

國立清華大學 電機工程學系

實作專題研究成果摘要

Optimization and Analysis of Schedule
Diversity Decoding for Polar Codes on LDPC-
Like Sparse Graphs

Polar Codes 在 LDPC-like 稀疏圖下的多排
程解碼優化與分析

專題領域： 通訊領域

組 別： B584

指導教授：翁詠祿 教授

組員姓名：郭昱君 陳映彤 梁恩慈

研究期間：2025年1月至2025年11月止，共 10 個月

摘要

本研究旨在探討極化碼(Polar Codes)在 Residual Belief-Propagation(RBP) 解碼下的排程多樣性策略，透過模擬文獻演算法驗證研讀成果並嘗試優化解碼策略。首先，我們透過剪枝(Pruning)技術刪減及合併節點，將原本十分複雜的 Polar Codes 因數圖 (Factor graph)簡化成類似 LDPC Codes 的稀疏圖，以利後續使用 RBP 解碼時，能降低計算複雜度，提升運算效率。

其次，殘差節點化置信度傳播(Node-Wise Residual Belief Propagation, NW-RBP)降低了傳統殘差置信度傳播(Residual Belief Propagation, RBP)之貪婪特性對解碼造成的負面影響，同時，也大幅提升了解碼效率。而改良殘差節點化置信度傳播(Modified NW-RBP)以及以不同解碼策略組合而成的多樣化排程(Schedule Diversity)更是增加了解碼過程的變化性，提高錯誤修正機率。本研究以系統極化碼(1024, 512)作為實驗平台，驗證了 Modified NW-RBP with Schedule Diversity 演算法對於降低錯誤率之效益。

針對解碼過程可能遇到的貪婪群組(Greedy Group)問題，本研究進一步結合 Quota-based RBP(Q-RBP)限制訊息邊更新次數之概念，提出 Quota-based NWRBP(Q-NW-RBP)更新策略，限制 Check Node(CN)在每次解碼時的最大更新次數，以此避免貪婪群組被過度更新、占用大部分解碼資源。在中高信噪比(Signal-to-Noise Rate, SNR) 區域容易產生因貪婪群組造成位元錯誤率(Bit Error Rate, BER)無法順利降低，形成 Error floor 的問題。上述提出的方法能有效減少此現象的發生，讓 BER 降至更低。本研究成果為極化碼解碼器的解碼成效優化提供具體可行之方法與參考依據。

目錄

1. 研究背景及目的.....	1
1.1 研究背景.....	1
1.2 研究目的.....	1
2. 研究方法.....	1
2.1 編碼/解碼之流程與架構.....	1
2.2 剪枝算法實現.....	2
2.3 分層置信度傳播(LAYER BELIEF PROPAGATION, LBP).....	3
2.4 改良殘差節點化置信度傳播(MODIFIED NW-RBP).....	4
2.4.1 殘差置信度傳播(Residual Belief Propagation, RBP).....	4
2.4.2 殘差節點化置信度傳播(Node-wise Residual Belief Propagation, NW-RBP).....	4
2.4.3 改良殘差節點化置信度傳播(Modified NW-RBP).....	5
2.5 排程多樣化(SCHEDULE DIVERSITY).....	6
2.6 文獻驗證結果.....	7
3. 研究結果.....	7
3.1 QUOTA-BASED-NW RBP(Q-NW-RBP).....	7
3.1.1 貪婪群組問題.....	7
3.1.2 Q-NW-RBP 演算法.....	8
3.2 Q-NW-RBP WITH SCHEDULE DIVERSITY.....	9
3.3 最大更新次數 Λ 選擇.....	9
3.4 結果分析與比較.....	9
4. 總結.....	10
5. 參考文獻.....	10
6. 心得感想.....	11

1. 研究背景及目的

1.1 研究背景

Polar Codes 由 Arikan 在2009年提出後，以其通道極化特性成為5G 通訊技術中，主要編碼技術。Polar Codes 目前最常用、且表現最佳的解碼方法為 SCL 解碼。然而，SCL 屬於序列式運算架構，其高記憶體需求不利於高速或低延遲通訊系統。

相較之下，Belief Propagation (BP) 解碼具備高度平行化之優勢，成為解碼的新選擇。然而，Polar Codes 的原生 factor graph 含有大量短迴圈，使得傳統 BP 解碼在有限碼長下存在收斂緩慢、BER 性能不足與錯誤傳遞等問題。

為改善上述問題，文獻[2]提出的剪枝方法，將 Polar code 的 factor graph 轉換為 LDPC-like 的稀疏 Tanner graph，有效降低短迴圈密度，並且近年有許多研究透過不同排程來優化 BP 效能，其中，以殘差(Residual)為導向排成的 Residual Belief Propagation (RBP) 解碼能根據訊息變化量挑選最需更新的節點，在加速局部收斂與提升整體效率方面有顯著效果。因此，本研究以 RBP 為核心概念，並在其基礎上發展多種延伸排程策略。

1.2 研究目的

本研究重於驗證並優化文獻[1]中所提出的 Schedule Diversity 解碼流程。

研究第一階段，我們重現文獻[1]的解碼流程，先將 Polar Code 所對應的 dense H matrix 透過剪枝(pruning)算法轉換為稀疏的 LDPC-like matrix，以降低訊息傳遞過程中的運算複雜度。接著，基於 Layered Belief Propagation (LBP) 與 Modified Node-Wise Residual Belief Propagation (Modified NW-RBP) 的 Schedule Diversity 架構進行解碼，以驗證該流程的效能與可行性上的表現。

研究第二階段，我們在此基礎上，進一步結合文獻[2]提出的 Quota-based Residual Belief Propagation (Q-RBP) 的概念，於 Schedule Diversity 流程中加入對 Check Node (CN) 更新次數的限制。藉由減少因貪婪群組 (Greedy Group) 頻繁更新而造成其他區域錯誤無法有效收斂的現象，我們成功達到優化解碼性能的目的。

2. 研究方法

2.1 編碼/解碼之流程與架構

Low-Density Parity-Check Code (LDPC) 透過稀疏矩陣結構所對應的 Tanner graph，使用 Belief Propagation (BP) 訊息傳遞演算法進行解碼，CN 與 VN 互相傳遞訊息、更新，直到成功解碼或達到最大迭代次數。其錯誤更正性能能夠逼近最大似然 (Maximum

Likelihood, ML) 解碼所能達到的理論最佳值。

LDPC 中對於一個大小為(N, M)的 H 矩陣，有以下編碼流程：

1. Input 一個碼長為 K 的二進位訊息 $u = (u_1, u_2, \dots, u_k)$
2. 用生成矩陣 G 對其進行編碼，由 $x = uG$ ，可得碼長為 N 的碼字 x
3. 將碼字 x 進行二進制相位調變， $s(t) = \begin{cases} 1, & \text{if } x(t) = 0 \\ -1, & \text{if } x(t) = 1 \end{cases}$
4. 加入可加性白高斯雜訊通道 AWGN，產生 Output 為碼字 y

解碼流程如下：

1. 取得碼字 y 後，計算初始 LLR(log-likelihood ratio) 作為 VN 對應的初始訊息 $L_{j \rightarrow i} = L_j$

$$L_j = L(v_j | y_j) = \log \left(\frac{\Pr(v_j = 0 | y_j)}{\Pr(v_j = 1 | y_j)} \right) = \frac{2y_j}{\sigma^2}$$

2. CN update：其中 $L_{i \rightarrow j}$ 為 c_i 傳遞至 v_j 的訊息

$$L_{i \rightarrow j} = 2 \tanh^{-1} \left(\prod_{j' \in N(i) - \{j\}} \tanh \left(\frac{1}{2} L_{j' \rightarrow i} \right) \right)$$

3. VN update：其中 $L_{j \rightarrow i}$ 為 v_j 傳遞至 c_i 的訊息

$$L_{j \rightarrow i} = L_j + \sum_{i' \in N(j) - \{i\}} L_{i' \rightarrow j}$$

4. 重複 update 迭代，並計算碼字的 total LLR

$$L_j^{total} = L_j + \sum_{i \in N(j)} L_{i \rightarrow j}$$

5. 最後得到解碼完的碼字 $v_j = \begin{cases} 1, & \text{if } L_j^{total} < 0 \\ 0, & \text{else} \end{cases}$

6. 透過 $Hv \triangleq 0$ ，可驗證解碼是否成功，若為 0 則解碼成功，停止迭代，若不為 0 則跳回步驟 2. 直到到達最大迭代次數

2.2 剪枝算法實現

由於 Polar code 原本的 BP 解碼圖具有階段式結構，節點數量龐大，且生成的 Parity Check 矩陣(matrix H) 稠密、存在大量短迴圈，直接在此矩陣上執行 SPA/BP 解碼通常會失敗，剪枝的目的是將原本具有多層結構的 Factor Graph，透過以下的剪枝方法化簡為單層 Tanner Graph，並生成稀具有稀疏結構的 LDPC-like 矩陣，以此降低運算複雜度。

文獻[3]提出了一個完整的剪枝方法如下：依據節點連接方式，將以下無效的節點移除或合併：

1. Frozen VN

Frozen bit 的輸入 $LLR = \infty$ ，其訊息不會更新，因此可直接移除。

2. Degree-1 CN

Degree-1 CN 所傳遞的訊息只有在其相連的 VN = bit 0 時才存在，且需 $= \infty$ ，因此該 CN 以及與其相連的皆可移除。

3. Degree-1 VNCH 與 degree-2 CN (如圖1-(a))

若 CN 連接一個 VNCH (有固定 L_{ch}) 與一個 VNH，此 VNH 傳遞的訊息也是來自 VNCH 的 L_{ch} ，因此 CN 可以移除，並將 VNH 由 VNCH 替代。

4. Degree-1 VNH (如圖1-(b))

因其輸入 $LLR = 0$ ，更新後永遠回傳 0，可連同其 CN 一併移除。

5. Degree-2 VNH (如圖1-(c))

Degree-2 VNH 只是簡單的將訊息從 CN1 轉傳至 CN2，無法帶來其他訊息，因此可將其移除，並將兩個 CN 合併。

6. Degree-2 CN (如圖1-(d))

CN 連接到兩個 VNH 無法帶來任何額外訊息，因此可移除此 CN，並合併2個 VNH。

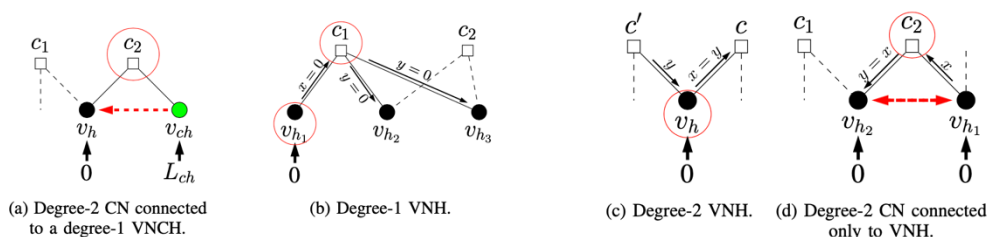


圖1. 無效節點連接情況 (引用自文獻[3])

2.3 分層置信度傳播(Layer Belief Propagation, LBP)

LBP 為 BP 解碼的一種，被廣泛應用在 LDPC 解碼中。它將 parity check 矩陣(即 H 矩陣)的 CN 分成多個層(layers)，一列為一層，每次只更新一層的 CN 節點並立即將結果回傳給 VN，讓下一列的更新得以使用新的資訊，使得資訊更快收斂。相比傳統全圖同時更新的 Flooding BP，LBP 能在相同的迭代次數下達到更低的錯誤率及更快的收斂速度。其演算法如下：

Algorithm 1 : Layer BP

- 1: Initialize all the counter, $I=0$
 - 2: Initialize all $m_{c_j \rightarrow j} = 0$, $m_{v_j \rightarrow c_i} = L_j$
 - 3: **for** every row R_k **do**
 - 4: **for** every $c_i \in R_k$ **do**
-

```

5:         for every  $v_j \in \mathcal{N}(c_i)$  do
6:             Generate and propagate  $m_{c_i \rightarrow v_j}$ 
7:             for every  $c_a \in \mathcal{N}(v_j) \setminus c_i$  do
8:                 Generate and propagate  $m_{v_j \rightarrow c_a}$ 
9:             end for
10:        end for
11:    end for
12: end for
13: if reach stopping rule
14:    break
15: else
16:    Return to Position 3

```

2.4 改良殘差節點化置信度傳播(Modified NW-RBP)

2.4.1 殘差置信度傳播(Residual Belief Propagation, RBP)

過去的 Flooding BP 解碼器為所有 Variable Nodes(VNs)及 Check Nodes(CNs)各同步更新一次，以此完成一輪迭代。固定排程 BP 解碼(Standard Sequential Scheduling, SSS)則是事先預設好 VNs 和 CNs 的更新順序，但以上兩種演算法的收斂速度皆會因此受限。

RBP 則是透過預先計算每條訊息更新時的殘差(Residual)，並選取擁有最大殘差的邊進行更新，而殘差為當前訊息和更新後訊息間的差異量。若有一條訊息 m_k ，其對應的更新函數為 $f_k(\mathbf{m})$ ，則該訊息的殘差定義為：

$$\mathcal{R}_k = |f_k(\mathbf{m}) - m_k|$$

此作法為動態排程(Informed Dynamic Scheduling, IDS)的演算法策略，可以確保每次優先更新圖中收斂程度較低的部分，加速整體的迭代及收斂的速度。

2.4.2 殘差節點化置信度傳播(Node-wise Residual Belief Propagation, NW-RBP)

RBP 會優先更新信度(Reliability)相對最低，也就是錯誤機率最高的 VN，但若該 VN 已經是正確的，再進行傳遞訊息的更新，反而可能造成額外錯誤，需要更多次迭代更新來修正。

文獻[5]中提出 Node-wise RBP 解碼演算法，傳統 RBP 只會更新殘差最大的訊息，Node-wise RBP 則是選出擁有最大訊息殘差邊的 CN，將有連接到該 CN 的 Check-to-Variable 訊息皆更新一遍。其演算法如下：

Algorithm 2 : Node-wise RBP

```

1: Initialize all  $m_{c \rightarrow v} = 0$ 
2: Initialize all  $m_{v_j \rightarrow c_i} = C_j$ 
3: Compute all  $r(m_{c \rightarrow v})$  and generate  $Q$ 
4: Let  $m_{v_j \rightarrow c_i}$  be the first message in  $Q$ 
5: for every  $v_k \in \mathcal{N}(c_i)$  do
6:     Generate and propagate  $m_{c_i \rightarrow v_k}$ 
7:     Set  $r(m_{c_i \rightarrow v_k}) = 0$  and re-order  $Q$ 
8:     for every  $c_a \in \mathcal{N}(v_k) \setminus c_i$  do
9:         Generate and propagate  $m_{v_k \rightarrow c_a}$ 
10:        for every  $v_b \in \mathcal{N}(c_a) \setminus v_k$  do
11:            Compute  $r(m_{c_a \rightarrow v_b})$  and re-order  $Q$ 
12:        end for
13:    end for
14: end for
15: if Stopping rule is not satisfied then
16:     Position=4;
17: end if

```

此方法相較傳統 RBP，是貪婪程度較低的演算法。NW-RBP 和 LBP 更新方式相似，但 LBP 預先決定了更新順序，NW-RBP 則引入了殘差動態排程的概念。

2.4.3 改良殘差節點化置信度傳播(Modified NW-RBP)

基於 NW-RBP 的概念，文獻[1]提出 Modified NW-RBP 演算法，可以自行設定選擇第 α 大的訊息進行更新，增加解碼的彈性。

Algorithm 3 : Modified NW-RBP

```

1: Initialize all  $m_{c \rightarrow v} = 0, m_{v \rightarrow c} = L_v^{ini}$ 
2: Generate all  $m_{c \rightarrow v}^{pre}$ 
3: Set all  $r_{new}(c \rightarrow v) = m_{c \rightarrow v}^{pre}, r_{old}(c \rightarrow v) = 0$ 
4: Compute all residual  $r(c \rightarrow v)$  and find the  $\alpha$ -th largest residual one
5: Set all  $r_{old} = r_{new}$ 
6: Let the edge which connects  $c_i$  and  $v_j$  has the  $\alpha$ -th largest residual value
7: for every  $v_k \in \mathcal{N}(c_i)$  do
    Generate and propagate  $m_{c_i \rightarrow v_k}$ 
    Set  $r(m_{c_i \rightarrow v_k}) = 0$ 
    for every  $c_a \in \mathcal{N}(v_k) \setminus c_i$  do
        Generate and propagate  $m_{v_k \rightarrow c_a}$ 
    end for
    for every  $c_b \in \mathcal{N}(v_k) \setminus c_i$  do

```

```

    for every  $v_a \in \mathcal{N}(c_b) \setminus v_k$  do
        Generate  $r_{new}(c_b \rightarrow v_a) = m_{c_b \rightarrow v_a}^{pre}$ 
    end for
end for
for every  $c_b \in \mathcal{N}(v_k) \setminus c_i$  do
    for every  $v_a \in \mathcal{N}(c_b) \setminus v_k$  do
        Compute all residual  $r(c_b \rightarrow v_a)$ 
    end for
end for
Set all  $r_{old} = r_{new}$ 
end for
8: if all the CNs are satisfied or the maximum iteration number is achieved
    Break
else
    Return to 6.
9: Return the decoded results in the last decoder

```

2.5 排程多樣化(Schedule Diversity)

文獻[1]提出了結合 LBP 及 Modified NW-RBP 多樣化排程的解碼方式，演算法如下。解碼流程的第一步會先計算收到的 Codeword 似然比(L_v^{ini})，並以 LBP 進行解碼，增加與其他排程解碼結果的差異性。若 LBP 解碼不成功，則繼續使用 L_v^{ini} 代入 Modified NW-RBP 進行 $I_{max} - 1$ 次解碼。在解碼的每一輪迭代中，將殘差前 Γ 大的邊所連接的 CNs 放入待選集合中，若其中有訊息連接到同一個 CN，則繼續向下選取，確保每輪有 Γ 個 CNs 可進行選擇。每輪更新時，Modified NW-RBP 中的 α 值皆為隨機從 $1 \sim \Gamma$ 中選取，並不一定是選取殘差最大的邊所連接的 CN，以此增加解碼多樣性。

Algorithm 4 : Modified NW-RBP with Schedule Diversity

- 1: Save all L_v^{ini} values
 - 2: Set the flag PC = false.
 - 3: Set decoding attempt number $i = 1$ and assume the maximum decoding attempt number to be I_{max}
 - 4: Perform the LBP considering all L_v^{ini} values. $i = i + 1$
 - 5: **If** all the CNs are satisfied, set PC = true
 - 6: **While** $i \leq I_{max}$ and PC = false

Perform Modified NW-RBP considering all L_v^{ini} values. However, α is randomly determined from $\mathcal{A} = \{1, 2, \dots, \Gamma\}$ in each update. If more than one edge that has the first Γ largest residual value locates in the same CN, increase the size of \mathcal{A} such that we always have Γ CNs to choose in an update.
-

$i = i + 1.$

If all the CNs are satisfied, set PC = true

End While

7: Return the decoded results in the last decoder.

2.6 文獻驗證結果

本研究使用1024的總碼長，其中包含512位的資訊位(Code rate = 0.5)，並以高斯白噪音(AWGN)作為通道進行模擬。Schedule Diversity 的部分，設定 LBP 進行50次迭代，Modified NW-RBP 總共進行9次解碼，每輪解碼迭代100次。模擬結果和文獻[1]的錯誤率吻合(如圖2)，圖中的黑線為[1]之結果，A 線為 LBP，B 線為 Modified NW-RBP with Schedule Diversity，C 線為傳統 SCL 解碼(L=8)，D 線則是 SCL 解碼(L=32)。而紅、藍兩線則是本研究模擬出的 BER 結果。

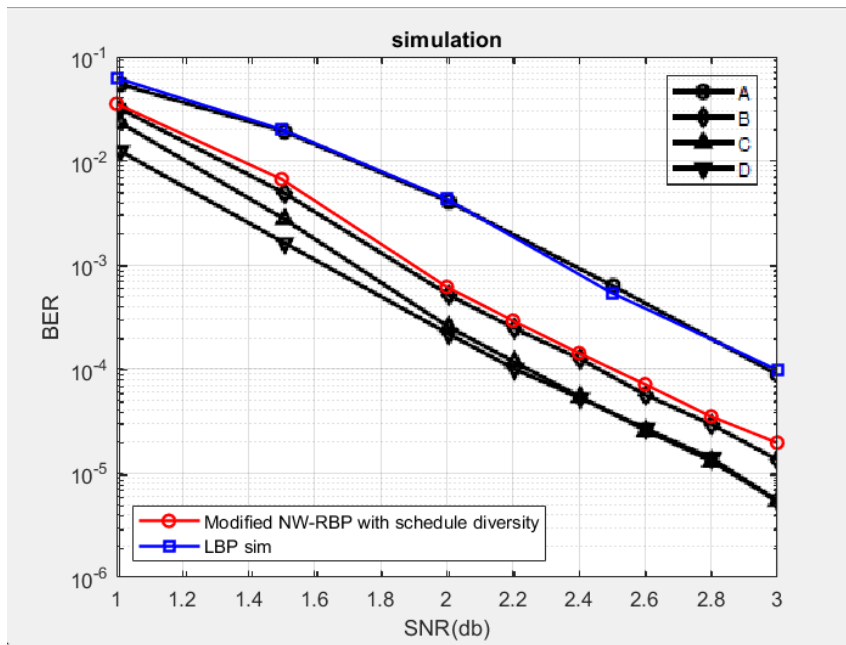


圖2. LBP 及 Modified NW-RBP with Schedule Diversity 模擬結果，(A)LBP, (B)modified NW-RBP with Schedule Diversity, (C)SCL when L=8, (D)SCL when L=32

3. 研究結果

3.1 Quota-Based-NW RBP(Q-NW-RBP)

3.1.1 貪婪群組問題

在 RBP 解碼中，因為某些邊或節點的殘差持續較大，演算法排序時總是擁有更高的優先級，使得更新行為長期集中在同一群節點稱為貪婪群組 (Greedy group)。即使加入排程多樣性的概念，Modified NW-RBP 在 CN 的選擇上仍然受限於某群殘差較大的節點群。這是由於在將 Polar codes 轉換成 LDPC-like 稀疏圖的過程中會產生含有大量短迴圈，導致在更新時殘差會在這些局部迴路中來回傳遞並反覆強化。一旦選中某個少數邊殘差較大的 CN，該 CN 的所有邊，包含不太需要被更新的邊全部都會被更新，反而放

大局部偏差。此外，資訊傳遞具有正回饋效果，更新某些 CN 會使相關 LLR 在下一輪更偏向極值，導致這些前段 CN 的優先級得以維持。綜上所述，排程多樣性的隨機性雖然一定程度上降低了排程的固化性，但殘差整體的排序仍相似，使候選 CN 在每次迭代中重複性仍舊居高不下。

3.1.2 Q-NW-RBP 演算法

基於上述貪婪群組的問題，我們參考文獻[2]提出的 Quota-based RBP(Q-RBP)概念進行改良，提出了 Q-NW-RBP。Q-RBP 以邊為單位限制每條訊息的更新次數，可以有效改善貪婪群組的問題。我們將其結合 Modified NW-RBP 的概念，以節點為單位限制每個 CN 在一次解碼中的更新次數，將 λ 定義為最大更新次數。其演算法如下：

Algorithm 5 : Proposed Q-NW-RBP

- 1: Initialize all $m_{c \rightarrow v} = 0, m_{v \rightarrow c} = L_v^{ini}, \eta_{c_i} = 0$
 - 2: Generate all $m_{c \rightarrow v}^{pre}$
 - 3: Set all $r_{new}(c \rightarrow v) = m_{c \rightarrow v}^{pre}, r_{old}(c \rightarrow v) = 0$
 - 4: Compute all residual $r(c \rightarrow v) = r_{new}(c \rightarrow v) - r_{old}(c \rightarrow v)$
 - 5: Find $r(c_{select} \rightarrow v_{select}) = \alpha$ -th largest $\{r(c_i \rightarrow v_j) | i \in \{1 \dots M\}, v_j \in \mathcal{N}(c_i), \eta_{c_i} < \lambda\}$
 - 6: Set all $r_{old} = r_{new}$
 - 7: **for every** $v_k \in \mathcal{N}(c_{select})$ **do**
 Generate and propagate $m_{c_{select} \rightarrow v_k}$
 Set $r(c_{max} \rightarrow v_k) = 0$
 for every $c_a \in \mathcal{N}(v_k) \setminus c_{select}$ **do**
 Generate and propagate $m_{v_k \rightarrow c_a}$
 end for
 for every $c_b \in \mathcal{N}(v_k) \setminus c_{select}$ **do**
 for every $v_a \in \mathcal{N}(c_b) \setminus v_k$ **do**
 Generate $r_{new}(c_b \rightarrow v_a) = m_{c_b \rightarrow v_a}^{pre}$
 end for
 end for
 for every $c_b \in \mathcal{N}(v_k) \setminus c_{select}$ **do**
 for every $v_a \in \mathcal{N}(c_b) \setminus v_k$ **do**
 Compute all residual $r(c_b \rightarrow v_a)$
 end for
 end for
 Set all $r_{old} = r_{new}$
end for
 - 8: $\eta_{c_{select}} = \eta_{c_{select}} + 1$
-

- 9: **if** all the CNs are satisfied or the maximum iteration number is achieved
 Break
 else
 Return to 5.
- 10: Return the decoded results in the last decoder
-

3.2 Q-NW-RBP with Schedule Diversity

在 Q-NW-RBP 的基礎上，我們延續文獻[1]中提出的排程多樣性做法，將原本 Algorithm 4 中的 Modified NW-RBP 更換成 Q-NW-RBP， \mathcal{A} 集合中可供選擇的 CN 皆須符合尚未達到最大更新次數 λ 的條件。

3.3 最大更新次數 λ 選擇

為了找到 BER 表現最好的 λ ，我們對特定 SNR 下進行每個 CN 在一次迭代中的限制更新次數對 BER 變化的模擬。如圖3所示，我們選擇可以同時反映出低 SNR 和高 SNR 解碼特性的中間 SNR—2.2dB 來實驗。可以看到在 $\lambda < 120$ 時，部分 CN 因為受到更新次數限制而未能將資訊完整傳輸，導致解碼錯誤率上升。在 $\lambda = 120 \sim 150$ 時，可以優於原始演算法的結果，表示限制 CN 更新次數能有效抑制貪婪群組的發生，使解碼更穩定且更具全域性。若是再放寬對 CN 的限制，效能改善幅度降低，顯示過大的 λ 會使 Q-NW-RBP 的行為逐漸接近 Modified NW-RBP，因而失去 Quota 控制帶來的益處。

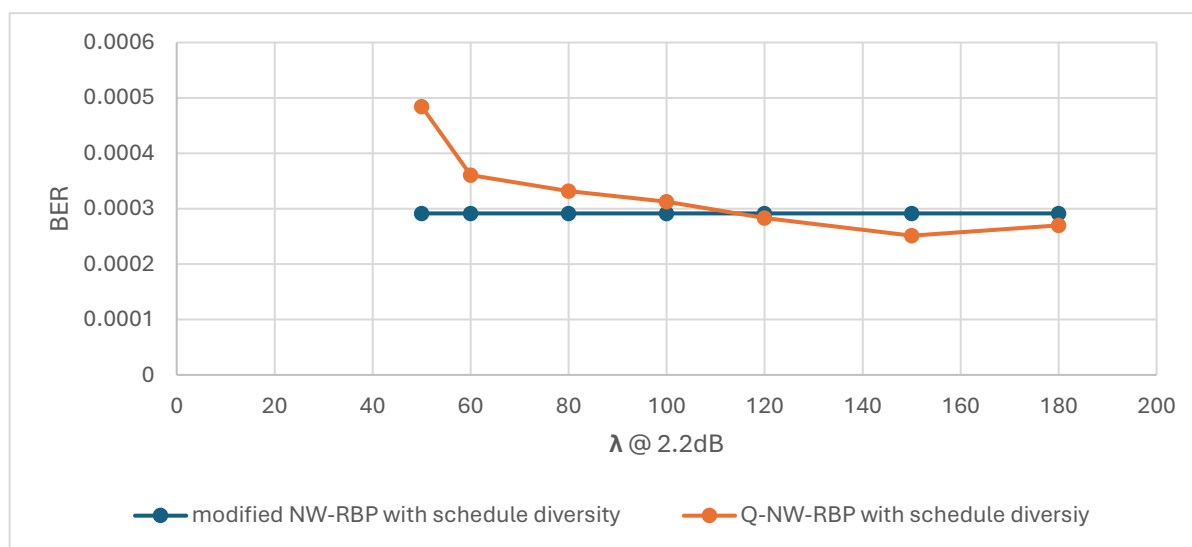


圖3. Q-NW-RBP with Schedule Diversity 在2.2dB 下的 CN 限制次數表現圖

3.4 結果分析與比較

我們將3.3得到的最佳 λ 帶入 Q-NW-RBP with Schedule Diversity 會得到如圖4.的結果。在低 SNR 下，解碼過程由通道雜訊主導，所有初始 LLR 都極度不可靠，導致大部分 CN 的殘差都偏大且差異不明顯，演算法必須對大量 CN 進行頻繁更新才能開始收斂，因此加入 λ 實際上無法有效區分「應該優先更新」與「不需要頻繁更新」的節點。

鑑於以上原因，我們無法在圖4.中看出低 SNR 區域明顯的效能優化。但在高 SNR 區域，BER 表現有較大幅度的提升。這是因為在 SNR 區域，通道提供的 LLR 已經相當可靠，解碼過程受到隨機通道雜訊的影響較小。BER 表現受限於 Modified NW-RBP 本身的過度修正與貪婪群組問題。在原始的演算法中，某些殘差較大的 CN 會在短迴路的正回饋作用下被反覆選中並過度更新，使相關的 LLR 在迭代間不斷被放大，導致訊息偏壓集中，造成局部震盪、收斂不穩定、甚至 Error floor 上升。因此在加入 λ 限制一次迭代中的 CN 更新次數後可以抑制高頻更新的 CN，讓其餘 CN 有更新的機會，整張二分圖的傳遞更加均勻，避免局部 LLR 的過度放大，也減少了解碼資源過度集中於特定區域的問題。

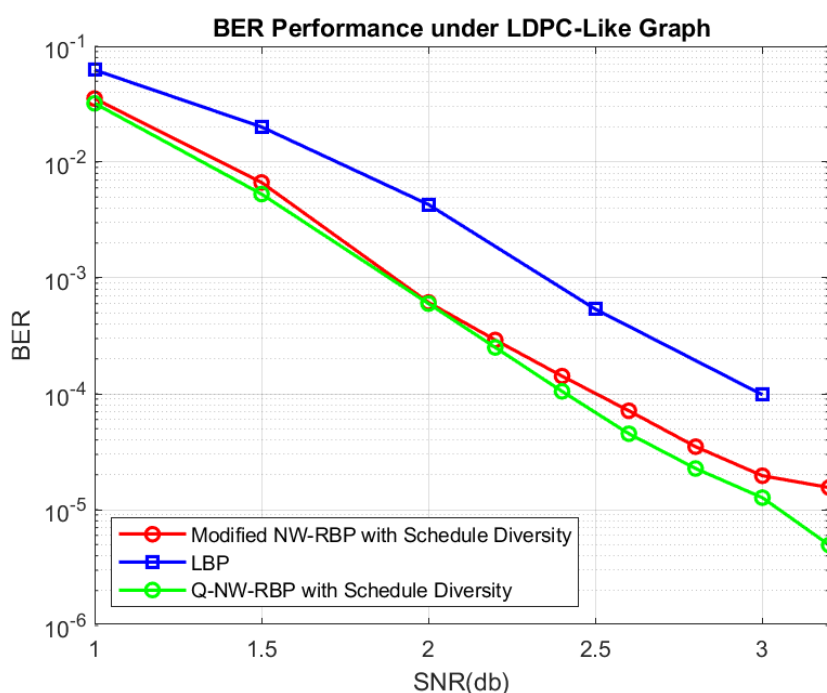


圖4. Q-NW-RBP with Schedule Diversity 與其他演算法的性能比較

4. 總結

本研究基於 Polar Codes 在 LDPC-like 稀疏圖下的多排程解碼，結合 LBP 與 Modified NW-RBP 執行 Schedule Diversity 演算法，驗證文獻[1]所提出的解碼流程，並設計Q-NW-RBP，解決因短迴圈產生貪婪群組導致錯誤無法更正的問題。

研究結果顯示，第一階段，LBP 與 Modified NW-RBP 執行 Schedule Diversity演算法，相較於單純的LBP解碼，整體的錯誤率表現明顯提升，且在高SNR區域較傳統 SCL 解碼器僅有小幅的錯誤率表現差距，卻有顯著減少運算次數與較低的記憶體需求等優勢。第二階段，在加入Q-NW-RBP的概念後，在低SNR區保持一樣表現，而在高SNR區時，Trapping set 造成的錯誤被改善，Error floor的現象減少，因此錯誤率表現相較第一階段進一步優化。

5. 參考文獻

- [1] Yen-Ming Chen, Yong-Jan Yang, Bo-Lun Huang, Huang-Chang Lee, and Chih-Peng Li, “Polar Decoding with Schedule Diversity Based on LDPC-like Sparse Graphs”, 2020 IEEE International Symposium on Broadband Multimedia Systems and Broadcasting (BMSB)
- [2] Huang-Chang Lee, Yeong-Luh Ueng, Shan-Ming Yeh, and Wen-Yen Weng, “Two Informed Dynamic Scheduling Strategies for Iterative LDPC Decoders” , IEEE Transaction in Communication, Vol.61, NO.3, MARCH 2013e
- [3] Sebastian Cammerer, Moustafa Ebada, Ahmed Elkelesh and Stephan ten Brink, “Sparse Graphs for Belief Propagation Decoding of Polar Codes”, 2018 IEEE International Symposium on Information Theory (ISIT)
- [4] Dale E. Hocevar, “A reduced complexity decoder architecture via layered decoding of LDPC codes”, IEEE Workshop on Signal Processing Systems, 2004. SIPS 2004.
- [5] A. I. Vila Casado, M. Griot, and R. Wesel, “Informed dynamic scheduling for belief-propagation decoding of LDPC codes,” in Proc. 2007 IEEE International Conference on Communications (IEEE ICC), Glasgow, Scotland, June 2007.

6. 心得感想

透過此專題，我們從通訊領域中最基礎的 BPSK 開始學習，了解何謂編碼、解碼以及模擬資訊傳遞的流程。接著，接觸更加複雜的 LDPC codes 以及 BP、RBP 和其衍生出的多種變化演算法。最後，學習將 5G 通訊中最常使用的 Polar codes 透過剪枝技術轉換成 LDPC-like 的圖，並以 LBP、NW-RBP 和 Schedule Diversity 等不同解碼方式進行模擬，觀察其錯誤率。

我們原本對通訊領域十分陌生，但經歷了循序漸進的學習後，對於通訊的理論、實務上都有更深刻的了解。同時，我們也學會如何在有疑問的時候，尋找相關資源解決問題，並且透過研讀文獻、與老師及學長姊和同學相互討論，思考如何在原本的知識基礎上，加入更多創新的想法。

過程中藉由程式模擬來驗證文獻中提出的演算法，同時，對我們創新的部分不斷進行測試與改良。從成功的結果中獲得成就感，也從失敗的過程裡學會發現問題、解決困境。除此之外，也練習如何進行分工、彼此合作，並且合理的安排計畫的進行、控管進度。綜上所述，專題是一個很好的學習經驗，不僅使我們更了解通訊領域的實務內容，也從中學習到許多學術研究上的重要技能。