이 학 석 사 학 위 논 문

비선형 수축생성기의 분석

2011 년 2 월

부 경 대 학 교 대 학 원

응 용 수 학 과

권 숙 희

이 학 석 사 학 위 논 문

비선형 수축생성기의 분석

지도교수 조 성 진

이 논문을 이학석사 학위논문으로 제출함.

2011 년 2 월

부 경 대 학 교 대 학 원

응 용 수 학 과

권 숙 희

권숙희의 이학석사 학위논문을 인준함.

2011년 2월 25일

주 심 이학박사 표 용 수 (인)

위 원 이학박사 박 진 한 (인)

위 원 이학박사 조 성 진 (인)

<목 차>

<그림 목차>

<표 목차>

Analysis of the Nonlinear Shrinking Generator

Suk-Hee Kwon

Department of Applied Mathematics, The Graduate School,

Pukyoung National University

Abstract

Linear Feedback Shift Register(LFSR)s produce sequences having large periods and good statistical properties, and are readily analyzed using algebraic techniques. But the output sequences of LFSRs are also easily predictable, if we know a proper successive sequence of the sequences.

Cellular Automata(CA) is a discrete dynamical system, which consists of a uniform array of memories called cells. The states of cells in the array are updated according to a rule : the state of a cell at a given time depends only on its own state and the state of its neighbors at the previous step. Since CA has a simple, regular, modular and cascadable structure, it is useful for hardware implementation for VLSI.

In this paper, we propose a new shrinking generator which is called LCSG(Shrinking Generator based on LFSR and CA) using an LFSR with control register and CA with generator register. The proposed shrunken sequences generated by LCSG have longer periods and high complexities than the shrunken sequences generated by the known method. And we analyze the generated sequences using LCSG. Also, we propose a method for recovering the original sequence from intercepted bits by analyzing phase shifts of the output sequence using the properties of sequences generated from control register.

1. 서론

암호학과 통신공학에 많이 응용되는 의사난수열(Pseudo-Random Sequences)은 최대 주기, 자기 상관관계, 런분포 등과 같은 많은 성질을 가지고 있다. 최근 키 수열의 주기가 길고 선형복잡도가 큰 스트림 암호(Stream Cipher)를 설계하는 다양한 기법이 도입되고 있지만 대부분 LFSR(Linear Feedback Shift Register)을 다양한 형태로 결합함으로써 생성하고 있다[4,18].

분석이 용이한 수학적 구조로 되어 있는 LFSR은 유한상태기계들로 되어 있어서 최대 주기의 수열을 얻을 수 있고 우수한 통계적 성질을 갖기 때문에 스트림 암호의 기본 요소로 사용되고 있지만, 적당한 길이의 출력수열만 알면 그 다음에 출력될 모든 값을 알 수 있는 단점이 있다[15]. 이러한 이유로 비선형여과생성기, 비선형결합생성기 그리고 시각제어생성기등과 같은 LFSR의 비선형인 논리를 도입하기 위한 방법들을 스트림 암호에 도입하였다[22]. Coppersmith 등에 의해서 제안된 수축생성기(Shrinking Generator, SG)는 구현이 용이하고 작동방법이 간단하여 고속암호화가 요구되는 응용분야에 적합한 스트림 암호로 인식되고 있다[10].

셀룰라 오토마타(Cellular Automata, CA)는 구조가 단순하고 국소 상호작용을 하며, 대량의 정보를 병렬처리 할 수 있는 특징이 있다[3,6,16]. Sabater 등은 90/150 CA를 기반으로 하는 선형모델에서 수축생성기를 변환하는 알고리즘을 제안하였다[19]. 하지만 그들은 CA에서 각각의 셀들이 같은 특성다항식을 가지는 수열을 생성하는 것과 비록 90/150 CA의 주기와 수축생성기에 의해서 생성된 수열의 주기가 같을지라도 초기벡터에 따라서 짧은 주기를 가지는 수열을 가질 수 있다는 것을 고려하지 않았다.

본 논문에서는 기존의 수축생성기를 변형시킨 새로운 수축생성기로서 LFSR을 제어 레지스터로 사용하고 CA를 생성 레지스터로 사용하는 수축생성기를

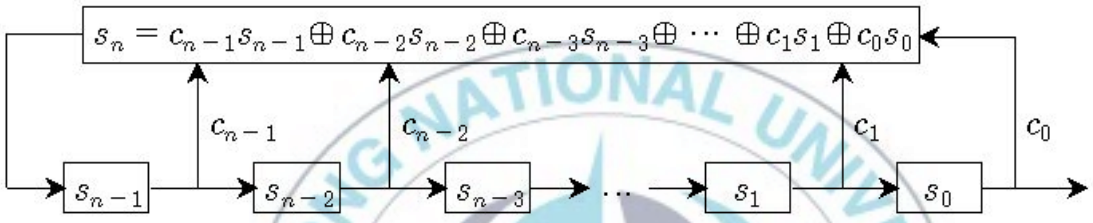
제안한다. 제안된 수축생성기에 의하여 생성된 수축수열은 기존의 수축수열보다 더욱 긴 주기를 가지고 선형복잡도가 크다. 그리고 이 수축생성기를 이용하여 생성된 수축수열을 p-비트 단위로 처리하여 기존의 방법보다 더욱 긴 주기를 갖는 수열을 생성하고 분석한다.

또한, 수축생성기에 의해 생성되는 비선형 수열을 삽입수열로 해석하여 제어레지스터에서 생성되는 수열의 성질을 이용하여 출력된 수축수열의 위상이동차를 분석하여 가로챈 일부 수열로부터 원래 수열을 복원해내는 방법을 제안한다.

2. LFSR과 CA에 대한 배경지식

n차 LFSR(Linear Feedback Shift Register)은 그림 2.1과 같이 n개의 셀과 선형 피드백 함수 $f(s_{n-1}, s_{n-2}, \ldots, s_1, s_0)$로 구성된다. LFSR로 생성되는 수열은 수학적으로는 초깃값이 $(s_0, s_1, \ldots, s_{n-1})$인 다음과 같은 점화수열이다.

$$f(s_{n-1}, s_{n-2}, \ldots, s_1, s_0) = c_{n-1}s_{n-1} \oplus c_{n-2}s_{n-2} \oplus \cdots \oplus c_1 s_1 \oplus c_0 s_0 \qquad (2.1)$$



<그림 2.1> LFSR의 구조

이러한 LFSR에 대하여 $f(x) = x^n + c_{n-1}x^{n-1} + \cdots + c_0$를 특성다항식이라 한다. 만약 LFSR에 초깃값이 모두 "0"이면 출력 값은 모두 "0"이 된다. 초깃값 중 적어도 하나가 "0"이 아니라고 가정하자. 그러면 LFSR의 단이 가질 수 있는 상태는 $2^n - 1$보다 작거나 같다. 그러므로 LFSR의 출력 수열은 주기를 갖게 되며 주기의 최댓값은 $2^n - 1$이 된다.

임의의 이진 수열을 (s_k)에 대하여 이 수열의 각 항을 계수로 하는 급수를 다음과 같이 정의한다.

$$r(x) = s_0 + s_1 x + s_2 x^2 \cdots \tag{2.2}$$

만약 주어진 수열이 주기가 있는 수열이며 주기가 t이면 대응하는 급수는 다음과 같이 표시될 수 있다.

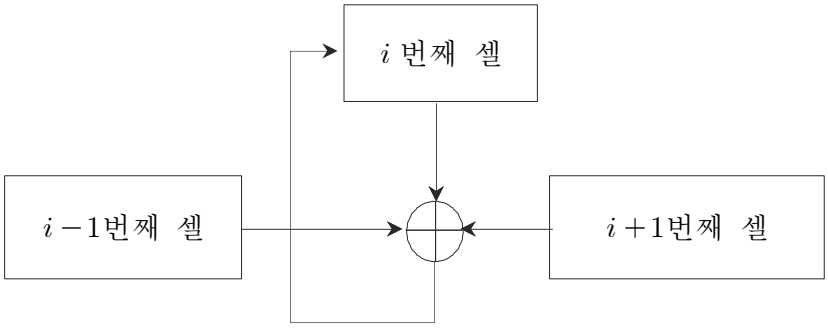
$$\begin{aligned} r(x) &= s_0 + s_1 x + s_2 x^2 + \cdots + s_{t-1} x^{t-1} + s_0 x^t + s_1 x^{t+1} + \cdots \\ &= (s_0 + s_1 x + s_2 x^2 + \cdots + s_{t-1} x^{t-1})(1 + x^t + x^{2t} + \cdots) \\ &= \frac{s_0 + s_1 x + s_2 x^2 + \cdots + s_{t-1} x^{t-1}}{1 - x^t} \\ &= \frac{g(x)}{f^*(x)} \end{aligned} \tag{2.3}$$

여기서 $(g(x), f^*(x)) = 1$이고 $f^*(x)$는 $f(x)$의 상반다항식이다[1,2].

CA란 이산 시간 하에서 동적 시스템으로 셀(cell)이라는 기본 단위 메모리의 배열로 이루어진다. 이 시스템에서 셀의 다음 상태는 어떤 규칙에 따라 정해진다. 즉, 각 셀들은 자기 자신과 이웃 셀의 함숫값에 의해 다음 상태가 결정되어 동시에 갱신된다. CA는 간단하고, 규칙적이며, 작은 단위로 확장 연결할 수 있는 구조이기 때문에 VLSI(Very Large Scale Integration) 하드웨어 구현에 알맞다[12].

셀이 선형으로 배열되어 있는 CA인 1차원 선형 CA(1-D CA)는 이산 시간의 동적 시스템으로 셀이라는 기본 단위 메모리의 배열이 선형으로 이루어져 있고, 1-D CA(One dimensional CA) 중에서 국소적 상호작용이 세 개의 셀, 즉 자기 자신과 인접한 두 개의 셀에 의해 이루어진 CA를 3-이웃 선형 CA(3-neighborhood linear CA)라고 한다. 3-이웃 선형 CA의 셀 구조는 그

림 2.2와 같다.



<그림 2.2> 3-이웃 선형 CA의 셀 구조

이 시스템에서 셀의 다음 상태는 어떤 규칙에 따라 정해진다. 즉, 각 셀들은 자기 자신과 이웃 셀의 함숫값에 의해 다음 상태가 결정되어 동시에 갱신되는데, 세 개의 셀을 가지는 CA에 대한 다음 상태전이 함수는 식 (2.4)와 같다[11].

$$q_i(t+1) = f(q_{i-1}(t), q_i(t), q_{i+1}(t)) \tag{2.4}$$

여기서 i는 일차원으로 배열되어 있는 각 셀의 위치이고 t는 시간단계이고 f는 i번째 셀의 결합논리를 가지는 국소전이(local transition) 함수이다. i번째 셀의 다음 상태는 $i-1$번째 셀, i번째 셀 그리고 $i+1$번째 셀의 현재 상태들의 일차결합을 나타내는 함숫값으로 주어진다. 그러므로 $q_i(t)$는 시각 t에서 i번째 셀의 상태를 나타낸다면 $q_i(t+1)$은 시각 $t+1$에서 i번째 셀의 상태를 나타낸다. 함수 f는 3개의 변수를 가지는 부울함수로서 2^{2^3}개가 있으며, 이것을 CA의 전이규칙(state transition rule)이라고 한다[11].

위의 규칙에 대한 결합논리는 표 2.1, 표 2,2와 같이 표현할 수 있고 $\oplus$는

XOR 논리를 나타낸다[5,8].

<표 2.1> 전이규칙

전이규칙	전이함수
60	$q_i(t+1) = q_{i-1}(t) \oplus q_i(t)$
90	$q_i(t+1) = q_{i-1}(t) \oplus q_{i+1}(t)$
102	$q_i(t+1) = q_i(t) \oplus q_{i+1}(t)$
150	$q_i(t+1) = q_{i-1}(t) \oplus q_i(t) \oplus q_{i+1}(t)$
170	$q_i(t+1) = q_{i-1}(t)$
204	$q_i(t+1) = q_i(t)$
240	$q_i(t+1) = q_{i+1}(t)$

<표 2.2> 여원전이규칙

전이규칙	전이함수
195	$q_i(t+1) = \overline{q_{i-1}(t) \oplus q_i(t)}$
165	$q_i(t+1) = \overline{q_{i-1}(t) \oplus q_{i+1}(t)}$
153	$q_i(t+1) = \overline{q_i(t) \oplus q_{i+1}(t)}$
105	$q_i(t+1) = \overline{q_{i-1}(t) \oplus q_i(t) \oplus q_{i+1}(t)}$
85	$q_i(t+1) = \overline{q_{i-1}(t)}$
51	$q_i(t+1) = \overline{q_i(t)}$
15	$q_i(t+1) = \overline{q_{i+1}(t)}$

예를 들어 표 2.3은 전이규칙 90과 150을 나타낸 표이다.

<표 2.3> 전이규칙 90과 150

이웃상태	111	110	101	100	011	010	001	000	전이규칙
다음상태	0	1	0	1	1	0	1	0	90
다음상태	1	0	0	1	0	1	1	0	150

n개의 셀로 이루어진 n-셀 90/150 $GF(2)$ 선형 CA의 상태전이함수는 $n \times n$ 행렬로 나타낼 수 있으며, 이를 상태전이행렬(state-transition matrix)이라고 한다. 이때 상태전이행렬 T는 식 (2.5)과 같은 삼중대각행렬이다[7, 9].

$$T = \begin{pmatrix} d_1 & 1 & 0 & 0 & \cdots & 0 & 0 & 0 \\ 1 & d_2 & 1 & 0 & \cdots & 0 & 0 & 0 \\ 0 & 1 & d_3 & 1 & \cdots & 0 & 0 & 0 \\ 0 & 0 & 1 & d_4 & \cdots & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & \cdots & d_{n-2} & 1 & 0 \\ 0 & 0 & 0 & 0 & \cdots & 1 & d_{n-1} & 1 \\ 0 & 0 & 0 & 0 & \cdots & 0 & 1 & d_n \end{pmatrix} \tag{2.5}$$

이후부터는 표기상의 편리함을 위하여 상태전이행렬 T를 식 (2.6)와 같이 나타내기로 한다.

$$T = < d_1, d_2, \ldots, d_i, \ldots, d_{n-1}, d_n > \tag{2.6}$$

여기서 $d_i \in GF(2)$이고 식 (2.7)와 같이 전이규칙이 90이면 “0”으로 150이면 “1”로 나타낸다.

$$d_i = \begin{cases} 0, & i\text{번째 셀의 전이규칙} = 90 \\ 1, & i\text{번째 셀의 전이규칙} = 150 \end{cases} \tag{2.7}$$

주어진 n-셀 CA의 상태전이행렬 T의 특성다항식(characteristic polynomial) $c(x)$는 $GF(2)$위에서 식 (2.8)과 같다.

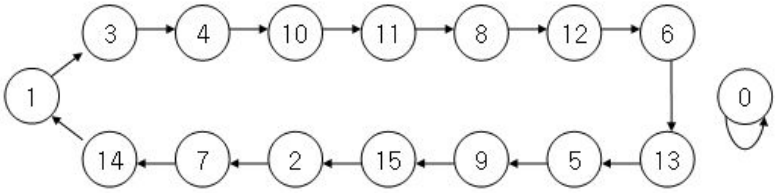
$$c(x)=|T\oplus xI| \tag{2.8}$$

여기서, I는 n차 단위행렬이다. 또, 특성다항식의 인수 중 T를 근으로 갖는 차수가 가장 낮은 다항식을 최소다항식(minimal polynomial)이라 한다. 그룹 CA의 상태전이 그래프에서 사이클의 구조는 CA의 최소다항식에 의하여 특성화된다. 90/150 CA에 대하여 상태전이행렬 T에 대한 특성다항식과 최소다항식은 같다[21].

$GF(2)$ 위에서 90/150 CA를 생각해 보자. 예를 들어 4 셀 90/150 $GF(2)$ CA $\mathbb{C}$의 상태전이규칙이 $<1,0,1,1>$이라 하고 CA-다항식이 $f(x)=x^4+x^3+1$ 이라 하면 상태전이행렬 T는 식 (2.9)과 같다.

$$T=\begin{pmatrix}1&1&0&0\\1&0&1&0\\0&1&1&1\\0&0&1&1\end{pmatrix} \tag{2.9}$$

여기서 첫 번째 셀의 왼쪽 이웃이 없으므로 첫 번째 셀의 왼쪽 이웃을 “0”이라 두고 마찬가지로 마지막 셀의 오른쪽 이웃도 “0”이라 두기로 한다. 이때 상태전이 그래프는 그림 2.3과 같다.

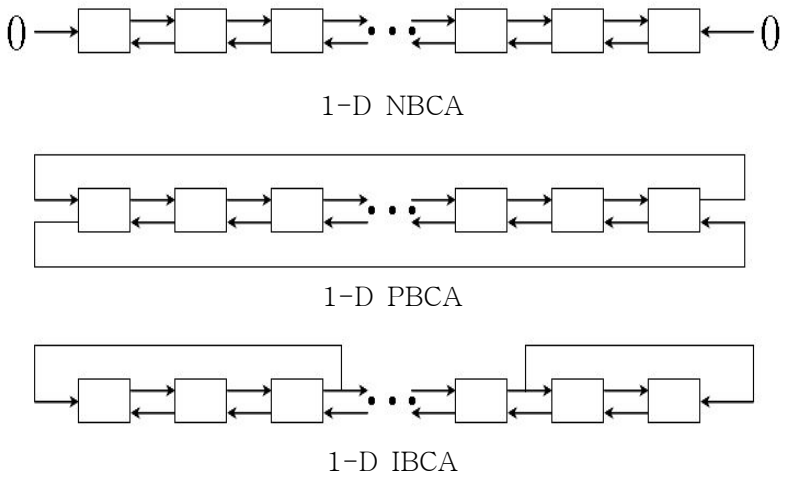


<그림 2.3> 전이규칙이 <1,0,1,1>인 상태전이 그래프

<정의 2.1[3]> CA 셀의 규칙이 XOR논리만 포함하고 있으면 선형규칙(linear rule)이라 하고 XNOR를 포함하는 규칙은 여원규칙(complemented rule)이라 한다. 모든 셀들이 선형규칙을 갖는 CA는 선형 CA라 하고 반면에 XOR 규칙과 XNOR 규칙의 조합을 갖는 CA는 가산 CA(additive CA)라 한다. AND-OR논리를 포함하는 규칙의 조합을 갖는 CA는 비가산 CA(nonadditive CA)이라 한다.

<정의 2.2[3]> 1차원 CA의 모든 셀들이 같은 규칙을 따르는 경우 그 CA는 uniform CA라 하고 그렇지 않으면 hybrid CA라 한다.

<정의 2.3[3]> CA의 제일 왼쪽과 오른쪽의 셀들이 “0” 상태와 연결되어 있으면 NBCA(null boundary CA), 양끝의 셀들이 서로 연결되어 있으면 PBCA(periodic boundary CA), 첫 번째 셀의 왼쪽 이웃을 세 번째 셀로 정의하고 마지막 셀의 오른쪽 이웃을 마지막 셀로부터 두 번째 왼쪽 셀로 정의되면 IBCA(intermediate boundary CA)라 한다.



<그림 2.4> 1-D NBCA, PBCA, IBCA 구조

이 논문에서는 특별히 언급하지 않은 CA는 모두 NBCA를 의미한다[14]. n 개의 셀로 이루어진 n-셀 CA는 $GF(2)$ 에서 연산이 이루어지는 $n \times n$ 상태전이행렬로 나타낼 수 있다. 상태전이행렬 $T = (t_{ij})$ 는 다음과 같이 만들어질 수 있다.

- $t_{ij} = 1$: i 번째 셀의 다음 상태가 현재 j 번째 셀에 영향을 받는 경우
- $t_{ij} = 0$: 그 외의 경우

CA의 다음 상태는 현재 상태 벡터와 행렬의 곱으로 얻어진다. 만약 $f_t(x)$ 가 시간 t인 순간 CA의 상태를 나타낸다면, 시간 $t+1$ 순간의 상태와 $t+2$ 인 순간의 상태는 아래의 식으로 표현될 수 있다 :

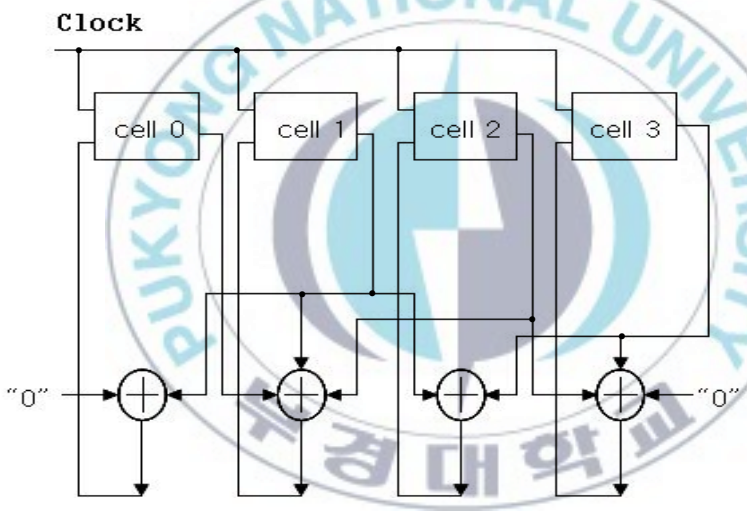
$$f_{t+1}(x) = T \cdot f_t(x),\ f_{t+2}(x) = T \cdot f_{t+1}(x) = T^2 \cdot f_t(x) \qquad (2.10)$$

같은 방법으로 p 단계 후의 상태는 다음과 같다.

$$f_{t+p}(x) = T^p \cdot f_t(x) \tag{2.11}$$

<정의 2.4> 인수분해 되지 않는 n차 다항식 $p(x)$가 $x^m - 1$을 나눌 때, m의 최솟값이 $2^n - 1$ 인 다항식을 n차 원시 다항식(primitive polynomial)이라 한다.

최대길이를 갖는 CA(Maximum Length CA, MLCA)라는 것과 그 상태전이 행렬의 특성다항식이 원시다항식이라는 것이 동치임은 잘 알려져 있다.



<그림 2.5> 전이규칙 $<0,1,0,1>$를 갖는 최대길이 NBCA

<예제 2.5> rule $<0,1,0,1>$를 갖는 4-셀 NBCA(그림 2.5)는 아래의 상태전이행렬로 표현될 수 있다 :

$$T=\begin{bmatrix}0&1&0&0\\1&1&1&0\\0&1&0&1\\0&0&1&1\end{bmatrix}$$

현재 상태 벡터가 $f_t(x)=[0101]$이면 그 다음 상태 벡터는

$$f_{t+1}(x)=T\cdot f_t(x)=\begin{bmatrix}0&1&0&0\\1&1&1&0\\0&1&0&1\\0&0&1&1\end{bmatrix}\cdot\begin{bmatrix}0\\1\\0\\1\end{bmatrix}=\begin{bmatrix}1\\1\\0\\1\end{bmatrix}$$

으로 얻을 수 있다. 이후부터는 역상태전이는 행렬론의 방법으로 구하겠다.

T의 특성다항식 $p(x)$는 x^4+x+1이고 이것은 원시다항식이므로 $<0,1,0,1>$는 MLCA이다.

그룹 CA(group CA)는 정칙인 행렬 T에 대응되므로 역행렬의 관점에서 CA의 특성화도 가능하다. 그룹 CA에 대해서는 아래의 식을 만족하는 정수 k가 존재한다[17].

$$T^k=I \quad \Leftrightarrow \quad T^{-1}=T^{k-1}$$

앞으로 진행되는 상태전이가

$$f_{t+1}(x)=T\cdot f_t(x)$$

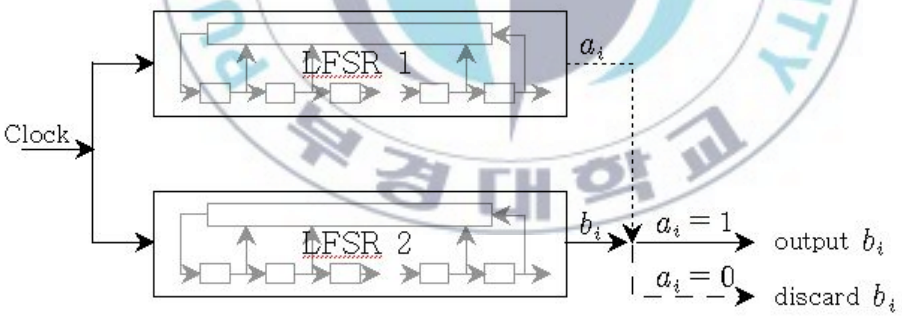
으로 표현된다면 역으로 진행되는 상태전이는

$$f_t(x) = T^{-1} \cdot f_{t+1}(x)$$
$$= T^{k-1} \cdot f_{t+1}(x)$$

와 같이 표현된다.

3. LFSR과 CA 기반의 수축생성기(LCSG)

이 장에서는 기존의 수축생성기를 변형시킨 새로운 수축생성기로서 LFSR을 제어 레지스터로 사용하고 CA를 생성 레지스터로 사용하는 수축생성기를 제안한다. 스트림 암호에서 키 수열의 생성기로 잘 알려져 있는 수축생성기는 그 작동방식이 간단하고 또한 구현이 용이하여 고속의 암호화가 요구되는 응용분야에 적합한 스트림 암호로 인식되고 있다. 이 생성기는 두 개의 LFSR R_1과 R_2가 이용되며 R_1이 R_2에 의해서 생성되는 수열의 일부분을 선택하기 위한 목적으로 사용된다. 따라서 수축생성기에 의해서 생성되는 키 수열은 LFSR R_2에 의해서 생성되는 수열의 수축된 형태가 된다. 즉, LFSR R_1과 R_2에 클럭 신호가 가해짐에 따라서 LFSR R_1의 출력이 "1"이면 LFSR R_2의 출력이 키 수열의 일부가 되고, 반면에 LFSR R_1의 출력이 "0"이면 LFSR R_2의 출력이 무시된다. 그림 3.1은 수축수열을 생성하는 수축생성기 구조이다.



<그림 3.1> 수축생성기

예를 들어 R_1의 길이가 3이라 하자. 특성다항식이 $x^3 + x^2 + 1$이고 초기 수

열벡터가 $\{100\}$이라고 하면 R_1에 의해 생성되는 수열 a_i는 식 (3.1)과 같이 $\{1001110\ldots\}$이고 주기는 7이다. 그리고 R_2의 길이가 4이고 특성다항식이 x^4+x^3+1이며 초기 수열벡터가 $\{1000\}$이라고 하면 R_2에 의해 생성되는 수열 b_i는 식 (3.2)와 같이 $\{100011110101100\ldots\}$이고 주기는 15이다. 이때 얻어지는 수축수열 c_i는 식 (3.3)과 같이 주기가 15×4=60인 수열이다.

$$a_i : 1001110100111010011101001110100111010011101001110\ldots \quad (3.1)$$

$$b_i : 100011110101100100011111010110010001\ldots \quad (3.2)$$

$$c_i : 10111011000111010000\ldots \quad (3.3)$$

<정리 3.1[10]> 두 개의 LFSR R_1과 R_2로 구성된 수축생성기에 대하여 R_1의 길이가 L_1이고, R_2의 길이가 L_2이면 수축생성기에 의해 생성된 수축수열의 주기 Ord와 선형복잡도 LC는 다음 식 (3.4)과 식 (3.5)을 만족한다.

$$Ord = (2^{L_2}-1)2^{(L_1-1)} \quad (3.4)$$

$$L_2 2^{(L_1-2)} < LC \leq L_2 2^{(L_1-1)} \quad (3,5)$$

Sabater 등은 90/150 CA에 의해서 기술된 수축생성기 선형모델을 고려하였다[20]. 그들은 LFSR 기반인 수축생성기와 동등한 수축생성기로 90/150 CA를 합성하는 알고리즘을 제안하였다. 이 CA는 최대길이를 갖는 90/150 CA와 CA의 성분을 변형하는 mirror image를 사용하여 구성한다. 알고리즘에 의해 생성된 90/150 CA의 특성다항식은 LFSR 기반인 수축생성기의 특성다항식과 같다.

이 수축생성기에 의해서 생성된 수축수열의 주기와 선형복잡도가 같은 수열을 생성하기 위해 특성다항식이 $f(x)^n$인 90/150 CA가 필요하다. 여기서 $f(x)$는 차수 L_2(LFSR의 단의 개수)인 원시다항식이고 n은 $2^{(L_1-2)} < n \leq 2^{(L_1-1)}$을 만족한다. 이를 이용하여 특성다항식 $(x^5+x^2+1)^4$인 CA를 합성한다. 그러나 모든 사이클이 같은 주기를 갖는 것이 아니다. 표 3.1은 특성다항식이 $(x^5+x^2+1)^4$인 CA의 전이규칙과 사이클 구조이다.

<표 3.1> 수축생성기 CA의 전이규칙과 사이클 구조

특성다항식	전이규칙	사이클 구조
x^5+x^2+1	<11110>	1(1), 1(31)
$(x^5+x^2+1)^2$	<1111111111>	1(1), 1(31), 16(62)
$(x^5+x^2+1)^4$	<1111111100111111111>	1(1), 1(31), 16(62), 844(124)
$(x^5+x^2+1)^8$	<1111111100111111110 0111111110011111111>	1(1), 1(31), 16(62), 844(124), 4435510400(248)

표 3.1에서 16(62)은 길이 62인 사이클이 16개 있음을 의미한다. 표 3.1에 의하면 특성다항식의 형태가 $f(x)^n$인 90/150 CA에 의해 생성된 수열의 주기는 초기벡터에 따라 일정하지 않음을 알 수 있다. 이것은 90/150 CA가 LFSR기반의 수축생성기와 달리 특정한 초기벡터에 대해 안전하지 않고, CA는 LFSR과 달리 각 셀에서 같은 특성다항식을 갖는 서로 다른 수열이 출력되며 기준이 되는 한 셀에 대한 수열의 위상이동차가 불규칙적이다. 이러한 CA의 성질이 같은 클럭에서 LFSR보다 훨씬 많은 수열을 출력을 할 수 있게 한다.

<정리 3.2[10]> 두 개의 LFSR R_1과 R_2로 구성된 수축생성기에 대하여 R_1, R_2의 길이가 각각 L_1과 L_2라 하자. R_2의 특성다항식이 $f_2(x)$일 때 수축생성

기에 의해 생성된 수열의 특성다항식 $h(x)$는 식 (3.6)를 만족한다.

$$h(x) = f_2^*(x^q) = [f_2^*(x)]^q \tag{3.6}$$

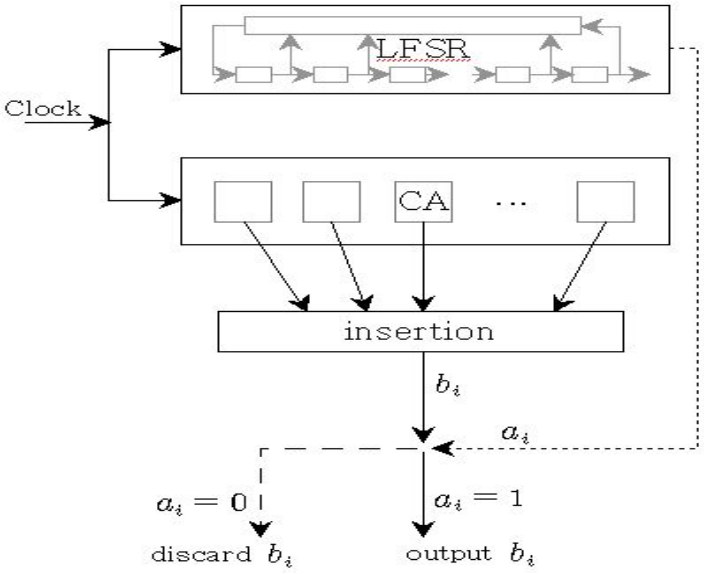
여기서 $q = 2^{L_1} - 1$이고 λ는 $f_2(x)$의 원시 근이며 $f_2^*(x)$는 다음 식 (3.7)과 같다.

$$f_2^*(x) = (x-\lambda^q)(x-\lambda^{2q})(x-\lambda^{2^2q})\cdots(x-\lambda^{2^{L_2-1}q}) \tag{3.7}$$

CA는 LFSR과 달리 각 셀에서 같은 특성다항식을 갖는 서로 다른 수열이 출력되며 기준이 되는 한 셀에 대한 수열의 위상이동차가 불규칙적이다[22]. 이러한 CA의 성질이 같은 클럭에서 LFSR보다 훨씬 많은 수열을 출력을 할 수 있게 한다.

제안하는 LCSG는 특성다항식이 원시다항식인 최대길이 LFSR과 CA 각 한 개로 이루어지는데 이 때 LFSR과 CA의 길이는 서로소가 되게 구성한다. 이 생성기에서 LFSR은 제어 레지스터로 사용되고, CA는 생성 레지스터로 사용된다. LCSG는 LFSR의 길이가 L_1 이고 CA의 길이가 L_2일 때 기존의 수축생성기와 달리 한 클럭에 L_2비트씩 출력할 수 있다. 또한 LCSG에 의해 생성된 키스트림은 CA에 의해 생성된 수열의 수축된 형태가 된다.

그림 3.2는 제안된 LCSG의 구조이다. 그림 2에서 LFSR과 CA에 클럭 신호가 가해짐에 따라서 LFSR의 출력이 “1”이면 CA의 L_2비트 출력이 키스트림으로 출력되고 LFSR의 출력이 “0”이면 CA의 출력이 제거된다.



<그림 3.2> LFSR과 CA 기반의 수축생성기

<정리 3.3> 길이가 L_1 특성다항식이 원시다항식인 LFSR과 이고 CA의 길이가 L_2이며 CA의 특성다항식 $c(x)$인 LCSG에 의하여 생성되는 수축수열의 주기는 식 (3.8)과 같다.

$$2^{L_1-1} \cdot (2^{L_2}-1) \cdot L_2 \tag{3.8}$$

(증명) LFSR에 의해 생성되는 수열의 주기는 $2^{L_1}-1$이다. 또한 CA의 특성다항식이 L_2차 원시다항식이므로 CA에 의해 생성되는 $GF(2^{L_2})$ 위에서의 수열의 주기는 $2^{L_2}-1$이다. 따라서 두 수열의 주기에 대하여 L_1과 L_2가 서로소이므로 $\gcd(2^{L_1}-1, 2^{L_2}-1)=1$이다. 그러므로 LCSG에 의해 생성되는 수열의 한 주기를 생성하기 위해서 $(2^{L_1}-1)\cdot(2^{L_2}-1)$ 클럭이 필요하고 LFSR에 의해 생

성되는 수열의 한 주기 동안 “1”은 2^{L_1-1}번 생성되므로[14] $(2^{L_1}-1)\cdot(2^{L_2}-1)$ 클럭 동안 CA에 의해 생성된 수열 중 키스트림에 포함되는 경우는 $2^{L_1-1}\cdot(2^{L_2}-1)$번 이므로 이것이 LCSG에 의해 생성된 수열의 주기이다. □

<정리 3.4> 길이가 L_1인 LFSR과 길이가 L_2인 CA로 구성된 LCSG에 대하여 CA의 특성다항식이 $c(x)$일 때 LCSG에 의하여 생성되는 수열의 최소다항식 $m(x)$은 식 (3.9)와 같다.

$$m(x)=f^*((x^{L_2})^{2^{L_1-1}})=[f_2^*(x)]^{2^{L_1-1}} \tag{3.9}$$

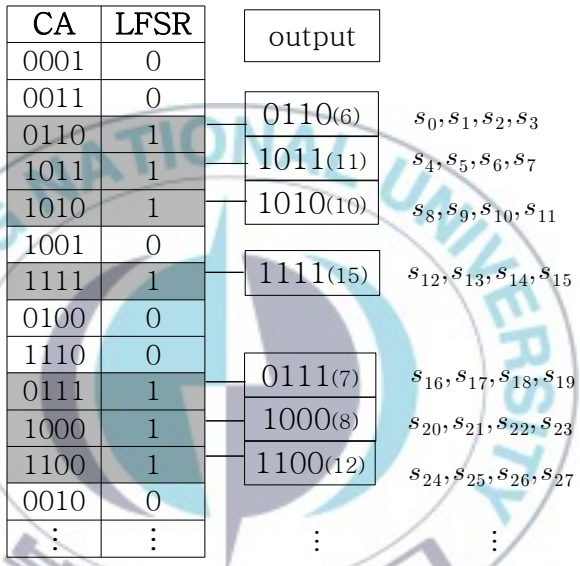
여기서 $f^*(x)$는 $c(x)$의 원시근을 α라 하고 $q=2^{L_1}-1$라 할 때, 식 (3.10)과 같다.

$$f^*(x)=(x-\alpha^q)(x-\alpha^{2q})\cdots(x-\alpha^{2^{L_2-1}q}) \tag{3.10}$$

(증명) LCSG에 의해 생성된 수열을 $s_0s_1s_2\ldots s_{t-1}s_t\,s_{t+1}\ldots s_{2t-1}s_{2t}\ldots$라 하면 LCSG에 의해 생성된 수열을 $t(=2^{L_1-1})$개씩 잘라 나열하였을 때 i열에 놓이는 수열은 s_i, s_{t+i}, s_{2t+i}, ...과 같다. 이 수열은 CA의 상태를 출력한 $GF(2^{L_2})$ 수열의 수축된 형태로 shrunken 수열이다. 그러므로 그 특성다항식은 식 (14)에 의해 $f^*(x)$이다. 따라서 수열 $\{S\}$는 2^{L_1-1}개의 열이 삽입된 형태이므로 $GF(2^{L_2})$ 상에서 식 (3.9)과 같이 확장시킬 수 있으므로 수열 $\{S\}$의 최소다항식 $m(x)=f^*(x^{2^{L_1-1}})$ 이다. □

<예제 3.5> LFSR 의 길이가 $L_1=3$이고 특성다항식이 $f_1(x)=x^3+x^2+1$이라 하자. 또한 길이가 $L_2=4$이고 특성다항식이 $f_2(x)=x^4+x^3+1$인 90/150 CA의 상태전이행렬 $T=<1,1,0,1>$로 구성된 LCSG에 의해 생성되는 수열은 그림 3.3과 같고 이 shrunken 수열의 주기는 식 (3.8)에 의하여 식 (3.11)과 같이 주기가 240으로 표 3.2와 같다.

$$2^{L_1-1} \cdot \left(2^{L_2}-1\right) \cdot L_2 = 4\times 15\times 4 = 240 \tag{3.11}$$



<그림 3.3> 수축수열의 생성과정

또 이 수축수열의 최소다항식 $m(x)$는 식 (3.9)에 의하여 식 (3.12)과 같다.

$$m(x)=\left(\left(x^{L_2}\right)^{2^{L_1-1}}\right)^4+\left(\left(x^{L_2}\right)^{2^{L_1-1}}\right)+1=((x^4)^4)^4+((x^4)^4)+1=(x^4+x+1)^{16} \tag{3.12}$$

<표 3.2> 수축수열을 $L_2 = 4$열로 나열

0110	1011	1010	1111	0111	1000	1100	0101	0011	0110
1011	1001	1110	0111	1000	0010	0011	0110	1010	0100
1110	0111	1100	1101	0001	0011	1011	1111	0100	1110
1000	0101	1101	0001	1011	1010	1001	0100	1000	1100
0010	1101	0110	1011	1010	1111	0111	1000	1100	0101
0011	0110	1011	1001	1110	0111	1000	0010	0001	0011
…									

표 3.2는 예제 3.5의 LCSG에 의해 생성되는 수열을 $GF(2^4)$의 원소로 나타내어 $L_2 = 4$열로 나열한 것이다.

4. 수축수열의 분석

이 장에서는 제 3장에서 제안한 특성다항식이 원시다항식인 최대길이 LFSR과 CA로 이루어진 LCSG에 대하여 분석한다.

<정리 4.1> 길이가 L_1인 LFSR과 길이가 L_2인 CA로 구성된 LCSG에 의하여 생성된 $GF(2^{L_2})$수열을 L_2개의 열로 나열하면 각 열들의 상태전이행렬 M은 식 (4.1)과 같다.

$$M = T^{2^{L_1}-1} \tag{4.1}$$

<표 4.1> $L_1 = 3$, $L_2 = 4$인 수축수열

6	11	10	15
7	8	12	5
3	6	11	9
14	7	8	2
1	3	6	10
4	14	7	12
13	1	3	11
15	4	14	8
5	13	1	6
9	15	4	7
2	5	13	3
10	9	15	14
12	2	5	1
11	10	9	4
8	12	2	13
⋮	⋮	⋮	⋮

(증명) LCSG에 의해 생성된 수열을 $s_0 s_1 s_2 \ldots s_{t-1} s_t\, s_{t+1} \ldots s_{2t-1} s_{2t} \ldots$라 하면 LCSG에 의해 생성된 수열을 $t(=2^{L_1-1})$개씩 잘라 나열하였을 때 i열에 놓

이는 수열은 s_i, s_{t+i}, s_{2t+i}, ...과 같다. s_i, s_{t+i}, s_{2t+i}, ...의 특성다항식은 정리 3.4에 의해 식 (3.9)과 같다. 즉 이는 CA의 상태전이행렬이 T일 때 $s_{t+i} = T^{2^{L_1-1}} \cdot s_i$임을 만족한다. □

<예제 4.2> LFSR $\mathbb{R}$의 길이가 $L_1 = 3$이고 특성다항식이 $f_1(x) = x^3 + x^2 + 1$이라 하자. 또한 길이가 $L_2 = 4$이고 특성다항식이 $f_2(x) = x^4 + x^3 + 1$인 90/150 CA $\mathbb{C}$의 상태전이행렬 T는 $T = <1,1,0,1>$이다. 이 LCSG에 의하여 생성된 수열을 $2^{L_1-1} = 4$개의 열로 나열하면 표 4.1과 같은데 이때 표 4.1의 각 열들의 상태전이행렬 M은 식 (4.1)에 의하여 식 (4.2)과 같다.

$$M = T^{2^3-1} = T^7 = \begin{pmatrix} 0 & 1 & 1 & 0 \\ 1 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \end{pmatrix} \tag{4.2}$$

이때 초기벡터를 $1\,(GF(2^4))$이라고 하면 M에 의해서 생성되는 수열은 식 (4.3)과 같다.

$$1, 4, 13, 15, 5, 9, 2, 10, 12, 11, 8, 6, 7, 3, 14, \ldots \tag{4.3}$$

표 4.1의 각 열들은 식 (4.3) 수열의 이동으로 표현할 수 있다. 표 4.1의 길이가 60인 수열 중에서 연이은 12개의 수열을 알면 그것들의 조합으로 16개의 수열(8개 1쌍과 서로 다른 위치를 갖는 4개 2쌍)과 그 위치를 알아본다. 길이가 L_1인 LFSR과 길이가 L_2인 CA로 구성된 LCSG에 의하여 생성된 수열 중에서 임의로 연이은 12개의 수열인 6, 11, 10, 15, 7, 8, 12, 15, 3, 6, 11, 9 를

안다고 하면 L_2만큼 차이를 두어 나열한 길이 8인 수열과 길이 4인 수열 7, 8, 12, 5, 3, 6, 11, 9와 3, 6, 11, 9을 구하면 표 4.2와 같이 길이 12인 수열은 α^0이라 하고 길이 8인 수열은 α^1이라 하고 길이 4인 수열은 α^2이라고 할 수 있다.

<표 4.2> 길이가 12, 8, 4인 수열

1	6	11	10	15	7	8	12	5	3	6	11	9
	0110	1011	1010	1111	0111	1000	1100	0101	0011	0110	1011	1001
α	7	8	12	5	3	6	11	9				
	0111	1000	1100	0101	0011	0110	1011	1001				
α^2	3	6	11	9								
	0011	0110	1011	1001								

$x^4+x+1=0$의 원시근을 α라고 하면 표 4.3 에 의하여 $1+\alpha=0011=\alpha^4$가 되고 여기서 "+"는 $GF(2^4)$에서의 덧셈을 의미한다. 이는 LCSG에 의해 생성된 수열을 2^2개 씩 나누어 표 4.1와 같이 표현했을 때 각 열의 수열이 연이은 두 성분 s_i, s_{t+i}에 대하여 두 성분의 합이 처음 성분에 대하여 4번 이후에 나타남을 의미한다. 여기서 $t=2^{L_1-1}$이다. 즉 $s_i+ s_{t+i}= s_{4t+i}$이다. 따라서 표 4.1의 각 열의 성분 6과 7에 대하여 $s_{4t+i}=6+7=1$임을 의미한다. 또 표 4.3에 의하여 $1+\alpha^2=0101=\alpha^8$이므로 같은 이유로 $s_i + s_{2t+i}= s_{8t+i}$이므로 표 4.2에서 $s_{8t+i}=s_i+s_{2t+i}=6+3=5$이다. 같은 방법으로 표 4.3에 의하여 $1+\alpha+\alpha^2=0011=\alpha^4$이므로 $s_i+s_{t+i}+s_{2t+i}=s_{4t+i}$ 이다. 따라서 성분 6, 7, 3에 대하여 $s_{7t+i}=s_i+s_{t+i}+s_{2t+i}= 6+7+3=4$이다. 그리하여 표 4.4와 같이 $1+\alpha$, $1+\alpha^2$, $1+\alpha+\alpha^2$의 값을 알 수가 있고 표 4.5와 같이 60개의 수열 중에서 연이은 12개의 수를 알 때 16개(8개 1쌍과 서로 다른 위치를 갖는 4개 2쌍)를 알 수 있다.

<표 4.3> $x^4+x+1=0$에 의해서 생성된 유한체

i	α^i
0	0001
1	0010
2	0100
3	1000
4	0011
5	0110
6	1100
7	1011
8	0101
9	1010
10	0111
11	1110
12	1111
13	1101
14	1001

<표 4.4> $1\oplus\alpha$, $1\oplus\alpha^2$, $1\oplus\alpha\oplus\alpha^2$에 의한 계산 결과

$1\oplus\alpha$	0001	0011	0110	1010	0100	1110	0111	1100
	1	3	6	10	4	14	7	12
$1\oplus\alpha^2$	0101	1101	0001	0110				
	5	13	1	6				
$1\oplus\alpha\oplus\alpha^2$	0010	0101	1101	0011				
	2	5	13	3				

<표 4.5> 부분적으로 알고 있는 수열로부터 수축생성기로
알아낸 수열과 그 위치

6	11	10	15	7	8	12	5	3	6	11	9	14	7	8	2	1	3	6	10
4	14	7	12	13	1	3	11	15	4	14	8	5	13	1	6	9	15	4	7
2	5	13	3	10	9	15	14	12	2	5	1	11	10	9	4	8	12	2	13

5. 위상이동차를 이용한 수축수열의 복원

이 장에서는 위상이동차를 이용하여 일부 수열로부터 수축수열의 복원을 다룬다.

차수가 m인 원시다항식에 의해서 생성된 수열을 PN 수열(Pseudo-Noise sequence) 또는 최대길이를 갖는 수열이라 하고 PN 수열에서 가장 주목해야 할 점은 런분포 성질(run distribution property)이다.

<표 5.1> 원시다항식에 의해 생성된 수축생성기

원시다항식	PN 수열
x^3+x+1	0010111…
x^3+x^2+1	0011101…
x^4+x+1	000100110101111…
x^4+x^3+1	000111101011001…
x^5+x^2+1	000010010110011111000110111010 1…
x^5+x^3+1	000010101110110001111100110100 1…

<예제 5.1> 초기수열벡터가 00001일 때 원시다항식 x^5+x^3+1에 의해 생성된 PN 수열 000010101110110001111100110100 1...에서 0이 연이어 4개인 런을 길이가 4인 0-런이라 하며 중간에 11111과 같이 연이어 1인 5개인 런을 길이가 5인 1-런이라 한다. 원시다항식 x^5+x^2+1에 의해 생성된 PN 수열은 길이가 1인 0-런은 4개이며 길이가 2인 0-런은 2개이며 길이가 3인 0-런은 1개 이며 길이가 4인 0-런은 1개가 있다. 또한 길이가 1인 1-런은 4개이며 길이가 2인 1-런은 2개이며 길이가 3인 1-런은 1개 이며 길이가 4인 1-런은 없으며 길이가 5인 1-런은 1개이다.

위 예제를 일반화 하면 2^m-1인 PN 수열의 런분포는 표 5.2와 같다.

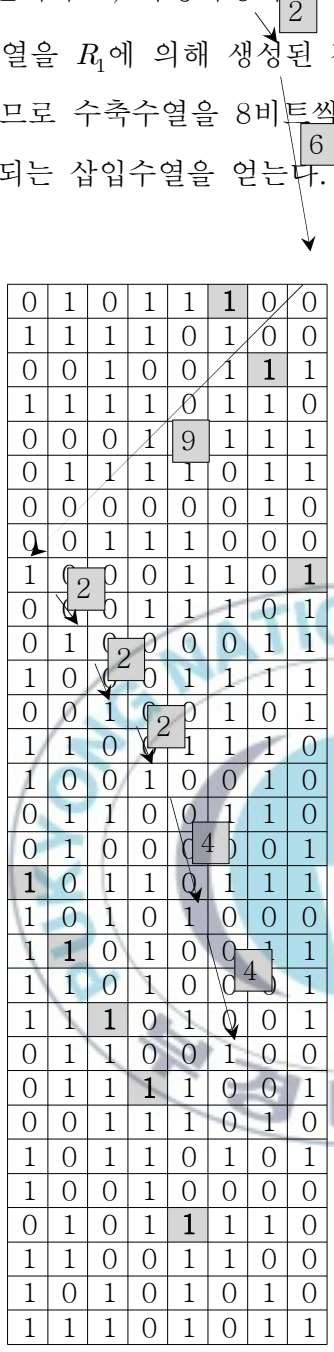
<표 5.2> PN 수열의 런분포

길 이	0-런	1-런
1	2^{m-3}	2^{m-3}
2	2^{m-4}	2^{m-4}
⋮	⋮	⋮
r	2^{m-r-2}	2^{m-r-2}
⋮	⋮	⋮
$m-2$	1	1
$m-1$	1	0
m	0	1
합 계	2^{m-2}	2^{m-2}

수축생성기에서 R_1의 길이가 4, 특성다항식이 x^4+x^3+1, 초기수열벡터가 0001이고, R_2의 길이가 5, 특성다항식이 x^5+x^3+1, 초기수열벡터가 00001일 때, 생성된 수축수열을 R_1에 의해 생성된 길이 2^4-1인 PN 수열의 한 주기에 1의 개수가 8개이므로 수축수열을 8비트씩 행으로 나열하면 그림 2와 같은 31×8행렬로 표현되는 삽입수열을 얻는다. 각 열의 수열이 연이은 두 성분 s_i, s_{t+i}에 대하여 두 성분의 합이 처음 성분에 대하여 4번 이후에 나타남을 의미한다.

수축생성기에서 R_1의 길이가 4, 특성다항식이 x^4+x^3+1, 초기수열벡터가

0001이고, R_2의 길이가 5, 특성다항식이 x^5+x^3+1, 초기수열벡터가 0001일 때, 생성된 수축수열을 R_1에 의해 생성된 길이 2^4-1인 PN 수열의 한 주기에 1의 개수가 8개이므로 수축수열을 8비트씩 행으로 나열하면 그림 5.1과 같은 31×8행렬로 표현되는 삽입수열을 얻는다.



R_1: x^4+x^3+1

R_2: x^5+x^3+1

각 열의 특성다항식

; x^5+x^2+1

각 열의 PN수열의 위상이동차

; (9,2,2,2,4,4,2,6)

<그림 5.1> 삽입수열로 표현된 $L_1=4, L_2=5$ 수축수열

위와 같은 방법으로 구한 삽입수열의 8개의 열은 x^5+x^3+1에 의해 생성된 PN 수열 중에서 x^4+x^3+1에 의해 생성된 PN 수열의 1이 있는 위치에서 15칸씩 건너뛰면서 만들어진 PN 수열로서, x^5+x^3+1의 상반다항식(reciprocal polynomial)에 의해 생성된 PN 수열들이다. 삽입수열의 각 PN 수열들은 다른 PN 수열들의 위치를 이동함으로써 얻을 수 있다. 8개의 PN 수열의 위상이동차는 R_1에 의해 생성된 PN 수열로부터 분석이 가능하다.

표 5.1에 의하면 x^4+x^3+1에 의해 생성된 주기 2^4-1인 PN 수열 000111101011001에 길이가 4인 1-런이 1개이므로 위상이동차 2가 연이어 3번 나타나고, 길이가 2인 1-런이 1개이므로 위상이동차 2가 한 번 더 나타난다. 한편 길이가 1인 0-런이 2개 존재하므로 위상이동차 4가 한번 나타나고 길이가 2인 0-런이 1개 존재하므로 위상이동차 6이 한번 나타난다. PN 수열의 주기가 31이므로 위상이동차 $31-(2+2+2+4+4+2+6)=9$가 한번 나타난다.

<정리 5.2> 차수가 L_1인 원시다항식을 특성다항식으로 가지는 LFSR R_1과 차수가 L_2인 원시다항식을 특성다항식으로 가지는 LFSR R_2로 구성된 수축생성기에 의하여 생성된 수열을 2^{L_1-1}비트씩 잘라서 나열한 행렬을 A라하면, A의 각 열은 같은 수열이며 위상이동차를 갖는다. (단, $L_1<L_2$)

(증명) L_1차 원시다항식을 특성다항식으로 가지는 LFSR R_1과 L_2차 원시다항식을 특성다항식으로 가지는 LFSR R_2로 구성된 수축생성기의 수열 생성과정은 LFSR R_1에 의해 생성된 수열의 1의 위치에서 LFSR R_2에 의해 생성된

수열을 출력하는 것이다. 따라서 LFSR R_1에 의하여 생성된 수열의 주기가 2^{L_1-1}이므로 LFSR R_1에 의하여 생성된 수열에서 첫 번째 1이 j_1번째 존재한다면 $j_1+2^{L_1}-1$번째에도 1이 존재한다. 이를 일반화하면 i번째 1이 j_i번째 존재한다면 $j_i+2^{L_1}-1$번째에도 1이 존재한다. 여기서 i는 LFSR R_1에 의해 생성된 수열의 한 주기 동안 1의 개수가 2^{L_1-1}개이므로 $1 \le i \le 2^{L_1-1}$이다. 따라서 LFSR R_2에 의해 생성된 수열을 $b_1 b_2 \ldots$라 하면, 수축생성기에 의하여 생성된 수열은 $b_{j_1} b_{j_2} \ldots b_{j_{2^{n-1}}} b_{j_1+2^n-1} b_{j_2+2^n-1} \ldots b_{j_{2^{n-1}}+2^n-1} \ldots$이다. □ (5.1)

한편, 수축생성기에 의하여 생성된 수열을 2^{L_1-1}비트씩 끊어서 나열하면 다음과 같다.

$$
\begin{array}{cccc}
b_{j_1} & b_{j_2} & \cdots & b_{j_{2^{L_1-1}}} \\
b_{j_1+2^{L_1}-1} & b_{j_2+2^{L_1}-1} & \cdots & b_{j_{2^{L_1-1}}+2^{L_1}-1} \\
\vdots & & &
\end{array}
$$

위의 각 열에서 k번째 원소와 $k+1$번째 원소는 LFSR R_2에 의하여 생성된 수열 $b_1 b_2 \cdots$에서 2^{L_1-1}칸씩 건너 띄어 가며 출력하게 되므로 각 열은 위상이동차를 갖지만 같은 수열임을 알 수 있다.

<정리 5.3> 차수가 L_1인 원시다항식으로 구성된 LFSR R_1과 차수가 L_2인 원시다항식으로 구성된 LFSR R_2로 이루어진 수축생성기에 의해 생성된 수축수열을 $(2^{L_2}-1)\times 2^{L_1-1}$ 행렬로 표현되는 삽입수열로 나타내었을 때, R_1에 의해 생성된 PN 수열에서 1이 처음 발생한 위치에서 발생한 PN 수열의 위상이동차 n은 다음을 만족한다.

$$(i+1)+(2^{L_1}-1)n \equiv i \pmod{2^{L_2}-1} \tag{5.2}$$

(증명) L_1차 원시다항식을 특성다항식으로 가지는 LFSR R_1과 L_2차 원시다항식으로 구성된 LFSR R_2로 구성된 수축 생성기에 의하여 생성된 수열을 2^{L_1-1}비트씩 끊어서 나열하면 식 (5.1)과 같고 정리 5.2에 의하여 각 열은 같은 수열이다. 여기서 i번째 열에 대한 $i+1$번째 열의 위상이동차를 k라 하면 $i+1$번째 열의 첫 번째 원소 $b_{j_{i+1}}$가 $2^{L_1}-1$씩 띄어 갈 때 k번 만에 b_{j_i}와 같아지는 것을 의미한다. 이 때 LFSR R_2에 의해 생성되는 수열의 주기가 $2^{L_2}-1$이므로 다음을 만족하는 k가 위상이동차가 된다.

$$j_i \equiv j_{i+1}+(2^{L_1}-1) \times k \,(\mathrm{mod} 2^{L_2}-1) \tag{5.3}$$

□

그림 5.1의 예제에서 $L_1=4$, $L_2=5$이다. x^4+x^3+1에 의해 생성된 주기 2^4-1인 PN 수열 000111101011001에서 $2^{4-1}=8$개의 1중에서 1이 연이어 나타나는 곳이 첫 번째와 두 번째이므로 삽입수열의 1(i)열과 2($i+1$)열의 PN 수열의 위상이동차 n은 $2+15 \cdot n=1(\mathrm{mod}31)$을 풀면 2임을 알 수 있다.

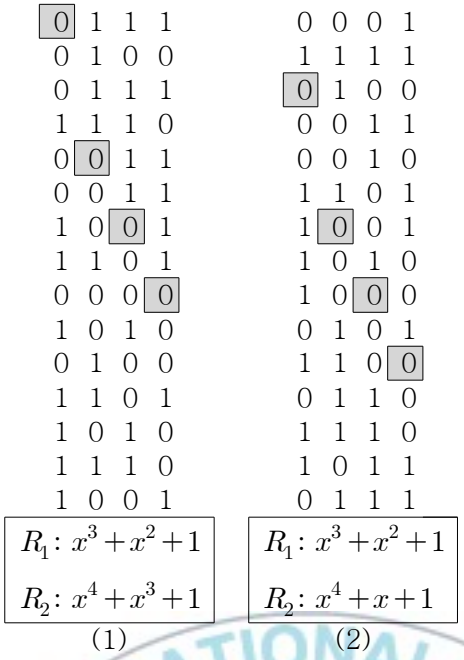
차수가 L_1인 원시다항식으로 구성된 LFSR R_1과 차수가 L_2인 원시다항식으로 구성된 LFSR R_2로 이루어진 수축생성기에 의해 생성된 수축수열을 $(2^{L_2}-1)\times 2^{L_1-1}$ 행렬로 표현되는 삽입수열로 나타내었을 때, R_1에 의해 생성된 PN 수열에서 1이 처음 발생한 위치에서 발생한 PN 수열을 삽입수열의 1열이라 하자. R_1에 의해 생성된 PN 수열에서 i번째 1과 $i+1$번째 1사이에 0이 k번 나타난다면 삽입수열의 i열과 $i+1$열의 PN 수열의 위상이동차 n은 식

(5.4)을 만족한다.

$$\{(i+1)+k\}+(2^{L_1}-1)n \equiv i \pmod{2^{L_2}-1} \tag{5.4}$$

그림 5.1의 예제에서 x^4+x+1에 의해 생성된 주기 2^4-1인 PN 수열 000111101011001에서 $2^{4-1}=8$개의 1중에서 오른쪽에서 첫 번째 1과 두 번째 1사이에 0이 두 번 나타나므로 삽입수열의 7열과 8열의 PN 수열의 위상이동차 n은 $(8+2)+15 \cdot n=7$을 풀면 6임을 알 수 있다. 같은 방법을 이용하여 2열부터 8열까지의 위상이동차를 계산하면 2, 2, 2, 4, 4, 2, 6 이고 8열과 1열사이의 위상이동차는 $31-(2+2+2+4+4+2+6)=9$임을 알 수 있다.

그러므로 R_1에 의해 생성되는 PN 수열의 1의 위치를 알면 수축수열을 삽입수열로 표현했을 때, 각 삽입수열의 위상이동차를 알 수 있다. 이러한 특성은 출력수열 중 일부만을 알 때 위상이동차를 이용하여 나머지 수열 전체를 알 수 있음을 보여주고 있다. 그림 5.2는 R_1이 같고 R_2는 길이만 같고 그 특성다항식이 다른 경우 두 수축생성기에서 출력된 키스트림을 삽입수열로 해석한 경우이다. 각 생성기에서 출력된 두 수열은 그 특성다항식이 서로 다르지만 주어진 수열을 2^{L_1-1}개의 열로 나열하여 삽입수열로 해석하였을 때 두 생성기의 수열은 달라도 각 열 사이의 위상이동차가 7, 4, 2, 2로 동일하다. 이러한 사실은 수축생성기의 구성요소인 R_1, R_2중 R_1의 정보가 더 중요함을 알 수 있다. 즉 R_1의 정보에 대해 취약하다는 것이다. [13]에서 분석한 자료와 동일한 조건으로 가로챈 출력수열의 크기를 그림 5.1의 24비트라 하고 이 수열이 000111010100001110001111이라 하자.



<그림 5.2> R_1이 같고 R_2의 길이가 다른 수축생성기

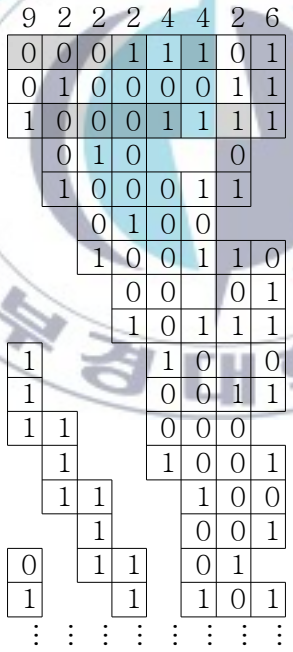
R_1의 특성다항식이 x^4+x^3+1이므로 초기벡터가 0001일 때 생성되는 수열의 주기는 000111101011001이므로 R_2의 길이가 5이므로 출력수열의 위상이동차는 9, 2, 2, 2, 4, 4, 2, 6이 된다. 가로챈 수열을 2^{L_1-1}개씩 잘라 행으로 나열하면 그림 5.3과 같다.

0	0	0	1	1	1	0	1
0	1	0	0	0	0	1	1
1	0	0	0	1	1	1	1

<그림 5.3> 가로챈 수열

PN 수열의 특성 중 4차 PN 수열은 반드시 런의 길이가 4인 경우가 1개

있으므로 이는 위상이동차가 2인 경우가 연이어 3번 나온다는 것을 의미한다. 그림 5.3에서 위상이동차가 2가 될 가능성이 있는 부분은 1열과 2열부터 3열과 4열 사이일 수 있으며 또는 2열과 3열부터 4열과 5열 사이가 될 수 있다. 그런데 주어진 R_1의 특성다항식에 의해 만들어지는 수열을 정확히 알고 있으므로 연이은 위상이동차 $2, 2, 2$에 이어 위상이동차가 2인 부분을 확인하면 그림 5.4의 윗부분에 있는 각 열의 위상이동차가 되며 첫 째 열의 위상이동차 9는 마지막 열과 첫 째 열사이의 위상이동차로 31에서 위상이동차 7개의 값을 모두 더하여 뺀 값으로 계산된다. 그림 5.4는 각 열에 대한 위상이동차를 바탕으로 각 열사이의 관계를 통하여 점차적으로 새롭게 알게 되는 출력비트들을 찾아가는 과정이다. 마지막 열에서 알게 된 비트수가 9개 이므로 이를 이용하여 출력된 수축수열의 특성다항식을 구할 수 있게 되므로 나머지 부분까지 모두 알 수 있다.



<그림 5.4> 위상이동차를 이용한 출력수열의 복원과정

6. 결론

본 논문에서는 기존의 수축생성기를 변형시킨 새로운 수축생성기로서 LFSR을 제어 레지스터로 사용하고 CA를 생성 레지스터로 사용하는 새로운 수축생성기인 LCSG를 제안하였다. 제안된 LCSG에 의하여 생성된 수열은 기존의 수축수열보다 더욱 긴 주기를 가지고 선형복잡도가 크다. 그리고 이 LCSG를 이용하여 생성된 수축수열을 p-비트 단위로 처리하여 효과적으로 분석하는 방법을 제안하였다.

또한 수축생성기에 의해 생성되는 수축수열을 삽입수열로 해석하여 위상이동차를 분석하여 일부 수열로부터 원래 수열을 복원해내는 새로운 공격방법을 제안하였다. [13]에서 제안한 방법으로 분석하게 되면 출력수열의 일부만을 재구성할 수 있으나 본 논문에서 제안된 위상이동차를 이용한 공격으로 수열 전체를 알 수 있음을 보였다.

참고문헌

[1] 조성진, "유한체 및 그 응용", 교우사, pp. 97-148 , 2007.

[2] 조성진, 김한두, 최언숙, "첨단공학을 위한 정보수학", 교우사, pp. 269-325, 2006.

[3] P.P. Chaudhuri, D.R. Chowdhury, S. Nandy and Chattopadhyay, "Additive cellular Automata Theory and Applications", Volume 1, IEEE Computer Society Press, California, 1997.

[4] S.A. Choi and K. Yang, "Balanced Shrinking Generators", LNCS 2587, pp. 213-226, 2003.

[5] S.J. Cho, U.S. Choi and H.D. Kim, "Behavior of complemented CA whose complement vector is a cyclic in a linear TPMACA", Math. Comput. Model., Vol. 36, No. 9/10, pp. 979-986, 2002.

[6] S.J. Cho, U.S. Choi, Y.H. Hwang and H.D. Kim, "Analysis of hybrid group cellular automata", ACRI 2006, LNCS, 4173, pp. 222-231, 2006.

[7] S.J. Cho, U.S. Choi, Y.H. Hwang, Y.S. Pyo, H.D. Kim and S.H. Heo, "Computing phase shifts of maximum-length 90/150 cellular automata sequences", LNCS, 3305, pp. 31-39, 2004.

[8] S.J. Cho, U.S. Choi and H.D. Kim, "Analysis of complemented CA derived from a linear hybrid group CA", Comput. Math. Appl., Vol. 53, No. 1, pp. 54-63, 2007.

[9] S.J. Cho, U.S. Choi, H.D. Kim, Y.H. Hwang, J.G. Kim and S.H. Heo, "New synthesis of one-dimensional 90/150 linear hybrid group cellular automata", IEEE Trans. Comput.-Aided Des. Integr. Circuits Syst, Vol. 26, No. 9, pp. 1720-1724, 2007.

[10] D. Coppersmith, H. Krawczyk, and Y. Mansour, "The shrinking generator", LNCS 773, pp. 22-39, 1993.

[11] A.K. Das and P.P. Chaudhuri, "Efficient characterization of cellular automata", in Proc. IEE(Part E) Vol. 137, No. 1, pp. 81-87, 1990.
[12] A.K. Das and P.P. Chaudhuri, "Vector space theoretic analysis of additive cellular automata and its application for pseudo-exhaustive test pattern generation", IEEE Trans. Comput. Vol. 42, No. 3, pp. 340-352, 1993.
[13] A. Fuster-Sabater, P. Caballero-Gil, "Concatenated automata in cryptanalysis of stream ciphers", Proc. of ACRI 2006, LNCS 4173, Springer, pp. 611-616, 2006.
[14] R.J. McEliece, Finite Fields for Computer Scientist and Engineers, Kluwer Academic Publishers, 2001.
[15] A.J. Menezes, P.C. van Oorschot and S.A. Vanstone, Handbook of Applied Cryptography, CRC Press, 1997.
[16] S. Nandi, B.K. Kar and P.P. Chaudhuri, "Theory and application of cellular automata in cryptography", IEEE Trans. Comput., Vol. 43, No.12, pp. 1346-1357, 1994.
[17] S. Nandi and P.P. Chaudhuri, "Analysis of periodic and intermediate boundary 90/150 cellular automata", IEEE Trans. Comput. Vol. 45, No. 1, pp. 1-12, 1996.
[18] A.F. Sabater and D.de.la Guia-Marinez, "Simple cellular automata-based linear models for the shrinking generator", ITW 2003, pp. 143-146, 2003.
[19] A.F. Sabater and P.C. Gil, "Concatenated automata in cryptanalysis of stream ciphers", LNCS 4173, pp. 611-616, 2006.
[20] A.F. Sabater and D.G. Martinez, "Modelling nonlinear sequence generators in terms of linear cellular automata", Appl. Math. Model., Vol. 31, pp. 226-235, 2007.

[21] M. Serra, T. Slater, J.C. Muzio and D.M. Miller, "The analysis of one dimensional linear cellular automata and their aliasing properties", IEEE Trans. Comput-Aided Design, Vol. 9, pp. 767-778, 1990.

[22] H.S. Stone, Discrete Mathematical Structures and Their Applications, Science Research, Chicago, IL, 1973.