

國立交通大學

電信工程學系

碩士論文



強健性寬頻語音編碼系統之研究
A study of robust wideband speech coding

研究生：蔡知鑑

指導教授：張文輝博士

中華民國九十六年六月

強健性寬頻語音編碼系統之研究

A study of robust wideband speech coding

研究生：蔡知鑑

Student：Chih-Chien Tsai

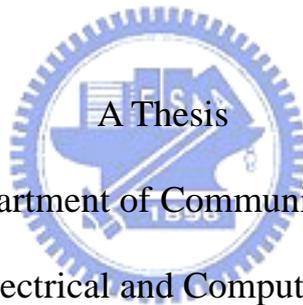
指導教授：張文輝

Advisor：Wen-Whei Chang

國立交通大學

電信工程學系

碩士論文



Submitted to Department of Communication Engineering

College of Electrical and Computer Engineering

National Chiao Tung University

in Partial Fulfillment of Requirements

for the Degree of

Master of Science

in Communication Engineering

June 2007

Hsainchu, Taiwan, Republic of China

中華民國 九十六 年 六 月

強健性寬頻語音編碼系統之研究

學生：蔡知鑑

指導教授：張文輝 博士

國立交通大學電信工程學系碩士班



中文摘要

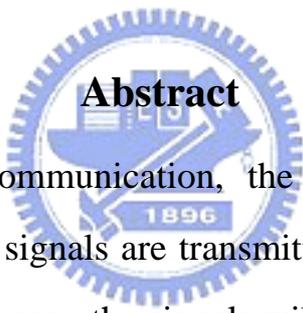
在數位傳輸系統，訊源控制通道解碼的研究旨在補償通道錯誤所造成的解碼失真。訊源編碼輸出的量化索引序列存在殘餘冗息，若能在通道解碼過程妥善地利用這一項事前資訊，將可對接收端的解碼效能有所提昇。我們提出一個訊源控制通道解碼機制，利用寬頻語音編碼處理後所殘留的訊源冗息提昇其參數重建的強健性。有別於以往基於單一敘述的演算法，我們將引入多重敘述向量量化，並改良其解碼演算法以估測最大後驗機率。實驗模擬顯示在重建寬頻語音編碼參數上，訊源冗息與多重敘述傳輸機制的整合能有效提昇其解碼效能。

A study of robust wideband speech coding

Student:Chih-Chien Tsai Advisor:Dr. Wen-Whei Chang

Department of Communication Engineering

National Chiao Tung University

The logo of National Chiao Tung University is a circular emblem with a gear-like outer border. Inside the circle, there is a stylized building and a banner with the year '1896'. The word 'Abstract' is overlaid on the logo in a bold, black font.

Abstract

For digital speech communication, the system performance will degrade after compressed signals are transmitted over noisy channels. In order to combat channel errors, the signals will be protected by means of source-controlled channel decoding methods. When signals are quantized by the source encoder, quantizer indexes have residual redundancy due to non-uniform distribution and correlations between them. By taking advantage of such a priori knowledge, we propose a source-controlled channel decoding scheme which incorporates the source residual redundancy into the channel decoding process. Also proposed is a modified decoding algorithm based on the multiple description vector quantizer. Experiments on the wideband speech coding showed that the proposed multiple description transmission scheme outperforms single description scheme in parameter reconstruction.

致謝

本篇論文的完成，首先要由衷地感謝我的指導教授張文輝老師，由於老師的細心教導與耐心指正，讓我瞭解到做研究時所應該持有的態度與精神，讓我受益良多。另外也要感謝實驗室的學長，同學以及學弟，不管是在課業或者是在研究上都協助我解決許多問題。同時也要感謝陪伴我的朋友們，尤其在我準備研究所考試以及就讀研究所其間一路上給我鼓勵。最後，僅將此論文獻給我的父母以及家人。

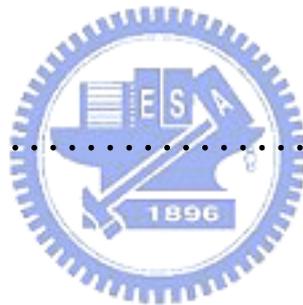


目錄

中文摘要.....	i
英文摘要.....	ii
致謝.....	iii
目錄.....	iv
圖目錄.....	vii
表目錄.....	viii
第一章 緒論	1
1.1 研究動機與方向.....	1
1.2 章節概要.....	3
第二章 寬頻語音編碼系統	4
2.1 AMR-WB 系統概述.....	5
2.2 編碼輸出冗息.....	9
2.3 ISF 索引間相互訊息.....	11
2.4 分析討論.....	13
第三章 訊源控制通道解碼機制	16

3.1	傳輸系統.....	17
3.2	訊息冗息的應用.....	20
3.3	錯誤控制碼與訊源冗息之整合.....	23
3.3.1	迴旋編碼架構.....	23
3.3.2	最後後驗機率的推導.....	25
第四章 多重敘述的訊源控制通道解碼機制		28
4.1	多重敘述向量量化.....	28
4.1.1	系統架構.....	28
4.1.2	最佳 MDVQ 解碼器.....	30
4.2	多重敘述索引指定設計.....	33
4.2.1	指定矩陣的評分機制.....	34
4.2.2	多重描述二元置換演算法.....	35
4.3	多重敘述量化與通道解碼之整合.....	36
第五章 實驗模擬與結果分析		40
5.1	冗息應用於訊源解碼之實驗.....	40
5.1.1	系統模擬之步驟說明.....	40
5.1.2	結果分析.....	43

5.2 訊源控制通道解碼之實驗.....	45
5.2.1 系統模擬之步驟說明.....	45
5.2.2 結果分析.....	46
5.3 整合多重敘述與訊源控制通道解碼之實驗.....	47
5.3.1 系統模擬之步驟說明.....	47
5.3.2 結果分析.....	48
第六章 結論與未來展望.....	50
參考文獻.....	52



圖目錄

圖 2.1	AMR-WB ACELP 編碼器的方塊圖	7
圖 2.2	AMR-WB ACELP 解碼器的方塊圖	8
圖 2.3	ISF 係數的分次多階向量量化處理	9
圖 2.4	框架內關聯性與框架間關聯性	12
圖 3.1	訊源控制通道解碼架構	17
圖 3.2	訊號傳輸系統方塊圖	19
圖 3.3	迴旋編碼器架構	25
圖 4.1	多重敘述傳輸系統架構	30
圖 4.2	指定矩陣的配置範例	30
圖 4.3	MD-SMAP1 的傳輸架構	39
圖 5.1	各種訊源解碼機制的平均頻譜失真結果	43
圖 5.2	訊源控制通道解碼器平均頻譜失真結果	46
圖 5.3	多重敘述通道解碼的平均頻譜失真結果	48

表目錄

表 2.1	AMR-WB 不同編碼模式的位元配置.....	8
表 2.2	ISF 索引值框架間的相互訊息.....	14
表 2.3	ISF 索引值框架內的相互訊息.....	15
表 5.1	各種訊源解碼機制的參數重建結果(1dB AWGN)	41
表 5.2	各種訊源解碼機制的參數重建結果(4dB AWGN)	42
表 5.3	各種訊源解碼機制的參數重建結果(7dB AWGN)	42
表 5.4	各索引所配置的指定矩陣	47
表 5.5	不同解碼機制的位元傳輸數目	49



第一章 緒論

1.1 研究動機與方向

在典型的數位通訊系統中，由於通道的多變性以及環境干擾的影響，在接收端往往會因為通道錯誤或衰退現象而造成原訊號的嚴重失真。尤其是語音 (speech)、音訊 (audio) 或是影像 (video) 的多媒體訊號，受限於通道環境而影響其輸出品質。有鑑於此，多媒體通訊一直是一項很重要的研究主題，其關鍵在於設計訊源編碼 (source coding) 以降低傳輸位元率，運用通道編碼 (channel coding) 為了提升對抗雜訊的強健性能。

在傳統的通訊系統設計中，訊源編碼和通道編碼這兩個系統區塊往往是分開考量且獨立設計，此設計概念源自於沈農 (shannon) 的消息理論[1]。相關理論是在個別的編碼架構規劃時，假設另一個編碼器已完成最佳化設計，但是這並不符合真實的通訊環境。經由消息理論的計算可以發現，訊號源在經由訊源編碼器的處理之後，無法完全去除訊號之間的關聯性。其編碼輸出的位元序列之間存在著非均勻機率分佈和某種形式的記憶性，而這些殘留的資訊通稱為殘餘冗息 (residual redundancy)。至於通道編碼的主要功能，是對訊源編碼

器輸出的位元序列附加一些保護位元，最常見的是區段碼 (block code) 以及迴旋碼 (convolutional code)。在通過雜訊通道環境時，這些保護位元可以在解碼過程中修正通道錯誤以達到強健的效果。

訊源殘餘冗息的優點，是可以在不需要提升傳輸位元率的情形之下，提供通道編碼一些事前資訊 (a priori information)。因此，若我們能在接收端妥善地利用這些事前資訊，相信應能對系統的解碼效能有所提升。在接收端將訊源冗息和通道環境之效應一併納入系統設計考量的研究統稱為合併訊源通道解碼 (joint source-channel decoding, JSCD) [2][3][4]，而在通道解碼過程充分地利用訊源殘餘冗息的相關研究又稱為訊源控制通道解碼 (source-controlled channel decoding, SCCD) [5]。

在本論文中所要提出的是通道位元錯誤的隱匿機制 (error concealment)，以訊源控制通道解碼演算法為主軸，並將整合多重敘述向量量化 (Multiple Description Vector Quantizer, MDVQ) 的概念。此外，由於寬頻語音編碼已視為下一代語音編碼標準架構，我們將所提出的演算法應用在重建寬頻語音編碼訊號上。

1.2 章節概要

第二章將先介紹寬頻語音編碼系統，再針對經由訊源編碼器之後的量化輸出索引做殘餘冗息分析。第三章我們探討利用訊源控制通道解碼機制來增加解碼的可靠性。第四章則是整合訊源控制通道編碼與多重敘述向量量化。第五章則為實驗模擬，驗證本論文所提出的解碼機制，並作效能評估與比較。第六章則為結論與未來展望。



第二章 寬頻語音編碼系統

在現存針對窄頻帶〔200-3400Hz〕電話語音所開發的的語音編碼技術，已經將窄頻帶語音的品質與效能發揮到極致。為了提供更好的通話品質並增加語音的自然真實性，必須增加聲音的頻寬及其取樣頻率才能達成。然而，在增加取樣頻率同時勢必會增加資料傳輸量，因此要使用更有效率的語音編碼演算法，才能將其壓縮到可以接受的資料量。隨著無線通訊普及率已提高，下一代行動通訊系統的主要目標是提供更佳的通話服務品質。而在有線網路傳輸上，因為網路頻寬提昇且成本大幅降低，其應用層面逐漸擴大，未來將可取代部份電信產品，並提供全新整合式數位語音之服務，同時亦將改變現有的電信服務之模式。因此不管是 ITU 或 ETSI/3GPP，均朝向高品質的寬頻語音通訊服務前進，無獨有偶的均選定適應性多重速率寬頻語音編碼技術(Adaptive Multi-Rate Wideband, AMR-WB)做為下一代語音編碼的標準。

本論文想要提出的是一軟性輸出通道解碼機制，而解碼效能所依靠的資訊將為訊源編碼輸出索引之間所存在的相互訊息(Mutual Information)。可想而知的是，若能針對索引之間所存在的相互訊息

進行深入分析，在據以開發的解碼機制必定有效改善其對抗通道雜訊的強健效能必定會是很大的助益。所以接下來先要概述 AMR-WB 系統，進而利用相互訊息的概念針對 AMR-WB 的頻譜參數進行分析。

2.1 AMR-WB 系統概述

行動電信網路的未來發展趨勢，將是結合無線通訊與網際網路，突破時空限制以提供更人性化的隨身資訊服務。目前的行動電話服務以第 2 代的 GSM 與第 2.5 代的 GPRS 為主，其核心技術為碼激發線性預估(Code-Excited Linear Prediction, CELP)編碼演算法[6]。主要是藉由合成分析處理取得模擬人體發聲過程的線性預估係數和激發源參數，再作壓縮處理以因應無線通訊的低位元率傳輸需求。問題是現有電信網路均受限於其窄頻帶(200-3400 Hz)，無法提供自然逼真的語音品質，使用寬頻語音(50-7000 Hz)作為傳輸媒介則可大幅提昇通話品質。相較於窄頻語音，寬頻語音向下延伸的 50-200Hz 頻段可以增加聲音的臨場感和自然真實性，向上延伸的 3400-7000Hz 頻段則會有效改善語意辨識度和語音清晰度。因此在 2001 年 3 月，歐規第 3 代標準組織 ETSI/3GPP 和 ITU-T 達成共識，選定適應性多重速率寬頻技術(Adaptive Multi-Rate WideBand, AMR-WB)為新的語音編碼標準[7]。AMR-WB 具有一項訊源控制速率功能，可支援 6.6 到 23.85

kbits/sec 九種編碼速率，能適時因應通道環境的改變而動態調整其編碼模式與錯誤更正位元配置。

AMR-WB 編碼器與解碼器的系統方塊圖分別示於圖 2.1 及圖 2.2，而其不同編碼模式的位元配置則列在表 2.1。編碼器的運作是根據代數碼簿激發線性預測演算法 (Algebraic Code-Excited Linear Prediction, ACELP)，以在 20 毫秒的音框，採用 12.8 kHz 取樣頻率來分析。在每一個語音訊框內，分析語音得到線性預測係數 (Linear Prediction Coefficients, LPC)、適應性編碼簿索引值 (Adaptive Codebook index, ACB)、固定性編碼簿索引值 (Fixed Codebook index, FCB) 與增益 (gains) 等 ACELP 模型參數。除了這些參數外，在 23.85 kbps 的模式下需要額外計算語音的高頻帶增益索引值 (High-band gain index)。這些參數經過編碼後再傳送，以期有效降低系統的整體傳輸位元率。

本論文將研究重點置於線性預測係數的重建上，因為線性預測係數主要呈現每個音框的頻譜資訊，其重建的成敗對語音輸出品質扮演重要的角色。在這 AMR-WB 編碼過程中，先執行線性預測預測取得的 16 個線性預測參數，再據以轉換為成量化失真靈敏度較低的導納頻譜頻率 (Impedance Spectrum Frequency, ISF) 係數。因應無線電信網路傳輸的低位元率需求，ISF 係數的編碼壓縮處理有其必要性，主

要是將 16 個 ISF 係數進行分次多階向量量化(split-multistage vector quantization, S-MSVQ)，其量化過程的位元分配如圖 2.3 所示。所謂的分次多階向量量化是分兩階段處理：第一階段將 16 個 ISF 係數 X 分成 9 和 7 個元素的兩組向量，個別執行向量量化求得兩個索引值(Index)，這兩個索引值所對應的 ISF 係數設定為 X_1 ；第二階段則將 X 減去 X_1 所得到誤差值 X_2 ，再依序以四組三個係數與一組四個係數分成五組向量，最後個別求得五個向量量化索引值。

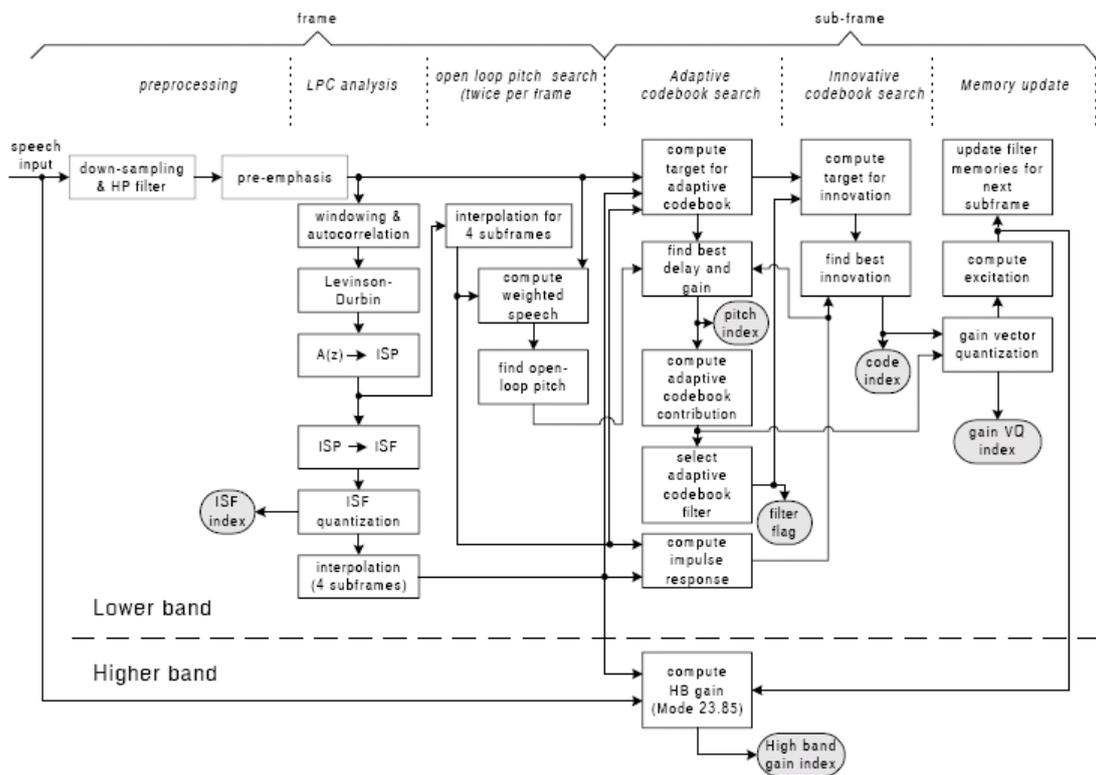


圖 2.1 AMR-WB ACELP 編碼器的方塊圖

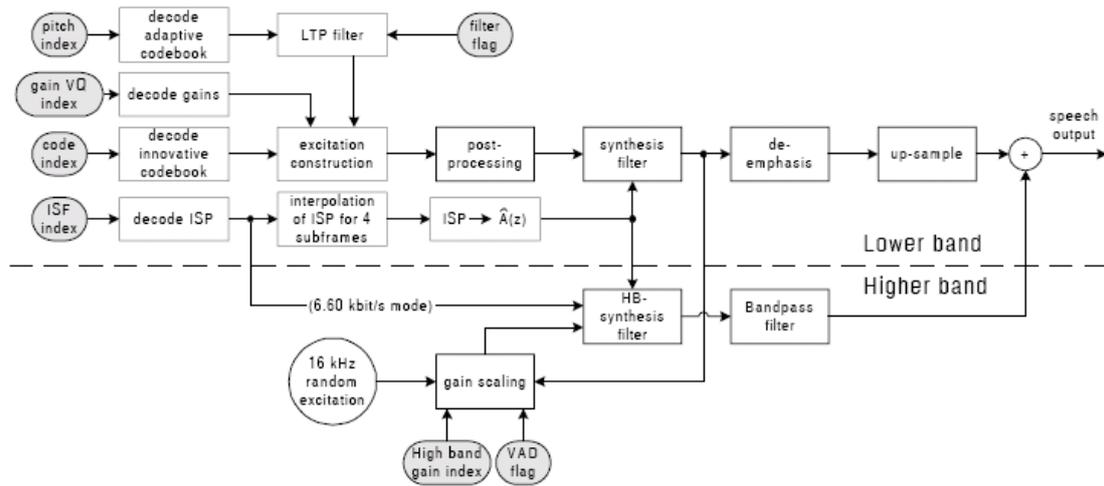


圖 2.2 AMR-WB ACELP 解碼器的方塊圖

parameter	Codec Mode(kbps)								
	6.60	8.85	12.65	14.25	15.85	18.25	19.85	23.05	23.85
VAD	1	1	1	1	1	1	1	1	1
LTP flag	0	0	4	4	4	4	4	4	4
ISF	36	46	46	46	46	46	46	46	46
ACB	23	26	30	30	30	30	30	30	30
FCB	48	80	144	176	208	256	288	352	352
GAIN	24	24	28	28	28	28	28	28	28
High-band Gain	0	0	0	0	0	0	0	0	16
Total	132	177	253	285	317	365	397	461	477

表 2.1 AMR-WB 不同編碼模式的位元配置

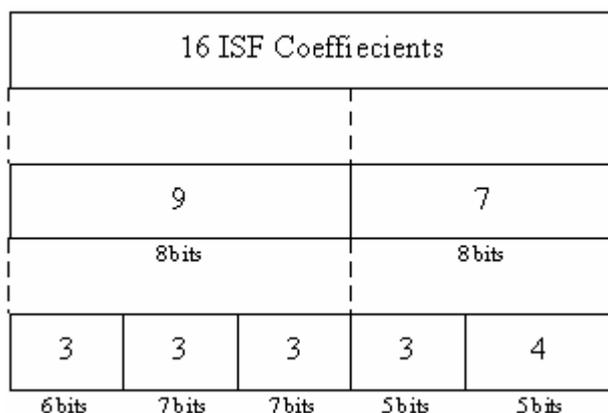


圖 2.3 ISF 係數的分次多階向量量化處理

2.2 編碼輸出冗息

當訊號源通過編碼器（或量化器）時，對於通道錯誤的敏感度將增大，所以需要使用某些技術以更正傳輸所衍生的位元錯誤。標準的通道編碼技術藉由加入了一些保護位元於編碼位元序列中，以期當位元因通道環境的影響而產生錯誤下仍可被復原。訊源冗息則提供另一解決方案，其基本概念為在訊源編碼器的輸出序列中存在一些冗餘訊息，我們選擇不移除這些冗餘訊息，並且在通道解碼的過程整合這些資訊來提供錯誤更正。事實上，如果某些冗息被原封不動的保留下來，那麼在索引之間所存在的資訊將散佈於壓縮序列中的數個索引之間。換句話說，如果雜訊使得這些索引之間的任一個產生錯誤，那麼關於此索引的資訊可以由觀察其相鄰的索引獲得。這就是在通道編碼之中，更正錯誤的方法。接下來就是為什麼我們要討論訊餘冗息的三

大原因：

[1] 殘餘冗息總是存在於編碼输出的符號串之間

就像之前所討論的，殘餘冗息總是存在於編碼後的索引之間。對於一般多媒體訊號源，如語音、音訊、視訊，經由訊號源編碼之後的殘餘冗息是非常值得注意的，這是因為這些訊號源總是呈現高度地不穩定，所以很難完全地壓縮。即使採用一些高壓縮率的語音編碼器，例如混和激發線性預估（the mixed-excitation linear prediction, MELP）訊源編碼器，仍有大量的冗息出現在編碼後的參數之間。

[2] 不需要更大的傳輸頻寬



有別於前向錯誤更正碼（forward error correcting code, FEC）[8]，利用殘餘冗息對抗通道雜訊不需要增加任何多餘的保護位元，亦即在傳輸頻寬的要求上不需要再多做增加，這在有限頻寬的通訊系統上是一項很大的優點。在[9]，Sayood *et al.* 模擬了一個考慮預估錯誤（prediction error）之間殘餘冗息的訊源解碼機制。預估錯誤序列被模擬為一階馬可夫過程，相鄰訊號源之相關性則表示為訊源轉移機率。再利用通道轉移機率將通道的狀態資訊納入考量，這兩個機率組合而為一個正比於接收序列之可能性的計量（metric）。

[3] 當通道雜訊很嚴重時，殘餘冗息將提供解碼所需的主要資訊

如同在[5]，訊源控制通道解碼器的判斷主要是依據通道的輸出以及訊號源的事前資訊。當通道環境良好，通道解碼器大部分依賴接收的序列；但當通道環境惡劣時，衰退現象非常嚴重，解碼時的參考選擇將依賴訊號源的事前消息甚過於所接收的序列。值得注意的是，在極差的通道環境將使得通道解碼喪失其錯誤更正能力，甚至還會引進更多錯誤出現。然而在惡劣通道環境，殘餘冗息在錯誤緩和 (error concealment) 的應用上扮演一個很重要的角色，這觀念在現在的行動通訊系統上也是很常使用，在[5]將 SCCD 應用於 GSM 的語音編碼上將提升 2dB 到 3dB 的改善，顯著的改善亦可在[2]中發現。



2.3 ISF 索引間的相互訊息

經由以上的介紹，可以知道殘餘冗息在現今或未來的傳輸系統中，在解碼部分將是一項重要的資訊。在下面文章之中，我們將分析不同程度的殘餘冗息，並在第五章的實驗模擬中證實了我們分析所得的假設。接下來，我們將分析語音訊號經過 AMR-WB 編碼之後 ISF 索引間所存在的殘餘冗息。

假設在第 n 個音框第 k 組索引 $X_n^{(k)}$ 所屬集合為 $U^{(k)}$ ，其索引為 $M^{(k)}$ 個位元所組成，我們想要探討的是索引存在的殘餘冗息。第一種情況是假設每個索引之間互相獨立，則第 k 個索引的熵值 (entropy) 定

義為：

$$H(X^{(k)}) \triangleq -\sum_{x^{(k)}} P(x^{(k)}) \log_2 P(x^{(k)}) \quad (2.1)$$

接下來要探討的是當索引呈現非均勻分佈 (non-uniform distribution)，以及框架間 (inter-frame) 和框架內 (intra-frame) 的記憶性，這裡我們將模擬為一階馬可夫程序。

當索引間的相互訊息不等於0時，表示不同索引之間有關聯性存在，進一步可以區分為框架間與框架內的關聯性。框架間關聯性定義為在連續兩個相鄰框架，不同框架中相同位置的索引之間所存在的相關性，而框架內關聯性定義為在同一個框架內不同兩個索引之間的相關性，如圖 2.4 所示。

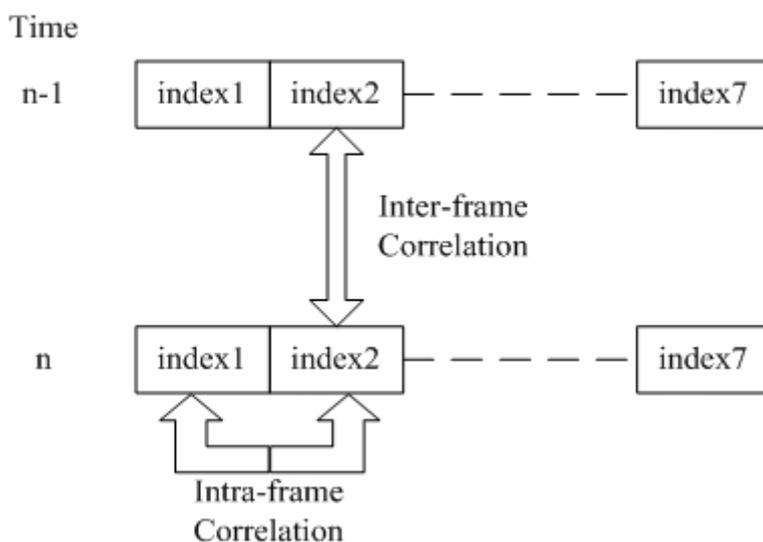


圖 2.4 框架內關聯性與框架間關聯性

如果只考慮框架間關聯性，且將索引的統計特性模擬為一階馬可夫模型，那麼第 k 組索引的熵值為

$$H(X_n^{(k)} | X_{n-1}^{(k)}) = - \sum_{x_n^{(k)}} \sum_{x_{n-1}^{(k)}} P(x_n^{(k)}, x_{n-1}^{(k)}) \log_2 P(x_n^{(k)} | x_{n-1}^{(k)}) \quad (2.2)$$

而其相對應的相互訊息為

$$I(X_n^{(k)}; X_{n-1}^{(k)}) = H(X_n^{(k)}) - H(X_n^{(k)} | X_{n-1}^{(k)}) \quad (2.1)$$

基於相同的推導與假設，針對第 k 組索引同一框架內與其他索引間相關性所得的熵值為

$$H(X_n^{(k)} | X_n^{(k')}) = - \sum_{x_n^{(k)}} \sum_{x_n^{(k')}} P(x_n^{(k)}, x_n^{(k')}) \log_2 P(x_n^{(k)} | x_n^{(k')}) \quad (2.3)$$

其中 k' 為同一框架內任一組索引，相對應的相互訊息為

$$I(X_n^{(k)}; X_n^{(k')}) = H(X_n^{(k)}) - H(X_n^{(k)} | X_n^{(k')}) \quad (2.4)$$

2.4 分析討論

接下來，我們將用上述的討論來分析 AMR-WB 中 ISF 索引框架間與框架內所存在的相互訊息。在訊源編碼處理部分，利用 16 kHz 取樣的大量語音檔經由 3GPP 所公佈的 AMR-WB 語音編碼器，得到 194881 個音框，再計算不同索引之間所存在的相互訊息（框架間與框架內的關聯性），相互訊息的計算方法如 (2.3) 與 (2.5) 式所示。表 2.2 與 2.3 分別為七組 ISF 索引框架間記憶性與框架內記憶性的相互訊息，表 2.4 為七組 ISF 索引結合框架間與光架內記憶性的相互訊息。

在表 2.2 中，我們可以發現索引一與索引二的相鄰框架間的相互訊息比較高。由圖 2.3 了解，因為這兩索引是直接對 ISF 係數量化，所以在相鄰兩框架間有比較高的相互訊息。至於其他索引，由於是對第一級量化後所剩餘的係數，再進一步量化，所以相鄰框架間的相互訊息也就不高，反而與所屬前一級索引會有較高的相互訊息，從表 2.3 中，我們可以發現這點。索引三、索引四與索引五是利用索引一量化所剩餘的係數再分別作量化所得，所以與索引一相互訊息反而比索引本身相鄰框架間的相互訊息還來的高。在索引六及索引七與索引二間的框架間相互訊息也來的比較高。



k	1	2	3	4	5	6	7
$I\{X_n^{(k)}, X_{n-1}^{(k)}\}$	2.041	1.244	0.183	0.135	0.184	0.037	0.020

表 2.2 ISF 索引值框架間的相互訊息

$I(X_n^{(k)}, X_n^{(k')})$	k'	1	2	3	4	5	6	7
k								
1		7.353	1.211	0.678	0.515	0.503	0.150	0.108
2		1.211	7.451	0.194	0.208	0.262	0.235	0.231
3		0.677	0.194	5.770	0.075	0.052	0.016	0.020
4		0.515	0.208	0.075	6.917	0.120	0.025	0.021
5		0.502	0.262	0.053	0.120	6.862	0.042	0.024
6		0.150	0.235	0.016	0.025	0.042	4.946	0.014
7		0.110	0.232	0.021	0.021	0.024	0.015	4.918

表 2.3 ISF 索引值框架內的相互訊息



第三章 訊源控制通道解碼機制

有別於傳統的通道解碼架構，訊源控制通道解碼器除了接收訊號 \hat{Y}_n 之外，其輸入端還需要額外提供訊源的事前訊息。標準的 SCCD 架構如圖 3.1 所示。當訊號通過一良好的通道環境時，通道解碼器將大部分依賴接收到的索引值。事前訊息的最大好處是在通道環境惡劣時，通道解碼效能將依賴索引值的事前資訊甚過於收到的索引值。這是因為在非常惡劣的通道下，接收到的索引將存在太多的通道錯誤，使得錯誤更正機制很難有效地正確解碼。而最常被使用的事前資訊，是根據訊號源量化索引的機率分佈作分析，可以由大量的訓練序列而得。

原始訊號在經過訊源編碼處理之後，被壓縮的語音參數對於通道錯誤的抵抗能力明顯降低。尤其當通道環境極差的時候，在遭受無可更正的錯誤之後，系統效能將會嚴重地下降。在這種情況之下，SCCD 將會是一個針對接收序列，既不增加傳輸頻寬能有效改善輸出品質的有效方法。其關鍵在於有效整合框架間或框架內的相關性，計算所有可能索引值的後驗機率。接下來，我們將要針對不同程度的殘餘冗息，規劃設計所其對應的索引後驗機率演算法。

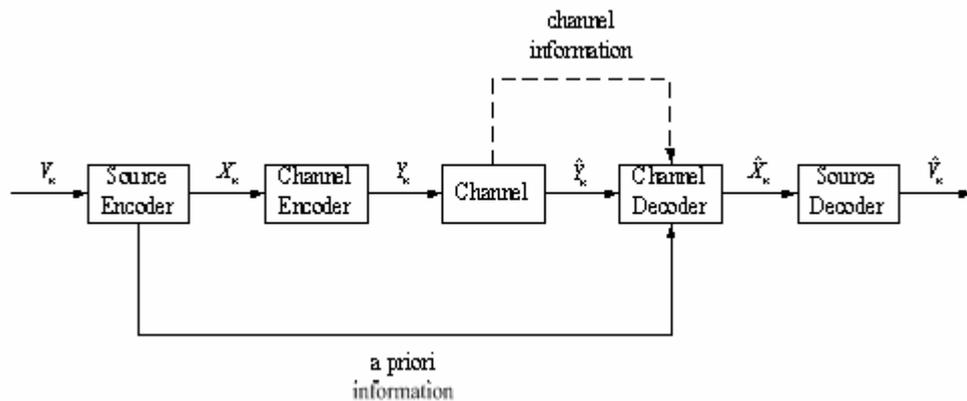


圖 3.1 訊源控制通道解碼架構

3.1 傳輸系統

圖 3.2 為本章節所使用的系統方塊圖，接下來將分別簡介每個工作單元的功能。



S-MSVQ/量化碼書查表(VQ codebook lookup)

S-MSVQ 主要目的是壓縮訊號源參數以降低傳輸速率，其量化方法已在 2.1 節描述，主要是將 16 個 ISF 映射到七組 ISF 索引值上。而在第 n 個音框，被量化的訊號源 ISF_n 將以索引值 $X_n = (X_n^{(1)}, X_n^{(2)}, \dots, X_n^{(7)})$ 代表。在接收端，訊源解碼器將依據解碼而得的索引值 \hat{X}_n 經由量化碼書查表比對重建其訊號為 \widehat{ISF}_n 。

通道編碼/解碼器(Channel Encoder/Decoder)

通道編碼/解碼器是通訊系統中極為必要的一個部分，因為它對於訊號源提供了對抗錯誤的保護功能。藉由增加一些多餘的保護位元在資訊位元上，不僅可以提供錯誤偵測還有錯誤的更正，使得傳送的資訊受到通道錯誤所產生的影響減小。假設編碼器的編碼率為 $1/R$ ，若令傳送第 k 組位元數為 $M^{(k)}$ 的索引 $X_n^{(k)}$ ，通道編碼器輸出為 $Y_n^{(k)} = (y_{n,0}^{(k)}, y_{n,1}^{(k)}, \dots, y_{n,M^{(k)}-1}^{(k)})$ ，而被通道錯誤影響後所接收到的序列為 $\hat{Y}_n^{(k)} = (\hat{y}_{n,0}^{(k)}, \hat{y}_{n,1}^{(k)}, \dots, \hat{y}_{n,M^{(k)}-1}^{(k)})$ ，則通道解碼器的輸出為一近似 X_n 的索引 \hat{X}_n 。



可加性高斯白雜訊通道(AWGN Channel)

通道是使傳輸訊號受到干擾與雜訊的傳輸媒介，因此接收到的訊號 $\hat{Y}_n^{(k)}$ 會與傳送的訊號 $Y_n^{(k)}$ 不一致。傳統的通道錯誤包括高斯雜訊(Gaussian noise)，多路徑衰退(multipath fading)，相對路徑延遲(relative path delay)，杜卜勒效應(Doppler shift)等。而本文則僅考慮可加性高斯白雜訊通道。

最大後驗機率解碼器(MAP 解碼器)

一般最常見的兩種估測方法為最小均方錯誤解碼器(MMSE decoder)

[10]與最大後驗機率解碼器 [11]。對於一個離散系統，MMSE 解碼器首在為是減小原始與估計的索引值之間的均方誤差的最佳方法，而 MAP 解碼器則是以降低解碼的位元錯誤率(bit error rate, BER)為目標。本論文中我們將使用的是 MAP 解碼器。

在接收端收到序列 $\hat{\mathbf{Y}}_n = (\hat{Y}_n^{(1)}, \hat{Y}_n^{(2)}, \dots, \hat{Y}_n^{(7)})$ 後，針對 7 組索引值個別計算其後驗機率，我們將會得到 $2^{M^{(k)}}$ 個軟性輸出 $P(X_n^{(k)} | \hat{\mathbf{Y}}_n)$ ，其中 $k=1, 2, \dots, 7$ 。再利用最大後驗機率的估計準則

$$\hat{X}_n^{(k)} = \arg \max_{X_n^{(k)}} P(X_n^{(k)} | \hat{\mathbf{Y}}_n) \quad (3.1)$$

可估測得到符號值 $\hat{X}_n^{(k)}$ ，再利用量化碼書查表比對查詢可得 $\widehat{ISF}_n^{(k)}$ 。

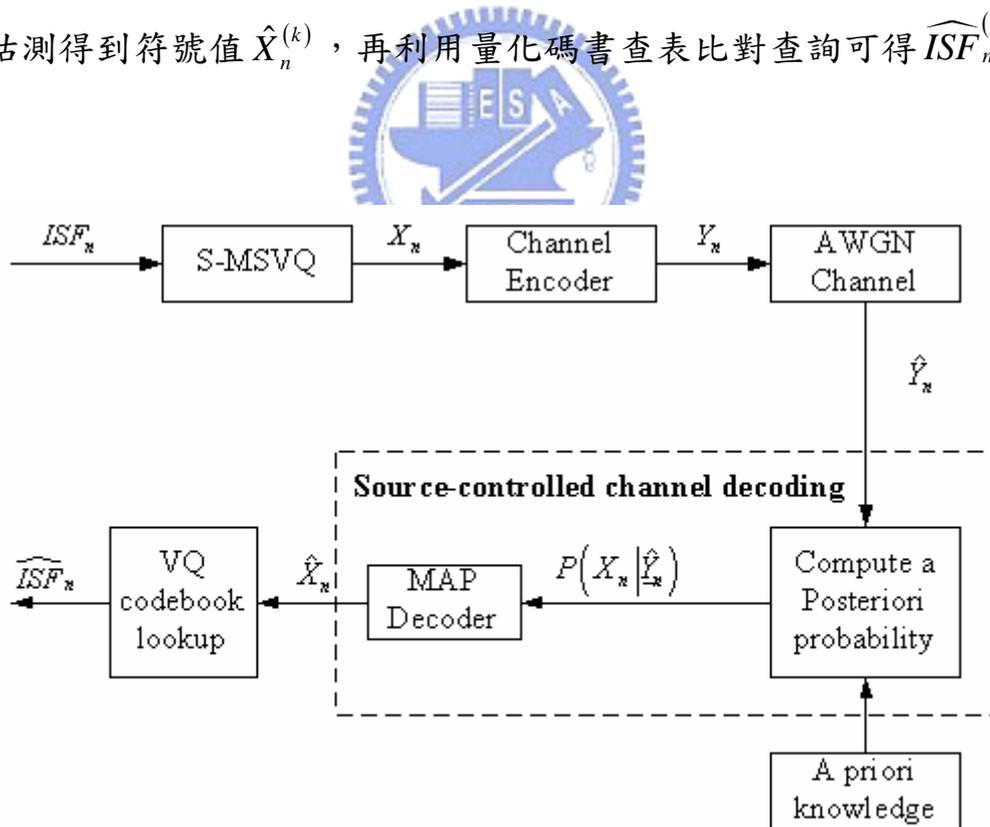


圖 3.2 訊號傳輸系統方塊圖

3.2 訊源冗息的運用

由於編碼器的不完美導致其輸出索引間存在殘餘冗息，而這些冗息是相鄰索引間的記憶性或索引本身的不均勻機率分佈所造成。我們主要是要利用這些殘餘冗息，設計一有效重建 ISF 係數的最大後驗機率解碼器。為了簡化機率的推導流程，我們將暫先不考慮通道編碼處理，也就是假設 $X_n^{(k)} = Y_n^{(k)}$ ，僅利用不同層級的殘餘冗息來設計後驗機率演算法，相關研究可參考[12]。

[1] Instantaneous MAP (IMAP)

IMAP 解碼器是最基本的 MAP 解碼器，是假設係數量化後索引框架間或框架內不存在任何相關性，且計算後驗機率時不考慮過去所收到的訊息。其解碼機制可表示成

$$\hat{X}_n^{(k)} = \arg \max_{X_n^{(k)}} P\left(X_n^{(k)} \mid \hat{Y}_n^{(k)}\right) \quad (3.2)$$

進一步將 $P\left(X_n^{(k)} \mid \hat{Y}_n^{(k)}\right)$ 展開成

$$P\left(X_n^{(k)} \mid \hat{Y}_n^{(k)}\right) = C \cdot P\left(X_n^{(k)}\right) \cdot P\left(\hat{Y}_n^{(k)} \mid X_n^{(k)}\right) \quad (3.3)$$

其中 $C = P\left(\hat{Y}_n^{(k)}\right)$ 為正規化因子， $P\left(X_n^{(k)}\right)$ 為事前訓練的索引機率， $P\left(\hat{Y}_n^{(k)} \mid X_n^{(k)}\right)$ 則為通道轉移機率(channel transition probability)，基於無記憶性通道(memoryless channel)的假設，通道轉移機率可展開成

$$P(\hat{Y}_n^{(k)} | X_n^{(k)}) = \prod_{m=1}^{M^{(k)}} P(y_{n,m}^{(k)} | x_{n,m}^{(k)}) \quad (3.4)$$

一個位元錯誤率 ε 的在二進制對稱通道(Binary Symmetric Channel) 中，是根據索引的接收值 $\hat{Y}_n^{(k)}$ 與傳輸索引 $X_n^{(k)}$ 比較，再求得通道轉移機

$$P(\hat{Y}_n^{(k)} | X_n^{(k)}) = \varepsilon^{h(X_n^{(k)}, \hat{Y}_n^{(k)})} (1-\varepsilon)^{M^{(k)}-h(X_n^{(k)}, \hat{Y}_n^{(k)})} \quad (3.5)$$

其中 $h(X_n^{(k)}, \hat{Y}_n^{(k)})$ 為 $X_n^{(k)}$ 與 $\hat{Y}_n^{(k)}$ 兩索引的漢明距離(Hamming Distance)。

[2] Sequence MAP1 (SMAP1)

有別於 IMAP, SMAP1 的後驗機率計算同時考慮目前與過去所有接收序列，以期能額外運用特定係數索引不同框架間的相關性。其 MAP 解碼器可表示成

$$\hat{X}_n^{(k)} = \arg \max_{X_n^{(k)}} P(X_n^{(k)} | \hat{Y}_n^{(k)}) \quad (3.6)$$

其中接收序列表示為 $\hat{Y}_n^{(k)} = (\hat{Y}_1^{(k)}, \hat{Y}_2^{(k)}, \dots, \hat{Y}_n^{(k)})$ 。利用貝氏定理可將

$P(X_n^{(k)} | \hat{Y}_n^{(k)})$ 展開為

$$P(X_n^{(k)} | \hat{Y}_n^{(k)}) = C \cdot P(X_n^{(k)}, \hat{Y}_n^{(k)}) \quad (3.7)$$

其中 $C = P(\hat{Y}_n^{(k)})$ 為正規化因子。在此考量第 k 組索引在相鄰兩框架間的相關性，可分解成

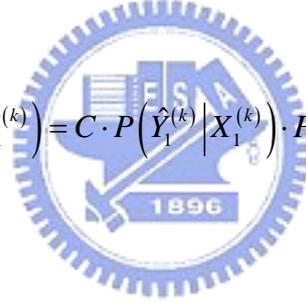
$$\begin{aligned}
P\left(X_n^{(k)}, \hat{Y}_n^{(k)}\right) &= \sum_{X_{n-1}^{(k)}} P\left(X_n^{(k)}, X_{n-1}^{(k)}, \hat{Y}_n^{(k)}\right) \\
&= \sum_{X_{n-1}^{(k)}} P\left(X_{n-1}^{(k)}, \hat{Y}_{n-1}^{(k)}\right) \cdot P\left(X_n^{(k)}, \hat{Y}_n^{(k)} \mid X_{n-1}^{(k)}, \hat{Y}_{n-1}^{(k)}\right) \\
&= \sum_{X_{n-1}^{(k)}} P\left(X_{n-1}^{(k)}, \hat{Y}_{n-1}^{(k)}\right) \cdot P\left(\hat{Y}_n^{(k)} \mid X_n^{(k)}, X_{n-1}^{(k)}, \hat{Y}_{n-1}^{(k)}\right) \cdot P\left(X_n^{(k)} \mid X_{n-1}^{(k)}, \hat{Y}_{n-1}^{(k)}\right)
\end{aligned} \tag{3.8}$$

其中第二項根據無記憶性通道特性可簡化成 $P\left(\hat{Y}_n^{(k)} \mid X_n^{(k)}, X_{n-1}^{(k)}, \hat{Y}_{n-1}^{(k)}\right) = P\left(\hat{Y}_n^{(k)} \mid X_n^{(k)}\right)$ 。至於式中第三項是考量框架間索引的關聯性， $P\left(X_n^{(k)} \mid X_{n-1}^{(k)}, \hat{Y}_{n-1}^{(k)}\right) = P\left(X_n^{(k)} \mid X_{n-1}^{(k)}\right)$ 這項可經由大量語料事前訓練而求得。最後(3.7)可整理成

$$P\left(X_n^{(k)} \mid \hat{Y}_n^{(k)}\right) = C \cdot \sum_{X_{n-1}^{(k)}} P\left(X_{n-1}^{(k)}, \hat{Y}_{n-1}^{(k)}\right) \cdot P\left(\hat{Y}_n^{(k)} \mid X_n^{(k)}\right) \cdot P\left(X_n^{(k)} \mid X_{n-1}^{(k)}\right) \tag{3.9}$$

其初始值的設定為

$$P\left(X_1^{(k)} \mid \hat{Y}_1^{(k)}\right) = C \cdot P\left(\hat{Y}_1^{(k)} \mid X_1^{(k)}\right) \cdot P\left(X_1^{(k)}\right) \tag{3.10}$$



[3] SMAP2

SMAP2 與 SAMP1 不同的地方是在計算後驗機率時，額外考慮了框架內不同參數索引值之建的相關性，其 MAP 解碼器可表示成

$$\hat{X}_n^{(k)} = \arg \max_{X_n^{(k)}} P\left(X_n^{(k)} \mid \hat{Y}_n^{(k)}, \hat{Y}_n^{(k')}\right) \tag{3.11}$$

其中 $\hat{Y}_n^{(k')}$ 代表第 k 個音框內第 k' 組索引的接收值。根據貝式定理，可將 $P\left(X_n^{(k)} \mid \hat{Y}_n^{(k)}, \hat{Y}_n^{(k')}\right)$ 展開成

$$P\left(X_n^{(k)} \mid \hat{Y}_n^{(k)}, \hat{Y}_n^{(k')}\right) = C \cdot P\left(X_n^{(k)}, \hat{Y}_n^{(k)}\right) \cdot P\left(\hat{Y}_n^{(k')} \mid X_n^{(k)}, \hat{Y}_n^{(k)}\right) \tag{3.12}$$

其中 $C = P\left(\hat{Y}_n^{(k)}, \hat{Y}_n^{(k')}\right)$ 為正規化參數。第二項已在(3.8)式所描述。第三

項可簡化成 $P(\hat{Y}_n^{(k')} | X_n^{(k)}, \hat{Y}_n^{(k)}) = P(\hat{Y}_n^{(k')} | X_n^{(k)})$ ，再進一步展開成

$$\begin{aligned} P(\hat{Y}_n^{(k')} | X_n^{(k)}) &= \sum_{X_n^{(k')}} P(\hat{Y}_n^{(k')}, X_n^{(k')} | X_n^{(k)}) \\ &= \sum_{X_n^{(k')}} P(\hat{Y}_n^{(k')} | X_n^{(k')}, X_n^{(k)}) \cdot P(X_n^{(k')} | X_n^{(k)}) \end{aligned} \quad (3.13)$$

其中 $P(\hat{Y}_n^{(k')} | X_n^{(k')}, X_n^{(k)}) = P(\hat{Y}_n^{(k')} | X_n^{(k)})$ 為通道轉移機率。至於 $P(X_n^{(k')} | X_n^{(k)})$

這，則是考慮框架內第 k 組與第 k' 組索引間的相關性，也是由大量語料事前訓練所求得。

3.3 錯誤控制碼與訊源冗息之整合

與 3.2 節不同是加入了錯誤控制碼，整個系統方塊圖如圖 3.2 所示。以下我們鎖定迴旋碼，先討論其編碼架構，再深入探討最大後驗機率的相關演算。



3.3.1 迴旋編碼的架構

考慮一個暫存記憶體為 L ，限制長度 (constraint length) 為 $L+1$ ，而編碼率為 $1/R$ 的通道編碼器。一般而言，在時刻 n 索引 $X_n^{(k)}$ 所屬 $M^{(k)}$ 個位元序列為 $x_{n,0}^{(k)}, x_{n,1}^{(k)}, \dots, x_{n,M^{(k)}-1}^{(k)}$ 輸入至編碼器，其輸出端會產生另一組位元序列 $Y_n^{(k)} = (y_{n,0}^{(k)}, y_{n,1}^{(k)}, \dots, y_{n,M^{(k)}-1}^{(k)})$ ，其中位元 $x_{n,m}^{(k)}$ 所對應的碼字為 $y_{n,m}^{(k)} = (y_{n,m,0}^{(k)}, y_{n,m,1}^{(k)}, \dots, y_{n,m,R-1}^{(k)})$ 。有別於傳統依單一位元編碼處理模式，我們將要探討每一個單位時間輸入一個 $M^{(k)}$ 位元的索引，致

使編碼器的移位暫存器(shift register)在同一單位時間內也移動了 $M^{(k)}$ 個位元。在下一段，我們將詳盡也探討此基於索引之迴旋編碼器。

迴旋碼的產生多項式(encoder generator polynomials)定義為：

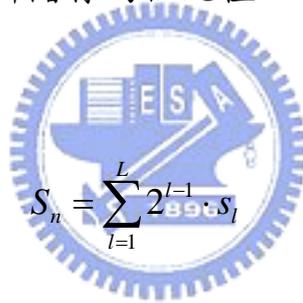
$$G_r(D) = g_0^r + Dg_1^r + D^2g_2^r + \dots + D^Lg_L^r, \quad r = 0, 1, \dots, R-1 \quad (3.14)$$

而對個別碼字位元 $y_{n,m,r}^{(k)}, m = 0, 1, \dots, M^{(k)} - 1, r = 0, 1, \dots, R-1$ 而言，

$$y_{n,m,r}^{(k)} = x_{n,m}^{(k)} \cdot g_0^r + \sum_{l=1}^L g_l^r \cdot s_l \pmod{2}, \quad g_l^r \in \{0, 1\} \quad (3.15)$$

其中 s_l 為移位暫存器中所儲存的位元值，而暫存器的狀態 S_n 由 s_l ，

$l = 1, \dots, L$ 所決定。



$$S_n = \sum_{l=1}^L 2^{l-1} \cdot s_l \quad (3.16)$$

值得注意的是，暫存器的狀態是由前一時刻輸入索引的位元組合完全進入暫存器之後計算得到。 S_n 將由最近 M 個輸入位元決定，其中若 $M > L$ ，則由單位時間內輸入至編碼器的最後 L 個位元決定，前 $M - L$ 個位元將不影響狀態值 S_n 。反之，若 $M < L$ ，則 S_n 將由 $n - 1$ 時刻輸入的 M 位元以及 $n - 2$ 時刻所輸入至編碼器的後 $L - M$ 位元決定。由(3.16)式可得知，此暫存器共可產生 2^L 種可能出現的狀態。另外，迴旋編碼器的起始狀態 S_1 需設定為0。全部的輸入序列傳送完畢之後，必須額外傳送 L 個為0的尾巴位元(tail bits)進入編碼器，以迫

使編碼器的狀態回歸至0。

在第五章系統模擬實驗將用到的一個典型迴旋編碼器架構如圖

3.3 所示，此編碼器的限制長度為5，編碼率為1/2，產生多項式為

$$\begin{aligned} G_1(D) &= 1 + D^3 + D^4 \\ G_2(D) &= 1 + D^1 + D^3 + D^4 \end{aligned} \quad (3.17)$$

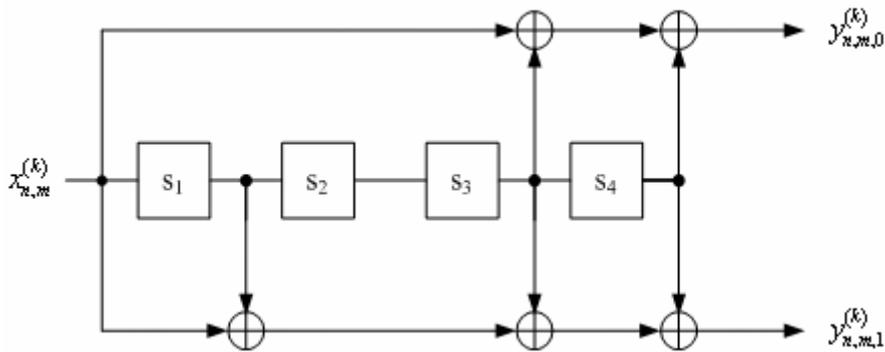


圖 3.3 迴旋編碼器架構

3.3.2 最大後驗機率的推導

整合 3.2 節所提訊源冗息的運用，我們可將 IMAP 與 SMAP1 做適度的改變，以期可應用在同時執行通道編碼處理的傳輸系統上。以下我們將對上述的兩種演算法做進一步的探討。

[1] IMAP

IMAP 解碼器與(3.2)式相同，但必須額外考慮迴旋碼碼器存在狀態轉移的情形，所以將(3.3)式展開成

$$\begin{aligned}
P\left(X_n^{(k)} \mid \hat{Y}_n^{(k)}\right) &= C \cdot \sum_{S_n^{(k)}} P\left(X_n^{(k)}, S_n^{(k)}, \hat{Y}_n^{(k)}\right) \\
&= C \cdot \sum_{S_n^{(k)}} P\left(\hat{Y}_n^{(k)} \mid X_n^{(k)}, S_n^{(k)}\right) \cdot P\left(X_n^{(k)}, S_n^{(k)}\right)
\end{aligned} \tag{3.18}$$

其中 $C = P(\hat{Y}_n^{(k)})$ 為正規化因子，基於高斯無記憶性通道假設，可將 $P(\hat{Y}_n^{(k)} \mid X_n^{(k)}, S_n^{(k)})$ 做進一步的分解成

$$P\left(\hat{Y}_n^{(k)} \mid X_n^{(k)}, S_n^{(k)}\right) = K_i \cdot \exp\left(\frac{1}{\sigma^2} \cdot \sum_{m=0}^{M^{(k)}-1} \sum_{r=0}^{R-1} \left(y_{n,m,r}^{(k)} - \hat{y}_{n,m,r}^{(k)}\right)^2\right) \tag{3.19}$$

其中 K_i 為一常數， σ^2 為 AWGN 通道雜訊的變異數(variance)， $y_{n,m,r}^{(k)}$ 為當時狀態 $S_n^{(k)}$ 與輸入索引 $X_n^{(k)}$ 共同決定的碼字位元。接著考慮(3.18)式中的 $P(X_n^{(k)}, S_n^{(k)})$ ，我們假設此刻狀態 $S_n^{(k)}$ 與輸入索引值 $X_n^{(k)}$ 無關，而狀態機率 $P(X_n^{(k)})$ 為一常數，所以最後可以將(3.18)式整理成

$$P\left(X_n^{(k)} \mid \hat{Y}_n^{(k)}\right) = C \cdot P\left(X_n^{(k)}\right) \sum_{S_n^{(k)}} P\left(\hat{Y}_n^{(k)} \mid X_n^{(k)}, S_n^{(k)}\right) \tag{3.20}$$

為了避免不必要的路徑產生影響後驗機率計算，其初始值設定為

$$P\left(X_1^{(k)} \mid \hat{Y}_1^{(k)}\right) = C \cdot P\left(X_1^{(k)}\right) \cdot P\left(\hat{Y}_1^{(k)} \mid X_1^{(k)}, S_1^{(k)} = 0\right) \tag{3.21}$$

之所以會有這種情形，是因為當初暫存器的起始狀態為 0。

[2] SMAP1

MAP 解碼器與(3.6)式相同，但由於有狀態轉移的現象，首先我們要將 $P(X_n^{(k)} \mid \hat{Y}_n^{(k)})$ 展開成

$$P\left(X_n^{(k)} \mid \hat{Y}_n^{(k)}\right) = C \cdot \sum_{S_n^{(k)}} P\left(X_n^{(k)}, S_n^{(k)}, \hat{Y}_n^{(k)}\right) \tag{3.22}$$

其中 $C = P(\hat{Y}_n^{(k)})$ 為正規化因子，根據貝氏定理可將 $P(X_n^{(k)}, S_n^{(k)}, \hat{Y}_n^{(k)})$ 分解

成

$$\begin{aligned}
 P\left(X_n^{(k)}, S_n^{(k)}, \hat{Y}_n^{(k)}\right) &= \sum_{X_{n-1}^{(k)}} \sum_{S_{n-1}^{(k)}} P\left(X_n^{(k)}, X_{n-1}^{(k)}, S_n^{(k)}, S_{n-1}^{(k)}, \hat{Y}_n^{(k)}\right) \\
 &= \sum_{X_{n-1}^{(k)}} \sum_{S_{n-1}^{(k)}} P\left(X_n^{(k)}, S_n^{(k)}, \hat{Y}_n^{(k)}\right) \cdot \gamma_{X_n, X_{n-1}}\left(S_n^{(k)}, S_{n-1}^{(k)}, \hat{Y}_n^{(k)}\right)
 \end{aligned} \tag{3.23}$$

其中 $\gamma_{X_n, X_{n-1}}\left(S_n^{(k)}, S_{n-1}^{(k)}, \hat{Y}_n^{(k)}\right) = P\left(X_n^{(k)}, S_n^{(k)}, \hat{Y}_n^{(k)} \mid X_{n-1}^{(k)}, S_{n-1}^{(k)}, \hat{Y}_{n-1}^{(k)}\right)$ 是由通道轉移機率與編碼器狀態轉移所組成的。再利用貝氏定理進一步分解

$$\begin{aligned}
 \gamma_{X_n, X_{n-1}}\left(S_n^{(k)}, S_{n-1}^{(k)}, \hat{Y}_n^{(k)}\right) &= P\left(\hat{Y}_n^{(k)} \mid X_n^{(k)}, S_n^{(k)}, X_{n-1}^{(k)}, S_{n-1}^{(k)}, \hat{Y}_{n-1}^{(k)}\right) \\
 &\quad \cdot P\left(X_n^{(k)} \mid S_n^{(k)}, X_{n-1}^{(k)}, S_{n-1}^{(k)}, \hat{Y}_{n-1}^{(k)}\right) \\
 &\quad \cdot P\left(S_n^{(k)} \mid X_{n-1}^{(k)}, S_{n-1}^{(k)}, \hat{Y}_{n-1}^{(k)}\right)
 \end{aligned} \tag{3.24}$$

其中 $P\left(\hat{Y}_n^{(k)} \mid X_n^{(k)}, S_n^{(k)}, X_{n-1}^{(k)}, S_{n-1}^{(k)}, \hat{Y}_{n-1}^{(k)}\right)$ 再無記憶通道可簡化成 $P\left(\hat{Y}_n^{(k)} \mid X_n^{(k)}, S_n^{(k)}\right)$ 如(3.19)式所描述。式中第二項 $P\left(X_n^{(k)} \mid S_n^{(k)}, X_{n-1}^{(k)}, S_{n-1}^{(k)}, \hat{Y}_{n-1}^{(k)}\right) = P\left(X_n^{(k)} \mid X_{n-1}^{(k)}\right)$ 意謂著相鄰框架間索引的關聯性。比較值得討論的是 $P\left(S_n^{(k)} \mid X_{n-1}^{(k)}, S_{n-1}^{(k)}\right)$ ，這項機率是由於採用迴旋碼編碼器所造成的，其物理意義就是當前一刻的索引值 $X_{n-1}^{(k)}$ 及暫存器狀態 $S_{n-1}^{(k)}$ 已決定後，下一刻的暫存器狀態 $S_n^{(k)}$ 也會被決定，也就是說這項機率不是 0 就是 1，決定了相鄰兩個時間點之間狀態的轉移與路徑行徑方向。最後(3.27)式可整理成

$$\gamma_{X_n, X_{n-1}}\left(S_n^{(k)}, S_{n-1}^{(k)}, \hat{Y}_n^{(k)}\right) = P\left(\hat{Y}_n^{(k)} \mid X_n^{(k)}, S_n^{(k)}\right) \cdot P\left(X_n^{(k)} \mid X_{n-1}^{(k)}\right) \cdot P\left(S_n^{(k)} \mid X_{n-1}^{(k)}, S_{n-1}^{(k)}\right) \tag{3.25}$$

第四章

多重敘述的訊源控制通道解碼機制

多重敘述向量量化主要將原始訊號送入編碼器，使其輸出多重敘述再利用個別的獨立通道傳送，以期對抗封包漏失對解碼的影響。舉例而言，將訊號利用 MDVQ 編碼器使其輸出兩個索引，再利用兩個獨立通道個別傳輸。在接收端，如果同時接收到兩個索引，則可獲得最好的訊號重建品質。反之，若只收到單一索引，MDVQ 解碼器仍然可重建一相當程度的訊號品質。在以前的研究，多重敘述向量量化主要用來克服封包漏失問題。我們則將近一不擴展其應用在位元錯誤通道上。針對於此，我們將整合 3.3 節的理論與多重敘述向量量化，使系統的整體解碼效能有所提升。接下來，我們要先說明多重敘述向量量化系統架構。

4.1 多重敘述向量量化

4.1.1 系統架構

多重敘述向量量化之目的是在網路傳輸頻寬限制下，進行資料壓縮去除其多餘訊息再作傳輸，同時保證解碼端仍能還原出高品質的合

成訊號，圖 4.1 為其傳輸架構。其編碼模式可視為向量量化和指定索引(Index Assignment)兩項對映函數的組合。量化器的主要構成元素為一組能涵蓋訊號特徵變化的範本碼書 $W = (w_1, w_2, \dots, w_{2^M})$ ，其內容是由 2^M 個維度為 d 之碼字向量所組成。在編碼端，與輸入向量 V 依序比對選擇其中最近似的碼字，再輸出其對應的 M 位元索引 X ，其索引集合為 $U = (1, 2, \dots, 2^M)$ ，經由後續的索引指定函數產生兩個敘述 $I = \delta_i(X)$ 和 $J = \delta_j(X)$ 。假設敘述 I 和 J 均屬於同一集合 $(1, 2, \dots, N)$ ，並且分別視其為一個 $N \times N$ 維指定矩陣 (assignment matrix) 的列索引(row index)和行索引(column index)。舉例而言， $M=3$ 且 $N=4$ 的指定矩陣如圖 4.2 所示，其中 $Row_i = \{X | \delta_i(X) = i\}$ 和 $Col_j = \{X | \delta_j(X) = j\}$ ，分別代表指定矩陣中第 i 列和第 j 行所屬的量化索引集合。索引指定的關鍵研究課題，是如何在該矩陣所屬的 N^2 位置選擇配置 2^M 個量化索引，以期能緩和封包漏失所衍生的解碼輸出失真。在最佳化計算領域，這是一個 NP (Non-deterministic Polynomial) 的複雜問題，因為總共有 $\binom{N^2}{2^M} \cdot 2^M!$ 種可能的配置組合，嘗試全面性搜尋以求最佳的索引指定並不可行，針對此議題，我們 4.2 節介紹。接下來要先介紹最佳 MDVQ 解碼器。

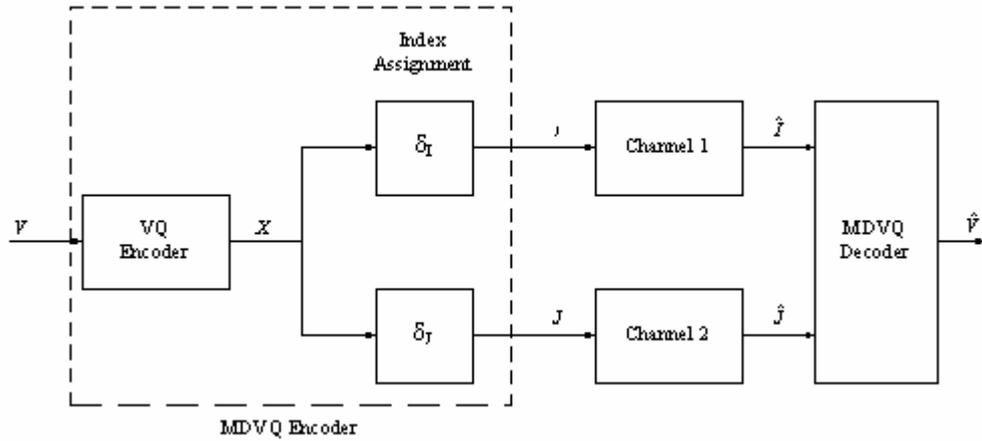


圖 4.1 多重敘述傳輸系統架構

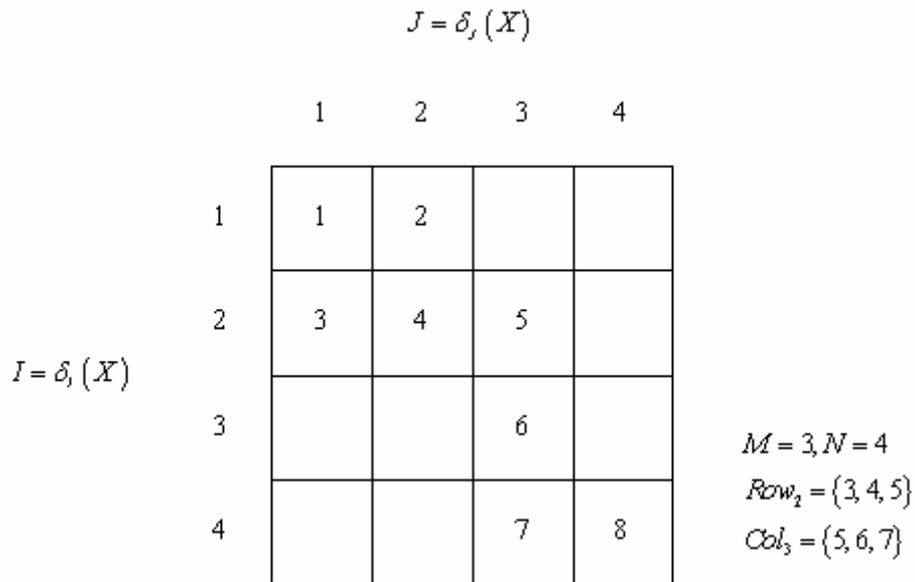


圖 4.2 指定矩陣的配置範例

4.1.2 最佳 MDVQ 解碼器

最佳 MDVQ 解碼器主要是以(4.1)式為準則

$$\hat{V}(\hat{I}, \hat{J}) = \arg \min_{\hat{V}} \left\{ E \left[\|w_X - \hat{V}\|^2 \mid \hat{I} = \hat{i}, \hat{J} = \hat{j} \right] \right\} \quad (4.1)$$

其中 w_X 為索引 X 所對應的碼字向量。 $\hat{V}(\hat{I}, \hat{J})$ 為最佳量化重建向量，是

根據所接收兩通道訊息 \hat{I} 與 \hat{J} 所重建。(4. 1)式的目標是在減少 w_x 與 $\hat{V}(\hat{I}, \hat{J})$ 之間的均方誤差，根據預估理論可得

$$\hat{V}(\hat{I}, \hat{J}) = \sum_{t=1}^{2^M} w_t \cdot P(X=t | \hat{I}=\hat{i}, \hat{J}=\hat{j}) \quad (4.2)$$

其關鍵在於準確計算個別索引的探討事後機率 $P(X=t | \hat{I}=\hat{i}, \hat{J}=\hat{j})$ 。若考慮兩通道間互相獨立特性，可利用貝式定理將 $P(X=t | \hat{I}=\hat{i}, \hat{J}=\hat{j})$ 展開成

$$\begin{aligned} & P(X=t | \hat{I}=\hat{i}, \hat{J}=\hat{j}) \\ &= \frac{P(X=t) \cdot P(\hat{I}=\hat{i}, \hat{J}=\hat{j} | X=t)}{P(\hat{I}=\hat{i}, \hat{J}=\hat{j})} \\ &= \frac{P(X=t) \cdot P(\hat{I}=\hat{i}, \hat{J}=\hat{j} | I=\delta_I(t), J=\delta_J(t))}{P(\hat{I}=\hat{i}, \hat{J}=\hat{j})} \quad (4.3) \\ &= \frac{P(X=t) \cdot P(\hat{I}=\hat{i} | I=\delta_I(t)) \cdot P(\hat{J}=\hat{j} | J=\delta_J(t))}{P(\hat{I}=\hat{i}, \hat{J}=\hat{j})} \\ &= \frac{P(X=t) \cdot P(\hat{I}=\hat{i} | I=\delta_I(t)) \cdot P(\hat{J}=\hat{j} | J=\delta_J(t))}{\sum_{t=1}^{2^M} P(X=t) \cdot P(\hat{I}=\hat{i} | I=\delta_I(t)) \cdot P(\hat{J}=\hat{j} | J=\delta_J(t))} \end{aligned}$$

若令 p_I 與 p_J 分別為兩通道個別的封包漏失率，則

$$P(\hat{I}=\hat{i} | I=\delta_I(t)) = \begin{cases} p_I & , \text{ if } \hat{i} = \emptyset \\ 1-p_I & , \text{ if } \hat{i} = \delta_I(t) \end{cases} \quad (4.4)$$

$$P(\hat{J}=\hat{j} | J=\delta_J(t)) = \begin{cases} p_J & , \text{ if } \hat{j} = \emptyset \\ 1-p_J & , \text{ if } \hat{j} = \delta_J(t) \end{cases} \quad (4.5)$$

其中 \emptyset 為表示封包漏失的空集合。如今考慮傳送索引值 $X=x$ ，若只收到單一敘述情況下，也就是 $\hat{i}=i=\delta_I(t)$ 且 $\hat{j}=\emptyset$ ，根據(4.3)、(4.4)與(4.5)式。可以得到其後驗機率為

$$P(X=t | \hat{I}=\hat{i}, \hat{J}=\emptyset) = \frac{P(X=t) \cdot (1-p_I) \cdot p_J}{P(\hat{I}=\hat{i}, \hat{J}=\emptyset)} \quad (4.6)$$

其中

$$P(\hat{I} = \hat{i}, \hat{J} = \emptyset) = (1 - p_I) \cdot p_J \cdot \sum_{\forall t \in Row_i} P(X = t) \quad (4.7)$$

將(4.7)式代入(4.6)式可得到

$$P(X = t | \hat{I} = \hat{i}, \hat{J} = \emptyset) = \begin{cases} \frac{P(X = t)}{\sum_{\forall t \in Row_i} P(X = t)} & , if \delta_I(t) = \hat{i} \\ 0 & , if \delta_I(t) \neq \hat{i} \end{cases} \quad (4.8)$$

其意義為當指定矩陣的行索引 \hat{j} 漏失時，在計算後驗機率時，僅需考慮所收到列索引 \hat{i} 的索引集合 Row_i ，利用此方法來縮小判斷傳輸索引的範圍。我們利用圖 4.2 說明，當 $\hat{I} = 2$ 與 $\hat{J} = \emptyset$ 時，在計算索引後驗機率時，只會考慮第二列的索引集合 $Row_2 = (3, 4, 5)$ 。同理可推，當 $\hat{i} = \emptyset$ 且 $\hat{j} = j = \delta_J(t)$ 的事後機率為

$$P(X = t | \hat{I} = \emptyset, \hat{J} = \hat{j}) = \begin{cases} \frac{P(X = t)}{\sum_{\forall t \in Col_j} P(X = t)} & , if \delta_J(t) = \hat{j} \\ 0 & , if \delta_J(t) \neq \hat{j} \end{cases} \quad (4.9)$$

接下來考慮當兩個通道都接收到時，其事後機率表示成

$$P(X = t | \hat{I} = \delta_I(x), \hat{J} = \delta_J(x)) = \begin{cases} 1 & , if t = x \\ 0 & , else \end{cases} \quad (4.10)$$

其意謂為當接收到兩個敘述時，必定能對應到指定矩陣中當初所傳送的索引 x 。最後考慮當兩敘述都沒接收到時，根據(4.6)與(4.7)可得到

$$P(X = t | \hat{I} = \emptyset, \hat{J} = \emptyset) = P(X = t) \quad (4.11)$$

由於接收端並沒有收到任何敘述，解碼時只能根據索引事前機率來做

為接收端的事後機率。依據上述四種可能的接收情形，最後可將(4.2)

整理成

$$\hat{V}(\hat{i}, \hat{j}) = \begin{cases} w_x & , \text{if } \hat{i} = i, \hat{j} = j \\ \sum_{t \in \text{Row}_i} \frac{w_t \cdot P(X=t)}{\sum_{q \in \text{Row}_i} P(X=q)} & , \text{if } \hat{i} = i, \hat{j} = \emptyset \\ \sum_{t \in \text{Col}_j} \frac{w_m \cdot P(X=t)}{\sum_{q \in \text{Col}_j} P(X=q)} & , \text{if } \hat{i} = \emptyset, \hat{j} = j \\ \bar{W} = \sum_{t=1}^{2^M} w_t \cdot P(X=t) & , \text{if } \hat{i} = \emptyset, \hat{j} = \emptyset \end{cases} \quad (4.12)$$

上式中的第一項代表接收到兩敘述可以對應到指定矩陣索引 x 的碼字向量，第二項則意謂著當只收到列索引 \hat{i} 時，會計算索引集合 Row_i 所對應碼字的期望值。利用圖 4.2 說明，當 $\hat{i} = 2$ 與 $\hat{j} = \emptyset$ 時， $\hat{V}(\hat{i}, \hat{j})$ 可表示成

$$\hat{V}(\hat{i} = 2, \hat{j} = \emptyset) = \frac{w_2 \cdot P(2) + w_3 \cdot P(3) + w_4 \cdot P(4)}{P(2) + P(3) + P(4)} \quad (4.13)$$

至於(4.12)式中第三項的意義，就是計算索引集合 Col_j 所對應碼字的期望值。第四項則代表當兩敘述同時未接收到時，解碼輸出的重建向量為整個碼字的期望值 \bar{W} 。

4.2 多重敘述索引指定設計

在 4.1 節中，我們已經詳述 MDVQ 解碼器的最佳重建向量。在本小節中，我們將利用最佳重建向量進一步設計指定矩陣，特別著重在

多重敘述二元置換演算法(Multiple Description Binary Switching Algorithm, MD-BSA)[13]。在介紹 MD-BSA 之前，我們要先訂立指定矩陣的評分機制。

4.2.1 指定矩陣的評分機制

指定矩陣的設計原則是希望能降低封包漏失所衍生的解碼失真 D ，其定義如下：

$$D = E \left[\left\| w_x - \hat{V}(\hat{i}, \hat{j}) \right\|^2 \right] \quad (4.14)$$

其中最佳重建向量 $\hat{V}(\hat{i}, \hat{j})$ 已在(4.12)描述。將(4.14)展開得到

$$D = \sum_{t=1}^{2^M} P(X=t) \cdot \sum_{\forall i \in U_i} \sum_{\forall j \in U_j} \left\| w_t - \hat{V}(\hat{i}, \hat{j}) \right\| \cdot P(\hat{I}=\hat{i}, \hat{J}=\hat{j} | X=t) \quad (4.15)$$

其中 $U_i = (i, \emptyset)$ 且 $U_j = (\emptyset, j)$ 。定義一個成本函數 $C(w_t)$ ，其表示如下

$$C(w_t) = P(X=t) \cdot \sum_{\forall i \in U_i} \sum_{\forall j \in U_j} \left\| w_t - \hat{V}(\hat{i}, \hat{j}) \right\| \cdot P(\hat{I}=\hat{i}, \hat{J}=\hat{j} | X=t) \quad (4.16)$$

考慮通道間互相獨立特性，可將 $C(w_t)$ 展開成

$$C(w_t) = P(X=t) \sum_{\forall i \in U_i} \sum_{\forall j \in U_j} \left\| w_t - \hat{V}(\hat{i}, \hat{j}) \right\| \cdot P(\hat{I}=\hat{i} | I = \delta_i(t)) \cdot P(\hat{J}=\hat{j} | J = \delta_j(t)) \quad (4.17)$$

利用(4.12)代入(4.17)，經過整理可得

$$C(w_t) = P(X=t) \cdot \left\{ \begin{array}{l} \left\| w_t - \hat{V}(\hat{i}, \emptyset) \right\|^2 \cdot (1-p_i) \cdot p_j \\ + \left\| w_t - \hat{V}(\emptyset, \hat{j}) \right\|^2 \cdot p_i \cdot (1-p_j) \\ + \left\| w_t - \hat{V}(\emptyset, \emptyset) \right\|^2 \cdot p_i \cdot p_j \end{array} \right\} \quad (4.18)$$

其中最後一項 $\left\| w_t - \hat{V}(\emptyset, \emptyset) \right\|^2$ 為一常數與所選擇的指定矩陣無關。假設兩通道的封包漏失率相同， $p = p_i = p_j$ ，將(4.18)代入(4.14)可分成兩部

份並重新整理成

$$D = \underbrace{(1-p) \cdot p \cdot \sum_{t=1}^{2^M} \Delta C(w_t)}_{=\Delta D} + \underbrace{p^2 \cdot \sum_{t=1}^{2^M} P(X=t) \cdot \|w_t - \hat{V}(\emptyset, \emptyset)\|^2}_{\text{positive constant}} \quad (4.19)$$

其中 $\Delta C(w_t)$ 為第 t 個索引的成本值(cost value)，定義如下

$$\Delta C(w_t) = P(X=t) \left\{ \|w_t - \hat{V}(\hat{i}, \emptyset)\|^2 + \|w_t - \hat{V}(\emptyset, \hat{j})\|^2 \right\} \quad (4.20)$$

在(4.19)中，值得注意的是 $\sum_{t=1}^{2^M} P(X=t) \cdot \|w_t - \hat{V}(\emptyset, \emptyset)\|^2$ 為碼字向量的變異數，與所選擇的指定矩陣無關。側失真(side distortion) ΔD 則與其對應的指定矩陣有關，只要降低 ΔD ， D 也會隨著降低。 ΔD 可視為指定矩陣中每一行與每一列索引集合的變異數的相加，也就是說指定矩陣的設計準則為如何使每一行或每一列索引集合的變異數降到最低。而且從(4.19)中也可以發現 ΔD 與封包漏失率 p 無關，但是這情形只會存在於兩通道的封包漏失率相同時。

4.2.2 多重描述二元置換演算法

MD-BSA 的主要目標是找出一個最佳的指定矩陣，以期降低封包漏失所衍生的解碼失真，其運作如下：

步驟一：首先選定一個起始矩陣，利用(4.19)與(4.20)分別計算置換前的失真 $D_{initial}$ 與其每個索引的成本值，選擇其中最高成本值的索引作為要置換的目標。

步驟二：將要置換的索引與矩陣中任一位置對換，置換完畢後，再利用(4.19)計算置換後矩陣的失真 D 。如果 $D < D_{initial}$ 代表置換成功，將這兩位置相互對調後的矩陣取代起始矩陣，回到步驟一重新執行。反之， $D \geq D_{initial}$ 代表該置換失敗，最高成本值的索引會與矩陣內所有位置繼續調換，如果有置換成功的情形，將置換好的矩陣取代起始矩陣再從步驟一出發。當最高成本值的索引與矩陣內所有位置都嘗試置換失敗，則接著執行步驟三。

步驟三：由於最高成本值的索引與矩陣內所有位置都置換失敗，此時選擇將第二高成本值的索引作為要置換的目標，其置換過程如步驟二所示。若此索引與矩陣中內任一位置置換成功，將置換好的矩陣重新從步驟一開始。假設此索引還是置換失敗則選擇第三高成本值的索引做為置換目標，以此類推。

步驟四：直到選擇到最小成本值的索引，而且此索引與矩陣上所有位置都置換失敗，代表此演算法結束，最後所保留的矩陣就是最佳的指定矩陣。

4.3 多重敘述量化與通道解碼之整合

圖 4.3 為此小節的系統傳輸架構。當訊號源 ISF_n 經過量化得到索引 X_n ，利用索引指定矩陣將其轉換成兩個敘述， $I_n = \delta_l(X_n)$ 與 $J_n = \delta_j(X_n)$ ，個別執行通道編碼處理取得 \tilde{I}_n 與 \tilde{J}_n 後再傳輸。在接收端，

根據所接收到的敘述 \hat{I}_n 與 \hat{J}_n 計算其後驗機率，利用 MAP 解碼器得到輸出索引 \hat{X}_n ，最後查表取得輸出訊號 \widehat{ISF}_n 。

有別於 3.3 節中的 SMAP1 演算法是利用現在與過去所接收的訊息以及特定索引在相鄰框架間的相關性來計算其後驗機率，本小節所探討的 MD-SMAP1 則是要同時考慮兩通道現在與過去所接收的訊息。在 4.2 節中，最佳 MDVQ 解碼器由於通道不考慮位元錯誤，所以在接收端若同時接收到兩敘述 \hat{I}_n 與 \hat{J}_n ，勢必可以在指定矩陣上找到其對應的索引。但是如今我們要將考慮位元錯誤，也就是說，如果我們還是個別對單一通道進行解碼，其兩通道個別解碼後的索引，可能對應在指定矩陣上會是個空集合。為了避免這類的情形發生，以下我們會基於(3.22)式的理念，來重新推導多重敘述的訊源控制通道解碼演算法。MD-SMAP1 解碼器的判定原則為

$$\hat{X}_n = \arg \max_{x_n} P(X_n | \hat{I}_n, \hat{J}_n) \quad (4.21)$$

其中兩接收序列為 $\hat{I}_n = (\hat{I}_1, \hat{I}_2, \dots, \hat{I}_n)$ 與 $\hat{J}_n = (\hat{J}_1, \hat{J}_2, \dots, \hat{J}_n)$ 。將 $P(X_n | \hat{I}_n, \hat{J}_n)$ 展開成

$$P(X_n | \hat{I}_n, \hat{J}_n) = C \cdot \sum_{S_{I_n}} \sum_{S_{J_n}} P(X_n, S_{I_n}, S_{J_n}, \hat{I}_n, \hat{J}_n) \quad (4.22)$$

其中 $C = P(\hat{I}_n, \hat{J}_n)$ 為正規化因子， S_{I_n} 與 S_{J_n} 分別為兩通道的位移暫存器狀態，將展開得

$$\begin{aligned}
P(X_n, S_{I_n}, S_{J_n}, \hat{I}_n, \hat{J}_n) &= \sum_{X_{n-1}} \sum_{S_{I_{n-1}}} \sum_{S_{J_{n-1}}} P(X_n, X_{n-1}, S_{I_n}, S_{J_n}, S_{I_{n-1}}, S_{J_{n-1}}, \hat{I}_n, \hat{J}_n) \\
&= \sum_{X_{n-1}} \sum_{S_{I_{n-1}}} \sum_{S_{J_{n-1}}} P(X_{n-1}, S_{I_{n-1}}, S_{J_{n-1}}, \hat{I}_{n-1}, \hat{J}_{n-1}) \\
&\quad \cdot \gamma_{X_n, X_{n-1}}(S_{I_n}, S_{J_n}, S_{I_{n-1}}, S_{J_{n-1}}, \hat{I}_{n-1}, \hat{J}_{n-1})
\end{aligned} \tag{4.23}$$

其中

$$\gamma_{X_n, X_{n-1}}(S_{I_n}, S_{J_n}, S_{I_{n-1}}, S_{J_{n-1}}, \hat{I}_{n-1}, \hat{J}_{n-1}) = P(X_n, S_{I_n}, S_{J_n}, \hat{I}_n, \hat{J}_n | X_{n-1}, S_{I_{n-1}}, S_{J_{n-1}}, \hat{I}_{n-1}, \hat{J}_{n-1})$$

與兩通道的轉移機率與暫存器狀態轉移有關。可進一步可分解成三項

$$\begin{aligned}
&\gamma_{X_n, X_{n-1}}(S_{I_n}, S_{J_n}, S_{I_{n-1}}, S_{J_{n-1}}, \hat{I}_{n-1}, \hat{J}_{n-1}) \\
&= P(\hat{I}_n, \hat{J}_n | X_n, S_{I_n}, S_{J_n}, X_{n-1}, S_{I_{n-1}}, S_{J_{n-1}}, \hat{I}_{n-1}, \hat{J}_{n-1}) \\
&\quad \cdot P(X_n | S_{I_n}, S_{J_n}, X_{n-1}, S_{I_{n-1}}, S_{J_{n-1}}, \hat{I}_{n-1}, \hat{J}_{n-1}) \\
&\quad \cdot P(S_{I_n}, S_{J_n} | X_{n-1}, S_{I_{n-1}}, S_{J_{n-1}}, \hat{I}_{n-1}, \hat{J}_{n-1})
\end{aligned} \tag{4.24}$$

第一項基於兩通道間互相獨立且為無記憶性通道的假設下，可展開成

$$\begin{aligned}
&P(\hat{I}_n, \hat{J}_n | X_n, S_{I_n}, S_{J_n}, X_{n-1}, S_{I_{n-1}}, S_{J_{n-1}}, \hat{I}_{n-1}, \hat{J}_{n-1}) \\
&= P(\hat{I}_n, \hat{J}_n | X_n, S_{I_n}, S_{J_n}) \\
&= P(\hat{I}_n, \hat{J}_n | I_n = \delta_I(X_n), J_n = \delta_J(X_n), S_{I_n}, S_{J_n}) \\
&= P(\hat{I}_n | I_n, S_{I_n}) \cdot P(\hat{J}_n | J_n, S_{J_n})
\end{aligned} \tag{4.25}$$

其中 $P(\hat{I}_n | I_n, S_{I_n})$ 與 $P(\hat{J}_n | J_n, S_{J_n})$ 為通道轉移機率，如(3.19)式所描述。

第二項為相鄰框架間索引的相關性，可簡化成 $P(X_n | X_{n-1})$ 。第三項則

為兩通道在相鄰兩時間點狀態轉移與路徑行進方向，基於兩通道的暫

存器狀態是獨立的假設，可將第三項分解成

$$P(S_{I_n}, S_{J_n} | X_{n-1}, S_{I_{n-1}}, S_{J_{n-1}}, \hat{I}_{n-1}, \hat{J}_{n-1}) = P(S_{I_n} | I_{n-1}, S_{I_{n-1}}) \cdot P(S_{J_n} | J_{n-1}, S_{J_{n-1}}) \tag{4.26}$$

最後整理(4.24)式成為

$$\begin{aligned} \gamma_{X_n, X_{n-1}}(S_{I_n}, S_{J_n}, S_{I_{n-1}}, S_{J_{n-1}}, \hat{I}_{n-1}, \hat{J}_{n-1}) &= P(\hat{I}_n | I_n, S_{I_n}) \cdot P(\hat{J}_n | J_n, S_{J_n}) \\ &\cdot P(X_n | X_{n-1}) \\ &\cdot P(S_{I_n} | I_{n-1}, S_{I_{n-1}}) \cdot P(S_{J_n} | J_{n-1}, S_{J_{n-1}}) \end{aligned} \quad (4.27)$$

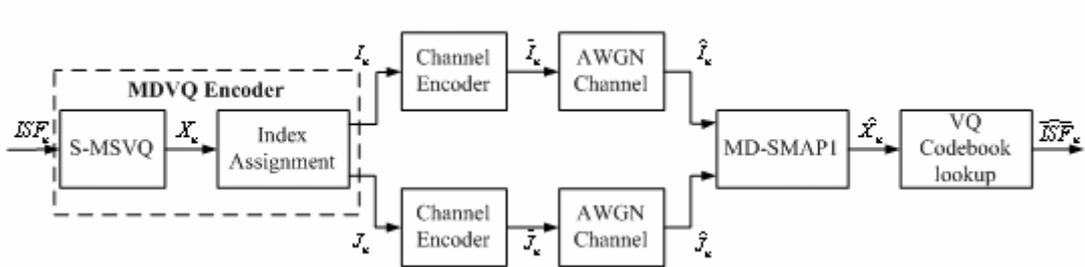


圖 4.3 MD-SMAP1 的傳輸架構



第五章 實驗模擬與結果分析

在 SCCD 的架構中，若在接收端利用訊源的殘餘冗息能有效地提升系統的解碼效能。在 2.4 節中，我們發現 AMR-WB 編碼輸出的量化索引在框架間與框架內存在不同程度的殘餘冗息。在本章的內容將分為三小節來進行實驗之模擬與結果之分析討論。在前兩小節中，我們將模擬第三章中所推導出的解碼演算法，並利用不同程度的索引殘餘冗息在接收端進行解碼效能的評比與分析。在第三小節中，將第四章所推導出的解碼演算法應用於系統模擬，以期驗證演算法之正確性。



5.1 冗息應用於訊源解碼之模擬

5.1.1 系統模擬之步驟說明

在此節之內容中，主要將 3.2 節之訊源解碼演算法進行系統模擬與結果分析。利用大量包含了男聲與女聲的語料經過 AMR-WB 編碼器得到 194881 組 *ISF* 索引，並記錄其個別索引、索引在相鄰框架間與框架內的機率做為事前機率使用。在這些語料內中取出一段 200 秒長度的語音作為訊號源，利用 AMR-WB 編碼器產生出 10000 組 *ISF* 索引，通過二位元相位鍵移 (BPSK) 調變，把每一位元由原來的 {0,1} 映對成

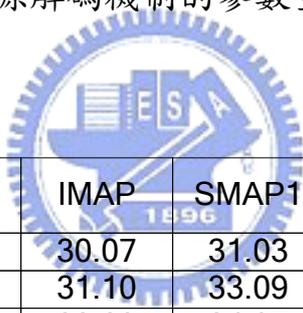
{1,-1}。接著經由通道傳送至接收端，模擬的通道環境使用白色高斯雜訊通道。接收到的訊號先執行硬判斷後，再分別利用在 3.2 節所推導出的 IMAP、SMAP1 及 SMAP2 後驗機率來預估索引 \hat{X} ，並經由量化碼書查表比對重建出訊號 \widehat{ISF} 。其實驗結果如表 5.1、表 5.2、表 5.3 與圖 5.1 所示。

SNR	HD	IMAP	SMAP1	SMAP2	SMAP2*
ISF_1	11.43	12.52	13.52	14.18	13.78
ISF_2	13.04	13.83	15.09	15.58	15.48
ISF_3	14.98	15.22	16.60	17.50	17.47
ISF_4	19.76	19.82	21.01	21.73	21.72
ISF_5	22.24	22.35	23.86	24.58	24.55
ISF_6	24.55	24.58	25.97	26.65	26.63
ISF_7	26.77	27.11	28.68	29.49	27.45
ISF_8	28.09	28.44	29.84	30.60	30.56
ISF_9	29.39	29.51	30.49	31.32	31.29
ISF_10	30.33	30.35	31.57	32.69	32.69
ISF_11	31.42	31.64	32.79	34.05	34.05
ISF_12	32.06	32.61	33.87	34.96	34.96
ISF_13	33.97	34.57	35.44	36.31	36.31
ISF_14	35.03	35.09	35.84	36.53	36.53
ISF_15	37.23	37.25	37.73	38.16	38.16
ISF_16	27.31	27.34	27.61	27.89	27.89

表 5.1 各種訊源解碼機制的參數重建結果(1dB AWGN)

SNR	HD	IMAP	SMAP1	SMAP2	SMAP2*
ISF_1	17.61	18.30	19.06	19.74	19.39
ISF_2	19.03	19.60	21.00	21.67	21.55
ISF_3	21.18	21.34	23.03	23.97	23.95
ISF_4	26.06	26.08	27.53	28.41	28.44
ISF_5	28.77	28.78	30.48	31.27	31.29
ISF_6	30.88	30.92	32.31	33.29	33.25
ISF_7	33.12	33.33	35.02	36.00	35.97
ISF_8	34.26	33.45	35.98	37.03	37.00
ISF_9	35.59	35.70	36.78	37.70	37.66
ISF_10	36.48	36.61	37.74	38.91	38.91
ISF_11	37.62	37.90	39.29	40.33	40.33
ISF_12	38.24	38.69	40.25	41.40	41.40
ISF_13	40.32	40.47	41.81	42.72	42.72
ISF_14	41.31	41.29	42.11	42.70	42.70
ISF_15	43.43	43.47	44.02	44.56	44.56
ISF_16	33.48	33.52	33.82	34.26	34.26

表 5.2 各種訊源解碼機制的參數重建結果(4dB AWGN)



SNR	HD	IMAP	SMAP1	SMAP2	SMAP*
ISF_1	30.06	30.07	31.03	31.83	31.38
ISF_2	31.07	31.10	33.09	33.82	33.56
ISF_3	33.06	33.09	34.97	35.70	35.69
ISF_4	37.72	37.75	39.80	40.24	40.24
ISF_5	41.57	41.57	43.55	44.46	44.46
ISF_6	43.39	43.42	44.86	45.66	45.66
ISF_7	45.76	45.81	47.29	48.13	48.10
ISF_8	47.06	47.03	48.51	49.39	49.38
ISF_9	47.77	47.70	48.91	49.49	49.37
ISF_10	48.60	48.60	49.72	51.38	51.38
ISF_11	49.61	49.61	50.54	52.26	52.26
ISF_12	50.49	50.45	51.66	53.13	53.13
ISF_13	52.24	52.24	53.60	58.84	54.84
ISF_14	53.91	53.91	53.95	55.25	55.25
ISF_15	55.90	55.90	56.34	57.11	57.11
ISF_16	45.49	45.49	45.98	46.21	46.21

表 5.3 各種訊源解碼機制的參數重建結果(7dB AWGN)

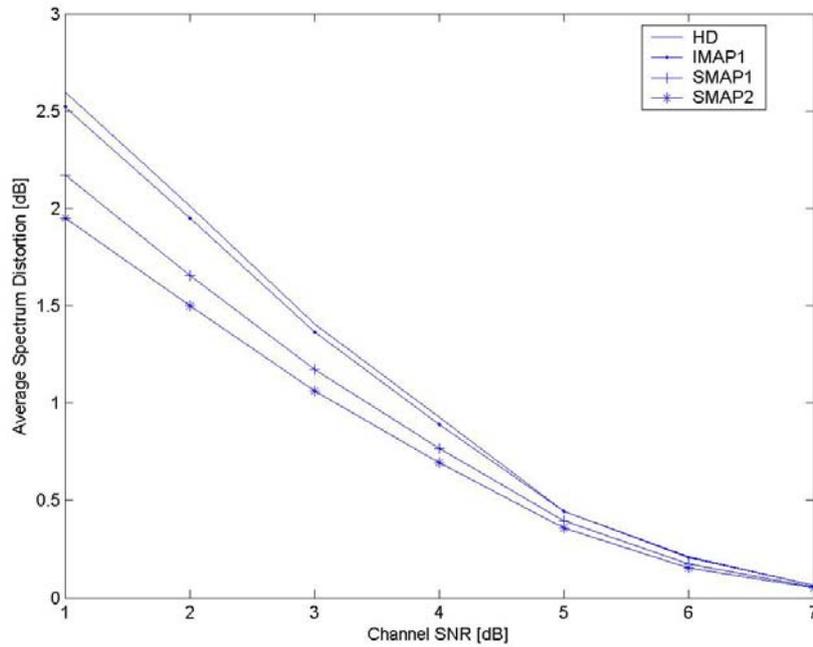


圖 5.1 各種訊源解碼機制的平均頻譜失真結果



5.1.2 結果分析

表 5.1、表 5.2 與表 5.3 中，分別為在不同 AWGN 通道環境下，針對重建的 \widehat{ISF}_n 係數與索引 X_n 利用量化碼書查表得到 \widetilde{ISF}_n 係數之間的訊雜比(Signal-to-Noise Ratio, SNR)作為衡量標準：

$$SNR = 10 \log_{10} \frac{\sum_{i=1}^{10000} \widetilde{ISF}_i^2}{\sum_{i=1}^{10000} (\widetilde{ISF}_i - \widehat{ISF}_i)^2} \quad [\text{dB}] \quad (5.1)$$

表中 ISF_1 代表 ISF 第一組係數。HD(Hard Decision)是指接收到的訊息，解碼時僅執行硬判斷，不做任何後驗機率的運算。從表中可以

發現 IMAP1 利用不均勻的索引分布機率而取得比 HD 更好的解碼效能。SMAP1 則是利用過去與現在的接收訊號，同時考慮相鄰兩框架間的相關性，因此，其效能也比 IMAP 好。SMAP2 除了延續 SMAP1，同時也額外加入了框架內相關性，實驗模擬的結果也顯示出 SMAP2 的解碼效能是最佳的。從實驗模擬中可以發現殘餘冗息確實可增加解碼的可靠性，利用越多的殘餘冗息，解碼的效能也就越佳。此外，我們試著將 SMAP2 做簡化，也就是表中的 SMAP2*，主要是利用 SMAP2 演算法計算 AMR-WB 中第一組與第二組的索引後驗機率，其他的 5 組索引則是利用 SMAP1 演算法。從中我們可以發現，利用此方法所得到結果與 SMAP2 十分接近。這意謂著，相較於 AMR-WB 的其他索引，索引一與索引二在重建訊號上扮演比較重要的角色。這些模擬結果都是針對單一 ISF 係數的解碼效能呈現，若要考量整體的解碼效果，則可利用平均頻譜失真(Spectrum Distortion)。對第 n 個音框而言，其頻譜失真為

$$SD_n = \sqrt{\frac{1}{f_2 - f_1} \cdot \int_{f_1}^{f_2} \left[10 \log_{10}(\tilde{P}_n(f)) - 10 \log_{10}(\hat{P}_n(f)) \right]^2 df} \quad (5.2)$$

其中 $f_1 = 50$ 、 $f_2 = 6400$ 。

$\tilde{P}_n(f) = \frac{1}{|\tilde{A}_n(\exp(j2\pi f / F_s))|^2}$ 與 $\hat{P}_n(f) = \frac{1}{|\hat{A}_n(\exp(j2\pi f / F_s))|^2}$ 為量化

後與重建的功率頻譜密度，其中 $\tilde{A}_n(z)$ 與 $\hat{A}_n(z)$ 為量化後與重建的 LPC

濾波器， $F_s = 12800$ 。其模擬結果如圖 5.1 所示。從圖中也不難發現當利用越多的索引間相關性，其解碼效果也就越好。更值得一提的是，在 AWGN 通道雜訊比較大時，不同訊源解碼機制的優劣差異會更明顯呈現。

5.2 訊源控制通道解碼之模擬

5.2.1 系統模擬之步驟說明

本小節主要是將 3.3 節之解碼演算法進行系統模擬與結果分析，同時採用迴旋碼(Convolutional Code, cc)的通道編碼器，如圖 3.3 所示。所用的事前機率與訊號源與 5.1 節模擬一樣，不同的是將訊號源經過量化處理所得的索引通過迴旋編碼器再進行 BPSK 調變，然後送進 AWGN 通道。在接收端分別利用 3.3 節所推導出來的 IMAP 與 SMAP1 計算其後驗機率，所預估出來的索引 \hat{X} 再比對量化碼書可得 \widehat{ISF} 。實驗結果如圖 5.2 所示。

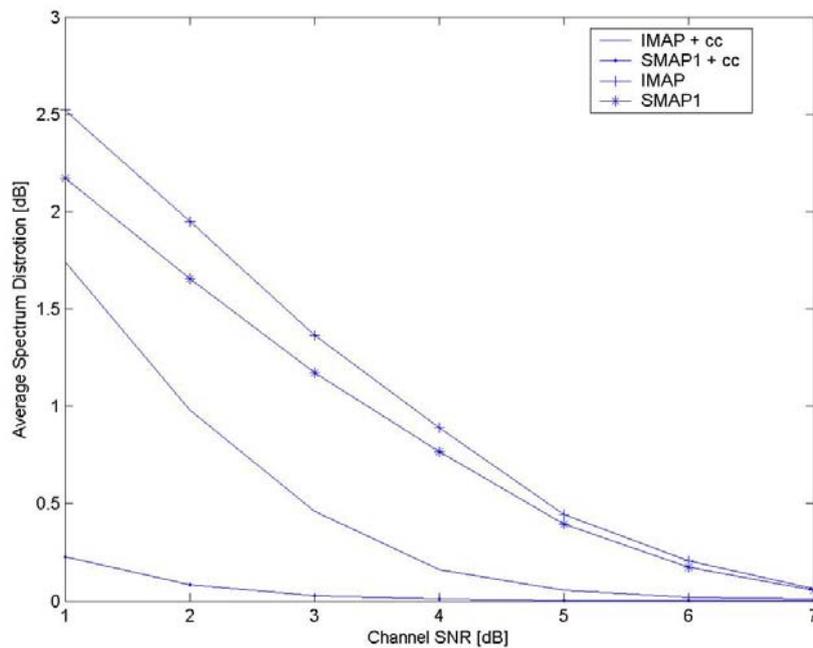


圖 5.2 訊源控制通道解碼器平均頻譜失真結果



5.2.2 結果分析

在圖 5.2 中，由於 SMAP1 比 IMAP 多利用了相鄰框架間索引的殘餘冗息，從圖中可以發現 SMAP1 的解碼效能會比 IMAP 來的好。在前一小節中的模擬顯示出在通道較好的環境下，訊源解碼所能提升的效能並不明顯。由於本節加入的通道編碼處理，不僅在通道環境不好時，SMAP1 比起 IMAP 的解碼效能有明顯上升外；在通道環境好時，解碼效能也有不錯提升。除此之外，與 5.1 節的訊源解碼實驗比較，不難發現加入迴旋碼能有效提昇其解碼效，而所付出的代價是較高的位元傳輸率。

5.3 整合多重敘述與訊源控制通道解碼之實驗

5.3.1 系統模擬之步驟說明

圖 4.3 為本小節系統模擬之架構。其事前機率與訊號源與前面的實驗相同。當訊號源經過 S-MSVQ 後，會產生出七組索引，將七組索引 X_n 分別利用其對應的索引指定矩陣分成兩敘述 I_n 與 J_n 。在這裡我們將模擬兩種索引指定矩陣對於重建訊號的影響，第一種為亂數索引指定矩陣，其產生方法為在 1000 個任意產生的指定矩陣中，選擇最小失真 D 的矩陣。第二種矩陣則是將亂數索引指定矩陣視為起始矩陣，利用 4.2 節所提的 MD-BSA 方法，求得最佳的指定矩陣，而各索引所配置的指定矩陣大小，如表 5.4 所示。輸出兩敘述分別利用迴旋編碼器後送入兩個獨立的 AWGN 通道。在接收端接收到兩通道過去與現在的訊息 \hat{I}_n 與 \hat{J}_n ，利用 (4.22) 的最大後驗機率解碼器預估出的索引 \hat{X}_n ，最後查詢量化碼書重建出訊號 \widehat{ISF}_n 。其實驗結果如圖 5.3 所示。



INDEX	M(bits)	N
1	8	32
2	8	32
3	6	16
4	7	16
5	7	16
6	5	8
7	5	8

表 5.4 各索引所配置的指定矩陣

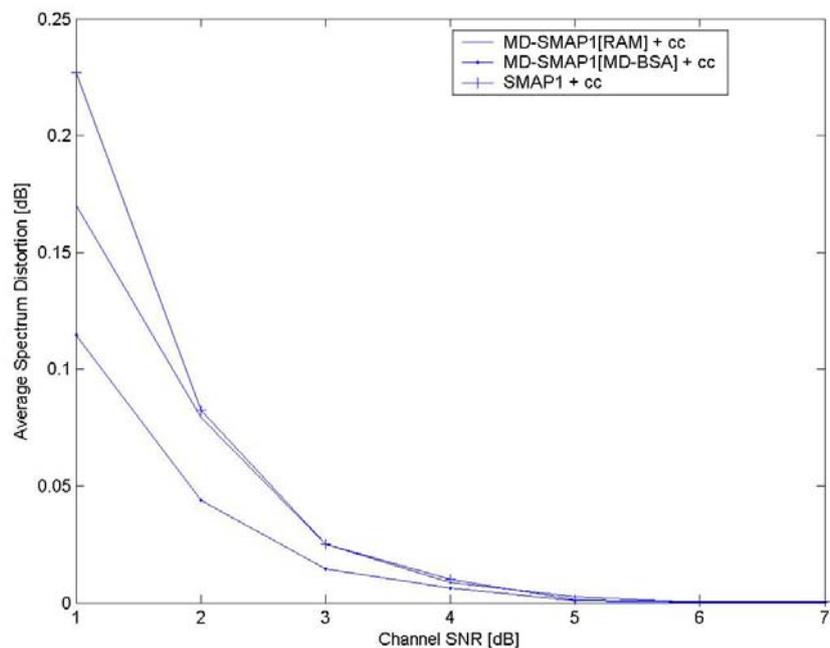


圖 5.3 多重敘述通道解碼的平均頻譜失真結果



5.3.2 結果分析

RAM 表示此系統模擬的索引指定矩陣是從 1000 組任一產生矩陣中找出最小失真的矩陣，MD-BSA 代表索引指定矩陣是有利用 MD-BSA 設計。相較於 RAM，我們可以發現 MD-BSA 所設計的索引指定矩陣能有效提昇其解碼效能。

與前一小節實驗模擬中的 SMAP1 相比，不論是使用 RAM 產生或是 MD-BSA 所設計的索引指定矩陣，多重敘述的編碼模式都能進一步減少解碼的頻譜失真。但特別強調的是，多重敘述編碼模式會增加額外傳輸位元率，如表 5.5 所示。

Decoder	bits per frame
SMAP1	46
SMAP1 + cc	92
MD-SMAP1 + cc	112

表 5.5 不同解碼機制的位元傳輸數目



第六章 結論與未來展望

本論文提出一個基於軟性輸出通道解碼演算法的錯誤隱匿機制，主要是整合訊號源在經由訊源編碼後索引間的殘餘冗息以及多重敘述向量量化，以提昇在接收端通道解碼的錯誤更正能力。首先分析訊號源在經由 AMR-WB 編碼器量化後，紀錄並分析其不同層級的索引間所殘留的訊源冗息。實驗模擬證實利用越多的索引間的殘餘冗息，在接收端越能提昇解碼效能，尤其當通道環境劣時改善效能格外明顯。最後，並將所記錄的事前訊息應用於所提出的 SMAP1 及 MD-SMAP1 的演算法中，對於解碼效能都能提供很大的幫助。

在第五章的實驗模擬證實，結合訊號源的殘餘冗息與多重敘述向量量化，確實大幅提昇解碼器的錯誤更正的能力。但目前模擬中所使用的通道環境，僅考慮無記憶性的可加性高斯白雜訊通道，而這樣的通道假設並不符合真實通道環境觀察到的叢發現位元錯誤。因此在未來，可以針對通道模擬環境加以改善，並且考慮具有記憶性的通道環境，例如 Glibert 通道，。若能通道記憶特性整合於解碼過程，相信亦能提昇寬頻語音系統的強健性。除此之外，在本論文中所推導的後驗機率演算法，當索引位元數目增大，其計算量更為龐大且複雜，

在解碼的效率上將有所延遲，其即時製作的具體實現有賴於演算法的進一步簡化。還有第四章所提出的編碼架構，其輸出位元數目約為原本的 2.4 倍，應考慮設計的更低位元率的編碼器。



參考文獻

- [1] C.E. Shannon, "A Mathematical Theory of Communication," *Bell Syst. Tech. J.*, vol.27, pp.379-423, pp.623-656, 1948.
- [2] T. Fingscheidt and P. Vary, "Softbit speech decoding: A New Approach to Error Concealment," *IEEE Trans. Speech and Audio Processing*, vol.9, no 3, pp 240-251, March 2001.
- [3] A.M. Peinado, V. Sanchez, J.L. Perez-Cordoba, and A. Torre, "HMM-based channel error mitigation and its application to distributed speech recognition," *Speech Communication*, vol.41, pp. 549-531, 2003.
- [4] H.U. Reinhold, and I. Valentin, "Soft features for improved distributed speech recognition over wireless networks," *8th International Conference on spoken Language Processing*, pp.2125-2128, Jeju Island, Korea, Oct 2004.
- [5] J. Hagenauer, "Source-Controlled Channel Decoding," *IEEE Trans. Commun.*, vol.43, pp.2449-2457, Sep. 1995.
- [6] L. Hanzo, F. C. A. Somerville, and J. P. Woodard, *Voice Compression and Communications*. New York: Wiley, 2001.
- [7] B. Bessette, R. Salami, R. Lefebvre, M. Jelinek, J. Rotola-Pukkila, J. Mikkola, H. Vainio, and K. Jarvinen, "The Adaptive Multirate Wideband Speech Codec (AMR-WB)." *IEEE Transactions on speech and audio processing*, vol.10, no.8, pp.620-636, Nov. 2002.
- [8] S. Lin and D. J. Costello, Jr., *Error Control Coding*. Englewood Cliffs, NJ: Prentice-Hall, 2004.

[9]K. Sayood and J.C. Borckenhagen, "Use of residual redundancy in the design of joint source/channel coders," *IEEE Trans. Commun.*, vol. 39, No. 6, pp. 838-846, June 1991.

[10]V. Cuperman, F.H. Liu, and P. Ho, "Robust vector quantization for noisy channel using soft decision and sequential decoding," *Eur. Trans. Telecomm.*, vol.5, no.5, pp.7-18, Sept 1994.

[11]K.P. Ho, "Soft-decoding vector quantizer using reliability information from turbo-codes," *IEEE Commun. Lett.*, vol.3, pp.208-210, July 1999.

[12]F. Lahouti and A. K. Khandani, "Soft Reconstruction of Speech in the Presence of Noise and Packet Loss," Technical Report 2003-4, Department of E&CE, University of Waterloo, 2004, available at www.cst.uwaterloo.ca, Submitted to *IEEE Trans. Inform. Theory*, Revised Expected publication.

[13]N. Gortz, and P. Leelapornchai, "Optimization of the index assignment for multiple description vector quantizers," *IEEE Trans. Commun.*, vol.51, no.3, pp.336-340, Mar. 2003.