The Comparison of SC Decoding and SCL Decoding of Polar Code

極化碼 SC 及 SCL 解碼之比較

陳永憲

指導教授:翁詠祿

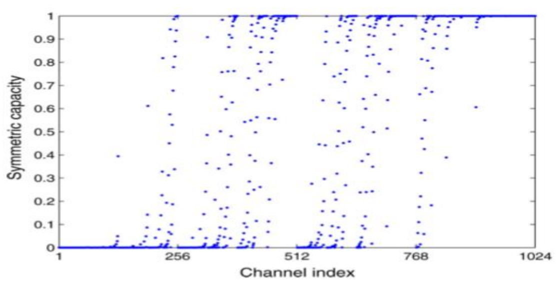
摘要

極化碼是一種用來做通訊傳輸的編碼方法，是第一個可以被證實達成 Shannon capacity 的編碼方式，目前廣泛運用在5G 傳輸方面。由於極化碼構造設計，透過通道極化(channel polarization)，使一部分的通道容量接近1(information bit)，另一部分通道容量接近0(frozen bit)，並利用資訊位元作為傳輸通道。極化碼解碼是透過編碼的結構特性，由 Erdal Arikan 教授提出了 SC 的解碼方式。然而傳統的 Successive Cancelation (SC)解碼，極化碼的性能較其他錯誤更正編碼差，有一種改進的解碼方式 Successive Cancelation List (SCL)，可以改善 SC 解碼，有更好的解碼效能。本專題主要是討論 SC 及 SCL 解碼算法的原理和結構的不同，並比較 SC 及 SCL 在不同的信噪比(SNR)及編碼率(coding rate)的解碼性能。

介紹

在討論編碼前，我們先討論通道極化，以下討論時我們都是用 B-DMC 通道，在還沒通道極化時，B-DMC 通道 W:X→Y，其轉移機率為 W(y|u) ,u∈X ,y∈Y。經過通道極化之後，會得到新的極化信道 $W^{(i)}_N$：$X \rightarrow Y^N \times X^{i-1}$，其轉移機率為 $W^{(i)}_N$（Y_1^N, u_1^{i-1} | u_i）

兩個子通道一個通道容量增加，另一個減少，而隨著碼長的增加，此種極化效果會更加明顯，通道極化會讓一部分的信道容量趨近於1，另一部分的通道容量趨近於0，如下圖所示而我們再利用訊息通道還進行傳輸，即為極化碼編碼的流程。



而極化碼 SC 解碼 可分為 f 函數 及 g 函數，當要對奇數通道解碼時，就須使用 f 函數，當要對偶數通道解碼時，則須使用 g 函數，自然對數似然比的公式如下

f 函數:

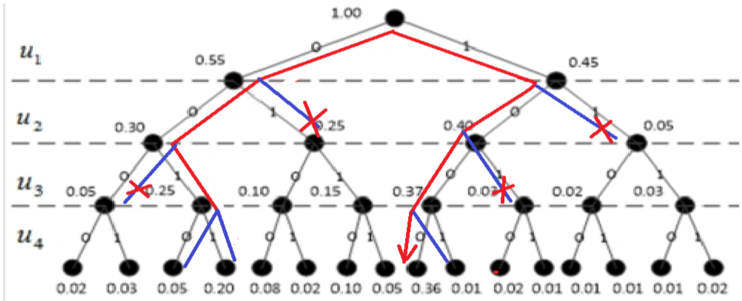
$$L_{2N}^{2i-1}\left(y_1^{2N}, u_1^{2i-2}\right) = \mathrm{sign}(L_1)\mathrm{sign}(L_2)\cdot\min(|L_1|, |L_2|) \quad (4)$$

g 函數:

$$L_{2N}^{2i}\left(y_1^{2N}, \hat{u}_1^{2i-1}\right) = (-1)^{\hat{u}_1^{2i-1}} \cdot L_1 + L_2 \quad (5)$$

如$\hat{u}_i$為訊息位元，且該位元的對數依然比如為正，則判斷$\hat{u}_i$=0，如該對數依然比為負，則判斷$\hat{u}_i$=1；如$\hat{u}_i$為凍結位元，則判斷$\hat{u}_i$=0

由 g 跟 f 函數可以發現 SC 解碼有個最大的缺點就是，前面的解碼結果會影響到後面的解碼，會產生嚴重的錯誤傳遞，並不能有效降低這種錯誤傳遞方生，SCL 解碼，可以大幅補足 SC 解碼所遇到的問題。



SCL 主要是增加每一層搜索路徑保留的數量，如當搜索路徑超過 L 時，會對解碼樹進行剪枝(prune)，並保留轉移概率最大的 L 條路徑，上圖為 SCL 的解碼樹，L=2。

在解碼時，會透過計算各節點的 PM(Path Metrics)值來決定保留的路徑，由於 PM 跟節點的轉移概率成反比，所以在路徑選擇上需要選擇 PM 最小的路徑。

在 SCL 解碼過程中，對於 L 條路徑進行搜索，對於任意一條路經$l \in \{1,\ 2, \dots, L\}$，發送任意位元$u_i (i \in \{1, 2, \dots, N\})$，其 PM 計算定義如下

$$PM_l^{(i)} \triangleq \sum_{j=1}^{i} \ln\left(1 + \exp\left(-(1 - 2\hat{u}_j[l]) \cdot L_N^{(j)}\right)\right) \qquad (6)$$

經過簡化後，可得下列規則：

如$\hat{u}_i[l] = \delta\left(L_N^{(i)}[l]\right)$，則此節點 PM 值就繼承父節點的 PM 值；

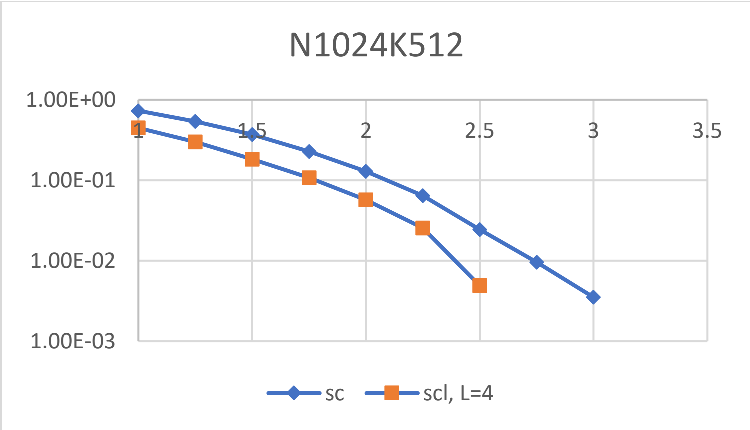
如$\hat{u}_i[l] \neq \delta\left(L_N^{(i)}[l]\right)$，則此節點的 PM 值則為父節點的 PM 值加上此節點 llr 值的絕對值；

如$\hat{u}_i$為凍結位元，$\hat{u}_i[l]$=1，則 PM 值為無限大，其中δ公式如下

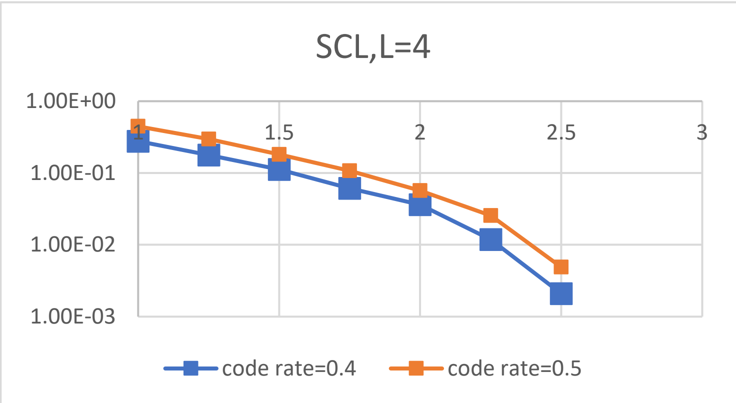
$$\delta(x) = \frac{1}{2}(1 - sign(x)) \qquad (7)$$

實驗方法

利用隨機產生的1和0位元，當作輸入訊號，並在通道中使用random函數模擬雜訊，然後進行編碼及解碼，在解碼完後比較輸出跟輸入是否有錯誤，並記錄錯誤的次數，每筆數據皆為測試十萬筆後的數據。



SC解碼及SCL解碼的性能比較，從圖中可以看出，相較於SC解碼，SCL解碼更具有更低的錯誤率，隨著L的增加，SCL的效能也會有顯著的增加。



不同編碼率的性能比較，從圖中可以看出，對於編碼率較小的解碼，其錯誤率也相對較低，因為編碼率的公式為資訊位元除以所有位元，所以編碼率小也代表在通道極化後，可以選擇容量較接近1的通道，極化較完全的通道，如圖三所示，而中間那些因為碼長有限而通道極化不完全的位元，會優先被排除掉，所以解碼錯誤率自然會降低。

結論

對於不同的參數，我們可以得到不同的錯誤率，而錯誤率的趨勢也跟我們在原理中提到的一致。而從結果數據可以比較出來，SCL的確有改善SC在有限的碼長的解碼中錯誤率過高的問題。而SCL系統複雜及記憶體所需要占用的空間也較大，這也增加了硬體實現的難度，在此本專題就不多加討論硬體實現的問題。