2001년 한국정보처리학회 춘계 학술발표논문집 제8권 제1호

셀룰라 오토마타를 이용한 ElGamal 알고리즘의 구현

°이준석*, 조현호*, 이경현**, 조경연**
*부경대학교 전자계산학과
**부경대학교 전자컴퓨터정보통신공학부
e-mail:jsmagic@unicorn.pknu.ac.kr

Implementation of ElGamal algorithm using cellular automata

°Jun-Seok Lee*, Hyun-Ho Cho*, Kyung-Hyune Rhee**, Gyeong-Yeon Cho**
*Dept of Computer Science, Pukyong National University
**Division of Electronic, Computer & Telecommunication Engineering, Pukyong National University

요약

본 논문에서는 셀룰라 오토마타(Cellular Automata : CA)를 이용한 다항식 모듈라 멱승 알고리즘을 제안한다. 또한 이를 이용하여 공개키 암호 알고리즘인 ElGamal 알고리즘을 구현한다. 기존의 모듈라 멱승 알고리즘은 대부분 선형 귀환 시프트 레지스트(Linear Feedback Shift Register : LFSR)를 이용하여 구현하였다. 그러나 LFSR을 이용한 구조는 기저가 자주 변경되는 연산에 대하여 구현하기에 곤란한 단점을 가지고 있다. 본 논문에서 제안된 알고리즘은 CA의 병렬성과 높은 적응성을 이용함으로써 기저가 자주 변경되는 멱승 연산 알고리즘에 쉽게 적용할 수 있는 장점이 있다.

1. 서론

셀룰라 오토마타(Cellular Automata : CA)는 Von Neumann에 의해 최초로 제안되었고, Wolfram에 의해서 수학적 분석이 이루어졌으며 처음으로 암호학에 도입되었다[1]. 이후 Das 등에 의해서 매트릭스 대수학으로 분석이 이루어졌으며[2,3], Chaudhuri, Nandi 등이 많은 응용분야에 CA를 폭넓게 활용하였다[4]. 그리고 Muzio, Kavin 등에 의해서 LFSR에 대응하는 CA에 대한 연구가 이루어졌으며, 최소비용으로 최대 길이를 갖는 CA를 찾는 연구가 수행되었다[5,6,7]. 최근에는 Imai 등에 의해서 고속 암호 알고리즘 구현에 CA가 이용되었다[8,9].

본 논문에서는 LFSR에 기반한 연산 구조를 APCA(Advanced Programmable CA)를 이용하여 모듈라 멱승 연산을 수행할 수 있음을 보이고, 이를 이용하여 ElGamal 암호 알고리즘을 구현한다. 2장에서 CA의 기본이론과 ElGamal 암호 알고리즘을 소개하고[10], 3장에서 APCA를 이용한 모듈라 멱승 연산을 수행하기 위한 연산구조를 제안한다. 마지막으로 4장에서 결론 및 향후 연구과제에 대하여 논한다.

2. CA와 ElGamal 암호 알고리즘

2.1 셀룰라 오토마타(Cellular Automata : CA)

CA는 규칙적으로 상호 연결된 많은 셀들로 구성된 유한 상태 머신(finite state machine : FSM)이다. CA의 각 셀들은 상호 연결된 이웃의 현재 상태와 특별한 법칙에 따라 이산 시간에 동시에 새로운 상태로 갱신된다. CA를 구성하는 중요한 요소는 다음 3가지로 구분할 수 있다.

(1) 상태(state) : 셀의 값
(2) 법칙(rule) : 셀의 갱신 방정식
(3) 이웃(neighborhood) : 법칙에 참여하는 셀

상태(state)는 일반적으로 *GF(2)*상의 원소를 가지는 2-상태가 많이 응용되고 있으며, 고속 연산을 위해 *GF(q)*, *GF(2^n)* *CA*를 이용하기도 한다.

[표 1] 2-상태 3-이웃 1-차원 CA의 셀 법칙

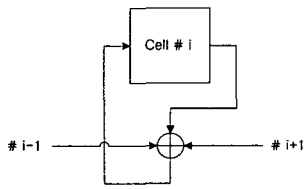
111	110	101	100	011	010	001	000
2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
0	1	0	1	1	0	1	0
1	0	0	1	0	1	1	0

[표 1]에서 가장 많은 응용이 이루어지고 있는 2-상태 3-이웃 1-차원 CA에 대하여 나타내었다. 표에서 첫 번째 행은 3-이웃으로 가능한 셀의 상태를 보여준다. 두 번째 행은 첫 번째 행의 상태 계수를 의미한다. 세 번째와 네 번째 행은 법칙90과 법칙150

을 나타낸다(90과 150은 세 번째 행과 네 번째 행의 십진 값을 말한다). 따라서, 법칙90은 셀의 갱신이 현재 상태의 R_{90} 왼쪽 이웃과 R_{90} 오른쪽 이웃의 상태 값을 이용하여 Xor 연산함으로써 얻을 수 있고, 법칙150은 왼쪽, 오른쪽 그리고 자신의 현재 상태 값을 Xor 연산한 결과이다. 이를 식으로 나타내면 다음과 같이 표현할 수 있다.

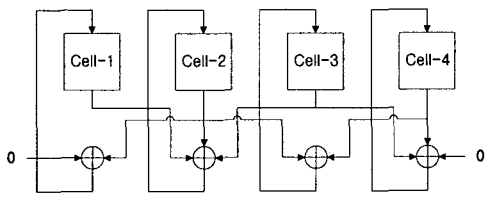
$$s_i^{t+1} = s_{i-1}^t \oplus s_{i+1}^t$$
$$s_i^{t+1} = s_{i-1}^t \oplus s_i^t \oplus s_{i+1}^t$$

여기서, s_i^t는 시간 t에서 i번째 셀의 상태를 나타내고, $\oplus$는 Xor 연산을 의미한다. 아래 (그림 1)은 3-이웃 CA의 셀 구조를 보여준다.



(그림 1) 3-이웃 CA의 셀 구조

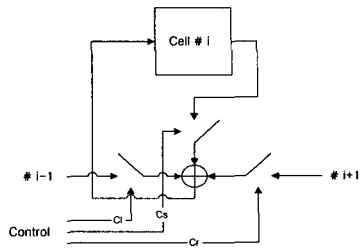
CA는 다음과 같이 여러 가지 방법으로 구분할 수 있다. 셀의 갱신에 적용된 법칙에 따라 Linear CA, Non-Linear CA 또는 Additive CA로 구분한다. CA에 적용된 법칙의 개수에 따라 Uniform CA, Hybrid CA로 구분할 수 있고, 셀이 배열되어있는 형태로 1-차원, 2-차원, 3차원 CA로 구분할 수 있다. 그리고, 첫 번째 셀/마지막 셀은 왼쪽이웃/오른쪽 이웃이 존재하지 않는다. 이를 경계조건으로 정의한다. 이 정의에 따라 NBCA(Null Boundary CA), PBCA(Periodic Boundary CA), IBCA(Intermediate Boundary CA)로 구분할 수 있다.



(그림 2) 1-D 3-이웃 NBCA의 구조

(그림 2)는 1-차원 3-이웃 NBCA의 구조를 보여준다. 여기에 사용된 법칙은 <90,150,90,150>이다.

법칙의 적용을 보다 효율적으로 하기 위해 PCA(Programmable CA) 구조가 소개되었다. 이는 Xor 게이트의 입력을 제어함으로써 갱신 시 적용되는 법칙을 조절할 수 있도록 구성한 것이다. (그림 3)이 3-이웃 PCA의 셀 구조를 보여준다.



(그림 3) 3-이웃 PCA의 셀 구조

여기서, 제어선 Cl, Cs, Cr은 각각 셀의 왼쪽, 자신 그리고 오른쪽 이웃으로부터의 입력을 제어하는 값이다. 즉, 제어 값이 '1'일 경우 입력을 받아들이는 것이다. 따라서 이산 시간에 따라 같은 셀에 다른 법칙이 적용될 수 있다. 예를 들어, Rule90과 150은 자신의 셀에 대한 제어선 Cs의 값에 따라서 같은 셀 구조로써 구성이 가능하다. 3-이웃 Linear CA일 경우 2^3개의 가능한 갱신 법칙이 존재하게 된다.

2.2 ElGamal 암호 알고리즘

ElGamal 암호 알고리즘은 현대 공개키 암호 알고리즘의 2가지 부류 중 하나인 이산대수문제에 기반한 공개키 암호 알고리즘이다.

ElGamal 암호 알고리즘은 다음 절차를 따른다.

전제 : 유한체 $GF(q)^*$와 생성원 g가 알려져 있다. 여기서 q는 임의의 큰 소수이다.

(1) 사용자 A는 $1 < a < q-1$ 사이의 정수를 임의로 선택하고 비밀키로 간직한다.

(2) 사용자 A는 $g^a \bmod q$를 계산하여 이를 암호화키로 공개한다.

(3) 사용자 B가 비밀 메시지 M을 A에게 보내고자 한다면, B는 임의의 정수 k를 선택하고, $g^{ak} \bmod q$를 계산한 후, 두 원소를 A에게 전달한다.

$$(g^k,\ Mg^{ak})$$

(4) A는 첫 번째 원소 g^k와 자신의 비밀키 a를 이용하여 g^{ak}를 계산하고, 두 번째 원소에 이를 나누어 메시지 M을 얻는다.

이때 a를 알지 못하고 $(g^k,\ Mg^{ak})$만으로 메시지 M을 구하는 것이 어려운 일임을 알 수 있다.

이 시스템을 구현하기 위해 필요한 알고리즘들은 유한체 상의 연산을 수행할 수 있는 알고리즘 즉, 모듈라 멱승 알고리즘, 곱셈 알고리즘, 역원 알고리즘 등이다.

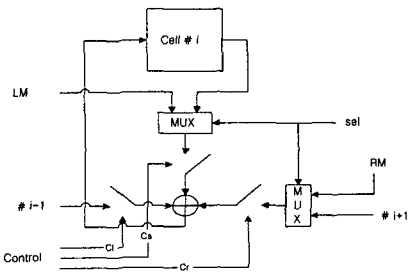
3. 제안 알고리즘

본 논문에서 제안하는 알고리즘은 $GF(2^n)$상의 원소를 이용하여 ElGamal 알고리즘을 구현하는 것이다. 이를 위해 APCA(Advanced PCA)를 소개하고 이

를 이용한 다항식의 모듈라 곱셈 알고리즘과 멱승 알고리즘, 그리고 역원 알고리즘을 제안하고 이를 구현한다.

3.1 APCA(Advanced PCA)

APCA는 PCA의 구조를 모듈라 곱셈 연산을 수행할 수 있도록 개선한 구조이다. (그림 4)가 3-이웃 APCA의 셀 구조를 보여준다.

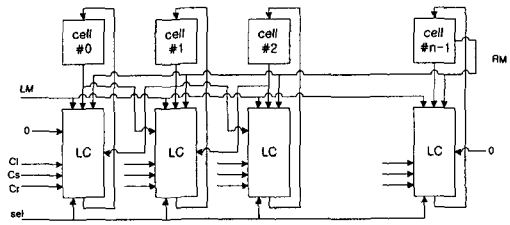


(그림 4) 3-이웃 PCA의 셀구조

여기서, LM은 곱셈, RM은 나눗셈을 위한 다항식이다. 그리고 sel은 멀티플렉서의 선택선이다. 이는 자신의 상태와 LM, 오른쪽 이웃의 상태와 RM를 선택할 수 있도록 한다. 그리고, 제어선 Cl, Cs, Cr은 Xor의 입력을 제어하는 제어선이다. 예를 들어서, Cl=1, Cs=Cr=0로 설정한다면 이 구조는 쉬프트 레지스트와 동일한 역할을 수행할 것이다. 또한 sel=1일 경우 Xor의 자신의 상태와 오른쪽 이웃의 입력으로 LM과 RM을 받아들이게 할 수 있다.

따라서 sel, Cl, Cs, Cr의 값을 적절하게 조절함으로써 APCA의 구조를 쉬프트 레지스트, 곱셈회로, 나눗셈 회로 등으로 활용할 수 있다.

3.2 APCA를 이용한 모듈라 곱셈 알고리즘



(그림 5) APCA를 이용한 곱셈 회로

일반적인 모듈라 곱셈 회로는 곱셈과 나눗셈을 동시에 수행하도록 함으로써 구성할 수 있다[11]. 이를 (그림 4)에서 보여진 APCA의 구조를 이용하여 재구성할 수 있다. 즉, sel=1, Cl=1, Cs, Cr은 곱셈과 나누셈 다항식의 계수로 설정함으로써 APCA를 이용하여 곱셈기를 구현할 수 있고, 이를 이용하여 멱승 연산을 수행할 수 있다. 이를 위한 구성을 (그림 5)에 나타내었다.

3.3 모듈라 멱승 알고리즘

$E(x)^D \bmod P(x)$와 같은 다항식의 모듈라 멱승 알고리즘은 3.2절에서 제안한 곱셈회로 2개를 이용하여 곱셈과 제곱을 동시에 수행함으로써 구현할 수 있다. 여기서 D는 임의의 n-tuple 2진 벡터이다. 즉, $(d_0, d_1, \cdots, d_{n-1})$와 같다.

알고리즘은 다음 철차를 따라 수행된다.

(1) $C = 1,\ E = E(x)$

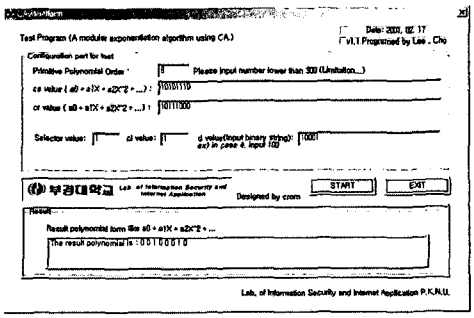
(2) 만약 $d_i = 1$ 이면, $C = C \cdot E \bmod P(x)$

(3) $E = E(x)^2 \bmod P(x)$

(4) $0 \le i \le n-1$ 동안 (2)와 (3)을 반복

(5) $C = E(x)^D \bmod P(x)$

여기서 연산은 곱셈 회로를 이용하여 (2)를 수행함으로써 얻을 수 있고, (3)의 과정은 또 하나의 곱셈회로를 응용하여 제곱 연산을 수행할 수 있다. 따라서 n번의 클럭으로 최종 연산의 결과를 얻을 수 있다.



(그림 6) 시뮬레이션 결과 화면

시뮬레이션은 Visual C++를 이용하여 수행하였으며, (그림 6)은 $E(x) = 1 + x^2 + x^4 + x^5 + x^6 = (10101110)$, $D = 17$, $P(x) = 1 + x^2 + x^3 + x^4 + x^8 = (101110001)$에 대한 결과 화면을 (그림 6)에 나타내었다.

3.4 역원 알고리즘

$g(x)$를 $GF(2^n)^*$의 생성다항식이라고 하자. 역원을 구하는 알고리즘은 $g(x)^{-a} = g(x)^{2^n-1-a}$임이 명백함으로 이전 절에서 제안된 멱승 알고리즘을 이용하여 $(2^n-1)-a$번의 멱승 연산을 수행함으로써 쉽게 구할 수 있다.

3.5 ElGamal 알고리즘의 구현

ElGamal 알고리즘의 구성에 있어서 가장 중요한 부분은 모듈라 멱승 부분이다. 이는 제안된 APCA 구조를 이용하여 용이하게 구현될 수 있다. 비밀 메시지를 암호화하여 보내고자 하는 송신측에서는 모듈라 멱승과 곱셈 알고리즘을 이용하여 송신메시지를 구성할 수 있고, 비밀 메시지를 수신하여 복호하

는 수신측에서는 멱승 알고리즘을 이용하여 역수를 구하고 이를 곱셈 알고리즘을 적용하여 메시지를 복호 할 수 있다. 알고리즘의 구현은 다음 단계를 통해 이루어진다.

· 송신측 알고리즘

step1 : $g(x)^k \bmod P(x)$

step2 : $(g(x)^a)^k \bmod P(x)$

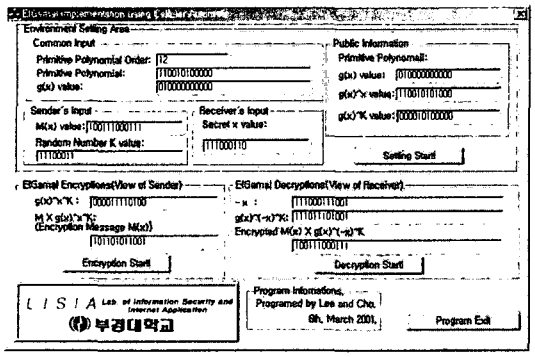
step3 : $M(x) \cdot g(x)^{ak} \bmod P(x)$

· 수신측 알고리즘

step1 : $g(x)^{-ak} = g(x)^{(2^n-1-a)k} \bmod P(x)$

step2 : $M(x) = (M(x)g(x)^{ak}) \cdot (g(x)^{-ak}) \bmod P(x)$

여기에서, $P(x)$는 n차 원시다항식이고, $g(x)$는 $GF(2^n)^*$에서의 생성다항식이다. $M(x)$는 송신측에서 수신측으로 보내고자하는 비밀 메시지에 대응하는 $GF(2^n)^*$상의 원소이다. a와 k $(1 \le a,\ k \le 2^n-2)$는 수신측과 송신측의 비밀 메시지인 임의의 정수이다.



(그림 7) 시뮬레이션 결과 화면

여기서, $P(x)=1+x+x^4+x^6+x^{12}$, $g(x)=x$를 사용하였고, $M(x)=1+x^3+x^4+x^5+x^9+x^{10}+x^{11}$, $k=227$, $a=454$ 를 사용하여 시뮬레이션 하였다.

4. 결론 및 향후 연구과제

본 논문에서는 주어진 유한체 상의 임의의 원소에 대한 모듈라 멱승 연산을 CA를 이용하여 구현하였다. 제안 방법은 APCA를 이용함으로써 승수와 피승수의 변화에도 매우 효과적으로 적용할 수 있는 장점이 있다.

또한 제안된 구조를 이용하여 이산대수문제를 기반으로 하는 공개키 암호 알고리즘인 ElGamal 암호 알고리즘의 연산을 수행할 수 있는 알고리즘을 제안하였고, 간략한 예에 대하여 Visual C++를 이용하여 구현한 결과를 나타내었다.

향후 연구과제로는 구현된 알고리즘이 실제로 활용 가능한 암호화 강도까지 알고리즘을 수정해야 할 것이고, 또한 알고리즘이 보다 고속으로 수행될 수 있도록 연구할 것이다. 이를 위해서 연산의 회수를 줄일 수 있는 연구와 하드웨어 구현이 뒤따라야 할 것으로 간주된다. 또한 제안된 알고리즘이 소인수 분해 문제를 기반으로 하는 RSA와 같은 공개키 암호 알고리즘에 적용할 수 있는 방안에 대해서도 추후 연구를 진행할 예정이다.

참고문헌

[1] S. Wolfram, *"Cellular Automata and Compexity"*, Addison-Welsely Publishing Company, 1994.

[2] A. K. Das, P. P. Chaudhuri, *"Efficient characterization of cellular automata"*, IEE Proceeding E, Vol.137, Iss.1, 1990.1, pp.81-87.

[3] A. K. Das and P. P. Chaudhuri, *"Vector space Theoretic Analysis of Addtive Cellular Automata and Its Application for Pseudoexhaustive Test Pattern Generation"*, IEEE Transaction on Computers, Vol.42, Iss.3, 1993.5, pp.340-352.

[4] P. P. Chaudhuri, et.al., *"Additive Cellular Automata Theory and Applications"*, volume 1, IEEE Computer Society Press, California, USA, 1997.

[5] K. Cattell, M. Serra, *"The Analysis of One Dimensional Multiple-Value Linear Cellular Automata"*, IEEE Transaction on Computer-Aided Design of Integrated Circuits and Systems, Vol.9, Iss.7, pp.767-778.

[6] K. Cattell, J. Muzio, *"Analysis of One-Dimensional Linear Hybrid Cellular Automata over GF(q)"*, IEEE Transactions on Computers, Vol.45, No.7, 1996.7, pp.782-792.

[7] K. Cattel, J. Muzio, *"Synthesis of One-Dimensional Linear Hybrid Cellular Automata"*, IEEE Transactions on Computer-Aided Design of Integrated Circuit and Systems, Vol.15, No.3, 1996.3, pp.325-335.

[8] M. Mihaljevic, Y. Zheng, H. Imai, *"A Fast and Secure Stream Cipher based on Cellular Automata over GF(q)"*, IEEE Global Telecommunications Conference, GLOBECOM '98, Vol.6, 1998, pp.3250-3255.

[9] M. Mihaljevic, H. Imai, *"A Family of Fast Keystream Generations based on Programmable Linear Cellular Automata over Gf(q) and Time-Variant Table"*, IEICE Transactions on Fundamentals, Vol.E82-A, No.1, 1999.1, pp.32-39.

[10] A. Menezes, P. van Oorschot, and S. Vanstone, *"Handbook of Applied Cryptography"*, CRC Press, 1996.

[11] S. Lin, D. J. Costello, Jr., *"Error Control Coding: Fundamentals and Applications"*, Prentice-Hall, Inc. Englewood Cliffs, New Jersey, 1983.