國立清華大學 電機工程學系

實作專題研究成果摘要

Unequal Error Protection Low-Density Parity-Check Codes
非均等錯誤保護低密度偶校碼編解碼實現

專題領域: 通訊

組別: A190

指導教授: 趙啟超教授

研究期間:110 年 9 月至 111 年 5 月，計 9 個月

摘要

當訊息經過通道(Channel)後，會有雜訊(noise)加入而產生訊息上的錯誤，透過錯誤更正碼能夠更正訊號的錯誤。而低密度奇偶校驗碼(Low-Density Parity-Check Codes，LDPC code)是錯誤更正碼的一種，在 1960 年代被 Gallager 所提出，為更正能力表現最佳的一種錯誤更正碼。

在此次專題中，使用 C 語言來作為模擬的工具，並用 MATLAB 來進行繪圖，以畫出 SNR-BER 的曲線圖。

這次專題研究 LDPC 編碼器(encoder)和解碼器(decoder)的實作，並且探討均等錯誤保護碼(Equal error protection code，EEP code)與不均等錯誤保護碼(Unequal error protection code，UEP code)對於不同 level 間有不同保護能力。除此之外，用相同大小的 EEP code 和 UEP code 在 BPSK 通道上傳送，並在此通道加上高斯白雜訊(Additive white Gaussian noise)，觀察兩者之間對訊息的保護能力是否符合預期。

對於需要有不同程度上的資料保護，UEP QC-LDPC code 是一項不錯的選擇。

二、報告內容

(一)前言

在 1960 年代，低密度奇偶校驗碼(Low-Density Parity-Check Codes)被 Gallager 所提出，是一種線性區段碼(Linear block code)並且是接近向農限制(Shonnon limit)，其更正能力表現最佳的一種錯誤更正碼，但因為複雜度過高而當時電腦無法處理，直到 1990 年代，重新被廣為討論。

本次專題主要了解低密度奇偶校驗碼的編碼器(encoder)和解碼器(decoder)的實作，並且了解其解碼時所使用的 iterative message-passing decoding algorithm 也稱之為 belief propagation 或是 sum-product algorithm，用 C 語言進行模擬。在專題中，也模擬均等錯誤保護碼(Equal error protection code，EEP code)的情況，和模擬不均等錯誤保護碼(Unequal error protection code，UEP code)對於不同 level 具有不相同的更正能力，且實作是模擬通過二元相移鍵控(BPSK)通道並加上高斯白雜訊(Additive white Gaussian noise)。

(二)原理分析與系統設計

2.1 原理分析

2.1.1 錯誤更正碼(Error correcting codes)

在通訊過程中，會有一些資料經過通道，而此通道會有許多雜訊，然後造成資料的錯誤，藉由錯誤更正碼把錯誤更正回來。錯誤更正碼主要架構在數位通訊系統(digital communication systems)裡的 channel coding，分別為通道編碼器(channel encoder)和通道解碼器(channel decoder)，而複雜度主要在 channel decoder 上。

用線性代數的方式，定義 code C 為奇偶校驗矩陣(parity-check matrix，H)的零空間(nullspace)，此時有一個 matrix 能夠形成所有 code C，而稱之為生成矩陣(generator matrix，G)，此 code C 為 generator matrix 的列空間(row space)。此時如果 G 是$[I_k|A]_{k\times n}$的形式，則 H 為 $[-A^T|I_k]_{(n-k)\times n}$，反之亦然。若 G 是一個大小為$k\times n$的矩陣，此 code 會稱為 (n,k) code，code rate 為 k / n。

對於進入到編碼過程(encoding process)，有一個 k bits 的 information bits u = $(u_1,u_2,\ldots,u_k)$，形成相對應的 n bit codeword x = $(x_1,x_2,\ldots,x_n)$，此編碼過程是 x = uG。經過通道後會加上雜訊(noise) n bit z = $(z_1,z_2,\ldots,z_n)$，通道外會收到 n bit garbled codeword y = $(y_1,y_2,\ldots,y_n)$，而 z = y – x。接著進入到解碼過程(decoding process)，把收到的訊息 vector y 乘上 H，$s^T=Hy^T$，稱計算出來的 vector s 為 syndrome，如果 s = 0，表示 y 為一組 codeword；如果 s ≠ 0，便找出一組 codeword y’能使Hy'^T=0，而此時

y’會是離 y 最近的一組 codeword。

2.1.2 線性區段碼(Linear block code)

奇偶校驗矩陣(parity-check matrix)，會用 H 表示。此時有一個大小是 r×n 的奇偶校驗矩陣 H，使原本訊息為 r bit 經過編碼後形成一個 n bits 的訊息，稱之為一個塊(block)。而在 2.1.1 錯誤更正碼中以提及 code C 和 matrix H 和生成矩陣 G 的關係，如何形成 G 則可以利用高斯消去法，先將 H 轉成 reduced row echelon form 的形式，藉由 column 之間的調換把 H 的右半邊形成單位矩陣，左半邊為剩下的 column，便把此 H 的左半邊經由轉置後放入 G 的右半邊，而 G 的左半邊補上單位矩陣，經過一連串步驟即可得到生成矩陣。

2.1.3 低密度奇偶校驗碼(Low-Density Parity-Check Codes，以下簡稱 LDPC code)

LDPC code 是一種線性區段碼(linear block code)，而 LDPC code 的 parity-check matrix H 滿足了以下四個特點：第一點是任一行(column)包含 dv 個 1；第二點是任一列(row)包含 dc 個 1；第三點是對於有 n 個 column 和 rc 個 row 的 parity-check matrix H，dc 和 dv 的個數比 rc 和 n 長度小很多，且滿足了所定義的公式 2.1.3；第四點是任何兩個 column 不會有任何 2 個 1 出現在同樣的位置上，也同樣地任何兩個 row 不會有兩個 1 同樣的位置上。

$$\frac{total\ number\ of\ 1's}{nr_c} = \frac{dc}{n} = \frac{dv}{r_c} \quad \text{公式 2.1.3}$$

LDPC code 可以藉由 Tanner graph 來表示，而此時的 Tanner graph 是 bipartite graph。此 bipartite graph 的頂點(vertex)集合分成兩個子集合，一個為 variable node 的子集合對應到 H 的行，另一個為 check node 子集合對應到列。比如現在有一個 parity-check matrix $H = \begin{bmatrix} 1 & 0 & 1 \\ 1 & 1 & 0 \end{bmatrix}$，而其 bipartite graph 如圖 2.1.1，此時有兩個 check node 分別為 s1 和 s2，三個 variable node 分別為 v1、v2 和 v3。

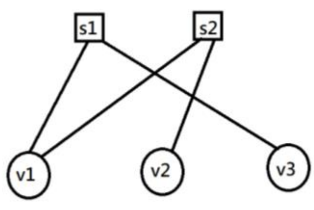

圖 2.1.1

有了對 LDPC definition 和 tanner graph 的了解，進入 decoding 的部分，使用 iterative message-passing decoding algorithm。首先，定義 log likelihood ratio (LLR)，如公式 2.1.4：

$L(p_0, p_1) = ln\frac{p_0}{p_1} = ln\lambda$，公式 2.1.4

$p_0 + p_1 = 1$，

$p_0 = P(x = 0)$等同於傳 0 的機率，$p_1 = P(x = 1)$等同於傳 1 的機率

並且也定義兩種運算，分別為 variable-node operation 和 check-node operation，如公式 2.1.5 和公式 2.1.6。

$VAR(L_1, L_2) = L_1 + L_2$，公式 2.1.5

$CHK(L_1, L_2) = sgn(L_1)sgn(L_2)min(|L_1|, |L_2|) + \Delta(L_1, L_2)$，公式 2.1.6

$\Delta(L_1, L_2) = ln\frac{1+e^{-|L_1+L_2|}}{1+e^{-|L_1-L_2|}}$，

而$\Delta(L_1, L_2)$的計算需要很大計算量，因此建立一個表格，藉由查表的方式即可。此兩種運算滿足 the associative law，分別為:

$VAR(L_1, L_2, \ldots, L_l) = VAR(L_1, VAR(L_2, \ldots, L_l))$，

$CHK(L_1, L_2, \ldots, L_l) = CHK(L_1, CHK(L_2, \ldots, L_l))$

然後定義 L(i)、$M(j)$、$q_{i,j}$、$u_{i,j}$和L_j：

(1) $L(i) = \{j: H_{i,j} = 1\}$，i=1,2,…,$r_c$

(2) $M(j) = \{i : H_{i,j} = 1\}$，j=1,2,…,n

(3) $q_{i,j}$，是在 codeword 中第 j 個 bit 的 LLR，除了 check-node i 之外，其他 check-node 告訴 variable-node j 應該給 check-node i 是 1 還是 0 的資訊。

(4) $u_{i,j}$，是在 codeword 中第 j 個 bit 的 LLR，除了 variable-node j 之外，其他 variable-node 告訴 check-nod*-e i 應該給 variable-node j 是 1 還是 0 的資訊。

(5) 令傳送的 codeword 為 $x = (x_1, x_2, \ldots, x_n)$和通道上的 noise vector 為 $n = (n_1, n_2, \ldots, n_n)$，接收到的 vector 為$y = (y_1, y_2, \ldots, y_n)$。

並定義$L_j = ln\frac{P(y_j|x_j=0)}{P(y_j|x_j=1)}$,是通道輸出的 LLR。當通道為 binary input 高斯白雜訊(additive white Gaussian noise channel)，$L_j = 4 * \frac{E_s}{N_0} * y_j$，$\frac{E_s}{N_0}$是 symbol signal-to-noise ratio (SNR)。

進入到 sum-product algorithm：

Initialization:

對於每一個(i,j)有$H_{i,j} = 1$，設定$q_{i,j} = L_j$。

Message passing:

1.Bottom-up: 對於每一個(i,j)，當此時$H_{i,j} = 1$成立時，計算$u_{i,j}$的公式如下:

$$u_{i,j} = CHK_{j' \in L(i) \backslash j}(q_{i,j'})$$

2.Top-down: 對於每一個(i,j)，當此時$H_{i,j} = 1$成立時，計算$q_{i,j}$的公式如下:

$$q_{i,j} = VAR(VAR_{i' \in M(j) \backslash i}(u_{i',j}), L_j)$$

Decision:

對於 j = 1,2,,…,n，計算 the log a posteriori probability q_j，公式如下:

$$q_j = VAR(VAR_{i\in M(j)}(u_{i,j}), L_j)$$

並且對於 j = 1,2,,…,n，$x_j = \begin{cases} 0, & if\ q_j \geq 0 \\ 1, & if\ q_j < 0 \end{cases}$，如果對於解出來的 vector x 與 parity-check matrix H 相乘，$Hx^T = 0$，則此 vector x 即為解碼後得到的 codeword；若是 $Hx^T \neq 0$，則到 message passing 繼續計算。

每做完一次 Message passing 和 Decision 視為一次 iteration，而當 iteration 的數目超過預期設計好的閾值，則稱為解碼失敗(decoding failure)。

2.1.4 不均等錯誤保護碼(Unequal error protection code，以下簡稱 UEP code)

在 UEP code 中，會將奇偶校驗矩陣(parity-check matrix)設計成特殊架構，而此特殊架構是把 code 分成多個 level，每個 level 對訊息保護程度是不相同的，有些 level 對訊息的保護程度較好，因此可以放入重要的訊息於此段 level 中；而有些 level 則對訊息的保護程度較差，此時放入較不重要的訊息與此段 level，藉由此種方法能夠達到有效分配訊息，並排放重要性不同層級的訊息到不同 level。

2.2 系統設計

2.2.1 模擬方式與流程

此次專題，模擬訊息皆使用 binary (1,0)的形式來描述。奇偶校驗矩陣(Parity-check matrix)則使用 LDPC code 的 Parity-check matrix，通道則使用 BPSK 調變方式加上高斯白雜訊(Additive ehite Gaussian noise，以下簡稱 AWGN)。在模擬中，會設定不同訊雜比(Signal-to-noise，以下簡稱 SNR)來調整標準差(variance)的大小，而 SNR=E_b/N_0，當 SNR 愈大的話 noise variance 會愈小，錯誤率就會隨之變低。

首先，我會先用 C 語言對需實作的奇偶校驗矩陣(Parity-check matrix)進行轉換，然後產生生成矩陣(Generator matrix)，並把生成矩陣的資訊轉成一個檔案輸出。並繼續使用 C 語言進行模擬，會先將奇偶校驗矩陣和生成矩陣資料進行讀取，接著進行 LDPC code 的模擬，本次專題對於整個 LDPC code 的模擬過程為：首先進入到編碼過程(encoding process)，編碼器(encoder)是會有一段原始信息 vector u 和生成矩陣 G 相乘形成一個 codeword，做法為$u \times G = x$，形成一段$1 \times n$的 vector x，為進入到 channel 前的 input 訊號，在此通道會加上 noise $1 \times n$ vector z，於 channel 的 output 得到 vector y，是 vector x 加上 vector z，最後進入到解碼過程(decoding process)，解碼器(decoder)是使用 sum-product algorithm 找出使 $Hx^T = 0$成立的 codeword，對於沒有在設定好的 iteration 次數內找到 $Hx^T = 0$，而稱為一次 decoding failure，而此 block 為 error block。

2.2.2 奇偶校驗矩陣(Parity-check matrix)的架構

此次專題模擬部分分成三個部分：

第一部分是模擬不同大小的 LDPC code，分別(n,k)=(816,408)，R=0.5 的 code 和(n,k)=(8000,4000)，R=0.5 的 code。第二部分是使用一種 UEP QC-LDPC code，稱其為C1，C1 為(7376,3688) code，R = 0.5，有 1/2 是保護程度較好的 level，information bits level-1=1844 ， information bits level-2=1844 ， 其 中 每 一 個 circulant permutation matrix=461 × 461。第三部分是使用一種EEP QC-LDPC code，稱其為C*，C*為(7376,3688) code，R = 0.5，其中每一個 circulant permutation matrix=922 × 922，和第二部分的相同大小的 UEP QC-LDPC code 做比較。

三、實驗結果

在此次專題中，分別對上述三個部份的 code 放入到 LDPC code 的模擬流程圖中，並利用 matlab 畫出 SNR 對 BER 的作圖。

3.1 不同大小的 LDPC code 模擬，n=816 和 n=8000

圖 3.1.1 是對 codeword 長度 n=816 進行模擬，而所畫出 SNR 對 BER 的作圖，此時 SNR 的單位為 dB；圖 3.1.2 是對 codeword 長度 n=8000 進行模擬，而所畫出 SNR 對 BER 的作圖，此時 SNR 的單位亦為 dB。

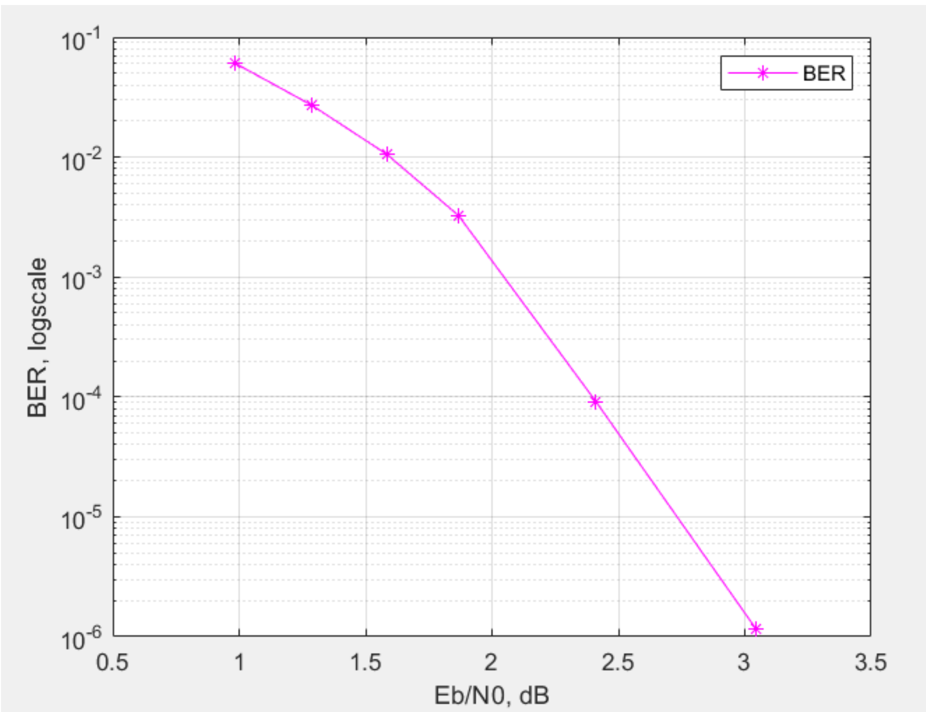

圖 3.1.1

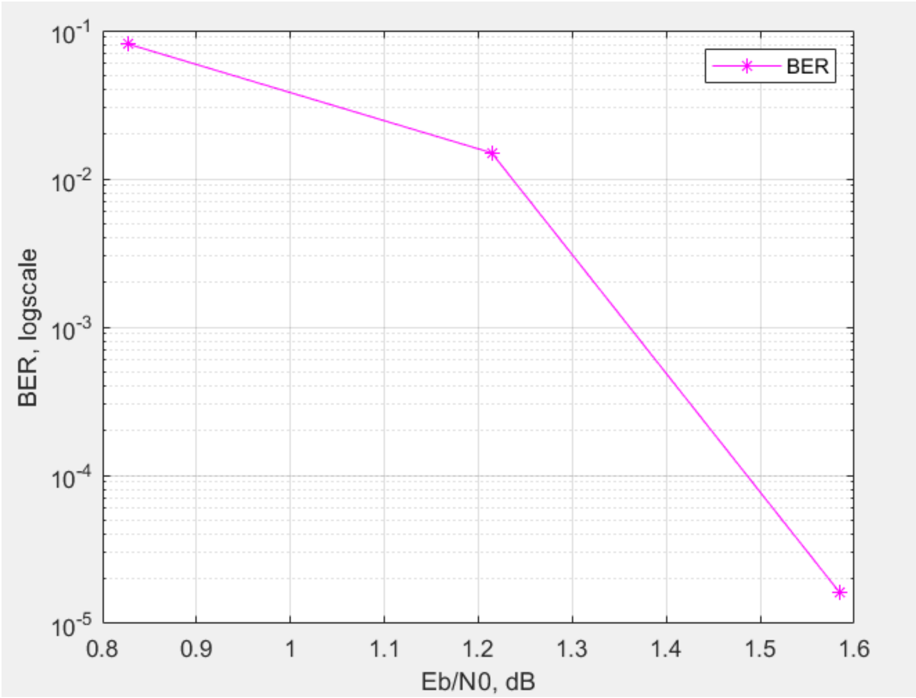

圖 3.1.2

正常情形下，隨著 SNR 的增加，BPSK 通道中的雜訊標準差(noise variance)會變小，而 BER 會隨之下降，此兩圖都能看到兩條曲線皆斜率為負。

3.2 UEP QC-LDPC code 模擬

圖 3.2.1 是對一組 UEP QC-LDPC code 並在 BPSK 通道上進行模擬，而所畫出 SNR 對 BER 的作圖，此時 SNR 的單位為 dB，圖中兩條曲線分別為 level1 和 level2 的 BER。

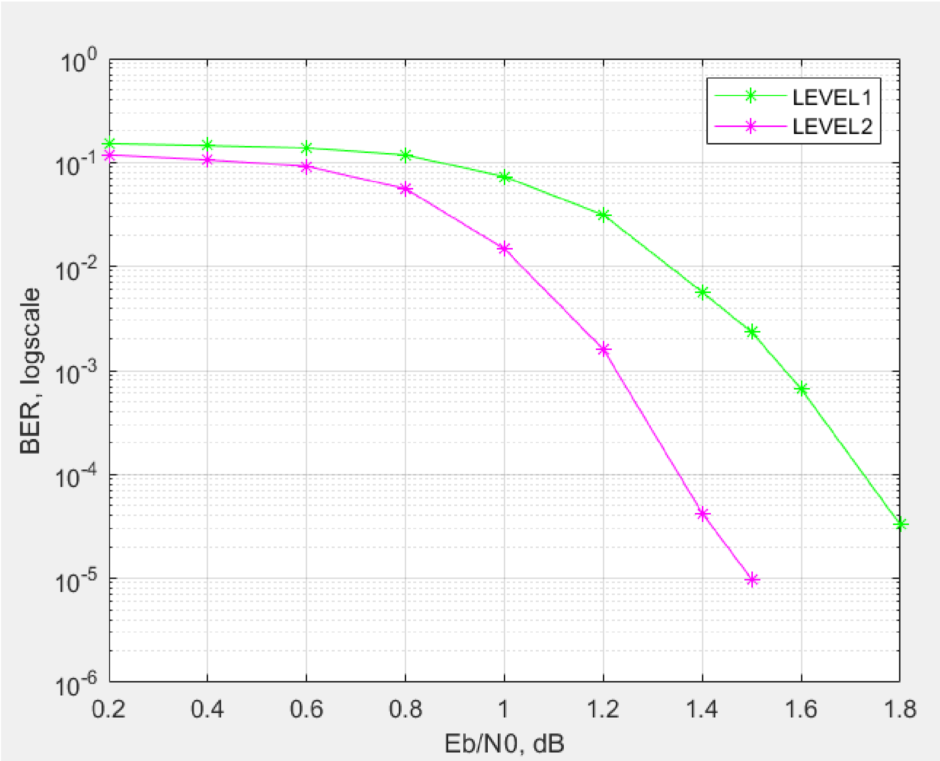

圖 3.2.1

正常情形下，隨著 SNR 的增加，BPSK 通道中的雜訊標準差(noise variance)會變小，而 BER 會隨之下降，此圖能看到兩條曲線(level1 和 level2)皆斜率為負。對於相同 SNR 情形，level2 的 BER 會低於 level1 的 BER，是因為在架構此 UEP code 時，對 level2 的保護程度較對 level1 的保護程度較好，同時也能看到相同的 BER，對 level2 而言所需要的 SNR 較 level1 所需要的 SNR 來的小，可以看到 UEP code 會對 level1 和 level2 進行不同程度上的保護。

3.3 EEP QC-LDPC code (C*)和 UEP QC-LDPC code (C1)的比較

圖 3.3.1 是對一組 EEP QC-LDPC code 並在 BPSK 通道上進行模擬，而所畫出 SNR 對 BER 的作圖，此時 SNR 的單位為 dB，並和 3.2 的 C1 所畫出的 SNR 對 BER 的曲線放在同一張圖上。

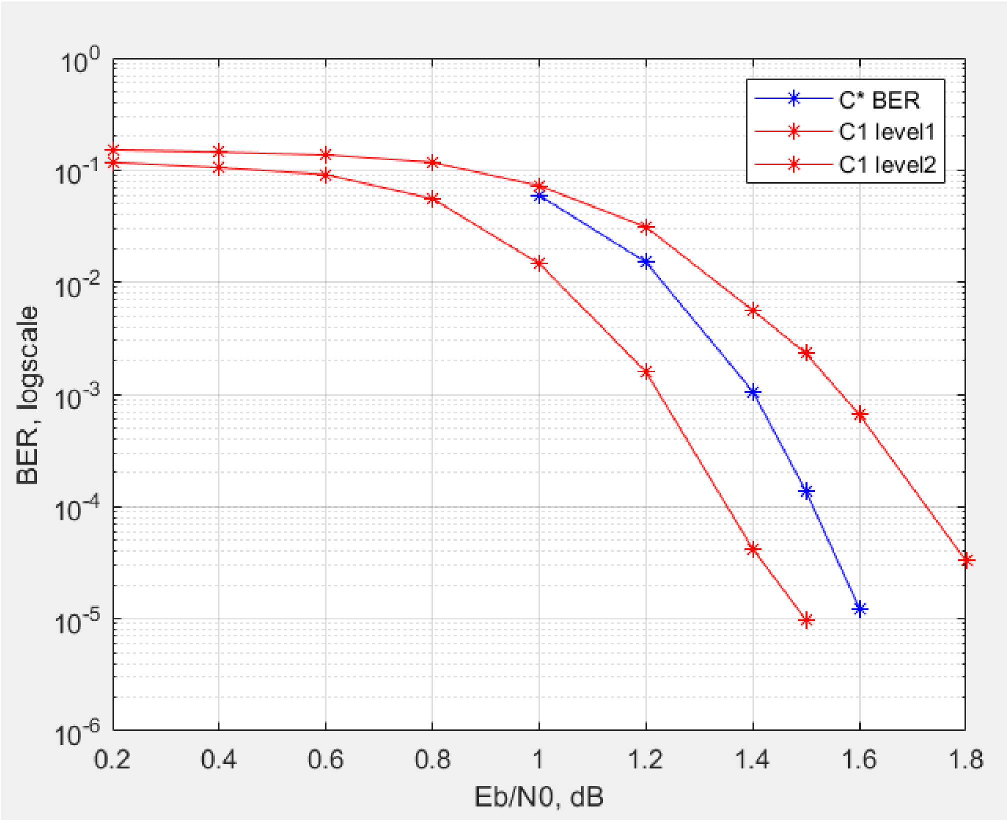

圖 3.3.1

正常情形下，C* code 隨著 SNR 的增加，BPSK 通道中的雜訊標準差(noise variance)會變小，而 BER 會隨之下降，此圖能看到曲線的斜率為負。C1 code 和 C* code 比較，此兩種 code 皆為相同的 code length 和 code rate，對於相同 SNR 下會看到 C1 code 會犧牲一些 level1 的 BER 效能來換取 level2 的 BER 效能，而達到 UEP code 的目的—對不同 level 有不同的保護能力。

(四)結論

這次專題為對 EEP QC-LDPC code 和 UEP QC-LDPC code 進行模擬訊號通過 BPSK 通道後，傳送的訊息能夠得到保護。而 UEP code 和 EEP code 相比是分成幾個 level 進行不同程度的保護，因此當傳遞一段資料可以分那些資料較為重要而使用較好的 level 去保護，並且也能看到隨著 SNR 的上升，BER 會隨之下降。

三、心得感想

首先，在這兩學期中，感謝趙啟超教授在每次 meeting 中給予指導還有實驗室的學長會給予一些問題上的解惑。

一開始選擇做這個專題是因為還沒有特別深入了解整個通訊領域在做甚麼,因此給予我很大的好奇心想藉此次專題可以更早的接觸到這個領域,並且聽完教授講解錯誤更正碼的內容就喜歡這個專題並決定從大三上開始做這項專題,因此得以從大三上開始接觸關於通訊領域中的錯誤更正碼,因此收穫了以前未曾學習過的內容,也能夠從此次專題使用到過去學習過的機率、線性代數和通訊系統的一些知識。

因為這次是一個人進行實作專題,有時候在進行 C 語言模擬時會遇到一些瓶頸,有時無法自己去看到盲點,可能換成兩個人的話會更好,能夠討論各自遇到的問題,並互相幫忙,可能會節省一些尋找問題的時間。