정보집합 디코딩을 위한 패리티 검사행렬 변환에 관한 연구

박민진, 이영도, 이윤진, 김동찬
국민대학교 정보보안암호수학과
{minjin_99, catsmusic, lucia3736, dckim}@kookmin.ac.kr

A Study on Transformation of Parity Check Matrix for Information Set Decoding

Minjin Park, Youngdo Lee, Yunjin Lee, Dong-Chan Kim
Kookmin University

요약

정보집합 디코딩은 주어진 패리티 검사행렬을 표준형 행렬로 변환해서 신드롬 디코딩 문제의 해를 찾는 공격 알고리듬이다. 최초로 제안된 정보집합 디코딩 알고리듬으로는 Prange 알고리듬이 있다. 패리티 검사행렬을 표준형 행렬로 변환하기 위해서 가우스 소거법을 이용한다. 하지만 모든 패리티 검사행렬이 표준형 행렬로 변환할 수 있는 것은 아니므로 이를 해결하는 연산의 방법에 대해 소개한다.

Ⅰ. 서론

랜덤선형부호의 디코딩은 안전한 공개키 암호 체계를 구축할 수 있는 난제로 이용되어 왔다. 주어진 신드롬값 s에 해당하는 해밍무게 w인 오류 벡터를 도출하는 것은 NP-hard 문제임이 입증되었다[2]. 1960년대 초 E. Prange의 연구[1] 이후, 정보집합 디코딩(Information Set Decoding, 이하 ISD)를 통해 선형부호 디코딩 문제를 해결하는 연구들이 진행되었다.

신드롬 디코딩 문제(Syndrome Decoding Problem, 이하 SDP)는 패리티 검사행렬 $H \in Mat_{(n-k)\times n}(\mathbb{F}_2)$와 신드롬 $s \in Mat_{1\times(n-k)}(\mathbb{F}_2)$, $w \in \{1,\dots,n\}$에 대해 $eH^T = s$, $w_H(e) = w$을 만족하는 오류벡터 $e \in Mat_{1\times n}(\mathbb{F}_2)$를 찾는 문제이다. 부호기반 암호 시스템에서의 패리티 검사행렬을 표준형 행렬로 변환하면 SDP를 전수조사 공격에 비해 낮은 계산 복잡도로 해를 찾을 수 있다. Prange 알고리듬[5], Lee and Brickell 알고리듬[4], Stern 알고리듬[6]에서의 ISD 알고리듬은 변환된 패리티 검사행렬을 기반으로 구성된다. 따라서 공격의 효율성을 높이기 위해서는 패리티 검사행렬을 표준형 행렬로 변환하는 것이 중요하다.

본 논문에서는 Fisher-Yates 셔플링 알고리듬[3]과 가우스 소거법[7]을 이용하여 패리티 검사행렬을 표준형 행렬로 변환하는 방법에 대해 소개한다.

본 논문의 구성은 다음과 같다. Ⅱ장에서 기호를 정의하고, Ⅲ장에서 대표적인 ISD 알고리듬인 Prange 알고리듬에 대해 소개한다. Ⅳ, Ⅴ장에서는 패리티 검사행렬을 표준형 행렬로 변환하는 과정에서 사용하는 가우스 소거법과 Fisher-Yates 셔플링 알고리듬을 소개한다. Ⅵ장에서는 패리티 검사행렬을 표준형 행렬로 변환하는 전체적인 과정을 기술한다.

Ⅱ. 정의와 기호

본 논문은 다음의 정의와 기호를 사용한다.

$w_H(u)$	벡터 u의 해밍무게 $(w_H(u) \coloneqq \lvert\{i : u_i \neq 0\}\rvert)$
$Mat_{m\times n}(\mathbb{F}_2)$	$\mathbb{F}_2$상의 성분으로 이루어진 $m \times n$ 행렬들의 집합
I_k	$k \times k$ 항등행렬
표준형 행렬	$[I_n\lvert M]$ 형태의 행렬 ($M : n \times m$행렬)
H_{sys}	H의 표준형 행렬

Ⅲ. Prange 알고리듬

Prange 알고리듬은 첫 번째로 알려진 ISD 알고리듬이다. 1962년 E. Prange는 패리티 검사행렬 H를 표준형 행렬로 변환하여 SDP의 해를 찾는 알고리듬을 제안하였다. 이 알고리듬은 SDP에서 찾을 수 있는 오류벡터 e를 추측하는 것에 기반을 두었다.

다음은 Prange 알고리듬의 계산과정이다.

(단계 1) 치환행렬 $P \in Mat_{n\times n}(\mathbb{F}_2)$를 랜덤하게 선택한다.

(단계 2) UHP가 다음과 같이 표준형 행렬이 되도록 가우스 소거법을 이용하여 가역행렬 $U \in Mat_{(n-k)\times(n-k)}(\mathbb{F}_2)$를 계산한다.

$$UHP = [I_{n-k}\lvert H'] \quad for \;\; H' \in Mat_{(n-k)\times k}(\mathbb{F}_2).$$

UHP가 표준형 행렬이 되지 않는 경우 (단계 1)로 돌아간다.

(단계 3) $w_H(sU^T) \neq w$이면 (단계 1)로 돌아간다.
$w_H(sU^T) = w$이면 오류벡터 $e' = (sU^T\lvert\lvert 0)$는 다음의 식을 만족한다.

$$(UHP)e'^T = [I_{n-k}\lvert M](sU^T\lvert\lvert 0)^T = sU^T.$$

(단계 4) $e = e'P^{-1}$를 반환한다.

Prange 알고리듬의 유사부호는 다음과 같다.

입력 : $H \in Mat_{(n-k)\times n}(\mathbb{F}_2),\ s \in Mat_{1\times(n-k)}(\mathbb{F}_2), w$
출력 : $e \in Mat_{1\times n}(\mathbb{F}_2)\ such\ that\ eH^T = s,\ w_H(e) = w$

```
repeat
    repeat
        Generate random permutation matrix P_{n×n}
        < U, [V|H'] > ← GaussianElimination(HP)
    until V = I_{n-k}
until w_H(sU^T) = w
return [sU^T|0_{1×k}]P^{-1}
```

Ⅳ. 가우스 소거법

가우스 소거법은 기본행연산을 이용해서 행사다리꼴 행렬로 변환하는 알고리듬이다. $m \times n$ 이진행렬 H에 대해 가우스 소거법을 수행하면 다음과 같이 진행된다.

(단계 1) $r \leftarrow 0,\ c \leftarrow 0$으로 초기화 한다.

(단계 2) r번째 행부터 c번째 열에 0이 아닌 값이 존재하는 행을 찾아 r번째 행과 교환한다. $(0 \le r \le m-1,\ r \le c \le n-1)$

c번째 열에 0이 아닌 값이 존재하는 행이 없다면 $c \leftarrow c+1$을 한 후, (단계 2)로 돌아간다.

(단계 3) 0번째 행부터 $r-1$번째 행까지 c번째 열이 0이 되도록 r번째 행을 행덧셈한다. $r+1$번째 행부터 $m-1$번째 행까지 c번째 열이 0이 되도록 r번째 행을 행덧셈한다.

(단계 4) $r \leftarrow r+1$, $c \leftarrow c+1$을 한 후, H가 행사다리꼴이 될 때까지 (단계 2), (단계 3)을 반복하여 수행한다.

V. Fisher-Yates 셔플링

Fisher-Yates 셔플링은 유한 집합에 대한 랜덤 전 단사 함수를 생성하는 알고리듬이다.

이 알고리듬의 유사부호는 다음과 같다.

입력 : 원소의 개수가 N개인 집합 $X = [X_0, X_1, \dots, X_{N-1}]$
출력 : $Shuffled\ array\ X$

1: ***for*** $i = N-1$ to 0 ***do***
2: $Pick\ random\ integer\ j\ such\ that\ 0 \le j < i$
3: $Swap\ X_i\ and\ X_j$
4: ***return*** X

랜덤한 수를 생성하는 함수의 계산복잡도를 $O(1)$이라고 가정할 때, 집합 X의 원소의 갯수가 N개이면 계산복잡도는 $O(N)$이다.

VI. 표준형 행렬로 변환

패리티 검사행렬 H를 표준형 행렬로 변환하는 과정은 다음과 같다.

(단계 1) Fisher-Yates 셔플링 알고리듬을 이용하여 $n \times n$ 치환행렬 P를 랜덤하게 선택한다.

(단계 2) HP를 계산한 후 가우스 소거법을 적용한다. 동시에 $(n-k) \times (n-k)$ 항등행렬 U에 HP의 가우스 소거법 과정에서 발생하는 연산들을 적용한다.

(단계 3) HP가 표준형 행렬로 변환되지 않는다면 다시 (단계 1)로 돌아간다.

(단계 4) 가역행렬 U, 치환행렬 P를 반환한다.

위 연산 과정의 유사부호는 다음과 같다.

입력 : $H \in Mat_{(n-k)\times n}(\mathbb{F}_2)$
출력 : $U \in Mat_{(n-k)\times(n-k)}(\mathbb{F}_2),\ P \in Mat_{n\times n}(\mathbb{F}_2)$

1: ***repeat***
2: $Generate\ random\ permutation\ matrix\ P_{n\times n}$
3: $H', U,\ r \leftarrow H \times P,\ I_{n-k},\ 0$
4: ***while*** $r < n-k$ ***do***
5: $flag = 0$
6: ***for*** $i = r$ to $n-k-1$ ***do***
7: ***if*** $H'[i][r] = 1$ && $flag = 0$ ***then***
8: $Swap\ rth\ and\ jth\ row\ of\ H'$
9: $Swap\ rth\ and\ jth\ row\ of\ U$
10: $flag = 1$
11: ***end if***
12: ***if*** $H'[i][r] = 1$ && $flag = 1$ ***then***
13: $H'[i] \leftarrow H'[i] + H'[r]$
14: $U[i] \leftarrow U[i] + U[r]$
15: ***end if***
16: ***end for***
17: $r \leftarrow r + flag$
18: ***end while***
19: $[V|H''] \leftarrow H'$
20: ***until*** $V = I_{n-k}$
21: ***return*** U, P

다음은 패리티 검사행렬 H를 표준형 행렬로 변환하는 과정의 예시이다.

패리티 검사행렬 H가 다음과 같을 때

$$H = \begin{pmatrix} 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 1 & 0 & 1 & 1 & 1 & 1 \end{pmatrix},$$

가역행렬 U를 다음과 같이 항등행렬로 설정한다.

$$U = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}.$$

치환행렬 P를 랜덤하게 선택한다.

$$P = \begin{pmatrix} 1 & 0 & 1 & 1 & 0 & 0 \\ 1 & 1 & 1 & 0 & 0 & 1 \\ 1 & 1 & 0 & 0 & 1 & 1 \\ 1 & 0 & 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 0 & 1 & 1 \\ 1 & 1 & 1 & 0 & 1 & 0 \end{pmatrix} \rightarrow HP = \begin{pmatrix} 0 & 1 & 1 & 0 & 1 & 1 \\ 1 & 0 & 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 \end{pmatrix}$$

HP에 가우스 소거법을 적용하여 HP_{sys}를 생성하고, 가우스 소거법 과정에서 발생하는 연산들을 동시에 U에 적용한다.

$$HP_{sys} = \begin{pmatrix} 1 & 0 & 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 \end{pmatrix}, \quad U = \begin{pmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 1 & 0 & 1 \end{pmatrix}$$

패리티 검사행렬 H를 UHP로 변환하게 되면 아래의 성질을 이용하여 SDP를 변형해 전수조사에 비해 낮은 복잡도로 해를 찾을 수 있다.

가역행렬 $U \in Mat_{(n-k)\times(n-k)}(\mathbb{F}_2)$, 치환행렬 $P \in Mat_{n\times n}(\mathbb{F}_2)$라고 할 때 다음의 성질이 만족한다.

$$eH^T = s \iff e'H'^T = s'$$
$$where \quad H' = UHP, s' = sU^T, e' = eP$$

위의 성질을 증명하면 다음과 같다.

$$e'H'^T = (eP)(UHP)^T = (eP)P^TH^TU^T = eH^TU^T = sU^T = s'$$

VII. 결론

본 논문에서는 패리티 검사행렬을 표준형 행렬로 변환하는 방법과 이 방법을 이용한 가장 기본적인 알고리듬인 Prange 알고리듬에 대해 알아보았다.

VI장 (단계 3)에서 HP_{sys}가 표준형 행렬이 되지 않을 때 바로 (단계 1)로 돌아가는 방법은 효율적이지 않다고 생각하였다. 따라서 (단계 3)에서 HP_{sys}의 계수가 미리 설정한 파라미터 t보다 클 때 적절하게 열을 바꿔 표준형 행렬로 변환을 하는 방법을 추가로 고안했다. 이 방법을 추가한 것과 추가하지 않는 것에 대한 계산 복잡도 비교 연구를 추후 진행할 예정이다.

Prange 알고리듬 이외에도 가우스 소거법 연산 횟수를 줄이는 방법을 적용한 Lee and Brickell 알고리듬과 이를 일반화한 Stern 알고리듬이 있다. 이 알고리듬들에 대한 계산 복잡도 비교 또한 추후 연구 예정이다.

참고 문헌

[1] Baldi, Marco, Barenghi, Alessandro, Chiaraluce, Franco, Pelosi, Gerardo, Santini, Paolo., "A Finite Regime Analysis of Information Set Decoding Algorithms. Algorithms," 2019

[2] E. Berlekamp, R. McEliece, and H. van Tilborg. "On the inherent intractability of certain coding problems,", 1978

[3] R. A. Fisher, F. Yates, "Statistical tables for biological, agricultural and medical research," 1938

[4] P. J. Lee and E. F. Brickell, "An observation on the security of mceliece's public-key cryptosystem," 1988

[5] E. Prange, "The use of information sets in decoding cyclic codes," 1962

[6] J. Stern, "A method for finding codewords of small weight," 1989

[7] D. Tom, W. Andrew, "Linear Algebra in Twenty Five Lectures," 2012