무선암호시스템에서 FDI와 GDI를 이용한 전송성능 분석

정회원 홍 진 근*, 황 찬 식**, 윤 장 홍*, 강 건 우*

Analysis of Transmission Performance Using Frame Domain Interleaving and Group Domain Interleaving on a Radio Encryption System

Jin-keun Hong*, Chan-sik Hwang**, Jang-hong Yoon*, Ken-woo Kang* ***Regular Members***

요 약

본 논문은 무선망의 링크암호에 적합한 스트림 암호시스템을 설계하였고 설계된 스트림 암호를 통해 암호문을 전송할 때 한 주기 동안 발생하는 동기패턴, 세션키, 암호문 정보에 적합한 인터리빙 기법을 설계하여 전송함으로써 버스트 오류로부터 암호문을 보호하고 전송성능을 개선하여 robust한 암호통신을 가능하도록 하였다. 본 논문에서는 한 주기의 동기패턴, 세션키, 암호문으로 구성된 FDI(Frame Domain Interleaving) 기법과 다수개의 프레임으로 구성된 GDI(Group Domain Interleaving) 기법을 제안한 후 이에 관한 성능을 분석하고 적합한 인터리빙 기법을 도출하였다.

ABSTRACT

In this paper, a synchronized stream cryptosystem for secure link layer communication on a radio channel is designed. We have proposed new interleaving schemes to randomize a burst error and experimented with different types of interleaving schemes. The proposed techniques of interleaving schemes are : (1) interleaving scheme based on frame, (2) interleaving scheme based on group. The proposed schemes are very robust in randomizing

Ⅰ. 서 론

정보보호 기술은 정보보호 기반기술, 네트워크 및 시스템 보호기술, 응용서비스 보호기술, 보안관리기술 등으로 나누고 이 가운데 정보보호 기반기술의 일종인 암호기반기술에는 대칭 암호시스템과 비대칭 암호시스템이 있고 암호처리 형태에 따라 블록 암호시스템과 스트림 암호시스템으로 나눈다.

스트림 암호시스템은 키수열 발생기를 통해 발생된 난수를 이용하여 암복호화를 수행함으로써 오류확산이 없고 주기, 선형복잡도, 상관면역도 등과 같은 비도 수준에 대한 정량화가 가능하고 하드웨어 구현이 용이하며 통신 지연이 없고, 복사전력에 제한을 받고 많은 잡음 채널로 전송채널이 심한 영향을 받는 이동통신망이나 위성통신망과 같은 무선전송 채널에서는 링크계층의 암호화 방식으로 오류확산이 없다는 장점을 가진다. 이와같은 암호시스템의 특성으로 인해 스트림 암호시스템은 채널구간 암호통신 방식에 많이 사용되고 있다.

스트림 암호시스템을 무선링크계층에 적용할 경우 멀티패스 페이딩, 간섭, 상대방의 고의적인 재밍 등의 채널 환경으로 인해 다량의 버스트 오류가 발생하고 특히 레일레이 페이딩과 같이 반사파로만 구성된 전파환경의 경우 페이딩 영향으로 인해 수

* 국가보안기술연구소, ** 경북대학교 전자전기공학부 데이터통신시스템 연구실
논문번호: 99453-1115, 접수일자: 1999년 11월 15일

십~수백 비트에 걸쳐서 발생한다. 이러한 채널환경에서 버스트 오류에 대한 대책이 없을 경우 무선 암호통신 시스템 성능은 심각한 영향을 받게 된다. Radio 채널의 전송계층에서 스트림 암호시스템을 사용하여 암호정보를 전송할 때 무선구간의 채널환경으로 인해 다량의 비트가 연속적인 형태로 발생하는 버스트 오류를 발생함으로써 주기적으로 전송하는 암호문의 경우 한 주기 전체의 암호문 프레임이 손실되고 암호통신은 통신불능 상태가 된다. 만일 버스트 오류가 발생하는 채널환경에서 암호문이 랜덤 오류에 대한 전송보호기법으로 처리된다면 이는 전송성능에 심각한 성능저하를 초래한다. 또한 버스트 오류특성을 고려한 전송기법의 요구는 버스트 오류가 몇 개의 블록으로 확산되어 나타나는 것보다 한 블록에 집중되어 나타나므로 버스트 오류에 대비한 정보의 확산과정을 통해 재전송시스템에서 재전송 횟수를 감소할 수 있다. 무선링크에 적용되는 동기식 스트림 암호시스템은 버스트 오류특성에 대비한 적합한 오류대책이 요구되고 이를 통해 전송성능이 개선된 암호통신시스템이 요구된다.

따라서 본 논문은 무선망의 링크암호에 적합한 스트림 암호시스템을 설계하였고 설계된 스트림 암호를 통해 암호문을 전송할 때 한 주기 동안 발생하는 동기패턴, 세션키, 암호문 정보를 인터리빙 기법을 적용하여 전송함으로써 버스트 오류로부터 암호문을 보호하고 전송성능을 개선하여 robust한 암호통신을 가능하도록 하였다. 본 논문에서는 한 주기의 동기패턴, 세션키, 암호문으로 구성된 프레임 영역 인터리빙(Frame Domain Interleaving) 기법과 다수개의 프레임으로 구성된 그룹영역 인터리빙(Group Domain Interleaving) 기법을 제안하여 전송성능을 개선하고자 한다. 동기식 스트림 암호시스템의 동기구조는 2400, 4800, 9600, 19200 비트를 갖도록 설계하였고 이 가운데 9600비트 동기구조에서 주로 실험하였다. 적용된 인터리빙 방식을 암호정보에 블록, 헬리컬, 랜덤, 확장된 랜덤 인터리빙을 수행하였고 850MHz와 220MHz 대역에서 수신전력 변동에 따른 평균 버스트 오류양을 검출하여 인터리빙 유무 영향, 인터리빙 depth에 따른 전송성능, 인터리빙 유형에 따른 성능평가 수행하여 적합한 인터리빙기법을 유도하기 위한 전송성능을 분석하였다.

Ⅱ. 무선채널의 특성

무선채널은 유선채널과 달리 자유공간 손실, 반사, 회절 등의 영향으로 인해 경로손실이 훨씬 높고 열악한 채널환경을 갖는다. 따라서 이러한 경로손실의 문제를 해결하기 위해서는 송신출력이나 안테나 이득증가가 요구되나 송신출력이나 안테나 이득의 증가는 운용상 제한되는 경우가 많다. 그러므로 대부분의 무선채널에서는 전달거리가 제한적이고 전달거리내 신호대 잡음비가 유선에 비해 상당히 낮아서 전송채널에서 다량의 정보손실이 발생한다. 특히 무선채널에서 버스트 오류 발생 원인인 페이딩 현상[4][7]은 짧은 시간간격동안 신호세기의 급격한 변화, 다른 멀티패스 신호에서 변화하는 도플러 변이에 의한 랜덤 주파수 변조, 멀티패스 전송지연에 의한 시간 분산 등과 같은 요인으로 인해 발생한다. 페이딩은 이동하거나 주위의 건물 등의 영향으로 전파의 반사로 수신전력이 변동하게 되고 만일 수신전력이 한계 값 이하로 떨어지게 되면 비트오류가 발생한다. 따라서 페이딩 현상으로 인한 버스트 오류로부터 전송성능을 개선하기 위해서는 버스트 오류[19][20]가 갖는 평균버스트 길이 이상으로 송신정보를 확산하여 전송하는 인터리빙기법이 요구된다. 일반적으로 이동통신 채널특성중 단구간 페이딩은 국부 산란과 건물에 의한 반사 등에 의한 다중경로, 수신기의 움직임으로 인한 도플러 효과 등에 의해 발생하고 시간과 이동국 속도의 함수이다. 빠른 페이딩은 기지국으로부터 직접파와 강한 반사파가 존재할 때 라이시안 분포(Rician distribution) 특성을 따르고 가시선(Line of Sight)가 존재하지 않고 순수한 반사파만이 이동국에 수신되는 시가지 환경에서는 레일레이 분포특성을 따른다. 본 논문은 채널 모델링시 통계적 분석을 수행하기 위해 확률밀도함수 *PDF(Probability Density Function)*, 누적확률분포 *CPD(Cumulative Probability Distribution)*, 레벨교차율 *LCR(Level Crossing Rate)*, 평균페이딩 구간 *ADF(Average Duration of Fades)*, 비트오류율 *BER(Bit Error Rate)* 등의 관계함수로부터 평균 버스트 길이를 유도하고 이를 통해 인터리빙 거리를 고려하여 실험을 수행하였다.

1. 라이시안 확률밀도함수(PDF)와 누적확률분포(CPD)

본 논문에서는 라이시안 분포[4][8]를 갖는 채널로 가정하였는데 이는 직접파 성분과 분산이 σ^2인 독립적인 가우시안 성분을 포함하는 반사파 성분으로 구성되므로 다른 페이딩 채널의 모델링이 가능하기 때문이다. 라이시안 채널의 환경에서γ 의 환경은 γ

$_0$가 100(20dB)으로 동일할 때 직접파 전력과 수신파 전력비가 다른 환경인 K값에 따른 평균 신호대 잡음비를 변수로 하는 여러 전파환경에서 γ에 대한 확률밀도함수는 그림1에서와 같다. 라이시안 페이딩 모델은 직접파 성분과 반사파 성분이 복합된 수신신호로 K값이 다른 여러 전파환경에서 포락선 크기(r)의 제곱을 γ로 정의하고 γ_0와 K를 파라메터로 하는 γ에 대한 확률밀도함수 $P_R(\gamma)$는 다음 식(1)에서와 같다.

$$P_R(\gamma)=\frac{K+1}{\gamma_0}\, e^{\left[-K-\frac{\gamma(K+1)}{\gamma_0}\right]} \cdot I_0\left[2\sqrt{\frac{\gamma K(K+1)}{\gamma_0}}\right] \tag{1}$$

γ는 순시 수신반송파대 잡음전력비, $I_0(\cdot)$는 0차 변형베셀 함수, K는 직접파전력($\frac{a^2}{2}$) 대 반사파 전력(σ^2)비로서 $K=\frac{a^2}{2\sigma^2}$ 의 값을 가진다. 라이시안 확률분포를 갖는 포락선 크기 r의 제곱 평균 $E(r^2)=a^2+2\sigma^2$를 γ_0로 정의하고 평균 수신반송파대 잡음전력비를 나타낸다. 라이시안 확률분포로부터 페이딩 신호레벨과 진폭의 확률분포 관계인 CPD를 다음 식 (2)에서와 같이 구할 수 있다.

$$P(\gamma\le L)=\int_{\gamma=0}^{L} P_R(\gamma)d\gamma \tag{2}$$

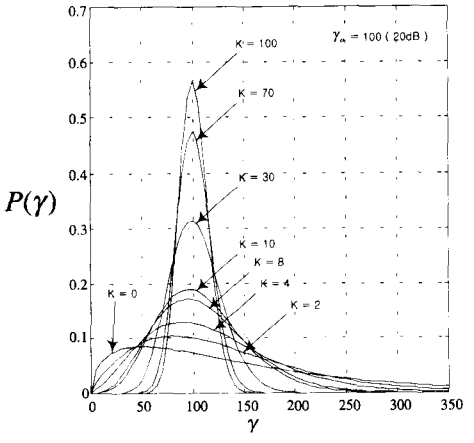


그림 1. 라이시안 채널 확률밀도함수(γ_0=100)

또한 라이시안 채널에서 오류율은 다음 식(3)와 같다.

$$P_e=\int_{\gamma=0}^{\infty} P_b \cdot P_R(\gamma)d\gamma \tag{3}$$

이때 P_b는 잡음영향으로 인한 비트 오류율이고 $P_R(\gamma)$는 라이시안 채널의 확률밀도함수이다. 지역에 따른 라이시안 분포의 K파라미터는 도심지역의 대도시 0.0~1.0, 중소도시 1.0~3.0, 외곽지역의 교외지역 3.0~4.0, 개활지 4.0~10.0의 값을 가지는 것으로 고려되고 있다. 그러나 교외지역일 경우 송수신기 위치에 따라 K파라미터는 큰 차이를 가질 수 있다. $K=0$인 경우 레일레이 페이딩 분포이고 페이딩이 매우 강한 전파환경에 해당한다. 이 환경은 직접파 성분이 없고 반사파만 수신하는 경우이고 $K=\infty$이면 페이딩이 거의 없는 전파환경으로 자유공간으로 볼 수 있다.

2. 레벨교차율(LCR)과 평균페이딩 구간(ADF)

레벨 L이 주어질 때 레벨 교차율은 전체 T 시간 동안 전체 레벨 교차 회수 N으로 결정되고 식 (4)에서와 같다.

$$n(\gamma-L)=\frac{N}{T} \tag{4}$$

그러므로 페이딩 신호의 레벨교차율(LCR)과 평균 페이딩 구간은 식 (5)와 (6)과 같이 계산할 수 있다.

$$n'(\gamma-L)=\frac{4}{T} \tag{5}$$

$$t'(\gamma-L)=\sum_{i=0}^{4}\frac{t_i}{4} \tag{6}$$

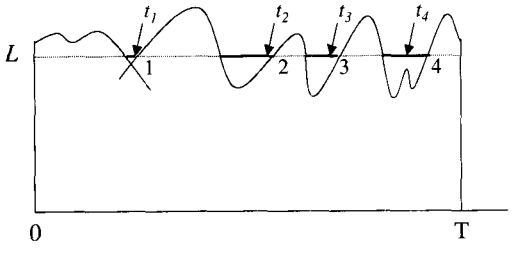


그림 2. 레벨교차율과 평균페이딩 구간

누적확률분포(CPD)는 다음 식 (7)과 같이 얻을 수 있다.

$$P(\gamma-L)=\sum_{i=0}^{N}\frac{t_i}{N} \tag{6}$$

레벨교차율(LCR), 평균페이딩 구간(ADF), 누적확률분포(CPD)의 3가지 파라미터의 관계식은 식 (7)에서와 같다.

$$LCR \cdot ADF = CPD \quad (7)$$

상기 관계식으로부터 페이딩의 분포가 버스트 길이에 미치는 영향을 살펴본다. $K=0$인 환경에서 확률밀도함수(*PDF*), 레벨교차율(*LCR*), 평균페이딩 구간(*ADF*), 누적확률분포(*CPD*)의 관계로부터 버스트 길이를 유도하고자 한다. 각 평균전력변이에 따른 레벨교차율, 평균페이딩 구간, 누적확률분포는 표 1에서와 같다. 먼저 850MHz의 주파수 대역에서 24Km/h의 속도로 이동체가 이동한다고 가정한다. 이때 평균 소요전력비에 비해 -10dB의 환경일 때 CPD는 $P(\gamma \leq -10dB)$의 값으로 0.099를 가진다. 평균레벨교차율 $n(L)$은 $n_0 \cdot n_L$로 결정된다. 정규화 인자 n_0는 $2.5 \cdot \frac{v}{\lambda} = 2.5 \cdot \frac{v}{c/f}$ 인 주파수(f)와 이동국 속도(v) 함수이다. c는 전파의 전달속도이다. n_L은 전계와 자계에서 신호전력 함수로서 페이딩의 PDF 관계함수이다. 그러므로 평균전력 레벨로부터 10dB 이하에서 예상되는 레벨교차율은 $n_0 \cdot n_L$이므로 47 × 0.284 = 13.35 crossings/sec 값으로 결정된다. 따라서 평균 페이딩 구간은 (*CPD*) / (*LCR*) = 0.007416 sec 이고 9600bps에서 전송할 경우 72비트의 버스트 오류가 발생한다. 페이딩은 신호강도가 40dB 범위에 걸쳐 변동한다. 평균페이딩 구간(*ADF*)은 평균전력에서 10dB까지는 상승하고 -30dB까지는 하강한다.

표 1. 평균전력변이에 따른 버스트 길이($K=0$, 주 파수 대역 *850MHz*, 이동속도 *24Km/h*)

평균전력변이 / 결정요소			-25dB	-10dB	0dB	5dB
$n_0(=2.5xv/\lambda)$			47			
n_L			0.066	0.284	0.33	0.05
$n(L)=n_0 \cdot n_L$ (*LCR*)			3회	13회	15회	2회
$t'(L)=t'_0 \cdot t'_L$ (*ADF*)			0.00097	0.007423	0.038728	0.40044
$CPD=P(\gamma \leq L)$			0.003	0.099	0.6	0.94
$t'_0 = 1/n_0$			0.0213			
$t'_L = CPD/n_L$			0.0455	0.3486	1.8182	18.8
전송속도	19200	*Burst Length(bits)*	20	144	742	7688
	9600		10	72	371	3844
	4800		5	36	186	1922
	2400		3	18	93	961

3. 평균버스트길이 (ABL, Average Burst Length)

본 논문에서는 무선채널을 라이시안 분포를 갖도록 하고 버스트 오류 발생위치를 랜덤하게 발생시켰으며 버스트 길이는 기하분포를 갖는 것으로 가정하였다. ITU-T 전신인 CCITT "Blue Book"[15]에서는 버스트 오류에 대해 "a group of bits in which two successive erroneous bits are always separated by less than a given number(x) of correct bits" 로 정의하고 있다. 평균 버스트 길이는 버스트 오류 성능을 평가하는 중요한 기준이다. 평균 버스트 길이 B는 모든 버스트의 전체 길이(L_t)를 버스트의 전체 수(N_t)로 나눈 값으로 결정된다.

$$B = \frac{L_t}{N_t} \quad (8)$$

평균버스트 길이가 주어질 때 버스트 길이가 갖는 확률은 기하분포를 갖고 식(9)에서와 같이 정의될 수 있다.

$$p(L) = \frac{1}{B}\left(1 - \frac{1}{B}\right)^{L-1} \quad (9)$$

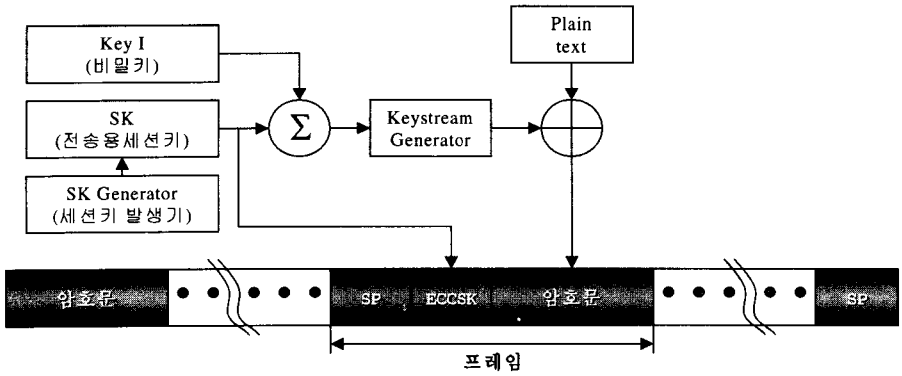
이때 B는 평균 버스트 길이이고, L은 버스트 길이이며 $p(L)$은 버스트길이 L일 때 버스트 길이 L이 발생할 확률을 의미하고 이 확률에 근거하여 비트 오류율 $10^{-1} \sim 10^{-6}$까지 기하분포[14][16][21]로 버스트 오류를 발생시켰다.

Ⅲ. 동기식 스트림 암호시스템

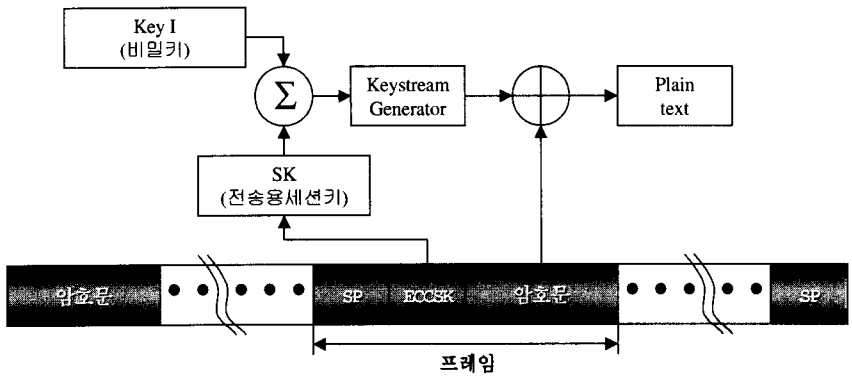
스트림 암호시스템에서 키수열 발생기는 외부 키입력을 시드값(seed number)로 하여 무한주기에 가까운 랜덤 키수열을 발생시킨다.

스트림 암호시스템의 암복호 과정은 다음 그림 3에서 제시한 바와 같다. 주기적으로 동기 패턴과 세션 키를 송신측에서 전송하고 수신측에서는 복호를 위해서 송수신측이 공유한 비밀키와 수신된 세션키를 사용하여 키수열 발생기의 초기 상태 값을 결정하고 주기적으로 동일한 동기패턴으로부터 동기를 유지한다.

먼저 주기적인 동기식 스트림 암호통신 방식을 운용하기 위해서 요구되는 동기패턴은 송수신이 동일한 값을 가져야 하며, *Gold Code Generator*[12]를 이용하여 발생시키고 그 패턴길이는 사용하는 채널



(a)



(b)

그림 3. 주기적인 동기방식을 이용한 스트림 암호시스템
(a) 암호화기 (b) 복호화기

환경에 적합하도록 결정한다. 암호기는 일정 길이를 갖는 동기패턴을 평균 비트오류율이 주어진 통신채널을 통해 전송하면 수신기는 동기검출 확률 P_D와 False Alarm 확률 P_{FA}를 유도하여 해당 동기패턴의 허용여부를 판별하게 된다. 세션키는 64비트 시드값을 세션키 발생기의 입력으로 하여 생성된 랜덤수열을 각 64비트씩 분할하여 사용하며 각 세션에 해당하는 암호문을 생성하기 위한 키수열 발생기의 키 값 중에 일부분으로 해당 세션키를 이용한다. 스트림 암호시스템에서의 비도수준은 암호공격에 강한 키수열 발생기의 설계에 의해 결정되므로 일반적으로 키 수열의 주기에 대한 최대값의 보장, 난수성이 좋음, 높은 상관 면역도를 가질 것, 큰 선형 복잡도를 지닐 것 등의 요구 사항을 만족해야 한다.

1. 동기패턴발생기

동기패턴 발생기는 자기상관특성이 우수한 *Gold Code Generator*를 이용하여 동기패턴을 발생시키고 복호기에서는 동기패턴 검출 과정을 수행하여 수신한 동기패턴이 검출되면 이어서 수신되는 세션키를 수신하여 암호문을 복호한다. 골드코드로 구성된 동기패턴 발생기는 다음 그림 4에서 제시하였다.

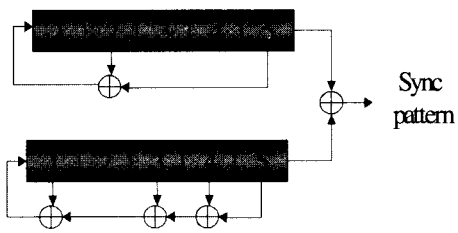


그림 4. 골드코드 구조

31비트의 키길이를 갖는 동기패턴발생기의 경우 5단 *LFSR1*, *LFSR2*를 *XOR* 과정을 수행하므로써 동기패턴을 얻게되고 원시다항식을 다음 식에서와 같이 구성할 수 있다.

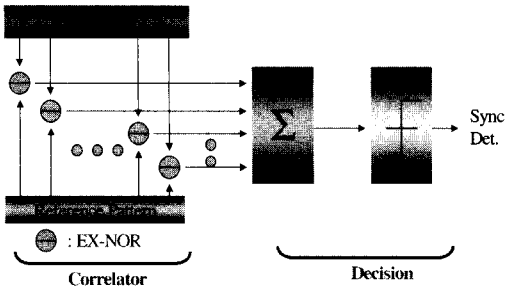


그림 5. 동기패턴 검출기 구조

$$f(x) = x^5 + x^2 + 1 \tag{10}$$

$$g(x) = x^5 + x^4 + x^3 + x + 1 \tag{11}$$

동기패턴 검출기는 *Beker & Piper*[2]의 동기패턴 검출기 모델을 이용하여 설계하였으며 그림 5에서와 같다. 송신패턴을 검출과정은 *Correlator*과 *Decision*으로 나누어 처리된다.

2. 세션키발생기

세션키 발생기는 다음 그림 6과 같이 비선형 키 수열 발생기로 한다.

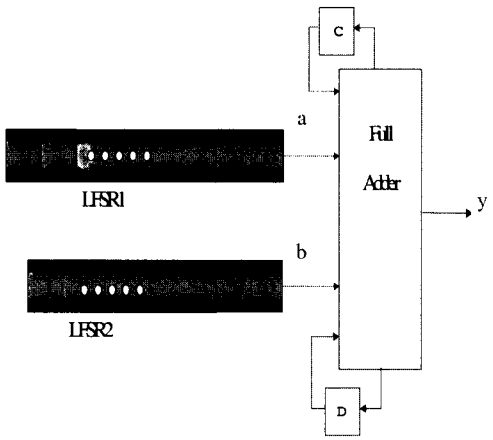


그림 6. 비선형 세션키 발생기

출력함수 y_j는 LFSR1의 출력 수열 a_j와 LFSR2의 출력 수열 b_j, 이전 carry c_{j-1}(carry의 초기값은 0), 이전 피드백 메모리 비트 D_{j-1} 의 전가산기를 통해 비선형으로 구해지고 각 출력은 다음 식과 같이 표현된다.

$$y_j = (a_j \oplus b_j \oplus C_{j-1}) \oplus D_{j-1} \tag{12}$$

$$C_j = a_j b_j \oplus (a_j \oplus b_j) C_{j-1} \tag{13}$$

$$D_j = b_j \oplus (a_j \oplus b_j) D_{j-1} \tag{14}$$

이때, $j=0,1,2,...$이다. 이 수열 발생기는 전가산기만으로 구성된 기존 발생기의 carry와 출력간의 상관 확률이 1/4로 매우 커서 상관공격에 취약하지만 feedback memory 함수를 추가하여 상관 확률이 1/2인 함수로 개선된 알고리즘을 사용하였다. 개선된 비선형 세션키 수열 발생기의 선형 복잡도는 주기 P가 $(2^{31}-1)(2^{61}-1)$에 근접하며 상관 면역도는 최고 차수를 갖는다.

3. 키수열발생기

키수열 발생기는 출력비트 수열의 선형 복잡도, 랜덤 특성, 상관 면역도 등을 고려하여 설계된 비선형 함수이다. 제안된 키수열 발생을 위해 사용된 비선형 키수열 발생기는 각 *LFSR* 주기의 단수가 29, 59, 89, 109가 서로 소로서 전가산기를 통해 구성하였으며 그림 7에서 제시한다.

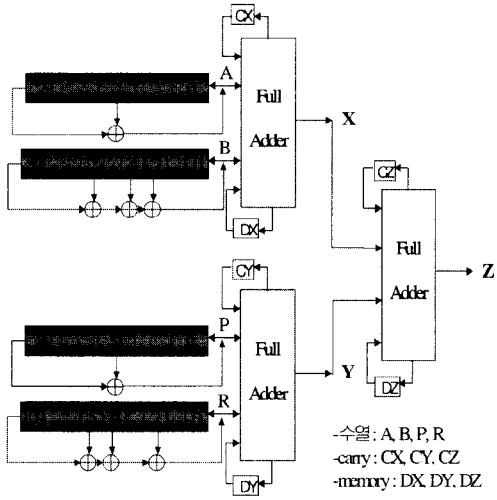


그림 7. 비선형 키수열 발생기

비선형 키수열 발생기는 비도측면에서 표 2에서 제시하는 바와 같이 무선 채널에 적용 가능한 비도 능력을 갖는다. 비선형 키수열 발생기가 갖는 비도(security level)은 주기가 $(2^{29}-1)\cdot(2^{59}-1)\cdot(2^{89}-1)\cdot(2^{109}-1)$으로 거의 10^{86} 값을 갖도록 설계하였으며, 선형 복잡도는 주기에 근접하도록 하였고 상관면역도는 최고의 차수를 갖도록 하였다. 함수 X_j는 *LFSR1*의 출력 수열 A_j와 *LFSR2*의 출력 수열 B_j, 이전 *carry* CX_{j-1}(*carry*의 초기값은 0), 이전 피드백

메모리 비트 DX_{j-1} 의 전가산기를 통해 비선형으로 구해지고 식 (15)~(17)에서와 같다.

$$X_j = (A_j \oplus B_j \oplus CX_{j-1}) \oplus DX_{j-1} \quad (15)$$

$$CX_j = A_jB_j \oplus (A_j \oplus B_j)CX_{j-1} \quad (16)$$

$$DX_j = B_j \oplus (A_j \oplus B_j)DX_{j-1} \quad (17)$$

함수 Y_j는 *LFSR2*의 출력 수열 P_j와 *LFSR3*의 출력 수열 R_j, 이전 *carry* CY_{j-1}(*carry*의 초기값은 0), 이전 피드백 메모리 비트 DY_{j-1}의 전가산기를 통해 식 (18)~(20)에서와 같이 비선형으로 구해진다.

$$Y_j = (P_j \oplus R_j \oplus CY_{j-1}) \oplus DY_{j-1} \quad (18)$$

$$CY_j = P_jR_j \oplus (P_j \oplus R_j)CY_{j-1} \quad (19)$$

$$DY_j = R_j \oplus (P_j \oplus R_j)DY_{j-1} \quad (20)$$

따라서 출력 Zj 는 식 (15)~(20)을 통해 식 (21)~(23)을 얻을 수 있다.

$$Z_j = (X_j \oplus Y_j \oplus CZ_{j-1}) \oplus DZ_{j-1} \quad (21)$$

$$CZ_j = X_jY_j \oplus (X_j \oplus Y_j)CZ_{j-1} \quad (22)$$

$$DZ_j = Y_j \oplus (X_j \oplus Y_j)DZ_{j-1} \quad (23)$$

이때, $j=0,1,2,...$이다.

4. 키수열 발생기의 랜덤특성 검정

키수열 발생기의 전체 주기로부터 발생된 키수열의 출력으로부터 랜덤 특성을 검정하는 것은 불가능하므로 적당한 비트 길이를 사용하여 국부적인 랜덤 특성 검정을 수행하였다. 이에 대한 검정 방법으로는 카이제곱검정(*chi-square test*)를 사용하여 적합도를 평가하며 이에 대한 유의수준 결정은 일반적으로 5%의 유의수준을 만족하게 되면 적합하다고 판정한다.

일반적인 검정 항목[18]은 *frequency test*, *serial test*, *t-serial test*, *poker test* 및 *autocorrelation test* 등이 있다. 설계된 키수열 발생기를 이용하여 얻은 출력 비트 20만 비트를 초기 값(initial seed number)를 달리하여 랜덤 검정을 수행한 결과를 나타낸 것으로 표2에서와 같다. 이때 검정 항목에 따라 얻은 결과를 살펴볼 때 정의하는 유의 수준을 통과함으로써 시스템에 적용할 때 적합하다고 평가된다.

표 2. 설계된 키수열 발생기의 랜덤특성 검정 결과

Test item	유의 수준 (5%)	test 결과 initial seed 1	test 결과 initial seed 2
frequency test	3.841	0.22	1.829
serial test	5.991	0.277	1.835
generalized t-serial test (t=3)	9.488	0.489	1.928
" (t=4)	15.507	9.29	7.626
" (t=5)	26.296	12.508	14.154
poker test (length=3)	14.067	3.88	12.173
" (length=4)	24.996	15.955	12.371
" (length=5)	44.654	37.906	33.992

Ⅳ. FDI(Frame Domain Interleaving)과 GDI(Group Domain Interleaving)

1. 인터리빙 기법

인터리빙 기법[24]은 정보전송 중에 발생하는 버스트 오류[15][16]를 분산시키는 방법으로 정보를 전송하기 전에 발생된 전송 비트열을 달리 배열하여 전송하고 복호측에서 원래의 순서대로 재배열하는 방식이다. 이는 전송중에 발생가능한 버스트 오류를 랜덤 오류와 같은 형태로 분산시키는 기능으로서 오류정정이 가능한 비트의 수가 제한된 순방향 오류정정 부호화의 효율을 극대화할 수 있다.

본 논문에서는 높은 보안수준을 유지하기 위해 무선채널에서 암호시스템을 적용할 때 무선채널 특성상 발생하는 버스트 오류에 대해 송신되는 암호정보를 보호하고 전송성능을 개선하기 위해 인터리빙 기법을 제시하고 또한 제안한 인터리빙 기법의 성능을 평가하고자 한다. 본 논문에서는 무선망에서 송신정보를 설계된 동기식 스트림 암호시스템을 이용해 암호화를 수행하고 이때 발생된 동기패턴, 세션키, 암호문을 제안한 인터리빙 방식을 수행함으로써 무선링크 암호시스템의 전송성능을 개선하였다. 이때 제안된 인터리빙 기법은 하나의 프레임을 기준으로 인터리빙을 수행하는 프레임영역 인터리빙 (Frame Domain Inter- leaving)과 다수개의 프레임이 하나의 그룹으로 구성되어 인터리빙이 수행되는 그룹영역 인터리빙 (Group Domain Inter- leaving)이다.

2. FDI(Frame Domain Interleaving)

프레임영역 인터리빙(FDI)는 동기식 스트림 암호

시스템에서 한 주기동안 발생하는 동기패턴(SP), 세션키(ECCSK), 암호문(Encrypted Data)으로 구성된 하나의 프레임단위로 인터리빙을 수행하고 그림 8에서와 같다.

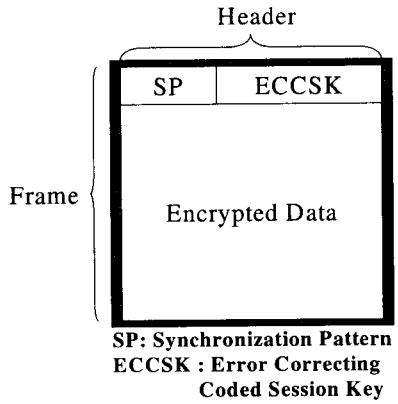


그림 8. 프레임영역 인터리빙 구조

이때 헤더(Header)는 동기패턴과 세션키로 구성한다. 프레임의 크기(N)은 동기패턴, 세션키, 암호문의 크기를 갖고 Nbits(2400, 4800, 9600, 19.2Kbps)마다 새로운 동기패턴, 세션키, 암호문을 인터리빙하여 수신측에 전송한다.

프레임영역 인터리빙은 프레임 전체영역을 인터리빙하고 동기패턴, 세션키, 암호문을 구분하지 않는다. 인터리빙은 하나의 동기구조를 기준하여 수행하고 인터리빙의 크기를 결정하는 블록크기는 프레임의 크기와 지연시간을 고려하여 구성한다. 프레임 내에서 블록인터리빙을 전체영역에 적용하고 이때 프레임의 구조는 다음 그림 9에서와 같다. 만일 하나의 동기주기를 갖는 동기패턴, 세션키, 암호문을 하나의 프레임으로 정할 때 프레임의 경계는 동기주기단위로 결정한다. 이때 하나의 프레임 크기(N)을 2400비트로 정한다면 블록 인터리빙을 위한 블록의 크기는 $50(m) \times 48(k)$로 구성한다. 가로 방향은 블록인터리버의 간격(m)으로 버스트오류가 발생할 때 오류없이 처리할 수 있는 버스트오류의 최대길이를 나타내고 세로 방향은 버스트 오류의 분산정도(k)를 의미한다.

2.1 블록인터리빙 적용

프레임 전체영역에서 블록 인터리빙 알고리듬을 적용하면 블록인터리버 거리를 m값으로 정할 때 입력 $x_0, x_1, x_2, \ldots, x_{(m-1)}, x_m, x_{(m+1)}, x_{(m+2)}, \ldots, x_{(m+k)}, x_{(m+k+1)}, x_{(m+k+2)}, \ldots, x_{(peri-1)}$ 인 $N(Peri)$개로 구성된 프레임에서 블록 인터리빙을 수행하면 $x_0, x_m, x_{2m}, \ldots, x_{mxk}, x_1, x_{(m+1)}, x_{(m+2)}, \ldots, x_{(m+k+1)}, x_2, x_{(m+2)}, \ldots, x_{(m+k+2)}, \ldots, \ldots, x_{(m-1)}, x_{(2m-1)}, \ldots, x_{(peri-1)}$ 구조를 가지고 수신측에 전송된다.

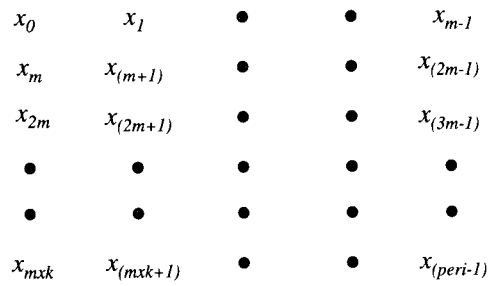


그림 9. 전체영역에 적용된 블록인터리빙 방식

2.2 헬리컬인터리빙 적용

인터리빙 수행방식은 대각으로 읽고 가로로 수행하여 수신측에 전송한 후 역으로 복호한다. 프레임 내의 전체영역에 헬리컬 인터리빙 알고리듬을 적용한다면 $m \times (m+1)$의 구조형식을 가져야 한다. 2400비트의 동기주기를 갖는 하나의 프레임이 있다면 헬리컬 인터리빙을 수행하기 위해서는 48×49의 구조를 갖도록 설계하고 이에 대해 구조는 그림 10에서와 같다.

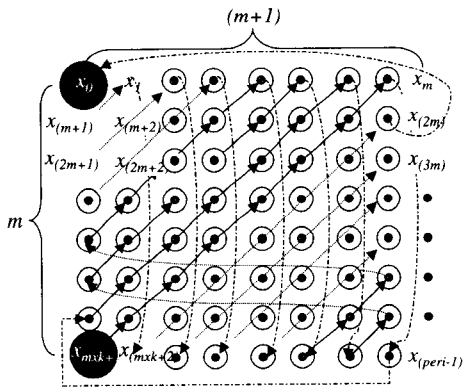


그림 10. 전체영역에서 헬리컬인터리빙 방식

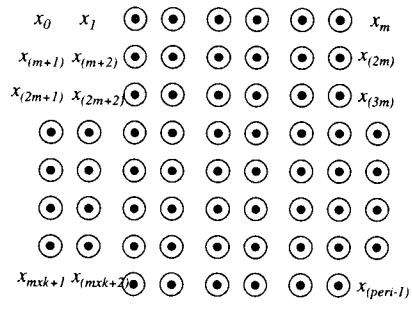
적용구조는 입력 $x_0, x_1, x_2, \ldots, x_{(m-1)}, x_m, x_{(m+1)}, x_{(m+2)}, \ldots, x_{(m+k)}, x_{(m+k+1)}, x_{(m+k+2)}, \ldots, x_{(peri-1)}$ 인 $N(Peri)$개로 구성된 프레임에서 헬리컬 인터리빙을 적용하면 수신측에 전송순서는 $x_{(mxk+1)}$에서 출발하여 $x_{(mx(k-1)+2)}, .., x_{(m-1)}, x_{(peri-1)}, x_{(mx(k-1)+1)}, x_{mx(k-2)+2)}, x_{(m-2)}, x_{(peri-2)}, x_{(mx(k-1)-1)}, \ldots, x_{(mx(k-2)+1)}, x_{(mx(k-3)+2)}, \ldots, x_{(2m-1)}, x_0$ 구조를 가지고 전송된다. 만일 프레임 구조가

2400 비트로 구성된다면 헬리컬 인터리빙을 수행하기 위해 2400비트내에서 48×49의 구조를 갖도록 구성한다. 이때 2352(48×49) 비트는 헬리컬인터리빙을 수행하고 잔여 8비트는 그대로 전송하고 버스트 오류는 $(m+1)$ 비트만큼 확산한다.

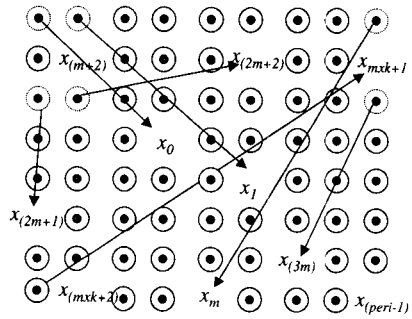
2.3 랜덤인터리빙 적용

랜덤 인터리빙 방식은 프레임내의 전체영역에 난수발생기에서 발생되는 난수열을 이용하여 특정위치로 분산시킨다. N비트(2400, 4800, 9600 등)의 동기주기를 갖는 동기식 스트림 암호시스템에서 인터리빙을 프레임내의 전체영역에 적용할 때 랜덤인터리빙은 입력된 정보 비트를 랜덤 테이블에 의해 재배열한다. 이때 사용되는 랜덤 테이블은 난수발생기에 의해 발생된 테이블을 근거하여 작성하고 랜덤 인터리빙 구조는 다음 그림 11에서와 같다. 프레임 전체영역에서 랜덤 인터리빙 알고리듬을 적용하기 위해 난수발생기를 통하여 랜덤 인터리버 테이블을 생성하고 생성된 테이블에 따라 입력 $x_0, x_1, x_2, \ldots, x_{(m-1)}, x_m, x_{(m+1)}, x_{(m+2)}, \ldots, x_{(m+k)}, x_{(m+k+1)}, x_{(m+k+2)}, \ldots, x_{(peri-1)}$ 인 $N(Peri)$개로 구성된 정보를 난수발생기의 랜덤 인터리버 테이블에 근거하여 위치를 결정한다. 그림 11에서 그림 (a)는 동기패턴, 세션키, 암호문으로 구성된 입력정보를 나타내고 그림 (b)는 랜덤 테이블에 의해 수행되는 인터리빙을 보여준다. 그러므로 그림 (a)의 $x_0, x_1, \ldots, x_{(peri-1)}$ 는 랜덤테이블을 통해 그림 (b)의 새로운 $x_0, x_1, \ldots, x_{(peri-1)}$의 위치로 특정된다.

이때 인터리빙 출력 순서는 그림 (b)에서 랜덤한 순서인 $\ldots, x_{(m+2)}, \ldots, x_{(2m+2)}, \ldots, x_{(mxk+1)}, \ldots, x_0, \ldots, x_{(2m+1)}, \ldots, x_1, \ldots, x_{(3m)}, \ldots, x_m, \ldots, x_{(peri-1)}$ 구조를 가지고 출력으로 전송된다. 따라서 랜덤 인터리빙의 인터리빙 효과는 난수발생기에서 결정하는 인터리빙 거리가 평균적으로 균등한 분포를 갖는지 여부에 좌우한다.



(a)



(b)

그림 11. 전체영역에서 랜덤인터리빙 방식
(a) 정보의 입력구조
(b) 랜덤인터리빙 위치결정

또한 랜덤 인터리빙의 효과는 랜덤 테이블의 크기에 따라 인터리빙의 확산거리에 비례하여 증가한다.

2.4 확장된 랜덤인터리빙 적용

확장된 랜덤인터리빙 방식은 종래의 랜덤인터리빙 방식에서 인터리빙되는 인터리빙거리의 제약조건을 고려한다. 이전 비트들을 고려하여 확산거리가 $\pm L$ 이내에 들지 않도록 설계해야 한다. 거리 L은 확산거리를 고려하여 결정한다. 확장된 랜덤인터리빙 방식을 프레임에 적용할 때 인터리빙 거리 $\pm L$의 크기를 결정하는 것은 지연시간과 메모리요소 등을 고려해야 한다. 만일 2400 비트의 동기주기를 갖는 암호시스템의 프레임일 경우 블록인터리빙에서 갖는 확산거리를 고려하여 인터리빙 거리를 특정하고 구조는 그림 12에서와 같다. 프레임 전체영역에서 확장된 랜덤 인터리빙 알고리듬 적용은 난수발생기를 통하여 랜덤 인터리버 테이블을 생성하고 생성된 테이블에서 이전 결정된 위치정보와 현재 결정해야 할 위치정보사이에 인터리빙 거리를 계산하여 인터리빙 거리가 $\pm L$이내에 존재여부를 판단하고 $\pm L$이상의 인터리빙 거리가 되도록 결정한다. 입력 $x_0, x_1, x_2, \ldots, x_{(m-1)}, x_m, x_{(m+1)}, x_{(m+2)}, \ldots, x_{(m+k)}, x_{(m+k+1)}, x_{(m+k+2)}, \ldots, x_{(peri-1)}$ 인 $N(Peri)$개로 구성된 정보를 난수발생기의 확장된 랜덤 인터리버 테이블에 근거하여 위치를 결정하고 그림 12에서 좌측 그림은 입력된 정보에 대한 확장랜덤 인터리빙을 적용시 우측 그림과 같이 인터리빙을 수행한다. 인터리빙 출력 순서는 이전 인터리빙 거리를 고려한 랜덤테이블에 의해 $\ldots, x_{(m+2)}, \ldots, x_{(2m+2)}, \ldots, x_{(mxk+1)}, \ldots, \underline{x_0} \ldots, x_{(2m+1)}, \ldots, \underline{x_1} \ldots, x_{(3m)}, \ldots, x_m, \ldots, x_{(peri-1)}$

구조를 갖는다. 이때 x_0 위치결정 후 x_1 위치를 결정할 때 난수발생기의 출력으로부터 결정되는 인터리빙 거리l은 기준 인터리빙 거리($\pm L$)이내에 존재하므로 기준 인터리빙 거리($\pm L$) 이상을 만족하는 x_1'위치로 재결정 과정을 수행하게 된다.

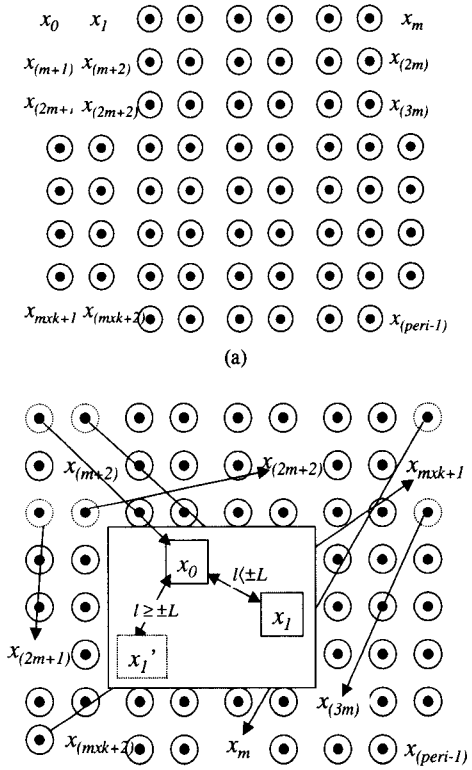


그림 12. 전체영역에 적용된 확장된 랜덤인터리빙 방식
(a) 정보의 입력구조
(b) 확장된 인터리빙 위치결정

3. GDI(Group Domain Interleaving)

그룹영역 인터리빙은 하나의 프레임이 동기식스트림 암호시스템으로부터 한 주기동안 발생되는 동기패턴(*SP*), 세션키(*ECCSK*), 암호문(*Encrypted Data*)으로 구성된 다수개의 프레임이 하나의 그룹을 이룬다. 이때 프레임의 크기(N)은 하나의 동기주기이고 *Nbits(2400, 4800, 9600, 19.2Kbits)*로 구성된 프레임이 모여 하나의 그룹을 이루어 인터리빙되어 수신측에 전송된다. 적용구조는 그림 13에서와 같고 인터리빙 방식은 다수개의 프레임이 구성된 그룹단위로 블록인터리빙을 수행한다. 프레임영역 블록 인터리빙 원리가 그룹영역 블록 인터리빙에도 동일하게 적용된다. 단지 그룹의 전체 크기가 달라지고 지연시간과 큰 메모리가 요구된다는 점을 제외하고는 블록 인터리빙 알고리듬과 동일한 원리이다. 그룹 인터리빙은 무선 채널 링크의 다중화에 사용할 수 있다. 링크계층, 매체제어계층, 물리계층에 이르기까지 해당 계층의 프레임에 암호화를 수행한 후 전송하는 정보에 다중화 인터리빙을 수행할 때 사용할 수 있다.

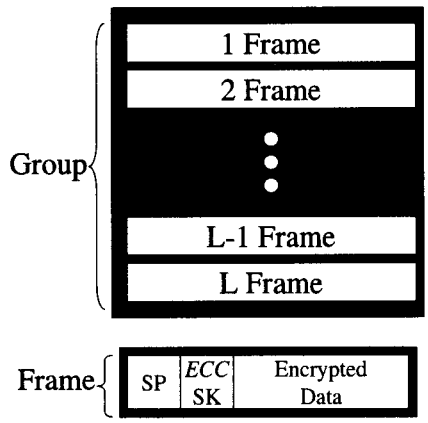


그림 13. 그룹단위 인터리빙 기준구조

하나의 그룹이 프레임 1(*Frame 1*), ..., 프레임 *N*(*Frame N*)개로 구성될 때 그룹 전체영역에서 블록인터리빙 수행은 다음 그림 14에서와 같다. 그룹 입력 *Frame1* {x_0, x_1, x_2, ..., $x_{(m-1)}$, x_m, $x_{(m+1)}$, $x_{(m+2)}$, ..., $x_{(m+k)}$, $x_{(m+k+1)}$, $x_{(m+k+2)}$, ..., $x_{(peri-1)}$ }, *Frame2* {x_0, x_1, x_2, ..., $x_{(m-1)}$, x_m, $x_{(m+1)}$, $x_{(m+2)}$, ..., $x_{(m+k)}$, $x_{(m+k+1)}$, $x_{(m+k+2)}$, ..., $x_{(peri-1)}$ }, ..., *FrameN* {x_0, x_1, x_2, ..., $x_{(m-1)}$, x_m, $x_{(m+1)}$, $x_{(m+2)}$, ..., $x_{(m+k)}$, $x_{(m+k+1)}$, $x_{(m+k+2)}$, ..., $x_{(peri-1)}$ }으로 구성될 때 인터리빙 거리 m을 갖도록 블록 인터리빙을 수행한다고 가정한다. 이때 그룹전체영역에서 블록인터리빙의 출력은 {*Frame1* {x_0}, *Frame2* {x_0}, ..., *FrameN* {x_0}}, {*Frame1* { $x_{(m+1)}$}, *Frame2* { $x_{(m+1)}$}, ..., *FrameN* {$x_{(m+1)}$}}, ..., {*Frame1* { $x_{(mxk+1)}$}, *Frame2* { $x_{(mxk+1)}$}, ..., *FrameN* {$x_{(mxk+1)}$}}, {*Frame1* {x_1}, *Frame2* {x_1}, ..., *FrameN* {x_1}}, {*Frame1* {x_{m+2}}, *Frame2* {x_{m+2}}, ..., *FrameN* {x_{m+2}}}, ..., {*Frame1* { $x_{(mxk+2)}$}, *Frame2* { $x_{(mxk+2)}$}, ..., *FrameN* {$x_{(mxk+2)}$}}, {*Frame1* {x_m}, *Frame2* {x_m}, ..., *FrameN* {x_m}}, {*Frame1* { $x_{(2m)}$}, *Frame2* { $x_{(2m)}$}, ..., *FrameN* {$x_{(2m)}$}}, {*Frame1* {x_{peri-1}}, *Frame2* {x_{peri-1}}, ..., *FrameN* {x_{peri-1}}}구조를 가진다.

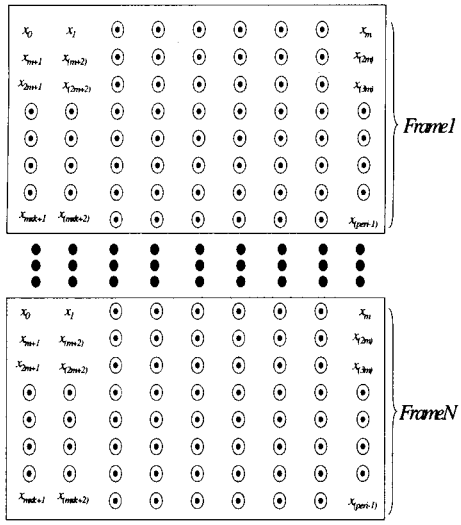


그림 14. 전체그룹에 적용된 블록인터리빙

그러므로 인터리빙 거리는 프레임의 개수에 따라 증가한다.

V. 시뮬레이션 결과 및 고찰

1. 암호시스템 시뮬레이션 환경

암호통신망 모델에 라이시안 채널환경으로 가정하였고 버스트 오류 발생위치는 랜덤 난수발생기를 이용하여 실험을 수행하였다. 버스트 오류 길이는 기하분포를 갖도록 가정하고 평균 버스트 길이를 달리하면서 실험을 수행하였다. AX.25 프로토콜 절차에는 링크설정, 정보전송, 링크해제 등의 순으로 이루어지고 그림 15에서와 같다. 먼저 링크를 설정하기 위해서는 송신국은 수신국에게 SABM(Set Asynchronous Balanced Mode) 명령 프레임을 송신하여 T1 타이머를 구동한다. 수신국이 정상상태로 암호통신이 가능하면 수신국은 SABM의 응답으로 UA (Unnumbered ACK) 프레임을 송신국에 보내고 수신국은 내부 상태변수 V(S), V(R)을 "0"으로 reset한다. 송신국이 UA 프레임을 수신하면 자신의 내부상태 변수 V(S), V(R)을 "0"으로 reset한다. 만일 연결 가능상태가 아니면 DM(Disconnect Mode) 프레임을 송신국에 전송한다. 링크가 설정되면 송신국은 정보전송을 수행한다.

이때 I 또는 S 프레임을 송수신한다. 정보전송상태에서 SABM 프레임을 수신하면 송신국은 reset 상태로 들어간다. 정보전송상태에서 링크를 해제하려고 하면 DISC (Disconnect) 명령을 수신국에 전

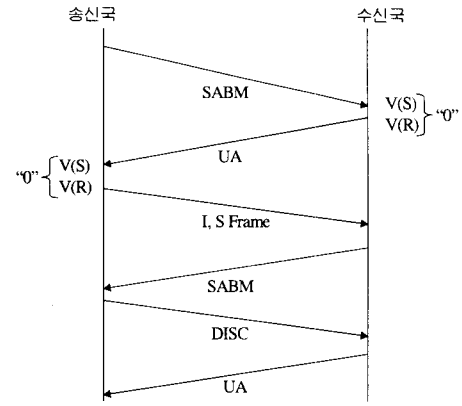


그림 15. AX.25 링크프로토콜 절차

송하고 수신국이 DISC 프레임을 수신하면 UA 프레임을 송신국에 전송하고 연결해제 단계에 들어간다. 송신국은 수신국에서 전송한 UA 프레임을 수신하면 T1 타이머를 정지시킨다. 만일 수신국으로부터 UA 프레임을 응답하기 전에 타이머가 time-out이 발생하면 DISC 프레임을 재전송하고 타이머를 재구동시킨다. 연결해제 상태에서 송수신국이 SABM 프레임을 수신하면 수신한 측에서는 SABM 링크 설정단계로 들어가고 DISC 프레임을 수신하면 DM 프레임으로 응답한다.

2. 암호시스템 시뮬레이션 결과

본 논문은 무선시스템에서 링크계층에 적용된 동기식 스트림 암호시스템의 전송성능 개선을 위해 암호시스템 내부에 암호화 구조에 적합한 인터리버를 추가하였다. 본 논문에서 사용된 실험 영상은 1600x1200 화소인 $1.536x10^7$ 비트 정보량을 갖는다. 본 논문의 실험내용은 첫째 암호시스템에서 850MHz대역과 220MHz대역에서의 인터리빙 영향, 둘째 인터리빙 depth에 따른 전송성능을 통해 제안한 FDI와 GDI의 성능을 평가한다.

2.1 주파수대역에 따른 페이딩 영향 분석

850MHz대역과 220MHz대역에서 페이딩 영향 분석을 위해 수신 평균전력을 -25dB, -10dB, 0dB, 5dB가 랜덤하게 변동시켜 발생시켰다. 프레임영역 인터리빙을 850MHz대역에 적용했을 때 표3에서와 같이 헬리컬, 랜덤, 확장랜덤 인터리빙 가운데 랜덤 인터리빙 성능이 우수하게 나타나고 블록인터리빙에서 인터리빙 depth를 증가함에 따라 블록 인터리빙 성능이 점차적으로 우수하게 나타났다. 그러므로 프레임영역 인터리빙에서는 블록인터리빙의 경우 충분

히 긴 인터리빙 depth를 제공하면 랜덤, 확장랜덤, 헬리컬보다 우수한 성능을 얻을 수 있다. 이는 버스트 오류가 랜덤 위치로 발생하고 랜덤, 확장랜덤 인터리빙의 랜덤테이블이 프레임 크기에 제한되어 오히려 인터리빙 depth가 작은 경우가 발생할 수 있다. 최적의 인터리빙은 블록인터리빙에서 충분한 인터리빙 depth를 가질 때 가장 우수한 성능을 얻게 된다. 인터리빙을 수행하지 않았을 때 850MHz 대역은 35.3%의 비트오류율을 220MHz 대역은 42.8%의 비트오류율을 얻는 데 이는 주파수 대역이 정규화 인자 n_0에 직접적인 영향을 주는 인자로 작용하기 때문이다. 850MHz의 주파수대역에서 인터리빙의 성능은 헬리컬의 경우 14.7%의 비트오류율을, 랜덤의 경우 13.4%, 확장랜덤의 경우 13.8%, 블록인터리빙의 경우 인터리빙 depth가 80~120사이에서 헬리컬, 랜덤, 확장랜덤과 유사한 성능을 얻고 200으로 증가함으로써 8%의 비트오류율을 나타낸다. 따라서 9600비트 구조에서 인터리빙을 수행할 때 블록 인터리빙은 인터리빙 depth를 80까지는 별다른 차이를 보이지 않지만 depth를 120~200의 40 값마다 2%씩 성능이 증가한다. 그러나 220MHz 주파수 대역의 경우에는 850MHz대역보다 평균 페이딩 구간이 넓게 분포되어 나타나므로 850MHz대역에 적용된 인터리빙 구조에 비해 인터리빙의 효과가 감소하고 헬리컬과 랜덤의 경우 블록에서 40~80사이 인터리빙 depth를 적용한 것과 유사한 성능을 얻을 수 있다.

표 3. 인터리빙 유형에 따른 비트오류양 (비트오류율:10^{-2})

주파수대역 / 인터리빙유형		850MHz	220MHz
No 인터리빙		5602602(35.3%)	6721440(42.8%)
블록 인터리빙	40	2791415(16.6%)	5056139(31.7%)
	80	2773119(16.5%)	3935026(24.2%)
	120	2171231(12.5%)	3489930(21.3%)
	160	1895486(10.6%)	3489176(21.3%)
	200	1485827(8%)	3486565(21.2%)
헬리컬 인터리빙		2504800(14.7%)	4455386(27.7%)
랜덤 인터리빙		2303184(13.4%)	4533132(28.2%)
확장랜덤 인터리빙		2367745(13.8%)	3794881(23.3%)

실제 220MHz 주파수 대역의 경우 블록인터리빙에서 인터리빙 depth가 200 이상이 요구된다고 예측 가능하지만 실제 인터리빙 depth를 무작정 증가시키는 것은 전송지연을 초래하므로 비효율적이다. 또한 페이딩 영향이 심한 채널의 경우 9600비트와 같은 맵 테이블이 정해진 영역에서 헬리컬, 랜덤, 확장랜덤 인터리빙은 비효율적이고 가장 적합한 인터리빙 구조는 블록인터리빙에서 충분한 인터리빙 depth를 가질 때이다.

2.2 블록인터리빙 depth에 따른 전송성능 영향

수신 평균전력변이는 평균페이딩 구간에 직접적인 영향을 미치고 평균페이딩 구간으로부터 유도된 평균 버스트 오류길이가 10, 37, …, 9600비트가 주어질 때 인터리빙 depth를 증가함에 따라 발생하는 전송성능 개선효과를 표4에서 제시하였다. 가로축은 인터리빙을 수행하지 않았을 때와 인터리빙 depth를 40, 80, 120, …, 200으로 결정하고 세로축의 평균 페이딩 구간의 영향으로 인해 발생하는 버스트 오류비트로부터 전송성능을 살펴본다. 적용 구조는 9600비트로 구성된 암호정보를 단위로 블록 인터리빙을 수행하였다. 평균 버스트 오류길이가 72(비트오류율이 $1.0\text{x}10^{-3}$)비트와 279비트(비트오류율이 $3.9\text{x}10^{-3}$)의 채널에서는 인터리빙 depth를 40비트로 정할 때 72비트의 경우 0.74%에 비해 279비트의 경우 42.8%의 비트오류율이 발생하고 depth를 80비트 이상으로 증가시킴으로써 버스트 오류가 279비트 채널환경에서도 3.2%이하로 크게 감소시킬 수 있다.

이는 인터리빙 depth 40과 80사이의 차이가 동기패턴, 세션키의 오류에 영향을 줌으로써 한 주기내의 암호정보의 손실유무를 결정하기 때문이다. 이때 버스트 오류가 371(비트오류율이 $3.86\text{x}10^{-2}$)비트 채널환경에서는 인터리빙 depth를 120으로 했을 때 15.5%, 160으로 했을 때 15.5%, 200으로 했을 때 3.7%의 비트오류율을 발생한다. 이는 인터리빙 depth가 120과 160의 경우 세션키와 동기패턴의 오류로 인한 프레임 전체의 손실영향이 50%정도로 볼 수 있고 depth가 200일 경우 손실영향을 감소시킬 수 있는 충분한 거리를 갖는 인터리빙 depth가 된다고 판단된다. 371(비트오류율이 $3.86\text{x}10^{-2}$)비트 이상의 버스트 오류가 발생하는 환경에서는 재전송 기법으로 정보의 재전송 처리가 요구되고 371비트 정도는 인터리빙 depth를 200으로 정하는 것이 적합할 것으로 판단된다. FDI을 수행했을 때 전송성

표 4. 블록인터리빙 depth에 따른 비트오류양

인터리빙 depth / 버스트오류 길이(비트오류율)	No 인터리빙	40	80	120	160	200
10(1.0×10^{-3})	6898534 (44%)	14319 (0.095%)	14870 (0.099%)	14319 (0.095%)	13219 (0.088%)	12667 (0.084%)
37(3.9×10^{-3})	7016989 (45%)	57825 (0.38%)	59475 (0.39%)	58927 (0.39%)	57829 (0.38%)	53426 (0.35%)
72(7.5×10^{-3})		112346 (0.74%)	113999 (0.75%)	114548 (0.76%)	115649 (0.77%)	111254 (0.74%)
279(2.91×10^{-2})		6721440 (42.8%)	489345 (3.2%)	447746 (2.9%)	447723 (2.9%)	441673 (2.9%)
371(3.86×10^{-2})			6721440 (42.8%)	3795555 (15.5%)	2636440 (15.5%)	591463 (3.7%)
1454(1.5×10^{-1})				6721440 (42.8%)	6721440 (42.8%)	6721440 (42.8%)
3844(4.0×10^{-1})						
9600(1.0×10^{1})						

능의 개선정도를 인터리빙 유형별로 살펴본 결과는 표 5에서와 같다. 블록, 헬리컬, 랜덤, 확장랜덤 가운데 가장 최적의 전송성능은 충분한 인터리빙 depth를 가진 블록인터리빙에서 얻을 수 있다. 이는 하나의 프레임이 9600비트로 구성될 때 랜덤, 확장랜덤의 경우 랜덤 테이블의 영역이 작으므로 블록인터리빙이 갖는 depth이하의 거리를 유지할 확률이 높기 때문에 비트오류율이 높게 나타난다. 따라서 주기적인 동기식 암호시스템에 인터리빙을 적용할 때 프레임영역으로 적용한다면 가장 적합한 구조가 블록인터리빙 방식에 인터리빙 depth를 충분히 제공할 때 우수한 인터리빙 효과를 얻을 수 있다. 버스트 오류가 279(2.91×10^{-2})비트 이상이 발생하는 환경에서는 블록이 2.9%의 비트오류율에 비해 헬리컬, 랜덤의 경우 20.6%, 28.8%의 비트오류율을 발생하는데 이는 헬리컬과 랜덤의 경우 동기패턴, 세션키로 인해 발생하는 오류가 전체 프레임 손실에 영향을 미치게 되어 프레임 손실이 존재하는 프레임이 있다고 판단되지만 랜덤과 블록은 암호문을 제외한 동기패턴이나 세션키 부분에서 오류가 발생함으로써 전체 프레임에 손실에 영향을 준 프레임이 없다고 판단된다. 371(비트오류율이 3.86×10^{-2})비트 이상의

표 5. 프레임영역방식에서 인터리빙 유형에 따른 비트오류양

인터리빙 유형 / 버스트 오류길이 (비트오류율)	블록 (depth=200)	헬리컬 (200x201)	랜덤 (9600 맵)	확장랜덤 (9600 맵, depth=200)
10(1.0×10^{-3})	12667 (0.084%)	15421 (0.1%)	15972 (0.1%)	15971 (0.1%)
37(3.9×10^{-3})	53426 (0.35%)	59474 (0.39%)	56177 (0.37%)	56170 (0.37%)
72(7.5×10^{-3})	111254 (0.74%)	115647 (0.77%)	110149 (0.73%)	109597 (0.73%)
279(2.91×10^{-2})	441673 (2.9%)	3390015 (20.6%)	4629109 (28.8%)	445268 (2.96%)
371(3.86×10^{-2})	591463 (3.7%)	5519403 (34.8%)	4679674 (29.2%)	4680875 (29.2%)
1454(1.5×10^{-1})	6721440 (42.8%)	6721440 (42.8%)	6721440 (42.8%)	6721440 (42.8%)
3844(4.0×10^{-1})				
9600(1.0×10^{1})				

버스트 오류가 발생하면 헬리컬, 랜덤, 확장랜덤 인터리빙을 수행했을 경우에도 세션키, 동기패턴에서 발생한 오류가 전체 프레임에 영향을 미치게 되어 프레임 손실이 발생함으로써 30%이상의 오류를 발생시킨다. 그러나 이 경우 블록 인터리빙은 세션키, 동기패턴으로 인해 발생하는 프레임 손실이 매우 적다고 판단할 수 있다.

따라서 9600비트의 동기구조를 갖는 동기식 스트림 암호정보를 전송할 때 전송성능 개선을 위한 FDI 구조는 충분한 인터리빙 depth를 갖는 블록 인터리빙 구조가 가장 적합하다. 실험을 통해 살펴볼 때 인터리빙 depth를 200으로 정했을 때 적합한 것을 판단된다.

FDI와 GDI의 비교는 표5와 표6에서와 같다. 짧은구간의 페이딩 영향에서는 FDI의 성능이 GDI의 성능보다 평균 0.01~0.02%정도 좋은 전송성능을 가지고 평균 버스트 오류길이가 279, 371비트가 발생하는 긴 구간 버스트 오류채널 환경에서는 GDI의 전송성능이 FDI 성능보다 우수하게 나타난다. 이는 짧은 페이딩 구간의 경우 인터리빙의 depth를 충분히 제공함으로써 버스트 길이로 인한 영향을 감소시킬 수 있으며 이때 영향을 받는 프레임은 두 개의 프레임에 걸쳐서 영향을 받을 수 있다. 그러나 평균 버스트 길이가 크게 나타날 경우 GDI의 구조가 FDI 구조보다 맵핑테이블 면적이 넓음으로 인해 인터리빙 depth를 증가시킬 수 있다. 또한 블록 인터리빙의 경우 FDI에 비해 인터리빙 depth가 2배 증가하게 된다. 279(비트오류율이 2.91×10^{-2})비트의 버스트 오류가 발생하는 경우 FDI에서 헬리컬, 랜덤구조는 세션키, 동기패턴의 오류가 발생함으로 인해 전체 프레임 손실이 발생함으로써 20.6%, 28.8%의 비트오류율이 발생한데 반해 GDI에서 헬리컬, 랜덤구조는 세션키, 동기패턴의 오류 영향이 적게 나타남으로 인해 프레임 손실률이 매우 낮아 2.78%, 2.77%의 비트오류율을 발생시킨다. 371(비트오류율이 3.86×10^{-2})비트의 버스트 오류가 발생하는 채널환경에서는 헬리컬, 랜덤, 확장랜덤의 경우 34.8%, 29.2%, 42.8%의 비트오류율이 3.77%, 3.76%, 3.56%의 비트오류율로 매우 낮은 프레임 손실률을 갖는다. 즉 GDI의 성능은 FDI의 성능에 비해 전반적으로 나은 전송성능을 가지고 특히 헬리컬, 랜덤, 확장랜덤의 경우 세션키, 동기패턴 오류 영향이 프레임 손실에 영향률이 감소하는 것으로 나타난다. 이 가운데 FDI의 경우 충분한 인터리빙 depth를 가진 블록이 확장랜덤보다 전반적으로 우수한 전송성능을 나타내지만 GDI의 경우 블록보다 확장랜덤에서 우수한 전송성능을 얻을 수 있다.

FDI과 GDI의 성능비교는 블록인터리빙 depth에 따라 비트오류율의 분포특성은 전반적으로 그룹영역 인터리빙 성능이 우수함을 나타내고 전반적으로 평균 버스트 오류가 371비트 이상 발생하는 채널환경에서는 인터리빙 기법만으로 처리한다면 전송성능 열화가 초래된다.

표 6. 그룹영역방식에서 인터리빙 유형에 따른 비트오류양

인터리빙 유형 / 버스트 오류길이 (비트오류율)	블록 (depth=200)	헬리컬 (200x201)	랜덤 (19200 맵)	확장랜덤 (19200 맵, depth=200)
10(1.0×10^{-3})	13220 (0.088%)	16520 (0.11%)	15859 (0.1%)	15530 (0.1%)
37(3.9×10^{-3})	56831 (0.37%)	60460 (0.40%)	57986 (0.38%)	57490 (0.38%)
72(7.5×10^{-3})	114651 (0.76%)	116960 (0.77%)	113167 (0.75%)	113659 (0.75%)
279(2.91×10^{-2})	441023 (2.74%)	447712 (2.78%)	445734 (2.77%)	430918 (2.67%)
371(3.86×10^{-2})	572984 (3.62%)	596392 (3.77%)	594258 (3.76%)	563929 (3.56%)
1454(1.5×10^{-1})	6721440 (42.8%)	6721440 (42.8%)	6721440 (42.8%)	6721440 (42.8%)
3844(4.0×10^{-1})				
9600(1.0×10^{1})				

Ⅵ. 결 론

본 논문은 무선망의 링크암호에 적합한 스트림 암호시스템을 설계하였고 설계된 스트림 암호를 통해 암호문을 전송할 때 한 주기 동안 발생하는 동기패턴, 세션키, 암호문 정보를 인터리빙 기법을 적용하여 전송함으로써 버스트 오류로부터 암호문을 보호하고 전송성능을 개선하여 robust한 암호통신을 가능하도록 하였다. 제안된 인터리빙 기법을 적용하기 위해 라이시안 페이딩 채널을 고려하여 페이딩 지수에 따른 평균적으로 발생하는 버스트 오류를 이론적으로 유도하였으며 주기적인 동기식 스트림 암호시스템에 인터리빙을 적용할 때 한 주기의 동기패턴, 세션키, 암호문으로 구성된 프레임영역 인터리빙과 다수개의 프레임으로 구성된 그룹영역 인터리빙으로 구분하여 인터리빙을 수행함으로써 인터리빙 기법 도입을 통하여 개선된 결과를 보였다.

동기식 스트림 암호시스템의 동기구조는 2400, 4800, 9600, 19200 비트를 갖도록 설계하였고 이 동기구조를 갖는 암호정보에 블록인터리빙을 수행하여 동기구조에 적합한 인터리빙 구조를 제안하였으며 인터리빙 유무를 통하여 인터리빙의 효율성을 제고하였다. 주기적인 동기식 암호시스템에 인터리빙을 적용할 때 FDI(Frame Domain Interleaving)은 충분한 인터리빙 depth를 가지는 블록인터리빙 방식에서 우수한 인터리빙 효과를 얻을 수 있었다. 블록 인터리빙이나 확장랜덤 인터리빙은 상대적으로 인터리빙 depth를 충분히 제공했을 때 헬리컬 인터리빙이 20.6%, 랜덤 인터리빙이 28.8%의 비트오류율을 발생하는 것에 비해 블록 인터리빙에서 2.7%, 확장랜덤 인터리빙에서 2.76% 비트오류율을 나타냄으로써 블록 인터리빙과 확장랜덤 인터리빙에서 헬리컬 인터리빙과 랜덤 인터리빙보다 나은 전송성능을 얻을 수 있다. GDI는 맵핑테이블이 증가함에 따라 블록 인터리빙보다 확장랜덤 인터리빙에서 우수한 전송성능을 얻을 수 있었다. 인터리빙 depth를 증가함에 따라 비트오류양을 감소시킬 수 있고 그룹영역 인터리빙은 버스트 오류 371비트의 환경에서 인터리빙 depth를 200으로 할 때 FDI에서 얻은 591463 비트(3.7%)의 비트오류양에 비해 572984비트(3.6%)의 비트오류양을 발생함으로써 더 나은 성능을 얻었다.

참 고 문 헌

[1] Van Til borg, H. C. A., *An Introduction to Cryptology*, KLUWER Academic Pub. Boston, etc.,1988.

[2] H. J. Beker and F. C. Piper, *Cipher Systems : The Protection of Communications*, Northwood Books, London, 1982.

[3] P. R. Geffe, "How to Protect Data with Ciphers that are really hard to break," *Electronics*, pp. 99-101, Jan. 1973.

[4] 안치훈, 김남, 박성균, "마이크로셀룰라 이동 무선시스템에서 Outage확률을 이용한 라이시안페이딩과 로그노말 새도우잉 영향에 관한 분석," *전자파학회논문지 제9권 제1호*, pp. 60-71, 1998년 12월.

[5] W. Meier and O. Staffelbach, "Correlation Properties of Combiners with Memory in stream ciphers," *Journal of Cryptology*, vol. 5, pp. 67-86, 1992.

[6] E. Dawson, "Cryptoanalysis of Summation Generator," *Advances in Cryptology AUSCRYPT '92, Lecture Notes in Computer Science*, Springer-verlag, pp. 209-215, 1993.

[7] M. Simon, Spread Spectrum Communications Handbook, McGraw-Hill, 1994.

[8] M. Y. Lee, *Error Correcting Coding Theory*, McGraw-Hill, 1989.

[9] J. L. Massey, "Shift Register Synthesis and BCH Decoding," *IEEE Trans. on Information Theory*, vol. IT-15, no. 1, pp. 122-127, Jan. 1969.

[10] T. Siegenthaler, "Correlation Immunity of Nonlinear Combining Function for Cryptographic Applications," *IEEE Trans. on Information Theory*, vol. IF-30, no. 5, pp. 776-780, Sept., 1984.

[11] R. A. Rueppel, "Correlation Immunity and the Summation Generator," Advances in cryptology, Proceedings of CRYPTO'85, pp. 260-272, 1985.

[12] R. C. Dixon, *Spread Spectrum Systems*, New York Wiley, 1976.

[13] M. Kimberley, "Comparison of two statistical

tests for keystream sequences," *Electronics Letters*, vol. 23, no. 8, pp. 365-366, Apr. 1987.

[14] A. Franchi & R. A. Harris, "On the Error Burst Properties of the "Standard" K=7, Rate-1/2 Convolutional Code with Soft-Decision Viterbi Decoding," *Submitted to European Transactions on Telecommunications.*

[15] CCITT "Blue Book", vol. 1, Fascicle I.3, "Terms and Definitions," *Rec. M.60* n. 34, *Rec. Q. 9* n.0222.

[16] D. E. Reed & W. A. Draheim, "Viterbi Decoding Burst Errors and their Effects on Digital Communication Performance," *MILCOM '85*, Boston, Mass., Oct. 1985.

[17] 최봉대, *Randomness 특성분석에 관한 연구*, 데이터보호의 기반기술연구(I), 전자통신연구소, 1990년.

[18] E. Costamagna and A. Schirru, "CHANNEL ERROR MODELS DERIVED FROM CHAOS EQUATIONS," *in Proc. 1994 IEEE Int. Symp. on Systems Man. and Cybernetics*, San Antonio. TX, 1994, vol. I, pp. 577-581.

[19] P. M. Crespo, R. M. Pelz, J. P. Cosmas, and J. G. Garcia-Frias, "Results of channel error profiles for DECT," *IEEE Trans. Commun.*, vol. 44, Aug. 1996.

[20] E. Costamagna, L. Favalli, P. Gamba, and P. Savazzi, "Block-Error Probilities for Mobile Radio Channels Derived from Chaos Equations," *IEEE Commun. Letters*, vol. 3, no. 3, march 1999.

홍 진 근(Jin-keun Hong) 정회원

2000년 2월 : 경북대학교 전자공학과 박사

2000년 10월 현재 : ETRI 부설 국가보안기술연구소 선임연구원

논문지 24권 6호 참조

황 찬 식(Chank-sik Hwang) 정회원

논문지 24권 6호 참조

윤 장 홍(Jang-hong Yoon) 정회원

1987년 2월 : 경북대학교 전자공학과 석사

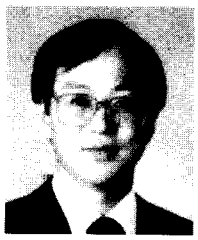
1998년 2월 : 경북대학교 전자공학과 박사

1987년 2월~2000년 1월 : ADD 개발팀장

2000년 2월~현재 : ETRI 부설 국가보안기술연구소 응용3팀장

<주관심 분야> 컴퓨터통신, 암호통신

강 건 우(Ken-woo Kang) 정회원



1973년 2월 : 연세대학교 전자공학과 학사

1987년 2월 : 한국과학기술원 전기전자공학과 석사

1996년 2월 : 한국과학기술원 전기전자공학과 박사

1973년 1월~1981년 11월 : 한국과학기술연구소

1981년 12월~2000년 1월 : ADD 개발부장

2000년 2월~현재 : ETRI 부설 국가보안기술연구소 응용기술개발부장

<주관심 분야> OFDM, 암호통신