Hindawi
Complexity
Volume 2019, Article ID 8765970, 11 pages
https://doi.org/10.1155/2019/8765970

Research Article

On Stability of Multi-Valued Nonlinear Feedback Shift Registers

Haiyan Wang,[1] Qiuzhen Lin,[1] Jianyong Chen,[1] Jianqiang Li,[1] Jianghua Zhong,[2] Dongdai Lin,[2] Jia Wang,[1] and Lijia Ma[1]

[1]*College of Computer Science and Software Engineering, Shenzhen University, Shenzhen 518060, China*
[2]*State Key Laboratory of Information Security, Institute of Information Engineering, Chinese Academy of Sciences, Beijing 100093, China*

Correspondence should be addressed to Qiuzhen Lin; qiuzhlin@szu.edu.cn

Received 12 October 2018; Revised 9 November 2018; Accepted 20 January 2019; Published 21 February 2019

Academic Editor: Eric Campos-Canton

Nonlinear feedback shift registers (NFSRs) are the main building blocks in many convolutional decoders, and a stable NFSR can limit decoding error propagation. Due to lack of efficient algebraic tools, the stability of multi-valued NFSRs has been much less studied. This paper studies the stability of multi-valued NFSRs using a logic network approach. A multi-valued NFSR can be viewed as a logic network. Based on its logic network representation, some sufficient and necessary conditions are provided for globally (locally) stable multi-valued NFSRs, explicit forms are given for the set of basins, and the algorithm for obtaining the set of basins is provided as well. Finally, a new method is presented for constructing stable $n+1$-stage NFSRs from stable n-stage NFSRs by the properties of D-morphism.

1. Introduction

Nonlinear feedback shift registers (NFSRs) are the main building blocks in many convolutional decoders. However, in the process of decoding, a decoding error tends to induce indefinitely long decoding errors. A stable NFSR is an alternative to limit this error propagation. Some studies have focused on the stability of NFSRs. In 1964, Massey and Liu [1] proposed that using a stable nonlinear feedback shift register (NFSR) as the main building block in a convolutional decoder is able to limit such an error propagation. In their NFSR-based decoder, the feedback function represents a decoding algorithm. They gave an example to highlight the application of the NFSR-based decoder. Mowle [2] proved that the number of n-stage globally stable NFSRs is 2^{2^n-n-1} and also showed that all these NFSRs are binomially distributed. In [3, 4], the author gave the enumeration and classification of stable FSRs and an algorithm to generate all of them. A direct algorithm for the synthesis of stable NFSRs was proposed [5]. It is notable to point out that only binary NFSRs were concerned in the above work. In addition, Lempel [6] gave some results on k-stable NFSRs. Since then, the stability of NFSRs has not been further studied due to lack of efficient mathematical tools, although numerous other efforts have been made on NFSRs over the past decades.

In [7–9], the authors studied the stability for binary NFSRs by viewing them as Boolean networks. A Boolean network is a finite state automaton evolving through Boolean functions. It was firstly introduced by Kauffman in 1969 to model a genetic network whose variables take only two possible values, "on" and "off" (or equivalently, 1 and 0, resp.) [10]. Over the last decades, Boolean networks have attracted much attention in many communities, such as biology [11–13], physics [14–16], system sciences [17–21], and control theory [22, 23]. In the community of system sciences, Cheng and his collaborators [24] developed an algebraic framework for Boolean networks using a semitensor product approach. In the algebraic framework, a Boolean network can be equivalently converted into a conventional discrete-time linear system. A logic network is a generalization of a Boolean network. The variables of a logical network usually take multiple values. If they take only two values, say 0 and 1, then the logical network is reduced to a Boolean network. The studies of multi-valued logical networks can refer to, for instance, [25–27].

Multi-valued NFSRs have been investigated in several studies. For example, some construction methods were given for de Bruijn sequences generated from multi-valued NFSRs [28–30]. A necessary and sufficient condition was given for the nonsingularity of multi-valued NFSRs [31]. Recently, the multi-valued NFSRs were studied in [32–34]. Some necessary and sufficient conditions were given for the stability of multi-valued NFSRs in [35].

In this paper, we study the stability of multi-valued NFSRs using a logic network approach. A multi-valued NFSR can be viewed as a logic network. Based on its logic network representation, we give the state transition matrix [34], which shows the simple relation with the truth table of the feedback function of the NFSR. From this viewpoint, it is more explicit than the state transition matrix introduced in [31], where the state transition matrices are expressed as the products of some structure matrices of the components of the vectorial function. In fact, from the cryptography perspective, it is very important and useful to show the explicit relation between the truth table of the feedback function and the state transition matrix in order to analyze and design an NFSR. This paper is an extension of our previous work [35], which is more complete and more substantial due to the following contributions:

(1) Because the stability of an NFSR completely depends on the basin of the NFSR, we give the explicit forms for the set of basins, and the algorithm for obtaining the set of basins is provided as well.

(2) A stable NFSR is an alternative to limit error propagation in the process of decoding; therefore we give a new method for constructing stable $n+1$-stage NFSRs from stable n-stage NFSRs over the binary field.

The remainder of this paper is organized as follows. Section 2 briefly reviews some related works on logic networks. Sections 3 and 4 are our main results. Some sufficient and necessary conditions are given for globally (locally) stable NFSRs in Section 3. In Section 4, we present the method to construct stable NFSRs, and examples are presented to show the effectiveness of the proposed method. The paper is concluded in Section 5.

2. Logic Network Representation of NFSR

In this section, we first briefly review the semitensor product of matrices and recall the multi-linear form of nonlinear logic function that is obtained by the semi-tensor product. Finally, we revisit the logic network representation of a multi-valued NFSR, which is very useful to investigate the stability of NFSRs.

For the statement ease, we first give some notations:

(i) $\mathscr{D}_k = \{0, 1, 2, \ldots, k-1\}$; when $k = 2$, we denote $\mathbb{F}_2$, that is, binary field.

(ii) $\mathscr{D}_k^n$: n-dimensional vectors over $\mathscr{D}_k$; when $k = 2$, we denote $\mathbb{F}_2^n$, that is, n-dimensional vector space over $\mathbb{F}_2$.

(iii) I_n : the identity matrix of dimension n.

(iv) δ_n^i: the i-th column of the identity matrix I_n.

(v) $\Delta_n = \{\delta_n^i | i = 1, 2, \ldots, n\}$.

(vi) Δ_n^m: the set of all m-dimensional vectors over Δ_n.

(vii) $C_j(A)$: the j-th column of a matrix A.

(viii) $ord(B)$: the order of a square matrix B of dimension n, that is, the least power p satisfying $B^p = I_n$.

(ix) $\mathscr{L}_{n\times m}$: the set of $n \times m$ matrices, whose columns belong to Δ_n. If $L \in \mathscr{L}_{n\times m}$, then it can be expressed as $L = \left[\delta_n^{i_1} \ \delta_n^{i_2} \ \cdots \ \delta_n^{i_n}\right]$. For the sake of compactness, it is briefly denoted by $L = \delta_n [i_1 \ i_2 \ \cdots \ i_n]$.

2.1. Semi-Tensor Product and Multilinear Form of Logic Network. Semi-tensor product of matrices was introduced by Cheng [24]. It is a generalization of the conventional matrix product and works for any two matrices regardless of their sizes, while it retains all major properties of the conventional matrix product. Before reviewing the semitensor product, we first recall what the Kronecker product is.

Definition 1 (see [36]). Let $A = (a_{ij})$ and B be matrices of dimensions $n \times m$ and $p \times q$, respectively. The Kronecker product of A and B is defined as an $np \times mq$ matrix, given by

$$A \otimes B = \begin{pmatrix} a_{11}B & a_{12}B & \cdots & a_{1m}B \\ a_{21}B & a_{22}B & \cdots & a_{2m}B \\ \vdots & \vdots & \vdots & \vdots \\ a_{n1}B & a_{n2}B & \cdots & a_{nm}B \end{pmatrix}. \tag{1}$$

Definition 2 (see [24]). Let A and B be matrices of dimensions $n \times m$ and $p \times q$, respectively, and let α be the least common multiple of m and p. The (left) semitensor product of A and B is defined as an $n\alpha/m \times q\alpha/p$ matrix, given by

$$A \ltimes B = \left(A \otimes I_{\alpha/m}\right)\left(B \otimes I_{\alpha/p}\right). \tag{2}$$

Clearly, if $m = p$, the semi-tensor product $A \ltimes B$ is reduced to their conventional matrix product AB.

A logical function f with n variables is a mapping from $\mathscr{D}_k^n$ to $\mathscr{D}_k$. Let i be the decimal number corresponding to the tuple $(i_1, i_2, \ldots, i_n) \in \mathscr{D}_k^n$ via the mapping $i = i_1 k^{n-1} + i_2 k^{n-2} + \cdots + i_n$. Then i ranges from 0 to $k^n - 1$. For the sake of simplicity, we denote $f(i) = f(i_1, i_2, \ldots, i_n)$. Then $[f(0), f(1), \ldots, f(k^n - 1)]$ is the truth table of f, arranged in the alphabet order, while $[f(k^n - 1), f(k^n - 2), \ldots, f(0)]$ is truth table of f, arranged in the reverse alphabet order.

Identify a variable $x \in \mathscr{D}_k$ as a vector $X = \delta_k^{k-x} \in \Delta_k$. Then a logic function f with n variable from $\mathscr{D}_k^n$ to $\mathscr{D}_k$ is changed to a function from Δ_k^n to Δ_k.

Lemma 3 (see [24]). *For any logical function $f(X_1, X_2, \ldots, X_n)$ with $X_i \in \Delta_k, i = 1, 2, \ldots, n$, let $[s_1, s_2, \ldots, s_{k^n}]$ be the truth table of f, arranged in the reverse alphabet order. Then f can be expressed as a multilinear form:*

$$f\left(X_1, X_2, \ldots, X_n\right) = M \ltimes X_1 \ltimes X_2 \ltimes \cdots \ltimes X_n, \tag{3}$$

where $M = \delta_k[k - s_1 k - s_2 \cdots k - s_{k^n}]$ *is called the structure matrix of* f.

Lemma 4 (see [24]). *Suppose*

$$\mathbf{x} = X_1 \ltimes X_2 \ltimes \cdots \ltimes X_n \tag{4}$$

with $X_i \in \Delta_k$, $i = 1, 2, \ldots, n$. *Then* $\mathbf{x} \in \Delta_{k^n}$. *Moreover, For any* $j \in \{1, 2, \ldots, k^n\}$, *the state* $\mathbf{x} = \delta_{k^n}^j \in \Delta_{k^n}$ *and the state* $(x_1, x_2, \ldots, x_n)^T \in \mathscr{D}_k^n$, *which satisfy* $k^{n-1}x_1 + k^{n-2}x_2 + \cdots + x_n = k^n - j$, *are one-to-one correspondent.*

Definition 5 (see [37]). Let $A = \begin{bmatrix} A_1 & A_2 & \cdots & A_n \end{bmatrix}$ and $B = \begin{bmatrix} B_1 & B_2 & \cdots & B_n \end{bmatrix}$ be matrices of dimensions $m \times n$ and $p \times n$, respectively, where A_i and B_i, $i = 1, 2, \ldots, n$, are the i-th column of matrices A and B, respectively. The Khatri-Rao product of A and B is defined as an $mp \times n$ matrix, given by

$$A * B = \begin{bmatrix} A_1 \otimes B_1 & A_2 \otimes B_2 & \cdots & A_n \otimes B_n \end{bmatrix}, \tag{5}$$

and a logical network with n-nodes can be described as the following system:

$$X(t+1) = \mathbf{g}(X(t)), \tag{6}$$

where $X = (x_1, x_2, \ldots, x_n)^T \in \mathscr{D}_k^n$, and $\mathbf{g} = (g_1, g_2, \ldots, g_n)^T$, with $g_i : \mathscr{D}_k^n \longrightarrow \mathscr{D}_k$ for all $i = 1, 2, \ldots, n$. Let G_i be the structure matrix of the function g_i for any $i \in \{1, 2, \ldots, n\}$. System (6) can be equivalently described as a linear system [21]:

$$\mathbf{x}(t+1) = L\mathbf{x}(t), \tag{7}$$

where $\mathbf{x} = X_1 \ltimes X_2 \ltimes \cdots \ltimes X_n \in \Delta_{k^n}$ is the state and $L = G_1 * G_2 * \cdots * G_n \in \mathscr{L}_{k^n \times k^n}$ is the state transition matrix.

2.2. Logic Network Representation of Multi-Valued NFSR. An n-stage multi-valued Fibonacci NFSR can be described as Figure 1. It is a collection of n storage devices, whose contents are denoted by the variables $x_1, x_2, \ldots, x_n$, taking values from the set $\mathscr{D}_k = \{0, 1, \ldots, k-1\}$. Here the logical function $f(x_1, x_2, \ldots, x_n)$ is called the feedback function of the NFSR. For any $i \in \{1, 2, \ldots, n-1\}$, the content x_{i+1} is shifted to x_i at each periodic interval determined by a master clock. However, to obtain a new value for the variable x_n, we compute the function $f(x_1, x_2, \ldots, x_n)$ of all the present contents in the shift register.

The state diagram of an n-stage k-valued NFSR is a directed graph consisting of k^n nodes and k^n directed edges. Each node represents a state of the NFSR, and an edge from state $\mathbf{X}$ to state $\mathbf{Y}$ means that $\mathbf{X}$ is shifted to the state $\mathbf{Y}$. $\mathbf{X}$ is called a predecessor of $\mathbf{Y}$, and $\mathbf{Y}$ is called the successor of $\mathbf{X}$. Every state of an NFSR has a unique successor but may have no predecessor or a single predecessor or m predecessors with a positive integer m satisfying $1 \le m \le k$. The state with more than one predecessor is called a branch state, while the state without predecessors is called a starting state. A sequence of p distinct states, $\mathbf{X}_1, \mathbf{X}_2, \ldots, \mathbf{X}_p$, is called a cycle of length p if $\mathbf{X}_1$ is the successor of $\mathbf{X}_p$, and $\mathbf{X}_{i+1}$ is a successor of $\mathbf{X}_i$ for any $i \in \{1, 2, \ldots, p-1\}$. Similarly, a sequence of p distinct states, $\mathbf{X}_1, \mathbf{X}_2, \ldots, \mathbf{X}_p$, is called a transient of length p, if the following conditions are satisfied: (1) none of them lies on a cycle; (2) $\mathbf{X}_1$ is a starting state; (3) $\mathbf{X}_{i+1}$ is a successor of $\mathbf{X}_i$ for any $i \in \{1, 2, \ldots, p-1\}$; (4) the successor of $\mathbf{X}_p$ lies on a cycle.

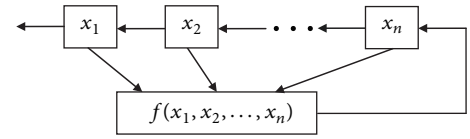

Figure 1: An n-stage nonlinear feedback shift register.

For the sake of statement simplicity, in the sequel an n-stage NFSR means an n-stage multi-valued Fibonacci NFSR over $\mathscr{D}_k$.

View the n-stage NSFR in Figure 1 as a logic network. Then it can be expressed as

$$X(t+1) = \mathbf{g}(X(t)), \tag{8}$$

where $X = (x_1, x_2, \ldots, x_n)^T \in \mathscr{D}_k^n$ is the state and $\mathbf{g} = (g_1, g_2, \ldots, g_n)^T$ is the state transition function, satisfying

$$\begin{aligned} g_1(X(t)) &= x_2(t), \\ &\vdots \\ g_{n-1}(X(t)) &= x_n(t), \\ g_n(X(t)) &= f(x_1(t), x_2(t), \ldots, x_n(t)). \end{aligned} \tag{9}$$

For any positive integer N, let $\mathbf{g}^{N+1}(X(t)) = \mathbf{g}(\mathbf{g}^N(X(t)))$, which indicates that the state $\mathbf{g}(X(t))$ is shifted N times from $X(t)$.

Lemma 6 (see [34]). *For an n-stage NFSR with a feedback function* f, *assume the truth table of* f *to be* $[s_1, s_2, \ldots, s_{k^n}]$, *arranged in the reverse alphabet order. Then the NFSR can be equivalently expressed as a linear system*

$$\mathbf{x}(t+1) = L\mathbf{x}(t), \tag{10}$$

where $\mathbf{x} \in \Delta_{k^n}$ *is the state and* $L \in \mathscr{L}_{k^n \times k^n}$ *is the state transition matrix, expressed as*

$$L = \delta_{k^n}[\eta_1 \ \eta_2 \ \cdots \ \eta_{k^n}], \tag{11}$$

where

$$\eta_m = \left\{\left[(m-1) \mod k^{n-1}\right] + 1\right\} k - s_m, \quad m = 1, 2, \ldots, k^n. \tag{12}$$

3. The Properties of Stable Multi-Valued NFSR

In this section, we first briefly review some existing basic concepts of the stability of NFSRs. Then we show that the error-propagation effect is closely related to the stability of an NFSR. Finally, we give some sufficient and necessary conditions for their stability.

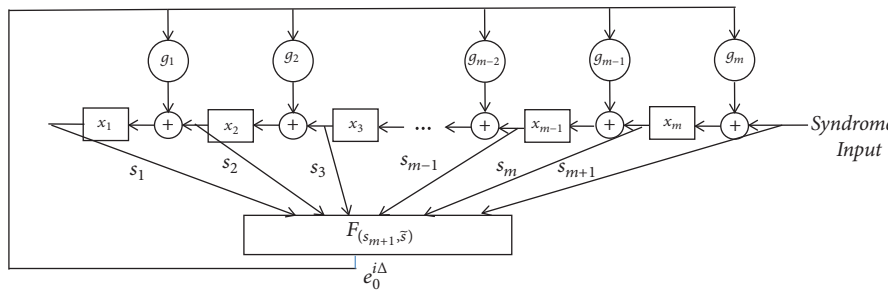

Figure 2: Decoder NFSR.

3.1. Basic Concepts

Definition 7 (see [2]). A state $\mathbf{X}(t)$ is called an equilibrium state of the logic network (6), if $\mathbf{g}(\mathbf{X}(t)) = \mathbf{X}(t)$. For a k-valued NFSR, the equilibrium state of its logic network representation (8) is also called an equilibrium state of the NFSR.

Note that an equilibrium state of an NFSR forms a cycle of length 1, that is, unit cycle, in the state diagram of the NFSR.

Definition 8. The set $\mathscr{E}$ is called the basin of an equilibrium state $\mathbf{X}$ of an NFSR, if $\mathscr{E}$ is a set of states eventually reaching the equilibrium state $\mathbf{X}$.

Definition 9 (see [1]). An n-stage NFSR is globally stable to the equilibrium state $\mathbf{0}$, if, for any state $\mathbf{X}(t)$, there exists a positive integer N such that the state transition function of its logic network representation (8) satisfies $\mathbf{g}^N(\mathbf{X}(t)) = 0$; that is, $\mathbf{0}$ is the unique equilibrium state and there are no other cycles in the state diagram of the NFSR.

Definition 10 (see [1]). An n-stage NFSR is locally stable to the equilibrium state $\mathbf{0}$, if there exists some state $\mathbf{X}(t) \neq \mathbf{0}$ such that for some positive integer N the state transition function of its logic network representation (8) satisfies $\mathbf{g}^N(\mathbf{X}(t)) = \mathbf{0}$.

Since an n-stage multi-valued NFSR has an equivalent logic network representation in a linear system (10), accordingly, we give an equivalent definition of globally (locally) stable multi-valued NFSR as follows.

Definition 11. An n-stage NFSR is globally stable to the equilibrium state $\mathbf{0}$, if, for any state $\mathbf{x}(t)$, there exists a positive integer N such that the state transition matrix L of its logic network representation (8) satisfies $L^N\mathbf{x}(t) = \delta_{k^n}^{k^n}$.

Definition 12. An n-stage NFSR is locally stable to the equilibrium state $\mathbf{0}$, if there exists some state $\mathbf{x}(t) \neq \delta_{k^n}^{k^n}$ such that for some positive integer N the state transition matrix L of its logic network representation (8) satisfies $L^N\mathbf{x}(t) = \delta_{k^n}^{k^n}$.

In the sequel, an NFSR is globally (resp., locally) stable meaning that an NFSR is globally (resp., locally) stable to the equilibrium state $\mathbf{0}$. From their definitions, it is easy to see that a globally stable NFSR must be locally stable but not the vice versa.

Definition 13 (see [2]). An NFSR is called a globally stable maximum transient NFSR if it is globally stable and has a single starting state.

3.2. Decoder NFSR.

We show below that the error-propagation effect is closely related to the stability of an NFSR. The relevant portion of a decoder is shown in Figure 2 and is seen to constitute an NFSR. The first m terms of the syndrome sequence are stored in the shift register, and the current input is s_{m+1}, at the time when the decoder forms $e_0^{i\Delta}$. Let the vector $\mathbf{s} = (s_1, s_2, \ldots, s_m)$ represent the shift register contents and let $\mathbf{0}$ denote the all-zero vector. $\mathbf{s}$ will be referred to as the state of the NFSR. The decoding algorithm is represented by the function $F(s_{m+1}, \mathbf{s})$; that is, $F(s_{m+1}, \mathbf{s}) = e_0^{i\Delta}$. For any reasonable decoding algorithm, $F(0, \mathbf{0}) = 0$, since this is the case where all parity checks are satisfied. From Figure 2, it should be clear that m consecutive correct decoding decisions will clear the decoder of any spurious symbols introduced by a decoding error and hence will terminate the error propagation. The ability of the decoder to affect such a "reconvergence" is conveniently studied by considering the shift register to be loaded with some initial states $\mathbf{s}$ and the syndrome input sequence to be all zeros; that is, all succeeding parity checks are satisfied. Finally, the shift register will enter state $\mathbf{0}$ when reconvergence has been achieved. Thus the problem of studying error propagation will be reduced to the stability analysis of an NFSR in Figure 1.

3.3. Necessary and Sufficient Conditions for Stability

Theorem 14. *An NFSR is locally stable if and only if the feedback function satisfies $f(0, \ldots, 0) = 0$, and there is at least one $i \in \{1, 2, \ldots, k-1\}$ such that $f(i, 0, \ldots, 0) = 0$.*

Proof. Necessity: Clearly, according to Definition 9, $f(0, \ldots, 0) = 0$ is a necessary condition for a locally stable NFSR. For any NFSR, the state $\mathbf{0}$ has the possible predecessors: itself and $(i, 0, \ldots, 0)^T$ with $i \in \{1, 2, \ldots, k-1\}$. If the NFSR is locally stable, then there exist some states $\mathbf{X}(t) \neq \mathbf{0}$ such that, for some integers N, $\mathbf{g}^N(\mathbf{X}(t)) = 0$. Thus,

0 has a predecessor different to itself. Hence, there is at least one $i \in \{1, 2, \ldots, k-1\}$ such that $f(i, 0, \ldots, 0) = 0$.

Sufficiency: $f(0, \ldots, 0) = 0$ implies that **0** is an equilibrium state of the NFSR. If there is at least an $i \in \{1, 2, \ldots, k-1\}$ such that $f(i, 0, \ldots, 0) = 0$, and then $(i, 0, \ldots, 0)^T$ is a predecessor of **0**. In other words, there exists a state $\mathbf{X}(t) = (i, 0, \ldots, 0)^T \neq \mathbf{0}$ such that $\mathbf{g}(\mathbf{X}(t)) = 0$, which implies that the NFSR is locally stable. □

Corollary 15. *Let $L = \delta_{k^n}[\eta_1 \ \eta_2 \ \ldots \ \eta_{k^n}]$ be the state transition matrix of the logic network representation (10) in a linear system of an n-stage NFSR. Then the NFSR is locally stable if and only if there exists at least one $i \in \{1, 2, \ldots, k-1\}$ such that $\eta_{k^n} = \eta_{(k-i)k^{n-1}} = k^n$.*

Proof. Let the truth table of the feedback function f of the NFSR be $[s_1 \ s_2 \ \cdots \ s_{k^n}]$, arranged in the reverse alphabet order. According to Theorem 14, the NFSR is locally stable if and only if the feedback function satisfies $f(0, \ldots, 0) = 0$, and there is at least one $i \in \{1, 2, \ldots, k-1\}$ such that $f(i, 0, \ldots, 0) = 0$, that is, $s_{k^n} = s_{(k-i)k^{n-1}} = 0$. Then the result follows from (12). □

Proposition 16. *Let L be the state transition matrix of the logic network representation (10) in a linear system of an n-stage NFSR. If $C_{k^n}(L) = \delta_{k^n}^{k^n}$, there exists an integer l such that $C_r(L^l) = \delta_{k^n}^{k^n}$ with some $r \neq k^n$. Then the NFSR is locally stable to $\delta_{k^n}^{k^n}$.*

Proof. Let $L = \delta_{k^n}[\eta_1 \ \eta_2 \ \ldots \ \eta_{k^n}]$, and let the truth table of the feedback function f of the NFSR be $[s_1 \ s_2 \ \cdots \ s_{k^n}]$, arranged in the reverse alphabet order. Since $C_{k^n}(L) = \delta_{k^n}^{k^n}$, we have $\eta_{k^n} = k^n$. According to (12), we have $s_{k^n} = \{[(k^n - 1) \bmod k^{n-1}] + 1\}k - \eta_{k^n} = 0$, that is, $f(0, \ldots, 0) = 0$. Since there exists an integer l, such that $C_r(L^l) = \delta_{k^n}^{k^n}$, we have $L^l \delta_{k^n}^r = \delta_{k^n}^{k^n}$. According to Definition 10, the NFSR is locally stable to $\delta_{k^n}^{k^n}$. □

Theorem 17. *If the NFSR is globally stable maximum transient, then there exists a unique $i \in \{1, 2, \ldots, k-1\}$ such that $f(0, 0, \ldots, 0) = f(i, 0, \ldots, 0) = 0$.*

Proof. If the NFSR is globally stable maximum transient, then except the starting state and the state **0**, the other states have their own unique predecessor and unique successor. The state **0** has the possible predecessors: itself and $(i, 0, \ldots, 0)^T$ with $i \in \{1, 2, \ldots, k-1\}$. Since the NFSR is globally stable maximum transient, **0** has a unique predecessor different to itself. Then the result follows. □

Remark 18. Theorem 14 shows that there is at least one $i \in \{1, 2, \ldots, k-1\}$ such that $f(0, 0, \ldots, 0) = f(i, 0, \ldots, 0) = 0$ is sufficient and necessary condition for a locally stable NFSR. However, Theorem 17 shows that there exists a unique $i \in \{1, 2, \ldots, k-1\}$ such that $f(0, 0, \ldots, 0) = f(i, 0, \ldots, 0) = 0$ for globally stable maximum transient NFSR, which is nothing but necessary condition.

Theorem 19. *Let L be the state transition matrix of the logic network representation (10) in a linear system of an n-stage NFSR. The NFSR is globally stable, if and only if there exists an integer $N \leq k^n - 1$ such that each column of L^N is equal to $\delta_{k^n}^{k^n}$. Moreover, the NFSR is globally stable maximum transient, if and only if each column of L^{k^n-1} is equal to $\delta_{k^n}^{k^n}$.*

Proof. Necessity: As the equilibrium state $\mathbf{0} \in \mathscr{D}_k^n$ is uniquely corresponding to the state $\delta_{k^n}^{k^n} \in \Delta_{k^n}$, that an n-stage NFSR is globally stale to the equilibrium state **0** is equivalent to that the n-stage NFSR is globally stable to the state $\delta_{k^n}^{k^n}$. Clearly, any state of an n-stage globally stable NFSR with one more starting state must be shifted fewer times to reach the equilibrium state **0** than the n-stage globally stable maximum transient NFSR. For an n-stage globally stable maximum transient NFSR, the starting state $\mathbf{x}_0 = \delta_{k^n}^i$ must shift $k^n - 1$ times to go through all other states and finally reaches the state $\delta_{k^n}^{k^n}$ (or, equivalently, the state **0**) and keeps staying at this state. Therefore, $N = k^n - 1$ is the largest power such that each column of L^N is equal to $\delta_{k^n}^{k^n}$.

Sufficiency: There exists an integer $N \leq k^n - 1$ such that each column of L^N is equal to $\delta_{k^n}^{k^n}$. Therefore, for the state $\delta_{k^n}^i$ with any $i \in \{1, 2, \ldots, k^n\}$, we have $L^N \delta_{k^n}^i = \delta_{k^n}^{k^n}$. According to Definition 9, the NFSR is globally stable. In particular, $N = k^n - 1$ means that the starting state $\delta_{k^n}^i$ for any $i \in \{1, 2, \ldots, k^n\}$ eventually reaches the equilibrium state $\delta_{k^n}^{k^n}$ and keeps staying at this state. Thus, the result follows. □

Theorem 20. *Given a globally stable maximum transient k-valued n-stage NFSR, its starting state is $(0, 0, \ldots, 0, i)^T$ with some $i \in \{1, 2, \ldots, k-1\}$.*

Proof. For a given globally stable maximum transient k-valued n-stage NFSR, the state $(0 \ 0 \ \cdots \ 0)^T$ has only two predecessors, itself and $(a, 0, \ldots, 0)^T$ with some $a \in \{1, 2, \ldots, k-1\}$. Assume that all $k-1$ states $(0, 0, \ldots, 0, i)^T$, for any $i \in \{1, 2, \ldots, k-1\}$, are not the starting state of the NFSR. Let the predecessor of any given state $(0, 0, \ldots, 0, i)^T$ be $(b_i, 0, \ldots, 0)^T$, with some $b_i \in \{1, 2, \ldots, k-1\}$. Then there exists b_{i_0} with some $i_0 \in \{1, 2, \ldots, k-1\}$ such that $(b_{i_0}, 0, \ldots, 0)^T = (a, 0, \ldots, 0)^T$, which implies that $b_{i_0} = a$. Note that the successor of $(b_{i_0}, 0, \ldots, 0)^T$ is $(0, 0, \ldots, i_0)^T$ with $i_0 \in \{1, 2, \ldots, k-1\}$, and the successor of $(a, 0, \ldots, 0)^T$ is $(0 \ 0 \ \cdots \ 0)^T$. Then $(a, 0, \ldots, 0)^T$ has two different successors, $(0, 0, \ldots, 0)^T$ and $(0, 0, \ldots, 0, i_0)^T$ with some $i_0 \in \{1, 2, \ldots, k-1\}$, which is a contradiction that any state has a unique successor. Hence, for any given globally stable maximum transient k-valued n-stage NFSR, there exists some $i \in \{1, 2, \ldots, k-1\}$ such that $(0, 0, \ldots, 0, i)^T$ is the starting state of the NFSR. □

Example 21. When $k = 3$ and $n = 2$, we consider two nonlinear feedback shift registers, NFSR1 and NFSR2. Their feedback functions are, respectively, as follows:

$$f_1(x_1, x_2) = x_2 + x_1\left(x_2^2 + 1\right) \tag{13}$$

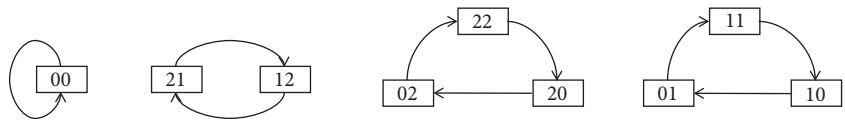

Figure 3: State diagram of NFSR1 in Example 21.

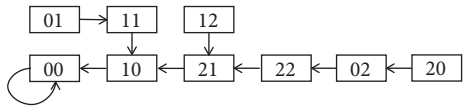

Figure 4: State diagram of NFSR2 in Example 21.

and

$$f_2(x_1, x_2) = 2x_1 + x_2 + x_1^2 + x_1x_2^2 + x_1^2x_2^2. \quad (14)$$

Computations show that the state transition matrices of the logic network representations of both NFSRs, respectively, are

$$L_1 = \delta_9[3\ 4\ 7\ 2\ 6\ 8\ 1\ 5\ 9] \quad (15)$$

and

$$L_2 = \delta_9[2\ 6\ 7\ 2\ 6\ 9\ 1\ 5\ 9]. \quad (16)$$

We use the same notations in previous sections. For the state transition matrix L_1, $\eta_3 \neq 9, \eta_6 \neq 9$. According to Corollary 15, we get that NFSR1 is not locally stable. Meanwhile, for the state transition matrix L_2, $\eta_6 = 9$, and $L_2^5 = \delta_9[9\ 9\ 9\ 9\ 9\ 9\ 9\ 9\ 9]$. From Theorem 17 and Theorem 19, we obtain that NFSR2 is globally stable. Moreover $\eta_1 = \eta_4 = 2, \eta_2 = \eta_5 = 6$, and according to Proposition13 in [34], the NFSR2 has two branch states: δ_9^2 and δ_9^6. Actually, the NFSR2 has three starting states. All those features are consistent with their state diagrams, which are shown in Figures 3 and 4.

Example 22. When $k = 3, n = 2$, we consider two nonlinear feedback shift registers, NFSR3 and NFSR4. Their feedback functions are, respectively, as follows:

$$f_3(x_1, x_2) = 2x_1 + x_2 + 2x_1x_2 + x_1x_2^2 + x_1^2 + x_1^2x_2 \quad (17)$$

and

$$f_4(x_1, x_2) = 2x_1 + x_2 + x_1x_2 + 2x_1^2 + 2x_1^2x_2. \quad (18)$$

Computations show that the state transition matrices of the logic network representations of both NFSRs, respectively, are

$$L_3 = \delta_9[2\ 6\ 7\ 3\ 4\ 9\ 1\ 5\ 9] \quad (19)$$

and

$$L_4 = \delta_9[2\ 6\ 9\ 3\ 4\ 8\ 1\ 5\ 9]. \quad (20)$$

We use the same notations in previous sections. For the state transition matrix L_3 and L_4, we obtain $L_3^8 = L_4^8 = \delta_9[9\ 9\ 9\ 9\ 9\ 9\ 9\ 9\ 9]$. According to Theorem 19, both NFSR3 and NFSR4 are globally stable maximum transient. Their state diagrams are shown in Figure 5.

Clearly, $(0,1)^T$ (resp., $(0,2)^T$) is the starting state of NFSR3 (resp., NFSR4), which is consistent with the result in Theorem 20. It also shows that different globally stable maximum transient k-valued NFSRs with $k > 2$ may have different starting states, which is unlike the globally stable maximum transient binary NFSRs whose starting states are the same, that is, $(0, 0, \ldots, 0, 1)^T$.

3.4. Basin of the Equilibrium State of NFSRs

Definition 23. The set $\mathscr{E}$ is called the basin of an equilibrium state $\mathbf{X}$ of an NFSR, if $\mathscr{E}$ is a set of states eventually reaching the equilibrium state $\mathbf{X}$.

We let $\mathscr{E}(\delta_{k^n}^{k^n})$ be the basin of the equilibrium state $\delta_{k^n}^{k^n}$. The stability of an NFSR in Figure 1 completely depends on the basin $\mathscr{E}(\delta_{k^n}^{k^n})$. In the following, we will focus on how to get the basin of the equilibrium state. Reference [34] gives a way to find all starting states of an NFSR, shown in the following lemma.

Lemma 24 (see [34]). *Let L be a state transition matrix of an n-stage NFSR. $\delta_{k^n}^i$ is a starting state if and only if $\delta_{k^n}^i$ is not a column of the state transition matrix L, where $i \in \{1, 2, \ldots, k^n\}$.*

Theorem 25. *Let L be a state transition matrix of an n-stage NFSR. Then the basin of the equilibrium state $\delta_{k^n}^{k^n}$ is $\mathscr{E}(\delta_{k^n}^{k^n}) = \{L^k\delta_{k^n}^i | 1 \leq k \leq K_i, K_i$ is the smallest k_i satisfying $L^k\delta_{k^n}^i = \delta_{k^n}^{k^n}$, and $\delta_{k^n}^i \in C(L)$ with some positive integer $i \leq k^n\}$.*

Proof. The result follows from Lemmas 6 and 24.

In fact, it is easy to get the whole state transition graph of an NFSR when its state transition matrix L is known. For any $L \in \mathscr{L}_{k^n \times k^n}, C_i(L) = \delta_{k^n}^j$, we have that $\delta_{k^n}^i$ is the predecessor state of $\delta_{k^n}^j$ and $\delta_{k^n}^j$ is the successor state of $\delta_{k^n}^i$; that is, $L\delta_{k^n}^i = C_i(L)$. For example, we consider the NFSR2 in Example 21. Its state transition matrix $L = \delta_9[2\ 6\ 7\ 2\ 6\ 9\ 1\ 5\ 9]$. Obviously, only $\delta_9^3, \delta_9^4, \delta_9^8 \in C(L)$, and according to Lemma 24, they are all starting states of NFSR2. For the state δ_9^4, it is easy to see that $L\delta_9^4 = C_4(L) = \delta_9^2$, and $L^2\delta_9^4 = L\delta_9^2 = C_2(L) = \delta_9^6$ and $L^3\delta_9^4 = L^2\delta_9^2 = L\delta_9^6 = C_6(L) = \delta_9^9$. Thus, according to Theorem 25, we have $\delta_9^4, \delta_9^2, \delta_9^6, \delta_9^9 \in \mathscr{E}(\delta_{k^n}^{k^n})$. Similarly, for the state δ_9^3, δ_9^8, we can also use the same method. Finally,

1: Set $l = 1$ and $\mathscr{E}(k^n) = \emptyset$.
2: Set $i = j_l$ and $\mathscr{E}_1 = \emptyset$.
3: Compute $\psi(i) = \eta_i$, and set $i = \psi(i)$.
4: If $i < k^n$ and ($i \notin \mathscr{E}_1$ or $i \notin \mathscr{E}(k^n)$, then $\mathscr{E}_1 = \mathscr{E}_1 \cup \{i\}$ and goto step 3;
if $i < k^n$ and ($i \in \mathscr{E}_1$ or $i \in \mathscr{E}(k^n)$, then $l = l + 1$;
if $i = k^n$, then set $\mathscr{E}(k^n) = \mathscr{E}(k^n) \cup \mathscr{E}_1 \cup \{i\}$.
5: If $l \leq |\mathscr{U}|$, then goto step 2. Otherwise output $\mathscr{E}(k^n)$ and stop.

Algorithm 1: Basin.

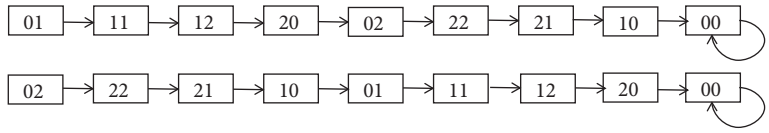

Figure 5: State diagram of NFSR1 in Example 22.

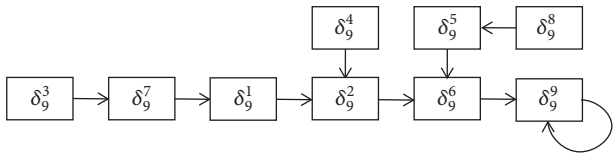

Figure 6: Basin diagram of NFSR2 in Example 22.

we have $\mathscr{E}(_{k^n}^{k^n}) = \{\delta_9^1, \delta_9^2, \delta_9^3, \delta_9^4, \delta_9^5, \delta_9^6, \delta_9^7, \delta_9^8, \delta_9^9\}$. All those features are consistent with the logic network expression of its state diagrams, which are shown in Figure 6. In the following, we strive to give an algorithm to obtain the basin $\mathscr{E}(_{k^n}^{k^n})$. We define $\mathscr{E}(k^n)$ to be the set of elements that represent the positions of the entry 1s of all elements in $\mathscr{E}(_{k^n}^{k^n})$. Precisely speaking, if $\mathscr{E}(_{k^n}^{k^n}) = \{\delta_{k^n}^{i_1}, \delta_{k^n}^{i_2}, \ldots, \delta_{k^n}^{i_m}\}$, then $\mathscr{E}(k^n) = \{i_1, i_2, \ldots, i_m\}$. For the sake of convenience, we also called $\mathscr{E}(k^n)$ the basin. For an n-stage NFSR, we first find its starting states according to Lemma 24. Let $\mathscr{U}$ be a set of starting states, and denote its cardinality as $|\mathscr{U}|$. Suppose that the starting state set $\mathscr{U}$ has been obtained in terms of Lemma 24, and its elements are denoted by $\delta_{k^n}^{j_l}, l = 1, 2, \ldots, |\mathscr{U}|$. Let $\mathscr{M} = \{j_1, j_2, \ldots, j_{|\mathscr{U}|}\}$, which is a set of the positions of the entry 1s of all the elements in the starting states $\mathscr{U}$. Second, we assume that the state transition matrix of the NFSR $L = \delta_{k^n}[\eta_1 \ \eta_2 \ \cdots \ \eta_{k^n}]$ is known. Following by $L\delta_{k^n}^i = C_i(L)$, we define a mapping

$$\psi : \psi(i) = \eta_i. \tag{21}$$

Actually, Note that any starting state of an NFSR eventually reaches a cycle and keeps staying on it. $\mathscr{E}(k^n)$ is constituted by the starting states that eventually reach the state $\delta_{k^n}^{k^n}$ and the states that those starting states go through. Finally, we need to take away repeat states.

Finally, we give Algorithm 1 to obtain the basin $\mathscr{E}(k^n)$ for an n-stage NFSR based on the mapping ψ and the set $\mathscr{M}$ if we knew the starting states of the NFSR. □

4. The Construction of Stable Feedback Shift Registers over the Binary Field

An n-stage k-valued NFSR can be described as Figure 1. Let the present state of the NFSR be $\mathbf{s} = (x_1, x_2, \ldots, x_n) \in \mathscr{D}_k^n$, and then the successor of $\mathbf{s}$ can be $(x_2, x_3, \ldots, x_n, a) \in \mathscr{D}_k^n, a \in \{1, 2, \ldots, k-1\}$; that is, the state $\mathbf{s}$ can have k different successors. Then we construct directly the stable $n+1$-stage k-valued NFSRs from the stable n-stage k-valued NFSRs by the properties of D-morphism, which is not a trivial work. We will consider it in another new work, in which we will define a new mapping. Therefore, in this section, we first give a new method for constructing stable $n+1$-stage NFSRs from stable n-stage NFSRs by the properties of D-morphism over the binary field.

4.1. D-Morphism. In this subsection, we will give an overview of the D-morphism. Let n be a positive integer, and let $\mathbf{s} = (x_1, x_2, \ldots, x_n) \in \mathbb{F}_2^n$. We define $\widehat{\mathbf{s}}$, the conjugate of $\mathbf{s}$, and $\overline{\mathbf{s}}$, the dual of $\mathbf{s}$, by

$$\widehat{\mathbf{s}} = (\overline{x_1}, x_2, \ldots, x_n) \in \mathbb{F}_2^n \tag{22}$$

and

$$\overline{\mathbf{s}} = (\overline{x_1}, \overline{x_2}, \ldots, \overline{x_n}) \in \mathbb{F}_2^n, \tag{23}$$

where $\overline{x_i}$ denotes the Boolean complement of x_i. We define a mapping $D : \mathbb{F}_2^n \longrightarrow \mathbb{F}_2^{n-1}$, $n \geq 2$, as follows. For $\mathbf{s} = (x_1, x_2, \ldots, x_n) \in \mathbb{F}_2^n$ and $\mathbf{s}' = (x_1 + x_2, x_2 + x_3, \ldots, x_{n-1} + x_n) \in \mathbb{F}_2^{n-1}$,

$$\begin{aligned} &D(x_1, x_2, \ldots, x_n) \\ &\quad = (x_1 + x_2, x_2 + x_3, \ldots, x_{n-1} + x_n), \end{aligned} \tag{24}$$

and, in the sequel, we denote (24) as $D\mathbf{s} = \mathbf{s}'$.

For $\mathbf{s} = (x_1, x_2, \ldots, x_n)$ and $\bar{\mathbf{s}} = (\overline{x_1}, \overline{x_2}, \ldots, \overline{x_n})$, we have

$$D\mathbf{s} = D(x_1, x_2, \ldots, x_n) = (x_1 + x_2, x_2 + x_3, \ldots, x_{n-1} + x_n), \quad (25)$$

$$D\bar{\mathbf{s}} = D(\overline{x_1}, \overline{x_2}, \ldots, \overline{x_n}) = (\overline{x_1} + \overline{x_2}, \overline{x_2} + \overline{x_3}, \ldots, \overline{x_{n-1}} + \overline{x_n}) = (x_1 + 1 + x_2 + 1, x_2 + 1 + x_3 + 1, \ldots, x_{n-1} + 1 + x_n + 1) = (x_1 + x_2, x_2 + x_3, \ldots, x_{n-1} + x_n). \quad (26)$$

Equations (25) and (26) imply that the mapping D is a 2-to-1 mapping, and it maps $(x_1, x_2, \ldots, x_n)$ and $(\overline{x}_1, \overline{x}_2, \ldots, \overline{x}_n)$ to the same element. Conversely, each $(y_1, y_2, \ldots, y_{n-1}) \in \mathbb{F}_2^{n-1}$ has two preimages in $\mathbb{F}_2^n$ under D, which are given by

$$D_t^{-1}(y_1, y_2, \ldots, y_{n-1}) = (t, t + y_1, t + y_1 + y_2, \ldots, t + y_1 + y_2 + \cdots + y_{n-1}), \quad t = 0, 1. \quad (27)$$

In the sequel, we denote (27) as $D_t^{-1}\mathbf{s}'$, $t = 0, 1$.

Some properties of D-morphism are recalled below. The n-th-order de Bruijn graph G_n is a directed graph with 2^n vertices, labeled by the elements of $\mathbb{F}_2^n$. The vertices x and y of G_n, $x, y \in \mathbb{F}_2^n$, are jointed by an arc, directed from x to y. A factor of G_n is a partial graph of G_n, and it includes all the vertices of G_n. For example, the state graph of every nonsingular n-stage NFSR in $\mathbb{F}_2^n$ is a factor of G_n.

Lemma 26 (see [38]). *Let $\mathbf{s}$ and $\hat{\mathbf{s}}$ be a pair of conjugate states in G_n. Then $\{D_0^{-1}\mathbf{s}, D_1^{-1}\hat{\mathbf{s}}\}$ are $\{D_1^{-1}\hat{\mathbf{s}}, \{D_1^{-1}\mathbf{s}\}$ two conjugate pairs in G_{n+1}.*

The mapping D induces a graph homomorphism (called D-morphism) from the n-th-order de Bruijn graph G_n to the $(n-1)$-th-order de Bruijn graph G_{n-1} [38]. If H is a subgraph of G_n, then its D-morphism image $D(H)$ is a subgraph of G_{n-1}. Obviously, the state diagram of an n-stage NFSR is a subgraph of G_n.

4.2. Synthesis Theory of Stable FSRs

Lemma 27 (see [39]). *Let C be a cycle in G_n, and let $\mathbf{s}$ be a state on C. Then the state $D_0^{-1}\mathbf{s}$ is on one of the cycles $D_0^{-1}C$ and $D_1^{-1}C$, and $D_1^{-1}\mathbf{s}$ is on the other one.*

By Lemma 27, it is easy to obtain the following corollary.

Corollary 28. *Let G_n' be a factor of G_n, and let $\mathbf{s}$ be a state on G_n'. Then the state $D_0^{-1}\mathbf{s}$ is on one of $D_0^{-1}G$ and $D_1^{-1}G$, and $D_1^{-1}\mathbf{s}$ is on the other one.*

Theorem 29. *Let SG_n be the state diagram of an n-stage NFSR, and let $\mathbf{s}_1, \mathbf{s}_2, \mathbf{s}_3$ be pairwise different states on SG_n. If $\mathbf{s}_1$ and $\mathbf{s}_2$ are two predecessors of $\mathbf{s}_3$, then $D_0^{-1}\mathbf{s}_1$ and $D_1^{-1}\mathbf{s}_2$ are two predecessors of one of $D_0^{-1}\mathbf{s}_3$ and $D_1^{-1}\mathbf{s}_3$, and $D_1^{-1}\mathbf{s}_1$ and $D_0^{-1}\mathbf{s}_2$ are two predecessors of the other one; or $D_1^{-1}\mathbf{s}_1$ and $D_0^{-1}\mathbf{s}_2$ are two predecessors of one of $D_0^{-1}\mathbf{s}_3$ and $D_1^{-1}\mathbf{s}_3$, and $D_0^{-1}\mathbf{s}_1$ and $D_1^{-1}\mathbf{s}_2$ are two predecessors of the other one.*

Proof. Since $\mathbf{s}_1$ and $\mathbf{s}_2$ are two predecessors of $\mathbf{s}_3$, we have

$$\mathbf{s}_1 = (x_1, x_2, \ldots, x_n), \quad \mathbf{s}_2 = (\overline{x}_1, x_2, \ldots, x_n), \quad \mathbf{s}_3 = (x_2, x_3, \ldots, x_n, y) \quad \text{with } \mathbf{s}_i \in B^n,\ i = 1, 2, 3. \quad (28)$$

Then, we have

$$\begin{aligned} D_0^{-1}\mathbf{s}_1 &= (0, x_1, x_1 + x_2, \ldots, x_1 + x_2 + \cdots x_n), \\ D_1^{-1}\mathbf{s}_1 &= (1, x_1, 1 + x_1 + x_2, \ldots, 1 + x_1 + x_2 + \cdots x_n), \\ D_0^{-1}\mathbf{s}_2 &= (0, \overline{x}_1, \overline{x}_1 + x_2, \ldots, \overline{x}_1 + x_2 + \cdots x_n), \\ D_1^{-1}\mathbf{s}_2 &= (1, \overline{x}_1, 1 + \overline{x}_1 + x_2, \ldots, 1 + \overline{x}_1 + x_2 + \cdots x_n), \\ D_0^{-1}\mathbf{s}_3 &= (0, x_2, x_2 + x_3, \ldots, x_2 + x_3 + \cdots x_n + y), \\ D_1^{-1}\mathbf{s}_3 &= (1, 1 + x_2, 1 + x_2 + x_3, \ldots, 1 + x_2 + x_3 + \cdots x_n + y). \end{aligned} \quad (29)$$

Thus, if $x_1 = 0$, we have that $D_0^{-1}\mathbf{s}_1$ and $D_1^{-1}\mathbf{s}_2$ are two predecessors of $D_0^{-1}\mathbf{s}_3$, and $D_1^{-1}\mathbf{s}_1$ and $D_0^{-1}\mathbf{s}_2$ are two predecessors of $D_1^{-1}\mathbf{s}_3$; if $x_1 = 1$, we have that $D_0^{-1}\mathbf{s}_1$ and $D_1^{-1}\mathbf{s}_2$ are two predecessors of $D_1^{-1}\mathbf{s}_3$, and $D_1^{-1}\mathbf{s}_1$ and $D_0^{-1}\mathbf{s}_2$ are two predecessors of $D_0^{-1}\mathbf{s}_3$. □

By Theorem 29, it is easy to obtain the following corollary.

Corollary 30. *SG_n is the state diagram of a stable n-stage NFSR; then there exists an $(n+1)$-stage NFSR such that $D^{-1}SG_n$ is the state diagram of NFSR. Moreover, $D^{-1}SG_n$ is two self-dual in G_{n+1}.*

Theorem 31. *If SG_n is the state diagram of an n-stage NFSR with the feedback function $f(x_1, x_2, \ldots, x_n)$, then there exists an $(n+1)$-stage NFSR such that $D^{-1}SG_n$ is its state diagram. Moreover, if the feedback function of the $(n+1)$-stage NFSR is f', then $f'(x_1, x_2, \ldots, x_{n+1}) = x_{n+1} + f(x_1 + x_2, x_2 + x_3, \ldots, x_n + x_{n+1})$.*

Proof. We define that the mappings F' and F are induced by the functions f' and f, respectively; that is, for any given $(x_1, x_2, \ldots, x_{n+1}) \in B^{n+1}$, $(y_1, y_2, \ldots, y_n) \in B^n$,

$$\begin{aligned} F' &: (x_1, x_2, \ldots, x_{n+1}) \longrightarrow (x_2, x_3, \ldots, x_{n+1}, f'(x_1, x_2 \ldots, x_{n+1})), \\ F &: (y_1, y_2, \ldots, y_n) \longrightarrow (y_2, y_3, \ldots, y_n, f(y_1,, y_2, \ldots, y_n)). \end{aligned} \quad (30)$$

According to Corollary 30, there exists a one-to-one correspondence between SG_n and the self-dual $D^{-1}SG_n$. More explicitly, for any given $\mathbf{s} = (x_1, x_2, \ldots, x_{n+1}) \in B^{n+1}$,

$$DF'\mathbf{s} = FD\mathbf{s}, \quad (31)$$

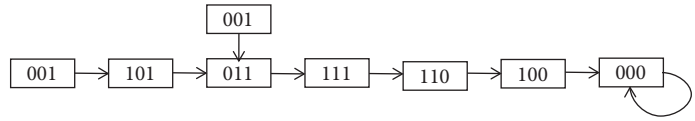

Figure 7: State diagram of NFSR with the feedback functions f in Example 32.

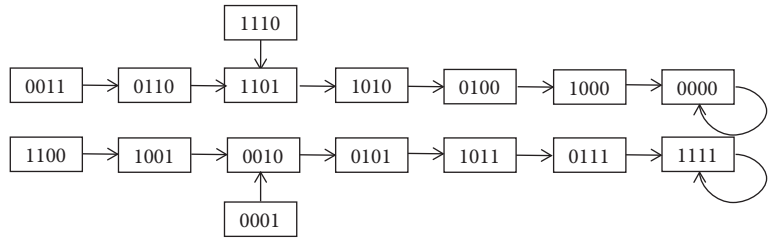

Figure 8: State diagram of NFSR with the feedback functions f' in Example 32.

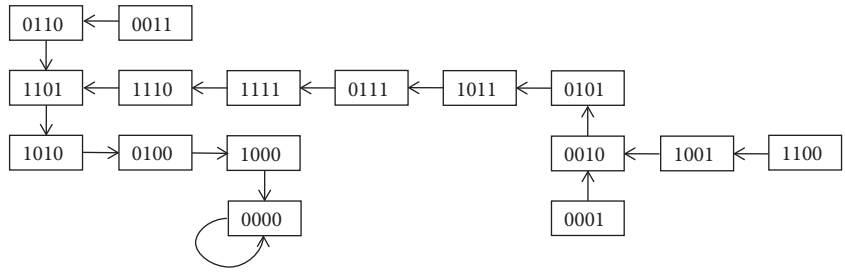

Figure 9: State diagram of NFSR with the feedback functions f'' in Example 32.

where

$$DF'\mathbf{s} = \left(x_2 + x_3, x_3 + x_4, \ldots, x_n + x_{n+1}, x_{n+1} + f'\left(x_1, x_2 \ldots, x_{n+1}\right)\right), \tag{32}$$

$$FD\mathbf{s} = \left(x_2 + x_3, x_3 + x_4, \ldots, x_n + x_{n+1}, x_{n+1} + f\left(x_1 + x_2, x_2 + x_3, \ldots, x_n + x_{n+1}\right)\right). \tag{33}$$

Then, the result follows from (31), (32), and (33); that is,

$$f'\left(x_1, x_2 \ldots, x_{n+1}\right) = x_{n+1} + f\left(x_1 + x_2, x_2 + x_3, \ldots, x_n + x_{n+1}\right). \tag{34}$$

□

Let $f(x_1, x_2, \ldots, x_n)$ be the feedback function of an n-stage NFSR1, and let

$$h\left(x_1, x_2, \ldots, x_n\right) = f\left(x_1, x_2, \ldots, x_n\right) + x_1^{a_1} x_2^{a_2} \cdots x_n^{a_n} \tag{35}$$

be the feedback function of an n-stage NFSR2, where $(a_1, a_2, \ldots, a_n) \in B^n$ and x_i^1 and x_i^0 denote x_i and $\overline{x}_i$, respectively. Note that $x_1^{a_1} x_2^{a_2} \cdots x_n^{a_n} = 1$ if and only if $x_i = a_i$ for $1 \le i \le n$. Then, the values of two functions h and f have different values only at the state $(a_1, a_2, \ldots, a_n)$.

Example 32. Consider a 3-stage stable NFSR with a feedback function

$$f\left(x_1, x_2, x_3\right) = x_2 + x_3 + x_1 x_2 + x_2 x_3. \tag{36}$$

According to Theorem 31, we obtain a 4-variant function, which is the feedback function of the 4-NFSR as follows:

$$f'\left(x_1, x_2, x_3, x_4\right) = x_4 + f\left(x_1 + x_2, x_2 + x_3, x_3 + x_4\right) = x_3 + x_1 x_2 + x_1 x_3 + x_2 x_4 + x_3 x_4. \tag{37}$$

According to (35), we obtain a 4-stage stable NFSR with a feedback function

$$f''\left(x_1, x_2, x_3, x_4\right) = f'\left(x_1, x_2, x_3, x_4\right) + x_1^1 x_2^1 x_3^1 x_4^1 = x_3 + x_1 x_2 + x_1 x_3 + x_2 x_4 + x_3 x_4 + x_1 x_2 x_3 x_4. \tag{38}$$

The state diagrams of the NFSRs with the feedback functions f, f', and f'' are shown in Figures 7, 8, and 9, respectively.

In summary, the theorems and corollaries in Section 4.1 presented a procedure for constructing stable $n+1$-stage NFSRs from stable n-stage NFSRs. Step 1 determines the feedback function of the $n+1$-stage NFSR from stable n-stage NFSR according to Theorem 31. Step 2 is used in finding the feedback function of $n+1$-stage stable NFSR from $n+1$-stage NFSR obtained by step 1.

5. Conclusion

A stable NFSR is an alternative to limit this error propagation. This paper studied the stability of multi-valued NFSRs using a logic network approach. A multi-valued NFSR can be viewed as a logic network. Based on its logic network representation, we first gave some sufficient and necessary conditions for globally (locally) stable multi-valued NFSRs. Then, explicit forms have been given for the set of basins, and the algorithm for obtaining the set of basins is provided as well. The approach used in this paper is helpful to theoretically analyze multi-valued NFSRs. Finally, the method of constructing stable NFSRs is presented, so that we can get a stable $n+1$-stage NFSR from stable n-stage NFSR by the properties of D-morphism. Nonlinear feedback shift registers are subject to impulsive effects and time-delay effects, which might be interesting to be considered in the future work.

Data Availability

No data were used to support this study.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This work was supported by the Joint Funds of the National Natural Science Foundation of China under Key Program Grant U1713212, the National Natural Science Foundation of China under Grants 61702341, 61876110, 61836005, and 61672358, the Natural Science Foundation of Guangdong Province under Grant 2017A030313338, and the Fundamental Research Project in the Science and Technology Plan of Shenzhen under Grant JCYJ20170817102218122.

References

[1] J. L. Massey and R. W. Liu, "Application of Lyapunovs direct method to the error-propagation effect in convolutional codes," *IEEE Transactions on Information Theory*, vol. 10, no. 3, pp. 248–250, 1964.

[2] F. J. Mowle, "Relations between P_n cycles and stable feedback shift registers," *IEEE Transactions on Electronic Computers*, vol. 15, pp. 375–378, 1966.

[3] F. J. Mowle, "Enumeration and classification of stable feedback shift-registers," Tech. Rept. EE-661, University of Notre Dame, Notre Dame, Ind., USA, January 1966.

[4] F. J. Mowle, "An Algorithm for Generating Stable Feedback Shift Registers of Order *n*," *Journal of the ACM*, vol. 14, no. 3, pp. 529–542, 1967.

[5] Y. Zhang, "A direct algorithm for synthesis of stable feedback shift registers," *International Journal of Electronics*, vol. 57, pp. 79–84, 1984.

[6] A. Lampel, "On *k*-stable feedback shift registers," *IEEE Transactions on Computers*, vol. 18, no. 7, pp. 652–660, 1969.

[7] J. Zhong and D. Lin, "Linearization of nonlinear filter generators and its application to cryptanalysis of stream ciphers," *Journal of Complexity*, vol. 35, pp. 29–45, 2016.

[8] J. Zhong and D. Lin, "Driven stability of nonlinear feedback shift registers with inputs," *IEEE Transactions on Computers*, vol. 64, no. 6, pp. 1–12, 2016.

[9] J. Zhong and D. Lin, "On minimum period of nonlinear feedback shift registers in grain-like structure," *IEEE Transactions on Information Theory*, vol. 64, no. 9, pp. 6429–6442, 2018.

[10] S. A. Kauffman, "Metabolic stability and epigenesis in randomly constructed genetic nets," *Journal of Theoretical Biology*, vol. 22, no. 3, pp. 437–467, 1969.

[11] S. E. Harris, B. K. Sawhill, A. Wuensche, and S. Kauffman, "A model of transcriptional regulatory networks based on biases in the observed regulation rules," *Complexity*, vol. 7, no. 4, pp. 23–40, 2002.

[12] P. Zhu and J. Han, "Stochastic multiple-valued gene networks," *IEEE Transactions on Biomedical Circuits and Systems*, vol. 8, no. 1, pp. 42–53, 2013.

[13] H. Zhang, X. Wang, and X. Lin, "Synchronization of Boolean networks with different update schemes," *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, vol. 11, no. 5, pp. 965–972, 2014.

[14] M. Aldana, "Boolean dynamics of networks with scale-free topology," *Physica D Nonlinear Phenomena*, vol. 185, pp. 45–66, 2003.

[15] J. Lizier, S. Pritam, and M. Prokopenko, "Information dynamics in small-world Boolean networks," *Artificial Life*, vol. 17, no. 4, pp. 293–314, 2011.

[16] H. Kowshik and P. R. Kumar, "Optimal computation of symmetric Boolean functions in collocated networks," *IEEE Journal on Selected Areas in Communications*, vol. 31, no. 4, pp. 639–654, 2013.

[17] D. Cheng, "Disturbance decoupling of Boolean control networks," *IEEE Transactions on Automatic Control*, vol. 56, no. 1, pp. 2–10, 2011 (Russian).

[18] G. Hochma, M. Margaliot, E. Fornasini, and M. E. Valcher, "Symbolic dynamics of Boolean control networks," *Automatica*, vol. 49, no. 8, pp. 2525–2530, 2013.

[19] H. Li and Y. Wang, "Logical matrix factorization with application to topological structure analysis of Boolean network," *IEEE Transactions on Automatic Control*, vol. 60, no. 5, pp. 1380–1385, 2015.

[20] Q. Lü and H. Li, "Event-triggered discrete-time distributed consensus optimization over time-varying graphs," *Complexity*, vol. 2017, Article ID 5385708, 12 pages, 2017.

[21] D. Cheng and H. Qi, "A linear representation of dynamics of Boolean networks," *IEEE Transactions on Automatic Control*, vol. 55, no. 10, pp. 2251–2258, 2010.

[22] Q. Liu, Q. Zeng, J. Huang, and D. Li, "Optimal intervention in semi-markov-based asynchronous probabilistic boolean networks," *Complexity*, vol. 2018, Article ID 8983670, 12 pages, 2018.

[23] S. Zhu, J. Lou, Y. Liu, Y. Li, and Z. Wang, "Event-triggered control for the stabilization of probabilistic boolean control networks," *Complexity*, vol. 2018, Article ID 9259348, 7 pages, 2018.

[24] D. Cheng, H. Qi, and Z. Li, *Analysis and Control of Boolean Networks*, Springer-Verlag, London, UK, 2011.

[25] Z. Li and D. Cheng, "Algebraic approach to dynamics of multi-valued networks," *International Journal of Bifurcation and Chaos*, vol. 20, pp. 561–582, 2010.

[26] F. Li and J. Sun, "Stability and stabilization of multivalued logical networks," *Nonlinear Analysis: Real World Applications*, vol. 12, pp. 3701–3712, 2011.

[27] C. Luo and X. Wang, "Algebraic representation of asynchronous multiple-valued networks and its dynamics," *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, vol. 10, no. 4, pp. 927–938, 2013.

[28] D. Yoshioka, "A construction method of maximum length NFSR sequences based on linear equations," in *Proceedings of the IEEE 11th International Symposium on Spread Spectrum Techniques and Applications*, pp. 185–188, Taiwan, China, 2010.

[29] H. Fredricksen and J. Maiorana, "Necklaces of beads in k colors and k-ary de Bruijn sequences," *Discrete Mathematics*, vol. 23, no. 3, pp. 207–210, 1978.

[30] T. Etzion, "An algorithm for constructing m-ary de Bruijn sequences," *Journal of Algorithms*, vol. 7, no. 3, pp. 331–340, 1986.

[31] X. Lai, "Condition for the nonsingularity of a feedback shiftregister over a general finite field," *IEEE Transactions on Information Theory*, vol. 33, pp. 747–757, 1987.

[32] H. Qi, "On shift register via semi-tensor product approach," in *Proceedings of the 32th Chinese Control Conference*, Xian, China, 2013.

[33] Z. Liu, Y. Wang, and Y. Zhao, "Nonsingularity of feedback shift registers," *Automatica*, vol. 55, pp. 247–253, 2015.

[34] H. Wang, J. Zhong, and D. Lin, "Linearization of multi-valued nonlinear feedback shift registers," *Journal of Systems Science and Complexity*, vol. 30, no. 2, pp. 494–509, 2017.

[35] H. Wang, J. Zhong, and D. Lin, "Stability of multi-valued nonlinear feedback shift registers," in *Proceedings of the IEEE International Conference on Information and Automation*, pp. 1764–1769, Ningbo, China, 2016.

[36] A. H. Roger and C. R. Johnson, *Topics in Matrix Analysis*, Cambridge University Press, UK, 1991.

[37] X. Zhang, Z. Yang, and C. Cao, "Inequalities involving Khatri-Rao products of positive semi-definite matrices," *Applied Mathematics E-Notes*, vol. 2, 2002.

[38] A. Lampel, "On a homomorphism of the de bruijn graph and its applications to the design of feedback shift registers," *IEEE Transactions on Computers*, vol. 19, no. 12, pp. 1204–1209, 1970.

[39] C. Li, X. Zeng, T. Helleseth, C. Li, and L. Hu, "The properties of a class of linear FSRs and their applications to the construction of nonlinear FSRs," *IEEE Transactions on Information Theory*, vol. 60, no. 5, pp. 3052–3061, 2014.