$. Therefore, $e_1 = e_2$ if and only if $e_1' = e_2'$.

Suppose $e_1 \neq e_2$. We then have $e_1' \neq e_2'$. It follows that there exists $i \in \{1, 2\}$ such that $e_1 = e_i'$. Let x_1, x_1' and $x_2' \in X$ satisfy $\delta(s, x_1) = s_1$ and $\delta(s', x_j') = s_j'$, $j = 1, 2$. We prove by reduction to absurdity that $\lambda(s, x_1) \neq \lambda(s', x_i')$. Suppose to the contrary that $\lambda(s, x_1) = \lambda(s', x_i')$. Since $e_1 = e_i'$, for any $x_2 \in X$ we have $\lambda(s_1, x_2) = \lambda(s_i', x_2)$. It follows that $\lambda(s, x_1x_2) = \lambda(s', x_i'x_2)$. Since s and s' be two different successor states of s_{-1} and s_{-1} is a 2-step state, we obtain $\lambda(s_{-1}, x_0x_1x_2) = \lambda(s_{-1}, x_0'x_i'x_2)$, where x_0 and x_0' are different elements in X satisfying $\delta(s_{-1}, x_0) = s$ and $\delta(s_{-1}, x_0') = s'$. This contradicts that s_{-1} is a 2-step state. We conclude that $\lambda(s, x_1) \neq \lambda(s', x_i')$. In the case of $e_1 = e_1'$, we have $i = 1$. This yields $\lambda(s, x_1) \neq \lambda(s', x_1')$. In the case of $e_1 \neq e_1'$, we have $i = 2$. This yields $\lambda(s, x_1) \neq \lambda(s', x_2')$. Since s' is a 0-step state, we have $\lambda(s', x_1') \neq \lambda(s', x_2')$. Thus $\lambda(s, x_1) = \lambda(s', x_1')$. Therefore, $e_1 = e_1'$ if and only if $\lambda(s, x_1) \neq \lambda(s', x_1')$.

(b) Suppose that $\lambda(s_1, X) = \lambda(s_2, X)$ and $|\lambda(s_1, X)| = 1$. Then s_1 and s_2 are not 0-step states. Since s is a successor state of a state in S_0, from Theorem 2.1.3 (b), s is a state in S_0, that is, $|W_{2,s}^M| = 2$. From Lemma 5.4.2 (b), s is a 0-step state. From (a), s_1' is not a 0-step state. Thus $|\lambda(s_1', X)| = 1$. Since $\lambda(s_1, X) = \lambda(s_2, X)$, using (a), we obtain $\lambda(s_1', X) = \lambda(s_2', X) \neq \lambda(s_1, X)$. □

Lemma 5.4.5. *Let $M = \langle X, Y, S, \delta, \lambda\rangle$ be a c-order semi-input-memory finite automaton $\mathcal{SIM}(M_a, f)$, and $M_a = \langle Y_a, S_a, \delta_a, \lambda_a\rangle$ be strongly cyclic. If $c \geqslant 2 = w_{2,M}$, $X = Y = \{0, 1\}$ and M is weakly invertible with delay 2, then there exist single-valued mappings h_0 from $X^{c-1} \times S_a$ to $\{0, 1\}$, h_1 from $X^{c-2} \times S_a$ to $\{0, 1\}$, f_0 from $X^c \times S_a$ to Y, f_1 from $X^{c-1} \times S_a$ to Y, and f_2 from $X^{c-2} \times S_a$ to Y, such that*

$$\begin{aligned} &h_0(x_{-2}, \ldots, x_{-c}, s_a) = 0 \to h_0(x_{-1}, \ldots, x_{-c+1}, \delta_a(s_a)) = 1, \\ &h_1(x_{-3}, \ldots, x_{-c}, s_a) = 1 \to h_0(x_{-3}, \ldots, x_{-c-1}, \delta_a^{-1}(s_a)) = 0, \end{aligned} \quad (5.2)$$

and

$$
\begin{aligned}
&f(x_0, \ldots, x_{-c}, s_a) \\
&\quad = \begin{cases}
f_2(x_{-3}, \ldots, x_{-c}, s_a) \oplus x_{-2}, \\
\qquad \text{if } h_0(x_{-2}, \ldots, x_{-c}, s_a) = 1 \ \&\ h_1(x_{-3}, \ldots, x_{-c}, s_a) = 1, \\
f_1(x_{-2}, \ldots, x_{-c}, s_a) \oplus x_{-1}, \\
\qquad \text{if } h_0(x_{-2}, \ldots, x_{-c}, s_a) = 1 \ \&\ h_1(x_{-3}, \ldots, x_{-c}, s_a) = 0, \\
f_0(x_{-1}, \ldots, x_{-c}, s_a) \oplus x_0, \\
\qquad \text{if } h_0(x_{-2}, \ldots, x_{-c}, s_a) = 0 \ \&\ h_1(x_{-2}, \ldots, x_{-c+1}, \delta_a(s_a)) = 1, \\
f_1(x_{-2}, \ldots, x_{-c}, s_a) \oplus x_0 \oplus x_{-1} \oplus x_{-1} h_2(x_{-2}, \ldots, x_{-c+1}, \delta_a(s_a)), \\
\qquad \text{if } h_0(x_{-2}, \ldots, x_{-c}, s_a) = 0 \ \&\ h_1(x_{-2}, \ldots, x_{-c+1}, \delta_a(s_a)) = 0,
\end{cases}
\end{aligned} \tag{5.3}
$$

where

$$
h_2(x_{-3}, \ldots, x_{-c}, s_a) = f_1(0, x_{-3}, \ldots, x_{-c}, s_a) \oplus f_1(1, x_{-3}, \ldots, x_{-c}, s_a). \tag{5.4}
$$

Proof. Define

$$
\begin{aligned}
f_0(x_{-1}, \ldots, x_{-c}, s_a) &= f(0, x_{-1}, \ldots, x_{-c}, s_a), \\
f_1(x_{-2}, \ldots, x_{-c}, s_a) &= f(0, 0, x_{-2}, \ldots, x_{-c}, s_a), \\
f_2(x_{-3}, \ldots, x_{-c}, s_a) &= f(0, 0, 0, x_{-3}, \ldots, x_{-c}, s_a).
\end{aligned}
$$

Define h_0 and h_1 as follows. $h_0(x_{-2}, \ldots, x_{-c}, s_a) = 1$ if and only if $\langle 0, x_{-2}, \ldots, x_{-c}, s_a \rangle$ is not a 0-step state. $h_1(x_{-3}, \ldots, x_{-c}, s_a) = 1$ if and only if $f(x_0, x_{-1}, 0, x_{-3}, \ldots, x_{-c}, s_a)$ does not depend on x_{-1} and x_0. Since M is a semi-input-memory finite automaton $\mathcal{SIM}(M_a, f)$ and M_a is strongly cyclic, M is strongly connected. From Theorem 2.1.3 (f), it follows that $|W^M_{2,s}| = 2$ holds for any state s of M. From Lemma 5.4.4 (b), $h_1(x_{-3}, \ldots, x_{-c}, s_a) = 1$ if and only if $f(x_0, x_{-1}, 1, x_{-3}, \ldots, x_{-c}, s_a)$ does not depend on x_{-1} and x_0.

To prove $h_0(x_{-2}, \ldots, x_{-c}, s_a) = 0 \rightarrow h_0(x_{-1}, \ldots, x_{-c+1}, \delta_a(s_a)) = 1$, suppose $h_0(x_{-2}, \ldots, x_{-c}, s_a) = 0$. Since M is weakly invertible with delay 2, any state of M is a j-step state for some j, $0 \leqslant j \leqslant 2$. From the definition of h_0, $\langle 0, x_{-2}, \ldots, x_{-c}, s_a \rangle$ is a 0-step state. Using Lemma 5.4.3 (a), this yields that $\langle 1, x_{-2}, \ldots, x_{-c}, s_a \rangle$ is a 0-step state. From Lemma 5.4.1, $\langle x_0, x_{-1}, \ldots, x_{-c+1}, \delta_a(s_a) \rangle$ is not a 0-step state for any x_{-1}, $x_0 \in X$. Therefore, $h_0(x_{-1}, \ldots, x_{-c+1}, \delta_a(s_a)) = 1$.

To prove $h_1(x_{-3}, \ldots, x_{-c}, s_a) = 1 \rightarrow h_0(x_{-3}, \ldots, x_{-c-1}, \delta_a^{-1}(s_a)) = 0$, suppose $h_1(x_{-3}, \ldots, x_{-c}, s_a) = 1$. Then $f(x_0, \ldots, x_{-c}, s_a)$ does not depend on x_0 and x_{-1} for any $x_{-2} \in X$. It follows that $\langle x_{-1}, \ldots, x_{-c}, s_a \rangle$ is not a 0-step state for any x_{-2}, $x_{-1} \in X$. We prove by reduction to absurdity that $h_0(x_{-3}, \ldots, x_{-c-1}, \delta_a^{-1}(s_a)) = 0$ holds for any $x_{-c-1} \in X$. Suppose to the contrary that $h_0(x_{-3}, \ldots, x_{-c-1}, \delta_a^{-1}(s_a)) = 1$ for some $x_{-c-1} \in X$. From the definition of h_0, $\langle 0, x_{-3}, \ldots, x_{-c-1}, \delta_a^{-1}(s_a) \rangle$ is not a 0-step

state. From Lemma 5.4.3 (a), $\langle 1, x_{-3}, \ldots, x_{-c-1}, \delta_a^{-1}(s_a)\rangle$ is not a 0-step state. Using Lemma 5.4.2 (b), we have $\lambda(\langle 0, x_{-2}, \ldots, x_{-c}, s_a\rangle, X) \neq \lambda(\langle 1, x_{-2}, \ldots, x_{-c}, s_a\rangle, X)$. Thus $f(x_0, \ldots, x_{-c}, s_a)$ depends on x_{-1}. Therefore, $h_1(x_{-3}, \ldots, x_{-c}, s_a) = 0$. This contradicts $h_1(x_{-3}, \ldots, x_{-c}, s_a) = 1$. We conclude that $h_0(x_{-3}, \ldots, x_{-c-1}, \delta_a^{-1}(s_a)) = 0$ holds for any x_{-c-1}.

Below we prove that f can be expressed by f_0, f_1, f_2, h_0, and h_1. There are four cases to consider.

Case $h_0(x_{-2}, \ldots, x_{-c}, s_a) = 1$ & $h_1(x_{-3}, \ldots, x_{-c}, s_a) = 1$:

Let $s_{x'_{-2}, x'_{-1}} = \langle x'_{-1}, x'_{-2}, x_{-3}, \ldots, x_{-c}, s_a\rangle$. Since $h_1(x_{-3}, \ldots, x_{-c}, s_a) = 1$, $\lambda(s_{x_{-2}, x_{-1}}, x_0)$ does not depend on x_{-1} and x_0, that is, $|\lambda(s_{x_{-2},0}, X)| = |\lambda(s_{x_{-2},1}, X)| = 1$ and $\lambda(s_{x_{-2},0}, X) = \lambda(s_{x_{-2},1}, X)$. Letting $e = \lambda(s_{x_{-2},0}, 0)$, we have $\lambda(s_{x_{-2},x_{-1}}, x_0) = e$ for any $x_{-1}, x_0 \in X$. Let $\bar{x}_{-2} \in X \setminus \{x_{-2}\}$. From Lemma 5.4.4 (b), $|\lambda(s_{\bar{x}_{-2},0}, X)| = 1$ and $\lambda(s_{\bar{x}_{-2},0}, X) = \lambda(s_{\bar{x}_{-2},1}, X) \neq \lambda(s_{x_{-2},0}, X)$. It follows that $\lambda(s_{\bar{x}_{-2},x_{-1}}, x_0) = e \oplus 1$ for any $x_{-1}, x_0 \in X$. Therefore,

$$\begin{aligned} f(x_0, x_{-1}, \ldots, x_{-c}, s_a) &= \lambda(\langle x_{-1}, x_{-2}, x_{-3}, \ldots, x_{-c}, s_a\rangle, x_0) \\ &= \lambda(\langle 0, x_{-2}, x_{-3}, \ldots, x_{-c}, s_a\rangle, 0) \\ &= \lambda(\langle 0, 0, x_{-3}, \ldots, x_{-c}, s_a\rangle, 0) \oplus x_{-2} \\ &= f_2(x_{-3}, \ldots, x_{-c}, s_a) \oplus x_{-2}. \end{aligned}$$

Case $h_0(x_{-2}, \ldots, x_{-c}, s_a) = 1$ & $h_1(x_{-3}, \ldots, x_{-c}, s_a) = 0$:

Since $h_0(x_{-2}, \ldots, x_{-c}, s_a) = 1$, from the definition of h_0, $s_{x_{-2},0}$ is not a 0-step state. Using Lemma 5.4.3 (a), $s_{x_{-2},1}$ is not a 0-step state. It follows that $\lambda(s_{x_{-2},x_{-1}}, 0) = \lambda(s_{x_{-2},x_{-1}}, 1)$ holds for any $x_{-1} \in X$. Since $h_1(x_{-3}, \ldots, x_{-c}, s_a) = 0$, we have $\lambda(s_{x_{-2},0}, 0) \neq \lambda(s_{x_{-2},1}, 0)$. Therefore,

$$\begin{aligned} f(x_0, x_{-1}, \ldots, x_{-c}, s_a) &= \lambda(\langle x_{-1}, x_{-2}, \ldots, x_{-c}, s_a\rangle, x_0) \\ &= \lambda(\langle x_{-1}, x_{-2}, \ldots, x_{-c}, s_a\rangle, 0) \\ &= \lambda(\langle 0, x_{-2}, \ldots, x_{-c}, s_a\rangle, 0) \oplus x_{-1} \\ &= f_1(x_{-2}, \ldots, x_{-c}, s_a) \oplus x_{-1}. \end{aligned}$$

Case $h_0(x_{-2}, \ldots, x_{-c}, s_a) = 0$ & $h_1(x_{-2}, \ldots, x_{-c+1}, \delta_a(s_a)) = 1$:

Since $h_0(x_{-2}, \ldots, x_{-c}, s_a) = 0$, from the definition of h_0, $s_{x_{-2},0}$ is a 0-step state. Using Lemma 5.4.3 (a), $s_{x_{-2},1}$ is a 0-step state. It follows that $\lambda(s_{x_{-2},x_{-1}}, x_0) = \lambda(s_{x_{-2},x_{-1}}, 0) \oplus x_0$. Therefore,

$$\begin{aligned} f(x_0, x_{-1}, \ldots, x_{-c}, s_a) &= \lambda(\langle x_{-1}, \ldots, x_{-c}, s_a\rangle, x_0) \\ &= \lambda(\langle x_{-1}, \ldots, x_{-c}, s_a\rangle, 0) \oplus x_0 \\ &= f_0(x_{-1}, \ldots, x_{-c}, s_a) \oplus x_0. \end{aligned}$$

Case $h_0(x_{-2}, \ldots, x_{-c}, s_a) = 0$ & $h_1(x_{-2}, \ldots, x_{-c+1}, \delta_a(s_a)) = 0$:

Since $h_0(x_{-2}, \ldots, x_{-c}, s_a) = 0$, as proven in the preceding case, $s_{x_{-2},0}$ and $s_{x_{-2},1}$ are 0-step states. It follows that $\lambda(s_{x_{-2},x_{-1}}, x_0) = \lambda(s_{x_{-2},x_{-1}}, 0) \oplus x_0$ for any $x_{-1}, x_0 \in X$. Using Lemma 5.4.1, $\langle x_0, x_{-1}, x_{-2}, \ldots, x_{-c+1}, \delta_a(s_a)\rangle$, denoted by s_{x_{-2},x_{-1},x_0}, is not a 0-step state for any $x_{-1}, x_0 \in X$. It follows that $\lambda(s_{x_{-2},x_{-1},x_0}, 0) = \lambda(s_{x_{-2},x_{-1},x_0}, 1)$ for any $x_{-1}, x_0 \in X$. On the other hand, from $h_1(x_{-2}, \ldots, x_{-c+1}, \delta_a(s_a)) = 0$, we have $\lambda(s_{x_{-2},x_{-1},0}, 0) \neq \lambda(s_{x_{-2},x_{-1},1}, 0)$ for any $x_{-1} \in X$. Using Lemma 5.4.4 (a), $\lambda(s_{x_{-2},0,0}, 0) = \lambda(s_{x_{-2},1,0}, 0)$ if and only if $\lambda(s_{x_{-2},0}, 0) \neq \lambda(s_{x_{-2},1}, 0)$. It follows that

$$\begin{aligned}
\lambda(s_{x_{-2},0}, 0) \oplus \lambda(s_{x_{-2},1}, 0) &= \lambda(s_{x_{-2},0,0}, 0) \oplus \lambda(s_{x_{-2},1,0}, 0) \oplus 1 \\
&= f(0,0,0,x_{-2},\ldots,x_{-c+1},\delta_a(s_a)) \oplus f(0,0,1,x_{-2},\ldots,x_{-c+1},\delta_a(s_a)) \oplus 1 \\
&= f_1(0, x_{-2}, \ldots, x_{-c+1}, \delta_a(s_a)) \oplus f_1(1, x_{-2}, \ldots, x_{-c+1}, \delta_a(s_a)) \oplus 1 \\
&= h_2(x_{-2}, \ldots, x_{-c+1}, \delta_a(s_a)) \oplus 1.
\end{aligned}$$

Thus $\lambda(s_{x_{-2},0}, 0) \oplus \lambda(s_{x_{-2},x_{-1}}, 0) = x_{-1}(h_2(x_{-2}, \ldots, x_{-c+1}, \delta_a(s_a)) \oplus 1)$. This yields

$$\begin{aligned}
\lambda(s_{x_{-2},x_{-1}}, 0) &= \lambda(s_{x_{-2},0}, 0) \oplus x_{-1}(h_2(x_{-2}, \ldots, x_{-c+1}, \delta_a(s_a)) \oplus 1) \\
&= f_1(x_{-2}, \ldots, x_{-c}, s_a) \oplus x_{-1}(h_2(x_{-2}, \ldots, x_{-c+1}, \delta_a(s_a)) \oplus 1).
\end{aligned}$$

Therefore,

$$\begin{aligned}
f(x_0, x_{-1}, \ldots, x_{-c}, s_a) &= \lambda(\langle x_{-1}, x_{-2}, \ldots, x_{-c}, s_a\rangle, x_0) \\
&= \lambda(\langle x_{-1}, x_{-2}, \ldots, x_{-c}, s_a\rangle, 0) \oplus x_0 \\
&= f_1(x_{-2}, \ldots, x_{-c}, s_a) \oplus x_{-1}(h_2(x_{-2}, \ldots, x_{-c+1}, \delta_a(s_a)) \oplus 1) \oplus x_0. \quad \square
\end{aligned}$$

Lemma 5.4.6. *Let $M = \langle X, Y, S, \delta, \lambda\rangle$ be a c-order semi-input-memory finite automaton $\mathcal{SIM}(M_a, f)$, and $M_a = \langle Y_a, S_a, \delta_a, \lambda_a\rangle$ be strongly cyclic. If $c \geqslant 2$, $w_{2,M} = 1$, $X = Y = \{0, 1\}$ and M is weakly invertible with delay 2, then there exists a single-valued mapping f_2 from $X^{c-2} \times S_a$ to Y such that*

$$f(x_0, \ldots, x_{-c}, s_a) = f_2(x_{-3}, \ldots, x_{-c}, s_a) \oplus x_{-2}.$$

Proof. Since M is a semi-input-memory finite automaton $\mathcal{SIM}(M_a, f)$ and M_a is strongly cyclic, M is strongly connected. From Theorem 2.1.3 (f), it follows that $|W^M_{2,s}| = 1$ holds for any state s of M. This yields that $\lambda(\langle x_{-1}, x_{-2}, \ldots, x_{-c}, s_a\rangle, x_0)$, $x_{-1}, x_0 = 0, 1$ are the same. Since M is weakly invertible with delay 2, we have $\lambda(\langle 0, 0, x_{-3}, \ldots, x_{-c}, s_a\rangle, 0) \neq \lambda(\langle 0, 1, x_{-3}, \ldots, x_{-c}, s_a\rangle, 0)$. Thus

$$\begin{aligned}
f(x_0, x_{-1}, \ldots, x_{-c}, s_a) &= \lambda(\langle x_{-1}, \ldots, x_{-c}, s_a\rangle, x_0) \\
&= \lambda(\langle 0, x_{-2}, \ldots, x_{-c}, s_a\rangle, 0) \\
&= \lambda(\langle 0, 0, x_{-3}, \ldots, x_{-c}, s_a\rangle, 0) \oplus x_{-2} \\
&= f_2(x_{-3}, \ldots, x_{-c}, s_a) \oplus x_{-2},
\end{aligned}$$

where $f_2(x_{-3}, \ldots, x_{-c}, s_a) = f(0, 0, 0, x_{-3}, \ldots, x_{-c}, s_a)$. □

Lemma 5.4.7. *Let $M = \langle X, Y, S, \delta, \lambda \rangle$ be a c-order semi-input-memory finite automaton $\mathcal{SIM}(M_a, f)$, and $M_a = \langle Y_a, S_a, \delta_a, \lambda_a \rangle$ be strongly cyclic. If $c \geqslant 2$, $w_{2,M} = 4$, $X = Y = \{0, 1\}$ and M is weakly invertible with delay 2, then there exists a single-valued mapping f_0 from $X^c \times S_a$ to Y such that*

$$f(x_0, \ldots, x_{-c}, s_a) = f_0(x_{-1}, \ldots, x_{-c}, s_a) \oplus x_0.$$

Proof. Since M is a semi-input-memory finite automaton $\mathcal{SIM}(M_a, f)$ and M_a is strongly cyclic, M is strongly connected. From Theorem 2.1.3 (f), it follows that $|W^M_{2,s}| = 4$ holds for any state s of M. This yields that $\lambda(\langle x_{-1}, \ldots, x_{-c}, s_a \rangle, 0) \neq \lambda(\langle x_{-1}, \ldots, x_{-c}, s_a \rangle, 1)$. Thus $f(x_0, x_{-1}, \ldots, x_{-c}, s_a) = \lambda(\langle x_{-1}, \ldots, x_{-c}, s_a \rangle, x_0) = \lambda(\langle x_{-1}, \ldots, x_{-c}, s_a \rangle, 0) \oplus x_0 = f_0(x_{-1}, \ldots, x_{-c}, s_a) \oplus x_0$, where $f_0(x_{-1}, \ldots, x_{-c}, s_a) = f(0, x_{-1}, \ldots, x_{-c}, s_a)$. □

Theorem 5.4.1. *Let $M = \langle X, Y, S, \delta, \lambda \rangle$ be a c-order semi-input-memory finite automaton $\mathcal{SIM}(M_a, f)$, and $M_a = \langle Y_a, S_a, \delta_a, \lambda_a \rangle$ be strongly cyclic. Let $c \geqslant 2$ and $X = Y = \{0, 1\}$. Then M is weakly invertible with delay 2 if and only if one of the following conditions holds:*

(a) *There exists a single-valued mapping f_0 from $X^c \times S_a$ to Y such that*

$$f(x_0, \ldots, x_{-c}, s_a) = f_0(x_{-1}, \ldots, x_{-c}, s_a) \oplus x_0.$$

(b) *There exists a single-valued mapping f_2 from $X^{c-2} \times S_a$ to Y such that*

$$f(x_0, \ldots, x_{-c}, s_a) = f_2(x_{-3}, \ldots, x_{-c}, s_a) \oplus x_{-2}.$$

(c) *There exist single-valued mappings h_0 from $X^{c-1} \times S_a$ to $\{0, 1\}$, h_1 from $X^{c-2} \times S_a$ to $\{0, 1\}$, f_0 from $X^c \times S_a$ to Y, f_1 from $X^{c-1} \times S_a$ to Y, and f_2 from $X^{c-2} \times S_a$ to Y, such that* (5.2) *and* (5.3) *hold, where h_2 is defined by* (5.4).

Proof. only if: From Theorem 2.1.3 (a), $w_{2,M}$ is in $\{1, 2, 4\}$. In the case of $w_{2,M} = 4$, from Lemma 5.4.7, the condition (a) holds. In the case of $w_{2,M} = 1$, from Lemma 5.4.6, the condition (b) holds. In the case of $w_{2,M} = 2$, from Lemma 5.4.5, the condition (c) holds.

if: Suppose that one of conditions (a), (b) and (c) holds. In the case of (a), clearly, M is weakly invertible with delay 0. Thus M is weakly invertible with delay 2. In the case of (b), it is easy to verify that M is weakly invertible with delay 2.

Below we discuss the case (c). Suppose that the condition (c) holds. Let $s = \langle x_{-1}, x_{-2}, \ldots, x_{-c}, s_a \rangle$, $s_i = \langle i, x_{-1}, \ldots, x_{-c+1}, \delta_a(s_a) \rangle$ and $s_{i,j} = \langle j, i, x_{-1}, \ldots, x_{-c+2}, \delta^2_a(s_a) \rangle$, $i, j = 0, 1$. To prove s is a t-step state for some t, $0 \leqslant t \leqslant 2$, there are several cases to consider.

In the case of $h_0(x_{-2},\ldots,x_{-c},s_a)=0$, from (5.3) in the condition (c), $\lambda(s,x_0)=f(x_0,\ldots,x_{-c},s_a)=f'_0(x_{-1},\ldots,x_{-c},s_a)\oplus x_0$ for any $x_0\in X$, where $f'_0(x_{-1},\ldots,x_{-c},s_a)=f_0(x_{-1},\ldots,x_{-c},s_a)$ or $f_1(x_{-2},\ldots,x_{-c},s_a)\oplus x_{-1}\oplus x_{-1}h_2(x_{-2},\ldots,x_{-c+1},\delta_a(s_a))$. It follows that s is a 0-step state.

In the case of $h_0(x_{-2},\ldots,x_{-c},s_a)=1$, from (5.2) in the condition (c), $h_1(x_{-2},\ldots,x_{-c+1},\delta_a(s_a))=0$ holds; and from (5.3) in the condition (c), we have $\lambda(s,0)=\lambda(s,1)$. We further consider $h_0(x_{-1},\ldots,x_{-c+1},\delta_a(s_a))$ and $h_1(x_{-1},\ldots,x_{-c+2},\delta_a^2(s_a))$. In the subcase of $h_0(x_{-1},\ldots,x_{-c+1},\delta_a(s_a))=1$, from (5.3) in the condition (c), $h_0(x_{-1},\ldots,x_{-c+1},\delta_a(s_a))=1$ & $h_1(x_{-2},\ldots,x_{-c+1},\delta_a(s_a))=0$ yields $\lambda(s_0,0)=\lambda(s_0,1)\neq\lambda(s_1,0)=\lambda(s_1,1)$. Thus s is a 1-step state. In the subcase of $h_0(x_{-1},\ldots,x_{-c+1},\delta_a(s_a))=0$ & $h_1(x_{-1},\ldots,x_{-c+2},\delta_a^2(s_a))=1$, from (5.2) in the condition (c), $h_0(x_{-1},\ldots,x_{-c+1},\delta_a(s_a))=0$ yields $h_0(x_0,\ldots,x_{-c+2},\delta_a^2(s_a))=1$ for any $x_0\in X$. From (5.3) in the condition (c), $h_0(x_0,\ldots,x_{-c+2},\delta_a^2(s_a))=1$ & $h_1(x_{-1},\ldots,x_{-c+2},\delta_a^2(s_a))=1$ yields $\lambda(s_{x_0,0},0)=\lambda(s_{x_0,0},1)=\lambda(s_{x_0,1},0)=\lambda(s_{x_0,1},1)$ for any $x_0\in X$ and $\lambda(s_{0,0},0)\neq\lambda(s_{1,0},0)$. Thus s is a $\leqslant 2$-step state. It follows that s is a 2-step or 1-step state. In the subcase of $h_0(x_{-1},\ldots,x_{-c+1},\delta_a(s_a))=0$ & $h_1(x_{-1},\ldots,x_{-c+2},\delta_a^2(s_a))=0$, from (5.3) in the condition (c), $\lambda(s_i,0)\neq\lambda(s_i,1)$ for $i=0,1$. From (5.2) in the condition (c), $h_0(x_{-1},\ldots,x_{-c+1},\delta_a(s_a))=0$ yields $h_0(x_0,\ldots,x_{-c+2},\delta_a^2(s_a))=1$ for any $x_0\in X$. For any $x_0\in X$, since $h_0(x_0,\ldots,x_{-c+2},\delta_a^2(s_a))=1$ & $h_1(x_{-1},\ldots,x_{-c+2},\delta_a^2(s_a))=0$, from (5.3) in the condition (c), we have $\lambda(s_{x_0,0},0)=\lambda(s_{x_0,0},1)\neq\lambda(s_{x_0,1},0)=\lambda(s_{x_0,1},1)$ and $\lambda(s_{x_0,0},0)=f_1(x_0,x_{-1},\ldots,x_{-c+2},\delta_a^2(s_a))$. Since $h_0(x_{-1},\ldots,x_{-c+1},\delta_a(s_a))=0$ & $h_1(x_{-1},\ldots,x_{-c+2},\delta_a^2(s_a))=0$, from (5.3) and (5.4) in the condition (c), we have $\lambda(s_0,0)=f_1(x_{-1},\ldots,x_{-c+1},\delta_a(s_a))$ and

$$\begin{aligned}\lambda(s_1,0)&=f_1(x_{-1},\ldots,x_{-c+1},\delta_a(s_a))\oplus 1\oplus h_2(x_{-1},\ldots,x_{-c+2},\delta_a^2(s_a))\\&=f_1(x_{-1},\ldots,x_{-c+1},\delta_a(s_a))\oplus 1\oplus f_1(0,x_{-1},\ldots,x_{-c+2},\delta_a^2(s_a))\\&\quad\oplus f_1(1,x_{-1},\ldots,x_{-c+2},\delta_a^2(s_a)).\end{aligned}$$

It follows that $\lambda(s_0,0)=\lambda(s_1,0)$ if and only if $f_1(0,x_{-1},\ldots,x_{-c+2},\delta_a^2(s_a))\neq f_1(1,x_{-1},\ldots,x_{-c+2},\delta_a^2(s_a))$. Since $\lambda(s_{x_0,0},0)=f_1(x_0,x_{-1},\ldots,x_{-c+2},\delta_a^2(s_a))$ for $x_0=0,1$, $\lambda(s_0,0)=\lambda(s_1,0)$ if and only if $\lambda(s_{0,0},0)\neq\lambda(s_{1,0},0)$. Noticing that $\lambda(s,0)=\lambda(s,1)$, $\lambda(s_{x_0},0)\neq\lambda(s_{x_0},1)$ and $\lambda(s_{x_0,0},0)=\lambda(s_{x_0,0},1)\neq\lambda(s_{x_0,1},0)=\lambda(s_{x_0,1},1)$ for $x_0=0,1$, thus s is a 2-step state.

To sum up, if the condition (c) holds, then any state s of M is a t-step state for some t, $0\leqslant t\leqslant 2$. Thus M is weakly invertible with delay 2. □

Theorem 5.4.2. *Let $M=\langle X,Y,S,\delta,\lambda\rangle$ be a c-order semi-input-memory finite automaton $\mathcal{SIM}(M_a,f)$, and $M_a=\langle Y_a,S_a,\delta_a,\lambda_a\rangle$ be strongly cyclic. Let $c\geqslant 2$ and $X=Y=\{0,1\}$. Then M is a feedforward inverse with delay 2 if and only if one of the following conditions holds:*

(a) *There exists a single-valued mapping* f_0 *from* $X^c \times S_a$ *to* Y *such that*

$$f(x_0, \ldots, x_{-c}, s_a) = f_0(x_{-1}, \ldots, x_{-c}, s_a) \oplus x_0.$$

(b) *There exists a single-valued mapping* f_2 *from* $X^{c-2} \times S_a$ *to* Y *such that*

$$f(x_0, \ldots, x_{-c}, s_a) = f_2(x_{-3}, \ldots, x_{-c}, s_a) \oplus x_{-2}.$$

(c) *There exist single-valued mappings* h_0 *from* $X^{c-1} \times S_a$ *to* $\{0,1\}$, h_1 *from* $X^{c-2} \times S_a$ *to* $\{0,1\}$, f_0 *from* $X^c \times S_a$ *to* Y, f_1 *from* $X^{c-1} \times S_a$ *to* Y, *and* f_2 *from* $X^{c-2} \times S_a$ *to* Y, *such that* (5.2) *and* (5.3) *hold, where* h_2 *is defined by* (5.4).

Proof. Since M, i.e., $\mathcal{SIM}(M_a, f)$, is a semi-input-memory finite automaton and M_a is strongly cyclic, M is strongly connected. Using Theorem 2.2.2, M is a feedforward inverse with delay 2 if and only if M is invertible with delay 2. From Theorem 5.4.1, M is a feedforward inverse with delay 2 if and only if one of the conditions (a), (b) and (c) holds. □

We discuss briefly h_0 and h_1 in (5.2). Suppose that (5.2) holds for any x_{-1}, …, x_{-c} in X and any s_a in S_a. Then we have

$$\begin{aligned}
h_0(x_{-2}, \ldots, x_{-c}, s_a) = {} & h_0(x_{-2}, \ldots, x_{-c}, s_a) \\
& \vee (h_0(0, x_{-2}, \ldots, x_{-c+1}, \delta_a(s_a)) \oplus 1) \\
& \vee (h_0(1, x_{-2}, \ldots, x_{-c+1}, \delta_a(s_a)) \oplus 1), \\
h_1(x_{-3}, \ldots, x_{-c}, s_a) = {} & h_1(x_{-3}, \ldots, x_{-c}, s_a) \\
& \& (h_0(x_{-3}, \ldots, x_{-c}, 0, \delta_a^{-1}(s_a)) \oplus 1) \\
& \& (h_0(x_{-3}, \ldots, x_{-c}, 1, \delta_a^{-1}(s_a)) \oplus 1),
\end{aligned}$$

where $\vee$ stands for the logical-or operation, that is, $1 \vee 1 = 1 \vee 0 = 0 \vee 1 = 1, 0 \vee 0 = 0$. Conversely, given arbitrarily single-valued mappings h_0' from $X^{c-1} \times S_a$ to $\{0,1\}$ and h_1' from $X^{c-2} \times S_a$ to $\{0,1\}$, define

$$\begin{aligned}
h_0(x_{-2}, \ldots, x_{-c}, s_a) = {} & h_0'(x_{-2}, \ldots, x_{-c}, s_a) \\
& \vee (h_0'(0, x_{-2}, \ldots, x_{-c+1}, \delta_a(s_a)) \oplus 1) \\
& \vee (h_0'(1, x_{-2}, \ldots, x_{-c+1}, \delta_a(s_a)) \oplus 1), \\
h_1(x_{-3}, \ldots, x_{-c}, s_a) = {} & h_1'(x_{-3}, \ldots, x_{-c}, s_a) \\
& \& (h_0(x_{-3}, \ldots, x_{-c}, 0, \delta_a^{-1}(s_a)) \oplus 1) \\
& \& (h_0(x_{-3}, \ldots, x_{-c}, 1, \delta_a^{-1}(s_a)) \oplus 1).
\end{aligned} \tag{5.5}$$

It is easy to see that h_0 and h_1 satisfy (5.2). We conclude that h_0 and h_1 satisfy (5.2) if and only if h_0 and h_1 can be defined by (5.5).

Historical Notes

The structure of feedforward inverse finite automata is first studied in [100] for delay 0 and for delay 1 in binary case. References [4, 5, 146] present a characterization for feedforward inverse finite automata with delay 1 of which sizes of the input and output alphabets are the same, and [129] introduces another characterization of them by means of mutual invertibility. Reference [153] gives the first characterization for binary feedforward inverse finite automata with delay 2, and [130] gives another characterization of them by means of mutual invertibility. Reference [141] deals with the structure of binary feedforward inverse finite automata with delay 3. Sections 5.1 and 5.2 are based on [100]. Section 5.3 is based on [129]. And Sect. 5.4 is based on [130].

6. Some Topics on Structure Problem

Renji Tao

Institute of Software, Chinese Academy of Sciences
Beijing 100080, China trj@ios.ac.cn

Summary.

This chapter investigates the following problem: given an invertible (respectively inverse, weakly invertible, weak inverse, and feedforward invertible) finite automaton, characterize the structure of the set of all its inverses (respectively original inverses, weak inverses, original weak inverses and weak inverses with bounded error propagation).

To characterize the set of all inverses (or weak inverses, or weak inverses with bounded error propagation) of a given finite automaton, the measures are, loosely speaking, first taking one member in the set and making a partial finite automaton by restricting its inputs, then constructing the set from this partial finite automaton. As an auxiliary tool, partial finite automata and partial semi-input-memory finite automata are defined.

To characterize the set of all original inverses (or original weak inverses) of a given finite automaton, we use the state tree method and results in Sect. 1.6 of Chap. 1.

Key words: *inverses, weak inverses, original inverses, original weak inverses, bounded error propagation*

This chapter investigates the following problem: given an invertible (respectively inverse, weakly invertible, weak inverse, feedforward invertible) finite automaton, characterize the structure of the set of all its inverses (respectively original inverses, weak inverses, original weak inverses, weak inverses with bounded error propagation).

To characterize the set of all inverses (or weak inverses, or weak inverses with bounded error propagation) of a given finite automaton, the measures are, loosely speaking, first taking one member in the set and making a partial finite automaton by restricting its inputs, then constructing the set from this partial finite automaton. As an auxiliary tool, partial finite automata and partial semi-input-memory finite automata are defined.

To characterize the set of all original inverses (or original weak inverses) of a given finite automaton, we use the state tree method and results in Sect. 1.6 of Chap. 1.

6.1 Some Variants of Finite Automata

6.1.1 Partial Finite Automata

A *partial finite automaton* is a quintuple $\langle X, Y, S, \delta, \lambda\rangle$, where X, Y and S are nonempty finite sets, δ is a single-valued mapping from a subset of $S \times X$ to S, and λ is a single-valued mapping from a subset of $S \times X$ to Y. X, Y and S are called the *input alphabet*, the *output alphabet* and the *state alphabet* of the partial finite automaton, respectively; and δ and λ are called the *next state function* and the *output function* of the partial finite automaton, respectively.

A partial finite automaton may naturally be expanded to a finite automaton. Taken a special symbol, say $\_$, to stand for the "undefined symbol", which is not in S or Y. Denote the domain of δ by Δ and the domain of λ by Λ. Let

$$\delta(s,x) = \_\,, \quad \text{if } (s,x) \in (S \times X) \setminus \Delta \text{ or } s = \_\,,$$
$$\lambda(s,x) = \_\,, \quad \text{if } (s,x) \in (S \times X) \setminus \Lambda \text{ or } s = \_\,.$$

$\langle X, Y \cup \{\_\}, S \cup \{\_\}, \delta, \lambda\rangle$ is a finite automaton, and is called the *trivial expansion* of M.

By expanding domains of δ and λ of the trivial expansion of the partial finite automaton M, the domain of δ of M may be expanded to $(S \cup \{\_\}) \times X^*$; the domain of λ of M may be expanded to $(S \cup \{\_\}) \times (X^* \cup X^\omega)$.

Let $s \in S$ and $\alpha \in X^*$. If $|\alpha| > 0$ yields $\delta(s, \alpha_1) \in S$, where α_1 is the prefix of α of length $|\alpha| - 1$, we say that α is *applicable* to s. Clearly, if $|\alpha| \leqslant 1$, then α is applicable to s.

Let $\alpha = a_1 \ldots a_r$ and $\beta = b_1 \ldots b_r$, where $a_1, b_1, \ldots, a_r, b_r \in Y \cup \{\_\}$. If for any i, $1 \leqslant i \leqslant r$, $a_i \neq \_$ and $b_i \neq \_$ implies $a_i = b_i$, we say that α and β are *compatible*, denoted by $\alpha \approx \beta$. If for any i, $1 \leqslant i \leqslant r$, $b_i \neq \_$ implies $a_i = b_i$, we say that α is *stronger* than β, denoted by $\beta \prec \alpha$. Notice that the relation $\approx$ over words is reflexive and symmetric and the relation $\prec$ over words is reflexive and transitive. It is easy to see that $\alpha \prec \beta$ and $\beta \prec \alpha$ if and only if $\alpha = \beta$.

Let $M = \langle X, Y, S, \delta, \lambda\rangle$ be a partial finite automaton. For any states s_1 and s_2 of M, if for any α in X^*, that α is applicable to s_1 and s_2 implies that $\lambda_1(s_1, \alpha) \approx \lambda_2(s_2, \alpha)$, we say that s_1 and s_2 are *compatible*, denoted by $s_1 \approx s_2$. Notice that the relation $\approx$ over states is reflexive and symmetric. It

is easy to see that for any s_1, s_2 in S and any α in X^*, if $s_1 \approx s_2$ and $\delta(s_i, \alpha)$ is defined, $i = 1, 2$, then $\delta(s_1, \alpha) \approx \delta(s_2, \alpha)$. For any nonempty subset T of S, if any two states in T are compatible, T is called a *compatible set* of M. If T is a compatible set of M and for any T', $T \subset T' \subseteq S$, T' is not a compatible set of M, T is called a *maximum compatible set* of M.

Let $C_1, \ldots, C_k$ be k compatible sets of M. If $\cup_{i=1}^{k} C_i = S$, and for any i, $1 \leqslant i \leqslant k$, and any x in X, there exists j, $1 \leqslant j \leqslant k$, such that $\delta(C_i, x)$, i.e., $\{\delta(s, x) \mid s \in C_i, \delta(s, x) \text{ is defined}\}$, is a subset of C_j, the sequence $C_1, \ldots, C_k$ is called a *closed compatible family* of M. Notice that C_i and C_h may be the same set, $i \neq h$.

Let $C_1, \ldots, C_k$ be a closed compatible family of $M = \langle X, Y, S, \delta, \lambda \rangle$. Let $X' = X$, $Y' = Y$, $S' = \{c_1, \ldots, c_k\}$,

$$\delta'(c_i, x) = \begin{cases} c_j, & \text{if } \delta(C_i, x) \neq \varnothing, \\ \text{undefined}, & \text{if } \delta(C_i, x) = \varnothing, \end{cases}$$

$$\lambda'(c_i, x) = \begin{cases} \lambda(s, x), & \text{if } \exists s_1 (s_1 \in C_i \ \& \ \lambda(s_1, x) \text{ is defined}), \\ \text{undefined}, & \text{otherwise}, \end{cases}$$

$$i = 1, \ldots, k, \ x \in X,$$

where j is an arbitrary integer satisfying $\delta(C_i, x) \subseteq C_j$, and s is an arbitrary state in C_i such that $\lambda(s, x)$ is defined. Since C_i is compatible, the value of $\lambda'(c_i, x)$ is independent of the selection of s. Let $M' = \langle X', Y', S', \delta', \lambda' \rangle$. Then M' is a partial finite automaton. We use $\mathcal{M}(C_1, \ldots, C_k)$ to denote the set of all such M'.

Let $M_i = \langle X_i, Y_i, S_i, \delta_i, \lambda_i \rangle$, $i = 1, 2$ be two partial finite automata with $X_1 = X_2$. Let s_i be in S_i, $i = 1, 2$. If for any α in X_1^*, that α is applicable to s_1 implies that α is applicable to s_2 and $\lambda_1(s_1, \alpha) \prec \lambda_2(s_2, \alpha)$, we say that s_2 is *stronger* than s_1, denoted by $s_1 \prec s_2$. If for any s_1 in S_1, there exists s_2 in S_2 such that $s_1 \prec s_2$, we say that M_2 is *stronger* than M_1, denoted by $M_1 \prec M_2$. If for any α in X_1^*, α is applicable to s_1 if and only if α is applicable to s_2, and $\lambda_1(s_1, \alpha) = \lambda_2(s_2, \alpha)$ whenever α is applicable to s_1, s_1 and s_2 are said to be *equivalent*, denoted by $s_1 \sim s_2$. If for any s_1 in S_1, there exists s_2 in S_2 such that $s_1 \sim s_2$, and for any s_2 in S_2, there exists s_1 in S_1 such that $s_2 \sim s_1$, M_1 and M_2 are said to be *equivalent*, denoted by $M_1 \sim M_2$.

Similar to the case of finite automata, for any s_1 in S_1, any s_2 in S_2, and any α in X_1^*, if $\delta_i(s_i, \alpha)$, $i = 1, 2$ are defined and $s_1 \sim s_2$, then $\delta_1(s_1, \alpha) \sim \delta_2(s_2, \alpha)$. And for any s_1 in S_1, any s_2 in S_2, and any α in X_1^*, if $\delta_i(s_i, \alpha)$, $i = 1, 2$ are defined and $s_1 \prec s_2$, then $\delta_1(s_1, \alpha) \prec \delta_2(s_2, \alpha)$.

From the definition, it is easy to show that for any positive integer k, any s_1 in S_1 and any s_2 in S_2, a sufficient and necessary condition of $s_1 \sim s_2$ is

the following: for any $\alpha \in X_1^*$ with $|\alpha| \leqslant k$, α is applicable to s_1 if and only if α is applicable to s_2, $\lambda_1(s_1, \alpha) = \lambda_2(s_2, \alpha)$ whenever α is applicable to s_1, and for any $\alpha \in X_1^*$ with $|\alpha| = k$, $\delta_1(s_1, \alpha)$ is defined if and only if $\delta_2(s_2, \alpha)$ is defined, and $\delta_1(s_1, \alpha) \sim \delta_2(s_2, \alpha)$ whenever $\delta_1(s_1, \alpha)$ is defined.

Notice that both the relation $\prec$ over states and the relation $\prec$ over partial finite automata are reflexive and transitive, and both the relation $\sim$ over states and the relation $\sim$ over partial finite automata are reflexive, symmetric and transitive. It is easy to see that $s_1 \sim s_2$ if and only if $s_1 \prec s_2$ and $s_2 \prec s_1$.

We point out that in the case of finite automata, relations $s_1 \prec s_2$, $s_1 \approx s_2$ and $s_1 \sim s_2$ are the same.

Let $M_i = \langle X_i, Y_i, S_i, \delta_i, \lambda_i \rangle$, $i = 1, 2$ be two partial finite automata with $X_1 = X_2$. Assume that for any i, $1 \leqslant i \leqslant 2$, any s_i in S_i and any x in X_1, $\delta_i(s_i, x)$ is defined if and only if $\lambda_i(s_i, x)$ is defined. Then for any s_i in S_i, $i = 1, 2$, $s_1 \sim s_2$ if and only if for any α in X_1^*, $\lambda_1(s_1, \alpha)$ is defined (i.e., each letter in $\lambda_1(s_1, \alpha)$ is defined) if and only if $\lambda_2(s_2, \alpha)$ is defined, and $\lambda_1(s_1, \alpha) = \lambda_2(s_2, \alpha)$ whenever they are defined. In fact, $s_1 \sim s_2$ if and only if for any α in X_1^* and any x in X_1, αx is applicable to s_1 if and only if αx is applicable to s_2, and for any α in X_1^* and any x in X_1, $\lambda_1(s_1, \alpha x) = \lambda_2(s_2, \alpha x)$ whenever αx is applicable to s_1. From the assumption that $\lambda_i(s_i, \alpha)$ is defined if and only if $\delta_i(s_i, \alpha)$ is defined, αx is applicable to s_i if and only if $\lambda_i(s_i, \alpha)$ is defined. Thus the condition that for any α in X_1^* and any x in X_1, αx is applicable to s_1 if and only if αx is applicable to s_2, is equivalent to the condition that for any α in X_1^*, $\lambda_1(s_1, \alpha)$ is defined if and only if $\lambda_2(s_2, \alpha)$ is defined. Similarly, the condition that for any α in X_1^* and any x in X_1, $\lambda_1(s_1, \alpha x) = \lambda_2(s_2, \alpha x)$ whenever αx is applicable to s_1, is equivalent to the condition that for any α in X_1^* and any x in X_1, $\lambda_1(s_1, \alpha x) = \lambda_2(s_2, \alpha x)$ whenever $\lambda_1(s_1, \alpha)$ is defined. Therefore, $s_1 \sim s_2$ if and only if for any $\alpha \in X_1^*$, $\lambda_1(s_1, \alpha)$ is defined if and only if $\lambda_2(s_2, \alpha)$ is defined, and for any $\alpha \in X_1^*$ and $x \in X_1$, $\lambda_1(s_1, \alpha x) = \lambda_2(s_2, \alpha x)$ whenever $\lambda_1(s_1, \alpha)$ is defined. Thus $s_1 \sim s_2$ if and only if for any $\alpha \in X_1^*$, $\lambda_1(s_1, \alpha)$ is defined if and only if $\lambda_2(s_2, \alpha)$ is defined, and for any $\alpha \in X_1^*$ and $x \in X_1$, $\lambda_1(s_1, \alpha x) = \lambda_2(s_2, \alpha x)$ whenever $\lambda_1(s_1, \alpha)$ and $\lambda_2(s_2, \alpha)$ are defined. We conclude that $s_1 \sim s_2$ if and only if for any α in X_1^*, $\lambda_1(s_1, \alpha)$ is defined if and only if $\lambda_2(s_2, \alpha)$ is defined, and $\lambda_1(s_1, \alpha) = \lambda_2(s_2, \alpha)$ whenever they are defined.

Let $M_i = \langle X_i, Y_i, S_i, \delta_i, \lambda_i \rangle$, $i = 1, 2$ be two partial finite automata with $X_1 = X_2$. Assume that for any i, $1 \leqslant i \leqslant 2$, any s_i in S_i and any x in X_1, $\delta_i(s_i, x)$ is defined if and only if $\lambda_i(s_i, x)$ is defined. Then for any s_i in S_i, $i = 1, 2$, $s_1 \prec s_2$ if and only if for any α in X_1^*, $\lambda_2(s_2, \alpha)$ is defined and $\lambda_1(s_1, \alpha) = \lambda_2(s_2, \alpha)$ whenever $\lambda_1(s_1, \alpha)$ is defined. In fact, $s_1 \prec s_2$ if and only if for any α in X_1^* and any x in X_1, αx is applicable to s_2 and $\lambda_1(s_1, \alpha x)$

$\prec \lambda_2(s_2, \alpha x)$ whenever αx is applicable to s_1. Thus $s_1 \prec s_2$ if and only if for any α in X_1^* and any x in X_1, $\lambda_2(s_2, \alpha)$ is defined and $\lambda_1(s_1, \alpha x) \prec \lambda_2(s_2, \alpha x)$ whenever $\lambda_1(s_1, \alpha)$ is defined. Therefore, $s_1 \prec s_2$ if and only if for any α in X_1^* and any x in X_1, $\lambda_2(s_2, \alpha x)$ is defined and $\lambda_1(s_1, \alpha x) = \lambda_2(s_2, \alpha x)$ whenever $\lambda_1(s_1, \alpha x)$ is defined. It follows that $s_1 \prec s_2$ if and only if for any α in X_1^*, $\lambda_2(s_2, \alpha)$ is defined and $\lambda_1(s_1, \alpha) = \lambda_2(s_2, \alpha)$ whenever $\lambda_1(s_1, \alpha)$ is defined.

Let $M_i = \langle X_i, Y_i, S_i, \delta_i, \lambda_i \rangle$, $i = 1, 2$ be two partial finite automata with $X_1 = X_2$. Assume that for any i, $1 \leqslant i \leqslant 2$, any s_i in S_i and any x in X_1, $\delta_i(s_i, x)$ is defined if and only if $\lambda_i(s_i, x)$ is defined. It is easy to show that for any positive integer k, any s_1 in S_1 and any s_2 in S_2, a sufficient and necessary condition of $s_1 \sim s_2$ is the following: for any $\alpha \in X_1^*$ with $|\alpha| \leqslant k$, $\lambda_1(s_1, \alpha)$ is defined if and only if $\lambda_2(s_2, \alpha)$ is defined, $\lambda_1(s_1, \alpha) = \lambda_2(s_2, \alpha)$ whenever they are defined, and $\delta_1(s_1, \alpha) \sim \delta_2(s_2, \alpha)$ whenever $\delta_1(s_1, \alpha)$ is defined and $|\alpha| = k$.

Lemma 6.1.1. *Let $M_i = \langle X, Y_i, S_i, \delta_i, \lambda_i \rangle$ be a partial finite automaton and $s_i \in S_i$, $i = 1, 2$.*

(a) *For any $s_i \in S_i$, $i = 1, 2$, and any $\alpha \in X^*$, if $s_1 \prec s_2$ and $\delta_1(s_1, \alpha)$ is defined, then $\delta_2(s_2, \alpha)$ is defined and $\delta_1(s_1, \alpha) \prec \delta_2(s_2, \alpha)$.*

(b) *For any s_0, $s_1 \in S_1$, and any $s_2 \in S_2$, if $s_0 \prec s_2$ and $s_1 \prec s_2$, then $s_0 \approx s_1$.*

Proof. (a) Let $\beta \in X^*$ be applicable to $\delta_1(s_1, \alpha)$. Then $\alpha\beta$ is applicable to s_1. From $s_1 \prec s_2$, $\alpha\beta$ is applicable to s_2 and $\lambda_1(s_1, \alpha\beta) \prec \lambda_2(s_2, \alpha\beta)$. Take $\beta_1 \in X$. Clearly, β_1 is applicable to $\delta_1(s_1, \alpha)$. Thus $\alpha\beta_1$ is applicable to s_2. It follows that $\delta_2(s_2, \alpha)$ is defined. Since $\alpha\beta$ is applicable to s_2, β is applicable to $\delta_2(s_2, \alpha)$. Since $\lambda_1(s_1, \alpha\beta) \prec \lambda_2(s_2, \alpha\beta)$, we have $\lambda_1(\delta_1(s_1, \alpha), \beta) \prec \lambda_2(\delta_2(s_2, \alpha), \beta)$. Therefore, $\delta_1(s_1, \alpha) \prec \delta_2(s_2, \alpha)$.

(b) Let α in X^* be applicable to s_0 and s_1. Since $s_0 \prec s_2$ and $s_1 \prec s_2$, α is applicable to s_2 and $\lambda_1(s_i, \alpha) \prec \lambda_2(s_2, \alpha)$, $i = 0, 1$. It follows that $\lambda_1(s_0, \alpha) \approx \lambda_1(s_1, \alpha)$. Therefore, $s_0 \approx s_1$. □

Lemma 6.1.2. *Let $M = \langle X, Y, S, \delta, \lambda \rangle$ be a partial finite automaton, and $C_1, \ldots, C_k$ a closed compatible family of M. For any $M' = \langle X, Y, \{c_1, \ldots, c_k\}, \delta', \lambda' \rangle$ in $\mathcal{M}(C_1, \ldots, C_k)$ and any s in C_i, $1 \leqslant i \leqslant k$, we have $s \prec c_i$.*

Proof. We prove by induction on the length of α that for any i, $1 \leqslant i \leqslant k$, any s in C_i, and any α in X^*, if α is applicable to s, then α is applicable to c_i and $\lambda(s, \alpha) \prec \lambda'(c_i, \alpha)$. *Basis* : $|\alpha| \leqslant 1$. Clearly, α is applicable to s and c_i. From the definition of λ', it is evident that $\lambda(s, \alpha) \prec \lambda'(c_i, \alpha)$. Thus $s \prec c_i$. *Induction step* : Suppose that for any α of length $j (\geqslant 1)$ the proposition has been proven. To prove the case $j + 1$, let $s \in C_i$ and $\alpha \in X^*$ with $|\alpha| = j + 1$. Suppose that α is applicable to s. Let $\alpha = x\alpha'$, where

$x \in X$. Then $|\alpha'| = j$. Since $|\alpha'| \geqslant 1$, $\delta(s,x)$ is defined. From the definition of δ', $\delta'(c_i,x)$ is defined and $\delta(s,x) \in C_h$, where $c_h = \delta'(c_i,x)$. Since α is applicable to s, α' is applicable to $\delta(s,x)$. From the induction hypothesis, α' is applicable to $\delta'(c_i,x)$ and $\lambda(\delta(s,x),\alpha') \prec \lambda'(\delta'(c_i,x),\alpha')$. Since $s \in C_i$, we have $\lambda(s,x) \prec \lambda'(c_i,x)$. It follows that $\lambda(s,\alpha) = \lambda(s,x)\lambda(\delta(s,x),\alpha') \prec \lambda'(c_i,x)\lambda'(\delta'(c_i,x),\alpha') = \lambda'(c_i,\alpha)$. □

Theorem 6.1.1. *Let M be a partial finite automaton, and $C_1, \ldots, C_k$ a closed compatible family of M. For any M' in $\mathcal{M}(C_1,\ldots,C_k)$, we have $M \prec M'$.*

Proof. Let $M' = \langle X, Y, \{c_1,\ldots,c_k\}, \delta', \lambda'\rangle$. Since $\cup_{i=1}^{k} C_i$ is the state alphabet of M, for any state s of M, there exists i, $1 \leqslant i \leqslant k$, such that $s \in C_i$. From Lemma 6.1.2, we have $s \prec c_i$. Therefore, $M \prec M'$. □

Let $M_i = \langle X_i, Y_i, S_i, \delta_i, \lambda_i\rangle$, $i = 1,2$ be two partial finite automata. If $X_1 \subseteq X_2$, $Y_1 \subseteq Y_2$, $S_1 \subseteq S_2$, and for any s in S_1 and any x in X_1, that $\delta_1(s,x)$ is defined implies that $\delta_2(s,x)$ is defined and $\delta_1(s,x) = \delta_2(s,x)$, and that $\lambda_1(s,x)$ is defined implies that $\lambda_2(s,x)$ is defined and $\lambda_1(s,x) = \lambda_2(s,x)$, M_1 is called a *partial finite subautomaton* of M_2, denoted by $M_1 \leqslant M_2$. For any nonempty subset S_2' of S_2 and any nonempty subset X_2' of X_2, if $\delta_2(S_2', X_2') = \{s' \mid \text{there exist } s_2 \in S_2' \text{ and } x \in X_2 \text{ such that } s' = \delta_2(s_2,x) \in S_2\} \subseteq S_2'$, S_2' is said to be *closed* with respect to X_2' in M_2. Clearly, if S_2' is closed with respect to X_2' in M_2, then $\langle X_2', Y_2, S_2', \delta_2|_{S_2' \times X_2'}, \lambda_2|_{S_2' \times X_2'}\rangle$ is a partial finite subautomaton of M_2, where $\delta_2|_{S_2' \times X_2'}$ and $\lambda_2|_{S_2' \times X_2'}$ are restrictions of δ_2 and λ_2 on $S_2' \times X_2'$, respectively.

Notice that the relation $\leqslant$ on partial finite automata is reflexive and transitive. It is easy to see that $M_1 \leqslant M_2$ implies $M_1 \prec M_2$ in the case of $X_1 = X_2$.

Let $M_i = \langle X_i, Y_i, S_i, \delta_i, \lambda_i\rangle$, $i = 1,2$ be two partial finite automata. M_1 and M_2 are said to be *isomorphic*, if $X_1 = X_2$, $Y_1 = Y_2$ and there exists a one-to-one mapping φ from S_1 onto S_2 such that for any s_1 in S_1 and any x in X_1, $\delta_1(s_1,x)$ is defined if and only if $\delta_2(\varphi(s_1),x)$ is defined, and $\varphi(\delta_1(s_1,x)) = \delta_2(\varphi(s_1),x)$ whenever they are defined, and $\lambda_1(s_1,x)$ is defined if and only if $\lambda_2(\varphi(s_1),x)$ is defined, and $\lambda_1(s_1,x) = \lambda_2(\varphi(s_1),x)$ whenever they are defined. φ is called an *isomorphism* from M_1 to M_2.

Notice that the isomorphic relation on partial finite automata is reflexive, symmetric and transitive. Clearly, if M_1 and M_2 are isomorphic, then $M_1 \sim M_2$ and $M_1 \prec M_2$.

Theorem 6.1.2. *Let $M = \langle X, Y, S, \delta, \lambda\rangle$ and $M'' = \langle X'', Y'', S'', \delta'', \lambda''\rangle$ be two partial finite automata. If $M \prec M''$ and $Y = Y''$, then there exist a partial finite subautomaton M''' of M'', a closed compatible family $C_1, \ldots,$*

C_k of M, and a partial finite automaton M' in $\mathcal{M}(C_1, \ldots, C_k)$ such that M' and M''' are isomorphic.

Proof. Suppose that $M \prec M''$. Then $X = X''$. Let $S''' = \{s'' \mid s'' \in S'', \exists s(s \in S \ \& \ s \prec s'')\}$. For any s''' in S''', let $\psi(s''') = \{s \mid s \in S, s \prec s'''\}$. Clearly, $\psi(s''') \neq \varnothing$. From $M \prec M''$, we have $\cup_{s''' \in S'''} \psi(s''') = S$. From Lemma 6.1.1 (b), for any s''' in S''', $\psi(s''')$ is a compatible set of M. Let s''' $\in S'''$, $x \in X$, and $\delta(\psi(s'''), x) \neq \varnothing$. Let s be in $\psi(s''')$, so that $\delta(s, x)$ is defined. Since $s \prec s'''$, from Lemma 6.1.1 (a), $\delta''(s''', x)$ is defined and $\delta(s, x)$ $\prec \delta''(s''', x)$. Thus $\delta(\psi(s'''), x) \subseteq \psi(\delta''(s''', x))$. Let states of S''' be s'''_1, …, s'''_k, where $k = |S'''|$. Let $C_i = \psi(s'''_i)$, $i = 1, \ldots, k$. We conclude that the sequence C_1, …, C_k is a closed compatible family of M.

We construct a partial finite automaton $M''' = \langle X'', Y'', S''', \delta''', \lambda''' \rangle$ as follows. For any s'''_i in S''' and any x in X, whenever $\delta(C_i, x) \neq \varnothing$, there exists s in $\psi(s'''_i)$ such that $\delta(s, x)$ is defined. In the preceding paragraph, we have proven that for any $s''' \in S'''$ and any $x \in X$, $\delta(\psi(s'''), x) \neq \varnothing$ yields $\delta(\psi(s'''), x) \subseteq \psi(\delta''(s''', x))$. Thus $\delta(\psi(s'''_i), x) \subseteq \psi(\delta''(s'''_i, x))$. Since $\delta(\psi(s'''_i), x) = \delta(C_i, x) \neq \varnothing$, $\delta''(s'''_i, x)$ is defined and in S'''. If $\delta(C_i, x) \neq \varnothing$, we define $\delta'''(s'''_i, x) = \delta''(s'''_i, x)$; otherwise, $\delta'''(s'''_i, x)$ is undefined. Whenever there exists s in C_i such that $\lambda(s, x)$ is defined, since x is applicable to s and $s \prec s'''_i$, we have $\lambda(s, x) \prec \lambda''(s'''_i, x)$. It follows that $\lambda''(s'''_i, x)$ is defined and $\lambda(s, x) = \lambda''(s'''_i, x)$. If there exists s in C_i such that $\lambda(s, x)$ is defined, we define $\lambda'''(s'''_i, x) = \lambda''(s'''_i, x)$; otherwise, $\lambda'''(s'''_i, x)$ is undefined. It is easy to see that M''' is a partial finite subautomaton of M''.

Take a partial finite automaton $\langle X, Y, S', \delta', \lambda' \rangle$ in $\mathcal{M}(C_1, \ldots, C_k)$ as M', where $S' = \{c_1, \ldots, c_k\}$,

$$\delta'(c_i, x) = \begin{cases} c_j, & \text{if } \delta(C_i, x) \neq \varnothing, \\ \text{undefined}, & \text{if } \delta(C_i, x) = \varnothing, \end{cases}$$

$$\lambda'(c_i, x) = \begin{cases} \lambda(s, x), & \text{if } \exists s_1(s_1 \in C_i \ \& \ \lambda(s_1, x) \text{ is defined}), \\ \text{undefined}, & \text{otherwise}, \end{cases}$$

$$i = 1, \ldots, k, \ x \in X,$$

j is the integer satisfying $\delta''(s'''_i, x) = s'''_j$, and s is an arbitrary state in C_i such that $\lambda(s, x)$ is defined. In the first paragraph of the proof, we have proven that for any $s''' \in S'''$ and any $x \in X$, $\delta(\psi(s'''), x) \neq \varnothing$ yields $\delta(\psi(s'''), x) \subseteq \psi(\delta''(s''', x))$. From $\delta''(s'''_i, x) = s'''_j$, we have $\delta(\psi(s'''_i), x) \subseteq \psi(\delta''(s'''_i, x)) = \psi(s'''_j)$, that is, $\delta(C_i, x) \subseteq C_j$. Thus M' is in $\mathcal{M}(C_1, \ldots, C_k)$ indeed.

We prove that M' and M''' are isomorphic. Let $\varphi(c_i) = s'''_i$, $i = 1, \ldots, k$. Clearly, φ is a one-to-one mapping from S' onto S'''. From the constructions of M''' and M', it is easy to see that $\delta'''(s'''_i, x)$ is defined if and only if

$\delta'(c_i, x)$ is defined, and that whenever they are defined, $\delta'''(s_i''', x) = s_j'''$ if and only if $\delta'(c_i, x) = c_j$. Similarly, $\lambda'''(s_i''', x)$ is defined if and only if $\lambda'(c_i, x)$ is defined, and whenever they are defined, $\lambda'''(s_i''', x) = \lambda'(c_i, x)$. Thus φ is an isomorphism from M' to M'''. We conclude that M' and M''' are isomorphic. □

Let $M = \langle X, Y, S, \delta, \lambda \rangle$ be a finite automaton, and $M' = \langle Y, X, S', \delta', \lambda' \rangle$ a partial finite automaton. For any states s in S and s' in S', if

$$(\forall \alpha)_{X^\omega}\ (\exists \alpha_0)_{(X \cup \{\_\})^*}\ [\ \lambda'(s', \lambda(s, \alpha)) = \alpha_0 \alpha\ \ \&\ \ |\alpha_0| = \tau\ ],$$

(s', s) is called a *match pair* with delay τ or say that s' *τ-matches* s. Clearly, if s' τ-matches s and $\beta = \lambda(s, \alpha)$ for some α in X^*, then $\delta'(s', \beta)$ τ-matches $\delta(s, \alpha)$. M' is called a *weak inverse* with delay τ of M, if for any s in S, there exists s' in S' such that (s', s) is a match pair with delay τ.

Let $M = \langle X, Y, S, \delta, \lambda \rangle$ be a partial finite automaton. The states of M is said to be *reachable* from a state s (respectively from a subset I of S), if for any state s' in S, there exists α in X^* such that $s' = \delta(s, \alpha)$ holds (respectively holds for some s in I).

6.1.2 Nondeterministic Finite Automata

A *nondeterministic finite automaton* is a quintuple $\langle X, Y, S, \delta, \lambda \rangle$, where X, Y and S are nonempty finite sets, δ is a single-valued mapping from $S \times X$ to $2^S \setminus \{\varnothing\}$, and λ is a single-valued mapping from $S \times X$ to $2^Y \setminus \{\varnothing\}$, where 2^T stands for the power set of a set T, that is, $2^T = \{T' \mid T' \subseteq T\}$. X, Y and S are called the *input alphabet*, the *output alphabet* and the *state alphabet* of the nondeterministic finite automaton, respectively; and δ and λ are called the *next state function* and the *output function* of the nondeterministic finite automaton, respectively.

The domain of δ may be expanded to $S \times X^*$ as follows.

$$\begin{aligned} &\delta(s, \varepsilon) = \{s\}, \\ &\delta(s, \alpha x) = \delta(\delta(s, \alpha), x), \text{i.e.,} \cup_{s' \in \delta(s,\alpha)} \delta(s', x), \\ &\quad s \in S,\ \alpha \in X^*,\ x \in X. \end{aligned}$$

It is easy to see that for any $s_0, s_l \in S$ and any $x_0, \ldots, x_{l-1} \in X$, $s_l \in \delta(s_0, x_0 \ldots l_{l-1})$ if and only if there exist $s_1, \ldots, s_{l-1} \in S$ such that $s_{i+1} \in \delta(s_i, x_i)$, $i = 0, 1, \ldots, l-1$.

The domain of λ may be expanded to $S \times (X^* \cup X^\omega)$ as follows. For any state $s_0 \in S$ and any l input letters $x_0, x_1, \ldots, x_{l-1} \in X$, $\lambda(s_0, x_0 x_1 \ldots x_{l-1})$ is a subset of the set of all the sequences of length l over Y satisfying the

condition: for any y_0, y_1, ..., y_{l-1} in Y, $y_0y_1 \dots y_{l-1}$ is in $\lambda(s_0, x_0x_1 \dots x_{l-1})$ if and only if there exist s_i, $i = 1, 2, \dots, l-1$ in S such that

$$s_{i+1} \in \delta(s_i, x_i), \quad i = 0, 1, \dots, l-2$$

and

$$y_i \in \lambda(s_i, x_i), \quad i = 0, 1, \dots, l-1.$$

In the case of $l = 0$, $x_0x_1 \dots x_{l-1}$ and $y_0y_1 \dots y_{l-1}$ mean the empty word ε. For any state $s_0 \in S$ and any infinite input letters $x_0, x_1, \dots \in X$, $\lambda(s_0, x_0x_1 \dots)$ is a subset of Y^ω satisfying the condition: for any y_0, y_1, ... in Y, $y_0y_1 \dots$ is in $\lambda(s_0, x_0x_1 \dots)$ if and only if for any $l \geqslant 0$, $y_0y_1 \dots y_{l-1}$ is in $\lambda(s_0, x_0x_1 \dots x_{l-1})$.

Let $M = \langle X, Y, S, \delta, \lambda \rangle$ be a nondeterministic finite automaton, and $M' = \langle Y, X, S', \delta', \lambda' \rangle$ a finite automaton. For any states s in S and s' in S', (s', s) is called a *match pair* with delay τ or say that s' *τ-matches* s, if for any $l \geqslant \tau$, any $x_0, x_1, \dots, x_l, z_0, z_1, \dots, z_l$ in X and any $y_0, y_1, \dots, y_l$ in Y, $y_0y_1 \dots y_l \in \lambda(s, x_0x_1 \dots x_l)$ and $z_0z_1 \dots z_l = \lambda'(s', y_0y_1 \dots y_l)$ yield $z_\tau z_{\tau+1} \dots z_l = x_0x_1 \dots x_{l-\tau}$. M' is called an *inverse* with delay τ of M, if for any s in S and any s' in S', (s', s) is a match pair with delay τ. M' is called a *weak inverse* with delay τ of M, if for any s in S, there exists s' in S' such that (s', s) is a match pair with delay τ.

Let $M = \langle X, Y, S, \delta, \lambda \rangle$ be a finite automaton, and $M' = \langle Y, X, S', \delta', \lambda' \rangle$ a nondeterministic finite automaton. For any states s in S and s' in S', (s', s) is called a *match pair* with delay τ or say that s' *τ-matches* s, if for any $l \geqslant \tau$, any $x_0, x_1, \dots, x_l, z_0, z_1, \dots, z_l$ in X and any $y_0, y_1, \dots, y_l$ in Y, $y_0y_1 \dots y_l = \lambda(s, x_0x_1 \dots x_l)$ and $z_0z_1 \dots z_l \in \lambda'(s', y_0y_1 \dots y_l)$ yield $z_\tau z_{\tau+1} \dots z_l = x_0x_1 \dots x_{l-\tau}$. M' is called an *inverse* with delay τ of M, if for any s in S and any s' in S', (s', s) is a match pair with delay τ.

Let $M = \langle X, Y, S, \delta, \lambda \rangle$ be a finite automaton, and $M' = \langle X, Y', S', \delta', \lambda' \rangle$ a nondeterministic finite automaton. For any states s in S and s' in S', if for any $\alpha \in X^*$, $\lambda(s, \alpha)$ is in $\lambda'(s', \alpha)$, s' is said to be *stronger* than s, denoted by $s \prec s'$. If for any s in S, there exists s' in S' such that $s \prec s'$, we say that M' is *stronger* than M, denoted by $M \prec M'$. It is easy to verify that whenever M' is a finite automaton also, $s \prec s'$ if and only if $s \sim s'$, and the definition of $M \prec M'$ here coincides with the definition in Sect. 1.2 of Chap. 1.

6.2 Inverses of a Finite Automaton

For any finite automaton $M = \langle X, Y, S, \delta, \lambda \rangle$, let δ_M be a single-valued mapping from $2^S \times Y$ to 2^S, defined by

$$\delta_M(T,y)=\{\delta(s,x)\mid s\in T,x\in X,y=\lambda(s,x)\},$$
$$T\subseteq S,\ y\in Y.$$

Notice that $\delta_M(T,y)=\varnothing$ holds if $y\neq\lambda(s,x)$ holds for any $s\in T$ and any $x\in X$. Expand the domain of δ_M to $2^S\times Y^*$ as follows:

$$\delta_M(T,\varepsilon)=T,$$
$$\delta_M(T,\beta y)=\delta_M(\delta_M(T,\beta),y),$$
$$T\subseteq S,\ \beta\in Y^*,\ y\in Y.$$

It is easy to prove by induction on l that for any T_0,$T_l\subseteq S$, and any $y_0,\ldots,y_{l-1}\in Y$, $\delta_M(T_0,y_0\ldots y_{l-1})=T_l$ if and only if there exist subsets $T_1,\ldots,T_{l-1}$ of S such that $T_{i+1}=\delta_M(T_i,y_i)$, $i=0,1,\ldots,l-1$. It immediately follows that $\delta_M(T,\alpha\beta)=\delta_M(\delta_M(T,\alpha),\beta)$ holds for any $T\subseteq S$ and any α, $\beta\in Y^*$. Moreover, we can prove by induction on l the following assertion: for any $T_0\subseteq S$ and any $y_0,\ldots,y_{l-1}\in Y$, if $\delta_M(T_0,y_0\ldots y_{l-1})\neq\varnothing$, then for any s_l in $\delta_M(T_0,y_0\ldots y_{l-1})$ there exist s_0 in T_0 and α in X^* of length l such that $s_l=\delta(s_0,\alpha)$ and $y_0\ldots y_{l-1}=\lambda(s_0,\alpha)$. Conversely, it is evident that for any $T_0\subseteq S$ and any $y_0,\ldots,y_{l-1}\in Y$, if there exist s_0 in T_0 and α in X^* of length l such that $y_0\ldots y_{l-1}=\lambda(s_0,\alpha)$, then $\delta_M(T_0,y_0\ldots y_{l-1})\neq\varnothing$ and $\delta(s_0,\alpha)\in\delta_M(T_0,y_0\ldots y_{l-1})$.

Given an invertible finite automaton $M=\langle X,Y,S,\delta,\lambda\rangle$ with delay τ, without loss of generality, we assume $\lambda(S,X)=Y$.

Let $R_M=\{\lambda(s,\alpha)\mid s\in S,\alpha\in X^*\}$. Let $M_{\text{out}}=\langle Y,2^S,\delta_M,S,S_M\setminus\{\varnothing\}\rangle$ be a finite automaton recognizer, where $S_M=\{\delta_M(S,\beta)\mid\beta\in Y^*\}$. We prove that M_{out} recognizes R_M. Suppose $y_0\ldots y_{l-1}\in R_M$. Then there exist $s_0\in S$ and $\alpha\in X^*$ such that $y_0\ldots y_{l-1}=\lambda(s_0,\alpha)$. It follows that $\delta_M(S,y_0\ldots y_{l-1})\neq\varnothing$. Thus $\delta_M(S,y_0\ldots y_{l-1})\in S_M$. Conversely, suppose $\delta_M(S,y_0\ldots y_{l-1})\in S_M$. Then $\delta_M(S,y_0\ldots y_{l-1})\neq\varnothing$. Take arbitrarily a state s_l in $\delta_M(S,y_0\ldots y_{l-1})$. Then there exist s_0 in S and α in X^* of length l such that $s_l=\delta(s_0,\alpha)$ and $y_0\ldots y_{l-1}=\lambda(s_0,\alpha)$. It follows that $y_0\ldots y_{l-1}\in R_M$. We conclude that M_{out} recognizes R_M. Point out that if $\beta\in R_M$, then there exists $y\in Y$ such that $\beta y\in R_M$.

Let $M'=\langle Y,X,S',\delta',\lambda'\rangle$ be an inverse finite automaton with delay τ of M. We construct a partial finite automaton $\bar{M}'=\langle Y,X,\bar{S}',\bar{\delta}',\bar{\lambda}'\rangle$ from M' and M, where

$$\bar{S}'=\{\langle\delta_M(S,\beta),\delta'(s',\beta)\rangle\mid s'\in S',\beta\in Y^*,\delta_M(S,\beta)\neq\varnothing\},$$

$$\bar{\delta}'(\langle t,s'\rangle,y)=\begin{cases}\langle\delta_M(t,y),\delta'(s',y)\rangle, & \text{if } \delta_M(t,y)\neq\varnothing,\\ \text{undefined}, & \text{otherwise},\end{cases}$$

$$\bar{\lambda}'(\langle t, s'\rangle, y) = \begin{cases} \lambda'(s', y), & \text{if } \delta_M(t, y) \neq \varnothing, \\ \text{undefined}, & \text{otherwise}, \end{cases}$$

$$\langle t, s'\rangle \in \bar{S}',\ y \in Y.$$

$\bar{M}'$ is referred to as the *input restriction* of M' by M.

Clearly, for any state $\langle t, s'\rangle$ of $\bar{M}'$ and any $\beta \in Y^*$, $\bar{\delta}'(\langle t, s'\rangle, \beta) = \langle \delta_M(t, \beta),\ \delta'(s', \beta)\rangle$ and $\bar{\lambda}'(\langle t, s'\rangle, \beta) = \lambda'(s', \beta)$ whenever $\delta_M(t, \beta) \neq \varnothing$; $\bar{\delta}'(\langle t, s'\rangle, \beta)$ and $\bar{\lambda}'(\langle t, s'\rangle, \beta)$ are undefined whenever $\delta_M(t, \beta) = \varnothing$, where $\bar{\lambda}'(\langle t, s'\rangle, \beta)$ is defined if and only if each letter of $\bar{\lambda}'(\langle t, s'\rangle, \beta)$ is defined. Thus we have $\langle t, s'\rangle \prec s'$.

It is easy to show that $S_1 \subseteq S_2$ yields $\langle S_1, s'\rangle \prec \langle S_2, s'\rangle$, $s' \in S'$, $S_1, S_2 \in S_M \setminus \{\varnothing\}$.

States of $\bar{M}'$ are reachable from $\{\langle S, s'\rangle, s' \in S'\}$. In fact, from the definition of $\bar{S}'$, for any state $\bar{s}'$ of $\bar{M}'$, there exist $s' \in S'$ and $\beta \in Y^*$ such that $\bar{s}' = \langle \delta_M(S, \beta), \delta'(s', \beta)\rangle$ and $\delta_M(S, \beta) \neq \varnothing$. From $\langle \delta_M(S, \beta), \delta'(s', \beta)\rangle = \bar{\delta}'(\langle S, s'\rangle, \beta)$, for any state $\bar{s}'$ of $\bar{M}'$, there exist a state $\langle S, s'\rangle$ of $\bar{M}'$ and $\beta \in Y^*$ such that $\bar{s}' = \bar{\delta}'(\langle S, s'\rangle, \beta)$.

Let $\bar{S}'_\tau = \{\bar{\delta}'(\bar{s}', \beta) \mid \bar{s}' \in \bar{S}', \beta \in Y^*, |\beta| = \tau\}$. Clearly, $\bar{S}'_\tau$ is closed with respect to Y in $\bar{M}'$. We use $\bar{M}'_\tau$ to denote the partial finite subautomaton $\langle Y,\ X,\ \bar{S}'_\tau,\ \bar{\delta}'|_{\bar{S}'_\tau \times Y},\ \bar{\lambda}'|_{\bar{S}'_\tau \times Y}\rangle$ of $\bar{M}'$. $\bar{M}'_\tau$ is referred to as the *τ-successor* of $\bar{M}'$.

Let $M'' = \langle Y, X, S'', \delta'', \lambda''\rangle$ be an inverse finite automaton with delay τ. Similarly, from M'' and M we can construct $\bar{M}''$, the input restriction of M'' by M, and $\bar{M}''_\tau$, the τ-successor of $\bar{M}''$.

Lemma 6.2.1. *If M' and M'' are inverse finite automata with delay τ of M, then $\bar{M}'_\tau$ and $\bar{M}''_\tau$ are equivalent.*

Proof. Let $\bar{s}'$ be a state of $\bar{M}'_\tau$. Then there exist $\bar{s}'_0$ in $\bar{S}'$ and β_τ of length τ in Y^* such that $\bar{s}' = \bar{\delta}'(\bar{s}'_0, \beta_\tau)$. From the construction of $\bar{M}'$, there exist s' in S' and β of length $\geqslant \tau$ in R_M such that $\bar{s}' = \bar{\delta}'(\langle S, s'\rangle, \beta)$. Let $\bar{s}'' = \bar{\delta}''(\langle S, s''\rangle, \beta)$, where s'' is an arbitrarily fixed state in S''. We prove $\bar{s}' \sim \bar{s}''$. For any β_1 in Y^*, we have

$$\bar{\lambda}'(\langle S, s'\rangle, \beta\beta_1) = \begin{cases} \lambda'(s', \beta)\lambda'(\delta'(s', \beta), \beta_1), & \text{if } \delta_M(S, \beta\beta_1) \neq \varnothing, \\ \text{undefined}, & \text{otherwise}, \end{cases}$$

$$\bar{\lambda}''(\langle S, s''\rangle, \beta\beta_1) = \begin{cases} \lambda''(s'', \beta)\lambda''(\delta''(s'', \beta), \beta_1), & \text{if } \delta_M(S, \beta\beta_1) \neq \varnothing, \\ \text{undefined}, & \text{otherwise}, \end{cases}$$

where "undefined" means that some letter is undefined. Noticing that M_{out} recognizes R_M, when $\delta_M(S, \beta\beta_1) \neq \varnothing$, there exist s in S and α in X^* such

that $\lambda(s,\alpha)=\beta\beta_1$. Since M' and M'' are inverse finite automata with delay τ of M, we have

$$\begin{aligned}\lambda'(s',\beta)\lambda'(\delta'(s',\beta),\beta_1)&=\lambda'(s',\beta\beta_1)\\&=\lambda'(s',\lambda(s,\alpha))=x'_{-\tau}\dots x'_{-1}x_0x_1\dots x_{r-\tau}\end{aligned}$$

and

$$\begin{aligned}\lambda''(s'',\beta)\lambda''(\delta''(s'',\beta),\beta_1)&=\lambda''(s'',\beta\beta_1)\\&=\lambda''(s'',\lambda(s,\alpha))=x''_{-\tau}\dots x''_{-1}x_0x_1\dots x_{r-\tau},\end{aligned}$$

for some $x'_{-\tau}$, …, x'_{-1}, $x''_{-\tau}$, …, x''_{-1} in X, where $r=|\alpha|-1$, x_0, x_1, …, $x_{r-\tau}\in X$, and $x_0x_1\dots x_{r-\tau}$ is a prefix of α. From $|\beta|\geqslant\tau$, it follows that $\lambda'(\delta'(s',\beta),\beta_1)=\lambda''(\delta''(s'',\beta),\beta_1)$. Since $\bar\lambda'(\bar s',\beta_1)=\lambda'(\delta'(s',\beta),\beta_1)$ and $\bar\lambda''(\bar s'',\beta_1)=\lambda''(\delta''(s'',\beta),\beta_1)$, we obtain $\bar\lambda'(\bar s',\beta_1)=\bar\lambda''(\bar s'',\beta_1)$. When $\delta_M(S,\beta\beta_1)=\varnothing$, $\bar\lambda'(\langle S,s'\rangle,\beta\beta_1)$ and $\bar\lambda''(\langle S,s''\rangle,\beta\beta_1)$ are undefined. Since $\bar\delta'(\langle S,s'\rangle,\beta)=\bar s'$ and $\bar\delta''(\langle S,s''\rangle,\beta)=\bar s''$, $\bar\lambda'(\langle S,s'\rangle,\beta)$ and $\bar\lambda''(\langle S,s''\rangle,\beta)$ are defined. Therefore, $\bar\lambda'(\bar s',\beta_1)$ and $\bar\lambda''(\bar s'',\beta_1)$ are undefined. We conclude that $\bar s'$ and $\bar s''$ are equivalent.

From symmetry, for any $\bar s''$ in $\bar S''_\tau$ there exists $\bar s'$ in $\bar S'_\tau$ such that $\bar s'$ and $\bar s''$ are equivalent. Thus $\bar M'_\tau$ and $\bar M''_\tau$ are equivalent. □

Since M is invertible with delay τ, there exists a τ-order input-memory finite automaton which is an inverse with delay τ of M. Given a τ-order input-memory finite automaton, say M', assume that M' is an inverse with delay τ of M. Let $\bar M'$ be the input restriction of M' by M, and $\bar M'_\tau$ the τ-successor of $\bar M'$.

We use $T'(Y,\tau-1)$ to denote a labelled tree with level $\tau-1$ in which any vertex with level $<\tau$ emits $|Y|$ arcs labelled by different letters in Y, respectively. Such labels are called input labels of arcs. For each vertex v of $T'(Y,\tau-1)$, the sequence of labels of arcs in the unique path from the root of $T'(Y,\tau-1)$ to the vertex v is called the input label sequence of the vertex v. Let $T(Y,\tau-1)$ be the subtree of $T'(Y,\tau-1)$ satisfying the following condition: a vertex of $T'(Y,\tau-1)$ is a vertex of $T(Y,\tau-1)$ if and only if its input label sequence is in R_M. From the definition of R_M, $\beta y\in R_M$ yields $\beta\in R_M$ for any $\beta\in Y^*$ and any $y\in Y$. Therefore, $T(Y,\tau-1)$ is a tree indeed and the root of $T(Y,\tau-1)$ is the root of $T'(Y,\tau-1)$. Since for any $\beta\in Y^*$, $\beta\in R_M$ yields $\beta y\in R_M$ for some $y\in Y$, any vertex with level $<\tau$ of $T(Y,\tau-1)$ emits at least one arc. It is easy to see that in the case of $\lambda(S,X)=Y$, the root of $T(Y,\tau-1)$ emits $|Y|$ arcs of which input labels consist of different letters in Y. For each arc of $T(Y,\tau-1)$, we assign a letter in X to it as its output label. Different assignments of the output labels give different trees; the set of all such trees is denoted by $\mathcal{T}'(Y,X,\tau-1)$. Let $\mathcal{T}'_0(Y,X,\tau-1)$ be

a non-empty subset of $\mathcal{T}'(Y, X, \tau - 1)$. We "join" trees in $\mathcal{T}'_0(Y, X, \tau - 1)$ to the partial finite automaton $\bar{M}'_\tau$ to get a partial finite automaton, say $M''' = \langle Y, X, S''', \delta''', \lambda''' \rangle$, as follows. The state alphabet S''' of M''' is the union set of $\bar{S}'_\tau$ and all vertices with level $< \tau$ of all trees in $\mathcal{T}'_0(Y, X, \tau - 1)$; we assume that states in $\bar{S}'_\tau$ and such vertices are different from each other. For any state t in S''' and any y in Y, we define $\delta'''(t, y)$ and $\lambda'''(t, y)$ as follows. In the case of $t \in \bar{S}'_\tau$, define $\delta'''(t, y) = \bar{\delta}'(t, y)$ and $\lambda'''(t, y) = \bar{\lambda}'(t, y)$. In the case where t is a vertex with level $< \tau - 1$ of some tree in $\mathcal{T}'_0(Y, X, \tau - 1)$, if there is an arc with input label y emitted from t, then define $\delta'''(t, y) = t'$ and $\lambda'''(t, y) = x$, where t' is the terminal vertex of the arc and x is the output label of the arc; if there is no arc with input label y emitted from t, then $\delta'''(t, y)$ and $\lambda'''(t, y)$ are undefined. In the case where t is a vertex with level $\tau - 1$ of some tree in $\mathcal{T}'_0(Y, X, \tau - 1)$, let y_0, y_1, ..., $y_{\tau-2}$ be the input label sequence of arcs in the path from the root to t. If there is an arc with input label y emitted from t, then define $\delta'''(t, y) = \langle \delta_M(S, y_0 \dots y_{\tau-2} y), \langle y, y_{\tau-2}, \dots, y_0 \rangle \rangle$, a state of $\bar{M}'_\tau$, and $\lambda'''(t, y) = x$, where x is the output label of the arc; if there is no arc with input label y emitted from t, then $\delta'''(t, y)$ and $\lambda'''(t, y)$ are undefined. The states corresponding to roots of trees in $\mathcal{T}'_0(Y, X, \tau - 1)$ are called *root states* of M'''. We use $\mathcal{J}'(M, M')$ to denote the set of all such M'''.

For any M''' in $\mathcal{J}'(M, M')$, states of M''' are reachable from root states of M'''. In fact, for any state t of M''', in the case where t is a vertex of some tree with root t_0, it is evident that there exists $\beta \in Y^*$ such that $\delta'''(t_0, \beta) = t$. Suppose that $t \in \bar{S}'_\tau$. From the definition of $\bar{M}'_\tau$, there exist a state $\bar{s}'$ of $\bar{M}'$ and β_2 in Y^* such that $|\beta_2| = \tau$ and $\bar{\delta}'(\bar{s}', \beta_2) = t$. From the definition of $\bar{M}'$, there exist a state $\langle S, s' \rangle$ of $\bar{M}'$ and β_1 in Y^* such that $\bar{s}' = \langle \delta_M(S, \beta_1), \delta'(s', \beta_1) \rangle$. Thus $\bar{s}' = \bar{\delta}'(\langle S, s' \rangle, \beta_1)$. It follows that $\bar{\delta}'(\langle S, s' \rangle, \beta_1 \beta_2) = t$. Let $\beta_1 \beta_2 = \beta_3 \beta_4$ with $|\beta_3| = \tau$, and $\bar{\delta}'(\langle S, s' \rangle, \beta_3) = \bar{s}'_0$. Then $\bar{s}'_0$ is a state of $\bar{M}'_\tau$, $\bar{s}'_0 = \langle \delta_M(S, \beta_3), \langle y_{\tau-1}, \dots, y_0 \rangle \rangle$, and $\bar{\delta}'(\bar{s}'_0, \beta_4) = t$, where $\beta_3 = y_0 \dots y_{\tau-1}$. Let t_0 be any root state of M'''. From the construction of M''', noticing that $\beta_3 \in R_M$ and β_3 is an input label sequence of trees in $\mathcal{T}'(Y, X, \tau - 1)$, we have $\delta'''(t_0, \beta_3) = \bar{s}'_0$. This yields that $\delta'''(t_0, \beta_3 \beta_4) = \delta'''(\delta'''(t_0, \beta_3), \beta_4) = \delta'''(\bar{s}'_0, \beta_4) = \bar{\delta}'(\bar{s}'_0, \beta_4) = t$. We conclude that for any state t of M''' there exist a root state t_0 and β in Y^* such that $\delta'''(t_0, \beta) = t$.

Let $\tilde{M} = \langle Y, X, \tilde{S}, \tilde{\delta}, \tilde{\lambda} \rangle$ be a partial finite automaton such that $\tilde{\lambda}(s, y)$ is defined if and only if $\tilde{\delta}(s, y)$ is defined. Each state s_0 of $\tilde{M}$ determines a labelled tree with level $\tau - 1$, denoted by $T^{\tilde{M}}_{\tau-1}(s_0)$, which can be recurrently constructed from $\tilde{M}$ and s_0 as follows. We assign s_0 to the root of $T^{\tilde{M}}_{\tau-1}(s_0)$ temporarily. For any vertex with level $< \tau$ of $T^{\tilde{M}}_{\tau-1}(s_0)$ and any y in Y, let s be the label of the vertex. If $\tilde{\delta}(s, y)$ is defined, then an arc is emitted from the vertex and y, called the input label, and $\tilde{\lambda}(s, y)$, called the output label, are

assigned to the arc, and $\tilde{\delta}(s,y)$ is temporarily assigned to the terminal vertex of the arc. Finally, deleting all labels of vertices results the tree $T^{\tilde{M}}_{\tau-1}(s_0)$.

We use $\mathcal{J}(M,M')$ to denote the set of $\tilde{M}$ in $\mathcal{J}'(M,M')$ satisfying the following condition: for any states s_1 and s_2 of $\tilde{M}$, if s_1 is a root state and s_2 is a successor state of s_1, then there exists a root state s_0 of $\tilde{M}$ such that $T^{\tilde{M}}_{\tau-1}(s_2)$ is a subtree of $T^{\tilde{M}}_{\tau-1}(s_0)$.

Lemma 6.2.2. *For any partial finite automaton $\tilde{M} = \langle Y, X, \tilde{S}, \tilde{\delta}, \tilde{\lambda}\rangle$ in $\mathcal{J}(M,M')$ and any state $\tilde{s}$ of $\tilde{M}$, there exists a root state $\tilde{t}$ of $\tilde{M}$ such that $\tilde{s} \prec \tilde{t}$.*

Proof. Since states of $\tilde{M}$ are reachable from root states of $\tilde{M}$, for any state $\tilde{s}$ of $\tilde{M}$, there exist a root state $\tilde{t}_0$ of $\tilde{M}$ and $\beta \in Y^*$ such that $\tilde{\delta}(\tilde{t}_0,\beta) = \tilde{s}$. It is sufficient to prove the following proposition: for any state $\tilde{s}$ of $\tilde{M}$, any root state $\tilde{t}_0$ of $\tilde{M}$ and any $\beta \in Y^*$, if $\tilde{\delta}(\tilde{t}_0,\beta) = \tilde{s}$, then there exists a root state $\tilde{t}$ of $\tilde{M}$ such that $\tilde{s} \prec \tilde{t}$. We prove the proposition by induction on the length of β. *Basis* : $|\beta| = 0$. $\tilde{s}$ is a root state of $\tilde{M}$. Take $\tilde{t} = \tilde{s}$. Then $\tilde{s} \prec \tilde{t}$. *Induction step* : Suppose that for any state $\tilde{s}$ of $\tilde{M}$, any root state $\tilde{t}_0$ of $\tilde{M}$ and any β of length l in Y^*, if $\tilde{\delta}(\tilde{t}_0,\beta) = \tilde{s}$, then there exists a root state $\tilde{t}$ of $\tilde{M}$ such that $\tilde{s} \prec \tilde{t}$. To prove the case of $|\beta| = l+1$, suppose that $\tilde{\delta}(\tilde{t}_0,\beta) = \tilde{s}$, $|\beta| = l+1$ and $\tilde{t}_0$ is a root state of $\tilde{M}$. Let $\beta = y\beta_1$, $y \in Y$, and $\tilde{\delta}(\tilde{t}_0,y) = \tilde{s}_1$. Since $\tilde{M} \in \mathcal{J}(M,M')$, there exists a root state $\tilde{t}_1$ of $\tilde{M}$ such that $T^{\tilde{M}}_{\tau-1}(\tilde{s}_1)$ is a subtree of $T^{\tilde{M}}_{\tau-1}(\tilde{t}_1)$. We prove $\tilde{s}_1 \prec \tilde{t}_1$. Let β_2 be in Y^*. Suppose that $\tilde{\lambda}(\tilde{s}_1,\beta_2)$ is defined. In the case of $|\beta_2| \leqslant \tau$, since $T^{\tilde{M}}_{\tau-1}(\tilde{s}_1)$ is a subtree of $T^{\tilde{M}}_{\tau-1}(\tilde{t}_1)$, $\tilde{\lambda}(\tilde{t}_1,\beta_2)$ is defined and $\tilde{\lambda}(\tilde{s}_1,\beta_2) = \tilde{\lambda}(\tilde{t}_1,\beta_2)$. In the case of $|\beta_2| > \tau$, let $\beta_2 = \beta_3\beta_4$ with $|\beta_3| = \tau$. Then $\tilde{\lambda}(\tilde{s}_1,\beta_3)$ is defined. Since $T^{\tilde{M}}_{\tau-1}(\tilde{s}_1)$ is a subtree of $T^{\tilde{M}}_{\tau-1}(\tilde{t}_1)$, $\tilde{\lambda}(\tilde{t}_1,\beta_3)$ is defined and $\tilde{\lambda}(\tilde{s}_1,\beta_3) = \tilde{\lambda}(\tilde{t}_1,\beta_3)$. Let $\tilde{\delta}(\tilde{s}_1,\beta_3) = \tilde{s}_2$ and $\tilde{\delta}(\tilde{t}_1,\beta_3) = \tilde{s}_3$. Then $\tilde{\delta}(\tilde{t}_0,y\beta_3) = \tilde{s}_2$. From the construction of $\tilde{M}$, we have $\tilde{s}_3 = \langle\delta_M(S,\beta_3),\langle y_{\tau-1},\ldots,y_0\rangle\rangle$ and $\tilde{s}_2 = \langle\delta_M(S,y\beta_3),\langle y_{\tau-1},\ldots,y_0\rangle\rangle = \langle\delta_M(S_1,\beta_3),\langle y_{\tau-1},\ldots,y_0\rangle\rangle$, where $S_1 = \delta_M(S,y)$, and $\beta_3 = y_0\ldots y_{\tau-1}$. From $S_1 \subseteq S$, we have $\delta_M(S_1,\beta_3) \subseteq \delta_M(S,\beta_3)$. Notice that for any states $\langle S_2,s'\rangle$ and $\langle S_3,s'\rangle$ of $\tilde{M}$, if $S_2 \subseteq S_3$, then $\langle S_2,s'\rangle \prec \langle S_3,s'\rangle$. Thus $\tilde{s}_2 \prec \tilde{s}_3$. Since $\tilde{\lambda}(\tilde{s}_1,\beta_3\beta_4)$ is defined and $\tilde{s}_2 = \tilde{\delta}(\tilde{s}_1,\beta_3)$, $\tilde{\lambda}(\tilde{s}_2,\beta_4)$ is defined. From $\tilde{s}_2 \prec \tilde{s}_3$, $\tilde{\lambda}(\tilde{s}_3,\beta_4)$ is defined and $\tilde{\lambda}(\tilde{s}_2,\beta_4) = \tilde{\lambda}(\tilde{s}_3,\beta_4)$. Therefore, $\tilde{\lambda}(\tilde{s}_1,\beta_2) = \tilde{\lambda}(\tilde{s}_1,\beta_3)$ $\tilde{\lambda}(\tilde{s}_2,\beta_4) = \tilde{\lambda}(\tilde{t}_1,\beta_3)$ $\tilde{\lambda}(\tilde{s}_3,\beta_4) = \tilde{\lambda}(\tilde{t}_1,\beta_2)$. We conclude that $\tilde{s}_1 \prec \tilde{t}_1$. Since $\tilde{s}_1 \prec \tilde{t}_1$ and $\tilde{s} = \tilde{\delta}(\tilde{s}_1,\beta_1)$, $\tilde{\delta}(\tilde{t}_1,\beta_1)$ is defined. Denoting $\tilde{s}_4 = \tilde{\delta}(\tilde{t}_1,\beta_1)$, then $\tilde{s} \prec \tilde{s}_4$. Since $|\beta_1| = l$, from the induction hypothesis, there exists a root state $\tilde{t}_2$ of $\tilde{M}$ such that $\tilde{s}_4 \prec \tilde{t}_2$. Using $\tilde{s} \prec \tilde{s}_4$, we have $\tilde{s} \prec \tilde{t}_2$. □

Lemma 6.2.3. *For any finite automaton $M'' = \langle Y, X, S'', \delta'', \lambda''\rangle$, M'' is an inverse with delay τ of M if and only if there exists $\tilde{M}$ in $\mathcal{J}(M,M')$ such that $\bar{M}''$ and $\tilde{M}$ are equivalent, where $\bar{M}''$ is the input restriction of M'' by M.*

Proof. only if : Suppose that M'' is an inverse with delay τ of M. Let $T_{\bar{M}''} = \{T_{\tau-1}^{\bar{M}''}(\langle S, s''\rangle) \mid s'' \in S''\}$. Clearly, $T_{\bar{M}''} \subseteq \mathcal{T}'(Y, X, \tau - 1)$. Joining trees in $T_{\bar{M}''}$ to $\bar{M}'_\tau$ results a partial finite automaton, say $\tilde{M} = \langle Y, X, \tilde{S}, \tilde{\delta}, \tilde{\lambda}\rangle$, where $\bar{M}'_\tau$ is the τ-successor of $\bar{M}'$, and $\bar{M}'$ is the input restriction of M' by M. Clearly, $\tilde{M}$ is in $\mathcal{J}'(M, M')$.

Below, the root state of $\tilde{M}$ corresponding to the root of $T_{\tau-1}^{\bar{M}''}(\langle S, s''\rangle)$ is called the root state of $\tilde{M}$ corresponding to s''. (Root states of $\tilde{M}$ corresponding to s'' and s''' may be the same for different s'' and s'''.) For any s'' in S'', if $\tilde{s}$ is the root state of $\tilde{M}$ corresponding to s'', then the state $\tilde{s}$ of $\tilde{M}$ and the state $\langle S, s''\rangle$ of $\bar{M}''$ are equivalent. To prove this assertion, notice that the following fact is evident: for any β in Y^* with $|\beta| \leqslant \tau$, $\bar{\lambda}''(\langle S, s''\rangle, \beta)$ is defined if and only if $\tilde{\lambda}(\tilde{s}, \beta)$ is defined, and $\bar{\lambda}''(\langle S, s''\rangle, \beta) = \tilde{\lambda}(\tilde{s}, \beta)$ whenever they are defined. We prove the fact that $\bar{\delta}''(\langle S, s''\rangle, \beta) \sim \tilde{\delta}(\tilde{s}, \beta)$ holds for any β of length τ in R_M. Letting $\beta \in R_M$ and $|\beta| = \tau$, from the proof of Lemma 6.2.1, we have $\bar{\delta}''(\langle S, s''\rangle, \beta) \sim \bar{\delta}'(\langle S, s'\rangle, \beta)$ for any state s' of M'. From the construction of $\tilde{M}$, we have $\tilde{\delta}(\tilde{s}, \beta) = \langle \delta_M(S, \beta), \langle y_{\tau-1}, \ldots, y_0\rangle\rangle$, where $\beta = y_0 \ldots y_{\tau-1}$. Since $\bar{\delta}'(\langle S, s'\rangle, \beta) = \langle \delta_M(S, \beta), \langle y_{\tau-1}, \ldots, y_0\rangle\rangle$, from the construction of $\tilde{M}$, it follows that $\bar{\delta}'(\langle S, s'\rangle, \beta) \sim \tilde{\delta}(\tilde{s}, \beta)$. Therefore, $\bar{\delta}''(\langle S, s''\rangle, \beta) \sim \tilde{\delta}(\tilde{s}, \beta)$. Using the two facts mentioned above, since $\bar{\delta}''(\langle S, s''\rangle, \beta)$ and $\tilde{\delta}(\tilde{s}, \beta)$ are undefined for $\beta \notin R_M$, the state $\tilde{s}$ of $\tilde{M}$ and the state $\langle S, s''\rangle$ of $\bar{M}''$ are equivalent.

Using the above assertion, for any state $\langle S, s''\rangle$ of $\bar{M}''$, there exists a state $\tilde{s}$ of $\tilde{M}$ such that $\tilde{s} \sim \langle S, s''\rangle$. Since states of $\bar{M}''$ are reachable from $\{\langle S, s''\rangle, s'' \in S''\}$, for any state of $\bar{M}''$, there exists a state of $\tilde{M}$ such that they are equivalent. Conversely, for any root state $\tilde{s}$ of $\tilde{M}$, there exists a state $\langle S, s''\rangle$ of $\bar{M}''$ such that $\tilde{s} \sim \langle S, s''\rangle$. Since states of $\bar{M}'$ are reachable from $\{\langle S, s'\rangle, s' \in S'\}$, states of $\bar{M}'_\tau$ are reachable from

$$\{\bar{\delta}'(\langle S, s'\rangle, \beta), s' \in S', \beta \in R_M, |\beta| = \tau\}$$
$$= \{\langle \delta_M(S, y_0 \ldots y_{\tau-1}), \langle y_{\tau-1}, \ldots, y_0\rangle\rangle, y_0 \ldots y_{\tau-1} \in R_M\}.$$

From the construction of $\tilde{M}$, states in $\{\langle \delta_M(S, y_0 \ldots y_{\tau-1}), \langle y_{\tau-1}, \ldots, y_0\rangle\rangle, y_0 \ldots y_{\tau-1} \in R_M\}$ are reachable from root states of $\tilde{M}$. It follows that states of $\tilde{M}$ are reachable from its root states. Thus for any state of $\tilde{M}$, there exists a state of $\bar{M}''$ such that they are equivalent. We conclude $\bar{M}'' \sim \tilde{M}$.

We prove $\tilde{M} \in \mathcal{J}(M, M')$. Suppose that $\tilde{s}$ is a root state of $\tilde{M}$. From the assertion shown previously, there exists s'' in S'' such that $\tilde{s} \sim \langle S, s''\rangle$. For any y in Y, if $\tilde{\delta}(\tilde{s}, y)$ is defined, then $\tilde{\delta}(\tilde{s}, y) \sim \bar{\delta}''(\langle S, s''\rangle, y)$. Clearly, $\bar{\delta}''(\langle S, s''\rangle, y) = \langle \delta_M(S, y), \delta''(s'', y)\rangle \prec \langle S, \delta''(s'', y)\rangle$. Let $\tilde{t}$ be the root state of $\tilde{M}$ corresponding to $\delta''(s'', y)$. Then we have $\tilde{t} \sim \langle S, \delta''(s'', y)\rangle$. Thus $\tilde{\delta}(\tilde{s}, y) \prec \tilde{t}$. It follows that $T_{\tau-1}^{\tilde{M}}(\tilde{\delta}(\tilde{s}, y))$ is a subtree of $T_{\tau-1}^{\tilde{M}}(\tilde{t})$. Therefore, $\tilde{M} \in \mathcal{J}(M, M')$.

if: Suppose that $\tilde{M} = \langle Y, X, \tilde{S}, \tilde{\delta}, \tilde{\lambda}\rangle \in \mathcal{J}(M, M')$ and $\bar{M}'' \sim \tilde{M}$. For any s in S, any s'' in S'' and any $\alpha = x_0 \dots x_l$, let $y_0 \dots y_l = \lambda(s, \alpha)$ and $z_0 \dots z_l = \lambda''(s'', y_0 \dots y_l)$, where $x_0, \dots, x_l \in X$, $y_0, \dots, y_l \in Y$, and $z_0, \dots, z_l \in X$. We prove $z_\tau \dots z_l = x_0 \dots x_{l-\tau}$ if $l \geqslant \tau$. Suppose that $l \geqslant \tau$. Let $\beta_1 = y_0 \dots y_{\tau-1}$, $\beta_2 = y_\tau \dots y_l$, and $\bar{s}'' = \langle S, s''\rangle$. Since $\bar{M}''$ and $\tilde{M}$ are equivalent, there exists $\tilde{s}$ in $\tilde{S}$ such that $\tilde{s} \sim \bar{s}''$. Noticing that domains of $\bar{\delta}''$ and $\bar{\lambda}''$ are the same and that domains of $\tilde{\delta}$ and $\tilde{\lambda}$ are the same, for any β in Y^*, $\bar{\lambda}''(\bar{s}'', \beta)$ is defined if and only if $\tilde{\lambda}(\tilde{s}, \beta)$ is defined, and $\bar{\lambda}''(\bar{s}'', \beta) = \tilde{\lambda}(\tilde{s}, \beta)$ holds whenever they are defined. Since $\beta_1\beta_2$, i.e., $\lambda(s, \alpha)$, is in R_M, $\bar{\lambda}''(\bar{s}'', \beta_1\beta_2)$ is defined. It follows that $\bar{\lambda}''(\bar{s}'', \beta_1\beta_2) = \tilde{\lambda}(\tilde{s}, \beta_1\beta_2)$. From $\lambda''(s'', \beta_1\beta_2) = \bar{\lambda}''(\bar{s}'', \beta_1\beta_2)$, it follows that $\tilde{\lambda}(\tilde{s}, \beta_1)\ \tilde{\lambda}(\tilde{\delta}(\tilde{s}, \beta_1), \beta_2) = \lambda''(s'', \beta_1\beta_2) = z_0 \dots z_l$. Therefore, we have $\tilde{\lambda}(\tilde{\delta}(\tilde{s}, \beta_1), \beta_2) = z_\tau \dots z_l$. On the other hand, from the construction of $\tilde{M}$, $\tilde{\delta}(\tilde{s}, \beta_1) = \langle \bar{s}, \langle y_{\tau-1}, \dots, y_0\rangle\rangle$ for some state $\langle \bar{s}, \langle y_{\tau-1}, \dots, y_0\rangle\rangle$ in $\bar{S}'_\tau$. Thus we have $\tilde{\lambda}(\tilde{\delta}(\tilde{s}, \beta_1), \beta_2) = \bar{\lambda}'(\langle \bar{s}, \langle y_{\tau-1}, \dots, y_0\rangle\rangle, \beta_2) = \lambda'(\langle y_{\tau-1}, \dots, y_0\rangle, \beta_2)$. Since M' is an inverse with delay τ of M, for any state s' of M', we have $\lambda'(s', \beta_1\beta_2) = x_{-\tau} \dots x_{-1}x_0 \dots x_{l-\tau}$ for some $x_{-\tau}, \dots, x_{-1}$ in X. Since M' is a τ-order input-memory finite automaton, it follows that $\lambda'(\langle y_{\tau-1}, \dots, y_0\rangle, \beta_2) = x_0 \dots x_{l-\tau}$. Thus $z_\tau \dots z_l = \tilde{\lambda}(\tilde{\delta}(\tilde{s}, \beta_1), \beta_2) = \lambda'(\langle y_{\tau-1}, \dots, y_0\rangle, \beta_2) = x_0 \dots x_{l-\tau}$. Therefore, M'' is an inverse with delay τ of M. □

For any partial finite automaton $\tilde{M} = \langle Y, X, \tilde{S}, \tilde{\delta}, \tilde{\lambda}\rangle$ in $\mathcal{J}(M, M')$, each root state $\tilde{t}$ determines a compatible set $C(\tilde{t}) = \{\tilde{s} \mid \tilde{s} \in \tilde{S}, \tilde{s} \prec \tilde{t}\}$. For any two different root states $\tilde{t}$ and $\tilde{t}'$ of $\tilde{M}$, if $\tilde{t} \prec \tilde{t}'$, then trees with roots $\tilde{t}$ and $\tilde{t}'$ are the same. Thus each $C(\tilde{t})$ only contains one root state. From Lemma 6.1.1 (b), $C(\tilde{t})$ is compatible. We use $C(\tilde{M})$ to denote the set $\{C(\tilde{t}) \mid \tilde{t}$ is a root state of $\tilde{M}\}$. For any $C(\tilde{t})$ in $C(\tilde{M})$ and any y in Y, using Lemma 6.1.1 (a), if $\tilde{\delta}(C(\tilde{t}), y) \neq \varnothing$, then $\tilde{s} \prec \tilde{\delta}(\tilde{t}, y)$ holds for any $\tilde{s}$ in $\tilde{\delta}(C(\tilde{t}), y)$. Since $\tilde{M}$ is in $\mathcal{J}(M, M')$, from Lemma 6.2.2, there exists a root state $\tilde{t}_1$ of $\tilde{M}$ such that $\tilde{\delta}(\tilde{t}, y) \prec \tilde{t}_1$. It follows that $\tilde{s} \prec \tilde{t}_1$ holds for any $\tilde{s}$ in $\tilde{\delta}(C(\tilde{t}), y)$. Thus $\tilde{\delta}(C(\tilde{t}), y) \subseteq C(\tilde{t}_1)$. Moreover, from Lemma 6.2.2, for any state $\tilde{s}$ of $\tilde{M}$, there exists a root state $\tilde{t}$ of $\tilde{M}$ such that $\tilde{s} \in C(\tilde{t})$. This yields $\cup_{C \in C(\tilde{M})} C = \tilde{S}$. Therefore, for any $C_1, \dots, C_k$, (it is not necessary that $i \neq j$ implies $C_i \neq C_j$,) if the set $\{C_1, \dots, C_k\}$ equals $C(\tilde{M})$, then the sequence $C_1, \dots, C_k$ is a closed compatible family of $\tilde{M}$. It is easy to see that in case of $\lambda(S, X) = Y$, each partial finite automaton in $\mathcal{M}(C_1, \dots, C_k)$ is a finite automaton whenever $\{C_1, \dots, C_k\} = C(\tilde{M})$. We use $\mathcal{M}_0(\tilde{M})$ to denote the union set of all $\mathcal{M}(C_1, \dots, C_k)$, $C_1, \dots, C_k$ ranging over all sequences so that the set $\{C_1, \dots, C_k\}$ is equal to $C(\tilde{M})$.

Theorem 6.2.1. *If $M = \langle X, Y, S, \delta, \lambda\rangle$ is an invertible finite automaton with delay τ and $\lambda(S, X) = Y$, then the set of all inverse finite automata with delay τ of M is $\cup_{\tilde{M} \in \mathcal{J}(M, M')} \mathcal{M}_0(\tilde{M})$ up to equivalence, i.e., for any*

finite automaton $M'' = \langle Y, X, S'', \delta'', \lambda'' \rangle$, M'' *is an inverse with delay* τ *of* M, *if and only if there exist* $\tilde{M}$ *in* $\mathcal{J}(M, M')$ *and* M''' *in* $\mathcal{M}_0(\tilde{M})$ *such that* $M''' \sim M''$.

Proof. only if: Suppose that M'' is an inverse finite automaton with delay τ of M. In the proof of the *only if* part of Lemma 6.2.3, we construct a partial finite automaton $\tilde{M} = \langle Y, X, \tilde{S}, \tilde{\delta}, \tilde{\lambda} \rangle$ in $\mathcal{J}(M, M')$ such that $\tilde{M}$ and $\bar{M}''$ are equivalent, where $\bar{M}''$ is the input restriction of M'' by M. Especially, any state $\langle S, s'' \rangle$ of $\bar{M}''$ and the root state of $\tilde{M}$ corresponding to s'' are equivalent; any root state of $\tilde{M}$ is a root state of $\tilde{M}$ corresponding to s'' for some state s'' in S'', and for any s'' in S'' there exists a root state of $\tilde{M}$ corresponding to s''.

Let $S'' = \{s''_1, \ldots, s''_h\}$ and $\bar{C}_i = \{\bar{s} \mid \bar{s} \text{ is a state of } \bar{M}'', \bar{s} \prec \langle S, s''_i \rangle\}$, $i = 1, \ldots, h$. Noticing that $\langle S_1, s'' \rangle \prec \langle S, s'' \rangle$ and $\bar{\delta}''(\langle S, s'' \rangle, y) \prec \langle S, \delta''(s'', y) \rangle$, using Lemma 6.1.1, it is easy to show that the sequence $\bar{C}_1, \ldots, \bar{C}_h$ is a closed compatible family of $\bar{M}''$.

Let $C_i = \{\tilde{s} \mid \tilde{s} \text{ is a state of } \tilde{M}, \text{ there exists } \bar{s} \in \bar{C}_i \text{ such that } \tilde{s} \sim \bar{s}\}$, $i = 1, \ldots, h$. We prove that for any root state $\tilde{t}$ of $\tilde{M}$ and any i, $1 \leqslant i \leqslant h$, if $\tilde{t}$ is a root state of $\tilde{M}$ corresponding to s''_i, then $C(\tilde{t}) = C_i$. Suppose that $\tilde{t}$ is a root state of $\tilde{M}$ corresponding to s''_i. Then $\tilde{t}$ is equivalent to the state $\langle S, s''_i \rangle$ of $\bar{M}''$. Therefore, for any $\tilde{s} \in \tilde{S}$, $\tilde{s} \in C(\tilde{t})$ if and only if $\tilde{s} \prec \tilde{t}$, if and only if $\tilde{s} \prec \langle S, s''_i \rangle$, if and only if there exists a state $\bar{s}$ of $\bar{M}''$ such that $\tilde{s} \sim \bar{s}$ and $\bar{s} \prec \langle S, s''_i \rangle$, if and only if there exists $\bar{s}$ in $\bar{C}_i$ such that $\tilde{s} \sim \bar{s}$, if and only if $\tilde{s} \in C_i$. It follows that $C(\tilde{t}) = C_i$. Since for any s'' in S'' there exists a root state of $\tilde{M}$ corresponding to s'', we have $\{C_1, \ldots, C_h\} \subseteq C(\tilde{M})$. Since any root state of $\tilde{M}$ is a root state of $\tilde{M}$ corresponding to s'' for some state s'' in S'', we have $C(\tilde{M}) \subseteq \{C_1, \ldots, C_h\}$. Thus $\{C_1, \ldots, C_h\} = C(\tilde{M})$. It follows that $C_1, \ldots, C_h$ is a closed compatible family of $\tilde{M}$.

From $\tilde{M} \sim \bar{M}''$, it is easy to prove that $\bar{C}_i = \{\bar{s} \mid \bar{s} \text{ is a state of } \bar{M}'', \text{ there exists } \tilde{s} \in C_i \text{ such that } \tilde{s} \sim \bar{s}\}$, $i = 1, \ldots, h$. We prove that $\bar{\delta}''(\bar{C}_i, y) \subseteq \bar{C}_j$ if and only if $\tilde{\delta}(C_i, y) \subseteq C_j$. Suppose $\bar{\delta}''(\bar{C}_i, y) \subseteq \bar{C}_j$. For any $\tilde{s} \in C_i$, there exists $\bar{s}$ in $\bar{C}_i$ such that $\tilde{s} \sim \bar{s}$. When $\tilde{\delta}(\tilde{s}, y)$ is defined, from $\tilde{s} \sim \bar{s}$, $\bar{\delta}''(\bar{s}, y)$ is defined and $\tilde{\delta}(\tilde{s}, y) \sim \bar{\delta}''(\bar{s}, y)$. From $\bar{\delta}''(\bar{C}_i, y) \subseteq \bar{C}_j$, we have $\bar{\delta}''(\bar{s}, y) \in \bar{C}_j$. It follows that $\tilde{\delta}(\tilde{s}, y) \in C_j$. Thus $\tilde{\delta}(C_i, y) \subseteq C_j$. Conversely, from the symmetry, $\tilde{\delta}(C_i, y) \subseteq C_j$ implies $\bar{\delta}''(\bar{C}_i, y) \subseteq \bar{C}_j$. We conclude that $\bar{\delta}''(\bar{C}_i, y) \subseteq \bar{C}_j$ if and only if $\tilde{\delta}(C_i, y) \subseteq C_j$.

Construct $\bar{M}''' = \langle Y, X, \{\bar{c}_1, \ldots, \bar{c}_h\}, \bar{\delta}''', \bar{\lambda}''' \rangle$, where

$$\begin{aligned} &\bar{\delta}'''(\bar{c}_i, y) = \bar{c}_j, \\ &\bar{\lambda}'''(\bar{c}_i, y) = \bar{\lambda}''(\langle S, s''_i \rangle, y), \\ &\qquad i = 1, \ldots, h, \ y \in Y, \end{aligned}$$

j is the integer satisfying $s_j'' = \delta''(s_i'', y)$. Using Lemma 6.1.1 (a), $\bar{s}'' \prec \langle S, s_i'' \rangle$ implies $\bar{\delta}''(\bar{s}'', y) \prec \langle \delta_M(S, y), \delta''(s_i'', y) \rangle$, therefore, implies $\bar{\delta}''(\bar{s}'', y) \prec \langle S, s_j'' \rangle$, that is, $\bar{\delta}''(\bar{s}'', y) \in \bar{C}_j$. It follows that $\bar{\delta}''(\bar{C}_i, y) \subseteq \bar{C}_j$. Since $\lambda(S, X) = Y$, for any $y \in Y$, $\bar{\delta}''(\langle S, s_i'' \rangle, y)$ and $\bar{\lambda}''(\langle S, s_i'' \rangle, y)$ are defined. It follows that $\bar{\delta}''(\bar{C}_i, y) \neq \varnothing$. Noticing $\langle S, s_i'' \rangle \in \bar{C}_i$, we obtain $\bar{M}''' \in \mathcal{M}(\bar{C}_1, \ldots, \bar{C}_h)$. Defining $\varphi(\bar{c}_i) = s_i''$, $i = 1, \ldots, h$, using $\bar{\lambda}''(\langle S, s_i'' \rangle, y) = \lambda''(s_i'', y)$, it is easy to verify that φ is an isomorphism from $\bar{M}'''$ to M''. Therefore, $\bar{M}'''$ and M'' are isomorphic.

Let $M''' = \langle Y, X, \{c_1, \ldots, c_h\}, \delta''', \lambda''' \rangle$, where $\delta'''(c_i, y) = c_j$ for the integer j satisfying $\bar{\delta}'''(\bar{c}_i, y) = \bar{c}_j$, and $\lambda'''(c_i, y) = \bar{\lambda}'''(\bar{c}_i, y)$. Since $\bar{\delta}'''(\bar{c}_i, y) = \bar{c}_j$ implies $\bar{\delta}''(\bar{C}_i, y) \subseteq \bar{C}_j$, $\delta'''(c_i, y) = c_j$ implies $\bar{\delta}''(\bar{C}_i, y) \subseteq \bar{C}_j$. It follows that $\delta'''(c_i, y) = c_j$ implies $\tilde{\delta}(C_i, y) \subseteq C_j$. Let $\tilde{t}$ be a root state corresponding to s_i''. Then $\tilde{t} \sim \langle S, s_i'' \rangle$. From $\langle S, s_i'' \rangle \in \bar{C}_i$, we have $\tilde{t} \in C_i$. Since $\bar{\delta}''(\langle S, s_i'' \rangle, y)$ is defined, $\tilde{\delta}(\tilde{t}, y)$ is defined. It follows that $\tilde{\delta}(C_i, y) \neq \varnothing$. From $\tilde{t} \sim \langle S, s_i'' \rangle$, we have $\bar{\lambda}''(\langle S, s_i'' \rangle, y) = \tilde{\lambda}(\tilde{t}, y)$. Since $\bar{\lambda}'''(\bar{c}_i, y) = \bar{\lambda}''(\langle S, s_i'' \rangle, y)$, we have $\lambda'''(c_i, y) = \bar{\lambda}'''(\bar{c}_i, y) = \tilde{\lambda}(\tilde{t}, y)$. Thus M''' is in $\mathcal{M}(C_1, \ldots, C_h)$. Clearly, M''' and $\bar{M}'''$ are isomorphic. Thus M''' and M'' are isomorphic. Therefore, M''' and M'' are equivalent. We conclude that for any inverse finite automaton M'' with delay τ of M, there exist $\tilde{M}$ in $\mathcal{J}(M, M')$ and M''' in $\mathcal{M}_0(\tilde{M})$ such that $M''' \sim M''$.

if : Suppose that $M'' \sim M'''$ for some $M''' \in \mathcal{M}_0(\tilde{M})$, where $\tilde{M} \in \mathcal{J}(M, M')$. Then there exists a closed compatible family $C_1, \ldots, C_h$ of $\tilde{M}$ such that $\{C_1, \ldots, C_h\} = C(\tilde{M})$ and $M''' \in \mathcal{M}(C_1, \ldots, C_h)$. Thus $M''' = \langle Y, X, \{c_1, \ldots, c_h\}, \delta''', \lambda''' \rangle$, where $\delta'''(c_i, y) = c_j$ for some j satisfying the condition $\varnothing \neq \tilde{\delta}(C_i, y) \subseteq C_j$, and $\lambda'''(c_i, y) = \tilde{\lambda}(\tilde{s}, y)$, $\tilde{s}$ being the root state in C_i. For any state s of M, any state c_i of M''', $1 \leqslant i \leqslant h$, and any x_0, …, x_l in X, let $\lambda(s, x_0 \ldots x_l) = y_0 \ldots y_l$ and $\lambda'''(c_i, y_0 \ldots y_l) = z_0 \ldots z_l$. We prove $z_\tau \ldots z_l = x_0 \ldots x_{l-\tau}$ in case of $\tau \leqslant l$. Suppose $\tau \leqslant l$ and let $\tilde{s}$ be the root state in C_i. Since $y_0 \ldots y_l \in R_M$, we have $\delta_M(S, y_0 \ldots y_l) \neq \varnothing$. It follows that $\tilde{\lambda}(\tilde{s}, y_0 \ldots y_l)$ is defined. Thus we have

$$\begin{aligned}
\lambda'''(c_i, \lambda(s, x_0 \ldots x_l)) &= \tilde{\lambda}(\tilde{s}, y_0 \ldots y_l) \\
&= \tilde{\lambda}(\tilde{s}, y_0 \ldots y_{\tau-1}) \tilde{\lambda}(\tilde{\delta}(\tilde{s}, y_0 \ldots y_{\tau-1}), y_\tau \ldots y_l) \\
&= \tilde{\lambda}(\tilde{s}, y_0 \ldots y_{\tau-1}) \bar{\lambda}'(\langle \delta_M(S, y_0 \ldots y_{\tau-1}), \langle y_{\tau-1}, \ldots, y_0 \rangle \rangle, y_\tau \ldots y_l) \\
&= \tilde{\lambda}(\tilde{s}, y_0 \ldots y_{\tau-1}) \lambda'(\langle y_{\tau-1}, \ldots, y_0 \rangle, y_\tau \ldots y_l) \\
&= \tilde{\lambda}(\tilde{s}, y_0 \ldots y_{\tau-1}) x_0 \ldots x_{l-\tau}.
\end{aligned}$$

It follows that $z_\tau \ldots z_l = x_0 \ldots x_{l-\tau}$. Thus M''' is an inverse finite automaton with delay τ of M. Since $M'' \sim M'''$, M'' is an inverse with delay τ of M.

□

Joining all trees in $\mathcal{T}'(Y, X, \tau - 1)$ to the partial finite automaton $\bar{M}'_\tau$, we get a partial finite automaton, denoted by $\tilde{M}_{\max}$. It is easy to see that $\tilde{M}_{\max} \in \mathcal{J}(M, M')$.

Theorem 6.2.2. *If $M = \langle X, Y, S, \delta, \lambda\rangle$ is an invertible finite automaton with delay τ and $\lambda(S, X) = Y$, then a finite automaton $M'' = \langle Y, X, S'', \delta'', \lambda''\rangle$ is an inverse with delay τ of M if and only if there exist a finite automaton M'''' in $\mathcal{M}_0(\tilde{M}_{\max})$ and a finite subautomaton M''' of M'''' such that $M''' \sim M''$.*

Proof. if : Suppose that $M'''' \in \mathcal{M}_0(\tilde{M}_{\max})$ and $M''' \sim M''$ for some finite subautomaton M''' of M''''. From $\lambda(S, X) = Y$, M'''' is a finite automaton. Since $\tilde{M}_{\max} \in \mathcal{J}(M, M')$, from Theorem 6.2.1, M'''' is an inverse with delay τ of M. It follows that M''' is an inverse with delay τ of M. From $M''' \sim M''$, M'' is an inverse with delay τ of M.

only if : Suppose that M'' is an inverse with delay τ of M. From Theorem 6.2.1, there exist $\tilde{M}$ in $\mathcal{J}(M, M')$ and M''' in $\mathcal{M}_0(\tilde{M})$ such that $M''' \sim M''$. Clearly, $\tilde{M}$ is a partial finite subautomaton of $\tilde{M}_{\max}$ and any root state of $\tilde{M}$ is a root state of $\tilde{M}_{\max}$. Let $M''' = \langle Y, X, \{c_1, \ldots, c_h\}, \delta''', \lambda'''\rangle \in \mathcal{M}(C_1, \ldots, C_h)$ for some closed compatible family $C_1, \ldots, C_h$ of $\tilde{M}$ with $\{C_1, \ldots, C_h\} = C(\tilde{M})$. Let $\tilde{t}_i$ be the root state in C_i, $i = 1, \ldots, h$. It is easy to prove that there exists a closed compatible family $C'_1, \ldots, C'_{h'}$ of $\tilde{M}_{\max}$ with $\{C'_1, \ldots, C'_{h'}\} = C(\tilde{M}_{\max})$ such that $h \leqslant h'$ and $\tilde{t}_i$ is the root state in C'_i for any i, $1 \leqslant i \leqslant h$. Clearly, $C_i = \{\tilde{s} \mid \tilde{s}$ is a state of $\tilde{M}, \tilde{s} \prec \tilde{t}_i\}$ and $C'_i = \{\tilde{s} \mid \tilde{s}$ is a state of $\tilde{M}_{\max}, \tilde{s} \prec \tilde{t}_i\}$, $i = 1, \ldots, h$. Since $\tilde{M}$ is a partial finite subautomaton of $\tilde{M}_{\max}$, we have $C_i \subseteq C'_i$, $i = 1, \ldots, h$. And for any $y \in Y$, $\tilde{\delta}(C_i, y) \subseteq C_j$ implies $\tilde{\delta}_{\max}(\tilde{t}_i, y) = \tilde{\delta}(\tilde{t}_i, y) \in C_j \subseteq C'_j$, therefore, using Lemma 6.1.1 (a), $\tilde{\delta}(C_i, y) \subseteq C_j$ implies $\tilde{\delta}_{\max}(C'_i, y) \subseteq C'_j$, $i, j = 1, \ldots, h$, where $\tilde{\delta}$ and $\tilde{\delta}_{\max}$ are the next functions of $\tilde{M}$ and $\tilde{M}_{\max}$, respectively. Thus we can construct $M'''' = \langle Y, X, \{c_1, \ldots, c_{h'}\}, \delta'''', \lambda''''\rangle$ in $\mathcal{M}(C'_1, \ldots, C'_{h'})$ such that M''' is a finite subautomaton of M''''. Clearly, $M'''' \in \mathcal{M}_0(\tilde{M}_{\max})$. This completes the proof of the theorem. □

Noticing that the condition "there exist a finite automaton M'''' in $\mathcal{M}_0(\tilde{M}_{\max})$ and a finite subautomaton M''' of M'''' such that $M''' \sim M''$" is equivalent to the condition "there exists a finite automaton M'''' in $\mathcal{M}_0(\tilde{M}_{\max})$ such that $M'' \prec M''''$ ", the theorem can be restated as the following corollary.

Corollary 6.2.1. *If $M = \langle X, Y, S, \delta, \lambda\rangle$ is an invertible finite automaton with delay τ and $\lambda(S, X) = Y$, then a finite automaton $M'' = \langle Y, X, S'', \delta'', \lambda''\rangle$ is an inverse with delay τ of M if and only if there exists a finite automaton M''' in $\mathcal{M}_0(\tilde{M}_{\max})$ such that $M'' \prec M'''$.*

We deal with the case $|X| = |Y|$. In this case, from Theorem 1.4.6, since M is invertible with delay τ, we have $R_M = Y^*$. Thus we can construct a finite automaton recognizer $M_{\text{out}} = \langle Y, \{S\}, \delta_M, S, \{S\}\rangle$ to recognize R_M, where $\delta_M(S, y) = S$ for any y in Y. It is easy to see that both $\bar{M}'$ and $\bar{M}'_\tau$ are finite automata. Therefore, the partial finite automaton $\tilde{M}_{\max}$ is a finite automaton. Using Lemma 6.1.2, it follows that each finite automaton in $\mathcal{M}_0(\tilde{M}_{\max})$ is equivalent to $\tilde{M}_{\max}$. Therefore, finite automata in $\mathcal{M}_0(\tilde{M}_{\max})$ are equivalent to each other. From Corollary 6.2.1, we obtain the following corollary.

Corollary 6.2.2. *If $M = \langle X, Y, S, \delta, \lambda\rangle$ is an invertible finite automaton with delay τ and $|X| = |Y|$, then a finite automaton $M'' = \langle Y, X, S'', \delta'', \lambda''\rangle$ is an inverse with delay τ of M if and only if $M'' \prec M'''$, where M''' is any finite automaton in $\mathcal{M}_0(\tilde{M}_{\max})$ or $M''' = \tilde{M}_{\max}$.*

Corollary 6.2.2 means that in the case of $|X| = |Y|$, the finite automaton $\tilde{M}_{\max}$ and each finite automaton in $\mathcal{M}_0(\tilde{M}_{\max})$ are a "universal" inverse with delay τ of M. And in the general case of $|X| \leqslant |Y|$, Theorem 6.2.2 means that all finite automata in the set $\mathcal{M}_0(\tilde{M}_{\max})$, not one finite automaton in it, constitute the "universal" inverses with delay τ of M. But a nondeterministic finite automaton can be constructed as follows, which is a "universal" inverse with delay τ of the finite automaton M.

From $\tilde{M}_{\max} = \langle Y, X, \tilde{S}, \tilde{\delta}, \tilde{\lambda}\rangle$, we construct a nondeterministic finite automaton $M''' = \langle Y, X, C(\tilde{M}_{\max}), \delta''', \lambda'''\rangle$ as follows. For any T in $C(\tilde{M}_{\max})$ and any y in Y, define $\delta'''(T, y) = \{W \mid W \in C(\tilde{M}_{\max}), \tilde{\delta}(T, y) \subseteq W\}$ and $\lambda'''(T, y) = \{\tilde{\lambda}(\tilde{s}, y)\}$, where $\tilde{s}$ is the root state of $\tilde{M}_{\max}$ in T.

Theorem 6.2.3. *The nondeterministic finite automaton M''' is an inverse with delay τ of the finite automaton M.*

Proof. Let C_0 be a state of M''', and s a state of M. We prove that C_0 τ-matches s. That is, for any $l \geqslant \tau$, any $x_0, x_1, \ldots, x_l, z_0, z_1, \ldots, z_l$ in X and any $y_0, y_1, \ldots, y_l$ in Y, $y_0\ y_1 \ldots y_l = \lambda(s, x_0x_1 \ldots x_l)$ and $z_0z_1 \ldots z_l \in \lambda'''(C_0, y_0y_1 \ldots y_l)$ imply $z_\tau z_{\tau+1} \ldots z_l = x_0x_1 \ldots x_{l-\tau}$. Suppose $y_0y_1 \ldots y_l = \lambda(s, x_0x_1 \ldots x_l)$ and $z_0z_1 \ldots z_l \in \lambda'''(C_0, y_0y_1 \ldots y_l)$, where $l \geqslant \tau$, $x_0, x_1, \ldots, x_l$, $z_0, z_1, \ldots, z_l \in X$, and $y_0, y_1, \ldots, y_l \in Y$. We prove $z_\tau z_{\tau+1} \ldots z_l = x_0x_1 \ldots x_{l-\tau}$. From the definition of λ''', there exist states $C_1, \ldots, C_l$ of M''' such that $C_{j+1} \in \delta'''(C_j, y_j)$ holds for $j = 0, 1, \ldots, l-1$ and $z_j \in \lambda'''(C_j, y_j)$ holds for $j = 0, 1, \ldots, l$. From the construction of M''', it follows that $\tilde{\delta}(C_j, y_j) \subseteq C_{j+1}$ holds for $j = 0, 1, \ldots, l-1$ and $z_j = \tilde{\lambda}(\tilde{t}_j, y_j)$ holds for $j = 0, 1, \ldots, l$, where $\tilde{t}_j$ is the root state of $\tilde{M}$ in C_j, $j = 0, 1, \ldots, l$. Let $\tilde{s}_0 = \tilde{t}_0$. Since $y_0 \ldots y_l \in R_M$, we can recursively define $\tilde{s}_{j+1} = \tilde{\delta}(\tilde{s}_j, y_j)$, $j = 0, 1, \ldots, l-1$. Since $y_0 \ldots y_l \in R_M$, $\tilde{\lambda}(\tilde{s}_j, y_j)$ is defined,

$j = 0, 1, \ldots, l$. From $\tilde{s}_0 \in C_0$ and $\tilde{\delta}(C_j, y_j) \subseteq C_{j+1}$, $j = 0, 1, \ldots, l-1$, we have $\tilde{s}_j \in C_j$, $j = 1, \ldots, l$. It follows that $\tilde{s}_j \prec \tilde{t}_j$, $j = 1, \ldots, l$. Thus $z_j = \tilde{\lambda}(\tilde{t}_j, y_j) = \tilde{\lambda}(\tilde{s}_j, y_j)$, $j = 0, 1, \ldots, l$. From the construction of $\tilde{M}_{\max}$, $\tilde{\lambda}(\tilde{s}_j, y_j) = \lambda'(\langle y_{j-1}, \ldots, y_{j-\tau}\rangle, y_j)$, $j = \tau, \ldots, l$. Since M' is an inverse of M and $y_0 y_1 \ldots y_l = \lambda(s, x_0 x_1 \ldots x_l)$, we have $\lambda'(\langle y_{j-1}, \ldots, y_{j-\tau}\rangle, y_j) = x_{j-\tau}$, $j = \tau, \ldots, l$. It follows that $z_j = \tilde{\lambda}(\tilde{s}_j, y_j) = \lambda'(\langle y_{j-1}, \ldots, y_{j-\tau}\rangle, y_j) = x_{j-\tau}$, $j = \tau, \ldots, l$. That is, $z_\tau z_{\tau+1} \ldots z_l = x_0 x_1 \ldots x_{l-\tau}$. We conclude that C_0 τ-matches s. Therefore, M''' is an inverse with delay τ of M. □

Theorem 6.2.4. *If $M = \langle X, Y, S, \delta, \lambda\rangle$ is an invertible finite automaton with delay τ and $\lambda(S, X) = Y$, then a finite automaton $M'' = \langle Y, X, S'', \delta'', \lambda''\rangle$ is an inverse with delay τ of M if and only if $M'' \prec M'''$.*

Proof. only if : Suppose that M'' is an inverse finite automaton with delay τ of M. From Corollary 6.2.1, there exist a closed compatible family $C_1, \ldots, C_h$ of $\tilde{M}_{\max}$ and a finite automaton $\bar{M}''' = \langle Y, X, \{c_1, \ldots, c_h\}, \bar{\delta}''', \bar{\lambda}'''\rangle$ in $\mathcal{M}(C_1, \ldots, C_h)$ such that $M'' \prec \bar{M}'''$ and $\{C_1, \ldots, C_h\} = C(\tilde{M}_{\max})$, where

$$\bar{\delta}'''(c_i, y) = \begin{cases} c_j, & \text{if } \tilde{\delta}(C_i, x) \neq \varnothing, \\ \text{undefined}, & \text{if } \tilde{\delta}(C_i, x) = \varnothing, \end{cases}$$

$$\bar{\lambda}'''(c_i, y) = \begin{cases} \tilde{\lambda}(s, y), & \text{if } \exists s_1 (s_1 \in C_i \ \&\ \tilde{\lambda}(s_1, y) \text{ is defined}), \\ \text{undefined}, & \text{otherwise}, \end{cases}$$

$$i = 1, \ldots, h, \ y \in Y,$$

j is an arbitrary integer satisfying $\tilde{\delta}(C_i, y) \subseteq C_j$, and s is an arbitrary state in C_i such that $\tilde{\lambda}(s, y)$ is defined.

We prove that $\bar{M}''' \prec M'''$. For any i, $1 \leqslant i \leqslant h$, c_i and C_i are states of $\bar{M}'''$ and M''', respectively. We prove $c_i \prec C_i$. For any $y_0, y_1, \ldots, y_l \in Y$, let $\bar{\lambda}'''(c_i, y_0 y_1 \ldots y_l) = x_0 x_1 \ldots x_l$, where $x_0, x_1, \ldots, x_l \in X$. Then there exist states $c_{i_j}, j = 0, 1, \ldots, l$ of $\bar{M}'''$ such that $c_{i_0} = c_i$, and

$$c_{i_{j+1}} = \bar{\delta}'''(c_{i_j}, y_j), \ j = 0, 1, \ldots l-1,$$
$$x_j = \bar{\lambda}'''(c_{i_j}, y_j), \ j = 0, 1, \ldots l.$$

Thus we have $\tilde{\delta}(C_{i_j}, y_j) \subseteq C_{i_{j+1}}$, $j = 0, 1, \ldots, l-1$. It follows that $C_{i_{j+1}} \in \delta'''(C_{i_j}, y_j)$, $j = 0, 1, \ldots, l-1$. From the definitions of $\bar{\lambda}'''$ and λ''', we have $\{\bar{\lambda}'''(c_{i_j}, y_j)\} = \{\tilde{\lambda}(t_j, y_j)\} = \lambda'''(C_{i_j}, x_j)$, therefore, $\lambda'''(C_{i_j}, y_j) = \{x_j\}$, $j = 0, 1, \ldots, l$, where t_j is the root state of $\tilde{M}$ in C_j. This yields $x_0\ x_1\ \ldots\ x_l \in \lambda'''(C_i,\ y_0 y_1 \ldots y_l)$. We conclude that $c_i \prec C_i$.

For any state s'' of M'', from $M'' \prec \bar{M}'''$, there exists a state c_i of $\bar{M}'''$ such that $s'' \prec c_i$. Using $c_i \prec C_i$, it is easy to verify that $s'' \prec C_i$. We conclude $M'' \prec M'''$.

if : Suppose that $M'' \prec M'''$. For any state s'' of M'', we can choose a state s''' of M''' such that $s'' \prec s'''$. For any state s of M, from Theorem 6.2.3, s''' τ-matches s. Using $s'' \prec s'''$, it is easy to verify that s'' τ-matches s. Thus M'' is an inverse with delay τ of M. □

6.3 Original Inverses of a Finite Automaton

Given an inverse finite automaton $M' = \langle Y, X, S', \delta', \lambda' \rangle$ with delay τ, let

$$f(y_\tau, \ldots, y_0) = \begin{cases} \lambda'(\delta'(s', y_0 \ldots y_{\tau-1}), y_\tau), & \text{if } |\lambda'(\delta'(S', y_0 \ldots y_{\tau-1}), y_\tau)| = 1, \\ \text{undefined}, & \text{otherwise}, \end{cases}$$
$$y_0, \ldots, y_\tau \in Y,$$

where s' is an arbitrarily fixed element in S'. We construct a labelled tree with level τ. Each vertex with level i, $0 \leqslant i \leqslant \tau$, emits $|X|$ arcs. Each arc has a label of the form (x, y), where $x \in X$ and $y \in Y$. x and y are called the *input label* and the *output label*, respectively. The input labels of $|X|$ arcs emitted from the same vertex are different letters in X. If the labels of arcs in a path from the root of length $\tau+1$ are (x_0, y_0), (x_1, y_1), …, (x_τ, y_τ), then $f(y_\tau, \ldots, y_0) = x_0$ holds. We use $\bar{\mathcal{T}}$ to denote the set of all such trees.

We use $\mathcal{M}(M')$ to denote the set of all $M(\mathcal{T}, \nu, \delta)$, $\mathcal{T}$ ranging over all nonempty closed subset of $\bar{\mathcal{T}}$. (For the construction of the finite automaton $M(\mathcal{T}, \nu, \delta)$, see Sect. 1.6 of Chap. 1.)

Lemma 6.3.1. *If $M' = \langle Y, X, S', \delta', \lambda' \rangle$ is an inverse finite automaton with delay τ, then M' is an inverse of any finite automaton in $\mathcal{M}(M')$.*

Proof. Let $M(\mathcal{T}, \nu, \delta)$ be a finite automaton in $\mathcal{M}(M')$. Denote $M(\mathcal{T}, \nu, \delta) = \langle X, Y, S, \delta, \lambda \rangle$. For any s in S and any x_i in X, $i = 0, 1, \ldots, \tau$, let $y_0 y_1 \ldots y_\tau = \lambda(s, x_0 x_1 \ldots x_\tau)$, where $y_i \in Y$, $i = 0, 1, \ldots, \tau$. From the construction of $M(\mathcal{T}, \nu, \delta)$, it is easy to show that if $s = \langle T, j \rangle$, where $T \in \mathcal{T}$, then $y_0 y_1 \ldots y_\tau$ is the output label of a path from the root of T of length $\tau + 1$ with input label $x_0 x_1 \ldots x_\tau$. Therefore, $f(y_\tau, \ldots, y_0) = x_0$ holds. From the definition of f, for any s' in S', we have $\lambda'(\delta'(s', y_0 y_1 \ldots y_{\tau-1}), y_\tau) = f(y_\tau, \ldots, y_0)$. It follows that

$$\begin{aligned} \lambda'(s', \lambda(s, x_0 x_1 \ldots x_\tau)) &= \lambda'(s', y_0 y_1 \ldots y_\tau) \\ &= \lambda'(s', y_0 y_1 \ldots y_{\tau-1}) \lambda'(\delta'(s', y_0 y_1 \ldots y_{\tau-1}), y_\tau) \\ &= \lambda'(s', y_0 y_1 \ldots y_{\tau-1}) f(y_\tau, \ldots, y_0) \\ &= \lambda'(s', y_0 y_1 \ldots y_{\tau-1}) x_0. \end{aligned}$$

We conclude that M' is an inverse of $M(\mathcal{T}, \nu, \delta)$ with delay τ. □

We use $\mathcal{T}_m$ to denote the maximum closed subset of $\bar{\mathcal{T}}$. Notice that $\mathcal{T}_m$ is unique. We use $\bar{\mathcal{M}}(M')$ to denote the set of all $M(\mathcal{T}_m, \nu, \delta)$.

Lemma 6.3.2. *If $M' = \langle Y, X, S', \delta', \lambda' \rangle$ is an inverse finite automaton with delay τ of $M = \langle X, Y, S, \delta, \lambda \rangle$, then there exists M_2 in $\bar{\mathcal{M}}(M')$ such that M and some finite subautomaton of M_2 are isomorphic.*

Proof. We construct $M(\mathcal{T}_1, \nu_1, \delta_1)$ as follows. Partition S so that s_1 and s_2 belong to the same block if and only if $T_\tau^M(s_1) = T_\tau^M(s_2)$. (For the definition of $T_\tau^M(s)$, see Sect. 1.6 of Chap. 1.) We use $C_1, \ldots, C_r$ to denote such blocks. In the case of $s \in C_j$, we take $\nu_1(T_\tau^M(s)) = |C_j|$. Take $\mathcal{T}_1 = \{T_\tau^M(s) \mid s \in S\}$ and $S_1 = \{\langle T, j \rangle \mid T \in \mathcal{T}_1, j = 1, \ldots, \nu_1(T)\}$.

Fix a one-to-one mapping φ from S onto S_1 such that $\varphi(s) = \langle T_\tau^M(s), j \rangle$ for some j. From the definition of ν_1, such a φ is existent. Define $\delta_1(\varphi(s), x) = \varphi(\delta(s, x))$. It is easy to verify that $M(\mathcal{T}_1, \nu_1, \delta_1)$ is a finite automaton and φ is an isomorphism from M to $M(\mathcal{T}_1, \nu_1, \delta_1)$. Therefore, M and $M(\mathcal{T}_1, \nu_1, \delta_1)$ are isomorphic.

For any s in S and any $x_0, \ldots, x_\tau$ in X, let $y_0 \ldots y_\tau = \lambda(s, x_0 \ldots x_\tau)$, where $y_0, \ldots, y_\tau \in Y$. Since M' is an inverse finite automaton with delay τ of M, for any s' in S', we have $\lambda'(\delta'(s', y_0 \ldots y_{\tau-1}), y_\tau) = x_0$. It follows that $f(y_\tau, \ldots, y_0) = x_0$. Thus $T_\tau^M(s)$ is in $\bar{\mathcal{T}}$. This yields $\mathcal{T}_1 \subseteq \bar{\mathcal{T}}$. Since $\mathcal{T}_1$ is closed, we have $\mathcal{T}_1 \subseteq \mathcal{T}_m$. Let

$$\nu_2(T) = \begin{cases} \nu_1(T), & \text{if } T \in \mathcal{T}_1, \\ 1, & \text{if } T \in \mathcal{T}_m \setminus \mathcal{T}_1, \end{cases}$$

$$\delta_2(\langle T, i \rangle, x) = \begin{cases} \delta_1(\langle T, i \rangle, x), & \text{if } T \in \mathcal{T}_1, \\ \langle \bar{T}, 1 \rangle, & \text{if } T \in \mathcal{T}_m \setminus \mathcal{T}_1, \end{cases}$$

where $\bar{T}$ is an arbitrarily fixed x-successor of T in $\mathcal{T}_m$. Then $M(\mathcal{T}_m, \nu_2, \delta_2) \in \bar{\mathcal{M}}(M')$. Choose $M(\mathcal{T}_m, \nu_2, \delta_2)$ as M_2. Clearly, $M(\mathcal{T}_1, \nu_1, \delta_1)$ is a finite subautomaton of M_2. □

From the proof of the above lemma, we have the following corollary.

Corollary 6.3.1. *If $M' = \langle Y, X, S', \delta', \lambda' \rangle$ is an inverse finite automaton with delay τ of $M = \langle X, Y, S, \delta, \lambda \rangle$, then there exists M_1 in $\mathcal{M}(M')$ such that M and M_1 are isomorphic.*

From Lemmas 6.3.1 and 6.3.2, we obtain the following theorem.

Theorem 6.3.1. *M' is an inverse finite automaton with delay τ of a finite automaton M if and only if there exist M_2 in $\bar{\mathcal{M}}(M')$ and a finite subautomaton M_1 of M_2 such that M and M_1 are isomorphic.*

Similarly, from Lemma 6.3.1 and Corollary 6.3.1, we obtain the following theorem.

Theorem 6.3.2. *M' is an inverse finite automaton with delay τ of a finite automaton M if and only if there exists M_1 in $\mathcal{M}(M')$ such that M and M_1 are isomorphic.*

Let $M(\mathcal{T}_m) = \langle X, Y, \mathcal{T}_m, \delta, \lambda\rangle$ be a nondeterministic finite automaton, where

$$\begin{aligned}&\delta(T,x) = \{\bar{T} \mid \bar{T} \in \mathcal{T}_m, \bar{T} \text{ is an } x\text{-successor of } T\},\\ &\lambda(T,x) = \{y\},\\ &\quad T \in \mathcal{T}_m, x \in X,\end{aligned}$$

and (x, y) is the label of an arc emitted from the root of T. Notice that $\delta(T, x)$ and $\lambda(T, x)$ are nonempty.

Theorem 6.3.3. (a) *M' is an inverse finite automaton with delay τ of the nondeterministic finite automaton $M(\mathcal{T}_m)$.*

(b) *If M' is an inverse finite automaton with delay τ of a finite automaton M, then $M \prec M(\mathcal{T}_m)$.*

Proof. (a) For any state T_0 of $M(\mathcal{T}_m)$ and any $x_0, x_1, \ldots, x_l$ in X, let $y_0y_1 \ldots y_l \in \lambda(T_0, x_0x_1 \ldots x_l)$, where $l \geqslant \tau$, and $y_0, y_1, \ldots \in Y$. Then there exist states $T_1, T_2, \ldots, T_{l+1}$ of $M(\mathcal{T}_m)$ such that $T_{i+1} \in \delta(T_i, x_i)$ and $y_i \in \lambda(T_i, x_i)$ hold for $i = 0, 1, \ldots, l$. From the definition of δ, it is easy to see that for any i, $0 \leqslant i \leqslant l - \tau$, $(x_i, y_i), (x_{i+1}, y_{i+1}), \ldots, (x_{i+\tau}, y_{i+\tau})$ are labels of arcs in a path from the root to a leaf of T_i. This yields $f(y_{i+\tau}, \ldots, y_i) = x_i$, $i = 0, 1, \ldots, l - \tau$. It follows that $\lambda'(\delta'(s', y_i \ldots y_{i+\tau-1}), y_{i+\tau}) = x_i$ holds for any state s' of M'. Thus for any state s'_0 of M', $\lambda'(s'_0, y_0y_1 \ldots y_l) = x'_0 \ldots x'_{\tau-1}x_0x_1 \ldots x_{l-\tau}$ holds for some $x'_0, \ldots, x'_{\tau-1}$ in X. Therefore, M' is an inverse finite automaton with delay τ of $M(\mathcal{T}_m)$.

(b) Suppose that M' is an inverse finite automaton with delay τ of a finite automaton M. From Lemma 6.3.2, there exist $M_2 = M(\mathcal{T}_m, \nu, \delta_2)$ and a finite subautomaton M_1 of M_2 such that M and M_1 are isomorphic. We prove $M_2 \prec M(\mathcal{T}_m)$. For any state $\langle T_0, j_0\rangle$ of M_2, T_0 is a state of $M(\mathcal{T}_m)$. To prove $\langle T_0, j_0\rangle \prec T_0$, let $x_0, x_1, \ldots, x_l$ be in X, and $y_0y_1 \ldots y_l = \lambda_2(\langle T_0, j_0\rangle, x_0x_1 \ldots x_l)$, where $y_0, y_1, \ldots, y_l$ are in Y, and λ_2 is the output function of M_2. Thus there exist states $\langle T_i, j_i\rangle$, $i = 1, \ldots, l+1$ of M_2 such that $\delta_2(\langle T_i, j_i\rangle, x_i) = \langle T_{i+1}, j_{i+1}\rangle$ and $\lambda_2(\langle T_i, j_i\rangle, x_i) = y_i$, $i = 0, 1, \ldots, l$, where δ_2 is the next state function of M_2. From the definition of $M(\mathcal{T}_m, \nu, \delta_2)$, T_{i+1} is an x_i-successor of T_i and (x_i, y_i) is the label of an arc emitted from the root of T_i. From the definition of $M(\mathcal{T}_m)$, we have $y_0y_1 \ldots y_l \in \lambda(T_0, x_0x_1 \ldots x_l)$.

Thus $\langle T_0, j_0\rangle \prec T_0$. We conclude that $M_2 \prec M(\mathcal{T}_m)$. Since M_1 is a finite subautomaton of M_2, this yields $M_1 \prec M(\mathcal{T}_m)$. Since M is isomorphic to M_1, we have $M \prec M(\mathcal{T}_m)$. □

Theorem 6.3.3 means that $M(\mathcal{T}_m)$ is a "universal" nondeterministic finite automaton for finite automata of which M' is an inverse with delay τ.

6.4 Weak Inverses of a Finite Automaton

Given a weakly invertible finite automaton $M = \langle X, Y, S, \delta, \lambda\rangle$ with delay τ, there exists a weak inverse finite automaton with delay τ of M. Let $M' = \langle Y, X, S', \delta', \lambda'\rangle$ be a weak inverse finite automaton with delay τ of M. For each s in S, we choose a state of M', say $\varphi(s)$, such that $\varphi(s)$ τ-matches s.

For any s in S, let $M_s = \langle Y, 2^S, \delta_M, \{s\}, S_s \setminus \{\varnothing\}\rangle$ be a finite automaton recognizer, where $S_s = \{\delta_M(\{s\}, \beta) \mid \beta \in Y^*\}$, δ_M is defined in the beginning of Sect. 6.2. Let $R_s = \{\lambda(s, \alpha) \mid \alpha \in X^*\}$. It is easy to verify that M_s recognizes R_s.

We construct a partial finite automaton $M'_0 = \langle Y, X, S'_0, \delta'_0, \lambda'_0\rangle$ from M' and M as follows. Let

$$S'_0 = T^{\geqslant\tau} \cup T^0,$$

where

$$\begin{aligned}
T_0 &= \bigcup_{s\in S} T_s^0,\\
T^{\geqslant\tau} &= \bigcup_{s\in S} T_s^{\geqslant\tau},\\
T_s^0 &= \{\langle s, \beta\rangle \mid \beta \in R_s, |\beta| < \tau\},\\
T_s^{\geqslant\tau} &= \{\langle \delta_M(\{s\}, \beta), \delta'(\varphi(s), \beta)\rangle \mid \beta \in R_s, |\beta| \geqslant \tau\}.
\end{aligned}$$

For any $\langle t, s'\rangle$ in $T^{\geqslant\tau}$ and any y in Y, let

$$\delta'_0(\langle t, s'\rangle, y) = \begin{cases} \langle \delta_M(t, y), \delta'(s', y)\rangle, & \text{if } \langle \delta_M(t, y), \delta'(s', y)\rangle \in T^{\geqslant\tau},\\ \text{undefined}, & \text{otherwise,}\end{cases}$$

$$\lambda'_0(\langle t, s'\rangle, y) = \begin{cases} \lambda'(s', y), & \text{if } \langle \delta_M(t, y), \delta'(s', y)\rangle \in T^{\geqslant\tau},\\ \text{undefined}, & \text{otherwise.}\end{cases}$$

For any $\langle s, \beta\rangle$ in T_s^0, $|\beta| < \tau - 1$ and any y in Y, let

$$\delta'_0(\langle s, \beta\rangle, y) = \begin{cases} \langle s, \beta y\rangle, & \text{if } \langle s, \beta y\rangle \in T_s^0,\\ \text{undefined}, & \text{otherwise,}\end{cases}$$

$$\lambda'_0(\langle s, \beta\rangle, y) = \text{undefined}.$$

For any $\langle s, \beta\rangle$ in T_s^0, $|\beta| = \tau - 1$ and any y in Y, let

$$\delta_0'(\langle s,\beta\rangle, y) = \begin{cases} \langle \delta_M(\{s\}, \beta y), \delta'(\varphi(s), \beta y)\rangle, & \text{if } \langle \delta_M(\{s\}, \beta y), \delta'(\varphi(s), \beta y)\rangle \in T_s^{\geqslant \tau}, \\ \text{undefined}, & \text{otherwise}, \end{cases}$$

$$\lambda_0'(\langle s, \beta\rangle, y) = \text{undefined}.$$

From the construction of M_0', for any $\beta \in Y^*$ with $|\beta| < \tau$, $\delta_0'(\langle s, \varepsilon\rangle, \beta) = \langle s, \beta\rangle$ if $\beta \in R_s$, $\delta_0'(\langle s, \varepsilon\rangle, \beta)$ is undefined otherwise. For any $\beta \in Y^*$ with $|\beta| = \tau$, $\delta_0'(\langle s, \varepsilon\rangle, \beta) = \langle \delta_M(\{s\}, \beta), \delta'(\varphi(s), \beta)\rangle$ if $\beta \in R_s$, $\delta_0'(\langle s, \varepsilon\rangle, \beta)$ is undefined otherwise. For any $\beta \in Y^*$ with $|\beta| > \tau$, $\delta_0'(\langle s, \varepsilon\rangle, \beta) = \delta_0'(\langle \delta_M(\{s\}, \beta'), \delta'(\varphi(s), \beta')\rangle, \beta'') = \langle \delta_M(\{s\}, \beta), \delta'(\varphi(s), \beta)\rangle$ if $\beta \in R_s$, $\delta_0'(\langle s, \varepsilon\rangle, \beta)$ is undefined otherwise, where $\beta = \beta'\beta''$ with $|\beta'| = \tau$. Therefore, $\delta_0'(\langle s, \varepsilon\rangle, \beta)$ is defined if and only if $\beta \in R_s$.

Lemma 6.4.1. *If M' is a weak inverse finite automaton with delay τ of M, then the partial finite automaton M_0' is a weak inverse with delay τ of M.*

Proof. Suppose that M' is a weak inverse finite automaton with delay τ of M. In the construction of M_0', for any s in S, we choose a state $\varphi(s)$ of M' such that $\varphi(s)$ τ-matches s. Let $\alpha = x_0 x_1 \ldots$, where $x_i \in X$, $i = 0, 1, \ldots$ Then we have

$$\lambda'(\varphi(s), \lambda(s, \alpha)) = x_{-\tau} \ldots x_{-1} x_0 x_1 \ldots,$$

for some $x_{-\tau}, \ldots, x_{-1}$ in X. Denote $\lambda(s, \alpha) = \beta_1\beta$, $|\beta_1| = \tau$. Then β_1 and any prefix of $\beta_1\beta$ are in R_s. It follows that for any prefix β' of $\beta_1\beta$, $\delta_0'(\langle s, \varepsilon\rangle, \beta')$ is defined. From the construction of M_0', this yields that for any prefix β'' of β, $\lambda_0'(\delta_0'(\langle s, \varepsilon\rangle, \beta_1), \beta'')$, i.e., $\lambda_0'(\langle \delta_M(\{s\}, \beta_1), \delta'(\varphi(s), \beta_1)\rangle, \beta'')$, is defined and it equals $\lambda'(\delta'(\varphi(s), \beta_1), \beta'')$. Thus $\lambda_0'(\langle \delta_M(\{s\}, \beta_1), \delta'(\varphi(s), \beta_1)\rangle, \beta)$ is defined and it equals $\lambda'(\delta'(\varphi(s), \beta_1), \beta)$. It follows that

$$\begin{aligned} \lambda_0'(\langle s, \varepsilon\rangle, \lambda(s, \alpha)) &= \lambda_0'(\langle s, \varepsilon\rangle, \beta_1\beta) \\ &= \lambda_0'(\langle s, \varepsilon\rangle, \beta_1)\lambda_0'(\delta_0'(\langle s, \varepsilon\rangle, \beta_1), \beta) \\ &= \lambda_0'(\langle s, \varepsilon\rangle, \beta_1)\lambda_0'(\langle \delta_M(\{s\}, \beta_1), \delta'(\varphi(s), \beta_1)\rangle, \beta) \\ &= x'_{-\tau} \ldots x'_{-1} \lambda'(\delta'(\varphi(s), \beta_1), \beta) \\ &= x'_{-\tau} \ldots x'_{-1} x_0 x_1 \ldots, \end{aligned}$$

where $x'_{-\tau} = \cdots = x'_{-1} = \_$. Therefore, $\langle s, \varepsilon\rangle$ τ-matches s. It follows that M_0' is a weak inverse with delay τ of M. □

Lemma 6.4.2. *Let M' be a weak inverse finite automaton with delay τ of M. For any partial finite automaton $M'' = \langle Y, X, S'', \delta'', \lambda''\rangle$, M'' is a weak inverse with delay τ of M if and only if $M_0' \prec M''$.*

Proof. only if : Suppose that M'' is a weak inverse with delay τ of M. Given any s in S and any s'_0 in $T_s^{\geqslant\tau} \cup T_s^0$, from the definitions of δ'_0 and S'_0, there exists $\bar{\beta}$ in R_s such that $s'_0 = \delta'_0(\langle s, \varepsilon\rangle, \bar{\beta})$. Since M'' is a weak inverse with delay τ of M, there exists s'' in S'' such that s'' τ-matches s. We prove that $\delta''(s'', \bar{\beta})$ is defined. From the definition of R_s, there exists β in Y^* such that $\beta \neq \varepsilon$, $\bar{\beta}\beta \in R_s$ and $|\bar{\beta}\beta| > \tau$. It follows that $\bar{\beta}\beta = \lambda(s, \alpha)$ for some α of length $|\bar{\beta}\beta|$ in X^*. Denote $\alpha = x_0\ x_1\ \ldots\ x_n$, where $x_0, x_1, \ldots, x_n \in X$, and $n = |\bar{\beta}\beta| - 1$. Since s'' τ-matches s, we have

$$\lambda''(s'', \bar{\beta}\beta) = \lambda''(s'', \lambda(s, \alpha)) = x_{-\tau} \ldots x_{-1} x_0 \ldots x_{n-\tau},$$

for some $x_{-\tau}, \ldots, x_{-1}$ in $X \cup \{\_\}$. From $\beta \neq \varepsilon$ and $x_{n-\tau} \in X$, it follows that $\delta''(s'', \bar{\beta})$ is defined.

Let $s''_0 = \delta''(s'', \bar{\beta})$. We prove $s'_0 \prec s''_0$. Assume that β in Y^* is applicable to s'_0. In the case of $\beta = \varepsilon$, it is obvious that β is applicable to s''_0 and $\lambda'_0(s'_0, \beta) \prec \lambda''(s''_0, \beta)$. In the case of $\beta \neq \varepsilon$, let $\beta = \beta_1 y$, where $y \in Y$. Since β is applicable to s'_0, $\delta'_0(s'_0, \beta_1)$, i.e., $\delta'_0(\langle s, \varepsilon\rangle, \bar{\beta}\beta_1)$ is defined. Thus $\bar{\beta}\beta_1 \in R_s$. It immediately follows that $\lambda(s, x_0 x_1 \ldots x_{n-1}) = \bar{\beta}\beta_1$ for some $x_0, x_1, \ldots, x_{n-1}$ in X, where $n = |\bar{\beta}\beta_1|$. Let $r = \max(\tau, n)$. Take arbitrarily $x_n, x_{n+1}, \ldots, x_r$ in X. Let $\lambda(s, x_0 x_1 \ldots x_r) = \bar{\beta}\beta_1\beta_2 y_r$, for some β_2 in Y^* and y_r in Y. Since s'' τ-matches s, we have

$$\lambda''(s'', \bar{\beta}\beta_1\beta_2 y_r) = \lambda''(s'', \lambda(s, x_0 x_1 \ldots x_r)) = x_{-\tau} \ldots x_{-1} x_0 \ldots x_{r-\tau},$$

for some $x_{-\tau}, \ldots, x_{-1}$ in $X \cup \{\_\}$. Since $r \geqslant \tau$, we have $x_{r-\tau} \in X$. It follows that $\delta''(s'', \bar{\beta}\beta_1\beta_2)$ is defined. From $\delta''(s'', \bar{\beta}\beta_1\beta_2) = \delta''(s''_0, \beta_1\beta_2) = \delta''(\delta''(s''_0, \beta_1), \beta_2)$, $\delta''(s''_0, \beta_1)$ is defined. Therefore, β is applicable to s''_0. We have proven that $\lambda(s,\ x_0 x_1 \ldots x_{n-1}) = \bar{\beta}\beta_1$ and $\lambda''(s'', \bar{\beta}\beta_1\beta_2 y_r) = x_{-\tau} \ldots x_{-1} x_0 \ldots x_{r-\tau}$. From $r \geqslant n$, we have $\lambda''(s'',\ \bar{\beta}\beta_1) = x_{-\tau}\ \ldots\ x_{-1}\ x_0\ \ldots\ x_{n-\tau-1}$. It follows that $\lambda''(s'', \bar{\beta}\beta) = x_{-\tau}\ \ldots\ x_{-1}\ x_0\ \ldots\ x_{n-\tau-1}\ \bar{x}_{n-\tau}$, for some $\bar{x}_{n-\tau}$ in $X \cup \{\_\}$. From the proof of Lemma 6.4.1, $\langle s, \varepsilon\rangle$ τ-matches s. When $\lambda'_0(\delta'_0(s'_0, \beta_1), y)$ is undefined, we have

$$\begin{aligned}\lambda'_0(\langle s, \varepsilon\rangle, \bar{\beta}\beta) &= \lambda'_0(\langle s, \varepsilon\rangle, \bar{\beta}\beta_1)\lambda'_0(\delta'_0(\langle s, \varepsilon\rangle, \bar{\beta}\beta_1), y) \\ &= x'_{-\tau} \ldots x'_{-1} x_0 \ldots x_{n-\tau-1} x'_{n-\tau},\end{aligned}$$

where $x'_{-\tau} = \cdots = x'_{-1} = x'_{n-\tau} = \_$. Since $x'_i \prec x_i$, $i = -\tau, \ldots, -1$, and $x'_{n-\tau} \prec \bar{x}_{n-\tau}$, we have $\lambda'_0(\langle s, \varepsilon\rangle, \bar{\beta}\beta) \prec \lambda''(s'', \bar{\beta}\beta)$. This yields that $\lambda'_0(s'_0, \beta) = \lambda'_0(\delta'_0(\langle s, \varepsilon\rangle, \bar{\beta}), \beta) \prec \lambda''(\delta''(s'', \bar{\beta}), \beta) = \lambda''(s''_0, \beta)$. When $\lambda'_0(\delta'_0(s'_0, \beta_1), y)$ is defined, from the construction of M'_0, $\delta'_0(\delta'_0(s'_0, \beta_1), y)$, i.e., $\delta'_0(\langle s, \varepsilon\rangle, \bar{\beta}\beta)$, is defined. It follows that $\bar{\beta}\beta \in R_s$. Thus $\lambda(s, x_0 x_1 \ldots x_n) = \bar{\beta}\beta$ for some x_0, $x_1, \ldots, x_n$ in X. Since $\langle s, \varepsilon\rangle$ τ-matches s, we have

$$\lambda'_0(\langle s, \varepsilon\rangle, \bar{\beta}\beta) = \lambda'_0(\langle s, \varepsilon\rangle, \lambda(s, x_0 x_1 \ldots x_n)) = x'_{-\tau} \ldots x'_{-1} x_0 \ldots x_{n-\tau},$$

where $x'_{-\tau} = \cdots = x'_{-1} = \_$. Since s'' τ-matches s, we have

$$\lambda''(s'', \bar{\beta}\beta) = \lambda''(s'', \lambda(s, x_0x_1 \ldots x_n)) = x_{-\tau} \ldots x_{-1}x_0 \ldots x_{n-\tau},$$

for some $x_{-\tau}, \ldots, x_{-1}$ in $X \cup \{\_\}$. Therefore, $\lambda'_0(\langle s, \varepsilon\rangle, \bar{\beta}\beta) \prec \lambda''(s'', \bar{\beta}\beta)$. This yields that $\lambda'_0(s'_0, \beta) \prec \lambda''(s''_0, \beta)$. We conclude $s'_0 \prec s''_0$.

Since for any s'_0 in S'_0, we can find s''_0 in S'' such that $s'_0 \prec s''_0$, we have $M'_0 \prec M''$.

if : Suppose that $M'_0 \prec M''$. From Lemma 6.4.1, M'_0 is a weak inverse with delay τ of M. Thus for each s in S, there exists s'_0 in S'_0 such that s'_0 τ-matches s. From $M'_0 \prec M''$, there exists s'' in S'' such that $s'_0 \prec s''$. We prove that s'' τ-matches s. Let $\alpha = x_0x_1 \ldots$, where $x_i \in X$, $i = 0, 1, \ldots$ Denote $y_0y_1 \ldots = \lambda(s, x_0x_1 \ldots)$, where $y_i \in Y$, $i = 0, 1, \ldots$ Then we have $\lambda'_0(s'_0, y_0y_1 \ldots) = x_{-\tau} \ldots x_{-1}\, x_0x_1 \ldots$, for some $x_{-\tau}, \ldots, x_{-1} \in X \cup \{\_\}$. Since $x_j \in X$ for any $j \geqslant 0$, $y_0y_1 \ldots y_i$ is applicable to s'_0 for any $i \geqslant 0$. From $s'_0 \prec s''$, $y_0y_1 \ldots y_i$ is applicable to s'', and $\lambda'_0(s'_0, y_0y_1 \ldots y_i) \prec \lambda''(s'', y_0y_1 \ldots y_i)$, i.e., $x_{-\tau}\, x_{-\tau+1} \ldots x_{i-\tau} \prec \lambda''(s'', y_0y_1 \ldots y_i)$, $i = 0, 1, \ldots$ Noticing that $x_i \in X$ for any $i \geqslant 0$, it follows that $\lambda''(s'', \lambda(s, \alpha)) = \lambda''(s'', y_0y_1 \ldots) = x''_{-\tau} \ldots x''_{-1}$ $x_0x_1 \ldots$, for some $x''_{-\tau}, \ldots, x''_{-1} \in X \cup \{\_\}$. Therefore, s'' τ-matches s. It follows that M'' is a weak inverse with delay τ of M. □

Since M is weakly invertible with delay τ, there is a finite automaton M' such that M' is a weak inverse with delay τ of M. Given a weak inverse finite automaton M' with delay τ of M, from M and M', we construct a partial finite automaton M'_0 as mentioned above.

For any closed compatible family $C_1, \ldots, C_k$ of M'_0, according to the discussion in Subsect. 6.1.1, we can construct a set $\mathcal{M}(C_1, \ldots, C_k)$ of partial finite automata. We use $\mathcal{M}_1(M'_0)$ to denote the union set of all $\mathcal{M}(C_1, \ldots, C_k)$, $C_1, \ldots, C_k$ ranging over all closed compatible family of M'_0.

Theorem 6.4.1. *If $M = \langle X, Y, S, \delta, \lambda\rangle$ is a weakly invertible finite automaton with delay τ, then a finite automaton $M'' = \langle Y, X, S'', \delta'', \lambda''\rangle$ is a weak inverse with delay τ of M if and only if there exist a partial finite automaton $\tilde{M}$ in $\mathcal{M}_1(M'_0)$ and a partial finite subautomaton M''' of M'' such that $\tilde{M}$ and M''' are isomorphic.*

Proof. From Lemma 6.4.2, for any (partial) finite automaton M'', M'' is a weak inverse with delay τ of M if and only if $M'_0 \prec M''$. From Theorems 6.1.1 and 6.1.2, $M'_0 \prec M''$ if and only if there exist a partial finite automaton $\tilde{M}$ in $\mathcal{M}_1(M'_0)$ and a partial finite subautomaton M''' of M'' such that $\tilde{M}$ and M''' are isomorphic. We then obtain the result of the theorem. □

6.5 Original Weak Inverses of a Finite Automaton

Given a weak inverse finite automaton $M' = \langle Y, X, S', \delta', \lambda' \rangle$ with delay τ, let

$$f(y_\tau, \ldots, y_0, s') = \lambda'(\delta'(s', y_0 \ldots y_{\tau-1}), y_\tau),$$
$$s' \in S',\ y_0, \ldots, y_\tau \in Y.$$

For any s' in S', construct a set $\mathcal{T}'_{s'}$ of labelled trees with level τ as follows. In any tree in $\mathcal{T}'_{s'}$, each vertex with level $\leqslant \tau$ emits $|X|$ arcs and each arc has a label of the form (x, y), where $x \in X$ and $y \in Y$. x and y are called the *input label* and the *output label* of the arc, respectively. Input labels of $|X|$ arcs emitted from the same vertex are different letters in X. Each vertex of the tree has a label also. The label of the root vertex is s'. For any vertex other than the root, if the labels of arcs in the path from the root to the vertex are (x_0, y_0), (x_1, y_1), ..., (x_i, y_i), then the label of the vertex is $\delta'(s', y_0 \ldots y_i)$. For any path from the root to a leaf, if the labels of arcs in the path are (x_0, y_0), (x_1, y_1), ..., (x_τ, y_τ), then $f(y_\tau, \ldots, y_0, s') = x_0$ holds.

Let $\bar{\mathcal{T}}' = \cup_{s' \in S'} \mathcal{T}'_{s'}$.

We use $\mathcal{M}'(M')$ to denote the set of all $M(\mathcal{T}, \nu, \delta)$, $\mathcal{T}$ ranging over all nonempty closed subset of $\bar{\mathcal{T}}'$. We use $\mathcal{T}'_m$ to denote the maximum closed subset of $\bar{\mathcal{T}}'$. Clearly, $\mathcal{T}'_m$ is unique. Let $\bar{\mathcal{M}}'(M') = \{M(\mathcal{T}'_m, \nu, \delta) \mid M(\mathcal{T}'_m, \nu, \delta) \in \mathcal{M}'(M')\}$. (For the construction of the finite automaton $M(\mathcal{T}, \nu, \delta)$, see the end of Sect. 1.6 of Chap. 1.)

Lemma 6.5.1. *If M' is a weak inverse finite automaton with delay τ, then M' is a weak inverse with delay τ of any finite automaton in $\mathcal{M}'(M')$.*

Proof. Let $M(\mathcal{T}, \nu, \delta)$ be a finite automaton in $\mathcal{M}'(M')$. Let $M' = \langle Y, X, S', \delta', \lambda' \rangle$ and $M(\mathcal{T}, \nu, \delta) = \langle X, Y, S, \delta, \lambda \rangle$. For any $s = \langle T, j \rangle$ in S and any x_i in X, $i = 0, 1, \ldots$, let $y_0 y_1 \ldots = \lambda(s, x_0 x_1 \ldots)$, where $y_i \in Y$, $i = 0, 1, \ldots$ Let s'_0 be the label of the root of the tree T. Given $i \geqslant 0$, we use $\langle T', j' \rangle$ to denote $\delta(\langle T, j \rangle, y_0 \ldots y_{i-1})$. From the construction of $M(\mathcal{T}, \nu, \delta)$, it is easy to show that the label of the root of T' is $\delta'(s'_0, y_0 \ldots y_{i-1})$, which is abbreviated to s'_i. From the construction of $M(\mathcal{T}, \nu, \delta)$, (x_i, y_i), ..., $(x_{i+\tau}, y_{i+\tau})$ are the labels of arcs in some path from the root of T'. Since $T' \in \mathcal{T}'_{s'_i}$, we have $f(y_{i+\tau}, \ldots, y_i, s'_i) = x_i$. Thus

$$\begin{aligned}\lambda'(s'_i, y_i \ldots y_{i+\tau}) &= \lambda'(s'_i, y_i \ldots y_{i+\tau-1})\lambda'(\delta'(s'_i, y_i \ldots y_{i+\tau-1}), y_{i+\tau}) \\ &= \lambda'(s'_i, y_i \ldots y_{i+\tau-1}) f(y_{i+\tau}, \ldots, y_i, s'_i) \\ &= \lambda'(s'_i, y_i \ldots y_{i+\tau-1}) x_i.\end{aligned}$$

It follows that $\lambda'(s'_0, y_0 y_1 \ldots) = x_{-\tau} \ldots x_{-1} x_0 x_1 \ldots$, for some $x_{-\tau}$, ..., x_{-1} in X. Therefore, s'_0 τ-matches s. We conclude that M' is a weak inverse of $M(\mathcal{T}, \nu, \delta)$ with delay τ. □

Lemma 6.5.2. *If M' is a weak inverse finite automaton with delay τ of a finite automaton M, then there exists $M(\mathcal{T}_1, \nu_1, \delta_1)$ in $\mathcal{M}'(M')$ such that M and $M(\mathcal{T}_1, \nu_1, \delta_1)$ are equivalent.*

Proof. Let $M' = \langle Y, X, S', \delta', \lambda' \rangle$ and $M = \langle X, Y, S, \delta, \lambda \rangle$. For any s in S and any s' in S', we assign labels to vertices of the tree $T_\tau^M(s)$ as follows. s' is assigned to the root. For any vertex other than the root, if the labels of arcs in the path from the root to the vertex are (x_0, y_0), (x_1, y_1), ..., (x_i, y_i), we assign $\delta'(s', y_0 \dots y_i)$ to the vertex. We use $T_\tau(s, s')$ to denote the tree, with arc label and vertex label, as mentioned above. To construct $M(\mathcal{T}_1, \nu_1, \delta_1)$, take $\mathcal{T}_1 = \{T_\tau(s, s') \mid s \in S, s' \in S', s' \text{ matches } s \text{ with delay } \tau\}$. Clearly, for any x in X, $T_\tau(\delta(s, x), \delta'(s', \lambda(s, x)))$ is an x-successor of $T_\tau(s, s')$. Thus $\mathcal{T}_1$ is closed. For any T in $\mathcal{T}_1$, we use s' to denote the label of the root of T and take $\nu_1(T)$ as the number of elements in the set $\{s \mid s \in S, T_\tau(s, s') = T,\ s' \text{ matches } s \text{ with delay } \tau\}$. For any s'' in S_0', fix a one-to-one mapping $\varphi_{s''}$ from the set $\{s \mid s \in S, s'' \text{ matches } s \text{ with delay } \tau\}$ onto the set $\{\langle T, j \rangle \mid T \in \mathcal{T}_1, 1 \leqslant j \leqslant \nu_1(T),\ \text{the label of the root of } T \text{ is } s''\}$, where $S_0' = \{s' \in S', \text{there exists } s \in S \text{ such that } s' \ \tau\text{-matches } s\}$. From the definition of ν_1, such a $\varphi_{s''}$ is existent. For any T in $\mathcal{T}_1$, any j, $1 \leqslant j \leqslant \nu_1(T)$, and any x in X, let $\delta_1(\langle T, j \rangle, x) = \varphi_{s''}(\delta(s, x))$, where $s = \varphi_{s'}^{-1}(\langle T, j \rangle)$, $s'' = \delta'(s', \lambda(s, x))$, and s' is the label of the root of T. It is easy to verify that if $\delta_1(\langle T, j \rangle, x) = \langle T', j' \rangle$, then T' is the x-successor of T.

Denote $M(\mathcal{T}_1, \nu_1, \delta_1) = \langle X, Y, S_1, \delta_1, \lambda_1 \rangle$. For any s in S, any s' in S' and any t in S_1, if $\varphi_{s'}(s) = t$, then there exists j such that $t = \langle T_\tau(s, s'), j \rangle$. From the definition of $T_\tau(s, s')$ and the construction of $M(\mathcal{T}_1, \nu_1, \delta_1)$, for any x in X, we have $\lambda(s, x) = \lambda_1(t, x)$. And from the definition of δ_1, we have $\delta_1(t, x) = \varphi_{s''}(\delta(s, x))$, where $s'' = \delta'(s', \lambda(s, x))$. To sum up, for any s in S, any s' in S' and any t in S_1, if $\varphi_{s'}(s) = t$, then for any x in X, we have $\lambda(s, x) = \lambda_1(t, x)$ and $\varphi_{s''}(\delta(s, x)) = \delta_1(t, x)$. Using this result repeatedly, it is easy to show that for any s in S, any s' in S' and any t in S_1, if $\varphi_{s'}(s) = t$, then $s \sim t$. Since M' is a weak inverse with delay τ of M, for any s in S, there exist s' in S' and t in S_1 such that $\varphi_{s'}(s) = t$. It follows that for any s in S, there exists t in S_1 such that $s \sim t$. Thus $M \prec M(\mathcal{T}_1, \nu_1, \delta_1)$. Conversely, for any t in S_1, there exist s in S and s' in S' such that $\varphi_{s'}(s) = t$. It follows that for any t in S_1, there exists s in S such that $t \sim s$. Thus $M(\mathcal{T}_1, \nu_1, \delta_1) \prec M$. We conclude that $M \sim M(\mathcal{T}_1, \nu_1, \delta_1)$.

We prove $M(\mathcal{T}_1, \nu_1, \delta_1) \in \mathcal{M}'(M')$. It is sufficient to prove that for any s in S and any s' in S', if s' τ-matches s, then $T_\tau(s, s') \in \mathcal{T}_{s'}'$. Suppose that (x_0, y_0), (x_1, y_1), ..., (x_τ, y_τ) are the labels of arcs in a path from the root of $T_\tau(s, s')$. Clearly, for any $t = \langle T_\tau(s, s'), j \rangle$ in S_1, we have $\lambda_1(t, x_0 \dots x_\tau) = y_0 \dots y_\tau$. Since s' τ-matches s, there exists j such that $\langle T_\tau(s, s'), j \rangle = \varphi_{s'}(s)$. From the result shown previously, it follows that $\langle T_\tau(s, s'), j \rangle \sim s$. Thus

$\lambda(s, x_0 \dots x_\tau) = y_0 \dots y_\tau$. Therefore,

$$\begin{aligned}\lambda'(s', y_0 \dots y_\tau) &= \lambda'(s', y_0 \dots y_{\tau-1})\lambda'(\delta'(s', y_0 \dots y_{\tau-1}), y_\tau) \\ &= \lambda'(s', y_0 \dots y_{\tau-1})x_0.\end{aligned}$$

It follows that $f(y_\tau, \dots, y_0, s') = x_0$. We conclude $T_\tau(s, s') \in \mathcal{T}_{s'}$. □

From Lemmas 6.5.1 and 6.5.2, we obtain the following theorem.

Theorem 6.5.1. *Let M' be a weak inverse finite automaton with delay τ. Then M' is a weak inverse with delay τ of a finite automaton M if and only if there exists M_1 in $\mathcal{M}'(M')$ such that M and M_1 are equivalent.*

Theorem 6.5.2. *Let $M' = \langle Y, X, S', \delta', \lambda' \rangle$ be a weak inverse finite automaton with delay τ. Then M' is a weak inverse with delay τ of a finite automaton $M = \langle X, Y, S, \delta, \lambda \rangle$ if and only if there exists M_2 in $\bar{\mathcal{M}}'(M')$ such that $M \prec M_2$.*

Proof. if : Suppose that there exists M_2 in $\bar{\mathcal{M}}'(M')$ such that $M \prec M_2$. Clearly, $M_2 \in \mathcal{M}'(M')$. From Theorem 6.5.1, M' is a weak inverse finite automaton with delay τ of M_2. Since $M \prec M_2$, M' is a weak inverse finite automaton with delay τ of M.

only if : Suppose that M' is a weak inverse finite automaton with delay τ of a finite automaton M. From Theorem 6.5.1, there exists $M(\mathcal{T}_1, \nu_1, \delta_1)$ in $\mathcal{M}'(M')$ such that $M \sim M(\mathcal{T}_1, \nu_1, \delta_1)$. Let

$$\nu_2(T) = \begin{cases} \nu_1(T), & \text{if } T \in \mathcal{T}_1, \\ 1, & \text{if } T \in \mathcal{T}'_m \setminus \mathcal{T}_1, \end{cases}$$

$$\delta_2(\langle T, i \rangle, x) = \begin{cases} \delta_1(\langle T, i \rangle, x), & \text{if } T \in \mathcal{T}_1, \\ \langle \bar{T}, 1 \rangle, & \text{if } T \in \mathcal{T}'_m \setminus \mathcal{T}_1, \end{cases}$$

where $\bar{T}$ is an arbitrarily fixed x-successor of T in $\mathcal{T}'_m$. Then $M(\mathcal{T}'_m, \nu_2, \delta_2) \in \bar{\mathcal{M}}'(M')$. Choose $M(\mathcal{T}'_m, \nu_2, \delta_2)$ as M_2. Clearly, $M(\mathcal{T}_1, \nu_1, \delta_1)$ is a finite subautomaton of M_2. From $M \sim M(\mathcal{T}_1, \nu_1, \delta_1)$, this yields $M \prec M_2$. □

Construct $M(\mathcal{T}'_m) = \langle X, Y, \mathcal{T}'_m, \delta, \lambda \rangle$ as follows:

$$\begin{aligned}&\delta(T, x) = \{\bar{T} \mid \bar{T} \in \mathcal{T}'_m \text{ is an } x\text{-successor of } T\}, \\ &\lambda(T, x) = \{y\}, \\ &\quad T \in \mathcal{T}'_m,\ x \in X,\end{aligned}$$

where (x, y) is the label of an arc emitted from the root of T. Clearly, $M(\mathcal{T}'_m)$ is a nondeterministic finite automaton.

Theorem 6.5.3. (a) *M' is a weak inverse with delay τ of the nondeterministic finite automaton $M(\mathcal{T}'_m)$.*

(b) *If M' is a weak inverse with delay τ of M, then $M \prec M(\mathcal{T}'_m)$.*

Proof. (a) For any state T_0 of $M(\mathcal{T}'_m)$ and any $x_0, x_1, \ldots, x_l$ in X, let $y_0 y_1 \ldots y_l \in \lambda(T_0,\ x_0 x_1 \ldots x_l)$, where $y_0, y_1, \ldots, y_l \in Y$. Then there exist states $T_1, T_2, \ldots, T_{l+1}$ of $M(\mathcal{T}'_m)$ such that $T_{i+1} \in \delta(T_i, x_i)$ and $y_i \in \lambda(T_i, x_i)$ hold for $i = 0, 1, \ldots, l$. From the definition of δ, it is easy to see that for any i, $0 \leqslant i \leqslant l - \tau$, (x_i, y_i), (x_{i+1}, y_{i+1}), …, $(x_{i+\tau}, y_{i+\tau})$ are labels of arcs in a path from the root to a leaf of T_i. This yields $f(y_{i+\tau}, \ldots, y_i, s'_i) = x_i$, $i = 0, 1, \ldots, l - \tau$, where s'_i is the label of the root vertex of T_i, $i = 0, 1, \ldots, l$. It follows that $\lambda'(\delta'(s'_i, y_i \ldots y_{i+\tau-1}), y_{i+\tau}) = x_i$, $i = 0, 1, \ldots, l - \tau$. From the definition of $M(\mathcal{T}'_m)$, it is easy to show that $s'_{i+1} = \delta'(s'_i, y_i)$, $i = 0, 1, \ldots, l$. Thus $\lambda'(s'_{i+\tau}, y_{i+\tau}) = x_i$, $i = 0, 1, \ldots, l - \tau$. Therefore, $\lambda'(s'_0, y_0 y_1 \ldots y_l) = x_{-\tau} \ldots x_{-1} x_0 x_1 \ldots x_{l-\tau}$ holds for some $x_{-\tau}, \ldots, x_{-1}$ in X. We conclude that M' is a weak inverse finite automaton with delay τ of $M(\mathcal{T}'_m)$.

(b) Similar to the proof of Theorem 6.3.3 (b), from Theorem 6.5.2, there exists $M_2 = M(\mathcal{T}'_m, \nu, \delta_2)$ such that $M \prec M_2$. We prove $M_2 \prec M(\mathcal{T}'_m)$. For any state $\langle T_0, j_0 \rangle$ of M_2, T_0 is a state of $M(\mathcal{T}'m)$. To prove $\langle T_0, j_0 \rangle \prec T_0$, let $x_0, x_1, \ldots, x_l \in X$ and $y_0 y_1 \ldots y_l = \lambda_2(\langle T_0, j_0 \rangle, x_0 x_1 \ldots x_l)$, where $y_0, y_1, \ldots, y_l$ are in Y, and λ_2 is the output function of M_2. Then there exist states $\langle T_i, j_i \rangle$, $i = 1, \ldots, l + 1$ of M_2 such that $\delta_2(\langle T_i, j_i \rangle, x_i) = \langle T_{i+1}, j_{i+1} \rangle$ and $\lambda_2(\langle T_i, j_i \rangle, x_i) = y_i$, $i = 0, 1, \ldots, l$, where δ_2 is the next state function of M_2. From the definition of $M(\mathcal{T}'_m, \nu, \delta_2)$, T_{i+1} is an x_i-successor of T_i and (x_i, y_i) is the label of an arc emitted from the root of T_i. From the definition of $M(\mathcal{T}'_m)$, $y_0 y_1 \ldots y_l \in \lambda(T_0, x_0 x_1 \ldots x_l)$. Thus $\langle T_0, j_0 \rangle \prec T_0$. We conclude $M_2 \prec M(\mathcal{T}'_m)$. From $M \prec M_2$, we have $M \prec M(\mathcal{T}'_m)$. □

This theorem means that $M(\mathcal{T}'_m)$ is a "universal" nondeterministic finite automaton for finite automata of which M' is a weak inverse with delay τ.

6.6 Weak Inverses with Bounded Error Propagation of a Finite Automaton

Let $M_a = \langle Y_a, S_a, \delta_a, \lambda_a \rangle$ be an autonomous finite automaton, and f a partial function from $X^{c+1} \times \lambda_a(S_a)$ to Y. We also use $\mathcal{SIM}(M_a, f)$ to denote a partial finite automaton $\langle X, Y, X^c \times S_a, \delta, \lambda \rangle$, where

$$\delta(\langle x_{-1}, \ldots, x_{-c}, s_a \rangle, x_0) = \langle x_0, \ldots, x_{-c+1}, \delta_a(s_a) \rangle,$$
$$\lambda(\langle x_{-1}, \ldots, x_{-c}, s_a \rangle, x_0) = f(x_0, x_{-1}, \ldots, x_{-c}, \lambda_a(s_a)),$$
$$x_0, x_{-1}, \ldots, x_{-c} \in X,\ s_a \in S_a.$$

$\mathcal{SIM}(M_a, f)$ is called a *c-order semi-input-memory partial finite automaton* determined by M_a and f. Clearly, a c-order semi-input-memory finite automaton is a special c-order semi-input-memory partial finite automaton.

Let $M = \langle X, Y, S, \delta, \lambda\rangle$ and $M' = \langle Y, X, S', \delta', \lambda'\rangle$ be two finite automata. Assume that M' is a weak inverse finite automaton with delay τ of M and that propagation of weakly decoding errors of M' to M is bounded with length of error propagation $\leqslant c$, where $c \geqslant \tau$. Similar to the proof of Theorem 1.5.1, we can construct a c-order semi-input-memory partial finite automaton as follows.

Since M' is a weak inverse finite automaton with delay τ of M and propagation of weakly decoding errors of M' to M is bounded with length of error propagation $\leqslant c$, for each s in S, we can choose a state of M', say $\varphi(s)$, such that $(s, \varphi(s))$ is a (τ, c)-match pair. Take a subset I of S with $\{\delta(s, \alpha) \mid s \in I, \alpha \in X^*\} = S$.

We first construct an autonomous finite automaton $M'' = \langle Y'', S'', \delta'', \lambda''\rangle$ mentioned in the proof of Theorem 1.5.1.

For any subset T of S, let $R(T) = \{\lambda(t, \alpha) \mid t \in T, \alpha \in X^*\}$. For any state $w_{s,i}$ of M'' and any $y_i, \ldots, y_{i-c}$ in Y, we define $f(y_i, \ldots, y_{i-c}, w_{s,i})$ as follows. When $i \geqslant c$ and $y_{i-c} \ldots y_i \in R(T_{s,i-c})$, define $f(y_i, \ldots, y_{i-c}, w_{s,i}) = \lambda'(\delta'(s'_{i-c}, y_{i-c} \ldots y_{i-1}), y_i)$, s'_{i-c} being a state in $T'_{s,i-c}$. From the proof of Theorem 1.5.1, this value of f is independent of the choice of s'_{i-c}. When $\tau \leqslant i < c$ and $y_0 \ldots y_i \in R(T_{s,0})$, define $f(y_i, \ldots, y_{i-c}, w_{s,i}) = x_{i-\tau}$, where $y_0 \ldots y_i = \lambda(s, x_0 \ldots x_i)$ for some x_0, …, x_i in X. From the proof of Theorem 1.5.1, the value of $x_{i-\tau}$ is uniquely determined by i and y_0, …, y_i. Otherwise, the value of $f(y_i, \ldots, y_{i-c}, w_{s,i})$ is undefined. We then construct a c-order semi-input-memory partial finite automaton $\mathcal{SIM}(M'', f)$ from M'' and f.

Lemma 6.6.1. *$\mathcal{SIM}(M'', f)$ is a weak inverse partial finite automaton with delay τ of M. Furthermore, for any s in I, the state $s''' = \langle y_{-c}, \ldots, y_{-1}, w_{s,0}\rangle$ of $\mathcal{SIM}(M'', f)$ τ-matches s, where y_{-1}, …, y_{-c} are arbitrary elements in Y.*

Proof. Similar to the proof of Theorem 1.5.1, for any $x_0, \ldots, x_j$ in X, let $y_0 \ldots y_j = \lambda(s, x_0 \ldots x_j)$ and $z_0 \ldots z_j = \lambda'''(s''', y_0 \ldots y_j)$, where λ''' is the output function of $\mathcal{SIM}(M'', f)$. We prove that $z_i = x_{i-\tau}$ holds for any $\tau \leqslant i \leqslant j$. In the case of $\tau \leqslant i < c$, since $y_0 \ldots y_i \in R(T_{s,0})$, from the construction of $\mathcal{SIM}(M'', f)$ and the definition of f, it immediately follows that $z_i = x_{i-\tau}$. In the case of $i \geqslant c$, take $h = i$ if $i < t_s + c + e_s$, and take $h = t_s + c + d$ if $i = t_s + c + d + ke_s$ for $k > 0$ and $0 \leqslant d < e_s$. Since $h - c \geqslant t_s$ and $h = i \pmod{e_s}$, or $h = i$, we have $(T_{s,i-c}, T'_{s,i-c}) = (T_{s,h-c}, T'_{s,h-c})$. Let $s_{i-c} = \delta(s, x_0 \ldots x_{i-c-1})$ and $s'_{i-c} = \delta'(\varphi(s), y_0 \ldots y_{i-c-1})$. Then we have $s_{i-c} \in$

$T_{s,h-c}$, $s'_{i-c} \in T'_{s,h-c}$, and $y_{i-c} \dots y_i \in R(T_{s,h-c})$. From the construction of $\mathcal{SIM}(M'', f)$ and the definition of f, it is easy to show that

$$\begin{aligned}
&\delta'''(s''', y_0 \dots y_{i-1}) = \langle y_{i-1}, \dots, y_{i-c}, w_{s,h} \rangle, \\
&z_i \ = \lambda'''(\langle y_{i-1}, \dots, y_{i-c}, w_{s,h} \rangle, y_i) \\
&\quad = f(y_i, y_{i-1}, \dots, y_{i-c}, w_{s,h}) = \lambda'(\delta'(s'_{i-c}, y_{i-c} \dots y_{i-1}), y_i),
\end{aligned}$$

where δ''' is the next state function of $\mathcal{SIM}(M'', f)$. Since $(s, \varphi(s))$ is a (τ, c)-match pair, we have

$$\lambda'(\delta'(s'_{i-c}, y_{i-c} \dots y_{i-1}), y_i) = \lambda'(\delta'(\varphi(s), y_0 \dots y_{i-1}), y_i) = x_{i-\tau}.$$

It follows that $z_i = x_{i-\tau}$. Therefore, s''' τ-matches s. It follows that $\delta'''(s''', \beta)$ τ-matches $\delta(s, \alpha)$, if $\beta = \lambda(s, \alpha)$. From $S = \{\delta(s, \alpha) \mid s \in I, \alpha \in X^*\}$, for any s in S, there exists a state s' of $\mathcal{SIM}(M'', f)$ such that s' τ-matches s. Thus $\mathcal{SIM}(M'', f)$ is a weak inverse partial finite automaton with delay τ of M. □

Lemma 6.6.2. *$\mathcal{SIM}(M'', f) \prec M'$.*

Proof. For any state $s''' = \langle y_{-1}, \dots, y_{-c}, w_{s,0} \rangle$ of $\mathcal{SIM}(M'', f)$, where $s \in I$ and y_{-1}, …, $y_{-c} \in Y$, we prove $s''' \prec \varphi(s)$. Let y_0, …, $y_j \in Y$, $j \geqslant 0$. From the construction of $\mathcal{SIM}(M'', f)$, $y_0 \dots y_j$ is applicable to s''', i.e., $j = 0$ or $\delta'''(s''', y_0 \dots y_{j-1})$ is defined, where δ''' is the next state function of $\mathcal{SIM}(M'', f)$. Let $\lambda'''(s''', y_0 \dots y_j) = x'''_0 \dots x'''_j$ and $\lambda'(\varphi(s), y_0 \dots y_j) = x'_0 \dots x'_j$, where λ''' is the output function of $\mathcal{SIM}(M'', f)$, and $x'''_i \in X \cup \{\text{-}\}$, $x'_i \in X$, $i = 0, 1, \dots, j$. We prove $x'''_i \prec x'_i$, i.e., $x'''_i = x'_i$ whenever x'''_i is defined, for $i = 0, 1, \dots, j$. There are three cases to consider.

In the case of $\tau \leqslant i < c$ and $y_0 \dots y_i \in R(T_{s,0})$, there exist x_0, …, x_i in X such that $\lambda(s, x_0 \dots x_i) = y_0 \dots y_i$. From the construction of $\mathcal{SIM}(M'', f)$, we have $x'''_i = x_{i-\tau}$. Since $(s, \varphi(s))$ is a match pair with delay τ, there exist $x_{-\tau}$, …, x_{-1} in X such that $\lambda'(\varphi(s), y_0 \dots y_i) = x_{-\tau} \dots x_{-1} x_0 \dots x_{i-\tau}$. It immediately follows that $x'_i = x_{i-\tau}$. Therefore, we have $x'''_i = x'_i$. This yields $x'''_i \prec x'_i$.

In the case of $i \geqslant c$ and $y_{i-c} \dots y_i \in R(T_{s,i-c})$, from the construction of $\mathcal{SIM}(M'', f)$, we have $x'''_i = \lambda'(\delta'(s'_{i-c}, y_{i-c} \dots y_{i-1}), y_i)$, for any s'_{i-c} in $T'_{s,i-c}$. Since $y_{i-c} \dots y_i$ is in $R(T_{s,i-c})$, there exist s_{i-c} in $T_{s,i-c}$ and x_{i-c}, …, x_i in X such that $\lambda(s_{i-c}, x_{i-c} \dots x_i) = y_{i-c} \dots y_i$. From the definition of $T_{s,i-c}$, there exist x_0, …, x_{i-c-1} in X such that $\delta(s, x_0 \dots x_{i-c-1}) = s_{i-c}$. Let $\lambda(s, x_0 \dots x_{i-c-1}) = y'_0 \dots y'_{i-c-1}$, where y'_0, …, $y'_{i-c-1} \in Y$. Then we have $\lambda(s, x_0 \dots x_i) = y'_0 \dots y'_{i-c-1}\ y_{i-c} \dots y_i$. Take $s'_{i-c} = \delta'(\varphi(s), y'_0 \dots y'_{i-c-1})$. Clearly, s'_{i-c} is in $T'_{s,i-c}$. Thus for this s'_{i-c}, $\lambda'(\delta'(s'_{i-c}, y_{i-c} \dots y_{i-1}), y_i) = x'''_i$. Since $(s, \varphi(s))$ is a (τ, c)-match pair, we have

$$\begin{aligned} x_i''' &= \lambda'(\delta'(s'_{i-c}, y_{i-c} \dots y_{i-1}), y_i) \\ &= \lambda'(\delta'(\varphi(s), y'_0 \dots y'_{i-c-1} y_{i-c} \dots y_{i-1}), y_i) \\ &= \lambda'(\delta'(\varphi(s), y_0 \dots y_{i-1}), y_i) \\ &= x'_i. \end{aligned}$$

It immediately follows that $x_i''' \prec x'_i$.

Otherwise, from the construction of $\mathcal{SIM}(M'', f)$, x_i''' is undefined. It immediately follows that $x_i''' \prec x'_i$.

We have proven that $\langle y_{-1}, \dots, y_{-c}, w_{s,0}\rangle \prec \varphi(s)$ for any $s \in I$ and any $y_{-1}, \dots, y_{-c} \in Y$. For any state $\bar{s}'''$ of $\mathcal{SIM}(M'', f)$, from the construction of $\mathcal{SIM}(M'', f)$, there exist s in I and $y_{i-c}, \dots, y_{i-1}$ in Y, $0 \leqslant i \leqslant t_s + c + e_s - 1$, such that $\bar{s}''' = \langle y_{i-1}, \dots, y_{i-c}, w_{s,i}\rangle$. Let $s''' = \langle y_{-1}, \dots, y_{-c}, w_{s,0}\rangle$ be a state of $\mathcal{SIM}(M'', f)$. Then $\delta'''(s''', y_0 \dots y_{i-1}) = \bar{s}'''$ holds for any $y_0, \dots, y_{i-c-1}$ in Y. Since $s''' \prec \varphi(s)$, we have $\delta'''(s''', y_0 \dots y_{i-1}) \prec \delta'(\varphi(s), y_0 \dots y_{i-1})$, i.e., $\bar{s}''' \prec \delta'(\varphi(s), y_0 \dots y_{i-1})$. We conclude that $\mathcal{SIM}(M'', f) \prec M'$. □

Let $\bar{M}' = \langle Y, X, \bar{S}', \bar{\delta}', \bar{\lambda}'\rangle$ be a finite automaton. Assume that $\bar{M}'$ is a weak inverse with delay τ of M and that propagation of weakly decoding errors of $\bar{M}'$ to M is bounded with length of error propagation $\leqslant \bar{c}$, where $\bar{c} \geqslant \tau$. Similar to the constructing method of $\mathcal{SIM}(M'', f)$ from M and M', we can construct a $\bar{c}$-order semi-input-memory partial finite automaton $\mathcal{SIM}(\bar{M}'', \bar{f})$ from M and $\bar{M}'$, replacing φ, $T'_{s,i}$, M'', f, c, δ', λ', … by $\bar{\varphi}$, $\bar{T}'_{s,i}$, $\bar{M}''$, $\bar{f}$, $\bar{c}$, $\bar{\delta}'$, $\bar{\lambda}'$, …, respectively.

Lemma 6.6.3. *If $\bar{c} \leqslant c$, then $\mathcal{SIM}(M'', f) \prec \mathcal{SIM}(\bar{M}'', \bar{f})$.*

Proof. From the construction of $\mathcal{SIM}(M'', f)$ and $\mathcal{SIM}(\bar{M}'', \bar{f})$, it is sufficient to prove that for any $s \in I$ and any $y_{-1}, \dots, y_{-c} \in Y$, the state $\bar{s}'''$ of $\mathcal{SIM}(\bar{M}'', \bar{f})$ is stronger than the state s''' of $\mathcal{SIM}(M'', f)$, where $s''' = \langle y_{-1}, \dots, y_{-c}, w_{s,0}\rangle$, $\bar{s}''' = \langle y_{-1}, \dots, y_{-\bar{c}}, w_{s,0}\rangle$. Since any β in Y^* is applicable to s''' and $\bar{s}'''$, $s''' \prec \bar{s}'''$ if and only if for any β in Y^*, $\lambda'''(s''', \beta) \prec \bar{\lambda}'''(\bar{s}''', \beta)$, where λ''' and $\bar{\lambda}'''$ are output functions of $\mathcal{SIM}(M'', f)$ and $\mathcal{SIM}(\bar{M}'', \bar{f})$, respectively.

For any $y_0, \dots, y_j$ in Y, $j \geqslant 0$, let $x_0''' \dots x_j''' = \lambda'''(s''', y_0 \dots y_j)$ and $\bar{x}_0''' \dots \bar{x}_j''' = \bar{\lambda}'''(\bar{s}''', y_0 \dots y_j)$, where x_i''', $\bar{x}_i'''$, $i = 0, 1, \dots, j$ are in $X \cup \{\text{-}\}$. We prove $x_i''' \prec \bar{x}_i'''$, $i = 0, 1, \dots, j$. There are four cases to consider.

In the case of $\tau \leqslant i < \bar{c}$ and $y_0 \dots y_i \in R(T_{s,0})$, there exist $x_0, \dots, x_i$ in X such that $\lambda(s, x_0 \dots x_i) = y_0 \dots y_i$. From the construction of $\mathcal{SIM}(M'', f)$ and $\mathcal{SIM}(\bar{M}'', \bar{f})$, using $\bar{c} \leqslant c$, we have $x_i''' = x_{i-\tau}$ and $\bar{x}_i''' = x_{i-\tau}$. It immediately follows $x_i''' = \bar{x}_i'''$, therefore, $x_i''' \prec \bar{x}_i'''$.

In the case of $\bar{c} \leqslant i < c$ and $y_0 \dots y_i \in R(T_{s,0})$, there exist $x_0, \dots, x_i$ in X such that $\lambda(s, x_0 \dots x_i) = y_0 \dots y_i$. From the construction of $\mathcal{SIM}(M'', f)$,

we have $x_i''' = x_{i-\tau}$. Using Lemma 6.6.1 (in the version of $\mathcal{SIM}(\bar{M}'', \bar{f})$), we have $\bar{\lambda}'''(\bar{s}''', y_0 \dots y_i) = x_{-\tau} \dots x_{-1} x_0 \dots x_{i-\tau}$ for some $x_{-\tau}$, …, x_{-1} in $X \cup \{\_\}$. It immediately follows that $\bar{x}_i''' = x_{i-\tau}$. Therefore, $x_i''' = \bar{x}_i'''$. This yields $x_i''' \prec \bar{x}_i'''$.

In the case of $i \geqslant c$ and $y_{i-c} \dots y_i \in R(T_{s,i-c})$, there exist s_{i-c} in $T_{s,i-c}$ and x_{i-c}, …, x_i in X such that $\lambda(s_{i-c}, x_{i-c} \dots x_i) = y_{i-c} \dots y_i$. From $s_{i-c} \in T_{s,i-c}$, there exist x_0, …, x_{i-c-1} in X such that $\delta(s, x_0 \dots x_{i-c-1}) = s_{i-c}$. Let $\lambda(s, x_0 \dots x_{i-c-1}) = y_0' \dots y_{i-c-1}'$, where y_0', …, $y_{i-c-1}' \in Y$. Then we have $\lambda(s, x_0 \dots x_i) = y_0' \dots y_{i-c-1}' y_{i-c} \dots y_i$. Using Lemma 6.6.1, it is easy to see that $\lambda'''(\delta'''(s''', y_0' \dots y_{i-c-1}' y_{i-c} \dots y_{i-1}), y_i) = x_{i-\tau}$. Similarly, using Lemma 6.6.1 (in the version of $\mathcal{SIM}(\bar{M}'', \bar{f})$), we have $\bar{\lambda}'''(\bar{\delta}'''(\bar{s}''', y_0' \dots y_{i-c-1}' \; y_{i-c} \dots y_{i-1}), y_i) = x_{i-\tau}$. Since

$$\begin{aligned}\lambda'''(\delta'''(s''', y_0 \dots y_{i-1}), y_i) &= \lambda'''(\langle y_{i-1}, \dots, y_{i-c}, \delta'''^{i}(w_{s,0})\rangle, y_i) \\ &= \lambda'''(\delta'''(s''', y_0' \dots y_{i-c-1}' y_{i-c} \dots y_{i-1}), y_i) = x_{i-\tau},\end{aligned}$$

we have $x_i''' = x_{i-\tau}$. Similarly, we can show $\bar{x}_i''' = x_{i-\tau}$. Therefore, $x_i''' = \bar{x}_i'''$. This yields $x_i''' \prec \bar{x}_i'''$.

Otherwise, from the construction of $\mathcal{SIM}(M'', f)$, x_i''' is undefined. It immediately follows that $x_i''' \prec \bar{x}_i'''$. □

Theorem 6.6.1. *Let $M = \langle X, Y, S, \delta, \lambda\rangle$ and $M' = \langle Y, X, S', \delta', \lambda'\rangle$ be two finite automata. Assume that M' is a weak inverse finite automaton with delay τ of M and that propagation of weakly decoding errors of M' to M is bounded with length of error propagation $\leqslant c$, where $c \geqslant \tau$. Let $\mathcal{SIM}(M'', f)$ be a c-order semi-input-memory partial finite automaton constructed from M and M'. Then for any finite automaton $\bar{M}' = \langle Y, X, \bar{S}', \bar{\delta}', \bar{\lambda}'\rangle$, $\bar{M}'$ is a weak inverse finite automaton with delay τ of M and propagation of weakly decoding errors of $\bar{M}'$ to M is bounded with length of error propagation $\leqslant c$, if and only if $\mathcal{SIM}(M'', f) \prec \bar{M}'$.*

Proof. only if : Suppose that $\bar{M}'$ is a weak inverse finite automaton with delay τ of M and that propagation of weakly decoding errors of $\bar{M}'$ to M is bounded with length of error propagation $\leqslant c$. Let $\mathcal{SIM}(\bar{M}'', \bar{f})$ be a c-order semi-input-memory partial finite automaton constructed from M and $\bar{M}'$. From Lemma 6.6.2 (in the version of $\mathcal{SIM}(\bar{M}'', \bar{f})$ and $\bar{M}'$), we have $\mathcal{SIM}(\bar{M}'', \bar{f}) \prec \bar{M}'$. From Lemma 6.6.3, $\mathcal{SIM}(M'', f) \prec \mathcal{SIM}(\bar{M}'', \bar{f})$ holds. It follows that $\mathcal{SIM}(M'', f) \prec \bar{M}'$.

if : Suppose $\mathcal{SIM}(M'', f) \prec \bar{M}'$. From Theorem 6.1.2, there exist a closed compatible family C_1, …, C_k of $\mathcal{SIM}(M'', f)$, a partial finite automaton M_1 in $\mathcal{M}(C_1, \dots, C_k)$, and a partial finite subautomaton M_2 of $\bar{M}'$ such that M_1 and M_2 are isomorphic. It follows that there exists a finite automaton $M_3 = \langle Y, X, S_3, \delta_3, \lambda_3\rangle$ such that M_1 is a partial finite subautomaton of M_3

and M_3 is isomorphic to $\bar{M}'$. Therefore, proving the *if* part is equivalent to proving that M_3 is a weak inverse finite automaton with delay τ of M and propagation of weakly decoding errors of M_3 to M is bounded with length of error propagation $\leqslant c$. Since $S = \{\delta(s, \alpha) \mid s \in I, \alpha \in X^*\}$, it is sufficient to prove that for any s in I, there exists a state s_3 of M_3 such that (s, s_3) is a (τ, c)-match pair.

Given arbitrarily s in I, we fix arbitrary c elements in Y, say $y_{-c}, \ldots, y_{-1}$, and use s''' to denote the state $\langle y_{-1}, \ldots, y_{-c}, w_{s,0}\rangle$ of $\mathcal{SIM}(M'', f)$. Suppose that $s''' \in C_h$ for some h, $1 \leqslant h \leqslant k$. Since $\cup_{1 \leqslant i \leqslant k} C_i$ is the state alphabet of $\mathcal{SIM}(M'', f)$, such an h is existent. From Lemma 6.6.1, s''' τ-matches s. From Lemma 6.1.2, we have $s''' \prec c_h$, where c_h is a state of M_1 corresponding to C_h. Since M_1 is a partial finite subautomaton of M_3, c_h is also a state of M_3 and $s''' \prec c_h$ also holds. This yields that c_h τ-matches s.

We consider the error propagation. Given arbitrarily $x_0, \ldots, x_l$ in X, $l \geqslant 0$, let $\lambda(s, x_0 \ldots x_l) = y_0 \ldots y_l$ and $\lambda_3(c_h, y_0 \ldots y_l) = x''_0 \ldots x''_l$, where $y_i \in Y$, $x''_i \in X$, $i = 0, 1, \ldots, l$. Given arbitrarily $y'_0, \ldots, y'_{n-1} \in Y$, $n \leqslant l$, let $\lambda_3(c_h, y'_0 \ldots y'_{n-1} y_n \ldots y_l) = x'_0 \ldots x'_l$, where $x'_i \in X$, $i = 0, 1, \ldots, l$. We prove $x''_{n+c} \ldots x''_l = x'_{n+c} \ldots x'_l$. In the case of $n + c > l$, this is trivial. In the case of $n + c \leqslant l$, let $W = \{w \mid \exists \bar{y}_{-c}, \ldots, \bar{y}_{-1} \in Y\ (\langle \bar{y}_{-1}, \ldots, \bar{y}_{-c}, w\rangle \in C_h)\}$. For any r, $n + c \leqslant r \leqslant l$, we have $r - c \geqslant n$. Since M_1 is a partial finite subautomaton of M_3 and $M_1 \in \mathcal{M}(C_1, \ldots, C_k)$, from the construction of M_1, it is easy to see that

$$C_i \supseteq \delta'''(C_h, y_0 \ldots y_{r-1}) = \{\langle y_{r-1}, \ldots, y_{r-c}, \delta'''^r(w)\rangle, w \in W\},$$
$$C_j \supseteq \delta'''(C_h, y'_0 \ldots y'_{n-1} y_n \ldots y_{r-1}) = \{\langle y_{r-1}, \ldots, y_{r-c}, \delta'''^r(w)\rangle, w \in W\},$$

where C_i and C_j correspond to $c_i = \delta_1(c_h, y_0 \ldots y_{r-1})$ and $c_j = \delta_1(c_h, y'_0 \ldots y'_{n-1}\ y_n \ \ldots \ y_{r-1})$, respectively, and δ''' and δ_1 are the next state functions of $\mathcal{SIM}(M'', f)$ and M_1, respectively. Let $s'''_r = \delta'''(s''', y_0 \ldots y_{r-1})$. From $r - c \geqslant n$, we have $s'''_r = \delta'''(s''', y'_0 \ldots y'_{n-1} y_n \ldots y_{r-1})$. It follows that $s'''_r \in C_i$ and $s'''_r \in C_j$. Since s''' τ-matches s, we have $\lambda'''(s'''_r, y_r) = x_{r-\tau}$, where λ''' is the output function of $\mathcal{SIM}(M'', f)$. From the construction of M_1, we have $\lambda_1(c_i, y_r) = \lambda'''(s'''_r, y_r) = x_{r-\tau}$ and $\lambda_1(c_j, y_r) = \lambda'''(s'''_r, y_r) = x_{r-\tau}$, where λ_1 is the output function of M_1. It follows that $\lambda_1(c_i, y_r) = \lambda_1(c_j, y_r)$. Since M_1 is a partial finite subautomaton of M_3, we have $\lambda_1(c_i, y_r) = x''_r$ and $\lambda_1(c_j, y_r) = x'_r$. Therefore, $x'_r = x''_r$.

We conclude that (s, c_h) is a (τ, c)-match pair. Taking $s_3 = c_h$, then s_3 satisfies the condition: (s, s_3) is a (τ, c)-match pair. □

Corollary 6.6.1. *Let $M = \langle X, Y, S, \delta, \lambda\rangle$ and $M' = \langle Y, X, S', \delta', \lambda'\rangle$ be two finite automata. Assume that M' is a weak inverse finite automaton with delay τ of M and that propagation of weakly decoding errors of M' to M is*

bounded with length of error propagation $\leqslant c$, *where* $c \geqslant \tau$. *Let* $\mathcal{SIM}(M'', f)$ *be a c-order semi-input-memory partial finite automaton constructed from* M *and* M'. *Then for any finite automaton* $\bar{M}' = \langle Y, X, \bar{S}', \bar{\delta}', \bar{\lambda}' \rangle, \bar{M}'$ *is a weak inverse finite automaton with delay* τ *of* M *and propagation of weakly decoding errors of* $\bar{M}'$ *to* M *is bounded with length of error propagation* $\leqslant c$, *if and only if there exist a closed compatible family* $C_1, \ldots, C_k$ *of* $\mathcal{SIM}(M'', f)$, *a partial finite automaton* M_1 *in* $\mathcal{M}(C_1, \ldots, C_k)$, *and a finite automaton* M_3 *such that* M_1 *is a partial finite subautomaton of* M_3 *and* M_3 *is isomorphic to* $\bar{M}'$.

Historical Notes

Partial finite automata are first discussed in [58, 73]. The concept of $\prec$ is introduced in [48], and the concept of compatibility is introduced in [3]. Subsection 6.1.1 is based on [80]. Nondeterministic finite automata are defined in [86] for the proof of Kleene Theorem, and a systematic development first appears in [94], see also [95]. Structures of some kinds of finite automata with invertibility are studied in [64]. Given a finite automaton, the structures of its inverses, its original inverses, its weak inverses, its original weak inverses, and its weak inverses with bounded error propagation are characterized in [111, 18, 19, 20]. Sections 6.2 and 6.4 are based on [19]. Sections 6.3 and 6.5 are based on [18]. Section 6.6 is based on [20].

7. Linear Autonomous Finite Automata

Renji Tao

Institute of Software, Chinese Academy of Sciences
Beijing 100080, China trj@ios.ac.cn

Summary.

Autonomous finite automata are regarded as sequence generators. For the general case, the set of output sequences of an autonomous finite automaton consists of ultimately periodic sequences and is closed under translation operation. From a mathematical viewpoint, such sets have been clearly characterized, although such a characterization is not very useful to cryptology. On the other hand, nonlinear autonomous finite automata can be linearized. So we confine ourself to the linear case in this chapter. Notice that each linear autonomous finite automaton with output dimension 1 is equivalent to a linear shift register and that linear shift registers as a special case of linear autonomous finite automata have been so intensively and extensively studied. In this chapter, we focus on the case of arbitrary output dimension. After reviewing some preliminary results of combinatory theory, we deal with representation, translation, period, and linearization for output sequences of linear autonomous finite automata. A result of decimation of linear shift register sequences is also presented.

Key words: *autonomous finite automata, linear shift register, root representation, translation, period, linearization, decimation*

Autonomous finite automata are regarded as sequence generators. For the general case, the set of output sequences of an autonomous finite automaton consists of ultimately periodic sequences and is closed under translation operation. From a mathematical viewpoint, such sets have been clearly characterized, although such a characterization is not very useful to cryptology. On the other hand, nonlinear autonomous finite automata can be linearized. So we confine ourself to the linear case in this chapter. Notice that each linear autonomous finite automaton with output dimension 1 is equivalent to a linear shift register and that linear shift registers as a special case of linear autonomous finite automata have been so intensively and extensively studied.

In this chapter, we focus on the case of arbitrary output dimension. After reviewing some preliminary results of combinatory theory, we give several representations for output sequences of linear autonomous finite automata. Then translation, period, and linearization for output sequences of linear autonomous finite automata are discussed. Finally, we present a result of decimation of linear shift register sequences.

7.1 Binomial Coefficient

Let n and r be two integers. We define

$$\begin{aligned}
&\binom{n}{r} = n(n-1)\dots(n-r+1)/r!, \qquad \text{if } r > 0,\\
&\binom{n}{0} = 1,\\
&\binom{n}{r} = 0, \qquad \text{if } n \geqslant 0,\ r < 0,
\end{aligned}$$

where $r!$ stands for the factorial of r, that is, $r! = \prod_{i=1}^{r} i$ for $r > 0$, and $0! = 1$.

Clearly,

$$\binom{n}{r} = \binom{n}{n-r}, \qquad \text{if } n \geqslant r \geqslant 0. \tag{7.1}$$

We prove

$$\binom{n}{r} = \binom{n-1}{r} + \binom{n-1}{r-1}, \qquad \text{if } r > 0 \text{ or } n > 0. \tag{7.2}$$

In the case of $r = 1$, (7.2) is evident. In the case of $r > 1$,

$$\begin{aligned}
&\binom{n-1}{r} + \binom{n-1}{r-1}\\
&\quad = (n-1)(n-2)\dots(n-r)/r! + (n-1)(n-2)\dots(n-r+1)/(r-1)!\\
&\quad = (n-1)(n-2)\dots(n-r+1)(n-r+r)/r!\\
&\quad = \binom{n}{r}.
\end{aligned}$$

In the case of $r \leqslant 0$ and $n > 0$, we have $\binom{n-1}{r-1} = 0$ and $\binom{n}{r} = \binom{n-1}{r}$; therefore, (7.2) holds.

We prove by induction on j the following formula:

$$\sum_{i=0}^{j} \binom{i+r-1}{i} = \binom{j+r}{j}, \qquad \text{if } j \geqslant 0. \tag{7.3}$$

Basis : $j = 0$. We have $\sum_{i=0}^{j} \binom{i+r-1}{i} = \binom{r-1}{0} = 1$ and $\binom{j+r}{j} = \binom{r}{0} = 1$. Thus the equation in (7.3) holds. *Induction step* : Suppose that the equation in (7.3) holds and $j \geqslant 0$. From the induction hypothesis and (7.2), we have

$$\sum_{i=0}^{j+1} \binom{i+r-1}{i} = \sum_{i=0}^{j} \binom{i+r-1}{i} + \binom{j+r}{j+1}$$
$$= \binom{j+r}{j} + \binom{j+r}{j+1} = \binom{j+1+r}{j+1}.$$

That is, the equation in (7.3) holds for $j+1$. We conclude that (7.3) holds.

From (7.3) and (7.1), we obtain

$$\sum_{i=0}^{j} \binom{i+r-1}{r-1} = \binom{j+r}{r}, \qquad \text{if } j \geqslant 0,\ r > 0. \tag{7.4}$$

A polynomial $f(x)$ over the real field is called an *integer-valued* polynomial if for any integer n, $f(n)$ is an integer.

For any nonnegative integer r, let $\binom{x+r}{r} = (x+r)(x+r-1)\ldots(x+1)/r!$ if $r > 0$, and $\binom{x}{0} = 1$. Clearly, $\binom{x+r}{r}$ is an integer-valued polynomial of degree r. It follows that if $a_0, \ldots, a_k$ are integers, then $\sum_{r=0}^{k} a_r \binom{x+r}{r}$ is an integer-valued polynomial. Below we prove that its reverse proposition also holds.

We define a difference operator ∇: $\nabla g(x) = g(x) - g(x-1)$. Let $\nabla^0 g(x) = g(x)$, $\nabla^{r+1} g(x) = \nabla(\nabla^r g(x))$. It is easy to verify that

$$\nabla \binom{x+i}{i} = \binom{x+i-1}{i-1}, \quad \text{if } i > 0, \tag{7.5}$$
$$\nabla \binom{x}{0} = 0.$$

From (7.5), it is easy to prove by induction on r that for any nonnegative integers r and i,

$$\nabla^r \binom{x+i}{i} = \begin{cases} \binom{x+i-r}{i-r}, & \text{if } i \geqslant r, \\ 0, & \text{if } i < r. \end{cases} \tag{7.6}$$

We prove

$$\nabla^r g(x) = \sum_{i=0}^{r} (-1)^i \binom{r}{i} g(x-i), \quad \text{if } r \geqslant 0 \tag{7.7}$$

by induction on r. *Basis* : $r = 0$. Clearly, the two sides of the equation in (7.7) are $g(x)$. Thus the equation in (7.7) holds. *Induction step* : Suppose that the equation in (7.7) holds. From the induction hypothesis and (7.2), we have

$$\nabla^{r+1} g(x) = \nabla^r g(x) - \nabla^r g(x-1)$$
$$= \sum_{i=0}^{r} (-1)^i \binom{r}{i} g(x-i) - \sum_{i=0}^{r} (-1)^i \binom{r}{i} g(x-i-1)$$

$$= \sum_{i=0}^{r}(-1)^i\binom{r}{i}g(x-i) - \sum_{i=1}^{r+1}(-1)^{i-1}\binom{r}{i-1}g(x-i)$$
$$= \sum_{i=0}^{r+1}(-1)^i\left[\binom{r}{i} + \binom{r}{i-1}\right]g(x-i)$$
$$= \sum_{i=0}^{r+1}(-1)^i\binom{r+1}{i}g(x-i).$$

Thus the equation in (7.7) holds for $r+1$. We conclude that (7.7) holds.

Theorem 7.1.1. *Any integer-valued polynomial $f(x)$ of degree k can be uniquely expressed in form $\sum_{r=0}^{k} a_r\binom{x+r}{r}$, where $a_0, a_1, \ldots, a_k$ are integers and $a_k \neq 0$, and*

$$a_r = \nabla^r f(x)|_{x=-1} = \sum_{i=0}^{r}(-1)^i\binom{r}{i}f(-1-i), \qquad (7.8)$$
$$r = 0, 1, \ldots, k.$$

Proof. It is easy to prove by induction on the degree k of $f(x)$ that $f(x)$ can be expressed in the form $\sum_{r=0}^{k} a_r\binom{x+r}{r}$ with $a_k \neq 0$.

Now suppose that

$$f(x) = \sum_{r=0}^{k} a_r\binom{x+r}{r}. \qquad (7.9)$$

We prove (7.8). Using (7.6), taking ∇ operation r times on two sides of (7.9) gives

$$\nabla^r f(x) = \sum_{i=r}^{k} a_i\binom{x+i-r}{i-r},$$
$$r = 0, 1, \ldots, k.$$

It follows that

$$\nabla^r f(x)|_{x=-1} = \sum_{i=r}^{k} a_i\binom{i-r-1}{i-r} = a_r,$$
$$r = 0, 1, \ldots, k.$$

From (7.7), (7.8) holds. □

Applying Theorem 7.1.1 to $f(\tau) = \binom{\tau+c+k-1}{k-1}$ for $k \geqslant 1$, since

$$\nabla^r\binom{\tau+c+k-1}{k-1} = \binom{\tau+c+k-1-r}{k-1-r}$$

and $\binom{\tau+c+k-1}{k-1}$ is a degree $k-1$ polynomial of the variable τ, we have

$$\begin{aligned}\binom{\tau+c+k-1}{k-1} &= \sum_{r=0}^{k-1}\binom{-1+c+k-1-r}{k-1-r}\binom{\tau+r}{r}\\ &= \sum_{h=1}^{k}\binom{k-h+c-1}{k-h}\binom{\tau+h-1}{h-1}, \qquad (7.10)\\ &k = 1, 2, \ldots\end{aligned}$$

Applying Theorem 7.1.1 to $f(\tau) = \binom{u\tau+k-1}{k-1}$, we have

$$\begin{aligned}\binom{u\tau+k-1}{k-1} &= \sum_{r=0}^{k-1}\Big[\sum_{i=0}^{r}(-1)^i\binom{r}{i}\binom{-u(i+1)+k-1}{k-1}\Big]\binom{\tau+r}{r}\\ &= \sum_{a=1}^{k}\Big[\sum_{i=0}^{a-1}(-1)^i\binom{a-1}{i}\binom{-u(i+1)+k-1}{k-1}\Big]\binom{\tau+a-1}{a-1}, \qquad (7.11)\\ &k = 1, 2, \ldots\end{aligned}$$

Expanding two sides of the equation (7.11) into polynomials of the variable τ and comparing the coefficients of the highest term τ^{k-1}, we obtain that the coefficient of the highest term $\binom{\tau+k-1}{k-1}$ in the right side of (7.11) is u^{k-1}.

Applying Theorem 7.1.1 to $f(\tau) = \prod_{a=1}^{h}\binom{\tau+k_a-1}{k_a-1}$, we have

$$\begin{aligned}\prod_{a=1}^{h}\binom{\tau+k_a-1}{k_a-1} &= \sum_{r=0}^{k_1+\cdots+k_h-h}\Big[\sum_{c=0}^{r}(-1)^c\binom{r}{c}\prod_{a=1}^{h}\binom{-1-c+k_a-1}{k_a-1}\Big]\binom{\tau+r}{r}\\ &= \sum_{k=1}^{k_1+\cdots+k_h-h+1}\Big[\sum_{c=0}^{k-1}(-1)^c\binom{k-1}{c}\prod_{a=1}^{h}\binom{k_a-2-c}{k_a-1}\Big]\binom{\tau+k-1}{k-1}, \qquad (7.12)\\ &k_1, \ldots, k_h = 1, 2, \ldots\end{aligned}$$

Since $k_a - 1 > k_a - 2 - c \geqslant k_a - 1 - k \geqslant 0$ whenever $0 \leqslant c < k < k_a$, we have $\binom{k_a-2-c}{k_a-1} = 0$. Thus (7.12) can be written as

$$\begin{aligned}\prod_{a=1}^{h}\binom{\tau+k_a-1}{k_a-1} &= \sum_{k=\max(k_1,\ldots,k_h)}^{k_1+\cdots+k_h-h+1}\Big[\sum_{c=0}^{k-1}(-1)^c\binom{k-1}{c}\prod_{a=1}^{h}\binom{k_a-2-c}{k_a-1}\Big]\binom{\tau+k-1}{k-1},\\ &k_1, \ldots, k_h = 1, 2, \ldots \qquad (7.13)\end{aligned}$$

In the case of $h = 2$, for computing the value in the square brackets in (7.13), we may use the following formula.

$$\sum_{c=0}^{r}(-1)^c\binom{r}{c}\binom{-1-c+r_1}{r_1}\binom{-1-c+r_2}{r_2} = (-1)^{r_1+r_2+r}\binom{r_2}{r-r_1}\binom{r}{r_2}$$

$$
\begin{aligned}
&= (-1)^{r_1+r_2+r}\tbinom{r_1}{r-r_2}\tbinom{r}{r_1} \qquad (7.14)\\
&= \frac{(-1)^{r_1+r_2+r} r!}{(r-r_1)!(r-r_2)!(r_1+r_2-r)!}\ ,\\
&r, r_1, r_2 = 0, 1, \ldots,
\end{aligned}
$$

where $(-n)! = \infty$ if $n > 0$ by convention. We prove (7.14) by induction on r. *Basis* : $r = 0$. The two sides of the equation (7.14) take 0 whenever $r_1 \neq 0$ or $r_2 \neq 0$, take 1 otherwise. Thus (7.14) holds. *Induction step* : Suppose that (7.14) holds for $r-1$ ($\geqslant 0$), that is,

$$
\begin{aligned}
&\sum_{c=0}^{r-1}(-1)^c\tbinom{r-1}{c}\tbinom{-1-c+r_1}{r_1}\tbinom{-1-c+r_2}{r_2} = (-1)^{r_1+r_2+r-1}\tbinom{r_2}{r-1-r_1}\tbinom{r-1}{r_2}\\
&\quad = (-1)^{r_1+r_2+r-1}\tbinom{r_1}{r-1-r_2}\tbinom{r-1}{r_1}. \qquad (7.15)
\end{aligned}
$$

We prove that (7.14) holds for r. There are four cases to consider. In the case of $r_1, r_2 > 0$, using (7.2) and (7.15), we have

$$
\begin{aligned}
&\sum_{c=0}^{r}(-1)^c\tbinom{r}{c}\tbinom{-1-c+r_1}{r_1}\tbinom{-1-c+r_2}{r_2}\\
&\quad = \sum_{c=0}^{r}(-1)^c\left[\tbinom{r-1}{c} + \tbinom{r-1}{c-1}\right]\tbinom{r_1-1-c}{r_1}\tbinom{r_2-1-c}{r_2}\\
&\quad = \sum_{c=0}^{r}(-1)^c\tbinom{r-1}{c}\tbinom{r_1-1-c}{r_1}\tbinom{r_2-1-c}{r_2} - \sum_{c=-1}^{r-1}(-1)^c\tbinom{r-1}{c}\tbinom{r_1-2-c}{r_1}\tbinom{r_2-2-c}{r_2}\\
&\quad = \sum_{c=0}^{r-1}(-1)^c\tbinom{r-1}{c}\tbinom{r_1-1-c}{r_1}\tbinom{r_2-1-c}{r_2}\\
&\qquad - \sum_{c=0}^{r-1}(-1)^c\tbinom{r-1}{c}\left[\tbinom{r_1-1-c}{r_1} - \tbinom{r_1-2-c}{r_1-1}\right]\left[\tbinom{r_2-1-c}{r_2} - \tbinom{r_2-2-c}{r_2-1}\right]\\
&\quad = \sum_{c=0}^{r-1}(-1)^c\tbinom{r-1}{c}\tbinom{r_1-2-c}{r_1-1}\tbinom{r_2-1-c}{r_2} + \sum_{c=0}^{r-1}(-1)^c\tbinom{r-1}{c}\tbinom{r_1-1-c}{r_1}\tbinom{r_2-2-c}{r_2-1}\\
&\qquad - \sum_{c=0}^{r-1}(-1)^c\tbinom{r-1}{c}\tbinom{r_1-2-c}{r_1-1}\tbinom{r_2-2-c}{r_2-1}\\
&\quad = (-1)^{r_1+r_2+r}\left[\tbinom{r_2}{r-r_1}\tbinom{r-1}{r_2} + \tbinom{r_2-1}{r-1-r_1}\tbinom{r-1}{r_2-1} + \tbinom{r_2-1}{r-r_1}\tbinom{r-1}{r_2-1}\right]\\
&\quad = (-1)^{r_1+r_2+r}\tbinom{r_2}{r-r_1)}\tbinom{r}{r_2}.
\end{aligned}
$$

In the case of $r_1 = r_2 = 0$, using (7.2) and (7.15), we have

$$
\sum_{c=0}^{r}(-1)^c\tbinom{r}{c}\tbinom{-1-c+r_1}{r_1}\tbinom{-1-c+r_2}{r_2} = \sum_{c=0}^{r}(-1)^c\left[\tbinom{r-1}{c} + \tbinom{r-1}{c-1}\right]
$$

$$
\begin{aligned}
&= \sum_{c=0}^{r}(-1)^c\tbinom{r-1}{c} - \sum_{c=-1}^{r-1}(-1)^c\tbinom{r-1}{c} \\
&= \sum_{c=0}^{r-1}(-1)^c\tbinom{r-1}{c} - \sum_{c=0}^{r-1}(-1)^c\tbinom{r-1}{c} \\
&= 0 = (-1)^{r_1+r_2+r}\tbinom{r_2}{r-r_1}\tbinom{r}{r_2}.
\end{aligned}
$$

In the case of $r_1 > 0$ and $r_2 = 0$, using (7.2) and (7.15), we have

$$
\begin{aligned}
\sum_{c=0}^{r}(-1)^c\tbinom{r}{c}\tbinom{-1-c+r_1}{r_1}\tbinom{-1-c+r_2}{r_2} &= \sum_{c=0}^{r}(-1)^c\tbinom{r}{c}\tbinom{-1-c+r_1}{r_1} \\
&= \sum_{c=0}^{r}(-1)^c\left[\tbinom{r-1}{c} + \tbinom{r-1}{c-1}\right]\tbinom{-1-c+r_1}{r_1} \\
&= \sum_{c=0}^{r-1}(-1)^c\tbinom{r-1}{c}\tbinom{-1-c+r_1}{r_1} + \sum_{c=1}^{r}(-1)^c\tbinom{r-1}{c-1}\tbinom{-1-c+r_1}{r_1} \\
&= \sum_{c=0}^{r-1}(-1)^c\tbinom{r-1}{c}\tbinom{-1-c+r_1}{r_1} - \sum_{c=0}^{r-1}(-1)^c\tbinom{r-1}{c}\left[\tbinom{-1-c+r_1}{r_1} - \tbinom{-2-c+r_1}{r_1-1}\right] \\
&= \sum_{c=0}^{r-1}(-1)^c\tbinom{r-1}{c}\tbinom{-1-c+r_1-1}{r_1-1} \\
&= \sum_{c=0}^{r-1}(-1)^c\tbinom{r-1}{c}\tbinom{-1-c+r_1-1}{r_1-1}\tbinom{-1-c+r_2}{r_2} \\
&= (-1)^{r_1+r_2+r-2}\tbinom{r_2}{r-1-(r_1-1)}\tbinom{r-1}{r_2} \\
&= (-1)^{r_1+r_2+r}\tbinom{r_2}{r-r_1}\tbinom{r}{r_2}.
\end{aligned}
$$

In the case of $r_2 > 0$ and $r_1 = 0$, from symmetry, the above case yields

$$
\sum_{c=0}^{r}(-1)^c\tbinom{r}{c}\tbinom{-1-c+r_1}{r_1}\tbinom{-1-c+r_2}{r_2} = (-1)^{r_1+r_2+r}\tbinom{r_1}{r-r_2}\tbinom{r}{r_1}.
$$

We conclude that (7.14) holds for $r, r_1, r_2 \geqslant 0$.

Taking $r_2 = 0$, (7.14) is reduced to

$$
\begin{aligned}
\sum_{c=0}^{r}(-1)^c\tbinom{r}{c}\tbinom{-1-c+r_1}{r_1} &= (-1)^{r_1+r}\tbinom{0}{r-r_1}\tbinom{r}{0} \\
&= \begin{cases} 1, & \text{if } r = r_1, \\ 0, & \text{if } r \neq r_1, \end{cases} \\
& r, r_1 = 0, 1, \ldots
\end{aligned}
$$

Taking $r_1 = r_2 = 0$, (7.14) is reduced to

$$\sum_{c=0}^{r}(-1)^c\binom{r}{c} = (-1)^r\binom{0}{r}\binom{r}{0} = \begin{cases} 1, & \text{if } r = 0, \\ 0, & \text{if } r \neq 0. \end{cases}$$

From (7.12),(7.13) and (7.14), we have

$$\binom{\tau+k_1-1}{k_1-1}\binom{\tau+k_2-1}{k_2-1} = \sum_{k=1}^{k_1+k_2-1}(-1)^{k_1+k_2+k-1}\binom{k_2-1}{k-k_1}\binom{k-1}{k_2-1}\binom{\tau+k-1}{k-1}$$

$$= \sum_{k=\max(k_1,k_2)}^{k_1+k_2-1}(-1)^{k_1+k_2+k-1}\binom{k_2-1}{k-k_1}\binom{k-1}{k_2-1}\binom{\tau+k-1}{k-1},$$

$$k_1, k_2 = 1, 2, \ldots$$

Similar to Theorem 7.1.1, we can prove the following theorem.

Theorem 7.1.2. *Any integer-valued polynomial $f(x)$ of degree k can be uniquely expressed in the form $\sum_{r=0}^{k} a_r\binom{x}{r}$, where $a_0, a_1, \ldots, a_k$ are integers and $a_k \neq 0$, and*

$$a_r = \nabla^r f(x)|_{x=r} = \sum_{i=0}^{r}(-1)^{r-i}\binom{r}{i}f(i),$$
$$r = 0, 1, \ldots, k.$$

Applying Theorem 7.1.2 to $f(w) = \binom{a+we}{h}$, e being a positive integer, we have

$$\binom{a+we}{h} = \sum_{k=0}^{h}\Big[\sum_{i=0}^{k}(-1)^{k-i}\binom{k}{i}\binom{a+ie}{h}\Big]\binom{w}{k}, \tag{7.16}$$
$$h = 0, 1, \ldots$$

Expanding the two sides of the equation (7.16) into polynomials of w and comparing the coefficients of the highest term w^h, we obtain that the coefficients of the highest term $\binom{w}{h}$ in the right side of (7.16) is e^h.

Let p be a prime number. Since $\gcd(r, p) = 1$ for any r, $0 < r < p$, we have

$$\binom{p}{r} = 0 \pmod{p}, \quad 0 < r < p\,. \tag{7.17}$$

It is easy to see that

$$(x+y)^n = \sum_{r=0}^{n}\binom{n}{r}x^r y^{n-r}. \tag{7.18}$$

From (7.17) and (7.18), we have

$$(x+y)^p = x^p + y^p \pmod{p}, \quad \text{if } p \text{ is prime.}$$

Theorem 7.1.3. (*Lucas*) *Let p be a prime number. If*

$$n = \sum_{i=0}^{k} n_i p^i,\ 0 \leqslant n_i < p\,, \quad r = \sum_{i=0}^{k} r_i p^i,\ 0 \leqslant r_i < p\,, \tag{7.19}$$

then

$$\binom{n}{r} = \prod_{i=0}^{k} \binom{n_i}{r_i} \pmod{p}. \tag{7.20}$$

Proof. Since $(1+x)^p = 1 + x^p \pmod{p}$, we have $(1+x)^{mp+a} = (1+x^p)^m (1+x)^a \pmod{p}$. Using (7.18), this yields

$$\sum_{i=0}^{mp+a} \binom{mp+a}{i} x^i = \Big(\sum_{i=0}^{m} \binom{m}{i} x^{pi}\Big)\Big(\sum_{i=0}^{a} \binom{a}{i} x^i\Big) \pmod{p}.$$

Expanding two sides of the above equation and comparing the coefficients of the term x^{hp+b}, we obtain

$$\binom{mp+a}{hp+b} = \begin{cases} 0 \pmod{p}, & \text{if } 0 \leqslant a < b < p\,, \\ \binom{m}{h}\binom{a}{b} \pmod{p}, & \text{if } 0 \leqslant b \leqslant a < p\,. \end{cases}$$

Therefore,

$$\binom{mp+a}{hp+b} = \binom{m}{h}\binom{a}{b} \pmod{p}, \quad \text{if } 0 \leqslant a, b < p\,. \tag{7.21}$$

We prove by induction on k that (7.20) holds for any k, n and r satisfying (7.19). *Basis* : $k = 0$. That is, $n = n_0$ and $r = r_0$. Thus $\binom{n}{r} = \binom{n_0}{r_0} \pmod{p}$. *Induction step* : Suppose that (7.20) holds for any n and r satisfying (7.19). Let

$$n' = \sum_{i=0}^{k+1} n_i p^i,\ 0 \leqslant n_i < p\,, \quad r' = \sum_{i=0}^{k+1} r_i p^i,\ 0 \leqslant r_i < p\,.$$

Then $n' = np + n_0$ and $r' = rp + r_0$, where

$$n = \sum_{i=0}^{k} n_{i+1} p^i, \quad r = \sum_{i=0}^{k} r_{i+1} p^i.$$

From (7.21) and the induction hypothesis, we have

$$\begin{aligned}\binom{n'}{r'} &= \binom{np+n_0}{rp+r_0} = \binom{n}{r}\binom{n_0}{r_0} \pmod p \\ &= \Big(\sum_{i=0}^{k}\binom{n_{i+1}}{r_{i+1}}\Big)\binom{n_0}{r_0} \pmod p \\ &= \sum_{i=0}^{k+1}\binom{n_i}{r_i} \pmod p.\end{aligned}$$

We conclude that (7.20) holds for any k, n and r satisfying (7.19). □

7.2 Root Representation

Consider (generalized) polynomials over $GF(q)$. Let $\psi(z) = \sum_{i=k}^{h} a_i z^i$ be a (generalized) polynomials over $GF(q)$, where h, k are integers, $h \geqslant k$, and $a_i \in GF(q)$, $i = k, k+1, \ldots, h$. $\max i\,[\,a_i \neq 0\,]$ is referred to as the *high degree* of ψ, and $\min i\,[\,a_i \neq 0\,]$ is referred to as the *low degree* of ψ. In the case of the zero polynomial, its high degree is ∞ and its low degree is $-\infty$. Clearly, if the high degree and the low degree of $\psi_i(z)$ are h_i and k_i, respectively, $i = 1, 2$, then the high degree and the low degree of the product of $\psi_1(z)$ and $\psi_2(z)$ are $h_1 + h_2$ and $k_1 + k_2$, respectively. For any polynomial ψ and any nonzero polynomial φ, it is easy to show that there exist uniquely polynomials $q(z)$ and $r(z)$ such that

$$\psi(z) = q(z)\varphi(z) + r(z),$$

$r(z) = 0$ or the low degree of $r(z) \geqslant$ the low degree of $\varphi(z)$, and $q(z) = 0$ or the high degree of $q(z) < 0$. Denote the unique $q(z)$ and $r(z)$ by $\mathrm{Quo}'(\psi(z), \varphi(z))$ and $\mathrm{Res}'(\psi(z), \varphi(z))$, respectively. It is easy to verify that for any nonzero polynomial $\chi(z)$, we have $\mathrm{Quo}'(\chi(z)\psi(z), \chi(z)\varphi(z)) = \mathrm{Quo}'(\psi(z), \varphi(z))$ and $\mathrm{Res}'(\chi(z)\psi(z), \chi(z)\varphi(z)) = \chi(z)\ \mathrm{Res}'(\psi(z), \varphi(z))$.

Let $M = \langle Y, S, \delta, \lambda\rangle$ be a linear autonomous finite automaton over $GF(q)$ with structure parameters m, n and structure matrices A, C. That is, Y and S are column vector spaces of dimensions m and n over $GF(q)$, respectively, $\delta(s) = As$, $\lambda(s) = Cs$, and A and C are $n \times n$ and $m \times n$ matrices over $GF(q)$, respectively. A and C are referred to as the *state transition matrix* and the *output matrix* of M, respectively.

For any $s \in S$, the infinite output sequence generated by s means the sequence $y_0 y_1 \ldots\ y_i \ldots$, where $y_i = \lambda(\delta^i(s))$ for $i \geqslant 0$. We use $\Phi_M(s)$ to denote $[y_0,\ y_1,\ \ldots,\ y_i,\ \ldots]$, and use $\Phi_M(s, z)$ to denote its z-transformation $\sum_{i=0}^{\infty} y_i z^i$. For any i, $1 \leqslant i \leqslant m$, we use $\Phi_M^{(i)}(s)$ and $\Phi_M^{(i)}(s, z)$ to denote the i-th row of $\Phi_M(s)$ and the i-th component of $\Phi_M(s, z)$, respectively. Clearly, $\Phi_M^{(i)}(s, z)$ is the z-transformation of $\Phi_M^{(i)}(s)$.

Similarly, for any $s \in S$, we use $\Psi_M(s)$ to denote the infinite state sequence generated by s

$$\Psi_M(s) = [s, \delta(s), \ldots, \delta^i(s), \ldots],$$

and use $\Psi_M(s,z)$ to denote its z-transformation $\sum_{i=0}^{\infty} \delta^i(s) z^i$. For any i, $1 \leqslant i \leqslant n$, we use $\Psi_M^{(i)}(s)$ and $\Psi_M^{(i)}(s,z)$ to denote the i-th row of $\Psi_M(s)$ and the i-th component of $\Psi_M(s,z)$, respectively. Clearly, $\Psi_M^{(i)}(s,z)$ is the z-transformation of $\Psi_M^{(i)}(s)$. Notice that the elements of $\Phi_M(s,z)$ and $\Psi_M(s,z)$ are formal power series of z and can be expressed as rational fractions of z.

Define

$$\begin{array}{ll} \Phi_M = \{\Phi_M(s),\ s \in S\}, & \Phi_M(z) = \{\Phi_M(s,z),\ s \in S\}, \\ \Psi_M = \{\Psi_M(s),\ s \in S\}, & \Psi_M(z) = \{\Psi_M(s,z),\ s \in S\}, \\ \Psi_M^{(1)} = \{\Psi_M^{(1)}(s),\ s \in S\}, & \Psi_M^{(1)}(z) = \{\Psi_M^{(1)}(s,z),\ s \in S\}. \end{array}$$

Let $f(z) = z^n + a_{n-1}z^{n-1} + \cdots + a_1 z + a_0$ be a polynomial over $GF(q)$. Recall that $P_{f(z)}$ is used to denote the $n \times n$ matrix

$$\begin{bmatrix} 0 & 1 & \cdots & 0 & 0 \\ 0 & 0 & \cdots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \cdots & 0 & 1 \\ -a_0 & -a_1 & \cdots & -a_{n-2} & -a_{n-1} \end{bmatrix}.$$

If $A = P_{f(z)}$ for some $f(z)$, M is called a *shift register*. If M is a shift register and $A = P_{f(z)}$, then $f(z) = |zE - A|$, where E stands for the $n \times n$ identity matrix. $f(z)$ is referred to as the *characteristic polynomial* of M.

Theorem 7.2.1. *Let M be a linear shift register over $GF(q)$, and $f(z)$ the characteristic polynomial of M. Let $g(z)$ be the reverse polynomial of $f(z)$, i.e., $g(z) = z^n f(z^{-1})$. Then for any state s of M, we have*

$$\Psi_M^{(1)}(s,z) = \sum_{k=0}^{n-1} h_k z^k / g(z), \tag{7.22}$$

where

$$\begin{bmatrix} h_0 \\ h_1 \\ \vdots \\ h_{n-1} \end{bmatrix} = Q_{f(z)} s, \quad Q_{f(z)} = \begin{bmatrix} 1 & 0 & \cdots & 0 & 0 \\ a_{n-1} & 1 & \cdots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ a_2 & a_3 & \cdots & 1 & 0 \\ a_1 & a_2 & \cdots & a_{n-1} & 1 \end{bmatrix}. \tag{7.23}$$

Proof. Let M' be a linear shift register with structure parameters n, n and structure matrices A, E. Then $\Psi_M(s) = \Phi_{M'}(s)$. Since the free response matrix of M' is $E(E-zA)^{-1} = (E-zA)^{-1}$, we have $\Psi_M(s,z) = (E-zA)^{-1}s$. Since $|zE - A| = f(z)$, we have $g(z) = |E - zA|$. Letting $s = [s_0, \ldots, s_{n-1}]^{\mathrm{T}}$ and $a_n = 1$, using (1.12),(1.13) and (1.14) in Sect. 1.3 of Chap. 1, we have

$$\begin{aligned}
\Psi_M^{(1)}(s,z) &= \sum_{j=0}^{n-1} \big(1 + \sum_{i=1}^{n-1-j} a_{n-i} z^i\big) z^j s_j \,/\, |E - zA| \\
&= \sum_{j=0}^{n-1} \big(\sum_{i=0}^{n-1-j} a_{n-i} z^i\big) z^j s_j / g(z) \\
&= \sum_{j=0}^{n-1} \big(\sum_{k=j}^{n-1} a_{n-k+j} s_j\big) z^k / g(z) \\
&= \sum_{k=0}^{n-1} \big(\sum_{j=0}^{k} a_{n-k+j} s_j\big) z^k / g(z) \\
&= \sum_{k=0}^{n-1} h_k z^k / g(z).
\end{aligned}$$

Thus (7.22) holds. □

Corollary 7.2.1. *Let M be a linear shift register over $GF(q)$, and $g(z)$ the reverse polynomial of the characteristic polynomial of M. Then $1/g(z)$, $z/g(z)$, …, $z^{n-1}/g(z)$ form a basis of $\Psi_M^{(1)}(z)$.*

Proof. Let $h = [h_0, \ldots, h_{n-1}]^{\mathrm{T}}$. From Theorem 7.2.1, for any $\Psi_M^{(1)}(s,z)$ in $\Psi_M^{(1)}(z)$, we have $\Psi_M^{(1)}(s,z) = \sum_{k=0}^{n-1} h_k z^k / g(z)$, where $h = Q_{f(z)}s$. Conversely, for any $h_0, \ldots, h_{n-1}$ in $GF(q)$, since $Q_{f(z)}$ is nonsingular, there is s in S such that $h = Q_{f(z)}s$. From Theorem 7.2.1, we have $\Psi_M^{(1)}(s,z) = \sum_{k=0}^{n-1} h_k z^k / g(z)$, Thus

$$\sum_{k=0}^{n-1} h_k z^k / g(z) \in \Psi_M^{(1)}(z).$$

Clearly, if $h_0, \ldots, h_{n-1}$ are not identically equal to zero, then $\sum_{k=0}^{n-1} h_k z^k / g(z) \neq 0$. We conclude that $1/g(z)$, $z/g(z)$, …, $z^{n-1}/g(z)$ form a basis of $\Psi_M^{(1)}(z)$. □

The basis $[1/g(z), z/g(z), \ldots, z^{n-1}/g(z)]$ is referred to as the *polynomial basis* of $\Psi_M^{(1)}(z)$ and $[h_0, \ldots, h_{n-1}]^{\mathrm{T}}$ is referred to as the *polynomial coordinate* of $\sum_{k=0}^{n-1} h_k z^k / g(z)$.

Let

$$c_i(z) = \sum_{k=1}^{n} c_{ik} z^{1-k},\ i = 1, \ldots, m,$$

where $C = [c_{ij}]_{m \times n}$ is the output matrix of M. $c_i(z), i = 1, \ldots, m$ are referred to as the *output polynomials* of M, and

$$f'(z) = f(z)/\gcd(f(z), c_1(z^{-1}), \ldots, c_m(z^{-1}))$$

is called the *second characteristic polynomial* of M.

For any infinite sequence $\Omega = [b_0, b_1, \ldots, b_i, \ldots]$, we use $D(\Omega)$ to denote the sequence $[b_1, b_2, \ldots, b_i, \ldots]$, the (one digit) translation of Ω. Similarly, we use $D(\sum_{i=0}^{\infty} b_i z^i)$ to denote $\sum_{i=0}^{\infty} b_{i+1} z^i$. We define $D^0(\Omega) = \Omega$, $D^{c+1}(\Omega) = D(D^c(\Omega))$, $D^0(\Omega(z)) = \Omega(z)$, $D^{c+1}(\Omega(z)) = D(D^c(\Omega(z)))$.

Theorem 7.2.2. *Let M be a linear shift register over $GF(q)$ with structure parameters m, n. Let $g(z)$ be the reverse polynomial of the characteristic polynomial of M, and $c_k(z), k = 1, \ldots, m$ the output polynomials of M. Let n' be the degree of the second characteristic polynomial of M. Then the dimension of $\Phi_M(z)$ is n', and $\Omega(z) \in \Phi_M(z)$ if and only if there exists a polynomial $h(z)$ of degree $< n$ over $GF(q)$ such that $z^{n-n'} | h(z)$ and*

$$\Omega(z) = \begin{bmatrix} \mathrm{Res}'(c_1(z)h(z), g(z))/g(z) \\ \vdots \\ \mathrm{Res}'(c_m(z)h(z), g(z))/g(z) \end{bmatrix}. \tag{7.24}$$

Proof. We first prove the following result: for any $s \in S$ and any polynomial $h(z) = \sum_{i=0}^{n-1} h_i z^i$, if $[h_0, \ldots, h_{n-1}]^{\mathrm{T}} = Q_{f(z)} s$, then

$$\Phi_M(s, z) = \begin{bmatrix} \mathrm{Res}'(c_1(z)h(z), g(z))/g(z) \\ \vdots \\ \mathrm{Res}'(c_m(z)h(z), g(z))/g(z) \end{bmatrix}, \tag{7.25}$$

where $f(z)$ is the characteristic polynomial of M. In fact, from Theorem 7.2.1, $\Psi_M^{(1)}(s, z) = h(z)/g(z)$. It follows that $\Psi_M(s, z) = [D^0(h(z)/g(z)), D^1(h(z)/g(z)), \ldots, D^{n-1}(h(z)/g(z))]^{\mathrm{T}}$. Thus

$$\begin{aligned} \Phi_M(s, z) &= C[D^0(h(z)/g(z)), D^1(h(z)/g(z)), \ldots, D^{n-1}(h(z)/g(z))]^{\mathrm{T}} \\ &= [c_1(D^{-1})(h(z)/g(z)), \ldots, c_{n-1}(D^{-1})(h(z)/g(z))]^{\mathrm{T}}. \end{aligned}$$

Clearly, $D^k(h(z)/g(z))$ is the nonnegative power part of $z^{-k}(h(z)/g(z))$. Thus for any i, $1 \leqslant i \leqslant m$, $\sum_{k=1}^{n} c_{ik} D^{k-1}(h(z)/g(z))$ is the nonnegative power part of $\sum_{k=1}^{n} c_{ik} z^{1-k}\ (h(z)/g(z)) = c_i(z)h(z)/g(z)$. Therefore, $\Phi_M(s, z)$ is

the nonnegative power part of $[c_1(z)h(z)/g(z), \ldots, c_m(z)h(z)/g(z)]^{\mathrm{T}}$, that is, (7.25) holds.

Suppose that (7.24) holds for a polynomial $h(z)$ of degree $< n$ over $GF(q)$. Let $s = Q_{f(z)}^{-1}[h_0, \ldots, h_{n-1}]^{\mathrm{T}}$, where $h(z) = \sum_{i=0}^{n-1} h_i z^i$. From (7.24) and (7.25), we have $\Omega(z) = \Phi_M(s, z)$; therefore, $\Omega(z) \in \Phi_M(z)$.

We prove a proposition: If $h(z) = \sum_{i=0}^{n-1} h_i z^i$ and $\bar{h}(z) = \sum_{i=0}^{n-1} \bar{h}_i z^i$ are two polynomials over $GF(q)$, then

$$\begin{bmatrix} \mathrm{Res}'(c_1(z)h(z), g(z))/g(z) \\ \vdots \\ \mathrm{Res}'(c_m(z)h(z), g(z))/g(z) \end{bmatrix} = \begin{bmatrix} \mathrm{Res}'(c_1(z)\bar{h}(z), g(z))/g(z) \\ \vdots \\ \mathrm{Res}'(c_m(z)\bar{h}(z), g(z))/g(z) \end{bmatrix} \tag{7.26}$$

holds if and only if $f'(z)|(\bar{h}'(z) - h'(z))$, where $h'(z) = z^{n-1}h(1/z)$ and $\bar{h}'(z) = z^{n-1}\bar{h}(1/z)$. In fact, since for any i, $1 \leqslant i \leqslant m$, $c_i'(z) = c_i(1/z)$ is a common polynomial, there exist uniquely common polynomials $q_i'(z)$ and $r_i'(z)$ such that $c_i'(z)h'(z) = q_i'(z)f(z) + r_i'(z)$, $r_i'(z) = 0$ or the degree of $r_i'(z) < n$. It follows that $c_i(z)h(z) = z^{n-1}c_i'(1/z)h'(1/z) = (z^{-1}q_i'(1/z))(z^n f(1/z)) + z^{n-1}r_i'(1/z) = (z^{-1}q_i'(1/z))g(z) + z^{n-1}r_i'(1/z)$. Thus

$$z^{-1}q_i'(1/z) = \mathrm{Quo}'(c_i(z)h(z), g(z)),\ z^{n-1}r_i'(1/z) = \mathrm{Res}'(c_i(z)h(z), g(z)).$$

Similarly, there exist uniquely common polynomials $\bar{q}_i'(z)$ and $\bar{r}_i'(z)$ such that $c_i'(z)\bar{h}'(z) = \bar{q}_i'(z)f(z) + \bar{r}_i'(z)$, $\bar{r}_i'(z) = 0$ or the degree of $\bar{r}_i'(z) < n$. It follows that $z^{-1}\bar{q}_i'(1/z) = \mathrm{Quo}'(c_i(z)\bar{h}(z), g(z))$, $z^{n-1}\bar{r}_i'(1/z) = \mathrm{Res}'(c_i(z)\bar{h}(z), g(z))$. Thus for any i, $1 \leqslant i \leqslant m$, $\mathrm{Res}'(c_i(z)h(z), g(z)) = \mathrm{Res}'(c_i(z)\bar{h}(z), g(z))$ if and only if $z^{n-1}r_i'(1/z) = z^{n-1}\bar{r}_i'(1/z)$, if and only if $f(z)|c_i'(z)(\bar{h}'(z) - h'(z))$. Thus (7.26) holds if and only if $f(z)|c_i'(z)(\bar{h}'(z) - h'(z))$, $i = 1, \ldots, m$. Let $d'(z) = \gcd(c_1'(z), \ldots, c_m'(z))$. It is easy to see that

$$\gcd(c_1'(z)(\bar{h}'(z) - h'(z)), \ldots, c_m'(z)(\bar{h}'(z) - h'(z))) = d'(z)(\bar{h}'(z) - h'(z)).$$

It follows that (7.26) holds if and only if $f(z)|d'(z)(\bar{h}'(z) - h'(z))$, if and only if $f'(z)|(\bar{h}'(z) - h'(z))$.

Suppose that $\Omega(z) \in \Phi_M(z)$. We prove that there exists a polynomial $h(z)$ of degree $< n$ over $GF(q)$ such that $z^{n-n'}|h(z)$ and (7.24) hold. Let $\Omega(z) = \Phi(\bar{s}, z)$ for some $\bar{s} \in S$. Denote $[\bar{h}_0, \ldots, \bar{h}_{n-1}]^{\mathrm{T}} = Q_{f(z)}\bar{s}$ and $\bar{h}(z) = \sum_{i=0}^{n-1} \bar{h}_i z^i$. Let $\bar{r}'(z)$ be a common polynomial of degree $< n'$, and $\bar{r}'(z) = \bar{h}'(z) \pmod{f'(z)}$, where $\bar{h}'(z) = z^{n-1}\bar{h}(1/z)$. Take $h(z) = z^{n-1}\bar{r}'(1/z)$. Then $z^{n-n'}|h(z)$ and $h'(z) = z^{n-1}h(1/z) = \bar{r}'(z)$. It follows that $f'(z)|(\bar{h}'(z) - h'(z))$. From the proposition proven in the preceding paragraph, (7.26) holds. From $\Omega(z) = \Phi(\bar{s}, z)$ and using (7.25) (in version of $\bar{s}$ and $\bar{h}$), this yields

$$\Omega(z) = \begin{bmatrix} \mathrm{Res}'(c_1(z)\bar{h}(z), g(z))/g(z) \\ \vdots \\ \mathrm{Res}'(c_m(z)\bar{h}(z), g(z))/g(z) \end{bmatrix} = \begin{bmatrix} \mathrm{Res}'(c_1(z)h(z), g(z))/g(z) \\ \vdots \\ \mathrm{Res}'(c_m(z)h(z), g(z))/g(z) \end{bmatrix}.$$

In the case where $\bar{h}(z)$ and $h(z)$ have divisor $z^{n-n'}$, degrees of $\bar{h}'(z)$ and $h'(z)$ are $< n'$. Thus $f'(z) \not| (\bar{h}'(z) - h'(z))$. Therefore, (7.26) does not hold. Since the number of polynomials of degree $< n$ over $GF(q)$ which have divisor $z^{n-n'}$ is $q^{n'}$, the dimension of $\Phi_M(z)$ is n'. □

Corollary 7.2.2. *Let n' be the dimension of $\Phi_M(z)$, then*

$$\begin{bmatrix} \mathrm{Res}'(c_1(z)z^{n-n'+k}, g(z))/g(z) \\ \vdots \\ \mathrm{Res}'(c_m(z)z^{n-n'+k}, g(z))/g(z) \end{bmatrix}, \quad k = 0, 1, \ldots, n'-1$$

form a basis of $\Phi_M(z)$.

This basis is called the *polynomial basis* of $\Phi_M(z)$. If (7.24) holds and $h(z) = \sum_{i=0}^{n'-1} h_i z^{n-n'+i}$, $[h_0, \ldots, h_{n'-1}]^{\mathrm{T}}$ is called the *polynomial coordinate* of $\Omega(z)$.

Corollary 7.2.3. *For any $s \in S$, if*

$$\sum_{i=0}^{n'-1} h'_i z^{n'-1-i} = \sum_{i=0}^{n-1} h_i z^{n-1-i} \pmod{f'(z)}$$

and $[h_0, \ldots, h_{n-1}]^{\mathrm{T}} = Q_{f(z)}s$, then $[h'_0, \ldots, h'_{n'-1}]^{\mathrm{T}}$ is the polynomial coordinate of $\Phi_M(s, z)$, where $f(z)$ and $f'(z)$ are the characteristic polynomial and the second characteristic polynomial of M, respectively, and n' is the degree of $f'(z)$.

Assume that $GF(q^*)$ is a splitting field of the second characteristic polynomial $f'(z)$ of M. Let M^* be the *natural extension* of M over $GF(q^*)$, i.e., the state transition matrices and the output matrices of M and M^* are the same, respectively.

Theorem 7.2.3. *Assume that the second characteristic polynomial $f'(z)$ of the linear shift register M has the factorization*

$$f'(z) = z^{l_0} \prod_{i=1}^{r} \prod_{j=1}^{n_i} (z - \varepsilon_i^{q^{j-1}})^{l_i}, \tag{7.27}$$

where $\varepsilon_1, \ldots, \varepsilon_r$ are nonzero elements in $GF(q^*)$ of which minimal polynomials over $GF(q)$ are coprime and have degrees $n_1, \ldots, n_r$, respectively, and

$l_0 \geqslant 0$, $l_1 > 0$, …, $l_r > 0$. *Then there exist uniquely column vectors* R_{0k}, $k = 1, \ldots, l_0$, R_{ijk}, $i = 1, \ldots, r$, $j = 1, \ldots, n_i$, $k = 1, \ldots, l_i$ *of dimension* m *over* $GF(q^*)$ *such that*

$$\begin{bmatrix} c_1(z)z^{n-1}/g(z) \\ \vdots \\ c_m(z)z^{n-1}/g(z) \end{bmatrix} = \sum_{k=1}^{l_0} R_{0k} z^{k-1} + \sum_{i=1}^{r} \sum_{j=1}^{n_i} \sum_{k=1}^{l_i} R_{ijk}/(1 - \varepsilon_i^{q^{j-1}} z)^k, \quad (7.28)$$

where $g(z) = z^n f(1/z)$, $f(z)$ *is the characteristic polynomial of* M. *Moreover, if* (7.28) *holds, then* $R_0(k)\Gamma_0(z)$, $k = 1, \ldots, l_0$, $R_{ij}(k)\Gamma_{ij}(z)$, $i = 1, \ldots, r$, $j = 1, \ldots, n_i$, $k = 1, \ldots, l_i$ *form a basis of* $\Phi_{M^*}(z)$, *where* $R_0(k)$ *and* $R_{ij}(k)$ *are matrices of dimension* $m \times l_0$ *and* $m \times l_i$, *respectively, and*

$$\begin{aligned} &R_0(k) = [R_{0(l_0+1-k)} \ \ldots \ R_{0l_0} \ 0 \ \ldots \ 0], \quad k = 1, \ldots, l_0, \\ &R_{ij}(k) = [R_{ij(l_i+1-k)} \ \ldots \ R_{ijl_i} \ 0 \ \ldots \ 0], \\ &\quad i = 1, \ldots, r, \ j = 1, \ldots, n_i, \ k = 1, \ldots, l_i, \\ &\Gamma_0(z) = \begin{bmatrix} 1 \\ z \\ \vdots \\ z^{l_0-1} \end{bmatrix}, \\ &\Gamma_{ij}(z) = \begin{bmatrix} 1/(1 - \varepsilon_i^{q^{j-1}} z) \\ 1/(1 - \varepsilon_i^{q^{j-1}} z)^2 \\ \vdots \\ 1/(1 - \varepsilon_i^{q^{j-1}} z)^{l_i} \end{bmatrix}, \quad i = 1, \ldots, r, \ j = 1, \ldots, n_i. \end{aligned} \quad (7.29)$$

Proof. Let $g'(z) = z^{n'} f'(1/z)$, where n' is the degree of $f'(z)$. From (7.27), we have $n' = l_0 + \sum_{i=1}^{r} n_i l_i$ and

$$g'(z) = \prod_{i=1}^{r} \prod_{j=1}^{n_i} (1 - \varepsilon_i^{q^{j-1}} z)^{l_i}. \quad (7.30)$$

Let $c_i'(1/z) = c_i(1/z)/d(z)$, $i = 1, \ldots, m$, where $d(z) = \gcd(f(z), c_1(1/z), \ldots, c_m(1/z))$. Since $g(z) = z^n f(1/z) = (z^{n'} f'(1/z))\ (z^{n-n'} d(1/z)) = g'(z)\ (z^{n-n'} d(1/z))$ and $c_i(z)z^{n-1} = c_i'(z)z^{n'-1}\ (z^{n-n'} d(1/z))$, $i = 1, \ldots, m$, we have $c_i(z)z^{n-1}/g(z) = c_i'(z)z^{n'-1}/g'(z)$, $i = 1, \ldots, m$. From (7.30), there exists uniquely column vectors R_{0k}, $k = 1, \ldots, l_0$, R_{ijk}, $i = 1, \ldots, r$, $j = 1, \ldots, n_i$, $k = 1, \ldots, l_i$ of dimension m over $GF(q^*)$ such that

$$\begin{bmatrix} c_1(z)z^{n-1}/g(z) \\ \vdots \\ c_m(z)z^{n-1}/g(z) \end{bmatrix} = \begin{bmatrix} c_1'(z)z^{n'-1}/g'(z) \\ \vdots \\ c_m'(z)z^{n'-1}/g'(z) \end{bmatrix} \quad (7.31)$$

$$
\begin{aligned}
&= \sum_{k=1}^{l_0} R_{0k} z^{k-1} + \sum_{i=1}^{r} \sum_{j=1}^{n_i} \sum_{k=1}^{l_i} R_{ijk} / (1 - \varepsilon_i^{q^{j-1}} z)^k \\
&= R\Gamma(z),
\end{aligned}
$$

where

$$
\begin{aligned}
R &= [R_0 \; R_{11} \; \ldots \; R_{1n_1} \; \ldots\ldots \; R_{r1} \; \ldots \; R_{rn_r}], \\
R_0 &= [R_{01} \; \ldots \; R_{0l_0}], \\
R_{ij} &= [R_{ij1} \ldots \; R_{ijl_i}], \; i = 1, \ldots, r, \; j = 1, \ldots, n_i,
\end{aligned}
$$

$$
\Gamma(z) = \begin{bmatrix}
\Gamma_0(z) \\
\Gamma_{11}(z) \\
\vdots \\
\Gamma_{1n_1}(z) \\
\vdots \\
\Gamma_{r1}(z) \\
\vdots \\
\Gamma_{rn_r}(z)
\end{bmatrix}.
$$

Let

$$
T_0 = \begin{bmatrix}
0 & 0 & \ldots & 0 & 0 \\
1 & 0 & \ldots & 0 & 0 \\
\vdots & \vdots & \ddots & \vdots & \vdots \\
0 & 0 & \ldots & 0 & 0 \\
0 & 0 & \ldots & 1 & 0
\end{bmatrix},
$$

$$
T_i = \begin{bmatrix}
1 & 0 & \ldots & 0 & 0 \\
1 & 1 & \ldots & 0 & 0 \\
\vdots & \vdots & \ddots & \vdots & \vdots \\
1 & 1 & \ldots & 1 & 0 \\
1 & 1 & \ldots & 1 & 1
\end{bmatrix}, \quad T_{ij} = \varepsilon_i^{q^{j-1}} T_i,
$$

$$
i = 1, \ldots, r, \quad j = 1, \ldots, n_i,
$$

$$
T = \begin{bmatrix}
T_0 & & & & & & & \\
& T_{11} & & & & & & \\
& & \ddots & & & & & \\
& & & T_{1n_1} & & & & \\
& & & & \ddots & & & \\
& & & & & T_{r1} & & \\
& & & & & & \ddots & \\
& & & & & & & T_{rn_r}
\end{bmatrix},
$$

where the dimension of T_i is $l_i \times l_i$, $i = 0, 1, \ldots, r$. Since $1/z(1-\varepsilon z)^k = 1/z + \sum_{i=1}^{k} \varepsilon/(1-\varepsilon z)^i$, the nonnegative term part of $1/z(1-\varepsilon z)^k$ is $\sum_{i=1}^{k} \varepsilon/(1-\varepsilon z)^i$. It follows that the nonnegative term part of $z^{-1}\Gamma_{ij}(z)$ is $T_{ij}\Gamma_{ij}(z)$. Clearly, the nonnegative term part of $z^{-1}\Gamma_0(z)$ is $T_0\Gamma_0$. Thus the nonnegative term part of $z^{-1}\Gamma(z)$ is $T\Gamma(z)$. From (7.31), it is easy to prove by simple induction that

$$\begin{bmatrix} \mathrm{Res}'(c_1(z)z^{n-n'+k}, g(z))/g(z) \\ \vdots \\ \mathrm{Res}'(c_m(z)z^{n-n'+k}, g(z))/g(z) \end{bmatrix} = RT^{n'-1-k}\Gamma(z), \ k = 0, \ldots, n'-1.$$

From Corollary 7.2.2, $RT^{n'-1-k}\Gamma(z)$, $k = 0, \ldots, n'-1$ form the polynomial basis of $\Phi_{M^*}(z)$.

Suppose that $\Omega(z) \in \Phi_{M^*}(z)$ has the polynomial coordinate $[h_0, \ldots, h_{n'-1}]^{\mathrm{T}}$. Then

$$\Omega(z) = \sum_{k=0}^{n'-1} h_k R T^{n'-1-k}\Gamma(z) = R\Big(\sum_{k=0}^{n'-1} h_k T^{n'-1-k}\Big)\Gamma(z).$$

We use $\mathrm{Res}(a(z), b(z))$ to denote the remainder of $a(z)$ on division by $b(z)$. Let $h_{ij}(z) = \mathrm{Res}(\sum_{k=0}^{n'-1} h_k z^{n'-1-k}, (z-\varepsilon_i^{q^{j-1}})^{l_i})$, $i = 1, \ldots, r$, $j = 1, \ldots, n_i$. Let $h_0(z) = \mathrm{Res}(\sum_{k=0}^{n'-1} h_k z^{n'-1-k}, z^{l_0})$. Since the minimal polynomial of T_{ij} is $(z-\varepsilon_i^{q^{j-1}})^{l_i}$, $i = 1, \ldots, r$, $j = 1, \ldots, n_i$ and the minimal polynomial of T_0 is z^{l_0}, we have

$$\Omega(z) = R \begin{bmatrix} h_0(T_0) & & & & & & & & \\ & h_{11}(T_{11}) & & & & & & & \\ & & \ddots & & & & & & \\ & & & h_{1n_1}(T_{1n_1}) & & & & & \\ & & & & \ddots & & & & \\ & & & & & h_{r1}(T_{r1}) & & & \\ & & & & & & \ddots & & \\ & & & & & & & h_{rn_r}(T_{rn_r}) \end{bmatrix} \Gamma(z).$$

It follows that

$$\Omega(z) = R_0 h_0(T_0)\Gamma_0(z) + \sum_{i=1}^{r}\sum_{j=1}^{n_i} R_{ij} h_{ij}(T_{ij})\Gamma_{ij}(z).$$

Define the $l_i \times l_i$ matrix H_i

$$H_i = \begin{bmatrix} 0 & 0 & \dots & 0 & 0 \\ 1 & 0 & \dots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \dots & 0 & 0 \\ 0 & 0 & \dots & 1 & 0 \end{bmatrix}, \ i = 0, 1, \dots, r. \tag{7.32}$$

Then we have $H_0 = T_0$, $\varepsilon_i^{q^{j-1}} \sum_{k=0}^{l_i-1} H_i^k = T_{ij}$, $i = 1, \dots, r$, $j = 1, \dots, n_i$. It is evident that the minimal polynomial of H_i is z^{l_i}, $i = 1, \dots, r$. Let $h'_{ij}(z) = \mathrm{Res}(h_{ij}(\varepsilon_i^{q^{j-1}} \sum_{k=0}^{l_i-1} z^k), z^{l_i})$, $i = 1, \dots, r$, $j = 1, \dots, n_i$, $h'_0(z) = h_0(z)$. Then we have

$$\Omega(z) = R_0 h'_0(H_0) \Gamma_0(z) + \sum_{i=1}^{r} \sum_{j=1}^{n_i} R_{ij} h'_{ij}(H_i) \Gamma_{ij}(z).$$

Since $l_0 + \sum_{i=1}^{r} n_i l_i = n'$ and the dimension of $\Phi_{M^*}(z)$ is n', $R_0 H_0^{k-1} \Gamma_0(z)$,$k = 1, \dots, l_0$, $R_{ij} H_i^{k-1} \Gamma_{ij}(z)$, $i = 1, \dots, r$, $j = 1, \dots, n_i$, $k = 1, \dots, l_i$ form a basis of $\Phi_{M^*}(z)$. Since R_{ijk} is the k-th column of R_{ij} and R_{0k} is the k-th column of R_0, we have

$$\begin{aligned} &R_0 H_0^{k-1} = [R_{0k} \ \dots \ R_{0l_0} \ 0 \ \dots \ 0] = R_0(l_0 + 1 - k), \quad k = 1, \dots, l_0, \\ &R_{ij} H_i^{k-1} = [R_{ijk} \ \dots \ R_{ijl_i} \ 0 \ \dots \ 0] = R_{ij}(l_i + 1 - k), \\ &\quad i = 1, \dots, r, \ j = 1, \dots, n_i, \ k = 1, \dots, l_i. \end{aligned}$$

Therefore, $R_0(k)\Gamma_0(z)$,$k = 1, \dots, l_0$, $R_{ij}(k)\Gamma_{ij}(z)$, $i = 1, \dots, r$, $j = 1, \dots, n_i$, $k = 1, \dots, l_i$ form a basis of $\Phi_{M^*}(z)$. □

The basis mentioned in the theorem is called the $(\varepsilon_1, \dots, \varepsilon_r)$ *root basis* of $\Phi_{M^*}(z)$. For any state s of M^*, any $\beta_k \in GF(q^*)$, $k = 0, \dots, l_0 - 1$, any $\beta_{ijk} \in GF(q^*)$, $i = 1, \dots, r$, $j = 1, \dots, n_i$, $k = 1, \dots, l_i$,

$$\begin{aligned} \beta = [&\beta_0, \dots, \beta_{l_0-1}, \ \beta_{111}, \dots, \beta_{1n_11}, \ \dots, \ \beta_{11l_1}, \dots, \beta_{1n_1l_1}, \\ &\dots\dots, \ \beta_{r11}, \dots, \beta_{rn_r1}, \ \dots, \ \beta_{r1l_r}, \dots, \beta_{rn_rl_r}]^{\mathrm{T}} \end{aligned}$$

is called the $(\varepsilon_1, \dots, \varepsilon_r)$ *root coordinate* of $\Phi_{M^*}(s, z)$, if

$$\Phi_{M^*}(s, z) = \sum_{k=1}^{l_0} \beta_{k-1} R_0(k) \Gamma_0(z) + \sum_{i=1}^{r} \sum_{j=1}^{n_i} \sum_{k=1}^{l_i} \beta_{ijk} R_{ij}(k) \Gamma_{ij}(z).$$

Let

$$\begin{aligned} &\Gamma_k = [\underbrace{0, \dots, 0}_{k}, 1, 0, \dots, 0, \dots], \ k = 0, \dots, l_0 - 1, \\ &\Gamma_k(\varepsilon_i^{q^{j-1}}) = \left[1, \tbinom{k}{k-1}\varepsilon_i^{q^{j-1}}, \dots, \tbinom{\tau+k-1}{k-1}\varepsilon_i^{\tau q^{j-1}}, \dots\right], \\ &\quad i = 1, \dots, r, \ j = 1, \dots, n_i, \ k = 1, \dots, l_i. \end{aligned}$$

Clearly, the generating function of Γ_k is z^k, $k = 0, \ldots, l_0 - 1$. We prove by induction on k that $1/(1-\varepsilon z)^k = \sum_{\tau=0}^{\infty} \binom{\tau+k-1}{k-1} \varepsilon^\tau z^\tau$. *Basis* : $k = 1$. $1/(1-\varepsilon z) = \sum_{\tau=0}^{\infty} (\varepsilon z)^\tau = \sum_{\tau=0}^{\infty} \binom{\tau+0}{0} \varepsilon^\tau z^\tau$. *Induction step* : Suppose that $1/(1-\varepsilon z)^k = \sum_{\tau=0}^{\infty} \binom{\tau+k-1}{k-1} \varepsilon^\tau z^\tau$. Using (7.4), we then have

$$\begin{aligned} 1/(1-\varepsilon z)^{k+1} &= (1/(1-\varepsilon z)^k)(1/(1-\varepsilon z)) \\ &= \Big(\sum_{\tau=0}^{\infty} \tbinom{\tau+k-1}{k-1} \varepsilon^\tau z^\tau\Big)\Big(\sum_{\tau=0}^{\infty} \varepsilon^\tau z^\tau\Big) \\ &= \sum_{\tau=0}^{\infty} \sum_{i=0}^{\tau} \tbinom{i+k-1}{k-1} \varepsilon^\tau z^\tau \\ &= \sum_{\tau=0}^{\infty} \tbinom{\tau+k}{k} \varepsilon^\tau z^\tau. \end{aligned}$$

Therefore, the generating function of $\Gamma_k(\varepsilon_i^{q^{j-1}})$ is $1/(1-\varepsilon_i^{q^{j-1}} z)^k$, $i = 1, \ldots, r$, $j = 1, \ldots, n_i$, $k = 1, \ldots, l_i$. Then

$$R_0(k) \begin{bmatrix} \Gamma_0 \\ \vdots \\ \Gamma_{l_0-1} \end{bmatrix}, \ k = 1, \ldots, l_0,$$

$$R_{ij}(k) \begin{bmatrix} \Gamma_1(\varepsilon_i^{q^{j-1}}) \\ \vdots \\ \Gamma_{l_i}(\varepsilon_i^{q^{j-1}}) \end{bmatrix},$$

$$i = 1, \ldots, r, \ j = 1, \ldots, n_i, \ k = 1, \ldots, l_i$$

form a basis of Φ_{M^*}, which is referred to as the $(\varepsilon_1, \ldots, \varepsilon_r)$ root basis of Φ_{M^*}. Similarly, coordinates relative to the $(\varepsilon_1, \ldots, \varepsilon_r)$ root basis are called $(\varepsilon_1, \ldots, \varepsilon_r)$ root coordinates.

Let β be the $(\varepsilon_1, \ldots, \varepsilon_r)$ root coordinate of $\Omega = [y_0, y_1, \ldots, y_\tau, \ldots]$ in Φ_{M^*}. Then

$$\begin{aligned} \Omega &= \sum_{k=1}^{l_0} \beta_{k-1} R_0(k) \begin{bmatrix} \Gamma_0 \\ \vdots \\ \Gamma_{l_0-1} \end{bmatrix} + \sum_{i=1}^{r} \sum_{j=1}^{n_i} \sum_{k=1}^{l_i} \beta_{ijk} R_{ij}(k) \begin{bmatrix} \Gamma_1(\varepsilon_i^{q^{j-1}}) \\ \vdots \\ \Gamma_{l_i}(\varepsilon_i^{q^{j-1}}) \end{bmatrix} \\ &= \sum_{k=1}^{l_0} \beta_{k-1} \sum_{h=1}^{k} R_{0(l_0+h-k)} \Gamma_{h-1} + \sum_{i=1}^{r} \sum_{j=1}^{n_i} \sum_{k=1}^{l_i} \beta_{ijk} \sum_{h=1}^{k} R_{ij(l_i+h-k)} \Gamma_h(\varepsilon_i^{q^{j-1}}) \\ &= \sum_{h=1}^{l_0} \Big(\sum_{k=h}^{l_0} \beta_{k-1} R_{0(l_0+h-k)} \Big) \Gamma_{h-1} \end{aligned}$$

$$+\sum_{i=1}^{r}\sum_{j=1}^{n_i}\sum_{h=1}^{l_i}\Big(\sum_{k=h}^{l_i}\beta_{ijk}R_{ij(l_i+h-k)}\Big)\Gamma_h(\varepsilon_i^{q^{j-1}})$$

$$=\sum_{h=1}^{l_0}\Big(\sum_{k=h}^{l_0}\beta_{l_0+h-k-1}R_{0k}\Big)\Gamma_{h-1}$$

$$+\sum_{i=1}^{r}\sum_{j=1}^{n_i}\sum_{h=1}^{l_i}\Big(\sum_{k=h}^{l_i}\beta_{ij(l_i+h-k)}R_{ijk}\Big)\Gamma_h(\varepsilon_i^{q^{j-1}}).$$

Thus we have

$$y_\tau=\sum_{k=\tau+1}^{l_0}\beta_{l_0+\tau-k}R_{0k}+\sum_{i=1}^{r}\sum_{j=1}^{n_i}\sum_{h=1}^{l_i}\Big(\sum_{k=h}^{l_i}\beta_{ij(l_i+h-k)}R_{ijk}\Big)\binom{\tau+h-1}{h-1}\varepsilon_i^{\tau q^{j-1}},$$
$$\tau=0,1,\ldots \tag{7.33}$$

Clearly, whenever (7.33) holds, β is the $(\varepsilon_1,\ldots,\varepsilon_r)$ root coordinate of Ω.

We discuss a special case, where $m=1$ and the output matrix of M is $[1,0,\ldots,0]$. Thus we have $c_1(z)=1$. It follows that the second characteristic polynomial and the characteristic polynomial of M are the same. Since $R_0(1)\Gamma_0(z)$ and $R_{ij}(1)\Gamma_{ij}(z)$ are basis vectors, they are not zero; therefore, $R_{0l_0}\neq 0$ and $R_{ijl_i}\neq 0$. Noticing $m=1$, from Theorem 7.2.3, $R'_0(k)\Gamma_0(z)$,$k=1,\ldots,l_0$, $R'_{ij}(k)\Gamma_{ij}(z)$, $i=1,\ldots,r$, $j=1,\ldots,n_i$, $k=1,\ldots,l_i$ form a basis of $\Phi_{M^*}(z)$, where

$$R'_0(k)=[\underbrace{0,\ldots,0}_{k-1},1,\underbrace{0,\ldots,0}_{l_0-k}],\quad k=1,\ldots,l_0,$$
$$R'_{ij}(k)=[\underbrace{0,\ldots,0}_{k-1},1,\underbrace{0,\ldots,0}_{l_i-k}],$$
$$i=1,\ldots,r,\ j=1,\ldots,n_i,\ k=1,\ldots,l_i.$$

That is, $1,z,\ldots,z^{l_0-1}$, $1/(1-\varepsilon_i^{q^{j-1}}z)^k$, $i=1,\ldots,r$, $j=1,\ldots,n_i$, $k=1,\ldots,l_i$ form a basis of $\Phi_{M^*}(z)$. We complete a proof of the following.

Corollary 7.2.4. *Assume that the characteristic polynomial $f(z)$ of the linear shift register M has the factorization*

$$f(z)=z^{l_0}\prod_{i=1}^{r}\prod_{j=1}^{n_i}(z-\varepsilon_i^{q^{j-1}})^{l_i},$$

where $\varepsilon_1,\ldots,\varepsilon_r$ are nonzero elements in $GF(q^*)$ of which minimal polynomials over $GF(q)$ are coprime and have degrees $n_1,\ldots,n_r$, respectively, and $l_0\geqslant 0$, $l_1>0,\ldots,l_r>0$. Then $1,z,\ldots,z^{l_0-1}$, $1/(1-\varepsilon_i^{q^{j-1}}z)^k$, $i=1,\ldots,r$, $j=1,\ldots,n_i$, $k=1,\ldots,l_i$ form a basis of $\Psi^{(1)}_{M^*}(z)$.

The basis mentioned in the corollary is called the $(\varepsilon_1, \ldots, \varepsilon_r)$ *root basis* of $\Psi^{(1)}_{M^*}(z)$. For any $\Omega(z) \in \Psi^{(1)}_{M^*}(z)$, any $\beta_k \in GF(q^*)$, $k = 0, \ldots, l_0 - 1$, any $\beta_{ijk} \in GF(q^*)$, $i = 1, \ldots, r$, $j = 1, \ldots, n_i$, $k = 1, \ldots, l_i$,

$$\begin{aligned}\beta = [&\beta_0, \ldots, \beta_{l_0-1},\ \beta_{111}, \ldots, \beta_{1n_11},\ \ldots,\ \beta_{11l_1}, \ldots, \beta_{1n_1l_1},\\ &\ldots\ldots,\ \beta_{r11}, \ldots, \beta_{rn_r1},\ \ldots,\ \beta_{r1l_r}, \ldots, \beta_{rn_rl_r}]^{\mathrm{T}}\end{aligned}$$

is called the $(\varepsilon_1, \ldots, \varepsilon_r)$ *root coordinate* of $\Omega(z)$, if

$$\Omega(z) = \sum_{k=1}^{l_0} \beta_{k-1} z^{k-1} + \sum_{i=1}^{r} \sum_{j=1}^{n_i} \sum_{k=1}^{l_i} \beta_{ijk} / (1 - \varepsilon_i^{q^{j-1}} z)^k.$$

Corresponding to $\Psi^{(1)}_{M^*}$, $\Gamma_0, \ldots, \Gamma_{l_0-1}$, $\Gamma_k(\varepsilon_i^{q^{j-1}})$, $i = 1, \ldots, r$, $j = 1, \ldots, n_i$, $k = 1, \ldots, l_i$ form a basis of $\Psi^{(1)}_{M^*}$ which is referred to as the $(\varepsilon_1, \ldots, \varepsilon_r)$ root basis of $\Psi^{(1)}_{M^*}$. Similarly, coordinates relative to the $(\varepsilon_1, \ldots, \varepsilon_r)$ root basis are called $(\varepsilon_1, \ldots, \varepsilon_r)$ root coordinates.

Let β be the $(\varepsilon_1, \ldots, \varepsilon_r)$ root coordinate of $\Omega = [s_0, s_1, \ldots, s_\tau, \ldots]$ in $\Psi^{(1)}_{M^*}$. Then

$$\Omega = \sum_{k=1}^{l_0} \beta_{k-1} \Gamma_{k-1} + \sum_{i=1}^{r} \sum_{j=1}^{n_i} \sum_{k=1}^{l_i} \beta_{ijk} \Gamma_k(\varepsilon_i^{q^{j-1}}).$$

Thus we have

$$\begin{aligned} s_\tau &= \beta_\tau + \sum_{i=1}^{r} \sum_{j=1}^{n_i} \sum_{k=1}^{l_i} \beta_{ijk} \tbinom{\tau+k-1}{k-1} \varepsilon_i^{\tau q^{j-1}}, \\ &\tau = 0, 1, \ldots, \end{aligned} \tag{7.34}$$

where $\beta_\tau = 0$ whenever $\tau \geqslant l_0$.

We define matrices over $GF(q^*)$

$$A_i = \begin{bmatrix} 1 & 1 & \ldots & 1 \\ \varepsilon_i & \varepsilon_i^q & \ldots & \varepsilon_i^{q^{n_i-1}} \\ \vdots & \vdots & \ddots & \vdots \\ \varepsilon_i^{n-1} & \varepsilon_i^{(n-1)q} & \ldots & \varepsilon_i^{(n-1)q^{n_i-1}} \end{bmatrix},\ i = 1, \ldots, r,$$

$$B_k = \begin{bmatrix} \binom{k-1}{k-1} & & & \\ & \binom{k}{k-1} & & \\ & & \ddots & \\ & & & \binom{n-2+k}{k-1} \end{bmatrix},\ k = 1, 2, \ldots, \tag{7.35}$$

$$D_0 = [B_1 A_1, \ldots, B_{l_1} A_1, \ldots, B_1 A_r, \ldots, B_{l_r} A_r],$$

$$E_0' = \begin{bmatrix} E_{l_0} \\ 0 \end{bmatrix},$$

$$D(\varepsilon_1, \ldots, \varepsilon_r, M) = [E_0', D_0],$$

where E_{l_0} is the $l_0 \times l_0$ identity matrix over $GF(q)$, and the dimension of E_0' is $n \times l_0$. It is easy to verify that

$$s_\tau = \beta_\tau + \sum_{i=1}^{r} \sum_{j=1}^{n_i} \sum_{k=1}^{l_i} \beta_{ijk} \binom{\tau+k-1}{k-1} \varepsilon_i^{\tau q^{j-1}}, \quad \tau = 0, 1, \ldots, n-1$$

can be written as $[s_0, \ldots, s_{n-1}]^{\mathrm{T}} = D(\varepsilon_1, \ldots, \varepsilon_r, M)\beta$. Since $[s_0, s_1, \ldots, s_\tau, \ldots]$ $= \Psi_{M^*}^{(1)}(s)$ implies $s = [s_0, \ldots, s_{n-1}]^{\mathrm{T}}$, $\Psi_{M^*}^{(1)}(s) = \Psi_{M^*}^{(1)}(s')$ if and only if $s = s'$. Thus $D(\varepsilon_1, \ldots, \varepsilon_r, M)$ is nonsingular.

Corollary 7.2.5. *For any state s of M^*, the $(\varepsilon_1, \ldots, \varepsilon_r)$ root coordinate of $\Psi_{M^*}^{(1)}(s)$ is $D(\varepsilon_1, \ldots, \varepsilon_r, M)^{-1}s$.*

We turn to characterizing sequences over $GF(q)$. For any integer u, let

$$F_u(\varepsilon_i) = \Big\{ \sum_{j=1}^{n_i} b_j \varepsilon_i^{u+j} \mid b_1, \ldots, b_{n_i} \in GF(q) \Big\}, \ i = 1, \ldots, r. \quad (7.36)$$

It is easy to verify that $F_u(\varepsilon_i)$ is a subfield of $GF(q^*)$ of which elements consist of all roots of $z^{q^{n_i}} - z$. Thus $F_u(\varepsilon_i) = GF(q^{n_i})$. Since elements in $GF(q)$ consist of all roots of $z^q - z$ and $z^q - z$ is a divisor of $z^{q^{n_i}} - z$, $GF(q)$ is a subfield of $F_u(\varepsilon_i)$.

Theorem 7.2.4. *Let $\Omega \in \Psi_{M^*}^{(1)}$. Then $\Omega \in \Psi_M^{(1)}$ if and only if in the $(\varepsilon_1, \ldots, \varepsilon_r)$ root coordinate β of Ω, $\beta_k \in GF(q)$, $k = 0, \ldots, l_0 - 1$, β_{ijk} $\in GF(q^{n_i})$ and $\beta_{ijk} = \beta_{i1k}^{q^{j-1}}$, $i = 1, \ldots, r$, $j = 1, \ldots, n_i$, $k = 1, \ldots, l_i$.*

Proof. Suppose that the condition in the theorem holds. Then there exist $b_{ihk} \in GF(q)$, $i = 1, \ldots, r$, $h = 1, \ldots, n_i$, $k = 1, \ldots, l_i$ such that $\beta_{i1k} = \sum_{h=1}^{n_i} b_{ihk} \varepsilon_i^{u+h}$, $i = 1, \ldots, r$, $k = 1, \ldots, l_i$. Let $\Omega = [s_0, s_1, \ldots, s_\tau, \ldots]$. From (7.34), we have

$$\begin{aligned} s_\tau &= \beta_\tau + \sum_{i=1}^{r} \sum_{j=1}^{n_i} \sum_{k=1}^{l_i} \beta_{i1k}^{q^{j-1}} \binom{\tau+k-1}{k-1} \varepsilon_i^{\tau q^{j-1}} \\ &= \beta_\tau + \sum_{i=1}^{r} \sum_{j=1}^{n_i} \sum_{k=1}^{l_i} \sum_{h=1}^{n_i} b_{ihk} \binom{\tau+k-1}{k-1} \varepsilon_i^{(u+h+\tau)q^{j-1}}, \\ &\tau = 0, 1, \ldots, \end{aligned}$$

where $\beta_\tau = 0$ if $\tau \geqslant l_0$. From $\varepsilon_i^{q^{n_i}} = \varepsilon_i$, $\sum_{j=1}^{n_i} \varepsilon_i^{(u+h+\tau)q^j} = \sum_{j=1}^{n_i} \varepsilon_i^{(u+h+\tau)q^{j-1}}$ holds; this yields

$$
\begin{aligned}
s_\tau^q &= \beta_\tau + \sum_{i=1}^{r}\sum_{j=1}^{n_i}\sum_{k=1}^{l_i}\sum_{h=1}^{n_i} b_{ihk}\tbinom{\tau+k-1}{k-1}\varepsilon_i^{(u+h+\tau)q^j} \\
&= \beta_\tau + \sum_{i=1}^{r}\sum_{j=1}^{n_i}\sum_{k=1}^{l_i}\sum_{h=1}^{n_i} b_{ihk}\tbinom{\tau+k-1}{k-1}\varepsilon_i^{(u+h+\tau)q^{j-1}}, \\
&= s_\tau, \\
&\tau = 0, 1, \ldots
\end{aligned}
$$

Thus $s_\tau \in GF(q)$, $\tau = 0, 1, \ldots$ We conclude that $\Omega \in \Psi_M^{(1)}$.

Since the number of β's which satisfy the condition in the theorem is equal to $q^{l_0+\sum_{i=1}^{r} n_i l_i} = q^n$ and the dimension of $\Psi_M^{(1)}$ is n, such β's determine the subset $\Psi_M^{(1)}$ of $\Psi_{M^*}^{(1)}$; this completes the proof of the theorem. □

Corollary 7.2.6. *Assume that the characteristic polynomial $f(z)$ of the linear shift register M has the factorization*

$$
f(z) = z^{l_0}\prod_{i=1}^{r}\prod_{j=1}^{n_i}(z - \varepsilon_i^{q^{j-1}})^{l_i},
$$

where $\varepsilon_1, \ldots, \varepsilon_r$ are nonzero elements in $GF(q^*)$ of which minimal polynomials over $GF(q)$ are coprime and have degrees $n_1, \ldots, n_r$, respectively, and $l_0 \geqslant 0$, $l_1 > 0$, $\ldots$, $l_r > 0$. Let u be an integer. Then

$$
\begin{aligned}
\Gamma_k &= \big[\underbrace{0, \ldots, 0}_{k}, 1, 0, \ldots, 0, \ldots\big], \ k = 0, \ldots, l_0 - 1, \\
\Gamma_{hk}(\varepsilon_i, u) &= \big[\sum_{j=1}^{n_i}\varepsilon_i^{(u+h)q^{j-1}}, \tbinom{k}{k-1}\sum_{j=1}^{n_i}\varepsilon_i^{(u+h+1)q^{j-1}}, \ldots, \\
&\qquad\qquad \tbinom{\tau+k-1}{k-1}\sum_{j=1}^{n_i}\varepsilon_i^{(u+h+\tau)q^{j-1}}, \ldots\big], \\
&i = 1, \ldots, r, \ h = 1, \ldots, n_i, \ k = 1, \ldots, l_i
\end{aligned}
$$

form a basis of $\Psi_M^{(1)}$.

Proof. From Theorem 7.2.4 and its proof, for any $\Omega = [s_0, s_1, \ldots, s_\tau, \ldots]$ in $\Psi_M^{(1)}$, there uniquely exist $b_0, \ldots, b_{l_0-1}$, b_{ihk}, $i = 1, \ldots, r$, $h = 1, \ldots, n_i$, $k = 1, \ldots, l_i$ in $GF(q)$ such that

$$
\begin{aligned}
s_\tau &= b_\tau + \sum_{i=1}^{r}\sum_{j=1}^{n_i}\sum_{k=1}^{l_i}\sum_{h=1}^{n_i} b_{ihk}\tbinom{\tau+k-1}{k-1}\varepsilon_i^{(u+h+\tau)q^{j-1}}, \\
&\tau = 0, 1, \ldots,
\end{aligned}
$$

where $b_\tau = 0$ if $\tau \geqslant l_0$. That is,

$$\Omega = \sum_{k=1}^{l_0} b_{k-1}\Gamma_{k-1} + \sum_{i=1}^{r}\sum_{h=1}^{n_i}\sum_{k=1}^{l_i} b_{ihk}\Gamma_{hk}(\varepsilon_i, u). \tag{7.37}$$

On the other hand, since each $\Gamma_{hk}(\varepsilon_i, u)$ is a sequence over $GF(q)$, such Ω with the above expression is in $\Psi_M^{(1)}$. Thus Γ_k, $k = 0, \ldots, l_0 - 1$, $\Gamma_{hk}(\varepsilon_i, u)$, $i = 1, \ldots, r$, $h = 1, \ldots, n_i$, $k = 1, \ldots, l_i$ form a basis of $\Psi_M^{(1)}$. □

The basis mentioned in the corollary is called the $(\varepsilon_1, \ldots, \varepsilon_r, u)$ *root basis* of $\Psi_M^{(1)}$. If Ω is expressed as (7.37),

$$\begin{aligned} b = [&b_0, \ldots, b_{l_0-1},\ b_{111}, \ldots, b_{1n_11},\ \ldots,\ b_{11l_1}, \ldots, b_{1n_1l_1},\\ &\ldots\ldots,\ b_{r11}, \ldots, b_{rn_r1},\ \ldots,\ b_{r1l_r}, \ldots, b_{rn_rl_r}]^{\mathrm{T}} \end{aligned}$$

is called the $(\varepsilon_1, \ldots, \varepsilon_r, u)$ *root coordinate* of Ω.

Using Corollary 7.2.5, we have the following.

Corollary 7.2.7. *Let*

$$G_i(u) = \begin{bmatrix} \varepsilon_i^{u+1} & \varepsilon_i^{u+2} & \cdots & \varepsilon_i^{u+n_i} \\ \varepsilon_i^{(u+1)q} & \varepsilon_i^{(u+2)q} & \cdots & \varepsilon_i^{(u+n_i)q} \\ \vdots & \vdots & \ddots & \vdots \\ \varepsilon_i^{(u+1)q^{n_i-1}} & \varepsilon_i^{(u+2)q^{n_i-1}} & \cdots & \varepsilon_i^{(u+n_i)q^{n_i-1}} \end{bmatrix},\ i = 1, \ldots, r,$$

$$G(u) = \begin{bmatrix} E_{l_0} & & & & & & \\ & G_1(u) & & & & & \\ & & \ddots & & & & \\ & & & G_1(u) & & & \\ & & & & \ddots & & \\ & & & & & G_r(u) & \\ & & & & & & \ddots \\ & & & & & & G_r(u) \end{bmatrix} \tag{7.38}$$

with l_i $G_i(u)$ for $i = 1, \ldots, r$, where E_{l_0} is the $l_0 \times l_0$ identity matrix. Then $DG(u)$ is a nonsingular matrix over $GF(q)$ and for any Ω in $\Psi_M^{(1)}$ with $(\varepsilon_1, \ldots, \varepsilon_r, u)$ root coordinate b, $\Omega = \Psi_M^{(1)}(s)$ holds, where $s = D(\varepsilon_1, \ldots, \varepsilon_r, M)$ $G(u)b$, $D(\varepsilon_1, \ldots, \varepsilon_r, M)$ is defined in (7.35).

The following corollary is a z-transformational version for Theorem 7.2.4.

Corollary 7.2.8. *Assume that the characteristic polynomial $f(z)$ of the linear shift register M has the factorization*

$$f(z) = z^{l_0} \prod_{i=1}^{r} \prod_{j=1}^{n_i} (z - \varepsilon_i^{q^{j-1}})^{l_i},$$

where $\varepsilon_1, \ldots \varepsilon_r$ are nonzero elements in $GF(q^*)$ of which minimal polynomials over $GF(q)$ are coprime and have degrees $n_1, \ldots, n_r$, respectively, and $l_0 \geqslant 0$, $l_1 > 0, \ldots, l_r > 0$. For any $\Omega(z) \in \Psi^{(1)}_{M^*}(z)$, $\Omega(z)$ is in $\Psi^{(1)}_M(z)$ if and only if there exist $\beta_k \in GF(q)$, $k = 0, \ldots, l_0 - 1$, $\beta_{ik} \in GF(q^{n_i})$, $i = 1, \ldots, r$, $k = 1, \ldots, l_i$ such that

$$\Omega(z) = \sum_{k=0}^{l_0-1} \beta_k z^k + \sum_{i=1}^{r} \sum_{j=1}^{n_i} \sum_{k=1}^{l_i} \beta_{ik}^{q^{j-1}} / (1 - \varepsilon_i^{q^{j-1}} z)^k.$$

Moreover, whenever such β's exist, they are unique.

Using Theorem 7.2.1, Corollary 7.2.8 implies the following corollary.

Corollary 7.2.9. *Let $f(z)$ be a polynomial of degree n over $GF(q)$, and $g(z) = z^n f(1/z)$. Assume that*

$$f(z) = z^{l_0} \prod_{i=1}^{r} \prod_{j=1}^{n_i} (z - \varepsilon_i^{q^{j-1}})^{l_i},$$

where $\varepsilon_1, \ldots, \varepsilon_r$ are nonzero elements in $GF(q^*)$ of which minimal polynomials over $GF(q)$ are coprime and have degrees $n_1, \ldots, n_r$, respectively, and $l_0 \geqslant 0$, $l_1 > 0, \ldots, l_r > 0$. For any polynomial $h(z)$ of degree $< n$ over $GF(q^*)$, $h(z)$ is a polynomial over $GF(q)$ if and only if there exist $\beta_k \in GF(q)$, $k = 0, \ldots, l_0 - 1$, $\beta_{ik} \in GF(q^{n_i})$, $i = 1, \ldots, r$, $k = 1, \ldots, l_i$ such that

$$h(z)/g(z) = \sum_{k=0}^{l_0-1} \beta_k z^k + \sum_{i=1}^{r} \sum_{j=1}^{n_i} \sum_{k=1}^{l_i} \beta_{ik}^{q^{j-1}} / (1 - \varepsilon_i^{q^{j-1}} z)^k.$$

Moreover, whenever such β's exist, they are unique.

Theorem 7.2.5. *Let $\Omega \in \Phi_{M^*}$. Then Ω is in Φ_M if and only if in the $(\varepsilon_1, \ldots, \varepsilon_r)$ root coordinate β of Ω, $\beta_k \in GF(q)$, $k = 0, \ldots, l_0 - 1$, $\beta_{ijk} \in GF(q^{n_i})$ and $\beta_{ijk} = \beta_{i1k}^{q^{j-1}}$, $i = 1, \ldots, r$, $j = 1, \ldots, n_i$, $k = 1, \ldots, l_i$.*

Proof. Suppose that the condition in the theorem holds. Let $\Omega = [y_0, y_1, \ldots, y_\tau, \ldots]$ and $y_\tau = [y_{\tau 1}, \ldots, y_{\tau m}]^{\mathrm{T}}$, $\tau = 0, 1, \ldots$ Let $R_{0k} = [r_{0k1}, \ldots, r_{0km}]^{\mathrm{T}}$, $k = 1, \ldots, l_0$, $R_{ijk} = [r_{ijk1}, \ldots, r_{ijkm}]^{\mathrm{T}}$, $i = 1, \ldots, r$, $j = 1, \ldots, n_i$, $k = 1, \ldots, l_i$. From (7.33), we have

$$y_{\tau c} = \sum_{k=\tau+1}^{l_0} \beta_{l_0+\tau-k} r_{0kc} + \sum_{i=1}^{r} \sum_{j=1}^{n_i} \sum_{h=1}^{l_i} \Big(\sum_{k=h}^{l_i} \beta_{ij(l_i+h-k)} r_{ijkc} \Big) \binom{\tau+h-1}{h-1} \varepsilon_i^{\tau q^{j-1}},$$
$$\tau = 0, 1, \ldots, \ c = 1, \ldots, m.$$

From $\beta_{i1k} \in GF(q^{n_i})$, there exist b_{idk}, $i = 1, \ldots, r$, $d = 1, \ldots, n_i$, $k = 1, \ldots, l_i$ such that

$$\beta_{i1k} = \sum_{d=1}^{n_i} b_{idk} \varepsilon_i^{u+d}, \quad i = 1, \ldots, r, \ k = 1, \ldots, l_i.$$

From (7.28), using Corollary 7.2.9, we have $r_{0kc} \in GF(q)$, $k = 1, \ldots, l_0$, $c = 1, \ldots, m$, and $r_{ijkc} = r_{i1kc}^{q^{j-1}} \in GF(q^{n_i})$, $i = 1, \ldots, r$, $j = 1, \ldots, n_i$, $k = 1, \ldots, l_i$, $c = 1, \ldots, m$. Thus there exist p_{iekc} in $GF(q)$, $i = 1, \ldots, r$, $e = 1, \ldots, n_i$, $k = 1, \ldots, l_i$, $c = 1, \ldots, m$ such that

$$r_{i1kc} = \sum_{e=1}^{n_i} p_{iekc} \varepsilon_i^{u+e}, \quad i = 1, \ldots, r, \ k = 1, \ldots, l_i, \ c = 1, \ldots, m.$$

It follows that

$$\begin{aligned} y_{\tau c} = & \sum_{k=\tau+1}^{l_0} \beta_{l_0+\tau-k} r_{0kc} \\ & + \sum_{i=1}^{r} \sum_{j=1}^{n_i} \sum_{h=1}^{l_i} \sum_{k=h}^{l_i} \sum_{d=1}^{n_i} \sum_{e=1}^{n_i} b_{id(l_i+h-k)} p_{iekc} \tbinom{\tau+h-1}{h-1} \varepsilon_i^{(2u+d+e+\tau)q^{j-1}}, \\ & \tau = 0, 1, \ldots, \ c = 1, \ldots, m. \end{aligned}$$

From $\varepsilon_i^{q^{n_i}} = \varepsilon_i$ and β_k, r_{0kc}, b_{idk}, $p_{iekc} \in GF(q)$, we have

$$\begin{aligned} y_{\tau c}^q = & \sum_{k=\tau+1}^{l_0} \beta_{l_0+\tau-k}^q r_{0kc}^q \\ & + \sum_{i=1}^{r} \sum_{j=1}^{n_i} \sum_{h=1}^{l_i} \sum_{k=h}^{l_i} \sum_{d=1}^{n_i} \sum_{e=1}^{n_i} b_{id(l_i+h-k)}^q p_{iekc}^q \tbinom{\tau+h-1}{h-1}^q \varepsilon_i^{(2u+d+e+\tau)q^j} \\ = & \sum_{k=\tau+1}^{l_0} \beta_{l_0+\tau-k} r_{0kc} \\ & + \sum_{i=1}^{r} \sum_{j=1}^{n_i} \sum_{h=1}^{l_i} \sum_{k=h}^{l_i} \sum_{d=1}^{n_i} \sum_{e=1}^{n_i} b_{id(l_i+h-k)} p_{iekc} \tbinom{\tau+h-1}{h-1} \varepsilon_i^{(2u+d+e+\tau)q^{j-1}} \\ = & \ y_{\tau c}, \\ & \tau = 0, 1, \ldots, \ c = 1, \ldots, m. \end{aligned}$$

Thus $y_{\tau c} \in GF(q)$, $\tau = 0, 1, \ldots$, $c = 1, \ldots, m$. We conclude that $\Omega \in \Phi_M$.

Since the number of β's which satisfy the condition in the theorem is equal to $q^{l_0 + \sum_{i=1}^{r} n_i l_i} = q^{n'}$ and the dimension of Φ_M is n', such β's determine the subset Φ_M of Φ_{M^*}; this completes the proof of the theorem. □

Let M be a linear autonomous finite automaton over $GF(q)$, with structure parameters m, n and structure matrices A, C. It is well known that there exists a nonsingular matrix P over $GF(q)$ such that

$$PAP^{-1} = \begin{bmatrix} P_{f^{(1)}(z)} & & \\ & \ddots & \\ & & P_{f^{(v)}(z)} \end{bmatrix},$$

where $f^{(1)}(z), \ldots, f^{(v)}(z)$ are elementary divisors of A. Let M' be the autonomous linear finite automaton over $GF(q)$ with structure parameters m, n and structure matrices PAP^{-1}, CP^{-1}. It is easy to show that M and M' are equivalent. Thus $\Phi_M(z) = \Phi_{M'}(z)$. Let $CP^{-1} = [C_1, \ldots, C_v]$, where C_i has $n^{(i)}$ columns, $n^{(i)}$ is the degree of $f^{(i)}(z)$, $i = 1, \ldots, v$. For each i, $1 \leqslant i \leqslant v$, define a linear autonomous finite automaton M_i' with structure parameters m, n_i and structure matrices $P_{f^{(i)}(z)}$, C_i. It is easy to show that $\Phi_M(z) = \Phi_{M'}(z) = \Phi_{M_1}(z) + \cdots + \Phi_{M_v}(z)$. In the case where M is minimal, the space sum is the direct sum; therefore, all bases of $\Phi_{M_1}(z)$, ..., $\Phi_{M_v}(z)$ together form a basis of $\Phi_M(z)$.

We discuss how to obtain a basis of $\Phi_M(z)$ from bases of $\Phi_{M_1}(z)$, ..., $\Phi_{M_v}(z)$ for general M. Let $g^{(i)}(z) = z^{n^{(i)}} f^{(i)}(1/z)$, $i = 1, \ldots, v$. We use $f^{(i)'}(z)$ to denote the second characteristic polynomial of $M^{(i)}$, and $n^{(i)'}$ the degree of $f^{(i)'}(z)$, $i = 1, \ldots, v$. Let $g^{(i)'}(z) = z^{n^{(i)'}} f^{(i)'}(1/z)$, $i = 1, \ldots, v$. We use $f'(z)$ to denote the least common multiple of $f^{(1)'}(z)$, ..., $f^{(v)'}(z)$. Assume that $GF(q^*)$ is a splitting field of $f'(z)$ and that (7.27) holds. Then

$$f^{(h)'}(z) = z^{l_0^{(h)}} \prod_{i=1}^{r} \prod_{j=1}^{n_i} (z - \varepsilon_i^{q^{j-1}})^{l_i^{(h)}}, \quad h = 1, \ldots, v, \tag{7.39}$$

for some $l_i^{(h)} \geqslant 0$, $i = 0, 1, \ldots, r$, $h = 1, \ldots, v$. It follows that

$$n^{(h)} = l_0^{(h)} + \sum_{i=1}^{r} n_i l_i^{(h)}, \quad h = 1, \ldots, v.$$

Let

$$\begin{bmatrix} c_1^{(h)}(z) z^{n^{(h)}-1}/g^{(h)}(z) \\ \vdots \\ c_m^{(h)}(z) z^{n^{(h)}-1}/g^{(h)}(z) \end{bmatrix} = \sum_{k=1}^{l_0^{(h)}} R_{0k}^{(h)} z^{k-1} + \sum_{i=1}^{r} \sum_{j=1}^{n_i} \sum_{k=1}^{l_i^{(h)}} R_{ijk}^{(h)} / (1 - \varepsilon_i^{q^{j-1}} z)^k,$$

$$R_0^{(h)}(k) = [R_{0(l_0^{(h)}+1-k)}^{(h)} \ \ldots \ R_{0l_0^{(h)}}^{(h)} \ 0 \ \ldots \ 0], \quad k = 1, \ldots, l_0^{(h)}, \ h = 1, \ldots, v,$$

$$R_{ij}^{(h)}(k) = [R_{ij(l_i^{(h)}+1-k)}^{(h)} \ \ldots \ R_{ijl_i^{(h)}}^{(h)} \ 0 \ \ldots \ 0], \tag{7.40}$$

$$i = 1, \ldots, r, \ j = 1, \ldots, n_i, \ k = 1, \ldots, l_i^{(h)}, \ h = 1, \ldots, v,$$

where $c_k^{(h)}(z), k = 1, \ldots, m$ are the output polynomials of $M^{(h)}$, $h = 1, \ldots, v$. We use $M^{(h)*}$ to denote the natural extension of $M^{(h)}$ over $GF(q^*)$, $h = 1, \ldots, v$. From Theorem 7.2.3, $R_0^{(h)}(k)\Gamma_0(z), k = 1, \ldots, l_0^{(h)}$, $R_{ij}^{(h)}(k)\Gamma_{ij}(z), i = 1, \ldots, r$, $j = 1, \ldots, n_i$, $k = 1, \ldots, l_i^{(h)}$ form a basis of $\Phi_{M^{(h)*}}(z)$, $h = 1, \ldots, v$, where $\Gamma_0(z)$, $\Gamma_{ij}(z)$ are defined in (7.29). We use $\mathcal{S}_0$ to denote the set consisting of $R_0^{(h)}(k)\Gamma_0(z)$, $k = 1, \ldots, l_0^{(h)}$, $h = 1, \ldots, v$, $R_{ij}^{(h)}(k)\Gamma_{ij}(z)$, $i = 1, \ldots, r$, $j = 1, \ldots, n_i$, $k = 1, \ldots, l_i^{(h)}$, $h = 1, \ldots, v$. Clearly, $\mathcal{S}_0$ generates $\Phi_{M^*}(z)$. We use $\mathcal{S}_{00}$ to denote the set consisting of $R_0^{(h)}(k)\Gamma_0(z)$,$k = 1, \ldots, l_0^{(h)}$, $h = 1, \ldots, v$, use $\mathcal{S}_{ij0}$ to denote the set consisting of $R_{ij}^{(h)}(k)\Gamma_{ij}(z)$, $k = 1, \ldots, l_i^{(h)}$, $h = 1, \ldots, v$ for any i, $1 \leqslant i \leqslant r$, and any j, $1 \leqslant j \leqslant n_i$. Evidently, elements in $\mathcal{S}_0$ are linearly independent over $GF(q^*)$ if and only if elements in $\mathcal{S}_{00}$ are linearly independent over $GF(q^*)$ and for any i, $1 \leqslant i \leqslant r$ and any j, $1 \leqslant j \leqslant n_i$, elements in $\mathcal{S}_{ij0}$ are linearly independent over $GF(q^*)$.

Proposition 7.2.1. *Elements in $\mathcal{S}_{00}$ are linearly dependent over $GF(q^*)$ if and only if $R_{0l_0^{(h)}}^{(h)}$, $h = 1, \ldots, v$ are linearly dependent over $GF(q)$.*

Proof. Suppose that $R_{0l_0^{(h)}}^{(h)}$, $h = 1, \ldots, v$ are linearly dependent over $GF(q)$. Since all columns of $R_0^{(h)}(1)$ are 0 except the first column $R_{0l_0^{(h)}}^{(h)}$, $R_0^{(h)}(1)\Gamma_0(z)$, $h = 1, \ldots, v$ are linearly dependent over $GF(q)$. From $R_0^{(h)}(1)$ $\Gamma_0(z) \in \mathcal{S}_{00}$, elements in $\mathcal{S}_{00}$ are linearly dependent over $GF(q)$; therefore, elements in $\mathcal{S}_{00}$ are linearly dependent over $GF(q^*)$.

Suppose that elements in $\mathcal{S}_{00}$ are linearly dependent over $GF(q^*)$. Then there exist $a_{hk} \in GF(q^*)$, $h = 1, \ldots, v$, $k = 1, \ldots, l_0^{(h)}$ such that $a_{hk} \neq 0$ for some h, k and

$$\sum_{h=1}^{v} \sum_{k=1}^{l_0^{(h)}} a_{hk} R_0^{(h)}(k)\Gamma_0(z) = 0.$$

It follows that

$$\sum_{h=1}^{v} \sum_{k=1}^{l_0^{(h)}} a_{hk} R_0^{(h)}(k) = 0.$$

We use k' to denote the maximum k satisfying the condition $a_{hk} \neq 0$ for some h, $1 \leqslant h \leqslant v$. Then we have

$$\sum_{h=1}^{v} a_{hk'} R_{0l^{(h)}}^{(h)} = 0.$$

Thus $R_{0l_0^{(h)}}^{(h)}$, $h = 1, \ldots, v$ are linearly dependent over $GF(q^*)$. Since elements in $R_{0l_0^{(h)}}^{(h)}$, $h = 1, \ldots, v$ are in $GF(q)$, $R_{0l_0^{(h)}}^{(h)}$, $h = 1, \ldots, v$ are linearly dependent over $GF(q)$. □

Similarly, we can prove the following.

Proposition 7.2.2. *For any i, $1 \leqslant i \leqslant r$ and any j, $1 \leqslant j \leqslant n_i$, elements in $\mathcal{S}_{ij0}$ are linearly dependent over $GF(q^*)$ if and only if $R^{(h)}_{ijl_i^{(h)}}$, $h = 1, \ldots, v$ are linearly dependent over $GF(q^{n_i})$.*

Proposition 7.2.3. *For any i, $1 \leqslant i \leqslant r$ and any j, $1 \leqslant j \leqslant n_i$, $R^{(h)}_{ijl_i^{(h)}}$, $h = 1, \ldots, v$ are linearly dependent over $GF(q^{n_i})$ if and only if $R^{(h)}_{i1l_i^{(h)}}$, $h = 1, \ldots, v$ are linearly dependent over $GF(q^{n_i})$. Moreover, for any a_h in $GF(q^{n_i})$, $h = 1, \ldots, v$, $\sum_{h=1}^{v} a_h R^{(h)}_{i1l_i^{(h)}} = 0$ implies $\sum_{h=1}^{v} a_h^{q^{j-1}} R^{(h)}_{ijl_i^{(h)}} = 0$, and $\sum_{h=1}^{v} a_h R^{(h)}_{ijl_i^{(h)}} = 0$ implies $\sum_{h=1}^{v} a_h^{q^{n_i-j+1}} R^{(h)}_{i1l_i^{(h)}} = 0$.*

Proof. Let $R^{(h)}_{i1l_i^{(h)}} = [r_{1h}, \ldots, r_{mh}]^{\mathrm{T}}$. Using Corollary 7.2.9, r_{kh} is in $GF(q^{n_i})$, for $k = 1, \ldots, m$, and $R^{(h)}_{ijl_i^{(h)}} = [r_{1h}^{q^{j-1}}, \ldots, r_{mh}^{q^{j-1}}]^{\mathrm{T}}$. Suppose that $\sum_{h=1}^{v} a_h R^{(h)}_{i1l_i^{(h)}} = 0$ for some $a_1, \ldots, a_v \in GF(q^{n_i})$. Then

$$\sum_{h=1}^{v} a_h r_{kh} = 0, \ k = 1, \ldots, m.$$

Thus

$$\sum_{h=1}^{v} a_h^{q^{j-1}} r_{kh}^{q^{j-1}} = 0, \ k = 1, \ldots, m,$$

that is, $\sum_{h=1}^{v} a_h^{q^{j-1}} R^{(h)}_{ijl_i^{(h)}} = 0$.

Conversely, suppose that $\sum_{h=1}^{v} a_h R^{(h)}_{ijl_i^{(h)}} = 0$ for some $a_1, \ldots, a_v \in GF(q^{n_i})$. Then

$$\sum_{h=1}^{v} a_h r_{kh}^{q^{j-1}} = 0, \ k = 1, \ldots, m.$$

From $a^{q^{n_i}} = a$ for any $a \in GF(q^{n_i})$, we have

$$\sum_{h=1}^{v} a_h^{q^{n_i-j+1}} r_{kh}^{q^{n_i}} = \sum_{h=1}^{v} a_h^{q^{n_i-j+1}} r_{kh} = 0, \quad k = 1, \ldots, m,$$

that is, $\sum_{h=1}^{v} a_h^{q^{n_i-j+1}} R^{(h)}_{i1l_i^{(h)}} = 0$. □

For any i, $1 \leqslant i \leqslant r$, construct $\mathcal{S}_{ij1}$ from $\mathcal{S}_{ij0}$, $j = 1, \ldots, n_i$, as follows. Whenever elements in $\mathcal{S}_{i10}$ are linearly independent, from Propositions 7.2.2 and 7.2.3, elements in $\mathcal{S}_{ij0}$ are linearly independent; we take $\mathcal{S}_{ij1} = \mathcal{S}_{ij0}$,

$j = 1, \ldots, n_i$. Whenever elements in $\mathcal{S}_{i10}$ are linearly dependent, from Proposition 7.2.2, $R^{(h)}_{i1l_i^{(h)}}$, $h = 1, \ldots, v$ are linearly dependent over $GF(q^{n_i})$; therefore, there exist $a_h \in GF(q^{n_i})$, $h = 1, \ldots, v$ such that $a_h \neq 0$ for some h and $\sum_{h=1}^{v} a_h R^{(h)}_{i1l_i^{(h)}} = 0$. From Proposition 7.2.3, this yields $\sum_{h=1}^{v} a_h^{q^{j-1}} R^{(h)}_{ijl_i^{(h)}} = 0$, $j = 1, \ldots, n_i$. Let $I = \{h \mid a_h \neq 0, h = 1, \ldots, v\}$. Take arbitrarily an integer h' in I with $l_i^{(h')} \leqslant l_i^{(h)}$ for any h in I. Let $R'_{ij}(k) = \sum_{h\in I} a_h^{q^{j-1}} R^{(h)}_{ij}(k)$, $k = 1, \ldots, l_i^{(h')}$. Since $R^{(h)}_{ij}(k) = [R^{(h)}_{ij(l_i^{(h)}+1-k)} \ \ldots \ R^{(h)}_{ijl_i^{(h)}} \ 0 \ \ldots \ 0]$, $k = 1, \ldots, l_i^{(h)}$, we have $R'_{ij}(1) = 0$. Let $\bar{l}_i^{(h')} = 0$ whenever $R'_{ij}(l_i^{(h')}) = 0$, and $\bar{l}_i^{(h')} = \max\, k\{\text{the } k\text{-th column of } R'_{ij}(l_i^{(h')}) \neq 0\}$ otherwise. It is easy to verify that $R^{(h')}_{ij}(k) = R^{(h')}_{ij}(l_i^{(h')}) H_i^{l^{(h')}-k}$, that is, shifting $R^{(h')}_{ij}(l_i^{(h')})$ $l^{(h')} - k$ columns to the left, $k = 1, \ldots, l_i^{(h')}$, where H_i is defined by (7.32). Therefore, $R'_{ij}(k) = 0$ if and only if $k \leqslant l^{(h')} - \bar{l}^{(h')}$. Since $R'_{ij}(1) = 0$, we have $\bar{l}^{(h')} < l^{(h')}$. Let $\mathcal{S}_{ij1}$ be the set obtained from $\mathcal{S}_{ij0}$ by deleting $R^{(h')}_{ij}(k)\Gamma_{ij}(z)$, $k = 1, \ldots, l_i^{(h')}$ and adding $R'_{ij}(k)\Gamma_{ij}(z)$, $k = l^{(h')} - \bar{l}^{(h')} + 1, \ldots, l_i^{(h')}$. Clearly, the space generated by $\mathcal{S}_{ij0}$ and the space generated by $\mathcal{S}_{ij1}$ are the same, $j = 1, \ldots, n_i$. Since $R^{(h)}_{ij}(k)$ can be obtained by taking each element in $R^{(h)}_{i1}(k)$ to the q^{j-1}-th power, $R'_{ij}(k)$ is equal to the matrix obtained by taking each element in $R'_{i1}(k)$ to the q^{j-1}-th power. Thus the fashion of $\mathcal{S}_{ij1}$ is the same as the fashion of $\mathcal{S}_{ij0}$, but the number of elements in $\mathcal{S}_{ij1}$ is less than the number of $\mathcal{S}_{ij0}$. Similarly, from $\mathcal{S}_{ij1}$ we construct $\mathcal{S}_{ij2}$, and so on. We stop the process until some $\mathcal{S}_{ijc}$ in which elements are linearly independent.

Similar to constructing $\mathcal{S}_{ij1}$, from $\mathcal{S}_{00}$ we can construct $\mathcal{S}_{01}$. Repeatedly, we obtain $\mathcal{S}_{02}$, $\mathcal{S}_{03}$, ..., until some $\mathcal{S}_{0c}$ in which elements are linearly independent.

To sum up, by this method we can obtain a basis of $\Phi_{M^*}(z)$.

7.3 Translation and Period

7.3.1 Shift Registers

For any nonnegative integer c, the *c-translation* of an infinite sequence $(a_0, a_1, \ldots)$ means the infinite sequence $(a_c, a_{c+1}, \ldots)$. Correspondingly, $\sum_{i=0}^{\infty} a_{i+c} z^i$ is called the *c-translation* of $\sum_{i=0}^{\infty} a_i z^i$.

Let M be a linear shift register over $GF(q)$. Let $GF(q^*)$ be a splitting field of the second characteristic polynomial of M, and M^* the natural extension of M over $GF(q^*)$.

Theorem 7.3.1. *Let β be the $(\varepsilon_1, \ldots, \varepsilon_r)$ root coordinate of $\Omega(z)$ in $\Phi_{M^*}(z)$. If $\Omega'(z)$ is the c-translation of $\Omega(z)$ and*

$$\begin{aligned}
&\beta'_k = \beta_{c+k},\ k = 0, 1, \dots, l_0 - c - 1,\\
&\beta'_k = 0,\ k = l_0 - c, \dots, l_0 - 1,\\
&\beta'_{ijh} = \sum_{k=h}^{l_i} \tbinom{k-h+c-1}{k-h} \beta_{ijk} \varepsilon_i^{cq^{j-1}},\\
&\qquad i = 1, \dots, r,\ j = 1, \dots, n_i,\ h = 1, \dots, l_i,
\end{aligned} \tag{7.41}$$

then

$$\begin{aligned}
\beta' = [&\beta'_0, \dots, \beta'_{l_0-1},\ \beta'_{111}, \dots, \beta'_{1n_1 1},\ \dots,\ \beta'_{11l_1}, \dots, \beta'_{1n_1 l_1},\\
&\dots\dots,\ \beta'_{r11}, \dots, \beta'_{rn_r 1},\ \dots,\ \beta'_{r1l_r}, \dots, \beta'_{rn_r l_r}]^{\mathrm{T}}
\end{aligned}$$

is the $(\varepsilon_1, \dots, \varepsilon_r)$ *root coordinate of* $\Omega'(z)$.

Proof. Let $\Omega(z) = \sum_{\tau=0}^{\infty} y_\tau z^\tau$ and $\Omega'(z) = \sum_{\tau=0}^{\infty} y'_\tau z^\tau$. From (7.33), we have

$$\begin{aligned}
y_\tau = &\sum_{k=\tau+1}^{l_0} \beta_{l_0+\tau-k} R_{0k} + \sum_{i=1}^{r} \sum_{j=1}^{n_i} \sum_{h=1}^{l_i} \Big(\sum_{k=h}^{l_i} \beta_{ij(l_i+h-k)} R_{ijk} \Big) \tbinom{\tau+h-1}{h-1} \varepsilon_i^{\tau q^{j-1}},\\
&\tau = 0, 1, \dots
\end{aligned}$$

Thus

$$\begin{aligned}
y'_\tau = y_{c+\tau} = &\sum_{k=c+\tau+1}^{l_0} \beta_{l_0+c+\tau-k} R_{0k}\\
&+ \sum_{i=1}^{r} \sum_{j=1}^{n_i} \sum_{k=1}^{l_i} \Big(\sum_{d=k}^{l_i} \beta_{ij(l_i+k-d)} R_{ijd} \Big) \tbinom{c+\tau+k-1}{k-1} \varepsilon_i^{(c+\tau)q^{j-1}},\\
&\tau = 0, 1, \dots
\end{aligned}$$

Using (7.10), we have

$$\begin{aligned}
y'_\tau = &\sum_{k=c+\tau+1}^{l_0} \beta_{l_0+c+\tau-k} R_{0k}\\
&+ \sum_{i=1}^{r} \sum_{j=1}^{n_i} \sum_{k=1}^{l_i} \Big(\sum_{d=k}^{l_i} \beta_{ij(l_i+k-d)} R_{ijd} \Big) \sum_{h=1}^{k} \tbinom{k-h+c-1}{k-h} \tbinom{\tau+h-1}{h-1} \varepsilon_i^{(c+\tau)q^{j-1}}.
\end{aligned}$$

Thus

$$\begin{aligned}
y'_\tau = &\sum_{k=c+\tau+1}^{l_0} \beta_{l_0+c+\tau-k} R_{0k}\\
&+ \sum_{i=1}^{r} \sum_{j=1}^{n_i} \sum_{1 \leqslant h \leqslant k \leqslant d \leqslant l_i} \beta_{ij(l_i+k-d)} R_{ijd} \tbinom{k-h+c-1}{k-h} \tbinom{\tau+h-1}{h-1} \varepsilon_i^{(c+\tau)q^{j-1}}
\end{aligned}$$

$$
\begin{aligned}
&= \sum_{k=c+\tau+1}^{l_0} \beta_{l_0+c+\tau-k} R_{0k} \\
&\quad + \sum_{i=1}^{r}\sum_{j=1}^{n_i}\sum_{h=1}^{l_i}\sum_{d=h}^{l_i}\sum_{k=h}^{d} \binom{k-h+c-1}{k-h} \beta_{ij(l_i+k-d)} \varepsilon_i^{cq^{j-1}} R_{ijd} \binom{\tau+h-1}{h-1} \varepsilon_i^{\tau q^{j-1}} \\
&= \sum_{k=c+\tau+1}^{l_0} \beta_{l_0+c+\tau-k} R_{0k} \\
&\quad + \sum_{i=1}^{r}\sum_{j=1}^{n_i}\sum_{h=1}^{l_i} \Big[\sum_{d=h}^{l_i} \Big(\sum_{k=l_i+h-d}^{l_i} \binom{k-l_i-h+d+c-1}{k-l_i-h+d} \beta_{ijk} \varepsilon_i^{cq^{j-1}} \Big) R_{ijd} \Big] \binom{\tau+h-1}{h-1} \varepsilon_i^{\tau q^{j-1}} \\
&= \sum_{k=\tau+1}^{l_0} \beta'_{l_0+\tau-k} R_{0k} + \sum_{i=1}^{r}\sum_{j=1}^{n_i}\sum_{h=1}^{l_i} \Big[\sum_{d=h}^{l_i} \beta'_{ij(l_i+h-d)} R_{ijd} \Big] \binom{\tau+h-1}{h-1} \varepsilon_i^{\tau q^{j-1}}, \\
&\tau = 0, 1, \ldots
\end{aligned}
$$

Therefore, β' is the $(\varepsilon_1, \ldots, \varepsilon_r)$ root coordinate of $\Omega'(z)$. □

For an $(\varepsilon_1, \ldots, \varepsilon_r)$ root coordinate β, let $l_{ij} = \min h \ [h \geqslant 0, \beta_{ijk} = 0 \ \text{if} \ h < k \leqslant l_i], i = 1, \ldots, r, \ j = 1, \ldots, n_i$. $\max\{ l_{ij}, i = 1, \ldots, r, \ j = 1, \ldots, n_i \}$ is called the *efficient multiplicity* of β. Let $i_1, \ldots, i_{r_1}$ be different elements in $\{i \mid 1 \leqslant i \leqslant r, \exists j (1 \leqslant j \leqslant n_i \ \& \ l_{ij} > 0)\}$. Denote the order of ε_i by e_i, $i = 1, \ldots, r$. The least common multiple of $e_{i_1}, \ldots, e_{i_{r_1}}$ is called the *basic period* of β.

M is said to be *nonsingular*, if its state transition matrix is nonsingular.

Theorem 7.3.2. *Assume that M is nonsingular. Then any $\Omega(z)$ in $\Phi_{M^*}(z)$ is periodic and its period is ep^a, where p is the characteristic of $GF(q)$, e is the basic period of the $(\varepsilon_1, \ldots, \varepsilon_r)$ root coordinate β of $\Omega(z)$, $a = \lceil \log_p l \rceil$ [1] and l is the efficient multiplicity of β.*

Proof. Let $\Omega'(z)$ be the c-translation of $\Omega(z)$, i.e., $\Omega'(z) = D^c(\Omega(z))$. Let β and β' be the $(\varepsilon_1, \ldots, \varepsilon_r)$ root coordinates of $\Omega(z)$ and $\Omega'(z)$, respectively. From Theorem 7.3.1, noticing $l_0 = 0$, we have

$$
\begin{aligned}
\beta'_{ijh} &= \sum_{k=h}^{l_i} \binom{k-h+c-1}{k-h} \beta_{ijk} \varepsilon_i^{cq^{j-1}}, \qquad (7.42) \\
&\quad i = 1, \ldots, r, \ j = 1, \ldots, n_i, \ h = 1, \ldots, l_i.
\end{aligned}
$$

Without loss of generality, assume that there exists j such that $l_{ij} > 0$ whenever $1 \leqslant i \leqslant r_1$ and that $l_{ij} = 0$ whenever $r_1 < i \leqslant r$. Then

[1] $\lceil x \rceil$ stands for the minimal integer $\geqslant x$.

$$\beta'_{ijh} = 0,\ i = 1, \dots, r_1,\ j = 1, \dots, n_i,\ h = l_{ij} + 1, \dots, l_i,$$
$$\beta'_{ijh} = 0,\ i = r_1 + 1, \dots, r,\ j = 1, \dots, n_i,\ h = 1, \dots, l_i.$$

From (7.42), $\Omega'(z) = \Omega(z)$ is equivalent to the equations

$$\beta_{ijh} = \sum_{k=h}^{l_{ij}} \tbinom{k-h+c-1}{k-h} \beta_{ijk} \varepsilon_i^{cq^{j-1}}, \tag{7.43}$$
$$i = 1, \dots, r_1,\ j = 1, \dots, n_i,\ h = 1, \dots, l_{ij}.$$

Equations for $h = l_{ij}$ in (7.43) are

$$\beta_{ijl_{ij}} = \beta_{ijl_{ij}} \varepsilon_i^{cq^{j-1}},\ i = 1, \dots, r_1,\ j = 1, \dots, n_i. \tag{7.44}$$

Since $\beta_{ijl_{ij}} \neq 0$ whenever $l_{ij} > 0$, (7.44) is equivalent to the equations

$$\varepsilon_i^c = 1,\ i = 1, \dots, r_1.$$

Since $\varepsilon_i^c = 1$ if and only if $e_i | c$, this yields that (7.44) is equivalent to $e|c$. Thus (7.43) is equivalent to the equations

$$e|c,$$
$$\sum_{k=h+1}^{l_{ij}} \tbinom{k-h+c-1}{k-h} \beta_{ijk} = 0,$$
$$i = 1, \dots, r_1,\ j = 1, \dots, n_i,\ h = 1, \dots, l_{ij} - 1.$$

It is equivalent to the equations

$$e|c,$$
$$\tbinom{l_{ij}-h+c-1}{l_{ij}-h} = -\beta_{ijl_{ij}}^{-1} \sum_{k=h+1}^{l_{ij}-1} \tbinom{k-h+c-1}{k-h} \beta_{ijk},$$
$$i = 1, \dots, r_1,\ j = 1, \dots, n_i,\ h = 1, \dots, l_{ij} - 1,$$

that is,

$$e|c,$$
$$\tbinom{h+c-1}{h} = -\beta_{ijl_{ij}}^{-1} \sum_{k=l_{ij}-h+1}^{l_{ij}-1} \tbinom{k-l_{ij}+h+c-1}{k-l_{ij}+h} \beta_{ijk},$$
$$i = 1, \dots, r_1,\ j = 1, \dots, n_i,\ h = 1, \dots, l_{ij} - 1.$$

Clearly, $\binom{h+c-1}{h} = -\beta_{ijl_{ij}}^{-1} \sum_{k=l_{ij}-h+1}^{l_{ij}-1} \binom{k-l_{ij}+h+c-1}{k-l_{ij}+h} \beta_{ijk}$, $h = 1, \dots, l_{ij} - 1$ if and only if $\binom{h+c-1}{h} = 0 (\mathrm{mod}\ p)$, $h = 1, \dots, l_{ij} - 1$. Thus (7.43) is equivalent to the equations

$$
\begin{aligned}
&e|c,\\
&\tbinom{h+c-1}{h} = 0 \pmod p, \quad h = 1, \ldots, l-1.
\end{aligned} \tag{7.45}
$$

Let

$$c = c_0 + we,\ 0 \leqslant c_0 < e.$$

Then (7.45) is equivalent to the equations

$$
\begin{aligned}
&c_0 = 0,\\
&\tbinom{h+c_0-1+we}{h} = 0 \pmod p, \quad h = 1, \ldots, l-1.
\end{aligned} \tag{7.46}
$$

Since $\gcd(e, p) = 1$, $e^{-1} \pmod p$ exists. From (7.16), we have

$$
\begin{aligned}
\tbinom{h+c_0-1+we}{h} = \sum_{k=0}^{h} \Big[\sum_{i=0}^{k} (-1)^{k-i} \tbinom{k}{i} \tbinom{h+c_0-1+ie}{h} \Big] \tbinom{w}{k},\\
h = 1, \ldots, l-1.
\end{aligned}
$$

Thus (7.46) is equivalent to the equations

$$
\begin{aligned}
&c_0 = 0,\\
&\sum_{k=0}^{h} \Big[\sum_{i=0}^{k} (-1)^{k-i} \tbinom{k}{i} \tbinom{h+c_0-1+ie}{h} \Big] \tbinom{w}{k} = 0 \pmod p,\\
&\quad h = 1, \ldots, l-1.
\end{aligned} \tag{7.47}
$$

Since the coefficient of the term $\binom{w}{h}$ in (7.47) is e^h, (7.47) is equivalent to the equations

$$
\begin{aligned}
&c_0 = 0,\\
&\tbinom{w}{h} = -e^{-h} \sum_{k=0}^{h-1} \Big[\sum_{i=0}^{k} (-1)^{k-i} \tbinom{k}{i} \tbinom{h-1+ie}{h} \Big] \tbinom{w}{k} \pmod p,\\
&\quad h = 1, \ldots, l-1.
\end{aligned} \tag{7.48}
$$

It is easy to prove by induction on h that (7.48) is equivalent to the equations

$$
\begin{aligned}
&c_0 = 0,\\
&\tbinom{w}{h} = q_h \pmod p, \quad h = 1, \ldots, l-1,
\end{aligned} \tag{7.49}
$$

where $0 \leqslant q_1, \ldots, q_{l-1} < p$, and

$$
\begin{aligned}
&q_0 = 1,\\
&q_h = -e^{-h} \sum_{k=0}^{h-1} \Big[\sum_{i=0}^{k} (-1)^{k-i} \tbinom{k}{i} \tbinom{h-1+ie}{h} \Big] q_k \pmod p,\\
&\quad h = 1, \ldots, l-1.
\end{aligned}
$$

Therefore, (7.43) is equivalent to (7.49). Let

$$w = \sum_{i=1}^{a} w_i p^{i-1} + w' p^a,$$
$$0 \leqslant w_1, \ldots, w_a < p, \quad w' \geqslant 0.$$

From Theorem 7.1.3, since $h < l \leqslant p^a$, (7.49) is equivalent to the system of equations $c_0 = 0$ and

$$\binom{\sum_{i=1}^{a} w_i p^{i-1}}{h} = q_h \pmod{p}, \quad h = 1, \ldots, l-1. \tag{7.50}$$

From Theorem 7.1.3, we have $w_i = \binom{\sum_{j=1}^{a} w_j p^{j-1}}{p^{i-1}} (\bmod\ p)$, $i = 1, \ldots, a$. Thus (7.50) implies

$$w_i = q_{p^{i-1}}, \ i = 1, \ldots, a. \tag{7.51}$$

Conversely, (7.51) implies (7.50). In fact, since $\Omega(z)$ is periodic, we may take a positive integer c such that $D^c(\Omega(z)) = \Omega(z)$. Then (7.50) holds for such a c. It follows that (7.51) holds for such a c. From (7.50) and (7.51), we have

$$\binom{\sum_{i=1}^{a} q_{p^{i-1}} p^{i-1}}{h} = q_h \pmod{p}, \quad h = 1, \ldots, l-1, \tag{7.52}$$

which is independent of c. Using (7.52), (7.51) implies (7.50). Thus (7.50) and (7.51) are equivalent. Clearly, (7.51) is equivalent to the equation

$$c = c_0 + e \sum_{i=1}^{a} q_{p^{i-1}} p^{i-1} + w' e p^a.$$

Therefore, (7.43) is equivalent to the equations

$$c = e \sum_{i=1}^{a} q_{p^{i-1}} p^{i-1} + w' e p^a. \tag{7.53}$$

We conclude that $D^c(\Omega(z)) = \Omega(z)$ if and only if (7.53) holds for some nonnegative integer w'.

Since $\Omega(z)$ is periodic, there exists a positive integer c_1 such that $D^{c_1}(\Omega(z)) = \Omega(z)$. It follows that there exists w_1' such that $c_1 = e \sum_{i=1}^{a} q_{p^{i-1}} p^{i-1} + w_1' e p^a$. Therefore, for any nonnegative integer c, $D^c(\Omega(z)) = \Omega(z)$ if and only if $c = c_1 \pmod{ep^a}$. From $D^0(\Omega(z)) = \Omega(z)$, it follows that $0 = c_1 \pmod{ep^a}$. Thus for any nonnegative integer c, $D^c(\Omega(z)) = \Omega(z)$ if and only if $c = 0 \pmod{ep^a}$. This yields that the period of $\Omega(z)$ is ep^a. □

From the definitions, whenever $\beta = 0$, the efficient multiplicity l of β is 0 and the basic period e of β is 1; whenever $\beta \neq 0$,

$$l = \max k\,(\exists i\,\exists j\,(1 \leqslant i \leqslant r\ \&\ 1 \leqslant j \leqslant n_i\ \&\ 1 \leqslant k \leqslant l_i\ \&\ b_{ijk} \neq 0)), \quad (7.54)$$

e is the least common multiple of e_{i_1}, ..., $e_{i_{r_1}}$, where e_{i_j} is the order of ε_{i_j}, $j = 1, \ldots, r_1$,

$$\{i_1, \ldots, i_{r_1}\} = \{i \mid 1 \leqslant i \leqslant r, \exists j\,\exists k\,(1 \leqslant j \leqslant n_i\ \&\ 1 \leqslant k \leqslant l_i\ \&\ b_{ijk} \neq 0)\}.$$

Corollary 7.3.1. *Assume that the state transition matrix of M is nonsingular. Let β_i and u_i be the $(\varepsilon_1, \ldots, \varepsilon_r)$ root coordinate and the period of $\Omega_i(z) \in \Phi_{M^*}(z)$, respectively, $i = 1, \ldots, h$. If the number of nonzero components of β_1, ..., β_h at each position is at most one, then the period of $\Omega_1(z) + \cdots + \Omega_h(z)$ is the least common multiple of u_1, ..., u_h.*

If the maximum period of the sequences in $\Phi_M(z)$ is h and the number of the sequences in $\Phi_M(z)$ with period i is c_i, $i = 1, \ldots, h$, $\sum_{i=1}^{h} c_i z^i$ is called the *period distribution polynomial* of $\Phi_M(z)$, denoted by $\xi_M(z)$. For any subset of $\Phi_M(z)$, we define similarly its period distribution polynomial. We define an addition operation on period distribution polynomials similar to common polynomials, and define $*$ operation:

$$\Big(\sum_{i=1}^{k} c_i z^i\Big) * \Big(\sum_{j=1}^{h} d_j z^j\Big) = \sum_{i=1}^{k}\sum_{j=1}^{h} c_i d_j z^{\mathrm{lcm}(i,j)}. \quad (7.55)$$

Take an $(\varepsilon_1, \ldots, \varepsilon_r)$ root basis of $\Phi_{M^*}(z)$. Let β be the root coordinate of $\Omega(z) \in \Phi_M(z)$. If components of β are zero but β_{ijk}, $j = 1$, ..., n_i for some i and k, then the period of β is $e_i p^{\lceil \log_p k \rceil}$, where e_i is the order of ε_i. Clearly, given i and k, $1 \leqslant i \leqslant r$, $1 \leqslant k \leqslant l_i$, the number of such β_{ijk} is $q^{n_i} - 1$. Consider the subset of $\Phi_M(z)$ in which in components of the root coordinate β of any sequence, $\beta_{i'jk'} = 0$ holds whenever $i' \neq i$ or $k' \neq k$. Then the subset's period distribution polynomial is $z + (q^{n_i} - 1)z^{e_i p^{\lceil \log_p k \rceil}}$. From Corollary 7.3.1, we have

$$\xi_M(z) = \prod_{i=1}^{r}\prod_{k=1}^{l_i}(z + (q^{n_i} - 1)z^{e_i p^{\lceil \log_p k \rceil}}), \quad (7.56)$$

where $\prod_{i=1}^{1} \varphi(i) = \varphi(1)$, $\prod_{i=1}^{n+1} \varphi(i) = (\prod_{i=1}^{n} \varphi(i)) * \varphi(n+1)$. Since for any $u \geqslant 1$, any c and any d, $(z + (c-1)z^u) * (z + (d-1)z^u) = z + (cd-1)z^u$ holds, we have

$$\prod_{k=p^{h-1}+1}^{p^{h-1}+t} (z + (q^{n_i} - 1)z^{e_i p^h}) = z + (q^{n_i t} - 1)z^{e_i p^h}. \quad (7.57)$$

Since $p^{\lceil \log_p k \rceil} = p^h$ for $k = p^{h-1} + 1$, ..., p^h, using (7.56) and (7.57), we have

$$\xi_M(z) = \prod_{i=1}^{r}[(z + (q^{n_i} - 1)z^{e_i}) * \prod_{h=1}^{\lceil \log_p l_i \rceil - 1} (z + (q^{n_i(p^h - p^{h-1})} - 1)z^{e_i p^h}) * (z + (q^{n_i(l_i - p^{\lceil \log_p l_i \rceil - 1})} - 1)z^{e_i p^{\lceil \log_p l_i \rceil}})]. \tag{7.58}$$

It is easy to prove by induction on k that

$$(z + (q^{n_i} - 1)z^{e_i}) * \prod_{h=1}^{k}(z + (q^{n_i(p^h - p^{h-1})} - 1)z^{e_i p^h}) \tag{7.59}$$
$$= z + (q^{n_i} - 1)z^{e_i} + \sum_{h=1}^{k}(q^{n_i p^h} - q^{n_i p^{h-1}})z^{e_i p^h}.$$

Using (7.58) and (7.59), it is easy to show that

$$\xi_M(z) = \prod_{i=1}^{r}\big[z + (q^{n_i} - 1)z^{e_i} + \sum_{h=1}^{\lceil \log_p l_i \rceil - 1} (q^{n_i p^h} - q^{n_i p^{h-1}})z^{e_i p^h} + (q^{n_i l_i} - q^{n_i p^{\lceil \log_p l_i \rceil - 1}})z^{e_i p^{\lceil \log_p l_i \rceil}}\big]. \tag{7.60}$$

Let $f_i(z)$ be the minimal polynomial of ε_i, $i = 1, \ldots, r$. It is easy to show that $f_1(z)$, ..., $f_r(z)$ are coprime with each other of which degrees are n_1, ..., n_r, respectively, and that the second characteristic polynomial of M is $\prod_{i=1}^{r} f_i(z)^{l_i}$. Notice that the period of $f_i(z)$, i.e., $\min c[c > 0, f_i(z)|(z^c - 1)]$, equals the order of ε_i. We then obtain the following theorem.

Theorem 7.3.3. *Assume that M is nonsingular. Let $\prod_{i=1}^{r} f_i(z)^{l_i}$ be the second characteristic polynomial of M, where $f_1(z)$, ..., $f_r(z)$ are irreducible polynomials over $GF(q)$ and coprime with each other, and l_1, ..., l_r are positive integers. Let n_i and e_i be the degree and the period of $f_i(z)$, respectively, $i = 1, \ldots, r$. Then the period distribution polynomial of M is given by* (7.56) *or* (7.60).

7.3.2 Finite Automata

For any linear autonomous finite automaton M, From Theorems 1.3.4 and 1.3.5, we can find linear autonomous registers $M^{(1)}$, ..., $M^{(v)}$ such that the union of $M^{(1)}$, ..., $M^{(v)}$ is minimal and equivalent to M. It follows that $\Phi_M(z)$ equals the direct sum of $\Phi_{M^{(1)}}(z)$, ..., $\Phi_{M^{(v)}}(z)$.

Let $GF(q^*)$ be a splitting field of the second characteristic polynomial of $M^{(h)}$, $h = 1, \ldots, v$. Let M^* be the natural extension of M over $GF(q^*)$, and $M^{(h)*}$ the natural extension of $M^{(h)}$ over $GF(q^*)$, $h = 1, \ldots, v$. Clearly, the union of $M^{(1)*}$, ..., $M^{(v)*}$ is minimal and equivalent to M^*. It follows that $\Phi_{M^*}(z)$ equals the direct sum of $\Phi_{M^{(1)*}}(z)$, ..., $\Phi_{M^{(v)*}}(z)$.

Let $\eta_1^{(h)}, \ldots, \eta_{n^{(h)}}^{(h)}$ be a basis of $\Phi_{M^{(h)*}}(z)$, $h = 1, \ldots, v$. Then $\eta_1^{(h)}, \ldots, \eta_{n^{(h)}}^{(h)}$, $h = 1, \ldots, v$ together form a basis of $\Phi_{M^*}(z)$. For any $\Omega(z) \in \Phi_{M^*}(z)$, there uniquely exist $\Omega^{(h)}(z) \in \Phi_{M^{(h)*}}(z)$, $h = 1, \ldots, v$ such that $\Omega(z) = \Omega^{(1)}(z) + \cdots + \Omega^{(v)}(z)$. Thus $D^c(\Omega(z)) = D^c(\Omega^{(1)}(z)) + \cdots + D^c(\Omega^{(v)}(z))$. Let β be the coordinate of $\Omega(z)$, and $\beta^{(h)}$ the coordinate of $\Omega^{(h)}(z)$, $h = 1, \ldots, v$. Then we have $\beta = [\beta^{(1)}, \ldots, \beta^{(v)}]^{\mathrm{T}}$.

It is easy to show that $D^c(\Omega(z)) = \Omega(z)$ if and only if $D^c(\Omega^{(h)}(z)) = \Omega^{(h)}(z)$, $h = 1, \ldots, v$. From the proof of Theorem 7.3.2, $D^c(\Omega(z)) = \Omega(z)$ if and only if

$$c = 0 \pmod{e^{(h)} p^{a^{(h)}}}, \ h = 1, \ldots, v, \tag{7.61}$$

where $e^{(h)}$ and $l^{(h)}$ are the basic period and the efficient multiplicity of the $(\varepsilon_1^{(h)}, \ldots, \varepsilon_{r^{(h)}}^{(h)})$ root coordinate of $\Omega^{(h)}(z)$, respectively, $h = 1, \ldots, v$. Let e be the least common multiple of $e_1, \ldots, e_v$, and $a = \max(a^{(1)}, \ldots, a^{(v)})$. Then the least common multiple of $e^{(1)} p^{a^{(1)}}, \ldots, e^{(v)} p^{a^{(v)}}$ is ep^a. Thus (7.61) is equivalent to the equation

$$c = 0 \pmod{ep^a}. \tag{7.62}$$

It follows that $D^c(\Omega(z)) = \Omega(z)$ if and only if (7.62) holds. We obtain the following Theorem.

Theorem 7.3.4. *Assume that M is a nonsingular linear autonomous finite automaton and that the union of linear registers $M^{(1)}, \ldots, M^{(v)}$ is minimal and equivalent to M. Let $\Omega(z) = \Omega^{(1)}(z) + \cdots + \Omega^{(v)}(z)$, $\Omega^{(h)}(z) \in \Phi_{M^{(h)*}}(z)$, $h = 1, \ldots, v$. Let e be the least common multiple of $e_1, \ldots, e_v$, and $a = \max(a^{(1)}, \ldots, a^{(v)})$, where $e^{(h)}$ and $l^{(h)}$ are the basic period and the efficient multiplicity of the $(\varepsilon_1^{(h)}, \ldots, \varepsilon_{r^{(h)}}^{(h)})$ root coordinate of $\Omega^{(h)}(z)$, respectively, $h = 1, \ldots, v$. Then $D^c(\Omega(z)) = \Omega(z)$ if and only if $c = 0 \pmod{ep^a}$. Therefore, the period of $\Omega(z)$ is ep^a.*

Using Theorem 7.3.2 and Theorem 7.3.4, we obtain the following.

Corollary 7.3.2. *Assume that M is a nonsingular linear autonomous finite automaton and that the union of linear registers $M^{(1)}, \ldots, M^{(v)}$ is minimal and equivalent to M. Let $\Omega(z) = \Omega^{(1)}(z) + \cdots + \Omega^{(v)}(z)$, $\Omega^{(h)}(z) \in \Phi_{M^{(h)*}}(z)$, $h = 1, \ldots, v$. Let u_h be the period of $\Omega^{(h)}(z)$, $h = 1, \ldots, v$. Then the period of $\Omega(z)$ is the least common multiple of $u_1, \ldots, u_v$.*

From Corollary 7.3.2, it is easy to show the following.

Corollary 7.3.3. *Assume that M is a nonsingular linear autonomous finite automaton and that the union of linear registers $M^{(1)}, \ldots, M^{(v)}$ is minimal and equivalent to M. Then we have*

$$\xi_M(z) = \prod_{i=1}^{v} \xi_{M^{(i)}}(z). \tag{7.63}$$

Since the union of linear registers $M^{(1)}$, ..., $M^{(v)}$ is minimal, $M^{(h)}$ is minimal, $h = 1, \ldots, v$. Therefore, the characteristic polynomial and the second characteristic polynomial of $M^{(h)}$ are the same, say $f^{(h)}(z)$, $h = 1, \ldots, v$. Clearly, we can take $f^{(h)}(z)$, $h = 1, \ldots, v$ as the elementary divisors of the state transition matrix of the union of $M^{(1)}$, ..., $M^{(v)}$. Since elementary divisors of the state transition matrix of a linear finite automaton keep unchanged under similarity transformation, $f^{(h)}(z)$, $h = 1, \ldots, v$ are the elementary divisors of the state transition matrix of any minimal linear finite automaton of M. From Theorem 7.3.3 and Corollary 7.3.3, noticing that any elementary divisor is a positive power of a irreducible polynomial, we have the following theorem.

Theorem 7.3.5. *Assume that M is a nonsingular linear autonomous finite automaton and that $f^{(i)}(z)$, $i = 1, \ldots, v$ are the elementary divisors of the state transition matrix of any minimal linear finite automaton of M. Let $f^{(i)}(z) = f_i(z)^{l_i}$, where $f_i(z)$ is an irreducible polynomial over $GF(q)$ of degree n_i and with period e_i, $i = 1, \ldots, v$. Then we have*

$$\begin{aligned}\xi_M(z) &= \prod_{i=1}^{v} \prod_{k=1}^{l_i} (z + (q^{n_i} - 1) z^{e_i p^{\lceil \log_p k \rceil}}) \\ &= \prod_{i=1}^{v} \Big[z + (q^{n_i} - 1) z^{e_i} + \sum_{h=1}^{\lceil \log_p l_i \rceil - 1} (q^{n_i p^h} - q^{n_i p^{h-1}}) z^{e_i p^h} \\ &\qquad + (q^{n_i l_i} - q^{n_i p^{\lceil \log_p l_i \rceil - 1}}) z^{e_i p^{\lceil \log_p l_i \rceil}} \Big].\end{aligned}$$

7.4 Linearization

We use $\Re$ to denote the set of all linear shift registers over $GF(q)$ of which the output is the first component of the state. It is evident that any linear shift register in $\Re$ is uniquely determined by its characteristic polynomial. We define two operations on $\Re$.

Let $M_i \in \Re$, $i = 1, \ldots, h$. For any i, $1 \leqslant i \leqslant h$, let $f_i(z)$ be the characteristic polynomial of M_i and $G(M_i)$, or G_i for short, the set of all different roots of $f_i(z)$; let $l_i(\varepsilon)$ be the multiplicity of ε whenever ε is a root of $f_i(z)$, $l_i(\varepsilon) = 0$ otherwise. Let

$$f_\Sigma(z) = \prod_{\varepsilon \in G_1 \cup \cdots \cup G_h} (z - \varepsilon)^{\max(l_1(\varepsilon), \ldots, l_h(\varepsilon))}. \tag{7.64}$$

It is easy to show that $f_\Sigma(z)$ is the least common multiple of $f_1(z), \ldots, f_h(z)$ with leading coefficient 1. The linear shift register in $\Re$ with characteristic polynomial $f_\Sigma(z)$ is called the *sum* of $M_1, \ldots, M_h$, denoted by $M_1 + \cdots + M_h$ or $\sum_{i=1}^{h} M_i$. Clearly, the sum is independent of the order of $M_1, \ldots, M_h$ and we have

$$\begin{aligned} &(M_1 + \cdots + M_h) + (M_{h+1} + \cdots + M_k) = M_1 + \cdots + M_k, \\ &M + \cdots + M = M. \end{aligned} \tag{7.65}$$

This yields

$$M_1 + (M_1 + \cdots + M_h) = M_1 + \cdots + M_h. \tag{7.66}$$

We use p to denote the characteristic of $GF(q)$. Let

$$\begin{aligned} &Q = Q(M_1, \ldots, M_h) = \Big\{ \prod_{i=1}^{h} \varepsilon_i \mid \varepsilon_i \in G_i,\ i = 1, \ldots, h \Big\}, \\ &l(0) = \max(l_1(0), \ldots, l_h(0)), \\ &\quad v(k) = \min\ v(v \geqslant 0\ \&\ k \leqslant p^v),\ k = 1, 2, \ldots, \\ &\quad v(k_1, \ldots, k_h) = \max(v(k_1), \ldots, v(k_h)), \\ &\quad l(k_1, \ldots, k_h) = \min(k_1 + \cdots + k_h - h + 1, p^{v(k_1, \ldots, k_h)}), \\ &\qquad k_1, \ldots, k_h = 1, 2, \ldots, \\ &l(\varepsilon) = \max \big\{ l(l_1(\varepsilon_1), \ldots, l_h(\varepsilon_h)) \mid \varepsilon = \prod_{i=1}^{h} \varepsilon_i, \varepsilon_i \in G_i, i = 1, \ldots, h \big\}, \\ &\quad \varepsilon \in Q \setminus \{0\}. \end{aligned} \tag{7.67}$$

Clearly, whenever $f_i(z)$ has no nonzero repeated root for any i, $1 \leqslant i \leqslant h$, $l(\varepsilon) = 1$ if $0 \neq \varepsilon \in Q$. It is easy to see that $\varepsilon \in Q$ implies $\varepsilon^q \in Q$ and $l(\varepsilon) = l(\varepsilon^q)$. Let

$$f_\Pi(z) = \prod_{\varepsilon \in Q} (z - \varepsilon)^{l(\varepsilon)}. \tag{7.68}$$

It is easy to show that $f_\Pi(z)$ is a polynomial over $GF(q)$. The linear shift register in $\Re$ with characteristic polynomial $f_\Pi(z)$ is called the *product* of $M_1, \ldots, M_h$, denoted by $M_1 \ldots M_h$ or $\prod_{i=1}^{h} M_i$. Clearly, the product is independent of the order of $M_1, \ldots, M_h$. It is easy to verify that

$$M(M_1 + \cdots + M_h) = MM_1 + \cdots + MM_h. \tag{7.69}$$

We use M_E to denote the linear shift register in $\Re$ with characteristic polynomial $z - 1$. Then we have

$$MM_E = M_E M = M. \tag{7.70}$$

Notice that for any M_1, M_2 in $\Re$, $M_1 \prec M_2$ if and only if $\Psi^{(1)}_{M_1}(z) \subseteq \Psi^{(1)(z)}_{M_2}$, if and only if $f_1(z)|f_2(z)$. Thus $M_1 \prec M_2$ if and only if $M_1 + M_2 = M_2$. From (7.65) and (7.69), we have $M_1 + M_3 \prec M_2 + M_3$ and $M_1 M_3 \prec M_2 M_3$, if $M_1 \prec M_2$.

Point out that the associative law does not hold.

Theorem 7.4.1. *For any M_1, M_2 in $\Re$, we have*

$$\Psi^{(1)}_{M_1+M_2}(z) = \Psi^{(1)}_{M_1}(z) + \Psi^{(1)}_{M_1}(z), \tag{7.71}$$

therefore, $M_1 + M_2$ is equivalent to the union of M_1 and M_2.

Proof. Let $f(z)$ be the characteristic polynomial of M_1+M_2, and $f_i(z)$ the characteristic polynomial of M_i, $i = 1, 2$. Let $g(z)$ be the reverse polynomial of $f(z)$, and $g_i(z)$ the reverse polynomial of $f_i(z)$, $i = 1, 2$. It is easy to prove that for any polynomials φ and ψ, $\psi(z)|\varphi(z)$ implies $\bar{\psi}(z)|\bar{\varphi}(z)$, where $\bar{\psi}(z)$ and $\bar{\varphi}(z)$ are the reverse polynomials of $\psi(z)$ and $\varphi(z)$, respectively. Using the result, it is easy to show that $g(z)$ is the least common multiple of $g_1(z)$ and $g_2(z)$.

Any $\Omega(z) \in \Psi^{(1)}_{M_1+M_2}(z)$, $\Omega(z)$ can be expressed as the sum of a polynomial $b_0(z)$ and a proper fraction $h(z)/g(z)$, where the degree of $b_0(z)$ is less than the multiplicity of the divisor z of $f(z)$. Since $g(z)$ is the least common multiple of $g_1(z)$ and $g_2(z)$, $h(z)/g(z)$ can be decomposed into a sum of proper fractions $h_1(z)/g_1(z)$ and $h_2(z)/g_2(z)$. Since the multiplicity of the divisor z of $f(z)$ equals the multiplicity of the divisor z of $f_1(z)$ or of $f_2(z)$, we have $b_0(z)+h_1(z)/g_1(z) + h_2(z)/g_2(z) \in \Psi^{(1)}_{M_1}(z) + \Psi^{(1)}_{M_2}(z)$. Thus $\Psi^{(1)}_{M_1+M_2}(z) \subseteq \Psi^{(1)}_{M_1}(z) + \Psi^{(1)}_{M_2}(z)$. On the other hand, let $\Omega_i(z) \in \Psi^{(1)}_{M_i}(z)$, $i = 1, 2$. Then $\Omega_i(z)$ can be expressed as the sum of a polynomial $b_i(z)$ and a proper fraction $h_i(z)/g_i(z)$, where the degree of $b_i(z)$ is less than the multiplicity of the divisor z of $f_i(z)$, $i = 1, 2$. Thus $\Omega_1(z) + \Omega_2(z) = b_1(z) + b_2(z) + h_1(z)/g_1(z) + h_2(z)/g_2(z) = b_1(z) + b_2(z) + h(z)/g(z)$, where $h(z)/g(z)$ is a proper fraction. Since the degree of $b_1(z)+b_2(z)$ is less than the multiplicity of the divisor z of $f(z)$, we have $\Omega_1(z)+\Omega_2(z) \in \Psi^{(1)}_{M_1+M_2}(z)$. Therefore, $\Psi^{(1)}_{M_1}(z) + \Psi^{(1)}_{M_2}(z) \subseteq \Psi^{(1)}_{M_1+M_2}(z)$. We conclude $\Psi^{(1)}_{M_1+M_2}(z) = \Psi^{(1)}_{M_1}(z) + \Psi^{(1)}_{M_2}(z)$. □

Let $\Omega_j(z) = \sum_{i=0}^{\infty} \omega_{ji} z^i$, $j = 1, \ldots, n$. $\sum_{i=0}^{\infty} (\omega_{1i} \ldots \omega_{ri}) z^i$ is called the product of $\Omega_1(z), \ldots, \Omega_r(z)$, denoted by $\Omega_1(z) \ldots \Omega_r(z)$.

Theorem 7.4.2. *Assume that M, $M^{(1)}$, …, $M^{(h)} \in \Re$ and $M^{(1)} \ldots M^{(h)} \prec M$. Let $\beta^{(a)}$ be the $(\varepsilon_1^{(a)}, \ldots, \varepsilon_{r^{(a)}}^{(a)})$ root coordinate of $\Omega_a(z)$ in $\Psi^{(1)}_{M^{(a)}}(z)$, $a = 1, \ldots, h$. Then $\Omega_1(z) \ldots \Omega_h(z)$ is in $\Psi^{(1)}_M(z)$ and the $(\varepsilon_1, \ldots, \varepsilon_r)$ root coordinate β of $\Omega_1(z) \ldots \Omega_h(z)$ is determined by*

$$\beta_k = \prod_{a=1}^{h} s_k^{(a)} - \prod_{a=1}^{h} (s_k^{(a)} - \beta_k^{(a)}),\ k = 0, \dots, l_0 - 1,$$

$$\beta_{ijk} = \psi(i, j, k, \beta^{(1)}, \dots, \beta^{(h)}) = \sum_{(i_1, j_1, \dots, i_h, j_h) \in P_{ij}} \varphi(i_1, j_1, \dots, i_h, j_h, k),$$

$$i = 1, \dots, r,\ j = 1, \dots, n_i,\ k = 1, \dots, l_i, \tag{7.72}$$

where $s_k^{(a)}$ *is the coefficient of* z^k *in* $\Omega_a(z)$, $k = 0, 1, \dots$, $\beta_k^{(a)} = 0$ *in the case of* $k \geqslant l_0^{(a)}$, $a = 1, \dots, h$,

$$P_{ij} = \big\{(i_1, j_1, \dots, i_h, j_h) \mid \prod_{a=1}^{h} (\varepsilon_{i_a}^{(a)})^{q^{j_a - 1}} = \varepsilon_i^{q^{j-1}},\ i_a = 1, \dots, r^{(a)},$$

$$j_a = 1, \dots, n_{i_a}^{(a)},\ a = 1, \dots, h\big\},$$

$$i = 1, \dots, r,\ j = 1, \dots, n_i,$$

$$\varphi(i_1, j_1, \dots, i_h, j_h, k) \tag{7.73}$$

$$= \sum_{k_1=1}^{l_{i_1}^{(1)}} \cdots \sum_{k_{h-1}=1}^{l_{i_{h-1}}^{(h-1)}} \sum_{k_h = \max(1, k - k_1 - \cdots - k_{h-1} + h - 1)}^{l_{i_h}^{(h)}} d(k-1, k_1 - 1, \dots, k_h - 1) \prod_{a=1}^{h} \beta_{i_a j_a k_a}^{(a)},$$

$$i_a = 1, \dots, r^{(a)},\ j_a = 1, \dots, n_{i_a}^{(a)},\ a = 1, \dots, h,\ k = 1, 2, \dots,$$

$$d(k-1, k_1 - 1, \dots, k_h - 1) = \sum_{c=0}^{k-1} (-1)^c \tbinom{k-1}{c} \prod_{a=1}^{h} \tbinom{k_a - 2 - c}{k_a - 1},$$

$$k, k_1, \dots, k_h = 1, 2, \dots$$

Proof. Let $\Omega(z) = \Omega_1(z) \dots \Omega_h(z) = \sum_{i=0}^{\infty} s_i z^i$. For any a, $1 \leqslant a \leqslant h$, since the $(\varepsilon_1^{(a)}, \dots, \varepsilon_{r^{(a)}}^{(a)})$ root coordinate of $\Omega_a(z)$ is $\beta^{(a)}$, we have

$$s_\tau^{(a)} = \beta_\tau^{(a)} + \sum_{i_a=1}^{r^{(a)}} \sum_{j_a=1}^{n_{i_a}^{(a)}} \sum_{k_a=1}^{l_{i_a}^{(a)}} \beta_{i_a j_a k_a}^{(a)} \tbinom{\tau + k_a - 1}{k_a - 1} (\varepsilon_{i_a}^{(a)})^{\tau q^{j_a - 1}},\quad \tau = 0, 1, \dots,$$

where $\beta_\tau^{(a)} = 0$ in the case of $\tau \geqslant l_0^{(a)}$. Noticing $l_0^{(a)} \leqslant l_0$ for $a = 1, \dots, h$, it follows that

$$s_\tau = \prod_{a=1}^{h} s_\tau^{(a)}$$

$$= \beta_\tau + \sum_{i_1=1}^{r^{(1)}} \cdots \sum_{i_h=1}^{r^{(h)}} \sum_{j_1=1}^{n_{i_1}^{(1)}} \cdots \sum_{j_h=1}^{n_{i_h}^{(h)}} \sum_{k_1=1}^{l_{i_1}^{(1)}} \cdots \sum_{k_h=1}^{l_{i_h}^{(h)}} \prod_{a=1}^{h} \beta_{i_a j_a k_a}^{(a)}$$

$$\prod_{a=1}^{h} \tbinom{\tau + k_a - 1}{k_a - 1} \Big(\prod_{a=1}^{h} (\varepsilon_{i_a}^{(a)})^{q^{j_a - 1}}\Big)^{\tau},$$

$$\tau = 0, 1, \dots,$$

where $\beta_\tau = 0$ in the case of $\tau \geqslant l_0$. Since $G(M^{(a)}) \setminus \{0\} = \{(\varepsilon_{i_a}^{(a)})^{q^{j_a-1}} \mid i_a = 1, \dots, r^{(a)},\ j_a = 1, \dots, n_{i_a}^{(a)}\}$ and $M^{(1)} \dots M^{(h)} \prec M$, we have

$$\begin{aligned} &Q(M^{(1)}, \dots, M^{(h)}) \setminus \{0\} \\ &\quad = \Big\{ \prod_{a=1}^{h} (\varepsilon_{i_a}^{(a)})^{q^{j_a-1}} \mid i_a = 1, \dots, r^{(a)},\ j_a = 1, \dots, n_{i_a}^{(a)}, a = 1, \dots, h \Big\} \\ &\quad \subseteq \{\varepsilon_i^{q^{j-1}} \mid i = 1, \dots, r,\ j = 1, \dots, n_i\}. \end{aligned}$$

Thus

$$\begin{aligned} s_\tau = \beta_\tau + \sum_{i=1}^{r} \sum_{j=1}^{n_i} \sum_{(i_1, j_1, \dots, i_h, j_h) \in P_{ij}} \sum_{k_1=1}^{l_{i_1}^{(1)}} \cdots \sum_{k_h=1}^{l_{i_h}^{(h)}} \prod_{a=1}^{h} \beta_{i_a j_a k_a}^{(a)} \prod_{a=1}^{h} \binom{\tau + k_a - 1}{k_a - 1} \varepsilon_i^{\tau q^{j-1}}, \\ \tau = 0, 1, \dots \end{aligned} \tag{7.74}$$

From (7.12), i.e.,

$$\begin{aligned} \prod_{a=1}^{h} \binom{\tau + k_a - 1}{k_a - 1} = \sum_{k=1}^{k_1 + \cdots + k_h - h + 1} d(k-1, k_1 - 1, \dots, k_h - 1) \binom{\tau + k - 1}{k - 1}, \\ k_a = 1, \dots, l_{i_a}^{(a)},\ a = 1, \dots, h, \end{aligned} \tag{7.75}$$

we have

$$\begin{aligned} &\sum_{k_1=1}^{l_{i_1}^{(1)}} \cdots \sum_{k_h=1}^{l_{i_h}^{(h)}} \prod_{a=1}^{h} \beta_{i_a j_a k_a}^{(a)} \prod_{a=1}^{h} \binom{\tau + k_a - 1}{k_a - 1} \\ &\quad = \sum_{k_1=1}^{l_{i_1}^{(1)}} \cdots \sum_{k_h=1}^{l_{i_h}^{(h)}} \sum_{k=1}^{k_1 + \cdots + k_h - h + 1} d(k-1, k_1 - 1, \dots, k_h - 1) \prod_{a=1}^{h} \beta_{i_a j_a k_a}^{(a)} \binom{\tau + k - 1}{k - 1} \\ &\quad = \sum_{k=1}^{l_{i_1}^{(1)} + \cdots + l_{i_h}^{(h)} - h + 1} \Big[\sum_{k_1=1}^{l_{i_1}^{(1)}} \cdots \sum_{k_{h-1}=1}^{l_{i_{h-1}}^{(h-1)}} \sum_{k_h = \max(1, k - k_1 - \cdots - k_{h-1} + h - 1)}^{l_{i_h}^{(h)}} \\ &\qquad\qquad d(k-1, k_1 - 1, \dots, k_h - 1) \prod_{a=1}^{h} \beta_{i_a j_a k_a}^{(a)} \Big] \binom{\tau + k - 1}{k - 1}, \\ &\quad i_a = 1, \dots, r^{(a)},\ j_a = 1, \dots, n_{i_a}^{(a)},\ a = 1, \dots, h. \end{aligned} \tag{7.76}$$

Clearly, in (7.73), $\varphi(i_1, j_1, \dots, i_h, j_h, k) = 0$ whenever $k > l_{i_1}^{(1)} + \cdots + l_{i_h}^{(h)} - h + 1$. We prove $\varphi(i_1, j_1, \dots, i_h, j_h, k) = 0$ whenever $k > p^{v(l_{i_1}^{(1)}, \dots, l_{i_h}^{(h)})}$. From (7.73),

it is sufficient to prove that $d(k-1,k_1-1,\ldots,k_h-1)=0(\bmod p)$ whenever $k_1+\cdots+k_h-h+1\geqslant k>p^{v(l_{i_1}^{(1)},\ldots,l_{i_h}^{(h)})}$, $k_a=1,\ldots,l_{i_a}^{(a)}$, $i_a=1,\ldots,r^{(a)}$, $a=1,\ldots,h$. For any positive integer k, let $\Delta_k(z)=\sum_{\tau=0}^{\infty}\binom{\tau+k-1}{k-1}z^\tau$ over $GF(q)$. It is known that the period of $\Delta_k(z)$ is $p^{v(k)}$, where $v(k)$ is defined in (7.67). Let

$$\Delta(z)=\prod_{a=1}^{h}\Delta_{k_a}(z),$$

$$\Delta'(z)=\sum_{k=1}^{k_1+\cdots+k_h-h+1}d(k-1,k_1-1,\ldots,k_h-1)\Delta_k(z).$$

From (7.75), we have $\Delta(z)=\Delta'(z)$. It follows that the period u of $\Delta(z)$ and the period u' of $\Delta'(z)$ are the same. Since $k_a\leqslant l_{i_a}^{(a)}$, we have $v(k_a)\leqslant v(l_{i_a}^{(a)})$. Thus $p^{v(k_a)}$, the period of $\Delta_{k_a}(z)$, is a divisor of $p^{v(l_{i_a}^{(a)})}$, $a=1,\ldots,h$. Clearly, u is a divisor of the least common multiple of $p^{v(k_1)}$, …, $p^{v(k_h)}$. From $v(l_{i_1}^{(1)},\ldots,l_{i_h}^{(h)})=\max(v(l_{i_1}^{(1)}),\ldots,v(l_{i_h}^{(h)}))$, u is a divisor of $p^{v(l_{i_1}^{(1)},\ldots,l_{i_h}^{(h)})}$. Suppose to the contrary that there exists k, $k_1+\cdots+k_h-h+1\geqslant k>p^{v(l_{i_1}^{(1)},\ldots,l_{i_h}^{(h)})}$ such that $d(k-1,\,k_1-1,\ldots,k_h-1)\neq 0\ (\bmod p)$. Then the period u' of $\Delta'(z)$ is a multiple of $p^{v(k)}$ and $v(k)\geqslant v(l_{i_1}^{(1)},\ldots,l_{i_h}^{(h)})+1$. It follows that u' is a multiple of $p^{v(l_{i_1}^{(1)},\ldots,l_{i_h}^{(h)})+1}$. Thus $u<u'$. This contradicts $u=u'$. We conclude that $d(k-1,k_1-1,\ldots,k_h-1)=0(\bmod p)$ whenever $k_1+\cdots+k_h-h+1\geqslant k>p^{v(l_{i_1}^{(1)},\ldots,l_{i_h}^{(h)})}$. Since $k>p^{v(l_{i_1}^{(1)},\ldots,l_{i_h}^{(h)})}$ implies $\varphi(i_1,j_1,\ldots,i_h,j_h,k)=0$, from (7.76), we have

$$\begin{aligned}&\sum_{k_1=1}^{l_{i_1}^{(1)}}\cdots\sum_{k_h=1}^{l_{i_h}^{(h)}}\prod_{a=1}^{h}\beta_{i_aj_ak_a}^{(a)}\prod_{a=1}^{h}\tbinom{\tau+k_a-1}{k_a-1}\\&\quad=\sum_{k=1}^{l(l_{i_1}^{(1)},\ldots,l_{i_h}^{(h)})}\varphi(i_1,j_1,\ldots,i_h,j_h,k)\tbinom{\tau+k-1}{k-1},\\&\quad i_a=1,\ldots,r^{(a)},\ j_a=1,\ldots,n_{i_a}^{(a)},\ a=1,\ldots,h,\end{aligned}$$

where $l(l_{i_1}^{(1)},\ldots,l_{i_h}^{(h)})$ is defined in (7.67). Replacing the leftside by the rightside of the above equation in (7.74), we obtain

$$\begin{aligned}s_\tau=\beta_\tau+\sum_{i=1}^{r}\sum_{j=1}^{n_i}\sum_{(i_1,j_1,\ldots,i_h,j_h)\in P_{ij}}\sum_{k=1}^{l(l_{i_1}^{(1)},\ldots,l_{i_h}^{(h)})}\varphi(i_1,j_1,\ldots,i_h,j_h,k)\tbinom{\tau+k-1}{k-1}\varepsilon_i^{\tau q^{j-1}},\\ \tau=0,1,\ldots\end{aligned}$$

From $M^{(1)}\dots M^{(h)} \prec M$, we have $l(l_{i_1}^{(1)},\dots,l_{i_h}^{(h)}) \leqslant l_i$ whenever $(i_1,j_1,\dots,i_h,j_h)\in P_{ij}$. Noticing $\varphi(i_1,j_1,\dots,i_h,j_h,k)=0$ whenever $k > l(l_{i_1}^{(1)},\dots,l_{i_h}^{(h)})$, this yields

$$\begin{aligned} s_\tau &= \beta_\tau + \sum_{i=1}^{r}\sum_{j=1}^{n_i}\sum_{(i_1,j_1,\dots,i_h,j_h)\in P_{ij}}\sum_{k=1}^{l_i}\varphi(i_1,j_1,\dots,i_h,j_h,k)\tbinom{\tau+k-1}{k-1}\varepsilon_i^{\tau q^{j-1}} \\ &= \beta_\tau + \sum_{i=1}^{r}\sum_{j=1}^{n_i}\sum_{k=1}^{l_i}\Big[\sum_{(i_1,j_1,\dots,i_h,j_h)\in P_{ij}}\varphi(i_1,j_1,\dots,i_h,j_h,k)\Big]\tbinom{\tau+k-1}{k-1}\varepsilon_i^{\tau q^{j-1}}, \\ &\tau = 0,1,\dots \end{aligned}$$

We conclude that $\Omega(z)\in \Psi_M^{(1)}(z)$ and its $(\varepsilon_1,\dots,\varepsilon_r)$ root coordinate β is determined by (7.72). □

Since $\Omega = \Omega_1\dots\Omega_h$ is a sequence over $GF(q)$, in its $(\varepsilon_1,\dots,\varepsilon_r)$ root coordinate β, components satisfy $\beta_{ijk} = \beta_{i1k}^{q^{j-1}}$. Therefore, it is not necessary to compute β_{ijk} using (7.72) for $j>1$.

It is easy to see that all characteristic polynomials of $M^{(1)},\dots,M^{(h)}$ have no nonzero repeated root if and only if the characteristic polynomial of $M^{(1)}\dots M^{(h)}$ has no nonzero repeated root. Assume that the characteristic polynomial of M has no nonzero repeated root. Then we have

$$\varphi(i_1,j_1,\dots,i_h,j_h,k) = \prod_{a=1}^{h}\beta_{i_aj_ak_a}^{(a)}$$

for $k=1$, and $\varphi(i_1,j_1,\dots,i_h,j_h,k)=0$ for $k>1$. Thus the formula (7.72) for computing β_{i1k} may be simplified into the following

$$\begin{aligned} &\beta_{ij1} = \psi(i,j,1,\beta^{(1)},\dots,\beta^{(h)}) = \sum_{(i_1,j_1,\dots,i_h,j_h)\in P_{ij}}\prod_{a=1}^{h}\beta_{i_aj_a1}^{(a)}, \\ &\beta_{ijk} = 0,\ k=2,\dots,l_i, \\ &\quad i=1,\dots,r,\ j=1,\dots,n_i. \end{aligned} \tag{7.77}$$

For any autonomous finite automaton $M=\langle Y,S,\delta,\lambda\rangle$, if Y and S are the column vector spaces over $GF(q)$ of dimensions m and n, respectively, and $\delta(s)=As$ for some $n\times n$ matrix A over $GF(q)$, M is called a *linear backward* autonomous finite automaton over $GF(q)$. A is referred to as the *state transition matrix* of M, and $|zE-A|$ is referred to as the *characteristic polynomial* of M, where E stands for the $n\times n$ identity matrix.

Below we discuss the problem: given a linear backward autonomous finite automaton M, find linear autonomous finite automaton $\bar{M}$ with $M\prec\bar{M}$.

Let M be a linear backward autonomous finite automaton over $GF(q)$ with state transition matrix

$$A = \begin{bmatrix} P_{f^{(1)}(z)} & & \\ & \ddots & \\ & & P_{f^{(v)}(z)} \end{bmatrix}, \tag{7.78}$$

where $f^{(i)}$ is a polynomial of degree $n^{(i)}$ with leading coefficient 1, $i = 1, \ldots, v$. Let M_i be in $\Re$ with characteristic polynomial $f^{(i)}$, $i = 1, \ldots, v$. Given c, $1 \leqslant c \leqslant m$, assume that the c-th component function λ_c of the output function λ of M is given by

$$\begin{aligned} &\lambda_c(s_{11}, \ldots, s_{1n^{(1)}}, \ldots, s_{v1}, \ldots, s_{vn^{(v)}}) \\ &\quad = \sum_{\substack{h_{ij}=0,\ldots,q-1 \\ i=1,\ldots,v \\ j=1,\ldots,n^{(i)}}} c_{h_{11}\ldots h_{1n^{(1)}}\ldots h_{v1}\ldots h_{vn^{(v)}}} s_{11}^{h_{11}} \ldots s_{1n^{(1)}}^{h_{1n^{(1)}}} \ldots s_{v1}^{h_{v1}} \ldots s_{vn^{(v)}}^{h_{vn^{(v)}}}, \end{aligned} \tag{7.79}$$

where $c_{h_{11}\ldots h_{1n^{(1)}}\ldots h_{v1}\ldots h_{vn^{(v)}}} \in GF(q)$, $h_{ij} = 0, \ldots, q-1$, $i = 1, \ldots, v$, $j = 1, \ldots, n^{(i)}$. Let

$$M_{\lambda_c} = \sum_{\substack{h_{ij}=0,\ldots,q-1 \\ i=1,\ldots,v,\ j=1,\ldots,n^{(i)} \\ c_{h_{11}\ldots h_{1n^{(1)}}\ldots h_{v1}\ldots h_{vn^{(v)}}} \neq 0}} M_1^{h_{11}+\cdots+h_{1n^{(1)}}} \ldots M_v^{h_{v1}+\cdots+h_{vn^{(v)}}}, \tag{7.80}$$

where $M_i^0 = M_E$. Let $\Phi_M^{(c)}(z) = \{\Phi_M^{(c)}(s, z) \mid s \in S\}$.

Theorem 7.4.3. *Assume that $\bar{M} \in \Re$ and $M_{\lambda_c} \prec \bar{M}$. Then we have $\Phi_M^{(c)}(z) \subseteq \Psi_{\bar{M}}^{(1)}(z)$. Moreover, if the $(\varepsilon_1^{(a)}, \ldots, \varepsilon_{r^{(a)}}^{(a)})$ root coordinate of $\Psi_{M_a}^{(1)}(s^{(a)}, z)$ is $\beta^{(a)}$, $a = 1, \ldots, v$ and the $(\bar{\varepsilon}_1, \ldots, \bar{\varepsilon}_{\bar{r}})$ root coordinate of $\Phi_M^{(c)}([s^{(1)}, \ldots, s^{(v)}]^{\mathrm{T}}, z)$ in $\Psi_{\bar{M}}^{(1)}(z)$ is $\bar{\beta}$, then*

$$\bar{\beta} = \sum_{\substack{h_{ij}=0,\ldots,q-1 \\ i=1,\ldots,v \\ j=1,\ldots,n^{(i)}}} c_{h_{11}\ldots h_{1n^{(1)}}\ldots h_{v1}\ldots h_{vn^{(v)}}} \beta^{[h_{11}\ldots h_{1n^{(1)}}\ldots h_{v1}\ldots h_{vn^{(v)}}]},$$

where

$$\begin{aligned} \beta_k^{[h_{11}\ldots h_{1n^{(1)}}\ldots h_{v1}\ldots h_{vn^{(v)}}]} &= \prod_{a=1}^{v} \prod_{b=1}^{n^{(a)}} (s_{b-1+k}^{(a)})^{h_{ab}} - \prod_{a=1}^{v} \prod_{b=1}^{n^{(a)}} (s_{b-1+k}^{(a)} - \beta_{b-1+k}^{(a)})^{h_{ab}}, \\ k &= 0, \ldots, \bar{l}_0 - 1, \end{aligned}$$

$$\beta_{ijk}^{[h_{11}\dots h_{1n^{(1)}}\dots h_{v1}\dots h_{vn^{(v)}}]} = \psi(i,j,k,\underbrace{\beta^{[11]},\dots,\beta^{[11]}}_{h_{11}},\dots,\underbrace{\beta^{[1n^{(1)}]},\dots,\beta^{[1n^{(1)}]}}_{h_{1n^{(1)}}},$$
$$\dots,\underbrace{\beta^{[v1]},\dots,\beta^{[v1]}}_{h_{v1}},\dots,\underbrace{\beta^{[vn^{(v)}]},\dots,\beta^{[vn^{(v)}]}}_{h_{vn^{(v)}}}),$$
$$i=1,\dots,\bar{r},\ j=1,\dots,\bar{n}_i,\ k=1,\dots,\bar{l}_i, \tag{7.81}$$
$$\beta_k^{[ab]} = \beta_{b-1+k}^{(a)},\ k=0,\dots,l_0^{(a)}-b,$$
$$\beta_k^{[ab]} = 0,\ k=l_0^{(a)}-b+1,\dots,l_0^{(a)}-1,$$
$$\beta_{ijh}^{[ab]} = \prod_{k=h}^{l_i^{(a)}} \beta_{ijk}^{(a)} \binom{k-h+b-2}{k-h} (\varepsilon_i^{(a)})^{(b-1)q^{j-1}},$$
$$i=1,\dots,r^{(a)},\ j=1,\dots,n_i^{(a)},\ h=1,\dots,l_i^{(a)},$$
$$a=1,\dots,v,\ b=1,\dots,n_i^{(a)},$$

$\Psi_{M_a}^{(1)}(s^{(a)},z) = \sum_{i=0}^{\infty} s_i^{(a)} z^i$, $\beta_\tau^{(a)} = 0$ *whenever* $\tau \geqslant l_0^{(a)}$, $a=1,\dots,v$, *and* ψ *is defined by* (7.72) *and* (7.73).

Proof. Since M_a is a shift register, from (7.79) we have

$$\Phi_M^{(c)}([s^{(1)},\dots,s^{(v)}]^{\mathrm{T}},z) \tag{7.82}$$
$$= \sum_{\substack{h_{ij}=0,\dots,q-1\\ i=1,\dots,v\\ j=1,\dots,n^{(i)}}} c_{h_{11}\dots h_{1n^{(1)}}\dots h_{v1}\dots h_{vn^{(v)}}} \prod_{a=1}^{v}\prod_{b=1}^{n^{(a)}} (D^{b-1}(\Psi_{M_a}^{(1)}(s^{(a)},z)))^{h_{ab}},$$

where the 0-th power of any sequence is the 1 sequence. From Theorem 7.4.1 and Theorem 7.4.2, we have $\Phi_M^{(c)}(z) \subseteq \Psi_{M_{\lambda_c}}^{(1)}(z)$. Since $M_{\lambda_c} \prec \bar{M}$, $\Phi_M^{(c)}(z) \subseteq \Psi_{\bar{M}}^{(1)}(z)$ holds.

We give some explanation on root basis mentioned in the theorem. Let $GF(q^*)$ be a splitting field of $f^{(1)}(z)$, …, $f^{(v)}(z)$ and the characteristic polynomial $\bar{f}(z)$ of $\bar{M}$. Then the factorizations

$$\bar{f}(z) = z^{\bar{l}_0} \prod_{i=1}^{\bar{r}} \prod_{j=1}^{\bar{n}_i} (z - \bar{\varepsilon}_i^{q^{j-1}})^{\bar{l}_i},$$
$$f^{(a)}(z) = z^{l_0^{(a)}} \prod_{i=1}^{r^{(a)}} \prod_{j=1}^{n_i^{(a)}} (z - (\varepsilon_i^{(a)})^{q^{j-1}})^{l_i^{(a)}},$$
$$a=1,\dots,v$$

determine the $(\bar{\varepsilon}_1,\dots,\bar{\varepsilon}_{\bar{r}})$ root basis of $\Psi_{\bar{M}^*}^{(1)}(z)$, the $(\varepsilon_1^{(a)},\dots,\varepsilon_{r^{(a)}}^{(a)})$ root basis of $\Psi_{M_a^*}^{(1)}(z)$, $a=1,\dots,h$, where M^* is the natural extension of M over $GF(q^*)$,

and M_a^* is the natural extension of M_a over $GF(q^*)$, $a = 1, \ldots, v$. Since the $(\varepsilon_1^{(a)}, \ldots, \varepsilon_{r^{(a)}}^{(a)})$ root coordinate of $\Psi_{M_a}^{(1)}(s^{(a)}, z)$ is $\beta^{(a)}$, from Theorem 7.3.1, $\beta^{[ab]}$ is the $(\varepsilon_1^{(a)}, \ldots, \varepsilon_{r^{(a)}}^{(a)})$ root coordinate of $D^{b-1}(\Psi_{M_a}^{(1)}(s^{(a)}, z))$, where components of $\beta^{[ab]}$ are given in (7.81), $a = 1, \ldots, v$, $b = 1, \ldots, n^{(a)}$. Suppose that $c_{h_{11}\ldots h_{1n^{(1)}}\ldots h_{v1}\ldots h_{vn^{(v)}}} \neq 0$. Then $M_1^{h_{11}+\cdots+h_{1n^{(1)}}} \ldots M_v^{h_{v1}+\cdots+h_{vn^{(v)}}} \prec M_{\lambda_c} \prec \bar{M}$. From Theorem 7.4.2, $\prod_{a=1}^{v}\prod_{b=1}^{n^{(a)}}(D^{b-1}(\Psi_{M_a}^{(1)}(s^{(a)}, z)))^{h_{ab}}$ is in $\Psi_{\bar{M}}^{(1)}(z)$, and its $(\bar{\varepsilon}_1, \ldots, \bar{\varepsilon}_{\bar{r}})$ root coordinate is $\beta^{[h_{11}\ldots h_{1n^{(1)}}\ldots h_{v1}\ldots h_{vn^{(v)}}]}$, of which components are given in (7.81). From (7.82), we have

$$\Phi_M^{(c)}([s^{(1)}, \ldots, s^{(v)}]^{\mathrm{T}}, z) \in \Psi_{\bar{M}}^{(1)}(z)$$

and its $(\bar{\varepsilon}_1, \ldots, \bar{\varepsilon}_{\bar{r}})$ root coordinate is given by (7.81). □

Corollary 7.4.1. *Let M be a linear backward shift register over $GF(q)$, of which the c-th component function of the output function is given by*

$$\lambda_c(s_1, \ldots, s_n) = \sum_{h_1,\ldots,h_n=0}^{q-1} c_{h_1\ldots h_n} s_1^{h_1} \ldots s_n^{h_n}. \tag{7.83}$$

Assume that $M_1 \in \Re$ and characteristic polynomials of M and M_1 are the same. Assume that $\bar{M} \in \Re$ and $M_1^{h_1+\cdots+h_n} \prec \bar{M}$ whenever $c_{h_1\ldots h_n} \neq 0$, $h_1, \ldots, h_n = 0, \ldots, q-1$. Then we have $\Phi_M^{(c)}(z) \subseteq \Psi_{\bar{M}}^{(1)}(z)$. Moreover, if the $(\varepsilon_1, \ldots, \varepsilon_r)$ root coordinate of $\Psi_{M_1}^{(1)}(s, z)$ is β, and the $(\bar{\varepsilon}_1, \ldots, \bar{\varepsilon}_{\bar{r}})$ root coordinate of $\Phi_M^{(c)}(s, z)$ in $\Psi_{\bar{M}}^{(1)}(z)$ is $\bar{\beta}$, then

$$\bar{\beta} = \sum_{h_1,\ldots,h_n=0}^{q-1} c_{h_1\ldots h_n} \beta^{[h_1\ldots h_n]}, \tag{7.84}$$

where

$$\begin{aligned}
&\beta_k^{[h_1\ldots h_n]} = \prod_{b=1}^{n} s_{b-1+k}^{h_b} - \prod_{b=1}^{n}(s_{b-1+k} - \beta_{b-1+k})^{h_b},\ k = 0, \ldots, \bar{l}_0 - 1,\\
&\beta_{ijk}^{[h_1\ldots h_n]} = \psi(i, j, k, \underbrace{\beta^{[1]}, \ldots, \beta^{[1]}}_{h_1}, \ldots, \underbrace{\beta^{[n]}, \ldots, \beta^{[n]}}_{h_n}),\\
&\quad i = 1, \ldots, \bar{r},\ j = 1, \ldots, \bar{n}_i,\ k = 1, \ldots, \bar{l}_i,\\
&\beta_k^{[b]} = \beta_{b-1+k},\ k = 0, \ldots, l_0 - b,\\
&\beta_k^{[b]} = 0,\ k = l_0 - b + 1, \ldots, l_0 - 1,\\
&\beta_{ijh}^{[b]} = \prod_{k=h}^{l_i} \beta_{ijk}^{\binom{k-h+b-2}{k-h}\varepsilon_i^{(b-1)q^{j-1}}},\\
&\quad i = 1, \ldots, r,\ j = 1, \ldots, n_i,\ h = 1, \ldots, l_i,\ b = 1, \ldots, n,
\end{aligned} \tag{7.85}$$

$\Psi^{(1)}_{M_1}(s,z) = \sum_{i=0}^{\infty} s_i z^i$, $\beta_\tau = 0$ *whenever* $\tau \geqslant l_0$, *and* ψ *is defined by* (7.72) *and* (7.73).

For a linear backward autonomous finite automaton $M' = \langle Y, S, \delta', \lambda' \rangle$ over $GF(q)$ of which the state transition matrix A' is not in the form of (7.78), we can transform it to the form of (7.78) by similarity transformation, that is, we can find a nonsingular matrix P over $GF(q)$ such that $A = P^{-1}A'P$ can be expressed in the form of (7.78). Let $M = \langle Y, S, \delta, \lambda \rangle$, where $\delta(s) = As$, $\lambda(s) = \lambda'(Ps)$. Then M is a linear backward autonomous finite automaton over $GF(q)$. It is easy to verify that M and M' are isomorphic and the state s' of M' and the state $P^{-1}s'$ of M are equivalent.

Notice that if the $(\varepsilon_1, \ldots, \varepsilon_r)$ root coordinate of a periodic Ω in Φ_M is β, then the linear complexity of Ω equals to $\sum_{i=1}^{r} n_i l_{i1}$, where $l_{i1} = \min h[h \geqslant 0, \beta_{i1k} = 0 \text{ if } h < k \leqslant l_i]$, $i = 1, \ldots, r$, the linear complexity of Ω means the minimal state space dimension of linear shift registers over $GF(q)$ which generate Ω.

We finish this section by the following theorem.

Theorem 7.4.4. *For any autonomous finite automaton* $M = \langle Y, S, \delta, \lambda \rangle$, *if* Y *is a column vector space over* $GF(q)$, *then there exists a linear autonomous finite automaton* $\bar{M} = \langle Y, \bar{S}, \bar{\delta}, \bar{\lambda} \rangle$ *over* $GF(q)$ *such that the dimension of* $\bar{S} \leqslant |S|$ *and* M *is isomorphic to a finite subautomaton of* $\bar{M}$.

Proof. Let $S = \{s_1, \ldots, s_n\}$ and

$$\begin{aligned} \delta(s_i) &= s_{j_i}, \\ \lambda(s_i) &= \begin{bmatrix} y_{1i} \\ \vdots \\ y_{mi} \end{bmatrix}, \\ i &= 1, \ldots, n, \end{aligned} \tag{7.86}$$

where $y_{ji} \in GF(q)$, $i = 1, \ldots, n$, $j = 1, \ldots, m$. Let $\bar{S}$ be the column vector space over $GF(q)$ of dimension n. Let $\bar{\gamma}_i$ be the column vector of dimension n of which the i-th component is 1 and 0 elsewhere, $i = 1, \ldots, n$. Define

$$\bar{\delta}(\bar{s}) = \bar{A}\bar{s}, \ \bar{\lambda}(\bar{s}) = \bar{C}\bar{s}, \ \bar{s} \in \bar{S}, \tag{7.87}$$

where

$$\begin{aligned} \bar{A} &= [\bar{\gamma}_{j_1}, \ldots, \bar{\gamma}_{j_n}], \\ \bar{C} &= \begin{bmatrix} y_{11} & y_{12} & \cdots & y_{1n} \\ y_{21} & y_{22} & \cdots & y_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ y_{m1} & y_{m2} & \cdots & y_{mn} \end{bmatrix}. \end{aligned} \tag{7.88}$$

Take

$$\tilde{S} = \{\bar{\gamma}_1, \ldots, \bar{\gamma}_n\}.$$

From (7.87) and (7.88), $\bar{\delta}(\bar{\gamma}_i) = \bar{\gamma}_{j_i}$, $i = 1, \ldots, n$. Therefore, $\tilde{S}$ is closed in $\bar{M}$. It follows that $\tilde{M} = \langle Y, \tilde{S}, \tilde{\delta}, \tilde{\lambda} \rangle$ is a finite subautomaton of $\bar{M}$, where $\tilde{\delta}(\bar{s}) = \bar{\delta}(\bar{s})$, $\tilde{\lambda}(\bar{s}) = \bar{\lambda}(\bar{s})$, for $\bar{s} \in \tilde{S}$. Let

$$\varphi(s_i) = \bar{\gamma}_i, \ i = 1, \ldots, n.$$

Clearly, φ is a bijection from S to $\tilde{S}$. From $\bar{\delta}(\varphi(s_i)) = \bar{\delta}(\bar{\gamma}_i) = \bar{\gamma}_{j_i} = \varphi(s_{j_i}) = \varphi(\delta(s_i))$, we have

$$\bar{\delta}(\varphi(s)) = \varphi(\delta(s)), \ s \in S.$$

Since $\bar{\lambda}(\varphi(s_i)) = \bar{\lambda}(\bar{\gamma}_i) = [y_{1i}, \ldots, y_{mi}]^{\mathrm{T}} = \lambda(s_i)$, we have

$$\bar{\lambda}(\varphi(s)) = \lambda(s), \ s \in S.$$

Thus φ is an isomorphism from M to $\tilde{M}$. We conclude that M and $\tilde{M}$ are isomorphic. □

7.5 Decimation

For any sequence $\omega = [w_0, w_1, \ldots, w_\tau, \ldots]$ and any positive integer u, the sequence $[w_0, w_u, \ldots, w_{u\tau}, \ldots]$ is called the *u-decimation* of ω.

Let M be a linear shift register over $GF(q)$ with structure parameters m, n and structure matrices A, C. Let $f(z)$ be the characteristic polynomial of M.

Assume that $GF(q^*)$ is a splitting field of $f(z)$. Let M^* be the natural extension of M over $GF(q^*)$. Take an $(\varepsilon_1, \ldots, \varepsilon_r)$ root basis of $\Psi^{(1)}_{M^*}(z)$, where

$$f(z) = z^{l_0} \prod_{i=1}^{r} \prod_{j=1}^{n_i} (z - \varepsilon_i^{q^{j-1}})^{l_i}.$$

Given a positive integer u, let R be a relation on $\{1, \ldots, r\}$, where iRj if and only if ε_i^u and ε_j^u are conjugate on $GF(q)$, i.e., $\varepsilon_i^u = (\varepsilon_j^u)^{q^k}$ for some nonnegative integer k . Clearly, the relation R is reflexive, symmetric and transitive. Let

$$P_1, P_2, \ldots, P_{\bar{r}} \tag{7.89}$$

be the equivalence classes of R. For each h, $1 \leqslant h \leqslant \bar{r}$, fix an integer m_h in P_h. Let

$$\bar{\varepsilon}_h = \varepsilon_{m_h}^u. \tag{7.90}$$

From the definition of P_h, for any i in P_h, ε_i^u and $\varepsilon_{m_h}^u$ are conjugate on $GF(q)$; therefore, there exists $k \geqslant 0$ such that $\varepsilon_i^u = \varepsilon_{m_h}^{uq^k}$, i.e., $\varepsilon_i^u = \bar{\varepsilon}_h^{q^k}$. Let

$$\begin{aligned} v_i &= \min k(k \geqslant 0 \ \& \ \varepsilon_i^u = \bar{\varepsilon}_h^{q^k}), \\ &\quad i \in P_h, \ h = 1, \ldots, \bar{r}. \end{aligned} \tag{7.91}$$

We use $\bar{n}_h$ to denote the degree of the minimal polynomial over $GF(q)$ of $\bar{\varepsilon}_h$. Then $\bar{\varepsilon}_h^{q^{\bar{n}_h}} = \bar{\varepsilon}_h$. Thus

$$v_i < \bar{n}_h, \ i \in P_h, \ h = 1, \ldots, \bar{r}. \tag{7.92}$$

Let $i \in P_h$. Then the minimal polynomials over $GF(q)$ of ε_i^u and $\bar{\varepsilon}_h$ are the same. Thus $\bar{n}_h = \min k(k > 0 \ \& \ \varepsilon_i^{uq^k} = \varepsilon_i^u)$. Since $\varepsilon_i^{uq^{n_i}} = (\varepsilon_i^{q^{n_i}})^u = \varepsilon_i^u$, we have $\bar{n}_h | n_i$. Let

$$q_i = n_i/\bar{n}_h, \ i \in P_h, \ h = 1, \ldots, \bar{r}. \tag{7.93}$$

Then q_i is a positive integer. Let $u = p^w u'$, where u' and p are coprime. Take[1]

$$\begin{aligned} \bar{l}_0 &= \min \ k\,(\,ku \geqslant l_0\,), \\ \bar{l}_h &= \max \ k\,(\,k - 1 = \lfloor (l_i - 1)/p^w \rfloor, \, i \in P_h\,), \\ &\quad h = 1, \ldots, \bar{r}. \end{aligned} \tag{7.94}$$

Let

$$\bar{f}(z) = z^{\bar{l}_0} \prod_{h=1}^{\bar{r}} \prod_{j=1}^{\bar{n}_h} (z - \bar{\varepsilon}_h^{q^{j-1}})^{\bar{l}_h}. \tag{7.95}$$

It is easy to show that $\bar{f}(z)$ is a polynomial over $GF(q)$ with leading coefficient 1. Let $\bar{M}$ be a linear autonomous shift register over $GF(q)$ with characteristic polynomial $\bar{f}(z)$. Let $\bar{M}^*$ be the natural extension of $\bar{M}$ over $GF(q^*)$. Then $\bar{\varepsilon}_1, \ldots, \bar{\varepsilon}_{\bar{r}}$ determine an $(\bar{\varepsilon}_1, \ldots, \bar{\varepsilon}_{\bar{r}})$ root basis of $\Psi_{\bar{M}^*}^{(1)}$.

Assume that $\Omega = [s_0, s_1, \ldots, s_\tau, \ldots]$ is in $\Psi_M^{(1)}$ and its $(\varepsilon_1, \ldots, \varepsilon_r)$ root coordinate is β. Then

$$\begin{aligned} s_\tau &= \beta_\tau + \sum_{i=1}^{r} \sum_{j=1}^{n_i} \sum_{k=1}^{l_i} \beta_{ijk} \tbinom{\tau+k-1}{k-1} \varepsilon_i^{\tau q^{j-1}}, \\ &\quad \tau = 0, 1, \ldots, \end{aligned}$$

[1] $\lfloor x \rfloor$ stands for the maximal integer $\leqslant x$.

where $\beta_\tau = 0$ whenever $\tau \geqslant l_0$. Let $\bar{\Omega} = [\bar{s}_0, \bar{s}_1, \ldots, \bar{s}_\tau, \ldots]$ be the u-decimation of Ω. Then

$$\begin{aligned}
\bar{s}_\tau = s_{u\tau} &= \beta_{u\tau} + \sum_{i=1}^{r}\sum_{j=1}^{n_i}\sum_{k=1}^{l_i}\beta_{ijk}\binom{u\tau+k-1}{k-1}\varepsilon_i^{u\tau q^{j-1}} \\
&= \beta_{u\tau} + \sum_{h=1}^{\bar{r}}\sum_{i\in P_h}\sum_{j=1}^{n_i}\sum_{k=1}^{l_i}\beta_{ijk}\binom{u\tau+k-1}{k-1}\bar{\varepsilon}_h^{\tau q^{v_i+j-1}} \\
&= \beta_{u\tau} + \sum_{h=1}^{\bar{r}}\sum_{i\in P_h}\sum_{j=1}^{\bar{n}_h}\sum_{c=0}^{q_i-1}\sum_{k=1}^{l_i}\beta_{i,c\bar{n}_h+j,k}\binom{u\tau+k-1}{k-1}\bar{\varepsilon}_h^{\tau q^{v_i+c\bar{n}_h+j-1}} \\
&= \beta_{u\tau} + \sum_{h=1}^{\bar{r}}\sum_{i\in P_h}\sum_{j=1}^{\bar{n}_h}\sum_{c=0}^{q_i-1}\sum_{k=1}^{l_i}\beta_{i,c\bar{n}_h+j,k}\binom{u\tau+k-1}{k-1}\bar{\varepsilon}_h^{\tau q^{v_i+j-1}} \qquad (7.96) \\
&= \beta_{u\tau} + \sum_{h=1}^{\bar{r}}\sum_{i\in P_h}\sum_{j=v_i+1}^{v_i+\bar{n}_h}\sum_{c=0}^{q_i-1}\sum_{k=1}^{l_i}\beta_{i,c\bar{n}_h+j-v_i,k}\binom{u\tau+k-1}{k-1}\bar{\varepsilon}_h^{\tau q^{j-1}} \\
&= \beta_{u\tau} + \sum_{h=1}^{\bar{r}}\sum_{i\in P_h}\sum_{j=1}^{\bar{n}_h}\sum_{c=0}^{q_i-1}\sum_{k=1}^{l_i}\beta_{i,c\bar{n}_h+(j-v_i)(\bmod \bar{n}_h),k}\binom{u\tau+k-1}{k-1}\bar{\varepsilon}_h^{\tau q^{j-1}} \\
&= \beta'_\tau + \sum_{h=1}^{\bar{r}}\sum_{j=1}^{\bar{n}_h}\sum_{k=1}^{\tilde{l}_h}\beta'_{hjk}\binom{u\tau+k-1}{k-1}\bar{\varepsilon}_h^{\tau q^{j-1}}, \\
&\tau = 0, 1, \ldots,
\end{aligned}$$

where $a(\bmod \bar{n}_h) = \min\ b(b > 0\ \&\ a = b\ (\bmod \bar{n}_h))$, $\tilde{l}_h = \max\{l_i, i \in P_h\}$, $h = 1, \ldots, \bar{r}$,

$$\begin{aligned}
&\beta'_\tau = 0,\ \text{if } \tau \geqslant \bar{l}_0, \\
&\beta'_\tau = \beta_{u\tau},\ \tau = 0, \ldots, \bar{l}_0 - 1, \qquad (7.97) \\
&\beta'_{hjk} = \sum_{i\in P_h}\sum_{c=0}^{q_i-1}\beta_{i,c\bar{n}_h+(j-v_i)(\bmod \bar{n}_h),k}\ , \\
&\qquad h = 1, \ldots, \bar{r},\ j = 1, \ldots, \bar{n}_h, k = 1, \ldots, \tilde{l}_h,
\end{aligned}$$

and $\beta_{ijk} = 0$ for $i \in P_h$, $h = 1, \ldots, \bar{r}$, $j = 1, \ldots, n_i$, $k = l_i + 1, \ldots, \tilde{l}_h$. We use $\theta(c)$ to denote $\lfloor c/p^w \rfloor$. From Theorem 7.1.3, it is easy to show that

$$\begin{aligned}
\binom{u\tau+k-1}{k-1} &= \binom{p^w(u'\tau+\theta(k-1))+\nu}{p^w\theta(k-1)+\nu} \\
&= \binom{u'\tau+\theta(k-1)}{\theta(k-1)}\binom{\nu}{\nu} \pmod p \\
&= \binom{u'\tau+\theta(k-1)}{\theta(k-1)} \pmod p,
\end{aligned}$$

where $0 \leqslant \nu < p^w$ and $\nu = k - 1 (\bmod\ p^w)$. From (7.11), for any k, we have

$$\binom{u'\tau+k-1}{k-1} = \sum_{a=1}^{k} c(u', k-1, a-1)\binom{\tau+a-1}{a-1},$$

$$c(u', k-1, a-1) = \sum_{i=1}^{a-1}(-1)^i \binom{a-1}{i}\binom{-u'(i+1)+k-1}{k-1},$$

$$a = 1, \ldots, k.$$

Thus

$$\binom{u'\tau+\theta(k-1)}{\theta(k-1)} = \sum_{a=1}^{\theta(k-1)+1} c(u', \theta(k-1), a-1)\binom{\tau+a-1}{a-1}.$$

It follows that

$$\binom{u\tau+k-1}{k-1} = \sum_{a=1}^{\theta(k-1)+1} c(u', \theta(k-1), a-1)\binom{\tau+a-1}{a-1} \pmod{p}.$$

Replacing the leftside by the rightside of the above equation in (7.96), letting $c(u', k', a) = 0$ for $a > k'$, we have

$$\begin{aligned}
\bar{s}_\tau &= \beta'_\tau + \sum_{h=1}^{\bar{r}}\sum_{j=1}^{\bar{n}_h}\sum_{k=1}^{\tilde{l}_h}\beta'_{hjk}\sum_{a=1}^{\theta(k-1)+1} c(u', \theta(k-1), a-1)\binom{\tau+a-1}{a-1}\bar{\varepsilon}_h^{\tau q^{j-1}} \\
&= \beta'_\tau + \sum_{h=1}^{\bar{r}}\sum_{j=1}^{\bar{n}_h}\sum_{k=1}^{\tilde{l}_h}\beta'_{hjk}\sum_{a=1}^{k} c(u', \theta(k-1), a-1)\binom{\tau+a-1}{a-1}\bar{\varepsilon}_h^{\tau q^{j-1}} \\
&= \beta'_\tau + \sum_{h=1}^{\bar{r}}\sum_{j=1}^{\bar{n}_h}\sum_{a=1}^{\tilde{l}_h}\Big(\sum_{k=a}^{\tilde{l}_h} c(u', \theta(k-1), a-1)\beta'_{hjk}\Big)\binom{\tau+a-1}{a-1}\bar{\varepsilon}_h^{\tau q^{j-1}} \qquad (7.98) \\
&= \beta'_\tau + \sum_{h=1}^{\bar{r}}\sum_{j=1}^{\bar{n}_h}\sum_{a=1}^{\theta(\tilde{l}_h-1)+1}\Big(\sum_{k=a}^{\tilde{l}_h} c(u', \theta(k-1), a-1)\beta'_{hjk}\Big)\binom{\tau+a-1}{a-1}\bar{\varepsilon}_h^{\tau q^{j-1}}, \\
&\tau = 0, 1, \ldots
\end{aligned}$$

Since $\tilde{l}_h = \max\{l_i, i \in P_h\}$, from (7.94), we have $\bar{l}_h = \theta(\tilde{l}_h - 1) + 1$, $h = 1, \ldots, \bar{r}$. Let

$$\begin{aligned}
&\bar{\beta}_\tau = \beta'_\tau,\ \tau = 0, \ldots, \bar{l}_0 - 1, \\
&\bar{\beta}_{hja} = \sum_{k=a}^{\tilde{l}_h} c(u', \theta(k-1), a-1)\beta'_{hjk}, \qquad (7.99) \\
&\qquad h = 1, \ldots, \bar{r},\ j = 1, \ldots, \bar{n}_h, a = 1, \ldots, \bar{l}_h.
\end{aligned}$$

Then (7.98) can be written as

$$\bar{s}_\tau = \bar{\beta}_\tau + \sum_{h=1}^{\bar{r}} \sum_{j=1}^{\bar{n}_h} \sum_{a=1}^{\bar{l}_h} \bar{\beta}_{hja} \binom{\tau+a-1}{a-1} \bar{\varepsilon}_h^{\tau q^{j-1}},$$
$$\tau = 0, 1, \ldots,$$

where $\bar{\beta}_\tau = 0$ whenever $\tau \geqslant \bar{l}_0$. Therefore, $\bar{\Omega}$ is a sequence in $\Psi_{\bar{M}}^{(1)}$ of which the $(\bar{\varepsilon}_1, \ldots, \bar{\varepsilon}_{\bar{r}})$ root coordinate is $\bar{\beta}$.

(7.97) and (7.99) can be written in matrix form. We use E_i to denote the $i \times i$ identity matrix, 0_{ij} to denote the $i \times j$ zero matrix; and the leftmost j columns and the rightmost $i - j$ columns of E_i are denoted by $E_i(j)$ and $E_i(-j)$, respectively. Let

$$I_0(i) = [E_{\bar{n}_h}(-v_i), \underbrace{E_{\bar{n}_h}, \ldots, E_{\bar{n}_h}}_{n_i/\bar{n}_h - 1}, E_{\bar{n}_h}(v_i)], \quad i \in P_h,\ h = 1, \ldots, \bar{r},$$

$$I_0 = \begin{bmatrix} I_0(0) & & & \\ & DIA_{I_0(1),l_1} & & \\ & & \ddots & \\ & & & DIA_{I_0(r),l_r} \end{bmatrix},$$

$$I_{hkic} = \begin{cases} E_{\bar{n}_h}, & \text{if } i \in P_h, k = c, \\ 0_{\bar{n}_h \bar{n}_{h'}}, & \text{if } i \in P_{h'}, h \neq h' \text{or } k \neq c, \end{cases}$$
$$h = 1, \ldots, \bar{r},\ k = 1, \ldots, \tilde{l}_h,\ i = 1, \ldots, r,\ c = 1, \ldots, l_i, \tag{7.100}$$

$$I_{hk} = [I_{hk11}\ \ldots\ I_{hk1l_1}\ \ldots\ I_{hkr1}\ \ldots\ I_{hkrl_r}],\ h = 1, \ldots, \bar{r},\ k = 1, \ldots, \tilde{l}_h,$$

$$I_1' = \begin{bmatrix} I_{11} \\ \vdots \\ I_{1\tilde{l}_1} \\ \vdots \\ I_{\bar{r}1} \\ \vdots \\ I_{\bar{r}\tilde{l}_{\bar{r}}} \end{bmatrix}, \quad I_1 = \begin{bmatrix} E_{\bar{l}_0} & \\ & I_1' \end{bmatrix},$$

$$I_2(h) = \begin{bmatrix} c(0,0) & c(1,0) & \ldots & c(\bar{l}_h - 1, 0) & \ldots & c(\tilde{l}_h - 1, 0) \\ 0 & c(1,1) & \ldots & c(\bar{l}_h - 1, 1) & \ldots & c(\tilde{l}_h - 1, 1) \\ \vdots & \vdots & \ddots & \vdots & \ddots & \vdots \\ 0 & 0 & \ldots & c(\bar{l}_h - 1, \bar{l}_h - 1) & \ldots & c(\tilde{l}_h - 1, \bar{l}_h - 1) \end{bmatrix},$$
$$h = 1, \ldots, \bar{r},$$

$$I_2 = \begin{bmatrix} E_{\bar{l}_0} & & & & \\ & I_2(1) & & & \\ & & I_2(2) & & \\ & & & \ddots & \\ & & & & I_2(\bar{r}) \end{bmatrix},$$

where the definition of the symbol DIA is in Sect. 4.1 of Chap. 4, $I_0(0)$ is an $\bar{l}_0 \times l_0$ matrix of which the element at row i and column $u(i-1)+1$ is 1 and 0 elsewhere, and $c(i,j)$ stands for $c(u',\theta(i),j)E_{\bar{n}_h}$. It is easy to verify that (7.97) can be written as $\beta' = I_1 I_0 \beta$ and that (7.99) can be written as $\bar{\beta} = I_2 \beta'$. Therefore, $\bar{\beta} = I_2 I_1 I_0 \beta$. We complete the proof of the following theorem.

Theorem 7.5.1. *Let Ω be a sequence in $\Psi_M^{(1)}$, and β the $(\varepsilon_1, \ldots, \varepsilon_r)$ root coordinate of Ω. If $\bar{\Omega}$ is the u-decimation of Ω, then $\bar{\Omega}$ is in $\Psi_{\bar{M}}^{(1)}$ and its $(\bar{\varepsilon}_1, \ldots, \bar{\varepsilon}_{\bar{r}})$ root coordinate $\bar{\beta}$ is determined by*

$$\bar{\beta} = I_2 I_1 I_0 \beta.$$

Corollary 7.5.1. *Let*

$$J = D(\bar{\varepsilon}_1, \ldots, \bar{\varepsilon}_{\bar{r}}, \bar{M}) I_2 I_1 I_0 D(\varepsilon_1, \ldots, \varepsilon_r, M)^{-1},$$

where $D(\cdot)$ is defined by (7.35). Then $\Psi_{\bar{M}}^{(1)}(\bar{s})$ is the u-decimation of $\Psi_M^{(1)}(s)$ if and only if $\bar{s} = Js$; therefore, J is a matrix over $GF(q)$.

By the way, if the characteristic polynomial $f(z)$ of M has no nonzero repeated root, then $\tilde{l}_h = \bar{l}_h = 1$, $h = 1, \ldots, \bar{r}$. Thus $I_2(h)$ in (7.100) equals $E_{\bar{n}_h}$; therefore I_2 is the identity matrix.

Theorem 7.5.2. *Each sequence in $\Psi_{\bar{M}}^{(1)}$ is u-decimations of $q^{n-\bar{n}}$ sequences in $\Psi_M^{(1)}$, and u-decimations of $\Psi_M^{(1)}(s_1)$ and $\Psi_M^{(1)}(s_2)$ are the same if and only if $Js_1 = Js_2$, where $\bar{n} = \bar{l}_0 + \sum_{h=1}^{\bar{r}} \bar{n}_h \bar{l}_h$.*

Proof. It is easy to see that the rank of I_0 is $n' = \bar{l}_0 + \sum_{h=1}^{\bar{r}} \sum_{i \in P_h} \bar{n}_h l_i$ and the rank of I_1 is $\tilde{n} = \bar{l}_0 + \sum_{h=1}^{\bar{r}} \bar{n}_h \tilde{l}_h$. Since u' and p are coprime, we have $c(u', a, a) = (u')^a \neq 0 \pmod{p}$. Using this fact, noticing $c(u', k', a) = 0$ for $k' < a$, we can prove that the rank of $I_2(h)$ is $\bar{n}_h \bar{l}_h$; therefore, the rank of I_2 is $\bar{n}$. Clearly, $D(\bar{\varepsilon}_1, \ldots, \bar{\varepsilon}_{\bar{r}}, \bar{M})$ and $D(\varepsilon_1, \ldots, \varepsilon_r, M)$ are nonsingular matrices. Using Sylvester inequality, that is, "the rank of G + the rank of H − the number of rows of $H \leqslant$ the rank of $GH \leqslant$ the rank of G, the rank of H", it is easy to see that the rank of J is $\bar{n}$. Since $\bar{n}$ is the number of rows of J, for any state $\bar{s}$ of $\bar{M}$, the equation $\bar{s} = Js$ has $q^{n-\bar{n}}$ solutions. From Corollary 7.5.1, $\Psi_{\bar{M}}^{(1)}(\bar{s})$ is u-decimations of $q^{n-\bar{n}}$ sequences in $\Psi_M^{(1)}$, i.e., $\Psi_M^{(1)}(s)$, $s \in S$, $\bar{s} = Js$.

From Corollary 7.5.1, u-decimations of $\Psi_M^{(1)}(s_1)$ and $\Psi_M^{(1)}(s_2)$ are the same if and only if $Js_1 = Js_2$. □

Corollary 7.5.2. *If $\bar{n} = n$, then each sequence in $\Psi_{\bar{M}}^{(1)}$ is the u-decimation of one sequence in $\Psi_M^{(1)}$; therefore, the u-decimation of any nonzero sequence in $\Psi_{\bar{M}}^{(1)}$ is nonzero.*

It is easy to prove that in the case of $u > 1$, $\bar{n} = n$ holds if and only if the following conditions hold: $\varepsilon_1^u, \ldots, \varepsilon_r^u$ are not conjugate on $GF(q)$ with each other, degrees of minimal polynomials of ε_i and ε_i^u are the same, $i = 1, \ldots, r$, 0 is not a repeated root of $f(z)$, $f(z)$ has no nonzero repeated root or u and p are coprime. In particular, whenever $u > 1$ and $f(z)$ is irreducible other than z, $\bar{n} = n$ holds if and only if degrees of minimal polynomials of ε_1 and ε_1^u are the same.

Historical Notes

Each component sequence of an output sequence of a linear autonomous finite automaton over a finite field is equivalent to a linear shift register sequence over a finite field. A great deal of work has been done on linear shift register sequences (equivalently, linear autonomous finite automata with 1-dimensional output), see [51] and its references for example. In particular, a root representation for linear shift register sequences is found in [54], pp. 20–22. References [97, 98] devote mainly to general (viz. the output dimension $\geqslant 1$) linear autonomous finite automata over finite fields. The material of this chapter is taken out from Appendix III and Chap. 3 of [98], but Theorem 7.3.2 is narrowed and its proof is slightly simplified.

8. One Key Cryptosystems and Latin Arrays

Renji Tao

Institute of Software, Chinese Academy of Sciences
Beijing 100080, China trj@ios.ac.cn

Summary.
In the first seven chapters, theory of finite automata is developed. From now on, some applications to cryptography are presented. This chapter proposes a canonical form for one key cryptosystems in the sense: for any one key cryptosystem without data expansion and with bounded error propagation implementable by a finite automaton, we always find a one key cryptosystem in canonical form such that they are equivalent in behavior. This assertion is affirmative by results concerned on feedforward invertibility in Sects. 1.5 and 5.2. Under the framework of the canonical form, the next is to study its three components: an autonomous finite automaton, a family of permutations, and a nonlinear transformation. Theory of autonomous finite automata has been discussed in the preceding chapter. As to permutational family, theory of Latin arrays, a topic on combinatory theory, is presented in this chapter also.

Key words: *one key cryptosystem, canonical form, Latin array*

In the first seven chapters, theory of finite automata is developed. From now on, some applications to cryptography are presented. This chapter proposes a canonical form for one key cryptosystems in the sense: for any one key cryptosystem without data expansion and with bounded error propagation implementable by a finite automaton, we always find a one key cryptosystem in canonical form such that they are equivalent in behavior. This assertion is affirmative by results concerned on feedforward invertibility in Sects. 1.5 and 5.2. Under the framework of the canonical form, the next is to study its three components: an autonomous finite automaton, a family of permutations, and a nonlinear transformation. Theory of autonomous finite automata has been discussed in the preceding chapter. As to permutational family, theory of Latin arrays, a topic on combinatory theory, is presented in this chapter also.

8.1 Canonical Form for Finite Automaton One Key Cryptosystems

From a mathematical viewpoint, a *cryptographic system*, or *cryptosystem* for short, is a family of transformations $\{f_k, k \in K\}$ depended on a parameter k called the *key*, where K is called the *key space*, f_k is called the cryptographic transformation which is an injective mapping from a set P (the plaintext space) to a set C (the ciphertext space). For sending a message α which is referred to as plaintext through an insecure channel, where it may be tapped by an adversary, using this cryptosystem, the sender first encrypts α by applying f_k to it and then sends the result $f_k(\alpha)$ which is referred to as ciphertext over the channel. The receiver decrypts the ciphertext $f_k(\alpha)$ by applying f_k^{-1} to it and retrieves the plaintext α, where f_k^{-1} is an inverse transformation of f_k. The receiver and the sender share the key k; this cryptosystem is referred to as a one key cryptosysyem.

An example of cryptosystems is the stream cipher of which the key string is a pseudorandom sequence generated by a binary shift register of order n. The key space is the vector space of dimension n over $GF(2)$, the plaintext space and the ciphertext space consist of all words over $GF(2)$, and the cryptographic transformation f_k is defined by $f_k(x_0x_1 \dots x_{l-1}) = y_0y_1 \dots y_{l-1}$, where $y_i = s_i \oplus x_i$, $i = 0, 1, \dots, l-1$, and the key string $s_0s_1 \dots s_{l-1}$ is the first l digits of the output of the shift register for the initial state k. It is well known that shift register sequences have received extensive attentions in cryptology community since the 1950s. Although shift registers are important sequence generators in stream ciphers, they are merely a special kind of autonomous finite automata. Finite automata have been regarded as a natural mathematical model of cryptosystems from an implementing viewpoint, where the plaintext space and the ciphertext space consist of all words over some finite sets, and the cryptographic transformation $f_k(\alpha)$ equals $\lambda(k, \alpha)$, λ being the output function of some weakly invertible finite automaton, and the key space is a set of weakly invertible finite automata and their initial states.

Assume that a finite automaton $M = \langle X, Y, S, \delta, \lambda \rangle$ is chosen as an encoder to implement encryption. Then M must satisfy some conditions on invertibility. In the case where M is invertible with delay τ, we may choose its inverse finite automaton with delay τ, say a τ-order input-memory finite automaton $M' = \langle Y, X, S', \delta', \lambda' \rangle$, as the corresponding decoder to implement decryption. To encrypt a plaintext $x_0 \dots x_{l-1}$ in X^*, we first expand randomly τ letters $x_l, \dots, x_{l+\tau-1}$ in X to its end and choose randomly a state s of M, then compute

$$y_0 \dots y_{l+\tau-1} = \lambda(s, x_0 \dots x_{l+\tau-1}).$$

$y_0 \dots y_{l+\tau-1}$ is a ciphertext of $x_0 \dots x_{l-1}$. For decryption, we compute

$$x'_0 \dots x'_{l+\tau-1} = \lambda'(s', y_0 \dots y_{l+\tau-1})$$

for any state s' of M', then the plaintext $x_0 \dots x_{l-1}$ equals $x'_\tau \dots x'_{l+\tau-1}$. In this case, the key is the structure of M. In general, the variable structure of the encoder leads its implementation to inconvenient. In the case where M is weakly invertible with delay τ, we may choose its weak inverse finite automaton with delay τ, say $M' = \langle Y, X, S', \delta', \lambda' \rangle$, as the corresponding decoder. To encrypt a plaintext $x_0 \dots x_{l-1}$ in X^*, we first expand randomly τ letters $x_l, \dots, x_{l+\tau-1}$ in X to its end, then compute

$$y_0 \dots y_{l+\tau-1} = \lambda(s, x_0 \dots x_{l+\tau-1}).$$

$y_0 \dots y_{l+\tau-1}$ is a ciphertext of $x_0 \dots x_{l-1}$. For decryption, we compute

$$x'_0 \dots x'_{l+\tau-1} = \lambda'(s', y_0 \dots y_{l+\tau-1}),$$

where the state s' of M' τ-matches s with delay τ. Then the plaintext $x_0 \dots x_{l-1}$ equals $x'_\tau \dots x'_{l+\tau-1}$. In this case, the key is the state s of M if the structure of M is fixed.

In invertible case, since M' is an input-memory finite automaton, an error letter in cipher causes at most $\tau + 1$ error letters in decryption. But in weakly invertible case, sometimes an error letter in cipher can cause infinite error letters in decryption as pointed out in p.35. From Theorem 1.5.2, to guarantee bounded propagation of decoding errors, encoders must be feedforward invertible and their feedforward inverses are taken as the corresponding decoders.

From Theorem 1.4.5, if $M = \langle X, Y, S, \delta, \lambda \rangle$ is taken as an encoder, then $|Y| \geqslant |X|$. To represent all ciphertexts for all plaintexts of length l ($l \log_2 |X|$ bits), we need $(l+\tau) \log_2 |Y|$ bits. Thus $l \log_2 |X| \leqslant (l+\tau) \log_2 |Y|$. Therefore, there is no plaintext expansion if and only if $l \log_2 |X| = (l + \tau) \log_2 |Y|$, if and only if $|Y| = |X|$ and the delay step $\tau = 0$.

For one key cryptosystems implemented by finite automata without plaintext expansion and with bounded propagation of decoding errors, decoders may be chosen from weakly inverse semi-input-memory finite automata with delay 0 in which the input alphabet and the output alphabet of a finite automaton have the same size. Theorem 5.2.2 characterizes the structure of feedforward inverses with delay 0; using this result, we give a canonical form of such cryptosystems as follows.

The decoder $M' = \langle Y, X, S', \delta', \lambda' \rangle$ is a c-order semi-input-memory finite automaton $\mathcal{SIM}(M_a, f)$, where $X = Y$, $M_a = \langle Y_a, S_a, \delta_a, \lambda_a \rangle$ is an autonomous finite automaton, f is a single-valued mapping from $Y^{c+1} \times \lambda_a(S_a)$

to X with $|f(Y, y_{c-1}, \ldots, y_0, \lambda_a(s_a))| = |X|$ for any $s_a \in S_a$ and any $y_0, \ldots, y_{c-1} \in Y$. For any $y_a \in \lambda_a(S_a)$ and any $y_0, \ldots, y_{c-1} \in Y$, let $f_{y_{c-1},\ldots,y_0,y_a}$ be a single-valued mapping from Y to X defined by

$$f_{y_{c-1},\ldots,y_0,y_a}(y_c) = f(y_c, \ldots, y_0, y_a),\ y_c \in Y.$$

Clearly, $f_{y_{c-1},\ldots,y_0,y_a}$ is a permutation on Y (or X). Then there exists a single-valued mapping h from $Y^c \times \lambda_a(S_a)$ to W such that

$$\begin{aligned} f(y_c, y_{c-1}, \ldots, y_0, y_a) &= g^{-1}_{h(y_{c-1},\ldots,y_0,y_a)}(y_c), \\ y_a \in \lambda_a(S_a),\ & y_0, \ldots, y_c \in Y \end{aligned}$$

for some finite set W, where g_w^{-1} is a bijection from Y to X, for any w in W. Fig.8.1.1 (b) gives a pictorial form of the decoder M'. For any initial state $s'_0 = \langle y_{-1}, \ldots, y_{-c}, s_{a0}\rangle$ and any input sequence (ciphertext) $y_0 \ldots y_{l-1}$ of M', the output sequence (plaintext) $x_0 \ldots x_{l-1}$ of M' can be computed by

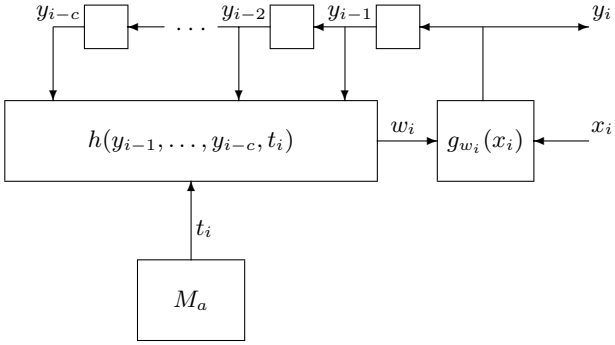
$$\begin{aligned} &s_{a,i+1} = \delta_a(s_{ai}), \\ &t_i = \lambda_a(s_{ai}), \\ &w_i = h(y_{i-1}, \ldots, y_{i-c}, t_i), \\ &x_i = g^{-1}_{w_i}(y_i), \\ &\quad i = 0, 1, \ldots, l-1. \end{aligned}$$

Among others, a corresponding encoder may be chosen as a finite automaton $M = \langle X,\ Y,\ Y^c \times S_a,\ \delta,\ \lambda\rangle$, of which a pictorial form is given by Fig.8.1.1 (a), where

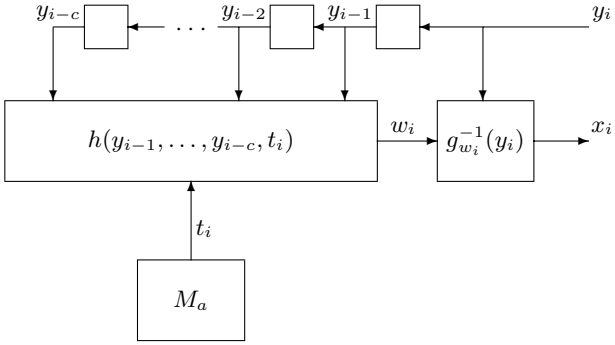
$$\begin{aligned} &\delta(\langle y_{-1}, \ldots, y_{-c}, s_a\rangle,\ x_0) = \langle y_0, y_{-1}, \ldots, y_{-c+1}, \delta_a(s_a)\rangle, \\ &\lambda(\langle y_{-1}, \ldots, y_{-c}, s_a\rangle,\ x_0) = y_0, \\ &\quad w_0 = h(y_{-1}, \ldots, y_{-c}, \lambda_a(s_a)), \\ &\quad y_0 = g_{w_0}(x_0), \\ &\quad \langle y_{-1}, \ldots, y_{-c}, s_a\rangle \in Y^c \times S_a,\ x_0 \in X. \end{aligned}$$

That is to say, for any initial state $s_0 = \langle y_{-1}, \ldots, y_{-c}, s_{a0}\rangle$ and any input sequence (plaintext) $x_0 \ldots x_{l-1}$ of M, the output sequence (ciphertext) $y_0 \ldots y_{l-1}$ of M can be computed by

$$\begin{aligned} &s_{a,i+1} = \delta_a(s_{ai}), \\ &t_i = \lambda_a(s_{ai}), \\ &w_i = h(y_{i-1}, \ldots, y_{i-c}, t_i), \\ &y_i = g_{w_i}(x_i), \\ &\quad i = 0, 1, \ldots, l-1. \end{aligned}$$



(a) Encoder M



(b) Decoder M'

Figure 8.1.1

Other decoders are possible; results in Sect. 6.6 of Chap. 6 show that, from a behavior viewpoint, they are slightly different from the above decoder.

On the other hand, other encoders are possible. We may even take nondeterministic encoders from results in Sect. 6.5 of Chap. 6. But they are more complex than the above encoder from a structural viewpoint.

As a special case ($c = 0$), for one key cryptosystems implemented by finite automata without expansion of the plaintext and without propagation of decoding errors, the canonical form is as follows.

The decoder $M' = \langle Y, X, S_a, \delta', \lambda' \rangle$ is a 0-order semi-input-memory finite automaton $\mathcal{SIM}(M_a, g')$, where $X = Y$,

$$\begin{aligned} \delta'(s_a, y) &= \delta_a(s_a), \\ \lambda'(s_a, y) &= g_w^{-1}(y), \\ w &= \lambda_a(s_a), \end{aligned}$$

$$s_a \in S_a,\ y \in Y,$$

$M_a = \langle W, S_a, \delta_a, \lambda_a \rangle$ is an autonomous finite automaton, g_w^{-1} is a bijection from Y to X for any w in W, and $g_w^{-1}(y) = g'(y, w)$. For any initial state s_{a0} and any input sequence (ciphertext) $y_0 \dots y_{l-1}$ of M', the output sequence (plaintext) $x_0 \dots x_{l-1}$ of M' can be computed by

$$\begin{aligned} s_{a,i+1} &= \delta_a(s_{ai}), \\ w_i &= \lambda_a(s_{ai}), \\ x_i &= g_{w_i}^{-1}(y_i), \\ & i = 0, 1, \dots, l-1. \end{aligned}$$

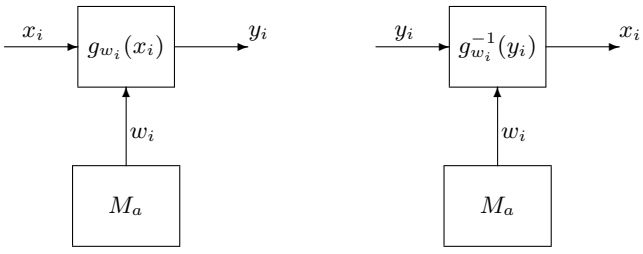
A corresponding encoder may be chosen as a finite automaton $M = \langle X, Y, S_a, \delta, \lambda \rangle$, where $X = Y$,

$$\begin{aligned} \delta(s_a, x) &= \delta_a(s_a), \\ \lambda(s_a, x) &= g_w(x), \\ w &= \lambda_a(s_a), \\ & s_a \in S_a,\ x \in X. \end{aligned}$$

That is to say, $M = \langle X, Y, S_a, \delta, \lambda \rangle$ is also a 0-order semi-input-memory finite automaton $\mathcal{SIM}(M_a, g)$, where $g(x, w) = g_w(x)$. For any initial state s_{a0} and any input sequence (plaintext) $x_0 \dots x_{l-1}$ of M, the output sequence (ciphertext) $y_0 \dots y_{l-1}$ of M can be computed by

$$\begin{aligned} s_{a,i+1} &= \delta_a(s_{ai}), \\ w_i &= \lambda_a(s_{ai}), \\ y_i &= g_{w_i}(x_i), \\ & i = 0, 1, \dots, l-1. \end{aligned}$$

Fig. 8.1.2 gives a pictorial form of the canonical form.



(a) Encoder M (b) Decoder M'

Figure 8.1.2

Block ciphers and stream ciphers (in the narrow sense) are special cases of the above canonical form. For block ciphers, δ_a is the identity function. For binary stream ciphers, $g_w(v) = g_w^{-1}(v) = w \oplus v$.

Example 8.1.1. To give a cipher pictorialized by Fig. 8.1.1, let X and Y be the set of all 8 bits 0,1 strings. Take $c = 6$. M_a consists of a binary shift register with characteristic polynomial $x^{128} \oplus x^8 \oplus x$ and an autonomous finite automaton M_I of which the next state function is the identity function. t_i is $\langle s_i, \varphi \rangle$, where s_i is the state of the shift register and φ is the output of M_I which represents an involution (i.e., $\varphi^{-1} = \varphi$) of 8 bits 0,1 strings. w_i is $\langle w_{i1}, w_{i2}, \varphi \rangle$, where w_{i1} and w_{i2} are 8 bits 0,1 strings. $g_{\langle w_1, w_2, \varphi \rangle}(x)$ is $\varphi(w_1 - (w_2 \oplus (w_1 - \varphi(x))))$, where $-$ stands for subtraction modulo 256; therefore, $g^{-1}_{\langle w_1, w_2, \varphi \rangle}(y) = g_{\langle w_1, w_2, \varphi \rangle}(y)$. The key consists of the initial state of M_a.

If the characteristic polynomial of the binary shift register is variable which may be taken as the product of x and any primitive polynomial of degree 127 over $GF(2)$, then the key consists of the initial state of M_a and the characteristic polynomial, in other words, the key consists of M_a and its initial state. Formally, the structure of M_a is variable; but after redefining the autonomous finite automaton M_a by expanding its state to include the coefficient of the characteristic polynomial, the key still consists of the initial state of M_a.

8.2 Latin Arrays

8.2.1 Definitions

The problem of designing one key cryptosystems implemented by finite automata without plaintext expansion and with bounded propagation of decoding errors may be reduced to choosing suitable parameters such as the size of alphabets and the length c of the ciphertext history and designing three components in the above canonical form (Fig.8.1.1) – an autonomous finite automaton M_a, a transformation h and a permutational family $\{g_w, w \text{ in } W\}$ – so that the systems are both efficient and secure.

Assume that the distribution of elements in the derived key sequence w_0 w_1 ... in the above canonical form is uniform. Let $\{g_w, w \text{ in } W\}$ be a family of permutations on X. For resisting the known plaintext attack, under the above assumption, the requirement in Property 1 is very natural.

Property 1. For any x, y in X, $|\{w | w \text{ in } W, g_w(x) = y\}| = \text{constant}$.

From the viewpoint of uniformity for permutations, the following property is also desired.

Property 2. For any w' in W, $|\{w|w \text{ in } W, g_w = g_{w'}\}| = \text{constant}$.

Specify an order for elements of X, say $x_1, \ldots, x_n$, and an order for elements of W, say $w_1, \ldots, w_m$. Let A be an $n \times m$ matrix, of which the element at row i and column j is $g_{w_j}(x_i)$. Then each column of A is a permutation of elements in X. Clearly, fixing orders of elements for X and W, the family of permutations $\{g_w, w \text{ in } W\}$ is one-to-one correspondent with A. Corresponding to Property 1, we introduce the following concept.

Let A be an $n \times nk$ matrix over $N = \{1, \ldots, n\}$. If each element of N occurs exactly once in each column of A and k times in each row of A, A is called an (n, k)-*Latin array*.

Corresponding to Properties 1 and 2, we introduce the following concept.

Let A be an (n, k)-Latin array. If each column of A occurs exactly r times in columns of A, A is called an (n, k, r)-*Latin array*.

Notice that $(n, 1)$-Latin arrays are n-order Latin squares in literature.

Let A and B be $n \times m$ matrices over N. If B can be obtained from A by rearranging rows, rearranging columns and renaming elements, we say that A and B are *isotopic*; and the transformation $\langle \alpha, \beta, \gamma \rangle$ is called an *isotopism* from A to B, where α, β and γ are the row arranging, the renaming and the column arranging, respectively. It is easy to verify that the isotopy relation is reflexive, symmetric and transitive. Clearly, if A is an (n, k)-Latin array and if A and B are isotopic, then B is an (n, k)-Latin array. Similarly, if A is an (n, k, r)-Latin array and if A and B are isotopic, then B is an (n, k, r)-Latin array.

For (n, k)-Latin arrays or (n, k, r)-Latin arrays, any equivalence class of the isotopy relation is also called an *isotopy class*.

8.2.2 On (n, k, r)-Latin Arrays

We use $U(n, k)$ to denote the number of all (n, k)-Latin arrays, $U(n, k, r)$ the number of all (n, k, r)-Latin arrays, $I(n, k)$ the number of all isotopy classes of (n, k)-Latin arrays, and $I(n, k, r)$ the number of all isotopy classes of (n, k, r)-Latin arrays.

Let A_i be an $n \times m_i$ matrix, $i = 1, \ldots, t$. The $n \times (m_1 + \cdots + m_t)$ matrix $[A_1, \ldots, A_t]$ is called the *concatenation* of $A_1, \ldots, A_t$. The concatenation of t identical matrices A is called the t-fold concatenation of A, denoted by $A^{(t)}$.

It is easy to see that the concatenation $[A_1, \ldots, A_t]$ of the (n, k_i)-Latin array A_i, $i = 1, \ldots, t$ is an $(n, k_1 + \cdots + k_t)$-Latin array.

Let A and B be $n \times m$ matrices over N. If B can be obtained from A by rearranging A's columns, we say that A and B are *column-equivalent*.

Clearly, the column-equivalence relation is reflexive, symmetric, and transitive. For (n, k)-Latin arrays or (n, k, r)-Latin arrays, the equivalence classes of the column-equivalence relation are called the *column-equivalence classes*.

For any matrix A, we use $b(A)$ to denote the matrix obtained from A by deleting repeated columns but the leftmost ones.

Lemma 8.2.1. (a) *Let A be an $(n, k, 1)$-Latin array. Then $A^{(r)}$ is an (n, k, r)-Latin array.*

(b) *Let A be an (n, k, r)-Latin array. Then $r|k$, $b(A)$ is an $(n, k/r, 1)$-Latin array, and A and the r-fold concatenation of $b(A)$ are isotopic and column-equivalent.*

Proof. (a) From the definition, the result is evident.

(b) From the definitions, it is easy to see that A and $b(A)^{(r)}$, the r-fold concatenation of $b(A)$, are column-equivalent; therefore, A and $b(A)^{(r)}$ are isotopic. Let k' be the number of occurrences of an element y in row i of $b(A)$. Then the number of occurrences of the element y in row i of $b(A)^{(r)}$ is rk'. Since $b(A)^{(r)}$ and A are column-equivalent and A is an (n, k, r)-Latin array, $b(A)^{(r)}$ is an (n, k, r)-Latin array. It follows that $k = rk'$. Thus $r|k$. Since $k' = k/r$, k' is independent of y. Therefore, $b(A)$ is an $(n, k/r, 1)$-Latin array. □

Lemma 8.2.2. *Let A and B be two (n, k, r)-Latin arrays.*

(a) *A and B are isotopic if and only if $b(A)$ and $b(B)$ are isotopic.*

(b) *A and B are column-equivalent if and only if $b(A)$ and $b(B)$ are column-equivalent.*

Proof. (a) Suppose that $b(A)$ and $b(B)$ are isotopic. It is easy to see that $b(A)^{(r)}$ and $b(B)^{(r)}$ are isotopic. From Lemma 8.2.1(b), $b(A)^{(r)}$ and A are isotopic, and $b(B)^{(r)}$ and B are isotopic. Therefore, A and B are isotopic.

Conversely, suppose that A and B are isotopic. Then there is an isotopism $\langle \alpha, \beta, \gamma \rangle$ from A to B. Let A' be the result obtained by applying row arranging α and renaming β to A. Then B can be obtained by applying column arranging γ from A'. Clearly, columns i and j of A are the same if and only if columns i and j of A' are the same. This yields that if $b(A)$ consists of columns $j_1, \ldots, j_{k/r}$ of A, then $b(A')$ consists of columns $j_1, \ldots, j_{k/r}$ of A'. Thus applying row arranging α and renaming β to $b(A)$ results $b(A')$. Since A' and B are column-equivalent, it is easy to see that $b(A')$ and $b(B)$ are column-equivalent. It follows that $b(A)$ and $b(B)$ are isotopic.

(b) The proof of part (b) is similar to part (a). □

Theorem 8.2.1. (a) $I(n, k, r) = I(n, k/r, 1)$.

(b) $U(n, k, r) = U(n, k/r, 1)(nk)!/((nk/r)!(r!)^{nk/r})$.

Proof. (a) We define a mapping φ from the isotopy classes of $(n, k/r, 1)$-Latin arrays to the isotopy classes of (n, k, r)-Latin arrays by taking $\varphi(C)$ as the isotopy class containing $A^{(r)}$, where A is an arbitrary element in C. From Lemma 8.2.1(a), $A^{(r)}$ is an (n, k, r)-Latin array. Noticing $b(A^{(r)}) = A$ for any $(n, k, 1)$-Latin array A, from Lemma 8.2.2 (a), it is easy to show that φ is single-valued and injective. From Lemma 8.2.1 (b), φ is surjective. Thus we have $I(n, k, r) = I(n, k/r, 1)$.

(b) Let C be an isotopy class of $(n, k/r, 1)$-Latin array, and $C' = \varphi(C)$. Since no column occurs repeatedly within any Latin array in C, each column-equivalence class of C has $(nk/r)!$ elements. Denote the number of column-equivalence classes of C by x. Then the number of Latin arrays in C is $|C| = (nk/r)!x$.

We define a mapping ψ from the column-equivalence classes of C to the column-equivalence classes of C' by taking $\psi(D)$ as the column-equivalence class containing $A^{(r)}$, where A is an arbitrary element in D. From Lemma 8.2.2(b), it is easy to show that ψ is single-valued and injective. From Lemma 8.2.1(b), ψ is surjective. Thus the number of column-equivalence classes of C is equal to the number of column-equivalence classes of C'.

For any column-equivalence class D' of C', it is easy to see that all elements of D' can be obtained from an arbitrary specific element of D' by rearranging columns. Since any Latin array in D' has nk columns and each column occurs exactly r times, there are $(nk)!$ ways to rearrange columns of a specific Latin array of D', and there are exactly $(r!)^{nk/r}$ ways generating the same result. Therefore, the number of elements in D' is $(nk)!/(r!)^{nk/r}$.

Using proven results, we conclude that the number of Latin arrays in C' is

$$\begin{aligned} |C'| &= ((nk)!/(r!)^{nk/r})x \\ &= ((nk)!/(r!)^{nk/r})|C|/(nk/r)! \\ &= |C|(nk)!/((nk/r)!(r!)^{nk/r}). \end{aligned}$$

From (a), it follows that $U(n, k, r) = U(n, k/r, 1)(nk)!/((nk/r)!(r!)^{nk/r})$. □

Let A be an $(n, k, 1)$-Latin array, and A' an $(n, (n-1)! - k, 1)$-Latin array. If the columns of the concatenation of A and A' consist of all permutations on N, A' is called a *complement* of A.

Clearly, if A' is a complement of A, then A is a complement of A'. Any two complements of A are column-equivalent.

Lemma 8.2.3. *Let A_i be an $(n, k, 1)$-Latin array, and A'_i a complement of A_i, $i = 1, 2$.*

(a) *A_1 and A_2 are isotopic if and only if A'_1 and A'_2 are isotopic.*

(b) *A_1 and A_2 are column-equivalent if and only if A'_1 and A'_2 are column-equivalent.*

Proof. (a) Suppose that A_1 and A_2 are isotopic. Then there is an isotopism from A_1 to A_2, say $\langle \alpha, \beta, \gamma \rangle$. Let γ' be a column arranging of $n \times (n!)$ matrices so that the restriction of γ' on the first nk columns is γ and γ' keeps the last $n! - nk$ columns unchanged. Let $[A_3, A'_3]$ be the result of applying the transformation $\langle \alpha, \beta, \gamma' \rangle$ to $[A_1, A'_1]$. Then we have $A_3 = A_2$ and that $\langle \alpha, \beta, e \rangle$ is an isotopism from A'_1 to A'_3, where e stands for the identical transformation. Since the columns of $[A_1, A'_1]$ consist of all permutations on N, the columns of $[A_3, A'_3]$ consist of all permutations on N. Thus A'_3 is a complement of A_2. If follows that there exists a column arranging γ'' from A'_3 to A'_2. Thus $\langle \alpha, \beta, \gamma'' \rangle$ is an isotopism from A'_1 to A'_2. Therefore, A'_1 and A'_2 are isotopic.

From symmetry, if A'_1 and A'_2 are isotopic, then A_1 and A_2 are isotopic.

(b) From the proof of (a), taking α and β as the identity transformation results a proof of (b). □

Theorem 8.2.2. *Let $1 \leqslant k < (n-1)!$.*

(a) $I(n, k, 1) = I(n, (n-1)! - k, 1)$.

(b) $U(n, (n-1)! - k, 1) = U(n, k, 1)(n! - nk)!/(nk)!$.

(c) $I(n, (n-1)!, 1) = 1, U(n, (n-1)!, 1) = (n!)!$.

Proof. (a) We define a mapping φ from the isotopy classes of $(n, k, 1)$-Latin arrays to the isotopy classes of $(n, (n-1)! - k, 1)$-Latin arrays so that φ maps the isotopy class containing A to the isotopy class containing a complement of A. From Lemma 8.2.3(a), φ is single-valued and injective. Since complements of any $(n, (n-1)! - k, 1)$-Latin array are existent, φ is surjective. Therefore, we have $I(n, k, 1) = I(n, (n-1)! - k, 1)$.

(b) For any isotopy class of $(n, k, 1)$-Latin arrays C, let $C' = \varphi(C)$. Clearly, the number of elements of any column-equivalence class of C is $(nk)!$. Denote the number of column-equivalence classes of C by x. Then the number of Latin arrays in C is $|C| = (nk)!x$.

We define a mapping ψ from the column-equivalence classes of C to the column-equivalence classes of C' so that ψ maps the column-equivalence class containing A to the column-equivalence class containing a complement of A. From Lemma 8.2.3(b), ψ is single-valued and injective. Since for each $(n, (n-k)! - k, 1)$-Latin array in C' there is an $(n, k, 1)$-Latin array in C as its complement, ψ is surjective. Therefore, the number of column-equivalence classes of C is equal to the number of column-equivalence classes of C'. Clearly, the number of elements of any column-equivalence class of C' is $(n! - nk)!$. It follows that the number of Latin arrays in C' is $|C'| = (n! - nk)!x$.

Using proven results, we conclude that the number of Latin arrays in C' is $|C'| = (n! - nk)!x = (n! - nk)!(|C|/(nk)!) = |C|(n! - nk)!/(nk)!$. From (a), it follows that $U(n, (n-1)! - k, 1) = U(n, k, 1)(n! - nk)!/(nk)!$.

(c) Evident. □

8.2.3 Invariant

Let A be an (n, k)-Latin array.

For any column in a matrix, the multiplicity of the column means the occurrence number of the column in the matrix. We use c_i to denote the number of distinct columns of A with multiplicity i, for $i = 1, \ldots, k$. $c_k c_{k-1} \ldots c_2$ is called the *column characteristic value* of A.

For any sequence $(x_1, \ldots, x_k)$, x_i taking value from an arbitrary set with $n - 1$ elements, $n_k n_{k-1} \ldots n_2$ is called the *type* of the sequence, where n_j is the number of distinct x_i's with multiplicity j. In the case of $k = 2$, possible types are 1 and 0 which are referred to as *twins* and *all different*, respectively. In the case of $k = 3$, possible types are 10, 01 and 00 which are referred to as *trio*, *twins* and *all different*, respectively. In the case of $k = 4$, possible types are 100, 010, 002 and 001 which are referred to as *quad*, *trio*, *double twins* and *twins*, respectively.

For any different i and j, we use $A(i, j, a)$ to denote the j-th row of the submatrix consisting of A's columns of which the elements at row i are a. Let c_t be the number of a, $1 \leqslant a \leqslant n$, such that the type of $A(i, j, a)$ is t; denote $T_1(i, j) = \sum t \cdot c_t$, t ranging over all types. Noting $\sum c_t = n$, any c_h can be determined by other c_t's. Fixing a permutation of all types, say $t_r, \ldots, t_1$, $T_1(i, j)$ is also represented by $c_{t_r} c_{t_{r-1}} \ldots c_{t_2}$. For example, in the case of $(4, 2)$-Latin array, we permute types as 1, 0 and represent $T_1(i, j)$ by c_1; in the case of $(4, 3)$-Latin array, we permute types as 10, 01, 00 and represent $T_1(i, j)$ by $c_{10} c_{01}$; in the case of $(4, 4)$-Latin array, we permute types as 100, 010, 002, 001 and represent $T_1(i, j)$ by $c_{100} c_{010} c_{002}$. Given different i and j, $i \neq j$, for any $a, 1 \leqslant a \leqslant n$, if in the type $n_k n_{k-1} \ldots n_2$ of $A(i, j, a)$ the nonzero n_h with the maximal subscript h takes value 1, then we define $\pi(a)$ as the element in $A(i, j, a)$ with the maximal multiplicity. If the mapping π is bijective, π is called the *derived permutation* from row i to row j, denoted by $\pi(i, j)$. A derived permutation can be expressed as a product of disjoint cycles of length > 1. The distribution of these lengths of cycles is called the *type* of the derived permutation, denoted by $T_2(i, j)$. If the derived permutation does not exist and if the maximal multiplicity of elements occurring in $A(i, j, 1), \ldots, A(i, j, n)$, say r, is great than $k/2$, $|I \cap J|$ is called the *intersection number* from row i to row j, denoted by $T_3(i, j)$, where $I = \{a \mid a \in N$, the maximal multiplicity of elements in $A(i, j, a)$ is $r\}$, and $J = \{b \mid$ there

is an $a \in I$, such that the multiplicity of b in $A(i, j, a)$ is $r\}$. Let $T(i, j) = (T_1(i, j), T_2(i, j), T_3(i, j))$; $T_2(i, j)$ and $T_3(i, j)$ may be undefined. The set consisting of $T(i, j)$, $i, j = 1, \ldots, n, i \neq j$ (repetition allowable) is called the *row characteristic set* of A. Let GR_A be a (directed) graph with vertex set N and arc set $(N \times N) \setminus \{(i, i), i \in N\}$, of which each arc, say (i, j), is labelled by $T(i, j)$; GR_A is called the *row characteristic graph* of A. GR_A is said to be *symmetric*, if $T(i, j) = T(j, i)$ holds for any $i \neq j$; it is considered as undirected, that is, the two arcs (i, j) and (j, i) are merged into an *edge* with *endpoints* i and j, which is also denoted by (i, j) or (j, i).

Theorem 8.2.3. *Let A and B be two (n, k)-Latin arrays. If A and B are isotopic, then* (a) *the column characteristic values of A and B are the same,* (b) *the row characteristic graphs of A and B are isomorphic, and* (c) *the row characteristic sets of A and B are the same.*

Proof. (a) Since the identity between two columns keeps unchanged under row arrangings and renamings and the column characteristic value keeps unchanged under column arrangings, the column characteristic values of A and B are the same.

(b) Since the type of a sequence $(x_1, \ldots, x_k)$ keeps unchanged under column arrangings and renamings, for any two rows i and j of an (n, k)-Latin array, $T_1(i, j)$ keeps unchanged under column arrangings and renamings. Clearly, $\pi(i, j)$ keeps unchanged under column arrangings, and a renaming for A yields the same renaming for $\pi(i, j)$. Thus $T_2(i, j)$ keeps unchanged under column arrangings and renamings. It is easy from the definition to prove that $T_3(i, j)$ keeps unchanged under column arrangings and renamings. Let $\langle \alpha, \beta, \gamma \rangle$ be an isotopism from A to B. Let A' be the result of transforming A by renaming β and column arranging γ. From the above results, GR_A and $GR_{A'}$ are the same. Since row arranging α transforms A' to B, α transforms GR_A $(= GR_{A'})$ to GR_B. It follows that the row characteristic graphs of A and B are isomorphic.

(c) Immediately obtained from (b). □

Corollary 8.2.1. *For any two (n, k)-Latin arrays, if the column characteristic values or the row characteristic sets of them are distinct, then they are not isotopic.*

Theorem 8.2.4. *For $n = 4, 2 \leqslant k \leqslant 4$, the row characteristic graph of any (n, k)-Latin array is symmetric.*

Proof. Let A be a $(4, k)$-Latin array. For any i, j, $1 \leqslant i, j \leqslant 4$, $i \neq j$, we use A_{ij} to denote the submatrix of A consisting of its rows i, j.

Case $k = 2$: For any i, j, $1 \leqslant i, j \leqslant 4$, $i \neq j$, let c be the number of different columns of A_{ij} with column multiplicity > 1. It is easy to see that

$T_1(i,j) = T_1(j,i) = 1 \cdot c + 0 \cdot (4-c)$. Whenever $c = 4$, $\pi(i,j)$ and $\pi(j,i)$ exist and $\pi(j,i)$ is the inverse permutation of $\pi(i,j)$; therefore $T_2(i,j) = T_2(j,i)$. Whenever $c < 4$, $\pi(i,j)$ and $\pi(j,i)$ do not exist. In the case of $0 < c < 4$, it is easy to see that $T_3(i,j)$ is equal to the number of the elements in the intersection of the elements in the two rows of the submatrix of A_{ij} consisting of columns with column multiplicity 2. Thus $T_3(i,j) = T_3(j,i)$. To sum up, we obtain $T(i,j) = T(j,i)$.

Case $k = 3$: For any i, j, $1 \leqslant i, j \leqslant 4$, $i \neq j$, it is easy to see that the number of different columns of A_{ij} with column multiplicity 3 is equal to the the number of A_{ija}, $a = 1, \ldots, 4$ with type trio and that the number of different columns of A_{ij} with column multiplicity 2 is equal to the number of $A_{ija}, a = 1, \ldots, 4$ with type twins. Since A_{ij} and A_{ji} are the same, we have $T_1(i,j) = T_1(j,i)$. Let c_3 (respectively c_2) be the numbers of $A_{ija}, a = 1, \ldots, 4$ with type trio (respectively twins). Clearly, $\pi(i,j)$ is existent if and only if $c_3 + c_2 = 4$. Thus $\pi(i,j)$ is existent if and only if $\pi(j,i)$ is existent, and $\pi(j,i)$ is the inverse permutation of $\pi(i,j)$ whenever they are existent. It follows that $T_2(i,j) = T_2(j,i)$ whenever $c_3 + c_2 = 4$. In the case of $0 < c_3 + c_2 < 4$, let A'_{ij} be the submatrix of A_{ij} consisting of its columns with column multiplicity 3 if $c_3 \neq 0$, with column multiplicity 2 if $c_3 = 0$. It is easy to see that $T_3(i,j)$ is equal to the number of the elements in the intersection of the elements in the two rows of A'_{ij}. Thus $T_3(i,j) = T_3(j,i)$. To sum up, we obtain $T(i,j) = T(j,i)$.

Case $k = 4$: For any i, j, $1 \leqslant i, j \leqslant 4$, $i \neq j$, it is easy to see that the number of different columns of A_{ij} with column multiplicity 4 (respectively 3) is equal to the the number of A_{ija}, $a = 1, \ldots, 4$ with type quad (respectively trio). Since A_{ij} and A_{ji} are the same, the number of A_{ija}, $a = 1, \ldots, 4$ with type quad (respectively trio) and the number of A_{jia}, $a = 1, \ldots, 4$ with type quad (respectively trio) are the same. We prove the following proposition. If A_{ija} has type double twins and consists of b, b, c, c, and if A_{jib} and A_{jic} have no type double twins, then A_{jib} and A_{jic} have type twins and one of the following conditions holds: (a) A_{ijb} and A_{ijd} have type twins, A_{ijc} has type trio, A_{jia} has type double twins, and A_{jid} has type trio; (b) A_{ijb} has type trio, A_{ijc} and A_{ijd} have type twins, A_{jia} has type double twins, and A_{jid} has type trio; (c) A_{ijb}, A_{ijc}, A_{ijd} and A_{jia} have type twins, and A_{jid} has type double twins, where $\{a, b, c, d\} = \{1, 2, 3, 4\}$. In fact, since A_{ija} consists of b, b, c, c and A_{jib} (respectively A_{jic}) has no type double twins, A_{jib} (respectively A_{jic}) consists of a, a, c, d (respectively a, a, b, d). Since d does not occur in A_{jid} and occurs 4 times in each row of A, A_{jia} consists of d, d, b, b, or d, d, c, c, or d, d, b, c. In the case where A_{jia} consists of d, d, b, b, A_{jid} consists of b, c, c, c; therefore, the condition (a) holds. In the case where A_{jia} consists of d, d, c, c, A_{jid} consists of b, b, b, c; therefore, the condition (b) holds. In the case where

A_{jia} consists of d, d, b, c, A_{jid} consists of b, b, c, c; therefore, the condition (c) holds. Let $S_2(i,j) = \{a|\ A_{ija}$ has type double twins, $a = 1, \ldots, 4\}$ and $n_2(i,j) = |S_2(i,j)|$. From the proposition, for any i, j, $1 \leqslant i, j \leqslant 4$, $i \neq j$, if $n_2(i,j) = 1$, then $n_2(j,i) \geqslant 1$. We prove that for any i, j, $1 \leqslant i, j \leqslant 4$, $i \neq j$, if $n_2(i,j) = 2$, then $n_2(j,i) \geqslant 2$. In fact, suppose that A_{ija} has type double twins and consists of b, b, c, c and that A_{ije} has type double twins and consists of f, f, g, g, where $\{a, b, c, d\} = \{1, 2, 3, 4\}$ and $e \in \{b, c, d\}$. Consider the intersection set $S = \{b, c\} \cap \{f, g\}$. In the case of $|S| = 2$, we have $\{b, c\} = \{f, g\}$. Thus A_{jib} and A_{jic} have type double twins. Therefore, $n_2(j,i) \geqslant 2$. In the case of $|S| = 1$, without loss of generality, we suppose that $S = \{b\}$. It follows that $\{f, g\} = \{a, b\}$ or $\{f, g\} = \{b, d\}$. Whenever $\{f, g\} = \{a, b\}$, we have $e = c$ or $e = d$. We prove $e \neq c$ by reduction to absurdity. Suppose to the contrary that $e = c$. Since A_{ijd} does not contain d and b, A_{ijd} consists of a, a, c, c. Thus $n_2(i,j) \geqslant 3$. This contradicts $n_2(i,j) = 2$. Thus we have $e = d$. Since A_{ijc} contains no c, A_{ijb} has two occurrences of c. Therefore, A_{jib} and A_{jic} have type double twins; that is, $n_2(j,i) \geqslant 2$. Whenever $\{f, g\} = \{b, d\}$, we have $e = c$. Since A_{ijd} contains no d, A_{ijb} has two occurrences of d. Therefore, A_{jib} and A_{jid} have type double twins; that is, $n_2(j,i) \geqslant 2$. In the case of $|S| = 0$, we have $\{f, g\} = \{a, d\}$. It follows that $e = b$ or $e = c$. Without loss of generality, we suppose that $e = b$. Since A_{ijc} contains no c, A_{ijd} has two occurrences of c; since A_{ijd} contains no d, A_{ijc} has two occurrences of d. Therefore, A_{jic} and A_{jid} have type double twins; that is, $n_2(j,i) \geqslant 2$. This completes the proof of that $n_2(i,j) = 2$ implies $n_2(j,i) \geqslant 2$. It is easy to prove that for any i, j, $1 \leqslant i, j \leqslant 4$, $i \neq j$, if $n_2(i,j) = 3$ and $\{1, 2, 3, 4\} \setminus S_2(i,j) = \{a'\}$, then $A_{ija'}$ has type quad; therefore, $n_2(j,i) = 3$. Clearly, for any i, j, $1 \leqslant i, j \leqslant 4$, $i \neq j$, if $n_2(i,j) = 4$, then $n_2(j,i) = 4$. To sum up, for any i, j, $1 \leqslant i, j \leqslant 4$, $i \neq j$, we have $n_2(i,j) \leqslant n_2(j,i)$. This yields that $n_2(j,i) \leqslant n_2(i,j)$. Thus $n_2(i,j) = n_2(j,i)$. Noticing that for $k = 4$ types of any A_{ija} are quad, trio, double twins, and twins, from results proven above, we conclude that $T_1(i,j) = T_1(j,i)$.

We prove $T_2(i,j) = T_2(j,i)$. From the definition, if $\pi(i,j)$ exists, then the type of A_{ija} is not double twins, $a = 1, \ldots, 4$. For any i, j, $1 \leqslant i, j \leqslant 4$, $i \neq j$, suppose that $\pi(i,j)$ exists. We prove that $\pi(j,i)$ is the inverse permutation of $\pi(i,j)$; since types of a permutation and its inverse permutation are the same, from the definition, this yields $T_2(i,j) = T_2(j,i)$. Let r be the number of a such that A_{ija} has type quad or type trio, $a = 1, \ldots, 4$. In the case of $r = 4$, it is evident that $\pi(j,i)$ is the inverse permutation of $\pi(i,j)$. In the case of $r = 3$, suppose that A_{ija}, A_{ijb} and A_{ijc} have type quad or type trio, where $\{a, b, c, d\} = \{1, 2, 3, 4\}$. Then A_{ijd} has type twins. Without loss of generality, A_{ijd} consists of a, b, c, c. It follows that A_{ija} and A_{ijb} contain one c. It reduces to two cases: (a) A_{ijc} consists of d, d, d, d, A_{ija} consists of

b, b, b, c, and A_{ijb} consists of a, a, a, c; or (b) A_{ija} consists of d, d, d, c, A_{ijb} consists of a, a, a, c, and A_{ijc} consists of b, b, b, d. It is easy to verify that $\pi(i, j) = \pi(j, i) = (ab)(cd)$ in case (a) and that $\pi(i, j) = (dcba)$ and $\pi(j, i) = (abcd)$ in case (b). Thus $\pi(j, i)$ is the inverse permutation of $\pi(i, j)$. In the case of $r = 2$, suppose that A_{ija} and A_{ijb} have type quad or type trio, where $\{a, b, c, d\} = \{1, 2, 3, 4\}$. Then A_{ijc} and A_{ijd} have type twins. Thus A_{ijc} contains a, b, d and A_{ijd} contains a, b, c. This yields that A_{ija} and A_{ijb} have type trio; therefore, without loss of generality, A_{ija} consists of c, c, c, b and A_{ijb} consists of d, d, d, a. Then two cases are possible: (a) A_{ijc} consists of a, a, b, d, and A_{ijd} consists of a, b, b, c; or (b) A_{ijc} consists of a, b, b, d, and A_{ijd} consists of a, a, b, c. It is easy to verify that $\pi(i, j) = \pi(j, i) = (ac)(bd)$ in case (a) and that $\pi(i, j) = (acbd)$ and $\pi(j, i) = (dbca)$ in case (b). Thus $\pi(j, i)$ is the inverse permutation of $\pi(i, j)$. In the case of $r = 1$, suppose that A_{ija} has type quad or type trio, where $\{a, b, c, d\} = \{1, 2, 3, 4\}$. Then A_{ijb}, A_{ijc} and A_{ijd} have type twins. Thus A_{ijb} contains a, c, d, A_{ijc} contains a, b, d, and A_{ijd} contains a, b, c. Since the number of occurrences of each element in a row of A is 4, the number of occurrences of any element in A_{ija} is at most 2; this contradicts that A_{ija} has type quad or type trio. Thus, the case $r = 1$ should not happen. In the case of $r = 0$, A_{ije}, $e = 1, 2, 3, 4$ have type twins. Let $\pi(e)$ be the element which occurs two times in A_{ije}, $e = 1, \ldots, 4$. Then A_{ije} consists of $\pi(e)$ and elements in $\{1, 2, 3, 4\} \setminus \{e\}$, $e = 1, \ldots, 4$. Since the number of occurrences of each element in a row of A is 4, we have $\{\pi(e) \mid e = 1, 2, 3, 4\} = \{1, 2, 3, 4\}$. Clearly, $\pi(i, j) = \pi$. It is easy to verify that $\pi(j, i) = \pi^{-1}$. Thus $\pi(j, i)$ is the inverse permutation of $\pi(i, j)$. To sum up, we conclude $T_2(i, j) = T_2(j, i)$.

In final, from the definition of $T_3(i, j)$, it is easy to see that for any i, j, $1 \leqslant i, j \leqslant 4$, $i \neq j$, $T_3(i, j) = T_3(j, i)$ holds. Since for any i, j, $1 \leqslant i, j \leqslant 4$, $i \neq j$, $T_a(i, j) = T_a(j, i)$ holds for $a = 1, 2, 3$, we obtain $T(i, j) = T(j, i)$ for any i, j, $1 \leqslant i, j \leqslant 4$, $i \neq j$. □

8.2.4 Autotopism Group

For any n and any k, the group consisting of all isotopisms of (n, k)-Latin arrays equipped with the composition operation is called the *isotopism group* of (n, k)-Latin arrays, denoted by G. Clearly, G can be represented as $S_n \times S_n \times S_{nk}$, where S_r stands for the symmetric group of degree r for any positive integer r. We denote an element of G by $\langle \alpha, \beta, \gamma \rangle$, where α (the row arranging), β (the renaming) and γ (the column arranging) are permutations of the first n, n and nk positive integers, respectively.

For any (n, k)-Latin array A, an isotopism from A to itself is called an *autotopism* on A. All autotopisms on A constitute a subgroup of G, denoted by G_A, which is referred to as the *autotopism group* of A.

Using a result in group theory, see Theorem 3.2 in [142] for example, we have $|G_A||A^G| = |G|$, where A^G stands for the isotopy class containing A. Since the order of the isotopism group G of (n,k)-Latin arrays is $(n!)^2(nk)!$, the cardinal number of the isotopy class containing A is $|A^G| = (n!)^2(nk)!/|G_A|$. Therefore, for enumeration of (n,k)-Latin arrays, we may first find out all isotopy classes of (n,k)-Latin arrays, then evaluate the cardinal number of each isotopy class by means of computing the autotopism group of any Latin array in the isotopy class. The following results are useful in computing autotopism groups.

We use e to denote the identity element of G, and $(i_1 i_2 \dots i_k)$ to denote the cyclic permutation which carries i_j into i_{j+1} for $j < k$ and i_k into i_1. Therefore, (1) denotes the identity permutation.

Given an arbitrary (n,k)-Latin array A, let

$$\begin{aligned} G'''_A &= \{\langle \alpha, \beta, \gamma\rangle \in G_A \mid \alpha = (1), \beta = (1)\},\\ G''_A &= \{\langle \alpha, \beta, \gamma\rangle \in G_A \mid \alpha = (1)\},\\ G'_A &= \{\langle \alpha, \beta, \gamma\rangle \in G_A \mid \alpha(1) = 1\}. \end{aligned}$$

It is easy to see that $G'''_A \leqslant G''_A \leqslant G'_A \leqslant G_A$, where the symbol $\leqslant$ in group theory stands for "is a subgroup of". Similarly, let

$$\begin{aligned} G''' &= \{\langle \alpha, \beta, \gamma\rangle \in G \mid \alpha = (1), \beta = (1)\},\\ G'' &= \{\langle \alpha, \beta, \gamma\rangle \in G \mid \alpha = (1)\},\\ G' &= \{\langle \alpha, \beta, \gamma\rangle \in G \mid \alpha(1) = 1\}. \end{aligned}$$

Clearly, we have $G''' \leqslant G'' \leqslant G' \leqslant G$, and $G'''_A \leqslant G'''$, $G''_A \leqslant G''$, $G'_A \leqslant G'$.

G_A can be obtained by computing G'''_A, G''_A, G'_A and G_A in turn. The subgroup G'''_A can be determined as follows. Partition the column labels of A into equivalence classes according to the identity relation of columns, i.e., i and j belong to the same equivalence class if and only if the i-th column and the j-th column of A are the same. Denote the equivalence classes with cardinal number > 1 by $I_1, \dots, I_r$. We use S^{I_j} to denote the symmetric group on I_j. It is easy to show that G'''_A is isomorphic to $S^{I_1} \times S^{I_2} \times \dots \times S^{I_r}$. Let $c_k \dots c_2$ be the column characteristic value of A. Then $c_i = |\{j \mid 1 \leqslant j \leqslant r, |I_j| = i\}|$. It follows that the order of G'''_A is $\prod_{i=2}^{k}(i!)^{c_i}$.

For any subgroup H of a group H', and any elements h_1 and h_2 in H', we use $h_1 \equiv h_2 (\mathrm{mod}\ H)$ to denote the condition $h_1 h_2^{-1} \in H$.

Similar to the case of Latin square, we can prove the following.

Theorem 8.2.5. *Let* $\langle \alpha_i, \beta_i, \gamma_i\rangle \in G_A$, $i = 1, 2$.

(a) *If* $\alpha_1 = \alpha_2$ *and* $\beta_1 = \beta_2$, *then* $\gamma_1 \equiv \gamma_2 \ (\mathrm{mod}\ S^{I_1} \times S^{I_2} \times \dots \times S^{I_r})$.

(b) *If* $\alpha_1 = \alpha_2$ *and* $\gamma_1 \equiv \gamma_2 \ (\mathrm{mod}\ S^{I_1} \times S^{I_2} \times \dots \times S^{I_r})$, *then* $\beta_1 = \beta_2$.

(c) *If rows of A are distinct from each other, $\beta_1 = \beta_2$ and $\gamma_1 \equiv \gamma_2 \pmod{S^{I_1} \times S^{I_2} \times \cdots \times S^{I_r}}$, then $\alpha_1 = \alpha_2$.*

Proof. Let $\langle \alpha, \beta, \gamma \rangle = \langle \alpha_1 \alpha_2^{-1}, \beta_1 \beta_2^{-1}, \gamma_1 \gamma_2^{-1} \rangle$. Then $\langle \alpha, \beta, \gamma \rangle \in G_A$.

(a) Since $\alpha_1 = \alpha_2$ and $\beta_1 = \beta_2$, we have $\alpha = (1)$ and $\beta = (1)$. It follows that $\langle \alpha, \beta, \gamma \rangle \in G_A'''$. Therefore, we have $\gamma \in S^{I_1} \times S^{I_2} \times \cdots \times S^{I_r}$. That is, $\gamma_1 \equiv \gamma_2 \pmod{S^{I_1} \times S^{I_2} \times \cdots \times S^{I_r}}$.

(b) Since $\alpha_1 = \alpha_2$ and $\gamma_1 \equiv \gamma_2 \pmod{S^{I_1} \times S^{I_2} \times \cdots \times S^{I_r}}$, we have $\alpha = (1)$ and $\gamma \in S^{I_1} \times S^{I_2} \times \cdots \times S^{I_r}$. Thus A keeps unchanged under row arranging α and column arranging γ. From $\langle \alpha, \beta, \gamma \rangle \in G_A$, this yields that A keeps unchanged under renaming β. Since each column of A is a permutation of elements in N, we have $\beta = (1)$. This yields $\beta_1 = \beta_2$.

(c) Since $\beta_1 = \beta_2$ and $\gamma_1 \equiv \gamma_2 \pmod{S^{I_1} \times S^{I_2} \times \cdots \times S^{I_r}}$, we have $\beta = (1)$ and $\gamma \in S^{I_1} \times S^{I_2} \times \cdots \times S^{I_r}$. Thus A keeps unchanged under renaming β and column arranging γ. From $\langle \alpha, \beta, \gamma \rangle \in G_A$, this yields that A keeps unchanged under row arranging α. Since rows of A are distinct from each other, we have $\alpha = (1)$. Thus $\alpha_1 = \alpha_2$. □

Theorem 8.2.6. (a) *Let G_1 be a subgroup of G_2 and of G_A. For any g in G_2, the (right) coset $gG_1 \subseteq G_A$ if and only if $g \in G_A$.*

(b) *Let $g_i = \langle \alpha_i, \beta_i, \gamma_i \rangle, i = 1, 2$. If $g_1, g_2 \in G_A''$, then $g_1 G_A''' = g_2 G_A'''$ if and only if $\beta_1 = \beta_2$.*

(c) *If $g_1, g_2 \in G_A'$, then $g_1 G_A'' = g_2 G_A''$ if and only if $\alpha_1 = \alpha_2$.*

(d) *If $g_1, g_2 \in G_A$, then $g_1 G_A' = g_2 G_A'$ if and only if $\alpha_1(1) = \alpha_2(1)$.*

Proof. (a) Evident from the definition.

(b) Suppose $g_1, g_2 \in G_A''$. Then $\alpha_1 = \alpha_2 = (1)$. Thus $g_2^{-1} g_1 = \langle (1), \beta_2^{-1}\beta_1, \gamma_2^{-1}\gamma_1 \rangle$. Therefore, $\beta_1 = \beta_2$ if and only if $g_2^{-1} g_1 \in G_A'''$, if and only if $g_1 G_A''' = g_2 G_A'''$.

(c) Suppose $g_1, g_2 \in G_A'$. Since $g_2^{-1} g_1 = \langle \alpha_2^{-1}\alpha_1, \beta_2^{-1}\beta_1, \gamma_2^{-1}\gamma_1 \rangle$, $\alpha_1 = \alpha_2$ if and only if $g_2^{-1} g_1 \in G_A''$, if and only if $g_1 G_A'' = g_2 G_A''$.

(d) Suppose $g_1, g_2 \in G_A$. Clearly, $\alpha_1(1) = \alpha_2(1)$ if and only if $\alpha_2^{-1}\alpha_1(1) = 1$ (i.e., $\alpha_2^{-1}(\alpha_1(1)) = 1$), if and only if $g_2^{-1} g_1 \in G_A'$, if and only if $g_1 G_A' = g_2 G_A'$. □

Although parallel results for left coset hold, in this section, it is enough to use right coset; hereafter, "coset" means "right coset".

An (n, k)-Latin array A is said to be *canonical*, if the first row of A is $1 \ldots 1\ 2 \ldots 2 \ldots n \ldots n$. Partition a canonical (n, k)-Latin array A into n blocks such that all the elements in the first row of the h-th block of A, denoted by A_h, are h, $h = 1, \ldots, n$. For any block A_h, let c_i be the number of distinct columns of A_h which occur exactly i times in A_h; $c_k c_{k-1} \ldots c_2$ is called the *column type* of A_h. Let p_i be the type of the i-th row of A_h; the sequence $(p_2, p_3, \ldots, p_n)$ is called the *row type* of A_h.

Theorem 8.2.7. *Assume that (n,k)-Latin arrays A and B are canonical and that A can be transformed into B by an isotopism $\langle(1),\beta,\gamma\rangle$. Then for any $i, 1 \leqslant i \leqslant n$, column types (row types) of blocks A_i and $B_{\beta(i)}$ are the same. Moreover, if there is a block of A, say A_h, such that the pair of the row type and the column type of A_h is distinct from the ones of other blocks and that for some positive integer r there is only one column of A_h with multiplicity r, then β can be determined by such a column.*

Proof. Clearly, for any $i, 1 \leqslant i \leqslant n$, the block A_i can be transformed into the block $B_{\beta(i)}$ by some renaming and some column arranging within block. Since the column type and the row type of a block keep unchanged under renamings and column arrangings within block, the column type and the row type of A_i are the same with ones of $B_{\beta(i)}$, respectively.

Suppose that A_h satisfies the conditions mentioned in the theorem. For any isotopism $\langle(1),\beta,\gamma\rangle$ which transforms A into B, using the first part of the theorem, there is only one block of B of which the column type and the row type are the same with ones of A_h, respectively, and such a block is the block $B_{\beta(h)}$. Clearly, there is only one column in the block $B_{\beta(h)}$ with column multiplicity r, and such a column can be obtained from transforming the column of A_h with multiplicity r by renaming β. Since each column of A is a permutation, such two corresponding columns determine β. □

8.2.5 The Case $n = 2, 3$

From the definitions, it is easy to see that any $(2,k)$-Latin array is a $(2,k,k)$-Latin array. Therefore, if A is a $(2,k,r)$-Latin array, then $r = k$ holds.

Theorem 8.2.8. *For any positive integer k, we have $I(2,k) = 1$, $U(2,k) = \binom{2k}{k}$.*

Proof. Let A be a canonical $(2,k)$-Latin array. Since each column of A is a permutation of 1 and 2, elements in the second row of A are 2 at the first k columns and 1 at the last k columns. Such a Latin array is denoted by $A12k$. Clearly, any $(2,k)$-Latin array can be transformed into $A12k$ by some column arranging. Consequently, $(2,k)$-Latin arrays have a unique isotopy class.

Since a $(2,k)$-Latin array can be uniquely determined by any row and the multiplicity of 1 and of 2 in any row are k, the number of distinct $(2,k)$-Latin arrays is $\binom{2k}{k}$. □

Theorem 8.2.9. *For any positive integer k, we have*

$$I(3,k) = \begin{cases} (k+1)/2, & \text{if } k \text{ is odd,} \\ k/2+1, & \text{otherwise,} \end{cases}$$

$$U(3,k) = \begin{cases} \sum_{h=(k+1)/2}^{k} 2(3k)!/(h!(k-h)!)^3, & \text{if } k \text{ is odd,} \\ \sum_{h=k/2+1}^{k} 2(3k)!/(h!(k-h)!)^3 + (3k)!/((k/2)!)^6, & \text{otherwise,} \end{cases}$$

and $I(3,k,1) = U(3,k,1) = 0$ *if* $k > 2$, $I(3,2,1) = 1$, $U(3,2,1) = 6!$, $I(3,1,1) = 1$, $U(3,1,1) = 3!2!$.

Proof. For any $h, 1 \leqslant h \leqslant k$, let $A(k-h+1)3k$ be a canonical $(3,k)$-Latin array of the form

$$\underbrace{\begin{bmatrix} 1\ldots1 & 1\ldots1 & 2\ldots2 & 2\ldots2 & 3\ldots3 & 3\ldots3 \\ 2\ldots2 & 3\ldots3 & 3\ldots3 & 1\ldots1 & 1\ldots1 & 2\ldots2 \\ 3\ldots3 & 2\ldots2 & 1\ldots1 & 3\ldots3 & 2\ldots2 & 1\ldots1 \end{bmatrix}}_{h \quad k-h \quad h \quad k-h \quad h \quad k-h}.$$

We first prove that for any $(3,k)$-Latin array A there exists $h, 1 \leqslant h \leqslant k$, such that A and $A(k-h+1)3k$ are isotopic. We transform A into its canonical form, say A', by some column arranging. Then using column arranging within block we transform the second row of A' in the form $2\ldots2\,3\ldots3\,3\ldots3\,1\ldots1$ $1\ldots1\,2\ldots2$ and denote the result by A''. Let h be the number of 2 in block 1 row 2 of A''. Then the number of 3 in block 1 row 2 of A'' is $k-h$. Since each row of a $(3,k)$-Latin array contains exactly k elements i, $1 \leqslant i \leqslant 3$, the number of 3 in block 2 row 2 of A'' is h and the number of 2 in block 3 row 2 of A'' is $k-h$. Consequently, the number of 1 in block 2 row 2 of A'' is $k-h$, and the number of 1 in block 3 row 2 of A'' is h. Therefore, the first two rows of A'' and of $A(k-h+1)3k$ are the same. Since each column of a $(3,k)$-Latin array is a permutation of 1, 2 and 3, row 3 is uniquely determined by rows 1 and 2. Thus A'' is equal to $A(k-h+1)3k$. We conclude that A and $A(k-h+1)3k$ are isotopic. Notice that $A(h+1)3k$ and $A(k-h+1)3k$ can be mutually obtained by applying renaming (23) and some column arranging. From the above results, any $(3,k)$-Latin array A is isotopic to $A(k-h+1)3k$, for some h, $k \geqslant h \geqslant \lceil k/2 \rceil$. We next prove that $A(k-h+1)3k$, $h = k, k-1, \ldots, \lceil k/2 \rceil$ are not isotopic to each other. Whenever $h = k/2$ and k is even, in the column characteristic value of $A(k-h+1)3k$, namely, $c_k \ldots c_2$, we have $c_h = 6$, and $c_i = 0$ for $i \neq h$. Whenever $h = k$, in the column characteristic value $c_k \ldots c_2$ of $A(k-h+1)3k$, we have $c_h = 3$, and $c_i = 0$ for $i \neq h$. Whenever $h = k-1, \ldots, \lceil k/2 \rceil$ and $h > k/2$, in the column characteristic value $c_k \ldots c_2$ of $A(k-h+1)3k$, we have $c_h = c_{k-h} = 3$, and $c_i = 0$ for $i \neq h, k-h$. Therefore, column characteristic values of $A(k-h+1)3k$, $h = k, k-1, \ldots, \lceil k/2 \rceil$ are different from each other. From Corollary 8.2.1, they are not isotopic to each other. To sum up, $I(3,k) = (k+1)/2$ if k is odd, and $k/2+1$ otherwise.

For $U(3,k)$, we compute the order of the autotopism group of $A(k-h+1)3k$. Clearly, the order of $G'''_{A(k-h+1)3k}$ is $(h!(k-h)!)^3$. We now compute coset representatives of $G'''_{A(k-h+1)3k}$ in $G''_{A(k-h+1)3k}$. In the case of $h > k-h$,

it is easy to see that there is a column arranging $\gamma = (3k, 3k-1, \ldots, 1)^k$ such that the isotopism $g = \langle(1), (321), \gamma\rangle$ keeps $A(k-h+1)3k$ unchanged. Consequently, g^2 and $g^3 = e$ also keep $A(k-h+1)3k$ unchanged, where e stands for the identity isotopism. Since $h \neq k-h$, for any transposition β and any column arranging γ the isotopism $\langle(1), \beta, \gamma\rangle$ is not an autotopism of $A(k-h+1)3k$. From Theorem 8.2.6 (b), the coset representatives are g, g^2 and e. It follows that the order of $G''_{A(k-h+1)3k}$ is equal to $3|G'''_{A(k-h+1)3k}| = 3(h!(k-h)!)^3$. In the case of $h = k-h$, we have $h = k/2$. It is easy to prove that there exist column arrangings γ_1, γ_2 such that $\langle(1), (23), \gamma_1\rangle$ and $\langle(1), (12), \gamma_2\rangle$ are autotopisms of $A(k-h+1)3k$. Since transpositions (23) and (12) can generate all permutations of 1, 2 and 3, for any renaming β there exists a column arranging γ such that $\langle(1), \beta, \gamma\rangle$ is an autotopism of $A(k-h+1)3k$. Using Theorem 8.2.6 (b), we have $|G''_{A(k-h+1)3k}| = 3!|G'''_{A(k-h+1)3k}| = 6(h!)^6$. We turn to computing coset representatives of $G''_{A(k-h+1)3k}$ in $G'_{A(k-h+1)3k}$. It is easy to see that there exists a column arranging γ such that $\langle(23), (23), \gamma\rangle$ keeps $A(k-h+1)3k$ unchanged. From Theorem 8.2.6 (c), it is easy to prove that $|G'_{A(k-h+1)3k}| = 2|G''_{A(k-h+1)3k}|$. We finally compute coset representatives of $G'_{A(k-h+1)3k}$ in $G_{A(k-h+1)3k}$. It is easy to show that there exists a column arranging γ such that $g = \langle(321), (1), \gamma\rangle$ keeps $A(k-h+1)3k$ unchanged. From Theorem 8.2.6 (d), we have that g, g^2 and e are the all coset representatives. It immediately follows that $|G_{A(k-h+1)3k}| = 3|G'_{A(k-h+1)3k}|$. To sum up, the order of $G_{A(k-h+1)3k}$ is equal to $3 \cdot 2 \cdot 3(h!(k-h)!)^3$ in the case of $h > k-h$, or $3 \cdot 2 \cdot 6(h!)^6$ in the case of $h = k-h$. Therefore, the cardinal number of the isotopy class containing $A_{(k-h+1)3k}$ is equal to $(3!)^2(3k)!/(3 \cdot 2 \cdot 3(h!(k-h)!)^3)$ $= 2(3k)!/(h!(k-h)!)^3$ in the case of $h > k-h$, or $(3!)^2(3k)!/(3 \cdot 2 \cdot 6(h!)^6)$ $= (3k)!/((k/2)!)^6$ in the case of $h = k-h$. In the preceding paragraph we have proven that the isotopy classes containing $A_{(k-h+1)3k}$, $h = k, k-1, \ldots, \lceil k/2 \rceil$ are all isotopy classes of $(3, k)$-Latin array. Thus the formula of $U(3, k)$ in the theorem holds.

Since the number of columns of a $(3, k)$-Latin array is $3k$ and the number of permutations on $\{1, 2, 3\}$ is 3!=6, for any $(3, k, 1)$-Latin array we have $3k \leqslant 6$, that is, $k \leqslant 2$. Thus $I(3, k, 1) = U(3, k, 1) = 0$ if $k > 2$.

It is easy to see that the columns of any $(3, 2, 1)$-Latin array consists of all permutations on $\{1, 2, 3\}$. Thus any two $(3, 2, 1)$-Latin arrays are column-equivalent. Therefore, $I(3, 2, 1) = 1$ and $U(3, 2, 1) = 6!$.

Consider a $(3, 1, 1)$-Latin array of which the first row and the first column are x_1, x_2, x_3. Then the Latin array is uniquely determined by these elements. In fact, since any row and any column are some permutations of x_1, x_2 and x_3, its elements at the positions (2,3) and (3,2) take x_1. It follows that its elements at the positions (2,2) and (3,3) are x_3 and x_2, respectively.

Since any $(3,1,1)$-Latin array can be transformed by rearranging rows and columns into a $(3,1,1)$-Latin array of which the first row and the first column are 1, 2, 3, from the result in the preceding paragraph, we have $I(3,1,1) = 1$. Since the number of permutations on $\{1,2,3\}$ is 3!, we have $U(3,1,1) = 3!2!$. □

8.2.6 The Case $n = 4$, $k \leqslant 4$

Enumeration of $(4,2)$-Latin Arrays

It is known that the number of isotopy classes of Latin squares of order 4 is 2 and the number of Latin squares of order 4 is $(4!)^2$, see [31] for example. Since both (4,1)-Latin arrays and (4,1,1)-Latin arrays coincide with Latin squares of order 4, we have $I(4,1) = I(4,1,1) = 2$ and $U(4,1) = U(4,1,1) = (4!)^2$. Another proof of the results using Theorem 8.2.1 will be given later.

Let $A142, \ldots, A1142$ be (4,2)-Latin arrays as follows:

$$\begin{bmatrix} 11223344 \\ 22114433 \\ 34341212 \\ 43432121 \end{bmatrix} \quad \begin{bmatrix} 11223344 \\ 22334411 \\ 34411223 \\ 43142132 \end{bmatrix} \quad \begin{bmatrix} 11223344 \\ 22134413 \\ 34341221 \\ 43412132 \end{bmatrix} \quad \begin{bmatrix} 11223344 \\ 22341413 \\ 34432121 \\ 43114232 \end{bmatrix}$$
$$A142 \qquad A242 \qquad A342 \qquad A442$$

$$\begin{bmatrix} 11223344 \\ 22341413 \\ 34412132 \\ 43134221 \end{bmatrix} \quad \begin{bmatrix} 11223344 \\ 23144123 \\ 34431212 \\ 42312431 \end{bmatrix} \quad \begin{bmatrix} 11223344 \\ 22114433 \\ 33441122 \\ 44332211 \end{bmatrix} \quad \begin{bmatrix} 11223344 \\ 22114433 \\ 33441221 \\ 44332112 \end{bmatrix}$$
$$A542 \qquad A642 \qquad A742 \qquad A842$$

$$\begin{bmatrix} 11223344 \\ 22114433 \\ 33442211 \\ 44331122 \end{bmatrix} \quad \begin{bmatrix} 11223344 \\ 22134413 \\ 33441221 \\ 44312132 \end{bmatrix} \quad \begin{bmatrix} 11223344 \\ 22344113 \\ 33412421 \\ 44131232 \end{bmatrix}.$$
$$A942 \qquad A1042 \qquad A1142$$

Lemma 8.2.4. *$A142, \ldots, A1142$ are not isotopic to each other.*

Proof. We compute the column characteristic value and the row characteristic set of $Ax42$ and represent them in the format:"x: the column characteristic value of $Ax42$; the row characteristic sets of $Ax42$", where the column characteristic value is in the form c_2, the row characteristic set is in the form $T_1(i,j)\ T_2(i,j)$, in order of $ij = 12, 13, 14, 23, 24, 34$, $T_3(i,j)$ is not listed. We list the results of column characteristic values and row characteristic sets of $A142$, ..., $A1142$ as follows:

$1:0;42,00,00,00,00,42$ $2:0;44,00,00,00,00,00$
$3:0;20,00,00,00,00,20$ $4:0;10,00,10,10,00,10$
$5:0;10,00,00,10,10,00$ $6:0;00,00,00,00,00,00$
$7:4;42,42,42,42,42,42$ $8:2;42,20,20,20,20,42$
$9:4;42,44,44,44,44,42$ $10:1;20,20,10,10,20,20$
$11:1;10,10,10,10,10,10$

where $T_2(i,j) = 4,2,0$ mean "a cycle of length 4", "two transpositions", "no derived permutation", respectively. It is easy to verify that for any two distinct $Ai42$'s either their column characteristic values are different, or their row characteristic sets are different. From Corollary 8.2.1, it immediately follows that $A142,\ldots,A1142$ are not isotopic to each other. □

Lemma 8.2.5. *Any* $(4,2)$*-Latin array is isotopic to one of* $A142,\ldots,A1142$*; any* $(4,2,1)$*-Latin array is isotopic to one of* $A142,\ldots,A642$ *and any* $(4,2,2)$*-Latin array is isotopic to* $A742$ *or* $A942$*.*

Proof. Let A be a (4,2)-Latin array. In the proof, by $A(i,j)$ denote the element of A at row i column j; by $A(i,j-h)$ denote elements of A at row i columns j to h. Since Latin arrays can be reduced to canonical ones by rearranging columns, without loss of generality, we suppose that A is canonical. From Theorem 8.2.4, instead of "from row i to row j" we can say "between rows i and j", for example, the intersection number between rows i and j; and from $T_1(i,j) = T_1(j,i) = 1\cdot c_1 + 0\cdot c_0$, we can say "the number of twins between rows i and j is c_1", and so on.

We prove by exhaustion that A is isotopic to $Ai42$ for some i, $1 \leqslant i \leqslant 11$. There are two cases to consider. Case 1: columns of A are different. Case 2: otherwise.

Case 1: no repeated columns in A. There are five alternatives according to the numbers of twins between rows. Case 11: there are two rows of A between which the number of twins is 4. Case 12: not the case 11 and there are two rows of A between which the number of twins is 3. Case 13: not the cases 11 and 12 and there are two rows of A between which the number of twins is 2. Case 14: not the cases 11 to 13 and there are two rows of A between which the number of twins is 1. Case 15: there is no twins between any two rows of A.

In case 11, in the sense of row arranging and column arranging we assume that the number of twins between rows 1 and 2 of A is 4. We subdivide this case into two subcases according to the derived permutation from row 1 to row 2 of A. Case 111: the derived permutation can be decomposed into a product of two disjoint transpositions. Case 112: the derived permutation is a cycle of length 4.

In case 111, in the sense of isotopy we assume that $A(2, 1-8) = 22114433$. Note that for the last two rows of any block of A, whenever one has type twins, so has the other. This yields that A has repeated columns. But this is impossible in case 1. Therefore, in the sense of column transposition within block we have $A(3, 1-8) = 34341212$. Since each column of A is a permutation, for any column of A the element at any row can be uniquely determined by others in that column. It follows that $A(4, 1-8) = 43432121$. [Hereafter, this deduction and the like are abbreviated to the form: "since last element(s), ...".] Therefore, A is isotopic to $A142$.

In case 112, in the sense of isotopy we assume that $A(2, 1-8) = 22334411$. Since A has no repeated columns, in the sense of column transposition within block we have $A(3, 1-8) = 34411223$. Since last elements, we obtain $A(4, 1-8) = 43142132$. Therefore, A is isotopic to $A242$.

In case 12, since each element occurs exactly two times in any row of A, between any two rows of A if the number of twins is at least 3 then it is equal to 4. This contradicts not the case 11. Therefore, this case can not occur.

In case 13, in the sense of isotopy we suppose that the number of twins between rows 1 and 2 of A is 2. We subdivide this case into three subcases according to the intersection number between rows 1 and 2 of A. Case 131: the intersection number is 2. Case 132: the intersection number is 1. Case 133: the intersection number is 0.

In case 131, in the sense of isotopy we assume $A(2, 1-4) = 2211$. Since each element occurs exactly once in each column and 2 times in each row of A, elements in block 3 row 2 of A are uniquely determined, namely, $A(2, 5-6) = 44$. [Hereafter, this deduction and the like are abbreviated to the form: "since unique value(s), ...".] This contradicts not the cases 11 and 12. Therefore, this case can not occur.

In case 132, in the sense of isotopy we assume $A(2, 1-4) = 2233$. Since unique values, we have $A(2, 7-8) = 11$. This contradicts not the cases 11 and 12. Therefore, this case can not occur.

In case 133, in the sense of isotopy we assume $A(2, 1-2) = 22$, $A(2, 5-6) = 44$. Since row 2 has already two occurrences of 2 and of 4, elements in blocks 2 and 4 row 2 are 1 or 3. Since block 2 row 2 and block 4 row 2 are not twins, in the sense of column transposition within block we have $A(2, 3-4) = A(2, 7-8) = 13$. [Hereafter, this deduction and the like are abbreviated to the form: "since no twins, ...".] Since each element occurs once in each column, in the sense of row transposition we have $A(4, 3) = 4$. Since last element, we have $A(3, 3) = 3$. Since A has no repeated columns, in the sense of column transposition within block we have $A(3, 1-2) = 34, A(3, 5-6) = 12$. Since last elements, we obtain $A(4, 1-2) = 43, A(4, 5-6) = 21$. Consider the columns in which the elements at row 3 are not determined yet and the

elements determined so far do not contain 4. Since such a column is unique, the place in row 3 which can take value 4 is unique. It immediately follows that $A(3,4) = 4$. [Hereafter, this deduction and the like are abbreviated to the form: "since unique place(s), ..."] Since unique value, we have $A(3,7) = 2$. Since unique value, we have $A(3,8) = 1$. Since last elements, we obtain $A(4,4) = 1, A(4,7-8) = 32$. Therefore, A is isotopic to $A342$.

In case 14, in the sense of isotopy we assume that there is one twins between rows 1 and 2, say $A(2,1-2) = 22$. Since no twins, we have $A(2,5-6) = 14$ and $A(2,7-8) = 13$. Since each element occurs exactly two times in each row of A, elements 3 and 4 in row 2 which are not determined yet so far can only occur in block 2 row 2. Consequently, in the sense of column transposition we have $A(2,3-4) = 34$. [Hereafter, this deduction and the like are abbreviated to the form: "since row sum, ..."] Since each element occurs once in each column, in the sense of row transposition we have $A(4,3) = 1$. Since last element, we obtain $A(3,3) = 4$. Since A has no repeated columns, in the sense of column transposition within block we have $A(3,1-2) = 34$. Since last elements, we have $A(4,1-2) = 43$. Since unique place, we have $A(4,5) = 4$. Since last element, we have $A(3,5) = 2$. At this point, there are two alternatives according to the value of $A(3,4)$, 3 for the case 141, and 1 for the case 142.

In case 141, since unique places, we have $A(3,6) = A(3,8) = 1, A(4,7) = 3$. Since last elements, we have $A(4,4) = 1, A(4,6) = A(4,8) = A(3,7) = 2$. Therefore, A is isotopic to $A442$.

In case 142, since unique place, we have $A(3,7) = 3$. Since last elements, we have $A(4,4) = 3, A(4,7) = 2$. Denoting $A(4,8) = a$, it is easy to see that a takes values 1 or 2. Since row sum, we have $A(4,6) = a'$, where $1' = 2, 2' = 1$. Since last elements, we have $A(3,8) = a', A(3,6) = a$. Whenever $a = 1$, A is isotopic to $A542$. Whenever $a = 2$, A can be transformed into $A542$ by row transposition (12), renaming (12)(34) and some column arranging.

In case 15, in the sense of isotopy we assume $A(2,1-2) = 23$. We have $A(2,8) = 3$ in the sense of isotopy. (In fact, when 3 does not occur in block 4 row 2, 2 occurs in it since no twins. Thus we can transform A in advance by renaming (23) and some column arranging.) Since no twins, we have $A(2,3-4) = 14$. Since unique places, in the sense of column transposition we have $A(2,5) = 4$. Denoting $A(2,7) = b$, it is easy to see that b takes values 1 or 2. Since row sum, we have $A(2,6) = b'$. In the sense of row transposition we assume $A(3,1) = 3, A(4,1) = 4$. Since no twins, we have $A(4,2) = 2$. Since last element, we have $A(3,2) = 4$. We prove $A(3,3) \neq 3$ by reduction to absurdity. Suppose to the contrary that $A(3,3) = 3$. Since last element, we have $A(4,3) = 4$. It follows that there is a twins between rows 3 and 4. This contradicts the case 15. Therefore, we obtain $A(3,3) = 4$. Since

last element, we have $A(4,3) = 3$. Since no twins, we have $A(4,4) = 1$. Since no twins between rows 2 and 4 exists, we have $A(4,8) = 1$. Since last elements, we have $A(3,4) = 3, A(3,8) = 2$. Since unique places, we have $A(4,6) = 4, A(4,7) = 3$. Since row sum, we have $A(4,5) = 2$. Since last elements, we have $A(3,5-7) = 1bb'$. Since no twins between rows 2 and 4 exists, we have $b \neq 1$. It immediately follows that $b = 2$. Therefore, A is isotopic to $A642$.

Case 2: A has repeated columns. Subdivide this case into three subcases according to the number of twins between rows. Case 21: there are two rows of A between which the number of twins is 4. Case 22: not the case 21 but there are two rows of A between which the number of twins is 2. Case 23: otherwise.

In case 21, in the sense of isotopy we assume that there are four twins between rows 1 and 2. We subdivide this case into two subcases according to the derived permutation from rows 1 to 2 of A. Case 211: the derived permutation can be decomposed into a product of two disjoint transpositions. Case 212: the derived permutation is a cycle of length 4.

In case 211, in the sense of isotopy we assume $A(2,1-8) = 22114433$, $A(3,1-2) = 33$, $A(4,1-2) = 44$. Since unique values, we have $A(3,3-4) = 44, A(4,3-4) = 33$. Denoting $A(4,7-8) = ab$, clearly, a and b take values 1 or 2. Since row sum, in the sense of column transposition we have $A(4,5-6) = a'b'$. Since last elements, we have $A(3,5-8) = aba'b'$. In the sense of column transposition ab may take three values 11,12 and 22, it follows that A is isotopic to $A742, A842$ and $A942$, respectively.

In case 212, in the sense of isotopy we assume $A(2,1-8) = 22334411$, $A(3,1-2) = 33$, $A(4,1-2) = 44$. Since unique values, we have $A(4,3-4) = 11, A(3,7-8) = 22$. Since last elements, we have $A(3,3-4) = 44, A(4,7-8) = 33$. Since row sum, we have $A(3,5-6) = 11, A(4,5-6) = 22$. It is easy to see that A can be transformed into $A942$ by row transposition (23), renaming (23) and some column arranging.

In case 22, in the sense of isotopy we assume that there are two twins between rows 1 and 2 of A. We subdivide this case into three subcases according to the intersection number between rows 1 and 2 of A. Case 221: the intersection number is 2. Case 222: the intersection number is 1. Case 223: the intersection number is 0.

In case 221, in the sense of isotopy we assume $A(2,1-4) = 2211$. Since unique places, we have $A(2,5-6) = 44, A(2,7-8) = 33$. This contradicts not the case 21. Therefore, this case can not occur.

In case 222, in the sense of isotopy we assume $A(2,1-2) = 22$, $A(2,3-4) = 33$. Since unique values, we have $A(2,7-8) = 11$. Since row sum, we have

$A(2,5-6)=44$. This contradicts not the case 21. Therefore, this case can not occur.

In case 223, in the sense of isotopy we assume $A(2,1-2)=22$, $A(2,5-6)=44$, $A(3,1-2)=33$, $A(4,1-2)=44$. Since no twins, we have $A(2,3-4)=A(2,7-8)=13$. Since unique values, we have $A(4,3-4)=31, A(3,7)=2$. Since last elements, we have $A(3,3-4)=44, A(4,7)=3$. Since no twins, we have $A(3,8)=1$. Since row sum, in the sense of column transposition we have $A(3,5-6)=12$. Since last elements, we have $A(4,5-6)=21$ and $A(4,8)=2$. Therefore, A is isotopic to $A1042$.

In case 23, it is easy to show that the number of twins between two rows of A is equal to 4 whenever it is at least 3. Thus in this case the number of twins between some two rows of A is equal to 1. In the sense of isotopy we assume that columns 1 and 2 are the same and $A(2,1-2)=22, A(3,1-2)=33, A(4,1-2)=44$. Since no twins, we have $A(2,5-6)=41, A(2,7-8)=13$. Since row sum, in the sense of column transposition we have $A(2,3-4)=34$. Since unique values, we have $A(3,4)=1, A(3,7)=A(4,6)=2$. Since last elements, we have $A(4,4)=A(4,7)=3, A(3,6)=4$. Since no twins, we have $A(3,8)=1$. Since unique places, we have $A(3,3)=4, A(3,5)=2$. Since last elements, we have $A(4,8)=2, A(4,3)=A(4,5)=1$. Therefore, A is isotopic to $A1142$.

Since the column characteristic value of any $(4,2,2)$-Latin array is 4, $A742$ and $A942$ are all distinct isotopy class representatives of $(4,2,2)$-Latin array. That is, any $(4,2,2)$-Latin array is isotopic to $A742$ or $A942$. □

Theorem 8.2.10. *$I(4,2)=11$, $I(4,2,1)=6$, $I(4,2,2)=2$.*

Proof. This is immediate from Lemmas 8.2.4 and 8.2.5. □

Theorem 8.2.11. *$U(4,2)=12640320$, $U(4,2,1)=10281600$, $U(4,2,2)=60480$.*

Proof. Denote the order of autotopism group G_{Ai42} of $Ai42$ by $n_i, i=1,\ldots,11$. Then the number of elements in the isotopy class containing $Ai42$ is $4!4!8!/n_i$. Therefore, we have

$$U(4,2)=\sum_{i=1}^{11}4!4!8!/n_i,\ U(4,2,1)=\sum_{i=1}^{6}4!4!8!/n_i,\ U(4,2,2)=\sum_{i=7,9}4!4!8!/n_i.$$

We compute G_{A142}. In GR_{A142}, labels of edges $(1,2)$ and $(3,4)$ are the same, say "red"; labels of other edges are the same and not red, say "green".

Since columns of $A142$ are different, we have $G'''_{A142}=\{e\}$, where e stands for the identity isotopism. It immediately follows that its order is 1.

To compute the coset representatives of G'''_{A142} in G''_{A142}, note that an isotopism $\langle\alpha,\beta,\gamma\rangle$ on a Latin array can be decomposed into a product of $\langle\alpha,(1),(1)\rangle$ (the row arranging α), $\langle(1),\beta,(1)\rangle$ (the renaming β), and $\langle(1),(1),\gamma\rangle$ (the column arranging γ), independent of their order. Clearly, for any isotopism in G'' its row arranging component is (1). We then consider the renaming component β so that the renaming β and some column arranging γ keep $A142$ unchanged. Since each column of $A142$ is a permutation, β can be uniquely determined by any two columns, say j and h, of $A142$ such that β transforms column h into column j. For each $h\in\{1,\ldots,8\}$, take β as the permutation which transforms column h into column 1. For example, in the case of $h=3$, the renaming β transforms the column 2134 into the column 1234; that is, $\beta=(12)$. In this way, we obtain eight candidates for β: (1), (34), (12), (12)(34), (13)(24), (1423), (1324) and (14)(23), which can be generated by (34) and (1324) for example. It is easy to verify that β transforms $A142$ into a $(4,2)$-Latin array which can be further transformed into $A142$ by some column arranging, for $\beta=(34)$ and (1324). It follows that β transforms $A142$ into a $(4,2)$-Latin array which can be further transformed into $A142$ by some column arranging, for all eight candidates. From Theorem 8.2.6 (b), autotopisms with different renamings correspond to different cosets, and autotopisms with the same renaming correspond to the same coset. Therefore, the coset representatives of all different cosets are $(\langle(1),(34),\cdot\rangle,\langle(1),(1423),\cdot\rangle)$, here and elsewhere a dot $\cdot$ in an isotopism represents some column arranging, and $(g_1,\ldots,g_r)$ stands for the set generated by $g_1,\ldots,g_r$ (the column arranging component is neglected in the case of coset representatives of G'''_A in G''_A). That is, $G''_{A142}=(\langle(1),(34),\cdot\rangle,\langle(1),(1423),\cdot\rangle)$ $G'''_{A142}=(\langle(1),(34),\cdot\rangle,\langle(1),(1423),\cdot\rangle)$. It immediately follows that the order of G''_{A142} is 8.

To compute the coset representatives of G''_{A142} in G'_{A142}, let $\langle\alpha,\beta,\gamma\rangle\in G'$, where the row arranging α is a permutation of 2,3,4. From the proof of Theorem 8.2.3 (b), if $\langle\alpha,\beta,\gamma\rangle$ is an autotopism of $A142$, then the row arranging α is an automorphism of GR_{A142}. It follows that edges $(1,2)$ and $(\alpha(1),\alpha(2))$, that is, $(1,\alpha(2))$, have the same color. Since the edge $(1,2)$ is the unique red edge with endpoint 1, we have $\alpha(2)=2$ whenever $\langle\alpha,\beta,\gamma\rangle\in G'_{A142}$. Thus candidates of α are (34) and (1) in this case. Transform rows of $A142$ by row transposition (34). Clearly, the result can be further transformed into $A142$ by some column arranging. We then obtain a coset representative $\langle(34),(1),\cdot\rangle$. Together with another coset representative e, from Theorem 8.2.6 (c), they are coset representatives of all distinct cosets; that is, $G'_{A142}=(\langle(34),(1),\cdot\rangle)\ G''_{A142}$, here and elsewhere $(g_1,\ldots,g_r)$ stands for the set generated by $g_1,\ldots,g_r$ but redundant autotopisms with identical row ar-

ranging component are omitted except one in the case of coset representatives of G''_A in G'_A. Consequently, the order of G'_{A142} is $2 \cdot 8$.

To compute the coset representatives of G'_{A142} in G_{A142}, let $\langle \alpha, \beta, \gamma \rangle \in G_{A142}$. Thus α is an automorphism of PR_{A142}. Try $\alpha(3) = 1$. Since the edge $(3, 4)$ is red, the edge $(\alpha(3), \alpha(4))$, that is, $(1, \alpha(4))$, is red. Since the edge $(1, 2)$ is the unique red edge with endpoint 1, we have $\alpha(4) = 2$. It follows that $\alpha = (1423)$ or $(13)(24)$. Try $\alpha = (1423)$. $A142$ can be transformed into

$$A' = \begin{bmatrix} 33 & 44 & 11 & 22 \\ 44 & 33 & 22 & 11 \\ 12 & 12 & 34 & 34 \\ 21 & 21 & 43 & 43 \end{bmatrix}$$

by row arranging (1423) and some column arranging. It is evident that A' can be transformed into $A142$ by some column arranging. Therefore, $g = \langle (1423), (1), \cdot \rangle$ is an autotopism of $A142$. It follows from Theorem 8.2.6 (d) that g, g^2, g^3 and $g^4 = e$ are coset representatives of all distinct cosets of G'_{A142} in G_{A142}. That is, $G_{A142} = (\langle (1423), (1), \cdot \rangle)\, G'_{A142}$, here and elsewhere $(g_1, \ldots, g_r)$ stands for the set generated by $g_1, \ldots, g_r$ but redundant autotopisms with identical value of the row arranging at 1 are omitted except one in the case of coset representatives of G'_A in G_A. Consequently, the order of G_{A142} is $4 \cdot 2 \cdot 8$.

We turn to computing n_{11}. From the column characteristic value, the order of G'''_{A1142} is 2! =2.

To compute coset representatives of G'''_{A1142} in G''_{A1142}, since the row arranging is restricted to be (1), from Theorem 8.2.7, for the result obtained from $A1142$ by applying a renaming β and reducing to a canonical form, the distribution of column types and row types of its blocks are coincided with ones for $A1142$, in particular, β transforms repeated columns of $A1142$ into repeated columns of the result. Since there is only one block of $A1142$ with repeated columns, β transforms the block with repeated columns into itself. It follows that $\beta = (1)$. Therefore, there is only one coset and e is a coset representative. It immediately follows that $G''_{A1142} = G'''_{A1142}$.

To compute coset representatives of G''_{A1142} in G'_{A1142}, let $\langle \alpha, \beta, \gamma \rangle \in G'$, where the row arranging α is a permutation of 2,3,4. Try $\alpha = (234)$. The row arranging (234) transforms $A1142$ into

$$A' = \begin{bmatrix} 11 & 22 & 33 & 44 \\ 44 & 13 & 12 & 32 \\ 22 & 34 & 41 & 13 \\ 33 & 41 & 24 & 21 \end{bmatrix}.$$

Suppose that A' can be transformed into $A1142$ by a renaming β and some column arranging. Since column types of the first block of A' and the first

block of $A1142$ are the same and different from column types of other blocks, from Theorem 8.2.7, the first block of A' is transformed into the first block of $A1142$ by renaming β and some column arranging. Thus β transforms the column 1423 into the column 1234; that is, $\beta = (234)$. It is easy to verify that $\langle (1), (234), \cdot \rangle$ transforms A' into $A1142$ indeed. Thus $\langle (234), (234), \cdot \rangle$ keeps $A1142$ unchanged. Similarly, it is easy to see that $\langle (34), (34), \cdot \rangle$ keeps $A1142$ unchanged. Note that the autotopisms generated by the two autotopisms contain six different autotopisms of which row arrangings are all the possible. From Theorem 8.2.6 (c), $(\langle (234), (234), \cdot \rangle, \langle (34), (34), \cdot \rangle)$ are coset representatives of all distinct cosets. That is, $G'_{A1142} = (\langle (234), (234), \cdot \rangle, \langle (34), (34), \cdot \rangle)$ G''_{A1142}. Therefore, the order of G'_{A1142} is $6 \cdot 2$.

To compute coset representatives of G'_{A1142} in G_{A1142}, let $\langle \alpha, \beta, \gamma \rangle \in G$. Try $\alpha = (1234)$. $A1142$ can be transformed into

$$A'' = \begin{bmatrix} 11 & 22 & 33 & 44 \\ 23 & 34 & 24 & 11 \\ 34 & 13 & 41 & 22 \\ 42 & 41 & 12 & 33 \end{bmatrix}$$

by row arranging (1234) and some column arranging. Suppose that A'' can be transformed into $A1142$ by a renaming β and some column arranging. Since column types of the fourth block of A'' and the first block of $A1142$ are the same and different from column types of other blocks, from Theorem 8.2.7, the fourth block of A'' is transformed into the first block of $A1142$ by renaming β and some column arranging. Thus β transforms the column 4123 into the column 1234; that is, $\beta = (1234)$. It is easy to verify that $\langle (1), (1234), \cdot \rangle$ transforms A'' into $A1142$ indeed. Thus $\langle (1234), (1234), \cdot \rangle$ keeps $A1142$ unchanged. From Theorem 8.2.6 (d), $(\langle (1234), (1234), \cdot \rangle)$ are coset representatives of all distinct cosets; that is, $G_{A1142} = (\langle (1234), (1234), \cdot \rangle)$ G'_{A1142}. Therefore, n_{11}, the order of G_{A1142}, is $4 \cdot 6 \cdot 2$.

Similarly, we can compute values of $n_2, \ldots, n_{10}$.

The following is the computing results in the format: "x: the order of G'''_{Ax42}, the number of cosets of G'''_{Ax42} in G''_{Ax42}, the number of cosets of G''_{Ax42} in G'_{Ax42}, the number of cosets of G'_{Ax42} in G_{Ax42} (the product of the four numbers is n_x); the set of coset representatives of G'''_{Ax42} in G''_{Ax42}; the set of coset representatives of G''_{Ax42} in G'_{Ax42}; the set of coset representatives of G'_{Ax42} in G_{Ax42}." γ in a coset representative $\langle \alpha, \beta, \gamma \rangle$ is omitted.

1 : $1, 8, 2, 4$; $(\langle (1), (34) \rangle, \langle (1), (1423) \rangle)$; $(\langle (34), (1) \rangle)$; $(\langle (1423), (1) \rangle)$.

2 : $1, 4, 2, 2$; $(\langle (1), (1234) \rangle)$; $(\langle (34), (1) \rangle)$; $(\langle (12), (12)(34) \rangle)$.

3 : $1, 2, 1, 4$; $(\langle (1), (13)(24) \rangle)$; e; $(\langle (13)(24), (1) \rangle, \langle (14)(23), (12)(34) \rangle)$.

4 : $1, 2, 2, 4$; $(\langle (1), (34) \rangle)$; $(\langle (24), (12) \rangle)$; $(\langle (4321), (1) \rangle)$.

5 : $1,1,2,3;\ e;\ (\langle(34),(34)\rangle);\ (\langle(431),(234)\rangle)$.

6 : $1,4,6,4;\ (\langle(1),(12)(34)\rangle,\langle(1),(14)(23)\rangle);\ (\langle(34),(23)\rangle,\langle(234),(234)\rangle);$
$(\langle(4321),(12)\rangle)$.

7 : $2^4,4,6,4;\ (\langle(1),(12)(34)\rangle,\langle(1),(13)(24)\rangle);\ (\langle(234),(234)\rangle,\langle(34),(34)\rangle);$
$(\langle(4321),(24)\rangle)$.

8 : $2^2,2,2,4;\ (\langle(1),(12)(34)\rangle);\ (\langle(34),(34)\rangle);\ (\langle(1423),(1423)\rangle)$.

9 : $2^4,4,2,4;\ (\langle(1),(1423)\rangle);\ (\langle(34),(34)\rangle);\ (\langle(1423),(1)\rangle)$.

10 : $2,1,2,4;\ e;\ (\langle(23),(23)\rangle);\ (\langle(1243),(1243)\rangle)$.

11 : $2,1,6,4;\ e;\ (\langle(234),(234)\rangle,\langle(34),(34)\rangle);\ (\langle(1234),(1234)\rangle)$.

Using the formulae at the beginning of the proof, we then have

$$
\begin{aligned}
U(4,2) &= \sum_{i=1}^{11} 4!4!8!/n_i \\
&= 4!4!8!(1/(8\cdot 2\cdot 4)+1/(4\cdot 2\cdot 2)+1/(2\cdot 4)+1/(2\cdot 2\cdot 4)+1/(2\cdot 3) \\
&\qquad +1/(4\cdot 6\cdot 4)+1/(2^4\cdot 4\cdot 6\cdot 4)+1/(2^2\cdot 2\cdot 2\cdot 4) \\
&\qquad +1/(2^4\cdot 4\cdot 2\cdot 4)+1/(2\cdot 2\cdot 4)+1/(2\cdot 6\cdot 4)) \\
&= 3\cdot 4!7!+3!3!8!+4!4!7!+3!3!8!+4\cdot 4!8!+3!8! \\
&\qquad +3\cdot 7!+2\cdot 3!3!7!+9\cdot 7!+3!3!8!+2\cdot 3!8! \\
&= 7!(72+576+3+72+9)+8!(36+36+96+6+36+12) \\
&= 7!732+8!222 = 7!2508 = 12640320, \\
U(4,2,1) &= \sum_{i=1}^{6} 4!4!8!/n_i \\
&= 3\cdot 4!7!+3!3!8!+4!4!7!+3!3!8!+4\cdot 4!8!+3!8! \\
&= 7!(72+576)+8!(36+36+96+6) \\
&= 7!648+8!174 = 7!2040 = 10281600, \\
U(4,2,2) &= 4!4!8!/n_7+4!4!8!/n_9 \\
&= 4!4!8!/(2^4\cdot 4\cdot 6\cdot 4)+4!4!8!/(2^4\cdot 4\cdot 2\cdot 4) \\
&= 3\cdot 7!+9\cdot 7! = 60480.
\end{aligned}
$$

We obtain the results of the theorem. □

Corollary 8.2.2. $I(4,1) = I(4,1,1) = 2$, $U(4,1) = U(4,1,1) = (4!)^2$.

Proof. From Theorem 8.2.1, we have $I(4,1) = I(4,1,1) = I(4,2,2) = 2$, and $U(4,1,1) = U(4,2,2)4!(2!)^4/8!$. Thus $U(4,1) = U(4,1,1) = 60480/105 = 576 = (4!)^2$. □

Enumeration of (4, 3)-Latin Arrays

Let $A143, \ldots, A4643$ be (4,3)-Latin arrays as follows:

$$\begin{bmatrix} 111222333444 \\ 222111444333 \\ 333444111222 \\ 444333222111 \end{bmatrix} A143 \quad \begin{bmatrix} 111222333444 \\ 222111444333 \\ 333444222111 \\ 444333111222 \end{bmatrix} A243 \quad \begin{bmatrix} 111222333444 \\ 222111444333 \\ 333444112221 \\ 444333221112 \end{bmatrix} A343 \quad \begin{bmatrix} 111222333444 \\ 222111444333 \\ 333444221112 \\ 444333112221 \end{bmatrix} A443$$

$$\begin{bmatrix} 111222333444 \\ 222133444113 \\ 333444112221 \\ 444311221332 \end{bmatrix} A543 \quad \begin{bmatrix} 111222333444 \\ 222131444313 \\ 333444112221 \\ 444313221132 \end{bmatrix} A643 \quad \begin{bmatrix} 111222333444 \\ 222434141133 \\ 333141424212 \\ 444313212321 \end{bmatrix} A743 \quad \begin{bmatrix} 111222333444 \\ 222431144133 \\ 333144421221 \\ 444313212312 \end{bmatrix} A843$$

$$\begin{bmatrix} 111222333444 \\ 222111444333 \\ 334443112221 \\ 443334221112 \end{bmatrix} A943 \quad \begin{bmatrix} 111222333444 \\ 222111444333 \\ 334443221112 \\ 443334112221 \end{bmatrix} A1043 \quad \begin{bmatrix} 111222333444 \\ 222333444111 \\ 334441112322 \\ 443114221233 \end{bmatrix} A1143 \quad \begin{bmatrix} 111222333444 \\ 222331444113 \\ 334443112221 \\ 443114221332 \end{bmatrix} A1243$$

$$\begin{bmatrix} 111222333444 \\ 222313444113 \\ 334441112232 \\ 443134221321 \end{bmatrix} A1343 \quad \begin{bmatrix} 111222333444 \\ 222313444113 \\ 334441212231 \\ 443134121322 \end{bmatrix} A1443 \quad \begin{bmatrix} 111222333444 \\ 222131444331 \\ 334443212112 \\ 443314121223 \end{bmatrix} A1543 \quad \begin{bmatrix} 111222333444 \\ 222131444331 \\ 334443112212 \\ 443314221123 \end{bmatrix} A1643$$

$$\begin{bmatrix} 111222333444 \\ 222113444331 \\ 334441112223 \\ 443334221112 \end{bmatrix} A1743 \quad \begin{bmatrix} 111222333444 \\ 222443114133 \\ 334111442322 \\ 443334221211 \end{bmatrix} A1843 \quad \begin{bmatrix} 111222333444 \\ 222314441331 \\ 334441222113 \\ 443133114222 \end{bmatrix} A1943 \quad \begin{bmatrix} 111222333444 \\ 222134441331 \\ 334413114222 \\ 443341222113 \end{bmatrix} A2043$$

$$\begin{bmatrix} 111222333444 \\ 222341441331 \\ 334413114222 \\ 443134222113 \end{bmatrix} A2143 \quad \begin{bmatrix} 111222333444 \\ 222334441113 \\ 334441212321 \\ 443113124232 \end{bmatrix} A2243 \quad \begin{bmatrix} 111222333444 \\ 222334441113 \\ 334411124322 \\ 443143212231 \end{bmatrix} A2343 \quad \begin{bmatrix} 111222333444 \\ 222334441113 \\ 334411224321 \\ 443143112232 \end{bmatrix} A2443$$

$$\begin{bmatrix} 111222333444 \\ 222314441133 \\ 334431124221 \\ 443143212312 \end{bmatrix} A2543 \quad \begin{bmatrix} 111222333444 \\ 222314441133 \\ 334431224211 \\ 443143112322 \end{bmatrix} A2643 \quad \begin{bmatrix} 111222333444 \\ 222314441133 \\ 334443112221 \\ 443131224312 \end{bmatrix} A2743 \quad \begin{bmatrix} 111222333444 \\ 222314441133 \\ 334443122211 \\ 443131214322 \end{bmatrix} A2843$$

$$\begin{bmatrix}111222333444\\222314441133\\334143214221\\443431122312\end{bmatrix}\ A2943 \qquad \begin{bmatrix}111222333444\\223443421311\\334114242132\\442331114223\end{bmatrix}\ A3043 \qquad \begin{bmatrix}111222333444\\223114144233\\334443211122\\442331422311\end{bmatrix}\ A3143 \qquad \begin{bmatrix}111222333444\\223114144233\\334443212121\\442331421312\end{bmatrix}\ A3243$$

$$\begin{bmatrix}111222333444\\223441124331\\334134241212\\442313412123\end{bmatrix}\ A3343 \qquad \begin{bmatrix}111222333444\\223441114332\\334134242121\\442313421213\end{bmatrix}\ A3443 \qquad \begin{bmatrix}111222333444\\223443114123\\334314242211\\442131421332\end{bmatrix}\ A3543 \qquad \begin{bmatrix}111222333444\\234134124123\\342341241231\\423413412312\end{bmatrix}\ A3643$$

$$\begin{bmatrix}111222333444\\234134124123\\342413241312\\423341412231\end{bmatrix}\ A3743 \qquad \begin{bmatrix}111222333444\\234134124123\\423341241231\\342413412312\end{bmatrix}\ A3843 \qquad \begin{bmatrix}111222333444\\332441124123\\424313241231\\243134412312\end{bmatrix}\ A3943 \qquad \begin{bmatrix}111222333444\\332441124123\\424313241312\\243134412231\end{bmatrix}\ A4043$$

$$\begin{bmatrix}111222333444\\332441124123\\424313412231\\243134241312\end{bmatrix}\ A4143 \qquad \begin{bmatrix}111222333444\\234334114221\\423141242313\\342413421132\end{bmatrix}\ A4243 \qquad \begin{bmatrix}111222333444\\234331441221\\423414212313\\342143124132\end{bmatrix}\ A4343 \qquad \begin{bmatrix}111222333444\\224113442331\\343434121212\\432341214123\end{bmatrix}\ A4443$$

$$\begin{bmatrix}111222333444\\224113441332\\343434212121\\432341124213\end{bmatrix}\ A4543 \qquad \begin{bmatrix}111222333444\\224113442331\\343341214212\\432434121123\end{bmatrix}\ A4643 .$$

Lemma 8.2.6. *$A143, \ldots, A4643$ are not isotopic to each other.*

Proof. We compute the column characteristic value and the row characteristic set of $Ax43$ and represent them in the format: "x: the column characteristic value of $Ax43$; the row characteristic set of $Ax43$", where the column characteristic value is in the form c_3c_2, the row characteristic set is in the form $T_1(i,j)\ T'(i,j)$, in order of $ij = 12, 13, 14, 23, 24, 34$, $T'(i,j) = T_2(i,j)$ if the derived permutation from row i to row j exists, $T'(i,j) = T_3(i,j)$ otherwise. $T_2(i,j) = 4$ means "a cycle of length 4"; $T_2(i,j) = 2$ means "two transpositions". We list the results of column characteristic values and row characteristic sets of $A143$, ..., $A4643$ as follows:

$1 : 40; 402, 402, 402, 402, 402, 402$ $\quad 2 : 40; 402, 404, 404, 404, 404, 402$

$3 : 22; 402, 222, 222, 222, 222, 402$ $\quad 4 : 22; 402, 224, 224, 224, 224, 402$

$5 : 13; 224, 222, 134, 134, 222, 224$ $\quad 6 : 12; 222, 222, 120, 120, 222, 222$

$7 : 12; 134, 134, 120, 120, 134, 134$ $\quad 8 : 10; 120, 120, 120, 120, 120, 120$

$9 : 04; 402, 042, 042, 042, 042, 402$ $10 : 04; 402, 044, 044, 044, 044, 402$

$11 : 04; 404, 042, 044, 044, 042, 044$ $12 : 04; 224, 042, 044, 044, 042, 224$

$13 : 02; 224, 042, 020, 020, 042, 042$ $14 : 02; 224, 032, 032, 032, 032, 042$

$15 : 03; 222, 044, 032, 032, 044, 222$ $16 : 02; 222, 042, 020, 020, 042, 222$

$17 : 04; 222, 042, 042, 042, 042, 222$ $18 : 04; 134, 134, 042, 042, 134, 134$

$19 : 03; 120, 134, 134, 032, 032, 042$ $20 : 03; 120, 120, 120, 042, 042, 042$

$21 : 03; 120, 120, 120, 120, 120, 120$ $22 : 02; 134, 032, 033, 044, 020, 032$

$23 : 01; 134, 032, 020, 020, 032, 020$ $24 : 02; 134, 033, 032, 032, 033, 020$

$25 : 01; 120, 020, 020, 020, 020, 120$ $26 : 03; 120, 032, 032, 032, 032, 120$

$27 : 02; 120, 042, 032, 032, 042, 120$ $28 : 02; 120, 044, 033, 033, 044, 120$

$29 : 01; 120, 020, 020, 032, 032, 042$ $30 : 02; 033, 033, 044, 044, 033, 033$

$31 : 04; 042, 042, 042, 042, 042, 042$ $32 : 02; 042, 044, 020, 020, 044, 042$

$33 : 01; 032, 020, 020, 032, 032, 020$ $34 : 01; 044, 032, 020, 020, 032, 044$

$35 : 01; 032, 032, 032, 032, 032, 032$ $36 : 00; 000, 000, 000, 044, 044, 044$

$37 : 00; 000, 000, 000, 000, 000, 000$ $38 : 00; 000, 000, 000, 033, 033, 033$

$39 : 00; 020, 020, 000, 033, 032, 032$ $40 : 00; 020, 020, 000, 044, 020, 020$

$41 : 00; 020, 020, 000, 000, 020, 020$ $42 : 00; 032, 032, 000, 000, 032, 032$

$43 : 00; 033, 033, 000, 000, 033, 033$ $44 : 00; 042, 042, 000, 000, 042, 042$

$45 : 00; 042, 044, 000, 000, 044, 042$ $46 : 00; 042, 020, 020, 020, 020, 042$

It is easy to verify that for any two distinct $Ai43$'s either their column characteristic values are different, or their row characteristic sets are different. From Corollary 8.2.1, it immediately follows that $A143, \ldots, A4643$ are not isotopic to each other. □

Lemma 8.2.7. *Any* $(4, 3)$*-Latin array is isotopic to one of* $A143, \ldots, A4643$*; and any* $(4, 3, 1)$*-Latin array is isotopic to one of* $A3643, \ldots, A4643$.

Proof. The proof of this lemma is similar to Lemma 8.2.5 but more tedious. We omit the details of the proof for the sake of space. □

Theorem 8.2.12. $I(4, 3) = 46,\ I(4, 3, 1) = 11.$

Proof. This is immediate from Lemmas 8.2.6 and 8.2.7. □

Theorem 8.2.13. $U(4, 3, 1) = 306561024000,\ U(4, 3) = 805929062400.$

Proof. For any $Ax43$, G'''_{Ax43} is easy to determine from positions of repeated columns. For computing the order of G_{Ax43}, we find out the set of

coset representatives of G'''_{Ax43} in G''_{Ax43}, the set of coset representatives of G''_{Ax43} in G'_{Ax43} and the set of coset representatives of G'_{Ax43} in G_{Ax43}.

Below we give an example for G_{A1743}

$$A1743 = \begin{bmatrix} 111\ 222\ 333\ 444 \\ 222\ 113\ 444\ 331 \\ 334\ 441\ 112\ 223 \\ 443\ 334\ 221\ 112 \end{bmatrix}.$$

In GR_{A1743}, labels of edges $(1,2)$ and $(3,4)$ are the same, say "red"; labels of other edges are the same and not red, say "green".

G'''_{A1743} consists of the product of the following permutations: permutations of columns 1 and 2, permutations of columns 4 and 5, permutations of columns 7 and 8, permutations of columns 10 and 11; its order is $(2!)^4$.

Let $\langle (1), \beta, \gamma \rangle \in G''_{A1743}$. Although the column types of various blocks of $A1743$ are the same, the first block and the third block have the same row type which is different from row types of other blocks. From Theorem 8.2.7, the first block should be transformed into itself or the third block by renaming β and some column arranging. In the case of being transformed into itself, the renaming β transforms the repeated column 1234 into itself; that is, $\beta = (1)$. This gives the coset representative e. In the case of being transformed into the third block, β transforms the repeated column 1234 into the repeated column 3412; that is, $\beta = (13)(24)$. It is easy to verify that $\langle (1), (13)(24), \cdot \rangle$ keeps $A1743$ unchanged indeed; this gives another coset representative. From Theorem 8.2.6 (b), it follows that $G''_{A1244} = (\langle (1), (13)(24), \cdot \rangle)\ G'''_{A1244}$, of which the order is $2 \cdot (2!)^4$.

To find G'_{A1743}, let $\langle \alpha, \beta, \gamma \rangle \in G'$, where the row arranging α is a permutation of 2,3,4. From the proof of Theorem 8.2.3 (b), if $\langle \alpha, \beta, \gamma \rangle$ is an autotopism of $A1743$, then the row arranging α is an automorphism of GR_{A1743}. It follows that edges $(1,2)$ and $(\alpha(1), \alpha(2))$, that is, $(1, \alpha(2))$, have the same color. Since the edge $(1,2)$ is the unique red edge with endpoint 1, we have $\alpha(2) = 2$ whenever $\langle \alpha, \beta, \gamma \rangle \in G'_{A244}$. Thus the candidates of α are (34) and (1). Try $\alpha = (34)$. The row arranging (34) transforms $A1743$ into

$$A' = \begin{bmatrix} 111\ 222\ 333\ 444 \\ 222\ 113\ 444\ 331 \\ 443\ 334\ 221\ 112 \\ 334\ 441\ 112\ 223 \end{bmatrix}.$$

If A' can be transformed into $A1743$ by a renaming β and some column arranging, from Theorem 8.2.7, then the first block of A' should be transformed into the first block or the third block of $A1743$ by renaming β and some column arranging. In the case of being transformed into the first block,

β transforms the repeated column 1243 into the repeated column 1234; that is, $\beta = (34)$. It is easy to see that β transforms the column 2341 into the column 2431, which is not a column of $A1743$. It follows that A' can not be transformed into $A1743$ by $\langle (1), (34), \cdot \rangle$. In the case of being transformed into the third block, β transforms the repeated column 1243 into the repeated column 3412; that is, $\beta = (1324)$. It is easy to see that β transforms the column 2143 into the column 4312, which is not a column of $A1743$. It follows that A' can not be transformed into $A1743$ by $\langle (1), (1324), \cdot \rangle$. To sum up, $\langle (34), \beta, \cdot \rangle$ can not keep $A1743$ unchanged for any β. Therefore, from Theorem 8.2.6 (c), e is a coset representative of the unique coset. That is, $G'_{A1743} = G''_{A1743}$, of which the order is $2 \cdot (2!)^4$.

To find G_{A1743}, let $\langle \alpha, \beta, \gamma \rangle \in G_{A1743}$. Thus α is an automorphism of PR_{A1743}. Try $\alpha(3) = 1$. Since the edge $(3, 4)$ is red, the edge $(\alpha(3), \alpha(4))$, that is, $(1, \alpha(4))$, is red. Since the edge $(1, 2)$ is the unique red edge with endpoint 1, we have $\alpha(4) = 2$. It follows that $\alpha = (1423)$ or $(13)(24)$. Try $\alpha = (1423)$. $A1743$ can be transformed into

$$A'' = \begin{bmatrix} 111 & 222 & 333 & 444 \\ 422 & 111 & 442 & 333 \\ 344 & 433 & 221 & 211 \\ 233 & 344 & 114 & 122 \end{bmatrix}$$

by row arranging (1423) and some column arranging. If A'' can be transformed into $A1743$ by a renaming β and some column arranging, from Theorem 8.2.7, then the second block of A'' should be transformed into the first block or the third block of $A1743$ by renaming β and some column arranging. In the case of being transformed into the first block, β transforms the repeated column 2134 into the repeated column 1234; that is, $\beta = (12)$. It is easy to see that β transforms the column 1432 into the column 2431, which is not a column of $A1744$. In the case of being transformed into the third block, β transforms the repeated column 2134 into the repeated column 3412; that is, $\beta = (1423)$. It is easy to see that β transforms the column 1432 into the column 4213, which is not a column of $A1744$. To sum up, for any renaning β, $\langle (1), \beta, \cdot \rangle$ can not transform A'' into $A1743$. Thus $\langle (1324), \beta, \cdot \rangle$ can not keep $A1743$ unchanged for any β. Try $\alpha = (13)(24)$. $A1743$ can be transformed into

$$A''' = \begin{bmatrix} 111 & 222 & 333 & 444 \\ 422 & 111 & 442 & 333 \\ 233 & 344 & 114 & 122 \\ 344 & 433 & 221 & 211 \end{bmatrix}$$

by row arranging (13)(24) and some column arranging. If A''' can be transformed into $A1743$ by a renaming β and some column arranging, from Theorem 8.2.7, then the second block of A''' should be transformed into the first

block or the third block of $A1743$ by renaming β and some column arranging. Try the first block. In this case, β transforms the repeated column 2143 into the repeated column 1234; that is, $\beta = (12)(34)$. It is easy to verify that A''' can be transformed into $A1743$ by $\langle (1), (12)(34), \cdot \rangle$ indeed. Therefore, $\langle (13)(24), (12)(34), \cdot \rangle$ keeps $A1743$ unchanged. Similarly, Try $\alpha(2) = 1$. Since the edge $(1, 2)$ is red, the edge $(\alpha(1), \alpha(2))$, that is, $(\alpha(1), 1)$, is red. Since the edge $(2, 1)$ is the unique red edge with endpoint 1, we have $\alpha(1) = 2$. It follows that $\alpha = (12)(34)$ or (12). Try $\alpha = (12)(34)$. $A1743$ can be transformed into

$$A'''' = \begin{bmatrix} 111\ 222\ 333\ 444 \\ 224\ 111\ 244\ 333 \\ 332\ 443\ 411\ 221 \\ 443\ 334\ 122\ 112 \end{bmatrix}$$

by row arranging $(12)(34)$ and some column arranging. Suppose that A'''' can be transformed into $A1743$ by a renaming β and some column arranging. From Theorem 8.2.7, the second block of A'''' should be transformed into the first block or the third block of $A1743$ by renaming β and some column arranging. Try the first block. In this case, β transforms the repeated column 2143 into the repeated column 1234; that is, $\beta = (12)(34)$. It is easy to verify that A'''' can be transformed into $A1743$ by $\langle (1), (12)(34), \cdot \rangle$ indeed. Therefore, $\langle (12)(34), (12)(34), \cdot \rangle$ keeps $A1743$ unchanged. From $(13)(24) \cdot (12)(34) = (14)(23)$ and $(12)(34) \cdot (12)(34) = (1)$, $\langle (14)(23), (1), \cdot \rangle$ keeps $A1743$ unchanged. From Theorem 8.2.6 (d), we then have $G_{A1743} = (\langle (13)(24), (12)(34), \cdot \rangle, \langle (12)(34), (12)(34), \cdot \rangle)\, G'_{A1743}$, of which the order is $4 \cdot 2 \cdot (2!)^4$.

Similarly, we can compute other G_{Ax43}, using GR_{Ax43} to reduce the trying scope for row arranging, and using column types and row types of blocks to reduce the trying scope for renaming.

Denote the order of autotopism group G_{Ai43} of $Ai43$ by $n_i, i = 1, \ldots, 46$. On autotopism group of $Ax43$ and its order n_x, the computing results are represented in the format: "x: the order of G'''_{Ax43}, the number of cosets of G'''_{Ax43} in G''_{Ax43}, the number of cosets of G''_{Ax43} in G'_{Ax43}, the number of cosets of G'_{Ax43} in G_{Ax43} (the product of the four numbers is n_x); the set of coset representatives of G'''_{Ax43} in G''_{Ax43}; the set of coset representatives of G''_{Ax43} in G'_{Ax43}; the set of coset representatives of G'_{Ax43} in G_{Ax43}." γ in a coset representative $\langle \alpha, \beta, \gamma \rangle$ is omitted.

1 : $(3!)^4, 4, 6, 4;\ (\langle (1), (12)(34) \rangle, \langle (1), (13)(24) \rangle);$
$(\langle (234), (234) \rangle, \langle (23), (23) \rangle);\ (\langle (1234) \rangle, \langle (4321) \rangle).$

2 : $(3!)^4, 4, 2, 4;\ (\langle (1), (1423) \rangle);\ (\langle (34), (34) \rangle);\ (\langle (1423), (1) \rangle).$

3 : $(3!2!)^2, 2, 2, 4;\ (\langle (1), (12)(34) \rangle);\ (\langle (34), (34) \rangle);\ (\langle (1324), (1423) \rangle).$

4 : $(3!2!)^2, 2, 2, 4;\ (\langle(1), (12)(34)\rangle);\ (\langle(34), (34)\rangle);\ (\langle(1324), (1324)\rangle)$.

5 : $3!2^3, 1, 1, 4;\ e;\ e;\ (\langle(12)(34), (12)(34)\rangle, \langle(14)(23), (14)(23)\rangle)$.

6 : $3!2^2, 1, 2, 4;\ e;\ (\langle(23), (23)\rangle);\ (\langle(1243), (1243)\rangle)$.

7 : $3!2^2, 1, 2, 4;\ e;\ (\langle(23), (23)\rangle);\ (\langle(1243), (1243)\rangle)$.

8 : $3!, 1, 6, 4;\ e;\ (\langle(234), (234)\rangle, \langle(34), (34)\rangle);\ (\langle(1234), (1234)\rangle)$.

9 : $2^4, 4, 2, 4;\ (\langle(1), (12)(34)\rangle, \langle(1), (13)(24)\rangle);\ (\langle(34), (34)\rangle);\ (\langle(1324), (34)\rangle)$.

10 : $2^4, 4, 2, 4;\ (\langle(1), (1423)\rangle);\ (\langle(34), (34)\rangle);\ (\langle(1324), (1)\rangle)$.

11 : $2^4, 4, 1, 2;\ (\langle(1), (1234)\rangle);\ e;\ (\langle(12)(34), (12)(34)\rangle)$.

12 : $2^4, 2, 1, 4;\ (\langle(1), (13)(24)\rangle);\ e;\ (\langle(12)(34), (12)(34)\rangle, \langle(13)(24), (1)\rangle)$.

13 : $2^2, 2, 1, 2;\ (\langle(1), (13)(24)\rangle);\ e;\ (\langle(12)(34), (12)(34)\rangle)$.

14 : $2^2, 1, 2, 2;\ e;\ (\langle(34), (13)(24)\rangle);\ (\langle(12), (14)(23)\rangle)$.

15 : $2^3, 1, 1, 4;\ e;\ e;\ (\langle(12)(34), (12)(34)\rangle, \langle(13)(24), (13)(24)\rangle)$.

16 : $2^2, 2, 1, 4;\ (\langle(1), (13)(24)\rangle);\ e;\ (\langle(12)(34), (12)(34)\rangle, \langle(13)(24), (1)\rangle)$.

17 : $2^4, 2, 1, 4;\ (\langle(1), (13)(24)\rangle);\ e;\ (\langle(13)(24), (12)(34)\rangle, \langle(12)(34), (12)(34)\rangle)$.

18 : $2^4, 1, 2, 4;\ e;\ (\langle(23), (12)(34)\rangle);\ (\langle(1243), (1)\rangle)$.

19 : $2^3, 1, 2, 1;\ e;\ (\langle(34), (34)\rangle);\ e$.

20 : $2^3, 1, 6, 1;\ e;\ (\langle(23), (14)\rangle, \langle(34), (34)\rangle);\ e$.

21 : $2^3, 1, 6, 4;\ e;\ (\langle(234), (143)\rangle, \langle(34), (34)\rangle);\ (\langle(1234), (4321)\rangle)$.

22 : $2^2, 1, 1, 1;\ e;\ e;\ e$.

23 : $2, 1, 1, 2;\ e;\ e;\ (\langle(12)(34), (12)(34)\rangle)$.

24 : $2^2, 1, 1, 2;\ e;\ e;\ (\langle(12)(34), (12)(34)\rangle)$.

25 : $2, 1, 2, 4;\ e;\ (\langle(34), (34)\rangle);\ (\langle(1423), (1423)\rangle)$.

26 : $2^3, 1, 2, 4;\ e;\ (\langle(34), (34)\rangle);\ (\langle(1423), (1423)\rangle)$.

27 : $2^2, 1, 1, 4;\ e;\ e;\ (\langle(12)(34), (12)(34)\rangle, \langle(14)(23), (12)(34)\rangle)$.

28 : $2^2, 1, 1, 4;\ e;\ e;\ (\langle(13)(24), (34)\rangle, \langle(14)(23), (12)\rangle)$.

29 : $2, 1, 2, 1;\ e;\ (\langle(34), (34)\rangle);\ e$.

30 : $2^2, 1, 2, 4;\ e;\ (\langle(23), (12)(34)\rangle);\ (\langle(1243), (1243)\rangle)$.

31 : $2^4, 4, 3, 4;\ (\langle(1), (12)(34)\rangle, \langle(1), (13)(24)\rangle);$
$(\langle(234), (234)\rangle);\ e, \langle(143), (143)\rangle, \langle(13)(24), (13)(24)\rangle, \langle(123), (123)\rangle$.

32 : $2^2, 2, 1, 4;\ (\langle(1), (12)(34)\rangle);\ e;\ (\langle(12)(34), (1)\rangle, \langle(14)(23), (14)(23)\rangle)$.

33 : $2, 1, 1, 3;\ e;\ e;\ (\langle(134), (134)\rangle)$.

34 : $2, 1, 1, 4;\ e;\ e;\ (\langle(12)(34), (12)(34)\rangle, \langle(14)(23), (14)(23)\rangle)$.

35 : $2, 1, 3, 4;\ e;\ (\langle(234), (234)\rangle);$
$e, \langle(124), (124)\rangle, \langle(421), (421)\rangle, \langle(132), (132)\rangle$.

36 : $1,4,6,1;\ (\langle(1),(1234)\rangle);\ (\langle(23),(24)\rangle,\langle(24),(24)\rangle);\ e.$
37 : $1,12,6,4;\ \langle(1),\pi\rangle,\pi$ is an even permutation;
$(\langle(234),(1)\rangle,\langle(23),(1234)\rangle);\ (\langle(1234),(1234)\rangle).$
38 : $1,3,6,1;\ (\langle(1),(124)\rangle);\ (\langle(234),(1)\rangle,\langle(23),(24)\rangle);\ e.$
39 : $1,1,2,1;\ e;\ (\langle(23),(12)\rangle);\ e.$
40 : $1,2,2,2;\ (\langle(1),(12)(34)\rangle);\ (\langle(23),(34)\rangle);\ (\langle(14)(23),(13)(24)\rangle).$
41 : $1,2,2,4;\ (\langle(1),(12)(34)\rangle);\ (\langle(23),(12)\rangle);\ (\langle(1342),(1423)\rangle).$
42 : $1,1,2,4;\ e;\ (\langle(23),(23)\rangle);\ (\langle(1243),(14)\rangle).$
43 : $1,3,2,4;\ (\langle(1),(234)\rangle);\ (\langle(23),(34)\rangle);\ (\langle(1342),(1)\rangle).$
44 : $1,4,2,4;\ (\langle(1),(12)(34)\rangle,\langle(1),(14)(23)\rangle);(\langle(23),(23)\rangle);(\langle(1342),(1342)\rangle).$
45 : $1,4,1,4;\ (\langle(1),(1423)\rangle);\ e;\ (\langle(12)(34),(12)\rangle,\langle(14)(23),(1)\rangle).$
46 : $1,2,2,4;\ (\langle(1),(14)(23)\rangle);\ (\langle(34),(12)(34)\rangle);\ (\langle(1423),(1)\rangle).$

Thus $n_1,\ldots,n_{46}$ are

$$3^5 2^9, 3^4 2^9, 3^2 2^8, 3^2 2^8, 3\cdot 2^6, 3\cdot 2^6, 3\cdot 2^6, 3^2 2^4, 2^9, 2^9,$$
$$2^7, 2^7, 2^4, 2^4, 2^5, 2^5, 2^7, 2^7, 2^4, 3\cdot 2^4,$$
$$3\cdot 2^6, 2^2, 2^2, 2^3, 2^4, 2^6, 2^4, 2^4, 2^2, 2^5,$$
$$3\cdot 2^8, 2^5, 3\cdot 2, 2^3, 3\cdot 2^3, 3\cdot 2^3, 3^2 2^5, 3^2 2, 2, 2^3,$$
$$2^4, 2^3, 3\cdot 2^3, 2^5, 2^4, 2^4,$$

respectively. Since the number of elements in the isotopy class containing $Ai43$ is $4!4!12!/n_i$, noticing $4!4!12! = 3^2 2^6 12! = 3^7 2^{14} 7700$, we have

$$\begin{aligned} U(4,3,1) &= \sum_{i=36}^{46} 4!4!12!/n_i \\ &= 12!(24+2+32+288+72+36+72+24+18+36+36) \\ &= 12!640 = 479001600\cdot 640 = 306561024000, \\ U(4,3) &= \sum_{i=1}^{46} 4!4!12!/n_i = 805929062400. \end{aligned}$$

□

Enumeration of (4, 4)-Latin Arrays

Notice that the first row of any canonical (4,4)-Latin array is 111122223333 4444. Since each column of any canonical (4,4)-Latin array is a permutation of 1,2,3 and 4, any canonical (4,4)-Latin array may be determined by its rows 2 and 3. For example, if rows 2 and 3 of a canonical (4,4)-Latin array A are 2222111144443333 and 3333444411112222, respectively, then we have

$$A = \begin{bmatrix} 1111222233334444 \\ 2222111144443333 \\ 3333444411112222 \\ 4444333322221111 \end{bmatrix}.$$

We give a (4,4)-Latin array $Ax44$ in the format: "x: the second row of $Ax44$, the third row of $Ax44$". Let $A144, \ldots, A20144$ be 201 (4,4)-Latin arrays given as follows:

1 : 2222111144443333, 3333444411112222
2 : 2222111144443333, 3333444422221111
3 : 2222111144443333, 3333444411122221
4 : 2222111144443333, 3333444422211112
5 : 2222111144443333, 3333444411222211
6 : 2222111144443333, 3334444311122221
7 : 2222111144443333, 3334444322211112
8 : 2222111144443333, 3334444311222211
9 : 2222111144443333, 3344443311222211
10 : 2222333344441111, 3334444111122223
11 : 2222333344441111, 3344441111223322
12 : 3333444421111222, 4444333112222113
13 : 3333444422211112, 4444333111122223
14 : 3333444422111122, 4444333111222213
15 : 3333444422112211, 4444331111223322
16 : 3333444411122221, 4442333122241113
17 : 3333444411222211, 4442333122141123
18 : 3333444412222111, 4442333121141223
19 : 3333444421111222, 4442333112242113
20 : 3333444422111122, 4442333111242213
21 : 3333444422211112, 4442333111142223
22 : 3333444411121222, 4442111322243331
23 : 3333444421121221, 4442111312243332
24 : 3333444412221211, 4442331121143322
25 : 3333444412121212, 4442331121243321
26 : 3333444411121222, 4442331122243311
27 : 3333444412212211, 4442331121143322
28 : 3333444411212221, 4442331122143312
29 : 3333444411122122, 2244113344211233
30 : 3333444411221122, 2244113344112233
31 : 3333444412122121, 2244113344211233
32 : 3333444412221121, 2244113344112233
33 : 2222333444411113, 3333444121142221
34 : 2222113444413331, 3333444111242212
35 : 2222113444413331, 3333444112242112
36 : 2222331444413311, 3333444122141122
37 : 2222443311441133, 3333114444222211
38 : 2222333444411113, 3334444111222231
39 : 2222333444411113, 3334444111122232
40 : 2222333444411113, 3334444311122221
41 : 2222333444411113, 3334441112242231
42 : 2222333444411113, 3334441111242232
43 : 2222113444413331, 3334434122241112

44 : 2222113444413331, 3334434122141122
45 : 2222113444413331, 3334434121141222
46 : 2222113444413331, 3334434111142222
47 : 2222113444413331, 3334444322121112
48 : 2222113444413331, 3334444321121122
49 : 2222113444413331, 3334444311121222
50 : 2222113444413331, 3334441322141122
51 : 2222113444413331, 3334441321141222
52 : 2222113444413331, 3334441311142222
53 : 2222133444411133, 3334344122142211
54 : 2222133444411133, 3334344121142212
55 : 2222133444411133, 3334344111142222
56 : 2222133444411133, 3334414312142221
57 : 2222133444411133, 3334414322142211
58 : 2222133444411133, 3334414121142322
59 : 2222133444411133, 3334414121242321
60 : 2222133444411133, 3334414122242311
61 : 2222133444411133, 3334444311122212
62 : 2222133444411133, 3334444321122211
63 : 2222133444411133, 3334444111122322
64 : 2222334444111122, 3334413121442221
65 : 2222334444111133, 3334413122442211
66 : 2222334444111133, 3334443111422221
67 : 2222334444111133, 3334443121422211
68 : 2222334444111133, 3334441111423222
69 : 2222333444411113, 3334411122143322
70 : 2222333444411113, 3334411122243321
71 : 2222333444411113, 3334411321142322
72 : 2222113444413331, 3334334121141222
73 : 2222113444413331, 3334334121241212
74 : 2222113444413331, 3334334122241112
75 : 2222113444413331, 3334431121242213
76 : 2222113444413331, 3334434111221223
77 : 2222133444411133, 3334344321122211
78 : 2222133444411133, 3334314321142221
79 : 2222133444411133, 3334314322142211
80 : 2222133444411133, 3334314121142322
81 : 2222133444411133, 3334314122142312
82 : 2222133444411133, 3334314122242311
83 : 2222334444111133, 3334443311222211
84 : 2222334444111133, 3334413311422221
85 : 2222334444111133, 3334413111422322
86 : 2222334444111133, 3334413121422312
87 : 2222334444111133, 3334441111223322
88 : 2222334444111133, 3334441112223312
89 : 2223111444413332, 3334444311122221
90 : 2223111444413332, 3334444311222211
91 : 2223111444413332, 3334444312222111
92 : 2223111444413332, 4332443321242111
93 : 2223111444413332, 4332443321142121
94 : 2223111444423331, 4332443121242113
95 : 2223111444413332, 4332443121242113
96 : 2223111444413332, 4332443121142123

97 : 2223111444423331, 4332443121142123
98 : 2223111444413332, 3344443321121221
99 : 2223333444411112, 3334441122143221
100 : 2224333144421113, 3332441411243221
101 : 2223333444411112, 3444411112222333
102 : 2223333444411112, 3444411321123223
103 : 2223333444411112, 3444411312222331
104 : 2224333144421113, 4333411412143222
105 : 2224333144421113, 4333411412243221
106 : 2224333144421113, 4433114421213232
107 : 2224333144421113, 4433414321212231
108 : 2224333144411123, 3332144421142312
109 : 2224333144411123, 3332144422142311
110 : 2224333144411123, 3342444311222311
111 : 2224333144411123, 3343444311222211
112 : 2224333144411123, 3442411412123332
113 : 2224333144411123, 3442411412223331
114 : 2224333144411123, 3442414312123231
115 : 2224333144411123, 3442411312143232
116 : 2224333144411123, 3442411312243231
117 : 2224333144411123, 3442414312123312
118 : 2224333144411123, 3442414312223311
119 : 2224333144411123, 3443411312242231
120 : 2224333144411123, 3443411312242312
121 : 2224333144411123, 3432414312142231
122 : 2224333144411123, 3432411412142332
123 : 2224333444111123, 3332441122443211
124 : 2224333444111123, 3332444121423211
125 : 2224333444111123, 4333411321442212
126 : 2224333444111123, 4333411322442211
127 : 2224333444111123, 4333411122442312
128 : 2224333444111123, 4333411122442231
129 : 2224333444111123, 4333441321422211
130 : 2224333444111123, 4333441121422312
131 : 2224333444111123, 4333441122422311
132 : 2224333444111123, 4333441112422231
133 : 2224333444111123, 4433144312222311
134 : 2224333444111123, 3342144311242231
135 : 2224333444111123, 3342144312243211
136 : 2224333444111123, 3342114321443212
137 : 2224333444111123, 3342441122243311
138 : 2223311444411332, 3334444311222121
139 : 2223311444411332, 3334444111223221
140 : 2223311444411332, 4332434121142123
141 : 2223311444411332, 4332434321142121
142 : 2223311444411332, 4332434121143221
143 : 2223311444411332, 4332144321143221
144 : 2223311444411332, 4332434122142113
145 : 2223311444411332, 4332434322142111
146 : 2223311444411332, 4332434122143211
147 : 2223311444411332, 4332144122143213
148 : 2223311444411332, 4332444311223211
149 : 2223311444411332, 3344144311223221

150 : 2223311444411332, 3344434311222121
151 : 2223331444411132, 3334144122143221
152 : 2223331444411132, 4332144321142213
153 : 2223331444411132, 4332144121243321
154 : 2223331444411132, 4332144321143221
155 : 2223331444411132, 4332144321243211
156 : 2223331444411132, 4332144121242313
157 : 2223331444411132, 4332443121242311
158 : 2223331444411132, 4332443121142213
159 : 2223331444411132, 4332444121123321
160 : 2223331444411132, 3344443311222211
161 : 2223331444411132, 3344414311222321
162 : 2223331444411132, 3344143122142321
163 : 2223113444421133, 4332434112142321
164 : 2223113444421133, 4332441112143322
165 : 2223113444421133, 4332444112113322
166 : 2223113444421133, 3344434321112212
167 : 2223113444421133, 3344441321112322
168 : 2223113444421133, 3344341311242212
169 : 2224113344421133, 4333434421212211
170 : 2224113344421133, 4333434121242211
171 : 2224113344421133, 3342441111243322
172 : 2224113344421133, 3342434111243212
173 : 2223314444123311, 4332141321441232
174 : 2223314444123311, 4332441321411232
175 : 2223314444123311, 4332441311241223
176 : 2223314444123311, 4332431121441232
177 : 2223314444123311, 3344431112241232
178 : 2223314444123311, 3344143311241222
179 : 2223314444123311, 3344141311242232
180 : 2223314444123311, 3344141312242132
181 : 2223314444111332, 3334441312242211
182 : 2223314444111332, 4332431122442113
183 : 2223314444111332, 3344443311222211
184 : 2223314444111332, 3344441311223221
185 : 2223314444111332, 3344441312223211
186 : 2223314444111332, 3344433111422221
187 : 2223314444111332, 3344433121422211
188 : 2223314444111332, 3344141321423221
189 : 2223334444113211, 3344141321422132
190 : 2233441111442233, 3344334422111122
191 : 2233314444211123, 3344141312423212
192 : 2233314444211123, 3442133112442231
193 : 2233314444211123, 3442433112142231
194 : 2234334144121123, 3342413411242312
195 : 2234114344123321, 3342443111242213
196 : 3344334411221122, 2423141324142313
197 : 3344114411222233, 2423341324141312
198 : 3344134411221322, 2423411324143213
199 : 3344113412241223, 2423341141423132
200 : 3344113412241223, 2423341341422131
201 : 2344133412241123, 4223314141422331

Lemma 8.2.8. *$A144, \ldots, A20144$ are not isotopic to each other.*

Proof. We compute the column characteristic value and the row characteristic set of $Ax44$ and represent them in the format: "x: the column characteristic value of $Ax44$; the row characteristic set of $Ax44$", where the column characteristic value is in the form $c_4c_3c_2$, the row characteristic set is in the form $T_1(i,j)\ T'(i,j)$, in order of $ij = 12, 13, 14, 23, 24, 34$, $T'(i,j) = T_2(i,j)$ if the derived permutation from row i to row j exists, $T'(i,j) = T_3(i,j)$ otherwise. $T_2(i,j) = 4$ means "a cycle of length 4"; $T_2(i,j) = 2$ means "two transpositions". We list the results of column characteristic values and row characteristic sets of $A144$, ..., $A19544$ as follows:

$$
\begin{array}{r}
1 : 400; 4002, 4002, 4002, 4002, 4002, 4002 \\
2 : 400; 4002, 4004, 4004, 4004, 4004, 4002 \\
3 : 220; 4002, 2202, 2202, 2202, 2202, 4002 \\
4 : 220; 4002, 2204, 2204, 2204, 2204, 4002 \\
5 : 204; 4002, 2020, 2020, 2020, 2020, 4002 \\
6 : 040; 4002, 0402, 0402, 0402, 0402, 4002 \\
7 : 040; 4002, 0404, 0404, 0404, 0404, 4002 \\
8 : 024; 4002, 0220, 0220, 0220, 0220, 4002 \\
9 : 008; 4002, 0040, 0040, 0040, 0040, 4002 \\
10 : 040; 4004, 0402, 0404, 0404, 0402, 0404 \\
11 : 008; 4004, 0040, 0040, 0040, 0040, 0040 \\
12 : 121; 2202, 1202, 2202, 2202, 1202, 2202 \\
13 : 130; 2204, 1304, 2202, 2202, 1304, 2204 \\
14 : 113; 2020, 1110, 2202, 2202, 1110, 2020 \\
15 : 106; 2020, 1030, 2020, 2020, 1030, 2020 \\
16 : 040; 2202, 0402, 0402, 0402, 0402, 2202 \\
17 : 022; 2020, 0202, 0402, 0402, 0202, 0402 \\
18 : 022; 2204, 0204, 0402, 0402, 0204, 0220 \\
19 : 022; 2202, 0202, 0402, 0402, 0202, 2202 \\
20 : 022; 2020, 0204, 0402, 0402, 0204, 2020 \\
21 : 040; 2204, 0404, 0402, 0402, 0404, 2204 \\
22 : 031; 2202, 0404, 0304, 0304, 0404, 2202 \\
23 : 022; 2020, 0304, 0304, 0304, 0304, 0402 \\
24 : 014; 2204, 0120, 0210, 0210, 0120, 0220 \\
25 : 013; 2020, 0110, 0210, 0210, 0110, 0402 \\
26 : 023; 2202, 0220, 0210, 0210, 0220, 2202 \\
27 : 015; 2020, 0120, 0220, 0220, 0120, 2020 \\
28 : 014; 2202, 0110, 0220, 0220, 0110, 2202 \\
29 : 006; 2202, 0020, 0040, 0040, 0020, 2202 \\
30 : 008; 2020, 0040, 0040, 0040, 0040, 2020 \\
31 : 004; 2020, 0020, 0020, 0020, 0020, 0402 \\
32 : 006; 2204, 0040, 0020, 0020, 0040, 0220 \\
33 : 121; 1304, 1202, 1304, 1304, 1202, 1304 \\
34 : 103; 1202, 1202, 1202, 1202, 1202, 1202 \\
35 : 103; 1202, 1110, 1110, 1110, 1110, 1202 \\
36 : 104; 1110, 1110, 1030, 1030, 1110, 1110 \\
37 : 106; 1030, 1030, 1030, 1030, 1030, 1030 \\
38 : 022; 1304, 0210, 0211, 0404, 0202, 0304 \\
39 : 031; 1304, 0402, 0304, 1304, 1202, 0304
\end{array}
$$

40 : 040; 1304, 0402, 1304, 1304, 0402, 1304
41 : 013; 1304, 0110, 0120, 0120, 0110, 0110
42 : 013; 1304, 0210, 0204, 0204, 0210, 0110
43 : 030; 1202, 0304, 0304, 0304, 0304, 1202
44 : 012; 1202, 0110, 0110, 0110, 0110, 1202
45 : 012; 1202, 0202, 0202, 0202, 0202, 1202
46 : 030; 1202, 1202, 1202, 1202, 1202, 1202
47 : 022; 1202, 0404, 0211, 0211, 0404, 1202
48 : 013; 1202, 0220, 0110, 0110, 0220, 1202
49 : 022; 1202, 0402, 0210, 0210, 0402, 1202
50 : 013; 1202, 0110, 0110, 0304, 0304, 0402
51 : 013; 1202, 0202, 0202, 0210, 0210, 0402
52 : 031; 1202, 1202, 1202, 0402, 0402, 0402
53 : 014; 1110, 0110, 0120, 0120, 0110, 0110
54 : 013; 1110, 0202, 0204, 0204, 0202, 1110
55 : 023; 1110, 1202, 1110, 1110, 1202, 1110
56 : 012; 1110, 0202, 0204, 0110, 0210, 0210
57 : 013; 1110, 0110, 0120, 0211, 0304, 0210
58 : 012; 1110, 0210, 0202, 0202, 0210, 0202
59 : 011; 1110, 0110, 0110, 0110, 0110, 0202
60 : 021; 1110, 0211, 0304, 0304, 0211, 0202
61 : 022; 1110, 0402, 0304, 0304, 0402, 1110
62 : 014; 1110, 0220, 0211, 0211, 0220, 1110
63 : 022; 1110, 0402, 0210, 1110, 1202, 0210
64 : 012; 1030, 0204, 0204, 0110, 0110, 0110
65 : 014; 1030, 0120, 0120, 0211, 0211, 0110
66 : 013; 1030, 0202, 0304, 0304, 0202, 0204
67 : 013; 1030, 0110, 0211, 0211, 0110, 0204
68 : 014; 1030, 0210, 0210, 1110, 1110, 0110
69 : 005; 1304, 0120, 0020, 0020, 0120, 0110
70 : 014; 1304, 0211, 0210, 0210, 0211, 0110
71 : 005; 1304, 0110, 0120, 0010, 0040, 0020
72 : 005; 1202, 0110, 0110, 0110, 0110, 1202
73 : 005; 1202, 0020, 0020, 0020, 0020, 1202
74 : 023; 1202, 0210, 0210, 0210, 0210, 1202
75 : 004; 1202, 0010, 0040, 0040, 0010, 1202
76 : 004; 1202, 0020, 0020, 0110, 0110, 0402
77 : 006; 1110, 0040, 0120, 0120, 0040, 1110
78 : 004; 1110, 0110, 0120, 0010, 0020, 0210
79 : 005; 1110, 0020, 0040, 0110, 0110, 0210
80 : 004; 1110, 0110, 0110, 0110, 0110, 0202
81 : 003; 1110, 0010, 0020, 0020, 0010, 0202
82 : 013; 1110, 0110, 0210, 0210, 0110, 0202
83 : 008; 1030, 0040, 1030, 1030, 0040, 1030
84 : 005; 1030, 0110, 0211, 0120, 0020, 0110
85 : 004; 1030, 0110, 0110, 0110, 0110, 0002
86 : 002; 1030, 0010, 0010, 0010, 0010, 0002
87 : 008; 1030, 0040, 0040, 1030, 1030, 0040
88 : 006; 1030, 0120, 0120, 0120, 0120, 0040
89 : 040; 0402, 0402, 0402, 0402, 0402, 0402
90 : 022; 0402, 0220, 0202, 0202, 0220, 0402
91 : 022; 0402, 0404, 0204, 0204, 0404, 0402
92 : 004; 0402, 0110, 0110, 0110, 0110, 0402

93 : 004; 0402, 0020, 0202, 0202, 0020, 0402
94 : 004; 0402, 0004, 0404, 0040, 0004, 0402
95 : 004; 0402, 0004, 0220, 0220, 0004, 0402
96 : 004; 0402, 0002, 0402, 0402, 0002, 0402
97 : 004; 0402, 0002, 0220, 0220, 0002, 0402
98 : 004; 0402, 0040, 0002, 0002, 0040, 0402
99 : 013; 0404, 0110, 0211, 0211, 0110, 0120
100 : 013; 0404, 0202, 0120, 0120, 0202, 0404
101 : 004; 0404, 0404, 0002, 0040, 0004, 0004
102 : 004; 0404, 0120, 0010, 0120, 0010, 0010
103 : 004; 0404, 0204, 0020, 0020, 0204, 0004
104 : 004; 0404, 0210, 0010, 0010, 0210, 0040
105 : 004; 0404, 0110, 0110, 0110, 0110, 0040
106 : 004; 0404, 0040, 0002, 0002, 0040, 0404
107 : 004; 0404, 0020, 0020, 0020, 0020, 0040
108 : 012; 0304, 0202, 0204, 0110, 0210, 0110
109 : 012; 0304, 0204, 0120, 0211, 0110, 0110
110 : 012; 0304, 0110, 0120, 0120, 0110, 0304
111 : 013; 0304, 0220, 0204, 0204, 0220, 0304
112 : 004; 0304, 0120, 0110, 0020, 0010, 0110
113 : 004; 0304, 0211, 0202, 0210, 0004, 0110
114 : 003; 0304, 0010, 0110, 0040, 0004, 0110
115 : 003; 0304, 0010, 0304, 0020, 0010, 0020
116 : 003; 0304, 0004, 0210, 0210, 0004, 0020
117 : 004; 0304, 0010, 0110, 0110, 0211, 0020
118 : 004; 0304, 0110, 0210, 0210, 0110, 0020
119 : 004; 0304, 0010, 0110, 0110, 0010, 0020
120 : 003; 0304, 0010, 0110, 0002, 0120, 0110
121 : 004; 0304, 0002, 0120, 0120, 0002, 0304
122 : 003; 0304, 0020, 0110, 0110, 0020, 0020
123 : 013; 0211, 0120, 0120, 0211, 0211, 0120
124 : 020; 0211, 0204, 0211, 0211, 0204, 0211
125 : 004; 0211, 0204, 0211, 0010, 0040, 0010
126 : 005; 0211, 0120, 0120, 0120, 0120, 0010
127 : 004; 0211, 0211, 0010, 0004, 0120, 0120
128 : 005; 0211, 0211, 0010, 0211, 0010, 0010
129 : 003; 0211, 0110, 0204, 0110, 0020, 0004
130 : 002; 0211, 0110, 0004, 0004, 0110, 0010
131 : 003; 0211, 0211, 0010, 0010, 0010, 0010
132 : 003; 0211, 0110, 0004, 0211, 0002, 0004
133 : 003; 0211, 0110, 0110, 0110, 0110, 0010
134 : 004; 0211, 0002, 0211, 0211, 0002, 0211
135 : 002; 0211, 0004, 0120, 0010, 0010, 0004
136 : 003; 0211, 0004, 0120, 0004, 0120, 0010
137 : 004; 0211, 0120, 0110, 0110, 0120, 0010
138 : 012; 0202, 0220, 0110, 0110, 0220, 0202
139 : 013; 0202, 0210, 0202, 0210, 0202, 0210
140 : 002; 0202, 0002, 0202, 0202, 0002, 0202
141 : 002; 0202, 0020, 0204, 0002, 0020, 0202
142 : 003; 0202, 0002, 0202, 0210, 0110, 0210
143 : 004; 0202, 0002, 0202, 0202, 0040, 0202
144 : 003; 0202, 0004, 0110, 0110, 0004, 0202
145 : 003; 0202, 0110, 0120, 0004, 0110, 0202

146 : 002; 0202, 0004, 0110, 0020, 0010, 0210
147 : 003; 0202, 0010, 0220, 0220, 0010, 0202
148 : 003; 0202, 0110, 0110, 0002, 0020, 0210
149 : 004; 0202, 0020, 0002, 0002, 0020, 0202
150 : 002; 0202, 0040, 0004, 0004, 0040, 0202
151 : 011; 0204, 0110, 0110, 0110, 0110, 0204
152 : 003; 0204, 0002, 0204, 0110, 0110, 0110
153 : 003; 0204, 0010, 0110, 0020, 0120, 0110
154 : 002; 0204, 0002, 0204, 0002, 0040, 0002
155 : 002; 0204, 0004, 0120, 0004, 0120, 0002
156 : 002; 0204, 0010, 0110, 0110, 0010, 0002
157 : 003; 0204, 0004, 0120, 0010, 0010, 0110
158 : 004; 0204, 0002, 0204, 0204, 0002, 0204
159 : 004; 0204, 0110, 0110, 0110, 0110, 0040
160 : 004; 0204, 0040, 0204, 0204, 0040, 0204
161 : 002; 0204, 0020, 0004, 0004, 0020, 0002
162 : 002; 0204, 0010, 0010, 0010, 0010, 0204
163 : 002; 0210, 0002, 0110, 0110, 0002, 0210
164 : 005; 0210, 0020, 0220, 0220, 0020, 0210
165 : 005; 0210, 0210, 0210, 0210, 0210, 0210
166 : 003; 0210, 0220, 0004, 0004, 0220, 0210
167 : 004; 0210, 0210, 0002, 0002, 0210, 0210
168 : 003; 0210, 0110, 0010, 0010, 0110, 0210
169 : 005; 0220, 0220, 0010, 0010, 0220, 0220
170 : 004; 0220, 0110, 0110, 0110, 0110, 0220
171 : 006; 0220, 0020, 0040, 0040, 0020, 0220
172 : 002; 0220, 0002, 0020, 0020, 0002, 0220
173 : 001; 0110, 0004, 0110, 0020, 0010, 0002
174 : 001; 0110, 0002, 0110, 0002, 0110, 0002
175 : 002; 0110, 0002, 0110, 0110, 0002, 0110
176 : 002; 0110, 0004, 0110, 0110, 0004, 0110
177 : 002; 0110, 0010, 0004, 0120, 0010, 0110
178 : 004; 0110, 0110, 0004, 0120, 0110, 0110
179 : 003; 0110, 0110, 0002, 0110, 0002, 0002
180 : 001; 0110, 0010, 0004, 0010, 0004, 0002
181 : 010; 0110, 0110, 0110, 0110, 0110, 0110
182 : 005; 0110, 0010, 0110, 0110, 0010, 0110
183 : 004; 0110, 0040, 0110, 0110, 0040, 0110
184 : 004; 0110, 0020, 0020, 0110, 0110, 0020
185 : 002; 0110, 0110, 0010, 0004, 0020, 0020
186 : 003; 0110, 0110, 0110, 0110, 0110, 0110
187 : 001; 0110, 0020, 0004, 0004, 0020, 0110
188 : 002; 0110, 0010, 0002, 0002, 0010, 0110
189 : 001; 0120, 0010, 0004, 0004, 0010, 0004
190 : 008; 0040, 0040, 0040, 0040, 0040, 0040
191 : 002; 0010, 0010, 0010, 0004, 0004, 0004
192 : 001; 0010, 0010, 0010, 0010, 0010, 0010
193 : 001; 0010, 0004, 0004, 0004, 0004, 0010
194 : 002; 0004, 0002, 0004, 0004, 0002, 0004
195 : 004; 0002, 0002, 0002, 0002, 0002, 0002

We can verify that for any two distinct $Ax44$'s, either their column characteristic values are different, or their row characteristic sets are different. From

Corollary 8.2.1, it immediately follows that $A144, \ldots, A19544$ are not isotopic to each other. Since $A19644$, ..., $A20144$ are complements of $A142, \ldots$, $A642$, respectively, from Lemma 8.2.3 (a) and Lemma 8.2.4, $A19644$, ..., $A20144$ are not isotopic to each other. For any i, $1 \leqslant i \leqslant 195$, and any j, $196 \leqslant j \leqslant 201$, since $Ai44$ has repeated columns and columns of $Aj44$ are different, $Ai44$ and $Aj44$ are not isotopic. Therefore, $A144, \ldots, A20144$ are not isotopic to each other. □

Lemma 8.2.9. *Any* $(4,4)$*-Latin array is isotopic to one of* $A144, \ldots, A20144$*; and any* $(4,4,1)$*-Latin array is isotopic to one of* $A19644, \ldots, A20144$.

Proof. The proof of this lemma is similar to Lemma 8.2.5 but more tedious. We omit the details of the proof for the sake of space. □

Theorem 8.2.14. $I(4,4) = 201,\ I(4,4,1) = 6.$

Proof. This is immediate from Lemmas 8.2.8 and 8.2.9. $I(4,4,1)$ can also be obtained from Theorem 8.2.2 (a) and Theorem 8.2.10, that is, $I(4,4,1) = I(4,2,1) = 6$. □

Theorem 8.2.15. $U(4,4,1) = 5335311421440000$,
$U(4,4) = 80306439693480000$.

Proof. For any $Ax44$, G'''_{Ax44} is easy to determine from positions of repeated columns. For computing the order of G_{Ax44}, we find out the set of coset representatives of G'''_{Ax44} in G''_{Ax44}, the set of coset representatives of G''_{Ax44} in G'_{Ax44} and the set of coset representatives of G'_{Ax44} in G_{Ax44}. Below we give three examples for G_{A144}, G_{A244} and G_{A4844}.

$$\begin{bmatrix} 1111\ 2222\ 3333\ 4444 \\ 2222\ 1111\ 4444\ 3333 \\ 3333\ 4444\ 1111\ 2222 \\ 4444\ 3333\ 2222\ 1111 \end{bmatrix} \quad \begin{bmatrix} 1111\ 2222\ 3333\ 4444 \\ 2222\ 1111\ 4444\ 3333 \\ 3333\ 4444\ 2222\ 1111 \\ 4444\ 3333\ 1111\ 2222 \end{bmatrix} \quad \begin{bmatrix} 1111\ 2222\ 3333\ 4444 \\ 2222\ 1134\ 4441\ 3331 \\ 3334\ 4443\ 2112\ 1122 \\ 4443\ 3311\ 1224\ 2213 \end{bmatrix}$$
$$A144 \qquad A244 \qquad A4844$$

From the form of $A144$, each block consists of four identical columns. Thus G'''_{A144} consists of all column permutations within blocks; therefore, its order is $(4!)^4$.

To find G''_{A144}, let $\langle (1), \beta, \gamma \rangle \in G''$ and $\beta(i_j) = j$, where i_1, i_2, i_3, i_4 are a permutation of 1,2,3,4. Whenever $\langle (1), \beta, \gamma \rangle$ keeps $A144$ unchanged, from the form of $A144$, the renaming β transforms the i_1-th block of $A144$ into the first block of $A144$. Thus β transforms the first column of the i_1-th block of $A144$ into the first column of first block of $A144$. Therefore, the renaming β is uniquely determined by i_1. In the case of $i_1 = 2$, the renaming β should transform the fifth column 2143 into the first column 1234. Thus

β should be the permutation (12)(34). It is easy to verify that β transforms the first, the third and the fourth blocks of $A144$ into the second, the fourth and the third blocks of $A144$, respectively. Thus $\langle (1), (12)(34), \cdot \rangle$ keeps $A144$ unchanged. Recall that a dot $\cdot$ in the place of the column arranging means "some column arranging γ". Similarly, in the case of $i_1 = 3$, it is easy to verify that $\langle (1), (13)(24), \cdot \rangle$ keeps $A144$ unchanged; in the case of $i_1 = 4$, it is easy to verify that $\langle (1), (14)(23), \cdot \rangle$ keeps $A144$ unchanged. Using $(12)(34) \cdot (13)(24) = (14)(23)$, from Theorem 8.2.6 (b), we have $G''_{A144} = (\langle (1), (12)(34), \cdot \rangle, \langle (1), (13)(24), \cdot \rangle)\ G'''_{A144}$, of which the order is $4(4!)^4$.

To find G'_{A144}, let $\langle \alpha, \beta, \gamma \rangle \in G'$, where the row arranging α is a permutation of 2,3,4. Try $\alpha = (432)$. The row arranging (432) transforms $A144$ into

$$A' = \begin{bmatrix} 1111\ 2222\ 3333\ 4444 \\ 3333\ 4444\ 1111\ 2222 \\ 4444\ 3333\ 2222\ 1111 \\ 2222\ 1111\ 4444\ 3333 \end{bmatrix}.$$

If A' can be transformed into $A144$ by a renaming β and some column arranging, then the first block of A' is transformed into the $\beta(1)$-th block of $A144$. From the form of $A144$ and A', the first column of A' is transformed into the first column of the $\beta(1)$-th block of $A144$. Try $\beta(1) = 1$. Then β transforms the column 1342 into the column 1234; that is, $\beta = (432)$. It is easy to verify that $\langle (1), (432), \cdot \rangle$ transforms A' into $A144$ indeed. Thus $\langle (432), (432), \cdot \rangle$ keeps $A144$ unchanged. Similarly, we can choose the row arranging (34) and the renaming (34) so that $\langle (34), (34), \cdot \rangle$ keeps $A144$ unchanged. Noticing that permutations (432) and (34) can generate all permutations on $\{2, 3, 4\}$, from Theorem 8.2.6 (c), we have $G'_{A144} = (\langle (432), (432), \cdot \rangle, \langle (34), (34), \cdot \rangle)\ G''_{A144}$, of which the order is $6 \cdot 4(4!)^4$.

To find G_{A144}, let $\langle \alpha, \beta, \gamma \rangle \in G$. Try $\alpha = (4321)$. The row arranging (4321) transforms $A144$ into

$$A'' = \begin{bmatrix} 2222\ 1111\ 4444\ 3333 \\ 3333\ 4444\ 1111\ 2222 \\ 4444\ 3333\ 2222\ 1111 \\ 1111\ 2222\ 3333\ 4444 \end{bmatrix}.$$

If A'' can be transformed into $A144$ by a renaming β and some column arranging, then the first block of A'' is transformed into the $\beta(2)$-th block of $A144$. From the form of $A144$ and A'', the first column of A'' is transformed into the first column of the $\beta(2)$-th block of $A144$. Try $\beta(2) = 1$. Then β transforms the column 2341 into the column 1234; that is, $\beta = (4321)$. It is easy to verify that $\langle (1), (4321), \cdot \rangle$ transforms A'' into $A144$ indeed. Thus $\langle (4321), (4321), \cdot \rangle$ keeps $A144$ unchanged. Noticing that the permuta-

tion (4321) generates (1234), (13)(24) and (1), from Theorem 8.2.6 (d), we have $G_{A144} = (\langle (4321), (4321), \cdot \rangle)\ G'_{A144}$, of which the order is $4 \cdot 6 \cdot 4(4!)^4$.

We compute G_{A244}. In GR_{A244}, labels of edges $(1, 2)$ and $(3, 4)$ are the same, say "red"; labels of other edges are the same and not red, say "green".

G'''_{A244} coincides with G'''_{A144}; its order is $(4!)^4$.

To find G''_{A244}, let $\langle (1), \beta, \gamma \rangle \in G''_{A244}$ and $\beta(i_j) = j$, where (i_1, i_2, i_3, i_4) is a permutation of 1,2,3,4. Similar to the discussion on $A144$, since $\langle (1), \beta, \gamma \rangle$ keeps $A244$ unchanged, the renaming β is uniquely determined by i_1. It follows that there are at most four choices for i_1. Try $i_1 = 3$. Similar to the discussion for $A144$, β transforms the column 3421 into the column 1234; that is, $\beta = (1423)$. It is easy to verify that $\langle (1), (1423), \cdot \rangle$ keeps $A244$ unchanged indeed. Since the permutation (1423) generates four permutations corresponding to four choices for i_1, from Theorem 8.2.6 (b), we have $G''_{A244} = (\langle (1), (1423), \cdot \rangle)\ G'''_{A244}$, of which the order is $4 \cdot (4!)^4$.

To find G'_{A244}, let $\langle \alpha, \beta, \gamma \rangle \in G'$, where the row arranging α is a permutation of 2,3,4. From the proof of Theorem 8.2.3 (b), if $\langle \alpha, \beta, \gamma \rangle$ is an autotopism of $A244$, then the row arranging α is an automorphism of GR_{A244}. In this case, edges $(1, 2)$ and $(\alpha(1), \alpha(2))$, that is, $(1, \alpha(2))$, have the same color. Since the edge $(1, 2)$ is the unique red edge with endpoint 1, we have $\alpha(2) = 2$ whenever $\langle \alpha, \beta, \gamma \rangle \in G'_{A244}$. Thus the choices of α are (34) and (1) in this case. The row arranging (34) transforms $A244$ into

$$A' = \begin{bmatrix} 1111\ 2222\ 3333\ 4444 \\ 2222\ 1111\ 4444\ 3333 \\ 4444\ 3333\ 1111\ 2222 \\ 3333\ 4444\ 2222\ 1111 \end{bmatrix}.$$

It is easy to verify that A' can be transformed into $A244$ by renaming (34) and some column arranging. Thus $\langle (34), (34), \cdot \rangle$ keeps $A244$ unchanged. Therefore, from Theorem 8.2.6 (c), we have $G'_{A244} = (\langle (34), (34), \cdot \rangle)\ G''_{A244}$, of which the order is $2 \cdot 4 \cdot (4!)^4$.

To find G_{A244}, let $\langle \alpha, \beta, \gamma \rangle \in G_{A244}$. Thus α is an automorphism of PR_{A244}. Try $\alpha(4) = 1$. Since the edge $(3, 4)$ is red, the edge $(\alpha(3), \alpha(4))$, that is, $(\alpha(3), 1)$, is red. Since the edge $(2, 1)$ is the unique red edge with endpoint 1, we have $\alpha(3) = 2$. It follows that $\alpha = (1324)$ or $(14)(23)$. Try $\alpha = (1324)$. $A244$ can be transformed into

$$A'' = \begin{bmatrix} 4444\ 3333\ 1111\ 2222 \\ 3333\ 4444\ 2222\ 1111 \\ 1111\ 2222\ 3333\ 4444 \\ 2222\ 1111\ 4444\ 3333 \end{bmatrix}$$

by row arranging (1324). It is evident that A'' can be transformed into $A244$ by some column arranging. Therefore, $\langle (1324), (1), \cdot \rangle$ keeps $A244$ unchanged.

Noticing that the permutation (1324) generates (1423), (12)(34) and (1), from Theorem 8.2.6 (d), we then have $G_{A244} = (\langle(1324), (1), \cdot\rangle)\ G'_{A244}$, of which the order is $4 \cdot 2 \cdot 4 \cdot (4!)^4$.

We compute G_{A4844}. In GR_{A4844}, labels of edges $(1, 4)$ and $(2, 3)$ are the same, say "red"; labels of edges $(1, 2)$ and $(3, 4)$ are the same and not red, say "green"; labels of edges $(1, 3)$ and $(2, 4)$ are the same, not red and not green, say "blue".

G'''_{A4844} consists of the product of the following permutations: permutations of columns 1,2 and 3, permutations of columns 5 and 6, permutations of columns 10 and 11, permutations of columns 13 and 14; its order is $3!(2!)^3$.

To find G''_{A4844}, let $\langle(1), \beta, \gamma\rangle \in G''_{A4844}$. Since the column type of the first block of $A4844$ are different from the column types of other blocks, from Theorem 8.2.7 and its proof, the renaming β transforms the column 1234 into itself; that is, $\beta = (1)$. From Theorem 8.2.6 (b), it follows that $G''_{A1244} = G'''_{A1244}$, of which the order is $3!(2!)^3$.

To find G'_{A4844}, let $\langle\alpha, \beta, \gamma\rangle \in G'_{A4844}$, where the row arranging α is a permutation of 2, 3, 4. Thus α is an automorphism of GR_{A4844}. It follows that edges $(1, i)$ and $(\alpha(1), \alpha(i))$, that is, $(1, \alpha(i))$, have the same color for $i = 2, 3, 4$. Since the colors of edges with endpoint 1 are different, we have $\alpha(i) = i$ for $i = 2, 3, 4$; that is, $\alpha = (1)$. From Theorem 8.2.6 (c), we have $G'_{A4844} = G''_{A4844}$, of which the order is $3!(2!)^3$.

To find G_{A4844}, let $\langle\alpha, \beta, \gamma\rangle \in G_{A4844}$. Thus α is an automorphism of PR_{A4844}. Try $\alpha(4) = 1$. Since the edge $(1, 4)$ is red, the edge $(\alpha(1), \alpha(4))$, that is, $(\alpha(1), 1)$, is red. Since the edge $(4, 1)$ is the unique red edge with endpoint 1, we have $\alpha(1) = 4$. It follows that $\alpha = (14)$ or $(14)(23)$. $A4844$ can be transformed into

$$A' = \begin{bmatrix} 1111\ 2222\ 3333\ 4444 \\ 4322\ 1111\ 4442\ 3332 \\ 3443\ 4433\ 2111\ 2221 \\ 2234\ 3344\ 1224\ 1113 \end{bmatrix}$$

by row arranging $(14)(23)$ and some column arranging. Since no pair of the row type and the column type of a block of A' coincides with one of the first block of $A4844$, from Theorem 8.2.7, no isotopism $\langle(1), \beta', \gamma'\rangle$ from A' to $A4844$ exists. It follows that no autotopism of $A4844$ with row arranging $(14)(23)$ exists. For the result of transforming A' by row arranging (23), no pair of the row type and the column type of a block of its coincides with one of the first block of $A4844$. Thus no isotopism $\langle(1), \beta', \gamma'\rangle$ from it to $A4844$ exists. It follows that no autotopism of $A4844$ with row arranging (14) exists. We next try $\alpha(2) = 1$. Since the edge $(1, 2)$ is green, the edge $(\alpha(1), \alpha(2))$, that is, $(\alpha(1), 1)$, is green. Since the edge $(2, 1)$ is the unique green edge with endpoint 1, we have $\alpha(1) = 2$. It follows that $\alpha = (12)$ or $(12)(34)$. $A4844$

can be transformed into

$$A'' = \begin{bmatrix} 1111\ 2222\ 3333\ 4444 \\ 2234\ 1111\ 2444\ 2333 \\ 3343\ 4443\ 1221\ 1122 \\ 4422\ 3334\ 4112\ 3211 \end{bmatrix}$$

by row arranging (12)(34) and some column arranging. From Theorem 8.2.7 and its proof, since the column with multiplicity 3 is unique in A'' and in $A4844$, the renaming β should transform the column 2143 of A'' into the column 1234 of $A4844$. It follows that $\beta = (12)(34)$. It is easy to verify that A'' can be transformed into $A4844$ by renaming β and some column arranging indeed. Thus $\langle (12)(34), (12)(34), \cdot \rangle$ keeps $A4844$ unchanged. Finally, for any row arranging α with $\alpha(3) = 1$, the product $\alpha \cdot (12)(34)$ brings row 4 to row 1. Thus such $\langle \alpha, \beta, \cdot \rangle$ is not an autotopism of $A4844$; otherwise there exists an autotopism of $A4844$ in which the row arranging bring row 4 to row 1, this is impossible as shown previously. From Theorem 8.2.6 (d), we obtain $G_{A4844} = (\langle (12)(34), (12)(34), \cdot \rangle)\ G'_{A4844}$, of which the order is $2 \cdot 3(2!)^3$.

Similarly, we can compute other G_{Ax44}, using GR_{Ax44} to reduce the trying scope for row arranging, and using column types and row types of blocks to reduce the trying scope for renaming.

Denote the order of autotopism group G_{Ai44} of $Ai44$ by $n_i, i = 1, \ldots, 201$. On the autotopism group of $Ax44$ and its order n_x, the computing results are represented by the format: "x: the order of G'''_{Ax44}, the number of cosets of G'''_{Ax44} in G''_{Ax44}, the number of cosets of G''_{Ax44} in G'_{Ax44}, the number of cosets of G'_{Ax44} in G_{Ax44} (the product of the four numbers is n_x); the set of coset representatives of G'''_{Ax44} in G''_{Ax44}; the set of coset representatives of G''_{Ax44} in G'_{Ax44}; the set of coset representatives of G'_{Ax44} in G_{Ax44}." γ in a coset representative $\langle \alpha, \beta, \gamma \rangle$ is omitted. For the sake of space, we only list a part of results as follows:

1 : $(4!)^4, 4, 6, 4;\ (\langle (1), (12)(34) \rangle, \langle (1), (13)(24) \rangle);$
$(\langle (432), (432) \rangle, \langle (34), (34) \rangle);\ (\langle (4321), (4321) \rangle).$

2 : $(4!)^4, 4, 2, 4;\ (\langle (1), (1423) \rangle);\ (\langle (34), (34) \rangle);\ (\langle (1324), (1) \rangle).$

3 : $(4!3!)^2, 2, 2, 4;\ (\langle (1), (12)(34) \rangle);\ (\langle (34), (34) \rangle);\ (\langle (1423), (1423) \rangle).$

4 : $(4!3!)^2, 2, 2, 4;\ (\langle (1), (12)(34) \rangle);\ (\langle (34), (34) \rangle);\ (\langle (1324), (1324) \rangle).$

5 : $(4!)^2 2^4, 2, 2, 4;\ (\langle (1), (12)(34) \rangle);\ (\langle (34), (34) \rangle);\ (\langle (1324), (1324) \rangle).$

6 : $(3!)^4, 4, 2, 4;\ (\langle (1), (12)(34) \rangle, \langle (1), (13)(24) \rangle);\ (\langle (34), (34) \rangle);$
$(\langle (1423), (1423) \rangle).$

7 : $(3!)^4, 4, 2, 4;\ (\langle (1), (1423) \rangle);\ (\langle (34), (34) \rangle);\ (\langle (1423), (1423) \rangle).$

. . . . . .

195 : $2^4, 4, 6, 4;\ (\langle(1), (12)(34)\rangle, \langle(1), (13)(24)\rangle);$
$(\langle(234), (234)\rangle, \langle(34), (34)\rangle);\ (\langle(4321), (4321)\rangle).$

196 : $1, 8, 2, 4;\ (\langle(1), (34)\rangle, \langle(1), (1423)\rangle);\ (\langle(34), (1)\rangle);\ (\langle(1423), (1)\rangle).$

197 : $1, 4, 2, 2;\ (\langle(1), (1234)\rangle);\ (\langle(34), (1)\rangle);\ (\langle(12), (12)(34)\rangle).$

198 : $1, 2, 1, 4;\ (\langle(1), (13)(24)\rangle);\ e;\ (\langle(13)(24), (1)\rangle, \langle(14)(23), (12)(34)\rangle).$

199 : $1, 2, 2, 4;\ (\langle(1), (34)\rangle);\ (\langle(24), (12)\rangle);\ (\langle(4321), (1)\rangle).$

200 : $1, 1, 2, 3;\ e;\ (\langle(34), (34)\rangle);\ (\langle(431), (234)\rangle).$

201 : $1, 4, 6, 4;\ (\langle(1), (12)(34)\rangle, \langle(1), (14)(23)\rangle);$
$(\langle(34), (23)\rangle, \langle(234), (234)\rangle);\ (\langle(4321), (12)\rangle).$

Thus we have

$n_1 = 3^5 2^{17},\ \ n_{89} = 3^5 2^8,\ \ n_2 = 3^4 2^{17},\ \ n_3 = n_4 = 3^4 2^{12},\ \ n_6 = n_7 = 3^4 2^9,$
$n_{13} = 3^4 2^8,\ \ n_i = 3^4 2^7,\ \ i = 10, 16, 21, 40,\ \ n_{46} = 3^4 2^6,\ \ n_{52} = 3^4 2^5,$
$n_{12} = n_{33} = 3^3 2^9,\ \ n_{22} = n_{43} = 3^3 2^6,\ \ n_{39} = 3^3 2^5,$
$n_5 = 3^2 2^{14},\ \ n_{37} = 3^2 2^{12},\ \ n_8 = 3^2 2^{10},\ \ n_{14} = n_{34} = 3^2 2^9,$
$n_{55} = n_{74} = 3^2 2^8,\ \ n_i = 3^2 2^7,\ \ i = 19, 20, 26, 90, 91,$
$n_i = 3^2 2^6,\ \ i = 17, 18, 23, 47, 49, 61,\ \ n_i = 3^2 2^5,\ \ i = 63, 124,$
$n_i = 3^2 2^4,\ \ i = 38, 60, 123, 139,\ \ n_{181} = 3^2 2^3,$
$n_{190} = 3^1 2^{13},\ \ n_{15} = 3^1 2^{12},\ \ n_{36} = 3^1 2^{10},\ \ n_i = 3^1 2^9,\ \ i = 35, 87, 195,$
$n_{27} = 3^1 2^8,\ \ n_i = 3^1 2^7,\ \ i = 28, 53, 165,$
$n_i = 3^1 2^6,\ \ i = 24, 44, 45, 54, 62, 65, 68, 70, 100, 111, 128,$
$n_i = 3^1 2^5,\ \ i = 25, 41, 42, 48, 50, 51, 66, 67, 82, 99, 110, 138, 186, 201,$
$n_i = 3^1 2^4,\ \ i = 57, 58, 64, 151, 179, 184, 192,$
$n_i = 3^1 2^3,\ \ i = 56, 59, 108, 109, 191,\ \ n_{174} = 3^1 2^2,\ \ n_{200} = 3^1 2^1,$
$n_9 = 2^{14},\ \ n_{11} = n_{83} = 2^{12},\ \ n_{30} = 2^{11},\ \ n_i = 2^9,\ \ i = 29, 96, 171,$
$n_i = 2^8,\ \ i = 32, 72, 73, 77, 88, 98, 106, 158, 160, 169, 182,$
$n_i = 2^7,\ \ i = 31, 75, 92, 93, 94, 95, 97, 101, 107, 134, 143, 149, 164, 167, 183,$
$n_i = 2^6,\ \ i = 69, 85, 103, 105, 121, 126, 140, 194, 196,$
$n_i = 2^5,\ \ i = 71, 76, 79, 80, 84, 86, 102, 104, 118, 119, 125, 127, 137, 144, 147,$
$150, 159, 162, 166, 168, 170, 172, 178,$
$n_i = 2^4,\ \ i = 78, 81, 112, 113, 115, 116, 117, 122, 131, 132, 133, 136, 141, 142,$
$145, 152, 154, 161, 163, 175, 188, 193, 197, 199,$
$n_i = 2^3,\ \ i = 114, 120, 129, 130, 148, 153, 155, 156, 157, 176, 177, 185, 187, 198,$
$n_i = 2^2,\ \ i = 135, 146, 180, 189,\ \ n_{173} = 2.$

Since the number of elements in the isotopy class containing $Ai44$ is $4!4!16!/n_i$, noticing $4!4!16! = 3^2 2^6 16! = 3^8 2^{18} 7007000$, we have

$$\begin{aligned} U(4,4,1) &= \sum_{i=196}^{201} 4!4!16!/n_i \\ &= 16!3^2 2^6 (1/2^6 + 1/2^4 + 1/2^3 + 1/2^4 + 1/(3\cdot 2) + 1/(3\cdot 2^5)) \\ &= 16!(3^2 + 3^2 2^2 + 3^2 2^3 + 3^2 2^2 + 3\cdot 2^5 + 3\cdot 2) \\ &= 16!255 = 5335311421440000, \\ U(4,4) &= \sum_{i=1}^{201} 4!4!16!/n_i = 11460887640 \cdot 7007000 = 80306439693480000. \end{aligned}$$

We point out that $U(4,4,1)$ can also be obtained from Theorem 8.2.2 (b) and Theorem 8.2.11, that is, $U(4,4,1) = U(4,2,1)(4! - 4\cdot 2)!/(4\cdot 2)! = 10281600 \cdot 16!/8! = 5335311421440000$. □

Enumerating high order Latin arrays is not easy. Among others, several useful permutational families corresponding to $(2^r, 2^r)$-Latin arrays are $g_{w_1 w_2}(x) = \varphi(w_1 - (w_2 \oplus (w_1 - \varphi(x))))$, $g_{w_1 w_2}(x) = \varphi(w_1 \oplus (w_2 - (w_1 \oplus \varphi(x))))$, $g_{w_1 w_2}(x) = w_1 \oplus \varphi(w_2 - \varphi(w_1 \oplus x))$, $g_{w_1 w_2}(x) = w_1 - \varphi(w_2 \oplus \varphi(w_1 - x))$, where φ is a bijection. In the case where φ is an involution (i.e., $\varphi^{-1} = \varphi$), such g_w's are involutions.

We return to giving an application of $(4,4)$-Latin array to cryptography. It is known that the following binary steam cipher is insecure: a plaintext $x_0 \dots x_{l-1}$ is encrypted into a ciphertext $y_0 \dots y_{l-1}$ by

$$y_i = x_i \oplus w_i,\ i = 0, \dots, l-1,$$

where the key string $w_0 \dots w_{l-1}$ is generated by a binary linear shift register. In fact, this cipher can not resist the plain-chosen attack. Assume that the key string satisfies the equation

$$w_i = a_1 w_{i-1} \oplus \cdots \oplus a_n w_{i-n},\ i = 0, \dots, l-1. \tag{8.1}$$

The key of the cipher is $a_1, \dots, a_n, w_{-n}, \dots, w_{-1}$. If one can obtain a segment of plaintext of length $2n$, say $x_j \dots x_{j+2n-1}$, then $w_j \dots w_{j+2n-1}$ may be evaluated by

$$w_i = x_i \oplus y_i,\ i = j, \dots, j+2n-1;$$

therefore, by solving the equation

$$w_i = a_1 w_{i-1} \oplus \cdots \oplus a_n w_{i-n},\ i = j+n, \dots, j+2n-1,$$

the coefficients $a_1, \dots, a_n$ can be easily found. By (8.1), the key bits $w_{j+2n}, \dots, w_{l-1}$ can be easily computed from $w_{j+n} \dots w_{j+2n-1}$, and the key bits $w_0, \dots,$

w_{j-1} can be easily computed from $w_j \dots w_{j+n-1}$. If we describe this cipher in the form in Fig.8.1.2, then M_a is a binary linear shift register and the permutational family $\{g_w, w = 0, 1\}$ is $\{w \oplus x, w = 0, 1\}$ which corresponds to a (2,2)-Latin array. The permutational family is too small! To improve this situation, one may adopt a more complex shift register or a permutational family corresponding to a higher order Latin array. For the latter, we give an example using (4,4)-Latin array which can resist the plain-chosen attack.

Example 8.2.1. A maximal linear shift register sequence plus (4,4)-Latin array cipher.

In Fig.8.1.2, let X and Y be $GF(2^2) = \{00, 01, 10, 11\}$. Let M_a be an n-order maximal linear shift register sequence generator (i.e., a linear shift register over $GF(2)$ with characteristic polynomial period $2^n - 1$) with 4 bits output. And the permutation g_w on $GF(2^2)$ corresponds to a (4,4)-Latin array A in such a way: for any $w = w_3w_2w_1w_0 \in GF(2^4)$, if the element of A at column $(w_3 2^3 + w_2 2^2 + w_1 2 + w_0) + 1$ and row $(x_1 2 + x_0) + 1$ is $(y_1 2 + y_0) + 1$, then $g_w(x_1x_0) = y_1y_0$. The key consists of g_w, M_a and its initial state; the amount of keys is the product of the number of (4,4)-Latin arrays, the number of the primitive polynomials of degree n over $GF(2)$ and $2^n - 1$. An analysis in [104] shows that this cipher can resist the plain-chosen attack.

If we use Latin arrays with larger order and if the family of permutations corresponding to the Latin array satisfies the following condition: for any x and y, components of w with $g_w(x) = y$ can not satisfy any linear equation, where w is represented as a vector of dimension $\lceil \log_2 nk \rceil$ over $GF(2)$, then for the counterpart of the above cipher in the case of (n, k)-Latin array, the characteristic polynomial of M_a may be excluded from the key, in other words, the structure of M_a can be fixed and the key consists of the Latin array and the initial state of M_a. In the following section, we will give the definition of such a kind Latin arrays, so-called Latin arrays with independence degree $\geqslant 1$, and discuss their generation problem by means of invertible Boolean vector functions with independence degree $\geqslant 1$.

8.3 Linearly Independent Latin Arrays

8.3.1 Latin Arrays of Invertible Functions

Let n and k be two positive integers. Denote $r = \lceil \log_2 nk \rceil$ and $N = \{1, \dots, n\}$. Let A be an (n, k)-Latin array. The vector $[u_1, \dots, u_r]$ over $GF(2)$ is called the *column label* of column $(u_1 2^{r-1} + u_2 2^{r-2} + \cdots + u_r) + 1$ of A. Let x, $y \in N$. If components of column labels of columns of A in which the elements at row x are y satisfy some nonzero polynomial in r variables of degree

$\leqslant c$ over $GF(2)$, A is said to be *c-dependent* with respect to (x, y), otherwise A is said to be *c-independent* with respect to (x, y). If A is c-dependent with respect to (x, y) for any x, $y \in N$, A is said to be *c-dependent*. If A is c-independent with respect to (x, y) for any x, $y \in N$, A is said to be *c-independent*. If A is c-dependent and not $(c-1)$-dependent, c is called the *dependent degree* of A, denoted by c_A. If A is c-independent and not $(c+1)$-independent, c is called the *independent degree* of A, denoted by I_A. Clearly, $c_A \geqslant I_A + 1$.

Proposition 8.3.1. *Let A be an (n, k)-Latin array and $r = \lceil \log_2 nk \rceil$. Then we have*

$$c_A \leqslant \min c\,[\,1 + \tbinom{r}{1} + \cdots + \tbinom{r}{c} > k\,].$$

Proof. Let c be a positive integer with $1 + \binom{r}{1} + \cdots + \binom{r}{c} > k$. For any $x, y \in N$, let the element at row x column j_h of A be y, $h = 1, \ldots, k$. Denote the column label of column j_h by $[w_{h1}, \ldots, w_{hr}]$. Let α_h be the row vector

$$[1,\ w_{h1}, \ldots, w_{hr},\ w_{h1}w_{h2}, \ldots, w_{h,r-1}w_{hr},$$
$$\ldots\ldots, w_{h1}\ldots w_{hc},\ \ldots,\ w_{h,r-c+1}\ldots w_{hr}],$$

for $h = 1, \ldots, k$. Let C be a matrix of which the h-th row is α_h, $h = 1, \ldots, k$. Since the number of C's columns is greater than the number of C's rows, columns of C are linearly dependent. Thus there is a nonzero column vector γ such that $C\gamma = 0$. It follows that A is c-dependent with respect to (x, y). Thus A is c-dependent. Therefore, $c_A \leqslant c$. □

We use R_2^r to denote the row vector space of dimension r over $GF(2)$. For any positive integer m, let f_m be a one-to-one mapping from R_2^m onto $\{0, 1, \ldots, 2^m - 1\}$ defined by $f_m(x_1, \ldots, x_m) = x_1 2^{m-1} + x_2 2^{m-2} + \cdots + x_m$. Let φ_1 and φ_2 be two permutations on R_2^r, and φ a transformation on R_2^r. Denote $\Phi = (\varphi_1, \varphi, \varphi_2)$. Construct a $2^r \times 2^{2r}$ matrix A_Φ over $\{1, \ldots, 2^r\}$ as follows: for any $x, w_1, w_2 \in R_2^r$, the element at row $f_r(x) + 1$ and column $f_{2r}(w_1, w_2) + 1$ is $f_r(\varphi_1(w_1) \oplus \varphi(\varphi_2(w_2) \oplus x)) + 1$, where $\oplus$ stands for the vector addition over $GF(2)$.

Proposition 8.3.2. *A_Φ is a $(2^r, 2^r)$-Latin array if and only if φ is a permutation.*

Proof. Whenever φ is a permutation, it is easy to prove that each column of A_Φ is a permutation on N and that for any $x, y \in N$, any $w_2 \in R_2^r$ there exists uniquely $w_1 \in R_2^r$ such that the element at row x column $f(w_1, w_2) + 1$ of A_Φ is y. Thus A_Φ is a $(2^r, 2^r)$-Latin array.

Whenever φ is not a permutation, it is easy to prove that each column of A_Φ is not a permutation on N. Thus A_Φ is not a $(2^r, 2^r)$-Latin array. □

Whenever φ is a permutation, A_Φ is called the $(2^r, 2^r)$-*Latin array* of Φ.

Let φ be a transformation on R_2^r with component functions $\varphi_1, \ldots, \varphi_r$. For any nonnegative integer c, if there is a nonzero polynomial h of degree $\leqslant c$ in $2r$ variables over $GF(2)$ such that

$$h(x_1, \ldots, x_r, \varphi_1(x_1, \ldots, x_r), \ldots, \varphi_r(x_1, \ldots, x_r)) = 0,\ x_1, \ldots, x_r \in GF(2),$$

φ is said to be *c-dependent*, and h is called a *dependent polynomial* of φ. If φ is not c-dependent, φ is said to be *c-independent*. If φ is c-dependent and $(c-1)$-independent, c is called the *dependent degree* of φ, denoted by c_φ, and $c-1$ is called the *independent degree* of φ, denoted by I_φ. In the case where φ is 1-dependent, φ is said to be *linearly dependent*. In the case where φ is 1-independent, φ is said to be *linearly independent*.

An affine transformation on R_2^r means a mapping $xC \oplus b$, where C is an $r \times r$ matrix over $GF(2)$, b is a row vector of dimension r over $GF(2)$.

Lemma 8.3.1. *Let φ be a transformation on R_2^r, and p and q two invertible affine transformations on R_2^r. Let $\varphi'(x) = p(\varphi(q(x)))$, $x \in R_2^r$. Then we have $c_\varphi = c_{\varphi'}$ and $I_\varphi = I_{\varphi'}$.*

Proof. Suppose that h is a dependent polynomial of φ. Since p and q are invertible affine transformations, p^{-1} and q^{-1} are invertible affine transformations. Let

$$h'(x, y) = h(q(x), p^{-1}(y)),\ x, y \in R_2^r.$$

Since h is a dependent polynomial of φ and q is invertible, it is easy to verify that h' is a dependent polynomial of φ'. Since q and p^{-1} are affine transformations, the degree of h' is not greater than the degree of h. Therefore, we have $c_{\varphi'} \leqslant c_\varphi$. Since p^{-1} and q^{-1} are invertible affine transformations and $\varphi(x) = p^{-1}(\varphi'(q^{-1}(x)))$, from symmetry, we have $c_\varphi \leqslant c_{\varphi'}$. Thus $c_{\varphi'} = c_\varphi$. From $I_{\varphi'} = c_{\varphi'} - 1$ and $I_\varphi = c_\varphi - 1$, we have $I_{\varphi'} = I_\varphi$. □

Theorem 8.3.1. *Let φ be a transformation on R_2^r, and φ_1 and φ_2 two invertible affine transformations on R_2^r. Let $\Phi = (\varphi_1, \varphi, \varphi_2)$, and A_Φ be the $(2^r, 2^r)$-Latin array of Φ. Then we have $c_{A_\Phi} = c_\varphi$, $I_{A_\Phi} = I_\varphi$, and $c_{A_\Phi} = I_{A_\Phi} + 1$.*

Proof. For any $x, y, w_1, w_2 \in R_2^r$, from the definition of A_Φ, the element of A_Φ at row $f_r(x) + 1$ column $f_{2r}(w_1, w_2) + 1$ is $f_r(y) + 1$ if and only if $y = \varphi_1(w_1) \oplus \varphi(\varphi_2(w_2) \oplus x)$, if and only if $w_1 = \varphi'_{xy}(w_2)$, where $\varphi'_{xy}(u) = p_y(\varphi(q_x(u)))$, $p_y(u) = \varphi_1^{-1}(u \oplus y)$, $q_x(u) = \varphi_2(u) \oplus x$, $u \in R_2^r$. It follows that A_Φ is $c_{\varphi'_{xy}}$-dependent with respect to $(f_r(x)+1, f_r(y)+1)$ and that A_Φ is $I_{\varphi'_{xy}}$-independent with respect to $(f_r(x) + 1, f_r(y) + 1)$. Since φ_1 and φ_2 are invertible affine transformations on R_2^r, p_y and q_x are invertible affine

transformations on R_2^r. From Lemma 8.3.1, we have $c_{\varphi'_{xy}} = c_\varphi$. Thus $I_{\varphi'_{xy}} = I_\varphi$. Therefore, A_Φ is c_φ-dependent and I_φ-independent. Since $c_\varphi = I_\varphi + 1$, we have $c_{A_\Phi} = c_\varphi$, $I_{A_\Phi} = I_\varphi$ and $c_{A_\Phi} = I_{A_\Phi} + 1$. □

Denote $c(r) = \min c\,[\,1 + \binom{2r}{1} + \cdots + \binom{2r}{c} > 2^r\,]$. We have the following.

Proposition 8.3.3. *For any transformation φ on R_2^r, we have $c_\varphi \leqslant c(r)$.*

Proof. We give a proof analogous to Proposition 8.3.1. For any $x \in R_2^r$, let α_x be the row vector

$$[1, w_1, \ldots, w_{2r}, w_1w_2, \ldots, w_{2r-1}w_{2r}, \ldots, w_1 \ldots w_{c(r)}, \ldots, w_{2r-c(r)+1} \ldots w_{2r}],$$

where $[\,x, \varphi(x)] = [w_1, \ldots, w_{2r}]$, $w_1, \ldots, w_{2r} \in GF(2)$. Let C be a matrix of which row $f_r(x) + 1$ is α_x, $x \in R_2^r$. Since the number of C's columns is greater than the number of C's rows, columns of C are linearly dependent. Thus there is a nonzero column vector γ such that $C\gamma = 0$. It follows that φ is $c(r)$-dependent. Therefore, $c_\varphi \leqslant c(r)$. □

Theorem 8.3.2. *Let $r \geqslant 3$. Then there is a permutation φ on R_2^r such that $c_\varphi \geqslant 2$.*

Proof. Denote $t_i = x_1 \ldots x_{i-1}x_{i+1} \ldots x_r$, $i = 1, \ldots, r$. Define a function φ on R_2^r so that $\varphi(x_1, \ldots, x_r) = [y_1, \ldots, y_r]$, where

$$\begin{aligned} y_1 &= x_1 \oplus t_1 \oplus t_2, \\ y_2 &= x_2 \oplus t_2 \oplus t_3, \\ &\cdots\cdots\cdots, \\ y_{r-1} &= x_{r-1} \oplus t_{r-1} \oplus t_r, \\ y_r &= x_r \oplus t_r. \end{aligned} \tag{8.2}$$

It is easy to verify that if the weight (number of 1) of $[x_1, \ldots, x_r]$ is less than $r-1$ then $\varphi(x_1, \ldots, x_r) = [x_1, \ldots, x_r]$, and that on the points of weights $r-1$ and r, φ is the cyclic permutation

$$(11 \ldots 110,\ 11 \ldots 101,\ \ldots,\ 101 \ldots 11,\ 011 \ldots 11,\ 111 \ldots 11).$$

Therefore, φ is invertible. We prove that φ is linearly independent. Suppose to the contrary that φ is linearly dependent. Then there exist c_0, …, c_r, $d_1, \ldots, d_r \in GF(2)$ such that at least one of them is nonzero and

$$c_0 \oplus c_1x_1 \oplus \cdots \oplus c_rx_r \oplus d_1y_1 \oplus \cdots \oplus d_ry_r = 0$$

holds for any $x_1, \ldots, x_r$ in $GF(2)$, where $y_1, \ldots, y_r$ are defined by (8.2). Notice that for the zero Boolean function all the coefficients of its polynomial

expression are zero. From the coefficients of t_1, we have $d_1 = 0$. From the coefficients of t_2, we have $d_2 = 0$. And so forth, from the coefficients of t_r, we have $d_r = 0$. It follows that $c_0 = \cdots = c_r = 0$. This is a contradiction. So φ is linearly independent. □

Corollary 8.3.1. *For any r, $1 \leqslant r \leqslant 6$, there is a permutation φ on R_2^r such that $c_\varphi = c(r)$.*

Proof. Whenever $r = 1, 2$, we have $c(r) = 1$. Since any permutation φ on R_2^r is 0-independent, from Proposition 8.3.3, $c_\varphi = c(r)$ holds.

Whenever $r = 3, \ldots, 6$, we have $c(r) = 2$. From Theorem 8.3.2, there is a permutation φ on R_2^r such that $c_\varphi \geqslant 2$. From Proposition 8.3.3, we have $c_\varphi \leqslant c(r) = 2$. Therefore, $c_\varphi = 2$ holds. □

8.3.2 Generation of Linearly Independent Permutations

Truth Table

Given $r > 0$, let φ be a transformation on R_2^r. Let W_i be a $\binom{r}{i} \times r$ matrix over $GF(2)$ of which rows consist of all difference vectors of dimension r with weight i, $i = 0, 1, \ldots, r$. We use I_t to denote the column vector of dimension $\binom{r}{t}$ of which each component is 1. For any i, $0 \leqslant i \leqslant r$, define a $\binom{r}{i} \times r$ matrix U_i over $GF(2)$ of which row j is the value of φ at row j of W_i, $1 \leqslant j \leqslant \binom{r}{i}$. Define a $2^r \times (1 + 2r)$ matrix

$$\Phi = \begin{bmatrix} I_0 & W_0 & U_0 \\ I_1 & W_1 & U_1 \\ \vdots & \vdots & \vdots \\ I_r & W_r & U_r \end{bmatrix}.$$

In this section, we refer to Φ as the *truth table* of φ, and denote the submatrix of the last r columns of Φ by U_φ. Notice that $W_0 = 0$. For convenience, we arrange rows of W_1 so that it is the identity matrix.

From the definitions, we have the following.

Proposition 8.3.4. (a) $c_\varphi > 1$ *if and only if columns of Φ are linearly independent.*

(b) *φ is invertible if and only if rows of U_φ are distinct.*

By E_t denote the $\binom{r}{t} \times \binom{r}{t}$ identity matrix. Let the $2^r \times 2^r$ matrix

$$P = \begin{bmatrix} I_0 & W_0 & & & & & & \\ I_1 & W_1 & & & & & & \\ I_2 & W_2 & E_2 & & & & & \\ 0 & W_3 & 0 & E_3 & & & & \\ I_4 & W_4 & 0 & 0 & E_4 & & & \\ \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & & \\ I'_{r-1} & W_{r-1} & 0 & 0 & 0 & \dots & E_{r-1} & \\ I'_r & W_r & 0 & 0 & 0 & \dots & 0 & E_r \end{bmatrix},$$

where $I'_j = I_j$ if j is even, $I'_j = 0$ otherwise. It is easy to verify that P is nonsingular and

$$P^{-1} = \begin{bmatrix} I_0 & W_0 & & & & & & \\ I_1 & W_1 & & & & & & \\ I_2 & W_2 & E_2 & & & & & \\ I_3 & W_3 & 0 & E_3 & & & & \\ I_4 & W_4 & 0 & 0 & E_4 & & & \\ \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & & \\ I_{r-1} & W_{r-1} & 0 & 0 & 0 & \dots & E_{r-1} & \\ I_r & W_r & 0 & 0 & 0 & \dots & 0 & E_r \end{bmatrix}.$$

Noticing that the first $1 + r$ columns of P^{-1} and of Φ are the same, we have the following.

Lemma 8.3.2. *$P\Phi$ is in the form*

$$P\Phi = \begin{bmatrix} I_0 & 0 & V_0 \\ 0 & E_1 & V_1 \\ 0 & 0 & V_2 \\ \vdots & \vdots & \vdots \\ 0 & 0 & V_r \end{bmatrix},$$

where $V_0 = U_0$, V_i is a $\binom{r}{i} \times r$ matrix, $i = 1, \dots, r$.

Let

$$V_\varphi = \begin{bmatrix} V_0 \\ \vdots \\ V_r \end{bmatrix}, \quad V_{\varphi -} = \begin{bmatrix} V_2 \\ \vdots \\ V_r \end{bmatrix}.$$

Since P is nonsingular, columns of Φ are linearly independent if and only if columns of $P\Phi$ are linearly independent. Using Lemma 8.3.2, columns of $P\Phi$ are linearly independent if and only if columns of $V_{\varphi -}$ are linearly independent. From Proposition 8.3.4 (a), we obtain the following proposition.

Proposition 8.3.5. *$c_\varphi > 1$ if and only if columns of $V_{\varphi-}$ are linearly independent.*

Lemma 8.3.3. *For any i, $1 \leqslant i \leqslant r$, and any $r \times r$ permutation matrix Q over $GF(2)$, there exists uniquely a $\binom{r}{i} \times \binom{r}{i}$ permutation matrix P_{iQ} such that $P_{iQ}W_i = W_iQ$.*

Proof. Notice that rows of W_i consist of all row vectors of dimension r with weight i. Since arranging columns of W_i keeps the weight of each row and sameness between rows unchanged, rows of W_iQ consist of all row vectors of dimension r with weight i. Therefore, there exists uniquely a permutation matrix P_{iQ} such that $P_{iQ}W_i = W_iQ$. □

For any $r \times r$ permutation matrix Q over $GF(2)$, let

$$D_Q = \begin{bmatrix} I_0 & & & & \\ & Q & & & \\ & & P_{2Q} & & \\ & & & \ddots & \\ & & & & P_{rQ} \end{bmatrix}.$$

Let $G'_r = \{D_Q \mid Q \text{ is an } r \times r \text{ permutation matrix over } GF(2)\}$. Clearly, in the case where Q is the identity matrix, P_{iQ} is the identity matrix. In the case of $P_{iQ}W_i = W_iQ$, we have $W_iQ^{-1} = P_{iQ}^{-1}(P_{iQ}W_i)Q^{-1} = P_{iQ}^{-1}(W_iQ)Q^{-1} = P_{iQ}^{-1}W_i$; therefore, $P_{iQ^{-1}} = P_{iQ}^{-1}$. In the case of $P_{iQ'}W_i = W_iQ'$ and $P_{iQ}W_i = W_iQ$, we have $P_{iQ}P_{iQ'}W_i = P_{iQ}W_iQ' = W_iQQ'$; therefore, $P_{iQ}P_{iQ'} = P_{i(QQ')}$. Thus G'_r is a group.

Let $G_r = \{\langle D_Q, \delta, C\rangle \mid Q$ is an $r \times r$ permutation matrix over $GF(2)$, δ is a row vector of dimension r over $GF(2)$, C is an $r \times r$ nonsingular matrix over $GF(2)$ $\}$.

For any $2^r \times r$ matrix V, partition it into blocks

$$V = \begin{bmatrix} V_0 \\ V_1 \\ \vdots \\ V_r \end{bmatrix},$$

where V_i has $\binom{r}{i}$ rows, $i = 0, 1, \ldots, r$. For any $\langle D_Q, \delta, C\rangle$ in G_r, define

$$V^{\langle D_Q,\delta,C\rangle} = D_Q(V \oplus \begin{bmatrix} \delta \\ 0 \\ \vdots \\ 0 \end{bmatrix})C = D_QVC \oplus \begin{bmatrix} \delta C \\ 0 \\ \vdots \\ 0 \end{bmatrix}.$$

For any $2^r \times r$ matrices V and V' over $GF(2)$, V and V' are said to be *equivalent* if there is $\langle D_Q, \delta, C\rangle$ in G_r such that $V^{\langle D_Q,\delta,C\rangle} = V'$. It is easy to verify that the equivalence relation is reflexive, symmetric and transitive.

Lemma 8.3.4. *Assume that* $V^{\langle D_Q,\delta,C\rangle} = V'$. *Then we have*

$$P^{-1}V' = D_Q(P^{-1}V)C \oplus \begin{bmatrix} \delta C \\ \delta C \\ \vdots \\ \delta C \end{bmatrix}.$$

Proof. Since P_{iQ} is a permutation matrix, we have $P_{iQ}I_i = I_i$. Using $P_{iQ}W_i = W_iQ$, it follows that

$$P^{-1}D_Q = \begin{bmatrix} I_0 & & & & \\ I_1 & Q & & & \\ I_2 & W_2Q & P_{2Q} & & \\ \vdots & \vdots & \vdots & \ddots & \\ I_r & W_rQ & 0 & \dots & P_{rQ} \end{bmatrix}$$

$$= \begin{bmatrix} I_0 & & & & \\ I_1 & Q & & & \\ I_2 & P_{2Q}W_2 & P_{2Q} & & \\ \vdots & \vdots & \vdots & \ddots & \\ I_r & P_{rQ}W_r & 0 & \dots & P_{rQ} \end{bmatrix} = D_QP^{-1}.$$

Since $V' = V^{\langle D_Q,\delta,C\rangle}$, we have

$$V' = D_QVC \oplus \begin{bmatrix} \delta C \\ 0 \\ \vdots \\ 0 \end{bmatrix}.$$

Therefore,

$$P^{-1}V' = P^{-1}D_QVC \oplus P^{-1}\begin{bmatrix} \delta C \\ 0 \\ \vdots \\ 0 \end{bmatrix} = D_Q(P^{-1}V)C \oplus \begin{bmatrix} \delta C \\ \delta C \\ \vdots \\ \delta C \end{bmatrix}. \qquad \square$$

For any $2^r \times r$ matrices V and V', denote the submatrices of V and of V' obtained by deleting their first $1 + r$ rows by V_- and V'_-, respectively.

Lemma 8.3.5. *Assume that two $2^r \times r$ matrices V and V' are equivalent.*

(a) *Columns of V_- are linearly independent if and only if columns of V'_- are linearly independent.*

(b) *Rows of $P^{-1}V$ are distinct if and only if rows of $P^{-1}V'$ are distinct.*

Proof. (a) Since V and V' are equivalent, there exists $\langle D_Q, \delta, C\rangle$ in G_r such that $V' = V^{\langle D_Q,\delta,C\rangle}$. Thus

$$V'_- = \begin{bmatrix} P_{2Q} & & \\ & \ddots & \\ & & P_{rQ} \end{bmatrix} V_- C.$$

Therefore, columns of V_- are linearly independent if and only if columns of V'_- are linearly independent.

(b) Since D_Q is a permutation matrix, rows of $P^{-1}V$ are distinct if and only if rows of $D_Q(P^{-1}V)$ are distinct. Since C is nonsingular, rows of $D_Q(P^{-1}V)$ are distinct if and only if rows of $D_Q(P^{-1}V)C$ are distinct. From Lemma 8.3.4, $P^{-1}V$ are distinct if and only if rows of $P^{-1}V'$ are distinct. □

Theorem 8.3.3. *Let φ and φ' be two transformations on R_2^r, and V_φ and $V_{\varphi'}$ be submatrices of the last r columns of $P\Phi$ and $P\Phi'$, respectively, where Φ and Φ' are truth tables of φ and φ', respectively. If V_φ and $V_{\varphi'}$ are equivalent, then the condition that $c_\varphi > 1$ and φ is invertible holds if and only if the condition that $c_{\varphi'} > 1$ and φ' is invertible holds.*

Proof. From Lemma 8.3.5, Proposition 8.3.4(b) and Proposition 8.3.5. □

On $S(V_0, V_1)$

For any row vector V_0 of dimension r over $GF(2)$ and any $r \times r$ matrix V_1 over $GF(2)$, we use $S(V_0, V_1)$ to denote a set of $2^r \times r$ matrices over $GF(2)$ such that $V \in S(V_0, V_1)$ if and only if the following conditions hold: the first row of V is V_0, the submatrix of rows 2 to $1 + r$ of V is V_1, columns of V_- are linearly independent, and rows of $P^{-1}V$ are distinct.

For any transformation φ on R_2^r, let V_φ be the last r columns of $P\Phi$, where Φ is the truth table of φ. Denote the first row of V_φ by $V_{\varphi 0}$, and the submatrix consisting of rows 2 to $r + 1$ of V_φ by $V_{\varphi 1}$. From Propositions 8.3.4 (b) and 8.3.5, if $c_\varphi > 1$ and φ is invertible, then $V_\varphi \in S(V_{\varphi 0}, V_{\varphi 1})$. Conversely, from Propositions 8.3.4 (b) and 8.3.5, for any $V \in S(V_0, V_1)$, if φ is the transformation on R_2^r with $V_\varphi = V$, then $c_\varphi > 1$ and φ is invertible.

Theorem 8.3.4. *Let δ be a row vector of dimension r over $GF(2)$, Q an $r \times r$ permutation matrix over $GF(2)$, and C an $r \times r$ nonsingular matrix over $GF(2)$. Then we have*

$$S((V_0 \oplus \delta)C, QV_1C) = \{V^{\langle D_Q, \delta, C\rangle} \mid V \in S(V_0, V_1)\},$$

and $|S((V_0 \oplus \delta)C, QV_1C)| = |S(V_0, V_1)|$.

Proof. For any $V \in S(V_0, V_1)$, from the definitions, using Lemma 8.3.5, we have $V^{\langle D_Q, \delta, C\rangle} \in S((V_0 \oplus \delta)C, QV_1C)$. Thus $S((V_0 \oplus \delta)C, QV_1C) \supseteq \{V^{\langle D_Q, \delta, C\rangle} \mid V \in S(V_0, V_1)\}$. Clearly, for any $V, \bar{V} \in S(V_0, V_1)$, if $V \neq \bar{V}$, then $V^{\langle D_Q, \delta, C\rangle} \neq \bar{V}^{\langle D_Q, \delta, C\rangle}$. It follows that $|S((V_0 \oplus \delta)C, QV_1C)| \geqslant |S(V_0, V_1)|$. On the other hand, we have $S(V_0, V_1) \supseteq \{V^{\langle D_{Q^{-1}}, \delta C, C^{-1}\rangle} \mid V \in S((V_0 \oplus \delta)C, QV_1C)\}$ and $|S((V_0 \oplus \delta)C, QV_1C)| \leqslant |S(V_0, V_1)|$. Thus we have $|S((V_0 \oplus \delta)C, QV_1C)| = |S(V_0, V_1)|$. It follows that $S((V_0 \oplus \delta)C, QV_1C) = \{V^{\langle D_Q, \delta, C\rangle} \mid V \in S(V_0, V_1)\}$. □

For any positive integer r, denote $G_r^{''} = \{\, \langle Q, C\rangle \mid Q$ is an $r \times r$ permutation matrix over $GF(2)$, C is an $r \times r$ nonsingular matrix over $GF(2\,\}$. Let $\cdot$ be an operation on $G_r^{''}$ defined by $\langle Q, C\rangle \cdot \langle Q', C'\rangle = \langle QQ', C'C\rangle$. It is easy to verify that $\langle G_r^{''}, \cdot\rangle$ is a group. For any $r \times r$ matrix V_1 over $GF(2)$ and any $\langle Q, C\rangle$ in $G_r^{''}$, denote $V_1^{\langle Q,C\rangle} = QV_1C$. V_1 and $V_1^{\langle Q,C\rangle}$ are said to be *equivalent* under $G_r^{''}$. It is easy to verify that the equivalence relation is reflexive, symmetric and transitive. Any equivalence class of the equivalence relation under $G_r^{''}$ includes $r \times r$ matrices in the form $\begin{bmatrix} E & 0 \\ B & 0 \end{bmatrix}$, where E is the identity matrix, columns of B are in decreasing order in some ordering. The minimum one in the ordering is called the *canonical form* of the equivalence class under $G_r^{''}$.

Notice that the property that rows of V_1 are nonzero and the distinct keeps unchanged under equivalence. Clearly, $S(0, V_1) \neq \varnothing$ implies that rows of V_1 are nonzero and distinct. From Theorem 8.3.4, generation of linear independent permutations can be reduced to generating $S(0, V_1)$, where V_1 ranges over canonical forms under $G_r^{''}$, and rows of V_1 are nonzero and distinct. For example, in the case of $r = 4$, V_1 has only three alternatives (see [38] for more details):

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 1 & 1 & 0 & 0 \end{bmatrix}, \quad \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 1 & 1 & 1 & 0 \end{bmatrix}.$$

Lemma 8.3.6. *Let*

$$R = \begin{bmatrix} I_0 & 0 & 0 \\ 0 & E_1 & 0 \\ 0 & W^* & P_R' \end{bmatrix}, \quad P_R = \begin{bmatrix} I_0 & & \\ & E_1 & \\ & & P_R' \end{bmatrix},$$

where P'_R *is a* $(2^r-1-r)\times(2^r-1-r)$ *permutation matrix over* $GF(2)$*, and*

$$W^* = W \oplus P'_R W, \quad W = \begin{bmatrix} W_2 \\ \vdots \\ W_r \end{bmatrix}.$$

Then we have $P_R P^{-1} = P^{-1}R$*, and* R *satisfying this equation is uniquely determined by* P_R*.*

Proof. Partition P^{-1} into blocks with 1, r, 2^r-1-r rows and columns in turn

$$P^{-1} = \begin{bmatrix} I_0 & 0 & 0 \\ I_1 & E_1 & 0 \\ I' & W & E' \end{bmatrix},$$

where I' is the column vector of dimension 2^r-1-r of which each component is 1, and E' is the $(2^r-1-r)\times(2^r-1-r)$ identity matrix. It is easy to verify that both $P_R P^{-1}$ and $P^{-1}R$ are equal to

$$\begin{bmatrix} I_0 & 0 & 0 \\ I_1 & E_1 & 0 \\ I' & P'_R W & P'_R \end{bmatrix}.$$

Therefore, $P_R P^{-1} = P^{-1}R$. We prove that if $P_R P^{-1} = P^{-1}R'$ then $R = R'$. Partition R' into blocks with 1, r, 2^r-1-r rows and columns in turn

$$R' = \begin{bmatrix} R_{11} & R_{12} & R_{13} \\ R_{21} & R_{22} & R_{23} \\ R_{31} & R_{32} & R_{33} \end{bmatrix}.$$

Since $P^{-1}R' = P_R P^{-1}$, we have $R_{11} = I_0$, $R_{12} = 0$, $R_{13} = 0$. It follows that $R_{21} = 0$, $R_{22} = E_1$, $R_{23} = 0$. Furthermore, we have $I' \oplus R_{31} = I'$, $W \oplus R_{32} = P'_R W$, $R_{33} = P'_R$. It follows that $R_{31} = 0$, $R_{32} = W \oplus P'_R W = W^*$. Therefore, $R' = R$. □

Lemma 8.3.7. *For any* $2^r \times r$ *matrices* V *and* V' *over* $GF(2)$*, the following two conditions are equivalent:* (a) *the first* $r+1$ *rows of* $P^{-1}V$ *and of* $P^{-1}V'$ *are the same, and rows of* $P^{-1}V'$ *are a permutation of rows of* $P^{-1}V$*;* (b) *the first* $r+1$ *rows of* V *and of* V' *are the same, and there exists a* $(2^r-1-r)\times(2^r-1-r)$ *permutation matrix* P'_R *such that* $V'_- = (E' \oplus P'_R)WV_1 \oplus P'_R V_-$*, where* V_- *and* V'_- *are the submatrices consisting of the last* 2^r-1-r *rows of* V *and* V'*, respectively,* V_1 *is the submatrix consisting of rows* 2 *to* $r+1$ *of* V*,* E' *is the* $(2^r-1-r)\times(2^r-1-r)$ *identity matrix, and* $W = \begin{bmatrix} W_2 \\ \vdots \\ W_r \end{bmatrix}$.

Proof. From the form of P^{-1}, it is easy to verify that the first $r+1$ rows of $P^{-1}V$ and $P^{-1}V'$ are the same if and only if the first $r+1$ rows of V and of V' are the same.

Suppose that the first $r+1$ rows of V and of V' are the same and that $V'_- = (E' \oplus P'_R)WV_1 \oplus P'_R V_-$. Let

$$R = \begin{bmatrix} I_0 & 0 & 0 \\ 0 & E_1 & 0 \\ 0 & (E' \oplus P'_R)W & P'_R \end{bmatrix}.$$

Then we have $V' = RV$. Therefore, $P^{-1}V' = P^{-1}RV$. Let

$$P_R = \begin{bmatrix} I_0 & & \\ & E_1 & \\ & & P'_R \end{bmatrix}.$$

From Lemma 8.3.6, we obtain $P^{-1}V' = P^{-1}RV = P_R P^{-1}V$. That is, rows of $P^{-1}V'$ are a permutation of rows of $P^{-1}V$.

Conversely, suppose that rows of $P^{-1}V'$ are a permutation of rows of $P^{-1}V$ and the first $r+1$ rows of them are the same. Then there exists a permutation matrix $P_R = \begin{bmatrix} I_0 & & \\ & E_1 & \\ & & P'_R \end{bmatrix}$ such that $P^{-1}V' = P_R P^{-1}V$. From Lemma 8.3.6, we obtain $P^{-1}V' = P^{-1}RV$. It follows that $V' = RV$. Therefore, $V'_- = (E' \oplus P'_R)WV_1 \oplus P'_R V_-$. □

Theorem 8.3.5. *Let V_0 and V_1 be $1 \times r$ and $r \times r$ matrices over $GF(2)$, respectively. Assume that rows of V_1 are distinct and nonzero. Let U_- be a $(2^r - 1 - r) \times r$ matrix over $GF(2)$ of which rows consist of all different row vectors of dimension r except V_0 and rows of $I_1V_0 \oplus V_1$. Then $S(V_0, V_1)$ is the set of all*

$$\begin{bmatrix} V_0 \\ V_1 \\ WV_1 \oplus P'_R(I'V_0 \oplus U_-) \end{bmatrix},$$

P'_R ranging over $(2^r - 1 - r) \times (2^r - 1 - r)$ permutation matrices such that columns of $WV_1 \oplus P'_R(I'V_0 \oplus U_-)$ are linearly independent, where I_1 and I' are column vectors of dimensions r and $2^r - 1 - r$ of which each component is 1, respectively.

Proof. Let

$$V = P \begin{bmatrix} V_0 \\ I_1V_0 \oplus V_1 \\ U_- \end{bmatrix}.$$

Clearly, the first row of V is V_0, the submatrix consisting of rows 2 to $r+1$ of V is V_1, and rows of $P^{-1}V$ are distinct. Since $U_- = I'V_0 \oplus WV_1 \oplus V_-$, we have $WV_1 \oplus V_- = I'V_0 \oplus U_-$.

Suppose that $V' \in S(V_0, V_1)$. Then the first row of V' is V_0, the submatrix consisting of rows 2 to $r+1$ of V' is V_1, columns of V'_- are linearly independent, and rows of $P^{-1}V'$ are distinct. Thus V and V' satisfy the condition (a) in Lemma 8.3.7. From Lemma 8.3.7, there exists a permutation matrix P'_R such that $V'_- = (E' \oplus P'_R)\ WV_1 \oplus P'_R V_- = WV_1 \oplus P'_R(WV_1 \oplus V_-)$. Since $WV_1 \oplus V_- = I'V_0 \oplus U_-$, we have $V'_- = WV_1 \oplus P'_R(I'V_0 \oplus U_-)$. Therefore, columns of $WV_1 \oplus P'_R(I'V_0 \oplus U_-)$ are linearly independent.

Conversely, suppose that P'_R is a permutation matrix and that columns of $WV_1 \oplus P'_R(I'V_0 \oplus U_-)$ are linearly independent. Let

$$V' = \begin{bmatrix} V_0 \\ V_1 \\ WV_1 \oplus P'_R(I'V_0 \oplus U_-) \end{bmatrix}.$$

Then columns of V'_- are linearly independent. Since $WV_1 \oplus V_- = I'V_0 \oplus U_-$, we have $V'_- = WV_1 \oplus P'_R(WV_1 \oplus V_-)$. Thus V and V' satisfy the condition (b) in Lemma 8.3.7; therefore, the condition (a) in Lemma 8.3.7 holds. Since rows of $P^{-1}V$ are distinct, rows of $P^{-1}V'$ are distinct. Therefore, $V' \in S(V_0, V_1)$. □

Problem $P(a_1, \ldots, a_k, b_1, \ldots, b_k)$

Given a row vector V_0 of dimension r over $GF(2)$ and an $r \times r$ matrix V_1 over $GF(2)$ with distinct and nonzero rows, we fix a $(2^r - 1 - r) \times r$ matrix U_- over $GF(2)$ such that rows of V_0, $I_1V_0 \oplus V_1$ and U_- are all different row vectors of dimension r. Denoting $A = WV_1$ and $B = I'V_0 \oplus U_-$, from Theorem 8.3.5, the problem on generation of $S(V_0, V_1)$ is reduced to choosing $(2^r - 1 - r) \times (2^r - 1 - r)$ permutation matrices P'_R's such that columns of $A \oplus P'_R B$ are linearly independent. The latter can be generalized to the following problem.

Problem $P(a_1, \ldots, a_k, b_1, \ldots, b_k)$: given row vectors $a_1, \ldots, a_k, b_1, \ldots, b_k$ of dimension r over $GF(2)$, find all permutations, say π, on $\{1, 2, \ldots, k\}$ such that columns of V_π are linearly independent, where

$$V_\pi = \begin{bmatrix} a_1 \oplus b_{\pi(1)} \\ a_2 \oplus b_{\pi(2)} \\ \vdots \\ a_k \oplus b_{\pi(k)} \end{bmatrix}.$$

Clearly, if $k < r$, then Problem $P(a_1, \ldots, a_k, b_1, \ldots, b_k)$ has no solution. Below we assume $k \geqslant r$.

For any sequences $(c_1, \ldots, c_r)$ and $(d_1, \ldots, d_r)$, if there exists i, $1 \leqslant i \leqslant r$, such that $c_1 = d_1$, ..., $c_{i-1} = d_{i-1}$ and $c_i < d_i$, $(c_1, \ldots, c_r)$ is said to be *less than* $(d_1, \ldots, d_r)$. For any solution π of Problem $P(a_1, \ldots, a_k, b_1, \ldots, b_k)$, consider the set of all sequences $(c_1, \ldots, c_r)$ such that $1 \leqslant c_1 < c_2 < \ldots < c_r \leqslant k$, rows $c_1, \ldots, c_r$ of V_π are linearly independent. The minimum sequence in the set is called the *rank-spectrum* of π. Since columns of V_π are linearly independent, the rank-spectrum of π is defined.

Denote the set of all solutions of Problem $P(a_1, \ldots, a_k, b_1, \ldots, b_k)$ with rank-spectrum $(c_1, \ldots, c_r)$ by $V(c_1, \ldots, c_r)$. Let

$$V(c_1, \ldots, c_r; d_1, \ldots, d_r) = \{\pi \mid \pi \in V(c_1, \ldots, c_r), \pi(c_1) = d_1, \ldots, \pi(c_r) = d_r\}.$$

Let $\pi \in V(c_1, \ldots, c_r; d_1, \ldots, d_r)$. Then $a_1 \oplus b_{\pi(1)} = \ldots = a_{c_1 - 1} \oplus b_{\pi(c_1 - 1)} = 0$ and $a_{c_1} \oplus b_{d_1} \neq 0$. Therefore, if for some $i < c_1$, a_i is different from each b_j, $j = 1, \ldots, k$, then $V(c_1, \ldots, c_r) = \varnothing$.

We use $R(c_1, \ldots, c_i, d_1, \ldots, d_i)$ to denote the vector space generated by $a_{c_1} \oplus b_{d_1}$, ..., $a_{c_i} \oplus b_{d_i}$, which is the 0-dimensional space $\{0\}$ in the case of $i = 0$. For any $I \subseteq \{1, \ldots, k\}$, let

$$\begin{aligned} &\Pi(I, h, c_1, \ldots, c_i, d_1, \ldots, d_i) \\ &\quad = \{j \mid j \in \{1, \ldots, k\} \setminus I,\ b_j \in a_h \oplus R(c_1, \ldots, c_i, d_1, \ldots, d_i)\}. \end{aligned}$$

Given $c_1, \ldots, c_r$ and $d_1, \ldots, d_r$ such that $1 \leqslant c_1 < c_2 < \cdots < c_r \leqslant k$ and that $d_1, \ldots, d_r$ are distinct elements in $\{1, \ldots, k\}$, denote $c_0 = 0$, $c_{r+1} = k+1$. Let

$$H_i = \{h \mid c_i < h < c_{i+1}\},\ i = 0, 1, \ldots, r.$$

Define a relation $\sim_i$ on H_i

$$h \sim_i h' \ \Leftrightarrow\ a_h \oplus a_{h'} \in R(c_1, \ldots, c_i, d_1, \ldots, d_i),$$

$0 \leqslant i \leqslant r$. Clearly, $\sim_i$ is reflexive, symmetric and transitive. We use H_{i1}, $H_{i2}, \ldots, H_{it_i}$ to denote all equivalence classes of $\sim_i$ on H_i.

Lemma 8.3.8. *Assume that* $1 \leqslant c_1 < c_2 < \cdots < c_r \leqslant k$ *and that* $d_1, \ldots, d_r$ *are distinct elements in* $\{1, \ldots, k\}$. *Let* $I \subseteq \{1, \ldots, k\}$. *Then for any* i, $0 \leqslant i \leqslant r$, *and any* h, $h' \in H_i$ *we have* (a) *if* $h \sim_i h'$ *holds, then* $\Pi(I, h, c_1, \ldots, c_i, d_1, \ldots, d_i) = \Pi(I, h', c_1, \ldots, c_i, d_1, \ldots, d_i)$, *and* (b) *if* $h \sim_i h'$ *does not hold, then* $\Pi(I, h, c_1, \ldots, c_i, d_1, \ldots, d_i)$ *and* $\Pi(I, h', c_1, \ldots, c_i, d_1, \ldots, d_i)$ *are disjoint.*

Proof. (a) Since $h \sim_i h'$ holds, we have $a_h \oplus a_{h'} \in R(c_1, \ldots, c_i, d_1, \ldots, d_i)$. Therefore, $a_h \oplus R(c_1, \ldots, c_i, d_1, \ldots, d_i) = a_{h'} \oplus R(c_1, \ldots, c_i, d_1, \ldots, d_i)$. It follows that $\Pi(I, h, c_1, \ldots, c_i, d_1, \ldots, d_i) = \Pi(I, h', c_1, \ldots, c_i, d_1, \ldots, d_i)$.

(b) Since $h \sim_i h'$ does not hold, $a_h \oplus a_{h'}$ is not in $R(c_1, \ldots, c_i, d_1, \ldots, d_i)$. It follows that the coset $a_h \oplus R(c_1, \ldots, c_i, d_1, \ldots, d_i)$ and the coset $a_{h'} \oplus R(c_1, \ldots, c_i, d_1, \ldots, d_i)$ are disjoint. Therefore, $\Pi\ (I, h, c_1, \ldots, c_i, d_1, \ldots, d_i)$ and $\Pi\ (I, h', c_1, \ldots, c_i, d_1, \ldots, d_i)$ are disjoint. □

Lemma 8.3.9. *Assume that* $1 \leqslant c_1 < c_2 < \cdots < c_r \leqslant k$ *and that* $d_1, \ldots, d_r$ *are distinct elements in* $\{1, \ldots, k\}$. *Let* $I \subseteq \{1, \ldots, k\}$. *Then for any* i *and* i', $0 \leqslant i' < i \leqslant r$, *any* $h \in H_i$ *and any* $h' \in H_{i'}$, $\Pi(I, h', c_1, \ldots, c_{i'}, d_1, \ldots, d_{i'})$ *is a subset of* $\Pi(I, h, c_1, \ldots, c_i, d_1, \ldots, d_i)$ *or they are disjoint.*

Proof. Suppose that the intersection set of $\Pi(I, h', c_1, \ldots, c_{i'}, d_1, \ldots, d_{i'})$ and $\Pi(I, h, c_1, \ldots, c_i, d_1, \ldots, d_i)$ is nonempty. Then the intersection set of $a_{h'} \oplus R(c_1, \ldots, c_{i'}, d_1, \ldots, d_{i'})$ and $a_h \oplus R(c_1, \ldots, c_i, d_1, \ldots, d_i)$ is nonempty. This yields $h \sim_i h'$, since $R(c_1, \ldots, c_{i'}, d_1, \ldots, d_{i'})$ is a subspace of $R(c_1, \ldots, c_i, d_1, \ldots, d_i)$. Thus $a_{h'} \oplus R(c_1, \ldots, c_{i'}, d_1, \ldots, d_{i'}) \subseteq a_{h'} \oplus R(c_1, \ldots, c_i, d_1, \ldots, d_i) = a_h \oplus R(c_1, \ldots, c_i, d_1, \ldots, d_i)$. It follows that $\Pi(I, h', c_1, \ldots, c_{i'}, d_1, \ldots, d_{i'}) \subseteq \Pi(I, h, c_1, \ldots, c_i, d_1, \ldots, d_i)$. □

Let $N_{0j} = \varnothing$ for any j. For any i, j, $1 \leqslant i \leqslant r$, $1 \leqslant j \leqslant t_i$, let

$$N_{ij} = \{\, (i', j') \mid 0 \leqslant i' < i,\ 1 \leqslant j' \leqslant t_{i'},\ (\exists h)_{H_{ij}} (\exists h')_{H_{i'j'}} [\Pi(I, h', c_1, \ldots, c_{i'}, d_1, \ldots, d_{i'}) \subseteq \Pi(I, h, c_1, \ldots, c_i, d_1, \ldots, d_i)] \,\}.$$

Theorem 8.3.6. *Assume that* $1 \leqslant c_1 < c_2 < \cdots < c_r \leqslant k$ *and that* $d_1, \ldots, d_r \in \{1, \ldots, k\}$ *are distinct. Let* $I = \{d_1, \ldots, d_r\}$. *Then for any transformation* π *on* $\{1, \ldots, k\}$, π *is in* $V(c_1, \ldots, c_r; d_1, \ldots, d_r)$ *if and only if the following conditions hold:*

(a) $\pi(c_j) = d_j$, $j = 1, \ldots, r$;

(b) $a_{c_1} \oplus b_{d_1}, \ldots, a_{c_r} \oplus b_{d_r}$ *are linearly independent;*

(c) *any* i, j, $0 \leqslant i \leqslant r$, $1 \leqslant j \leqslant t_i$, $|\pi(H_{ij})| = |H_{ij}|$ *and*

$$\pi(H_{ij}) \subseteq \Pi(I, h_{ij}, c_1, \ldots, c_i, d_1, \ldots, d_i) \setminus \bigcup_{(i', j') \in N_{ij}} \pi(H_{i', j'}),$$

where h_{ij} *is an arbitrary element in* H_{ij}.

Proof. only if : Suppose that $\pi \in V(c_1, \ldots, c_r; d_1, \ldots, d_r)$. From the definition of $V(c_1, \ldots, c_r; d_1, \ldots, d_r)$, (a) and (b) are obvious. Since π is a permutation, numbers of elements of H_{ij} and $\pi(H_{ij})$ are the same. For any i, i', $0 \leqslant i' < i \leqslant r$, since H_i and $H_{i'}$ are disjoint, $\pi(H_i)$ and

$\pi(H_{i'})$ are disjoint. It follows that $\pi(H_{ij})$ and $\bigcup_{(i',j')\in N_{ij}} \pi(H_{i',j'})$ are disjoint. On the other hand, for any h in H_{ij}, since $(c_1, \ldots, c_r)$ is the rank-spectrum of π, $a_h \oplus b_{\pi(h)}$ is in $R(c_1, \ldots, c_i, d_1, \ldots, d_i)$. Since π is a permutation, $\pi(h)$ is not in I. Therefore, $\pi(h) \in \Pi(I, h, c_1, \ldots, c_i, d_1, \ldots, d_i)$. Since $h \sim_i h_{ij}$, from Lemma 8.3.8, we obtain $\Pi(I, h, c_1, \ldots, c_i, d_1, \ldots, d_i) = \Pi(I, h_{ij}, c_1, \ldots, c_i, d_1, \ldots, d_i)$. Thus $\pi(h) \in \Pi(I, h_{ij}, c_1, \ldots, c_i, d_1, \ldots, d_i)$. It follows that $\pi(H_{ij}) \subseteq \Pi(I, h_{ij}, c_1, \ldots, c_i, d_1, \ldots, d_i)$. Therefore, (c) holds.

if : Suppose that the transformation π satisfies conditions (a), (b) and (c). First of all, we prove a proposition by reduction to absurdity: for any i, i', $0 \leqslant i' < i \leqslant r$, $\pi(H_i)$ and $\pi(H_{i'})$ are disjoint. Suppose to the contrary that there exist i and i', $0 \leqslant i' < i \leqslant r$, such that $\pi(H_i)$ and $\pi(H_{i'})$ intersect. Then there exist j and j', $1 \leqslant j \leqslant t_i$, $1 \leqslant j' \leqslant t_{i'}$, such that $\pi(H_{ij})$ and $\pi(H_{i'j'})$ intersect. Since (c) holds, $\Pi(I, h, c_1, \ldots, c_i, d_1, \ldots, d_i)$ and $\Pi(I, h', c_1, \ldots, c_{i'}, d_1, \ldots, d_{i'})$ intersect, for any $h \in H_{ij}$, $h' \in H_{i'j'}$. Using Lemma 8.3.9, $\Pi(I, h', c_1, \ldots, c_{i'}, d_1, \ldots, d_{i'})$ is a subset of $\Pi(I, h, c_1, \ldots, c_i, d_1, \ldots, d_i)$. Therefore, $(i', j') \in N_{ij}$. Since $\pi(H_{ij})$ and $\pi(H_{i'j'})$ intersect, $\pi(H_{ij})$ and $\bigcup_{(s,t)\in N_{ij}} \pi(H_{st})$ intersect. This contradicts the condition (c). Thus the proposition holds. From (a), (c) and the proposition, it is easy to show that π is a permutation if and only if for any i, $0 \leqslant i \leqslant r$, any j and j', $1 \leqslant j' < j \leqslant t_i$, $\pi(H_{ij})$ and $\pi(H_{ij'})$ are disjoint. Using Lemma 8.3.8, whenever $h \sim_i h'$ does not hold, Π $(I, h, c_1, \ldots, c_i, d_1, \ldots, d_i)$ and Π $(I, h', c_1, \ldots, c_i, d_1, \ldots, d_i)$ are disjoint. From (c), it is easy to see that for any i, $0 \leqslant i \leqslant r$, any j and j', $1 \leqslant j' < j \leqslant t_i$, $\pi(H_{ij})$ and $\pi(H_{ij'})$ are disjoint. Thus π is a permutation. From (b) and (c), it is easy to prove that $(c_1, \ldots, c_r)$ is the rank-spectrum of π. It immediately follows, using (a), that $\pi \in V(c_1, \ldots, c_r, d_1, \ldots, d_r)$. □

Solutions π and π' of Problem $P(a_1, \ldots, a_k, b_1, \ldots, b_k)$ are said to be *equivalent*, if rank-spectra of π and π' are the same, say $(c_1, \ldots, c_r)$, and $\pi(c_i) = \pi'(c_i)$ for $i = 1, \ldots, r$, $\pi(H_{ij}) = \pi'(H_{ij})$ for $i = 0, 1, \ldots, r$ and $j = 1, \ldots, t_i$. (In the definition of H_{ij}, the value of d_i is taken as $\pi(c_i)$, $i = 1, \ldots, r$.) It is easy to verify that the equivalent relation is reflexive, symmetric and transitive.

Corollary 8.3.2. *If $V(c_1, \ldots, c_r; d_1, \ldots, d_r) \neq \varnothing$, then the number of solutions in each equivalence class is $\prod_{0\leqslant i\leqslant r, 1\leqslant j\leqslant t_i} |H_{ij}|!$, and the equivalence class containing π can be obtained by changing the restriction of π on H_{ij} to bijections from H_{ij} to $\pi(H_{ij})$ for $i = 0, 1, \ldots, r$ and $j = 1, \ldots, t_i$.*

Corollary 8.3.3. *Assume that $1 \leqslant c_1 < c_2 < \cdots < c_r \leqslant k$ and that $d_1, \ldots, d_r \in \{1, \ldots, k\}$ are distinct. Let $I = \{d_1, \ldots, d_r\}$. If there exist i, j, $0 \leqslant i < r$, $1 \leqslant j \leqslant t_i$, such that $|H_{ij}| + \sum_{(i',j')\in N_{ij}} |H_{i'j'}| >$*

$|\Pi(I, h_{ij}, c_1, \ldots, c_i, d_1, \ldots, d_i)|$, *where* $h_{ij} \in H_{ij}$, *then* $V(c_1, \ldots, c_r; d_1, \ldots, d_r) = \varnothing$.

Noticing that H_{ij} and $\Pi(I, h_{ij}, c_1, \ldots, c_i, d_1, \ldots, d_i)$ do not depend on parameters $c_{i+2}, \ldots, c_r, d_{i+1}, \ldots, d_r$, Corollary 8.3.3 can be generalized to the following.

Corollary 8.3.4. *Let* $0 \leqslant i < r$. *Assume that* $1 \leqslant c_1 < c_2 < \cdots < c_{i+1} \leqslant k - r + i + 1$ *and that* $d_1, \ldots, d_i \in \{1, \ldots, k\}$ *are distinct. Let* $I = \{d_1, \ldots, d_i\}$. *If there exists* j, $1 \leqslant j \leqslant t_i$ *such that* $|H_{ij}| + \sum_{(i',j') \in N_{ij}} |H_{i'j'}| > |\Pi(I, h_{ij}, c_1, \ldots, c_i, d_1, \ldots, d_i)|$, *where* $h_{ij} \in H_{ij}$, *then for any* $c_{i+2}, \ldots, c_r$, $d_{i+1}, \ldots, d_r$, *we have* $V(c_1, \ldots, c_r; d_1, \ldots, d_r) = \varnothing$.

Theorem 8.3.6 and Corollaries 8.3.3 and 8.3.4 give criteria to decide whether $V(c_1, \ldots, c_r; d_1, \ldots, d_r)$ is empty and a method of enumerating its elements in nonempty case. The first step is computing $a_{c_j} \oplus b_{d_j}$, $j = 1, \ldots, r$ and deciding whether they are linear independent; in the case of $V(c_1, \ldots, c_r; d_1, \ldots, d_r) \neq \varnothing$, they must be linearly independent. The second step is, for each i from 1 to r in turn, choosing $\pi(H_{ij})$, $1 \leqslant j \leqslant t_i$, so that the condition (c) in Theorem 8.3.6 holds. In the case of $V(c_1, \ldots, c_r; d_1, \ldots, d_r) \neq \varnothing$, this process can go on, that is, the circumstances without enough elements for choosing mentioned in Corollaries 8.3.3 and 8.3.4 can not happen. Point out that $t_r = 1$ and $\Pi(I, h, c_1, \ldots, c_r, d_1, \ldots, d_r) = \{1, \ldots, k\} - I$, so $\pi(H_{r1})$ is unique, i.e., $\pi(H_{r1}) = \{1, \ldots, k\} - \{\pi(i), i = 1, \ldots, c_r\}$. The third step is choosing π so that $\pi(c_i) = d_i$ for $i = 1, \ldots, r$ and that the restriction of π on H_{ij} is a surjection from H_{ij} to $\pi(H_{ij})$ (in fact, a bijection because of $|H_{ij}| = |\pi(H_{ij})|$) for $i = 0, 1, \ldots, r$ and $j = 1, \ldots, t_i$.

Notice that in the sense of equivalence π is determined by $\pi(H_{ij})$, $i = 0, 1, \ldots, r$, $j = 1, \ldots, t_i$. From the method mentioned above we have the following.

Corollary 8.3.5. *Assume that* $0 \leqslant c_1 < c_2 < \cdots < c_r \leqslant k$ *and that* $d_1, \ldots, d_r \in \{1, \ldots, k\}$ *are distinct. Let* $I = \{d_1, \ldots, d_r\}$. *For any* i, j, $1 \leqslant i \leqslant r$, $1 \leqslant j \leqslant t_i$, *let* $q_{ij} = |H_{ij}|$ *and*

$$s_{ij} = |\Pi(I, h, c_1, \ldots, c_i, d_1, \ldots, d_i)| - \sum_{(i',j') \in N_{ij}} q_{i'j'},$$

where h *is an arbitrary element in* H_{ij}.

(a) $V(c_1, \ldots, c_r; d_1, \ldots, d_r) \neq \varnothing$ *if and only if* $a_{c_1} \oplus b_{d_1}, \ldots, a_{c_r} \oplus b_{d_r}$ *are linearly independent and* $(\forall i)(\forall j)[0 \leqslant i \leqslant r \,\&\, 1 \leqslant j \leqslant t_i \to q_{ij} \leqslant s_{ij}]$.

(b) *For nonempty* $V(c_1, \ldots, c_r; d_1, \ldots, d_r)$, *the number of its equivalence classes is*

$$\prod_{0 \leqslant i < r, 1 \leqslant j \leqslant t_i} \binom{s_{ij}}{q_{ij}}$$

and the number of its elements is

$$\Big(\prod_{0\leqslant i<r,1\leqslant j\leqslant t_i}\tbinom{s_{ij}}{q_{ij}}\cdot q_{ij}!\,\Big)\cdot(k-c_r)!.$$

Example 8.3.1. Let a_i and b_i be the i-th row of A and B, respectively, where

$$A=\begin{bmatrix}1&1&1&0\\1&0&0&0\\0&1&1&0\\0&1&0&0\\1&0&1&0\\1&1&0&0\\1&0&1&0\\0&1&1&0\\0&0&0&0\\1&1&1&0\\0&0&1&0\end{bmatrix},\qquad B=\begin{bmatrix}0&0&0&1\\0&0&1&1\\0&1&0&1\\0&1&1&0\\1&0&0&1\\1&0&1&0\\0&1&1&1\\1&0&1&1\\1&1&0&1\\1&1&1&0\\1&1&1&1\end{bmatrix}.$$

Find $V(2,6,8,9;5,11,1,9)$.

We first compute

$$C=\begin{bmatrix}a_2\oplus b_5\\a_6\oplus b_{11}\\a_8\oplus b_1\\a_9\oplus b_9\end{bmatrix}=\begin{bmatrix}0&0&0&1\\0&0&1&1\\0&1&1&1\\1&1&0&1\end{bmatrix}.$$

Let R_j be the vector space spanned by the first j rows of C, $j=1,2,3$. Let $R_0=\{0\}$ be the vector space of dimension 0. We use Π_{ij} to denote the set of all $h\in I'$ such that $a_k\oplus b_h\in R_i$, where $I'=\{1,\dots,11\}\setminus\{5,11,1,9\}$, k is an arbitrarily given element in H_{ij}. We compute H_{ij} and Π_{ij}.

$H_0=H_{01}=\{1\}$, $\Pi_{01}=\{10\}$.

$H_1=\{3,4,5\}$, $H_{11}=\{3\}$, $H_{12}=\{4\}$, $H_{13}=\{5\}$, $\Pi_{11}=\{4,7\}$, $\Pi_{12}=\{3\}$, $\Pi_{13}=\{6,8\}$.

$H_2=H_{21}=\{7\}$, $\Pi_{21}=\{6,8\}$.

$H_3=\varnothing$.

$H_4=H_{41}=\{10,11\}$, $\Pi_{01}=I'$.

We define the permutation π on $\{1,\dots,11\}$.

First define $\pi(2)=5$, $\pi(6)=11$, $\pi(8)=1$, $\pi(9)=9$.

Then for points with $|H_{ij}|=|\Pi_{ij}|$, define $\pi(1)=10$, $\pi(4)=3$. Similarly, define $\pi(\{5,7\})=\{6,8\}$.

Now define $\pi(3)\in\{4,7\}$. In the case where $\pi(3)=4$, define $\pi(\{10,11\})=\{2,7\}$. In the case where $\pi(3)=7$, define $\pi(\{10,11\})=\{2,4\}$.

To sum up, the solutions of $\pi(1),\dots,\pi(11)$ are:

$$10, 5, 4, 3, 6, 11, 8, 1, 9, 2, 7;$$
$$10, 5, 4, 3, 6, 11, 8, 1, 9, 7, 2;$$
$$10, 5, 7, 3, 6, 11, 8, 1, 9, 2, 4;$$
$$10, 5, 7, 3, 6, 11, 8, 1, 9, 4, 2;$$
$$10, 5, 4, 3, 8, 11, 6, 1, 9, 2, 7;$$
$$10, 5, 4, 3, 8, 11, 6, 1, 9, 7, 2;$$
$$10, 5, 7, 3, 8, 11, 6, 1, 9, 2, 4;$$
$$10, 5, 7, 3, 8, 11, 6, 1, 9, 4, 2.$$

For any solution π in the example, let

$$V_0 = \begin{bmatrix} 0\ 0\ 0\ 0 \end{bmatrix}, \quad V_1 = \begin{bmatrix} 1\ 0\ 0\ 0 \\ 0\ 1\ 0\ 0 \\ 0\ 0\ 1\ 0 \\ 1\ 1\ 0\ 0 \end{bmatrix},$$

and P'_R be the 11×11 permutation matrix of which the element at row i column $\pi(i)$ is 1. From Theorem 8.3.5,

$$\begin{bmatrix} V_0 \\ V_1 \\ WV_1 \oplus P'_R(I'V_0 \oplus U_-) \end{bmatrix}$$

is in $S(V_0, V_1)$, where $WV_1 = A$, $U_- = B$, and rows of W_i in W are arranged in increasing order, that is,

$$W = \begin{bmatrix} 0\ 0\ 1\ 1 \\ 0\ 1\ 0\ 1 \\ 0\ 1\ 1\ 0 \\ 1\ 0\ 0\ 1 \\ 1\ 0\ 1\ 0 \\ 1\ 1\ 0\ 0 \\ 0\ 1\ 1\ 1 \\ 1\ 0\ 1\ 1 \\ 1\ 1\ 0\ 1 \\ 1\ 1\ 1\ 0 \\ 1\ 1\ 1\ 1 \end{bmatrix}.$$

Therefore,

$$P^{-1} \begin{bmatrix} V_0 \\ V_1 \\ WV_1 \oplus P'_R(I'V_0 \oplus U_-) \end{bmatrix} = \begin{bmatrix} 0 \\ V_1 \\ P'_R B \end{bmatrix}$$

gives the last 4 columns of the truth table of a permutation φ on R_2^4 with $c_\varphi > 1$.

Historical Notes

Finite automata are regarded as mathematical models of ciphers in [91, 88, 33] for example. Based on [99, 100], a one key cryptosystem is proposed in [101] to model the ciphers which can be realized by finite automata and possess features of bounded error propagation and no plaintext expansion. Section 8.1 is based on [101]. This model consists of four part: a segment of recent ciphertext history, an autonomous finite automaton, a discrete function, and a permutational family. In [115], the concept of Latin array is introduced for studying permutational families, which is a generalization of Latin square. Section 8.2 is based on [115, 116, 43] and an unpublished manuscript [114]; the omitted parts in the proofs of Lemmas 8.2.7 and 8.2.9 and Theorem 8.2.15 can be found in [114]. And Sect. 8.3 is based on [119].

9. Finite Automaton Public Key Cryptosystems

Renji Tao

Institute of Software, Chinese Academy of Sciences
Beijing 100080, China trj@ios.ac.cn

Summary.

Since the introduction of the concept of public key cryptosystems by Diffie and Hellman[32], many concrete cryptosystems had been proposed and found applications in the area of information security; almost all are block. In this chapter, we present a sequential one, the so-called finite automaton public key cryptosystem; it can be used for encryption as well as for implementing digital signatures. The public key is a compound finite automaton of $n+1\,(\geqslant 2\,)$ finite automata and states, the private key is the $n+1$ weak inverse finite automata of them and states; no feasible inversion algorithm for the compound finite automaton is known unless its decomposition is known. Chapter 3 gives implicitly a feasible method to construct the $2n+2$ finite automata. We restrict the $2n+2$ finite automata to memory finite automata in the first five sections; in the last section, we use pseudo-memory finite automata to construct generalized cryptosystems.

Security of finite automaton public key cryptosystems is discussed in Sects. 9.4 and 9.5, which is heavily dependent on Chap. 4 and Sect. 2.3.

Key words: *public key cryptosystem, FAPKC, finite automata*

Since the introduction of the concept of public key cryptosystems by Diffie and Hellman[32], many concrete cryptosystems had been proposed and found applications in the area of information security; almost all are block. In this chapter, we present a sequential one, the so-called finite automaton public key cryptosystem; it can be used for encryption as well as for implementing digital signatures. The public key is a compound finite automaton of $n+1(\geqslant 2)$ finite automata and states, the private key is the $n+1$ weak inverse finite automata of them and states; no feasible inversion algorithm for the compound finite automaton is known unless its decomposition is known.

Chapter 3 gives implicitly a feasible method to construct the $2n+2$ finite automata. We restrict the $2n+2$ finite automata to memory finite automata in the first five sections; in the last section, we use pseudo-memory finite automata to construct generalized cryptosystems.

Security of finite automaton public key cryptosystems is discussed in Sects. 9.4 and 9.5, which is heavily depend on Chap. 4 and Sect. 2.3.

9.1 Theoretical Fundamentals

Throughout this chapter, for any integer i, any positive integer k and any symbol string z, we still use $z(i,k)$ to denote the symbol string $z_i, z_{i-1}, \ldots, z_{i-k+1}$. For any (r,t)-order memory finite automaton $M = \langle X, Y, S, \delta, \lambda\rangle$, any (r',t')-order memory finite automaton $M' = \langle Y, X, S', \delta', \lambda'\rangle$, and any nonnegative integer τ, we use $PI(M, M', \tau)$ to denote the following condition:

> For any state $s = \langle y(-1,t), x(-1,r)\rangle$ of M and any state $s' = \langle x(-1,t'), y(\tau-1,r')\rangle$ of M', and any $x_0, x_1, \ldots \in X$, $y_\tau, y_{\tau+1}, \ldots \in Y$, if $y_0 y_1 \ldots = \lambda(s, x_0 x_1 \ldots)$, then $x_0 x_1 \ldots = \lambda'(s', y_\tau y_{\tau+1} \ldots)$.

For any i, $0 \leqslant i \leqslant n$, let X_i be the column vector space over $GF(q)$ of dimension l_i. Let Y be the column vector space over $GF(q)$ of dimension m, and $X = X_n$.

For any i, $1 \leqslant i \leqslant n$, let $M_i = \langle X_i, X_{i-1}, X_i^{r_i}, \delta_i, \lambda_i\rangle$ be an r_i-order input-memory finite automaton, $M_i^* = \langle X_{i-1}, X_i, X_i^{r_i} \times X_{i-1}^{\tau_i}, \delta_i^*, \lambda_i^*\rangle$ a (τ_i, r_i)-order memory finite automaton, and $\tau_i \leqslant r_i$.

Let $M_0 = \langle X_0, Y, Y^{t_0} \times X_0^{r_0}, \delta_0, \lambda_0\rangle$ be an (r_0, t_0)-order memory finite automaton, $M_0^* = \langle Y, X_0, X_0^{t_0^*} \times Y^{r_0^*}, \delta_0^*, \lambda_0^*\rangle$ an (r_0^*, t_0^*)-order memory finite automaton, and $\tau_0 \leqslant r_0$.

Theorem 9.1.1. *Assume that M_i^*, M_i and τ_i satisfy $PI(M_i^*, M_i, \tau_i)$, $i = 0, 1, \ldots, n$. Let $s_{-b_{i-1}}^{(i)*} = \langle x^{(i)}(-b_{i-1}-1, r_i), \bar{x}^{(i-1)}(-b_{i-1}-1, \tau_i)\rangle$ be a state of M_i^*, $i = 1, \ldots, n$, and let $x_{-b_0}^{(0)}, \ldots, x_{-1}^{(0)} \in X_0$,*

$$\begin{aligned} &x_{-b_{i-1}}^{(i)} \ldots x_{-1}^{(i)} = \lambda_i^*(s_{-b_{i-1}}^{(i)*}, x_{-b_{i-1}}^{(i-1)} \ldots x_{-1}^{(i-1)}),\\ &s_0^{(i)*} = \delta_i^*(s_{-b_{i-1}}^{(i)*}, x_{-b_{i-1}}^{(i-1)} \ldots x_{-1}^{(i-1)}),\\ &\quad i = 1, \ldots, n, \end{aligned} \tag{9.1}$$

where $b_0 = r_0 - \tau_0$, $b_i = b_{i-1} + r_i - \tau_i$, $i = 1, \ldots, n$. Let $s_0^{(0)*} = \langle x^{(0)}(-1, t_0^*), y(-1, r_0^*)\rangle$ be a state of M_0^*. If

$$\begin{aligned} &x_0^{(0)} x_1^{(0)} \ldots = \lambda_0^*(s_0^{(0)*}, y_0 y_1 \ldots),\\ &x_0^{(i)} x_1^{(i)} \ldots = \lambda_i^*(s_0^{(i)*}, x_0^{(i-1)} x_1^{(i-1)} \ldots),\\ &\quad i = 1, \ldots, n, \end{aligned} \tag{9.2}$$

then

$$y_0y_1\ldots=\lambda'(s',x_\tau^{(n)}x_{\tau+1}^{(n)}\ldots), \tag{9.3}$$

where λ' *is the output function of* $C'(M_n,\ldots,M_1,M_0)$, $s'=\langle y(-1,t_0),x^{(n)}(\tau-1,\,r)\rangle$, $r=r_0+\cdots+r_n$, *and* $\tau=\tau_0+\cdots+\tau_n$.

Proof. Suppose that (9.2) holds. From (9.1) and (9.2), it is easy to obtain that

$$x_{-b_{i-1}}^{(i)}\ldots x_{-1}^{(i)}x_0^{(i)}x_1^{(i)}\ldots=\lambda_i^*(s_{-b_{i-1}}^{(i)*},x_{-b_{i-1}}^{(i-1)}\ldots x_{-1}^{(i-1)}x_0^{(i-1)}x_1^{(i-1)}\ldots),$$

$i=1,\ldots,n$. Since $PI(M_i^*,M_i,\tau_i)$ holds, this yields that

$$x_{-b_{i-1}}^{(i-1)}\ldots x_{-1}^{(i-1)}x_0^{(i-1)}x_1^{(i-1)}\ldots=\lambda_i(s^{(i)},x_{-b_{i-1}+\tau_i}^{(i)}x_{-b_{i-1}+\tau_i+1}^{(i)}\ldots), \tag{9.4}$$

$i=1,\ldots,n$, where $s^{(i)}=\langle x^{(i)}(-b_{i-1}+\tau_i-1,r_i)\rangle$. Since $PI(M_0^*,M_0,\tau_0)$ holds, from the part on M_0^* in (9.2), we have

$$y_0y_1\ldots=\lambda_0(s^{(0)},x_{\tau_0}^{(0)}x_{\tau_0+1}^{(0)}\ldots), \tag{9.5}$$

where $s^{(0)}=\langle y(-1,t_0),x^{(0)}(\tau_0-1,r_0)\rangle$.

For $1\leqslant i\leqslant n-1$, let λ_i' be the output function of $C'(M_n,\ldots,M_i)$. From (9.4) for $i=n-1,n$, by Theorem 1.2.1 we have

$$x_{-b_{n-2}}^{(n-2)}x_{-b_{n-2}+1}^{(n-2)}\ldots=\lambda_{n-1}'(\langle x^{(n)}(-b_{n-2}+\tau_{n-1}+\tau_n-1,r_{n-1}+r_n)\rangle,\\ x_{-b_{n-2}+\tau_{n-1}+\tau_n}^{(n)}x_{-b_{n-2}+\tau_{n-1}+\tau_n+1}^{(n)}\ldots).$$

Suppose that we have proven that

$$x_{-b_i}^{(i)}x_{-b_i+1}^{(i)}\ldots=\lambda_{i+1}'(\langle x^{(n)}(-b_i+\tau_{i+1}+\cdots+\tau_n-1,r_{i+1}+\cdots+r_n)\rangle,\\ x_{-b_i+\tau_{i+1}+\cdots+\tau_n}^{(n)}x_{-b_i+\tau_{i+1}+\cdots+\tau_n+1}^{(n)}\ldots)$$

for $i\geqslant 1$, we prove the case of λ_i'. From the above equation and (9.4), noticing $-b_i=-b_{i-1}+\tau_i-r_i$, applying Theorem 1.2.1, we obtain

$$x_{-b_{i-1}}^{(i-1)}x_{-b_{i-1}+1}^{(i-1)}\ldots=\lambda_i'(\langle x^{(n)}(-b_{i-1}+\tau_i+\cdots+\tau_n-1,r_i+\cdots+r_n)\rangle,\\ x_{-b_{i-1}+\tau_i+\cdots+\tau_n}^{(n)}x_{-b_{i-1}+\tau_i+\cdots+\tau_n+1}^{(n)}\ldots).$$

Thus the above equation holds for the case of $i=1$, that is,

$$x_{-b_0}^{(0)}x_{-b_0+1}^{(0)}\ldots=\lambda_1'(\langle x^{(n)}(-b_0+\tau_1+\cdots+\tau_n-1,r_1+\cdots+r_n)\rangle,\\ x_{-b_0+\tau_1+\cdots+\tau_n}^{(n)}x_{-b_0+\tau_1+\cdots+\tau_n+1}^{(n)}\ldots).$$

Since (9.5) holds, from Theorem 1.2.1 it immediately follows that

$$y_0y_1\ldots=\lambda'(s',x_\tau^{(n)}x_{\tau+1}^{(n)}\ldots),$$

that is, (9.3) holds. □

Theorem 9.1.2. *Assume that* $PI(M_i, M_i^*, \tau_i)$, $i = 0, 1, \ldots, n$ *hold. Let* $s' = \langle y(-1, t_0), x^{(n)}(-1, r)\rangle$ *be a state of* $C'(M_n, \ldots, M_1, M_0)$ *with output function* λ', *where* $r = r_0 + \cdots + r_n$. *Let*

$$x^{(i-1)}_{-r_0-\cdots-r_{i-1}} \cdots x^{(i-1)}_{-1} = \lambda_i(\langle x^{(i)}(-r_0 - \cdots - r_{i-1} - 1, r_i)\rangle, x^{(i)}_{-r_0-\cdots-r_{i-1}} \ldots x^{(i)}_{-1}) \quad (9.6)$$

for $i = n, n-1, \ldots, 1$, *and*

$$y_0 y_1 \ldots = \lambda'(s', x_0^{(n)} x_1^{(n)} \ldots). \quad (9.7)$$

If

$$x_0^{(0)} x_1^{(0)} \ldots = \lambda_0^*(\langle x^{(0)}(-1, t_0^*), y(\tau_0 - 1, r_0^*)\rangle, y_{\tau_0} y_{\tau_0+1} \ldots), \quad (9.8)$$

and

$$x_0^{(i)} x_1^{(i)} \ldots = \lambda_i^*(\langle x^{(i)}(-1, r_i), x^{(i-1)}(\tau_i - 1, \tau_i)\rangle, x^{(i-1)}_{\tau_i} x^{(i-1)}_{\tau_i+1} \ldots) \quad (9.9)$$

for $i = 1, \ldots, n-1$, *then*

$$x_0^{(n)} x_1^{(n)} \ldots = \lambda_n^*(\langle x^{(n)}(-1, r_n), x^{(n-1)}(\tau_n - 1, \tau_n)\rangle, x^{(n-1)}_{\tau_n} x^{(n-1)}_{\tau_n+1} \ldots). \quad (9.10)$$

Proof. Let $s_i = \langle x^{(i)}(-1, r_i)\rangle$ for $1 \leqslant i \leqslant n$, $s'_n = s'$ and $s'_i = \langle\, y(-1, t_0),\, x^{(i)}(-1, r_0 + \cdots + r_i)\rangle$ for $0 \leqslant i \leqslant n-1$. For any i, $0 < i \leqslant n$, since (9.6) holds, from Theorem 1.2.1, the state s'_i of $C'(M_i, \ldots, M_1, M_0)$ and the state $\langle s_i, s'_{i-1}\rangle$ of $C(M_i, C'(M_{i-1}, \ldots, M_1, M_0))$ are equivalent. Thus, the state $s'(= s'_n)$ of $C'(M_n, \ldots, M_1, M_0)$ and the state $\langle s_n, \ldots, s_1, s'_0\rangle$ of $C(M_n, \ldots, M_1, M_0)$ are equivalent.

Suppose that (9.7) holds. Let

$$\bar{x}_0^{(n-1)} \bar{x}_1^{(n-1)} \ldots = \lambda_n(s_n, x_0^{(n)} x_1^{(n)} \ldots), \quad (9.11)$$

and

$$\bar{x}_0^{(i-1)} \bar{x}_1^{(i-1)} \ldots = \lambda_i(s_i, \bar{x}_0^{(i)} \bar{x}_1^{(i)} \ldots) \quad (9.12)$$

for $i = n-1, n-2, \ldots, 1$. Since s' and $\langle s_n, \ldots, s_1, s'_0\rangle$ are equivalent, (9.7) yields

$$y_0 y_1 \ldots = \lambda_0(s'_0, \bar{x}_0^{(0)} \bar{x}_1^{(0)} \ldots). \quad (9.13)$$

To show $x_0^{(n-1)} x_1^{(n-1)} \ldots = \bar{x}_0^{(n-1)} \bar{x}_1^{(n-1)} \ldots$, we now prove by simple induction that $x_0^{(i)} x_1^{(i)} \ldots = \bar{x}_0^{(i)} \bar{x}_1^{(i)} \ldots$ for $0 \leqslant i \leqslant n-1$. Since $PI(M_0, M_0^*, \tau_0)$ holds, by (9.13) we have

$$\bar{x}_0^{(0)} \bar{x}_1^{(0)} \ldots = \lambda_0^*(\langle x^{(0)}(-1, t_0^*), y(\tau_0 - 1, r_0^*)\rangle, y_{\tau_0} y_{\tau_0+1} \ldots).$$

From (9.8) it follows that $x_0^{(0)} x_1^{(0)} \ldots = \bar{x}_0^{(0)} \bar{x}_1^{(0)} \ldots$ Suppose that $x_0^{(i-1)} x_1^{(i-1)} \ldots = \bar{x}_0^{(i-1)} \bar{x}_1^{(i-1)} \ldots$ is true and $i \leqslant n-1$. We prove that

$x_0^{(i)}x_1^{(i)}\ldots = \bar{x}_0^{(i)}\bar{x}_1^{(i)}\ldots$ Since $PI(M_i, M_i^*, \tau_i)$ holds, noticing $x_j^{(i-1)} = \bar{x}_j^{(i-1)}$, (9.12) yields

$$\bar{x}_0^{(i)}\bar{x}_1^{(i)}\ldots = \lambda_i^*(\langle x^{(i)}(-1, r_i), x^{(i-1)}(\tau_i - 1, \tau_i)\rangle, x_{\tau_i}^{(i-1)}x_{\tau_i+1}^{(i-1)}\ldots).$$

From (9.9), we have $x_0^{(i)}x_1^{(i)}\ldots = \bar{x}_0^{(i)}\bar{x}_1^{(i)}\ldots$

Since $PI(M_n, M_n^*, \tau_n)$ holds, noticing $x_j^{(n-1)} = \bar{x}_j^{(n-1)}$, (9.11) yields

$$x_0^{(n)}x_1^{(n)}\ldots = \lambda_n^*(\langle x^{(n)}(-1, r_n), x^{(n-1)}(\tau_n - 1, \tau_n)\rangle, x_{\tau_n}^{(n-1)}x_{\tau_n+1}^{(n-1)}\ldots),$$

that is, (9.10) holds. □

Corollary 9.1.1. *If condition* (9.6) *is replaced by*

$$x_{-r_0'-r_1-\cdots-r_{i-1}}^{(i-1)}\ldots x_{-1}^{(i-1)} = \lambda_i(\langle x^{(i)}(-r_0' - r_1 - \cdots - r_{i-1} - 1, r_i)\rangle, x_{-r_0'-r_1-\cdots-r_{i-1}}^{(i)}\ldots x_{-1}^{(i)}),$$

where $r_0' = \min(r_0, t_0^*)$, $-r_0' - r_1 - \cdots - r_{i-1}$ *means* $-r_0'$ *in the case of* $i = 1$, *then Theorem* 9.1.2 *still holds.*

Proof. Since M_i, $i = 1, \ldots, n$ are input-memory finite automata, $x_{-1}^{(0)}, \ldots, x_{-t_0^*}^{(0)}$ in (9.8), $x_{-1}^{(i)}, \ldots, x_{-r_i}^{(i)}$, $i = 1, \ldots, n$ in (9.9) and (9.10) are independent of $x_{-r}^{(n)}$, $x_{-r+1}^{(n)}$, …, $x_{-r+r_0-r_0'-1}^{(n)}$. □

9.2 Basic Algorithm

A conventional cryptosystem, namely, a one key cryptosystem, is a family of pairs of encryption algorithms and decryption algorithms, each algorithm in any pair is indexed by its key. The key of an encryption algorithm and the key of its corresponding decryption algorithm are the same, or the latter can be easily derived from the former. Conventional cryptosystems require the sender and the receiver to share a key in secret.

According to Diffie and Hellman[32], a public key cryptosystem is a family of pairs of algorithms, say $\{(E_k, D_k), k \in K\}$, satisfying conditions: (a) for any $k \in K$, D_k is an inverse of E_k (for confidence application), and/or E_k is an inverse of D_k (for authenticity application), (b) for any $k \in K$, E_k and D_k are easy to compute, (c) for almost every $k \in K$, it is infeasible to derive an easily computed algorithm equivalent to D_k from E_k, and (d) for any $k \in K$, it is feasible to compute the pair of E_k and D_k. In applications of a public key cryptosystem, each user chooses a pair of E_k, the user's public key, and D_k, the user's private key, the user makes E_k public and keeps D_k secret.

Based on the results of the preceding section, a public key cryptosystem for both confidence and authenticity applications can be proposed. It is required that $m = l_0 = \cdots = l_n$.[1] We denote X_i by X for short.

Choose a common q and m for all users. Let both the alphabets X and Y be the same column vector space over $GF(q)$ of dimension m. The plaintext space X^* and the ciphertext space Y^* are the same.

A user, say A, choose his/her own public key and private key as follows.

(a) Construct an (r_0^*, t_0^*)-order memory finite automaton $M_0^* = \langle Y, X, S_0^*, \delta_0^*, \lambda_0^* \rangle$ and an (r_0, t_0)-order memory finite automaton $M_0 = \langle X, Y, S_0, \delta_0, \lambda_0 \rangle$ satisfying conditions $PI(M_0^*, M_0, \tau_0)$ and $PI(M_0, M_0^*, \tau_0)$.

(b) For each i, $1 \leqslant i \leqslant n$, construct an r_i-order input-memory finite automaton $M_i = \langle X, X, S_i, \delta_i, \lambda_i \rangle$ and a (τ_i, r_i)-order memory finite automaton $M_i^* = \langle X, X, S_i^*, \delta_i^*, \lambda_i^* \rangle$ satisfying conditions $PI(M_i^*, M_i, \tau_i)$ and $PI(M_i, M_i^*, \tau_i)$.

(c) Construct the finite automaton $C'(M_n, \ldots, M_1, M_0) = \langle X, Y, S, \delta, \lambda \rangle$ from $M_0, \ldots, M_n$.

(d) Let $b_0 = r_0 - \tau_0$, $b_i = b_{i-1} + r_i - \tau_i$, $i = 1, \ldots, n$. Assume that $b_0 = \ldots = b_{c-1} = 0$, i.e., $r_j = \tau_j$, $j = 0, \ldots, c-1$, for some c, $0 \leqslant c \leqslant n$. Choose arbitrary $y_{-1}, \ldots, y_{-t_0} \in Y$, $x_{-1}^{(c)}, \ldots, x_{-b_c}^{(c)} \in X$. For each i, $c+1 \leqslant i \leqslant n$, choose an arbitrary state $s_{-b_{i-1}}^{(i)*} = \langle x_{-b_{i-1}-1}^{(i)}, \ldots, x_{-b_{i-1}-r_i}^{(i)}, \bar{x}_{-b_{i-1}-1}^{(i-1)}, \ldots, \bar{x}_{-b_{i-1}-\tau_i}^{(i-1)} \rangle$ of M_i^*. Compute

$$x_{-b_{i-1}}^{(i)} \ldots x_{-1}^{(i)} = \lambda_i^*(s_{-b_{i-1}}^{(i)*}, x_{-b_{i-1}}^{(i-1)} \ldots x_{-1}^{(i-1)})$$

and

$$s_0^{(i)*} = \delta_i^*(s_{-b_{i-1}}^{(i)*}, x_{-b_{i-1}}^{(i-1)} \ldots x_{-1}^{(i-1)}),$$

for $i = c+1, \ldots, n$. Take $s_s^{(0),in} = \langle y_{-1}, \ldots, y_{-\min(t_0, r_0^*)} \rangle$, $s_s^{(c),out} = \langle x_{-1}^{(c)}, \ldots, x_{-b_c}^{(c)} \rangle$, $s_s^{(i)} = s_0^{(i)*}$, $i = c+1, \ldots, n$, $s_v^{out} = \langle y_{-1}, \ldots, y_{-t_0} \rangle$, $s_v^{in} = \langle x_{-1}^{(n)}, \ldots, x_{-b_n}^{(n)} \rangle$.

(e) Choose arbitrarily $y_{-1}, \ldots, y_{-r_0^*+\tau_0} \in Y$, and $x_{-1}^{(n)}, \ldots, x_{-r'}^{(n)} \in X$, where $r' = r_0' + r_1 + \cdots + r_n$, $r_0' = \min(r_0, t_0^*)$. Compute

$$\begin{aligned} &x_{-r_0'-r_1-\cdots-r_{i-1}}^{(i-1)} \ldots x_{-1}^{(i-1)} \\ &\quad = \lambda_i(\langle x_{-r_0'-r_1-\cdots-r_{i-1}-1}^{(i)}, \ldots, x_{-r_0'-r_1-\cdots-r_i}^{(i)} \rangle, x_{-r_0'-r_1-\cdots-r_{i-1}}^{(i)} \ldots x_{-1}^{(i)}), \\ &\quad i = n, n-1, \ldots, 1. \end{aligned}$$

[1] If the public key system is only for the confidence application, $l_n \leqslant \cdots \leqslant l_0 \leqslant m$ is enough. If it is only for the authenticity application, $l_n \geqslant \cdots \geqslant l_0 \geqslant m$ suffices.

Take $s_e^{out} = \langle y_{-1}, \ldots, y_{-\min(r_0^*-\tau_0, t_0)} \rangle$, $s_e^{in} = \langle x_{-1}^{(n)}, \ldots, x_{-r'}^{(n)} \rangle$, $s_d^{(0),out} = \langle x_{-1}^{(0)}, \ldots, x_{-\min(r_0, t_0^*)}^{(0)} \rangle$, $s_d^{(0),in} = \langle y_{-1}, \ldots, y_{-r_0^*+\tau_0} \rangle$, $s_d^{(i),out} = \langle x_{-1}^{(i)}, \ldots, x_{-r_i}^{(i)} \rangle$, $i = 1, \ldots, n$. [1]

(f) The public key of the user A is

$$C'(M_n, \ldots, M_1, M_0),\ s_v^{out}, s_v^{in},\ s_e^{out}, s_e^{in},\ \tau_0 + \cdots + \tau_n.$$

The private key of the user A is

$$M_0^*, \ldots, M_n^*,\ s_s^{(0),in}, s_s^{(c),out},\ s_s^{(c+1)}, \ldots, s_s^{(n)},$$
$$s_d^{(0),out}, s_d^{(0),in},\ s_d^{(1),out}, \ldots, s_d^{(n),out}, \tau_0, \ldots, \tau_n.$$

Encryption

Any user, say B, wants to send a plaintext $x_0 \ldots x_l$ to a user A. B first suffixes any $\tau_0 + \cdots + \tau_n$ digits, say $x_{l+1} \ldots x_{l+\tau_0+\cdots+\tau_n}$, to the plaintext. Then using A's public key $C'(M_n, \ldots, M_1, M_0)$, $s_e^{out} = \langle y_{-1}, \ldots, y_{-\min(r_0^*-\tau_0, t_0)} \rangle$ and $s_e^{in} = \langle x_{-1}^{(n)}, \ldots, x_{-r'}^{(n)} \rangle$, B computes the ciphertext $y_0 \ldots y_{n+\tau_0+\cdots+\tau_n}$ as follows:

$$y_0 \ldots y_{l+\tau_0+\cdots+\tau_n} = \lambda'(s', x_0 \ldots x_{l+\tau_0+\cdots+\tau_n}),$$

where

$$s' = \langle y_{-1}, \ldots, y_{-t_0}, x_{-1}^{(n)}, \ldots, x_{-r}^{(n)} \rangle,$$

$x_{-r+r_0-t_0^*-1}^{(n)}, \ldots, x_{-r}^{(n)}$ are arbitrarily chosen from X when $t_0^* < r_0$, and $y_{-r_0^*+\tau_0-1}, \ldots, y_{-t_0}$ are arbitrarily chosen from Y when $r_0^* - \tau_0 < t_0$.

Decryption

From the ciphertext $y_0 \ldots y_{l+\tau_0+\cdots+\tau_n}$, A can retrieve the plaintext as follows. Using $M_0^*, \ldots, M_n^*$, $s_d^{(0),out} = \langle x_{-1}^{(0)}, \ldots, x_{-\min(r_0, t_0^*)}^{(0)} \rangle$, $s_d^{(0),in} = \langle y_{-1}, \ldots, y_{-r_0^*+\tau_0} \rangle$, $s_d^{(i),out} = \langle x_{-1}^{(i)}, \ldots, x_{-r_i}^{(i)} \rangle$, $i = 1, \ldots, n$ in his/her private key, A computes

$$x_0^{(0)} x_1^{(0)} \ldots x_{l+\tau_1+\cdots+\tau_n}^{(0)}$$
$$= \lambda_0^*(\langle x_{-1}^{(0)}, \ldots, x_{-t_0^*}^{(0)}, y_{\tau_0-1}, \ldots, y_{\tau_0-r_0^*} \rangle, y_{\tau_0} y_{\tau_0+1} \ldots y_{l+\tau_0+\cdots+\tau_n}),$$
$$x_0^{(i)} x_1^{(i)} \ldots x_{l+\tau_{i+1}+\cdots+\tau_n}^{(i)}$$
$$= \lambda_i^*(\langle x_{-1}^{(i)}, \ldots, x_{-r_i}^{(i)}, x_{\tau_i-1}^{(i-1)}, \ldots, x_0^{(i-1)} \rangle, x_{\tau_i}^{(i-1)} x_{\tau_i+1}^{(i-1)} \ldots x_{l+\tau_i+\cdots+\tau_n}^{(i-1)}),$$
$$i = 1, \ldots, n,$$

where $x_{-r_0-1}^{(0)}, \ldots, x_{-t_0^*}^{(0)}$ may be arbitrarily chosen when $r_0 < t_0^*$. From Corollary 9.1.1, the plaintext $x_0 \ldots x_l$ is equal to $x_0^{(n)} x_1^{(n)} \ldots x_l^{(n)}$.

[1] For the simplicity of symbolization, we use the same symbols y_{-j} and $x_{-j}^{(i)}$ in (d) and in (e), but their intentions are different.

Signature

To sign a message $y_0 \dots y_l$, the user A first suffixes any $\tau_0 + \cdots + \tau_n$ digits, say $y_{l+1} \dots y_{l+\tau_0+\cdots+\tau_n}$, to the message. Then using his/her private key $M_0^*, \dots, M_n^*$, $s_s^{(0),in} = \langle y_{-1}, \dots, y_{-\min(t_0, r_0^*)} \rangle$, $s_s^{(c),out} = \langle x_{-1}^{(c)}, \dots, x_{-b_c}^{(c)} \rangle$, $s_s^{(i)}$, $i = c+1, \dots, n$, A computes

$$x_0^{(0)} x_1^{(0)} \dots x_{l+\tau_0+\cdots+\tau_n}^{(0)} = \lambda_0^*(s_s^{(0)}, y_0 y_1 \dots y_{l+\tau_0+\cdots+\tau_n}),$$
$$x_0^{(i)} x_1^{(i)} \dots x_{l+\tau_0+\cdots+\tau_n}^{(i)} = \lambda_i^*(s_s^{(i)}, x_0^{(i-1)} x_1^{(i-1)} \dots x_{l+\tau_0+\cdots+\tau_n}^{(i-1)}),$$
$$i = 1, \dots, n,$$

where

$$s_s^{(0)} = \langle x_{-1}^{(0)}, \dots, x_{-t_0^*}^{(0)}, y_{-1}, \dots, y_{-r_0^*} \rangle,$$
$$s_s^{(i)} = \langle x_{-1}^{(i)}, \dots, x_{-r_i}^{(i)}, \bar{x}_{-1}^{(i-1)}, \dots, \bar{x}_{-\tau_i}^{(i-1)} \rangle,$$
$$i = 1, \dots, c,$$

$x_{-b_0-1}^{(0)}, \dots, x_{-t_0^*}^{(0)}$ are arbitrarily chosen from X in the case of $c = 0$, $x_{-1}^{(0)}, \dots, x_{-t_0^*}^{(0)}, x_{-1}^{(i)}, \dots, x_{-r_i}^{(i)}$, $i = 1, \dots, c-1$, $x_{-b_c-1}^{(c)}, \dots, x_{-r_c}^{(c)}$, and $\bar{x}_{-1}^{(i-1)}, \dots, \bar{x}_{-\tau_i}^{(i-1)}$, $i = 1, \dots, c$, are arbitrarily chosen from X in the case of $c > 0$, and $y_{-t_0-1}, \dots, y_{-r_0^*}$ are arbitrarily chosen from Y in the case of $t_0 < r_0^*$. Then $x_0^{(n)} x_1^{(n)} \dots x_{l+\tau_0+\cdots+\tau_n}^{(n)}$ is a signature of the message $y_0 \dots y_l$.

Validation

Any user, say B, can verify the validity of the signature $x_0^{(n)} x_1^{(n)} \dots x_{l+\tau_0+\cdots+\tau_n}^{(n)}$ as follows. Using $C'(M_n, \dots, M_1, M_0)$, $s_v^{out} = \langle y_{-1}, \dots, y_{-t_0} \rangle$, $s_v^{in} = \langle x_{-1}^{(n)}, \dots, x_{-r+\tau}^{(n)} \rangle$ in A's public key, B computes

$$\lambda'(s', x_\tau^{(n)} x_{\tau+1}^{(n)} \dots x_{l+\tau}^{(n)}),$$

which would coincide with the message $y_0 \dots y_l$ from Theorem 9.1.1, where $s' = \langle y_{-1}, \dots, y_{-t_0}, x_{\tau-1}^{(n)}, \dots, x_{\tau-r}^{(n)} \rangle$, $r = r_0 + \cdots + r_n$, and $\tau = \tau_0 + \cdots + \tau_n$.

The public key cryptosystem based on finite automata mentioned above is abbreviated to FAPKC. Notice that a plaintext may have many ciphertexts and that a message may have many signatures. In encryption or signing, some digits of the initial state(s) may take arbitrary values. The number of such digits is referred to as the freedom, which depends on the choice of parameters. For example, in FAPKC3, a special case of FAPKC with $n = 1$, $r_0^* = t_0 + \tau_0$, and $t_0^* = r_0$ (cf. [131]), the freedom for signature is $2\tau_0$ if $\tau_0 < r_0$,

$2\tau_0+2\tau_1$ if $\tau_0 = r_0$, and the freedom for encryption is 0. In FAPKC4, a special case of FAPKC with $n = 1$, $r_0 = t_0^* + \tau_0$, and $t_0 = r_0^*$ (cf. [122]), the freedom for signature is 0 if $\tau_0 < r_0$, $2\tau_1$ if $\tau_0 = r_0$, and the freedom for encryption is $2\tau_0$.

We point out that finite automata M_i and M_i^* satisfying conditions $PI(M_i^*, M_i, \tau_i)$ and $PI(M_i, M_i^*, \tau_i)$ do exist. Recall some results in Chap. 3. Taking $p = -1$, let M be an (r, t)-order memory finite automaton M_f defined by

$$y_i = f(y_{i-1}, \ldots, y_{i-t}, x_i, \ldots, x_{i-r}),\ i = 0, 1, \ldots \tag{9.14}$$

Assume that $eq_0(i)$ is the equation

$$-y_i + f(y_{i-1}, \ldots, y_{i-t}, x_i, \ldots, x_{i-r}) = 0 \tag{9.15}$$

and that

$$eq_k(i) \overset{R_a[\varphi_k]}{\longrightarrow} eq'_k(i),\ eq'_k(i) \overset{R_b[r_{k+1}]}{\longrightarrow} eq_{k+1}(i),\quad k = 0, 1, \ldots, \tau - 1 \tag{9.16}$$

is an R_a R_b transformation sequence. Let f_τ^* be a single-valued mapping from $X^r \times Y^{\tau+t+1}$ to X, and $M^* = \langle Y, X, X^r \times Y^{\tau+t}, \delta^*, \lambda^* \rangle$ be a finite automaton $M_{f_\tau^*}$ defined by

$$x_i = f_\tau^*(x_{i-1}, \ldots, x_{i-r}, y'_i, \ldots, y'_{i-\tau-t}),\ i = 0, 1, \ldots$$

If $eq_\tau(i)$ has a solution f_τ^*, i.e., for any parameters x_{i-1}, …, x_{i-r}, $y_{i+\tau}$, …, y_{i-t}, $eq_\tau(i)$ has a solution x_i

$$x_i = f_\tau^*(x_{i-1}, \ldots, x_{i-r}, y_{i+\tau}, \ldots, y_{i-t}),$$

then from Lemma 3.1.1 we have $PI(M^*, M, \tau)$. If $eq_\tau(i)$ has at most one solution f_τ^*, i.e., for any x_i, …, x_{i-r}, $y_{i+\tau}$, …, y_{i-t}, $eq_\tau(i)$ implies $x_i = f_\tau^*(x_{i-1}, \ldots, x_{i-r}, y_{i+\tau}, \ldots, y_{i-t})$, then from Lemma 3.1.2 we have $PI(M, M^*, \tau)$. Thus if $eq_\tau(i)$ determines a single-valued function f_τ^*, then we have $PI(M^*, M, \tau)$ and $PI(M, M^*, \tau)$. Using R_a^{-1} R_b^{-1} transformation, in Sect. 3.2 of Chap. 3 a generation procedure of such a finite automaton M and such an R_a R_b transformation sequence are given. For example, choose an $m \times l(\tau+1)$ (l, τ)-echelon matrix $G_\tau(i)$ over $GF(q)$ and an R_a^{-1} R_b^{-1} transformation sequence

$$G_{k+1}(i) \overset{R_b^{-1}[r_{k+1}]}{\longrightarrow} G'_k(i),\ G'_k(i) \overset{R_a^{-1}[P'_k]}{\longrightarrow} G_k(i),\ k = \tau - 1, \ldots, 1, 0 \tag{9.17}$$

such that for any parameters x_{i-1}, …, $x_{i-\nu}$,

$$G_{0\tau}\psi^l_{-1,\nu}(x_i, \ldots, x_{i-\nu})$$

as a function of the variable x_i is a bijection. Take

$$f(y_{i-1},\ldots,y_{i-t},x_i,\ldots,x_{i-r}) = f'(y_{i-1},\ldots,y_{i-t}) + \sum_{j=0}^{r} G_{j0}\psi^{l}_{-1,\nu}(x_{i-j},\ldots,x_{i-j-\nu}),$$

where f' and G_{j0}, $j=\tau+1,\ldots,r$ are arbitrarily chosen such that the right side of the equation does not depend on $x_{i-r-1},\ldots,x_{i-r-\nu}$. Taking M as M_f and the R_a R_b transformation sequence corresponding to the reverse transformation sequence of (9.17), then the finite automaton and the R_a R_b transformation sequence satisfy the conditions mentioned above.

9.3 An Example of FAPKC

We give a pedagogical example with $q=2$, $m=8$, $n=1$ and $c=0$. Thus the alphabets X and Y are the column vector spaces over $GF(2)$ of dimension 8.

M_0 is an (r_0,t_0)-order nonlinear finite automaton, defined by

$$y_i = \sum_{j=1}^{t_0} A_j y_{i-j} \oplus \sum_{j=0}^{r_0} B_j x'_{i-j} \oplus \sum_{j=0}^{r_0-1} B'_j t'(x'_{i-j},x'_{i-j-1}),$$
$$i=0,1,\ldots,$$

and M_0^* is a nonlinear $(t_0+\tau_0,r_0)$-order memory finite automaton, defined by

$$x'_i = \sum_{j=1}^{r_0} A^*_j x'_{i-j} \oplus \sum_{j=1}^{r_0-1} A^{**}_j t'(x'_{i-j},x'_{i-j-1}) \oplus \sum_{j=0}^{t_0+\tau_0} B^*_j y_{i-j},$$
$$i=0,1,\ldots,$$

where t' is a nonlinear function from X^2 to X,

$$t'(x_0,x_{-1}) = \beta_{3+j},$$
$$j = 2^3 j_3 + 2^2 j_2 + 2j_1 + j_0,$$
$$j_3 = p_8(\beta_{20}\ \&\ x_0),$$
$$j_2 = p_8(\beta_{21}\ \&\ x_0)),$$
$$j_1 = p_8(\beta_{20}\ \&\ x_{-1}),$$
$$j_0 = p_8(\beta_{21}\ \&\ x_{-1}),$$
$$p_8([b_7,b_6,\ldots,b_0]^{\mathrm{T}}) = b_7 \oplus b_6 \oplus \cdots \oplus b_0,$$
$$[a_7,a_6,\ldots,a_0]^{\mathrm{T}}\ \&\ [b_7,b_6,\ldots,b_0]^{\mathrm{T}} = [a_7\ \&\ b_7, a_6\ \&\ b_6,\ldots,a_0\ \&\ b_0]^{\mathrm{T}},$$
$$x_0,x_{-1}\in X,\ a_i,b_i\in GF(2),\ i=0,\ldots,7.$$

M_1 is a nonlinear r_1-order input-memory finite automaton, defined by

$$x'_i = \sum_{j=0}^{r_1} F_j x_{i-j} \oplus \sum_{j=0}^{r_1-2} F'_j t(x_{i-j}, x_{i-j-1}, x_{i-j-2}),$$
$$i = 0, 1, \ldots,$$

and M_1^* is a nonlinear (τ_1, r_1)-order memory finite automaton, defined by

$$x_i = \sum_{j=1}^{r_1} F_j^* x_{i-j} \oplus \sum_{j=1}^{r_1-2} F_j^{**} t(x_{i-j}, x_{i-j-1}, x_{i-j-2}) \oplus \sum_{j=0}^{\tau_1} D_j^* x'_{i-j},$$
$$i = 0, 1, \ldots,$$

where t is a nonlinear function from X^3 to X,

$$t(x_0, x_{-1}, x_{-2}) = \beta_{3+j} \oplus (\beta_{19} \,\&\, \psi(x_0) \,\&\, \psi(x_{-1}) \,\&\, \psi(x_{-2})),$$
$$j = 2^3 j_3 + 2^2 j_2 + 2 j_1 + j_0,$$
$$j_3 = p_8((\beta_0 \,\&\, x_0) \oplus (\beta_2 \,\&\, x_{-1})),$$
$$j_2 = p_8(\beta_1 \,\&\, x_{-1}),$$
$$j_1 = p_8((\beta_0 \,\&\, x_{-1}) \oplus (\beta_2 \,\&\, x_{-2})),$$
$$j_0 = p_8(\beta_1 \,\&\, x_{-2}),$$
$$x_0, x_{-1}, x_{-2} \in X.$$

$\beta_0, \ldots, \beta_{21}$ are in X, and ψ is a single-valued mapping from X to X. In this example, β_0, β_1, ..., β_{21} are `f1`, `a2`, `57`, `00`, `00`, `00`, `00`, `01`, `01`, `01`, `01`, `00`, `20`, `40`, `64`, `80`, `a8`, `d0`, `fe`, `04`, `79`, `39`, respectively, and $\psi([0,0,0,0,0,0,0,0]^{\mathrm{T}})$ $\psi([0,0,0,0,0,0,0,1]^{\mathrm{T}})$ ... $\psi([1,1,1,1,1,1,1,1]^{\mathrm{T}})$ are

```
30 31 33 34 35 36 37 38 05 06 07 08 09 0a 0b 0c
0d 0e 0f 02 00 01 03 04 10 11 13 14 15 16 17 18
19 1a 1b 1c 1d 1e 1f 12 29 2a 2b 2c 20 21 23 24
25 26 27 28 2d 2e 2f 22 39 3a 3b 3c 3d 3e 3f 32
70 71 73 74 75 76 77 78 45 46 47 48 49 4a 4b 4c
4d 4e 4f 42 40 41 43 44 50 51 53 54 55 56 57 58
59 5a 5b 5c 5d 5e 5f 52 69 6a 6b 6c 60 61 63 64
65 66 67 68 6d 6e 6f 62 79 7a 7b 7c 7d 7e 7f 72
b0 b1 b3 b4 b5 b6 b7 b8 85 86 87 88 89 8a 8b 8c
8d 8e 8f 82 80 81 83 84 90 91 93 94 95 96 97 98
99 9a 9b 9c 9d 9e 9f 92 a9 aa ab ac a0 a1 a3 a4
a5 a6 a7 a8 ad ae af a2 b9 ba bb bc bd be bf b2
```

f0 f1 f3 f4 f5 f6 f7 f8 c5 c6 c7 c8 c9 ca cb cc
cd ce cf c2 c0 c1 c3 c4 d0 d1 d3 d4 d5 d6 d7 d8
d9 da db dc dd de df d2 e9 ea eb ec e0 e1 e3 e4
e5 e6 e7 e8 ed ee ef e2 f9 fa fb fc fd fe ff f2

respectively. In this example, to save space, we use symbols 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, a, b, c, d, e, f to denote 0000, 0001, 0010, 0011, 0100, 0101, 0110, 0111, 1000, 1001 1010, 1011 1100, 1101 1110, 1111 in matrices, and to denote column vectors $[0,0,0,0]^{\mathrm{T}}$, $[0,0,0,1]^{\mathrm{T}}$, $[0,0,1,0]^{\mathrm{T}}$, $[0,0,1,1]^{\mathrm{T}}$, $[0,1,0,0]^{\mathrm{T}}$, $[0,1,0,1]^{\mathrm{T}}$, $[0,1,1,0]^{\mathrm{T}}$, $[0,1,1,1]^{\mathrm{T}}$, $[1,0,0,0]^{\mathrm{T}}$, $[1,0,0,1]^{\mathrm{T}}$, $[1,0,1,0]^{\mathrm{T}}$, $[1,0,1,1]^{\mathrm{T}}$, $[1,1,0,0]^{\mathrm{T}}$, $[1,1,0,1]^{\mathrm{T}}$, $[1,1,1,0]^{\mathrm{T}}$, $[1,1,1,1]^{\mathrm{T}}$ in states and sequences (words), respectively. The concatenation of two such symbols, say s_1s_2, denotes $t_1t_2t_3t_4t_5t_6t_7t_8$ when s_1 denotes $t_1t_2t_3t_4$ and s_2 denotes $t_5t_6t_7t_8$, or denotes $[t_1,t_2,t_3,t_4,t_5,t_6,t_7,t_8]^{\mathrm{T}}$ when s_1 denotes $[t_1,t_2,t_3,t_4]^{\mathrm{T}}$ and s_2 denotes $[t_5,t_6,t_7,t_8]^{\mathrm{T}}$, where $t_i \in GF(2)$, $i = 1,2,\ldots,8$.

$C'(M_1,M_0)$ is a nonlinear (r_0+r_1,t_0)-order memory finite automaton.

In this example, take $\tau_0 = 4$, $\tau_1 = 6$, $r_0 = 5$, $t_0 = 3$, $r_1 = 8$, and $C'(M_1,M_0)$ in the public key can be expressed as

$$y_i = \sum_{j=1}^{t_0} A_j y_{i-j} \oplus \sum_{j=0}^{r_0+r_1} C_j x_{i-j} \oplus \sum_{j=0}^{r_0+r_1-2} C'_j t(x_{i-j}, x_{i-j-1}, x_{i-j-2}),$$
$$i = 0,1,\ldots,$$

where

$$[C_0 \ \ldots \ C_{13}] = \begin{bmatrix} 00\ 00\ a2\ 00\ db\ dd\ 21\ 3c\ 91\ d3\ bc\ cd\ 7d\ 69 \\ 00\ 00\ a2\ 00\ 92\ 6a\ 42\ 8a\ a6\ d8\ 6b\ 47\ 9e\ 25 \\ 00\ f1\ a6\ 1e\ 5c\ 52\ 39\ 1e\ bd\ ef\ 3c\ 85\ 18\ 01 \\ 00\ 00\ 00\ f1\ 6c\ e0\ d8\ b7\ ab\ d1\ c3\ 8a\ 2f\ ff \\ 00\ f1\ 57\ 00\ d0\ 73\ 76\ ba\ 05\ 2e\ d6\ cb\ 52\ 41 \\ 00\ 00\ 00\ a2\ 42\ ed\ f8\ c9\ 00\ 70\ 7f\ c5\ 26\ 25 \\ 00\ 00\ 53\ 57\ 1a\ c1\ 97\ 99\ b1\ 26\ 3b\ a2\ a6\ 90 \\ 00\ 00\ a2\ 00\ 68\ 6c\ e0\ 30\ 85\ 0b\ dc\ f2\ 51\ b3 \end{bmatrix},$$

$$[C'_0 \ \ldots \ C'_{11}] = \begin{bmatrix} 00\ b8\ b2\ 10\ c1\ b8\ f3\ 6a\ a6\ 38\ 9e\ 67 \\ 00\ b8\ b2\ 69\ d0\ 65\ 22\ 25\ 1b\ 5f\ 90\ 64 \\ 00\ 81\ ca\ 23\ 2c\ 0c\ e2\ b9\ 88\ 97\ 39\ 4d \\ 00\ 39\ 8b\ a0\ b2\ 77\ a7\ b8\ 9b\ 89\ d2\ d4 \\ 00\ b8\ 38\ 8a\ 19\ aa\ 8b\ 97\ 7e\ b4\ 41\ e6 \\ 00\ 00\ 00\ 90\ aa\ 4e\ fe\ 72\ 38\ 5e\ bc\ 64 \\ 00\ 81\ b8\ 78\ d1\ dd\ fa\ 6e\ 98\ eb\ 25\ ad \\ 00\ b8\ 33\ b9\ 70\ 77\ 9e\ 52\ e4\ fd\ 15\ d7 \end{bmatrix},$$

$$[A_1\ A_2\ A_3] = \begin{bmatrix} \texttt{00 43 80} \\ \texttt{00 21 c0} \\ \texttt{00 10 e0} \\ \texttt{00 08 70} \\ \texttt{00 04 38} \\ \texttt{00 02 1c} \\ \texttt{00 01 0e} \\ \texttt{00 00 87} \end{bmatrix}.$$

In the public key,

$$s_v^{out} = s_e^{out} = \langle \texttt{74}, \texttt{d1}, \texttt{4a} \rangle,$$
$$s_v^{in} = \langle \texttt{17}, \texttt{06}, \texttt{d2} \rangle,$$
$$s_e^{in} = \langle \texttt{17}, \texttt{06}, \texttt{d2}, \texttt{ef}, \texttt{52}, \texttt{a5}, \texttt{0c}, \texttt{de}, \texttt{58}, \texttt{37}, \texttt{9b}, \texttt{80}, \texttt{4d} \rangle.$$

In the private key, M_0^* is a nonlinear $(7, 5)$-order memory finite automaton, defined by

$$x_i' = \sum_{j=1}^{5} A_j^* x_{i-j}' \oplus \sum_{j=1}^{4} A_j^{**} t'(x_{i-j}', x_{i-j-1}'), \oplus \sum_{j=0}^{7} B_j^* y_{i-j},$$
$$i = 0, 1, \ldots,$$

where

$$[B_0^*\ \ldots\ B_7^*] = \begin{bmatrix} \texttt{00 c2 df 82 23 ef 27 00} \\ \texttt{c0 29 2b 1d 7a 5a 7f 00} \\ \texttt{c0 eb 62 c2 11 78 4b 00} \\ \texttt{00 00 d5 14 68 a1 6c 00} \\ \texttt{c0 6e f4 ca 0b bf 11 7f} \\ \texttt{00 47 49 8a 70 28 58 00} \\ \texttt{00 47 0a 9e 50 6b 34 00} \\ \texttt{00 00 00 a8 00 57 58 00} \end{bmatrix},$$

$$[A_1^*\ \ldots\ A_5^*] = \begin{bmatrix} \texttt{41 5e 1a 6e 00} \\ \texttt{89 80 1d 81 00} \\ \texttt{8a 92 52 7a 00} \\ \texttt{2f 67 09 14 00} \\ \texttt{01 f0 4e 67 81} \\ \texttt{8f 3d e1 ef 00} \\ \texttt{da 2d 8f fb 00} \\ \texttt{58 86 60 ef 00} \end{bmatrix},$$

$$[A_1^{**} \ \dots \ A_4^{**}] = \begin{bmatrix} \texttt{ba d3 11 00} \\ \texttt{7b 62 38 00} \\ \texttt{ab d3 09 00} \\ \texttt{7b c8 18 00} \\ \texttt{68 38 6b 38} \\ \texttt{40 81 29 00} \\ \texttt{63 41 31 00} \\ \texttt{40 b9 29 00} \end{bmatrix}.$$

M_1^* is a nonlinear $(6,8)$-order, $(6,6)$-order in essential, memory finite automaton, defined by

$$x_i = \sum_{j=1}^{6} F_j^* x_{i-j} \oplus \sum_{j=1}^{4} F_j^{**} t(x_{i-j}, x_{i-j-1}, x_{i-j-2}) \oplus \sum_{j=0}^{6} D_j^* x'_{i-j},$$
$$i = 0, 1, \dots,$$

where

$$[D_0^* \ \dots \ D_6^*] = \begin{bmatrix} \texttt{c2 eb 04 59 59 40 79} \\ \texttt{c2 58 59 42 1b 00 00} \\ \texttt{00 b3 5d 1b 22 00 00} \\ \texttt{00 b3 42 59 22 39 00} \\ \texttt{c2 eb 1b 1b 42 39 00} \\ \texttt{00 b3 eb 00 39 79 79} \\ \texttt{c2 58 59 42 7b 79 79} \\ \texttt{00 b3 42 59 42 79 00} \end{bmatrix},$$

$$[F_1^* \ \dots \ F_6^*] = \begin{bmatrix} \texttt{26 b5 6e 99 e1 be} \\ \texttt{23 b6 ff 29 aa 4b} \\ \texttt{bb 0b 4a 6d b6 4b} \\ \texttt{00 26 05 3d 43 4b} \\ \texttt{df 34 a4 5e e9 f5} \\ \texttt{ea 69 2c 90 00 00} \\ \texttt{9d be 24 f4 57 f5} \\ \texttt{e9 2e 7c 42 1c f5} \end{bmatrix},$$

$$[F_1^{**} \ \dots \ F_4^{**}] = \begin{bmatrix} \texttt{b9 ba 96 32} \\ \texttt{2f 28 0d d1} \\ \texttt{d8 6c be 50} \\ \texttt{bd 96 dc 50} \\ \texttt{01 91 1a 62} \\ \texttt{25 2c 00 00} \\ \texttt{61 d6 28 62} \\ \texttt{f3 e9 78 62} \end{bmatrix}.$$

In the private key,

$$s_s^{(0),in} = \langle \texttt{74}, \texttt{d1}, \texttt{4a} \rangle,$$
$$s_s^{(0),out} = \langle \texttt{69} \rangle,$$
$$s_s^{(1)} = \langle \texttt{17}, \texttt{06}, \texttt{d2}, \texttt{70}, \texttt{7f}, \texttt{a1}, \texttt{65}, \texttt{75}; \texttt{69}, \texttt{ec}, \texttt{bf}, \texttt{cc}, \texttt{c9}, \texttt{3e} \rangle,$$
$$s_d^{(0),in} = \langle \texttt{74}, \texttt{d1}, \texttt{4a} \rangle,$$
$$s_d^{(0),out} = \langle \texttt{59}, \texttt{c2}, \texttt{37}, \texttt{81}, \texttt{b6} \rangle,$$
$$s_d^{(1),out} = \langle \texttt{17}, \texttt{06}, \texttt{d2}, \texttt{ef}, \texttt{52}, \texttt{a5}, \texttt{0c}, \texttt{de} \rangle.$$

Let α be the sequence

$$\texttt{4e 6f 20 70 61 69 6e 73 2c 6e 6f 20 67 61 69 6e 73 2e}$$

over X, which is the ASCII code of the sentence "No pains,no gains.".

For encryption, taking randomly $\alpha_{10} = \texttt{89 b4 70 2a 92 07 ce cd 2a 4c}$, then

$$\lambda(s_e, \alpha\alpha_{10}) = \texttt{6f df a8 59 94 99 80 d7 91 d8 7f 65 ff 39 6d}$$
$$\texttt{a6 36 fe a6 7b 8b fc 08 78 03 75 13 e5}$$

is a ciphertext of α, where λ is the output function of $C'(M_1, M_0)$, and $s_e = \langle s_e^{out}, s_e^{in} \rangle = \langle \texttt{74}, \texttt{d1}, \texttt{4a}; \texttt{17}, \texttt{06}, \texttt{d2}, \texttt{ef}, \texttt{52}, \texttt{a5}, \texttt{0c}, \texttt{de}, \texttt{58}, \texttt{37}, \texttt{9b}, \texttt{80}, \texttt{4d} \rangle$.

For decryption, first compute

$$\lambda_0(s_d^{(0)}, \beta_1) = \texttt{14 d9 c2 26 af 38 fe 74 c2 2b 3b 74 1f 49 a0}$$
$$\texttt{69 c0 d7 15 43 58 a4 6a 55},$$

where $\beta_1 = \texttt{94 99 80 d7 91 d8 7f 65 ff 39 6d a6 36 fe a6 7b 8b fc 08 78 03 75 13 e5}$, $s_d^{(0)} = \langle \texttt{59}, \texttt{c2}, \texttt{37}, \texttt{81}, \texttt{b6}; \texttt{59}, \texttt{a8}, \texttt{df}, \texttt{6f}, \texttt{74}, \texttt{d1}, \texttt{4a} \rangle$. Then compute

$$\lambda_1(s_d^{(1)}, \beta_2) = \texttt{4e 6f 20 70 61 69 6e 73 2c 6e 6f 20 67 61 69 6e 73 2e}$$

which is equal to the plaintext α, where $\beta_2 = \texttt{fe 74 c2 2b 3b 74 1f 49 a0 69 c0 d7 15 43 58 a4 6a 55}$, $s_d^{(1)} = \langle \texttt{17}, \texttt{06}, \texttt{d2}, \texttt{ef}, \texttt{52}, \texttt{a5}, \texttt{0c}, \texttt{de}; \texttt{38}, \texttt{af}, \texttt{26}, \texttt{c2}, \texttt{d9}, \texttt{14} \rangle$.

For signing, taking randomly $\texttt{c9}, \texttt{c8}, \texttt{02}, \texttt{95}, \texttt{2e}, \texttt{76}, \texttt{b0}, \texttt{8d}$ and $\alpha'_{10} = \texttt{2d 49 df fb 14 69 63 d7 e6 8d}$, first compute

$$\lambda_0(s_s^{(0)}, \alpha\alpha'_{10}) = \texttt{35 2e 1a 75 74 92 9e 1c b0 14 4c a4 b0 3c 60}$$
$$\texttt{02 25 7d 9b 70 fb 62 11 88 c2 ec 76 b3}$$

where $s_s^{(0)} = \langle \texttt{69}, \texttt{c9}, \texttt{c8}, \texttt{02}, \texttt{95}; \texttt{74}, \texttt{d1}, \texttt{4a}, \texttt{2e}, \texttt{76}, \texttt{b0}, \texttt{8d} \rangle$. Then compute

$$\lambda_1(s_s^{(1)}, \lambda_0(s_s^{(0)}, \alpha\alpha'_{10})) = \texttt{33 bb bc 7b 95 95 87 2a 9d ec 1e 54 7a 18 fb}$$
$$\texttt{31 1f 4c 9c d8 4d a1 82 17 7a ce 25 2b}$$

which is a digital signature of α.

For verifying, compute

$$\lambda(s_v, \beta_3) = \texttt{4e 6f 20 70 61 69 6e 73 2c 6e 6f 20 67 61 69 6e 73 2e}$$

which is equal to α, where

$$\beta_3 = \texttt{1e 54 7a 18 fb 31 1f 4c 9c d8 4d a1 82 17 7a ce 25 2b},$$
$$s_v = \langle \texttt{74}, \texttt{d1}, \texttt{4a}; \texttt{ec}, \texttt{9d}, \texttt{2a}, 87, 95, 95, \texttt{7b}, \texttt{bc}, \texttt{bb}, 33, 17, 06, \texttt{d2} \rangle.$$

9.4 On Weak Keys

9.4.1 Linear R_a R_b Transformation Test

Notice that $C'(M_n, \ldots, M_0)$ in the public key of FAPKC is an $(r_0+\cdots+r_n, t_0)$-order memory finite automaton. Let $C'(M_n, \ldots, M_0)$ be M_f defined by (9.14), where $r = r_0+\cdots+r_n$, and $t = t_0$. Let (9.16) be a linear R_a R_b transformation sequence, where $\tau = \tau_0 + \cdots + \tau_n$, and $eq_0(i)$ is defined by (9.15). If for any parameters x_{i-1}, ..., x_{i-r}, $y_{i+\tau}$, ..., y_{i-t}, the equation $eq_\tau(i)$ has a (or at most one) solution x_i, then from results in Sect. 3.1 of Chap. 3 a $(\tau+t, r)$-order memory finite automaton $\bar{M}$ can be feasibly constructed from $eq_\tau(i)$ such that $C'(M_n, \ldots, M_0)$ is a weak inverse (or an original weak inverse) with delay τ of $\bar{M}$. Therefore, a check process should be included in a key-generator of FAPKC to sieve out such a $C'(M_n, \ldots, M_0)$, of which an original weak inverse (or a weak inverse) can be obtained by linear R_a R_b transformation method. If there is a linear R_a R_b transformation satisfying the condition mentioned above, then $C'(M_n, \ldots, M_0)$ is sieved out. Although the number of those linear R_a R_b transformations is huge, only one linear R_a R_b transformation sequence is enough to check due to the following results in Sect. 4.1 of Chap. 4: if for a linear R_a R_b transformation sequence (9.16) the condition holds, then for any linear R_a R_b transformation sequence (9.16) the condition holds too.

The key of FAPKC of which $C'(M_n, \ldots, M_0)$ is sieved out by linear R_a R_b transformation method is called a *weak key*. For $n = 1$, from results in Sect. 4.2 of Chap. 4, the following cases are weak key: linear M_0, 0 step delay M_1; linear M_0, quasi-linear M_1; linear M_0, nonlinear M_1 of which a weak inverse can be obtained by linear R_a R_b transformation method. The latter is a more general case.

9.4.2 On Attack by Reduced Echelon Matrix

In Sect. 4.3 of Chap. 4, a method by reduced echelon matrix to construct a weak inverse of a finite automaton is discussed, and it is shown that if

some inversion method by reduced echelon matrix based on injectiveness or surjectiveness of $D_{22}\psi_\nu^l(x(i,\nu{+}1))$ is applicable to a finite automaton M, so is the linear R_a R_b transformation method, where $M = \langle X, Y, Y^k \times X^{h+\nu}, \delta, \lambda\rangle$ is defined by

$$\begin{aligned} y_i &= \varphi_{out}(y(i-1,k)) + [C_0,\ldots,C_h]\psi_\nu^{lh}(x,i), \\ &\quad i = 0,1,\ldots, \end{aligned} \tag{9.18}$$

D_{22} is described in Theorem 4.3.1. From this result, it is not necessary to include a check process based inversion method by reduced echelon matrix in a key-generator of FAPKC. Only weak keys of FAPKC can be broken using the method by reduced echelon matrix!

9.4.3 On Attack by Canonical Diagonal Matrix Polynomial

Let $M = \langle X, Y, Y^k \times X^{h+\nu}, \delta, \lambda\rangle$ be a finite automaton defined by (9.18). Let

$$C(z) = \sum_{j=0}^{h} C_j z^j.$$

Formally, we use $z^j\psi_\nu^l(x_i,\ldots,x_{i-\nu})$ to denote $\psi_\nu^l(x_{i-j},\ldots,x_{i-j-\nu})$, and $z^j x'_i$ to denote x'_{i-j}. Then

$$\begin{aligned} [C_0,\ldots,C_h]\psi_\nu^{lh}(x,i) &= \sum_{j=0}^{h} C_j\psi_\nu^l(x_{i-j},\ldots,x_{i-j-\nu}) \\ &= \sum_{j=0}^{h} C_j(z^j\psi_\nu^l(x_i,\ldots,x_{i-\nu})) = C(z)\psi_\nu^l(x_i,\ldots,x_{i-\nu}). \end{aligned}$$

Suppose that $C(z) = D(z)F(z)$. Let M_1 be a finite automaton defined by

$$\begin{aligned} x'_i &= F(z)\psi_\nu^l(x_i,\ldots,x_{i-\nu}), \\ &\quad i = 0,1,\ldots, \end{aligned}$$

and M_0 a finite automaton defined by

$$\begin{aligned} y_i &= \varphi_{out}(y(i-1,k)) + D(z)x'_i, \\ &\quad i = 0,1,\ldots \end{aligned}$$

It is easy to see that $M = C'(M_1, M_0)$. Since M_0 is quasi-linear, its weak inverse can be easily constructed whenever it exists. Thus there is a feasible inversion method for M whenever there is a feasible inversion method for M_1. Therefore, if for any parameters x_{-1}, …, $x_{-\nu}$, $F(0)\psi_\nu^l(x_0,\ldots,x_{-\nu})$ as a

function of the variable x_0 is an injection, then M_1 is weakly invertible with delay 0 and a weak inverse with delay 0 of M_1 can be feasibly constructed.

It is well known that the process of reducing canonical diagonal form for matrix polynomials is feasible. We can feasibly find $0 \leqslant a_1 \leqslant \cdots \leqslant a_r$, $f_j(z)$, $j = 1, \ldots, r$, and two invertible matrix polynomials $P(z)$ and $Q(z)$ such that

$$C(z) = P(z)DIA_{m,m}(z^{a_1}f_1(z), \ldots, z^{a_r}f_r(z), 0, \ldots, 0)Q(z),$$

$f_j(z) \mid f_{j+1}(z)$ for $j = 1, \ldots, r-1$ and $f_j(0) \neq 0$ for $j = 1, \ldots, r$. Take $D(z) = P(z)DIA_{m,r}(z^{a_1}, \ldots, z^{a_r})$ and $F(z) = DIA_{r,m}(f_1(z), \ldots, f_r(z))Q(z)$. Then $C(z) = D(z)F(z)$ is referred to as the derived factorization of type 2 by canonical diagonal matrix polynomial. Similar, $C(z) = D'(z)F'(z)$ is referred to as the derived factorization of type 1 by canonical diagonal matrix polynomial, where $D'(z) = P(z)DIA_{m,r}(z^{a_1}f_1(z), \ldots, z^{a_r}f_r(z))$ and $F(z) = DIA_{r,m}(1, \ldots, 1)Q(z)$. If $C(z) = D(z)F(z)$ is a derived factorization of type 1 or 2 by canonical diagonal matrix polynomial and $F(0)\psi_\nu^l(x_0, \ldots, x_{-\nu})$ as a function of the variable x_0 is an injection for any parameters $x_{-1}, \ldots, x_{-\nu}$, then a weak inverse of M can be feasibly constructed as mentioned in the preceding paragraph. On the other hand, from Theorem 4.4.5, there exists a terminating and elementary R_a R_b transformation sequence

$$C_k(z) \overset{R_a[P_k]}{\longrightarrow} C'_k(z),\ C'_k(z) \overset{R_b[r_{k+1}]}{\longrightarrow} C_{k+1}(z), \quad k = 0, 1, \ldots, \tau' - 1,$$

where $C_0(z) = C(z)$. Let $\bar{Q}(z)$ be the submatrix of the first r rows of $C_{\tau'}(z)$. From Corollaries 4.4.4 and 4.4.5, if $F(0)\psi_\nu^l(x_0, \ldots, x_{-\nu})$ as a function of the variable x_0 is an injection for any parameters $x_{-1}, \ldots, x_{-\nu}$, then $\bar{Q}(0)\psi_\nu^l(x_0, \ldots, x_{-\nu})$ as a function of the variable x_0 is an injection for any parameters $x_{-1}, \ldots, x_{-\nu}$. It follows that a weak inverse of M can be feasibly constructed by linear R_a R_b transformation method. We conclude that keys of FAPKC which can be broken by canonical diagonal matrix polynomial method are weak keys. Thus it is not necessary to include a check process based on reducing canonical diagonal matrix polynomial in a key-generator of FAPKC.

9.5 Security

Since no theory exists to prove whether a public key system is secure or not, the only approach is to evaluate all ways to break it that one can think. We first consider ways that a cryptanalyst tries to deduce the private key from the public key, and then ways of deducing the plaintext (respectively signature) from the ciphertext (respectively message).

9.5.1 Inversion by a General Method

If one can find a finite automaton M^* which is a weak inverse of $C'(M_n, \ldots, M_0)$ with delay $\tau_0+\cdots+\tau_n$, then one can retrieve plaintexts from ciphertexts. For general nonlinear finite automata, the proof of Theorem 1.4.4 provides an inversion algorithm which requires computing each output for each state and for each input of length $\tau_0 + \cdots + \tau_n + 1$. For a state, the computing amount is $O(q^{m(\tau_0+\cdots+\tau_n)})$ ($O(2^{160})$ for $n = 1$, $q = 2$, $m = 8$, $\tau_0 + \tau_1 = 20$). Therefore, this method is impractical for moderate $\tau_0 + \cdots + \tau_n$.

Similarly, if one can find a finite automaton M^* of which $C'(M_n, \ldots, M_0)$ is a weak inverse with delay $\tau_0 + \cdots + \tau_n$, then one can forge signatures for messages. For general nonlinear finite automata, in Sect. 6.5 of Chap. 6 an inversion algorithm is provided, which requires computing an input-tree with level $\tau_0+\cdots+\tau_n$ for each state and each output of length $\tau_0+\cdots+\tau_n+1$. So this method spends more computing time and storage amount than the method mentioned in the preceding paragraph. We prefer to construct an original weak inverse of a finite automaton by means of constructing a weak inverse of the finite automaton based on mutual invertibility, see Theorem 2.2.1.

9.5.2 Inversion by Decomposing Finite Automata

From the finite automaton $C'(M_n, \ldots, M_0)$ in a public key of FAPKC, if one can feasibly find finite automata $\bar{M}_n$, ..., $\bar{M}_0$ so that $C'(M_n, \ldots, M_0) = C'(\bar{M}_n, \ldots, \bar{M}_0)$ and a weak inverse finite automaton of $\bar{M}_i$ can be feasibly constructed for each j, $0 \leqslant j \leqslant n$, then a weak inverse finite automaton of $C'(M_n, \ldots, M_0)$ can be feasibly constructed. No feasible decomposition method is known.

In some special cases, for example, in the example of FAPKC mentioned in Sect. 9.3, if M_0 is linear (i.e., $B'_j = 0$ for $j = 0, 1, \ldots, r_0 - 1$), then decomposing $C'(M_1, M_0)$ is reduced to factorizing the matrix polynomial

$$\bar{C}(z) = \sum_{j=0}^{r_0+r_1} [\, C_j \,,\, C'_j \,]\, z^j,$$

where $C'_{r_0+r_1-1} = C'_{r_0+r_1} = 0$. If $\bar{C}(z) = \bar{B}(z)\bar{F}(z)$, then $C'(M_1, M_0) = C'(\bar{M}_1, \bar{M}_0)$, where $\bar{M}_0$ is defined by

$$y_i = \sum_{j=1}^{t_0} A_j y_{i-j} \oplus \sum_{j=0}^{\bar{r}_0} \bar{B}_j x'_{i-j},$$
$$i = 0, 1, \ldots,$$

$\bar{M}_1$ is defined by

$$x'_i = \sum_{j=0}^{\bar{r}_1} \bar{F}_j x_{i-j} \oplus \sum_{j=0}^{\bar{r}_1} \bar{F}'_j t(x_{i-j}, x_{i-j-1}, x_{i-j-2}),$$
$$i = 0, 1, \ldots,$$

$\bar{B}(z) = \sum_{j=0}^{\bar{r}_0} \bar{B}_j z^j$, and $\bar{F}(z) = \sum_{j=0}^{\bar{r}_1} [\bar{F}_j, \bar{F}'_j] z^j$. Thus a weak inverse finite automaton of $C'(M_1, M_0)$ can be feasibly found whenever a weak inverse finite automaton of $\bar{M}_1$ can be feasibly constructed.

Although polynomial time algorithms for factorization of polynomials over $GF(q)$ are existent, no feasible algorithm is known for factorizing matrix polynomials over $GF(q)$. Some specific factorizing algorithms of matrix polynomials over $GF(q)$ are known, such as factorization by linear R_a R_b transformation and factorization by reducing canonical diagonal form for matrix polynomials. But those specific factorizations never lead to breaking the key except the weak key.

Let F be a finite field. $H(z)$ in $M_{m,n}(F[z])$ is said to be *linearly primitive*, if for any $H_1(z)$ in $M_{m,m}(F[z])$ with rank m and any $H_2(z)$ in $M_{m,n}(F[z])$, $H(z) = H_1(z)H_2(z)$ implies $H_1(z) \in GL_m(F[z])$.

$H(z)$ in $M_{m,n}(F[z])$ is said to be *left-primitive*, if for any positive integer r, any $H_1(z)$ in $M_{m,r}(F[z])$ with rank r and any $H_2(z)$ in $M_{r,n}(F[z])$, $H(z) = H_1(z)H_2(z)$ implies that the rank of $H_1(0)$ is r.

Two factorizations $A(z) = G(z)H(z)$ and $A(z) = G'(z)H'(z)$ are *equivalent*, if there is an invertible matrix polynomial $R(z)$ such that $G'(z) = G(z)R(z)^{-1}$ and $H'(z) = R(z)H(z)$.

It is known that the linearly primitive factorizations of $A(z)$ and the derived factorizations of type 1 by canonical diagonal matrix polynomial of $A(z)$ are coincided with each other and unique under equivalence and that the left-primitive factorizations of $A(z)$, the derived factorizations by linear R_a R_b transformations of $A(z)$ and the derived factorizations of type 2 by canonical diagonal matrix polynomial of $A(z)$ are coincided with each other and unique under equivalence. No other algorithm to give linearly primitive factorization or left-primitive factorization is known except ones by reducing canonical diagonal form and by linear R_a R_b transformation.

9.5.3 Chosen Plaintext Attack

A chosen plaintext attack for FAPKC is reduced to the problem of solving a nonlinear system of equations over $GF(q)$. We explain the claim for the example in Sect. 9.3.

Since the public key is available for anyone, one can encrypt any plaintext $x_0x_1 \ldots x_n$ and obtain corresponding ciphertext $y_0y_1 \ldots y_{n+\tau_0+\tau_1}$ using $C'(M_1, M_0)$ and its initial state s_e. Suppose that s_e is equivalent to the state

$\langle s_1, s_0\rangle$ of $C(M_1, M_0)$. Let $x'_0x'_1 \dots x'_n$ be the output of M_1 for the input $x_0x_1 \dots x_n$ on the state s_1. Then we have

$$x'_0 \dots x'_{n-\tau_0} = \lambda_0^*(\langle x'_{-1}, \dots, x'_{-r_0}, y_{\tau_0-1}, \dots, y_{-t_0}\rangle, y_{\tau_0} \dots y_n)$$

for some $x'_{-1}, \dots, x'_{-r_0}$ and $y_{-1}, \dots, y_{-t_0}$. It follows that

$$\begin{aligned}
&\sum_{j=0}^{r_1} F_j x_{i-j} \oplus \sum_{j=0}^{r_1-2} F'_j t(x_{i-j}, x_{i-j-1}, x_{i-j-2}) \\
&\quad = \sum_{j=1}^{r_0} A_j^* x'_{i-j} \oplus \sum_{j=1}^{r_0-1} A_j^{**} t'(x'_{i-j}, x'_{i-j-1}) \oplus \sum_{j=0}^{t_0+\tau_0} B_j^* y_{i+\tau_0-j}, \quad (9.19) \\
&\quad i = 0, 1, \dots, n-\tau_0,
\end{aligned}$$

where $s_1 = \langle x_{-1}, \dots, x_{-r_1}\rangle$. In (9.19), values of x_i and y_i are known, and unknown variables are F_j's, F'_j's, A_j^*'s, A_j^{**}'s, B_j^*'s, and x'_j's. In the case of $r_0 > 0$, (9.19) is nonlinear in essential. Finding out M_1 and M_0^* by solving the system of equations seems difficult, even if M_0^* is linear.

9.5.4 Exhausting Search and Stochastic Search

Exhausting Search Attack

Since the encryption algorithm is known for anyone, one may guess possible plaintexts and can encrypt them. When the result of encrypting some guessed plaintext coincides with the ciphertext, the guessed plaintext is the virtual plaintext. Notice that the public key cryptosystem based finite automata is sequential. Its block length m is small in order to provide a small key size. But small block length causes the cryptosystem fragile for the divide and conquer attack. In fact, the guess process can be reduced to guessing a piece of plaintext of length $\tau_0 + \cdots + \tau_n + 1$ and deciding its first digit. That is, guess a value of the first $\tau_0 + \cdots + \tau_n + 1$ digits of the plaintext first, and then encrypt it using the public key and compare the result with the first $\tau_0 + \cdots + \tau_n + 1$ digits of the virtual ciphertext. If they coincide, then the first digit of the guessed plaintext is indeed the first digit of the virtual plaintext. Repeat this process for guessing next digit of the plaintext, and so on. In Sect. 2.3 of Chap. 2, Algorithm 1 is such an exhausting search algorithm for encryption, where M is the finite automaton $C'(M_n, \dots, M_1, M_0)$ in a user's public key, and s is a state of M of which part of components is given by s_e^{out} and s_e^{in} in the public key. In the case of the example in Sect. 9.3, s is $\langle s_e^{out}, s_e^{in}\rangle$.

Formulae of the search amounts in average case and in worse case are deduced in Sect. 2.3 of Chap. 2 for a finite automaton M, which is equivalent

to $C(\bar{M}_0, D_{X,r_1}, \bar{M}_1, D_{X,r_2}, \bar{M}_2, \ldots, \bar{M}_{\tau-1}, D_{X,r_\tau}, \bar{M}_\tau)$, where $0 \leqslant r_1 \leqslant r_2 \leqslant \cdots \leqslant r_\tau \leqslant m$, $\bar{M}_i$ is a weakly invertible finite automaton with delay 0 for $i = 0, 1, \ldots, \tau$, and $\bar{M}_i$ is $(m - r_{i+1})$-preservable, $i = 1, \ldots, \tau - 1$. We point out that a finite automaton $M = C'(M'_1, M'_0)$ satisfies the above condition, where M'_0 is a weakly invertible finite automaton with delay 0, and M'_1 is generated by linear R_a R_b transformation method. A formula of the lower bound for the search amounts in average case is also given there, that is,

$$(l + 1 - \tau)\big(1 + \sum_{j=2}^{\tau} q^{r_j + \cdots + r_\tau}\big)/2.$$

According to the formula, in the case of the example, $(l+1-\tau)(2^{r_2+\cdots+r_\tau-1} + 2^{r_3+\cdots+r_\tau-1})$ is a lower bound for the search amounts in average case, taking $l = \tau = 15$, which is equal to $2^{58} + 2^{56}$ whenever $r_1, \ldots, r_{15}$ are 1, 2, 2, 3, 3, 3, 4, 4, 4, 5, 5, 5, 6, 6, 7, respectively, or $2^{67} + 2^{65}$ whenever $r_1, \ldots, r_{15}$ are 1, 2, 3, 4, 4, 4, 5, 5, 5, 5, 6, 6, 6, 6, 7, respectively.

Similarly, to forge a signature of a given message, one may guess a possible signature first, and then computes using the public key for verification and checks whether the computing result coincides with the message or not. Repeat this process until a coincident one is met. An attacker may adopt an exhausting search attack to forge a signature for a given message using a search algorithm like the following.

Algorithm 3

Input : a message $y_0 y_1 \ldots y_l$.

Output : the signature $x_0 x_1 \ldots x_{\tau+l}$ which satisfies

$$y_0 y_1 \ldots y_l = \lambda(s_{x_0,\ldots,x_{\tau-1}}, x_\tau \ldots x_{\tau+l}),$$

where $s_{x_0,\ldots,x_{\tau-1}}$ is a state $\langle y_{-1}, \ldots, y_{-t_0}, x_{\tau-1}, \ldots, x_{\tau-r}\rangle$ determined by $s_v^{out} = \langle y_{-1}, \ldots, y_{-t_0}\rangle$ and $s_v^{in} = \langle x_{-1}, \ldots, x_{-r+\tau}\rangle$ in the public key and $x_0, \ldots, x_{\tau-1}$.

Procedure :

1. *Guess the prefix of length τ of the signature.*

1.1. Take $X_\tau = X^\tau$.

1.2. If $X_\tau \neq \varnothing$, then choose an element in it as $x'_0, \ldots, x'_{\tau-1}$, delete this element from it and go to Step 2.1; otherwise (impossibly occurs) stop.

2. *Guess the main part of the signature.*

2.1. Set $i = 0$ and $s' = s_{x'_0,\ldots,x'_{\tau-1}}$.

2.2. Set $X_{s',x'_\tau x'_{\tau+1}\ldots x'_{\tau+i-1}} = \{x | x \in X,\ y_i = \lambda(\delta(s',\ x'_\tau\ x'_{\tau+1}\ \ldots\ x'_{\tau+i-1}),\ x)\}$ in the case of $i > 0$, or $\{x | x \in X, y_i = \lambda(s', x)\}$ otherwise.

2.3. If $X_{s',x'_\tau x'_{\tau+1}\dots x'_{\tau+i-1}} \neq \varnothing$, then choose an element in it as $x'_{\tau+i}$, delete this element from it, increase i by 1 and go to Step 2.4; otherwise, decrease i by 1 and go to Step 2.5.
2.4. If $i > l$, then output $x'_0 \dots x'_{\tau+l}$ as a signature $x_0 \dots x_{\tau+l}$ and stop; otherwise, go to Step 2.2.
2.5. If $i \geqslant 0$, go to Step 2.3; otherwise, go to Step 1.2.

An execution of Algorithm 3 consists of two phases. The first phase is to search several such initial states, say $s' = s_{x'_0\dots x'_{\tau-1}}$, for some values of $x'_0, \dots, x'_{\tau-1} \in X$ with $y_0 y_1 \dots y_l \notin W^M_{l+1,s'}$. The second phase is to search an initial state $s' = s_{x'_0\dots x'_{\tau-1}}$ with $y_0 y_1 \dots y_l \in W^M_{l+1,s'}$. We evaluate the search amount for the second phase. We point out that the phase 2 and Algorithm 1 in Sect. 2.3 of Chap. 2 are the same except the subscripts of x'_i in Algorithm 3 with offset τ. Thus for a valid $x'_0 \dots x'_{\tau-1}$, we can evaluate the search amount of the phase 2 in Algorithm 3 as well as to evaluate the search amount of Algorithm 1. Taking account of the search amount of the first phase of an execution of Algorithm 3, the complexity for signature is a little more than the complexity for encryption.

By the way, the initial state for encryption may be variable, whenever $M_0 = \langle X, Y, S_0, \delta_0, \lambda_0 \rangle$ is defined by

$$y_i = \sum_{j=1}^{t_0} A_j y_{i-j} + f_0(x'_i, \dots, x'_{i-r_0}),$$
$$i = 0, 1, \dots$$

and $M_0^* = \langle Y, X, S_0^*, \delta_0^*, \lambda_0^* \rangle$ is defined by

$$x'_i = f_0^*(x'_{i-1}, \dots, x'_{i-r_0}) + \sum_{j=0}^{t_0+\tau_0} B_j^* y_{i-j},$$
$$i = 0, 1, \dots,$$

where $f_0(0, \dots, 0) = f_0^*(0, \dots, 0) = 0$. Suppose that $\bar{y}_{-1}$, …, $\bar{y}_{-t_0}$ satisfy the condition

$$\lambda_0^*(\langle 0, \dots, 0, 0, \dots, 0, \bar{y}_{-1}, \dots, \bar{y}_{-t_0} \rangle, 00\dots) = 00\dots \tag{9.20}$$

Let

$$\lambda_0(\langle \bar{y}_{-1}, \dots, \bar{y}_{-t_0}, 0, \dots, 0, \rangle, 00\dots) = \bar{y}_0 \bar{y}_1 \dots \tag{9.21}$$

From $PI(M_0, M_0^*, \tau_0)$, this yields

$$\lambda_0^*(\langle 0, \dots, 0, \bar{y}_{\tau_0-1}, \dots, \bar{y}_0, \bar{y}_{-1}, \dots, \bar{y}_{-t_0} \rangle, \bar{y}_{\tau_0} \bar{y}_{\tau_0+1} \dots) = 00\dots$$

Subtracting two sides of (9.20) from two sides of the above equation, from the definition of M_0^*, we have

$$\lambda_0^*(\langle 0,\ldots,0,\bar{y}_{\tau_0-1},\ldots,\bar{y}_0,0,\ldots,0\rangle,\bar{y}_{\tau_0}\bar{y}_{\tau_0+1}\ldots)=00\ldots \tag{9.22}$$

Let

$$\lambda_0(\langle y_{-1},\ldots,y_{-t_0},x'_{-1},\ldots,x'_{-r_0}\rangle,x'_0x'_1\ldots)=y_0y_1\ldots, \tag{9.23}$$

and $y'_i=y_i+\bar{y}_i$, $i=0,1,\ldots$ Adding two sides of (9.21) to two sides of (9.23), from the definition of M_0, we have

$$\lambda_0(\langle y_{-1}+\bar{y}_{-1},\ldots,y_{-t_0}+\bar{y}_{-t_0},x'_{-1},\ldots,x'_{-r_0}\rangle,x'_0x'_1\ldots)=y'_0y'_1\ldots$$

On the other hand, from $PI(M_0,M_0^*,\tau_0)$, (9.23) yields

$$\lambda_0^*(\langle x'_{-1},\ldots,x'_{-r_0},y_{\tau_0-1},\ldots,y_0,y_{-1},\ldots,y_{-t_0}\rangle,y_{\tau_0}y_{\tau_0+1}\ldots)=x'_0x'_1\ldots$$

Adding two sides of (9.22) to two sides of the above equation, from the definition of M_0^*, we have

$$\lambda_0^*(\langle x'_{-1},\ldots,x'_{-r_0},y'_{\tau_0-1},\ldots,y'_0,y_{-1},\ldots,y_{-t_0}\rangle,y'_{\tau_0}y'_{\tau_0+1}\ldots)=x'_0x'_1\ldots$$

We conclude that an offset $\langle\bar{y}_{-1},\ldots,\bar{y}_{-t_0}\rangle$ which satisfies (9.20) may be added to the output part of the initial state for encryption.

The equation (9.20) is equivalent to

$$\begin{bmatrix} B^*_{\tau_0+1} & B^*_{\tau_0+2} & \ldots & B^*_{\tau_0+t_0-1} & B^*_{\tau_0+t_0}\\ B^*_{\tau_0+2} & B^*_{\tau_0+3} & \ldots & B^*_{\tau_0+t_0} & 0\\ \ldots & \ldots & \ldots & \ldots & \ldots\\ B^*_{\tau_0+t_0-1} & B^*_{\tau_0+t_0} & \ldots & 0 & 0\\ B^*_{\tau_0+t_0} & 0 & \ldots & 0 & 0 \end{bmatrix} \begin{bmatrix} \bar{y}_{-1}\\ \bar{y}_{-2}\\ \vdots\\ \bar{y}_{-t_0} \end{bmatrix} = \begin{bmatrix} 0\\ 0\\ \vdots\\ 0 \end{bmatrix} \tag{9.24}$$

which can be further reduced by row transformation. For example, in the example of FAPKC given in Sect. 9.3, the equation (9.24) can be further reduced to

$$\begin{bmatrix} \texttt{01101011 00010001 01111111}\\ \texttt{00010001 01111111 00000000}\\ \texttt{10111111 00110100 00000000}\\ \texttt{01010000 00010011 00000000}\\ \texttt{01111111 00000000 00000000}\\ \texttt{00110100 00000000 00000000}\\ \texttt{00010011 00000000 00000000}\\ \texttt{00000010 00000000 00000000}\\ \texttt{10001100 00000000 00000000} \end{bmatrix} \begin{bmatrix} \bar{y}_{-1}\\ \bar{y}_{-2}\\ \bar{y}_{-3} \end{bmatrix} = \begin{bmatrix} 0\\ 0\\ 0\\ 0\\ 0\\ 0\\ 0\\ 0\\ 0 \end{bmatrix}.$$

There are $2^{24-1-3-5}=2^{15}$ different solutions of $[\bar{y}_{-1},\bar{y}_{-2},\bar{y}_{-3}]$. This increases the complexity of the exhause search for encryption, because the virtual initial state for encryption is unknown.

Stochastic search attack

The above attack by exhausting search is a deterministic algorithm. It can be modified to a stochastic one. Algorithm 2 in Sect. 2.3 of Chap. 2 is a stochastic search algorithm to retrieve a plaintext $x_0x_1 \dots x_l$ from a ciphertext $y_0y_1 \dots y_l$ $(= \lambda(s, x_0x_1 \dots x_l))$, where M is the finite automaton $C'(M_n, \dots, M_1, M_0)$ in a user's public key, and s is a state of M in which partial components are given by s_e^{out} and s_e^{in} in the public key. In the case of the example in Sect. 9.3, s is $\langle s_e^{out}, s_e^{in} \rangle$.

A formula of the successful probability is deduced in Sect. 2.3 of Chap. 2 for a finite automaton M which is equivalent to $C(\bar{M}_0, D_{X,r_1}, \bar{M}_1, D_{X,r_2}, \bar{M}_2, \dots, \bar{M}_{\tau-1}, D_{X,r_\tau}, \bar{M}_\tau)$, where $0 \leqslant r_1 \leqslant r_2 \leqslant \cdots \leqslant r_\tau \leqslant m$, $\bar{M}_i$ is a weakly invertible finite automaton with delay 0 for $i = 0, 1, \dots, \tau$, and $\bar{M}_i$ is $(m - r_{i+1})$-preservable for $i = 1, \dots, \tau - 1$. According to the formula, in the case of the example, the probability of successfully choosing $x'_0, \dots, x'_l$ of Algorithm 2 is $2^{\sum_{i=0}^{\min(l,\tau-1)} r_{\tau-i} - (l+1)r_\tau}$. Taking $l = \tau = 15$, the probability is 2^{-52} whenever $r_1, \dots, r_{15}$ are 1, 2, 2, 3, 3, 3, 4, 4, 4, 5, 5, 5, 6, 6, 7, respectively, or 2^{-43} whenever $r_1, \dots, r_{15}$ are 1, 2, 3, 4, 4, 4, 5, 5, 5, 5, 6, 6, 6, 6, 7, respectively.

Similar to Algorithm 2, an attacker may adopt a stochastic search attack to forge a signature for a given message using a stochastic search algorithm like the following.

Algorithm 4

Input : a message $y_0y_1 \dots y_l$.

Output : the signature $x_0x_1 \dots x_{\tau+l}$ which satisfies

$$y_0y_1 \dots y_l = \lambda(s_{x_0,\dots,x_{\tau-1}}, x_\tau \dots x_{\tau+l}),$$

where $s_{x_0,\dots,x_{\tau-1}}$ is a state $\langle y_{-1}, \dots, y_{-t_0}, x_{\tau-1}, \dots, x_{\tau-r} \rangle$ determined by $s_v^{out} = \langle y_{-1}, \dots, y_{-t_0} \rangle$ and $s_v^{in} = \langle x_{-1}, \dots, x_{-r+\tau} \rangle$ in the public key and $x_0, \dots, x_{\tau-1}$.

Procedure :

1. *Guess the prefix of length τ of the signature.*
   Choose randomly $x'_0, \dots, x'_{\tau-1} \in X$.
2. *Guess the main part of the signature.*
2.1. Set $i = 0$ and $s' = s_{x'_0,\dots,x'_{\tau-1}}$.
2.2. Set $X_{s',x'_\tau x'_{\tau+1} \dots x'_{\tau+i-1}} = \{x | x \in X, y_i = \lambda(\delta(s', x'_\tau\ x'_{\tau+1} \dots x'_{\tau+i-1}), x)\}$ in the case of $i > 0$, or $\{x | x \in X, y_i = \lambda(s', x)\}$ otherwise.
2.3. If $X_{s',x'_\tau x'_{\tau+1} \dots x'_{\tau+i-1}} \neq \varnothing$, then choose randomly an element in it as $x'_{\tau+i}$, increase i by 1 and go to Step 2.4; otherwise, prompt a failure information and stop.

2.4. If $i > l$, then output $x'_0 \dots x'_{\tau+l}$ as a signature $x_0 \dots x_{\tau+l}$ and stop; otherwise, go to Step 2.2.

Let $p_{ix}^{y_0 \dots y_l}$ be the probability of successfully choosing an initial part $x'_0 \dots x'_{\tau-1}$ in Step 1 of Algorithm 4. Let $X_{\text{valid}}^{y_0 \dots y_l} = \{x'_0 \dots x'_{\tau-1} | I^M_{y_0 y_1 \dots y_l, s_{x'_0, \dots, x'_{\tau-1}}} \neq \varnothing\}$. Then $p_{ix}^{y_0 \dots y_l} = |X_{\text{valid}}^{y_0 \dots y_l}|/q^{m\tau}$.

We point out that the phase 2 and Algorithm 2 in Sect. 2.3 of Chap. 2 are the same except the subscripts of x'_i in Algorithm 4 with offset τ and replacing s by s'. From Theorem 2.3.8, the probability of successfully executing Algorithm 4 is $p_{ix}^{y_0 \dots y_l} q^{\sum_{i=0}^{\min(l,\tau-1)} r_{\tau-i} - (l+1) r_\tau}$, for a finite automaton M which is equivalent to $C(\bar{M}_0, D_{X,r_1}, \bar{M}_1, D_{X,r_2}, \bar{M}_2, \dots, \bar{M}_{\tau-1}, D_{X,r_\tau}, \bar{M}_\tau)$, where $0 \leqslant r_1 \leqslant r_2 \leqslant \cdots \leqslant r_\tau \leqslant m$, $\bar{M}_i$ is weakly invertible finite automata with delay 0 for $i = 0, 1, \dots, \tau$, and $\bar{M}_i$ is $(m - r_{i+1})$-preservable for $i = 1, \dots, \tau - 1$.

9.6 Generalized Algorithms

9.6.1 Some Theoretical Results

In this section, we deal with pseudo-memory finite automata instead of memory finite automata. Notice that a generation method of pseudo-memory finite automata with invertibility is discussed in Chap. 3.

Let $M = \langle X, Y, (U^{(1)})^{p_1+1} \times \cdots \times (U^{(c)})^{p_c+1} \times X^r, \delta, \lambda \rangle$ be a finite automaton defined by

$$\begin{aligned} y_i &= f(u^{(1)}(i, p_1 + 1), \dots, u^{(c)}(i, p_c + 1), x(i, r + 1)), \\ u_{i+1}^{(j)} &= g_j(u^{(1)}(i, p_1 + 1), \dots, u^{(c)}(i, p_c + 1), x(i, r + 1)), \\ & \quad j = 1, \dots, c, \quad i = 0, 1, \dots \end{aligned}$$

Let $M' = \langle Y, Z, Z^k \times (W^{(1)})^{n_1+1} \times \cdots \times (W^{(d)})^{n_d+1} \times Y^h, \delta', \lambda' \rangle$ be a finite automaton defined by

$$\begin{aligned} z_i &= \varphi(z(i-1, k), w^{(1)}(i, n_1 + 1), \dots, w^{(d)}(i, n_d + 1), y(i, h + 1)), \\ w_{i+1}^{(j)} &= \psi_j(z(i-1, k), w^{(1)}(i, n_1 + 1), \dots, w^{(d)}(i, n_d + 1), y(i, h + 1)), \\ & \quad j = 1, \dots, d, \quad i = 0, 1, \dots \end{aligned}$$

From M and M', a finite automaton $\langle X, Z, Z^k \times (W^{(1)})^{n_1+1} \times \cdots \times (W^{(d)})^{n_d+1} \times (U^{(1)})^{h+p_1+1} \times \cdots \times (U^{(c)})^{h+p_c+1} \times X^{h+r}, \delta'', \lambda'' \rangle$ is defined by

$$
\begin{aligned}
z_i &= \varphi(z(i-1,k), w^{(1)}(i,n_1+1),\dots,w^{(d)}(i,n_d+1),\\
&\qquad f(u^{(1)}(i,p_1+1),\dots,u^{(c)}(i,p_c+1),x(i,r+1)),\ \dots,\\
&\qquad f(u^{(1)}(i-h,p_1+1),\dots,u^{(c)}(i-h,p_c+1),x(i-h,r+1))),\\
w_{i+1}^{(j)} &= \psi_j(z(i-1,k), w^{(1)}(i,n_1+1),\dots,w^{(d)}(i,n_d+1),\\
&\qquad f(u^{(1)}(i,p_1+1),\dots,u^{(c)}(i,p_c+1),x(i,r+1)),\ \dots,\\
&\qquad f(u^{(1)}(i-h,p_1+1),\dots,u^{(c)}(i-h,p_c+1),x(i-h,r+1))),\\
&\quad j=1,\dots,d,\\
u_{i+1}^{(j)} &= g_j(u^{(1)}(i,p_1+1),\dots,u^{(c)}(i,p_c+1),x(i,r+1)),\\
&\quad j=1,\dots,c,\\
&\quad i=0,1,\dots
\end{aligned}
\tag{9.25}
$$

We still use $C'(M,M')$ to denote this finite automaton in this section.

Theorem 9.6.1. *For any state*

$$
\begin{aligned}
s'' = \langle & z(-1,k), w^{(1)}(0,n_1+1),\dots,w^{(d)}(0,n_d+1),\\
& u^{(1)}(0,h+p_1+1),\dots,u^{(c)}(0,h+p_c+1),x(-1,h+r)\rangle
\end{aligned}
$$

of $C'(M,M')$, let

$$
\begin{aligned}
s &= \langle u^{(1)}(0,p_1+1),\dots,u^{(c)}(0,p_c+1),x(-1,r)\rangle,\\
s' &= \langle z(-1,k), w^{(1)}(0,n_1+1),\dots,w^{(d)}(0,n_d+1),y(-1,h)\rangle,
\end{aligned}
$$

where

$$
\begin{aligned}
y_i &= f(u^{(1)}(i,p_1+1),\dots,u^{(c)}(i,p_c+1),x(i,r+1)),\\
&\quad i=-h,\dots,-1.
\end{aligned}
\tag{9.26}
$$

Then the state $\langle s,s'\rangle$ of $C(M,M')$ and s'' are equivalent.

Proof. Taking arbitrary $x_0,x_1,\dots\in X$, let

$$z_0z_1\dots=\lambda''(s'',x_0x_1\dots).$$

Then there exist $w^{(1)}(i)$, …, $w^{(d)}(i)$, $u^{(1)}(i)$, …, $u^{(c)}(i)$, $i=1,2,\dots$ such that (9.25) holds. Denoting

$$y_i = f(u^{(1)}(i,p_1+1),\dots,u^{(c)}(i,p_c+1),x(i,r+1)),\ i=0,1,\dots \tag{9.27}$$

and using (9.26), (9.25) yields

$$
\begin{aligned}
z_i &= \varphi(z(i-1,k), w^{(1)}(i,n_1+1),\dots,w^{(d)}(i,n_d+1),y(i,h+1)),\\
w_{i+1}^{(j)} &= \psi_j(z(i-1,k), w^{(1)}(i,n_1+1),\dots,w^{(d)}(i,n_d+1),y(i,h+1)),\\
&\quad j=1,\dots,d,\quad i=0,1,\dots
\end{aligned}
$$

From the definition of M', we have

$$z_0 z_1 \ldots = \lambda'(s', y_0 y_1 \ldots).$$

Using (9.25) and (9.27), from the definition of M, we obtain

$$y_0 y_1 \ldots = \lambda(s, x_0 x_1 \ldots).$$

Thus

$$\lambda''(s'', x_0 x_1 \ldots) = z_0 z_1 \ldots = \lambda'(s', \lambda(s, x_0 x_1 \ldots)) = \lambda'''(\langle s, s' \rangle, x_0 x_1 \ldots),$$

where λ''' is the output function of $C(M, M')$. Therefore, $\langle s, s' \rangle$ and s'' are equivalent. □

Let $M = \langle V, Z, Z^k \times W^{n+1} \times V^h, \delta, \lambda \rangle$ be a finite automaton, where

$$\begin{aligned}
&\lambda(\langle z(-1,k), w(0,n+1), v(-1,h) \rangle, v_0) = z_0, \\
&\delta(\langle z(-1,k), w(0,n+1), v(-1,h) \rangle, v_0) = \langle z(0,k), w(1,n+1), v(0,h) \rangle, \\
&\quad z_0 = \varphi(z(-1,k), w(0,n+1), v(0,h+1)), \\
&\quad w_1 = \psi(z(-1,k), w(0,n+1), v(0,h+1)).
\end{aligned}$$

Let $M^* = \langle Z, V, V^h \times W^{n+1} \times Z^{\tau+k}, \delta^*, \lambda^* \rangle$ be a finite automaton, where

$$\begin{aligned}
&\lambda^*(\langle v(-1,h), w(0,n+1), z(-1,\tau+k) \rangle, z_0) = v_0, \\
&\delta^*(\langle v(-1,h), w(0,n+1), z(-1,\tau+k) \rangle, z_0) \\
&\quad = \langle v(0,h), w(1,n+1), z(0,\tau+k) \rangle, \\
&\quad v_0 = \varphi^*_\tau(v(-1,h), w(0,n+1), z(0,\tau+k+1)), \\
&\quad w_1 = \psi(z(-\tau-1,k), w(0,n+1), v(0,h+1)).
\end{aligned}$$

We use $PI_1(M, M^*, \tau)$ to denote the following condition: for any state

$$s_0 = \langle z(-1,k), w(0,n+1), v(-1,h) \rangle$$

of M and any $v_0, v_1, \ldots \in V$, if

$$z_0 z_1 \ldots = \lambda(s_0, v_0 v_1 \ldots),$$

then

$$v_0 v_1 \ldots = \lambda^*(s^*_\tau, z_\tau z_{\tau+1} \ldots),$$

where

$$s^*_\tau = \langle v(-1,h), w(0,n+1), z(\tau-1, \tau+k) \rangle.$$

We use $PI_2(M^*, M, \tau)$ to denote the following condition: for any state

$$s_0^* = \langle v(-1,h), w(0,n+1), z(-1,\tau+k)\rangle$$

of M^* and any $z_0, z_1, \ldots \in Z$, if

$$v_0 v_1 \ldots = \lambda^*(s_0^*, z_0 z_1 \ldots),$$

then

$$z_0 z_1 \ldots = \lambda(s_\tau, v_\tau v_{\tau+1} \ldots),$$

where

$$\begin{aligned} s_\tau &= \langle z(-1,k), w(\tau,n+1), v(\tau-1,h)\rangle, \\ w_{i+1} &= \psi(z(i-\tau-1,k), w(i,n+1), v(i,h+1)), \\ &\quad i = 0,1,\ldots,\tau-1. \end{aligned}$$

For any i, $0 \leqslant i \leqslant n$, let X_i be the column vector space over $GF(q)$ of dimension l_i. Let Y be the column vector space over $GF(q)$ of dimension m, and $X = X_n$.

For any i, $1 \leqslant i \leqslant n$, let $M_i = \langle X_i, X_{i-1}, U_i^{p_i+1} \times X_i^{r_i}, \delta_i, \lambda_i\rangle$ be an $(r_i, 0, p_i)$-order pseudo-memory finite automaton determined by f_i and g_i, where

$$\begin{aligned} &\lambda_i(\langle u^{(i)}(0,p_i+1), x^{(i)}(-1,r_i)\rangle, x_0^{(i)}) = x_0^{(i-1)}, \\ &\delta_i(\langle u^{(i)}(0,p_i+1), x^{(i)}(-1,r_i)\rangle, x_0^{(i)}) = \langle u^{(i)}(1,p_i+1), x^{(i)}(0,r_i)\rangle, \\ &\quad x_0^{(i-1)} = f_i(u^{(i)}(0,p_i+1), x^{(i)}(0,r_i+1)), \\ &\quad u_1^{(i)} = g_i(u^{(i)}(0,p_i+1), x^{(i)}(0,r_i+1)), \end{aligned} \tag{9.28}$$

and let $M_i^* = \langle X_{i-1}, X_i, X_i^{r_i} \times U_i^{p_i+1} \times X_{i-1}^{\tau_i}, \delta_i^*, \lambda_i^*\rangle$ be a (τ_i, r_i, p_i)-order pseudo-memory finite automaton determined by f_i^* and g_i, where

$$\begin{aligned} &\lambda_i^*(\langle x^{(i)}(-1,r_i), u^{(i)}(0,p_i+1), x^{(i-1)}(-1,\tau_i)\rangle, x_0^{(i-1)}) = x_0^{(i)}, \\ &\delta_i^*(\langle x^{(i)}(-1,r_i), u^{(i)}(0,p_i+1), x^{(i-1)}(-1,\tau_i)\rangle, x_0^{(i-1)}) \\ &\quad = \langle x^{(i)}(0,r_i), u^{(i)}(1,p_i+1), x^{(i-1)}(0,\tau_i)\rangle, \\ &\quad x_0^{(i)} = f_i^*(x^{(i)}(-1,r_i), u^{(i)}(0,p_i+1), x^{(i-1)}(0,\tau_i+1)), \\ &\quad u_1^{(i)} = g_i(u^{(i)}(0,p_i+1), x^{(i)}(0,r_i+1)). \end{aligned} \tag{9.29}$$

Assume that $\tau_i \leqslant r_i$ for $1 \leqslant i \leqslant n$.

Let $M_0 = \langle X_0, Y, Y^{t_0} \times U_0^{p_0+1} \times X_0^{r_0}, \delta_0, \lambda_0\rangle$ be an (r_0, t_0, p_0)-order pseudo-memory finite automaton determined by f_0 and g_0, where

$$\begin{aligned} &\lambda_0(\langle y(-1,t_0), u^{(0)}(0,p_0+1), x^{(0)}(-1,r_0)\rangle, x_0^{(0)}) = y_0, \\ &\delta_0(\langle y(-1,t_0), u^{(0)}(0,p_0+1), x^{(0)}(-1,r_0)\rangle, x_0^{(0)}) \end{aligned}$$

$$
\begin{aligned}
&= \langle y(0,t_0), u^{(0)}(1,p_0+1), x^{(0)}(0,r_0)\rangle, \qquad (9.30)\\
&y_0 = f_0(y(-1,t_0), u^{(0)}(0,p_0+1), x^{(0)}(0,r_0+1)),\\
&u_1^{(0)} = g_0(y(-1,t_0), u^{(0)}(0,p_0+1), x^{(0)}(0,r_0+1)).
\end{aligned}
$$

Let $M_0^* = \langle Y, X_0, X_0^{r_0} \times U_0^{p_0+1} \times Y^{\tau_0+t_0}, \delta_0^*, \lambda_0^*\rangle$ be a (τ_0+t_0, r_0, p_0)-order pseudo-memory finite automaton determined by f_0^* and g_0 where

$$
\begin{aligned}
&\lambda_0^*(\langle x^{(0)}(-1,r_0), u^{(0)}(0,p_0+1), y(-1,\tau_0+t_0)\rangle, y_0) = x_0^{(0)},\\
&\delta_0^*(\langle x^{(0)}(-1,r_0), u^{(0)}(0,p_0+1), y(-1,\tau_0+t_0)\rangle, y_0)\\
&\quad = \langle x^{(0)}(0,r_0), u^{(0)}(1,p_0+1), y(0,\tau_0+t_0)\rangle, \qquad (9.31)\\
&\quad x_0^{(0)} = f_0^*(x^{(0)}(-1,r_0), u^{(0)}(0,p_0+1), y(0,\tau_0+t_0+1)),\\
&\quad u_1^{(0)} = g_0(y(-\tau_0-1,t_0), u^{(0)}(0,p_0+1), x^{(0)}(0,r_0+1)).
\end{aligned}
$$

Assume that $\tau_0 \leqslant r_0$.

Abbreviate $\tau_{i,j} = \tau_i + \tau_{i+1} + \cdots + \tau_j$, $r_{i,j} = r_i + r_{i+1} + \cdots + r_j$ and $p_{i,j} = r_i + r_{i+1} + \cdots + r_{j-1} + p_j$ for any integer i and j with $i \leqslant j$. Let $\tau_{i,j} = 0$ in the case of $i > j$. Thus $p_{j,j} = p_j$, $\tau_{j,j} = \tau_j$ and $r_{j,j} = r_j$. Let $f_{n,n} = f_n$ and $g_{n,n} = g_n$. For any i, $1 \leqslant i \leqslant n-1$, let

$$
\begin{aligned}
&f_{i,n}(u^{(i)}(0,p_{i,i}+1), u^{(i+1)}(0,p_{i,i+1}+1), \ldots, u^{(n)}(0,p_{i,n}+1), x^{(n)}(0,r_{i,n}+1))\\
&\quad = f_i(u^{(i)}(0,p_i+1),\\
&\qquad f_{i+1,n}(u^{(i+1)}(0,p_{i+1,i+1}+1), u^{(i+2)}(0,p_{i+1,i+2}+1), \ldots,\\
&\qquad\quad u^{(n)}(0,p_{i+1,n}+1), x^{(n)}(0,r_{i+1,n}+1)),\\
&\qquad \cdots\cdots,\\
&\qquad f_{i+1,n}(u^{(i+1)}(-r_i,p_{i+1,i+1}+1), u^{(i+2)}(-r_i,p_{i+1,i+2}+1), \ldots,\\
&\qquad\quad u^{(n)}(-r_i,p_{i+1,n}+1), x^{(n)}(-r_i,r_{i+1,n}+1)))
\end{aligned}
$$

and

$$
\begin{aligned}
&g_{i,n}(u^{(i)}(0,p_{i,i}+1), u^{(i+1)}(0,p_{i,i+1}+1), \ldots, u^{(n)}(0,p_{i,n}+1), x^{(n)}(0,r_{i,n}+1))\\
&\quad = g_i(u^{(i)}(0,p_i+1),\\
&\qquad f_{i+1,n}(u^{(i+1)}(0,p_{i+1,i+1}+1), u^{(i+2)}(0,p_{i+1,i+2}+1), \ldots,\\
&\qquad\quad u^{(n)}(0,p_{i+1,n}+1), x^{(n)}(0,r_{i+1,n}+1)),\\
&\qquad \cdots\cdots,\\
&\qquad f_{i+1,n}(u^{(i+1)}(-r_i,p_{i+1,i+1}+1), u^{(i+2)}(-r_i,p_{i+1,i+2}+1), \ldots,\\
&\qquad\quad u^{(n)}(-r_i,p_{i+1,n}+1), x^{(n)}(-r_i,r_{i+1,n}+1))).
\end{aligned}
$$

Let

$$
\begin{aligned}
&f_{0,n}(y(-1,t_0), u^{(0)}(0,p_{0,0}+1), u^{(1)}(0,p_{0,1}+1), \ldots,\\
&\qquad\qquad u^{(n)}(0,p_{0,n}+1), x^{(n)}(0,r_{0,n}+1))
\end{aligned}
$$

$$
\begin{aligned}
&= f_0(y(-1,t_0), u^{(0)}(0,p_0+1),\\
&\qquad f_{1,n}(u^{(1)}(0,p_{1,1}+1), u^{(2)}(0,p_{1,2}+1),\dots,\\
&\qquad\quad u^{(n)}(0,p_{1,n}+1), x^{(n)}(0,r_{1,n}+1)),\\
&\qquad \cdots\cdots,\\
&\qquad f_{1,n}(u^{(1)}(-r_0,p_{1,1}+1), u^{(2)}(-r_0,p_{1,2}+1),\dots,\\
&\qquad\quad u^{(n)}(-r_0,p_{1,n}+1), x^{(n)}(-r_0,r_{1,n}+1)))
\end{aligned}
$$

and

$$
\begin{aligned}
&g_{0,n}(y(-1,t_0), u^{(0)}(0,p_{0,0}+1), u^{(1)}(0,p_{0,1}+1),\dots,\\
&\qquad\qquad u^{(n)}(0,p_{0,n}+1), x^{(n)}(0,r_{0,n}+1))\\
&= g_0(y(-1,t_0), u^{(0)}(0,p_0+1),\\
&\qquad f_{1,n}(u^{(1)}(0,p_{1,1}+1), u^{(2)}(0,p_{1,2}+1),\dots,\\
&\qquad\quad u^{(n)}(0,p_{1,n}+1), x^{(n)}(0,r_{1,n}+1)),\\
&\qquad \cdots\cdots,\\
&\qquad f_{1,n}(u^{(1)}(-r_0,p_{1,1}+1), u^{(2)}(-r_0,p_{1,2}+1),\dots,\\
&\qquad\quad u^{(n)}(-r_0,p_{1,n}+1), x^{(n)}(-r_0,r_{1,n}+1))).
\end{aligned}
$$

Let $M_{n,n} = M_n$ and $M_{i,n} = C'(M_{i+1,n}, M_i)$ for $0 \leqslant i \leqslant n-1$. From the definition of compound finite automata, for any i, $1 \leqslant i \leqslant n$, we have

$$M_{i,n} = \langle X_n, X_{i-1}, U_i^{p_{i,i}+1} \times U_{i+1}^{p_{i,i+1}+1} \times \cdots \times U_n^{p_{i,n}+1} \times X_n^{r_{i,n}}, \delta_{i,n}, \lambda_{i,n}\rangle,$$

where

$$
\begin{aligned}
&\lambda_{i,n}(\langle u^{(i)}(0,p_{i,i}+1),\dots,u^{(n)}(0,p_{i,n}+1), x^{(n)}(-1,r_{i,n})\rangle, x_0^{(n)}) = x_0^{(i-1)},\\
&\delta_{i,n}(\langle u^{(i)}(0,p_{i,i}+1),\dots,u^{(n)}(0,p_{i,n}+1), x^{(n)}(-1,r_{i,n})\rangle, x_0^{(n)})\\
&\quad = \langle u^{(i)}(1,p_{i,i}+1),\dots,u^{(n)}(1,p_{i,n}+1), x^{(n)}(0,r_{i,n})\rangle,
\end{aligned}
$$

and

$$
\begin{aligned}
&x_0^{(i-1)} = f_{i,n}(u^{(i)}(0,p_{i,i}+1),\dots,u^{(n)}(0,p_{i,n}+1), x^{(n)}(0,r_{i,n}+1)),\\
&u_1^{(c)} = g_{c,n}(u^{(c)}(0,p_{c,c}+1),\dots,u^{(n)}(0,p_{c,n}+1), x^{(n)}(0,r_{c,n}+1)),\\
&\quad c = i,\dots,n.
\end{aligned}
$$

And we have

$$M_{0,n} = \langle X_n, Y, Y^{t_0} \times U_0^{p_{0,0}+1} \times U_1^{p_{0,1}+1} \times \cdots \times U_n^{p_{0,n}+1} \times X_n^{r_{0,n}}, \delta_{0,n}, \lambda_{0,n}\rangle,$$

where

$$
\begin{aligned}
&\lambda_{0,n}(\langle y(-1,t_0), u^{(0)}(0,p_{0,0}+1),\dots,u^{(n)}(0,p_{0,n}+1), x^{(n)}(-1,r_{0,n})\rangle, x_0^{(n)}) = y_0,\\
&\delta_{0,n}(\langle y(-1,t_0), u^{(0)}(0,p_{0,0}+1),\dots,u^{(n)}(0,p_{0,n}+1), x^{(n)}(-1,r_{0,n})\rangle, x_0^{(n)})\\
&\quad = \langle y(0,t_0), u^{(0)}(1,p_{0,0}+1),\dots,u^{(n)}(1,p_{0,n}+1), x^{(n)}(0,r_{0,n})\rangle,
\end{aligned}
$$

and

$$\begin{aligned}
y_0 &= f_{0,n}(y(-1,t_0), u^{(0)}(0,p_{0,0}+1), u^{(1)}(0,p_{0,1}+1),\dots,\\
&\qquad u^{(n)}(0,p_{0,n}+1), x^{(n)}(0,r_{0,n}+1)),\\
u_1^{(0)} &= g_{0,n}(y(-1,t_0), u^{(0)}(0,p_{0,0}+1), u^{(1)}(0,p_{0,1}+1),\dots,\\
&\qquad u^{(n)}(0,p_{0,n}+1), x^{(n)}(0,r_{0,n}+1)),\\
u_1^{(c)} &= g_{c,n}(u^{(c)}(0,p_{c,c}+1), u^{(c+1)}(0,p_{c,c+1}+1),\dots,\\
&\qquad u^{(n)}(0,p_{c,n}+1), x^{(n)}(0,r_{c,n}+1)),\\
&\quad c=1,\dots,n.
\end{aligned}$$

Theorem 9.6.2. *Assume that M_i^*, M_i and τ_i satisfy $PI_2(M_i^*, M_i, \tau_i)$, $i = 0, 1, \dots, n$. Let $s_{-b_{i-1}}^{(i)*} = \langle x^{(i)}(-b_{i-1}-1, r_i),\ u^{(i)}(-b_{i-1}, p_i+1),\ \bar{x}^{(i-1)}(-b_{i-1}-1, \tau_i)\rangle$ be a state of M_i^*, $i = 1, \dots, n$. Let $x_{-b_0}^{(0)}, \dots, x_{-1}^{(0)} \in X_0$,*

$$\begin{aligned}
&x_{-b_{i-1}}^{(i)} \dots x_{-1}^{(i)} = \lambda_i^*(s_{-b_{i-1}}^{(i)*}, x_{-b_{i-1}}^{(i-1)} \dots x_{-1}^{(i-1)}),\\
&s_0^{(i)*} = \delta_i^*(s_{-b_{i-1}}^{(i)*}, x_{-b_{i-1}}^{(i-1)} \dots x_{-1}^{(i-1)}),\\
&\quad i = 1, \dots, n,
\end{aligned} \tag{9.32}$$

and

$$u_{j+1}^{(i)} = g_i(u^{(i)}(j, p_i+1), x^{(i)}(j, r_i+1)) \tag{9.33}$$

for $i = 1, \dots, n$ and $j = -b_{i-1}, \dots, -1$, where $b_{-1} = \max\{\tau_0, -\sum_{j=0}^{i-1} r_j + \sum_{j=0}^{i} \tau_j,\ i = 1, \dots, n\}$, and $b_i = b_{i-1} + r_i - \tau_i$ for $0 \leqslant i \leqslant n$. Let $s_0^{(0)*} = \langle x^{(0)}(-1, r_0),\ u^{(0)}(0, p_0+1),\ y(-1, \tau_0+t_0)\rangle$ be a state of M_0^*. If

$$\begin{aligned}
&x_0^{(0)} x_1^{(0)} \dots = \lambda_0^*(s_0^{(0)*}, y_0 y_1 \dots),\\
&x_0^{(i)} x_1^{(i)} \dots = \lambda_i^*(s_0^{(i)*}, x_0^{(i-1)} x_1^{(i-1)} \dots),\\
&\quad i = 1, \dots, n,
\end{aligned} \tag{9.34}$$

then

$$y_0 y_1 \dots = \lambda_{0,n}(s, x_{\tau_{0,n}}^{(n)} x_{\tau_{0,n}+1}^{(n)} \dots),$$

where $s = \langle y(-1, t_0),\ u^{(0)}(\tau_{0,0}, p_{0,0}+1),\ u^{(1)}(\tau_{0,1}, p_{0,1}+1),\ \dots,\ u^{(n)}(\tau_{0,n}, p_{0,n}+1),\ x^{(n)}(\tau_{0,n}-1, r_{0,n})\rangle$, and $u_j^{(i)}$ in s for $0 < j \leqslant \tau_{0,i}$ are computed out according to the following formulae

$$\begin{aligned}
u_{j+1}^{(i)} &= g_{i,n}(u^{(i)}(j, p_{i,i}+1), u^{(i+1)}(j+\tau_{i+1,i+1}, p_{i,i+1}+1), \dots,\\
&\qquad u^{(n)}(j+\tau_{i+1,n}, p_{i,n}+1), x^{(n)}(j+\tau_{i+1,n}, r_{i,n}+1)),\\
&\quad j = 0, 1, \dots, \tau_{0,i}-1,\\
&\quad i = n, n-1, \dots, 1,
\end{aligned} \tag{9.35}$$

$$u_{j+1}^{(0)} = g_{0,n}(y(j-\tau_0-1,t_0), u^{(0)}(j,p_{0,0}+1), u^{(1)}(j+\tau_{1,1},p_{0,1}+1),\ldots,$$
$$u^{(n)}(j+\tau_{1,n},p_{0,n}+1), x^{(n)}(j+\tau_{1,n},r_{0,n}+1)),$$
$$j=0,1,\ldots,\tau_{0,0}-1.$$

Proof. Suppose that (9.34) holds. From (9.32) and the part on M_i^* in (9.34), it is easy to obtain that

$$x_{-b_{i-1}}^{(i)}\ldots x_{-1}^{(i)}x_0^{(i)}x_1^{(i)}\ldots = \lambda_i^*(s_{-b_{i-1}}^{(i)*}, x_{-b_{i-1}}^{(i-1)}\ldots x_{-1}^{(i-1)}x_0^{(i-1)}x_1^{(i-1)}\ldots),$$

$i=1,\ldots,n$. Since $b_{i-1}\geqslant\tau_i$ and (9.33) holds for $i=1,\ldots,n$ and $j=-b_{i-1},\ldots,-1$, from $PI_2(M_i^*,M_i,\tau_i)$, it follows that

$$x_{-b_{i-1}}^{(i-1)}\ldots x_{-1}^{(i-1)}x_0^{(i-1)}x_1^{(i-1)}\ldots = \lambda_i(s^{(i)}, x_{-b_{i-1}+\tau_i}^{(i)}x_{-b_{i-1}+\tau_i+1}^{(i)}\ldots), \quad (9.36)$$

$i=1,\ldots,n$, where $s^{(i)}=\langle u^{(i)}(-b_{i-1}+\tau_i,p_i+1), x^{(i)}(-b_{i-1}+\tau_i-1,r_i)\rangle$, $i=1,\ldots,n$. For any $i, 1\leqslant i\leqslant n$, letting (9.33) for $j=0,1,\ldots$, since (9.33) holds for $j=-b_{i-1},\ldots,-1$, we have

$$u_{j+1}^{(i)} = g_i(u^{(i)}(j,p_i+1), x^{(i)}(j,r_i+1)), \quad (9.37)$$
$$j=-b_{i-1},-b_{i-1}+1,\ldots$$

For such $u^{(i)}$'s which satisfy (9.37), from the definition of M_i and (9.36), we obtain

$$x_j^{(i-1)} = f_i(u^{(i)}(j+\tau_i,p_i+1), x^{(i)}(j+\tau_i,r_i+1)), \quad (9.38)$$
$$j=-b_{i-1},-b_{i-1}+1,\ldots,$$

$i=1,\ldots,n$. Since $PI_2(M_0^*,M_0,\tau_0)$ holds, from the part on M_0^* in (9.34), we have

$$y_0y_1\ldots = \lambda_0(s^{(0)}, x_{\tau_0}^{(0)}x_{\tau_0+1}^{(0)}\ldots), \quad (9.39)$$

where $s^{(0)}=\langle y(-1,t_0), u^{(0)}(\tau_0,p_0+1), x^{(0)}(\tau_0-1,r_0)\rangle$, and

$$u_{j+1}^{(0)} = g_0(y(j-\tau_0-1,t_0), u^{(0)}(j,p_0+1), x^{(0)}(j,r_0+1)) \quad (9.40)$$

for $j=0,\ldots,\tau_0-1$.

We prove by induction on i that for any i, $1\leqslant i\leqslant n$, we have

$$x_{-b_{i-1}}^{(i-1)}x_{-b_{i-1}+1}^{(i-1)}\cdots$$
$$= \lambda_{i,n}(\langle u^{(i)}(-b_{i-1}+\tau_{i,i},p_{i,i}+1), u^{(i+1)}(-b_{i-1}+\tau_{i,i+1},p_{i,i+1}+1),\ldots,$$
$$u^{(n)}(-b_{i-1}+\tau_{i,n},p_{i,n}+1), x^{(n)}(-b_{i-1}+\tau_{i,n}-1,r_{i,n})\rangle, \quad (9.41)$$
$$x_{-b_{i-1}+\tau_{i,n}}^{(n)}x_{-b_{i-1}+\tau_{i,n}+1}^{(n)}\ldots)$$

and

$$x_j^{(i-1)} = f_{i,n}(u^{(i)}(j+\tau_{i,i}, p_{i,i}+1), u^{(i+1)}(j+\tau_{i,i+1}, p_{i,i+1}+1), \ldots, u^{(n)}(j+\tau_{i,n}, p_{i,n}+1), x^{(n)}(j+\tau_{i,n}, r_{i,n}+1)), \tag{9.42}$$

$$u_{j+\tau_{i,c}+1}^{(c)} = g_{c,n}(u^{(c)}(j+\tau_{i,c}, p_{c,c}+1), u^{(c+1)}(j+\tau_{i,c+1}, p_{c,c+1}+1), \ldots, u^{(n)}(j+\tau_{i,n}, p_{c,n}+1), x^{(n)}(j+\tau_{i,n}, r_{c,n}+1)),$$

$$c = i, \ldots, n, \quad j = -b_{i-1}, -b_{i-1}+1, \ldots$$

Basis : $i = n$. The formula (9.41) is

$$x_{-b_{n-1}}^{(n-1)} x_{-b_{n-1}+1}^{(n-1)} \cdots = \lambda_{n,n}(\langle u^{(n)}(-b_{n-1}+\tau_{n,n}, p_{n,n}+1), x^{(n)}(-b_{n-1}+\tau_{n,n}-1, r_{n,n})\rangle, x_{-b_{n-1}+\tau_{n,n}}^{(n)} x_{-b_{n-1}+\tau_{n,n}+1}^{(n)} \cdots)$$

which is deduced by (9.36), using $\lambda_{n,n} = \lambda_n$, $r_{n,n} = r_n$, $p_{n,n} = p_n$ and $\tau_{n,n} = \tau_n$. Similarly, the formula (9.42) is

$$x_j^{(n-1)} = f_{n,n}(u^{(n)}(j+\tau_{n,n}, p_{n,n}+1), x^{(n)}(j+\tau_{n,n}, r_{n,n}+1)),$$
$$u_{j+\tau_{n,n}+1}^{(n)} = g_{n,n}(u^{(n)}(j+\tau_{n,n}, p_{n,n}+1), x^{(n)}(j+\tau_{n,n}, r_{n,n}+1)),$$
$$j = -b_{n-1}, -b_{n-1}+1, \ldots$$

which is deduced by (9.37) and (9.38), using $f_{n,n} = f_n$, $g_{n,n} = g_n$, $r_{n,n} = r_n$, $p_{n,n} = p_n$, and $\tau_{n,n} = \tau_n$. *Induction step* : Suppose that for $i \geqslant 1$ we have proven that

$$x_{-b_i}^{(i)} x_{-b_i+1}^{(i)} \cdots = \lambda_{i+1,n}(\langle u^{(i+1)}(-b_i+\tau_{i+1,i+1}, p_{i+1,i+1}+1), u^{(i+2)}(-b_i+\tau_{i+1,i+2}, p_{i+1,i+2}+1), \ldots, u^{(n)}(-b_i+\tau_{i+1,n}, p_{i+1,n}+1), x^{(n)}(-b_i+\tau_{i+1,n}-1, r_{i+1,n})\rangle, x_{-b_i+\tau_{i+1,n}}^{(n)} x_{-b_i+\tau_{i+1,n}+1}^{(n)} \cdots) \tag{9.43}$$

and

$$x_j^{(c-1)} = f_{c,n}(u^{(c)}(j+\tau_{c,c}, p_{c,c}+1), u^{(c+1)}(j+\tau_{c,c+1}, p_{c,c+1}+1), \ldots, u^{(n)}(j+\tau_{c,n}, p_{c,n}+1), x^{(n)}(j+\tau_{c,n}, r_{c,n}+1)) \tag{9.44}$$

for $c = i+1$ and $j = -b_i, -b_i+1, \ldots$,

$$u_{j+\tau_{i+1,c}+1}^{(c)} = g_{c,n}(u^{(c)}(j+\tau_{i+1,c}, p_{c,c}+1), u^{(c+1)}(j+\tau_{i+1,c+1}, p_{c,c+1}+1), \ldots, u^{(n)}(j+\tau_{i+1,n}, p_{c,n}+1), x^{(n)}(j+\tau_{i+1,n}, r_{c,n}+1)) \tag{9.45}$$

for $c = i+1, \ldots, n$ and $j = -b_i, -b_i+1, \ldots$ We prove (9.41) and (9.42) hold. Since $-b_i = -b_{i-1} + \tau_i - r_i$, (9.44) holds for $c = i+1$ and $-b_{i-1} + \tau_i - r_i \leqslant$

$j < -b_{i-1} + \tau_i$. From (9.43) and (9.36), noticing $-b_i \leqslant -b_i + r_i = -b_{i-1} + \tau_i$, applying Theorem 9.6.1, (9.41) holds. Since (9.37) holds and (9.44) holds for $c = i+1$ and $j \geqslant -b_i = -b_{i-1} + \tau_i - r_i$, from the definition of $g_{c,n}$, we have

$$u^{(c)}_{j+\tau_{i,c}+1} = g_{c,n}(u^{(c)}(j+\tau_{i,c}, p_{c,c}+1), u^{(c+1)}(j+\tau_{i,c+1}, p_{c,c+1}+1), \ldots, u^{(n)}(j+\tau_{i,n}, p_{c,n}+1), x^{(n)}(j+\tau_{i,n}, r_{c,n}+1)) \tag{9.46}$$

for $c = i$ and $j \geqslant -b_{i-1}$. From $-b_i = -b_{i-1} - r_i + \tau_i \leqslant -b_{i-1} + \tau_i$, (9.45) holds for $c = i+1, \ldots, n$ and $j \geqslant -b_{i-1} + \tau_i$. Replacing j in (9.45) by $j + \tau_i$, it follows immediately that (9.46) holds for $c = i+1, \ldots, n$ and $j \geqslant -b_{i-1}$. Therefore, (9.46) holds for $c = i, \ldots, n$ and $j \geqslant -b_{i-1}$. From (9.41), (9.44) holds for $c = i$ and $j \geqslant -b_{i-1}$; therefore, (9.44) holds for $c = i, \ldots, n$ and $j \geqslant -b_{i-1}$. We conclude that (9.42) holds.

Especially, equation (9.41) and (9.42) hold for the case of $i = 1$, that is,

$$\begin{aligned} x^{(0)}_{-b_0} x^{(0)}_{-b_0+1} \cdots &= \lambda_{1,n}(\langle u^{(1)}(-b_0+\tau_{1,1}, p_{1,1}+1), u^{(2)}(-b_0+\tau_{1,2}, p_{1,2}+1), \ldots, \\ &\quad u^{(n)}(-b_0+\tau_{1,n}, p_{1,n}+1), x^{(n)}(-b_0+\tau_{1,n}-1, r_{1,n})\rangle, \\ &\quad x^{(n)}_{-b_0+\tau_{1,n}} x^{(n)}_{-b_0+\tau_{1,n}+1} \cdots) \end{aligned} \tag{9.47}$$

and

$$\begin{aligned} x^{(0)}_j &= f_{1,n}(u^{(1)}(j+\tau_{1,1}, p_{1,1}+1), u^{(2)}(j+\tau_{1,2}, p_{1,2}+1), \ldots, \\ &\quad u^{(n)}(j+\tau_{1,n}, p_{1,n}+1), x^{(n)}(j+\tau_{1,n}, r_{1,n}+1)), \\ u^{(c)}_{j+\tau_{1,c}+1} &= g_{c,n}(u^{(c)}(j+\tau_{1,c}, p_{c,c}+1), u^{(c+1)}(j+\tau_{1,c+1}, p_{c,c+1}+1), \ldots, \\ &\quad u^{(n)}(j+\tau_{1,n}, p_{c,n}+1), x^{(n)}(j+\tau_{1,n}, r_{c,n}+1)), \\ &\quad c = 1, \ldots, n, \quad j = -b_0, -b_0+1, \ldots \end{aligned} \tag{9.48}$$

Using Theorem 9.6.1, from (9.47), (9.39) and (9.48), we obtain

$$y_0 y_1 \ldots = \lambda_{0,n}(\bar{s}, x^{(n)}_{\tau_{0,n}} x^{(n)}_{\tau_{0,n}+1} \ldots),$$

where $\bar{s} = \langle y(-1, t_0), u^{(0)}(\tau_{0,0}, p_{0,0}+1), u^{(1)}(\tau_{0,1}, p_{0,1}+1), \ldots, u^{(n)}(\tau_{0,n}, p_{0,n}+1), x^{(n)}(\tau_{0,n}-1, r_{0,n})\rangle$.

To prove $s = \bar{s}$, letting (9.40) for $j \geqslant \tau_0$, since (9.40) holds for $0 \leqslant j < \tau_0$, (9.40) holds for $j \geqslant 0$. From (9.48) and the definition of $g_{0,n}$, noticing $b_{-1} \geqslant \tau_0$, this yields that

$$u^{(0)}_{j+\tau_0+1} = g_{0,n}(y(j-1, t_0), u^{(0)}(j+\tau_{0,0}, p_{0,0}+1), u^{(1)}(j+\tau_{0,1}, p_{0,1}+1), \ldots, u^{(n)}(j+\tau_{0,n}, p_{0,n}+1), x^{(n)}(j+\tau_{0,n}, r_{0,n}+1)) \tag{9.49}$$

holds for $j \geqslant -\tau_0$. Notice that for any $i, 1 \leqslant i \leqslant n$, (9.46) holds for $c = i$ and $j \geqslant -b_{i-1}$. The condition $j \geqslant -b_{i-1}$ is equivalent to the condition $j+\tau_i \geqslant -b_{i-1}+\tau_i$ which can be deduced by the condition $j+\tau_i \geqslant 0$ because of $-b_{i-1} + \tau_i \leqslant 0$. Thus for any $i, 1 \leqslant i \leqslant n$, (9.46) holds for $c = i$ and $j + \tau_i \geqslant 0$, that is,

$$u^{(i)}_{j+\tau_i+1} = g_{i,n}(u^{(i)}(j+\tau_{i,i}, p_{i,i}+1), u^{(i+1)}(j+\tau_{i,i+1}, p_{i,i+1}+1), \ldots, u^{(n)}(j+\tau_{i,n}, p_{i,n}+1), x^{(n)}(j+\tau_{i,n}, r_{i,n}+1))$$

holds for $j + \tau_i \geqslant 0$. From (9.49), it follows that (9.35) holds. Therefore, $s = \bar{s}$. □

Let $r'_{0,j} = \tau_0 + t_0 + r_1 + \cdots + r_j$ and $p'_{0,j} = \tau_0 + t_0 + r_1 + \cdots + r_{j-1} + p_j$, for $j > 0$. Let

$$\begin{aligned}
&f'_{0,n}(y(-1, r_0), u^{(0)}(0, p_0+1), u^{(1)}(0, p'_{0,1}+1), \ldots, \\
&\qquad u^{(n)}(0, p'_{0,n}+1), x^{(n)}(0, r'_{0,n}+1)) \\
&= f^*_0(y(-1, r_0), u^{(i)}(0, p_0+1), \\
&\quad f_{1,n}(u^{(1)}(0, p_{1,1}+1), u^{(2)}(0, p_{1,2}+1), \ldots, \\
&\qquad u^{(n)}(0, p_{1,n}+1), x^{(n)}(0, r_{1,n}+1)), \\
&\quad \cdots\cdots, \\
&\quad f_{1,n}(u^{(1)}(-\tau_0 - t_0, p_{1,1}+1), u^{(2)}(-\tau_0 - t_0, p_{1,2}+1), \ldots, \\
&\qquad u^{(n)}(-\tau_0 - t_0, p_{1,n}+1), x^{(n)}(-\tau_0 - t_0, r_{1,n}+1)))
\end{aligned}$$

and

$$\begin{aligned}
&g'_{0,n}(y(0, r_0+1), u^{(0)}(0, p_0+1), u^{(1)}(0, p'_{0,1}+1), \ldots, \\
&\qquad u^{(n)}(0, p'_{0,n}+1), x^{(n)}(0, r'_{0,n}+1)) \\
&= g_0(f_{1,n}(u^{(1)}(-\tau_0 - 1, p_{1,1}+1), u^{(2)}(-\tau_0 - 1, p_{1,2}+1), \ldots, \\
&\qquad u^{(n)}(-\tau_0 - 1, p_{1,n}+1), x^{(n)}(-\tau_0 - 1, r_{1,n}+1)), \\
&\quad \cdots\cdots, \\
&\quad f_{1,n}(u^{(1)}(-\tau_0 - t_0, p_{1,1}+1), u^{(2)}(-\tau_0 - t_0, p_{1,2}+1), \ldots, \\
&\qquad u^{(n)}(-\tau_0 - t_0, p_{1,n}+1), x^{(n)}(-\tau_0 - t_0, r_{1,n}+1)), \\
&\quad u^{(0)}(0, p_0+1),\ y(0, r_0+1)).
\end{aligned}$$

We use $M'_{0,n}$ to denote $C'(M_{1,n}, M^*_0)$, where symbols of the input alphabet and the output alphabet of M^*_0 are interchanged. Then we have

$$M'_{0,n} = \langle X_n, Y, Y^{r_0} \times U_0^{p_0+1} \times U_1^{p'_{0,1}+1} \times \ldots \times U_n^{p'_{0,n}+1} \times X_n^{r'_{0,n}}, \delta'_{0,n}, \lambda'_{0,n} \rangle,$$

where

$$\begin{aligned}
&\lambda'_{0,n}(\langle y(-1,r_0), u^{(0)}(0,p_0+1), u^{(1)}(0,p'_{0,1}+1),\dots,\\
&\qquad u^{(n)}(0,p'_{0,n}+1), x^{(n)}(-1,r'_{0,n})\rangle,\ x_0^{(n)})\\
&\quad = y_0,
\end{aligned}$$

$$\begin{aligned}
&\delta'_{0,n}(\langle y(-1,r_0), u^{(0)}(0,p_0+1), u^{(1)}(0,p'_{0,1}+1),\dots,\\
&\qquad u^{(n)}(0,p'_{0,n}+1), x^{(n)}(-1,r'_{0,n})\rangle, x_0^{(n)})\\
&\quad = \langle y(0,r_0), u^{(0)}(1,p_0+1), u^{(1)}(1,p'_{0,1}+1),\dots,\\
&\qquad u^{(n)}(1,p'_{0,n}+1), x^{(n)}(0,r'_{0,n})\rangle,
\end{aligned}$$

and

$$\begin{aligned}
y_0 &= f'_{0,n}(y(-1,r_0), u^{(0)}(0,p_0+1), u^{(1)}(0,p'_{0,1}+1),\dots,\\
&\qquad u^{(n)}(0,p'_{0,n}+1), x^{(n)}(0,r'_{0,n}+1)),\\
u_1^{(0)} &= g'_{0,n}(y(0,r_0+1), u^{(0)}(0,p_0+1), u^{(1)}(0,p'_{0,1}+1),\dots,\\
&\qquad u^{(n)}(0,p'_{0,n}+1), x^{(n)}(0,r'_{0,n}+1)),\\
u_1^{(i)} &= g_{i,n}(u^{(i)}(0,p_{i,i}+1),\dots,u^{(n)}(0,p_{i,n}+1), x^{(n)}(0,r_{i,n}+1)),\\
&\quad i=1,\dots,n.
\end{aligned}$$

Theorem 9.6.3. *Assume that M_i^*, M_i and τ_i satisfy $PI_2(M_i^*, M_i, \tau_i)$, $i=1,\dots,n$ and that M_0^*, M_0 and τ_0 satisfy $PI_1(M_0, M_0^*, \tau_0)$. Let $s_{-b_{i-1}}^{(i)*} = \langle x^{(i)}(-b_{i-1}-1, r_i),\ u^{(i)}(-b_{i-1}, p_i+1),\ \bar{x}^{(i-1)}(-b_{i-1}-1, \tau_i)\rangle$ be a state of M_i^*, $i=1,\dots,n$. Let $x_{-b_0}^{(0)},\dots,x_{-1}^{(0)} \in X_0$,*

$$\begin{aligned}
&x_{-b_{i-1}}^{(i)}\dots x_{-1}^{(i)} = \lambda_i^*(s_{-b_{i-1}}^{(i)*}, x_{-b_{i-1}}^{(i-1)}\dots x_{-1}^{(i-1)}),\\
&s_0^{(i)*} = \delta_i^*(s_{-b_{i-1}}^{(i)*}, x_{-b_{i-1}}^{(i-1)}\dots x_{-1}^{(i-1)}),\\
&\qquad i=1,\dots,n,
\end{aligned}$$

and

$$u_{j+1}^{(i)} = g_i(u^{(i)}(j,p_i+1), x^{(i)}(j,r_i+1))$$

for $i=1,\dots,n$ and $j=-b_{i-1},\dots,-1$, where $b_{-1} = \max\{0, -\sum_{j=1}^{i-1} r_j + \sum_{j=1}^{i}\tau_j,\ i=1,\dots,n\}$, $b_0 = b_{-1}+t_0$, and $b_i = b_{i-1}+r_i-\tau_i$ for $1\leqslant i\leqslant n$. Let $s_0^{(0)} = \langle x^{(0)}(-1,t_0),\ u^{(0)}(0,p_0+1),\ y(-1,r_0)\rangle$ be a state of M_0. If

$$\begin{aligned}
&x_0^{(0)}x_1^{(0)}\dots = \lambda_0(s_0^{(0)}, y_0y_1\dots),\\
&x_0^{(i)}x_1^{(i)}\dots = \lambda_i^*(s_0^{(i)*}, x_0^{(i-1)}x_1^{(i-1)}\dots),\\
&\qquad i=1,\dots,n,
\end{aligned}\tag{9.50}$$

then

$$y_0y_1\ldots = \lambda'_{0,n}(s, x^{(n)}_{\tau_{0,n}} x^{(n)}_{\tau_{0,n}+1}\ldots),$$

where $s = \langle y(-1, r_0), u^{(0)}(0, p_0+1), u^{(1)}(\tau_{0,1}, p'_{0,1}+1), \ldots, u^{(n)}(\tau_{0,n}, p'_{0,n}+1), x^{(n)}(\tau_{0,n}-1, r'_{0,n})\rangle$, *and* $u^{(i)}_j$ *in* s *for* $0 < j \leqslant \tau_{0,i}$ *are computed out according to the following formulae*

$$\begin{aligned} u^{(i)}_{j+1} = g_{i,n}(u^{(i)}(j, p_{i,i}+1), u^{(i+1)}(j+\tau_{i+1,i+1}, p_{i,i+1}+1), \ldots, \\ u^{(n)}(j+\tau_{i+1,n}, p_{i,n}+1), x^{(n)}(j+\tau_{i+1,n}, r_{i,n}+1)), \\ i = n, n-1, \ldots, 1, \quad j = 0, 1, \ldots, \tau_{0,i}-1. \end{aligned}$$

Proof. The proof of this theorem is similar to Theorem 9.6.2. Suppose that (9.50) holds. From the proof of Theorem 9.6.2, (9.47) and (9.48) hold. Since $PI_1(M_0, M_0^*, \tau_0)$ holds, from the part on M_0 in (9.50), we have

$$y_0y_1\ldots = \lambda^*_0(s^{(0)*}, x^{(0)}_{\tau_0} x^{(0)}_{\tau_0+1}\ldots), \tag{9.51}$$

where $s^{(0)*} = \langle y(-1, r_0), u^{(0)}(0, p_0+1), x^{(0)}(\tau_0-1, \tau_0+t_0)\rangle$. Using Theorem 9.6.1, from (9.47), (9.51) and (9.48), we obtain

$$y_0y_1\ldots = \lambda'_{0,n}(\bar{s}, x^{(n)}_{\tau_{0,n}} x^{(n)}_{\tau_{0,n}+1}\ldots),$$

where $\bar{s} = \langle y(-1, r_0), u^{(0)}(0, p_0+1), u^{(1)}(\tau_{0,1}, p'_{0,1}+1), \ldots, u^{(n)}(\tau_{0,n}, p'_{0,n}+1), x^{(n)}(\tau_{0,n}-1, r'_{0,n})\rangle$. From the proof of Theorem 9.6.2 (neglecting (9.49)), $\bar{s}$ coincides with s. □

Lemma 9.6.1. *Assume that* $PI_1(M_i, M_i^*, \tau_i)$ *holds for any* i, $1 \leqslant i \leqslant n$. *Let*

$$s = \langle u^{(1)}(0, p_1+1), u^{(2)}(0, p_{1,2}+1), \ldots, u^{(n)}(0, p_{1,n}+1), x^{(n)}(-1, r_{1,n})\rangle$$

be a state of $C'(M_n, \ldots, M_1)$. *Let*

$$\begin{aligned} x^{(c-1)}_j = f_{c,n}(u^{(c)}(j, p_{c,c}+1), \ldots, u^{(n)}(j, p_{c,n}+1), x^{(n)}(j, r_{c,n}+1)), \\ j = -r_{c-1}, \ldots, -1 \end{aligned} \tag{9.52}$$

for $c = 2, \ldots, n$. *If*

$$x^{(0)}_0 x^{(0)}_1\ldots = \lambda_{1,n}(s, x^{(n)}_0 x^{(n)}_1\ldots), \tag{9.53}$$

then

$$\begin{aligned} &x^{(n)}_0 x^{(n)}_1\ldots \\ &\quad = \lambda^*_n(\langle x^{(n)}(-1, r_n), u^{(n)}(0, p_n+1), x^{(n-1)}(\tau_n-1, \tau_n)\rangle, x^{(n-1)}_{\tau_n} x^{(n-1)}_{\tau_n+1}\ldots), \end{aligned} \tag{9.54}$$

where

$$x_0^{(i)}x_1^{(i)}\ldots \tag{9.55}$$
$$= \lambda_i^*(\langle x^{(i)}(-1,r_i), u^{(i)}(0,p_i+1), x^{(i-1)}(\tau_i-1,\tau_i)\rangle, x_{\tau_i}^{(i-1)}x_{\tau_i+1}^{(i-1)}\ldots)$$

for $i = 1,\ldots,n-1$.

Proof. Let $s_i = \langle u^{(i)}(0,p_i+1), x^{(i)}(-1,r_i)\rangle$ for $i = 1,\ldots,n-1$, and $s'_i = \langle u^{(i+1)}(0,p_{i+1}+1),\ u^{(i+2)}(0,p_{i+1,i+2}+1),\ \ldots,\ u^{(n)}(0,p_{i+1,n}+1),\ x^{(n)}(-1,r_{i+1,n})\rangle$ for $i = 0,1,\ldots,n-1$. Suppose that (9.53) holds. We prove by induction on i that

$$x_0^{(i-1)}x_1^{(i-1)}\ldots = \lambda_{i,n}(s'_{i-1}, x_0^{(n)}x_1^{(n)}\ldots) \tag{9.56}$$

holds for any i, $1 \leqslant i \leqslant n$. The case of $i = 1$ is trivial because of $s = s'_0$. Suppose that (9.56) holds for i and $i \leqslant n-1$. We prove (9.56) holds for $i+1$, that is,

$$x_0^{(i)}x_1^{(i)}\ldots = \lambda_{i+1,n}(s'_i, x_0^{(n)}x_1^{(n)}\ldots). \tag{9.57}$$

Since (9.52) holds for $c = i+1$, applying Theorem 9.6.1, the state s'_{i-1} of $C'(M_n,\ \ldots,\ M_i)$ and the state $\langle s'_i, s_i\rangle$ of $C(C'(M_n,\ldots,M_{i+1}), M_i)$ are equivalent. Letting

$$\bar{x}_0^{(i)}\bar{x}_1^{(i)}\ldots = \lambda_{i+1,n}(s'_i, x_0^{(n)}x_1^{(n)}\ldots), \tag{9.58}$$

from (9.56), we have

$$x_0^{(i-1)}x_1^{(i-1)}\ldots = \lambda_i(s_i, \bar{x}_0^{(i)}\bar{x}_1^{(i)}\ldots).$$

Since $PI_1(M_i, M_i^*, \tau_i)$ holds, the above equation deduces

$$\bar{x}_0^{(i)}\bar{x}_1^{(i)}\ldots = \lambda_i^*(\langle x^{(i)}(-1,r_i), u^{(i)}(0,p_i+1), x^{(i-1)}(\tau_i-1,\tau_i)\rangle, x_{\tau_i}^{(i-1)}x_{\tau_i+1}^{(i-1)}\ldots).$$

From (9.55), it follows immediately that $\bar{x}_0^{(i)}\bar{x}_1^{(i)}\ldots = x_0^{(i)}x_1^{(i)}\ldots$ Therefore, (9.58) implies (9.57).

Since $PI_1(M_n, M_n^*, \tau_n)$ holds, from the case $i = n$ of (9.56), (9.54) holds. □

Theorem 9.6.4. *Assume that* $PI_1(M_i, M_i^*, \tau_i)$ *holds for any* i, $0 \leqslant i \leqslant n$. *Let* $s = \langle y(-1,t_0),\ u^{(0)}(0,p_0+1),\ u^{(1)}(0,p_{0,1}+1),\ \ldots,\ u^{(n)}(0,p_{0,n}+1),\ x^{(n)}(-1,r_{0,n})\rangle$ *be a state of* $C'(M_n,\ \ldots,\ M_1,\ M_0)$. *Assume that* (9.52) *holds for* $c = 1,\ldots,n$ *and*

$$y_0y_1\ldots = \lambda_{0,n}(s, x_0^{(n)}x_1^{(n)}\ldots). \tag{9.59}$$

If

$$x_0^{(0)}x_1^{(0)}\ldots \tag{9.60}$$
$$= \lambda_0^*(\langle x^{(0)}(-1,r_0), u^{(0)}(0,p_0+1), y(\tau_0-1,\tau_0+t_0)\rangle, y_{\tau_0}y_{\tau_0+1}\ldots)$$

holds and (9.55) *holds for* $i = 1,\ldots,n-1$, *then* (9.54) *holds.*

Proof. Let $s_0 = \langle y(-1, t_0),\ u^{(0)}(0,\ p_0 + 1),\ x^{(0)}(-1, r_0)\rangle$ and $s'_0 = \langle u^{(1)}(0, p_1 + 1),\ u^{(2)}(0, p_{1,2} + 1),\ \ldots,\ u^{(n)}(0, p_{1,n} + 1),\ x^{(n)}(-1, r_{1,n})\rangle$. Since (9.52) holds for $c = 1$, the state s of $C'(M_n, \ldots, M_1, M_0)$ and the state $\langle s'_0, s_0\rangle$ of $C(C'(M_n, \ldots, M_1),\ M_0)$ are equivalent. Letting

$$\bar{x}_0^{(0)} \bar{x}_1^{(0)} \ldots = \lambda_{1,n}(s'_0, x_0^{(n)} x_1^{(n)} \ldots), \tag{9.61}$$

since (9.59) holds, we have

$$y_0 y_1 \ldots = \lambda_0(s_0, \bar{x}_0^{(0)} \bar{x}_1^{(0)} \ldots).$$

From $PI_1(M_0, M_0^*, \tau_0)$, it follows immediately that

$$\bar{x}_0^{(0)} \bar{x}_1^{(0)} \ldots = \lambda_0^*(\langle x^{(0)}(-1, r_0), u^{(0)}(0, p_0 + 1), y(\tau_0 - 1, \tau_0 + t_0)\rangle, y_{\tau_0} y_{\tau_0+1} \ldots).$$

Using (9.60), this yields $\bar{x}_0^{(0)} \bar{x}_1^{(0)} \ldots = x_0^{(0)} x_1^{(0)} \ldots$ From (9.61), we obtain

$$x_0^{(0)} x_1^{(0)} \ldots = \lambda_{1,n}(s'_0, x_0^{(n)} x_1^{(n)} \ldots).$$

From Lemma 9.6.1, we obtain (9.54). □

Theorem 9.6.5. *Assume that $PI_2(M_0^*, M_0, \tau_0)$ holds and $PI_1(M_i, M_i^*, \tau_i)$ holds for any i, $1 \leqslant i \leqslant n$. Let $s = \langle y(-1, r_0),\ u^{(0)}(0, p_0 + 1),\ u^{(1)}(0, p'_{0,1} + 1),\ \ldots,\ u^{(n)}(0, p'_{0,n} + 1),\ x^{(n)}(-1, r'_{0,n})\rangle$ be a state of $C'(M_n, \ldots, M_1, M_0^*)$. Assume that*

$$\begin{aligned} x_j^{(c-1)} &= f_{c,n}(u^{(c)}(j, p_{c,c} + 1), \ldots, u^{(n)}(j, p_{c,n} + 1), x^{(n)}(j, r_{c,n} + 1)), \\ &\quad j = -r'_{c-1}, \ldots, -1 \end{aligned} \tag{9.62}$$

for $c = 1, \ldots, n$ and

$$y_0 y_1 \ldots = \lambda'_{0,n}(s, x_0^{(n)} x_1^{(n)} \ldots), \tag{9.63}$$

where $r'_0 = \tau_0 + t_0$, $r'_i = r_i$, $i = 1, \ldots, n$. If

$$\begin{aligned} &x_0^{(0)} x_1^{(0)} \ldots = \lambda_0(\langle x^{(0)}(-1, t_0), u^{(0)}(\tau_0, p_0 + 1), y(\tau_0 - 1, r_0)\rangle, y_{\tau_0} y_{\tau_0+1} \ldots), \\ &u_{j+1}^{(0)} = g_0(x^{(0)}(j - \tau_0 - 1, t_0), u^{(0)}(j, p_0 + 1), y(j, r_0 + 1)), \\ &\quad j = 0, 1, \ldots, \tau_0 - 1 \end{aligned} \tag{9.64}$$

hold and (9.55) *holds for $i = 1, \ldots, n - 1$, then* (9.54) *holds.*

Proof. Let $s_0 = \langle y(-1, r_0),\ u^{(0)}(0, p_0 + 1),\ x^{(0)}(-1, \tau_0 + t_0)\rangle$ and $s'_0 = \langle u^{(1)}(0,\ p_1 + 1),\ u^{(2)}(0, p_{1,2} + 1),\ \ldots,\ u^{(n)}(0, p_{1,n} + 1),\ x^{(n)}(-1, r_{1,n})\rangle$. Since (9.62) holds for $c = 1$, the state s of $C'(M_n, \ldots, M_1, M_0^*)$ and the state $\langle s'_0, s_0\rangle$

of $C(C'(M_n, \ldots, M_1), M_0^*)$ are equivalent. Letting (9.61), from (9.63), we have

$$y_0 y_1 \ldots = \lambda_0^*(s_0, \bar{x}_0^{(0)} \bar{x}_1^{(0)} \ldots).$$

Since $PI_2(M_0^*, M_0, \tau_0)$ holds, it follows immediately that

$$\bar{x}_0^{(0)} \bar{x}_1^{(0)} \ldots = \lambda_0(\langle x^{(0)}(-1, t_0), u^{(0)}(\tau_0, p_0 + 1), y(\tau_0 - 1, r_0)\rangle, y_{\tau_0} y_{\tau_0+1} \ldots).$$

From (9.64), this yields $\bar{x}_0^{(0)} \bar{x}_1^{(0)} \ldots = x_0^{(0)} x_1^{(0)} \ldots$ Therefore, (9.61) implies

$$x_0^{(0)} x_1^{(0)} \ldots = \lambda_{1,n}(s_0', x_0^{(n)} x_1^{(n)} \ldots).$$

From Lemma 9.6.1, we obtain (9.54). □

9.6.2 Two Algorithms

FAPKC3x-*n*

Using the results of the preceding subsection, we now generalize the so-called basic algorithm of the public key cryptosysytem based on finite automata in Sect. 9.2 to two public key cryptosystems so that component finite automata of the compound finite automata in public keys are finite automata with auxiliary state discussed in the preceding subsection. The two cryptosystems are designed for both encryption and signature. Therefore, we require that all input alphabets and all output alphabets have the same dimension, say m.

We first propose a cryptosystem relied upon Theorem 9.6.2 and Theorem 9.6.4. Let $n \geqslant 1$. Choose a common q and m for all users. Let all the alphabets $X_0, \ldots, X_n$ and Y be the same column vector space over $GF(q)$ of dimension m.

A user, say A, choose his/her own public key and private key as follows.

(a) Construct pseudo-memory finite automata M_i, M_i^*, $i = 0, 1, \ldots, n$ defined by (9.30), (9.28), (9.31) and (9.29), respectively, which satisfy conditions $PI_1(M_i, M_i^*, \tau_i)$ and $PI_2(M_i^*, M_i, \tau_i)$ for some $\tau_i \leqslant r_i$, $i = 0, 1, \ldots, n$.

(b) Construct the finite automaton $C'(M_n, \ldots, M_1, M_0) = \langle X, Y, S, \delta_{0,n}, \lambda_{0,n}\rangle$ from M_0, M_1, …, M_n.

(c) Let $b_{-1} = \max\{\tau_0, -\sum_{j=0}^{i-1} r_j + \sum_{j=0}^{i} \tau_j, i = 1, \ldots, n\}$, and $b_i = b_{i-1} + r_i - \tau_i$, $i = 0, 1, \ldots, n$. Choose arbitrary $x_{-b_0}^{(0)}, \ldots, x_{-1}^{(0)} \in X_0$. For each i, $1 \leqslant i \leqslant n$, choose an arbitrary state

$$s_{-b_{i-1}}^{(i)*} = \langle x^{(i)}(-b_{i-1} - 1, r_i), u^{(i)}(-b_{i-1}, p_i + 1), \bar{x}^{(i-1)}(-b_{i-1} - 1, \tau_i)\rangle$$

of M_i^*. Compute

$$x_{-b_{i-1}}^{(i)} \dots x_{-1}^{(i)} = \lambda_i^*(s_{-b_{i-1}}^{(i)*}, x_{-b_{i-1}}^{(i-1)} \dots x_{-1}^{(i-1)}),$$
$$s_0^{(i)*} = \delta_i^*(s_{-b_{i-1}}^{(i)*}, x_{-b_{i-1}}^{(i-1)} \dots x_{-1}^{(i-1)}),$$
$$i = 1, \dots, n,$$

and

$$u_{j+1}^{(i)} = g_i(u^{(i)}(j, p_i + 1), x^{(i)}(j, r_i + 1)),$$
$$i = 1, \dots, n,\ j = -b_{i-1}, \dots, -1.$$

Choose an arbitrary state $s_0^{(0)*} = \langle x^{(0)}(-1, r_0),\ u^{(0)}(0, p_0+1),\ y(-1, \tau_0 + t_0)\rangle$ of M_0^*. Take $s_s^{(i)} = s_0^{(i)*}$, $i = 0, 1, \dots, n$, $s_v^{out} = \langle y_{-1}, \dots, y_{-\tau_0 - t_0}\rangle$, $s_v^{aux,i} = \langle u_0^{(i)}, u_{-1}^{(i)}, \dots, u_{-p^{(i)}}^{(i)}\rangle$, $i = 0, 1, \dots, n$, $s_v^{in} = \langle x_{-1}^{(n)}, \dots, x_{-r^{(n)}}^{(n)}\rangle$, where

$$r^{(n)} = \max(r_n,\ r_{n-1,n} - \tau_{n,n}, \dots, r_{1,n} - \tau_{2,n},\ r_{0,n} - \tau_{1,n}),$$
$$p^{(0)} = p_0,$$
$$p^{(i)} = \max(p_i,\ p_{i-1,i} - \tau_{i,i}, \dots, p_{1,i} - \tau_{2,i},\ p_{0,i} - \tau_{1,i}),\ i = 1, \dots, n.$$

(d) Choose an arbitrary state $s_e = \langle y(-1, t_0),\ u^{(0)}(0, p_0+1),\ u^{(1)}(0, p_{0,1}+1),\ \dots,\ u^{(n)}(0, p_{0,n}+1),\ x^{(n)}(-1, r_{0,n})\rangle$ of $C'(M_n, \dots, M_1, M_0)$. Compute

$$x_j^{(c-1)} = f_{c,n}(u^{(c)}(j, p_{c,c} + 1), \dots, u^{(n)}(j, p_{c,n} + 1), x^{(n)}(j, r_{c,n} + 1)),$$
$$c = 1, 2, \dots, n,\ j = -r_{c-1}, \dots, -1.$$

Take $s_d^{(0),in} = \langle y_{-1}, \dots, y_{-t_0}\rangle$, $s_d^{(i),aux} = \langle u_0^{(i)}, u_{-1}^{(i)}, \dots, u_{-p_i}^{(i)}\rangle$ and $s_d^{(i),out} = \langle x_{-1}^{(i)}, \dots, x_{-r_i}^{(i)}\rangle$, $i = 0, 1, \dots, n$.[1]

(e) The public key of the user A is

$$C'(M_n, \dots, M_1, M_0),\ s_v^{out}, s_v^{in}, s_v^{aux,0}, \dots, s_v^{aux,n},\ s_e,\ \tau_{0,n}.$$

The private key of the user A is

$$M_0^*, \dots, M_n^*,\ s_s^{(0)}, \dots, s_s^{(n)},\ s_d^{(0),in},\ s_d^{(0),aux}, \dots, s_d^{(n),aux},$$
$$s_d^{(0),out}, \dots, s_d^{(n),out}, \tau_0, \dots, \tau_n.$$

Encryption Any user, say B, wants to send a plaintext $x_0^{(n)} x_1^{(n)} \dots x_l^{(n)}$ to a user A. B first suffixes any $\tau_{0,n}$ digits, say $x_{l+1}^{(n)} \dots x_{l+\tau_{0,n}}^{(n)}$, to the plaintext. Then using $C'(M_n, \dots, M_1, M_0)$ and s_e in A's public key, B computes the ciphertext $y_0\ y_1\ \dots\ y_{l+\tau_{0,n}}$ as follows:

$$y_0 y_1 \dots y_{l+\tau_{0,n}} = \lambda_{0,n}(s_e, x_0^{(n)} x_1^{(n)} \dots x_{l+\tau_{0,n}}^{(n)}).$$

[1] For the simplicity of symbolization, we use the same symbols y_{-j}, $u_j^{(i)}$, $x_{-j}^{(i)}$ in (c) and in (d), but their intentions are different.

Decryption From the ciphertext $y_0y_1\dots y_{l+\tau_{0,n}}$, according to Theorem 9.6.4, A can retrieve the plaintext $x_0^{(n)}x_1^{(n)}\dots x_l^{(n)}$ as follows. Using $M_0^*,\dots,M_n^*$, $s_d^{(0),in}$ $s_d^{(i),aux}$, $s_d^{(i),out}$, $i=0,1,\dots,n$ in his/her private key, A computes

$$x_0^{(0)}x_1^{(0)}\dots x_{l+\tau_{1,n}}^{(0)} = \lambda_0^*(\langle x^{(0)}(-1,r_0),u^{(0)}(0,p_0+1),y(\tau_0-1,\tau_0+t_0)\rangle, y_{\tau_0}y_{\tau_0+1}\dots y_{l+\tau_{0,n}})$$

and

$$x_0^{(i)}x_1^{(i)}\dots x_{l+\tau_{i+1,n}}^{(i)} = \lambda_i^*(\langle x^{(i)}(-1,r_i),u^{(i)}(0,p_i+1),x^{(i-1)}(\tau_i-1,\tau_i)\rangle, x_{\tau_i}^{(i-1)}x_{\tau_i+1}^{(i-1)}\dots x_{l+\tau_{i,n}}^{(i-1)}),\quad i=1,\dots,n,$$

where $s_d^{(0),in}=\langle y_{-1},\dots,y_{-t_0}\rangle$, $s_d^{(i),aux}=\langle u_0^{(i)},u_{-1}^{(i)},\dots,u_{-p_i}^{(i)}\rangle$ and $s_d^{(i),out}=\langle x_{-1}^{(i)},\dots,x_{-r_i}^{(i)}\rangle$, $i=0,1,\dots,n$.

Signature To sign a message $y_0y_1\dots y_l$, the user A first suffixes any $\tau_{0,n}$ digits, say $y_{l+1}\dots y_{l+\tau_{0,n}}$, to the message. Then using $M_0^*,\dots,M_n^*$, $s_s^{(0)},\dots,s_s^{(n)}$ in his/her private key, A computes

$$x_0^{(0)}x_1^{(0)}\dots x_{l+\tau_{0,n}}^{(0)} = \lambda_0^*(s_s^{(0)},y_0y_1\dots y_{l+\tau_{0,n}}),$$
$$x_0^{(i)}x_1^{(i)}\dots x_{l+\tau_{0,n}}^{(i)} = \lambda_i^*(s_s^{(i)},x_0^{(i-1)}x_1^{(i-1)}\dots x_{l+\tau_{0,n}}^{(i-1)}),\quad i=1,\dots,n.$$

Then $x_0^{(n)}x_1^{(n)}\dots x_{l+\tau_{0,n}}^{(n)}$ is a signature of $y_0y_1\dots y_l$.

Validation Any user, say B, can verify the validity of the signature $x_0^{(n)}x_1^{(n)}\dots x_{l+\tau_{0,n}}^{(n)}$ as follows. Using $C'(M_n,\dots,M_1,M_0)$, $s_v^{out},s_v^{in},s_v^{aux,i}$, $i=0,1,\dots,n$ in A's public key, B first computes

$$u_{j+1}^{(i)} = g_{i,n}(u^{(i)}(j,p_{i,i}+1),u^{(i+1)}(j+\tau_{i+1,i+1},p_{i,i+1}+1),\dots,$$
$$u^{(n)}(j+\tau_{i+1,n},p_{i,n}+1),x^{(n)}(j+\tau_{i+1,n},r_{i,n}+1)),$$
$$j=0,1,\dots,\tau_{0,i}-1,\quad i=n,n-1,\dots,1,$$
$$u_{j+1}^{(0)} = g_{0,n}(y(j-\tau_0-1,t_0),u^{(0)}(j,p_{0,0}+1),u^{(1)}(j+\tau_{1,1},p_{0,1}+1),\dots,$$
$$u^{(n)}(j+\tau_{1,n},p_{0,n}+1),x^{(n)}(j+\tau_{1,n},r_{0,n}+1)),$$
$$j=0,1,\dots,\tau_{0,0}-1,$$

where $s_v^{out}=\langle y_{-1},\dots,y_{-\tau_0-t_0}\rangle$, $s_v^{aux,i}=\langle u_0^{(i)},u_{-1}^{(i)},\dots,u_{-p^{(i)}}^{(i)}\rangle$, $i=0,1,\dots,n$, $s_v^{in}=\langle x_{-1}^{(n)},\dots,x_{-r^{(n)}}^{(n)}\rangle$. Letting $s=\langle y(-1,t_0),\ u^{(0)}(\tau_{0,0},p_{0,0}+1),\ u^{(1)}(\tau_{0,1},p_{0,1}+1),\dots,\ u^{(n)}(\tau_{0,n},p_{0,n}+1),\ x^{(n)}(\tau_{0,n}-1,r_{0,n})\rangle$, A then computes

$$\lambda_{0,n}(s, x^{(n)}_{\tau_{0,n}} x^{(n)}_{\tau_{0,n}+1} \dots x^{(n)}_{l+\tau_{0,n}})$$

which would coincide with the message $y_0 y_1 \dots y_l$ from Theorem 9.6.2.

The special case of $n = 1$ of the above cryptosystem may be regarded as a generation of the cryptsystem FAPKC3 (cf. [123]). The cryptosystem is referred to as FAPKC3x-n.

FAPKC4x-*n*

Similar to FAPKC3x-n, we propose a cryptosystem relied upon Theorem 9.6.3 and Theorem 9.6.5. Let $n \geqslant 1$. Choose a common q and m for all users. Let all the alphabets $X_0, \dots, X_n$ and Y be the same column vector space over $GF(q)$ of dimension m.

A user, say A, choose his/her own public key and private key as follows.

(a) Construct pseudo-memory finite automata M_i, M_i^*, $i = 0, 1, \dots, n$ defined by (9.30), (9.28), (9.31) and (9.29), respectively, which satisfy conditions $PI_1(M_i, M_i^*, \tau_i)$ and $PI_2(M_i^*, M_i, \tau_i)$ for some $\tau_i \leqslant r_i$, $i = 0, 1, \dots, n$.

(b) Construct the finite automaton $C'(M_n, \dots, M_1, M_0^*) = \langle X, Y, S', \delta'_{0,n}, \lambda'_{0,n} \rangle$ from M_0^*, M_1, …, M_n.

(c) Let $b_{-1} = \max\{0, -\sum_{j=0}^{i-1} r_j + \sum_{j=0}^{i} \tau_j, i = 1, \dots, n\}$, $b_0 = b_{-1} + t_0$, and $b_i = b_{i-1} + r_i - \tau_i$, $i = 1, \dots, n$. Choose arbitrary $x^{(0)}_{-b_0}, \dots, x^{(0)}_{-1} \in X_0$. For each i, $1 \leqslant i \leqslant n$, choose an arbitrary state $s^{(i)*}_{-b_{i-1}} = \langle x^{(i)}(-b_{i-1} - 1, r_i), u^{(i)}(-b_{i-1}, p_i + 1), \bar{x}^{(i-1)}(-b_{i-1} - 1, \tau_i) \rangle$ of M_i^*. Compute

$$\begin{aligned} x^{(i)}_{-b_{i-1}} \dots x^{(i)}_{-1} &= \lambda_i^*(s^{(i)*}_{-b_{i-1}}, x^{(i-1)}_{-b_{i-1}} \dots x^{(i-1)}_{-1}), \\ s^{(i)*}_0 &= \delta_i^*(s^{(i)*}_{-b_{i-1}}, x^{(i-1)}_{-b_{i-1}} \dots x^{(i-1)}_{-1}), \\ & i = 1, \dots, n, \end{aligned}$$

and

$$\begin{aligned} u^{(i)}_{j+1} &= g_i(u^{(i)}(j, p_i + 1), x^{(i)}(j, r_i + 1)) \\ & i = 1, \dots, n, \ j = -b_{i-1}, \dots, -1. \end{aligned}$$

Choose an arbitrary state $s^{(0)}_0 = \langle x^{(0)}(-1, t_0), u^{(0)}(0, p_0 + 1), y(-1, r_0) \rangle$ of M_0. Take $s^{(0)}_s = s^{(0)}_0$, $s^{(i)}_s = s^{(i)*}_0$, $i = 1, \dots, n$, $s^{out}_v = \langle y_{-1}, \dots, y_{-r_0} \rangle$, $s^{aux,i}_v = \langle u^{(i)}_0, u^{(i)}_{-1}, \dots, u^{(i)}_{-p^{(i)}} \rangle$, $i = 0, 1, \dots, n$, $s^{in}_v = \langle x^{(n)}_{-1}, \dots, x^{(n)}_{-r^{(n)}} \rangle$, where

$$\begin{aligned} r^{(n)} &= \max(r_n, \ r_{n-1,n} - \tau_{n,n}, \dots, r_{1,n} - \tau_{2,n}, \ r'_{0,n} - \tau_{0,n}), \\ p^{(0)} &= p_0, \\ p^{(i)} &= \max(p_i, \ p_{i-1,i} - \tau_{i,i}, \dots, p_{1,i} - \tau_{2,i}, \ p'_{0,i} - \tau_{0,i}), \ i = 1, \dots, n. \end{aligned}$$

(d) Choose an arbitrary state $s_e = \langle y(-1, r_0),\, u^{(0)}(0, p_0+1),\, u^{(1)}(0, p'_{0,1}+1),\, \ldots,\, u^{(n)}(0, p'_{0,n}+1),\, x^{(n)}(-1, r'_{0,n})\rangle$ of $C'(M_n, \ldots, M_1, M_0^*)$. Compute

$$x_j^{(c-1)} = f_{c,n}(u^{(c)}(j, p_{c,c}+1), \ldots, u^{(n)}(j, p_{c,n}+1), x^{(n)}(j, r_{c,n}+1)),$$
$$c = 1, 2, \ldots, n,\ j = -r'_{c-1}, \ldots, -1,$$

where $r'_0 = \tau_0 + t_0$, $r'_i = r_i$, $i = 1, \ldots, n$. Take $s_d^{(0),in} = \langle y_{-1}, \ldots, y_{-r_0}\rangle$, $s_d^{(i),aux} = \langle u_0^{(i)}, u_{-1}^{(i)}, \ldots, u_{-p_i}^{(i)}\rangle$, $i = 0, 1, \ldots, n$, $s_d^{(0),out} = \langle x_{-1}^{(0)}, \ldots, x_{-\tau_0-t_0}^{(0)}\rangle$, and $s_d^{(i),out} = \langle x_{-1}^{(i)}, \ldots, x_{-r_i}^{(i)}\rangle$, $i = 1, \ldots, n$.[1]

(e) The public key of the user A is

$$C'(M_n, \ldots, M_1, M_0^*),\ s_v^{out}, s_v^{in}, s_v^{aux,0}, \ldots, s_v^{aux,n},\ s_e,\ \tau_{0,n}.$$

The private key of the user A is

$$M_0, M_1^*, \ldots, M_n^*,\ s_s^{(0)}, \ldots, s_s^{(n)},\ s_d^{(0),in},\ s_d^{(0),aux}, \ldots, s_d^{(n),aux},$$
$$s_d^{(0),out}, \ldots, s_d^{(n),out}, \tau_0, \ldots, \tau_n.$$

Encryption Any user, say B, wants to send a plaintext $x_0^{(n)}x_1^{(n)}\ldots x_l^{(n)}$ to a user A. B first suffixes any $\tau_{0,n}$ digits, say $x_{l+1}^{(n)}\ldots x_{l+\tau_{0,n}}^{(n)}$, to the plaintext. Then using $C'(M_n, \ldots, M_1, M_0^*)$ and s_e in A's public key, B computes the ciphertext $y_0\ y_1\ \ldots\ y_{l+\tau_{0,n}}$ as follows:

$$y_0 y_1 \ldots y_{l+\tau_{0,n}} = \lambda'_{0,n}(s_e, x_0^{(n)}x_1^{(n)}\ldots x_{l+\tau_{0,n}}^{(n)}).$$

Decryption From the ciphertext $y_0 y_1 \ldots y_{l+\tau_{0,n}}$, according to Theorem 9.6.5, A can retrieve the plaintext $x_0^{(n)}x_1^{(n)}\ldots x_l^{(n)}$ as follows. Using $M_0, M_1^*, \ldots, M_n^*$, $s_d^{(0),in}$ $s_d^{(i),aux}$, $s_d^{(i),out}$, $i = 0, 1, \ldots, n$ in his/her private key, A computes

$$u_{j+1}^{(0)} = g_0(x^{(0)}(j-\tau_0-1, t_0), u^{(0)}(j, p_0+1), y(j, r_0+1)),\quad j = 0, \ldots, \tau_0 - 1,$$
$$x_0^{(0)}x_1^{(0)}\ldots x_{l+\tau_{1,n}}^{(0)}$$
$$= \lambda_0(\langle x^{(0)}(-1, t_0), u^{(0)}(\tau_0, p_0+1), y(\tau_0-1, r_0)\rangle, y_{\tau_0}y_{\tau_0+1}\ldots y_{l+\tau_{0,n}})$$

and

$$x_0^{(i)}x_1^{(i)}\ldots x_{l+\tau_{i+1,n}}^{(i)}$$
$$= \lambda_i^*(\langle x^{(i)}(-1, r_i), u^{(i)}(0, p_i+1), x^{(i-1)}(\tau_i-1, \tau_i)\rangle, x_{\tau_i}^{(i-1)}x_{\tau_i+1}^{(i-1)}\ldots x_{l+\tau_{i,n}}^{(i-1)}),$$
$$i = 1, \ldots, n,$$

where $s_d^{(0),in} = \langle y_{-1}, \ldots, y_{-r_0}\rangle$, $s_d^{(i),aux} = \langle u_0^{(i)}, u_{-1}^{(i)}, \ldots, u_{-p_i}^{(i)}\rangle$, $i = 0, 1, \ldots, n$, $s_d^{(0),out} = \langle x_{-1}^{(0)}, \ldots, x_{-\tau_0-t_0}^{(0)}\rangle$ and $s_d^{(i),out} = \langle x_{-1}^{(i)}, \ldots, x_{-r_i}^{(i)}\rangle$, $i = 1, \ldots, n$.

[1] For the simplicity of symbolization, we use the same symbols y_{-j}, $u_j^{(i)}$, $x_{-j}^{(i)}$ in (c) and in (d), but their intentions are different.

Signature To sign a message $y_0y_1 \dots y_l$, the user A first suffixes any $\tau_{0,n}$ digits, say $y_{l+1} \dots y_{l+\tau_{0,n}}$, to the message. Then using M_0, M_1^*, …, M_n^*, $s_s^{(0)}, \dots, s_s^{(n)}$ in his/her private key, A computes

$$\begin{aligned} x_0^{(0)}x_1^{(0)} \dots x_{l+\tau_{0,n}}^{(0)} &= \lambda_0(s_s^{(0)}, y_0y_1 \dots y_{l+\tau_{0,n}}), \\ x_0^{(i)}x_1^{(i)} \dots x_{l+\tau_{0,n}}^{(i)} &= \lambda_i^*(s_s^{(i)}, x_0^{(i-1)}x_1^{(i-1)} \dots x_{l+\tau_{0,n}}^{(i-1)}), \\ & \quad i = 1, \dots, n. \end{aligned}$$

Then $x_0^{(n)}x_1^{(n)} \dots x_{l+\tau_{0,n}}^{(n)}$ is a signature of $y_0y_1 \dots y_l$.

Validation Any user, say B, can verify the validity of the signature $x_0^{(n)}\ x_1^{(n)}\ \dots\ x_{l+\tau_{0,n}}^{(n)}$ as follows. Using $C'(M_n, \dots, M_1, M_0^*)$, $s_v^{out}, s_v^{in}, s_v^{aux,i}$, $i = 0, 1, \dots, n$ in A's public key, B first computes

$$\begin{aligned} u_{j+1}^{(i)} = g_{i,n}(&u^{(i)}(j, p_{i,i} + 1), u^{(i+1)}(j + \tau_{i+1,i+1}, p_{i,i+1} + 1), \dots, \\ &u^{(n)}(j + \tau_{i+1,n}, p_{i,n} + 1), x^{(n)}(j + \tau_{i+1,n}, r_{i,n} + 1)), \\ & i = n, n-1, \dots, 1, \quad j = 0, 1, \dots, \tau_{0,i} - 1, \end{aligned}$$

where $s_v^{aux,i} = \langle u_0^{(i)}, u_{-1}^{(i)}, \dots, u_{-p^{(i)}}^{(i)} \rangle$, $i = 0, 1, \dots, n$, $s_v^{in} = \langle x_{-1}^{(n)}, \dots, x_{-r^{(n)}}^{(n)} \rangle$. Letting $s = \langle y(-1, r_0), u^{(0)}(0, p_0 + 1), u^{(1)}(\tau_{0,1}, p'_{0,1} + 1), \dots, u^{(n)}(\tau_{0,n}, p'_{0,n} + 1), x^{(n)}(\tau_{0,n} - 1, r'_{0,n}) \rangle$, A then computes

$$\lambda'_{0,n}(s, x_{\tau_{0,n}}^{(n)} x_{\tau_{0,n}+1}^{(n)} \dots x_{l+\tau_{0,n}}^{(n)})$$

which would coincide with the message $y_0y_1 \dots y_l$ from Theorem 9.6.3, where $s_v^{out} = \langle y_{-1}, \dots, y_{-r_0} \rangle$.

The special case of $n = 1$ of the above cryptosystem may be regarded as a generation of the cryptsystem FAPKC4 (cf. [125]). The cryptosystem is referred to as FAPKC4x-n.

Historical Notes

Since introducing the concept of public key cryptosystems by Diffie and Hellman [32], many concrete block cryptosystems are proposed in [89, 74, 72, 34, 50, 76, 63, 90, 93, 1]. A sequential public key cryptosystem based on finite automata, referred to as FAPKC0, is given in [112] of which a public key contains a compound finite automaton of an invertible linear (τ, τ)-order memory finite automaton with delay τ and a weakly invertible nonlinear input-memory finite automaton with delay 0. Two other schemes, referred to as FAPKC1 and FAPKC2, are given in [113], where a public key for FAPKC1

contains a compound finite automaton of an linear inverse τ-order input-memory finite automaton with delay τ and a weakly invertible nonlinear input-memory finite automaton with delay 0. Reference [26] first proves that FAPKC0 is insecure in encryption. Reference [11] proves that FAPKC1 is insecure in encryption, and proposes a modification using quasi-linear finite automata, which is shown to be insecure in both encryption and signature in [108]. From [107], FAPKC0 and FAPKC1 are insecure in both encryption and signature. In [118] a method for generating a kind of nonlinear weakly invertible finite automata is developed; then two schemes, called FAPKC3 and FAPKC4, are proposed in [131, 122]. References [123, 125, 126] give some generalization of FAPKC3 and FAPKC4. In [45, 22], some schemes of public key cryptosystems based on finite automata are also proposed. Further works on security of public key cryptosystems based on finite automata can be found in [83, 108, 135, 137, 28, 132, 121, 109, 30, 110, 8, 82, 128]. In this chapter, Sects. 9.1 and 9.2 are based on [126]. Section 9.4 is in part based on [108, 83, 137, 28, 132, 121]. Section 9.5 is in part based on [45, 128]. And Sect. 9.6 is a further generalization of [123, 125] in respect of the total number of component automata.

References

1. M. Ajtai and C. Dwork, A public-key cryptosystem with worst-case/average-case equivalence, in *Proceedings of the Twenty-ninth Annual ACM Symposium on the Theory of Computing*, Association for Computing Machinery, 1997, 284–293.
2. S. V. Aleshin, Finite automata and the Burnside problem for torsion groups, *Mathematical Notes*, **29**(1972), 319–328. (in Russian)
3. D. D. Aufenkamp, Analysis of sequential machines II, *IRE Transactions on Electronic Computers*, **7**(1958), 299–306.
4. F. Bao, *On the Structure of n-ary Feedforward Inverses with Delay* 1, MA Thesis, Institute of Software, Chinese Academy of Sciences, Beijing, 1986. (in Chinese)
5. F. Bao, Limited error-propagation, self-synchronization and finite input-memory FSMs as weak inverses, in *Advances in Chinese Computer Science*, Vol. 3, World Scientific, Singapore, 1991, 1–24.
6. F. Bao, Composition and decomposition of weakly invertible finite automata, *Science in China*, Ser. A, **23** (1993), 759–765. (in Chinese)
7. F. Bao, Two results about the decomposition of delay step of weakly invertible finite automata, *Chinese Jounrnal of Computers*, **16**(1993), 629–632. (in Chinese)
8. F. Bao, Increasing ranks of linear finite automata and complexity of FA public key cryptosystem, *Science in China*, Ser. A, **37**(1994), 504–512.
9. F. Bao, R. H. Deng, X. Gao, and Y. Igarashi, Modified finite automata public key cryptosystem, in *Proceedings of the First International Workshop on Information Security*, Lecture Notes in Computer Science 1396, Springer-Verlag, Berlin, 1997, 82–95.
10. F. Bao and Y. Igarashi, A randomized algorithm to finite automata public key cryptosystem, in *Proceedings of the Fifth International Symposium on Algorithms and Computation*, Lecture Notes in Computer Science 834, Springer-Verlag, Berlin, 1994, 678–686.
11. F. Bao and Y. Igarashi, Break finite automata public key cryptosystem, in *Automata, Languages and Programming*, Lecture Notes in Computer Science 944, Springer-Verlag, Berlin, 1995, 147–158.
12. F. Bao, Y. Igarashi, and X. Yu, Some results on decomposition of weakly invertible finite automata, *IEICE Transactions on Information and systems*, **E79-D**(1996), 1–7.
13. C. Berge, *Théorie des Graphes et ses Applications*, Dunod, Paris, 1958.
14. E. Berlecamp, *Algebraic Coding Theory*, McGraw-Hill Book Co., New York, 1968.
15. E. Berlecamp, Factoring polynomial over large finite fields, *Mathematics of Computation*, **24**(1970), 713–735.

16. G. Birkhoff and S. MacLane, *A Survey of Modern Algebra*, Macmillan Publishing Co., Inc., New York, 1977.
17. S. H. Chen, On the structure of weak inverses of a weakly invertible linear finite automaton, *Chinese Jounrnal of Computers*, **4**(1981), 409–419. (in Chinese)
18. S. H. Chen, On the structure of finite automata of which M' is an (weak) inverse with delay τ, *Jounrnal of Computer Science and Technology*, **1**(1986), 54–59.
19. S. H. Chen, On the structure of (weak) inverses of an (weakly) invertible finite automaton, *Jounrnal of Computer Science and Technology*, **1**(1986), 92–100.
20. S. H. Chen and R. J. Tao, The structure of weak inverses of a finite automaton with bounded error propagation, *Kexue Tongbao*, **32**(1987), 713–714; full paper in *Advances in Chinese Computer Science*, Vol.**1**, World Scientific, Singapore, 1988, 205–211.
21. S. H. Chen and R. J. Tao, Invertibility of quasi-linear finite automata, in *Advances in Cryptology – CHINACRYPT'92*, Science Press, Beijing, 1992, 77–86. (in Chinese)
22. Xiaoming Chen, *The Invertibility Theory and Application of Quadratic Finite Automata*, Ph. D. Thesis, Institute of Software, Chinese Academy of Sciences, Beijing, 1996. (in Chinese)
23. H. J. Chizech, Inverse of finite group systems, *IEEE Transactions on Automatic Control*, **23** (1978), 66–70.
24. M. Cohn, Properties of linear machines, *Jounrnal of the Association for Computing Machinery*, **11**(1964), 296–301.
25. M. Cohn and S. Even, Identification and minimization of linear machines, *IEEE Transactions on Electronic Computers*, **14**(1965), 367–376.
26. D. W. Dai, K. Wu, and H. G. Zhang, Cryptanalysis on a finite automaton public key cryptosystem, *Science in China*, Ser. A, **39** (1996), 27–36.
27. Z. D. Dai, Invariants and invertibility of linear finite automata, in *Advances in Cryptology – CHINACRYPT'94*, Science Press, Beijing, 1994, 127–134. (in Chinese)
28. Z. D. Dai, A class of separable nonlinear finite automata – and an analysis of a certain typed FA based public key encryption and signature scheme, in *Advances in Cryptology – CHINACRYPT'96*, Science Press, Beijing, 1996, 87–94. (in Chinese)
29. Z. D. Dai and D. F. Ye, Weak invertibility of nonlinear finite automata over commutative rings, *Chinese Science Bulletin*, **40**(1995), 1357–1360. (in Chinese)
30. Z. D. Dai, D. F. Ye and K. Y. Lam, Weak Invertibility of Finite Automata and Cryptanalysis on FAPKC, in *Advances in Cryptology – ASIACRYPT'98*, Lecture Notes in Computer Science 1514, Springer Verlag, Berlin, 1998, 227–241.
31. J. Dénes and A. D. Keedwell, *Latin Squares and Their Applications*, Akadémiai Kiadó, Budapest, 1974.
32. W. Diffie and M. E. Hellman, New directions in cryptography, *IEEE Transactions on Information Theory*, **22** (1976), 644–654.
33. A. Ecker, Abstrakte Kryptographische Maschinen, *Angewandte Informatik*, **17** (1975), 201–205.
34. T. ElGamal, A public key cryptosystem and a signature scheme based on discrete logarithms, *IEEE Transactions on Information Theory*, **31**(1985), 469–472.
35. B. Elspas, The theory of autonomous linear sequential networks, *IRE Transactions on Circuit Theory*, **6**(1959), 45–60.

36. S. Even, On information lossless automata of finite order, *IEEE Transactions on Electronic Computers*, **14**(1965), 561–569.
37. P. R. Feng, *Finite Automaton Cryptosystems – Theoretical Research, Application and Implementation*, Ph. D. Thesis, Institute of Software, Chinese Academy of Sciences, Beijing, 1996. (in Chinese)
38. P. R. Feng and R. J. Tao, Some results on a kind of classification and enumeration of matrices, in *Advances in Cryptology – CHINACRYPT'94*, Science Press, Beijing, 1994, 214–224. (in Chinese)
39. G. D. Forney, Convolution codes I: algebraic structure, *IEEE Transactions on Information Theory*, **16**(1970), 720–738.
40. B. Friedland, Linear modular sequential circuits, *IRE Transactions on Circuit Theory*, **6**(1959), 61–68.
41. K. Fukunaga, A theory of linear sequential nets using z transforms, *IEEE Transactions on Electronic Computers*, **13**(1964), 310–312.
42. F. R. Gantmacher, *The Theory of Matrices*, Chelsea Publishing Company, New York, 1959.
43. X. Gao, A random (n, k)-Latin array generator and an algorithm for generating the representatives of (n, k)-Latin array's isotopy classes, *Advances in Cryptology – CHINACRYPT'92*, Science Press, Beijing, 1992, 209–215. (in Chinese)
44. X. Gao, An algorithm for constructing a class of higher dimension linear independent Latin arrays from lowers', *Advances in Cryptology – CHINACRYPT'94*, Science Press, Beijing, 1994, 232–237. (in Chinese)
45. X. Gao, *Finite Automaton Public Key Cryptosystems and Digital Signatures – Analysis, Design and Implementation*, Ph. D. Thesis, Institute of Software, Chinese Academy of Sciences, Beijing, 1994. (in Chinese)
46. X. Gao and F. Bao, Decomposition of binary weakly invertible finite automata, *Chinese Jounrnal of Computers*, **17**(1994), 330–337. (in Chinese)
47. A. Gill, The minimization of linear sequential circuits, *IEEE Transactions on Circuit Theory*, **12**(1965), 292–294.
48. S. Ginsburg, On the reduction of superfluous states in a sequential machine, *Jounrnal of the Association for Computing Machinery*, **6**(1959), 259–282.
49. I. Gohberg, P. Lancaster and L. Rodman, *Matrix Polynomials*, Academic Press, New York, 1982.
50. S. Goldwasser and S. Micali, Probabilistic encryption, *Jounrnal of Computer and System Science*, **28**(1984), 270–299.
51. S. W. Golomb, *Shift Register Sequences*, Revised Edition, Aegean Park Press, Laguna Hills, California, 1982.
52. E. J. Groth, Generation of binary sequences with controllable complexity, *IEEE Transactions on Information Theory*, **17**(1971), 288–296.
53. M. Gysin, A one key cryptosystem based on a finite nonlinear automaton, in *Proceedings of the International Conference on Cryptography: Policy and Algorithms*, Lecture Notes in Computer Science 1029, Springer-Verlag, Berlin, 1995, 165–173.
54. M. Hall, *Combinatorial Theory*, Blaisdell Publishing Company, London, 1967.
55. X. X. Han and G. Yao, The combined use of FAPKC without compromising the security of the cryptosystem, *Research and Development of Computers*, **42**(2005), 1692–1697. (in Chinese)
56. M. Q. He, *Enumeration of Involutive Latin Arrays*, MA Thesis, Institute of Software, Chinese Academy of Sciences, Beijing, 1991. (in Chinese)
57. T. Herlestam, On functions of linear shift register sequences, *Advances in Cryptology – EUROCRYPTO'85*, Lecture Notes in Computer Science 219, Springer-Verlag, Berlin, 1986, 119–129.

58. D. A. Huffman, The synthesis of sequential switching circuits, *Jounrnal of the Franklin Institute*, **257**(1954), 161–190, 275–303.
59. D. A. Huffman, The synthesis of linear sequential coding networks, in *Informationn Theory*, Academic Press Inc., New York, 1956, 77–95.
60. D. A. Huffman, Canonical forms for information-lossless finite-state logical machines, *IRE Transactions on Circuit Theory*, **6**(1959), special supplement, 41–59.
61. Y. Kambayashi and S. Yajima, The upper bound of k in k-lossless sequential machines, *Information and Control*, **19**(1971), 432–438.
62. S. C. Kleene, Representation of events in nerve nets and finite automata, in *Automata Studies*, Annals of Mathematical Studies 34, Princeton University Press, Princeton, 1956, 3–41.
63. N. Koblitz, Elliptic curve cryptosystems, *Mathematics of Computation*, **48**(1987), 203–209.
64. A. A. Kurmit, *Information Lossless Automata of Finite Order*, John Wiley, New York, 1974.
65. J. B. Li, *A Software Implementations of the Finite Automaton Public Key Cryptosystem and Digital Signature and Its Applications*, MA Thesis, Institute of Software, Chinese Academy of Sciences, Beijing, 1994. (in Chinese)
66. R. Lidl and H. Niederreiter, *Finite Fields*, Addison-Wesley, London, 1983.
67. Y. H. Long, Some results on the number of a class of linearly independent Latin arrays, in *Advances in Cryptology – CHINACRYPT'94*, Science Press, Beijing, 1994, 238–244. (in Chinese)
68. Y. H. Long, *Latin Arrays, Permutation Polynomials and Their Applications in 2NO Stream Ciphers*, Ph. D. Thesis, Institute of Software, Chinese Academy of Sciences, Beijing, 1995. (in Chinese)
69. Y. H. Long, The design of Latin arrays used in a class of stream cipher, *Chinese Jounrnal of Computers*, **19**(1996), 247–253. (in Chinese)
70. S. Z. Lü, Some results on the invertibility of linear finite automata over a ring, *Chinese Jounrnal of Computers*, **14**(1991), 570–578. (in Chinese)
71. J. L. Massey and M. K. Sain, Inverse of linear sequential circuits, *IEEE Transactions on Computers*, **17**(1968), 330–337.
72. R. J. McEliece, A public-key cryptosystem based on algebraic coding theory, DSN Progress Report, 42–44, 1978.
73. G. H. Mealy, A method for synthesizing sequential circuits, *The Bell System Technical Journal*, **34**(1955), 1045–1079.
74. R. C. Merkle and M. E. Hellman, Hiding information and signatures in trapdoor knapsacks, *IEEE Transactions on Information Theory*, **24**(1978), 525–530.
75. T. Meskanen, On finite automaton public key cryptosystems, Technical Report No. 408, Turku Centre for Computer Science, Turku, August 2001.
76. V. Miller, Uses of elliptic curves in cryptography, *Advances in Cryptology – CRYPTO'85*, Lecture Notes in Computer Science 218, Springer-Verlag, Berlin, 1986, 417–426.
77. B. C. Moore and L. M. Siverman, A new characterization of feedforward delay-free inverses, *IEEE Transactions on Information Theory*, **19**(1973), 126–129.
78. E. F. Moore, Gedunken-experiments on sequential machines, in *Automata Studies*, Annals of Mathematical Studies 34, Princeton University Press, Princeton, 1956, 129–153.
79. R. R. Olson, Note on feedforward inverses for linear sequential circuits, *IEEE Transactions on Computers*, **19**(1970), 1216–1221.

80. M. C. Paull and S. H. Unger, Minimizing the number of states in incompletely specified sequential functions, *IRE Transactions on Electronic Computers*, **8**(1959), 356–367.
81. Y. L. Qi, S. H. Chen, and R. J. Tao, A finite automaton cryptosystem and its software implementation, *Computer Research and Development*, **24**(1987), 6–14. (in Chinese)
82. Z. P. Qin and H. G. Zhang, Enumeration of sequences in finite automata with application to cryptanalysis, in *Advances in Cryptology – CHINACRYPT'94*, Science Press, Beijing, 1994, 112–119. (in Chinese)
83. Z. P. Qin and H. G. Zhang, Cryptanalysis of finite automaton public key cryptosystems, in *Advances in Cryptology – CHINACRYPT'96*, Science Press, Beijing, 1996, 75–86. (in Chinese)
84. Z. P. Qin and H. G. Zhang, ALT$^+$-algorithm for attacking cryptosystems, *Chinese Jounrnal of Computers*, **20**(1997), 546–550. (in Chinese)
85. Z. P. Qin, H. G. Zhang and X. Q. Cao, Linear primitive factorization to the loop product of the automaton-defining function, *Chinese Jounrnal of Computers*, **22**(1999), 11–15. (in Chinese)
86. M. O. Rabin and D. Scott, Finite automata and their decision problems, *IBM Jounrnal of Research and Development*, **3**(1959), 114–125.
87. G. N. Raney, Sequential functions, *Jounrnal of the Association for Computing Machinery*, **5**(1958), 177–180.
88. S. R. Reddy and M. J. Ashjaee, A class of serial cyphers, in *Proceedings of the 1975 Conference on Information Sciences and Systems*, Johns Hopkins University Press, Baltimore, 1975, 396–400.
89. R. Rivest, A. Shamir and L. Adleman, A method for obtaining digital signatures and public-key cryptosystems, *Communications of the Association for Computing Machinery*, **21**(1978), 120–126.
90. A. Salomaa, *Public-key Cryptography,* Springer-Verlag, Berlin, 1990.
91. J. E. Savage, Some simple self-synchronizing digital data scrambles, *Bell System Technical Journal*, **46**(1967), 449–487.
92. C. E. Shannon, Communication theory of securecy systems, *Bell System Technical Journal*, **28**(1949), 656–716.
93. P. Smith, LUC public-key encryption, *Dr.Dobb's Journal*, **18**(1993), 44–49.
94. P. H. Starke, Einige Bemerkungen über nicht-deterministische Automaten, *Elektronische Informationsverarbeitung und Kybernetik*, **2**(1966), 61–82.
95. P. H. Starke, *Abstract Automata*, North-Holland Pub. Co., Amsterdan, 1972.
96. J. C. Tao (R. J. Tao), Invertible linear finite automata, *Scientia Sinica*, **16**(1973), 565–581.
97. R. J. Tao, Linear feedback shift register sequences, *Computer Application and Applied Mathematics*, 1975, No.11, 21–51. (in Chinese)
98. R. J. Tao, *Invertibility of Finite Automata*, Science Press, Beijing, 1979. (in Chinese)
99. R. J. Tao, Relationship between bounded error propagation and feedforward invertibility, *Kexue Tongbao*, **27**(1982), 680–682.
100. R. J. Tao, Some results on the structure of feedforward inverses, *Scientia Sinica*, Ser. A, **27**(1984), 157–162.
101. R.J. Tao, Cryptology and mathematics, *Nature Journal*, **7**(1984), 527–534. (in Chinese)
102. R. J. Tao, Some mathematical problems in cryptology, *Chinese Quarterly Jounrnal of Mathematics*, **2**(1987), 73–90. (in Chinese)
103. R. J. Tao, Invertibility of linear finite automata over a ring, in *Automata, Languages and Programming*, Lecture Notes in Computer Science 317, Springer-Verlag, Berlin, 1988, 489–501.

104. R. J. Tao, An application of (4,4)-Latin arrays to cryptography, *Chinese Jounrnal of Computers*, **14**(1991), 423–431. (in Chinese)
105. R. J. Tao, On finite automaton one key cryptosystems, in *Fast Software Encryption*, Lecture Notes in Computer Science 809, Springer-Verlag, Berlin, 1994, 135–148.
106. R. J. Tao, On Latin arrays, in *Proceedings of the Third International Workshop on Discrete Mathematics and Algorithms*, Jinan University Press, Guangzhou, 1994, 1–14.
107. R. J. Tao, On invertibility of some compound finite automata, Technical Report No. ISCAS-LCS-95-06, Laboratory for Computer Science, Institute of Software, Chinese Academy of Sciences, Beijing, June 1995.
108. R. J. Tao, On R_a R_b transformation and inversion of compound finite automata, Technical Report No. ISCAS-LCS-95-10, Laboratory for Computer Science, Institute of Software, Chinese Academy of Sciences, Beijing, September 1995.
109. R. J. Tao, Several specific factorizations of matrix polynomials, *Chinese Jounrnal of Computers*, **22**(1999), 1–10. (in Chinese)
110. R. J. Tao, Remark on "Weak Invertibility of Finite Automata and Cryptanalysis on FAPKC", in *Advances in Cryptology – CHINACRYPT'2000*, Science Press, Beijing, 2000, 58–68.
111. R. J. Tao and S. H. Chen, Some properties on the structure of invertible and inverse finite automata with delay τ, *Chinese Jounrnal of Computers*, **3**(1980), 289–297. (in Chinese)
112. R. J. Tao and S. H. Chen, A finite automaton public key cryptosystem and digital signatures, *Chinese Jounrnal of Computers*, **8**(1985), 401–409. (in Chinese)
113. R. J. Tao and S. H. Chen, Two varieties of finite automaton public key cryptosystem and digital signatures, *Jounrnal of Computer Science and Technology*, **1**(1986), 9–18.
114. R. J. Tao and S. H. Chen, Enumeration of Latin arrays, unpublished manuscript, 1989. (in Chinese)
115. R. J. Tao and S. H. Chen, Enumeration of Latin arrays (I) – case $n \leqslant 3$, *Science in China*, Ser. A, **33**(1990), 1430–1438.
116. R. J. Tao and S. H. Chen, Enumeration of Latin arrays (II) – case $n = 4$, $k \leqslant 4$, *Science in China*, Ser. A, **34**(1991), 20–29.
117. R. J. Tao and S. H. Chen, An implementation of identity-based cryptosystems and digital signature schemes by finite automaton public key cryptosystems, in *Advances in Cryptology – CHINACRYPT'92*, Science Press, Beijing, 1992, 87–104. (in Chinese)
118. R. J. Tao and S. H. Chen, Generating a kind of nonlinear finite automata with invertibility by transformation method, Technical Report No. ISCAS-LCS-95-05, Laboratory for Computer Science, Institute of Software, Chinese Academy of Sciences, Beijing, June 1995.
119. R. J. Tao and S. H. Chen, A generation method for a kind of linear independent Latin arrays, *Science in China*, Ser. A, **38**(1995), 884–896.
120. R. J. Tao and S. H. Chen, On input-trees of finite automata, in *Advances in Cryptology – CHINACRYPT'96*, Science Press, Beijing, 1996, 65–74. (in Chinese)
121. R. J. Tao and S. H. Chen, A necessary condition on invertibility of finite automata, *Science in China*, Ser. E, **40**(1997), 637-643.
122. R. J. Tao and S. H. Chen, A variant of the public key cryptosystem FAPKC3, *Jounrnal of Network and Computer Applications*, **20**(1997), 283–303.

123. R. J. Tao and S. H. Chen, A note on the public key cryptosystem FAPKC3, in *Advances in Cryptology – CHINACRYPT'98*, Science Press, Beijing, 1998, 69–77.
124. R. J. Tao and S. H. Chen, Generation of some permutations and involutions with dependence degree > 1, *Journal of Software*, **9**(1998), 251–255.
125. R. J. Tao and S. H. Chen, The generalization of public key cryptosystem FAPKC4, *Chinese Science Bulletin*, **44** (1999), 784–790.
126. R. J. Tao and S. H. Chen, On finite automaton public-key cryptosystem, *Theoretical Computer Science*, **226**(1999), 143–172.
127. R. J. Tao and S. H. Chen, Constructing finite automata with invertibility by transformation method, *Jounrnal of Computer Science and Technology*, **15**(2000), 10–26.
128. R. J. Tao and S. H. Chen, Input-trees of finite automata and application to cryptanalysis, *Jounrnal of Computer Science and Technology*, **15**(2000), 305–325.
129. R. J. Tao and S. H. Chen, Structure of weakly invertible semi-input-memory finite automata with delay 1, *Jounrnal of Computer Science and Technology*, **17**(2002), 369–376.
130. R. J. Tao and S. H. Chen, Structure of weakly invertible semi-input-memory finite automata with delay 2, *Jounrnal of Computer Science and Technology*, **17**(2002), 682–688.
131. R. J. Tao, S. H. Chen, and Xuemei Chen, FAPKC3: a new finite automaton public key cryptosystem, *Jounrnal of Computer Science and Technology*, **12**(1997), 289–305.
132. R. J. Tao and P. R. Feng, On relations between R_a R_b transformation and canonical diagonal form of λ-matrix, *Science in China*, Ser. E, **40**(1997), 258–268.
133. R. J. Tao and D. F. Li, An implementation of finite-automaton one-key crypto-system on IC card and selection of involutions, in *Advances in Cryptology – CHINACRYPT'2004*, Science Press, Beijing, 2004, 460–467. (in Chinese)
134. A. M. Turing, On computing numbers, with an application to the entscheidungsproblem, *Proceedings of the London Mathematical Society*, Ser. 2, **42**(1936), 230–265. A correction, *ibid.*, **43**(1937), 544–546.
135. H. Wang, *Studies on Invertibility of Automata*, Ph. D. Thesis, Institute of Software, Chinese Academy of Sciences, Beijing, 1996. (in Chinese)
136. H. Wang, R_a R_b transformation of compound finite automata over commutative rings, *Jounrnal of Computer Science and Technology*, **12**(1997), 40–48.
137. H. Wang, The R_a R_b representation of a class of the reduced echelon matrices, *Jounrnal of Software*, **8**(1997), 772–780. (in Chinese)
138. H. Wang, On weak invertibility of linear finite automata, *Chinese Jounrnal of Computers*, **20**(1997), 1003–1008. (in Chinese)
139. H. Wang, A note on compound finite automata, *Research and Development of Computers*, **34**(1997), supplement, 108–113. (in Chinese)
140. H. J. Wang, Two results of decomposing weakly invertible finite automata, *Research and Development of Computers*, **42**(2005), 690–696. (in Chinese)
141. H. J. Wang, *The Structures and Decomposition of the Finite Automata with Invertibility*, Ph. D. Thesis, Institute of Software, Chinese Academy of Sciences, Beijing, 2005. (in Chinese)
142. H. Wielandt, *Finite Permutation Groups*, Academic Press Inc., New York, 1964.
143. A. S. Willsky, Invertibility of finite group homomorphic sequential systems, *Information and Control*, **27**(1975), 126–147.

144. G. A. Xu and Z. P. Qin, Isotopy classes of $(4, k)$-Latin array, *Jounrnal of Huazhong University of Science and Technology*, **26**(1998), No.3, 100–102. (in Chinese)
145. Z. Q. Yang and T. H. Zhou, The implementation of a finite automaton cryptosystem, *Computer Research and Development*, **22**(1985), 29–35,59. (in Chinese)
146. G. Yao, Two results on structure of feedforward inverse finite automata, *Jounrnal of Software*, **13**(2002), supplement, 252–258. (in Chinese)
147. G. Yao, Decomposing a kind of weakly invertible finite automata with delay 2, *Jounrnal of Computer Science and Technology*, **18**(2003), 354–360.
148. G. Yao, *Some Results of Finite Automata Invertibility*, Ph. D. Thesis, Institute of Software, Chinese Academy of Sciences, Beijing, 2003. (in Chinese)
149. D. F. Ye, Z. D. Dai and K. Y. Lam, Decomposing attacks on asymmetric cryptography based on mapping compositions, *Jounrnal of Cryptology*, **14**(2001), 137–150.
150. F. M. Yuan, Minimal memory inverse of linear sequential circuits, *IEEE Transactions on Computers*, **23**(1974), 1155–1163.
151. J. Zhou and Z. P. Qin, A fast algorithm of generating the representatives of (n, k)-Latin array's isotopy classes, *Jounrnal of Huazhong University of Science and Technology*, **28**(2000), No.1, 100–101,108. (in Chinese)
152. S. Y. Zhou, *On Weakly Invertibility of Type I Abelian Finite Group Homomorphic Sequential Systems*, MA Thesis, Institute of Computing Technology, Chinese Academy of Sciences, Beijing, 1981. (in Chinese)
153. X. J. Zhu, On the structure of binary feedforward inverses with delay 2, *Jounrnal of Computer Science and Technology*, **4**(1989), 163–171.
154. N. Zieler, Linear recurring sequences, *Jounrnal of the Society for Industrial and Applied Mathematics*, **7**(1959), 31–48.

Index