

RS Code의 오류 위치 정보를 이용하는 리스트 구 복호기

*박선호 **이혁 ***심병호

고려대학교

*medital@korea.ac.kr

List Sphere Decoding using error location information of RS code

*Park, Sunho **Lee, Hyuk ***Shim, Byonghyo

Korea University

요약

본 논문은 Shannon의 정리에 따른 채널 용량에 근접한 성능을 보이는 것으로 알려진 터보 복호기 기반의 반복적인 검출과 복호화(Iterative Detection and Decoding) 기법에서 반복적인 복호화를 수행할 시에 제외되었던 리스트 구 복호기(List Sphere Decoder)에서 사전 정보(prior information)를 이용할 수 있도록 하여 수정된 IDD 기법을 제안하였다.

기존의 기법에서는 사후확률(A posteriori probability)을 계산하기 위하여 리스트 구 복호기를 사용하였으나 반복적인 복호화 수행 시에는 사전 정보를 이용하지 않는 특성으로 인하여 제외된다. 만약 잡음(noise) 등의 이유로 검출된 심볼 벡터 목록이 원래의 것과 매우 다른 경우라도 재 검출을 하지 않기 때문에 반복적인 복호화를 수행하더라도 원래의 정보에 근접하기 어렵게 된다. 본 논문에서는 이러한 기존의 기법에서 리스트 구 복호기를 터보 복호기의 Log Likelihood Ratio (LLR) 값을 사전 정보로 이용할 수 있도록 수정된 리스트 구 복호기를 제안하였다. 수정된 리스트 복호기는 반복적인 복호화를 수행 시 이전의 복호화에서 얻은 정보를 이용하여 새로이 검출된 심볼 벡터 목록을 제공하게 된다.

실제의 통신환경과 유사한 모델의 실험을 통해 수정된 IDD 기법이 기존의 IDD로 구성되는 내부 피드백에 RS 복호기 기반의 외부 피드백으로 구성된 형태로 피드백 회수가 증가할수록 기존의 IDD에 비해 성능이 개선됨을 확인하였다.

1. 서론

최근 무선 통신 서비스에 대한 사용자의 요구가 증가되면서 다양한 멀티미디어 전송을 위한 대용량의 데이터를 고속으로 전송할 수 있는 기술들이 각광받고 있다. 특히 다중 입력 다중 출력 시스템에서 터보 부호를 이용한 반복적인 검출과 복호화(Iterative Detection and Decoding, IDD) 기법은 Shannon의 정리에 따른 채널 용량(channel capacity)에 가까운 성능을 보이는 기법으로 최근 많은 주목을 받고 있다[1]. IDD 기법은 검출과 복호화의 반복횟수에 따라 전체 수행의 복잡성(complexity)이 증가한다는 단점이 있지만 반복할수록 비트오류 비율(bit error rate, BER) 성능이 개선되는 장점이 있다.

기본적인 IDD에서는 APP 검출기에서 계산의 복잡성을 낮추면서도 성능의 감소를 최소화하는 효율적인 방법을 위해 리스트 스피어 디코딩(list sphere decoding, LSD) 기법을 사용한다. 리스트 스피어 디코딩은 ML 검출(Maximum likelihood detection)을 효율적으로 하기 위한 방법 중의 하나로 알려진 스피어 디코딩(sphere decoding, SD) 기법[2]-[8]을 변형한 형태의 기법이다. 기존의 스피어 디코딩이 한 개의 최적 심볼 벡터(symbol vector)만을 찾는다면, 리스트 스피어 디코딩은 L개의 최적 심볼 벡터를 찾음으로써 복잡성과 성능의 반비례 관계에서 가변적으로 선택할 수 있도록 하는 장점을 갖추고 있다. 그러나 기본적인 IDD 구조에서는 복호기(decoder)에서 나온 사후 확률이 다음 차례의 반복적인 검출에서 사전 정보로 이용될 때 기존 LSD는 사전 정보를 이용할 수 없기 때문에 제외된다. 이는 처음 LSD를 통해 L

개의 심볼 벡터를 찾을 때 잡음(noise)으로 인하여 LLR에서 비중이 낮은 심볼 벡터가 검출된 경우에도 다음 차례의 반복적인 검출에서 심볼 벡터의 정보가 바뀌지 않으므로 이러한 비중이 낮은 심볼 벡터를 계속적으로 이용할 수밖에 없다. 이러한 IDD 구조에서 LSD도 사전정보를 이용할 수 있도록 수정하여 반복적으로 검출할 때 점차 LLR에서 비중이 큰 심볼 벡터의 정보로 교체할 수 있다면 성능이 좀 더 개선될 여지가 있다.

본 논문에서는 기본적인 IDD 구조에서 기존의 LSD를 비선형적 슬라이싱(non-linear slicing)된 사전 정보를 이용할 수 있도록 수정하여 반복적인 검출에 LSD를 포함시킨 수정된 IDD 구조를 제안하고 기존의 IDD와 성능을 비교한다. 수정된 IDD에서는 내부 복호기(inner decoder)의 출력 정보뿐만 아니라 실제 통신 환경에서 사용되는 전방 오류수정(Forward Error Correction, FEC) 등의 비트 오류를 확인할 수 있는 기능을 갖춘 외부 복호기(outer decoder)의 출력도 이용한다. 특히 현재까지도 많은 분야에서 사용되는 강력한 부호로 알려진 Reed-Solomon 부호를 사용하여 사전 정보에 존재할 수 있는 비트 오류의 위치를 검출함으로써 오류전파 현상을 방지한다.

본 논문은 먼저 IDD에서 사용되는 기존의 LSD기법을 수정하여 사전 정보를 이용하는 LSD에 대해서 기술하고 다음으로 사전 정보를 이용하는 LSD를 위한 수정된 IDD를 제안한다. 수정된 IDD에 관련된 내용을 바탕으로 실험하여 결과를 분석하고 마지막으로 결론을 맺는다.

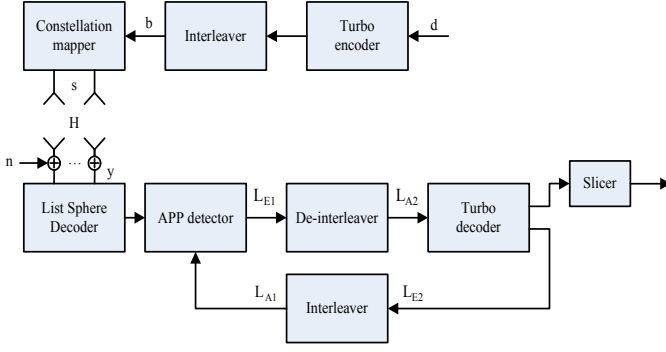


그림 1 기존의 IDD 구조

2. 반복적인 검출과 복호화 기법 (IDD)

그림 1은 기존의 IDD 구조를 나타낸 그림이다. 그림 1의 수신단에서 아래첨자 "1"은 APP 검출기에 관련된 정보임을 나타내고 아래첨자 "2"는 터보 복호기에 관련된 정보임을 나타낸다. APP 검출기에서 구해진 L_{E1} 값은 전송단의 인터리버에 대응하는 디인터리버(de-interleaver)를 거친 후 터보 복호기로 전달된다. 이 정보는 터보 복호기 관점에서는 사전 정보가 된다. 터보 복호기는 전달받은 사전 정보 L_{A2} 를 이용하여 복호화를 수행하고 이를 통해 터보 복호기 자신의 외부 LLR 정보인 L_{E2} 를 계산한다. 이 정보는 반복적인 검출이 수행 중인 경우에는 다시 전송단과 동일한 인터리버를 거쳐서 사전 정보 L_{A1} 으로 APP 검출기에 전달된다. APP 검출기는 전달받은 사전 정보 L_{A1} 를 이용하여 APP를 다시 계산한다. 이 과정은 반복적인 검출이 끝날 때까지 계속 순환하고 반복적인 검출이 끝난 상태인 경우 터보 복호기의 출력인 L_{E2} 가 슬라이서(slicer)를 통하여 검출된 정보 비트로 변환되어 출력된다.

3. 사전 정보를 이용하는 리스트 스피어 디코딩

가. 리스트 스피어 디코딩 기법

전형적인 MIMO 복소수 신호 모델(complex number signal model)에서 전송단과 수신단의 관계는 다음과 같다.

$$y_c = H_c s_c + n_c \quad (1)$$

식(1)에서 아래첨자 "c"는 복소수 벡터를 의미한다. y_c 는 수신단의 수신벡터, H_c 는 채널 매트릭스, s_c 는 전송단의 심볼 벡터이고 n_c 는 AWGN(additive white Gaussian noise)를 나타낸다. 일반적으로 채널 매트릭스 H_c 가 주어졌다고 가정하면 식(1)의 ML 해법(solution)은 다음과 같다.

$$\hat{s}_c = \operatorname{argmin}_{s_c \in \mathbb{C}^M} \|y_c - H_c s_c\|^2 \quad (2)$$

복소수 신호 모델에서 실수 신호 모델의 변환은 다음과 같이 나타낼 수 있다.

$$y = \begin{bmatrix} \mathcal{R}(y_c) \\ \mathcal{I}(y_c) \end{bmatrix}, H = \begin{bmatrix} \mathcal{R}(H_c) & -\mathcal{I}(H_c) \\ \mathcal{I}(H_c) & \mathcal{R}(H_c) \end{bmatrix}, s = \begin{bmatrix} \mathcal{R}(s_c) \\ \mathcal{I}(s_c) \end{bmatrix} \quad (3)$$

식(3)의 관계를 이용하여 식(1), (2)를 실수 신호 모델로 다시 표현하면 다음과 같다.

$$y = Hs + n \quad (4)$$

$$\hat{s} = \operatorname{argmin}_{s \in \mathbb{R}^{2M}} \|y - Hs\|^2 \quad (5)$$

식(5)의 H를 QR-분해(QR-decomposition)를 이용하여 다음과 같이 표현할 수 있다.

$$H = [QU] \begin{bmatrix} R \\ 0 \end{bmatrix} \quad (6)$$

식(6)에서 만약 H의 크기가 $M \times N$ 이라고 하면 Q와 U는 각각 $M \times M$, $M \times (M-N)$ 크기의 단위 행렬(unitary matrix)이고, R은 정방형($N \times N$) 역삼각 형태의 매트릭스, 0은 $(M-N) \times N$ 크기의 영행렬(zero matrix)이다. 식(6)을 이용하여 식(4), (5)를 다시 표현하면

$$Q^T y = Rs + Q^T n \quad (7)$$

$$\hat{s} = \operatorname{argmin}_{s \in \mathbb{R}^{2M}} \|y' - Rs\|^2 \quad (8)$$

식(8)에서 $y' = Q^T y$ 이다. 식(8)의 $y' - Rs$ 부분의 구조를 표현하면 다음과 같다.

$$\begin{bmatrix} y'_1 \\ y'_2 \\ \vdots \\ y'_{N-1} \\ y'_N \end{bmatrix} - \begin{bmatrix} r_{1,1} & r_{1,2} & \cdots & r_{1,N} \\ 0 & r_{2,2} & \cdots & r_{2,N} \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & r_{N-1,N} \\ 0 & 0 & \cdots & r_{N,N} \end{bmatrix} \begin{bmatrix} s_1 \\ s_2 \\ \vdots \\ s_{N-1} \\ s_N \end{bmatrix} \quad (9)$$

식(9)의 구조를 분석해보면 심볼 벡터 s 를 구하기 위해서는 계산량이 적은 s_N 부터 구하는 것이 간단하다는 것을 알 수 있다. 이용하여 식(8)을 다음과 같이 표현할 수 있다.

$$\hat{s} = \operatorname{argmin}_{s \in \mathbb{R}^{2M}} \sum_{j=1}^N B_j(s_j^N) \quad (10)$$

식(10)에서 각 계층(layer)의 가지 값(branch metric)으로 표현될 수 있는 $B_j(s_j^N)$ 는 다음과 같다.

$$B_j(s_j^N) = \left(y'_j - \sum_{k=j}^N r_{j,k} s_k \right)^2 \quad (11)$$

스피어 디코딩 기법에서는 현재까지 탐색된 계층이 $j+1$ 부터 N 이라고 가정하고 현재 계층을 j 라 가정하면 탐색된 계층의 가지 값들의 누적 합으로 나타내어지는 경로 값(path metric)과 현재 계층의 가지 값의 합이 주어진 고차원 구(hypersphere)의 자승 반지름보다 작아야 하는 조건을 만족해야 다음 계층으로의 탐색(tree search)이 가능하다. 이러한 조건은 다음과 같이 나타낼 수 있다.

$$d_0 \geq P_j = P_{j+1} + B_j = B_j + \sum_{k=j+1}^N B_k \quad (12)$$

식(11)을 이용하여 식(12)를 다시 표현하면 다음과 같다

$$d_0 - P_{j+1} \geq \left(y'_j - r_{j,j} s_j - \sum_{k=j+1}^N r_{j,k} s_k \right)^2 \quad (13)$$

식(13)를 통해 구하고자 하는 s_j 의 범위를 다음과 같이 구할 수 있다.

$$s_{j,\min} \leq s_j \leq s_{j,\max} \quad (14)$$

$$s_{j,\min} = \left\lceil \frac{1}{r_{j,j}} \left(y'_j - \sqrt{d_0 - P_{j+1}} - \sum_{k=j+1}^N r_{j,k} s_k \right) \right\rceil \quad (15)$$

$$s_{j,\max} = \left\lfloor \frac{1}{r_{j,j}} \left(y'_j + \sqrt{d_0 - P_{j+1}} - \sum_{k=j+1}^N r_{j,k} s_k \right) \right\rfloor \quad (16)$$

식(15), (16)의 $\lceil \cdot \rceil$, $\lfloor \cdot \rfloor$ 은 각각 올림 함수(ceiling function)와 내림 함수(floor function)이다.

스피어 디코딩에서는 처음 탐색을 시작 시에 식(7)을 이용하여 최하위 계층인 N번째 계층부터 각 계층마다 식(14)를 만족하는 심볼 s_j 를 구하고 최상위 계층인 1번째 계층에 도달하여 식(12)를 만족하는 경우 그 때의 경로 값을 새로운 반지름으로 교체하고 나머지 가지들을 탐색하여 ML 해법을 찾아낸다.

리스트 스피어 디코딩은 스피어 디코딩과 다르게 구해야 할 심볼 벡터 개수 L이 주어지고 이에 대응하는 심볼 벡터의 목록 \mathbb{L} 이 생성된다. 또한 최상위 계층인 1번째 계층에 도달하여 식(12)를 만족하는 경우라도 주어진 심볼 벡터의 목록(list) \mathbb{L} 이 다 채워지지 않은 경우, 다시 말해 지금까지 구한 심볼 벡터의 개수가 L개보다 적은 경우에는 반지름을 새로 교체하지 않고 현재의 심볼 벡터를 목록 \mathbb{L} 에 채워지게 된다. 만약 탐색이 종료되기 전에 심볼 벡터 목록 \mathbb{L} 이 다 채워진 경우에는 반지름을 심볼 벡터 목록 \mathbb{L} 의 L번째 (심볼 벡터 목록 중에서 반지름이 가장 큰) 심볼 벡터의 반지름으로 교체하고 탐색을 계속한다. 만약 탐색을 계속하던 중에 현재 반지름보다 작은 심볼 벡터를 구한 경우에는 심볼 벡터 목록 \mathbb{L} 에서 L번째 심볼 벡터를 제외하여 새로운 심볼 벡터를 갱신하고 그 때의 새로운 L번째 심볼 벡터의 반지름으로 교체하고 탐색이 끝날 때까지 이 과정을 반복한다. LSD의 경우에는 SD와 달리 조건을 만족하는 심볼 벡터를 찾더라도 경우에 따라 반지름이 새로 교체되지 않고 또한 교체된 반지름은 대부분의 경우 SD의 반지름보다 더 느슨하기 때문에 LSD의 복잡성(complexity)는 SD보다 높은 단점이 있다.

나. 사전 정보를 이용하는 LSD 기법

기존의 IDD에서는 처음 검출 시 구한 심볼 벡터 목록 \mathbb{L} 을 반복적으로 검출하는 과정에서도 그대로 사용하게 된다. 이는 APP 검출기에서 처음 검출 시 잡음 등의 어떠한 이유로 인하여 심볼 벡터 목록 \mathbb{L} 이 성능이 떨어지게 되는 심볼 벡터들을 가지게 되는 경우 반복적인 검출을 통하더라도 이미 구해진 심볼 벡터 목록 \mathbb{L} 로만 LLR을 계산하게 된다는 단점을 내포하게 된다. 이러한 점을 개선하기 위해서는 이전의 반복적인 검출 과정에서 주어지는 사전 정보를 LSD에서 이용하여 새로운 심볼 벡터 목록 \mathbb{L} 을 갱신하여 성능을 개선할 수 있는 LLR을 계산할 수 있도록 수정되어야 한다. 그러나 사전 정보를 현재의 LSD 구조에서는 직접 이용할 수 없기 때문에 간접적인 형태의 변환을 필요로 한다.

우선 사전 정보 L_A 를 LSD에서 사용할 수 있는 형태로 변환하기 위해 주어진 임계 값(threshold) L_{th} 에 대해 다음과 같은 비선형적 슬라이서(slicer)를 정의한다.

$$f_{th}(L_A) = \begin{cases} +1 & L_A \geq +L_{th} \\ -1 & L_A \leq -L_{th} \\ 0 & \text{otherwise} \end{cases} \quad (17)$$

식(17)을 통과한 사전 정보는 LLR로부터 슬라이스 된 최종적으로 검출되는 정보 비트와 유사한 형태를 가지게 되므로 이를 \hat{b}_{pr} 라 하고 다음과 같이 정의한다.

$$\hat{b}_{pr} = f_{th}(L_A(b)) \quad (18)$$

수정된 LSD에서는 식(12)에서 현재 계층의 가지 값을 계산할 때 현재 계산하려는 심볼 s_j 에 대해 식(14)의 조건뿐만 아니라 식(18)을 통해 구한 \hat{b}_{pr} 을 이용하여 현재 계층에 해당하는 비트 정보를 참조하

여 가지 값을 계산한다. 이 경우 \hat{b}_{pr} 의 원소(element)가 +1이나 -1인 경우에는 그에 해당하는 심볼의 범위는 제한되고 0인 경우에는 가변적이 된다. (예를 들어, 16-QAM을 사용하는 4x4 MIMO 복소수 신호 모델에서 현재 5번째 계층을 탐색하고 있고, 이에 대응되는 \hat{b}_{pr} 의 9번째, 10번째의 값이 0, -1 이라면 5번째 계층에서 찾게 되는 실수 신호 모델의 심볼 s_5 의 범위는 $-\frac{3}{\sqrt{10}}$ 과 $\frac{3}{\sqrt{10}}$ 의 두 가지가 된다. 만약 1, 0인 경우에는 $\frac{1}{\sqrt{10}}$ 과 $\frac{3}{\sqrt{10}}$ 이 된다. 0, 0의 경우에는 식(14)의 모든 범위가 된다.) 이러한 사전 정보를 이용함으로써 수정된 LSD에서는 일부 계층의 심볼의 범위가 제한되기 때문에 복잡성이 개선되는 이점이 생긴다.

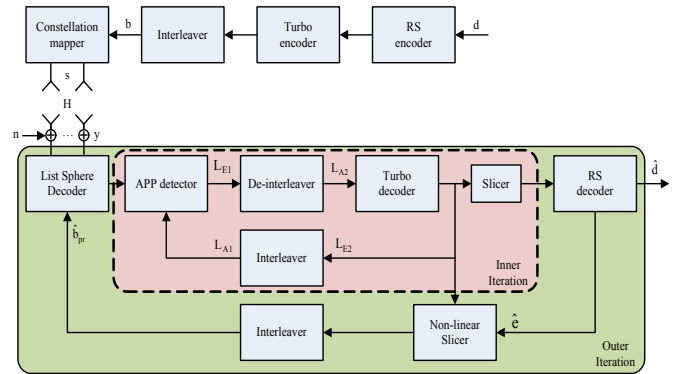


그림 2 수정된 IDD 구조

4. 사전 정보를 이용한 LSD를 위한 수정된 IDD

그림3은 앞에서 언급된 내용을 기존의 IDD에 적용하여 새로이 수정된 IDD 구조를 나타낸 것이다. 전달하고자 하는 정보인 d는 RS 부호기와 터보 부호기를 거친 후 연결 오류에 대항하기 위한 인터리버를 통한 뒤 심볼벡터로 변환되어 수신단으로 전달된다. 수신단에서는 수신된 벡터를 리스트 구 복호기를 통하여 검출한다. 이 때의 리스트 구 복호기는 사전 정보가 없는 상태이므로 기존의 리스트 구 복호기와 같은 동작을 하게 된다. 수정된 IDD 구조에서는 기존의 IDD 구조가 내부 피드백 형태를 구성하고 RS 복호기를 통한 외부 피드백이 추가된 형태의 구조를 이룬다. 이러한 다중 피드백의 구조는 피드백의 횟수에 따라 성능과 복잡도를 조절할 수 있다는 이점이 생기게 된다. 주어진 내부 피드백 반복 횟수만큼 터보 복호기의 반복적인 검출 작업이 수행된 후 검출된 정보는 외부 피드백의 RS 복호기로 전달된다. RS 복호기는 전달받은 정보를 복호화하여 오류 위치를 검출하게 된다. 검출된 오류 위치 정보는 비선형적 슬라이서로 전달된다.

RS 복호기를 통한 외부 피드백의 정보를 사전정보를 이용하는 리스트 구 복호기에 전달하기 위해서는 터보 복호기의 LLR 값을 비선형적으로 슬라이싱 한 정보와 RS 복호기의 오류 위치 정보를 병합(merge)하는 과정이 필요로 하게 된다. 이러한 과정을 그림 3의 외부 피드백에서 비선형적 슬라이서가 수행하게 된다. 터보 복호기에서 얻어진 L_{E2} 는 식(17)의 f_{th} 에 의해 슬라이싱 된다. 이러한 슬라이싱 된 정보 $f_{th}(L_{E2})$ 는 RS 복호기의 오류 위치 정보 \hat{e} 와 대조되어 오류가 있다고 확인된 위치에 있는 경우 이 정보는 사용되지 않고 오류가 없

다고 확인된 정보만이 사전 정보 \hat{b}_{pr} 이 되어 LSD에 전달된다. LSD는 전달받은 \hat{b}_{pr} 을 이용하여 재 검출을 수행하고 새로운 심볼 벡터 목록을 구한다. 구해진 심볼 벡터 목록은 다시 APP 검출기로 전달되고 기존의 IDD에 해당되는 내부 피드백 과정을 거치게 되면서 외부 피드백이 한차례 종료되게 된다. 표 1은 이러한 비선형적인 슬라이싱 과정을 $L_{th} = 5$ 인 경우에 대해 예를 들어서 나타낸 것이다.

L_{E2}	...	2.5	-5.3	7.1	-4.7	-6.4	5.6	-3.8	8.3	...
$f_{th}(L_{E2})$...	0	-1	1	0	-1	1	0	1	...
\hat{e}	...	0	0	0	1	1	0	0	0	...
\hat{b}_{pr}	...	0	-1	1	0	0	1	0	1	...

표 1 비선형적인 슬라이싱 ($L_{th} = 5$)

5. 모의실험

위에서 언급된 수정된 IDD의 성능을 검증하기 위하여 MATLAB을 통한 모의실험을 수행하였다. 그림 3은 정보비트가 1760이고 LSD는 64개의 심볼 벡터 목록을 검출한 경우의 외부 피드백 횟수에 따른 터보 복호기 관점의 BER 성능을 나타낸 것이다. 약 6dB 이전의 영역에서는 외부 피드백의 영향이 거의 없는 상태인데 그 이유는 RS(15,11)이 검출할 수 있는 오류의 개수는 4비트 크기의 심볼 2개이나 이 영역에서는 이를 넘어서는 오류가 발생하여 RS 복호기가 오류 발생 위치를 검출하지 못하기 때문이다. 다시 말해 이는 60비트 중 연립오류 형태의 8비트의 오류를 수정할 수 있는데 이 영역에서 이보다 많은 오류가 발생하기 때문에 RS(15,11) 복호기의 오류 위치 정보가 LSD의 심볼 벡터 목록 재 검출에 거의 영향을 미치지 못하는 상태가 된다. 외부 피드백의 영향이 나타나는 영역은 waterfall 현상이 일어나는 영역의 바로 이전 영역인 7dB에서 뚜렷하게 나타난다. 이 영역에서는 RS 복호기에 의해 검출된 오류 위치 정보가 사전 정보를 이용한 LSD 검출에 매우 큰 영향을 미치기 때문이다. 이는 APP 검출 시 원래의 정보에 더 가까워지는 영향을 주게 된다. BER이 약 10^{-1} 일 때 외부 피드백이 없는 기존의 IDD 보다 외부 피드백을 4회 수행한 결과가 약 0.3dB 개선됨을 확인 할 수 있다.

6. 결론

본 논문에서는 기존의 반복적인 검출 및 복호화(IDD) 기법에서 APP 검출기에 심볼 벡터 목록을 전달하는 리스트 구 복호기(List Sphere Decoder)를 사전 정보(prior information)를 이용하여 검출을 수행할 수 있도록 하여 이를 이용한 기존의 IDD보다 성능이 개선된 수정된 IDD를 제안하였다.

기존의 IDD는 반복적인 복호화를 수행할 때 LSD를 제외하고 수행하기 때문에 한번 검출된 심볼 벡터 목록이 계속적으로 사용되게 된다. 이는 첫 검출에서 원래의 정보와 매우 다른 형태로 검출된 경우 반복적인 검출을 수행하더라도 원래의 정보에 근접하기 힘들게 된다. 그러므로 이를 수정하기 위해 터보 복호기의 출력값인 LLR에 비선형적인 슬라이싱을 하여 이를 LSD에서 참고하여 새로운 심볼 벡터 목록을 검출할 수 있도록 하였다. 또한 Reed-Solomon 부호를 외부 부호로 사용하여 비트 오류의 위치를 검출하였다. 검출된 오류 위치 정보는 새로이 검출되는 심볼 벡터 목록의 정확도를 높이는 역할을 하도록 하였다.

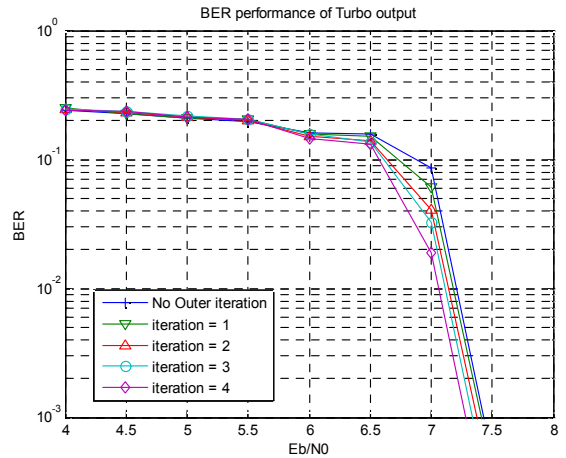


그림 3 모의실험 결과

이러한 구조는 기존의 IDD에 외부 피드백이 더해지는 형태로 구성되었다. 이러한 수정된 IDD는 모의실험을 통해 기존의 IDD보다 성능이 개선됨을 확인하였다.

감사의 말

이 연구에 참여한 연구자(의 일부)는 '2단계 BK21사업' 지원비를 받았음.

참고자료

- [1] B. M. Hochwald and S. ten Brink, "Achieving near-capacity on a multiple-antenna channel," IEEE Trans. Commun., Vol. 51, pp. 389-399, March 2003.
- [2] U. Fincke and M. Pohst, "Improved methods for calculating vectors of short length in a lattice, including a complexity analysis," Math. Computat., vol. 44, pp. 463.471, 1985.
- [3] M. Pohst, "On the computation of lattice vectors of minimal length, successive minima and reduced basis with applications," in Proc. ACM SIGSAM, 1981, vol. 15, pp. 37.44.
- [4] E. Viterbo and E. Biglieri, "A universal lattice decoder," in Proc. GRETSI 14-eme Colloque, Jun-les-pins, France, Sep. 1993, pp. 611.614.
- [5] C. P. Schnorr and M. Euchner, "Lattice basis reduction: Improved practical algorithms and solving subset sum problems," Math. Programm., vol. 66, pp. 181.191, 1994.
- [6] E. Agrell, T. Eriksson, A. Vardy, and K. Zeger, "Closest point search in lattices," IEEE Trans. Inf. Theory, vol. 48, pp. 2201.2214, Aug. 2002.
- [7] M. O. Damen, H. Gamel, and G. Caire, "On maximum-likelihood detection and the search for the closest lattice point," IEEE Trans. Inf. Theory, vol. 49, pp. 2389.2402, Oct. 2003.
- [8] E. Viterbo and J. Boutros, "A universal lattice code decoder for fading channels," IEEE Trans. Inf. Theory, vol. 45, pp. 1639.1642, Jul. 1999.
- [9] B. Shim and I. Kang, "Sphere decoding with a probabilistic tree pruning," IEEE Transaction on signal processing, vol. 56, no. 10, pp. 4867-4878, Oct. 2008.
- [10] J. C. Moreira and P. G. Farrell, Essentials of Error-Control Coding. England: John Wiley & Sons Ltd, 2006.
- [11] A. Papoulis and S. U. Pillai, Probability, Random Variables and Stochastic Processes. New York: McGraw-Hill, 2002.