

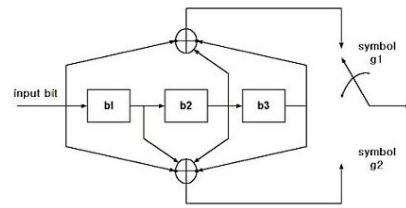
구속장 길이에 따른 Viterbi Decoder의 내부 메모리 오류에 대한 정정능력 평가

김호준, 김민수, 김종태
성균관대학교 휴대폰학과

Evaluation of the Error Correction Ability in the inner memory error for the Viterbi Decoder According to the Constraint Length

Hojun Kim, Min Su Kim, Jong Tae Kim
The Department of Mobile Systems Engineering, Sungkyungwan University

Abstract - 1967년 Andrew J. Viterbi에 의해 처음 제안된 Viterbi 알고리즘은 길쌈부호(convolution code)의 대표적인 복호방법으로 현재 통신 기술 중에서 가장 많이 쓰이는 것 중에 하나이다. Viterbi decoder는 사용되는 시스템의 사양에 따라 에러 수정 능력이 다른데 통신 channel 상의 오류뿐만 아니라 Viterbi decoder내부에 있는 메모리에서 발생하는 오류도 Viterbi decoder의 에러 수정 능력에 영향을 준다. 본 논문에서는 일반적으로 많이 확인되었던 channel상의 오류와 함께 Viterbi decoder내부에 있는 메모리에서 오류가 발생했을 때 복-부호기의 사양에 따른 에러정정능력을 분석하였다.



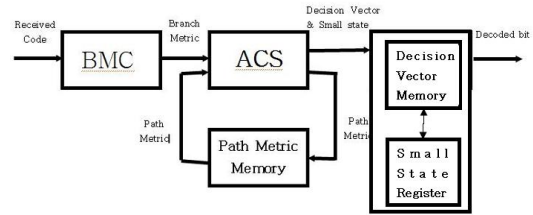
<그림 1> 부호율이 1/2이고 구속장 길이가 3인 길쌈부호기의 구조

1. 서 론

디지털 통신시스템에는 데이터 전송 시 channel상에서 data에 오류가 발생하는데 이러한 오류를 수신측에서 판단하여 올바른 data로 정정해 주어야 한다. 이렇게 오류정정이 가능하도록 보내고자 하는 정보를 변형시키는 과정을 오류 정정 부호화(channel coding)라고 하는데 channel coding중의 하나인 Viterbi decoder는 우주통신[1], 장거리통신, 이동통신, 방송 등에서 많이 사용되고 있다. Viterbi decoder의 내부에는 수신 비트와 중간 data 저장에 위해 다수의 메모리 소자가 필요하다. 이런 메모리 소자가 중성자 방사선에 노출이 되면 반도체 실리콘 기관과 방사선이 충돌하여, 다수의 양성자 및 중이온, 중성자 및 알파 입자를 발생시킨다. 이러한 고에너지 입자들은 전자 축적을 유도하여 저장 소자에 저장된 값을 변화시킬 수 있다.[2] 또는 저전력을 구현하기 위해 메모리에 공급하는 전압을 낮추면 메모리 배열에 영향을 줘서 데이터가 변하는 내부에러가 날 확률이 높아진다. 최근 국내에서는 항공우주 산업에 적합한 전자부품 개발 및 시험에 대한 관심이 매우 증대되어 있으나 단일 메모리 소자이외의 특정 부품에 대한 방사선 영향 평가 기술과 완화 기법에 대한 연구 실적이 매우 부족한 실정이다. 본 논문에서는 SRAM 기반의 Viterbi decoder의 내부메모리에 임의의 에러를 생성하여 시뮬레이션하고 결과를 분석하였다.

2.2 Viterbi 복호기

Viterbi decoder의 기본 구조는 <그림 2>와 같이 일반적으로 BMC(Branch Metric Calculation), ACS(Add Compare Select), SMM(Survivor Path Memory Management) 세부분으로 구성되어 있다.



<그림 2> Viterbi Decoder의 기본 구조

2. 본 론

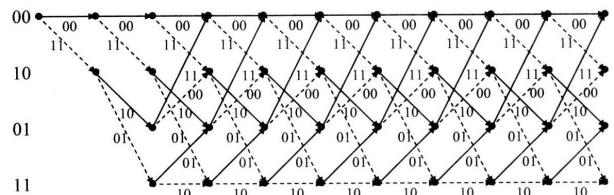
2.1 길쌈부호기

길쌈부호는 현재의 입력과 과거의 입력간의 상호 연관성에 기반을 두고 n개의 data bit를 m개의 codeword로 바꾸는 부호화 방법이다. 과거의 입력 값과 현재의 입력 값을 이용하여 codeword를 만들 때 <표 1>과 같은 두 가지의 생성 다항식(generator polynomial)을 이용한다. 생성다항식에서 각 미지수의 차수는 과거에 들어온 시점을 나타낸다. 가령 x의 경우는 바로 직전에 들어온 과거 값이고 x의 제곱승은 이전 이전의 과거 값이다. 그래서 길쌈부호기에는 <그림 1>과 같이 과거의 입력 값을 저장하는 register가 있어야 하는데 이 때 사용되는 register의 개수를 구속장의 길이(constraint length),K,라하고 n비트의 data를 m비트의 codeword로 부호화 할 때 n/m을 부호율이라 한다.

우선 BMC블록에서는 hamming distance를 이용해 처음 상태(길쌈부호기 register중 과거 값만을 뽑아낸 것)에서 다음 상태로 갈 때 만들어질 수 있는 codeword와 전송받은 codeword와의 차이인 branch metric을 계산한다. ACS블록에서는 BMC로부터 받은 branch metric을 Path metric memory에 저장되어 있는 각 상태의 누적 값에 더해져 최소가 되는 값을 찾고, 각 상태의 새로운 누적 값을 path metric memory에 다시 저장한다. 그리고 모든 상태 중에서 최소값인 small state와 각 상태에서의 최소값들이 어떤 이전상태에서 들어왔는지를 확인하여 이전상태 값을 적은 decision vector를 생성하여 SMM 블록으로 전송한다. SMM블록에서는 ACS로부터 받은 decision vector와 small state를 비교하며 역추적 하여 data값을 얻어낸다. 이때 한 비트의 data를 얻어내기 위해서는 하나의 small state를 구속장 길이 K의 3~4배 정도 되는 양의 decision vector와 비교해야 하는데 이때 역시 과거의 decision vector를 저장하기 위해서 memory가 필요하다. 이런 복호 과정은 <그림 3>과 같은 격자도(trellis diagram)를 이용해 나타낼 수 있다.

<표 1> 1/2부호율에 대한 생성 다항식

| | U1 | U2 |
|-----|-----------------------------|---------------------------|
| K=3 | $x^2 + x + 1$ | $x^2 + 1$ |
| K=4 | $x^3 + x^2 + x + 1$ | $x^3 + x^2 + 1$ |
| K=5 | $x^4 + x^3 + x^2 + 1$ | $x^4 + x + 1$ |
| K=6 | $x^5 + x^3 + x^2 + x + 1$ | $x^5 + x^4 + x^2 + 1$ |
| K=7 | $x^6 + x^5 + x^4 + x^3 + 1$ | $x^6 + x^4 + x^3 + x + 1$ |



<그림 3> 부호율이 1/2이고 구속장 길이가 3인 trellis diagram

2.3 실험 및 분석

시뮬레이션에는 100,000개의 랜덤한 Data를 사용하였고, 부호율이 1/2인 길쌈부호기를 통과시켜 부호화 하였다. channel 전송시에는 BPSK로 Modulation 한 뒤 전송받은 신호를 Viterbi decoder의 구속장 길이를 변형시켜 실행하며 성능을 확인하였다. 실험의 목적이 Viterbi decoder 내부에 있는 메모리에서 에러가 났을 경우 그것이 Viterbi decoder의 성능에 어떠한 영향을 주는지 알아보는 것이므로 내부메모리에 <표 2>와 같은 에러를 임의적으로 제공하였다.

<표 2> 실험에 사용한 BER

| | | | | |
|-----|---|-------|--------|----------|
| BER | 0 | 0.001 | 0.0001 | 0.000001 |
|-----|---|-------|--------|----------|

그리고 구속장의 길이가 길어질수록 중간 Data를 저장하기위한 메모리의 용량이 커진다. Decision Vector Memory의 크기는 구속장길이의 4 배, Path Metric Memory의 크기는 구속장길이 K의 2^K 을 적용하여 <표 3>에 나타난 크기로 하였다.

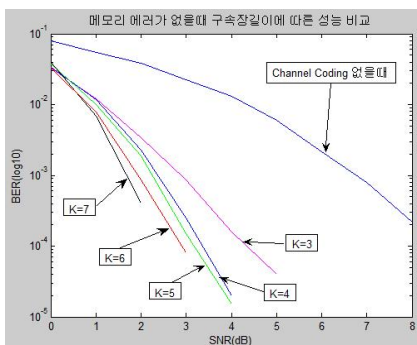
<표 3> Decision Vector Memory 및 Path Metric Memory의 크기

| 구속장길이 | Decision Vector Memory | | Path Metric Memory | |
|-------|------------------------|---------|--------------------|---------|
| | 가로(bit) | 세로(bit) | 가로(bit) | 세로(bit) |
| K=3 | 12 | 4 | 1 | 8 |
| K=4 | 16 | 8 | 1 | 16 |
| K=5 | 20 | 16 | 1 | 32 |
| K=6 | 24 | 32 | 1 | 64 |
| K=7 | 28 | 64 | 1 | 128 |

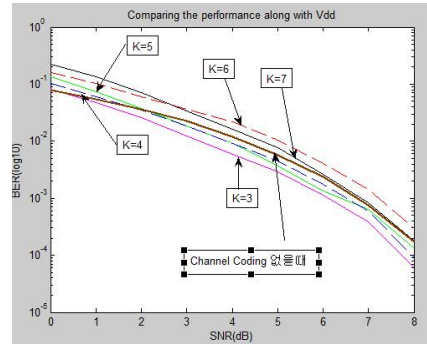
메모리에 에러가 발생하지 않고 채널 상에 생긴 오류를 정정한다면 <그림 4>에 나타난 것과 같이 Channel Coding을 사용하지 않았을 때보다 Channel Coding을 했을 때 성능이 훨씬 좋아졌고, Constraint Length가 길어질 때 마다 성능은 더 좋아진다는 것을 알 수 있다. 메모리 각 배열에 에러가 날 확률을 0.000001로 하여 적용하면, 구속장 길이가 길어질수록 성능이 좋아지던 Viterbi Decoder가 메모리에 에러가 생기자 <그림 5>에서와 같이 일정 SNR보다 커질 때 k=7일 때가 k=6일 때보다 성능이 나빠지고 k=5일 때가 k=4일 때보다 성능이 나빠지는 등 성능이 점점 구속장길이에 역으로 안 좋아지는 현상이 나타나기 시작한다.

메모리 각 배열에 에러가 날 확률을 0.0001로 조금 증가 하면 <그림 6>에서와 같이 K=3,4,5일 때는 모든 SNR에서 역전되어 역으로 구속장 길이가 길어질 때 BER이 안 좋아 지고 k=6일 때와 k=7일 때는 특정 SNR 이후에 역전이 된다.

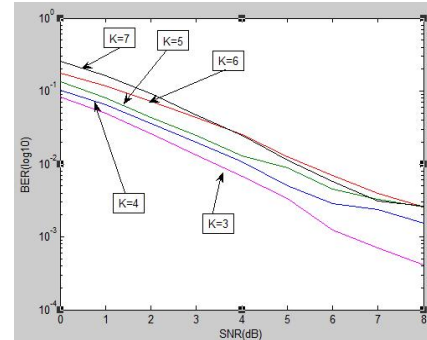
메모리 각 배열에 에러가 날 확률을 0.01까지 증가 하게 되면, <그림 7>에 나타난 것과 같이 메모리에 에러가 발생하지 않았을 때와는 정반대 상황인 구속장 길이가 길어질수록 정정능력이 떨어지는 것을 알 수 있다. 그리고 Channel Coding을 하지 않았을 때보다 성능이 더 떨어진 다.



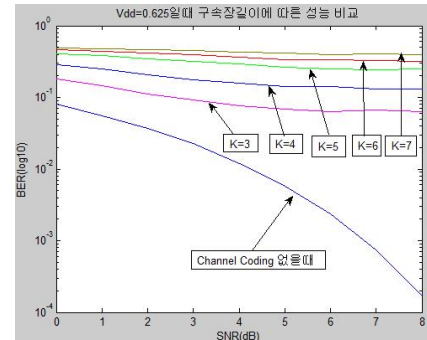
<그림 4> 메모리에 에러가 없을 때 구속장 길이에 따른 성능 비교



<그림 5> 메모리에 에러날 확률을 0.000001로 낮췄을 때



<그림 6> 메모리에 에러날 확률을 0.0001로 낮췄을 때



<그림 7> 메모리에 에러날 확률을 0.01로 낮췄을 때

3. 결 론

본 논문에서는 통신 분야에서 가장 많이 쓰이는 Channel Coding중의 하나인 Viterbi Decoder의 내부에 있는 메모리에서 에러가 났을 때 그것이 Viterbi Decoder의 정정능력에 미치는 영향과 구속장 길이에 따른 성능을 분석하였다. 구속장의 길이가 증가 할수록 중간 Data를 저장하기 위한 Memory의 저장용량이 증가하고 그만큼 에러가 발생할 수 있는 비트배열부분이 더 많아지기 때문에 성능이 나빠진다는 것을 확인하였다. 그러므로 통신할 때 방사선의 노출에 대비하여 강한 구조를 갖는 반도체소자 개발에 투자를 해야 하며 더불어 메모리에서의 에러를 대비한 정정 방법을 추가 하여야 한다.

[참 고 문 헌]

- [1] Heller. J. and Jacobs. I, "Viterbi Decoding for Satellite and Space Communication" IEEE Transactions on Communication, Vol. 19, Issue 5, Part 1, Oct 1971 pp.835-848
- [2] Martha V O'Bryan, "Recent single event effects results for candidate spacecraft electronics for NASA," Radiation IEEE Effects Data Workshop 2005, pp.26-35
- [3] Andrew J. Viterbi " Error bounds for convolutional codes and an asymptotically optimum decoding algorithm", IEEE Transactions on Information Theory, Vol. 13 Issue 2, Apr 1967, pp.260-269
- [4] Bernard Sklar, "Digital Communications Second Edition", Prentice Hall, 2001