

國立交通大學

電信工程研究所

碩士論文



基於多層級分散式訊源編碼的壓縮傳遞系統

**A Study of Multi-level Distributed Source Coding for
Compress-and-forward Relaying Systems**

研究生：許晉瑋

指導教授：張文輝 博士

中華民國一百零二年六月

基於多層級分散式訊源編碼的壓縮傳遞系統
A Study of Multi-level Distributed Source Coding for
Compress-and-forward Relaying Systems

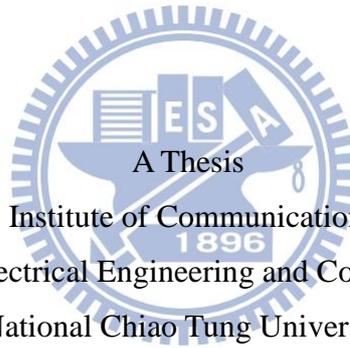
研究生：許晉瑋

Student: Chin-Wei Hsu

指導教授：張文輝

Advisor: Wen-Whei Chang

國立交通大學
電信工程研究所
碩士論文



Submitted to Institute of Communications Engineering
College of Electrical Engineering and Computer Science
National Chiao Tung University
in Partial Fulfillment of the Requirements
for the Degree of
Master
in

Communication Engineering

June 2013

Hsinchu, Taiwan, Republic of China

中華民國一百零二年六月

基於多層級分散式訊源編碼的壓縮傳遞系統

研究生：許晉璋

指導教授：張文輝 博士

國立交通大學
電信工程研究所

中文摘要

本論文針對合作式通訊在低通道訊雜比的應用環境下提出一個新的中繼端傳輸方案。中繼端利用純量量化器處理源自來源端的接收信號，再利用分散式訊源編碼的同位子法以及多層級編碼處理後，經由通道碼保護送出。目的端收到來源端與中繼端的訊號後，先還原其量化索引資訊，再與來源端直接鏈結路徑的接收信號合作解碼。模擬結果顯示在低通道訊雜比時，基於渦輪碼原則的疊代訊源通道解碼能有效還原中繼端的重建訊號。

關鍵字：合作式通訊、壓縮傳遞、分散式訊源編碼、同位子法、多層級編碼。



A study of Multi-level Distributed Source Coding for Compress-and-forward Relaying Systems

Student: Chin-Wei Hsu

Advisor: Dr. Wen-Whei Chang

Institute of Communications Engineering,

National Chiao Tung University

Hsinchu, Taiwan

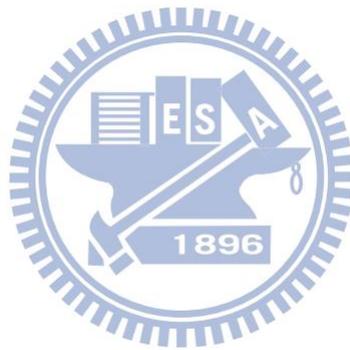
Abstract

In this thesis, we propose a new information relaying approach for cooperative communication with low source-to-destination channel SNR. The relay first performs scalar quantization on received source signal, and then jointly apply the parity approach of distributed source coding and multi-level coding prior to the transmission to destination with channel code. The destination estimates the received relay signal by a joint decoding of the signals from both the source node and relay node, followed by decoding of the source message using both the received source signal and reconstruction relay signal. Also proposed is a signal estimation scheme based on the turbo-like iterative source-channel decoding. Simulation results indicate that proposed relaying system yields better performances when source-to-destination channel SNR is low.

Keywords: cooperative communication, compress-and-forward, distributed source coding, parity approach, multi-level coding.

誌謝

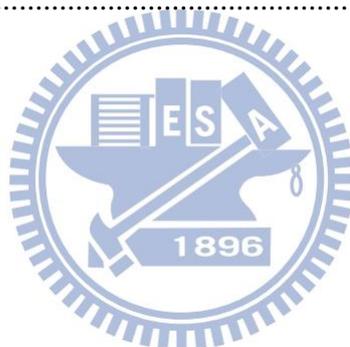
本篇論文的完成，首先要感謝我的指導教授張文輝老師這兩年來的悉心指導及指正，使我了解到作研究時該有的方向、想法以及態度。同時也要感謝實驗室的大家，博班學長吳鴻材，總是能告訴我問題點在哪以及該朝哪方面去做。已經畢業的學長林志賢、王韋超、曾鴻竣、蔡維盈，同屆的曾怡華、黎忠孝，還有學弟李育璋、呂柏廣、吳宜庭、潘建豪，大學同學兼研究所室友的劉哲璋、許智皓。大家總能一起討論、分享、以及提供意見，最後要特別感謝我的父母，不辭辛勞地供給我讀書，讓我能無後顧之憂，順利的作研究，謝謝你們！



目錄

中文摘要	i
Abstract.....	ii
誌謝	iii
目錄	iv
表目錄	vi
圖目錄	vii
第 1 章 緒論.....	1
1.1 研究動機與方向	1
1.2 章節概要	5
第 2 章 壓縮傳遞合作式通訊機制.....	6
2.1 系統傳輸模型	6
2.2 目的端前級解碼與效能評估	8
2.3 目的端的後級解碼	10
2.3.1 BCJR 演算法	10
2.3.2 結合多重資訊的 BCJR 演算法	12
2.4 頻寬擴展	13
第 3 章 基於單一同位檢查碼的編碼機制.....	14
3.1 中繼端傳輸架構	14
3.2 目的端解碼架構	15
3.3 BCJR 通道解碼演算法	17
3.4 軟式位元訊源解碼演算法	19
第 4 章 基於多層級分散式訊源編碼的壓縮傳遞機制	21
4.1 中繼端傳輸架構	21
4.2 目的端解碼架構	22
4.3 BCJR 通道解碼演算法	24
4.4 軟式位元訊源解碼演算法	26
4.5 分散式訊源編碼理論	27
4.6 基於多層級同位子法的編碼機制	29

4.6.1	中繼端傳輸架構.....	29
4.6.2	目的端解碼架構.....	30
4.6.3	BCJR 通道解碼演算法	32
4.6.4	基於多層級編碼的 BCJR 訊源解碼演算法	34
第 5 章	實驗模擬與結果分析.....	37
5.1	環境設定	37
5.2	實驗模擬	38
5.2.1	基於單一同位檢查碼的中繼端解碼機制.....	38
5.2.2	基於平行串接架構的中繼端解碼機制.....	41
5.2.3	基於多層級同位子法編碼的解碼機制.....	44
5.3	綜合討論	47
第 6 章	結論與未來展望.....	49
	參考文獻.....	50



表目錄

表 5.1	基於單一同位檢查碼的重建 SNR.....	39
表 5.2	基於單一同位檢查碼的位元錯誤率	39
表 5.3	基於平行串接架構的重建 SNR.....	42
表 5.4	基於平行串接架構的位元錯誤率	42
表 5.5	基於多層級同位子法的重建 SNR.....	45
表 5.6	基於多層級同位子法的位元錯誤率	45



圖目錄

圖 1.1	合作式通訊概念圖[2].....	2
圖 2.1	系統傳輸模型[15].....	6
圖 2.2	基於觀察點的壓縮[15].....	7
圖 2.3	基於對數相似比率的壓縮[15].....	7
圖 2.4	目的端解碼架構.....	9
圖 2.5	迴旋碼之位元層級籬柵圖.....	10
圖 3.1	中繼端傳輸架構.....	14
圖 3.2	目的端解碼架構.....	15
圖 3.3	AWGN 的量化輸出通道[22].....	20
圖 4.1	中繼端傳輸架構.....	21
圖 4.2	目的端解碼架構.....	22
圖 4.3	相關訊源的無失真編碼流程[18].....	28
圖 4.4	分散式訊源編碼之碼率範圍[18].....	28
圖 4.5	中繼端傳輸架構.....	29
圖 4.6	目的端解碼架構.....	30
圖 5.1	基於單一同位檢查碼的重建 SNR.....	40
圖 5.2	基於單一同位檢查碼解碼機制的模擬結果.....	40
圖 5.3	基於平行串接架構的重建 SNR.....	43
圖 5.4	基於平行串接架構解碼機制的模擬結果.....	43
圖 5.5	基於多層級同位子法的重建 SNR.....	46
圖 5.6	基於多層級同位子法解碼機制的模擬結果.....	46
圖 5.7	量化精度 $m = 1$ 的位元錯誤率.....	48
圖 5.8	量化精度 $m = 2$ 的位元錯誤率.....	48

第1章 緒論

1.1 研究動機與方向

由於科技日新月異，近年來人們大量使用無線通訊網路。且建造無線網路不像有線網路有其地形與電纜配置的限制，所以無線通訊得到廣泛的應用。無論在食、衣、住、行、育樂皆享受到無線通訊所帶來的好處，例如無線上網、影片、語音、音樂等。使用者越來越注重網路的覆蓋率、聲音及影像的品質。因此，一個穩定高品質的無線通訊網路是智慧化生活不可或缺的標準配備。

無線通訊仰賴無線傳輸來達成，例如衛星通訊、行動通訊、無線區域網路等，藉著電磁波將訊息從來源端(Source node)傳送至目的端(Destination node)。由於電磁波傳輸時並不都是直視性的(line-of-sight)，傳輸的過程中會碰到很多障礙物體，造成反射、散射及繞射後到達目的端，我們稱為多重路徑(multipath)。多重路徑會讓目的端收到很多個信號，其強度、相位與到達時間都不同，此即多重路徑衰減(multipath fading)。多樣性(diversity)即是無線通訊系統為了克服多重路徑衰減所使用的重要技術，主要概念是讓信號通過多個獨立的通道傳輸，來提高目的端的收訊效能。典型的多樣技術有三個：時間多樣(time diversity)、頻率多樣(frequency diversity)以及空間多樣(space diversity)。時間多樣是將一個信號於不同時間重覆傳送的多樣技術，可以克服都卜勒效應(Doppler effect)引起的衰減。頻率多樣是將一個信號於不同頻率下傳送並且在目的端合成或選擇的多樣技術，可以使信號傳送時的衰減互相獨立。空間多樣，又稱作天線多樣(antenna diversity)，是在來源端與(或)目的端裝置多根不同方向的天線的多樣技術，亦稱為多輸入輸出系統(multiple-input multiple output, MIMO)。如果天線之間的距離夠大，則信號受到衰減的影響就越小且越獨立。由於空間多樣是將傳送的電磁波分散在不同方向的空間，所以不需要額外使用時間與頻率的資源來達到多樣性，是目前最常用的技術。

[1]裡提到，使用空間多樣的 MIMO 系統近年來被廣泛的研究與探討，並已證明此系統可以有效增加多樣性與頻譜效益。但是在某些實際的通訊環境裡，MIMO 系統無法充分達到應有的效能，例如大樓的通訊死角、隧道內及細胞邊緣(使用者離基地台最遠處)等，使得訊號的接收嚴重不良。且由於通訊裝置大小的要求以及成本上的考量也會限制接收的效能，例如傳輸或接收天線增加的時候，天線的間距可能要縮短，將使得天線間之相關性增加，降低多樣性的程度。合作式通訊(cooperative communication) 是近

年來被提出的一種傳輸方式，可以解決因上述問題而降低的多樣性。主要的概念是透過中繼端(relay node)協助來源端至目的端之間的傳輸，如圖 1.1 所示。

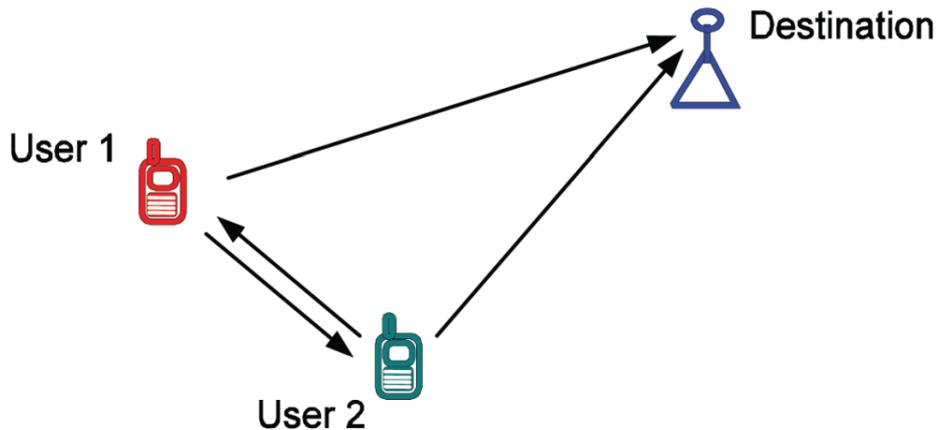


圖 1.1 合作式通訊概念圖[2]

其中來源端至目的端又稱為直接鏈結(direct link)，來源端至中繼端再到目的端則稱為中繼鏈結(relay link)。讓原本的信號不只透過直接鏈結，也透過中繼鏈結來幫助傳輸，改善單一鏈結的收訊問題。而且這些中繼端在空間中是分散配置，通道間的統計特性各自獨立，所以具有分散的空間多樣，又稱分散式空間多樣。就算各端點只有裝置單根天線，整體仍可以等校成 MIMO 系統的效能。

另外，[3]提到有關合作式通訊的主要技術起源於 Cover 與 El Gamal 在[4]探討三端網路(來源端、中繼端、目的端)模型的消息理論特性。他們的研究證明了在離散無記憶且附加性白色高斯雜訊(additive white Gaussian noise, AWGN)通道中，中繼端的通道容量大於來源端至目的端的通道容量，他們也探討了關於多個中繼端的架構。而且透過三種不同的隨機編碼方式可以得到中繼通道容量的下限值，分別為：

1. 簡易方法：中繼端不主動幫助來源端，是透過盡量減少干擾來源端來提高目的端的品質與可靠度。
2. 協同方法：中繼端接收到信號後先解回來源端的訊息，然後重新送至目的端。
3. 觀察方法：中繼端對來源端的信號的量化形式編碼後再傳至目的端。

根據這三種方法，合作式通訊有三種訊號處理方式。分別是：

1. 放大傳遞(amplify-and-forward)：放大傳遞是將中繼端收到的來源端發送信號直接放大後傳到目的端，目的端再將收到的兩條鏈結訊號合併處理，所以放大傳

遞是一種類比信號的處理方式。但是在放大信號的同時也會將雜訊放大，會導致雜訊傳播的問題。

2. 解碼傳遞(decode-and-forward):解碼傳遞是將中繼端收到的來源端發送信號進行解碼，若能正確無誤解出來源端的信息後再傳到目的端，是一種數位信號處理的方式。但如果來源端至中繼端之間的通道狀況不好時，會造成解碼錯誤導致錯誤傳播(error propagation)。
3. 壓縮傳遞(compress-and-forward):壓縮傳遞是介於放大傳遞與解碼傳遞之間的形式，將中繼端收到的來源端發送信號利用某種壓縮技術將接收信號壓縮、保護後送出。

而有關合作式通訊的傳輸模式，一般而言分為兩種，第一種為全雙工(full-duplex)，第二種為半雙工(half-duplex)。當在全雙工的模式下，中繼端可以同時傳送與接收，在半雙工的模式下，中繼端同時只能執行傳送或接收信號。雖然在全雙工模式時對於頻譜得使用率較好，但是在同時傳送跟接收的情形下，信號會有自我干擾的情形，因此皆以討論半雙工的模式居多。大部分的研究都在討論解碼傳遞與放大傳遞的實現，而壓縮傳遞的討論相對較少，例如[5]、[6]、[7]。所以我們將針對壓縮傳遞的合作式通訊進行討論，前面提到壓縮傳遞是中繼端將收到的來源端發送信號利用某種壓縮技術將接收信號壓縮、保護後送出。對於壓縮方面，最常使用純量量化(scalar quantization, SQ)，好處是簡單且無記憶的特性。主要概念是找到對應於輸入的分區(partition)與重建值(reconstruct value)，來最小化輸入與重建值間的均方誤差(mean square error)。保護方面，對於量化後的索引(index)我們會再通過額外的訊源編碼器(source encoder)與通道編碼器(channel encoder)以對抗通道雜訊的干擾。

傳統設計通訊系統時，訊源編碼與通道編碼這兩部分是各自獨立設計及運作的。訊源編碼著重在資料壓縮，壓縮過的資料要能越小越好，視情況允許能否有失真。通道編碼則著重在保護資料，以對抗通訊過程中信號因雜訊的干擾所造成的錯誤。這概念是源自於沈農(Shannon)的消息理論，主要是說當我們在設計某個編碼器時，已經假設其他編碼器的狀態是最佳化了。但如果考慮到實際的通訊環境下，信號的延遲及系統複雜度的限制會導致實現的困難。不過根據前人的實驗，觀察出當訊號經過訊源編碼器處理後，其序列間會存在某種形式的分均勻機率分布或隱含著相關性，這些資訊我們稱為殘存冗息(residual redundancy)。若能運用殘存冗息，可以在不提高通道頻寬的情形下，提供一些事前資訊(a priori information)給通道編碼使用，提高通道解碼的正確度。若目的端將

訊源解碼與通道解碼合併在一起考慮的話，我們稱為合併訊源通道解碼(joint source-channel decoding, JSCD)。大致可以分為三個種類：錯誤隱藏(error concealment)、訊源控制通道解碼(source-controlled channel decoding, SCCD)以及疊代訊源通道解碼(iterative source-channel decoding, ISCD)。錯誤隱藏部分，在[8]提到殘餘冗息可以讓訊源解碼器隱藏通道解碼器無法消除的殘餘錯誤，也稱為軟式訊源解碼(soft-bit source decoding, SBSB)。而訊源控制通道解碼由[9]提出，嘗試將殘餘冗息整合進通道解碼過程裡。不同於以往的通道解碼，訊源控制通道解碼的輸入部分包括了事前機率，大多來自事先統計好的訊源分佈資訊。疊代訊源通道解碼則是結合了以上兩個的優點，比單獨使用其中一種可以達到更好的效能。

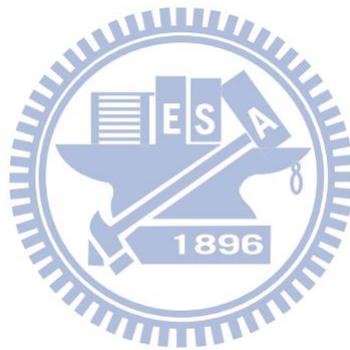
疊代訊源通道解碼參考了渦輪碼(turbo code)[10][11]解碼的模式，渦輪碼是近代通道解碼的重要突破，它由兩個迴旋編碼器(convolutional encoder)與一個交錯器(interleaver)組成。目的端將兩個解碼器產生出來的額外訊息(extrinsic information)互相交換給另一個解碼器作為事前訊息使用，在多次疊代的情況下其錯誤率得到很好的結果，遠超過其他編碼方式，因此得到廣泛的注目與發展。在疊代訊源解碼中，是將訊源解碼器與通道解碼器結合，利用訊源的殘餘冗息與通道解碼的錯誤更正能力，可以有效提升訊源重建的品質。

另外在合作式通訊中，也可以應用分散式訊源編碼(distributed source coding, DSC)。分散式訊源編碼起源於 1970 年代 Slepian 及 Wolf 針對兩相關訊源提出的無失真壓縮理論[12]，對於兩個有相關性的訊源，可藉著資源共享的合併編碼(joint encoding)的方式降低其理論熵(entropy)值。重要的是，即使在獨立編碼(separate encoding)的情況下，仍然可以合併解碼的方式得到相同的理論熵值。對於分散式訊源編碼的做法，一般而言有兩種：校驗子法(syndrome approach)與同位子法(parity approach)。校驗子法是在 2003 年由 Pradhan 和 Ramachandran 運用通道編碼理論的碼分級校驗子觀念提出的。將兩個訊源的位元序列分成固定長度的區塊，其中一個訊源經過通道編碼處理後送出其碼組(coset)的校驗子。另一個訊源則不進行處理，被視為目的端的邊訊息(side information)。而最重要的是必須事先建立好用來描述兩訊源間相關性的數學模型，通常兩訊源的相關性是以虛擬的二位元對稱通道(binary symmetric channel)來描述。而同位子法也是透過通道編碼的方式來達成，例如一個碼率為 $1/2$ 的系統性迴旋碼(recursive systematic convolutional code, RSCC)，它會輸出原本的輸入訊息位元以及具保護能力的同位位元(parity bits)，而同位子法則是只送出同位位元的部分，解碼端則利用收到的同位位元以及邊訊息來解碼。同

位子法的好處是做法較為簡單而且無須更改原本的通道解碼演算法。而在合作式通訊裡我們可以分別將來源端送至中繼端處理的資訊運用校驗子法或同位子法處理，而來源端至目的端的訊號則視為邊訊息，之後目的端再根據收到的信號還原出原始來源端的信息。

1.2 章節概要

本文將在第二章介紹系統的傳輸模型與壓縮策略，並說明如何評估解碼效能以及通道解碼所使用的演算法。第三章介紹前人提出的中繼端訊源編碼機制。第四章將介紹如何將分散式訊源編碼的作法應用於中繼端的傳輸方案中，進一步引入了分層編碼的概念，並說明解碼演算法如何運作。第五章進行實驗模擬與結果分析。最後第六章為結論與未來展望。



第2章 壓縮傳遞合作式通訊機制

關於壓縮傳遞的具體做法大致分為兩種，一為基於觀察點的壓縮，二為基於對數相似比率(log-likelihood ratio)的壓縮。經由分析得知，第一種做法可以取得近似分散式天線系統(distributed antenna system, DAS)的效能，DAS 是一種空間分離的概念，其多根天線皆連至同一接收端。根據第一種做法，利用聯合訊源通道編碼來處理信號，由於處理後的輸出位元多於輸入，我們運用了頻寬擴展(bandwidth expansion)的概念傳輸至目的端。而目的端則運用渦輪原則的疊代訊源通道解碼原出中繼端的接收信號，最後一步是結合中繼端的還原信號以及直接鏈結路徑的信號兩個資訊解回信息。本章節將分以下幾節介紹，第一節為整體系統的架構與壓縮傳遞的做法及分析，第二節為目的端前級解碼與疊代解碼的效能評估，第三節介紹目的端的後級解碼如何結合兩個路徑的資訊，最後第四節說明頻寬擴展的概念。

2.1 系統傳輸模型

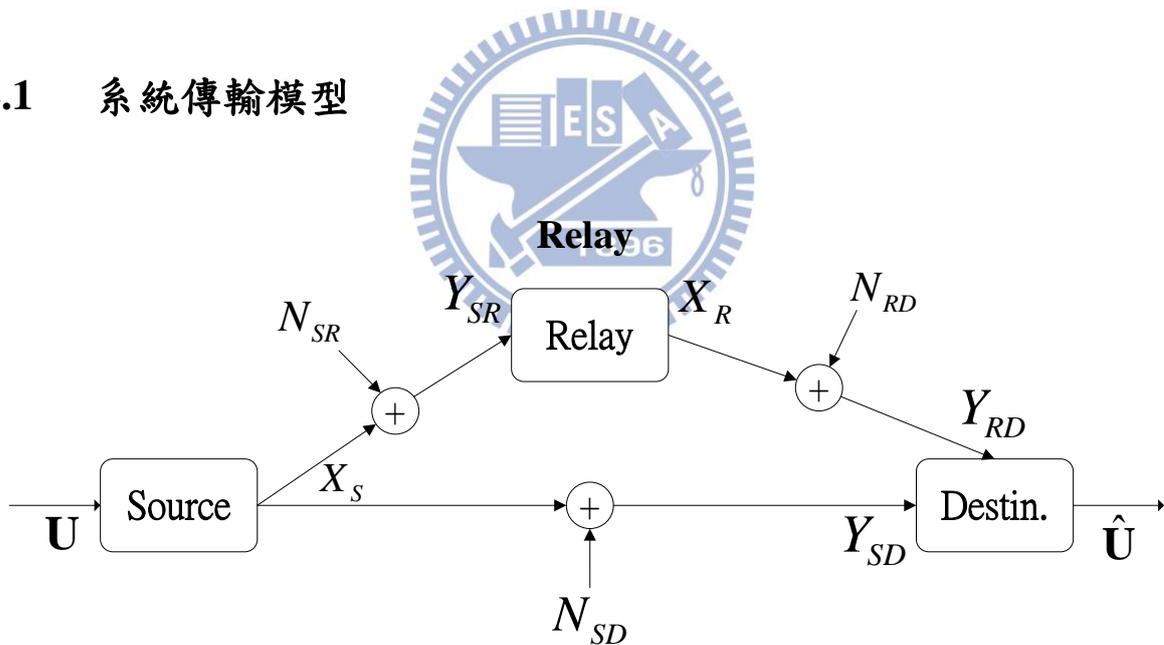


圖 2.1 系統傳輸模型[15]

如圖 2.1 所示，系統分為來源端(Source node)、中繼端(Relay node)、目的端(Destination node)三個部分。基本假設為半雙工高斯中繼通道(half-duplex Gaussian relay channel)且無記憶以及正交接收器，意即來源端至中繼端、來源端至目的端、中繼端至目的端三條鏈結都是獨立的 AWGN 通道且有各別的信號雜訊比(signal-to-noise ratio, SNR)。其中來源端至目的端又稱作直接鏈結(direct link)路徑。

來源端使用通道編碼處理均勻(uniform)且獨立同分布(independent and identically distributed, i.i.d)的信息 U ，並使用二位元鍵移(binary phase shift keying, BPSK)調變為 X_S 後進行傳輸。傳輸分為兩個階段：第一階段為來源端傳至中繼端與目的端，接收信號分別為 Y_{SR} 與 Y_{SD} ，接著中繼端將 Y_{SR} 進行信號處理與 BPSK 調變為 X_R 。第二階段為中繼端傳至目的端，接收信號為 Y_{RD} 。待目的端接收到 Y_{RD} 後，結合直接鏈結路徑的 Y_{SD} 解回信息 \hat{U} 。

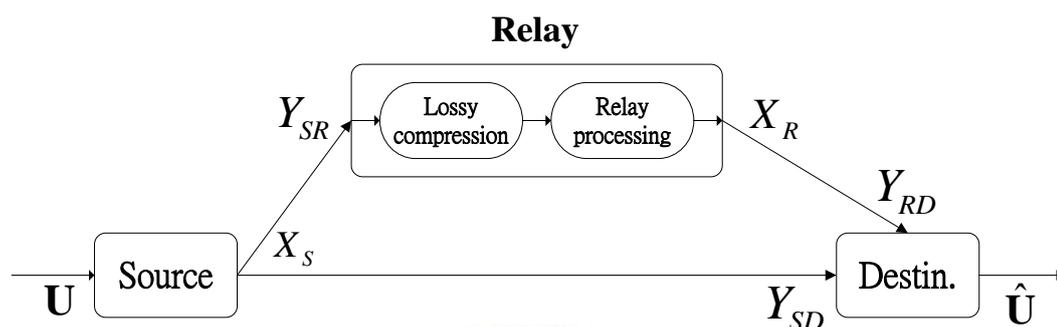


圖 2.2 基於觀察點的壓縮[15]

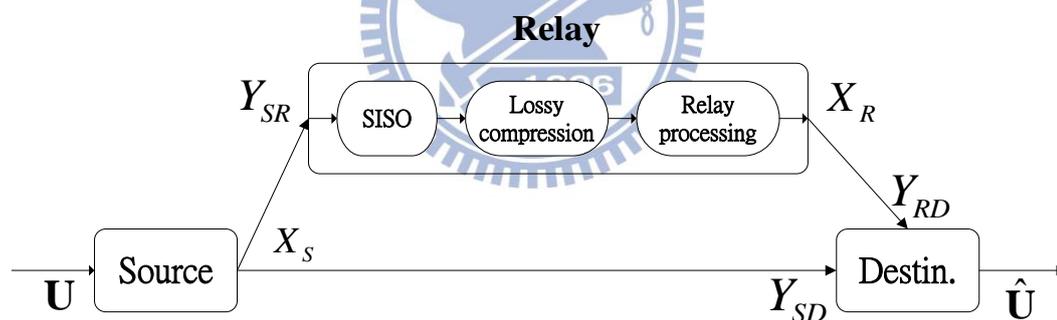


圖 2.3 基於對數相似比率的壓縮[15]

關於中繼端壓縮傳遞的做法大致上分為兩種：一為基於接收信號的壓縮(如圖 2.2)，二為基於對數相似比率的壓縮(如圖 2.3)，這兩種做法皆屬失真壓縮(lossy compression)。第一種做法是直接將來源端的接收信號 Y_{SR} 做壓縮(例如量化)，然後進一步處理(例如通道編碼)後輸出信號。第二種做法是先將 Y_{SR} 做軟式輸出輸入(soft-input soft-output, SISO)解碼得到每個信息的事後機率(a posteriori probability, APP)的對數相似值(log-likelihood ratio, LLR)，再將 LLR 壓縮後進一步處理送出。

若來源端使用一個碼率為 $R_s = k/N$ 的通道編碼器 C_s ，並針對一長度 k 的信息

$\mathbf{U} = [u_1, u_2, \dots, u_n, \dots, u_k]$ 進行編碼，則第二種方法只要壓縮 k 個信號點，而第一種方法則要壓縮 N 個信號點。表面上看起來第一種方法比較好(因為不需要壓縮 N 點)，不過我們的目的是要讓目的端利用 Y_{RD} 與 Y_{SD} 的整體解碼效能接近分散式天線系統(利用 Y_{SR} 與 Y_{SD} 解碼)的效能。在[13]裡簡單地證明，中繼端利用基於接收信號壓縮當作壓縮傳遞的第一種方案時，可以有更多的資訊幫助 \mathbf{C}_s 的解碼器。以下說明其證明過程，假設信息點 u_n 在中繼端與目的端相對應的接收信號點 Y_{SR} 與 Y_{SD} ，且 L_{SR} 為 Y_{SR} 的 LLR，則互資訊(mutual information) $I(u_n; Y_{SR}, Y_{SD})$ 可表示為：

$$\begin{aligned}
 I(u_n; Y_{SR}, Y_{SD}) &= I(u_n; Y_{SD}) + I(u_n; Y_{SR} | Y_{SD}) \\
 &= I(u_n; Y_{SD}) + I(u_n; Y_{SR}, L_{SR} | Y_{SD}) \\
 &= I(u_n; Y_{SD}) + I(u_n; L_{SR} | Y_{SD}) + I(u_n; Y_{SR} | Y_{SD}, L_{SR}) \quad (2.1) \\
 &\geq I(u_n; Y_{SD}) + I(u_n; L_{SR} | Y_{SD}) \\
 &= I(u_n; Y_{SD}, L_{SR})
 \end{aligned}$$

可以看到最後的結果為 $I(u_n; Y^R, Y^D) \geq I(u_n; Y^D, L^R)$ 。如[13]所提，此不等式無法滿足所有情況，例如來源端不編碼傳輸或使用重複碼(repetition code)編碼，這兩種方法會有同樣的效能。

2.2 目的端前級解碼與效能評估

如圖 2.4 所示，當目的端收到直接鏈結路徑的 Y_{SD} 與中繼端傳送的 Y_{RD} 後，分成兩個步驟來估測最後的輸出信息 $\hat{\mathbf{U}}$ 。前級處理先利用 Y_{SD} 與 Y_{RD} 解出 $\hat{\mathbf{Y}}_Q$ ，後級處理則利用 Y_{SD} 與 $\hat{\mathbf{Y}}_Q$ 解出 $\hat{\mathbf{U}}$ 。關於第一個步驟，我們利用疊代訊源通道解碼來處理，即圖 2.1 中的虛線部分。然後通過 MMSE 獲得估算的中繼端接收信號 $\hat{\mathbf{Y}}_Q$ 。而第二個步驟，則是利用解碼器 \mathbf{C}_s 解碼器對 Y_{SD} 與 $\hat{\mathbf{Y}}_Q$ 進行解碼。

目的端之前級解碼架構為疊代訊源通道解碼，其概念源自於渦輪解碼(turbo decoding)，主要核心是訊源與通道兩種 SISO 解碼器構成。在疊代解碼的過程中，兩個解碼器互相利用對方輸出的額外訊息來幫助解碼，隨著疊代次數增加，整體解碼的準確性也逐漸提高。這些額外訊息分別來自訊源的殘餘冗息與迴旋碼的錯誤檢查能力。

假設我們使用一個 m 位元的純量量化器作為訊源編碼器，直接把中繼端的接收信號 $Y_{SR,1}^N = [y_{SR,1}, y_{SR,2}, \dots, y_{SR,n}, \dots, y_{SR,N}]$ 量化為一串索引序列 $\mathbf{B} = [\mathbf{b}_1, \mathbf{b}_2, \dots, \mathbf{b}_n, \dots, \mathbf{b}_N]$ ，其中

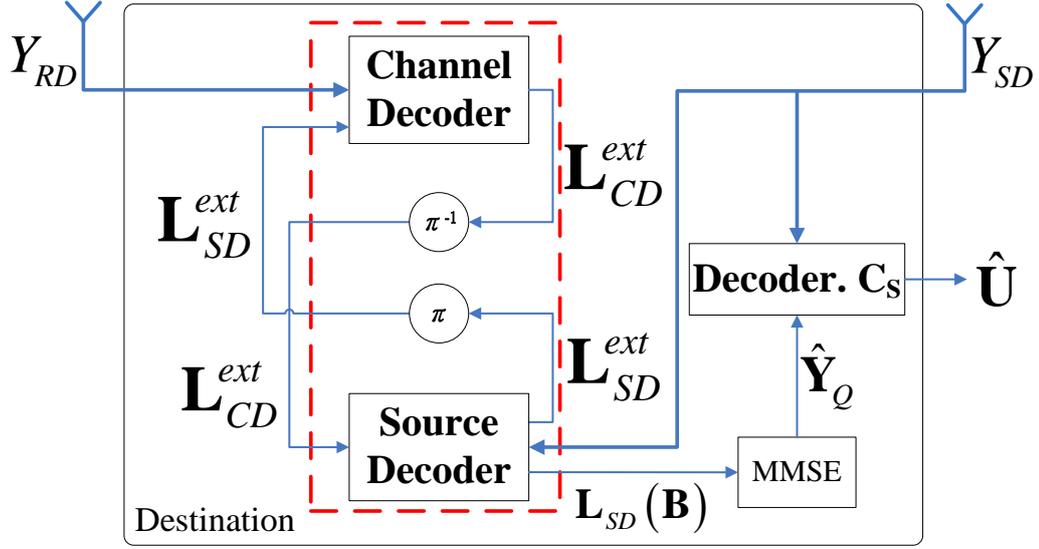


圖 2.4 目的端解碼架構

$\mathbf{b}_n = [b_{n,1}, b_{n,2}, \dots, b_{n,k}, \dots, b_{n,m}]$ 且相對應的重建信號值為 $\mathbf{Y}_{Q,1}^N = [y_{Q,1}, y_{Q,2}, \dots, y_{Q,n}, \dots, y_{Q,N}]$ 。訊源解碼器整合收到的訊息 \tilde{Y} ，包括直接鏈結路徑的 Y_{SD} 與通道解碼器回傳的額外訊息 $\mathbf{L}_{CD}^{ext}(\mathbf{B})$ ，用以解出每個索引值 \mathbf{b}_n 的事後對數相似比率 $\mathbf{L}_{SD}(\mathbf{b}_n)$ 。等到最終疊代完成之後，由 $\mathbf{L}_{SD}(\mathbf{b}_n)$ 計算 $p(\mathbf{b}_n | \tilde{Y})$ 與相對應的重建信號 $\hat{y}_{Q,n}$ 。最終目的是要讓解碼器依據最小均方誤差(minimum mean square error, MMSE)準則解出 $\hat{\mathbf{Y}}_Q$ ，能最佳化地估算中繼端的接收信號 Y_{SR} 。

所謂的最小均方誤差準則，是將原始訊號 $y_{SR,n}$ 與重建信號 $\hat{y}_{Q,n}$ 的平均誤差最小化，如下所示：

$$\hat{y}_{Q,n}^{MMSE} = \min_{\hat{y}_{Q,n}} E \left\{ (y_{SR,n} - \hat{y}_{Q,n})^2 | \tilde{Y} \right\} \quad (2.2)$$

即是將重建訊雜比(reconstruction signal-to-noise ratio, RSNR)最大化，重建訊雜比為：

$$RSNR = \frac{\sum_{n=1}^N (y_{SR,n})^2}{\sum_{n=1}^N (y_{SR,n} - \hat{y}_{Q,n})^2} \quad (2.3)$$

由(2.2)式可推得估算為：

$$\hat{y}_{Q,n}^{MMSE} = \sum_{Y_Q(k) \in \Psi} p(\mathbf{b}_n | \tilde{Y}) Y_Q(k) \quad (2.4)$$

其中 Ψ 表示所有量化索引重建值的集合。而整體解碼效能我們將用重建訊雜比來評比。

2.3 目的端的後級解碼

在來源端我們使用一碼率 $1/2$ 的遞迴系統迴旋碼的通道編碼器。所謂系統碼 (systematic code)，是在通道編碼的過程裡，會產生與輸入端相同的訊息位元及通道編碼演算法計算出的同位位元。相較於區塊碼(block code)中，碼率為 k/N 的編碼器輸入了 k 個訊息位元然後產生 N 位元的碼字，編碼過程是以一個區塊為單位進行，所以編碼器內必須有足夠空間儲存全部的訊息片段以產生碼字。但實際運作時訊息位元是以串列的方式輸入，因此區塊碼受限於大的儲存空間而較不實際。反之，迴旋碼的編碼是經由不斷地與輸入訊息運算來產生碼字，而且可以利用籬柵圖(trellis diagram)呈現輸入訊息、輸出訊息以及編碼器內暫存器狀態的轉移關係。以一碼率為 $1/2$ 的迴旋編碼器為例，其籬柵圖展示於圖 2.5。以下我們將分兩小節分別介紹基於迴旋碼的 BCJR 解碼演算法，以及目的端之後級解碼器如何整合運用 Y_{SD} 與 \hat{Y}_Q 這兩串資訊。

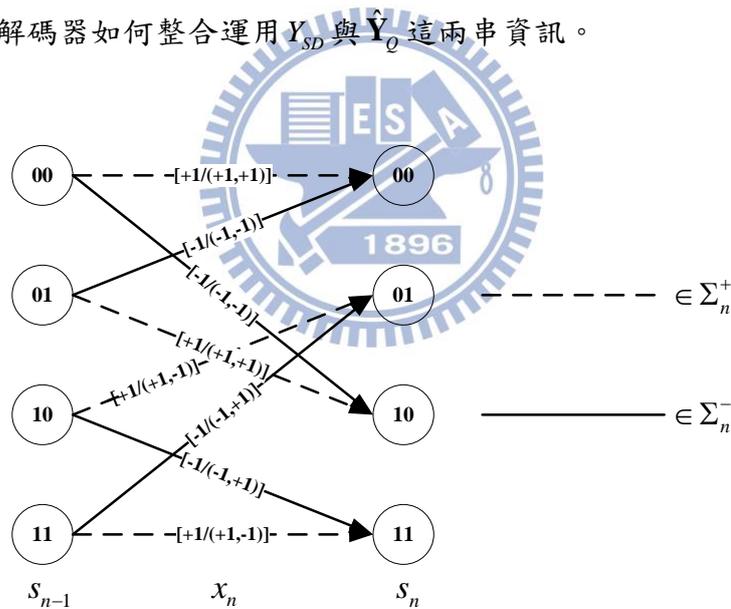


圖 2.5 迴旋碼之位元層級籬柵圖

2.3.1 BCJR 演算法

BCJR 演算法是基於一種最大事後機率 (maximum a posteriori probability, MAP) 法則。由 Bahl、Cocke、Jelinek 及 Raviv 提出，廣泛應用於迴旋碼的軟式輸出輸入解碼。主要是利用一接收序列解出每個位元的最大事後機率，可以增強解碼可靠度。由於迴旋碼具備籬柵結構，可據以推導出一個遞迴形式的演算法，進而大幅降低運算複雜度。這一節

將介紹基於單一接收信號序列(例如來源端至目的端的信號 Y_{SD})的事後機率。我們先定義 Y_{SD} 的形式，以 $Y_{SD,1}^{\tilde{k}}$ 表達 Y_{SD} 為一長度 \tilde{k} 的向量信號：

$$Y_{SD,1}^{\tilde{k}} = [y_{SD,1}, y_{SD,2}, \dots, y_{SD,n}, \dots, y_{SD,\tilde{k}}] = [Y_{SD}^s, Y_{SD}^p], \quad n=1, 2, \dots, \tilde{k} \quad (2.5)$$

$$y_{SD,n} = [y_{SD,n}^s, y_{SD,n}^p]$$

其中每個 $y_{SD,n}$ 有一個訊息位元 $y_{SD,n}^s$ 及同位位元 $y_{SD,n}^p$ 。假設為無記憶的AWGN通道，且通道雜訊比為 $\frac{E_b}{N_0}$ ，雜訊的變異數為 $\sigma^2 = \frac{N_0}{2}$ 。此時序列中第 n 位置訊息位元的通道訊

息 $p(y_{SD,n}^s | u_n)$ 與同位位元的通道訊息 $p(y_{RD,n}^p | u_n^p)$ 可由以下公式分別計算：

$$p(y_{SD,n}^s | u_n) = \frac{1}{\sqrt{2\pi\sigma^2}} \cdot \exp\left(-\frac{E_b}{N_0} (y_{SD,n}^s - u_n)^2\right) \quad (2.6)$$

$$p(y_{RD,n}^p | u_n^p) = \frac{1}{\sqrt{2\pi\sigma^2}} \cdot \exp\left(-\frac{E_b}{N_0} (y_{RD,n}^p - u_n^p)^2\right)$$

若使用上式公式計算時，碼字位元都視為已用BPSK調變過，以保持編解碼的一致性。如圖2.5中暫存器狀態之間的連線所示。接著進入推導過程，我們只考慮 $u_n = +1$ 的情況，因 $p(u_n = +1 | Y_{SD,1}^N)$ 與 $p(u_n = -1 | Y_{SD,1}^N)$ 的和等於1。首先展開 $p(u_n = +1 | Y_{SD,1}^N)$ 為：

$$p(u_n = +1 | Y_{SD,1}^N) = \frac{p(u_n = +1, Y_{SD,1}^N)}{p(Y_{SD,1}^N)} = \frac{\sum_{(s_{n-1}, s_n) \in \Sigma_n^+} p(s_{n-1}, s_n, Y_{SD,1}^N)}{p(Y_{SD,1}^N)} \quad (2.7)$$

$$= C \cdot \sum_{(s_{n-1}, s_n) \in \Sigma_n^+} p(s_{n-1}, s_n, Y_{SD,1}^N)$$

其中 C 為一常數， s_{n-1} 與 s_n 分別為前一個時刻與目前時刻的狀態， Σ_n^+ 表示在 $s_{n-1} \rightarrow s_n$ 間所有對應於 $u_n = +1$ 的狀態轉移分支的集合，如圖2.5所示。基於無記憶通道且輸入位元 u_n 獨立性的假設， $p(s_{n-1}, s_n, Y_{SD,1}^N)$ 可分解成三項：

$$p(s_{n-1}, s_n, Y_{SD,1}^N) = p(s_{n-1}, Y_{SD,1}^{n-1}) \cdot p(s_n, y_{SD,n} | s_{n-1}) \cdot p(Y_{SD,n+1}^N | s_n) \quad (2.8)$$

$$\triangleq \alpha_{n-1}(s_{n-1}) \cdot \gamma_n(s_{n-1}, s_n) \cdot \beta_n(s_n)$$

其中 $\alpha_n(s_n)$ 稱為前向計量(forward metric)：

$$\alpha_n(s_n) = \sum_{s_{n-1}} \gamma_n(s_{n-1}, s_n) \cdot \alpha_{n-1}(s_{n-1}) \quad (2.9)$$

$\beta_{n-1}(s_{n-1})$ 稱為後向計量(backward metric)：

$$\beta_{n-1}(s_{n-1}) = \sum_{s_{n-1}} \gamma_n(s_{n-1}, s_n) \cdot \beta_n(s_n) \quad (2.10)$$

$\gamma_l(s_{l-1}, s_l)$ 稱為分支計量(branch metric)：

$$\begin{aligned} \gamma_n(s_{n-1}, s_n) &= p(s_n, y_{SD,n} | s_{n-1}) \\ &= p(s_n | s_{n-1}) \cdot p(y_{SD,n} | s_{n-1}, s_n) \\ &= p(u_n = +1) \cdot p(y_{SD,n} | u_n = +1, u_n^p) \\ &= p(u_n = +1) \cdot p(y_{SD,n}^s | u_n = +1) \cdot p(y_{SD,n}^p | u_n^p) \end{aligned} \quad (2.11)$$

由於利用了迴旋碼的籬柵結構，前向計量與後向計量皆可以表達成與分支計量有關的遞迴運算形式。上式中的第三個等式是由 $(s_{n-1}, s_n) \in \Sigma_n^+$ 而來，第四個等式則利用了無記憶通道假設。由於迴旋碼通常都由全零的狀態開始編碼，處理完全部的輸入位元後再回到零的狀態，因此在解碼時設定前向與後向計量的初始狀態為：

$$\begin{aligned} \alpha_0(s_0) &= \begin{cases} 1, & s_0 = 0 \\ 0, & s_0 \neq 0 \end{cases} \\ \beta_N(s_N) &= \begin{cases} 1, & s_N = 0 \\ 0, & s_N \neq 0 \end{cases} \end{aligned} \quad (2.12)$$

2.3.2 結合多重資訊的 BCJR 演算法

當目的端執行第一步驟的疊代解碼之後，利用最小均方誤差準則可以得到中繼端的還原信號 $\hat{\mathbf{Y}}_Q$ 。我們定義 $\hat{\mathbf{Y}}_Q$ 的形式，以 $\hat{\mathbf{Y}}_{Q,1}^{\tilde{k}}$ 表達 $\hat{\mathbf{Y}}_Q$ 為一長度 \tilde{k} 的信號：

$$\begin{aligned} \hat{\mathbf{Y}}_{Q,1}^{\tilde{k}} &= [\hat{y}_{Q,1}, \hat{y}_{Q,2}, \dots, \hat{y}_{Q,n}, \dots, \hat{y}_{Q,\tilde{k}}] = [\hat{Y}_Q^s, \hat{Y}_Q^p], \quad n=1, 2, \dots, \tilde{k} \\ \hat{y}_{Q,n} &= [\hat{y}_{Q,n}^s, \hat{y}_{Q,n}^p] \end{aligned} \quad (2.13)$$

接著我們利用 $\hat{\mathbf{Y}}_{Q,1}^{\tilde{k}}$ 與直接鏈結路徑信號 Y_{SD} ，透過 BCJR 演算法來解出最後的信息 $\hat{\mathbf{U}}$ 。由於推導類似，我們將省略一些步驟。考慮 $u_n = +1$ 的情況，而 $p(u_n = +1 | Y_{SD,1}^{\tilde{k}}, \hat{\mathbf{Y}}_{Q,1}^{\tilde{k}})$ 與 $p(u_n = -1 | Y_{SD,1}^{\tilde{k}}, \hat{\mathbf{Y}}_{Q,1}^{\tilde{k}})$ 的和等於 1。首先展開 $p(u_n = +1 | Y_{SD,1}^{\tilde{k}}, \hat{\mathbf{Y}}_{Q,1}^{\tilde{k}})$ 為：

$$p(u_n = +1 | Y_{SD,1}^{\tilde{k}}, \hat{\mathbf{Y}}_{Q,1}^{\tilde{k}}) = C \cdot \sum_{(s_{n-1}, s_n) \in \Sigma_n^+} p(s_{n-1}, s_n, Y_{SD,1}^{\tilde{k}}, \hat{\mathbf{Y}}_{Q,1}^{\tilde{k}}) \quad (2.14)$$

而 $p(s_{n-1}, s_n, Y_{SD,1}^{\tilde{k}}, \hat{\mathbf{Y}}_{Q,1}^{\tilde{k}})$ 可分解成三項：

$$\begin{aligned}
& p(s_{n-1}, s_n, Y_{SD,1}^{\tilde{k}}, \hat{Y}_{Q,1}^{\tilde{k}}) \\
&= p(s_{n-1}, Y_{SD,1}^{n-1}, \hat{Y}_{Q,1}^{n-1}) \cdot p(s_n, y_{SD,n}, \hat{y}_{Q,n} | s_{n-1}) \cdot p(Y_{SD,n+1}^{\tilde{k}}, \hat{Y}_{Q,n+1}^{\tilde{k}} | s_n) \\
&\triangleq \alpha_{n-1}(s_{n-1}) \cdot \gamma_n(s_{n-1}, s_n) \cdot \beta_n(s_n)
\end{aligned} \tag{2.15}$$

其中前向計量 $\alpha_n(s_n)$ 與後向計量 $\beta_{n-1}(s_{n-1})$ 即其初始值亦分別如(2.9)式、(2.10)式與(2.12)式所述。由於額外結合了中繼端的輸出信號，所以分支計量 $\gamma_n(s_{n-1}, s_n)$ 的部分修改如下：

$$\begin{aligned}
\gamma_n(s_{n-1}, s_n) &= p(s_n, y_{SD,n}, \hat{y}_{Q,n} | s_{n-1}) \\
&= p(s_n | s_{n-1}) \cdot p(y_{SD,n}, \hat{y}_{Q,n} | s_{n-1}, s_n) \\
&= p(u_n = +1) \cdot p(y_{SD,n} | u_n = +1, u_n^p) \cdot p(\hat{y}_{Q,n} | u_n = +1, u_n^p) \\
&= p(u_n = +1) \cdot p(y_{SD,n}^s | u_n = +1) \cdot p(y_{SD,n}^p | u_n^p) \\
&\quad \cdot p(\hat{y}_{Q,n}^s | u_n = +1) \cdot p(\hat{y}_{Q,n}^p | u_n^p)
\end{aligned} \tag{2.16}$$

其中 $y_{SD,n}^s$ 與 $\hat{y}_{Q,n}^s$ 皆屬於系統位元的部分， $y_{SD,n}^p$ 與 $\hat{y}_{Q,n}^p$ 則屬於同位元部分。

2.4 頻寬擴展

由於壓縮傳遞中的系統架構，會造成中繼端的輸出位元比輸入多。我們引入頻寬擴展的概念，藉由增加中繼端至目的端路徑的頻寬來傳送更多的位元。且在[14]裡也提到，如果壓縮傳遞搭配頻寬擴展使用，系統的性能可以接近分散式天線系統。而該如何決定擴展多少頻寬，我們定義 γ 為頻寬擴展係數，它是一個倍率的關係。如果 $\gamma=2$ ，表示中繼端至目的端的頻寬會是來源端到中繼端與來源端到目的端的 2 倍。

第3章 基於單一同位檢查碼的編碼機制

本章節將介紹前人研究[15]針對中繼端(relay node)提出的合併訊源通道編碼(joint source-channel coding, JSCC)技術。透過對來源端之接收信號直接量化，配合使用單一同位檢查碼(single parity-check code, SPCC)與迴旋碼編碼，來提升直接鏈結的解碼效能。至於中繼端如何重建來源端之接收信號，將在以下做詳細的解說。

3.1 中繼端傳輸架構

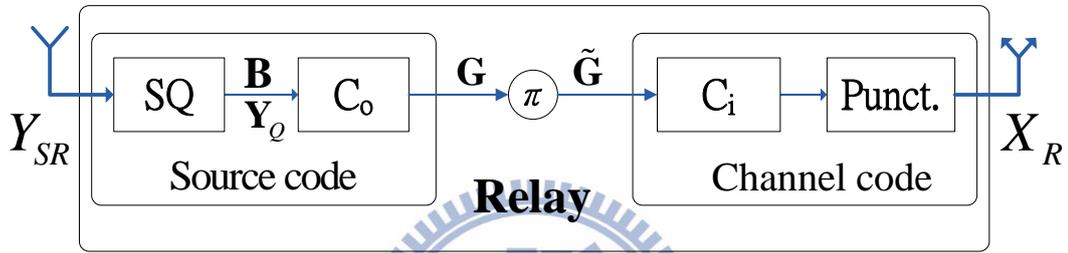


圖 3.1 中繼端傳輸架構

系統架構如圖3.1所示，大致分成三個部分組成，第一個是訊源編碼器，由純量量化器與單一同位檢查碼 C_0 組成，第二個是位元交錯器 π (bit interleaver)、第三個是由迴旋碼 C_i 與穿刺(puncture)組成的通道編碼器。

當中繼端收到被 AWGN 通道干擾的來源端信號序列 $Y_{S,R}^N = [y_{S,R,1}, y_{S,R,2}, \dots, y_{S,R,n}, \dots, y_{S,R,N}]$ ， $S \in \mathcal{R}$ ， $R \in \mathcal{R}$ ，每個信號點 $y_{S,R,n}$ 將經由純量量化器產生一個 m 位元的量化索引 \mathbf{b}_n 以及相對應的重建信號值 $y_{Q,n}$ 。接著 \mathbf{b}_n 經過一個單一同位檢查碼 C_0 產生 $(m+1)$ 位元的碼字 \mathbf{g}_n ，這個動作可以增加一個冗餘位元，以提高其歐式距離 (Euclidean distance)，歐式距離越高越能保護資料。以下我們定義：

$$\begin{aligned}
 \mathbf{Y}_{Q,1}^N &= [y_{Q,1}, y_{Q,2}, \dots, y_{Q,n}, \dots, y_{Q,N}] \\
 \mathbf{B} &= [\mathbf{b}_1, \mathbf{b}_2, \dots, \mathbf{b}_n, \dots, \mathbf{b}_N] \\
 \mathbf{G} &= [\mathbf{g}_1, \mathbf{g}_2, \dots, \mathbf{g}_n, \dots, \mathbf{g}_N] \\
 y_{Q,n} &\equiv \mathbf{b}_n = [b_{n,1}, b_{n,2}, \dots, b_{n,k}, \dots, b_{n,m}] \\
 &\equiv \mathbf{g}_n = [g_{n,1}, g_{n,2}, \dots, g_{n,k}, \dots, g_{n,m+1}]
 \end{aligned} \tag{3.1}$$

相對於接收到的 N 個信號， $\mathbf{Y}_{Q,1}^N$ 、 \mathbf{B} 與 \mathbf{G} 分別代表 N 個重建信號值、 N 個量化索引與 N

個有單一同位檢查碼字。粗體字表示內含多個位元的集合， $b_{n,k}$ 代表第 n 個索引值的第 k 個位元。然後將 \mathbf{G} 通過一個位元交錯器產生 $\tilde{\mathbf{G}} = [\tilde{g}_1, \tilde{g}_2, \dots, \tilde{g}_l, \dots, \tilde{g}_L]$ ，其中交錯器的大小 $L = (m+1)N$ ，而 \tilde{g}_l 對應到交錯後的 $g_{n,k}$ ：

$$\tilde{g}_l = \pi(g_{n,k}), \quad l=1,2,\dots,L \quad (3.2)$$

交錯器的功能是打散資料間的相關性，避免在進行疊代解碼的過程中用到重複的資訊。接著 $\tilde{\mathbf{G}}$ 通過一個碼率為 1 的迴旋碼 C_i ，只傳送碼率 1/2 遞迴系統性迴旋碼的同位位元部分而產生 $\tilde{\mathbf{G}}^p$ ：

$$\tilde{\mathbf{G}}^p = [\tilde{g}_1^p, \tilde{g}_2^p, \dots, \tilde{g}_l^p, \dots, \tilde{g}_{\hat{L}}^p], \quad l=1,2,\dots,\hat{L} \quad (3.3)$$

此處長度 \hat{L} 包含了 L 個輸入位元以及 $\hat{L} - L$ 個終止位元(termination bit)。接著把 $\tilde{\mathbf{G}}^p$ 進行穿刺後產生 mN 個位元的輸出，穿刺的作用是移除一些位元，以降低中繼端到目的端所需的頻寬擴展。接著進行 BPSK 調變後產生 X_R ，再經由 AWGN 通道傳送到目的端。

3.2 目的端解碼架構

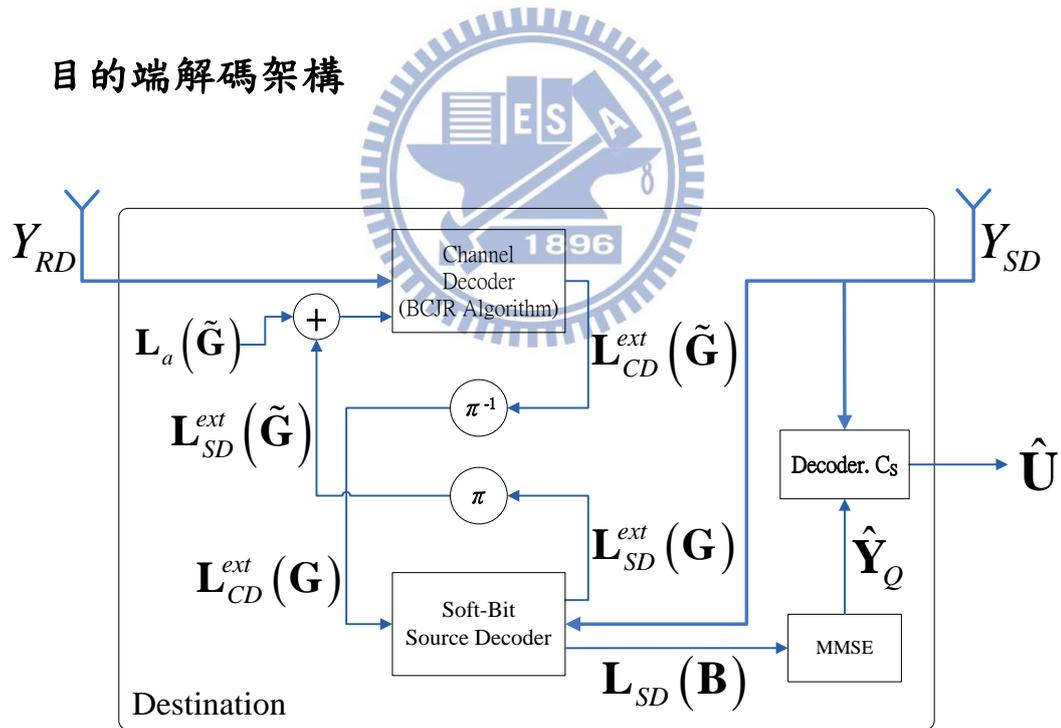


圖 3.2 目的端解碼架構

如圖 3.2 所示，當目的端收到直接鏈結路徑的 Y_{SD} 與中繼端傳送的 Y_{RD} 後，我們分成兩個步驟來估測最後的輸出信息 $\hat{\mathbf{U}}$ ：先利用 Y_{SD} 與 Y_{RD} 解出 $\hat{\mathbf{Y}}_Q$ ，再利用 Y_{SD} 與 $\hat{\mathbf{Y}}_Q$ 解出 $\hat{\mathbf{U}}$ 。關於第二個步驟，已於先前 2.3.2 節解釋如何利用 Y_{SD} 與 $\hat{\mathbf{Y}}_Q$ 解出 $\hat{\mathbf{U}}$ 。現在我們把焦點放

在第一個步驟。當目的端接收到序列後，因為中繼端的通道編碼器只輸出同位檢查位元且進行穿刺，所以將序列進行解穿刺(depuncture)，亦即在移除掉位元的位置上補上 0，還原如下：

$$Y_{RD,l}^{\hat{L}} = \left[y_{RD,1}^p, y_{RD,2}^p \dots y_{RD,l}^p \dots, y_{RD,\hat{L}}^p \right], \quad l = 1, 2, \dots, \hat{L} \quad (3.4)$$

假設中繼端至目的端為 AWGN 的無記憶通道，且通道雜訊比為 $\frac{E_b}{N_0}$ 、雜訊的變異數為 $\sigma^2 = \frac{N_0}{2}$ 。接收序列第 l 位置的同位位元的通道訊息可以表示為：

$$p(y_{RD,l}^p | \tilde{g}_l^p) = \frac{1}{\sqrt{2\pi\sigma^2}} \cdot \exp\left(-\frac{E_b}{N_0} (y_{RD,l}^p - \tilde{g}_l^p)^2\right) \quad (3.5)$$

為了還原出 Y_Q ，我們使用疊代訊源通道解碼，且在此時將以對數相似比率方式呈現。有關訊源解碼器與通道解碼器的詳細說明將在下兩節介紹，以下先說明疊代解碼的步驟：

1. 初始化：設定疊代次數 $\text{iter} = 1$ ，訊源解碼額外訊息的初始值為 $\mathbf{L}_{SD}^{\text{ext}}(\mathbf{G}) = 0$ ，其交錯值為 $\mathbf{L}_{SD}^{\text{ext}}(\tilde{\mathbf{G}}) = 0$ 。定義 $\mathbf{L}_{SD}^{\text{ext}}(\mathbf{G})$ 與 $\mathbf{L}_{SD}^{\text{ext}}(\tilde{\mathbf{G}})$ ：

$$\begin{aligned} \mathbf{L}_{SD}^{\text{ext}}(\mathbf{G}) &= \left[L_{SD}^{\text{ext}}(g_{1,1}), L_{SD}^{\text{ext}}(g_{1,2}), \dots, L_{SD}^{\text{ext}}(g_{n,k}), \dots, L_{SD}^{\text{ext}}(g_{N,m+1}) \right] \\ \mathbf{L}_{SD}^{\text{ext}}(\tilde{\mathbf{G}}) &= \left[L_{SD}^{\text{ext}}(\tilde{g}_1), L_{SD}^{\text{ext}}(\tilde{g}_2), \dots, L_{SD}^{\text{ext}}(\tilde{g}_l), \dots, L_{SD}^{\text{ext}}(\tilde{g}_L) \right] \end{aligned} \quad (3.6)$$

其中

$$L_{SD}^{\text{ext}}(\tilde{g}_l) = \log \frac{p_{SD}^{\text{ext}}(\tilde{g}_l = +1)}{p_{SD}^{\text{ext}}(\tilde{g}_l = -1)} \quad (3.7)$$

2. 通道解碼部分：三個輸入分別是源自中繼端傳送的 $Y_{RD,l}^{\hat{L}}$ 、事前訊息 $\mathbf{L}_a(\tilde{\mathbf{G}})$ 與訊源解碼額外訊息的交錯值 $\mathbf{L}_{SD}^{\text{ext}}(\tilde{\mathbf{G}})$ 。通道解碼器使用 BCJR 演算法，計算每個位元的通道解碼訊息 $\mathbf{L}_{CD}^{\text{ext}}(\tilde{\mathbf{G}})$ ，再通過位元解交錯器(bit deinterleaver)成為 $\mathbf{L}_{CD}^{\text{ext}}(\mathbf{G})$ 後提供給訊源解碼器使用。定義 $\mathbf{L}_{CD}^{\text{ext}}(\tilde{\mathbf{G}})$ 與 $\mathbf{L}_{CD}^{\text{ext}}(\mathbf{G})$ ：

$$\begin{aligned} \mathbf{L}_{CD}^{\text{ext}}(\tilde{\mathbf{G}}) &= \left[L_{CD}^{\text{ext}}(\tilde{g}_1), L_{CD}^{\text{ext}}(\tilde{g}_2), \dots, L_{CD}^{\text{ext}}(\tilde{g}_l), \dots, L_{CD}^{\text{ext}}(\tilde{g}_L) \right] \\ \mathbf{L}_{CD}^{\text{ext}}(\mathbf{G}) &= \left[L_{CD}^{\text{ext}}(g_{1,1}), L_{CD}^{\text{ext}}(g_{1,2}), \dots, L_{CD}^{\text{ext}}(g_{n,k}), \dots, L_{CD}^{\text{ext}}(g_{N,m+1}) \right] \end{aligned} \quad (3.8)$$

其中

$$L_{CD}^{ext}(\tilde{g}_l) = \log \frac{p_{CD}^{ext}(\tilde{g}_l = +1)}{p_{CD}^{ext}(\tilde{g}_l = -1)} \quad (3.9)$$

3. 訊源解碼部分：兩個輸入分別是直接鏈結路徑的 $Y_{SD,1}^N$ 與通道解碼額外訊息 $\mathbf{L}_{CD}^{ext}(\mathbf{G})$ ，計算出索引值的後驗對數相似比率 $\mathbf{L}_S(\mathbf{G}) = [L(s|g)_{p=1}, L(s|g)_{p=2}, \dots, L(s|g)_{p=n}]$ 。由於訊源編碼器使用的是簡單的單一同位檢查碼，所以可以輕易地由 $\mathbf{L}_{SD}(\mathbf{G})$ 得到 $\mathbf{L}_{SD}(\mathbf{B}) = [L(s|b)_{p=1}, L(s|b)_{p=2}, \dots, L(s|b)_{p=n}]$ 。至於訊源編碼額外訊息 $\mathbf{L}_{SD}^{ext}(\mathbf{G})$ ，則是將 $\mathbf{L}_{SD}(\mathbf{G})$ 直接減去通道編碼額外訊息 $\mathbf{L}_{CD}^{ext}(\mathbf{G})$ 。接著將 $\mathbf{L}_{CD}^{ext}(\mathbf{G})$ 通過交錯器得到 $\mathbf{L}_{CD}^{ext}(\tilde{\mathbf{G}})$ 再回傳給通道解碼器當事前訊息使用。
4. 下一次的疊代，如果未達最大疊代次數 `iter_max`，則回到步驟二繼續進行，反之則進行步驟五。
5. 利用步驟三算出的後驗對數相似比率 $\mathbf{L}_{SD}(\mathbf{B})$ 得到事後機率 $p(\mathbf{b}_n | y_{SD,n}, \mathbf{L}_{CD}^{ext}(\mathbf{g}_n))$ ，再使用最小均方誤差準則來合成新的重建信號值 \hat{Y}_Q ：

$$\hat{Y}_Q(n) = \sum_{Y_Q(k) \in \Psi} p(\mathbf{b}_n | y_{SD,n}, \mathbf{L}_{CD}^{ext}(\mathbf{g}_n)) \cdot Y_Q(k) \quad (3.10)$$

此處的 Ψ 表示所有量化索引重建值的集合。

3.3 BCJR 通道解碼演算法

前面章節已經介紹過碼率 1/2 的迴旋碼的 BCJR 演算法，我們接著推導碼率 1 (只傳送碼率 1/2 迴旋碼的同位位元部分) 迴旋碼的最大事後機率及其額外訊息。由於推導類似，所以我們省略一些推導步驟。先考慮 $\tilde{g}_l = +1$ 的情況，而 $p(\tilde{g}_l = +1 | Y_{RD,1}^L)$ 與 $p(\tilde{g}_l = -1 | Y_{RD,1}^L)$ 的和等於 1。首先展開 $p(\tilde{g}_l = +1 | Y_{RD,1}^L)$ 為：

$$p(\tilde{g}_l = +1 | Y_{RD,1}^L) = C \cdot \sum_{(s_{l-1}, s_l) \in \Sigma_l^+} p(s_{l-1}, s_l, Y_{RD,1}^L) \quad (3.11)$$

其中 C 為一常數， s_{l-1} 與 s_l 分別為前一個時刻跟目前時刻的狀態， Σ_l^+ 表示在 $s_{n-1} \rightarrow s_n$ 間所有對應於 $\tilde{g}_l = +1$ 的狀態轉移分支的集合，如圖 2.5 所示。因為假設為無記憶通道且輸

入位元 \tilde{g}_l 的獨立性，所以 $p(s_{l-1}, s_l, Y_{RD,1}^L)$ 可被分解成三項：

$$\begin{aligned} p(s_{l-1}, s_l, Y_{RD,1}^L) &= p(s_{l-1}, Y_{RD,1}^{l-1}) \cdot p(s_l, y_{RD,l} | s_{l-1}) \cdot p(Y_{RD,l+1}^L | s_l) \\ &\triangleq \alpha_{l-1}(s_{l-1}) \cdot \gamma_l(s_{l-1}, s_l) \cdot \beta_l(s_l) \end{aligned} \quad (3.12)$$

其中前項計量 $\alpha_l(s_l)$ 與後向計量 $\beta_{l-1}(s_{l-1})$ 亦分別如(2.9)、(2.10)式所述。由於碼率 1 迴旋碼只傳送同位位元，所以分支計量 $\gamma_n(s_{n-1}, s_n)$ 的部分如下：

$$\begin{aligned} \gamma_l(s_{l-1}, s_l) &= p(s_l, y_{RD,l} | s_{l-1}) \\ &= p(s_l | s_{l-1}) \cdot p(y_{RD,l} | s_{l-1}, s_l) \\ &= p(\tilde{g}_l = +1) \cdot p(y_{RD,l}^p | s_{l-1}) \end{aligned} \quad (3.13)$$

下一步我們將說明額外訊息 $L_{CD}^{ext}(\tilde{g}_l)$ 的部分，此額外訊息會經由解交錯器送到訊源解碼器。由於 $p_{CD}^{ext}(\tilde{g}_l = +1)$ 與 $p_{CD}^{ext}(\tilde{g}_l = -1)$ 的和不一定為 1，所以必須分開計算。由(3.13)式：

$$\begin{aligned} \gamma_l(s_{l-1}, s_l) &= p(\tilde{g}_l = +1) \cdot p(y_{RD,l}^p | \tilde{g}_l^p) \\ &\triangleq p(\tilde{g}_l = +1) \cdot \gamma_l^{ext}(s_{l-1}, s_l) \end{aligned} \quad (3.14)$$

此處 $\gamma_l^{ext}(s_{l-1}, s_l) = p(y_{RD,l}^p | \tilde{g}_l^p)$ 。而(3.12)式與(3.14)式及(3.11)式結合後，(3.11)式變為：

$$\begin{aligned} p(\tilde{g}_l = +1 | Y_{RD,1}^L) &= C \cdot \sum_{(s_{l-1}, s_l) \in \Sigma_l^+} p(s_{l-1}, s_l, Y_{RD,1}^L) \\ &= C \cdot \sum_{(s_{l-1}, s_l) \in \Sigma_l^+} \alpha_{l-1}(s_{l-1}) \cdot \gamma_l(s_{l-1}, s_l) \cdot \beta_l(s_l) \\ &= C \cdot p(\tilde{g}_l = +1) \cdot \sum_{(s_{l-1}, s_l) \in \Sigma_l^+} \alpha_{l-1}(s_{l-1}) \cdot \gamma_l^{ext}(s_{l-1}, s_l) \cdot \beta_l(s_l) \\ &= C \cdot p(\tilde{g}_l = +1) \cdot p_{CD}^{ext}(\tilde{g}_l = +1) \end{aligned} \quad (3.15)$$

其中 $p_{CD}^{ext}(\tilde{g}_l = +1) = \sum_{(s_{l-1}, s_l) \in \Sigma_l^+} \alpha_{l-1}(s_{l-1}) \cdot \gamma_l^{ext}(s_{l-1}, s_l) \cdot \beta_l(s_l)$ 即為額外訊息的部分。而在進行

一次疊代解碼後，通道解碼器的事前訊息可由訊源解碼器回傳的額外訊息 $p_{SD}^{ext}(\tilde{g}_l = +1)$ 得以更新：

$$p(\tilde{g}_l = +1) \leftarrow [p(\tilde{g}_l = +1) \cdot p_{SD}^{ext}(\tilde{g}_l = +1)] \quad (3.16)$$

故分支計量修正為：

$$\gamma_l(s_{l-1}, s_l) = p(\tilde{g}_l = +1) \cdot p_{SD}^{ext}(\tilde{g}_l = +1) \cdot \gamma_l^{ext}(s_{l-1}, s_l) \quad (3.17)$$

而事後機率為：

$$p(\tilde{g}_i = +1 | Y_{RD,1}^L) = C \cdot p(\tilde{g}_i = +1) \cdot p_{SD}^{ext}(\tilde{g}_i = +1) \cdot p_{CD}^{ext}(\tilde{g}_i = +1) \quad (3.18)$$

3.4 軟式位元訊源解碼演算法

有關軟式位元訊源解碼演算法，在[16]裡是基於一階事前訊息(AK1)的假設而推導。由於我們來源端的輸入信息為均勻且獨立同分布，所以事前訊息由原本的一階事前訊息(AK1)退化為零階事前訊息(AK0)。整合直接鏈結路徑的信號 $y_{SD,n}$ 與通道解碼器送來的額外訊息 $\mathbf{L}_{CD}^{ext}(\mathbf{g}_n)$ ，解碼器的目標是計算每個 $g_{n,k}$ 位元的對數相似比率。以下為訊源解碼演算法：

$$L_{SD}(g_{n,k}) = \log \frac{\sum_{\mathbf{b}_n \rightarrow g_{n,k}=0} p(\mathbf{b}_n | y_{SD,n}, \mathbf{L}_{CD}^{ext}(\mathbf{g}_n))}{\sum_{\mathbf{b}_n \rightarrow g_{n,k}=1} p(\mathbf{b}_n | y_{SD,n}, \mathbf{L}_{CD}^{ext}(\mathbf{g}_n))} \quad (3.19)$$

其中 $\mathbf{b}_n \rightarrow g_{n,k}=0$ 與 $\mathbf{b}_n \rightarrow g_{n,k}=1$ 分別表示 \mathbf{b}_n 對應到位元 $g_{n,k}$ 等於 0 跟 1。而 $p(\mathbf{b}_n | y_{SD,n}, \mathbf{L}_{CD}^{ext}(\mathbf{g}_n))$ 可進一步分解為：

$$\begin{aligned} p(\mathbf{b}_n | y_{SD,n}, \mathbf{L}_{CD}^{ext}(\mathbf{g}_n)) &= \frac{p(\mathbf{b}_n, y_{SD,n}, \mathbf{L}_{CD}^{ext}(\mathbf{g}_n))}{p(y_{SD,n}, \mathbf{L}_{CD}^{ext}(\mathbf{g}_n))} \\ &= C \cdot p(\mathbf{b}_n, y_{SD,n}, \mathbf{L}_{CD}^{ext}(\mathbf{g}_n)) \\ &= C \cdot p(\mathbf{b}_n) \cdot p(y_{SD,n}, \mathbf{L}_{CD}^{ext}(\mathbf{g}_n) | \mathbf{b}_n) \\ &= C \cdot p(\mathbf{b}_n) \cdot p(y_{SD,n} | \mathbf{b}_n) \cdot p(\mathbf{L}_{CD}^{ext}(\mathbf{g}_n) | \mathbf{b}_n) \end{aligned} \quad (3.20)$$

在第二章裡提到，假設來源端到中繼端、來源端到目的端、中繼端到目的端這三條通道互為獨立的狀態，即 $y_{SD,n}$ 與 $\mathbf{L}_{CD}^{ext}(\mathbf{g}_n)$ 分別對於 \mathbf{b}_n 來說是獨立的，所以 $p(y_{SD,n}, \mathbf{L}_{CD}^{ext}(\mathbf{g}_n) | \mathbf{b}_n)$ 可以分解為 $p(y_{SD,n} | \mathbf{b}_n)$ 跟 $p(\mathbf{L}_{CD}^{ext}(\mathbf{g}_n) | \mathbf{b}_n)$ 這兩項。對於 $p(y_{SD,n} | \mathbf{b}_n)$ ，可以分解為：

$$\begin{aligned} p(y_{SD,n} | \mathbf{b}_n) &= \frac{1}{C} \cdot \sum_{X_{S,n} \in \{\pm 1\}} p(y_{SD,n}, \mathbf{b}_n, X_{S,n}) \\ &= \frac{1}{C} \cdot \sum_{X_{S,n} \in \{\pm 1\}} p(y_{SD,n} | X_{S,n}) \cdot p(\mathbf{b}_n | X_{S,n}) \cdot p(X_{S,n}) \\ &= \frac{1}{C} \cdot \sum_{X_{S,n} \in \{\pm 1\}} p(y_{SD,n} | X_{S,n}) \cdot p(\mathbf{b}_n | X_{S,n}) \end{aligned} \quad (3.21)$$

(3.21)式的推導也是基於獨立的假設， $p(y_{SD,n} | X_{S,n})$ 表示通道相關資訊， $p(\mathbf{b}_n | X_{S,n})$ 表

示經由來源端送出的信號 $X_{s,n}$ ，經過 AWGN 通道干擾以及量化後的轉移機率，有關轉移機率的說明如下：

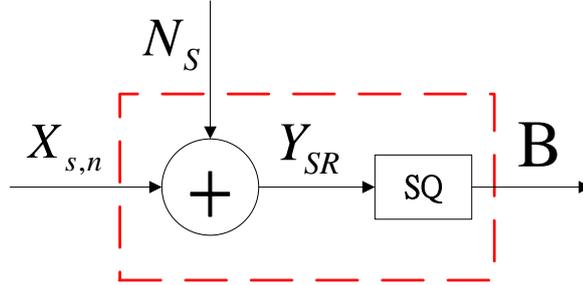


圖 3.3 AWGN 的量化輸出通道[22]

如圖 3.3，若發射端的訊號經由 BPSK 調變為 $X_{s,n}$ 後送出，接收端收到被離散無記憶的 AWGN 通道(雜訊的變異數為 σ^2)干擾後的序列 Y_{SR} ，然後將之量化為 \mathbf{B} 。對於純量量化器而言，會有 $K-1$ 個門檻值 $q = [q_1, q_2, \dots, q_{K-1}] \in \mathbb{R}^{K-1}$ 將訊號範圍切割為 K 個分區 $-\infty := q_0 < q_1 < q_2 < \dots < q_{K-1} < q_K := \infty$ ，所造成的量化轉移機率則為：

$$P(\mathbf{B} = \mathbf{b}_n | X = X_{s,n}) = Q\left(\frac{q_{n-1} - X_{s,n}}{\sigma}\right) - Q\left(\frac{q_n - X_{s,n}}{\sigma}\right) = p(\mathbf{b}_n | X_{s,n}) \quad (3.22)$$

對於 $p(\mathbf{L}_{CD}^{ext}(\mathbf{g}_n) | \mathbf{b}_n)$ ，可以分解為：

$$p(\mathbf{L}_{CD}^{ext}(\mathbf{g}_n) | \mathbf{b}_n) = \prod_{k=1}^{m+1} p(L_{CD}^{ext}(g_{n,k}) | g_{n,k}) = \prod_{k=1}^{m+1} \frac{e^{(1-g_{n,k})L_{CD}^{ext}(g_{n,k})}}{1 + e^{L_{CD}^{ext}(g_{n,k})}} \quad (3.23)$$

而額外訊息的部分，會由訊源解碼器傳出，經由交錯器後引入通道解碼器裡使用，對於額外訊息 $\mathbf{L}_{SD}^{ext}(\mathbf{G})$ 定義為：

$$\mathbf{L}_{SD}^{ext}(\mathbf{G}) = \mathbf{L}_{SD}(\mathbf{G}) - \mathbf{L}_{CD}^{ext}(\mathbf{G}) \quad (3.24)$$

由(3.19)式算出的 $\mathbf{L}_{SD}(\mathbf{G})$ 直接扣除由通道解碼器送進來的額外訊息 $\mathbf{L}_{CD}^{ext}(\mathbf{G})$ ，以保證疊代過程中通道解碼器不會使用到重複的訊息。

第4章 基於多層級分散式訊源編碼的壓縮傳遞機制

第三章所述的系統[15]在實作上為確保疊代解碼機制得以運作，因此對於中繼端及來源端至目的端的通道訊雜比有許多的限制。有鑑於此，本章將提出可推廣至來源端至目的端為低訊雜比時的編碼機制。在前人研究[17]中，提出一種編解碼架構，對接收之來源信號進行量化，配合使用一個碼率 1/2 的迴旋碼編碼後送出。我們將以這個架構為基礎，結合分散式訊源編碼的同位子法(parity approach)及多層級編碼(multi-level coding)的概念，進一步提升中繼端的解碼效能。以下我們將簡略介紹此架構作為中繼端傳輸方案的作法，之後詳述分散式訊源編碼理論如何結合多層級編碼的製作過程。

4.1 中繼端傳輸架構

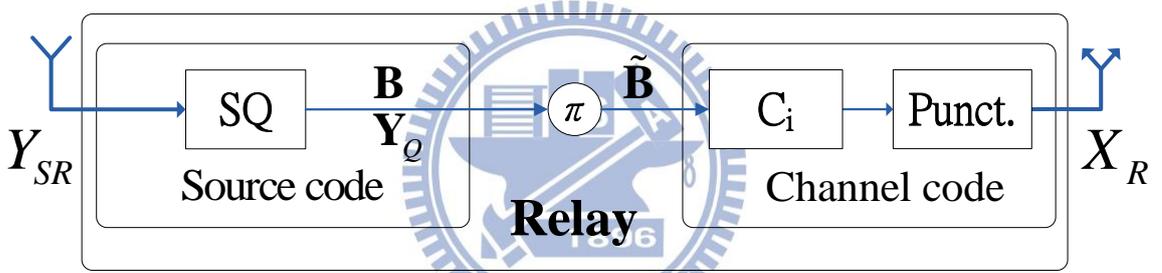


圖 4.1 中繼端傳輸架構

系統架構如圖 4.1 所示，與前一章的差別在於訊源編碼器只有使用量化器，沒有添加其他冗餘碼。作法也是將收到被 AWGN 通道干擾的來源端信號序列 $Y_{SR,1}^N = [y_{SR,1}, y_{SR,2}, \dots, y_{SR,n}, \dots, y_{SR,N}]$, $n=1, 2, \dots, N$ 進行量化，每個信號點 $y_{SR,n}$ 經由純量量化器產生一個 m 位元的量化索引 \mathbf{b}_n 以及相對應的重建信號值 $y_{Q,n}$ 。以下我們定義：

$$\begin{aligned} \mathbf{Y}_{Q,1}^N &= [y_{Q,1}, y_{Q,2}, \dots, y_{Q,n}, \dots, y_{Q,N}] \\ \mathbf{B} &= [\mathbf{b}_1, \mathbf{b}_2, \dots, \mathbf{b}_n, \dots, \mathbf{b}_N] \\ y_{Q,n} &\equiv \mathbf{b}_n = [b_{n,1}, b_{n,2}, \dots, b_{n,k}, \dots, b_{n,m}] \end{aligned} \quad (4.1)$$

$\mathbf{Y}_{Q,1}^N$ 與 \mathbf{B} 分別代表 N 個重建信號值及 N 個量化索引。粗體字表示內含多個位元的集合， $b_{n,k}$ 代表第 n 個索引值裡的第 k 個位元。然後將 \mathbf{B} 通過一個位元交錯器產生 $\tilde{\mathbf{B}} = [\tilde{b}_1, \tilde{b}_2, \dots, \tilde{b}_l, \dots, \tilde{b}_L]$ ，其中交錯器的大小 $L = mN$ ， \tilde{b}_l 對應到一個通過交錯器後的 $b_{n,k}$ ：

$$\tilde{b}_l = \pi(b_{n,k}), \quad l=1,2,\dots,L \quad (4.2)$$

接著把 $\tilde{\mathbf{B}}$ 通過一碼率 1/2 的遞迴系統性迴旋碼 \mathbf{C}_i 編碼後產生相對應的碼字序列 $\mathbf{C} = [\tilde{\mathbf{B}}, \tilde{\mathbf{B}}^p]$ ，其中訊息位元 $\tilde{\mathbf{B}}$ 及同位位元 $\tilde{\mathbf{B}}^p$ 分別為：

$$\begin{aligned} \tilde{\mathbf{B}} &= [\tilde{b}_1, \tilde{b}_2, \dots, \tilde{b}_l, \dots, \tilde{b}_{\hat{L}}], \quad l=1,2,\dots,\hat{L} \\ \tilde{\mathbf{B}}^p &= [\tilde{b}_1^p, \tilde{b}_2^p, \dots, \tilde{b}_l^p, \dots, \tilde{b}_{\hat{L}}^p], \quad l=1,2,\dots,\hat{L} \end{aligned} \quad (4.3)$$

此處長度 \hat{L} 包含了 L 個輸入位元以及 $\hat{L} - L$ 個終止位元，接著把 \mathbf{C} 通過一個穿孔後產生 $3mN/2$ 個位元的輸出。接著我們進行 BPSK 調變後產生 X_R ，再經由 AWGN 通道傳送到目的端。

4.2 目的端解碼架構

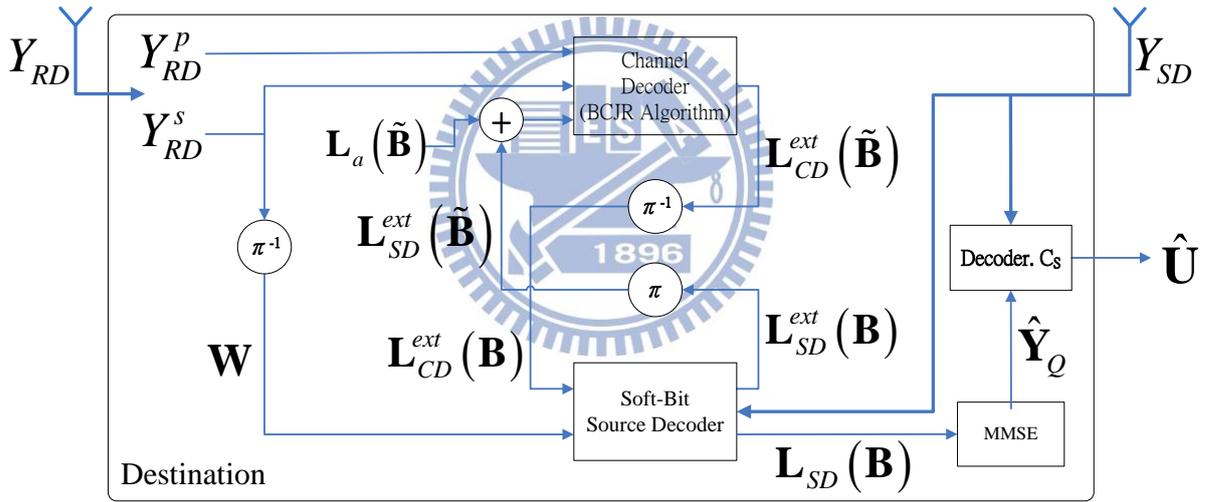


圖 4.2 目的端解碼架構

如圖 4.2 所示，目的端收到直接鏈結路徑的 Y_{SD} 與中繼端路徑的 Y_{RD} 後，會先將 Y_{RD} 序列解穿孔還原如下：

$$Y_{RD,l}^{\hat{L}} = [y_{RD,1}, y_{RD,2}, \dots, y_{RD,l}, \dots, y_{RD,\hat{L}}] \equiv [Y_{RD}^s, Y_{RD}^p] \quad (4.4)$$

其中

$$y_{RD,l} = [y_{RD,l}^s, y_{RD,l}^p], \quad l=1,2,\dots,\hat{L} \quad (4.5)$$

假設我們的通道為 AWGN 的無記憶通道，在通道雜訊比為 $\frac{E_b}{N_0}$ 、雜訊的變異數為 $\sigma^2 = \frac{N_0}{2}$ 。

此時序列第 l 位置的訊息位元的通道訊息 $p(y_{RD,l}^s | \tilde{b}_l)$ 與同位位元的通道訊息 $p(y_{RD,l}^p | \tilde{b}_l^p)$ 可由以下公式計算：

$$\begin{aligned} p(y_{RD,l}^s | \tilde{b}_l) &= \frac{1}{\sqrt{2\pi\sigma^2}} \cdot \exp\left(-\frac{E_b}{N_0} (y_{RD,l}^s - \tilde{b}_l)^2\right) \\ p(y_{RD,l}^p | \tilde{b}_l^p) &= \frac{1}{\sqrt{2\pi\sigma^2}} \cdot \exp\left(-\frac{E_b}{N_0} (y_{RD,l}^p - \tilde{b}_l^p)^2\right) \end{aligned} \quad (4.6)$$

在疊代訊源通道解碼的過程中，訊源解碼器與通道解碼器皆會收到量化索引的通道資訊 Y_{RD}^s ，我們稱為平行串接結構(parallel concatenated scheme)。關於通道解碼器，雖然在中繼端有做穿刺後傳出，但是並不影響通道解碼器的做法。關於通道解碼器與訊源解碼器的說明將在下兩節介紹。以下說明疊代解碼的步驟：

1. 初始化：設定疊代次數 $iter = 1$ ，訊源解碼額外訊息的初始值為 $\mathbf{L}_{SD}^{ext}(\mathbf{B}) = 0$ ，其交錯值為 $\mathbf{L}_{SD}^{ext}(\tilde{\mathbf{B}}) = 0$ 。定義 $\mathbf{L}_{SD}^{ext}(\mathbf{B})$ 與 $\mathbf{L}_{SD}^{ext}(\tilde{\mathbf{B}})$ ：

$$\begin{aligned} \mathbf{L}_{SD}^{ext}(\mathbf{B}) &= [L_{SD}^{ext}(b_{1,1}), L_{SD}^{ext}(b_{1,2}), \dots, L_{SD}^{ext}(b_{n,k}), \dots, L_{SD}^{ext}(b_{N,m})] \\ \mathbf{L}_{SD}^{ext}(\tilde{\mathbf{B}}) &= [L_{SD}^{ext}(\tilde{b}_1), L_{SD}^{ext}(\tilde{b}_2), \dots, L_{SD}^{ext}(\tilde{b}_l), \dots, L_{SD}^{ext}(\tilde{b}_L)] \end{aligned} \quad (4.7)$$

其中

$$L_{SD}^{ext}(\tilde{b}_l) = \log \frac{p_{SD}^{ext}(\tilde{b}_l = +1)}{p_{SD}^{ext}(\tilde{b}_l = -1)} \quad (4.8)$$

2. 通道解碼部分：三個輸入分別是源自中繼端傳送的 $Y_{RD,1}^L$ 、事前訊息 $\mathbf{L}_a(\tilde{\mathbf{B}})$ 與訊源解碼額外訊息的交錯值 $\mathbf{L}_{SD}^{ext}(\tilde{\mathbf{B}})$ 。通道解碼器使用 BCJR 演算法，計算每個位元的通道解碼訊息 $\mathbf{L}_{CD}^{ext}(\tilde{\mathbf{B}})$ ，再通過位元解交錯器成為 $\mathbf{L}_{CD}^{ext}(\mathbf{B})$ 後提供給訊源解碼器使用。定義 $\mathbf{L}_{CD}^{ext}(\tilde{\mathbf{B}})$ 與 $\mathbf{L}_{CD}^{ext}(\mathbf{B})$ ：

$$\begin{aligned} \mathbf{L}_{CD}^{ext}(\tilde{\mathbf{B}}) &= [L_{CD}^{ext}(\tilde{b}_1), L_{CD}^{ext}(\tilde{b}_2), \dots, L_{CD}^{ext}(\tilde{b}_l), \dots, L_{CD}^{ext}(\tilde{b}_L)] \\ \mathbf{L}_{CD}^{ext}(\mathbf{B}) &= [L_{CD}^{ext}(b_{1,1}), L_{CD}^{ext}(b_{1,2}), \dots, L_{CD}^{ext}(b_{n,k}), \dots, L_{CD}^{ext}(b_{N,m})] \end{aligned} \quad (4.9)$$

其中

$$L_{CD}^{ext}(\tilde{b}_l) = \log \frac{p_{CD}^{ext}(\tilde{b}_l = +1)}{p_{CD}^{ext}(\tilde{b}_l = -1)} \quad (4.10)$$

3. 訊源解碼部分：三個輸入分別是直接鏈結路經的 $Y_{SD,1}^N$ 、通道解碼額外訊息 $\mathbf{L}_{CD}^{ext}(\mathbf{B})$ 與 Y_{RD}^s 解交錯後得到的資訊 $\mathbf{W}=[\mathbf{w}_1, \mathbf{w}_2, \dots, \mathbf{w}_n, \dots, \mathbf{w}_N]$ ，我們稱為量化索引通道訊息，據以計算出索引值的後驗對數相似比率 $\mathbf{L}_{SD}(\mathbf{B})=[L(\tilde{b}_1), L(\tilde{b}_2), \dots, L(\tilde{b}_n), \dots, L(\tilde{b}_k)]$ 。至於訊源編碼額外訊息 $\mathbf{L}_{SD}^{ext}(\mathbf{B})$ ，則是將 $\mathbf{L}_{SD}(\mathbf{B})$ 減去通道編碼額外訊息 $\mathbf{L}_{CD}^{ext}(\mathbf{B})$ 與量化索引通道訊息 $\mathbf{L}(\mathbf{W}|\mathbf{B})$ 。接著將 $\mathbf{L}_{SD}^{ext}(\mathbf{B})$ 通過交錯器得到 $\mathbf{L}_{CD}^{ext}(\tilde{\mathbf{B}})$ 再回傳給通道解碼器當事前訊息使用。定義 $\mathbf{L}(\mathbf{W}|\mathbf{B})$ ：

$$\mathbf{L}(\mathbf{W}|\mathbf{B})=[L(w_{1,1}|b_{1,1}), \dots, L(w_{n,k}|b_{n,l}), \dots, L(w_{N,m}|b_{N,m})] \quad (4.11)$$

其中

$$L(w_{n,k}|b_{n,l})=\log \frac{p(w_{n,k}|b_{n,l}=+1)}{p(w_{n,k}|b_{n,l}=-1)} \quad (4.12)$$

4. 下一次的疊代，如果未達最大疊代次數 `iter_max`，則回到步驟二繼續進行，反之則進行步驟五。
5. 利用步驟三算出的後驗對數相似比率 $\mathbf{L}_{SD}(\mathbf{B})$ 得到事後機率 $p(\mathbf{b}_n|y_{SD,n}, \mathbf{L}_{CD}^{ext}(\mathbf{b}_n), \tilde{\mathbf{b}}_n)$ ，再使用最小均方誤差準則來合成新的重建信號值 \hat{Y}_Q ：

$$\hat{Y}_Q(n)=\sum_{Y_Q(k) \in \Psi} p(\mathbf{b}_n|y_{SD,n}, \mathbf{L}_{CD}^{ext}(\mathbf{b}_n), \tilde{\mathbf{b}}_n) \cdot Y_Q(k) \quad (4.13)$$

此處的 Ψ 表示所有量化索引重建值的集合。

4.3 BCJR 通道解碼演算法

在 3.3 節時我們推導了碼率 1 迴旋碼的事後機率以及額外訊息，我們接著要推導碼率 1/2 迴旋碼的事後機率以及額外訊息的計算方式。由於推導類似，所以我們省略一些推導步驟。先考慮 $\tilde{b}_l = +1$ 的情況，而 $p(\tilde{b}_l = +1|Y_{RD,1}^L)$ 與 $p(\tilde{b}_l = -1|Y_{RD,1}^L)$ 的和等於 1。首先展開 $p(\tilde{b}_l = +1|Y_{RD,1}^L)$ 為：

$$p(\tilde{b}_l = +1|Y_{RD,1}^L)=C \cdot \sum_{(s_{l-1}, s_l) \in \Sigma_l^+} p(s_{l-1}, s_l, Y_{RD,1}^L) \quad (4.14)$$

其中 C 為一常數， s_{l-1} 與 s_l 分別為前一個時刻跟目前時刻的狀態， Σ_l^+ 表示在 $s_{n-1} \rightarrow s_n$ 間所有對應於 $\tilde{b}_l = +1$ 的狀態轉移分支的集合，如圖 2.5 所示。因為假設為無記憶通道且輸入位元 \tilde{b}_l 的獨立性，所以 $p(s_{l-1}, s_l, Y_{RD,l}^{\tilde{L}})$ 可被分解為三項：

$$\begin{aligned} p(s_{l-1}, s_l, Y_{RD,l}^{\tilde{L}}) &= p(s_{l-1}, Y_{RD,l}^{l-1}) \cdot p(s_l, y_{RD,l} | s_{l-1}) \cdot p(Y_{RD,l+1}^{\tilde{L}} | s_l) \\ &\triangleq \alpha_{l-1}(s_{l-1}) \cdot \gamma_l(s_{l-1}, s_l) \cdot \beta_l(s_l) \end{aligned} \quad (4.15)$$

其中前項計量 $\alpha_l(s_l)$ 與後向計量 $\beta_{l-1}(s_{l-1})$ 亦分別如(2. 9)、(2. 10)式所述。而分支計量 $\gamma_l(s_{l-1}, s_l)$ 的部分如下：

$$\begin{aligned} \gamma_l(s_{l-1}, s_l) &= p(s_l, y_{RD,l} | s_{l-1}) \\ &= p(s_l | s_{l-1}) \cdot p(y_{RD,l} | s_{l-1}, s_l) \\ &= p(\tilde{b}_l = +1) \cdot p(y_{RD,l} | \tilde{b}_l = +1, \tilde{b}_l^p) \\ &= p(\tilde{b}_l = +1) \cdot p(y_{RD,l}^s | \tilde{b}_l = +1) \cdot p(y_{RD,l}^p | \tilde{b}_l^p) \end{aligned} \quad (4.16)$$

接著我們將說明額外訊息 $L_{CD}^{ext}(\tilde{b}_l)$ 的部分，此額外訊息會經由解交錯器送到訊源解碼器。

由於 $p_{CD}^{ext}(\tilde{b}_l = +1)$ 與 $p_{CD}^{ext}(\tilde{b}_l = -1)$ 的和不一定為 1，所以必須分開計算。由(4.16)式：

$$\begin{aligned} \gamma_l(s_{l-1}, s_l) &= p(\tilde{b}_l = +1) \cdot p(y_{RD,l}^s | \tilde{b}_l = +1) \cdot p(y_{RD,l}^p | \tilde{b}_l^p) \\ &\triangleq p(\tilde{b}_l = +1) \cdot p(y_{RD,l}^s | \tilde{b}_l = +1) \cdot \gamma_l^{ext}(s_{l-1}, s_l) \end{aligned} \quad (4.17)$$

此處 $\gamma_l^{ext}(s_{l-1}, s_l) = p(y_{RD,l}^p | \tilde{b}_l^p)$ 。而(4.15)式與(4.17)式及(4.14)式結合後，(4.14)式變為：

$$\begin{aligned} p(\tilde{b}_l = +1 | Y_{RD,l}^{\tilde{L}}) &= C \cdot \sum_{(s_{l-1}, s_l) \in \Sigma_l^+} p(s_{l-1}, s_l, Y_{RD,l}^{\tilde{L}}) \\ &= C \cdot \sum_{(s_{l-1}, s_l) \in \Sigma_l^+} \alpha_{l-1}(s_{l-1}) \cdot \gamma_l(s_{l-1}, s_l) \cdot \beta_l(s_l) \\ &= C \cdot p(\tilde{b}_l = +1) \cdot p(y_{RD,l}^s | \tilde{b}_l = +1) \\ &\quad \cdot \sum_{(s_{l-1}, s_l) \in \Sigma_l^+} \alpha_{l-1}(s_{l-1}) \cdot \gamma_l^{ext}(s_{l-1}, s_l) \cdot \beta_l(s_l) \\ &= C \cdot p(\tilde{b}_l = +1) \cdot p(y_{RD,l}^s | \tilde{b}_l = +1) \cdot p_{CD}^{ext}(\tilde{b}_l = +1) \end{aligned} \quad (4.18)$$

其中 $p_{CD}^{ext}(\tilde{b}_l = +1) = \sum_{(s_{l-1}, s_l) \in \Sigma_l^+} \alpha_{l-1}(s_{l-1}) \cdot \gamma_l^{ext}(s_{l-1}, s_l) \cdot \beta_l(s_l) = +1$ 即為額外訊息的部分。在進行

一次疊代解碼後，通道解碼器的事前訊息可由訊源解碼器回傳的額外訊息 $p_{SD}^{ext}(\tilde{b}_l = +1)$

得以更新：

$$p(\tilde{b}_l = +1) \leftarrow \left[p(\tilde{b}_l = +1) \cdot p_{SD}^{ext}(\tilde{b}_l = +1) \right] \quad (4.19)$$

故分支計量修正為：

$$\gamma_l(s_{l-1}, s_l) = p(\tilde{b}_l = +1) \cdot p_{SD}^{ext}(\tilde{b}_l = +1) \cdot p(y_{RD,l}^s | \tilde{b}_l = +1) \cdot \gamma_l^{ext}(s_{l-1}, s_l) \quad (4.20)$$

而事後機率為：

$$p(\tilde{b}_l = +1 | Y_{RD,1}^L) = C \cdot p(\tilde{b}_l = +1) \cdot p_{SD}^{ext}(\tilde{b}_l = +1) \cdot p(y_{RD,l}^s | \tilde{b}_l = +1) \cdot p_{CD}^{ext}(\tilde{b}_l = +1) \quad (4.21)$$

4.4 軟式位元訊源解碼演算法

軟式位元訊源解碼器的部分，因為存在平行串接結構，所以訊源解碼器也會收到有關量化索引通道資訊 \mathbf{W} ， \mathbf{W} 是由解穿刺信號序列 $Y_{RD,1}^L$ 的訊息位元 Y_{RD}^s 的通道訊息解交錯後得到。整合運用直接鏈結路徑的信號點 $y_{SD,n}$ 、通道解碼器送來的額外訊息 $\mathbf{L}_{CD}^{ext}(\mathbf{b}_n)$ 以及每個量化索引通道資訊 \mathbf{w}_n ，我們想解出每個 $b_{n,k}$ 位元的對數相似比率。以下為訊源解碼演算法：

$$L(b_{n,k}) = \log \frac{\sum_{\mathbf{b}_n \rightarrow b_{n,k}=0} p(\mathbf{b}_n | y_{SD,n}, \mathbf{L}_{CD}^{ext}(\mathbf{b}_n), \mathbf{w}_n)}{\sum_{\mathbf{b}_n \rightarrow b_{n,k}=1} p(\mathbf{b}_n | y_{SD,n}, \mathbf{L}_{CD}^{ext}(\mathbf{b}_n), \mathbf{w}_n)} \quad (4.22)$$

此處 $\mathbf{b}_n \rightarrow b_{n,k}=0$ 與 $\mathbf{b}_n \rightarrow b_{n,k}=1$ 表示 \mathbf{b}_n 分別對應到位元 $b_{n,k}$ 等於 0 跟 1，而

$p(\mathbf{b}_n | y_{SD,n}, \mathbf{L}_{CD}^{ext}(\mathbf{b}_n), \mathbf{w}_n)$ 可被分解為：

$$\begin{aligned} p(\mathbf{b}_n | y_{SD,n}, \mathbf{L}_{CD}^{ext}(\mathbf{b}_n), \mathbf{w}_n) &= \frac{p(\mathbf{b}_n, y_{SD,n}, \mathbf{L}_{CD}^{ext}(\mathbf{b}_n), \mathbf{w}_n)}{p(y_{SD,n}, \mathbf{L}_{CD}^{ext}(\mathbf{b}_n), \mathbf{w}_n)} \\ &= C \cdot p(\mathbf{b}_n, y_{SD,n}, \mathbf{L}_{CD}^{ext}(\mathbf{b}_n), \mathbf{w}_n) \\ &= C \cdot p(\mathbf{b}_n) \cdot p(y_{SD,n}, \mathbf{L}_{CD}^{ext}(\mathbf{b}_n), \mathbf{w}_n | \mathbf{b}_n) \\ &= C \cdot p(\mathbf{b}_n) \cdot p(y_{SD,n} | \mathbf{b}_n) \cdot p(\mathbf{L}_{CD}^{ext}(\mathbf{b}_n) | \mathbf{b}_n) \cdot p(\mathbf{w}_n | \mathbf{b}_n) \end{aligned} \quad (4.23)$$

由於假設來源端到中繼端、來源端到目的端、中繼端到目的端這三條通道是互為獨立的狀態，所以 $p(y_{SD,n}, \mathbf{L}_{CD}^{ext}(\mathbf{b}_n), \mathbf{w}_n | \mathbf{b}_n)$ 可以分解為 $p(y_{SD,n} | \mathbf{b}_n)$ 、 $p(\mathbf{L}_{CD}^{ext}(\mathbf{b}_n) | \mathbf{b}_n)$ 跟 $p(\mathbf{w}_n | \mathbf{b}_n)$ 這三項。如同前一章的訊源解碼器推導，對於 $p(y_{SD,n} | \mathbf{b}_n)$ ，可以分解為：

$$\begin{aligned}
p(y_{SD,n} | \mathbf{b}_n) &= \frac{1}{c} \cdot \sum_{X_{S,n} \in \{\pm 1\}} p(y_{SD,n}, \mathbf{b}_n, X_{S,n}) \\
&= \frac{1}{c} \cdot \sum_{X_{S,n} \in \{\pm 1\}} p(y_{SD,n} | X_{S,n}) \cdot p(\mathbf{b}_n | X_{S,n}) \cdot p(X_{S,n}) \\
&= \frac{1}{c} \cdot \sum_{X_{S,n} \in \{\pm 1\}} p(y_{SD,n} | X_{S,n}) \cdot p(\mathbf{b}_n | X_{S,n})
\end{aligned} \tag{4.24}$$

對於 $p(\mathbf{L}_{CD}^{ext}(\mathbf{g}_n) | \mathbf{b}_n)$ ，可以分解為：

$$p(\mathbf{L}_{CD}^{ext}(\mathbf{b}_n) | \mathbf{b}_n) = \prod_{k=1}^m p(L_{CD}^{ext}(b_{n,k}) | b_{n,k}) = \prod_{k=1}^{m+1} \frac{e^{(1-b_{n,k})L_{CD}^{ext}(b_{n,k})}}{1 + e^{L_{CD}^{ext}(b_{n,k})}} \tag{4.25}$$

對於量化索引的通道訊息 $p(\mathbf{w}_n | \mathbf{b}_n)$ 可以分解為：

$$p(\mathbf{w}_n | \mathbf{b}_n) = \prod_{k=1}^m p(w_{n,k} | b_{n,k}) \tag{4.26}$$

其中

$$p(w_{n,k} | b_{n,k}) = \frac{1}{\sqrt{2\pi\sigma^2}} \cdot \exp\left(-\frac{E_b}{N_0} (w_{n,k} - b_{n,k})^2\right) \tag{4.27}$$

而額外訊息的部分，會由訊源解碼器傳出，經由交錯器後引入通道解碼器裡使用。對於額外訊息 $\mathbf{L}_{SD}^{ext}(\mathbf{B})$ 定義為：

$$\mathbf{L}_{SD}^{ext}(\mathbf{B}) = \mathbf{L}_{SD}(\mathbf{B}) - \mathbf{L}_{CD}^{ext}(\mathbf{B}) - \mathbf{L}(\mathbf{W} | \mathbf{B}) \tag{4.28}$$

由(4.22)式算出的 $\mathbf{L}_{SD}(\mathbf{B})$ 直接扣除由通道解碼器送進來的額外訊息 $\mathbf{L}_{CD}^{ext}(\mathbf{B})$ 與量化索引通道資訊 $\mathbf{L}(\mathbf{W} | \mathbf{B})$ ，其中(4.11)式為有關量化索引通道資訊的定義，以保證疊代過程中通道解碼器不會使用到重覆的訊息。

4.5 分散式訊源編碼理論

分散式訊源編碼(distributed source coding, DSC)源自 1970 年代 Slepian 和 Wolf 的相關研究，基於訊息熵(entropy)值提出兩相關訊號源的無失真壓縮理論[12]。主要是對於兩個有相關性的訊源，可藉著資源共享的合併編碼方式降低其理論熵值。重要的是，即使在獨立編碼的情況下，仍然可以合併解碼的方式得到相同的理論熵值，如圖 4.3 所示。

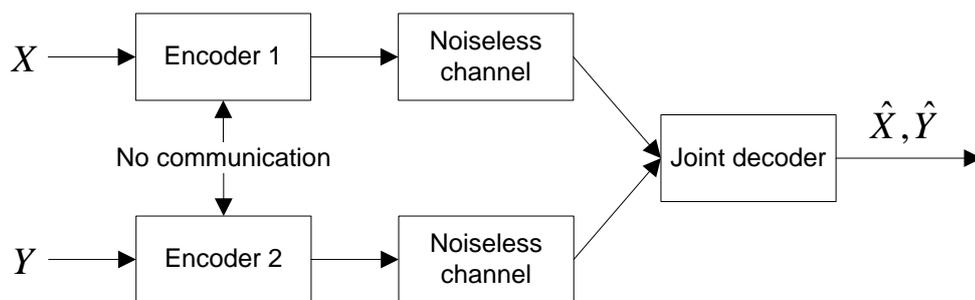


圖 4.3 相關訊源的無失真編碼流程[18]

假設 $H(X)$ 與 $H(Y)$ 分別為 X 跟 Y 兩個訊源的熵值，根據 Shannon 的理論，兩者合併編碼的最小碼率為其合併熵值 $H(X, Y)$ ，而 $H(X, Y) \leq H(X) + H(Y)$ 。而且分散式編碼理論也證明，即使編碼端分別針對 X 與 Y 做獨立編碼(separate encoding)，只要解碼端使用合併編碼(joint decoding)，就不會有碼率的損失，最小碼率仍可以逼近 $H(X, Y)$ 。舉例來說，若 Y 是以熵值 $H(Y)$ 編碼，則 X 的理論熵值可由 $H(X)$ 減少為 $H(X|Y) = H(X, Y) - H(Y)$ 。依此理論設計的無失真分散式訊源編碼器稱為 Slepian-Wolf 編碼器，其碼率範圍如圖 4.4 所示。

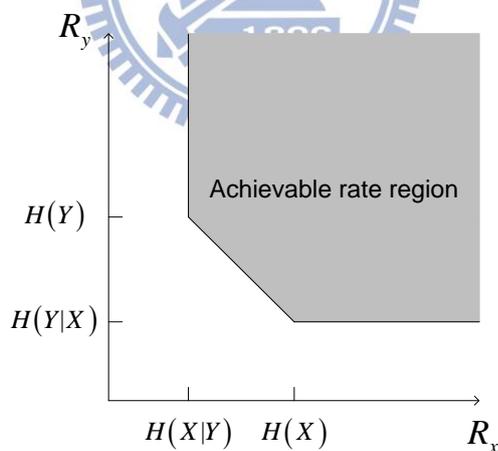


圖 4.4 分散式訊源編碼之碼率範圍[18]

在[19]裡談論有關實現分散式訊源編碼的做法，主要有兩類：一為校驗子法(syndrome approach)，二為同位子法(parity approach)。關於校驗子法，在處理任一 n 位元長度的輸入序列時，藉由傳遞該序列所屬碼組(bin)的 $n-k$ 位元的校驗子，達成 $n:(n-k)$ 的壓縮率。對於同位子法，可以使用碼率 $\frac{k}{n} \leq \frac{1}{2}$ 的系統通道碼生成，在處理任一 k 位元長

度的輸入序列時，只輸出碼字的同位位元部分，達成 $k:(n-k)$ 的壓縮率。相較於校驗子法，同位子的做法較為簡易且無須重新修改解碼演算法。重要的是，在有雜訊干擾的情況下，同位子法比校驗子法要好。因此，接下來要論述的多層級編碼將結合同位子法的概念，應用在中繼端傳輸上。

4.6 基於多層級同位子法的編碼機制

4.6.1 中繼端傳輸架構

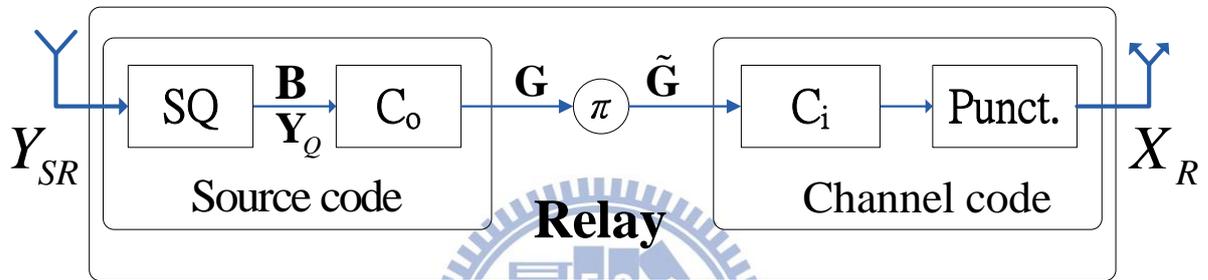


圖 4.5 中繼端傳輸架構

系統架構如圖 4.5，當中繼端收到被 AWGN 通道干擾的來源端信號序列 $Y_{SR,1}^N = [y_{SR,1}, y_{SR,2}, \dots, y_{SR,n}, \dots, y_{SR,N}]$ ， $n=1, 2, \dots, N$ ，每個信號點 $y_{SR,n}$ 將經由純量量化器產生一個 m 位元的量化索引 \mathbf{b}_n 以及相對應的重建信號值 $y_{Q,n}$ 。以下我們定義：

$$\begin{aligned} \mathbf{Y}_{Q,1}^N &= [y_{Q,1}, y_{Q,2}, \dots, y_{Q,n}, \dots, y_{Q,N}] \\ \mathbf{B} &= [\mathbf{b}_1, \mathbf{b}_2, \dots, \mathbf{b}_n, \dots, \mathbf{b}_N] \\ y_{Q,n} &\equiv \mathbf{b}_n = [b_{n,1}, b_{n,2}, \dots, b_{n,k}, \dots, b_{n,m}] \end{aligned} \quad (4.29)$$

接著將量化索引序列 \mathbf{B} 整理為 m 層的構造：

$$\begin{aligned} \mathbf{B} &= [\mathbf{B}_{1,1}^N, \mathbf{B}_{2,1}^N, \dots, \mathbf{B}_{k,1}^N, \dots, \mathbf{B}_{m,1}^N] \\ \mathbf{B}_{k,1}^N &= [b_{1,k}, b_{2,k}, \dots, b_{n,k}, \dots, b_{N,k}] \end{aligned} \quad (4.30)$$

其中 $\mathbf{B}_{1,1}^N$ 表示最高有效位元 (most significant bit, MSB)， $\mathbf{B}_{1,m}^N$ 表示最低有效位元 (least significant bit, LSB)。我們使用的編碼器 \mathbf{C}_o 是由一碼率 $1/2$ 的遞迴系統性迴旋碼而來，然後利用同位子法將這 m 層分別進行獨立通道編碼，並取其同位位元以組合成 \mathbf{G} ：

$$\begin{aligned} \mathbf{G} &= [\mathbf{g}_{1,1}^N, \mathbf{g}_{2,1}^N, \dots, \mathbf{g}_{k,1}^N, \dots, \mathbf{g}_{m,1}^N] \\ \mathbf{g}_{k,1}^N &= [g_{k,1}, g_{k,2}, \dots, g_{k,n}, \dots, g_{k,N}] \\ &= [b_{1,k}^p, b_{2,k}^p, \dots, b_{n,k}^p, \dots, b_{N,k}^p] \end{aligned} \quad (4.31)$$

\mathbf{G} 亦為 m 層，其中 $b_{n,k}^p = g_{k,n}$ 皆代表 $b_{n,k}$ 編碼後產生的同位位元。然後將 \mathbf{G} 通過一個交錯器產生 $\tilde{\mathbf{G}} = [\tilde{g}_1, \tilde{g}_2, \dots, \tilde{g}_l, \dots, \tilde{g}_L]$ ，其中交錯器的大小 $L = mN$ ，而 \tilde{g}_l 對應到一個通過交錯器後的 $g_{k,n}$ ：

$$\tilde{g}_l = \pi(g_{k,n}), \quad l=1,2,\dots,L \quad (4.32)$$

接著把 $\tilde{\mathbf{G}}$ 通過一個碼率 1/2 的遞迴系統性迴旋碼 \mathbf{C}_i 編碼後產生相對應的碼字序列 $\mathbf{C} = [\tilde{\mathbf{G}}, \tilde{\mathbf{G}}^p]$ ，其中同位位元部分 $\tilde{\mathbf{G}}^p$ ：

$$\tilde{\mathbf{G}}^p = [\tilde{g}_1^p, \tilde{g}_2^p, \dots, \tilde{g}_l^p, \dots, \tilde{g}_{\hat{L}}^p], \quad l=1,2,\dots,\hat{L} \quad (4.33)$$

此處長度 \hat{L} 包含 L 個輸入位元以及 $\hat{L} - L$ 個終止位元，接著把 \mathbf{C} 通過一個穿刺後產生 $3mN/2$ 個位元的輸出。接著我們進行 BPSK 調變後產生 X_R ，再經由 AWGN 通道傳送到目的端。



4.6.2 目的端解碼架構

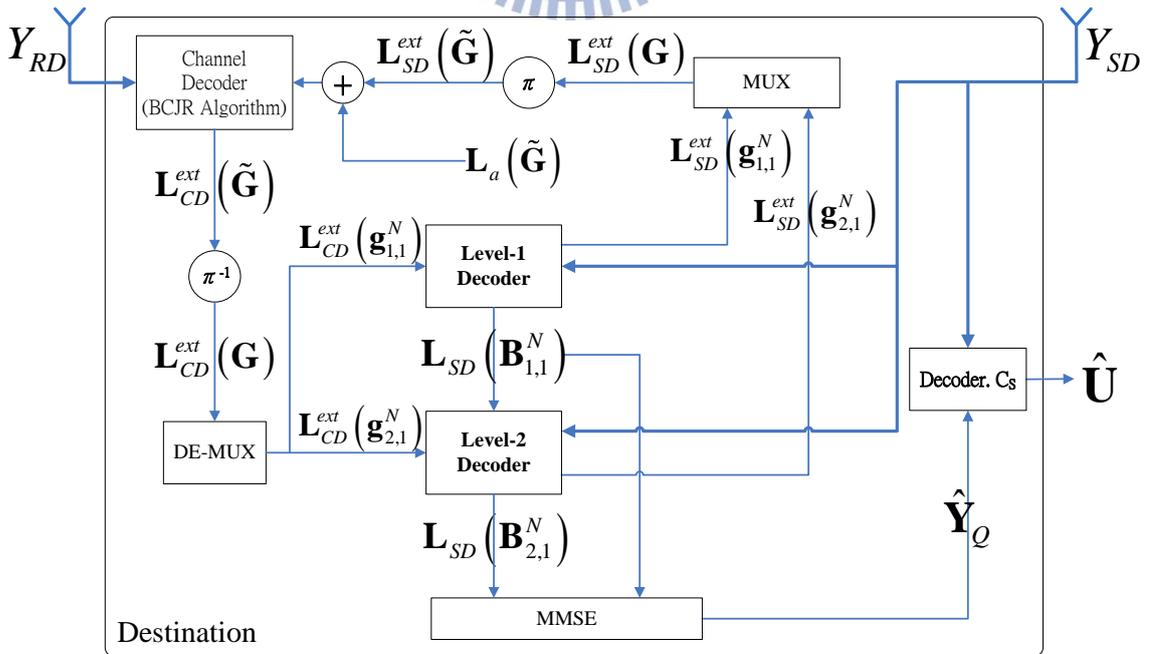


圖 4.6 目的端解碼架構

如圖 4.6 所示，目的端收到直接鏈結路徑的 Y_{SD} 與中繼端路徑的 Y_{RD} 後，如(4.4)式一

樣，會先將 Y_{RD} 序列解穿刺還原如下：

$$\begin{aligned} Y_{RD,l}^{\hat{L}} &= [y_{RD,1}, y_{RD,2}, \dots, y_{RD,l}, \dots, y_{RD,\hat{L}}] \equiv [Y_{RD}^s, Y_{RD}^p] \\ y_{RD,l} &= [y_{RD,l}^s, y_{RD,l}^p], \quad l=1, 2, \dots, \hat{L} \end{aligned} \quad (4.34)$$

假設我們的通道為 AWGN 的無記憶通道，在通道雜訊比為 $\frac{E_b}{N_0}$ 、雜訊的變異數為 $\sigma^2 = \frac{N_0}{2}$ 。

此時序列第 l 位置的通道訊息 $p(y_{RD,l}^s | \tilde{g}_l)$ 與同位位元的通道訊息 $p(y_{RD,l}^p | \tilde{g}_l^p)$ 可以由以下公式計算：

$$\begin{aligned} p(y_{RD,l}^s | \tilde{g}_l) &= \frac{1}{\sqrt{2\pi\sigma^2}} \cdot \exp\left(-\frac{E_b}{N_0} (y_{RD,l}^s - \tilde{g}_l)^2\right) \\ p(y_{RD,l}^p | \tilde{g}_l^p) &= \frac{1}{\sqrt{2\pi\sigma^2}} \cdot \exp\left(-\frac{E_b}{N_0} (y_{RD,l}^p - \tilde{g}_l^p)^2\right) \end{aligned} \quad (4.35)$$

接著我們將說明疊代訊源通道解碼的過程。為了方便說明，關於訊源解碼，我們只討論 2 層的情況，同樣的做法可以擴展到 m 層。有關通道解碼器與訊源解碼器的說明將在下節介紹。以下說明疊代解碼的步驟：

1. 初始化：設定疊代次數 $\text{iter} = 1$ ，訊源解碼額外訊息的初始值為 $\mathbf{L}_{SD}^{\text{ext}}(\mathbf{G}) = 0$ ，其交錯值為 $\mathbf{L}_{SD}^{\text{ext}}(\tilde{\mathbf{G}}) = 0$ 。定義 $\mathbf{L}_{SD}^{\text{ext}}(\mathbf{G})$ 與 $\mathbf{L}_{SD}^{\text{ext}}(\tilde{\mathbf{G}})$ ：

$$\begin{aligned} \mathbf{L}_{SD}^{\text{ext}}(\mathbf{G}) &= [L_{SD}^{\text{ext}}(g_{1,1}), L_{SD}^{\text{ext}}(g_{1,2}), \dots, L_{SD}^{\text{ext}}(g_{n,k}), \dots, L_{SD}^{\text{ext}}(g_{N,2})] \\ \mathbf{L}_{SD}^{\text{ext}}(\tilde{\mathbf{G}}) &= [L_{SD}^{\text{ext}}(\tilde{g}_1), L_{SD}^{\text{ext}}(\tilde{g}_2), \dots, L_{SD}^{\text{ext}}(\tilde{g}_l), \dots, L_{SD}^{\text{ext}}(\tilde{g}_L)] \end{aligned} \quad (4.36)$$

2. 通道解碼部分：三個輸入分別源自中繼端傳送的 $Y_{RD,l}^{\hat{L}}$ 、事前訊息 $\mathbf{L}_a(\tilde{\mathbf{G}})$ 與訊源解碼額外訊息的交錯值 $\mathbf{L}_{SD}^{\text{ext}}(\tilde{\mathbf{G}})$ 。通道解碼器使用 BCJR 演算法，計算每個位元的通道解碼訊息 $\mathbf{L}_{CD}^{\text{ext}}(\tilde{\mathbf{G}})$ ，再通過位元解交錯器成為 $\mathbf{L}_{CD}^{\text{ext}}(\mathbf{G})$ 後提供給訊源解碼器使用。定義 $\mathbf{L}_{CD}^{\text{ext}}(\tilde{\mathbf{G}})$ 與 $\mathbf{L}_{CD}^{\text{ext}}(\mathbf{G})$ ：

$$\begin{aligned} \mathbf{L}_{CD}^{\text{ext}}(\tilde{\mathbf{G}}) &= [L_{CD}^{\text{ext}}(\tilde{g}_1), L_{CD}^{\text{ext}}(\tilde{g}_2), \dots, L_{CD}^{\text{ext}}(\tilde{g}_l), \dots, L_{CD}^{\text{ext}}(\tilde{g}_L)] \\ \mathbf{L}_{CD}^{\text{ext}}(\mathbf{G}) &= [L_{CD}^{\text{ext}}(g_{1,1}), L_{CD}^{\text{ext}}(g_{1,2}), \dots, L_{CD}^{\text{ext}}(g_{n,k}), \dots, L_{CD}^{\text{ext}}(g_{N,2})] \end{aligned} \quad (4.37)$$

其中

$$L_{CD}^{\text{ext}}(\tilde{g}_l) = \log \frac{p_{CD}^{\text{ext}}(\tilde{g}_l = +1)}{p_{CD}^{\text{ext}}(\tilde{g}_l = -1)} \quad (4.38)$$

3. 訊源解碼部分：兩個輸入分別是直接鏈結路徑的 $Y_{SD,1}^N$ 與通道解碼額外訊息 $\mathbf{L}_{CD}^{ext}(\mathbf{G})$ 。因為在中繼端是每層獨立編碼，所以須將通道解碼額外訊息做解多工 (demultiplex) 將 $\mathbf{L}_{CD}^{ext}(\mathbf{G})$ 分成 2 部分得到 $\mathbf{L}_{CD}^{ext}(\mathbf{g}_{1,1}^N)$ 與 $\mathbf{L}_{CD}^{ext}(\mathbf{g}_{2,1}^N)$ 。然後分別輸入至第一層(Level-1)與第二層(Level-2)解碼器裡。解碼順序是先做第一層解碼，第一層解碼器會收到 $\mathbf{L}_{CD}^{ext}(\mathbf{g}_{1,1}^N)$ 與 $Y_{SD,1}^N$ 解碼得到索引位元部分的 $\mathbf{L}_{SD}(\mathbf{B}_{1,1}^N)$ 與同侖子法輸出部分的 $\mathbf{L}_{SD}(\mathbf{g}_{1,1}^N)$ ，其中把 $\mathbf{L}_{SD}(\mathbf{g}_{1,1}^N)$ 減去輸入的 $\mathbf{L}_{CD}^{ext}(\mathbf{g}_{1,1}^N)$ 就得到第一層的訊源解碼額外訊息 $\mathbf{L}_{SD}^{ext}(\mathbf{g}_{1,1}^N)$ ，把 $\mathbf{L}_{SD}(\mathbf{B}_{1,1}^N)$ 做硬式決策(hard decision)得到 $\hat{\mathbf{B}}_{1,1}^N$ 。接著進行第二層解碼，第二層解碼器會收到 $\mathbf{L}_{CD}^{ext}(\mathbf{g}_{2,1}^N)$ 、 $Y_{SD,1}^N$ 與自第一層送來的 $\hat{\mathbf{B}}_{1,1}^N$ ，然後解碼得到 $\mathbf{L}_{SD}(\mathbf{B}_{2,1}^N)$ 與 $\mathbf{L}_{SD}(\mathbf{g}_{2,1}^N)$ 。把 $\mathbf{L}_{SD}(\mathbf{g}_{2,1}^N)$ 減去輸入的 $\mathbf{L}_{CD}^{ext}(\mathbf{g}_{2,1}^N)$ 得到第二層的訊源解碼額外訊息 $\mathbf{L}_{SD}^{ext}(\mathbf{g}_{2,1}^N)$ 。之後利用多工器將額外訊息 $\mathbf{L}_{SD}^{ext}(\mathbf{g}_{1,1}^N)$ 與 $\mathbf{L}_{SD}^{ext}(\mathbf{g}_{2,1}^N)$ 合併為 $\mathbf{L}_{SD}^{ext}(\mathbf{G})$ ，然後將 $\mathbf{L}_{SD}^{ext}(\mathbf{G})$ 通過交錯器得到 $\mathbf{L}_{SD}^{ext}(\tilde{\mathbf{G}})$ 再回傳給通道解碼器當事前訊息使用。
4. 下一次的疊代，如果未達最大疊代次數 $iter_max$ ，則回到步驟二繼續進行，反之則進行步驟五。
5. 利用步驟三算出的後驗對數相似比率 $\mathbf{L}_{SD}(\mathbf{B}_{1,1}^N)$ 與 $\mathbf{L}_{SD}(\mathbf{B}_{2,1}^N)$ 得到事後機率 $p(\mathbf{B} | Y_{SD,1}^N, \mathbf{L}_{CD}^{ext}(\mathbf{G}))$ ，再使用最小均方誤差準則來合成新的重建信號值 \hat{Y}_Q ：

$$\hat{Y}_Q(n) = \sum_{Y_Q(k) \in \Psi} p(\mathbf{B} | Y_{SD,1}^N, \mathbf{L}_{CD}^{ext}(\mathbf{G})) \cdot Y_Q(k) \quad (4.39)$$

此處的 Ψ 表示所有量化索引重建值的集合。

4.6.3 BCJR 通道解碼演算法

這部分我們使用的通道碼與 4.1 節一樣，但由於解碼架構不同，因此訊源解碼器與通道解碼器之間交換的額外訊息的定義也不一樣。所以在這一節我們將說明其額外訊息的算法。由於推導與 4.3 節類似，所以我們省略一些推導步驟。考慮 $\tilde{g}_l = +1$ 的情況，而 $p(\tilde{g}_l = +1 | Y_{RD,1}^L)$ 與 $p(\tilde{g}_l = -1 | Y_{RD,1}^L)$ 的和等於 1。首先展開 $p(\tilde{g}_l = +1 | Y_{RD,1}^L)$ 為：

$$p(\tilde{g}_l = +1 | Y_{RD,1}^{\hat{L}}) = C \cdot \sum_{(s_{l-1}, s_l) \in \Sigma_l^+} p(s_{l-1}, s_l, Y_{RD,1}^{\hat{L}}) \quad (4.40)$$

其中 C 為一常數， s_{l-1} 與 s_l 分別為前一個時刻跟目前時刻的狀態， Σ_l^+ 表示在 $s_{l-1} \rightarrow s_l$ 間所有對應於 $\tilde{g}_l = +1$ 的狀態轉移分支的集合，如圖 2.5 所示。因假設為無記憶通道且輸入位元 \tilde{g}_l 的獨立性，所以 $p(s_{l-1}, s_l, Y_{RD,1}^{\hat{L}})$ 可被分解為三項：

$$\begin{aligned} p(s_{l-1}, s_l, Y_{RD,1}^{\hat{L}}) &= p(s_{l-1}, Y_{RD,1}^{l-1}) \cdot p(s_l, y_{RD,l} | s_{l-1}) \cdot p(Y_{RD,l+1}^{\hat{L}} | s_l) \\ &\triangleq \alpha_{l-1}(s_{l-1}) \cdot \gamma_l(s_{l-1}, s_l) \cdot \beta_l(s_l) \end{aligned} \quad (4.41)$$

其中前項計量 $\alpha_l(s_l)$ 與後向計量 $\beta_{l-1}(s_{l-1})$ 亦分別如(2.9)、(2.10)式所述。而分支計量 $\gamma_l(s_{l-1}, s_l)$ 的部分如下：

$$\begin{aligned} \gamma_l(s_{l-1}, s_l) &= p(s_l, y_{RD,l} | s_{l-1}) \\ &= p(s_l | s_{l-1}) \cdot p(y_{RD,l} | s_{l-1}, s_l) \\ &= p(\tilde{g}_l = +1) \cdot p(y_{RD,l} | \tilde{g}_l = +1, \tilde{g}_l^p) \\ &= p(\tilde{g}_l = +1) \cdot p(y_{RD,l}^s | \tilde{g}_l = +1) \cdot p(y_{RD,l}^p | \tilde{g}_l^p) \end{aligned} \quad (4.42)$$

接著將說明額外訊息 $L_{CD}^{ext}(\tilde{g}_l)$ 的部分，此處額外訊息部分與(4.17)式略有不同，由(4.42)式：

$$\begin{aligned} \gamma_l(s_{l-1}, s_l) &= p(\tilde{g}_l = +1) \cdot p(y_{RD,l}^s | \tilde{g}_l = +1) \cdot p(y_{RD,l}^p | \tilde{g}_l^p) \\ &\triangleq p(\tilde{g}_l = +1) \cdot \gamma_l^{ext}(s_{l-1}, s_l) \end{aligned} \quad (4.43)$$

此處 $\gamma_l^{ext}(s_{l-1}, s_l) = p(y_{RD,l}^s | \tilde{g}_l = +1) \cdot p(y_{RD,l}^p | \tilde{g}_l^p)$ 。而(4.41)式與(4.43)式及(4.40)式結合後，(4.40)式變為：

$$\begin{aligned} p(\tilde{g}_l = +1 | Y_{RD,1}^{\hat{L}}) &= C \cdot \sum_{(s_{l-1}, s_l) \in \Sigma_l^+} p(s_{l-1}, s_l, Y_{RD,1}^{\hat{L}}) \\ &= C \cdot \sum_{(s_{l-1}, s_l) \in \Sigma_l^+} \alpha_{l-1}(s_{l-1}) \cdot \gamma_l(s_{l-1}, s_l) \cdot \beta_l(s_l) \\ &= C \cdot p(\tilde{g}_l = +1) \cdot \sum_{(s_{l-1}, s_l) \in \Sigma_l^+} \alpha_{l-1}(s_{l-1}) \cdot \gamma_l^{ext}(s_{l-1}, s_l) \cdot \beta_l(s_l) \\ &= C \cdot p(\tilde{g}_l = +1) \cdot p_{CD}^{ext}(\tilde{g}_l = +1) \end{aligned} \quad (4.44)$$

其中 $p_{CD}^{ext}(\tilde{g}_l = +1) = \sum_{(s_{l-1}, s_l) \in \Sigma_l^+} \alpha_{l-1}(s_{l-1}) \cdot \gamma_l^{ext}(s_{l-1}, s_l) \cdot \beta_l(s_l)$ 即為額外訊息的部分。在進行一

次疊代解碼後，通道解碼器的事前訊息可由訊源解碼器回傳的額外訊息 $p_{SD}^{ext}(\tilde{g}_l = +1)$ 得以更新：

$$p(\tilde{g}_l = +1) \leftarrow \left[p(\tilde{g}_l = +1) \cdot p_{SD}^{ext}(\tilde{g}_l = +1) \right] \quad (4.45)$$

故分支計量修正為：

$$\gamma_l(s_{l-1}, s_l) = p(\tilde{g}_l = +1) \cdot p_{SD}^{ext}(\tilde{g}_l = +1) \cdot \gamma_l^{ext}(s_{l-1}, s_l) \quad (4.46)$$

而事後機率為：

$$p(\tilde{g}_l = +1 | Y_{RD,1}^L) = C \cdot p(\tilde{g}_l = +1) \cdot p_{SD}^{ext}(\tilde{g}_l = +1) \cdot p_{CD}^{ext}(\tilde{g}_l = +1) \quad (4.47)$$

4.6.4 基於多層級編碼的 BCJR 訊源解碼演算法

在這一節，我們將推導基於分散式訊源編碼中同位子法的多層級解碼。同位子法的原始解碼方式是將雜訊干擾的同位位元部分與系統位元部分合併視為一組完整的接收序列。由於中繼端一開始將接收信號序列做量化的關係，所以不同的地方在於系統位元部分的 $Y_{SD,1}^N$ 是未被量化過的序列，以及我們在解碼過程裡引入了前幾層的資訊幫助目前這一層的解碼。由於是基於原始 BCJR 演算法推得，所以一開始類似的推導將省略。以下我們將分兩部分推導，第一部分為層級訊息的傳遞，第二部分為額外資訊。

對於第 k 層的解碼器，其輸入有直接鏈結路徑的 $Y_{SD,1}^N$ 、第 k 層的通道解碼額外訊息 $\mathbf{L}_{CD}^{ext}(\mathbf{g}_{k,1}^N)$ 與已解出的第 1 至 $k-1$ 層量化索引位元 $[\hat{\mathbf{B}}_{1,1}^N, \dots, \hat{\mathbf{B}}_{k-1,1}^N]$ 。這些資訊用於解出第 k 層的量化索引位元對數相似比率 $L_{SD}(b_{n,k})$ ：

$$L_{SD}(b_{n,k}) = \log \frac{p(b_{n,k} = 0 | Y_{SD,1}^N, \mathbf{L}_{CD}^{ext}(\mathbf{g}_{k,1}^N), \hat{\mathbf{B}}_{1,1}^N, \dots, \hat{\mathbf{B}}_{k-1,1}^N)}{p(b_{n,k} = 1 | Y_{SD,1}^N, \mathbf{L}_{CD}^{ext}(\mathbf{g}_{k,1}^N), \hat{\mathbf{B}}_{1,1}^N, \dots, \hat{\mathbf{B}}_{k-1,1}^N)} \quad (4.48)$$

因為 $p(b_{n,k} = 1 | Y_{SD,1}^N, \mathbf{L}_{CD}^{ext}(\mathbf{g}_{k,1}^N), \hat{\mathbf{B}}_{1,1}^N, \dots, \hat{\mathbf{B}}_{k-1,1}^N) = 1 - p(b_{n,k} = 0 | Y_{SD,1}^N, \mathbf{L}_{CD}^{ext}(\mathbf{g}_{k,1}^N), \hat{\mathbf{B}}_{1,1}^N, \dots, \hat{\mathbf{B}}_{k-1,1}^N)$ ，我們先考慮 $b_{n,k} = 0$ 的情況。首先展開 $p(b_{n,k} = 0 | Y_{SD,1}^N, \mathbf{L}_{CD}^{ext}(\mathbf{g}_{k,1}^N), \hat{\mathbf{B}}_{1,1}^N, \dots, \hat{\mathbf{B}}_{k-1,1}^N)$ 為：

$$\begin{aligned} & p(b_{n,k} = 0 | Y_{SD,1}^N, \mathbf{L}_{CD}^{ext}(\mathbf{g}_{k,1}^N), \hat{\mathbf{B}}_{1,1}^N, \dots, \hat{\mathbf{B}}_{k-1,1}^N) \\ &= C \cdot \sum_{(s_{n-1}, s_n) \in \Sigma_n^+} p(s_{n-1}, s_n, Y_{SD,1}^N, \mathbf{L}_{CD}^{ext}(\mathbf{g}_{k,1}^N), \hat{\mathbf{B}}_{1,1}^N, \dots, \hat{\mathbf{B}}_{k-1,1}^N) \end{aligned} \quad (4.49)$$

其中 C 為一常數， s_{l-1} 與 s_l 分別為前一個時刻跟目前時刻的狀態， Σ_n^+ 表示在 $s_{n-1} \rightarrow s_n$ 間所有對應於 $\tilde{g}_l = +1$ 的狀態轉移分支的集合，如圖 2.5 所示。因為假設為無記憶通道且輸入位元 \tilde{g}_l 的獨立性，所以 $p(s_{n-1}, s_n, Y_{SD,1}^N, \mathbf{L}_{CD}^{ext}(\mathbf{g}_{k,1}^N), \hat{\mathbf{B}}_{1,1}^N, \dots, \hat{\mathbf{B}}_{k-1,1}^N)$ 可被分解成三項：

$$\begin{aligned}
& p\left(s_{n-1}, s_n, Y_{SD,1}^N, \mathbf{L}_{CD}^{ext}(\mathbf{g}_{k,1}), \hat{\mathbf{B}}_{1,1}^N, \dots, \hat{\mathbf{B}}_{k-1,1}^N\right) \\
&= p\left(s_{n-1}, Y_{SD,1}^{n-1}, \mathbf{L}_{CD}^{ext}(\mathbf{g}_{k,1}^{n-1}), \hat{\mathbf{B}}_{1,1}^{n-1}, \dots, \hat{\mathbf{B}}_{k-1,1}^{n-1}\right) \\
&\quad \cdot p\left(s_n, y_{SD,n}, L_{CD}^{ext}(g_{k,n}), \hat{b}_{n,1}, \dots, \hat{b}_{n,k-1} \mid s_{n-1}\right) \\
&\quad \cdot p\left(Y_{SD,n+1}^N, \mathbf{L}_{CD}^{ext}(\mathbf{g}_{k,n+1}), \hat{\mathbf{B}}_{1,n+1}^N, \dots, \hat{\mathbf{B}}_{k-1,n+1}^N \mid s_n\right) \\
&\triangleq \alpha_{n-1}(s_{n-1}) \cdot \gamma_n(s_{n-1}, s_n) \cdot \beta_n(s_n)
\end{aligned} \tag{4.50}$$

其中前項計量 $\alpha_n(s_n)$ 與後向計量 $\beta_{n-1}(s_{n-1})$ 亦分別如(2.9)、(2.10)式所述。而分支計量 $\gamma_n(s_{n-1}, s_n)$ 如下：

$$\begin{aligned}
\gamma_n(s_{n-1}, s_n) &= p\left(s_n, y_{SD,n}, L_{CD}^{ext}(g_{k,n}), \hat{b}_{n,1}, \dots, \hat{b}_{n,k-1} \mid s_{n-1}\right) \\
&= p(s_n \mid s_{n-1}) \cdot p\left(y_{SD,n}, L_{CD}^{ext}(g_{k,n}), \hat{b}_{n,1}, \dots, \hat{b}_{n,k-1} \mid s_{n-1}, s_n\right) \\
&= p(b_{n,k} = 0) \cdot p\left(y_{SD,n}, L_{CD}^{ext}(g_{k,n}), \hat{b}_{n,1}, \dots, \hat{b}_{n,k-1} \mid b_{n,k} = 0, b_{n,k}^p\right) \\
&= p(b_{n,k} = 0) \cdot p\left(L_{CD}^{ext}(g_{k,n}) \mid b_{n,k}^p\right) \cdot p\left(y_{SD,n}, \hat{b}_{n,1}, \dots, \hat{b}_{n,k-1} \mid b_{n,k} = 0\right)
\end{aligned} \tag{4.51}$$

其中 $p\left(y_{SD,n}, \hat{b}_{n,1}, \dots, \hat{b}_{n,k-1} \mid b_{n,k} = 0\right)$ 可以分解為：

$$\begin{aligned}
& p\left(y_{SD,n}, \hat{b}_{n,1}, \dots, \hat{b}_{n,k-1} \mid b_{n,k} = 0\right) \\
&= \frac{1}{p(b_{n,k} = 0)} \cdot p\left(y_{SD,n}, \hat{b}_{n,1}, \dots, \hat{b}_{n,k-1}, b_{n,k} = 0\right) \\
&= \frac{1}{p(b_{n,k} = 0)} \cdot \sum_{X_{S,n} \in \{\pm 1\}} p(X_{S,n}) \cdot p\left(y_{SD,n}, \hat{b}_{n,1}, \dots, \hat{b}_{n,k-1}, b_{n,k} = 0 \mid X_{S,n}\right) \\
&= \frac{1}{p(b_{n,k} = 0)} \cdot \sum_{X_{S,n} \in \{\pm 1\}} p(X_{S,n}) \cdot p\left(y_{SD,n} \mid X_{S,n}\right) \cdot p\left(\hat{b}_{n,1}, \dots, \hat{b}_{n,k-1}, b_{n,k} = 0 \mid X_{S,n}\right)
\end{aligned} \tag{4.52}$$

其中 $p\left(\hat{b}_{n,1}, \dots, \hat{b}_{n,k-1}, b_{n,k} = 0 \mid X_{S,n}\right)$ 我們稱為層級訊息，可以由量化轉移機率計算：

$$p\left(\hat{b}_{n,1}, \dots, \hat{b}_{n,k-1}, b_{n,k} = 0 \mid X_{S,n}\right) = \sum_{\mathbf{b}_n \in \{\hat{b}_{n,1}, \dots, \hat{b}_{n,k-1}, b_{n,k} = 0\}} p(\mathbf{b}_n \mid X_{S,n}) \tag{4.53}$$

故分支計量 $\gamma_n(s_{n-1}, s_n)$ 為：

$$\begin{aligned}
& \gamma_n(s_{n-1}, s_n) \\
&= p\left(L_{CD}^{ext}(g_{k,n}) \mid b_{n,k}^p\right) \\
&\quad \cdot \sum_{X_{S,n} \in \{\pm 1\}} p(X_{S,n}) \cdot p\left(y_{SD,n} \mid X_{S,n}\right) \cdot p\left(\hat{b}_{n,1}, \dots, \hat{b}_{n,k-1}, b_{n,k} = 0 \mid X_{S,n}\right)
\end{aligned} \tag{4.54}$$

最後由(4.48)式算出的 $L_{SD}(b_{n,k})$ ，經由硬式決策得到的 $\hat{b}_{n,k}$ 可以作為下一層解碼時使用。

接著是額外資訊的部分，由於送進通道編碼器是訊源編碼器輸出的同位元部分(即

$b_{n,k}^p$), 所以不同的部分在於 $s_{l-1} \rightarrow s_l$ 間狀態轉移分支的加總對象。在(4.49)式裡是對系統位元 $b_{n,k}$ 進行加總, 在此則是對同位位元 $b_{n,k}^p$ 加總, 對於要算的同位元對數相似比率 $L_{SD}(g_{k,n})$ 為:

$$L_{SD}(g_{k,n}) = \log \frac{p(b_{n,k}^p = 0 | Y_{SD,1}^N, \mathbf{L}_{CD}^{ext}(\mathbf{g}_{k,1}), \hat{\mathbf{B}}_{1,1}^N, \dots, \hat{\mathbf{B}}_{1,k-1}^N)}{p(b_{n,k}^p = 1 | Y_{SD,1}^N, \mathbf{L}_{CD}^{ext}(\mathbf{g}_{k,1}), \hat{\mathbf{B}}_{1,1}^N, \dots, \hat{\mathbf{B}}_{1,k-1}^N)} \quad (4.55)$$

其中 $p(b_{n,k}^p = 0 | Y_{SD,1}^N, \mathbf{L}_{CD}^{ext}(\mathbf{g}_{k,1}), \hat{\mathbf{B}}_{1,1}^N, \dots, \hat{\mathbf{B}}_{1,k-1}^N)$ 部分由於展開作法與(4.48)式一樣所以省略。而算出的 $L_{SD}(g_{k,n})$ 則直接減去由通道解碼器回傳的額外訊息 $L_{CD}^{ext}(g_{k,n})$ 就得到訊源解碼額外訊息:

$$L_{SD}^{ext}(g_{k,n}) = L_{SD}(g_{k,n}) - L_{CD}^{ext}(g_{k,n}) \quad (4.56)$$



第5章 實驗模擬與結果分析

在前面的章節我們介紹了各種中繼端傳輸方案的作法：第一種是以單一同位檢查碼為冗餘碼的架構，第二種是擁有平行串接模式的解碼架構，以及我們所提出的基於分散式訊源編碼中同位子法與多層級編碼的架構。在本章中，我們將各種架構實現於系統模擬，進一步驗證其效能。本章的內容將分三小節，第一節說明基本的環境設定，第二節介紹三種做法的實驗模擬與結果分析，第三節為綜合討論，並比較三種方法的效能優劣。

5.1 環境設定

本節主要是說明上述三種方法的基本設定，其餘的細節設定將在後續章節補充說明。在第二章我們提到，對於中繼端的做法為使用基於觀察點的壓縮方案，如圖 2.2 所示。而在這部分，我們使用一樣做法的有來源端的輸入位元特性與編碼器、中繼端的壓縮方式，以及三條通道的傳輸條件。對於來源端的輸入 U ，假設為均勻且獨立同分佈且長度 10^4 的訊息位元。來源端內部的通道編碼器 C_s 使用一碼率為 $1/2$ 的遞迴系統性迴旋碼，其生成矩陣 $G(D) = \left[1, \frac{1+D+D^2}{1+D^2} \right]$ ，將輸入 U 編成長度為 N 的碼字序列。

對於中繼端的壓縮方式，我們使用的是基於 Lloyd 演算法[20]的純量量化器，量化索引使用二位元映射(binary mapping)編碼方式。至於通道條件的部分，分別為 SR(來源端至中繼端)、SD(來源端至目的端)、RD(中繼端至目的端)。這三條通道互相獨立，且有各自的 SNR。參考前人研究[15]與[21]，我們設定 RD 通道的 $SNR_{RD} = 5dB$ ，SR 通道的 SNR_{SR} 為 SD 通道的 $SNR_{SD} + 2dB$ ，最後 SD 通道的 SNR 範圍為 $-2dB \sim 3dB$ 。

至於頻寬擴展的部分，我們在模擬上是將其訊號雜訊比的效應反應於中繼端的輸出位元上，使得 SR、SD 及 RD 三條通道擁有一致的 $\frac{E_s}{N_0}$ 。對於來源端 k 個輸入位元、中繼端 N_{out} 個輸出位元，給定頻寬擴展係數 γ ，其中繼端輸出的等效碼率 R_{relay} 為：

$$R_{relay} = \frac{k \cdot \gamma}{N_{out}} \quad (5.1)$$

5.2 實驗模擬

在這節我們將模擬比較上述三種方法的系統效能，由於在中繼端有量化器，來源端傳來的信號將會利用純量量化，量化為 m 位元的量化索引。模擬圖將呈現的是中繼端解碼所得的量化信號 Y_Q^{MMSE} 的重建訊雜比(RSNR)，並與中繼端原始接收信號 Y_{SR} 的重建訊雜比(Y_{SR} 與其量化重建值 Y_Q 計算而得)相比較，以及來源端輸入 U 的解碼位元錯誤率。每種方法都會與不使用合作式通訊(no cooperation, NC)及使用分散式天線系統(distributed antenna system, DAS)的情況一起比較效能。以下分三小節敘述，第一小節為基於單一同位檢查碼架構，第二小節為平行串接架構，第三小節為使用多層級同位子法編碼架構，以下分個別敘述。

5.2.1 基於單一同位檢查碼的中繼端解碼機制

[1] 參數設定：

如圖3. 1所示，由於中繼端利用單一同位檢查碼而增加一個冗餘位元，所以在此交錯器的大小為 $L = (m+1)N$ ， N 為源自來源端收到的信號長度。而通道碼為碼率1的遞迴

系統性迴旋碼 C_i ，其生成矩陣 $G(D) = \begin{bmatrix} 1+D \\ 1+D+D^2 \end{bmatrix}$ 。穿刺的部分則是偽隨機碼(pseudo

random code)將輸入序列穿刺為 mN 的長度。

[2] 實驗結果：

		SD 通道之 SNR(dB)					
		-2	-1	0	1	2	3
m=1	Y_{SR} RSNR(dB)	4.955997	5.161579	5.430702	5.769625	6.188015	6.689994
	Y_Q^{MMSE} RSNR(dB)	1.805517	2.21315	2.858359	3.635992	4.149959	4.665367
m=2	Y_{SR} RSNR(dB)	9.959769	10.15109	10.38048	10.6608	10.99741	11.40661
	Y_Q^{MMSE} RSNR(dB)	1.373613	1.61675	2.096087	3.281788	6.55659	12.46056

表 5.1 基於單一同位檢查碼的重建 SNR



		SD 通道之 SNR(dB)					
		-2	-1	0	1	2	3
BER	NC	0.17655	0.129444	0.080777	0.040755	0.015628	0.004434
	DAS	0.013573	0.003684	0.000713	9.66E-05	7.40E-06	2.00E-07
	m=1	0.098373	0.040917	0.010022	1.64E-03	2.96E-04	4.65E-05
	m=2	0.163146	0.102306	0.043448	8.59E-03	5.25E-04	1.06E-05

表 5.2 基於單一同位檢查碼的位元錯誤率

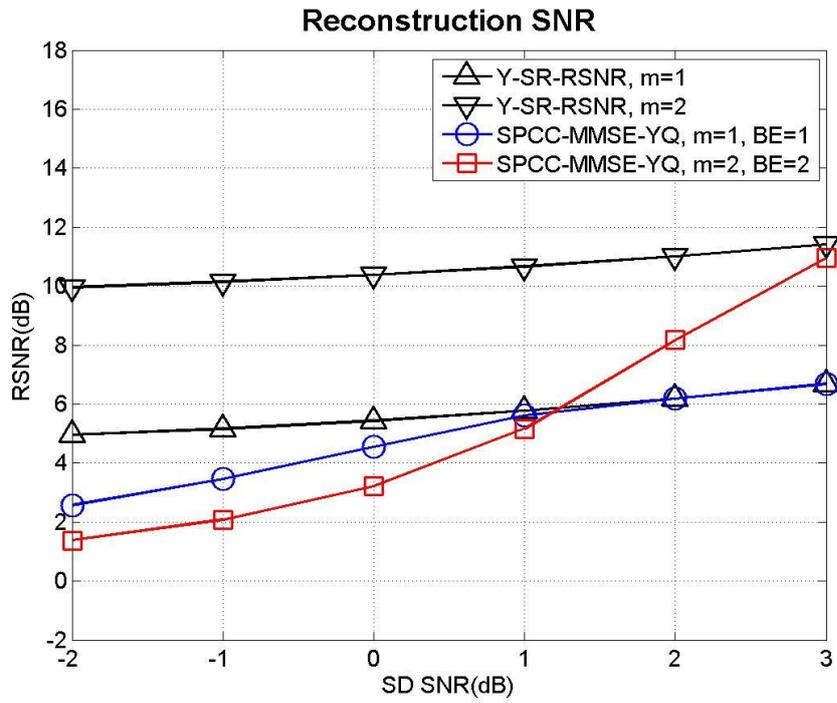


圖 5.1 基於單一同位檢查碼的重建 SNR

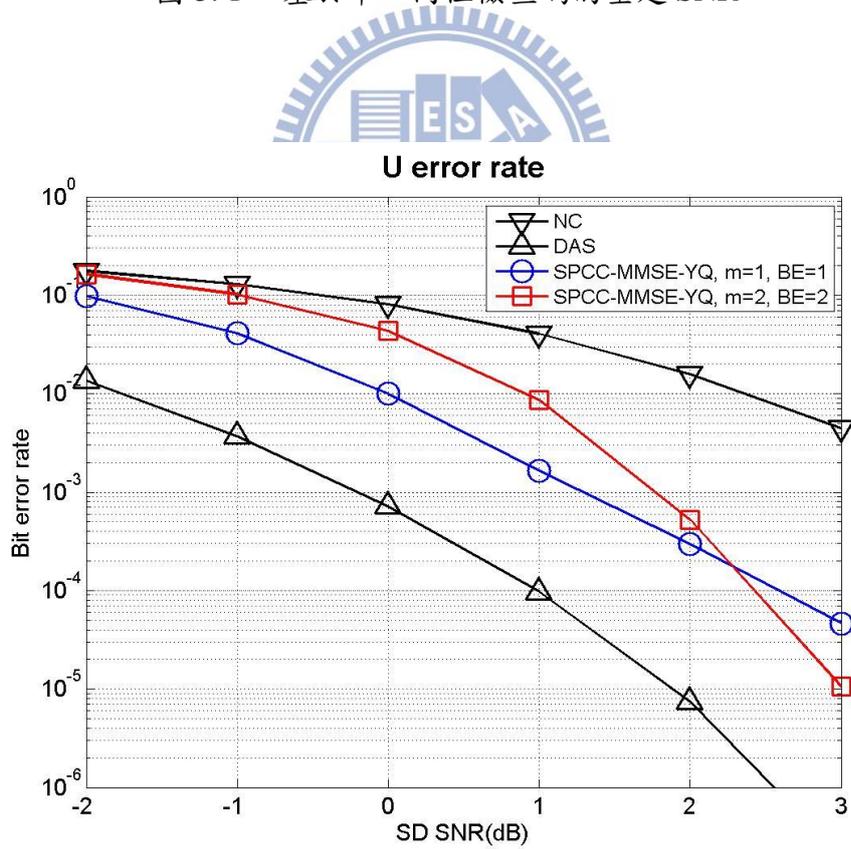


圖 5.2 基於單一同位檢查碼解碼機制的模擬結果

[3] 結果分析：

本實驗模擬第 3 章的編解碼架構，以 $m=1$ 與 $m=2$ 兩種不同的量化精度進行中繼端的信號處理。圖 5.1 所示為中繼端接收信號之重建訊雜比，圖 5.2 則為來源端信息的解碼位元錯誤率。參考圖 5.2，我們可以看到約在 2.3dB 之前， $m=2$ 的錯誤率比 $m=1$ 高。這種現象來自於我們在第 4 章所提及的，[15] 為確保疊代解碼機制得以運作，因此對通道訊雜比有許多的限制；特別是因其錯誤更正能力會受到 SD 通道的訊息所影響，因此來源端至目的端的通道訊雜比不能過低。而在目前的通道環境設定下，第三章之解碼架構並未完全進入工作區，也因此其效能會無法完全展現出來。參照圖 5.1，亦可發現此時中繼端接收信號之重建訊雜比有 $m=1$ 優於 $m=2$ 的現象，與最終的位元錯誤率表現大致相符。

另外，對於 $m=1$ 的量化精度，其中繼端至目的端的頻寬擴展係數 1，而 $m=1$ 時頻寬擴展係數為 2。我們可以看到在 $m=1$ 時，與 NC 的情況比較約有 1~2.5dB 的編碼增益，與 DAS 差約 1.5~1.8dB。 $m=2$ 時，與 NC 約有 0~1.7dB 的編碼增益，跟 DAS 則差約 1~2.8dB，顯示尚有改進空間。

5.2.2 基於平行串接架構的中繼端解碼機制

[1] 參數設定：

如圖 4.1 所示，由於量化結果直接進行通道編碼，其交錯器長度為 $L = mN$ ， N 為源自來源端收到的信號長度。通道碼為碼率 1/2 的遞迴系統性迴旋碼 C_i ，其生成矩陣 $G(D) = \left[1, \frac{1+D}{1+D+D^2} \right]$ ，與通道碼串接之穿刺部分則是偽隨機碼將輸入序列穿刺為 $3mN/2$ 的長度，兩者總合為一碼率 2/3 的通道碼。

[2] 實驗結果：

		SD 通道之 SNR(dB)					
		-2	-1	0	1	2	3
m=1	Y_{SR} RSNR(dB)	4.955997	5.161579	5.430702	5.769625	6.188015	6.689994
	Y_Q^{MMSE} RSNR(dB)	2.675452	2.842578	3.076296	3.394777	3.825495	4.39265
m=2	Y_{SR} RSNR(dB)	9.959769	10.15109	10.38048	10.6608	10.99741	11.40661
	Y_Q^{MMSE} RSNR(dB)	8.238618	8.706696	9.311459	10.10544	11.10771	12.43758

表 5.3 基於平行串接架構的重建 SNR



		SD 通道之 SNR(dB)					
		-2	-1	0	1	2	3
BER	NC	0.17655	0.129444	0.080777	0.040755	0.015628	0.004434
	DAS	0.013573	0.003684	0.000713	9.66E-05	7.40E-06	2.00E-07
	m=1	0.052596	0.023784	0.008125	2.17E-03	4.22E-04	6.51E-05
	m=2	0.023348	0.007916	0.002025	4.11E-04	6.80E-05	1.11E-05

表 5.4 基於平行串接架構的位元錯誤率

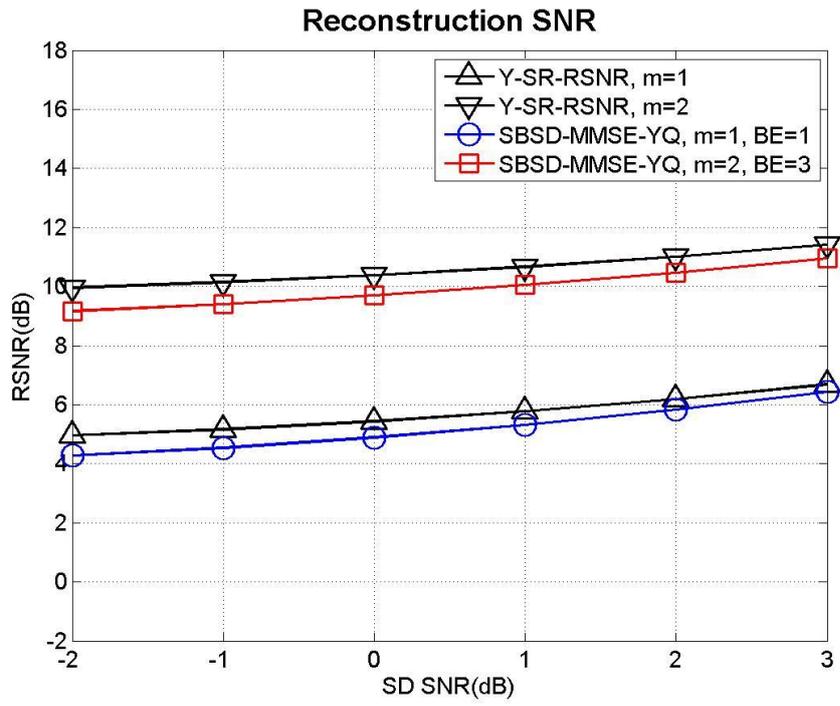


圖 5.3 基於平行串接架構的重建 SNR

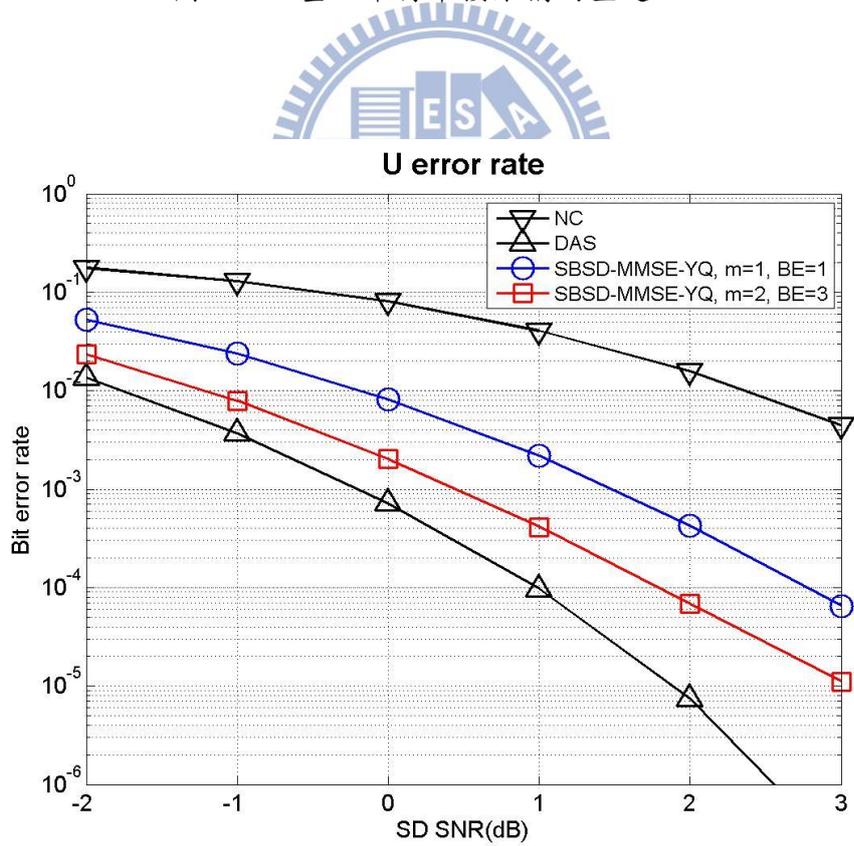


圖 5.4 基於平行串接架構解碼機制的模擬結果

[3] 結果分析：

本實驗模擬 4.1 節的平行串接解碼架構，以 $m=1$ 與 $m=2$ 兩種不同的量化精度進行中繼端的信號處理。圖 5.3 所示為中繼端接收信號之重建訊雜比，圖 5.4 為來源端信息的解碼位元錯誤率。隨著通道環境的改善，重建訊雜比也隨之提高，而位元錯誤率也跟著變低。不過，在重建訊雜比方面，離 Y_{SR} 的重建訊雜比還有些微差距。

另外，對於 $m=1$ 的量化精度，其中繼端至目的端擁有頻寬擴展係數 1，而 $m=2$ 時頻寬擴展係數則為 3。這是由於使用碼率 2/3 的迴旋碼且希望維持通道碼的解碼效能所設定。我們可以看到在 $m=1$ 時，與 NC 的情況約有 2.5dB 的編碼增益，與 DAS 差約 1.5~1.8dB。 $m=2$ 時，與 NC 可以達到約 3.5dB 的編碼增益，跟 DAS 則差約 0.5~1dB，在較低的通道訊雜比時能接近分散式天線系統的效能。此部分結果顯示我們已可克服前一個實驗中通道環境所帶來的影響，並在低的 SD 通道訊雜比下仍能保持一定的解碼能力。

5.2.3 基於多層級同位子法編碼的解碼機制

[1] 參數設定：

如圖 4.5 所示，量化後的序列會送至使用同位子法的編碼器 C_0 進行編碼，編碼器 C_0 。

使用一碼率 1/2 的遞迴系統性迴旋碼，其生成矩陣 $G(D) = \left[1, \frac{1+D+D^2+D^3}{1+D+D^3} \right]$ ，但只輸出

其同位位元，然後依據 4.6.1 節敘述的分層編碼處理後傳入通道編碼器中。在此的交錯器長度為 $L = mN$ ， N 為源自來源端收到的信號長度。通道碼為碼率 1/2 的遞迴系統性迴旋

碼 C_i ，其生成矩陣 $G(D) = \left[1, \frac{1+D}{1+D+D^2} \right]$ ，與通道碼串接之穿刺部分則是偽隨機碼將輸

入序列穿刺為 $3mN/2$ 的長度，總合為一碼率 2/3 的通道碼。

[2] 實驗結果：

		SD 通道之 SNR(dB)					
		-2	-1	0	1	2	3
m=1	Y_{SR} RSNR(dB)	4.955997	5.161579	5.430702	5.769625	6.188015	6.689994
	Y_Q^{MMSE} RSNR(dB)	3.116429	3.281909	3.493204	3.775151	4.15653	4.665312
m=2	Y_{SR} RSNR(dB)	9.959769	10.15109	10.38048	10.6608	10.99741	11.40661
	Y_Q^{MMSE} RSNR(dB)	9.901095	10.34612	10.90293	11.61152	12.4373	13.18082

表 5.5 基於多層級同位子法的重建 SNR



		SD 通道之 SNR(dB)					
		-2	-1	0	1	2	3
BER	NC	0.17655	0.129444	0.080777	0.040755	0.015628	0.004434
	DAS	0.013573	0.003684	0.000713	9.66E-05	7.40E-06	2.00E-07
	m=1	0.04278	0.01795	0.005817	1.44E-03	2.83E-04	3.78E-05
	m=2	0.020571	0.006706	0.001657	2.88E-04	4.75E-05	5.00E-06

表 5.6 基於多層級同位子法的位元錯誤率

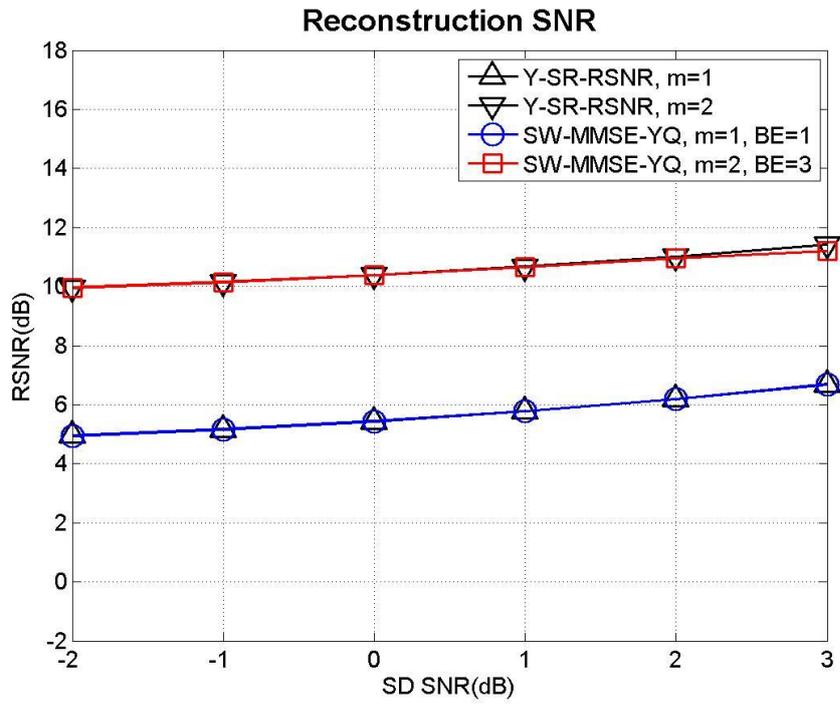


圖 5.5 基於多層級同位子法的重建 SNR

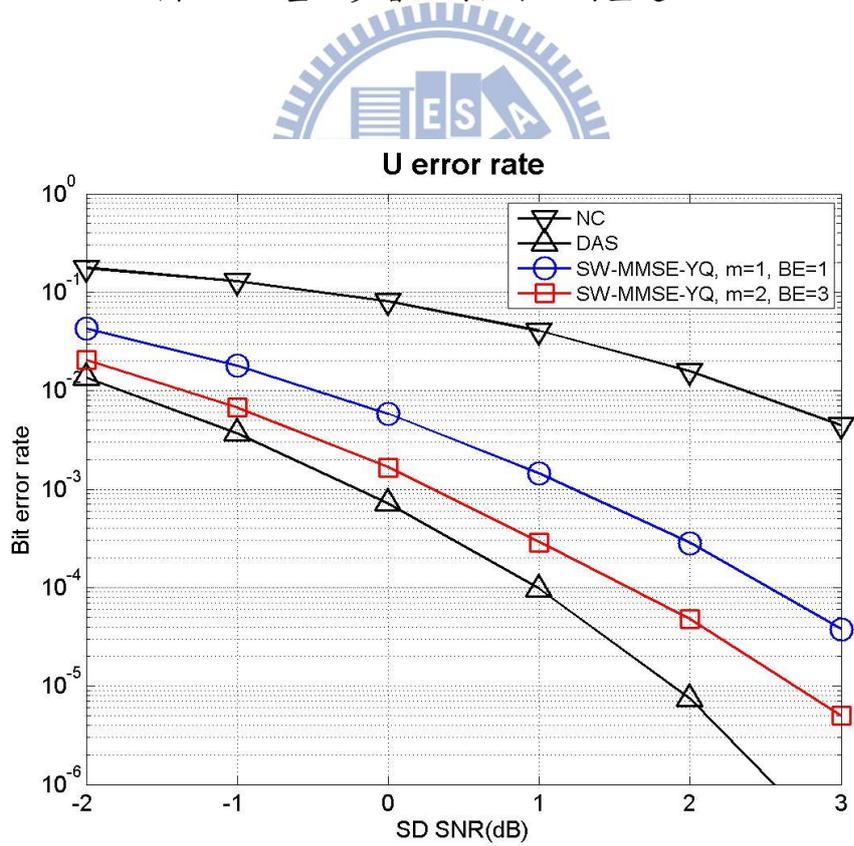


圖 5.6 基於多層級同位子法解碼機制的模擬結果

[3] 結果分析：

本實驗模擬 4.6 節的同位子法的多層級解碼架構，也以 $m=1$ 與 $m=2$ 兩種不同的量化精度進行中繼端的信號處理。圖 5.5 為中繼端接收信號之重建訊雜比，圖 5.6 為來源端信息的解碼位元錯誤率。我們可以看到圖 5.5 中的重建訊雜比，不論是 $m=1$ 或 $m=2$ 兩種量化精度，其重建訊雜比皆與 Y_{SR} 的重建訊雜比相貼合，顯示中繼端若使用同位子法的多層級解碼機制，確實可以進一步提升來源端的解碼效能。

另外，圖 5.6 中的 $m=1$ 與 $m=2$ 兩種量化精度所對應到的頻寬擴展係數也與 5.2.3 節相同，分別為 1 及 3。可以看到在 $m=1$ 時，與 NC 的情況約有 2.7dB 的編碼增益，與 DAS 差約 1.2~1.6dB。 $m=2$ 時，與 NC 可以達到約 4dB 的編碼增益，與 DAS 差約 0.4~0.85dB。另外，對於中繼端接收信號 Y_{SR} 的重建訊雜比，如圖 5.5，我們可以看到 $m=1$ 與 $m=2$ 兩種重建訊雜比幾乎與 Y_{SR} 的重建訊雜比相吻合。結果顯示中繼端若使用多層級同位子法的解碼機制，確實可以進一步提升來源端的解碼效能，在低的 SD 通道訊雜比下仍能保持一定的解碼能力。

5.3 綜合討論

在這一節，我們將分成 $m=1$ 與 $m=2$ 兩種量化精度來對上述三個架構進行討論。環境設定也與 5.1 節相同， $m=1$ 的頻寬擴展為 1， $m=2$ 的頻寬擴展為 3。

對於 $m=1$ 的情況下，由圖 5.7 可以看到雖然三種做法的位元錯誤率都很接近，但還是使用多層級同位子法的編碼架構最低。此外，由圖 5.5 所顯現的重建訊號之訊雜比來看，第三種架構的訊雜比是整個與原始重建信號重疊，我們可以推論在此狀況下可以得到的最好的錯誤率。

對於 $m=2$ ，即使第一種做法的頻寬擴展係數提高到 3，在低的 SD 通道訊雜比下的效能仍然較差，而另外兩種做法與分散式天線系統的效能比較起來則有不錯的效果。



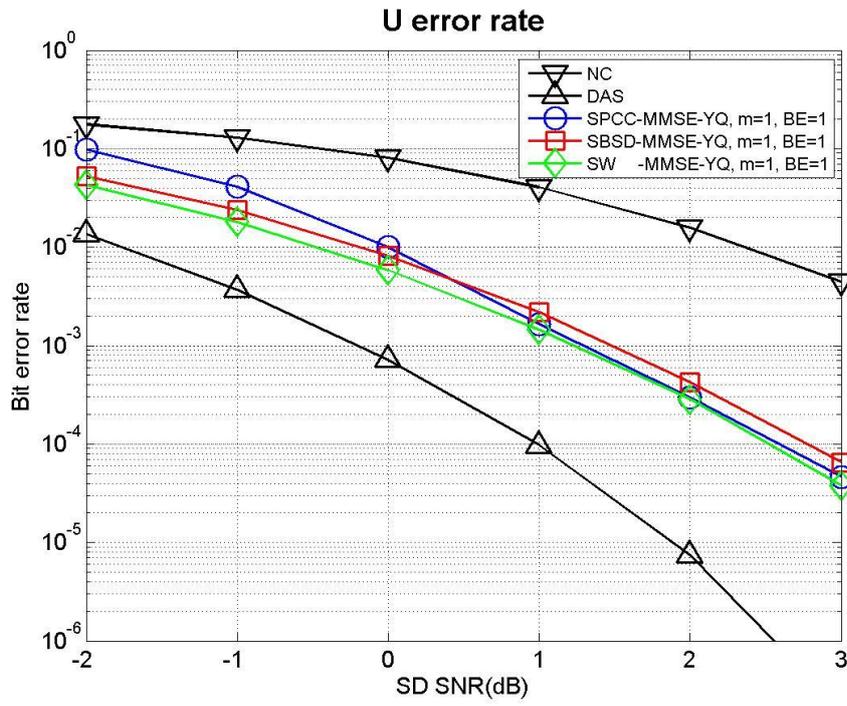


圖 5.7 量化精度 $m=1$ 的位元錯誤率

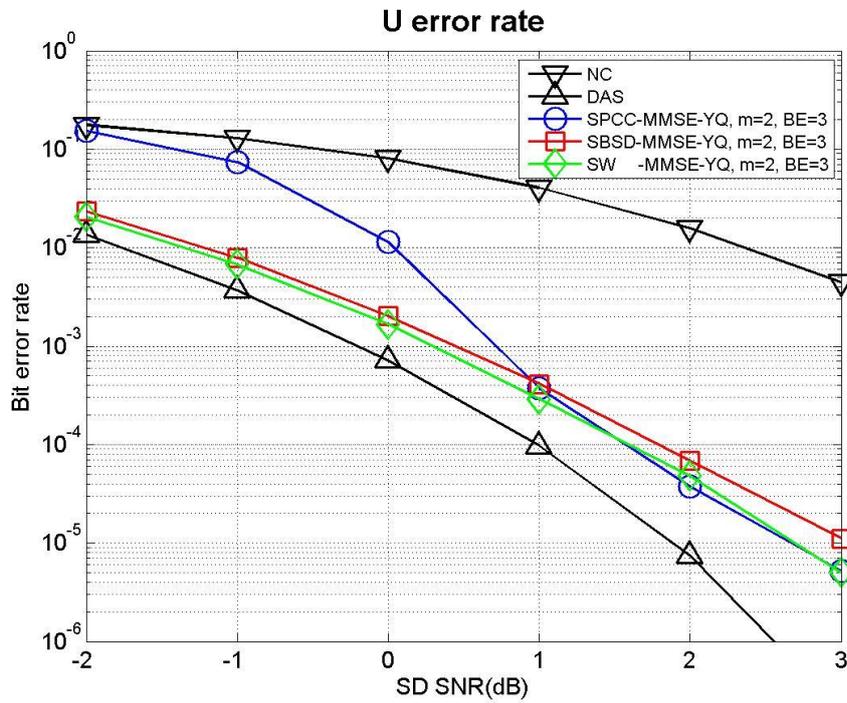


圖 5.8 量化精度 $m=2$ 的位元錯誤率

第6章 結論與未來展望

在現今科技生活中，無線通訊網路的涵蓋範圍越來越廣，隨之衍生出多樣化的媒體服務，諸如語音、短訊、音樂、影片等等，在容納那麼多服務的同時也要兼顧連線品質。加上現在各種建築物林立，而在進行無線通訊的時候，最為重要的就是克服多重路徑衰減。以往有一些方法被提出來處理，例如多天線系統。但是，若在一些基於規格的限制地方就無法使用，合作式通訊將是一個可行的方案。

合作式通訊是透過中繼端，協助來源端至目的端之間的通訊，而且研究[4]也證明了在 AWGN 通道中，中繼端的通道容量大於來源端至目的端的通道容量。並透過三種不同的編碼方式可以獲得中繼通道容量的下限值，分別為放大傳遞、解碼傳遞與壓縮傳遞，通常都以放大傳遞為中繼機制。不過在某些情況，例如低信號訊雜比下，放大傳遞就不是一個可行的方案，這是因為在放大訊號的同時會與雜訊一起放大，造成效能無法提升，而解碼傳遞則需要先解出原始訊號，導致解碼延遲。壓縮傳遞則是直接對接收信號量化後再編碼傳送至目的地端，因此不會有解碼延遲的問題。

本論文主要探討壓縮傳遞的編碼機制，基於單一同位檢查碼與碼率 1 的迴旋碼的設計方案，我們在量化器後使用了分散式訊源編碼的同位子法，以及多層級編碼的概念做為一個新的中繼端壓縮傳遞機制。對於同位子法我們採用一迴旋碼構成，而目的端在進行多層級解碼時，會有不同層級傳遞的訊息幫助解碼。在低訊雜比的情形下，可以接近分散式天線系統的效能。由於中繼端使用的是純量量化器，而純量量化器在同一個量化精度所造成的重建信號雜訊比有其限制。另外，同位子法的部分並非只能利用迴旋碼來構成，且我們的通道僅考慮為無記憶性的 AWGN 通道，並不是真實的通道環境，因此未來或許可以針對不同的量化方法、不同的同位子法與通道環境，例如衰落(fading)通道等加以改進探討。

參考文獻

- [1] 吳文榕，曾凡碩，謝弘道，「合作式通訊之現況與發展」，網路通訊國家型科技計劃簡訊，民國一百年。
- [2] 洪樂文，王淑賢，「合作式通訊系統之媒介存取與路由技術」，網路通訊國家型科技計劃簡訊，民國一百年。
- [3] 林高洲，「無線通信系統發展新趨勢-智慧型合作式通信網路」，中華民國電子零件認證委員會，民國一百年。
- [4] T. Cover and A. Gamal, "Capacity theorems for the relay channel," *Information Theory, IEEE Transactions on*, vol. 25, no. 5, pp. 572–584, Sep 1979.
- [5] Z. Liu, V. Stankovic, and Z. Xiong, "Wyner-Ziv coding for the half-duplex relay channel," in *Acoustics, Speech, and Signal Processing, 2005. Proceedings. (ICASSP 2005). IEEE International Conference on*, vol. 5, March 2005, pp. 1113–1116 Vol. 5.
- [6] R. Hu and J. Li, "Practical compress-forward in user cooperation: Wyner-Ziv cooperation," in *Information Theory, 2006 IEEE International Symposium on*, July 2006, pp. 489–493.
- [7] T. Bui and J. Yuan, "A decode and forward cooperation scheme with soft relaying in wireless communication," in *Signal Processing Advances in Wireless Communications, 2007. SPAWC 2007. IEEE 8th Workshop on*, June 2007, pp. 1–5.
- [8] T. Fingscheidt and P. Vary, "Softbit Speech Decoding: A New Approach to Error Concealment," *IEEE Trans. Speech Audio Processing.*, vol. 9, no.3, pp.240-251, 2001.
- [9] J. Hagenauer, "Source-Controlled Channel Decoding," *IEEE Trans. Commun.*, vol. 43, pp. 2449-2457, Sep. 1995.
- [10] C. Berrou, A. Glavieux, and P. Thitimajshima, "Near Shannon limit error-correcting coding and decoding: Turbo-codes. 1," in *Proc. IEEE International Conference on Communications (ICC)*, pp. 1064-1070, 1993.
- [11] C. Berrou and A. Glavieux, "Near Optimum Error Correcting Coding And Decoding: Turbo-Codes," *IEEE Trans. on Commun.*, vol. 44, no. 10, pp. 1261-1271, 1996.
- [12] D. Slepian and J. K. Wolf, "Noiseless Coding of Correlated Information Sources," *IEEE Trans. Inform. Theory*, vol. 19, pp. 471-480, Jul. 1973.
- [13] Blasco-Serrano, Ricardo. "Coding Strategies for Compress-and-Forward Relaying," Diss. KTH, 2010.
- [14] E. Yilmaz, R. Knopp, and D. Gesbert, "On the gains of fixed relays in cellular networks with intercell interference," in *Signal Processing Advances in Wireless Communications, 2009. SPAWC 2009. IEEE 10th Workshop on*, June 2009, pp. 603–607.
- [15] Blasco-Serrano, Ricardo, Ragnar Thobaben, and Mikael Skoglund. "Compress-and-forward relaying based on symbol-wise joint source-channel coding." *Communications (ICC), 2010 IEEE International Conference on*. IEEE, 2010.
- [16] T. Fingscheidt and P. Vary, "Softbit Speech Decoding: A New Approach to Error Concealment," *IEEE Trans. Speech Audio Processing*, vol. 9, no. 3, pp. 240-251, 2001.
- [17] 潘彥璋，「基於索引層級的疊代訊源通道解碼機制」，國立交通大學碩士論文，民國九十七年。
- [18] 王韋超，「基於低密度奇偶查核碼的分散式訊源編碼機制」，國立交通大學碩士論文，民國一百年。
- [19] Tan, Peiyu, Kai Xie, and Jing Li, "Slepian-Wolf coding using parity approach and syndrome approach," *Information Sciences and Systems, 2007. CISS'07. 41st Annual Conference on*. IEEE, 2007.

- [20] S. Lloyd, "Least squares quantization in PCM," *Information Theory, IEEE Transactions on*, vol. 28, no. 2, pp. 129–137, Mar 1982.
- [21] Yinan Qi, Reza Hoshyar, and Rahim Tafazolli, "Soft Multilevel Slepian-Wolf Decoding in Systems Using Turbo Joint Decoding and Decompressing," in *Proc. Vehicular Technology Conference, 2008(VTC, Spring 2008)*, pp.1514-1518, May 2008
- [22] Singh, Jaspreet, Onkar Dabeer, and Upamanyu Madhow, "Capacity of the discrete-time AWGN channel under output quantization." *Information Theory, 2008. ISIT 2008. IEEE International Symposium on. IEEE, 2008.*

