

國立交通大學

電信工程學系

碩士論文



通道匹配的多重描述量化索引指定之研究
Channel-Matched Index Assignment Design
In Multiple Description Quantization

研究生：曾啟翔

指導教授：張文輝博士

中華民國九十七年六月

通道匹配的多重描述量化索引指定之研究

**Channel-Matched Index Assignment Design In
Multiple Description Quantization**

研究生：曾啟翔

Student：Chi-Shung Tseng

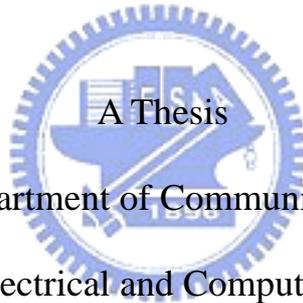
指導教授：張文輝

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 2008

Hsainchu, Taiwan, Republic of China

中華民國 九十七 年 六 月

通道匹配的多重描述量化索引指定之研究

學生：曾啟翔

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

本論文探討多重描述量化索引之最佳化指定，進一步應用於適應性多重速率寬頻語音編碼技術。多重描述量化技術常用來克服網路封包漏失的問題，本論文則推廣應用於位元錯誤及 AWGN 通道，在系統的傳輸端，多重描述量化器將語音參數量化，並根據索引指定函數產生多個描述，再通過互相獨立的通道傳輸。傳統的索引指定是利用二元置換演算法，在指定矩陣中兩兩交換索引位置直到最小失真度為止。為了要降低其運算複雜度以及增加其強健性，索引指定將分為位置指定及位元對應兩階段進行。前者決定量化索引集合在指定矩陣的位置，使其對應的列索引及行索引可合併成為一合法的線性區段碼字，此碼字結構有助於解碼端的錯誤偵測與更正。至於後續的位元對應，則是在規劃好的位置集合執行二元置換演算法，找尋匹配於當時通道環境之最佳索引指定。實驗結果顯示不同通道環境下，本論文所提出的索引指定機制具有更強健的性能表現。

Channel-Matched Index Assignment Design In Multiple Description Quantization

Student: Chi-Shung Tseng Advisor: Dr. Wen-Whei Chang

Department of Communication Engineering

National Chiao Tung University

Abstract

This study addresses the channel-matched index assignment design for multiple description quantization and its application to adaptive multi-rate wideband (AMR-WB) speech coding. In the multiple description quantizer, speech parameters are quantized and mapped to two or more descriptions according to a predefined index assignment. Conventional binary switching algorithm (BSA) aims to switch a pair of codevectors recursively until the expected distortion is minimized. In order to reduce its computational complexity and enhance robustness, we divide the index assignment into two steps: location assignment and bit mapping. The optimization criterion and a practical approach are presented for index assignment optimization under bit error and AWGN channels. A new location assignment algorithm is also proposed that ensures the combined row and column indexes form a logical linear block codeword. Simulation results show that the proposed index assignment algorithms achieve significant performance gain in channel robustness.

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 通道匹配的索引指定設計.....	4
2.1 系統架構.....	4
2.1.1 多重描述向量量化.....	4
2.1.2 多重描述量化解碼器.....	6
2.2 通道匹配的二元置換演算法.....	7
2.3 AMR-WB 編碼系統的相關應用.....	11
2.3.1 AMR-WB 概述.....	11

2.3.2	系統傳輸規格.....	15
第三章	基於位元錯誤通道的索引指定.....	17
3.1	位置指定機制.....	17
3.1.1	線性區段碼.....	17
3.1.2	特殊群組選擇.....	18
3.1.3	位置指定演算法.....	23
3.2	位元對應機制.....	25
3.2.1	位元對應.....	25
3.2.2	通道匹配的二元置換演算法.....	26
3.3	基於位元錯誤通道的多重描述向量量化系統.....	28
第四章	基於 AWGN 通道的索引指定.....	30
4.1	通道轉移機率.....	30
4.2	通道匹配的二元置換演算法.....	34
4.3	基於 AWGN 通道的多重描述向量量化系統.....	36
4.3.1	系統架構.....	36
4.3.2	trellis 解碼器.....	37

第五章	實驗模擬與結果分析	44
5.1	通道匹配二元置換的實驗	44
5.1.1	系統模擬之步驟說明	44
5.1.2	結果分析	45
5.2	位置指定之下的多重描述量化系統實驗	47
5.2.1	系統模擬之步驟說明	47
5.2.2	結果分析	48
5.3	AMR-WB 應用的實驗	49
5.3.1	系統模擬之步驟說明	49
5.3.2	結果分析	50
第六章	結論與未來展望	53
	參考文獻	55



圖目錄

圖 2.1	多重描述傳輸系統架構	6
圖 2.2	指定矩陣的配置範例	6
圖 2.3	二元置換演算法流程圖	10
圖 2.4	AMR-WB ACELP 編碼器的方塊圖	13
圖 2.5	AMR-WB ACELP 解碼器的方塊圖	14
圖 2.6	ISF 係數的分次多階向量量化處理	15
圖 3.1	特殊群組選擇流程圖	19
圖 3.2	證明線性區段碼示意圖	20
圖 3.3	$(n, k)=(4, 3)$ 的位置指定 $d=2$ 的例子	25
圖 3.4	二元置換過程示意圖	26
圖 3.5	多重路徑量化系統架構圖(位元錯誤)	28
圖 4.1	AWGN 通道的多重描述量化傳輸系統	31
圖 4.2	多重路徑量化系統架構圖(AWGN)	35
圖 4.3	柵狀解碼圖	41
圖 5.1	二元置換演算法置換過程訊雜比	45
圖 5.2	MDC1、MDC2 及 MDC3 訊雜比比較圖	48
圖 5.3	AMR-WB 在不同索引指定的頻譜失真	50
圖 5.4	AMR-WB 在不同索引指定的音質評估	51

表目錄

表 2.1	AMR-WB 不同編碼模式的位元配置.....	15
表 4.1	狀態分析表	38
表 5.1	位元錯誤率與訊雜比的對應關係	46
表 5.2	ISF 係數前 3 個索引的位元分配.....	48



第一章

緒論

1.1 研究動機與方向

全球網際網路的發展以及服務市場的需求，通訊科技與日俱增，主要以多媒體通訊的應用服務為主軸，尤其以語音 (speech)、音訊 (audio) 或是影像 (video) 為主要服務內容。不論在有線或無線的傳輸過程中，不同傳輸媒介將會遭遇各式各樣的錯誤通道，因而造成資訊的錯誤或遺失。較常用的通道模型有位元錯誤(bit error)通道、可加性高斯白雜訊通(addition white Gaussian noise, AWGN)通道、網路封包漏失(packet loss)通道以及衰弱(fading)通道。

針對多媒體通訊中網路封包漏失的問題，以及因編碼壓縮處理所衍生的量化失真，目前已有許多相關技術來解決上述的問題。例如：多重描述編碼技術(Multiple Description Coding, MDC)[1]以及前向錯誤控制技術(Forward Error Control, FEC)[8]等。多重描述量化技術可視為傳統的量化器(quantizer)與索引指定(index assignment)兩工作單元的整合，先將訊號源作純量或向量量化取得量化索引(quantizer index)，再將其藉由索引指定函數取得兩個或多個描述去表示。分別於兩個或多個獨立通道之下傳輸此描述資訊，即使有通道收到錯誤資訊或是發生封包漏失的情況，解碼端在合理的範圍之內

依然能重建一定程度的品質。隨者接收端收到的正確描述來源數目增多，重建的品質也相對地變好。至於前向錯誤控制技術，則是對訊源編碼器的輸出位元序列附加一些保護位元，最常見的是線性區段碼 (linear block code) 以及迴旋碼 (convolutional code)。在通過雜訊干擾的通道環境時，這些保護位元可以在通道解碼過程中修正通道錯誤以提升系統的強健性能。

在多重描述量化的相關研究中，索引指定的設計扮演關鍵且重要的角色，若能於編碼端設計最佳的指定矩陣 (assignment matrix)，有助提升解碼端的強健效果，於是指定矩陣中找尋合適的位置置放量化索引成為一個重要的課題。本論文將索引指定函數分為兩部分：位置指定 (location assignment) 及位元對應 (bit mapping)。前者預先在指定矩陣中決定一組位置，使其對應的列索引及行索引合併成為一合法線性區段碼字，而後者則利用二元置換演算法 (binary switching algorithm, BSA) 找尋通道匹配的索引排列方式。

本論文所提出的是通道匹配的索引指定機制，其關鍵在於針對位元錯誤通道以及可加性高斯白雜訊通道個別設計。在有限的頻寬下，為了不增加額外的保護位元，同時於索引指定引入了線性區段碼的概念，以期各通道傳輸的描述彼此結合成為一個合法碼字，進而更正一定程度的位元錯誤。此外，由於寬頻語音編碼已被視為下一代語音編

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

第二章將先介紹多重描述量化系統的基本架構，並針對索引指定函數做分析，使用二元置換演算法找到最佳的排列方式。第三章開始，我們從線性區段碼的概念出發，整合區段碼以及索引指定，以期對抗位元錯誤的通道。第四章則是進一步推廣運用到可加性高斯白雜訊通道下。第五章則為實驗模擬，驗證本論文所提出的各種多重描述量化系統，並作效能評估與比較。第六章則為結論與未來展望。



第二章

通道匹配的索引指定設計

多重描述向量量化 (Multiple Description Vector Quantization, MDVQ) 主要是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，以期對抗通道雜訊對解碼的影響。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。目前大部分多重描述量化的研究，主要用來克服封包漏失問題，本文所考量的通道環境則是位元錯誤通道以及常見的可加性高斯白雜訊 (AWGN) 通道。在接收端，我們收到經過通道雜訊干擾的兩個索引值，即使我們不能確定此索引值正確與否，透過 MDVQ 解碼器仍然可以重建一相當程度的訊號品質。多重描述量化的相關研究中，指定矩陣扮演非常關鍵的角色，如何尋找匹配於不同通道環境的指定矩陣，以及將此技術如何運用在 AMR-WB 上，會在接下來的章節一一介紹。首先先探討多重描述量化的系統架構。

2.1 系統架構

2.1.1 多重描述向量量化

多重描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 2.1 為其傳輸架構。其編碼模式可視為向量量化和索引指定(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 $B = (b_1, b_2, \dots, b_{2^M})$ ，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描述 $I = \delta_i(X)$ 和 $J = \delta_j(X)$ 。假設描述 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 2.2 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。索引指定研究的關鍵課題，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，以期能緩和通道錯誤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，我們將在 2.2 節進一步介紹。

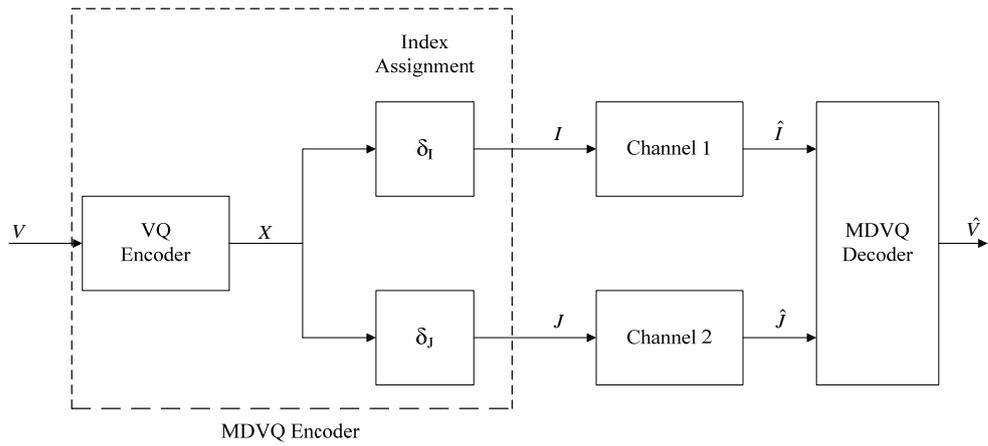


圖 2.1 多重描述傳輸系統架構

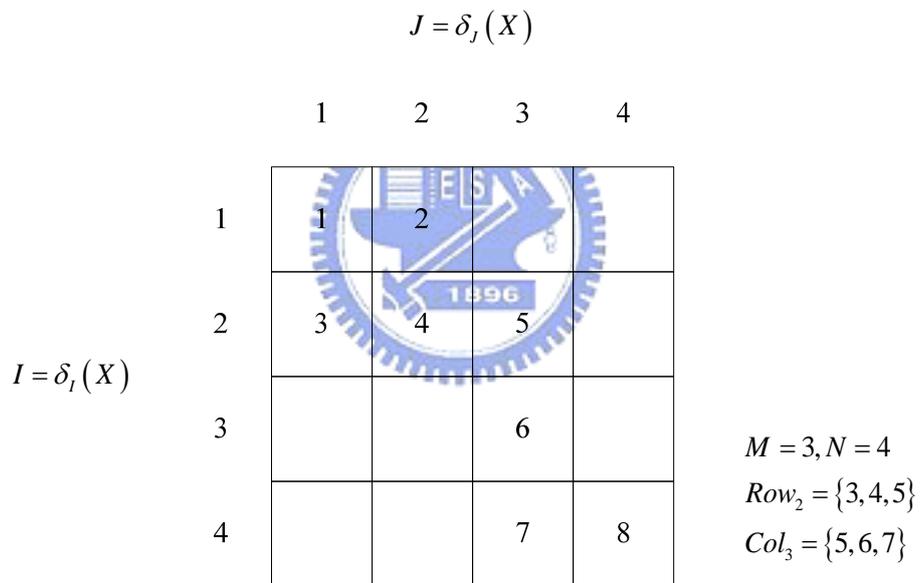


圖 2.2 指定矩陣的配置範例

2.1.2 多重描述量化解碼器

我們根據收到的列索引以及行索引，計算出所有可能判定結果的後驗機率(a posteriori probability)，再依此機率值決定最後結果。

首先我們先定義此後驗機率：

$$p(x|\hat{i}, \hat{j}) = cp(x)p(\hat{i}, \hat{j}|x) \quad (2.1)$$

$$p(\hat{i}, \hat{j}|x) = p(\hat{i}, \hat{j}|x_i, x_j) = p(\hat{i}|x_i)p(\hat{j}|x_j) \quad (2.2)$$

其中 c 為 $p(\hat{i}, \hat{j})^{-1}$ ， $\hat{i}, \hat{j} \in \{1, 2, \dots, N\}$ 分別為列索引 i 以及行索引 j 的接收值，而 x_i, x_j 為特定索引 x 在指定矩陣中對應的列索引以及行索引。至於不同索引的事前機率 $p(x)$ 則是經過大量訓練語料產生出來的平均出現機率。若針對兩個通道的索引做漢明距離並分別定義為 h_i 以及 h_j ，我們能求出：

$$p(\hat{i}, \hat{j}|x) = p(\hat{i}|x_i)p(\hat{j}|x_j) = (1 - BER_i)^{(1-h_i)} BER_i^{h_i} (1 - BER_j)^{(1-h_j)} BER_j^{h_j}$$

其中 BER_i, BER_j 分別為兩個獨立傳輸通道的位元錯誤率。配合事前大量訓練所得的範本碼書，當收到列索引以及行索引之後，計算最佳輸出向量：

$$\tilde{X}(\hat{i}, \hat{j}) = \sum_{x=1}^{2^M} b_x p(x|\hat{i}, \hat{j}) \quad (2.3)$$

其中 b_x 為事前訓練所得範本碼字書中的第 x 個碼字向量。

2.2 通道匹配的二元置換演算法

延續 2.1.1 節所衍生出的問題，找到一個最佳化的指定矩陣是一個 NP 複雜問題，全面性的搜尋並不可行。因此我們嘗試運用二元置換演算法(Binary Switching Algorithm)，來簡化最佳的指定矩陣設

計。在置換的過程中，每個不同的指定矩陣具有不同的失真度，演算法的目的就是盡可能尋找到一個能讓失真度最小的指定矩陣，以期能提升系統效能。接下來定義指定矩陣的失真度：

$$D = \sum_{x=1}^{2^M} p(x) \sum_{\hat{i}=1}^N \sum_{\hat{j}=1}^N \|b_x - \tilde{X}(\hat{i}, \hat{j})\|^2 p(\hat{i}, \hat{j} | x) \quad (2.4)$$

其中 $p(\hat{i}, \hat{j} | x)$ 及 $\tilde{X}(\hat{i}, \hat{j})$ 分別按照 2.1.2 節的方法計算出。

二元置換演算法的一個重要參數就是成本值(cost value)，據以決定置換的方向，及置換的順序等等。成本函數(cost function)可以從失真度的公式中看出端倪，定義索引 x 的成本函數為：

$$C(x) = p(x) \sum_{\hat{i}=1}^N \sum_{\hat{j}=1}^N \|b_x - \tilde{X}(\hat{i}, \hat{j})\|^2 p(\hat{i}, \hat{j} | x) \quad (2.5)$$

成本函數所計算出的數值我們稱成本值，其和失真度的關係為：

$$D = \sum_{x=1}^{2^M} C(x) \quad (2.6)$$

有了失真度、成本函數以及成本值的概念之後，開始探討通道匹配二元置換演算法的運作如下：

步驟一：首先選定一個起始矩陣，利用(2.4)與(2.5)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(2.3)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其餘位置繼續調換，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。

從(2.4)公式，不難發現在不同通道環境之下，所置換出的指定矩陣皆不相同，主要因為 $p(\hat{i}, \hat{j} | x)$ 在不同的通道環境其位元錯誤率都會跟著變化。根據不同的通道環境所得出的指定矩陣，就是最匹配於該環境的指定矩陣，而此演算法我們命名為通道匹配二元置換演算法，其流程圖如圖 2.3 所示：

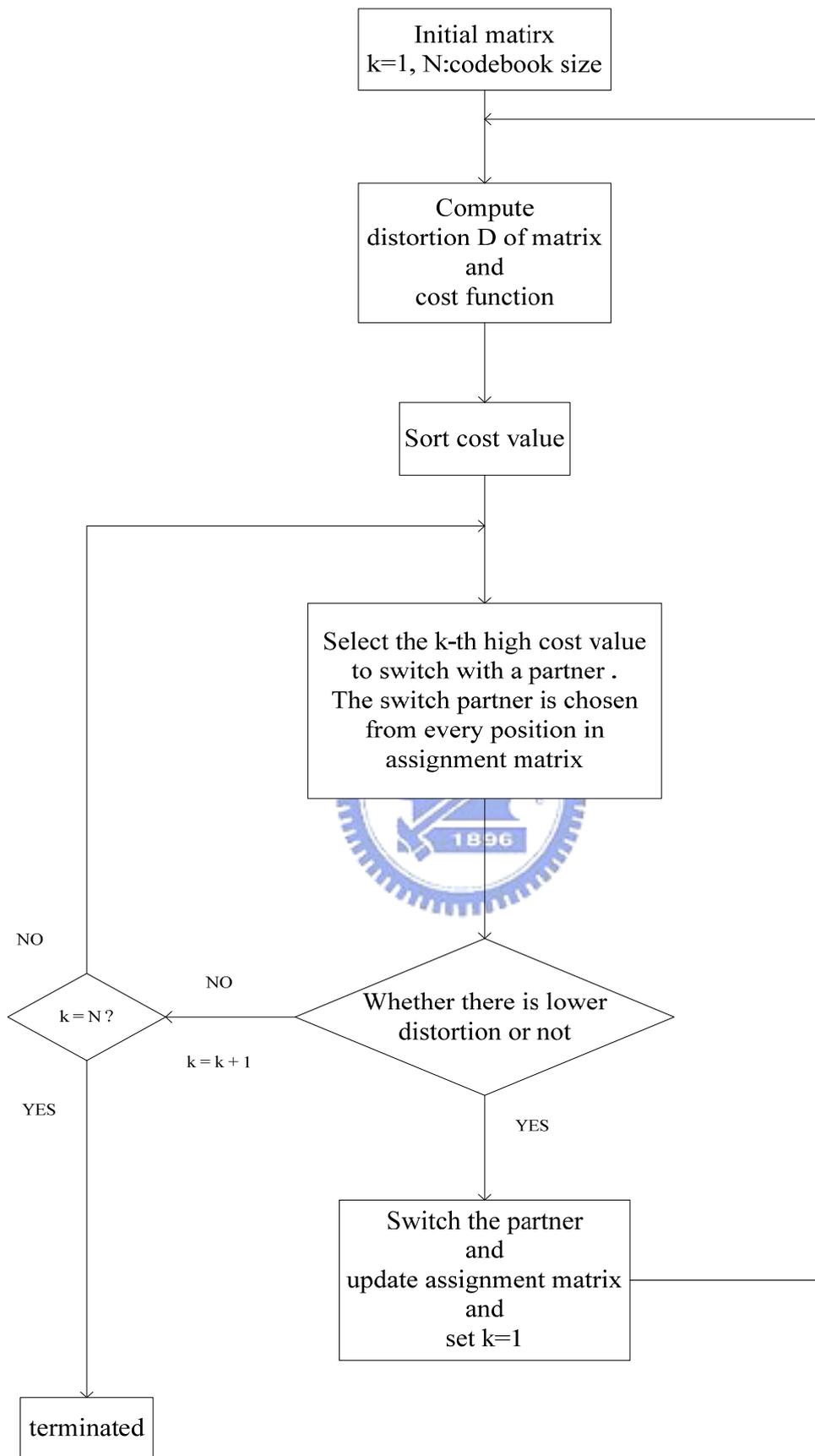


圖 2.3 二元置換演算法流程圖

2.3 AMR-WB 編碼系統的相關應用

2.3.1 AMR-WB 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[9]。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[10]。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 2.4 及圖 2.5，

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), 在 20 毫秒的音框為單位, 採用 12.8 kHz 取樣頻率來處理。在每一個語音訊框內, 分析得到線性預測係數 (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, 先執行線性預測分析取得 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 2.6 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的索引值。

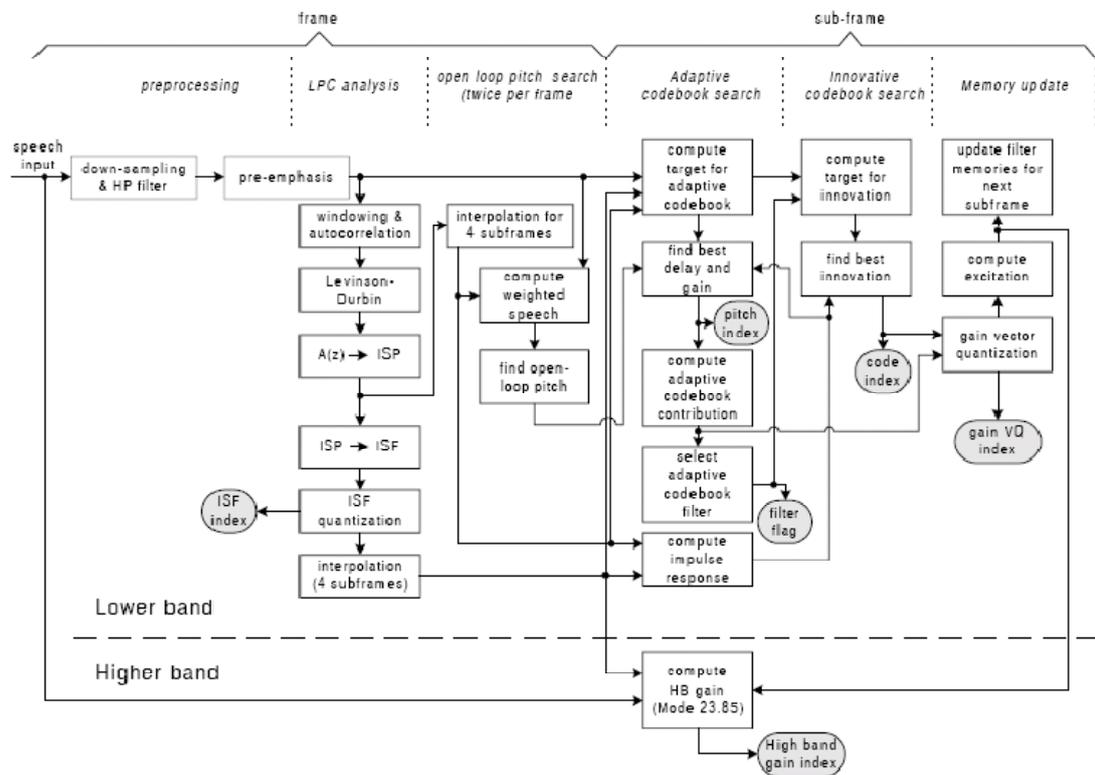


圖 2.4 AMR-WB ACELP 編碼器的方塊圖

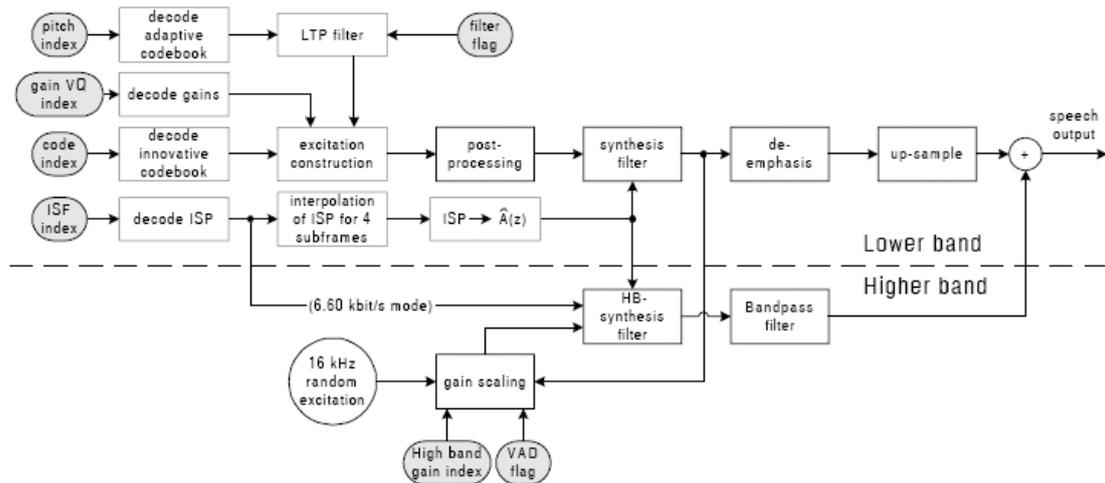


圖 2.5 AMR-WB ACELP 解碼器的方塊圖

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

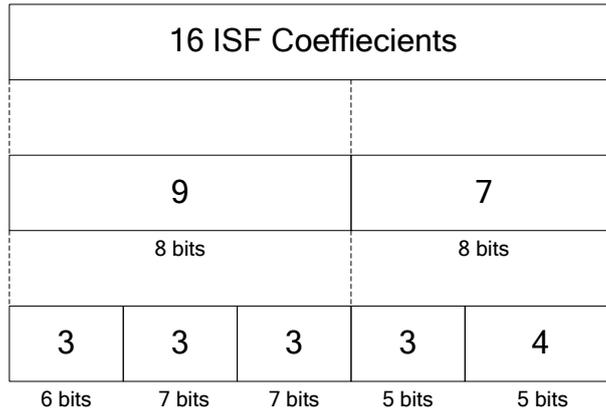


圖 2.6 ISF 係數的分次多階向量量化處理

2.3.2 系統傳輸規格

每一個音框經過 AMR-WB 編碼處理之後，會產生出 16 個線性預測係數，再據以轉換為成量化失真靈敏度較低的導納頻譜頻率 (ISF) 係數。配合事先大量訓練好的碼書，每個音框可以量化輸出 7 個索引值：ID1、ID2、ID3、ID4、ID5、ID6、ID7，分別用 8 bits、8 bits、6 bits、7 bits、7 bits、5 bits、5 bits 量化，如圖 2.6 所示。其中第一階段所量化輸出的兩個索引值 ID1、ID2，以及第二階段以 6 位元作量化的索引值 ID3，在本論文中皆做多重描述向量量化的處理，分別在兩個通道傳輸其在指定矩陣中所對應的座標值。ID1 以及 ID2 分別在兩個通道傳輸以 6 位元作量化的列索引值以及行索引值，而 ID3 則是傳輸以 5 位元作量化的列索引值以及行索引值。其餘的 4 個

索引值並沒有做多重描述向量量化的，ID4 以及 ID6 經由第一個通道做傳輸，ID5 以及 ID7 經由另一個通道做傳輸，若有一通道資訊發生錯誤則由另一通道做補償的動作。舉例而言，如果在傳輸過程中接受端發現 ID4 發生錯誤則由 ID5 做補償動作，若同時都發生錯誤則會運用前一時刻點的資訊做補償的動作。依此類推，ID6 以及 ID7 做相同的補償機制。



第三章

基於位元錯誤通道的索引指定

在此之前所探討的索引指定都是針對特定通道環境而設計，隨機產生一千種指定矩陣，藉由前一章所提供的數學公式，我們可以計算出每個指定矩陣相對應的失真度，從中選擇一個擁有最小失真度當作二元置換演算法的起始矩陣，經過多次的疊代置換找到一個通道匹配的指定矩陣。這章在某些環境與操作之下，我們提供了一種新的設計方法，不但擁有前一章所有的設計特性與優點，在索引指定的同時引入了線性區段碼的概念。利用線性區段碼所對應的查核矩陣，能更正一定程度的位元錯誤，達到提升系統效能的目的。索引指定的工作流程包括位置指定(location assignment)與位元對應(bit mapping)兩個步驟，前者是先在指定矩陣之中找尋適當的位置集合，決定個別位置的列索引以及行索引，使其合併可以成為一個合法的碼字。至於後續的位元對應，則是決定不同位置所應置放最合適的量化索引。

3.1 位置指定機制

3.1.1 線性區段碼

就我們對線性區段碼(linear block code)的一般認識而言，一

個 (n, k) 碼代表的是輸入訊號 u 為一個 k 位元的向量，經過產生矩陣 (generator matrix) G 的編碼處理，而得到一個 n 位元的合法碼字 x 輸出，其對應關係為 $x = uG$ 。在接收端的解碼部分，可以透過產生矩陣 G 得到查核矩陣 (parity-check matrix) H ，利用關係式 $s = xH^T$ 計算出接收資訊的徵狀 (syndrome)。藉由各種不同型式的徵狀，我們可以做一定程度的錯誤位元更正。至於不同的線性區段碼所能更正的錯誤位元數，我們可以從它的最小距離值看出端倪。最小距離值是定義在所有碼字裡最小的漢明距離值，而實際上最小距離值和產生矩陣有基本的關係存在，可以透過一些計算得到。任何一種線性區段碼都有對應的最小距離值，我們根據這最小距離值可以計算出每種線性區段碼可以更正的位元數，其兩者之間的對應關係為下列公式：

$$t = \frac{1}{2}(d_{\min} - 1)$$

其中 t 為可更正的錯誤位元數，而 d_{\min} 為最小距離值。線性區段碼更詳細的介紹可以參考[8]。

3.1.2 特殊群組選擇

首先，我們探討一個擁有特殊屬性的群組，並且驗證此特性與線性區段碼有關，此特性也會運用在後續的系統設計之上。一個 n 位

元數的集合總共有 2^n 個不同組合，我們從中挑選出的特殊群組需滿足一條件：此群組所屬的任意兩個位元組合之間的漢明距離必須不小於設定的門檻定值，而此定值可由使用者自行定義或設計。其群組選擇演算法流程如圖(3.1)所示。

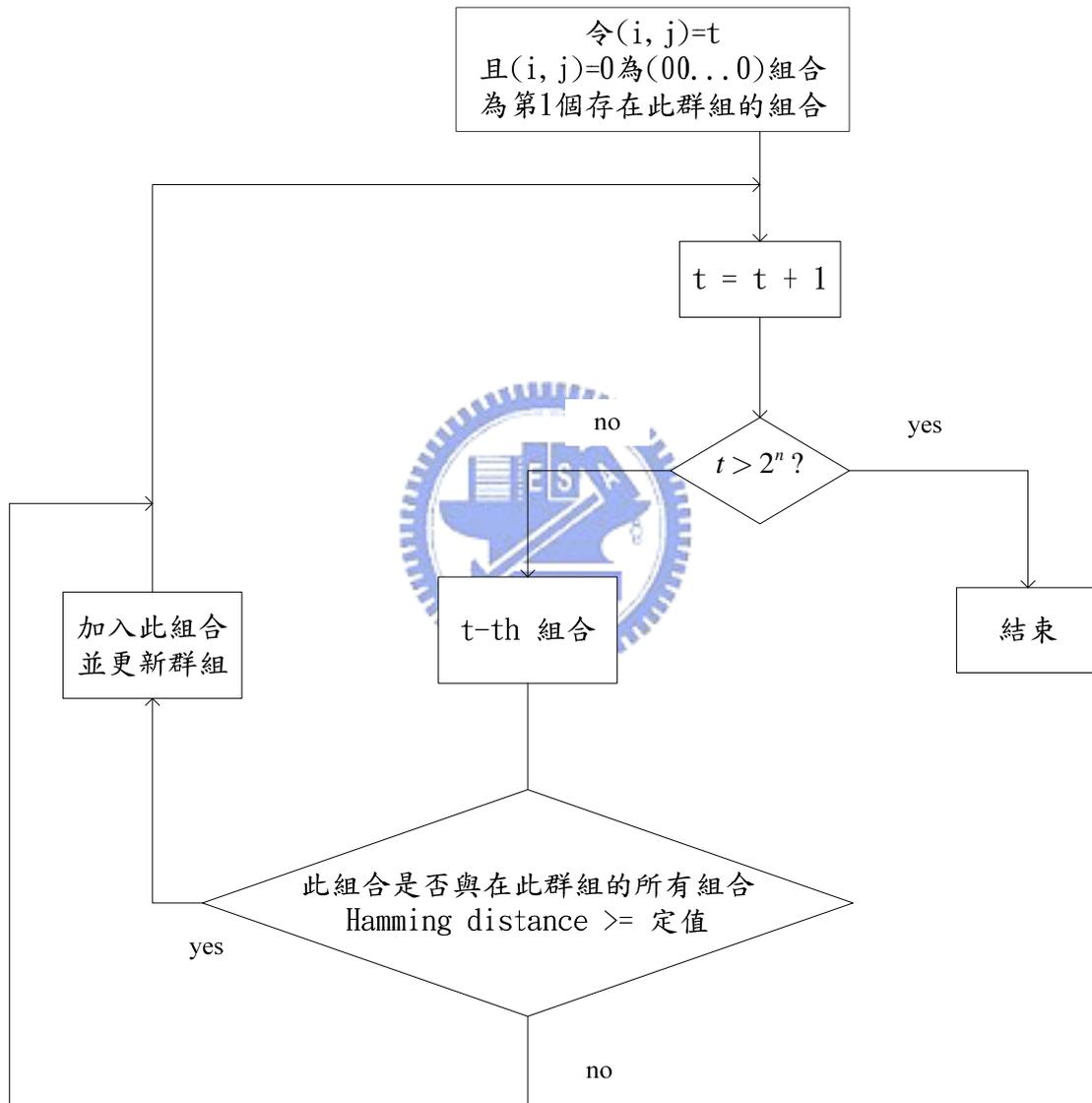


圖 3.1 特殊群組選擇流程圖

很明顯的，當定值越高則符合條件的位元組合相對越少，我們也發現不管定值設為多少，所成群組的組合個數永遠都是二的倍數次，因此

一定有其特殊的結構隱含其中。基於系統設計的考量，我們初步鎖定定值為 3 的例子做深入分析，同時擇定 0 的位元組合 $[000\cdots 000]$ 一定存在此群組之中。若令 a 和 b 為兩個從此群組挑出的位元組合，由於此群組中任意兩組合的漢明距離一定不小於 3，因此 a 和 b 作二進位加法所得的 c ，其結構中位元 1 的數目 k 必須不小於 3，而位元 0 的數目則為 $(n-k)$ 。意即 a 和 b 有 k 個位置是置放不一樣的位元，而其餘的 $(n-k)$ 個位置則代表同樣的位元。為了符合群組的特性， a 和 b 再分別和 0 的位元組合估算其漢明距離。茲針對此 k 個位置做分析，我們分別對四種不同情況做討論。

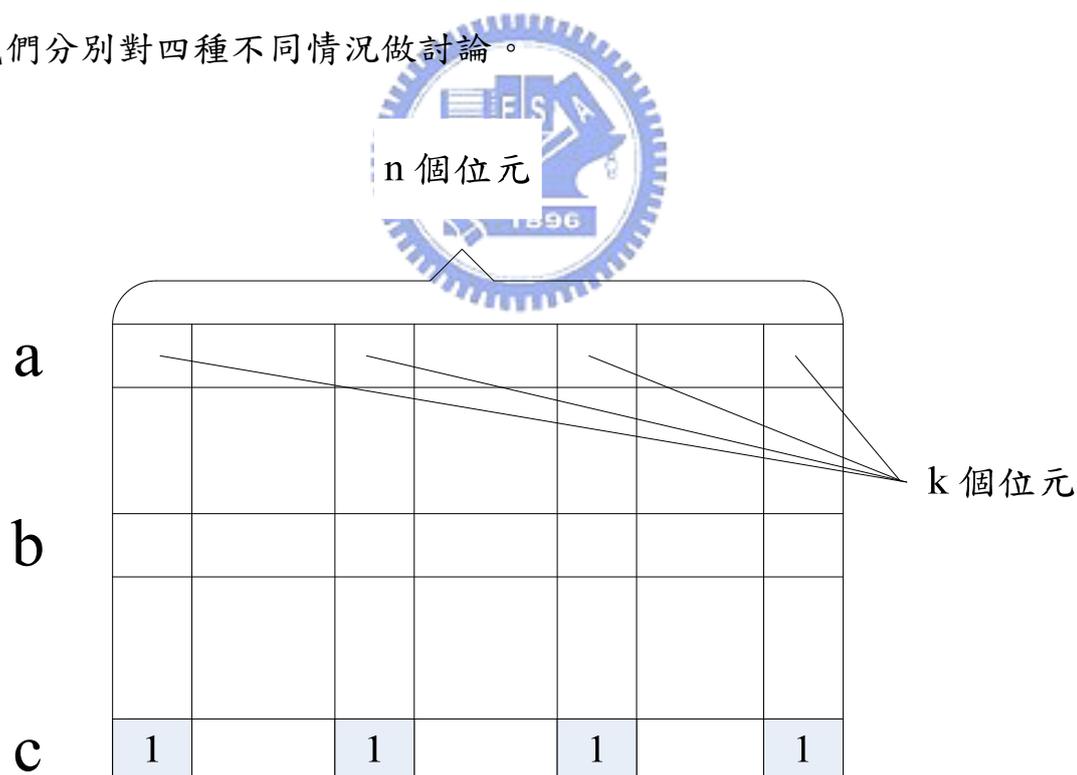


圖 3.2 證明線性區段碼示意圖

- 對於 a 其中 k 個位置皆為 0 位元，則 b 在相同的 k 個位置皆為 1 位元，兩者為了達到和 0 位元組合之間也同樣具有漢明距離

不小於 3 的條件，兩者剩餘的 $(n-k)$ 個位元一定得包含 3 個以上的 1 位元。再分別將 a、b 和 c 做漢明距離，針對 a 而言其漢明距離必定大於 $(k+3)$ ，而對 b 而言其漢明距離必定大於 3。

- 2、對於 a 其中 $(k-1)$ 個位置為 0 位元，只有一個位置為 1 位元，因此對於 b 而言剛好與 a 顛倒，只有一個位置為 0 位元， $(k-1)$ 個位置為 1 位元。兩者為了達到和 0 位元組合也同樣具有漢明距離不小於 3 的條件，兩者剩餘的 $(n-k)$ 個位元一定得包含 2 個以上的 1 位元。再分別將 a、b 和 c 做漢明距離，針對 a 而言其漢明距離必定大於 $(k-1+2)=k+1$ ，而對 b 而言其漢明距離必定大於 $1+2=3$ 。

- 3、對於 a 其中 $(k-2)$ 個位置為 0 位元，只有兩個位置為 1 位元，因此對於 b 而言剛好與 a 顛倒，只有兩個位置為 0 位元， $(k-2)$ 個位置為 1 位元。兩者為了達到和 0 位元組合也同樣具有漢明距離不小於 3 的條件，兩者剩餘的 $(n-k)$ 個位元中必定包含 2 個以上的 1 位元。再分別將 a、b 和 c 做漢明距離，針對 a 而言其漢明距離必定大於 $(k-2+2)=k$ ，而對 b 而言其漢明距離必定大於 $2+2=4$ 。

- 4、對於 a 其中 $(k-3)$ 個位置為 0 位元，有三個位置為 1 位元，因此對於 b 而言剛好與 a 顛倒，有三個位置為 0 位元， $(k-3)$ 個

位置為 1 位元。兩者為了達到和 0 位元組合也同樣具有漢明距離不小於 3 的條件，因此兩者剩餘的 $(n-k)$ 個位元中必定包含 3 個以上的 1 位元。再分別將 a 、 b 和 c 做漢明距離，針對 a 而言其漢明距離必定大於 $(k-3+3)=k$ ，而對 b 而言其漢明距離必定大於 $3+3=6$ 。

綜合以上 4 種情況的討論，可以發現 a 和 b 做二進位加法所得的 c 和 a 、 b 的漢明距離皆不小於 3。這代表此群組任意取兩個位元組合具有加法的封閉性，意即 c 仍然屬於在這群組之中，也說明了這種群組是一個子空間。再加上 0 位元組合必存在此群組之中，因此我們可以證明在定值為 3 的情況下，此群組就是某個合法的產生矩陣所製造出的所有合法碼字。如此也可解釋為什麼經過此演算法找出的群組，其所屬的位元組合數一直是 2 的倍數次方。我們甚至可以從這所有的合法碼字，找出其對應的產生矩陣以及查核矩陣。換句話說，經過此演算法所找出的群組本身就是一群具有位元更正能力的合法碼字。

以上的說明驗證了定值為 3 的情況，此群組為一合法的線性區段碼，只要定出我們希望合法碼字的位元數，就可以找出對應的產生矩陣以及查核矩陣。從線性區段碼的角度來看，所謂的定值其實就是在編碼理論中的最小距離值。依此列出經過證實的 (n, k) 線性區段碼，其最小距離皆為 3，即可更正 1 個發生錯誤的位元： $(6, 3)$ 、 $(8, 4)$ 、

(10, 6)、(12, 8)、(14, 10)，其中(10, 6)以及(12, 8)兩種線性區段碼將會運用在我們的實驗系統之中。另外，我們可以推廣到 3 以上的定值，利用同樣的方法也可證明出亦為線性區段碼的特性。

3.1.3 位置指定演算法

多重描述的索引指定是將一個訊號經過純量量化或向量量化處理之後所得的量化索引，經過一個指定函數產生不同的描述再分別傳輸。本實驗的環境都設定於兩個傳輸通道之下，因此任意一個量化索引經過索引指定之後，會針對此索引在指定矩陣的座標，分別在兩個通道傳輸此座標的列索引以及行索引。我們的構想是，如果此兩個索引值的位元表示式合併起來可以成為一個合法的碼字，接收端在還未進入多重描述解碼器之前，可以事先對此碼字做個別偵測或是更正。實際的做法就是利用查核矩陣盡可能找出可更正的錯誤位元並更正之，無法更正錯誤的位元就維持原來的接收碼字，再進入最後階段的多重描述解碼器執行錯誤隱匿(error concealment)的動作。按照 3.1.2 所介紹的演算法，可以找尋出符合條件的特殊群組，此群組包含一個產生矩陣所編碼輸出的合法碼字，將此所有合法碼字分別做對等的分割，該位元組合的前半部與後半部分別對應到指定矩陣的列索引與行索

引，依順序在指定矩陣中慢慢將對應位置出來。

我們可以想像一個多重描述量化器，輸入訊號經過向量量化或是純量量化，而得到一個 k 位元的量化索引，再經由指定矩陣的查詢，分別在兩通道傳輸 $n/2$ 位元。總體來看就像個黑盒子，輸入一個 k 位元的資訊，然後輸出為一個 n 位元的碼字，再做切割分別在兩個通道傳輸 $n/2$ 位元。藉由 3.1.2 所提出的演算法，為了有更正 1 個錯誤位元的能力，定值至少要 3 以上，再給定所需要的 n 位元值，就可以很容易地找出對應的產生矩陣以及所屬的合法碼字。將這所有的合法碼字做等分切割，前 $n/2$ 位元當作第一個通道傳輸的位元，而後 $n/2$ 位元則為第二個通道所傳輸的位元。若分別將這些位元組合以其十進位表示，即可在指定矩陣中找到一個相對應的位置代表該合法碼字。舉例來說，經過 3.1.2 的演算法，我們找出 $(n, k) = (12, 8)$ 的線性區段碼，其最小距離值為 3，代表其可以更正 1 位元發生錯誤的情況。此種線性區段碼運用在我們設計的多重描述量化之中，即代表輸入為 8 位元長度的訊號，在一個(6 位元 * 6 位元)的矩陣之中需要找出置放 256 索引的位置。透過以上的描述，分別對這 256 個合法碼字做分割，前 6 位元代表在此 64×64 矩陣的列數，後 6 位元則代表在此 64×64 矩陣的行數，就可以在一個指定矩陣找出這 256 個座標位置。它們所共同具有的特性，是任意

一個位置其對應的列數與行數的位元表示，合併起來一定是個合法的碼字。我們在解碼端可以有效利用此特性，藉由查核矩陣在有限的錯誤發生率更正錯誤位元，達到提升系統效能的目的。圖 3.3 是實際利用 3.1.2 節所介紹的演算法以及位置指定的方法，在(4*4)的指定矩陣中找尋特殊位置，其中圖中所標示的灰色位置即代表符合輸出位元為 4 以及定值為 2。換句話說，可以視為輸入為 3 位元的量化索引經過指定函數的對應，分別在兩個通道傳輸 2 位元的列索引以及行索引。

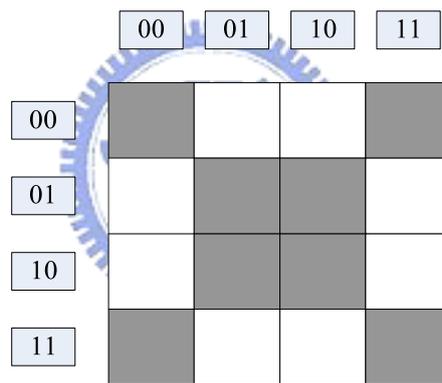


圖 3.3 $(n, k)=(4, 3)$ 且 $d=2$ 的位置指定

3.2 位元對應機制

3.2.1 位元對應

經過 3.1 節的介紹，事先在指定矩陣之中規畫好位置，再將量化索引值放入這些規畫好的位置之中，即可完成索引指定的整體設

計。我們繼續沿用 3.1.3 節的例子，經過位置指定之後，在(6 位元 * 6 位元)的矩陣之中規畫好所有 256 個合法碼字所屬的位置集合之後，才將 256 個量化索引分別放入其對應的位置。有別於前章的索引是隨機置入，而這裡一定得放入這些已規劃好的位置集合，才不會破壞原本碼字該有的結構。而接下來所探討的問題，就是在指定矩陣已規畫好的位置集合中，決定每個量化索引置放最合適的位置，所用的方法就是前章所提到的二元置換演算法，將於下一小節介紹。

3.2.2 通道匹配的二元置換演算法

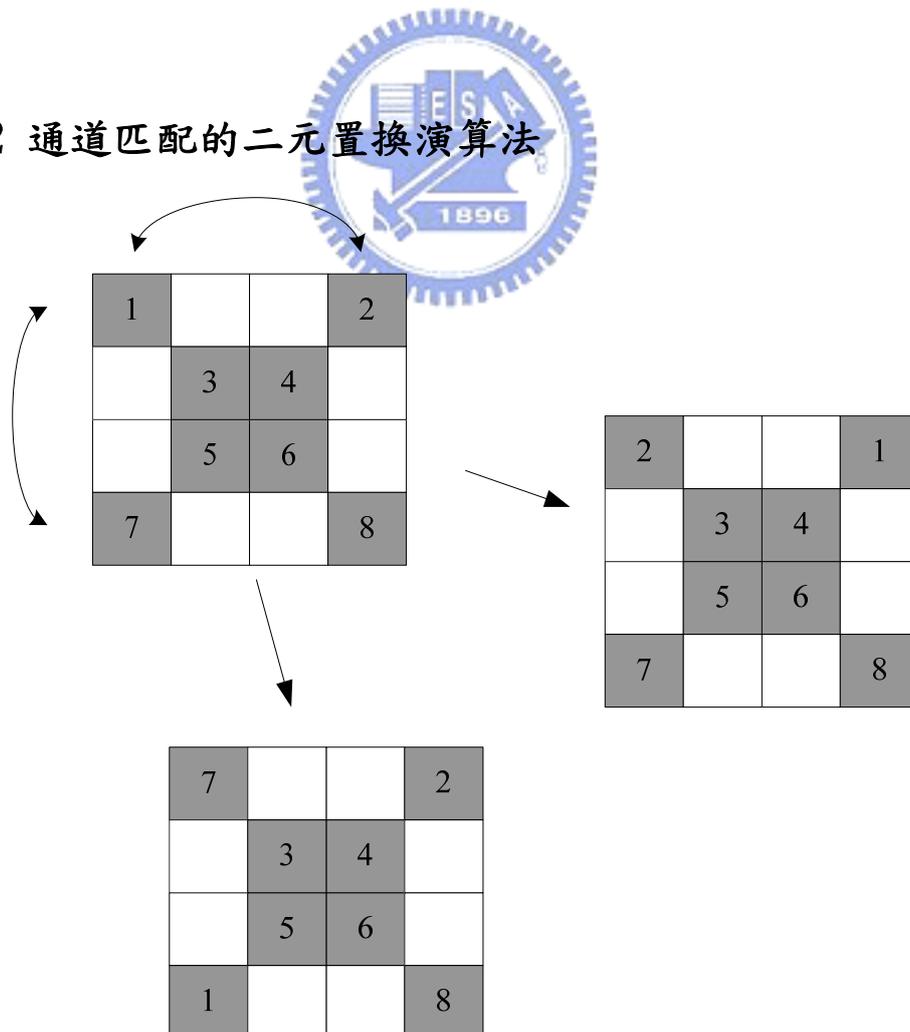


圖 3.4 二元置換過程示意圖

在前一章發現，利用二元置換演算法設計於匹配通道的索引指定，藉著持續的兩兩置換找到一個最佳的索引指定函數，而在不同的錯誤環境所得到的指定排列方式均不同。龐大計算量一直是不可避免的問題，前一章所提供的二元置換演算法，是根據不同的錯誤環境持續對每個位置做置換再計算其失真度，每做一次判斷均對矩陣中每個位置做計算的動作，不難發現當矩陣非常巨大的時候，計算量會成倍數的成長。3.1 節所提供的演算法為了要達到傳輸的位元是一個合法碼字的目的，事先在指定矩陣之中規畫好幾個特定的位置，這些位置都能產生出合法的碼字。不論在初始的索引指定或是二元置換演算法的過程中，所有的置換以及位元對應都只能在這已經規劃好的位置上做變動。此章的二元置換演算法大致上和前一章架構一樣，最大不同在於前一章要對所有的位置做運算，這邊我們只需對此特定規劃好的位置計算其失真度，與前一章比較起來，每次做的判斷計算量就減少很多，解碼的效果也因為隱含的線性區段碼結構而提升，實驗結果會在第五章更詳細的描述。

3.3 基於位元錯誤通道的多重描述向量量化系統

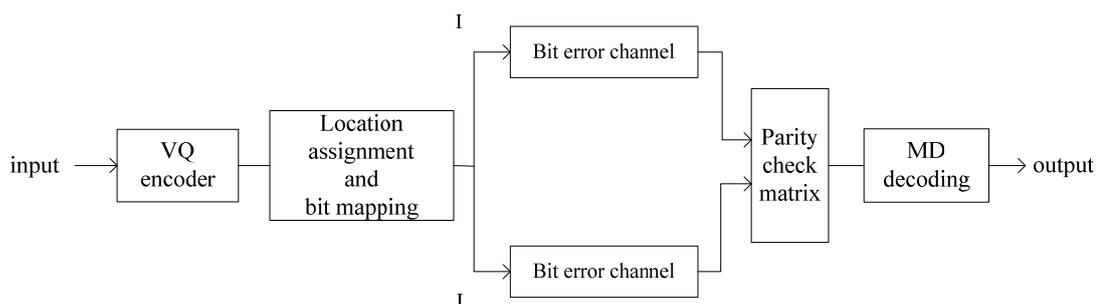


圖 3.5 位元錯誤通道的多重描述量化系統

此系統架構和前一章最大的不同就是增加了查核矩陣的錯誤更正機制，利用 3.1.2 的方法經過位置指定以及位元對應之後，可以確保 $[I, J]$ 的二進位表示式絕對是一個合法的碼字。經過通道雜訊的干擾，在做多重描述解碼之前，我們會將兩個通道的索引接收值經由查核矩陣做預先的判定與錯誤更正的動作。這時候會有下列各種情況發生：

- 1、發現錯誤位元，並正確的更正回來，才繼續做後續的多重描述解碼處理。
- 2、發現錯誤位元，卻更正成另外一個合法碼字，才繼續做後續的多重描述解碼處理。

- 3、 發現錯誤位元，卻無法更正回來，對碼字不做任何的變更動作，直接通過此查核矩陣檢查機制，交給之後的多重描述解碼器做最後解碼。
- 4、 未發現任何錯誤，直接通過此查核矩陣檢查機制，多重描述解碼器做最後解碼動作。

多重描述解碼器會根據收到的列索引以及行索引 \hat{i}, \hat{j} ，計算出所有可能判定結果的機率，再依此機率值決定最後結果，方法如同 2.3 節所介紹的：

$$\tilde{X}(\hat{i}, \hat{j}) = \sum_{x=1}^{2^M} b_x p(x | \hat{i}, \hat{j})$$

其中 b_x 為事前大量訓練出來的第 x 個碼字。

第四章

基於 AWGN 通道的索引指定

在前二章所探討的系統都是基於位元錯誤通道的考慮而設計，此章則將通道模型推廣到更複雜的 AWGN 環境。因應此通道環境，軟性輸入輸出系統的通道轉移機率計算以及解碼器也有所不同，而二元置換演算法也基於通道匹配的原則做最佳化的設計。這章和前一章的核心概念大致相同，包括資訊量化、具有線性區段碼結構的位置指定以及其位元對應。最大的不同在於解碼模組，由於經由 AWGN 雜訊干擾而接受到的訊號皆為實數，無法適用前一章採用的查核矩陣。有鑑於此，我們引入柵狀(trellis)解碼的機制，取代前一章的查核矩陣檢查機制以及多重描述解碼器。接下來，會依序介紹通道轉移機率，以及通道匹配的二進位置換演算法，最後介紹柵狀解碼機制。

4.1 通道轉移機率推導

第二章所推導的通道轉移機率是建構在位元錯誤環境之下，使用漢明距離並配合各自錯誤環境計算其錯誤機率，並據以設計通道匹配的二元置換演算法。此章的通道轉移機率雖然不再用漢明距離計算錯誤率，但仍先從後驗機率出發。

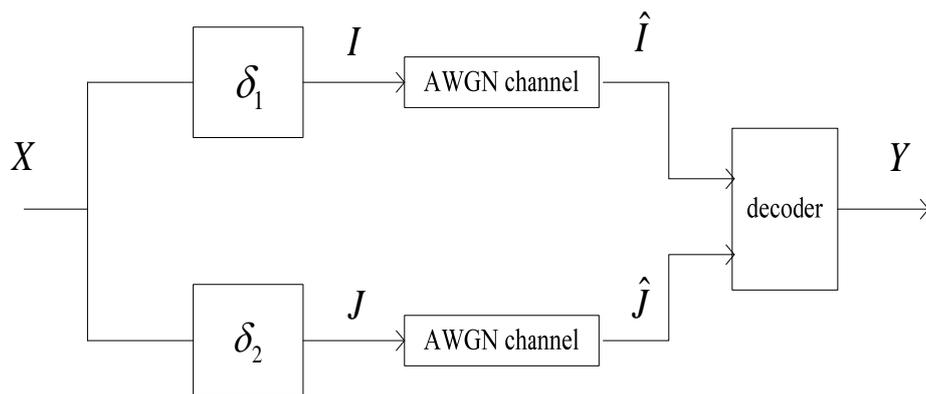


圖 4.1 AWGN 通道的多重描述量化系統

如圖 4.1 所示，向量量化編碼輸出的索引 X ，經過索引指定而產生指定矩陣中對應的列索引 I 以及行索引 J ，分別在兩個通道經過 AWGN 干擾之後而接收到 \hat{I} 及 \hat{J} 。令 Y 為解碼輸出的索引，而 Y_i 及 Y_j 分別為其在指定矩陣的列索引以及行索引。基於最大後驗機率準則，按照下式做解碼的判斷：

$$\begin{aligned}
 Y_{opt} &= \arg \max_y p(y | \hat{i}, \hat{j}) \\
 &= \arg \max_y p(\hat{i}, \hat{j} | y) p(y) \\
 &= \arg \max_y p(y) p(\hat{i} | y_i) p(\hat{j} | y_j) \\
 &= \arg \max_y p(y) \left(\prod_{n=1}^k p(\hat{s}_n^{(i)} | s_n^{(i)}(y)) \right) \left(\prod_{n=1}^k p(\hat{s}_n^{(j)} | s_n^{(j)}(y)) \right) \\
 &= \arg \max_y \log p(y) + \sum_{n=1}^k \log(p(\hat{s}_n^{(i)} | s_n^{(i)}(y))) + \sum_{n=1}^k \log(p(\hat{s}_n^{(j)} | s_n^{(j)}(y)))
 \end{aligned}$$

其中 $\hat{s}_n^{(i)}(y)$ 及 $\hat{s}_n^{(j)}(y)$ 分別代表 y_i 及 y_j 的二進位表示式中的第 n 個位元以，而其接收值則分別以 $\hat{s}_n^{(i)}$ 及 $\hat{s}_n^{(j)}$ 表示。因考量在 AWGN 通道環境之下，我們引入了高斯機率分布：

$$\begin{aligned}\log p(\hat{s}_n^{(i)} | s_n^{(i)}(y)) &= \log \frac{1}{\sqrt{2\pi\sigma^2}} + \log e^{-\frac{(\hat{s}_n^{(i)} - s_n^{(i)}(y))^2}{2\sigma^2}} \\ &= \log \frac{1}{\sqrt{2\pi\sigma^2}} - \frac{(\hat{s}_n^{(i)} - s_n^{(i)}(y))^2}{2\sigma^2} \\ \log p(\hat{s}_n^{(j)} | s_n^{(j)}(y)) &= \log \frac{1}{\sqrt{2\pi\sigma^2}} - \frac{(\hat{s}_n^{(j)} - s_n^{(j)}(y))^2}{2\sigma^2}\end{aligned}$$

將式子帶回上式，可以整理出：

$$Y_{opt} = \arg \max_y (-2k \log \sqrt{2\pi\sigma^2} - \sum_{n=1}^k \frac{(\hat{s}_n^{(i)} - s_n^{(i)}(y))^2}{2\sigma^2} - \sum_{n=1}^k \frac{(\hat{s}_n^{(j)} - s_n^{(j)}(y))^2}{2\sigma^2} + \log p(y))$$

接下來，在傳送索引為 X 的條件之下，定義 $MAP(Y' \rightarrow Y | X)$ 判

斷索引值 Y_{opt} 是 Y 而不是 Y' 的事件，亦即 $p(Y | \hat{i}, \hat{j}) > p(Y' | \hat{i}, \hat{j})$

$$\begin{aligned}& -2k \log \sqrt{2\pi\sigma^2} - \sum_{n=1}^k \frac{(\tilde{s}_n^{(i)}(X) - s_n^{(i)}(Y))^2}{2\sigma^2} - \sum_{n=1}^k \frac{(\tilde{s}_n^{(j)}(X) - s_n^{(j)}(Y))^2}{2\sigma^2} + \log p(Y) \\ & > -2k \log \sqrt{2\pi\sigma^2} - \sum_{n=1}^k \frac{(\tilde{s}_n^{(i)}(X) - s_n^{(i)}(Y'))^2}{2\sigma^2} - \sum_{n=1}^k \frac{(\tilde{s}_n^{(j)}(X) - s_n^{(j)}(Y'))^2}{2\sigma^2} + \log p(Y')\end{aligned}$$

接下來針對上式之事件做整理，可得：

$$\sum_{m=i,j} \sum_{n=1}^k \frac{(\hat{s}_n^{(m)}(X) - s_n^{(m)}(Y'))^2 - (\hat{s}_n^{(m)}(X) - s_n^{(m)}(Y))^2}{2\sigma^2} > \log \frac{p(Y')}{p(Y)}$$

其中 $\hat{s}_n^{(m)}(X) = s_n^{(m)}(X) + W_n^{(m)}$ ， $W_n^{(m)} \in \text{Gauss}(0, \sigma^2)$ $m = i, j$

定義 $d_n^{(m)}(X, Y) = s_n^{(m)}(X) - s_n^{(m)}(Y)$ ， $m = i, j$ ，以方便計算

$$\Rightarrow \sum_{m=i,j} \sum_{n=1}^k \frac{(d_n^{(m)}(X, Y') + W_n^{(m)})^2}{2\sigma^2} - \frac{(d_n^{(m)}(X, Y) + W_n^{(m)})^2}{2\sigma^2} > \log \frac{p(Y')}{p(Y)}$$

$$\begin{aligned} &\Rightarrow \sum_{m=i,j} \sum_{n=1}^k (d_n^{(m)}(X, Y') + W_n^{(m)})^2 - (d_n^{(m)}(X, Y) + W_n^{(m)})^2 > 2\sigma^2 \log \frac{p(Y')}{p(Y)} \\ &\Rightarrow \sum_{m=i,j} \sum_{n=1}^k d_n^{(m)}(X, Y')^2 - d_n^{(m)}(X, Y)^2 + 2 \sum_{m=i,j} \sum_{n=1}^k W_n^{(m)} d_n^{(m)}(Y, Y') > 2\sigma^2 \log \frac{p(Y')}{p(Y)} \\ &\Rightarrow \sum_{m=i,j} \sum_{n=1}^k W_n^{(m)} d_n^{(m)}(Y, Y') > \sigma^2 \log \frac{p(Y')}{p(Y)} - \frac{1}{2} \sum_{m=i,j} \sum_{n=1}^k (d_n^{(m)}(X, Y')^2 - d_n^{(m)}(X, Y)^2) \end{aligned}$$

注意左式的 $\sum_{m=i,j} \sum_{n=1}^k W_n^{(m)} d_n^{(m)}(Y, Y')$ ，可以發現是一群高斯隨機變數的線

性組合，因此可以證明其亦為一高斯隨機變數，其平均值為 0，而其

變異數則為 $\sum_{m=i,j} \sum_{n=1}^k \sigma^2 d_n^{(m)}(Y, Y')^2$ 按照機率的理論，任一隨機變數 Y

的機率，可從其機率密度函數 $f_Y(y)$ 估算而得 $p(Y > a) = \int_a^\infty f_Y(y) dy$ 。

習慣上，定義一個 Q 函數以利積分計算， $Q(a) = \frac{1}{\sqrt{2\pi}} \int_a^\infty e^{-y^2/2} dy$ 。

舉例而言，令 Y 為一個期望值 μ 為且變異數 σ^2 為高斯隨機變數，則

$$p(Y > a) = \frac{1}{\sqrt{2\pi\sigma^2}} \int_a^\infty e^{-\frac{(y-\mu)^2}{2\sigma^2}} dy = Q\left(\frac{a-\mu}{\sigma}\right)。$$

同理可推，事件 $MAP(Y' \rightarrow Y | X)$ 的機率為

$$p(Y' \rightarrow Y | X) = Q\left(\frac{\sigma^2 \log\left(\frac{p(Y')}{p(Y)}\right) - \frac{1}{2} \sum_{m=i,j} \sum_{n=1}^k (d_n^{(m)}(X, Y')^2 - d_n^{(m)}(X, Y)^2)}{\sqrt{\sum_{m=i,j} \sum_{n=1}^k \sigma^2 d_n^{(m)}(Y, Y')^2}}\right)$$

實際上，通道轉移機率為 $p(Y | X) = \text{Prob}\left\{ \bigcap_{Y': Y' \neq Y} MAP(Y' \rightarrow Y | X) \right\}$ ，

而其推導為一個難以分析的問題。因此我們參考[10]利用此式的上限來近似 $p(Y|X)$ ：

$$p(Y|X) \leq \text{prob}\{MAP(X \rightarrow Y|X)\} = \text{prob}(X \rightarrow Y|X)$$

$$p(Y|X) \approx p(X \rightarrow Y|X)$$

此一公式將會在接下來小節介紹的失真度公式中計算運用，提供二元置換演算法運算的依據。

4.2 通道匹配的二元置換演算法

前一章我們考量在位元錯誤通道下，因此在接收端所收到的訊號皆為二位元表示的列索引以及行索引。收到的兩個索引值透過解碼輸出的結果與編碼端的碼字數，配合通道轉移機率，計算出每個指定矩陣的失真度及矩陣中每個索引的成本值，以供二元置換演算法的置換根據。如式(2.4)所示，不難發現收到的列索引以及行索引已有範圍限制，意即在未傳輸之前解碼端已經明確知道將會收到的索引型態。此章則考量在 AWGN 的通道環境之下，在軟性輸入輸出的領域中，接收端收到的訊號可能為任意實數，無明確的範圍可供解碼器預先做解碼的動作。使用的失真度評定方式為：

$$D = \sum_{x=1}^{2^M} p(x) \sum_{y=1}^{2^M} \|b_x - b_y\|^2 p(y|x)$$

其中 $p(y|x)$ 就是 4.1 節所介紹的通道轉移機率，它也是造成和前幾章採用的評定方式之最大差別。如同前章的二進位置換演算法，在 AWGN 通道環境之下仍然需要根據成本值作置換，因此定義指定矩陣中每個索引對應的成本函數：

$$C(x) = p(x) \sum_{y=1}^{2^M} \|b_x - b_y\|^2 p(y|x)$$

按照第二章所提供的方法，分別計算出每個索引的成本值，依序由大排到小，選取最大的成本值當作目標，按照多重描述二元置換演算法找出最佳的指定索引。值得注意的是，此章同樣考慮了第三章所提供的概念，在指定矩陣中引入了線性區段碼的概念。因此事先做了位置指定，這些事先預定好的位置同樣可以組合成一個合法的碼字，提供在解碼端做解碼以及更正的動作。所以在置換的過程中，兩兩位置的交換也限於這些預先規劃好的位置中，不得置換到其他非指定的位置，大致上的置換原理都和前章相同。

4.3 基於 AWGN 通道的多重描述向量量化系統

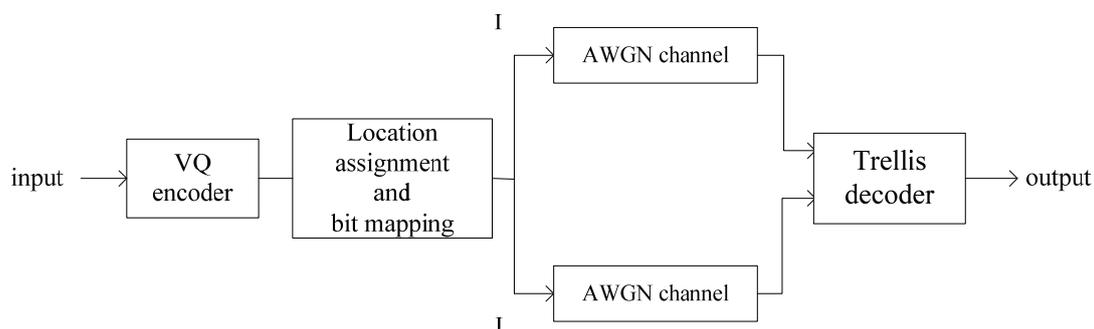


圖 4.2 多重路徑量化系統架構圖

4.3.1 系統架構

此章的系統架構圖如圖 4.2 所示，向量量化的輸出索引，經過位置指定以及位元對應之後，以 $[I, J]$ 的二進位表示某個產生矩陣所產生的合法碼字，再分別經由兩個通道傳送其列索引以及行索引。在經過 AWGN 雜訊干擾之後，在接收端收到的是實數，因此查核矩陣並不適用在此系統。取而代之的是可以運用在軟性空間領域的 trellis 解碼器，在解碼過程考慮所有可能的碼字，並輸出一個最大可能性的碼字。在最後判斷的時候再將此碼字分割為兩部分，前半部對應到列索引，後半部對應到行索引，再配合經過通道匹配二元置換演算法處理的指定矩陣，最後可以決定一個索引值出來。接下來介紹 trellis 解碼器。

4.3.2 trellis 解碼器

顧名思義，柵狀解碼器(trellis decoder)就是利用柵狀碼圖結構做解碼的動作，完整的柵狀圖(trellis diagram)包含路徑的輸出以及狀態的表示。為了簡化柵狀圖的建構，我們需要重新安排產生矩陣使其轉化成為一個 trellis-oriented form(TOF)的特殊格式。首先透過第三章所介紹的演算法，找出某個產生矩陣 G 的所有碼字，進一步利用高斯消去法找出其 TOF 格式的產生矩陣。參考 Shu Lin 所寫的錯誤更正碼[8]，按照兩個準則：

- 1、 產生矩陣之中每一列起頭 1(leading 1)的行索引，一定要小於該列以下其餘列起頭 1 的行索引。
- 2、 在矩陣之中任意兩列的收尾 1(trailing 1)需有不同的行索引。

按照這兩個準則，高斯消去法可以將任意一個產生矩陣轉成符合 TOF 格式的產生矩陣 G_{TOGM} ，其中 TOGM 代表 trellis oriented generator matrix，舉例而言，給定一產生矩陣：

$$G = \begin{bmatrix} 1 & 0 & 1 & 0 & 0 & 1 & 1 \\ 1 & 0 & 1 & 0 & 1 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 & 0 & 0 \end{bmatrix}$$

將第一列與第二列做二進位加法，再與第三列置換可得其一 TOF 為：

$$G_{TOGM} = \begin{bmatrix} 1 & 0 & 1 & 0 & 1 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 \end{bmatrix}$$

一輸入訊號 $\underline{a} = [a_0, a_1, \dots, a_{k-1}]$ 經過產生矩陣 G_{TOGM} 的編碼處理，而得到一個合法碼字 $\underline{V} = [v_0, v_1, \dots, v_{n-1}]$ 輸出，針對每時刻的狀態表示及狀態轉移所衍生出的路徑輸出，主要依據該時刻輸入訊號與 G_{TOGM} 的對應產生，其對應關係則需進一步統計 TOF 各個特性的列集合。值得注意的是，柵狀圖上任一條完整路徑上所有表示的輸出即為一合法碼字，換句話說，根據此 G_{TOGM} 所編碼而得的碼字可用此柵狀圖完整呈現。接下來將統計 G_{TOGM} 所有列向量的對應關係，將來柵狀圖就是根據這些畫分的列向量慢慢刻畫出來。

針對 G_{TOGM} 所屬的任一系列向量 $\mathbf{g} = (g_0, g_1, \dots, g_{n-1})$ ，我們定義其位元展開(bit span) $\phi(\mathbf{g}) = [i, j] = \{i, i+1, \dots, j\}$ ，代表此列向量之中包含所有非零元素的最小索引區間。換句話說， $g_i = 1$ 且 $g_j = 1$ 分別是此列集合的起頭 1 以及收尾 1。舉例而言：

$$\begin{array}{c}
 \text{time} \rightarrow 0 \quad 1 \quad 2 \quad 3 \quad 4 \quad 5 \quad 6 \quad 7 \\
 \\
 G_{TOGM} = \begin{bmatrix} \mathbf{g}_0 \\ \mathbf{g}_1 \\ \mathbf{g}_2 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 1 & 0 & 1 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 \end{bmatrix} \\
 \\
 \text{bit position} \rightarrow 0 \quad 1 \quad 2 \quad 3 \quad 4 \quad 5 \quad 6
 \end{array}$$

其中 $\phi(\mathbf{g}_0) = [0, 4]$ 、 $\phi(\mathbf{g}_1) = [1, 3]$ 以及 $\phi(\mathbf{g}_2) = [4, 6]$ 。接著定義

- 1、 G_i^p 為位元展開 $\phi(\mathbf{g})$ 包含於 $[0, i-1]$ 區間的所有列向量集合。
- 2、 G_i^f 為位元展開 $\phi(\mathbf{g})$ 包含於 $[i, n-1]$ 區間的所有列向量集合。
- 3、 G_i^s 為在同時間之內，不屬於以上兩個集合的其餘列向量。
- 4、 A_i^s 為對應於 G_i^s 中所屬列向量的資訊位元 (information bit) 集合
- 5、 ρ_i 為 G_i^s 所屬列向量的數目。

Time i	G_i^p	G_i^f	G_i^s	A_i^s	ρ_i
0	ϕ	$\{\mathbf{g}_0, \mathbf{g}_1, \mathbf{g}_2\}$	ϕ	ϕ	0
1	ϕ	$\{\mathbf{g}_1, \mathbf{g}_2\}$	$\{\mathbf{g}_0\}$	$\{a_0\}$	1
2	ϕ	$\{\mathbf{g}_2\}$	$\{\mathbf{g}_0, \mathbf{g}_1\}$	$\{a_0, a_1\}$	2
3	ϕ	$\{\mathbf{g}_2\}$	$\{\mathbf{g}_0, \mathbf{g}_1\}$	$\{a_0, a_1\}$	2
4	$\{\mathbf{g}_1\}$	$\{\mathbf{g}_2\}$	$\{\mathbf{g}_0\}$	$\{a_0\}$	1

5	$\{\mathbf{g}_0, \mathbf{g}_1\}$	ϕ	$\{\mathbf{g}_2\}$	$\{a_2\}$	1
6	$\{\mathbf{g}_0, \mathbf{g}_1\}$	ϕ	$\{\mathbf{g}_2\}$	$\{a_2\}$	1
7	$\{\mathbf{g}_0, \mathbf{g}_1, \mathbf{g}_2\}$	ϕ	ϕ	ϕ	0

表 4.1 狀態分析表

一個完整的柵狀圖，每條路徑上都具有兩種含意，一是個代表狀態(state)的轉移，另一個代表對應的輸出，藉由柵狀圖架構可計算每一路徑的 metric。以 AWGN 通道模型為例，所屬的 metric 為訊號接收值與狀態輸出的歐幾里得距離(Euclidean distance)平方。而解碼的程序依據 Viterbi 演算法，計算可能的碼字輸出所對應的累積 metric，其中具有最小 metric 的可行路徑解也將會是最大似然率(maximum likelihood)的解碼結果。有鑑於此，定義每個時刻狀態的表示及對應的輸出就顯得非常重要。

首先定義狀態的表示，舉例而言，針對第 3 個時刻 $G_3^s = \{\mathbf{g}_0, \mathbf{g}_1\}$ 因此對應 $A_3^s = \{a_0, a_1\}$ ，具有的狀態組合為 (a_0, a_1) ，所以共有 4 種狀態，分別為 $(0, 0)$ 、 $(0, 1)$ 、 $(1, 0)$ 以及 $(1, 1)$ ，而第 5 個時刻 $G_5^s = \{\mathbf{g}_2\}$ 對應 $A_5^s = \{a_2\}$ ，因此具有的狀態組合為 (a_2) ，所以其所有的狀態為 0 以及 1。如圖 4.3 所示。

接下來說明從第 i 個時刻到第 $i+1$ 個時刻的輸出位元。其值根據

有無 a^* 的出現有兩種情況發生。在第 i 個時刻其 G_i^f 若有列向量符合其起頭 1 出現在第 i 個位置的性質，我們定義此列向量為該時刻的 \mathbf{g}^* ，而其對應的資訊位元為 a^* 。舉例而言，第 1 時刻 $G_1^f = \{\mathbf{g}_1, \mathbf{g}_2\}$ ，其中 \mathbf{g}_1 起頭 1 正好為出現在第 1 個位置，因此其 $\mathbf{g}^* = \mathbf{g}_1$ 而 $a^* = a_1$ 。第 i 時刻點到第 $i+1$ 時刻點的輸出位元為：

$$v_i = a^* + \sum_{l=1}^{\rho_i} a_l^{(i)} g_{l,i}^{(i)}$$

反之，若該時刻點並無 \mathbf{g}^* 存在且也無 a^* 存在，則其輸出位元為：

$$v_i = \sum_{l=1}^{\rho_i} a_l^{(i)} g_{l,i}^{(i)}$$

其中 $a_l^{(i)}$ 代表第 i 時刻 A_i^s 集合中的第 l 個資訊位元，而 $g_{l,i}^{(i)}$ 代表 G_i^s 集合中第 l 個列向量 $\mathbf{g}_l^{(i)} = (g_{l0}^{(i)}, g_{l1}^{(i)}, \dots, g_{l,n-1}^{(i)})$ 的第 i 個位元。沿用以上的例子，我們計算第 1 個時刻到第 2 個時刻的輸出位元。從表 4.1 中我們可以知道在第 1 個時刻的狀態集合為 $A_1^{(s)} = \{a_0\}$ ，有兩種狀態 0 或者是 1，並且 G_1^f 內存有符合 \mathbf{g}^* 的列向量在 $\mathbf{g}^* = \mathbf{g}_1$ 。此外， $G_1^{(s)} = \{\mathbf{g}_0\}$ 且 $\rho_1 = 1$ ，因此 $v_1 = a_1 + a_0 g_{01}$ 。再舉個例子來說明，我們計算第 2 時刻到第 3 時刻的輸出位元，從表中可以知道在第 2 個時刻的狀態集合為 $A_2^{(s)} = \{a_0, a_1\}$ ，有四種組合狀態，分別為 $\{0, 0\}$ 、 $\{0, 1\}$ 、 $\{1, 0\}$ 以及 $\{1, 1\}$ ，並且在 G_2^f 內不存在符合 \mathbf{g}^* 的列集合。此外， $G_2^f = \{\mathbf{g}_0, \mathbf{g}_1\}$ 且 $\rho_2 = 2$ ，因此 $v_2 = 0 + a_0 g_{02} + a_1 g_{12}$ 。因此我們由這

個具有 TOF 的產生矩陣，可以刻畫出如圖 4.3 所示的柵狀圖：

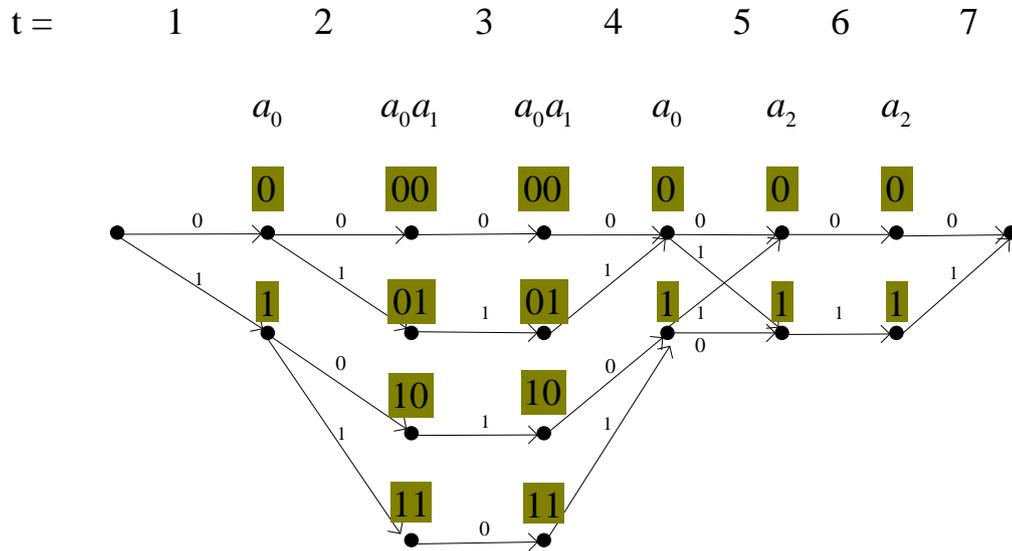


圖 4.3 柵狀解碼圖



解碼過程根據接收端收到的資訊，按照每個資訊收到的時刻和柵狀解碼圖中的時刻做一對一對應，計算該時刻點收到資訊和柵狀圖中相同時刻所有輸出的歐幾里得距離平方。每個狀態(state)會累積前一時刻的 metric，前一時刻所累積的 metric 和該時刻所得歐幾里得距離平方的相加即為該時刻該狀態的 metric。若該狀態具有兩個以上的來源路徑時，會互相比較其 metric，選擇其中具有最小 metric 的路徑當做該狀態的資訊。待最後一個時刻，亦即接受碼字結束之時，統計擁有最小 metric 的當作輸出結果。舉個例子來說，如圖 4.3 所示，若在第一個時刻狀態 0 和 1 所累積的 metric 分別 1.25、2.64，

當時第二個時刻收到的資訊為 3.14，則(01)狀態其 metric 為 $1.25 + (3.14 - (1))^2 = 5.8296$ ，而(10)狀態其 metric 則為 $2.64 + (3.14 - (-1))^2 = 19.7796$ 。

最後根據解碼出的碼字，將此碼字按照 [I, J] 的畫分，前半部的二進位表示可以對應到一個列索引，而碼字後半部的碼字對應一個行索引，這樣的解碼過程輸出結果絕對是一個合法的碼字，因此在指定矩陣之中絕對不會發生對應到空的位置而無法判斷的錯誤。此改良的結果會在下一章實驗結果中探討，並且運用在 AWR-WB 編碼參數上呈現整個解碼效果。



第五章

實驗模擬與結果分析

三種不同的系統設計，第一種即為第二章所介紹的系統架構，第二種及第三種則分別為第三章以及第四章所介紹的系統。為了方便區別，我們分別將之命名為 MDC1、MDC2 及 MDC3。第二章所介紹的 MDC1 系統架構中，使用通道匹配二元置換演算法經過一次次的置換運算，逐漸找到與當時通道環境匹配的最佳指定矩陣，此指定矩陣可有效提升系統的整體效能，將在第一節用實驗結果加以驗證。第二節則是在指定索引的同時，找尋一些特殊的位置，使在該位置的索引所對應的列索引及行索引可成為一個碼字。解碼端可利用此結構以改善系統效能，此技術將會依序運用在錯誤位元以及 AWGN 通道環境之下。第三節則是將三種不同的多重描述向量量化應用在 AMR-WB 系統，並比較不同設計的效能評比差異。

5.1 通道匹配二元置換的實驗

5.1.1 系統模擬之步驟說明

在此節之內容中，主要是將貫徹整篇論文的二元置換演算法進行系統模擬與結果分析。利用包含了男聲與女聲的大量語料，經過 AMR-WB 編碼器得到 194881 組 *ISF* 索引，並記錄其個別索引的機率做

為事前機率使用。在這些語料內中取出一段 200 秒長度的語音作為參考，利用 AMR-WB 編碼器產生出 10000 組 *ISF* 索引，每組 *ISF* 索引具有 7 個索引值，事前訓練好各個索引值的量化碼字書，提供解碼端解碼的依據。本節實驗將使用每組 *ISF* 索引的第一個索引當作訊號源，意即第二章所介紹的索引值 ID1，經過指定矩陣的對應每個索引值可以對應其列索引以及行索引，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}。列索引及行索引分別經過不同的通道傳送至接收端。模擬通道則使用位元錯誤通道環境，為了簡化模擬實驗，設定兩個通道的位元錯誤率是一致的。接收端接受來自兩個通道的資訊之後，隨即啟動多重描述解碼器，配合事前訓練的量化碼書執行解碼的動作。



5.1.2 結果分析

不同的指定矩陣所對應的列索引及行索引存在明顯差異，因而反應在解碼端的效能也有所不同。因此本實驗利用二元置換演算法持續做運算，並記錄每次置換之後的指定矩陣。而系統效能的評量方式則是根據每個索引在事前訓練的量化碼書所對應的碼字，和在多重描述解碼器輸出的結果做比較，最後再總合計算其訊雜比 (SNR)。實際的運算如下式：

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

其中 b_i 是第 i 個索引值在量化碼書所對應的碼字，而 output_i 則代表解碼端的輸出結果。執行二元置換演算法，逐次紀錄置換過後的指定矩陣，根據該指定矩陣做上述的系統傳輸動作，並計算此系統的訊雜比。實驗模擬結果如圖 5.1 所示，我們發現二元置換演算法在置換的過程中，確實可以逐次達到匹配於當時錯誤環境的最佳化指定矩陣。

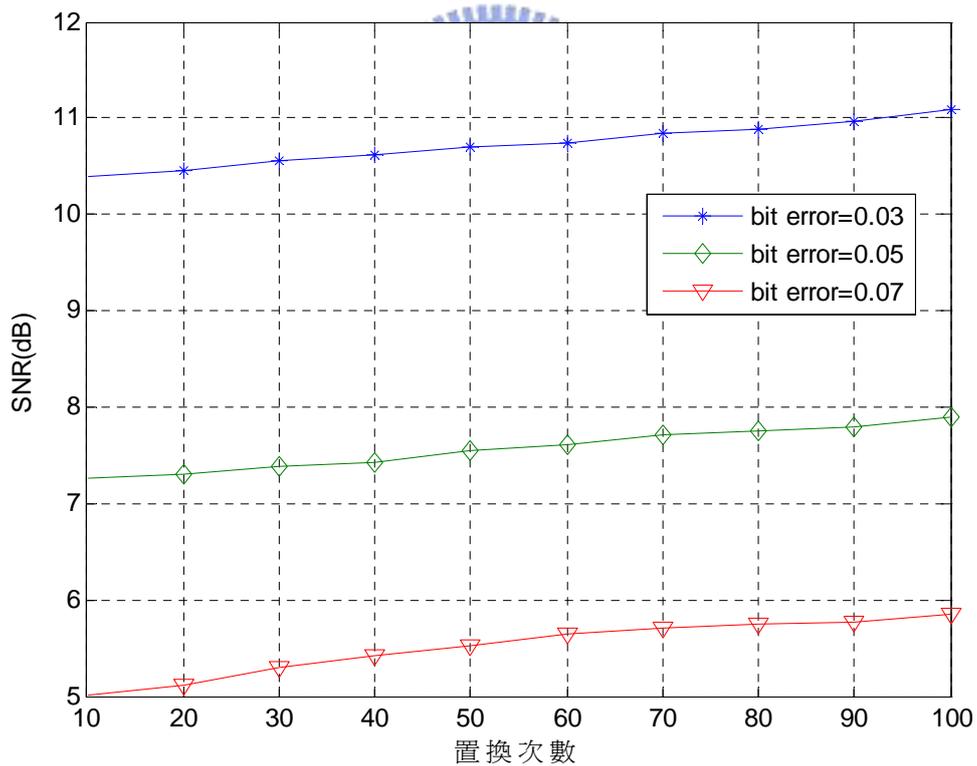


圖 5.1 二元置換過程的訊雜比

5.2 兩階段索引指定的實驗

5.2.1 系統模擬之步驟說明

本節實驗驗證第三章所介紹的方法，於指定矩陣中事先規劃好可以形成合法碼字的位置，再使用二元置換演算法尋找該通道環境之下的最佳排列方式。值得注意的是，為了公平比較位元錯誤通道以及 AWGN 通道對系統的影響，我們利用二位元相位鍵移 (BPSK) 調變的理想錯誤率公式：

$$BER = \frac{1}{2} \operatorname{erfc}\left(\sqrt{\frac{E_b}{N_0}}\right) \quad (5.2)$$

反推出各種不同的訊雜比之下的位元錯誤率，表 5.1 列出兩者之間的對應關係。

位元錯誤率	訊雜比($\frac{E_b}{N_0}$)
0.01	4.323dB
0.02	3.241dB
0.03	2.477dB
0.04	1.850dB
0.05	1.310dB
0.06	0.821dB
0.07	0.367dB
0.08	-0.057dB
0.09	-0.464dB
0.10	-0.856dB

表 5.1 位元錯誤率與訊雜比的對應關係

訊號源和第一節實驗使用的相同，第二章所介紹的量化索引值 ID1，經過根據不同的通道環境預先設計的指定矩陣，通過錯誤通道環境之後，在解碼端對接收訊號做解碼的動作。如果是位元錯誤通道，則會先經過查核矩陣的檢查之後再做多重描述解碼器的解碼動作，如果是 AWGN 通道，則直接利用柵狀解碼器做解碼的動作，其平均值設定為 0，變異數則根據上表的對照表作改變。

5.2.2 結果分析



採用訊雜比衡量三種不同的系統設計概念，第一種設計 MDC1 即第二章所介紹的系統架構，而 MDC2、MDC3 則分別為第三章以及第四章所介紹的系統架構。後兩種系統因為具有一定程度的錯誤更正的能力，我們預期其系統效能會勝於 MDC1 系統，實驗結果如圖 5.2 所示。MDC1 與 MDC2 兩個系統在錯誤率較高的環境下，其解碼效能其實非常接近，這是因為通道錯誤超出 MDC2 的查核矩陣可以更正的能力範圍。MDC3 系統則因為考量在軟性領域之下，所有的解碼並沒有事先做硬性的判別，同時具備錯誤更正的能力，使得其效能在各種錯誤環境之下都明顯優於其他兩個系統。

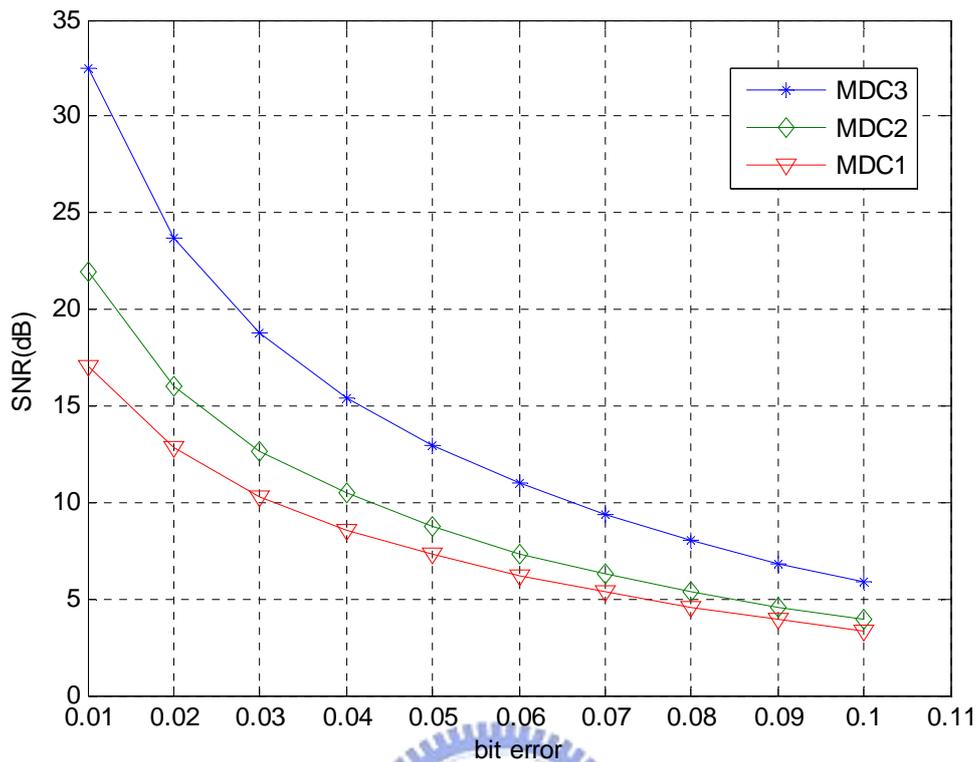


圖 5.2 MDC1、MDC2 及 MDC3 訊雜比比較圖

5.3 AMR-WB 應用的實驗

5.3.1 系統模擬之步驟說明

我們將三種不同的設計概念實際應用在 AMR-WB 系統之上，初步鎖定在導納頻譜頻率 (ISF) 係數的重建。針對語料編碼所得的 10000 組 ISF 索引，每組 ISF 索引取其 ID1、ID2 以及 ID3 三個索引值，分別依表 5.2 的位元分配去執行多重描述向量量化。

INDEX	M(bits)	N
ID1	8	64
ID2	8	64
ID3	6	32

表 5.2 ISF 係數前 3 個索引的位元分配

至於其餘的 4 個索引值，則未執行多重描述向量量化的動作，而將 ID4 以及 ID6 在第一個通道做傳輸，而 ID5 以及 ID7 則是在另一個通道做傳輸。若有一通道資訊發生錯誤，由另一通道做補償的動作；若同時發生錯誤，則會運用前一時刻的資訊做補償的動作。MDC1 及 MDC2 設計系統是在位元錯誤通道環境之下做測試，而 MDC3 系統是在 AWGN 通道環境之下做測試。為了公平比較，AWGN 雜訊的平均值設定為 0，而變異數則依據表 5.1 對照做調整。為了方便呈現模擬結果，設定兩個通道的錯誤環境一致。接收端分別對收到的資訊做解碼動作而輸出 7 個索引值，再綜合其他 AMR-WB 編碼參數以還原解碼之後的語音檔。



5.3.2 結果分析

第一節以及第二節的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：

$$SD_n = \sqrt{\frac{1}{f_2 - f_1} \cdot \int_{f_1}^{f_2} \left[10 \log_{10}(\tilde{P}_n(f)) - 10 \log_{10}(\hat{P}_n(f)) \right]^2 df} \quad (5.3)$$

其中 $f_1 = 50$ 、 $f_2 = 6400$ 。量化後與重建的功率頻譜密度為

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

$\tilde{A}_n(z)$ 與 $\hat{A}_n(z)$ 為量化後與重建的 LPC 濾波器， $F_s = 12800$ 。實驗結果如圖 5.3 所示，結果分析發現隨者環境越惡劣，失真度也逐漸的上升。在相同的錯誤環境之下，MDC1 及 MDC2 系統因包含錯誤更正的能力，在系統效能的衡量標準上都優於 MDC1，而 MDC3 系統不但具備錯誤更正的能力，又因為考量在軟性領域之下做解碼的動作，因此又更優於 MDC2 系統。

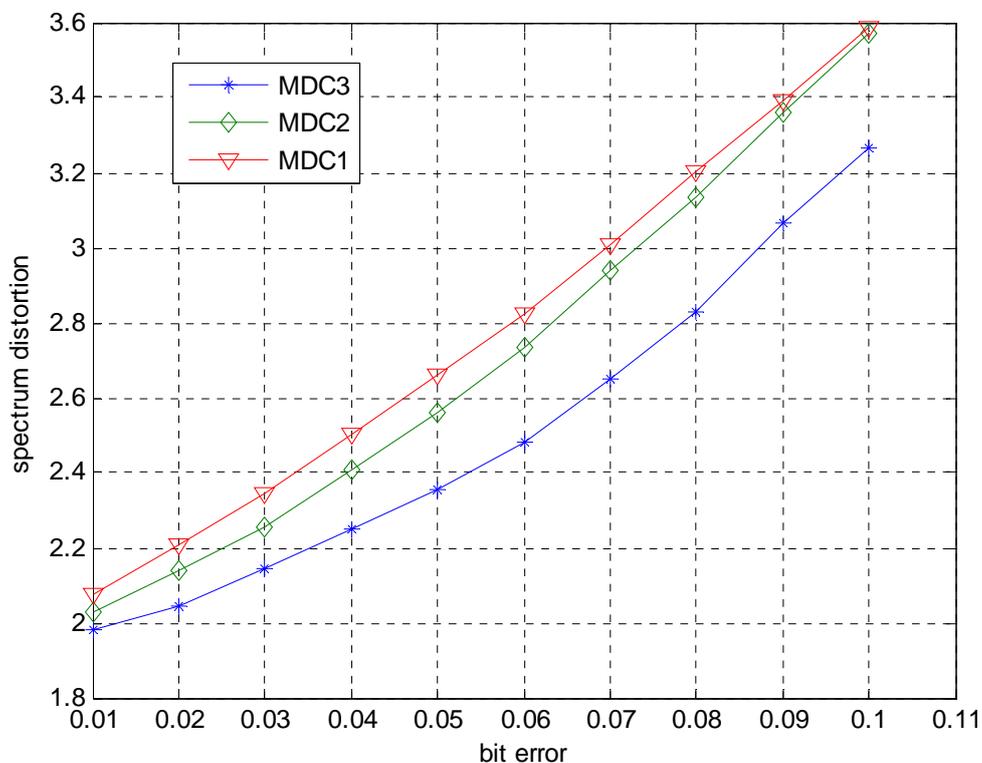


圖 5.3 AMR-WB 在不同索引指定的頻譜失真

AMR-WB 系統的整體解碼效能還有另外一種衡量方式：音質感知評估(Perceptual Evaluation of Speech Quality ; PESQ)，比較原始與解碼重

建的語音檔，評量分數(Mean Opinion Score, MOS)範圍從 0 到 4.5。評分 4.5 代表重建的音質與原音檔非常接近，0 則代表重建語音品質非常糟糕。實驗結果如圖 5.4 所示，我們發現 PESQ-MOS 所呈現的走勢和平均頻譜失真是一致的，音質好壞依序為 MDC3>MDC2>MDC1，而環境越惡劣時語音品質也相對越壞。

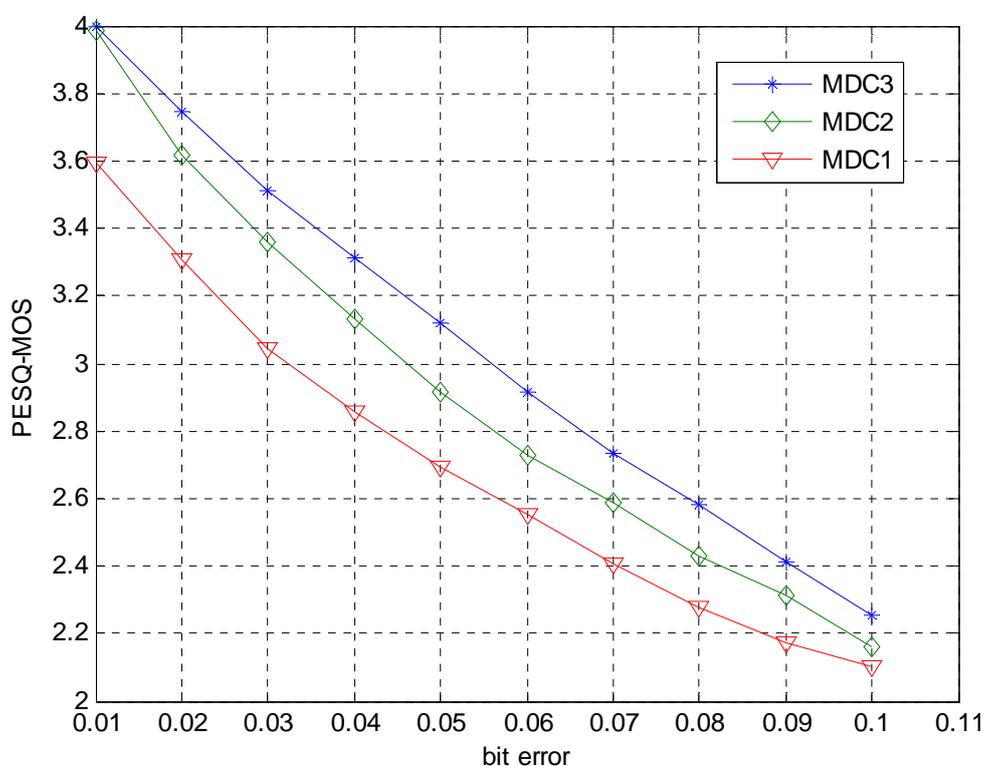


圖 5.4 AMR-WB 在不同索引指定的音質評估

第六章

結論與未來展望

不同的應用服務所對應的通道模型存在明顯的差異，例如位元錯誤通道、可加性高斯白雜訊通道、網路封包漏失通道以及雷利衰弱通道(Rayleigh fading channel)。多重描述量化原本是針對網路封包漏失而發展的訊源編碼技術，本論文將此技術推廣應用於行動通訊常見的位元錯誤通道以及 AWGN 通道。在多重描述量化中指定索引的設計扮演重要的角色，最佳化的索引指定設計有助提升解碼的效果。本論文主要的構想是將索引指定分解為位置指定與位元對應兩階段，在指定索引的同時預先決定好特定位置，讓對應的列索引以及行索引可以合併成為一個碼字。此結構於解碼端可更正一定程度的位元錯誤，進而有效提升解碼輸出的語音品質。在第五章的實驗模擬證實，我們提出的兩階段索引指定設計，不但大幅減少二元置換演算法的運算量，在位元錯誤或 AWGN 通道環境解碼輸出的音值也顯著改善。

目前研究是假設位元錯誤呈現隨機分佈，但這並不符合真實通道環境的叢發性位元錯誤。此外，我們只考量兩個通道下傳輸，未來可將此系統擴展到更多的通道傳輸，對於索引指定上的設計也更靈活。通道的擴展其實和多輸入多輸出(Multiple Input Multiple Output, MIMO)的系統架構有其相同之處，而雷利衰弱通道更是無線通

訊環境中最常使用的錯誤通道。為了抵抗 MIMO 通道環境的衰弱而產生的錯誤，時空碼是目前最常使用的錯誤更正碼，未來可以考慮在指定索引的同時，在位置的選擇上建立此錯誤更正的碼字結構。



參考文獻

- [1] V. A. Vaishampayan, "Design of multiple description scalar quantizers," *IEEE Trans. Inform. Theory*, vol. 39, pp. 821-834, May 1993.
- [2] Y. Zhou and W.-Y. Chan, "Multiple description quantizer design using a channel optimized quantizer approach," in *Proceedings of the 38th Annual Conference on Information Sciences and systems*, 2004.
- [3] P. Yahampath, "On index assignment and the design of multiple description qiantizers," in *Proceedings of 2004 International Conference on Acoustics, Speech, and Signal Processing*, Montreal, Quebec, Canada, vol. 4, pp. iv-597-iv-600, May 2004. 47
- [4] K. Zeger and A. Gersho, "Pseudo-Gray coding," *IEEE Trans. Commun.*, vol. 38,no. 12, pp. 2147-2158, Dec. 1990.
- [5] N. Gortz and P. Leelapornchai, "Optimization of the index assignments for multiple description vector quantizers," *IEEE Trans. Commun.*, vol. 51, no. 3, pp.336-340, Mar. 2003.
- [6] Rui Ma and Fabrice Labeau, "Robust index assignment for MDSQ encoder over noisy channels," *Proc. MMSP'06*, pp. 286–290, Oct. 2006.
- [7] Rui Ma and Farbrice Labeau, "Enhanced multiple description decoder for errorprone channels," 2006, accepted by ICIP '06.

[8]S. Lin and D. J. Costello, Jr., *Error Control Coding*. Englewood Cliffs, NJ: Prentice-Hall, 2004.

[9]L. Hanzo, F. C. A. Somerville, and J. P. Woodard, *Voice Compression and Communications*. New York: Wiley, 2001.

[10]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.

[11]Yugang Zhou,"Analysis and Design of Multiple Description Codes For Wired And Wireless Channels," qspace.library.queensu.ca , pp68-101 , September 2007

