

# 호핑 방식을 이용한 블록 인터리버의 성능 분석

공 형 윤<sup>†</sup> · 우 일 승<sup>††</sup>

## 요 약

본 논문에서는 호핑 방식을 사용한 블록 인터리버에 대해 연구하였다. 인터리버의 성능은 인접 데이터간 최소 거리에 의해 결정되어지므로, 인접 데이터간 최소 거리를 증가시키는 방법으로 인터리버의 성능을 향상 시켰다. 기존의 블록 인터리버는 메모리의 입/출력이 행과 열에 대해 차례대로 이루어지는 반면, 본 연구에서 제안된 방식은 행과 열을 건너뛰면서 입/출력이 이루어지므로 인접 데이터간 최소거리와 평균거리를 증가시킨다. 제안하는 인터리버의 또 다른 장점으로서는 다양한 호핑 방식의 사용이 가능하다는 것이다. 제안된 인터리버의 성능 분석과 비교를 위해 가우시안 잡음 환경 하에서 컴퓨터 모의 실험을 하였다.

## Performance Analysis of Block Interleaver by Using Hopping Method

Hyung-Yun Kong<sup>†</sup> · Il-Seung Woo<sup>††</sup>

### ABSTRACT

In this paper, we studied the block interleaver that used hopping method. In general, the performance of interleaver is determined by minimum distance between neighbor data, thus we could improve the performance of interleaver by increasing the distance of the nearest data. The input/output of the conventional block interleaver is achieved by the order of column and row, but our proposed method is achieved by hopping the column and row that increase the minimum distance and average distance between the nearest data. Another advantage of our proposed interleaver can have various hopping methods. To verify and compare the performance of our proposed interleaving method, the computer simulations have been performed under gaussian environments.

**키워드 :** 호핑방식(Hopping Method), 인터리버(Interleaver)

### 1. 서 론

IMT-2000(International Mobile Telecommunications in the year 2000) 시스템은 3세대 이동 통신 시스템으로서 다양한 서비스(음성, 화상, 데이터, 멀티미디어...)의 제공을 그 목표로 하고 있으며, 이러한 종류의 서비스 제공을 위해서는 데이터 율(data rate)이 2Mb/s 정도까지 증가하여야 한다[1]. 하지만, 무선 통신 채널의 비이상적(non-ideal) 특성 때문에 데이터 율의 증가에 제약을 받게 되는데, 무선 통신 채널의 특성으로는 전송 신호의 대역폭이 제한(Band-limit)되어 있다는 것과 여러 가지 요인들로 인해 전송 신호의 변형이 발생하게 된다는 것이다. 전송 신호의 변형 요인은 잡음(noise), 다중 경로(multi-path), 심벌간 간섭(ISI: Inter-Symbol Interference)등으로 인해 전송 신호는 변형 뿐만 아니라 전송 데이터의 오차(error)까지도 유발하게 된다. 이렇게 발생한 오차에 대응하기 위해 대부분의 무선 통

신 시스템에서는 채널 부호화(Channel Coding)를 사용하고 있으며[2, 3], 현재 많이 사용하고 있는 채널 부호화로는 콘볼루션 부호와 터보 부호가 있다. 콘볼루션 부호와 터보 부호는 현재의 데이터를 부호화 하는데 있어 과거의 데이터의 값의 영향을 받게 되는데, 이러한 특성으로 인해 콘볼루션 부호와 터보 부호는 연접 에러에 대해 매우 약한 특성을 가지며, 따라서 연접 에러를 랜덤 에러로 바꾸어주는 인터리버(Interleaver)의 사용을 필요로 하게 된다. 인터리버는 적당한 알고리즘을 통하여 데이터의 전송 순서를 조작하게 되며, 수신단에서는 송신단과 마찬가지로 적당한 알고리즘을 통하여 원래의 순서대로 복원을 하게 된다[4]. 본 논문에서는 블록 인터리버의 성능을 개선시킨 호핑(hopping) 방식의 블록 인터리버에 대해 연구하였으며, 제안하는 인터리버의 성능 분석을 위한 모의 실험 결과로부터 호핑 방식의 블록 인터리버가 우수함을 알 수 있었다. 모의 실험 결과, 신호 대 잡음비(Signal to Noise Ratio: SNR)가 7dB 이상에서 호핑 방식을 이용한 블록 인터리버의 성능이 우수하게 측정되었으며, 비트 오차율(Bit Error

\* 본 논문은 정보통신부 우수시범학교 결과물입니다.

† 정 회 원 : 울산대학교 전자공학과 교수

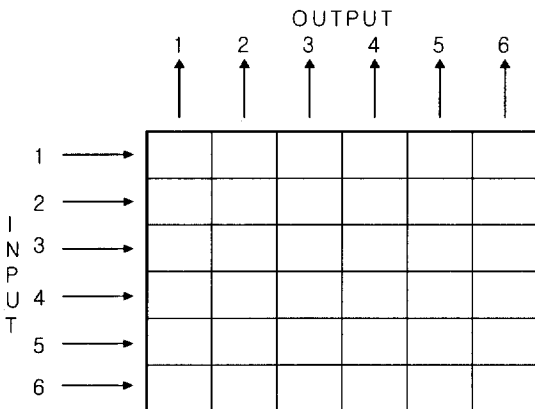
†† 준 회 원 : 이스텔 시스템즈 무선통신 연구4팀

논문접수 : 2000년 11월 27일, 심사완료 : 2001년 5월 7일

Rate : BER)이 일 때 1dB 정도의 인터리빙 이득(Interleaving gain)을 얻을 수 있었다. 본 논문의 구성을 살펴보면 다음과 같다. 제2장에서는 콘볼루션 부호와 블록 인터리버에 대해서 간단히 알아보고, 제3장에서는 호핑 방식의 블록 인터리버의 구조와 동작에 대해 설명하였다. 제4장에서는 호핑 방식의 성능 분석을 위하여 모의 실험을 하였으며, 제5장에서 결론을 맺도록 하겠다.

## 2. 채널 부호화와 인터리버

1948년 C. E. Shannon은 정보에 확률 개념을 도입하여 정보를 비트로 표현하고 잡음 채널과 무잡음 채널에서 오차 없이 정보를 전송하기 위한 이론을 제창하였다. 그 후 잡음이 있는 환경에서 오차 제어를 위한 채널 부호화 및 복호화에 관한 연구가 수행되어 왔고, 오차 제어를 위한 채널 부호화의 사용이 통신 시스템의 설계에 필수적인 요소가 되었다[5, 6]. 현재 많이 사용되는 채널 부호화로는 콘볼루션 부호와 터보 부호가 있는데, 이러한 채널 부호화 방법들은 현재의 데이터를 부호화 하는데 있어 과거 데이터의 영향을 받게 된다. 이러한 이유로 연접 에러(burst error)에 약하다는 특성을 가지게 되는데 이는 연접 에러가 발생하면 수신단에서 복호화에 어려움이 따르기 때문이다. 따라서, 채널 부호화의 성능 향상을 위해서 이러한 연접 에러를 랜덤 에러로 바꾸어주는 인터리버를 사용하고 있으며[7], 다양한 인터리버 중에서 가장 간단한 구조를 가지는 것이 블록 인터리버로 전송 데이터를 일정한 블록 단위로 나누어서 인터리빙을 하게 된다. 행의 크기가  $x$ , 열의 크기가  $y$  인 블록, 즉 블록의 크기는  $x \times y$ 가 되고, 한번에 인터리빙 되는 데이터의 수와 같다. 여기서,  $x$ 와  $y$ 의 선택에 있어서는 연접 에러의 길이가  $x$  또는  $y$ 보다 작다는 가정 하에서  $x$ 와  $y$ 를 선택하여야 한다. (그림 1)에서는 행( $x$ )과 열( $y$ )의 크기가 모두 6인 간단한 블록 인터리버를 나타내었다[8].



(그림 1) 블록 인터리버의 예

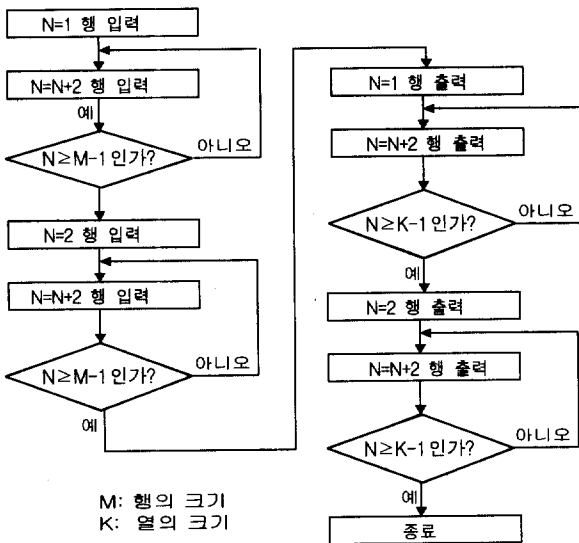
(그림 1)에서 보는 것처럼, 메모리 블록에의 입력은 행으

로 이루어지고 출력은 열로 하게 된다. 이런 방식으로 적당한 알고리즘에 의해 데이터의 전송 순서를 조작하면 전송도중 어떤 잡음이나, 페이딩, 심벌간 간섭 등과 같은 영향으로 연접 에러가 발생하더라도 그 역과정, 즉 디인터리빙을 하게 되면 연접 에러가 랜덤 에러로 바뀌게 된다. 블록 인터리버의 성능을 향상시키기 위한 방법의 한 예로 3차원 방식이 있는데, 이는 기존의 인터리버가  $x$ 축과  $y$ 축으로 구성되는 반면 3차원 방식은  $x, y$ 축 이외에  $z$ 축의 개념까지 추가한 것이다. 3차원 방식의 동작을 간단히 살펴보면 입력은  $x, y, z$ 순으로 이루어지고 출력은  $z, y, x$ 의 순을 따름으로서 인접 데이터간 거리를 증가시켜 인터리버의 성능을 향상시킨 방법이다[9]. 하지만 본 논문에서 제안하는 방식은 2차원의 평면에서 입력과 출력이 이루어지는 비교적 간단한 구조를 가지면서도 성능이 우수한 블록 인터리버이며, 다음장에서 상세히 설명하도록 하겠다.

## 3. Hopping 방식의 블록 인터리버 설계

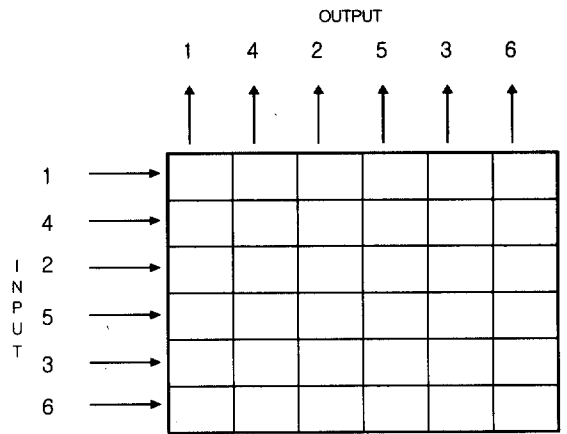
이번 장에서는 호핑 방식의 블록 인터리버의 구성과 동작에 대해서 알아보도록 하겠다. 인터리버란 전송하고자 하는 데이터를 적당한 알고리즘에 의해 전송 순서를 바꿈으로써 메모리를 가지는 전송 채널에 의한 연접 오류를 최소화시키는 방식이다. 인터리버의 성능은 인접 데이터간 최소 거리와 평균 거리에 의해 결정되는데, 제안하는 블록 인터리버는 기존의 블록 인터리버보다 인접 데이터간 최소 거리와 평균 거리를 증가시킨다는 장점을 가지고 있다. 호핑 방식의 블록 인터리버가 인접 데이터간 최소 거리와 평균 거리를 증가시키는 이유를 설명하면 다음과 같다. 기존의 블록 인터리버는 메모리 블록에의 입력과 출력이 행과 열에 대해 차례대로 진행되는데 반해 호핑 방식의 인터리버는 행과 열을 건너뛰면서 동작을 하게 된다. 이를 자세히 설명하면 메모리 블록에 입력되어 질 때, 1행씩 건너뛰는 방식으로 이루어지고, 출력 역시 1열씩 건너뛰는 방식으로 이루어진다. 따라서, 입력시 인접 데이터간 거리를 증가시키고, 마찬가지로 출력시에도 인접 데이터간 거리를 증가시켜 준다. 즉, 2번에 걸친 거리의 증가로 기존의 블록 인터리버보다 더 큰 인접 데이터간 최소 거리와 평균 거리를 가지게 된다. 우선, 호핑 방식을 사용하는 블록 인터리버중 가장 기본적인 구조를 가지는 mode-2에 대해서 알아보면 다음과 같다. Mode-2는 메모리 블록에 입력과 출력이 2칸마다 이루어지게 되는데, 즉 첫 번째 행의 입력이 끝나면 다음에는 세 번째 행, 다섯 번째 행의 순으로 홀수 행에 대해 먼저 입력이 이루어지게 되고 다음으로 짝수 행에 차례대로 입력이 이루어진다. 출력시에도 입력의 동작과 마찬가지로 홀수 행의 데이터를 먼저 출력하고 난 후 짝수 행의 데이터를 출력하게 된다. 다음으로 mode-3에 대해서 알아

보기로 하자. Mode-2는 2행, 2열 단위로 입력과 출력이 이루어지는데 반해, mode-3는 메모리 블록에의 입력과 출력이 3행, 3열마다 이루어지고 전체적인 동작은 mode-2와 비슷하다. 첫 번째 행의 입력이 끝나면 다음에는 네 번째 행, 일곱 번째 행의 순으로 입력이 이루어지고, 출력 역시 첫 번째, 네 번째, 일곱 번째 행의 순으로 이루어지게 된다. 즉, mode-x로 동작하는 호핑 방식의 블록 인터리버는 메모리 블록에의 입력과 출력이 x행, x열 단위로 이루어진다는 것을 알 수 있다. 호핑 방식의 블록 인터리버라고 이름이 붙여진 이유는 이 때문이다. 이러한 호핑 방식의 블록 인터리버의 사용을 위해서는 기존의 블록 인터리버 보다 복잡한 알고리즘을 필요로 하며, 실제 하드웨어의 구현에 있어 각각의 행과 열에 대해서 서로 다른 지연 시간을 할당하여야 하므로 어려움이 따르게 된다. 하지만, 블록 인터리버보다 복잡한 구조를 가지는 콘볼루션 인터리버, 랜덤 인터리버등과 비교하면 간단하면서도 기존의 블록 인터리버에 비해 인접 데이터간 최소거리가 더 크다는 장점을 가지게 된다. (그림 2)에서는 본 연구에서 제안하는 호핑 방식의 블록 인터리버(mode-2)에 대한 순서도를 나타내었다. (그림 2)에서는 mode-2의 동작에 대한 순서도만 나타내었는데, 호핑 방식이 mode-3 또는 mode-4등으로 바뀌게 되면 입/출력때의 초기값  $N=1, 2$ 뿐만 아니라  $N=1, 2, 3$  또는  $N=1, 2, 3, 4$ 로 설정되고, 증가값  $N=N+2$ 가 아니고  $N=N+3$  또는  $N=N+4$  등으로 바뀌어야 한다. 즉, mode의 변화가 있을 때마다 순서도에 변화를 가져오게 되지만 단순한 과정을 거쳐 각각의 mode에 적당한 순서도의 작성이 가능하게 된다.

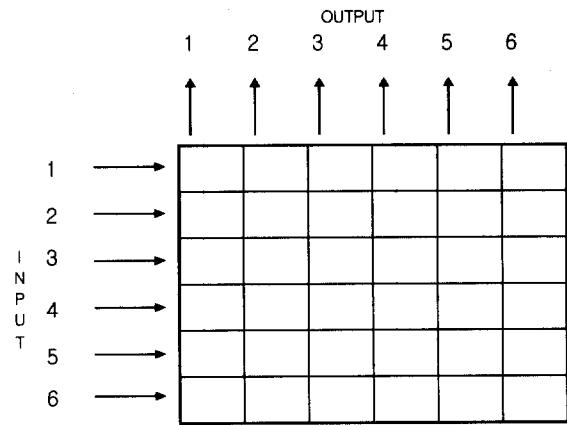


(그림 2) 호핑 방식을 사용한 블록 인터리버의 순서도

(그림 3)은 호핑 방식을 이용한 블록 인터리버의 동작을 나타낸 것으로 (그림3-a)는 제안하는 방식에서의 mode-2를 나타내었고, (그림 3-b)는 mode-3를 나타낸 것이다.



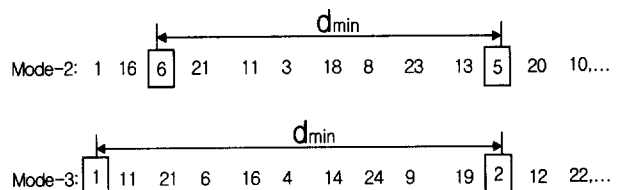
(a) mode-2



(b) mode-3

(그림 3) Hopping 방식의 블록 인터리버

여기서, mode-2와 mode-3는 레지스터 블록에 입/출력되는 호핑의 범위에 의해 결정된다. Mode-2는 2칸마다 입/출력이 이루어지고, mode-3는 3칸마다 입/출력이 이루어지는 블록 인터리버를 의미한다. 블록의 크기에 따라 언급된 mode-2와 mode-3 이외에도 필요에 따라 다양한 mode의 호핑 알고리즘을 선택하여 인터리버로 사용이 가능하다. (그림 4)에서는 블록의 크기가 5x5인 경우의 mode-2와 mode-3에 대한 출력 시퀀스를 비교해 보았다

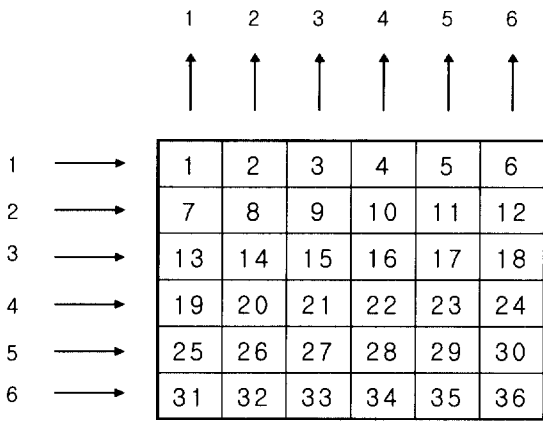


(그림 4) Mode-2와 mode-3에 대한 출력 시퀀스 비교

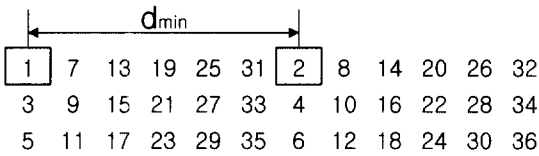
(그림 4)로부터 mode-2와 mode-3에 대한 출력 시퀀스가 서로 다르고, 인접 데이터간 최소거리도 다르다는 것을 알 수 있다. 메모리 블록의 크기가 같다고 하더라도 호핑

mode에 따라 서로 다른 출력 시퀀스를 가지며 인접 데이터간 최소 거리도 서로 다르게 되고, 따라서 서로 다른 성능을 가지게 될 것이다. 다음으로 호핑 방식을 사용하는 블록 인터리버의 성능을 기존의 블록 인터리버와 비교하기 위해 mode-2인 경우에 대해 간단한 예를 들어보기로 하겠다. (그림 5)는 인터리버의 사이즈가 6×6인 블록 인터리버의 입/출력 데이터를 나타낸 것이고, (그림 6)은 호핑 방식에서 mode-2인 블록 인터리버의 입/출력 데이터를 나타낸 것이다.

3.1 블록 인터리버



(a) 블록 인터리버



(b) 기존의 블록 인터리버에서 데이터의 출력 순서

(그림 5) 블록 인터리버

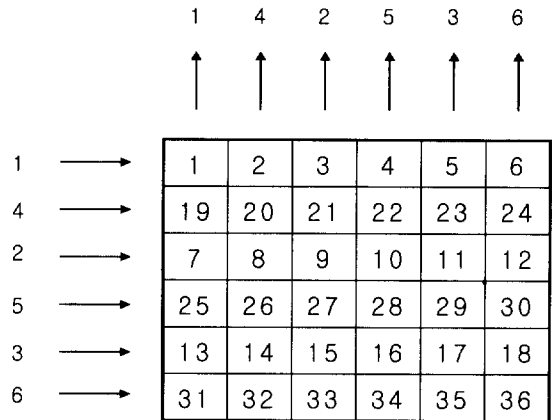
(그림 5)에 메모리 블록의 크기가 6×6인 기존의 블록 인터리버의 구조와 입/출력 시퀀스를 나타내었다. (그림 5)로부터 크기가 6×6인 기존의 블록 인터리버에서 데이터의 입/출력 순서를 살펴보면 다음과 같다. 입력의 1, 2, 3, 4, 5, 6, ...의 순서로 메모리에 입력이 되고, 출력의 순서는 1, 7, 13, 19, 25, 2, ...가 된다. 여기서, 전송 데이터의 인접 데이터간 최소 거리와 평균거리를 구해 보면 다음과 같다. 인접 데이터간 최소 거리는 (그림 5-b)에서 1과 2 사이의 거리인 6이 되며, 평균 거리는 인접 데이터간 거리의 평균값을 의미한다.

- 인접 데이터간 최소 거리( $d_{min}$ ) : 6
- 평균 거리( $d_{average}$ ) : 9.83

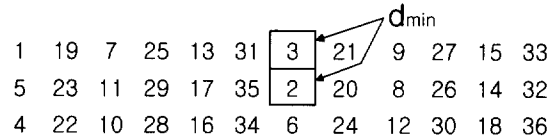
3.2 호핑 방식의 블록 인터리버

(그림 5)에서는 메모리 블록의 크기가 6×6인 호핑 방식

의 블록 인터리버에 대한 입/출력 시퀀스를 나타내었다.



(a) 호핑 방식의 블록 인터리버



(b) 호핑 방식의 블록 인터리버에서 데이터의 출력 순서

(그림 6) 호핑 방식의 블록 인터리버

(그림 6)으로부터 크기가 6×6인 호핑 방식의 블록 인터리버(mode-2)에서 인접 데이터간 최소 거리와 평균거리를 구해 보면 다음과 같이 구해진다.

- 인접 데이터간 최소 거리( $d_{min}$ ) : 12
- 평균 거리( $d_{average}$ ) : 18

위에서 기존의 인터리버와 호핑 방식의 블록 인터리버에서의 인접 데이터간 최소 거리와 평균 거리에 대한 예를 살펴보았으며, 예로부터 제안하는 방식의 인터리버가 인접 데이터간 최소 거리와 평균 거리에 있어 기존의 시스템에 비해 훨씬 큰 값을 가짐을 알 수 있었다. <표 1>에는 mode-2인 호핑 방식의 블록 인터리버와 기존의 블록 인터리버에서의 인접 데이터간 최소 거리와 평균 거리를 블록의 크기에 따라 정리하였다. <표 1>을 살펴보면 블록의 크기가 작은 경우(4×4)에는 기존의 블록 인터리버와 호핑 방식의 인터리버에서 인접 데이터간 최소 거리와 평균 거리가 별 차이를 보이지 않지만, 블록 사이즈가 증가할수록 인접 데이터간 최소 거리와 평균 거리의 차이도 더 커짐을 알 수 있다. 그리고, <표 1>에서는 기존의 인터리빙 방식에서 인접 데이터간 최소 거리는 블록의 행이나 열의 크기와 일치하는데 반해 제안하는 방식(호핑 방식)의 인터리빙에서는 행이나 열의 크기와 일치하지 않고, 행이나 열의 크기 이상임을 보여주고 있다. 그리고, 블록의 크기가 증가할수록 기존의 방식과 제안하는 방식의 거리의 차이는 점점 증가하고

있음을 보여준다. 즉, 블록의 크기가 증가할수록 기존의 방식과 제안하는 방식에서의 인접 데이터간 최소거리와 평균 거리의 차이는 점점 증가하게 될 것이고, 따라서 성능의 차이도 점점 증가할 것이라는 예측이 가능하다.

〈표1〉 블록의 사이즈에 따른 인접 데이터간 최소거리와 평균 거리

Block Size	System	d_min	d_average
4×4	Block	4	6
	Hopping	4	8.5
5×5	Block	5	8
	Hopping	8	12
6×6	Block	6	9.83
	Hopping	12	18
10×10	Block	10	17.9
	Hopping	50	62.8

본 연구에서 제안된 호핑 방식을 이용한 블록 인터리버의 특성을 요약하여 정리하면 다음과 같다.

- 콘볼루션 인터리버나 랜덤 인터리버에 비해 비교적 간단한 구조를 가지면서도 기존의 블록 인터리버보다 큰 인접 데이터간 최소 거리와 평균 거리를 가진다.
- 기본 호핑 방식인 mode-2를 바탕으로 하여 mode-3, mode-4 등과 같은 다양한 호핑 알고리즘의 적용이 가능하다.
- 블록의 크기가 증가할수록 기존의 블록 인터리버와 비교하여 인접 데이터간 최소거리와 평균 거리의 차이가 증가한다.
- 제안하는 방식에서는 메모리 블록에서 행과 열에 대해서도 다른 전송 지연 시간을 두어야 하므로 기존의 블록 인터리버에 비해 복잡한 알고리즘을 필요로 하며 실제 하드웨어의 구현시 어려움이 따르게 된다.

4. 모의 실험

이번 장에서는 제안하는 인터리버의 성능 분석을 위한 모의 실험을 하였으며, 모의 실험은 다음과 같은 환경에서 이루어졌다. 채널 부호화는 부호화 율(code rate)이 1/2인 콘볼루션 부호를 사용하였으며, 채널 환경으로는 가우시안 잡음 환경을 설정하였다. 사용된 블록의 크기는 IMT-2000 시스템의 무선 접속 규격 중 한 가지인 W-CDMA의 음성 채널에서 사용하고자 하는 블록 인터리버의 규격을 따랐으며[10], 호핑 알고리즘은 mode-2와 mode-3를 사용하였다. (그림 6)은 W-CDMA의 음성 채널 순방향 링크(Forward Link)에 대한 모의 실험의 결과로, (그림 6-a)는 블록의 크기가 29×10, (그림 6-b)는 13×10에 대한 결과이며, 호핑 방식의 블록 인터리버가 우수한 성능을 나타내었다. (그림 6)은 W-CDMA의 역방향 링크(Reverse Link)에 대한 모의 실험의 결과로, (그림 7-a)는 블록의 크기가 32×10, (그림

(그림 6) 순방향 링크에서의 성능 비교

7-b)는 16×10에 대한 결과이며 이 역시 호핑 방식의 블록 인터리버가 우수한 성능을 나타내었다. 전체적으로 기존의 블록 인터리버와 호핑 방식의 블록 인터리버의 성능을 비교해 보면, 호핑 방식을 사용한 블록 인터리버의 성능이 신호 대 잡음비가 7dB를 넘어 서면서 우수한 결과를 나타내었고, 비트 오차율이 에서 1dB 정도의 인터리빙 이득을 얻을 수 있었다. 그리고, 본 논문에서 제안하는 방식인 호핑 알고리즘에서 mode-2와 mode-3의 성능을 비교하면 비슷한 성능을 보였지만 신호 대 잡음비가 8dB 이상에서는 mode-3가 mode-2보다 약간 우수한 성능을 보였다. 이는 앞에서 언급되었지만 호핑의 mode에 따라서 인접 데이터간 최소 거리가 달라지게 되는데, mode-3의 인접 데이터간 최소 거리가 mode-2보다 더 크기 때문이다.

5. 결 론

본 논문에서는 블록 인터리버의 성능을 개선시킨 호핑

시키는 장점을 가지고 있으며, 기본 호핑 방식인 mode-2 이외에도 mode-3, mode-4 등과 같은 다양한 호핑 알고리즘의 적용이 가능하다. 따라서, 제안하는 방식인 호핑 방식의 블록 인터리버를 IMT-2000의 음성 채널에 인터리버로 적용한다면 보다 우수한 채널 부호화의 효과를 얻게 될 것이며, 블록 인터리버가 사용되는 많은 다른 시스템에 적용이 가능할 것이다. 향후 과제로는 최적의 호핑 알고리즘을 이용한 블록 인터리버의 구현 및 호핑 알고리즘을 적용한 다양한 인터리버의 설계, 즉 콘볼루션 인터리버, 랜덤 인터리버등의 설계이다.

### 참 고 문 헌

- [1] Raj Pandya, Davide Grillo, Edgar Lycksell, Philippe Mieybegue, Hideo Okinaka, Masami Yabusaki : 'IMT-2000 Standards : Network Aspects', IEEE Personal Communications, Volume : 44 , Aug. 1997, pp.20-29.
- [2] John G. Proakis : 'DIGITAL COMMUNICATIONS', Third Ed. McGraw-Hill, 1995.
- [3] 이문호 저 : '실용 정보이론', 1판, 복두출판사, 1998.
- [4] Raymond Steele : 'MOBILE RADIO COMMUNICATIONS,' IEEE Press, 1994.
- [5] 이문호 저, "실용 디지털 통신", 1판, 도서출판 영일, 1997.
- [6] 강창언 저, "디지털 통신 시스템", 2판, 복두출판사, 1998.
- [7] Danny T. Chi, "A NEW HELICAL INTERLEAVER", MILCOM '92, Vol.2, pp.799-804.
- [8] STEPHENG. WILSON, "DIGITAL MODULATION AND CODING," Prentice Hall, 1996.
- [9] 공형운, 이창희, "새로운 블록 인터리버를 이용한 터보코드의 성능분석", 2000년도 한국통신학회 하계학술 발표회, pp.645- 648.
- [10] Atsushi Fukasawa, Takuro Sato, Yumi Takizawa, Reed E. Fisher, Toshio Kato and Manabu Kawabe, "Wideband CDMA System," Spread Spectrum Techniques and Applications Proceedings, 1996., IEEE 4th International Symposium on Volume.1 pp.244-248, Vol.1.

(그림 7) 역방향 링크에서의 성능 비교

방식의 블록 인터리버에 대해서 연구하였다. 현재 무선 통신 시스템에서 많이 사용하고 있는 콘볼루션 부호나 터보 부호의 경우 현재의 부호화 값은 과거의 데이터 값과 상관성이 있기 때문에 연결 에러에 대해 약한 성능을 보인다. 따라서, 데이터의 전송 순서를 일정한 알고리즘에 의해 바꾸어 주는 인터리버의 사용을 필요로 하며, 본 논문에서 제안하는 방식인 호핑 방식의 블록 인터리버는 기존의 블록 인터리버보다 인접 데이터간 최소거리와 평균거리를 증가시킬 수 있는 방법으로, 개선된 알고리즘에 의해 데이터의 전송 효율을 증가시킨다. 제안된 인터리버의 성능 분석을 위해 가우시안 잡음 환경 하에서 모의 실험을 하였는데, 모의 실험 결과 호핑 방식을 이용한 인터리버의 성능이 신호대 잡음비가 7dB 이상에서 우수한 성능을 보였으며, 비트오차율이 에서 1dB정도의 인터리빙 이득을 얻을 수 있었다. 호핑 방식을 사용하는 블록 인터리버의 사용을 위해서는 기존의 블록 인터리버보다 복잡한 알고리즘을 필요로 하지만 콘볼루션 인터리버 또는 랜덤 인터리버에 비하면 간단한 구조를 가지면서 인접 데이터간 최소 거리를 증가

### 공 형 운

e-mail : hkong@uoulsan.ac.kr  
 1989년 New York Institute of Technolgy(학사)  
 1991년 Polytechnic University(석사)  
 1996년 Polytechnic University(박사)  
 1996년~1998년 LG 멀티미디어연구실 PCS 팀장 및 LG 그룹 회장실 전략사업개발단 과장  
 1998년~현재 울산대학교 전자공학과 조교수  
 관심분야 : 무선통신 (IMT-2000, LMDS, 위성시스템), 전송기술(등화기, 채널코딩, 변조방식)

### 우 일 승

e-mail : iswoo@eastelsystems.com  
 1999년 울산대학교 전자공학과 졸업(학사)  
 2001년 울산대학교 전자공학과 졸업(석사)  
 현재 이스텔 시스템즈 무선통신 연구4팀 근무  
 관심분야 : 무선통신, 변복조, B-WLL