Journal of KIIT. Vol. 19, No. 11, pp. 63-70, Nov. 30, 2021. pISSN 1598-8619, eISSN 2093-7571
http://dx.doi.org/10.14801/jkiit.2021.19.11.63

M 시퀀스 데시메이션 특성을 이용한 DSSS 신호의 블라인드 동기화

주영진*[1], 김윤지**[1], 김재윤***[1], 강현진***[2], 송정환*[2], 윤동원**[2]

Blind Synchronization of DSSS Signal using M-Sequence Decimation Property

Youngjin Ju*[1], Yoonji Kim**[1], Jaeyoon Kim***[1], Hyunjin Kang***[2], Junghwan Song*[2], and Dongweon Yoon**[2]

본 연구는 ㈜LIG넥스원 저피탐 DSSS 통신 신호 탐지 및 분석 기술 과제 수행 과제로 지원을 받아 수행되었음.

요 약

DSSS(Direct Sequence Spread Spectrum)신호를 복원하기 위해서는 수신 신호를 역확산시켜야 하므로 확산 부호 시작 위치를 정확하게 찾아 동기화해야 한다. 비협력 통신 환경에서는 동기화를 위한 정보가 없기 때문에 수신 신호만으로 동기화를 가능하게 하는 방법이 필요하다. 본 논문에서는 비협력 통신 환경에서 스크램블링된 DSSS 신호의 동기화 알고리즘을 제안한다. 제안한 알고리즘은 먼저 수신 신호로부터 확산 부호의 반복 패턴을 활용하여 선형 피드백 시프트 레지스터 (LFSR, Linear Feedback Shift Register) 출력인 m-시퀀스를 추출하고, 추출된 m-시퀀스의 데시메이션 특성을 이용하여 동기화를 수행한다. 모의실험을 통해 제안된 알고리즘의 동기화 성능을 분석한다.

Abstract

Since despreading must be performed to restore data from Direct Sequence Spread Spectrum(DSSS) signals, it is necessary to find the correct starting position of the spreading code and synchronize the signals. However, in a non-cooperative context, there is no information for synchronization. Therefore, a method that enables synchronization using only the received DSSS signal itself is required. In this paper, we propose an algorithm that synchronizes the scrambled DSSS signal in a non-cooperative context. To synchronize the received signal, we first extract the output of Linear Feedback Shift Register(LFSR), m-sequence, by using a repetitive pattern of the spreading code. Then, we use the decimation property of the extracted m-sequence. Through computer simulations, we analyze the synchronization performance of the proposed algorithm in a noisy environment.

Keywords
communication forensic, linear feedback shift register, estimation

* 한양대학교 수학과, 자연과학연구소(*[1]공동1저자)
- ORCID[1]: https://orcid.org/0000-0003-0676-1700
- ORCID[2]: https://orcid.org/0000-0002-8802-8141
** 한양대학교 융합전자공학과(**[1]공동1저자, **[2]교신저자)
- ORCID[1]: https://orcid.org/0000-0003-4247-5795
- ORCID[2]: https://orcid.org/0000-0001-9631-3500
*** LIG 넥스원 전자전연구소 수석연구원
- ORCID[1]: https://orcid.org/0000-0002-4692-8467
- ORCID[2]: https://orcid.org/0000-0003-1770-2634

· Received: Aug. 13, 2021, Revised: Sep. 15, 2021, Accepted: Sep. 18, 2021
· Corresponding Author: Dongweon Yoon
Dept. of Electronic Engineering, Hanyang University, 222 Wangsimni-ro, Seongdong-gu, Seoul 04763, Korea,
Tel.: +82-2220-0362, Email: dwyoon@hanyang.ac.kr

Ⅰ. 서 론

직접 시퀀스 대역 확산(DSSS, Direct Sequence Spread Spectrum)시스템은 잡음과 재밍에 강하며 저피탐 특성을 갖고 있어 군 통신에서 주로 사용되어 왔으며, 1990년 중반이후 CDMA(Code Division Multiple Access), WCDMA(Wideband CDMA), WLAN(Wireless Local Area Network) 등의 상용 통신 분야에서도 널리 사용되고 있다[1]-[4].

DSSS에서는 일반적으로 신호의 스펙트럼을 넓게 분포시키기 위한 확산 부호기와 함께 입력 데이터의 반복을 피하기 위해서 스크램블러를 사용한다. 대표적인 스크램블러로는 동기 스크램블러(Synchronous scrambler)와 자기 동기 스크램블러(Self-synchronous scrambler)가 있다. 동기 스크램블러는 선형 피드백 시프트 레지스터(LFSR, Linear Feedback Shift Register)로부터 생성된 m-시퀀스를 사용하며, 자기 동기 스크램블러는 시프트 레지스터의 출력과 메시지 비트가 더해진 값이 시프트 레지스터의 입력으로 사용되는 구조를 갖고 있다.

불법 전파 감시 또는 인지무선통신과 같은 비협력 통신 환경에서 신호를 복원하기 위해서는 수신 신호만을 이용하여 송신기에서 사용된 통신 파라미터를 추정해야 한다. 비협력 통신 환경에서 수신한 신호의 파라미터를 추정하는 방법에 대한 연구가 다양하게 진행되어 왔다[5]-[11]. [5]와 [6]에서는 채널 부호의 특성을 이용하여 인터리빙에 대한 정보를 추정하였다. 확산부호가 반복되는 성질을 통해 [7]에서는 동기 스크램블러를 추정하였고, [8]에서는 자기동기 스크램블러를 추정하였다. [9]에서는 선형 변환을 이용하여 동기 스크램블러를 추정하였으며, [10]과 [11]에서는 [9]에서 제안된 방법을 개선한 추정 알고리즘을 제안하였다.

기존 연구들에서는 스크램블링된 DSSS 신호의 스크램블러를 블라인드 추정하기 위하여 수신 신호의 시작이 확산 부호 시작 위치에 정확하게 동기화되었다고 가정하고 스크램블러를 추정한다. 일반적인 협력 통신에서는 수신기에서 송신기의 확산부호 및 스크램블러 정보를 알고 있으므로 신호 획득 및 추적 과정을 순차적으로 거쳐 신호의 동기화가 가능하지만, 비협력 통신에서는 미상의 신호를 탐지하고 추정하기 때문에 올바른 신호 동기화가 매우 어렵다. 또한 동기 스크램블러를 이용하여 스크램블링된 경우 가능한 LFSR의 피드백 다항식과 초기값에 대한 경우의 수가 무수히 많이 존재하므로 수신 신호만을 이용해서 스크램블러를 추정하는 것은 더욱 어렵다.

본 논문에서는 비협력 통신 환경에서 동기 스크램블링된 DSSS 신호를 수신하였을 때 확산부호의 시작점을 찾는 동기화 알고리즘을 제안한다. 제안하는 알고리즘은 LFSR 출력 스트림인 m-시퀀스를 데시메이션(Decimation)해서 생성한 시퀀스의 특성을 이용하여 동기화를 수행한다. 신호의 동기화 후에는 확산 부호의 반복 패턴을 활용하여 m-시퀀스를 추출하고 이를 통해 스크램블러를 추정한다[7].

본 논문의 구성은 다음과 같다. Ⅱ장에서는 m-시퀀스의 데시메이션 특성을 사용하여 수신 신호를 동기화하는 새로운 알고리즘을 제시한다. Ⅲ장에서는 제안한 알고리즘을 사용한 스크램블링된 DSSS 신호의 동기화 성능을 분석한다. 마지막으로 Ⅳ장에서는 본 논문의 결론을 맺는다.

Ⅱ. DSSS 신호 블라인드 동기화

잡음 환경에서 수신한 DSSS 신호로부터 데이터를 복원하기 위해서는 확산 부호 시작 위치를 정확하게 찾는 수신 신호의 동기화 과정을 필요로 한다. 신호 동기화에 대한 정보가 없는 비협력 통신 상황에서는 수신 신호만으로 동기를 맞추어야 한다. 가장 쉽게 생각할 수 있는 방법으로는 수신된 신호를 1 bit씩 shift해가면서 동기를 맞추는 방법을 생각할 수 있다. 하지만 이 경우 동기가 맞았는지 판단이 불가능하고, 무엇보다도 확산부호의 길이가 긴 경우에는 너무 많은 반복이 필요하다는 문제가 있다.

이 장에서는 m-시퀀스의 데시메이션 특성에 대해 살펴본 후, 이를 이용하여 블라인드 환경에서 수신 신호를 동기화 하는 새로운 방법에 대해 제안한다. 알고리즘 설명에 앞서 본 논문 전반에 걸쳐 사용하는 기호는 표 1에 표기하였다.

그림 1과 같이 확산부호로 확산 후 동기 스크램블링된 신호를 수신하였다고 가정하자.

표 1. 기호 정의
Table 1. Symbol notations

Symbol	Definition
$\oplus$	modulo-2 addition
$\ll(\gg)$	(right) shift
$\gcd(a,b)$	great commom divisior of a and b
$a \perp b$	a and b are relatively prime

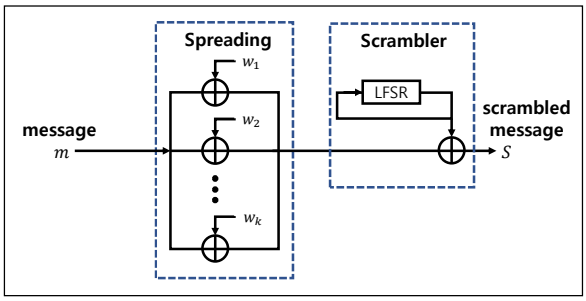


그림 1. 시스템 모델
Fig. 1. System model

메시지 m에 대해서 k비트 길이 확산부호 $w=(w_0w_1...w_{k-1})$를 이용하여 확산된 신호 T는 식 (1)과 같다.

$$T_{kt+i}=m_t\oplus w_i(i=0,...,k-1) \tag{1}$$

또한 확산된 신호 T에 동기 스크램블러로 생성한 m-시퀀스 p를 더하여 스크램블링된 신호 S는 식 (2)와 같다.

$$S_j=T_j\oplus p_j=m_t\oplus w_i\oplus p_j(j=kt+i) \tag{2}$$

본 논문에서 제안하는 동기화 알고리즘은 확산부호의 길이 k가 특정되었다고 가정하고 수신 신호로부터 m-시퀀스를 추출한 후, 추출된 m-시퀀스의 데시메이션 특성을 통해 동기화를 수행한다.

2.1 수신 신호로부터 m-시퀀스 추출

스크램블링에 사용된 LFSR의 피드백 다항식을 추정하기 위해서는 메시지, 확산부호, m-시퀀스가 섞여 있는 수신 신호로부터 m-시퀀스만을 추출해야 한다. m-시퀀스를 추출하기 위해서는 확산부호가 주기적으로 반복되는 성질을 이용하여 수신 신호를 shift한 후 더하는 방법을 사용한다[7]. [7]의 방법을 간단히 요약하면 다음과 같다.

우선 수신 신호의 동기가 맞았다고 가정하고 식 (3)과 같은 연산을 통해 생성되는 U를 생각하자.

$$U=S\oplus(S\ll 1)\oplus(S\ll k)\oplus(S\ll k+1) \tag{3}$$

식 (3)에서 U의 $j(=kt+i)$번째 비트 값은 S_j, S_{j+1}, S_{j+k}, S_{j+k+1}의 합이므로 $i\neq 0$인 경우에는 식 (2)에 따라 식 (4)와 같이 m-시퀀스의 합으로만 나타낼 수 있다.

$$U_j=p_j\oplus p_{j+1}\oplus p_{j+k}\oplus p_{j+k+1} \tag{4}$$

반면, $i=0$인 경우에는 식 (5)와 같이 LFSR 출력 시퀀스 이외에 메시지 비트의 영향이 있는 식으로 구성된다.

$$U_j=m_t\oplus m_{t+2}\oplus p_j\oplus p_{j+1}\oplus p_{j+k}\oplus p_{j+k+1} \tag{5}$$

여기서 $\hat{p}_j=p_j\oplus p_{j+1}\oplus p_{j+k}\oplus p_{j+k+1}$이라고 정의하면 m-시퀀스의 cycle-and-add property[12]에 의해서 $\hat{p}_j$에 대하여 $\hat{p}_j=p_{j+\alpha}$를 만족하는 α값이 존재한다. 즉, $\hat{p}$와 p는 동일한 피드백 다항식을 갖는 m-시퀀스이다. 따라서 U로부터 k비트마다 반복되는 1비트를 제외한 나머지 위치에서는 메시지와 확산부호가 제거된 순수한 LFSR 출력인 m-시퀀스 값을 추출 할 수 있다[7]. 또한, 순수한 m-시퀀스가 아닌 비트의 다음 비트를 기준으로 신호를 동기화하면 확산부호의 시작점과 신호를 일치시킬 수 있다.

2.2 M-시퀀스의 데시메이션 특성 및 추정방법

피드백 다항식의 차수가 n인 LFSR의 출력 시퀀스, m-시퀀스는 2^n-1길이의 주기를 갖게 된다. m-시퀀스에 대해서는 다양한 특성들이 연구되었는데 [12][13], 본 절에서는 데시메이션에 관련한 특성을 소개한다.

d비트 데시메이션이란 시퀀스 $X=x_1x_2x_3...$이 주어져 있을 때 d비트마다 1비트를 추출하여 시퀀스

$X' = x_1 x_{d+1} x_{2d+1} \cdots$ 을 생성하는 것을 말한다. 여기서 X'을 X의 d비트 데시메이션 시퀀스라고 한다. 피드백 다항식이 n차인 LFSR의 출력 시퀀스 X에 대한 d비트 데시메이션 시퀀스 X'에 대해 다음의 Theorem 1 이 성립한다[13].

Theorem 1. n차 피드백 다항식 $p_1(x)$를 사용하여 생성한 m-시퀀스에 대해서, d비트 데시메이션한 시퀀스를 생성할 수 있는 n차 피드백 다항식 $p_2(x)$가 항상 존재하며, $p_2(x)$는 $p_1(x)$와 d에 대해 유일하다. 이때, d비트 데시메이션한 시퀀스의 주기는 $\frac{2^n-1}{\gcd(d, 2^n-1)}$ 이다.

예를 들어 n차 피드백 다항식으로부터 생성된 m-시퀀스 $X = x_1 x_2 x_3 \cdots$이 주어졌다고 하자. 이때, X의 d비트 데시메이션 시퀀스 $X^{(a)} = x_a x_{d+a} x_{2d+a} \cdots$는 Theorem 1에 의해 어떤 a 에서든지 $X^{(a)}$를 생성하는 LFSR의 피드백 다항식이 항상 존재하고, 이 다항식은 a에 관계없이 동일하다는 것을 알 수 있다. 또한 이 다항식은 Theorem 1에 의해 $d \perp 2^n - 1$을 만족한다면 원시 다항식이다.

LFSR 출력 시퀀스가 주어져 있을 때, 피드백 다항식을 찾는 방법에 대해서 많이 연구되어 왔으며[14][15] 가장 대표적인 알고리즘으로는 BM(Berlekamp-Massey) 알고리즘이 있다[16]. BM 알고리즘은 피드백 다항식의 차수 n에 대해 최소 $2n$길이의 비트열을 입력으로 받아서 다항식을 찾는 알고리즘이다. 따라서 $2n$길이만큼의 LFSR 출력 시퀀스를 얻을 수 있는 경우 BM 알고리즘을 사용하면 간단하게 피드백 다항식을 찾을 수 있다.

2.3 m-시퀀스의 데시메이션을 이용한 신호 동기화 방법

2.1절에서 제시한 방법을 통해서 스크램블링된 DSSS 신호로부터 순수한 LFSR 출력 시퀀스인 m-시퀀스를 얻을 수 있다. 이번 절에서는 m-시퀀스를 얻었을 때 2.2절에서 설명한 데시메이션 특성을 활용하여 수신 신호를 동기화하는 방법을 제안한다.

식 (3)을 이용하여 U를 생성하면 식 (4), (5)와 같이 U는 각 k비트마다 1비트는 메시지의 영향을 받은 값이고 나머지 $k-1$비트들은 순수한 m-시퀀스이다. 만약 신호의 동기가 맞는 경우에는 S를 통해 U를 생성하면 $U_{j(=kt+i)}$에 대해서 식 (5)와 같이 $i=0$일 때의 비트가 메시지의 영향을 받은 비트다. 반면, 신호의 동기가 맞지 않는 경우에는 어떤 r값에 대해 $i=r$인 $U_{j(=kt+i)}$의 비트가 메시지의 영향을 받은 비트가 된다. 따라서 r을 아는 경우에는 수신 신호 S를 r만큼 left shift하게 되면 동기를 맞출 수 있다.

우선 신호에 잡음이 없는 이상적인 상황에서 신호 동기화 방법은 다음과 같다. 식 (3)을 이용하여 수신 신호 S로부터 U를 생성한 후 k비트 데시메이션을 수행하여 식 (6)을 생성할 수 있다.

$$U^{(a)} = U_a U_{k+a} U_{2k+a} \cdots \tag{6}$$

여기서 $a = 1,...,k$ 이다. $a \neq r$인 $U^{(a)}$는 식 (4)와 (5)에 따라 m-시퀀스의 k비트 데시메이션 시퀀스가 된다. 반면에 $a=r$일 때의 $U^{(\alpha)}$는 랜덤한 시퀀스의 k비트 데시메이션 시퀀스와 같게 된다. Theorem 1에 따르면 m-시퀀스의 k비트 데시메이션 시퀀스를 생성하는 LFSR의 피드백 다항식은 유일하다. 즉, $a \neq r$인 경우에는 a값에 관계없이 $U^{(a)}$를 생성하는 LFSR의 피드백 다항식이 존재하며 이 다항식은 모두 동일하다. 반면, $a=r$인 경우에는 이를 만족하지 않는다. 따라서 각 $a=1,...,k$에 대해 $U^{(a)}$을 BM 알고리즘에 적용하면 $a=r$인 경우를 제외하고는 모두 동일한 다항식을 얻을 수 있게 된다. 즉, 다른 다항식이 출력으로 나오는 값을 r라고 추정할 수 있으며, 최종적으로 수신 신호 S를 r만큼 left shift 해서 동기를 맞출 수 있다.

BM 알고리즘으로 피드백 다항식을 얻기 위해서는 n차 피드백 다항식을 통해서 만들어진 LFSR 출력 시퀀스인 경우, 각 a에 대해서 최소 $2n$비트의 입력이 필요하다. 즉, $2nk$길이의 U값이 필요하며 이를 생성하기 위해서는 $(2n+1)k+1$길이의 S값이 필요하다. 따라서 수신 신호에 잡음이 없는 상황에서 제시한 방법으로 동기화를 진행하기 위해서는

최소 $(2n+1)k+1$길이의 S가 필요하다.

한편, 수신 신호에 잡음이 있는 경우에는 $a \neq r$이더라도 $U^{(a)}$값이 잡음의 영향으로 순수한 LFSR 출력 시퀀스 값으로만 이루어지지 않는 경우도 존재한다. BM 알고리즘의 입력 시퀀스에 오류가 있는 경우 틀린 다항식을 얻게 되는 경우가 존재하므로 위의 방법을 그대로 적용하면 잡음 상황에서는 동기화를 올바르게 하지 못하는 경우가 발생한다. 따라서 이 경우 데시메이션 시퀀스에 대한 피드백 다항식을 찾는 과정을 반복 적용한다. 여러 번의 반복 과정을 거치면 $U^{(a)}$에 대한 생성 다항식 $p_2(x)$를 가장 많이 찾게 되고 이를 올바른 데시메이션 시퀀스에 대한 피드백 다항식이라고 추정하여 잡음이 없는 경우와 유사한 방법으로 동기화를 진행한다. 잡음환경에서 DSSS 신호 블라인드 동기화 과정은 알고리즘 1과 같이 구성된다.

먼저 데시메이션 시퀀스 $U^{(a)}$에 대한 올바른 피드백 다항식, $poly$를 찾는다. 이 때 $poly$는 $U^{(a)}$에 대한 피드백 다항식 추정을 반복적으로 수행하여 가장 많이 추정된 다항식으로 선정한다.

각 데시메이션 시퀀스 별로 $poly$를 찾은 후에는 $U^{(a)}$의 피드백 다항식을 구하여 이를 $poly$와 비교한다. 만약 $U^{(a)}$가 메시지의 영향을 받은 위치에서 데시메이션한 스트림이라면 아무리 반복을 많이 하더라도 BM 알고리즘의 결과로 데시메이션 시퀀스에 대한 $poly$가 나타나지 않는다. 반면 $U^{(a)}$가 메시지의 영향을 받지 않은 위치에서 데시메이션한 스트림이라면 피드백 다항식 추정을 반복하였을 때 적어도 1번은 데시메이션 시퀀스에 대한 $poly$를 추정할 가능성이 크다. 따라서 $k-1$개의 위치에서 데시메이션 시퀀스에 대한 올바른 다항식을 찾을 때까지 피드백 다항식 추정을 반복하여 메시지의 영향을 받은 위치를 찾는다. 그 후 메시지의 영향을 받은 위치만큼 수신 신호를 shift하여 동기화를 수행한다.

Algorithm 1의 과정을 위해 필요한 수신 신호의 최소 길이에 대해서 생각해보자. M이 N보다 큰 경우에는 4번 과정에서 필요한 S의 길이보다 8번 과정에서 필요한 S의 길이가 더 길기 때문에 8번 과정에서 필요한 S의 길이만 생각하면 된다.

Algorithm 1. DSSS 신호 블라인드 동기화

입력 : 수신 신호 S, 확산부호의 길이 k
출력 : 동기화된 신호
1. 반복 파라미터 N 설정
2. $N \times k$크기 행렬 $CandidatePoly$를 0으로 초기화
3. $U = S \oplus (S \ll 1) \oplus (S \ll k) \oplus (S \ll k+1)$ 생성
4. $i = 1$부터 N까지 다음의 과정 반복 수행
   4.1. $a = 1,...,k$에 대해 $U^{(a)}$생성
   4.2. BM 알고리즘에 $U^{(a)}$를 입력하여 $U^{(a)}$에 대한 피드백 다항식을 구함
   4.3. 추정한 다항식을 $CandidatePoly$의 (i,a)번째 위치에 저장
   4.4. U를 k비트 left shift하여 위 과정 반복
5. $CandidatePoly$ 중 가장 많이 선택된 다항식 $poly$선택
6. $U = S \oplus (S \ll 1) \oplus (S \ll k) \oplus (S \ll k+1)$ 다시 생성
7. k크기 배열 idx를 0으로 초기화
8. 다음의 과정 M회 반복 수행
   8.1. $a = 1,...,k$에 대해 $U^{(a)}$생성
   8.2. BM 알고리즘에 $U^{(a)}$를 입력하여 $U^{(a)}$에 대한 피드백 다항식을 구함
   8.3. 각 a에 대해 구한 다항식이 $poly$와 같은 경우 idx의 a번째 원소를 1로 변경
   8.4. idx의 원소 중 1의 개수가 $k-1$개인 경우 반복 종료
   8.5. U를 k비트 left shift하여 위 과정 반복
9. 배열 idx의 값이 0인 원소의 index를 $shift$값으로 저장
10. $S = S \ll shift$

M회 반복을 하는 경우 U가 처음 생성되었을 때보다 kM비트 left shift된 상황까지 반복을 진행해야한다. 가장 마지막 반복에서는 $2nk$길이의 U가 필요하므로 총 $k(M+2n)$길이의 U가 필요하다. 이를 생성하기 위해서는 $k(M+2n+1)+1$ 길이의 수신 신호 S가 필요하다. 따라서 Algorithm 1을 통해서 동기화를 하기 위해서는 $k(M+2n+1)+1$길이의 시퀀스가 필요하다.

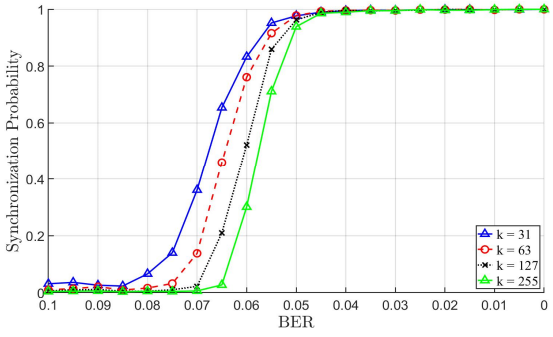
III. 모의 실험 및 성능 분석

이 장에서는 비협력 통신 환경에서 스크램블링된 DSSS 신호를 수신하였을 때 제안하는 동기화 알고리즘의 성능을 확산부호의 길이와 LFSR 피드백 다항식의 차수에 따라 분석한다. 또한 제안하는 알고리즘과 [7]의 스크램블러 추정 기법을 통해 동기가 맞지 않은 수신 신호로부터 LFSR 피드백 다항식을 추정하고 성능을 분석한다. 각 실험에 대해서는 1000번의 시도를 통해서 동기화 및 추정 성공률을 계산하였다.

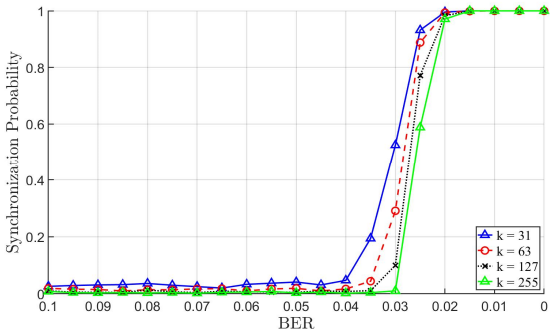
확산부호의 길이가 $k=31,63,127,255$일 때, LFSR 피드백 다항식 차수 n이 10인 경우와 20인 경우에 대해 각각 실험하고, 각 길이에 따른 동기화 확률을 그림 2에 나타내었다. 실험 결과, n이 10이고 k값이 31, 63, 127, 255인 경우 BER이 각각 0.06, 0.055, 0.055, 0.05 일 때, 90%의 추정 성능을 보였다. 그리고 n이 20인 경우에는 k값이 31, 63, 127, 255 인 경우 BER이 각각 0.025, 0.025, 0.02, 0.02 일 때, 추정 성능이 90%의 추정 성능을 보였다.

이를 통해 LFSR 피드백 다항식의 차수가 같은 경우 확산부호의 길이가 짧을수록 동기화 성능이 더 좋아지는 것을 확인할 수 있다. 이는 확산부호의 길이 k가 클수록 Algorithm 1의 8번 과정 내에서 더 많은 위치에 대해서 올바른 생성 다항식을 찾아야하기 때문으로 분석된다.

그림 3에 LFSR 피드백 다항식의 차수 n에 따른 동기화 확률을 나타내었다. 그림 3에서 확인할 수 있듯이 k가 31이고 n값이 10, 15, 20, 25 인 경우 BER이 각각 0.06, 0.035, 0.025, 0.02일 때, 90%의 추정 성능을 보였다. 그리고 k가 127이고 n값이 10, 15, 20, 25 인 경우 BER이 각각 0.055, 0.035, 0.02, 0.015 일 때, 90%의 추정 성능을 보였다. 그림 3의 결과를 통해 LFSR 피드백 다항식의 차수가 작을수록 동기화 성능이 좋은 것을 확인할 수 있다. 이는 LFSR 피드백 다항식을 찾기 위해 $2n$비트를 입력하는데, 입력 길이가 작을수록 잡음이 존재할 확률이 더 작기 때문으로 분석된다.



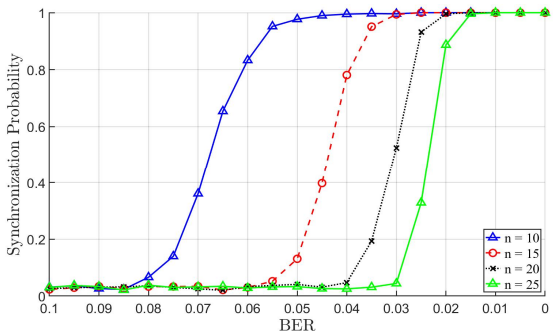
(a) n=10



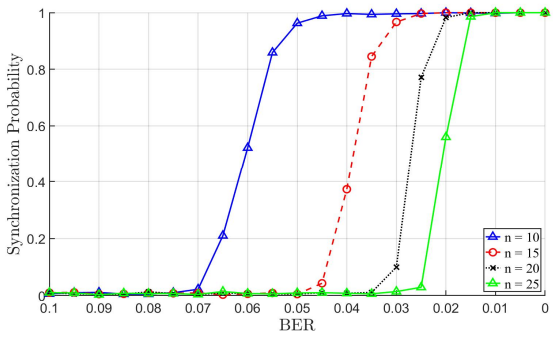
(b) n=20

그림 2. 확산부호의 길이에 따른 동기화 확률

Fig. 2. Synchronization probabilities for spreading code length



(a) k=31



(b) k=127

그림 3. LFSR 피드백 다항식의 차수에 따른 동기화 확률

Fig. 3. Synchronization probability for the degree of feedback polynomial of LFSR

Ⅳ. 결 론

본 논문에서는 비협력 통신 상황의 블라인드 환경에서 동기 스크램블러로 스크램블링된 DSSS 신호의 동기화 알고리즘을 제안하였다. 제안하는 알고리즘은 LFSR 출력 시퀀스에 대해서 d비트 데시메이션 시퀀스를 생성하는 생성 다항식이 유일하게 존재한다는 특성을 이용하여 DSSS 신호 동기화를 수행하였다. 모의실험 결과 제안하는 동기화 알고리즘은 확산부호의 길이가 짧을수록 그리고 스크램블러의 단수가 작을수록 동기화 성능이 더 좋은 것으로 분석되었다. 본 논문에서 제안하는 동기화 알고리즘을 이용하면 블라인드 상황에서도 동기화를 수행할 수 있게 되어, 동기화에 대한 제약 없이 기존의 스크램블러 추정 알고리즘을 이용하여 스크램블러의 제원 추정이 가능하다. 향후 동기 스크램블러로 스크램블링된 신호 뿐만 아니라 자기동기 스크램블러로 스크램블링된 신호를 블라인드 환경에서 추정하는 연구가 필요할 것으로 사료된다.

References

[1] J. G. Proakis and M. Salehi, "Digital communications", 5th edition, New York: McGraw-hill., pp. 1028-1053, Nov. 2007.

[2] K. Feher, "Wireless digital communications: modulation and spread spectrum applications", Prentice-Hall, Inc., pp. 241-332, May 1995.

[3] H. Holma and A. Toskala, "WCDMA for UMTS: Radio access for third generation mobile communications", John Wiley & Sons, pp. 31-58, Mar. 2001.

[4] P. Roshan and J. Leary, "802.11 Wireless LAN fundamentals", Cisco press, pp. 21-147, Dec. 2003.

[5] C. Choi and D. Yoon, "Blind Interleaver Parameter Estimation Using Kullback-Leibler Divergence", Journal of KIIT, Vol. 15, No. 12, pp. 109-115, Dec. 2017. http://dx.doi.org/10.14801/jkiit.2017.15.12.109.

[6] G. Kim, Y. Jang, and D. Yoon, "Blind Estimation for Interleaving Parameter using Probability Mass Function over Fading Channel", Journal of KIIT, Vol. 17, No. 5, pp. 39-46, May 2019. http://dx.doi.org/10.14801/jkiit.2019.17.5.39.

[7] D. Kim, J. Song, and D. Yoon, "On the estimation of synchronous scrambler in direct sequence spread spectrum systems", IEEE Access, Vol. 8, pp. 166450-466459, 2020. http://dx.doi.org/10.1109/ACCESS.2020.3023425.

[8] D. Kim and D. Yoon, "Blind estimation of self-synchronous scrambler in DSSS systems", IEEE Access, Vol. 9, pp. 76976-76982, May 2021. https://doi.org/10.1109/ACCESS.2021.3083071.

[9] M. Cluzeau, "Reconstruction of a linear scrambler", IEEE Trans. Computers, Vol. 56, No. 9, pp. 1283-1291, Sep. 2007. https://doi.org/10.1109/TC.2007.1055.

[10] X. B. Liu, S. N. Koh, X. W. Wu, and C. C. Chui, "Reconstructing a linear scrambler with improved detection capability and in the presence of noise", IEEE Trans. Inf. Forensics and Security, Vol. 7, No. 1, pp. 208-218, Feb. 2012. https://doi.org/10.1109/TIFS.2011.2169790.

[11] H. WenJia, "Reconstructing the feedback polynomial of a linear scrambler with the method of hypothesis testing", IET Commun., Vol. 9, No. 8, pp. 1044-1047, May 2015. https://doi.org/10.1049/iet-com.2014.0109.

[12] H. Y. Song, "Feedback shift register sequences, Wiley Encyclopedia of Telecommunications", Apr. 2003. https://doi.org/10.1002/0471219282.eot328.

[13] C. Lauradoux and A. Röck, "Parallel generation of l-sequences", in Proc. Springer SETA, Berlin, Heidelberg, pp. 299-312, Sep. 2008.

[14] D. Kim, Y. Kim, C. Park, J. Song, and D. Yoon, "Linear feedback shift register estimation using statistical method in a noisy environment", J. Korean Inst. Commun. Inf. Sci., Vol. 45, No. 3, pp. 616-621, Mar, 2020. http://dx.doi.org/10.7840/kics.2020.45.3.616.

[15] M. Jang, S. Ahn, U. Jung, and D. Yoon, "Estimation of linear feedback shift register in noisy environments", in Proc. KIEES Winter Conf., Jungsun, Korea, pp. 39, Feb. 2019.

[16] J. L. Massey, "Shift-register synthesis and BCH decoding", IEEE Trans. Inf. Theory, Vol. 15, No. 1, pp. 122-127, Jan. 1969. https://doi.org/10.1109/TIT.1969.1054260.

저자소개

주 영 진 (Youngjin Ju)

2018년 8월 : 한양대학교 수학과(이학사)
2018년 9월 ~ 현재 : 한양대학교 수학과 석박사과정 재학
관심분야 : 암호학, 암호 프로토콜, 신호추정

김 윤 지 (Yoonji Kim)

2019년 2월 : 한양대학교 융합전자공학부(학사)
2020년 3월 ~ 현재 : 한양대학교 융합전자공학과 석박사과정 재학
관심분야 : 무선 통신, 파라미터 추정, 신호추정

김 재 윤 (Jaeyoon Kim)

2002년 2월 : 한양대학교 전자공학과(학사)
2004년 2월 : 한양대학교 전자전기제어 계측공학과(석사)
2004년 1월 ~ 현재 : LIG넥스원 전자전연구소 수석연구원
관심분야 : 디지털신호처리, 신호분석 알고리즘

강 현 진 (Hyunjin Kang)

2002년 2월 : 중앙대학교 전자공학과(학사)
2008년 2월 : 중앙대학교 전자공학과(석사)
2011년 12월 ~ 현재 : LIG넥스원 전자전연구소 수석연구원
관심분야 : 전자전신호처리, 디지털통신

송 정 환 (Junghwan Song)

1984년 2월 : 한양대학교 수학과(이학사)
1989년 5월 : Syracuse University 수학과(이학석사)
1993년 5월 : Rensselaer Polytechnic Institute 수학과(이학박사)
1999년 3월 ~ 현재 : 한양대학교 수학과 교수
관심분야 : 암호학, 암호 프로토콜, 신호추정

윤 동 원 (Dongweon Yoon)

1989년 2월 : 한양대학교 전자통신공학과(공학사)
1992년 2월 : 한양대학교 전자통신공학과(공학석사)
1995년 8월 : 한양대학교 전자통신전파공학과(공학박사)
2021년 11월 현재 : 한양대학교 융합전자공학부 교수
관심분야 : 무선통신, 위성 및 우주통신, 신호추정