

國立交通大學

電信工程研究所

碩士論文

雙向性可變長度碼的軟性訊源解碼機制
Soft-Decision Decoding of Reversible
Variable-Length-Codes

研究生：洪肇遠

指導教授：張文輝 博士

中華民國一百年六月

雙向性可變長度碼的軟性訊源解碼機制

Soft-Decision Decoding of Reversible Variable-Length-Codes

研究 生：洪肇遠

Student : Chao-Yuan Hung

指 導 教 授：張文輝

Advisor : Wen-Whei Chang



National Chiao Tung University

in partial Fulfillment of the Requirements

for the Degree of

Master

in

Computer and Information Science

June 2011

Hsinchu, Taiwan, Republic of China

中華民國一百年六月

雙向性可變長度碼的軟性訊源解碼機制

學生：洪肇遠

指導教授：張文輝 博士

國立交通大學電信工程研究所碩士班



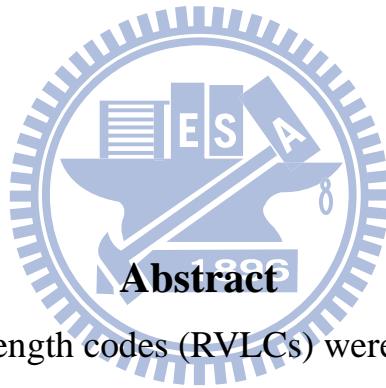
近年來多媒體通訊的快速發展，使得針對影音資料的壓縮已成為重要的研究課題，特別是多項國際標準制定的可變長度碼。本論文主要探討可變長度碼的軟性訊源解碼機制，並將其應用於心電圖訊號的壓縮。首先針對可變長度碼在索引層級的籬柵結構，推導其軟性訊源解碼演算法，並整合其軟性輸出與籬柵結構以發展基於維特比演算法的索引估測方法。其次，針對具有特定機率分佈的訊源，可建構其霍夫曼碼，並以此發展具有錯誤更正能力的雙向性可變長度碼。我們將引入訊源解碼的軟性資訊，配合一階相關性的事前資訊進行索引指定的設計。系統模擬證實我們提出的方法能有效對抗通道雜訊干擾，進而提升系統的強健性。

Soft-Decision Decoding of Reversible Variable-Length-Codes

Student : Chao-Yuan Hung Advisor : Dr. Wen-Whei Chang

Institute of Communications Engineering

National Chiao Tung University



Reversible variable length codes (RVLCs) were proposed for facilitating the bidirectional decoding, which mitigate the transmission errors in case of losing synchronization between the encode and decoder. This thesis focuses on the RVLC design and its application to ECG signal compression. Our work starts with a soft-decision source decoding algorithm based on a symbol-level trellis also proposed is a viterbi algorithm which exploit the soft output of source decoder to better estimate the transmitted index. For construction of RVLC, we proposed an index assignment scheme based on the channel-related information and the first-order correlation of source. Simulation results show that the proposed scheme can generate RVLC of higher robustness in comparison to other construct algorithms.

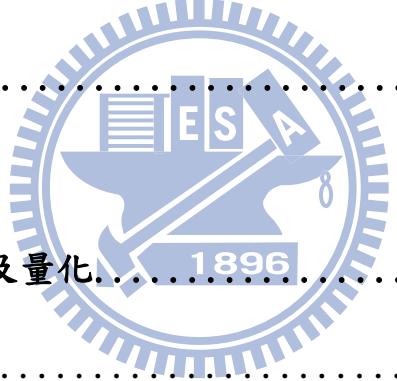
誌謝

得以完成本篇論文，首先要感謝我的指導教授張文輝老師，就讀研究所兩年中，老師一對一地細心指導與循循善誘，使我瞭解做研究所應有的態度和方法，並於其中獲得許多寶貴的專業知識。

其次要感謝在這兩年中從旁協助我研究的吳俊鋒學長，學長總能不厭其煩地解答我許多生活與研究上的問題。也要感謝吳鴻材、徐子凡、陳柏強三位同學這兩年的幫助與照顧，三位學弟王韋超、林志賢、曾鴻竣一年來的陪伴。最後要特別感謝我的父母與求學路上一路相伴的女朋友，沒有他們的支持，將讓我無法如此順利地完成學業，謹以此論文獻給所有照顧過我及我愛的人。



目錄

中文摘要.....	i
英文摘要.....	ii
致謝.....	iii
目錄.....	iv
圖目錄.....	vi
表目錄.....	vii
第一章 緒論.....	1
 The logo of the Electrical Society of China (ESCA) is centered on the page. It features a circular border with the text "中国电气学会" (Chinese Electrical Society) in Chinese characters. Inside the circle, there is a stylized representation of a power transmission system, including a central tower with "ESCA" written on it, flanked by two arrows pointing upwards, and a base with the year "1896".	
第二章 心電圖簡介及量化.....	5
2.1 心電圖簡介.....	5
2.2 心電圖量測.....	7
2.3 心電訊號的量化.....	10
第三章 索引層級軟性訊源解碼機制.....	15
3.1 系統架構.....	16
3.2 索引層級籬柵圖.....	17
3.3 索引後驗機率的推導.....	18

3.4 索引估算.....	21
第四章 雙向性可變長度碼之設計.....	23
4.1 雙向性可變長度碼.....	24
4.2 基於最小平均長度之設計.....	25
4.3 基於錯誤更正能力之設計.....	29
4.4 基於解碼效能之索引指定設計.....	32
第五章 實驗模擬與結果分析.....	38
5.1 實驗環境設定.....	38
5.2 實驗一.....	39
5.3 實驗二.....	43
5.4 實驗三.....	47
第六章 結論與結果展望.....	50
參考文獻.....	52

圖目錄

圖 2.1 心電圖之 PQRST 波.....	7
圖 2.2 Einthoven 三角形.....	8
圖 2.3 單極胸前導程電極.....	9
圖 2.4 MIT-BIH 編號 100 心電圖.....	11
圖 2.5 向量量化誤差($K=2$ 及 $M=4$).....	12
圖 2.6 向量量化誤差($K=2$ 及 $M=6$)	12
圖 2.7 向量量化誤差($K=4$ 及 $M=4$)	13
圖 2.8 向量量化誤差($K=4$ 及 $M=6$)	13
圖 3.1 可變長度碼之傳輸模型.....	16
圖 3.2 索引層級籬柵圖.....	18
圖 4.1 位元樹狀圖.....	24
圖 4.2 二位元置換演算法.....	37
圖 5.1 四種可變長度碼之軟性訊源解碼效能.....	42
圖 5.2 不同通道環境訓練所得之索引指定的效能比較.....	46
圖 5.3 $RVLC_1$ 之不同索引檢測方式比較.....	47
圖 5.4 $RVLC_2$ 之不同索引檢測方式比較.....	48
圖 5.5 $RVLC_3$ 之不同索引檢測方式比較.....	48

表目錄

表 2.1 MIT-BIH 編號 100 心電圖向量量化的 PRD 值.....	14
表 4.1 第三層碼字之可用候選碼字表.....	27
表 5.1 不同可變長度碼設計之碼書.....	42
表 5.2 與通道環境匹配之 $RVLC_3$ 碼書.....	45



第一章 緒論

1.1 研究動機與方向

隨著生活水準及醫療服務的日趨進步，高齡化將是現代人所要面臨的一個趨勢，許多先進國家也開始發展遠距醫療照護(wireless telemedicine)相關的服務。在台灣因為山脈地形的關係，容易造成城鄉醫療資源的差距，許多住在偏遠地區的民眾必須花費更多的時間與金錢才能獲得醫療品質的保障。遠距醫療照護將可縮短這些落差，例如澎湖離島地區已運用遠距視訊系統，配合醫生的同步會診以克服醫療人手不足的問題。基於預防勝於治療的觀念，遠距居家照護(Mobile e-Health)也成為醫療照護的重要項目。主要是利用隨身的無線感測器，隨時隨地偵測病患的重要生理訊號，如心電圖、腦電波、體溫…等等，並傳至醫院中讓醫生得以隨時監控此病人的身體狀況。

長久以來，心臟病一直是國人的常見死因，心電圖的偵測對患者而言尤為重要，因此本論文將針對心電圖深入探討。

隨時隨地監控患者所量測的心電圖訊號，其所需傳送的資料量是非常龐大的，且有即時性的要求，因此如何更有效率傳送這些資料成為一個重要的議題。由於多媒體通訊的相關服務蓬勃發展，國際標準化組織已制定發表了許多的影音資料壓縮標準，例如 MPEG 系列、

H.264…等且大部分皆使用了可變長度碼(Variable Length Code, VLC)作為編碼機制的核心。可變長度碼的壓縮率接近理論熵值，但常受限於傳輸錯誤而發生同步失序的問題。可變長度碼是利用訊源中資料機率分布不平均的特性，讓出現機率較大的索引使用較短的碼字而發生機率較小的索引則用較長的碼字，因此碼字平均長度將會短於固定長度碼(fixed length code)，其中最為人熟知的是霍夫曼碼(Huffman Code)，而不同的資料型態及傳輸目的也會有不同的編碼方法。可變長度碼因其傳遞每個索引時，用以編碼的位元數不如固定長度碼般的固定，因此對位元錯誤非常敏感，單一個位元錯誤就可能使得接下來連續性索引誤判。一個可行的解決方案是開發新的訊源解碼機制，利用訊源編碼之後殘存的冗息(residual redundancy)，除了希望能將低位元錯誤率外，也希望能控制連續性的索引錯誤。雙向性可變長度碼(Reversible VLC, RVLC)兼具前向與後向的解碼能力，可定位連續錯誤發生的位置，使其漸漸在標準中取代可變長度碼的地位。因此本論文的另一個研究方向是設計雙向性可變長度碼，使其在傳輸過程中有效提升系統的強健性。

軟性訊源解碼機制(soft-bit source decoding, SBSD)在[1]被提出，利用殘餘冗息使得訊源解碼器得以隱藏通道解碼器無法消除的殘餘錯誤，其處理模式可以分為位元[2]與索引層級[3]的可變長度

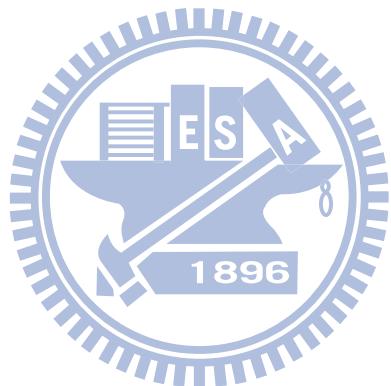
解碼機制。在前人研究[2]中，利用碼字間位元狀態的轉移，建構其樹狀圖(code tree)與籬柵圖(code trellis)，並以 BCJR(Bahl-Cocke-Jelinek-Raviv)演算法計算每個位元的後驗機率(A posteriori probability, APP)，藉以估測所傳遞的位元。而文獻[3]則在假設傳送端的索引與位元數目已知的前提下，以此建立索引層級的籬柵圖以計算每個索引的後驗機率，再透過最大後驗機率原則(Maximum a posteriori, MAP)估測傳送之索引。

在[4]中，首次提出了可提供即時性前向與後向解碼的雙向性可變長度碼。文獻[5]則是針對對稱性與非對稱性 RVLC 分別提出最小平均長度的演算法，並進一步設計一組兼具錯誤更正能力與最小平均長度的 RVLC[6]。[7]希望能利用索引指定，改變索引與對應碼字之間的關係，以提升訊源解碼的效能。同時提供一個找尋其區域最佳值(local optimal)的演算法，降低所需複雜度。

1.2 章節概要

第二章將介紹心電圖訊號的量測與特性，以及基於向量量化(vector quantization)的壓縮模式。第三章則介紹本論文所使用的傳輸系統及其軟性訊源解碼器，主要是利用索引層級籬柵圖提出一個軟性輸出的維特比演算法(viterbi)。第四章詳述設計非對稱性 RVLC

的演算法，其中包括基於最小長度與具有錯誤更正能力兩種設計，並以此延伸提出符合訊源解碼器的索引指定設計。第五章將提供實驗數據及其結果分析，第六章則為結論與未來展望。



第二章 心電圖簡介及量化

遠距居家照護的目標是希望能隨時監控在家病人的生理狀況，透過生理訊號的感測並將資料傳送至醫院，以便醫生得知病人是否有突發性狀況。根據行政院衛生署國民健康局的調查發現，國內 65 歲以上民眾幾乎每四人就有一人曾罹患心臟病，而急性心肌梗塞更持續高居台灣十大死因的第二位，因此心臟相關疾病的預防是一項重要課題。急性心肌梗塞病因是冠狀動脈血液供應急劇減少，以致心肌發生嚴重的急性缺血，而臨床表現症狀為心律不整或心臟衰竭。心電圖儀是評估心臟循環功能的主要工具，有助於診斷病患的心律不整症狀，如心跳速率、規則性、電脈衝的發源地及傳導路徑的異常。不間斷地傳送心電圖需要龐大的資料量，在本章中我們將透過對個別患者的心電圖進行訓練與向量量化，以期能壓縮傳送的資料量並兼顧其重建品質。

2.1 心電圖簡介

心電圖(Electrocardiography, ECG)是記錄心臟組織電壓變化的
一個圖形，心臟的肌肉是人體肌肉中，唯一具有自發性跳動及節律性
收縮的肌肉。心臟傳導系統發出電波，刺激整個肌肉纖維而產生收

縮。電波的產生及傳導，皆會產生微弱的電流分佈全身，若將心電儀的電極連接到身上不同的部位，就可記錄而得心電圖。心電圖可檢驗下列八種情況：

- (1) 心房肥大及心室肥大。
- (2) 心房及心室電波的傳導遲延。
- (3) 心肌缺血及心肌梗塞。
- (4) 判定心律不整來源及監視它的活動情形。
- (5) 心包膜炎。
- (6) 侵犯心臟的全身性疾病。
- (7) 判定心臟藥物的影響。
- (8) 電解質代謝障礙，特別是鉀¹⁸⁹⁶



心電圖訊號屬於低頻範圍 0.05 至 100HZ，且振幅微小僅 1 至 10mV，因此在心電圖訊號的擷取上便需要訊號放大器與濾波裝置。市面上常見的一般心電儀，其靈敏度便是 0.25mV，記錄的頻率範圍為 0.5HZ 至 80HZ，臨床上用以測量 ECG 訊號的電極通常為銀 / 氯化銀電極，其電位特性、雜訊以及回復特性適合用於生理訊號的感測。

美國 MIT-BIH 醫療機構公開提供的心電圖資料庫，已廣泛應用於心電訊號處理分析的相關研究。心跳的一個週期如圖 2.1 所示，起始於竇房結發出每秒約 60 次的電脈衝，以漸進方式傳遞至左右心房引

起收縮(P波)，P波區間通常不大於0.12秒且高度不大於2.5mv，抵達房室結後停滯約0.1秒讓血液充分流至心室(Q波)，接著藉由傳遞纖維傳至左右心室而引起收縮(R波)，之後心臟暫時靜止直到心室再極化(T波)。整體而言，一個心電圖週期以QRS波最為顯著，一個QRS複合波週期大約介於0.08至0.12秒之間，其位置的自動化偵測能提供病患心律不整的診斷資訊。

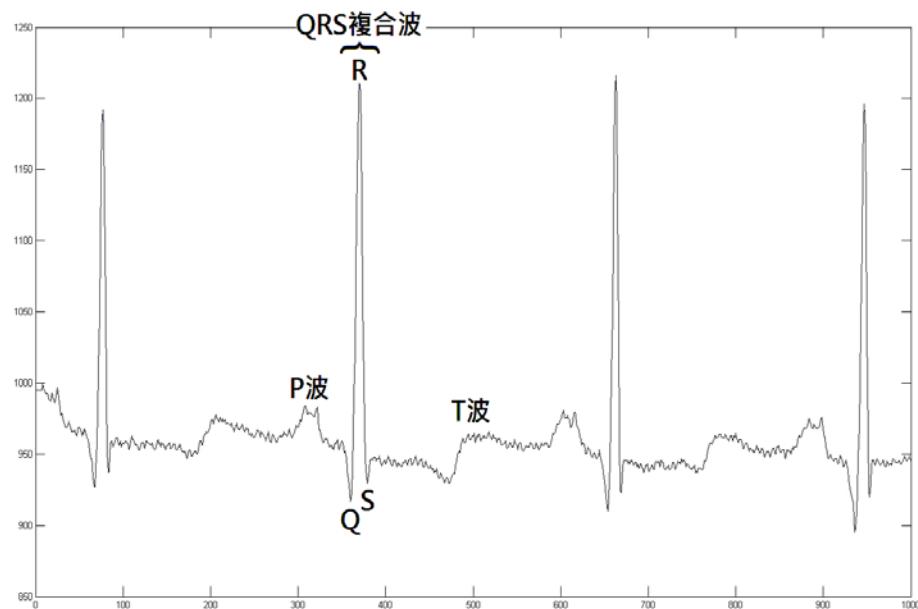
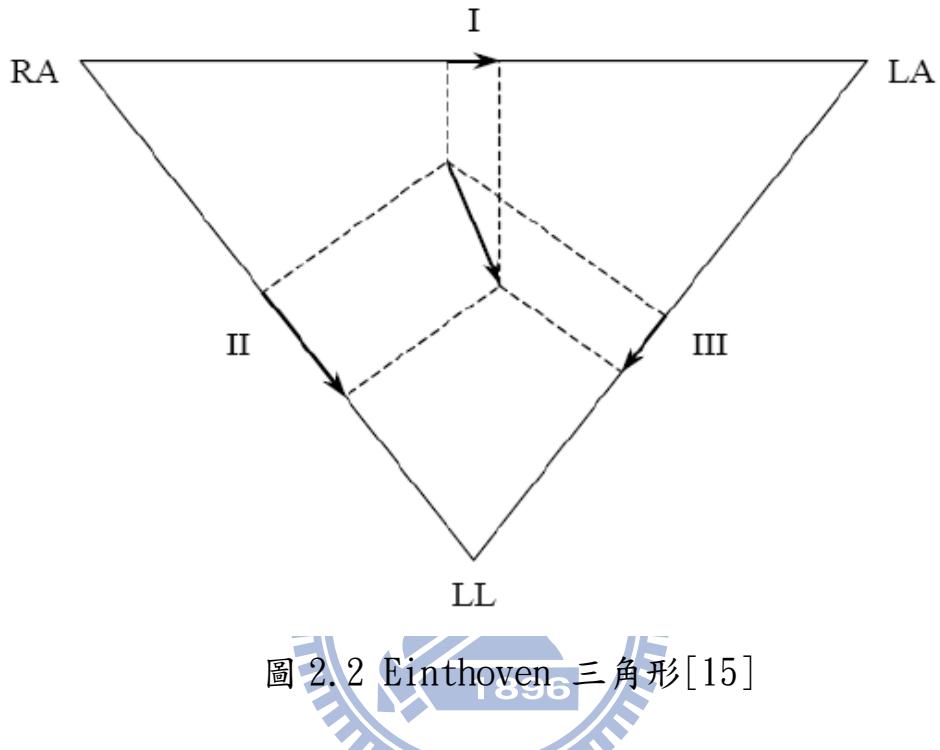


圖 2.1 心電圖 PQRST 波

2.2 心電圖量測

心電圖為身體表面記錄之心臟週期性電氣活動的電位變化，現階段臨牀上使用最多的為十二導程心電圖(12 leads ECG)，分為胸導程與肢端導程兩類。心電圖的六個單極胸前導程分別為V1、V2、V3、

V4、V5、V6，能夠從水平面上測得心臟的狀況。而來自四肢電極的情況，則可由六個肢端導程 I、II、III、aVR、aVL、aVF 從心臟垂直面測得，這十二個導程將整合成標準的心電圖。



六個肢端導程又細分為兩類，aVR、aVL、aVF 為單極肢端導程 (Unipolar limb Leads)，I、II、III 為標準雙極肢端導程 (Bipolar Standard Leads)，量測方法如下：

- Lead I 肢端導程：左手為正極，右手為負極，角度為 0 度，即
 $Lead\ I = LA - RA$ (左手電壓與右手電壓之電位差)。
- Lead II 肢端導程：左腳為正極，右手為負極，角度為 60 度，即
 $Lead\ II = LL - RA$ (左腳電壓與右手電壓之電位差)。
- Lead III 肢端導程：左腳為正極，左手為負極，角度為 120 度，

即 Lead III = LL - LA(左腳電壓與左手電壓之電位差)。

此三個導程之向量關係可用 $II = I + III$ 表示，相關位置如圖 2.2 所示。

- Lead aVL 肢端導程：左手為正極，左腳與右手為負極，角度為 -30 度， $aVL = I - (II/2)$ 。
- Lead aVR 肢端導程：右手為正極，左腳與左手為負極，角度為 -150 度， $aVR = -(I+II)/2$ 。
- Lead aVF 肢端導程：左腳為正極，右手與左手為負極，角度為 90 度， $aVF = II - (I/2)$ 。

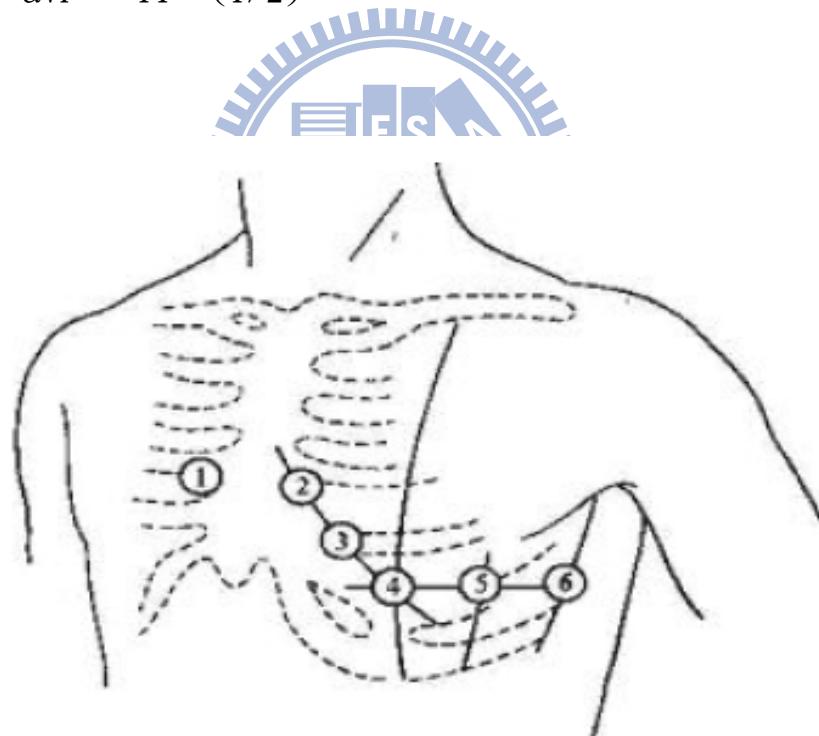


圖 2.3 單極胸前導程電極[14]

六個胸前導程電極位置如圖 2.3 所示，量測方法如下：

- V1 是放在胸骨右側的第四肋間。
- V2 是放在胸骨左側的第四肋間。

- V3 是放在 V2 和 V4 之間。
- V4 是放在第五肋間的左鎖骨中線之處。
- V5 是放在 V4 和 V6 之間。
- V6 是放在第五肋間的腋中線(midaxillary line)之處。

2.3 心電訊號的量化

量化處理為數位傳輸系統裡很重要的一環，因取樣過後的訊源取樣點其實數值的範圍是無限的。經過量化步驟後，可將這些數值分隔成若干個區域，分別以一個代表位階來表示，如此一來，所有的數值皆可以用有限的位階來代替，傳輸上也只要傳送這些代表數值的索引即可。向量量化(vector quantization)為資料壓縮領域的重要方法，因其具有良好的壓縮率與較小的失真。以 K 維 M 位元的向量量化為例，將訊源取樣點依序以 K 點組成一個向量，將此 K 維空間分隔成 2^M 區域，每個區域皆有一個事前訓練所得而具有代表性的碼向量，每個輸入向量將被劃分至與碼向量之失真(distortion)最小的區域。失真的評估方式很多，常見的是以較直觀的平方差為準。最原始的向量量化演算法在 1980 年由 Y. Linde、A. Buzo 和 R. Gray 共同提出[13]，後人稱其為 LBG 演算法。流程如下：

1. 計算所有向量的重心，並以此點為初始的代表向量。

2. 將代表向量分別加減一個特定向量後，分裂為兩個新代表向量。
 3. 所有的訓練向量皆計算其與代表向量間的失真，並劃分在與其失真最小的群。
 4. 分別計算各個分群裡向量的重心，若各個分群裡重心與代表向量的差異收斂在一極小的範圍內，則回到第 2 步驟，反之，則以新重心為代表向量，接著執行第 3 步驟。直至代表向量個數為所需個數，且分群重心皆收斂，演算法結束。
- 量化機制皆會產生量化誤差，雖可以透過碼書的設計來降低，但仍舊有其限制。圖 2.4 為 MIT-BIH 資料庫編號 100 的心電圖，圖 2.5 至 2.8 為其分別依不同 K 與 M 值做向量量化所產生的誤差圖。一般而言，維度 K 越小或碼書尺寸 2^M 越大，失真就越小，但如此一來，也會造成壓縮率的降低。

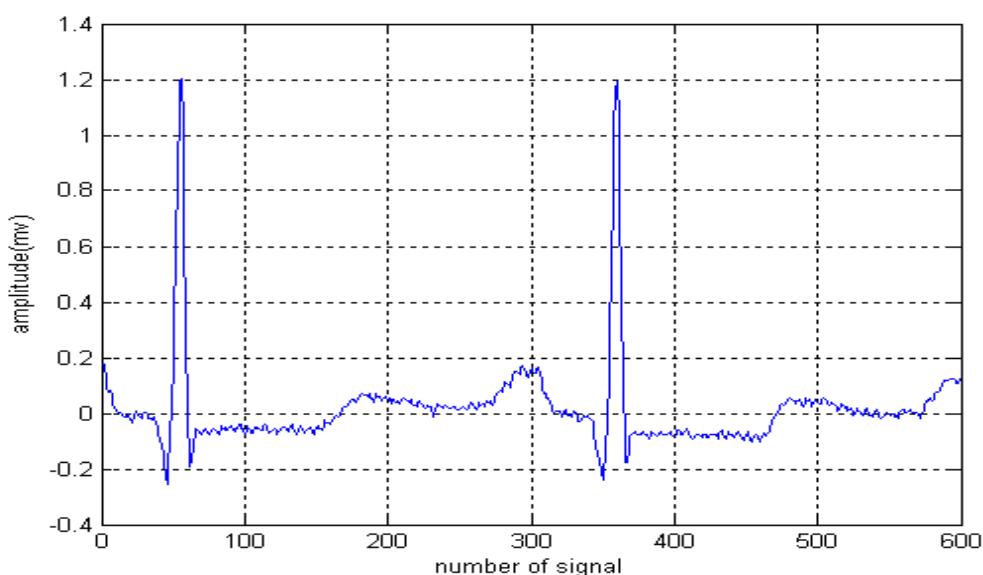


圖 2.4 MIT-BIH 編號 100 心電圖

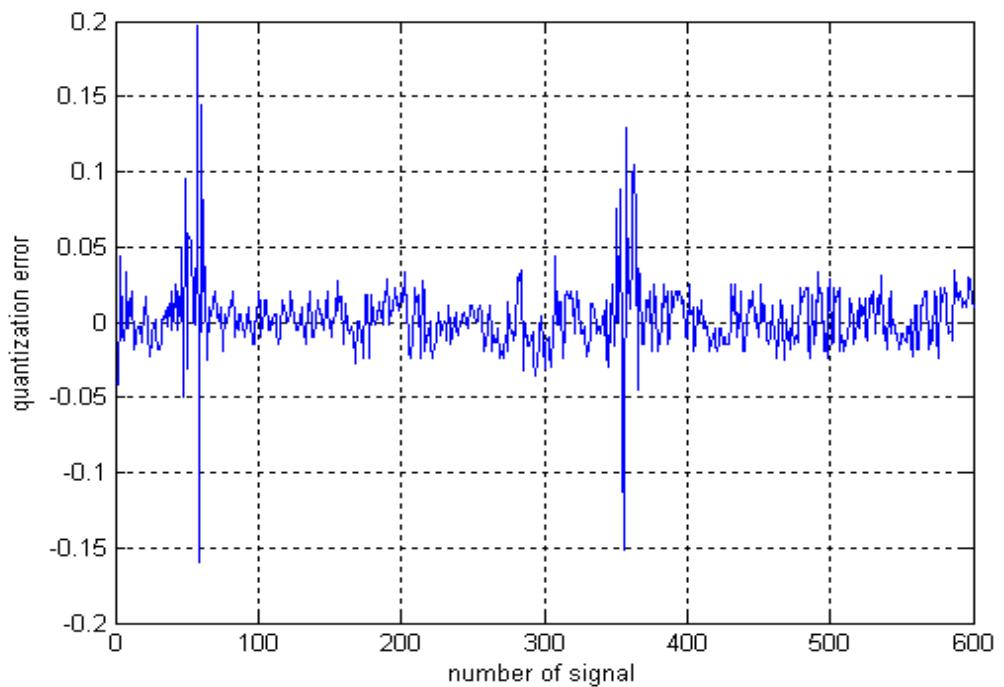


圖 2.5 向量量化誤差($K=2$ 及 $M=4$)

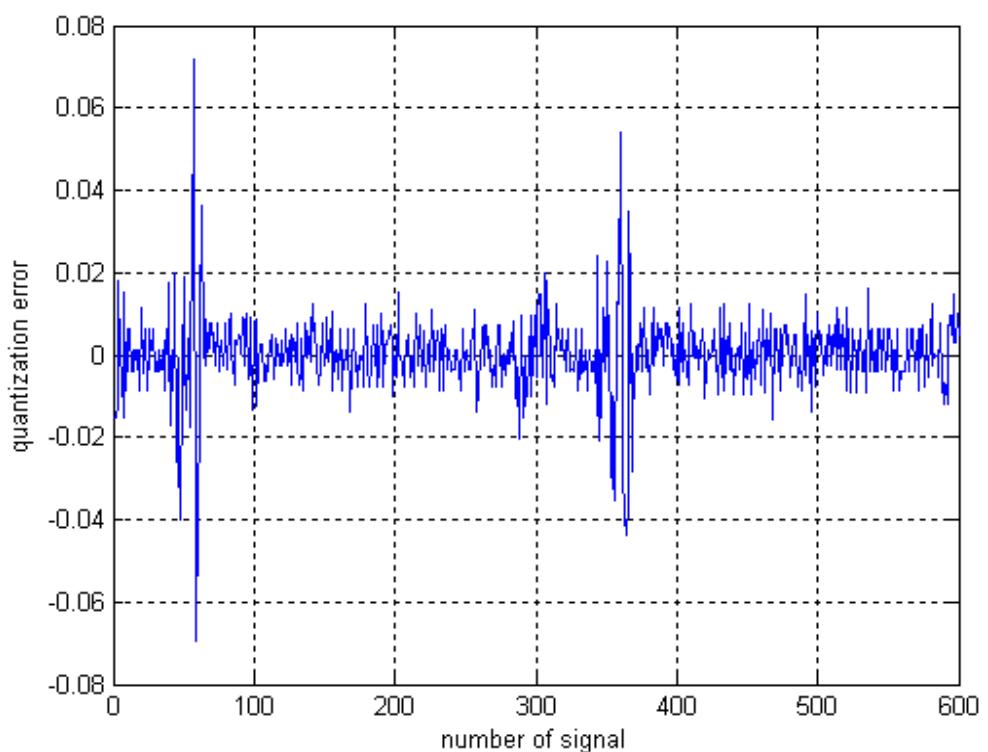


圖 2.6 向量量化誤差($K=2$ 及 $M=6$)

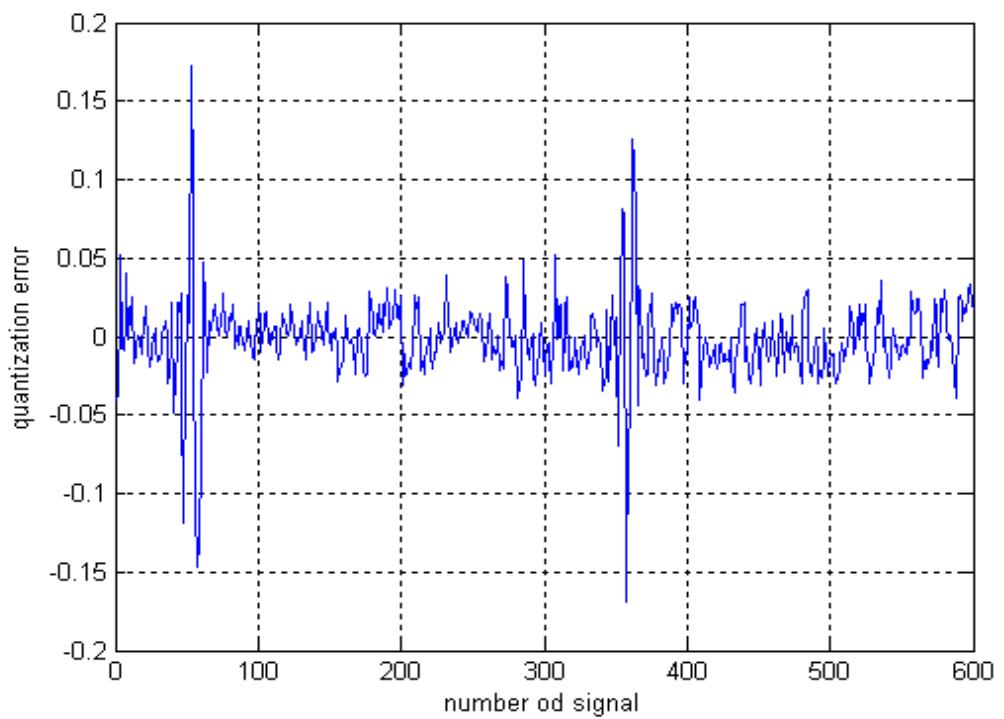


圖 2.7 向量量化誤差($K=4$ 及 $M=4$)

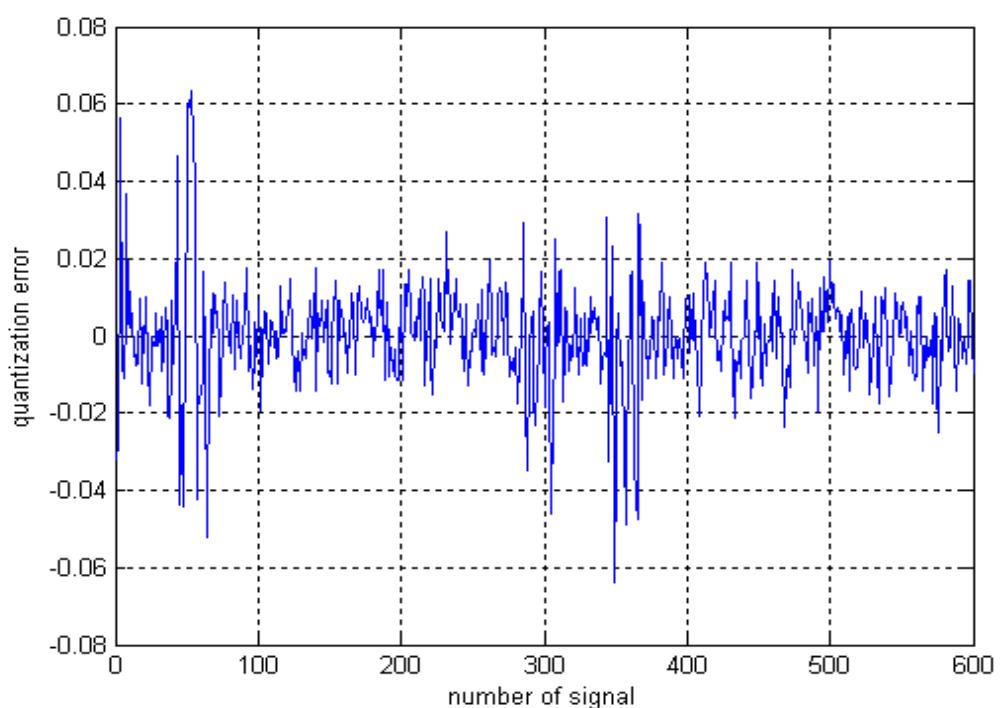


圖 2.8 向量量化誤差($K=4$ 及 $M=6$)

在本論文中將使用 LBG 演算法訓練心電圖的量化碼書，而我們採用的解碼機制之複雜度與 2^M 值大小成正比，但因心電圖的監控有其品質要求，若 M 值太小則使得失真太大，應妥善選擇 K 與 M 的折衷值。心電圖的品質大多以均方根誤差百分比(Percent Root-mean-square Difference, PRD)做為評比的標準，其與訊雜比有一轉換關係，公式如下：

$$PRD = \sqrt{\frac{1}{SNR}} \times 100\% \quad (2.1)$$

從式子(2.1)可得知，SNR 越高則 PRD 越低，量化品質也較好。一般而言，PRD 能低於 10%的心電圖是可接受的重建品質。由表 2.1 可發現達成低標以上的向量量化之(K, M)組合為(K, M)=(2, 5)、(2, 6)、(2. 7)、(4. 6)、(4. 7)，考慮到訊源解碼器複雜度的關係，在進行心電圖向量量化時暫不考慮 $M=7$ 的兩種組合，而在剩存的三種組合中，(K, M)=(2, 6)向量量化的 PRD 值最低，效果最好，故在本論文中對心電圖的向量量化將採用 $K=2$ 、 $M=6$ 的向量量化。

K/M	4	5	6	7
2	14.0328%	9.3334%	6.3987%	4.5674%
4	17.9685%	11.7524%	8.9175%	7.0909%

表 2.1 MIT-BIH 編號 100 心電圖向量量化的 PRD 值

第三章 索引層級軟性訊源解碼機制

在訊源編碼的相關研究中，可分為可變長度碼(variable length code, VLC)與固定長度碼(fixed length code)兩大類別，各有其不同之應用領域。針對具有特定機率分佈的訊號源，相較於固定長度碼而言，可變長度碼在資料壓縮處理上有較好的效率。然而其解碼過程對於錯誤位元相當敏感，少許的位元錯誤便可造成嚴重之索引同步的失序。早期的文獻中，軟性訊源解碼機制主要是針對固定長度碼而開發，其解碼演算法又可細分為位元層級與索引層級兩種，相關演算法也逐漸推廣應用在可變長度碼的軟性解碼機制。位元層級的訊源解碼機制[2]，主要是運用可變長度碼字位元間狀態(state)的轉移關係與其籬柵(trellis)結構，並配合BCJR演算法提供一個可靠的解碼輸出。然而此解碼方式並未充分利用相鄰索引間的相關性，而且在位元轉換成索引的同步失序問題上仍有待解決。索引層級的解碼則是以傳送序列之索引數目與位元數目當做邊訊息(side information)，依此建立其籬柵圖，配合前向與後向遞迴運算(forward-backward recursion)逐步計算每個索引的後驗機率，故可避免位元轉換索引時出現的失序問題。本章節將針對索引層級解碼器做深入探討，推導其解碼之軟性輸出，並以此輸出估算其最大可能性的傳送索引序列。

3.1 系統架構

系統傳輸模型如圖 3.1 所示。 $v_1^L = [v_1, v_2, \dots, v_L]$ 為一具有 L 個訊號源取樣點之序列，經由維度 K 且 M 位元之向量量化器處理後，產生一組長度為 T 之索引序列 $u_1^T = [u_1, u_2, \dots, u_T]$ ，其中 $T = L/K$ 。定義量化索引的集合為 $I = \{0, 1, \dots, 2^M - 1\}$ ，即 $u_t = \lambda \in I$ 。經由事前訓練得其索引的機率分佈 $P(\lambda)$ ，再據以產生一組可變長度碼的碼字 $C = \{c(0), c(1), \dots, c(2^M - 1)\}$ ，其中 $c(\cdot)$ 代表該索引所對應的可變長度碼字。訊源編碼器將輸入之索引序列 u_1^T 依序對應該碼書，接著使用二進位相位鍵移(Binary Phase Shift Keying)調變，產生一長度為 N 位元之二位元序列 $b_1^N = [b_1, b_2, \dots, b_N]$ ，其中第 n 個位元值表示為 $b_n \in \{1, -1\}$ 。經過附加性白高斯雜訊(Additive White Gaussian noise, AWGN)通道後，得到雜訊干擾的實數序列 $\tilde{b}_1^N = [\tilde{b}_1, \tilde{b}_2, \dots, \tilde{b}_N]$ ，令其對應的索引序列為 $\tilde{u}_1^T = [\tilde{u}_1, \tilde{u}_2, \dots, \tilde{u}_T]$ 。在接收端，將序列 \tilde{u}_1^T 經由索引層級訊源解碼器的處理，計算每一個索引值的後驗機率 $P(u_t = \lambda | \tilde{u}_1^T)$ ，並以此軟性輸出資訊進行索引 u_t 的最佳化估算，最後得到索引的估算法 $\hat{u}_1^{T \times \hat{\lambda}} = [u_1, u_2, \dots, u_T]$ ， $\hat{u}_t \in I$ 。

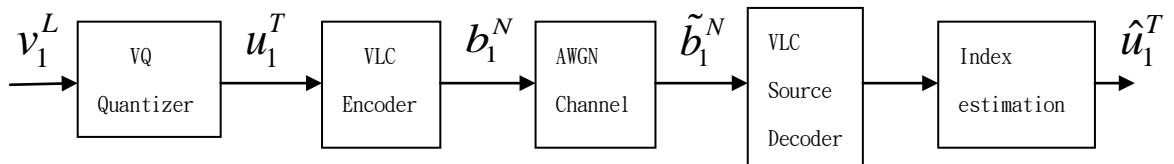


圖 3.1 可變長度碼之傳輸模型

3.2 索引層級籬柵圖

多媒體通訊中通常以封包作為一個傳輸的單位，在本論文中將視 T 個索引組成的序列 $u_1^T = [u_1, u_2, \dots, u_T]$ 為一個封包，並針對 T 已知的情形進行系統設計。而進行可變長度碼的編碼後，因為索引數目 T 固定，所以每個封包的位元總數大小 N 皆不固定，但接收端可依接收到的實數值數目來得知。所謂的索引層級籬柵圖，是在已知 T 及 N 兩個資訊的前提下建構，而每個封包所產生之籬柵圖也不盡相同。

圖 3.2 為 $T=5$ 及 $N=10$ 之索引層級籬柵圖，其中橫軸 t 為索引位置，縱軸 n 為位元數目。假設索引集合為 $I = \{0, 1, 2, 3\}$ ，對應的可變長度碼字集合為 $C = \{c(0) = [0], c(1) = [10], c(2) = [01], c(3) = [111]\}$ ，而碼字長度分別為 $l(c(0)) = 1$ 、 $l(c(1)) = 2$ 、 $l(c(2)) = 2$ 、 $l(c(3)) = 3$ ，其中 $l(\cdot)$ 為該碼字所對應的位元長度。此圖是在封包所屬的索引數目與位元長度已知的限制條件下，列舉出所有可能發生的索引組合。定義編碼進行到第 t 個索引時的節點狀態為 g_t ， $g_t = n$ 則代表前 t 個索引使用了 n 個位元做編碼。兩節點之間的狀態轉移路徑皆標記著一個碼字，例如， $g_1 = 2$ 轉移至 $g_2 = 4$ 表示第二個索引的編碼使用了 $4 - 2 = 2$ 個位元。而在此例子中，位元長度為 2 的碼字有兩個，分別為 $c(1)$ 與 $c(2)$ ，故在此狀態轉移中會有兩條平行路徑。直覺以為每個節點狀態的轉移皆應有四條路徑，因為碼書中共有四個碼字，然而因為封包索引數與位元長度的限

制，使得有些路徑不可能出現。例如從 $g_4=9$ 轉移出去的合法狀態只有 $g_5=10$ ，即碼字 $c(0)=[0]$ ，因不允許使用超過位元長度大於 1 的碼字做編碼。若使用 $c(3)=[111]$ 做編碼，將會使狀態轉移至 $g_5=12$ ，超過位元長度 $N=10$ 的限制。另一個例子是 $g_3=4$ 無法轉移至 $g_4=5$ 與 $g_4=6$ ，因碼字的最大位元長度為 3，造成下一個狀態最多只能轉移至 $g_5=8$ 與 $g_5=9$ ，並不符 N=10 位元長度的限制，故也應予以刪除。

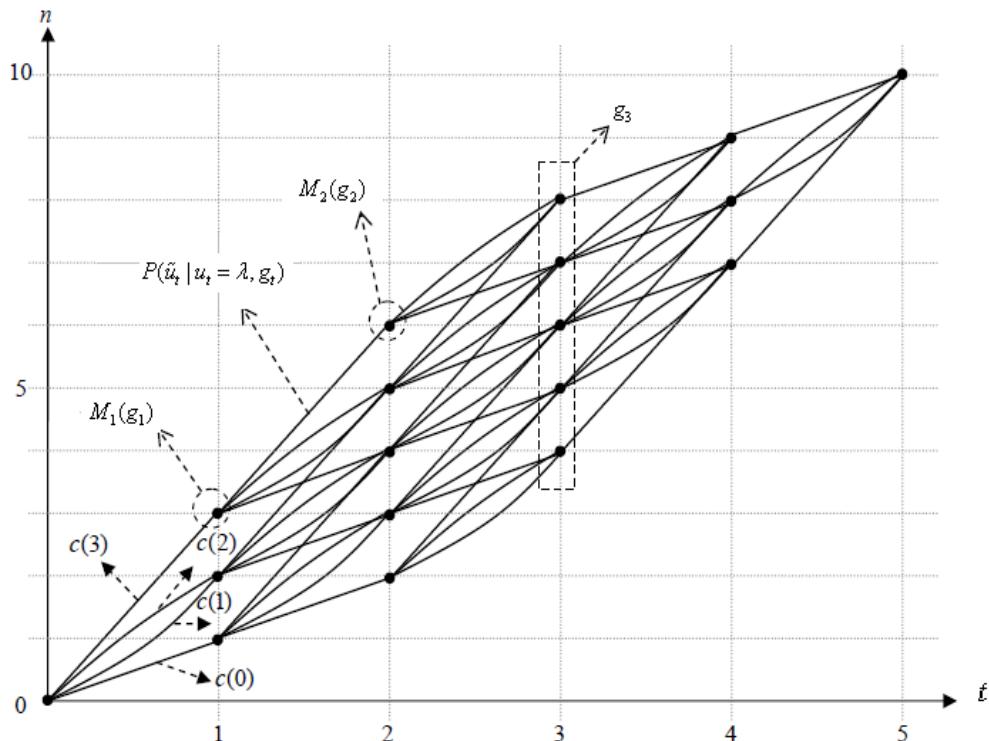


圖 3.2 索引層級籬柵圖

3.3 索引後驗機率的推導

本節將利用 3.2 節之索引層級籬柵圖，推導軟性輸入及軟性輸出(soft-input soft-output, SISO)的訊源解碼演算法，進而提供索

引層級的後驗機率 $P(u_t = \lambda | \tilde{u}_1^T)$ 。其中關鍵是引入通道的軟性資訊與訊源的一階事前相關性，並配合已知的 T 及 N 所建構之籬柵圖的狀態轉移，進行前向與後向遞迴計算。推導如下：

$$P(u_t = \lambda | \tilde{u}_1^T) = \sum_{g_t} \sum_{g_{t-1}} P(g_{t-1}, u_t = \lambda, g_t | \tilde{u}_1^T) \quad (3.1)$$

進一步使用貝氏定理(Bayes Theorem)，將 $P(g_{t-1}, u_t = \lambda, g_t | \tilde{u}_1^T)$ 此項拆解為

$$P(g_{t-1}, u_t = \lambda, g_t | \tilde{u}_1^T) = \alpha_t(\lambda, g_t) \cdot \beta_t(\lambda, g_t) / P(\tilde{u}_1^T) \quad (3.2)$$

其中前向機率 $\alpha_t(\lambda, g_t) = P(g_{t-1}, u_t = \lambda, g_t, \tilde{u}_1^t)$ ，而後向機率 $\beta_t(\lambda, g_t) = P(\tilde{u}_{t+1}^T | g_{t-1}, u_t = \lambda, g_t, \tilde{u}_1^t)$ 。接著利用量化索引間的一階馬可夫相關特性與 AWGN 通道之無記憶特性，推導此兩項機率的前向與後向遞迴運算公式為

$$\begin{aligned} \alpha_t(\lambda, g_t) &= \sum_{g_{t-2}} \sum_{q: l(q) = g_{t-1} - g_{t-2}} P(g_{t-2}, u_{t-1} = q, g_{t-1}, u_t = \lambda, g_t, \tilde{u}_1^t) \\ &= \sum_{g_{t-2}} \sum_{q: l(q) = g_{t-1} - g_{t-2}} P(u_t = \lambda, g_t, \tilde{u}_1^t | g_{t-2}, u_{t-1} = q, g_{t-1}, \tilde{u}_1^{t-1}) \\ &\quad \cdot P(g_{t-2}, u_{t-1} = q, g_{t-1}, \tilde{u}_1^{t-1}) \end{aligned} \quad (3.3)$$

$$= \sum_{g_{t-2}} \sum_{q: l(q) = g_{t-1} - g_{t-2}} \gamma_{\lambda, q}(u_t, g_t, g_{t-2}) \cdot \alpha_{t-1}(q, g_{t-1})$$

$$\beta_t(\lambda, g_t) = \sum_{g_{t+1}} \sum_{q: l(q) = g_{t+1} - g_t} P(\tilde{u}_{t+1}^T, g_{t+1}, u_{t+1} = q | g_{t-1}, u_t = \lambda, g_t, \tilde{u}_1^t) \quad (3.4)$$

$$= \sum_{g_{t+1}} \sum_{q: l(q) = g_{t+1} - g_t} P(\tilde{u}_{t+2}^T, \tilde{u}_{t+1}, g_{t+1}, u_{t+1} = q | g_{t-1}, u_t = \lambda, g_t, \tilde{u}_1^t)$$

$$= \sum_{g_{t+1}} \sum_{q: l(q) = g_{t+1} - g_t} P(\tilde{u}_{t+2}^T | \tilde{u}_{t+1}, g_{t+1}, u_{t+1} = q, g_{t-1}, u_t = \lambda, g_t, \tilde{u}_1^t)$$

$$\cdot P(\tilde{u}_{t+1}, g_{t+1}, u_{t+1} = q | g_{t-1}, u_t = \lambda, g_t, \tilde{u}_1^t)$$

$$\begin{aligned}
&= \sum_{g_{t+1}} \sum_{q: l(q) = g_{t+1} - g_t} P(\tilde{u}_{t+2}^T | \tilde{u}_1^{t+1}, g_{t+1}, u_{t+1} = q, g_{t-1}, u_t = \lambda, g_t) \\
&\quad \cdot P(\tilde{u}_{t+1}, g_{t+1}, u_{t+1} = q | g_{t-1}, u_t = \lambda, g_t, \tilde{u}_1^t) \\
&= \sum_{g_{t+1}} \sum_{q: l(q) = g_{t+1} - g_t} P(\tilde{u}_{t+2}^T | \tilde{u}_1^{t+1}, g_{t+1}, u_{t+1} = q, g_t) \\
&\quad \cdot P(\tilde{u}_{t+1}, g_{t+1}, u_{t+1} = q | g_{t-1}, u_t = \lambda, g_t, \tilde{u}_1^t) \\
&= \sum_{g_{t+1}} \sum_{q: l(q) = g_{t+1} - g_t} \gamma_{q, \lambda}(u_{t+1}, g_{t+1}, g_{t-1}) \cdot \beta_{t+1}(q, g_{t+1})
\end{aligned}$$

在式子(3.3)中

$$\begin{aligned}
\gamma_{\lambda, q}(u_t, g_t, g_{t-2}) &= P(u_t = \lambda, g_t, \tilde{u}_t | g_{t-2}, u_{t-1} = q, g_{t-1}, \tilde{u}_1^{t-1}) \quad (3.5) \\
&= P(\tilde{u}_t | u_t = \lambda, g_t, g_{t-2}, u_{t-1} = q, g_{t-1}, \tilde{u}_1^{t-1}) \\
&\quad \cdot P(u_t = \lambda, g_t | g_{t-2}, u_{t-1} = q, g_{t-1}, \tilde{u}_1^{t-1}) \\
&= P(\tilde{u}_t | u_t = \lambda, g_t) \cdot P(u_t = \lambda, g_t | u_{t-1} = q, g_{t-1})
\end{aligned}$$

進一步分析得知，式子(3.5)中的第二項可簡化為

$$P(u_t = \lambda, g_t | u_{t-1} = q, g_{t-1}) = \frac{1}{C(q, g_{t-1})} \left\{ \begin{array}{ll} P(u_t = \lambda | u_{t-1} = q) & , \quad l(c(\lambda)) = g_t - g_{t-1} \\ 0 & , \quad otherwise \end{array} \right. \quad (3.6)$$

其中 $P(u_t = \lambda | u_{t-1} = q)$ 為事前訓練所得的一階相關性資訊，而正規化常

數(normalization factor)則為

$$C(q, g_{t-1}) = \sum_{g_t} \sum_{\lambda \in I: l(c(\lambda)) = g_t - g_{t-1}} P(u_t = \lambda | u_{t-1} = q) \quad (3.7)$$

至於式子(3.5)中的第一項 $P(\tilde{u}_t | u_t = \lambda, g_t)$ 的計算，則可利用通道特性

做簡化。假設一附加性白高斯雜訊通道，其期望值為 0，變異數為

$\sigma_n^2 = N_0 / 2E_s$ ，訊號功率 $E_s = 1$ ，則

$$P(\tilde{u}_t | u_t = \lambda, g_t) = \prod_{m=1}^{l(c(\lambda))} p(\tilde{b}_{g_t-l(c(\lambda))+m} | a(m)) \quad (3.8)$$

$$= \left(\frac{1}{\sqrt{N_0 \pi}} \right)^M \cdot \exp \left(-\frac{1}{N_0} \sum_{m=1}^{l(c(\lambda))} (\tilde{b}_{g_t-l(c(\lambda))+m} - a(m))^2 \right)$$

其中 $[a(1), a(2), \dots, a(l(c(\lambda)))]$ 為碼字 $c(\lambda)$ 經過 BPSK 的雙極(bipolar)位元序列。

3.4 索引估算

經由 3.3 節之公式推導，我們可以得知傳送封包裡每個索引值的後驗機率 $P(u_t = \lambda | \tilde{u}_1^T)$ 。根據此軟性輸出之後驗機率，再以最大後驗機率(Maximum a posteriori, MAP)原則估算其索引序列 \hat{u}_t 如下：

$$\hat{u}_{t,MAP} = \arg \max_{u_t} \{ P(u_t = \lambda | \tilde{u}_1^T), \lambda \in I \} \quad (3.9)$$

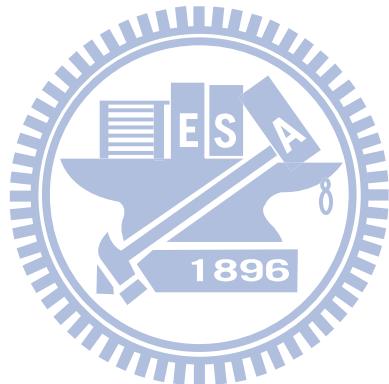
經此估算後所得的最佳索引序列为 $\hat{u}_{1,MAP}^{\wedge\wedge} = [\hat{u}_{1,MAP}, \hat{u}_{2,MAP}, \dots, \hat{u}_{T,MAP}]$ ，配合向量量化碼書的對應關係，即可還原其輸出訊號為 $\hat{v}_1^{\wedge\wedge} = [v_1, v_2, \dots, v_L]$ 。

最大後驗機率原則固然提供了一個簡易的估算方法，然而因其在估測過程中是針對單一索引個別處理，這可能造成索引序列 $\hat{u}_{1,MAP}^T$ 所對應的碼字序列为 $[c(\hat{u}_{1,MAP}^{\wedge\wedge}), c(\hat{u}_{2,MAP}^{\wedge\wedge}), \dots, c(\hat{u}_{T,MAP}^{\wedge\wedge})]$ 的位元總長度並不等於已知之長度 N 。針對此項議題，我們將使用 3.2 節之籬柵圖，在已知 T 及 N 條件下發展一個軟性輸出的維特比(viterbi)演算法。首先定義每個節點的權值(metric)，如圖 3.2 中，節點 g_t 在時間 t 的權值定義為 $M_t(g_t)$ 。擬發展的維特比演算法將引入一階馬可夫模型，並依下列遞

迴公式來計算每個節點之權值：

$$M_t(g_t) = \max\{P(u_t = \lambda | \tilde{u}_1^T)P(u_t = \lambda, g_t | u_{t-1} = q, g_{t-1}, g_{t-2})M_{t-1}(g_{t-1})\} \quad (3.10)$$

執行此演算法的遞迴運算直到 $M_T(g_T = N)$ ，再由此最終節點往回尋找機率最大化並符合 T 與 N 值的索引組合路徑，依此估測的索引序列所對應的碼字序列之位元長度也將自動符合 N 的條件。



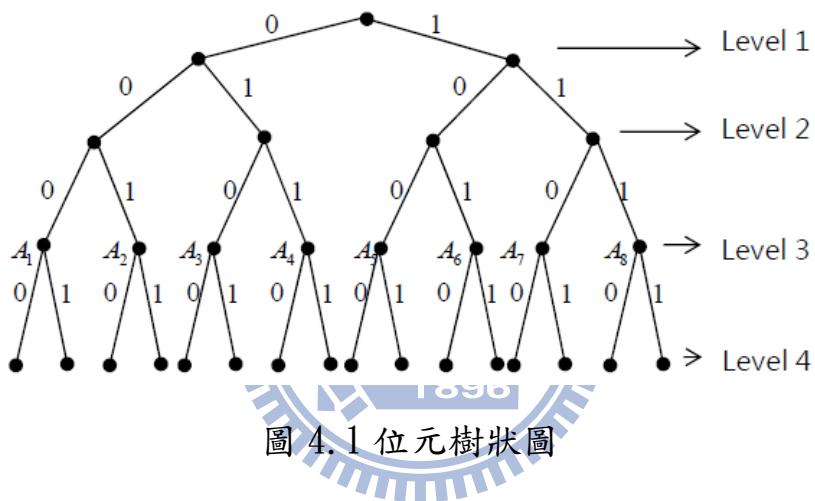
第四章 雙向性可變長度碼之設計

針對具有特定機率分佈的訊號源，可變長度碼可依照其機率大小使用不同長度的碼字加以編碼。機率較大的資料使用較短的碼字，機率相對較小的資料則使用較長的碼字，使其平均碼字長度能夠更接近理論熵值，以達到節省資料傳送量的目的。可變長度編碼方式眾多，其中以霍夫曼碼最為著名，它的前置條件(Prefix condition)允許前向解碼過程能夠符合索引同步，且其平均碼字長度經證實最接近理論熵值。可變長度碼對於位元的錯誤相當敏感，少數位元錯誤可能造成索引判斷的連續錯誤，如何克服位元錯誤的同步失序問題是可變長度解碼的一大課題。較理想的解決方案是雙向性可變長度碼(Reversible variable length code, RVLC)，同時符合前置條件與後置條件(Suffix condition)，因此能提供雙向性的即時解碼，並可定位連續錯誤發生的位置，進一步修復以加強解碼的錯誤更正能力。

本章將介紹非對稱性 RVLC 碼的不同編碼方式，其中包括基於最小平均長度的設計以及將其延伸而具有錯誤更正能力的設計，最後提出基於解碼效能的索引指定設計。

4.1 雙向性可變長度碼

雙向性可變長度碼之特性為兼具前置與後置條件，故可提供雙向性的解碼效能。前置條件是指任一碼字不得為其他較長碼字的字首，而後置條件則是指任一碼字不得為其他較長碼字的字尾。進一步又可分為對稱性(symmetric)與非對稱性(asymmetric)兩種，主要差別在於由左至右的位元組合是否與由右至左的位元組合相同。



雙向性可變長度碼的設計，一般是利用位元樹狀圖表示目標碼字選取的分佈情況，如圖 4.1 所示。Level n 代表碼字長度為 n 位元的層級，其中 A_1, A_2, \dots, A_8 代表位於第三層(位元長度為 3)中有可能被選取的碼字，定義為「候選碼字」，例如 A_3 即為碼字"010"。假設一個雙向性可變長度碼書內已選了"10"這個碼字，則為了符合前置條件，以碼字"10"為根節點的樹狀分支皆不能成為可用的候選碼字，代表 A_5, A_6 不能出現在碼書中。而為了符合後置條件， A_3, A_7 也不能留在接下來所挑選的碼字中。因此，若我們挑選"10"為「目標碼字」(即準備要

選取的碼字)，便會造成"010"、"110"、"100"、"101"，即 A_3, A_5, A_6, A_7 這四個碼字不能出現在接下來所要選的碼字中，此時第三層的「可用候選碼字」便只剩下 A_1, A_2, A_4, A_8 。另一個例子為目標碼字挑選"11"時，則依據前置條件 A_7, A_8 不能出現在碼書中，而後置條件則排除 A_4, A_8 兩個碼字，故「可用候選碼字」為 A_1, A_2, A_3, A_5, A_6 。由這兩個例子可以發現，選取不同的目標碼字時因受限於前置與後置條件的影響，使得可用的候選碼字數量存在不同的情形。

因此我們應該填選一組碼字集合，且所屬的碼字若皆符合前置與後置條件，便可稱此集合為一組雙向性可變長度碼。若是沒有效率的任選集合內的碼字，其平均長度可能遠離於理論熵值，因而使得訊源編碼的壓縮效率下降。以下我們將針對非對稱性 RVLC 碼，先介紹基於最小平均長度的設計，再討論兩種將其延伸而發展出具有錯誤修正能力的設計。

4.2 基於最小平均長度之設計

基於最小平均長度的設計理念，旨在將其編碼所得的平均長度儘可能接近理論熵值，而其中又以符合前置條件的霍夫曼碼為代表作。因此 RVLC 碼的設計一般是參考霍夫曼碼，並修改之以符合額外考量的後置條件，主要是讓每一層都能有最多的可用候選碼字提供我們選

擇。為了同時符合前置與後置條件，若已選碼字與霍夫曼碼一樣的情況下，則在下一層可挑選的碼字數目勢必小於或等於霍夫曼碼。參考4.1節的兩個例子也可以發現，不同碼字的選擇會影響到後續可用候選碼字數目的多寡。因此當我們要挑選某一層的碼字時，應配合其他選擇機制讓該層的候選碼字有其挑選的優先順序。針對非對稱性RVLC碼，台灣大學資工系吳家麟教授提出一個碼字選擇機制[5]，主要是依據碼字的最小重複間隔(Minimum Repetition Gap, MRG)，定義如下：

給定一個具有 k 位元的碼字 $C=C_1C_2\dots C_k$ ，若在其後加上一個 k 位元的碼字 X ，且碼字 X 內的每一個位元可任選是0或1，將此重新形成的碼字定義為 $D=CX$ 。 C 的重複間隔 q 代表在 D 的位元組合將每隔 q 位元重複一次，而其最小值則定義為最小重複間隔 g ，即

$$\min_{1 \leq g \leq k} \{D_i = D_j\}, \text{ for all } i \equiv j \pmod{g} \text{ and } 1 \leq i, j \leq 2k \quad (4.1)$$

舉例而言，給定一個碼字 $C="010"$ ，則 $D="010X_1X_2X_3"$ ，我們令 $X="101"$ ，因此 $D_1=D_3=D_5$ (i.e., $X_2)=0$ ， $D_2=D_4(X_1)=D_6(X_3)=1$ ，則 $q=2$ 。另一個例子為令 $X="010"$ ，因此 $D_1=D_4=0$ ， $D_2=D_5=1$ ， $D_3=D_6=0$ ， $q=3$ ，則碼字 $C="010"$ 的 $\text{MRG}=\min\{2, 3\}=2$ 。

當我們要排列某一層候選碼字的優先挑選順序時，首先將候選碼字中位元互補的碼字兩兩成為一組，因為位元互補的碼字其MRG的大小是一樣的。接著將不同的組別依MRG的值由小至大排列，若不同組

別有相同的 MRG 值，則以擁有十進位制中較小碼字的組別為優先。表

第三層		可用候選碼字數量		
候選碼字	MRG	第四層	第五層	第六層
000	1	13	24	44
111	1	13	24	44
010	2	12	21	37
101	2	12	21	37
001	3	12	20	33
110	3	12	20	33
011	3	12	20	33
100	3	12	20	33

表 4.1 第三層碼字之可用候選碼字表

4.1 列出第三層所有的候選碼字，並以 MRG 由小至大排序，並列出挑

選該碼字後在第四、五、六層可用候選碼字的數量。我們可以發現

MRG 的大小排序與後續可用候選碼字的數量息息相關，因此挑選 MRG

越小的碼字做為目標碼字，可保證我們在下一層挑選碼字時可用的候

選碼字數量較多。以下詳述其編碼演算法：

- 首先我們利用訓練資料求出每個索引的機率，並依此機率分佈進行霍夫曼編碼。將這組霍夫曼碼當作參考的可變長度碼，定義 $n(i)$ 為其第 i 層的碼字數量。定義 RVLC 目標碼字之第 i 層數量為 $n_{rev}(i)$ ，並以 $n(i)$ 為其初始值，即 $n_{rev}(i) = n(i)$ 。定義 c_r 為已選目標碼字之集合， m_{as} 為已選目標碼字之數量且初始值為 0。
- 針對第 i 層候選碼字而言，利用已選碼字的前置與後置條件刪去不可使用的碼字，保留下來的碼字即為可用候選碼字。定義可用候

選碼字數量為 $avail(i)$ ，再將可用候選碼字的優先挑選順序依 MRG

值由小至大依序排列，並定義排序後的可用候選碼字集合為：

$S_a(i) = \{c_a^1(i), c_a^2(i), \dots, c_a^{avail(i)}(i)\}$ 。接下來，針對兩種可能發生的情況分

別處理：

(1) 當 $n_{rev}(i) \leq avail(i)$ ，代表該層所需要的目標碼字數量小於或等於該層提供的可用候選碼字數量，則 $n_{rev}(i)$ 保持不變。

(2) 當 $n_{rev}(i) > avail(i)$ ，代表需要的目標碼字數量大於可用的候選碼字數量，因此無法在此層選到足夠的目標碼字，補救方法是將不足的數量 ($n_{rev}(i) - avail(i)$) 加到下一層所需要的碼字數量，即 $n_{rev}(i+1) = n_{rev}(i+1) + n_{rev}(i) - avail(i)$ ，且將本層的碼字數更改為 $n_{rev}(i) = avail(i)$ 。

3. 依照前述 MRG 的排序選取該層前 $n_{rev}(i)$ 個可用候選碼字納入目標碼

書，即 $c_r(u^{m_{as}+j}) = c_a^j(i)$, $j = 1, 2, \dots, n_{rev}(i)$ ，且 $m_{as} = m_{as} + n_{rev}(i)$ 。若 $m_{as} < n_{total}$ ，則 $i = i + 1$ 並重複上述步驟 2，反之則結束演算法，其中 n_{total} 為應選目標碼字的數目。

上述的演算法因參考最接近理論熵值的霍夫曼碼在每一層中挑選的碼字數目，在選取碼字時都儘可能達到目標數量，且 MRG 排序也能保證在下一層可用候選碼字儘可能達到最多，因此能夠找尋到一組具有最小平均長度的非對稱性雙向可變長度碼。

4.3 基於錯誤更正能力之設計

基於最小平均長度的設計，提供了一組高壓縮效率的雙向性可變長度碼，但我們希望能進一步改良以兼顧其錯誤更正能力。根據前人研究[6]，錯誤更正能力與可變長度碼本身的自由距離(free distance)有強烈的相依性。自由距離越大，則錯誤更正能力越強，但也因而造成更加嚴苛地碼字選擇條件。

考慮一組 M 位元的量化索引集合 $I = \{0, 1, 2, \dots, 2^{M-1}\}$ ，定義 $C = \{c(0), c(1), \dots, c(2^M - 1)\}$ 為索引集合 I 所對應的可變長度碼字集合。假設一個長度為 T 之索引序列 $u_1^T = \{u_1, u_2, \dots, u_T\}$ ，經過可變長度編碼後得到一組位元長度為 N 之二位元序列 $b_1^N = [c(u_1), c(u_2), \dots, c(u_T)] = [b_1, b_2, \dots, b_N]$ ，為了方便說明，我們將 T 個索引所對應的位元序列 b_1^N 視為一個 N 位元的碼字，並將所有可能的碼字集合定義為 B 。定義自由距離 d_f 為集合 B 中任兩個不同碼字 $\{b^{(m)}, b^{(n)}\}$ 之間的最小漢明距離(Hamming Distance)，而區塊距離 d_b 則定義為集合 C 中任兩個位元等長度的不同碼字之最小漢明距離。若令 d_H 為漢明距離，則 d_f 與 d_b 分別定義如下：

$$d_f = \min\{d_H(b^{(m)}, b^{(n)}) \mid b^{(m)}, b^{(n)} \in B; b^{(m)} \neq b^{(n)}\} \quad (4.2)$$

$$d_b = \min\{d_H(c(i), c(j)) \mid c(i), c(j) \in C, l(i) = l(j); i \neq j\} \quad (4.3)$$

其中 $l(i)$ 為索引 i 所對應之碼字長度。我們藉由觀察霍夫曼碼之碼字，可發現位元等長的碼字彼此間最小的漢明距離為 1，且因其只具有前

置特性，故 $d_f \geq 1$ 。而根據文獻[6]中所證明，雙向性可變長度碼具有前置與後置條件，所以具有 $d_f \geq \min(2, d_b)$ 之性質，故若能讓 $d_b \geq 2$ 則 $d_f \geq 2$ 。藉由上述的討論我們可以得知，若能將集合 C 中位元等長的不同碼字之最小漢明距離提升至 2，即 $d_b \geq 2$ ，則能夠使得雙向性可變長度碼的錯誤更正能力提高。以下之演算法將基於此特性，並配合 4.2 節最小平均長度的設計概念，找尋一組符合 $d_f \geq 2$ 條件下最小平均長度之雙向性可變長度碼：

1. 首先我們利用訓練資料求出每個索引的機率，並依此機率分佈進行霍夫曼編碼，將這組霍夫曼碼當作參考的可變長度碼，定義 $n(i)$ 為其第 i 層的碼字數量。定義目標雙向性可變長度碼之第 i 層數量為 $n_{rev}(i)$ ，並以 $n(i)$ 為其初始值⁸⁹，即 $n_{rev}(i) = n(i)$ ，定義 c_r 為已選目標碼字之集合， m_{as} 為已選目標碼字之數量且初始值為 0。
2. 針對第 i 層候選碼字而言，利用已選碼字的前置與後置條件刪去不可使用的碼字，保留下來的碼字為可用候選碼字。定義可用候選碼字數量為 $avail(i)$ ，再將可用候選碼字的優先挑選順序依 MRG 值由小至大依序排列，並定義排序後的可用候選碼字集合為：

$$S_a(i) = \{c_a^1(i), c_a^2(i), \dots, c_a^{avail(i)}(i)\}.$$

3. 在指定較優先選取的 $\min(n_{rev}(i), avail(i))$ 個目標碼字之前，先確認 $S_a(i)$ 集合的碼字最小漢明距離是否 ≥ 2 ，即：

$$\min\{d_H(c_a^j(i), c_a^k(i)) \mid 1 \leq j, k \leq \min(n_{rev}(i), avail(i)); j \neq k\} \geq 2 \quad (4.4)$$

若 ≥ 2 ，則進行步驟 4，反之則進行步驟 5。

4. 分為兩種情形處理：

(1) 當 $n_{rev}(i) \leq avail(i)$ ，代表該層所需要的碼字數量小於等於該層

可提供的可用候選碼字數量，則 $n_{rev}(i)$ 不變。

(2) 當 $n_{rev}(i) > avail(i)$ ，代表需要的碼字數量大於可用的候選碼字數

量，因此不能在此層提供足夠的碼字數目，此時將不足數量

的 $(n_{rev}(i) - avail(i))$ 加到下一層所需要的碼字數量中，即

$$n_{rev}(i+1) = n_{rev}(i+1) + n_{rev}(i) - avail(i) \text{，且將本層的碼字數更改為}$$

$$n_{rev}(i) = avail(i) \circ$$

依照前述 MRG 的排序選取該層前 $n_{rev}(i)$ 個可用候選碼字納入目標碼

書，即： $c_r(u^{m_{as}+j}) = c_a^j(i), j = 1, 2, \dots, n_{rev}(i)$ ，接著進行步驟 6。

5. 將 $S_a(i)$ 分成兩個集合， $S_o(i)$ 與 $S_e(i)$ ，其中 $S_o(i)$ 代表 $S_a(i)$ 中漢明權重

(Hamming weight)為奇數的碼字，而 $S_e(i)$ 則代表漢明權重為偶數

的碼字。因為此兩集合中碼字的漢明權重皆為偶數或者奇數，故

在同一集合內的不同碼字彼此之間漢明距離必定 ≥ 2 。定義此兩集

合的碼字數量分別為 $n_o(i)$ 與 $n_e(i)$ ，且令 $n_s(i) = \max(n_o(i), n_e(i))$ ，並從

碼字數量較多的集合中去挑選目標碼字，即：

$$c_r(u^{m_{as}+j}) = c_e^j(i), j = 1, \dots, \min(n_{rev}(i), n_s(i)) \quad \text{if } n_e(i) \geq n_o(i)$$

$$c_r(u^{m_{as}+j}) = c_o^j(i), j = 1, \dots, \min(n_{rev}(i), n_s(i)) \quad \text{if } n_e(i) < n_o(i) \quad (4.5)$$

- (1) 當 $n_{rev}(i) \leq n_s(i)$ ，代表該層所需要的碼字數量小於等於該層可提供的可用候選碼字數量，則 $n_{rev}(i)$ 不變。
- (2) 當 $n_{rev}(i) > n_s(i)$ ，代表需要的碼字數量大於可用的候選碼字數量，因此不能在此層選取足夠的碼字數目，此時將不足的數量 ($n_{rev}(i) - n_s(i)$) 加到下一層所需要的碼字數量中，即

$$n_{rev}(i+1) = n_{rev}(i+1) + n_{rev}(i) - n_s(i)，且將本層的碼字數更改為$$

$$n_{rev}(i) = n_s(i)。$$

依照前述 MRG 排序選取該層前 $n_{rev}(i)$ 個可用候選碼字編入碼書，即：

$$c_r(u^{m_{as}+j}) = c_a^j(i), j = 1, 2, \dots, n_{rev}(i)，接著進行步驟 6。$$

6. 更新已選碼字之數目 $m_{as} = m_{as} + n_{rev}(i)$ ，若 $m_{as} < n_{total}$ ，則 $i = i + 1$ 並重複上述步驟 2，反之則結束演算法。

4.4 基於解碼效能之索引指定設計

索引指定設計旨在置換 RVLC 碼書中碼字與其索引間之對應關係，以期能找出一組最佳的對應組合以提升訊源解碼效能。我們提出的索引指定設計，主要的置換對象是位元等長的碼字集合，以避免置換位元長度不同的碼字而導致碼字平均長度的增加。我們的構想是在置換過程中，先將碼書中的碼字依位元長度分成不同的集合，每次置

換只置換特定集合內的碼字，其他集合的索引指定則不變。進行置換之集合的順序是依位元長度由小至大，例如有位元長度 3、4、5 的三個碼字集合，將先置換位元長度 3 之集合內的碼字，完成後再置換位元長度 4 的集合，以此類推。

最佳的索引指定必須將訊源解碼效能納入設計的考量，在此論文中使用的軟性輸出訊源解碼器主要是利用 AWGN 通道資訊與一階事前相關性來做解碼，本節將利用此兩個資訊定義一項可具體反應訊源解碼效能的目標函數。評估解碼效能的一個重要指標參數是訊雜比

(Parameter Signal to Noise Ratio, PSNR)，我們利用此機制來定義

目標函數：

$$\rho = \frac{\sum_{l=1}^L v_l^2}{\sum_{l=1}^L (v_l - \hat{v}_l)^2} \quad (4.6)$$



令 $v_1^L = [v_1, v_2, \dots, v_L]$ 為一 L 個數值之訓練資料，經過一 K 維度為且 M 位元的向量量化器產生一索引序列 $u_1^T = [u_1, u_2, \dots, u_T]$ ，其中 $T = L / K$ 。再將 T 個索引依序進行可變長度編碼，令其位元長度依序為 $l_1^T = [l_1, l_2, \dots, l_T] = [l(c(u_1)), l(c(u_2)), \dots, l(c(u_T))]$ 。此外，接收端解碼輸出 $\hat{v}_1^L = [v_1, v_2, \dots, \hat{v}_L] = [q(\hat{u}_1), q(u_2), \dots, q(u_T)]$ ，其中 $q(\hat{u}_t)$ 為索引估測值 \hat{u}_t 對照查表所得的 K 維碼向量。至於索引 \hat{u}_t 的估測方式，我們將在下一段詳細說明。

可變長度碼的特性是針對不同的量化索引編碼所得之位元數並

不固定，因而造成傳輸錯誤所衍生的索引同步的問題。針對此問題，論文[3]中假設接收端已知傳送封包的位元總數 N 與索引總數 T ，提供一索引層級的最大後驗機率解碼演算法。其主要構想是建構可變長度碼索引層級籬柵圖，代表滿足 N 與 T 的所有可能組合序列，並基於此籬柵圖而計算每一個索引的後驗機率。問題是後驗機率的計算，必須考量到 N 及 T 限制下所有可能的路徑，整體運算量將隨著 T 的增長而無可預期。針對此議題我們將在接收端已知 N 、 T 及 $l_1^T = [l_1, l_2, \dots, l_T]$ 的假設下，重新推導其後驗機率 $P(u_t = i | \tilde{u}_1^t, l_1^t)$ ，並引入通道的軟性資訊與一階事前相關性進行遞迴運算。令前 t 個索引的接收序列為 $\tilde{u}_1^t = [\tilde{u}_1, \tilde{u}_2, \dots, \tilde{u}_t]$ ，其中 $\tilde{u}_t = [\tilde{b}_{l_1+l_2+\dots+l_{t-1}+1}, \tilde{b}_{l_1+l_2+\dots+l_{t-1}+2}, \dots, \tilde{b}_{l_1+l_2+\dots+l_t}]$ ， \tilde{b}_n 為接收端收到的第 n 個實數值。若令 $S(l_{t-1})$ 代表位元長度為 l_{t-1} 的可變長度碼字組成的集合，索引層級後驗機率可推導如下：

$$\begin{aligned}
& P(u_t = i | \tilde{u}_1^t, l_1^t) \\
&= \sum_{j \in S(l_{t-1})} P(u_t = i, u_{t-1} = j | \tilde{u}_1^{t-1}, \tilde{u}_t, l_1^t) \\
&= \frac{1}{P(\tilde{u}_1^t, l_1^t)} \sum_{j \in S(l_{t-1})} P(u_t = i, u_{t-1} = j, \tilde{u}_1^{t-1}, \tilde{u}_t, l_1^t) \\
&= C_1 \sum_{j \in S(l_{t-1})} P(\tilde{u}_t | u_t = i, u_{t-1} = j, \tilde{u}_1^{t-1}, l_1^t) \cdot P(u_t = i | u_{t-1} = j, \tilde{u}_1^{t-1}, l_1^t) \\
&\quad \cdot P(u_{t-1} = j | \tilde{u}_1^{t-1}, l_1^{t-1}) \tag{4.7}
\end{aligned}$$

其中正規化常數 $C_1 = P(\tilde{u}_1^{t-1}, l_1^{t-1}) / P(\tilde{u}_t, l_1^t)$ 。進一步考慮通道的無記憶性，

$P(\tilde{u}_t | u_t = i, u_{t-1} = j, \tilde{u}_1^{t-1}, l_1^t)$ 可簡化為 $P(\tilde{u}_t | u_t = i, l_1^t)$ 。而 $P(\tilde{u}_t | u_t = i, l_1^t) =$

$\prod_{l=1}^L P(\tilde{b}_{l_1+l_2+\dots+l_{t-1}+l} | b_{l_1+l_2+\dots+l_{t-1}+l})$ 。此外，基於事前資訊是一階馬可夫模型的假設， $P(u_t = i | u_{t-1} = j, \tilde{u}_1^{t-1}, l_1^t) = P(u_t = i | u_{t-1} = j)$ 。綜合整理後，可得出下式：

$$P(u_t = i | \tilde{u}_1^t, l_1^t) = C_1 \sum_{j \in S(l_{t-1})} P(\tilde{u}_t | u_t = i, l_1^t) \cdot P(u_t = i | u_{t-1} = j) \\ \cdot P(u_{t-1} = j | \tilde{u}_1^{t-1}, l_1^{t-1}) \quad (4.8)$$

其中 $P(\tilde{u}_t | u_t = i, l_1^t)$ 為通道的軟性資訊， $P(u_t = i | u_{t-1} = j)$ 為一階的事前資訊， $P(u_{t-1} = j | \tilde{u}_1^{t-1}, l_1^{t-1})$ 則是代表前一時刻的索引後驗機率。最後依據最大後驗機率原則進行估測：

$$\hat{u}_t = \arg \left\{ \max_i (P(u_t = i | \tilde{u}_1^t, l_1^t)) \right\}, i \in S(l_t) \quad (4.9)$$

不同的索引指定皆可以透過事先訓練而計算出 ρ ， ρ 越大則代表其訊源解碼效能更佳。因此一組最佳的索引指定必須使得 ρ 最大化，但此最佳化問題所需考慮的碼字置換組合數目相當的多。例如一個雙向性可變長度碼書，其中具有位元長度 3 的碼字五個、長度為 4 的六個、長度為 5 的八個，依前述的原則置換位元等長度的碼字，所有的排列組合共有 $5! \times 6! \times 8! = 3.4836 \times 10^9$ 種。為了得到一組具有最大 ρ 值的索引指定，必須針對所有可能組合個別計算其 ρ 值並做交叉比較，而這在實際運作上有其困難。因此在進行置換過程中，我們將採用具體可行的二位元置換演算法(Binary Switching Algorithm, BSA)方式。二位元置換演算法只能找出區域最佳值(local optimal)，但可以省去全域最佳值(global optimal)的龐大運算量。

進行二位元置換演算法時，置換的目標為索引所對應的碼字，在不改變索引順序之前前提下，改變索引所對應之碼字的順序。首先，根據 4.3 節之演算法產生具有錯誤更正能力的雙向性可變長度碼，將其所屬的碼字依其對應索引機率由大至小的排列，並將這組索引與碼字的對應設為初始的索引指定，計算此索引指定的 ρ ，且將其設為參考值 ρ^{ref} 。接著將碼字中最小及最大位元長度分別記為 min 和 max，不同位元長度組成的碼字集合定義為 $\{level(min), level(min+1), \dots, level(max)\}$ ，進行置換的集合順序從 $level(min)$ 至 $level(max)$ 。先考慮具有位元長度 $level(min)$ 之碼字集合，定義為 $c = \{c(1), c(2), \dots, c(n(level(min)))\}$ ， $n(level(min))$ 為碼字數量，初始設定 $i=1$ 及 $j=i+1$ ，即碼字 $c(i)$ 與 $c(j)$ 順序進行置換，置換後計算新的參考值 ρ_{new}^{ref} 。如果 ρ_{new}^{ref} 大於 ρ^{ref} ，則將此兩個碼字之順序進行置換，並且重新設定 $i=1$ 及 $j=i+1$ 。反之，則碼字索引維持不變。令 $j=j+1$ ，直到 $j=n(level(min))$ 且 ρ_{new}^{ref} 依然小於 ρ^{ref} 時，使得 $i=i+1$ 、 $j=i+1$ ，並重複上面動作，至 $i=n(level(min))$ 且已不再發生置換情形時，則此索引指定為 $level(min)$ 此層區域最佳之索引排列組合。這樣的步驟由 $level(min)$ 一直執行到 $level(max)$ ，即可找出每層的區域最佳索引指定。圖 4.2 為二位元置換演算法之流程圖：

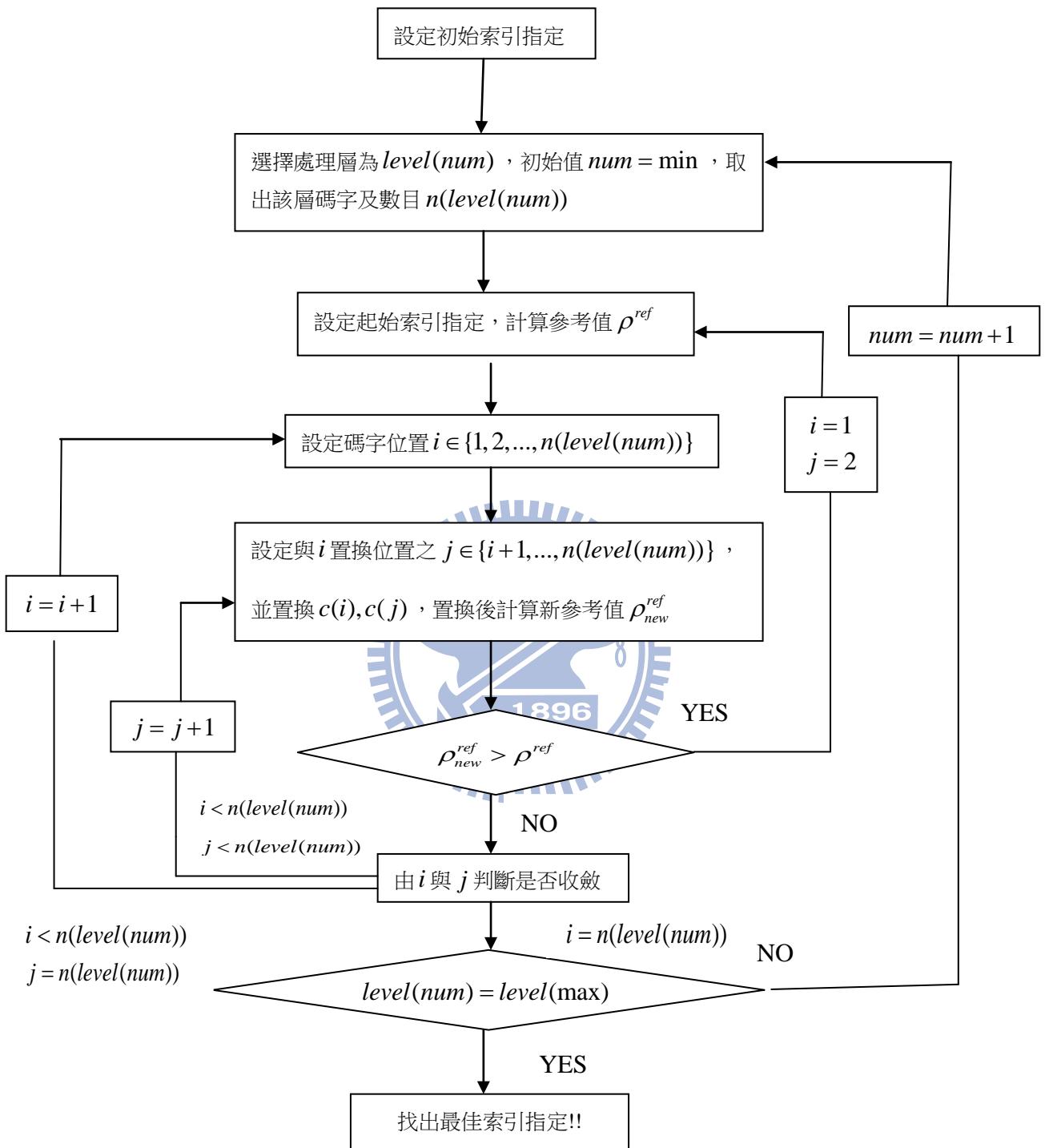


圖 4.2 二位元置換演算法

第五章 實驗模擬與結果分析

在前面的章節中，詳細地介紹傳輸系統的架構與非對稱性 RVLC 的編解碼過程，並將其運用在心電圖的傳輸上。可變長度碼字間存在的殘存冗息可運用於軟性訊源解碼，而基於不同的可變長度碼之殘存的冗息也有多寡之分。若一可變長度碼字的平均長度可達理論熵值，即代表著沒有任何冗餘，也可達到最好的壓縮效率。然而前面章節所介紹 RVLC 編碼雖使得碼字平均長度較傳統的霍夫曼碼來得大，但也使得解碼時有較多的殘存冗息來增進其強健效能。為了進一步驗證其效能，我們將在此章中模擬系統傳輸端與接收端，計算心電圖訊號源與接收端還原訊號之訊雜比，並進行結果的比較與分析。

本章將分為三個實驗，首先實驗一將對第四章中基於不同觀點所設計的 RVLC 及索引指定設計進行傳輸效益的探討。實驗二利用 4.5 節中訓練索引指定的方式，針對不同通道情況訓練 RVLC，並對其解碼效能加以討論比較。最後我們將利用第三章的軟性訊源解碼機制，針對兩種不同的索引估計方式做效能比較。

5.1 實驗環境設定

本節主要敘述後續實驗的基本設定，而對於每個實驗所需設定的細節將在往後的實驗過程中逐次說明。

整體架構如圖 3.1 所示，其中訊號源將採用 MIT-BIH 資料庫的第 100 號心電圖。每次傳送以一個封包為單位，一個封包內有 100 個心電圖的離散取樣值，經維度為 $K=2$ 且 $M=6$ 的向量量化處理而得 50 個量化索引。再使用二進位相位鍵移調變方式，位元 " 0 " $\rightarrow +1$ 、" 1 " $\rightarrow -1$ ，位元平均能量為 1。傳送通道的雜訊以無記憶可加性白色高斯雜訊模擬，其平均值為 0 且變異數為 $N_0/2$ 。接收端採用第三章所敘述的軟性訊源解碼機制，並以其軟性輸出進行索引的估測。至於心電圖重建品質之評估方式，我們是採用參數訊雜比(Parameter SNR, PSNR)。

在進行實驗之前，我們將會透過大量的訓練資料取得專屬 100 號患者的心電圖中每個索引發生的機率及其一階的事前資訊。

5.2 實驗一

目的：探討不同的可變長度碼設計對於軟性訊源解碼效能的影響。

模擬環境：透過事前的訓練可知每個索引發生的機率，並據以建構其霍夫曼碼，依 4.2、4.3 節敘述，可延伸出基於最小長度及具有錯誤更正能力的兩種設計。索引指定則同樣透過 MIT-BIH 資料庫 100 號心電圖的資料，依 4.4 節的流程進行訓練，而訓練時的通道環境設定為 3dB。接著依

5.1 節敘述進行相關的環境設定，再將訊源編碼的碼書

依序變更為霍夫曼碼與第四章中三種雙向性可變長度碼

的設計，索引的估算則採用 3.4 節的維特比演算法。

結 果：

量 化 索 引 值	索引機率	<i>Huffman</i>	<i>RVLC</i> ₁	<i>RVLC</i> ₂	<i>RVLC</i> ₃
0	0.051007	[1 0 1 0]	[0 0 0 0]	[0 0 0 0]	[1 0 1 0]
1	0.049087	[1 0 1 1]	[1 1 1 1]	[1 1 1 1]	[1 1 1 1]
2	0.04867	[1 1 0 0]	[0 1 0 1]	[0 1 0 1]	[0 1 0 1]
3	0.046467	[1 1 1 0]	[1 0 1 0]	[1 0 1 0]	[0 0 0 0]
4	0.045663	[1 1 1 1]	[0 0 1 0]	[0 1 1 0]	[0 1 1 0]
5	0.045483	[0 0 0 0 0]	[1 1 0 1 1]	[1 1 0 1 1]	[1 0 1 1 1]
6	0.045333	[0 0 0 0 1]	[0 1 0 0 1]	[0 1 0 0 1]	[0 1 0 0 1]
7	0.042283	[0 0 0 1 0]	[1 0 1 1 0]	[1 0 0 1 0]	[1 0 0 1 0]
8	0.042273	[0 0 0 1 1]	[0 1 1 0 1]	[1 1 1 0 1]	[1 1 1 0 1]
9	0.04196	[0 0 1 0 0]	[1 1 1 0 1] 896	[1 0 1 1 1]	[1 1 0 1 1]
10	0.03939	[0 0 1 1 0]	[0 0 1 1 0]	[1 0 0 0 1]	[1 0 0 0 1]
11	0.03728	[0 0 1 1 1]	[1 1 0 0 1]	[0 0 0 1 1]	[0 0 0 1 1]
12	0.036893	[0 1 0 0 0]	[0 1 0 0 0]	[1 1 0 0 0]	[1 1 0 0 0]
13	0.03678	[0 1 0 0 1]	[1 0 1 1 1]	[0 0 0 1 0 0]	[0 0 1 1 1 0]
14	0.032693	[0 1 1 0 0]	[0 1 1 0 0]	[0 0 1 0 0 0]	[0 0 1 0 0 0]
15	0.032203	[0 1 1 0 1]	[1 0 0 1 1]	[0 0 1 1 1 0]	[0 0 0 1 0 0]
16	0.03148	[0 1 1 1 0]	[0 1 1 1 0]	[0 1 1 1 0 0]	[0 1 1 1 0 0]
17	0.031467	[0 1 1 1 1]	[1 0 0 0 1]	[0 0 1 0 1 1]	[0 0 1 0 1 1]
18	0.029803	[1 0 0 0 0]	[0 0 0 1 1]	[1 1 0 1 0 0]	[1 1 0 1 0 0]
19	0.026457	[1 0 0 0 1]	[1 1 1 0 0]	[0 0 1 1 0 1]	[0 0 1 1 0 1]
20	0.025127	[1 0 0 1 1]	[0 0 1 1 1]	[1 0 1 1 0 0]	[1 0 1 1 0 0]
21	0.024163	[1 1 0 1 0]	[1 1 0 0 0]	[0 1 0 0 0 1 0]	[0 1 0 0 0 0 1]
22	0.022707	[1 1 0 1 1]	[1 0 0 1 0 0]	[0 0 1 1 0 0 1]	[0 0 1 1 0 0 1]
23	0.021273	[0 0 1 0 1 0]	[0 0 0 1 0 0]	[1 0 0 1 1 0 0]	[0 1 0 0 0 1 0]
24	0.01674	[0 1 0 1 0 1]	[0 1 1 1 1 0]	[1 1 1 0 0 1 1]	[1 1 1 0 0 1 1]
25	0.016337	[0 1 0 1 1 1]	[1 0 0 0 0 1]	[0 0 1 0 1 0 0]	[0 0 1 0 1 0 0]

26	0.01353	[100100]	[1101000]	[1101011]	[1101011]
27	0.00958	[0010111]	[1101011]	[1100111]	[1100111]
28	0.00841	[0101100]	[1001011]	[0100001]	[1001100]
29	0.0031533	[10010101]	[10010100]	[00100100]	[00100100]
30	0.0028133	[001011000]	[110101011]	[101101101]	[001111001]
31	0.0025367	[001011011]	[100101011]	[011101110]	[011101110]
32	0.0022433	[010100011]	[011111110]	[100111001]	[100111001]
33	0.0021267	[010100111]	[100000001]	[000101000]	[000101000]
34	0.0021133	[010110100]	[000101011]	[111000111]	[111000111]
35	0.0019933	[010110110]	[1001010100]	[001100001]	[001100001]
36	0.0016833	[100101000]	[0001010100]	[001111001]	[100111100]
37	0.0016633	[100101001]	[1000001011]	[010000010]	[010000010]
38	0.00155	[100101100]	[011111110]	[100111100]	[101101101]
39	0.0015467	[100101101]	[100000001]	[100001100]	[100001100]
40	0.0015467	[100101110]	[1101010100]	[001010100]	[001010100]
41	0.0015467	[100101111]	[10000010100]	[110101011]	[110101011]
42	0.0013533	[0010110010]	[11010101011]	[1100110011]	[0111011001]
43	0.0013067	[0010110011]	[00010111110]	[1110000111]	[0100000001]
44	0.0013067	[0010110100]	[10010101011]	[1100101110]	[1100101110]
45	0.00123	[0101000000]	[10000001011]	[0011111001]	[1001111100]
46	0.0012233	[0101000001]	[01111111110]	[0100000010]	[0100000010]
47	0.0011933	[0101000010]	[1000000001]	[1001111100]	[0011111001]
48	0.00118	[0101000011]	[00010101011]	[0111010011]	[1100101011]
49	0.00117	[0101000100]	[00010100001]	[1110010011]	[1110010011]
50	0.00117	[0101000101]	[100101010100]	[1101010011]	[1101010011]
51	0.0011267	[0101001000]	[100000010100]	[1100101011]	[1110000111]
52	0.0011133	[0101001001]	[000101010100]	[1100100111]	[1001101110]
53	0.0011067	[0101001010]	[01111100001]	[0011111100]	[0011111100]
54	0.0010967	[0101001011]	[100000111110]	[0100000001]	[1110000001]
55	0.00107	[0101001100]	[000101111110]	[1001101110]	[1100101101]
56	0.0010667	[0101001101]	[11010100001]	[0111011001]	[1100110011]
57	0.00105	[0101101010]	[011111010100]	[1000000010]	[1011011001]
58	0.00102	[0101101011]	[011111111110]	[1110000001]	[0111010011]
59	0.00097667	[0101101110]	[10000000001]	[1100101101]	[1100100111]
60	0.00089	[0101101111]	[110101010100]	[1011011001]	[1000000010]
61	0.00082	[00101101010]	[011111011110]	[00110001100]	[00110001100]
62	0.0004433	[001011010110]	[100000100001]	[100001000010]	[100001000010]

63	3.00E-05	[001011010111]	[0001010000001]	[011110011110]	[011110011110]
	平均長度	5.0824	5.1548	5.49	5.49

表 5.1 不同可變長度碼設計之碼書

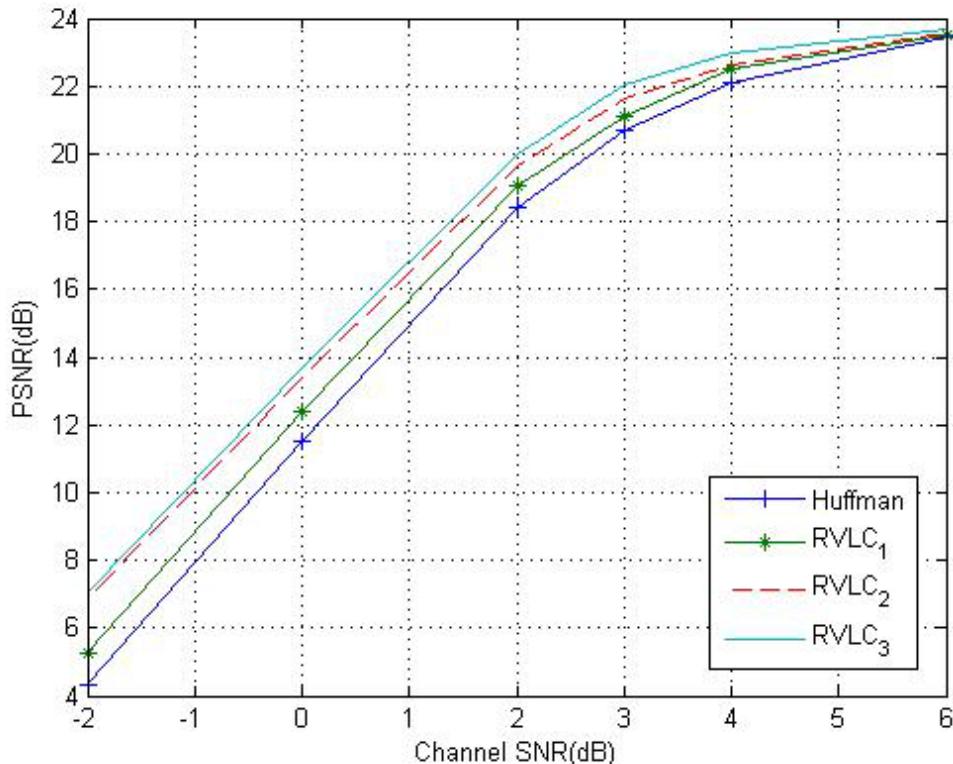


圖 5.1 四種可變長度碼之軟性訊源解碼效能

分析：RVLC₁為基於最小平均長度的設計，RVLC₂為具有錯誤更正能力的設計，而RVLC₃為4.4節所建議的索引指定設計。可以從表5.1中得知霍夫曼碼的碼字平均長度為5.0824，RVLC₁的選取較霍夫曼碼多了後置條件的這項限制，故其平均長度提升至5.1548，而RVLC₂則因為位元等長度的碼字間之漢明距離必須不小於2，使得平均長

度增加到 5.49。至於 $RVLC_3$ ，只是在 $RVLC_2$ 中位元等長度的碼字集合進行置換，而不是針對所有的碼字集合，因此使得整體的平均長度仍為 5.49。圖 5.1 比較此四種碼書應用在軟性訊源解碼機制上的效能，可以發現平均碼字長度的增加使得訊源編碼後產生較多的殘餘冗息，有助於軟性訊源解碼效能的提升，因此解碼效 $Huffman < RVLC_1 < RVLC_2$ 。至於額外考慮索引指定設計 $RVLC_3$ ，因其在訓練時的通道環境設定為 3dB，相較於 $RVLC_2$ 在通道環境 3dB 下有約 0.44dB 的顯著改進，但在通道環境小於 0dB 時，改進幅度則因通道環境不匹配而較不明顯，因此我們可以推測在設計索引指定時，用於訓練與測試的通道環境應匹配，才能取得最佳的解碼效能。

5.2 實驗二

目的：探討匹配於通道環境的索引指定設計。
模擬環境：分別依通道訊雜比(Channel SNR, CSNR)為 0dB 與 3dB 的訓練環境，進行 4.4 節之索引指定設計，將此兩種不同碼書依 5.1 節的環境設定進行實驗，索引的估算同樣採用 3.4 節的維特比演算法。

結果：

量化索引值	索引機率	$RVLC_3(3)$	$RVLC_3(0)$
0	0.051007	[1 0 1 0]	[1 0 1 0]
1	0.049087	[1 1 1 1]	[1 1 1 1]
2	0.04867	[0 1 0 1]	[0 1 0 1]
3	0.046467	[0 0 0 0]	[0 0 0 0]
4	0.045663	[0 1 1 0]	[0 1 1 0]
5	0.045483	[1 0 1 1 1]	[1 1 1 0 1]
6	0.045333	[0 1 0 0 1]	[0 1 0 0 1]
7	0.042283	[1 0 0 1 0]	[1 0 0 1 0]
8	0.042273	[1 1 1 0 1]	[1 1 0 1 1]
9	0.04196	[1 1 0 1 1]	[1 0 1 1 1]
10	0.03939	[1 0 0 0 1]	[1 0 0 0 1]
11	0.03728	[0 0 0 1 1]	[0 0 0 1 1]
12	0.036893	[1 1 0 0 0]	[1 1 0 0 0]
13	0.03678	[0 0 1 1 1 0]	[1 1 0 1 0 0]
14	0.032693	[0 0 1 0 0 0]	[0 0 1 0 0 0]
15	0.032203	[0 0 0 1 0 0]	[0 0 1 1 1 0]
16	0.03148	[0 1 1 1 0 0]	[0 1 1 1 0 0]
17	0.031467	[0 0 1 0 1 1] 396	[0 0 1 0 1 1]
18	0.029803	[1 1 0 1 0 0]	[0 0 0 1 0 0]
19	0.026457	[0 0 1 1 0 1]	[0 0 1 1 0 1]
20	0.025127	[1 0 1 1 0 0]	[1 0 1 1 0 0]
21	0.024163	[0 1 0 0 0 0 1]	[0 0 1 0 1 0 0]
22	0.022707	[0 0 1 1 0 0 1]	[0 0 1 1 0 0 1]
23	0.021273	[0 1 0 0 0 1 0]	[1 0 0 1 1 0 0]
24	0.01674	[1 1 1 0 0 1 1]	[1 1 1 0 0 1 1]
25	0.016337	[0 0 1 0 1 0 0]	[0 1 0 0 0 1 0]
26	0.01353	[1 1 0 1 0 1 1]	[1 1 0 1 0 1 1]
27	0.00958	[1 1 0 0 1 1 1]	[1 1 0 0 1 1 1]
28	0.00841	[1 0 0 1 1 0 0]	[0 1 0 0 0 0 1]
29	0.0031533	[0 0 1 0 0 1 0 0]	[0 0 1 0 0 1 0 0]
30	0.0028133	[0 0 1 1 1 1 0 0 1]	[1 1 0 1 0 1 0 1 1]
31	0.0025367	[0 1 1 1 0 1 1 1 0]	[0 0 1 0 1 0 1 0 0]
32	0.0022433	[1 0 0 1 1 1 0 0 1]	[1 0 0 1 1 1 0 0 1]
33	0.0021267	[0 0 0 1 0 1 0 0 0]	[0 0 0 1 0 1 0 0 0]

34	0.0021133	[1 1 1 0 0 0 1 1 1]	[0 0 1 1 0 0 0 0 1]
35	0.0019933	[0 0 1 1 0 0 0 0 1]	[1 1 1 0 0 0 1 1 1]
36	0.0016833	[1 0 0 1 1 1 0 0]	[0 0 1 1 1 1 0 0 1]
37	0.0016633	[0 1 0 0 0 0 0 1 0]	[0 1 0 0 0 0 0 1 0]
38	0.00155	[1 0 1 1 0 1 1 0 1]	[1 0 0 1 1 1 1 0 0]
39	0.0015467	[1 0 0 0 0 1 1 0 0]	[1 0 0 0 0 1 1 0 0]
40	0.0015467	[0 0 1 0 1 0 1 0 0]	[0 1 1 1 0 1 1 1 0]
41	0.0015467	[1 1 0 1 0 1 0 1 1]	[1 0 1 1 0 1 1 0 1]
42	0.0013533	[0 1 1 1 0 1 1 0 0 1]	[1 1 0 0 1 0 1 1 1 0]
43	0.0013067	[0 1 0 0 0 0 0 0 1]	[1 1 1 0 0 0 0 1 1 1]
44	0.0013067	[1 1 0 0 1 0 1 1 1 0]	[1 0 0 0 0 0 0 0 1 0]
45	0.00123	[1 0 0 1 1 1 1 1 0 0]	[0 0 1 1 1 1 1 1 0 0]
46	0.0012233	[0 1 0 0 0 0 0 0 1 0]	[0 1 0 0 0 0 0 0 1 0]
47	0.0011933	[0 0 1 1 1 1 1 0 0 1]	[0 0 1 1 1 1 1 0 0 1]
48	0.00118	[1 1 0 0 1 0 1 0 1 1]	[1 0 0 1 1 0 1 1 1 0]
49	0.00117	[1 1 1 0 0 1 0 0 1 1]	[1 1 1 0 0 1 0 0 1 1]
50	0.00117	[1 1 0 1 0 1 0 0 1 1]	[1 1 0 1 0 1 0 0 1 1]
51	0.0011267	[1 1 1 0 0 0 0 1 1 1]	[1 1 0 0 1 0 1 0 1 1]
52	0.0011133	[1 0 0 1 1 0 1 1 1 0]	[1 1 0 0 1 0 0 1 1 1]
53	0.0011067	[0 0 1 1 1 1 1 1 0 0]	[1 0 0 1 1 1 1 1 0 0]
54	0.0010967	[1 1 1 0 0 0 0 0 0 1]	[0 1 0 0 0 0 0 0 0 1]
55	0.00107	[1 1 0 0 1 0 1 1 0 1]	[1 1 0 0 1 0 1 1 0 1]
56	0.0010667	[1 1 0 0 1 1 0 0 1 1]	[1 1 0 0 1 1 0 0 1 1]
57	0.00105	[1 0 1 1 0 1 1 0 0 1]	[0 1 1 1 0 1 1 0 0 1]
58	0.00102	[0 1 1 1 0 1 0 0 1 1]	[1 1 1 0 0 0 0 0 0 1]
59	0.00097667	[1 1 0 0 1 0 0 1 1 1]	[0 1 1 1 0 1 0 0 1 1]
60	0.00089	[1 0 0 0 0 0 0 0 1 0]	[1 0 1 1 0 1 1 0 0 1]
61	0.00082	[0 0 1 1 0 0 0 1 1 0 0]	[0 0 1 1 0 0 0 1 1 0 0]
62	0.00044333	[1 0 0 0 0 1 0 0 0 1 0]	[0 1 1 1 0 0 1 1 1 1 0]
63	3.00E-05	[0 1 1 1 0 0 1 1 1 1 0]	[1 0 0 0 0 1 0 0 0 0 1 0]
	平均長度	5.49	5.49

表 5.2 與通道環境匹配之 $RVLC_3$ 碼書

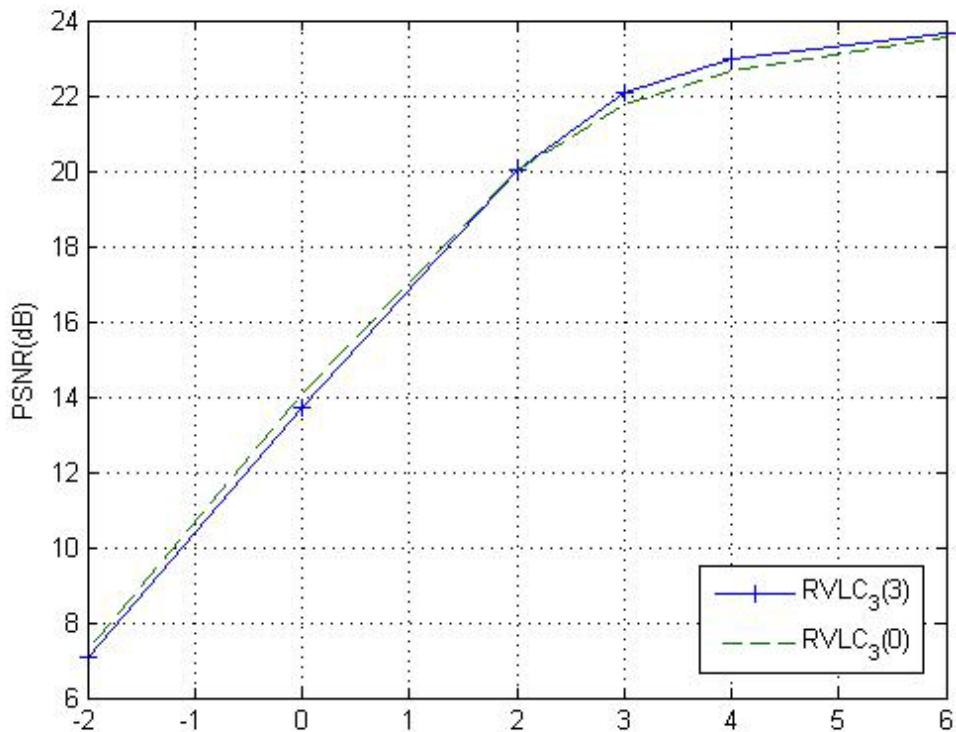


圖 5.2 不同通道環境訓練所得之索引指定的效能比較

分析： $RVLC_3(3)$ 為實驗一在通道訊雜比 3dB 環境下訓練所得的索引指定，而 $RVLC_3(0)$ 則為 0dB 通道環境下訓練所得的另一組碼書。由圖 5.2 可明顯看出 $RVLC_3(3)$ 在通道環境 3dB 時訊源解碼效能較 $RVLC_3(0)$ 約有 0.33dB 的改善，而 $RVLC_3(0)$ 則是在 0dB 附近時效能較 $RVLC_3(3)$ 好約 0.4dB。由此我們也得知訓練用通道環境的不同造成了最佳化索引指定的差異，進而影響了解碼效能，因此也可證明索引指定的最佳化設計確實具有通道匹配的特性。

5.3 實驗三

目 的：探討軟性解碼輸出採用不同檢測方式的效能

模擬環境：依 5.1 節的敘述進行相關設定，針對 3.4 節兩種不同的索引估計方法進行模擬，分別是最大後驗機率原則與建議之維特比演算法。

結 果：

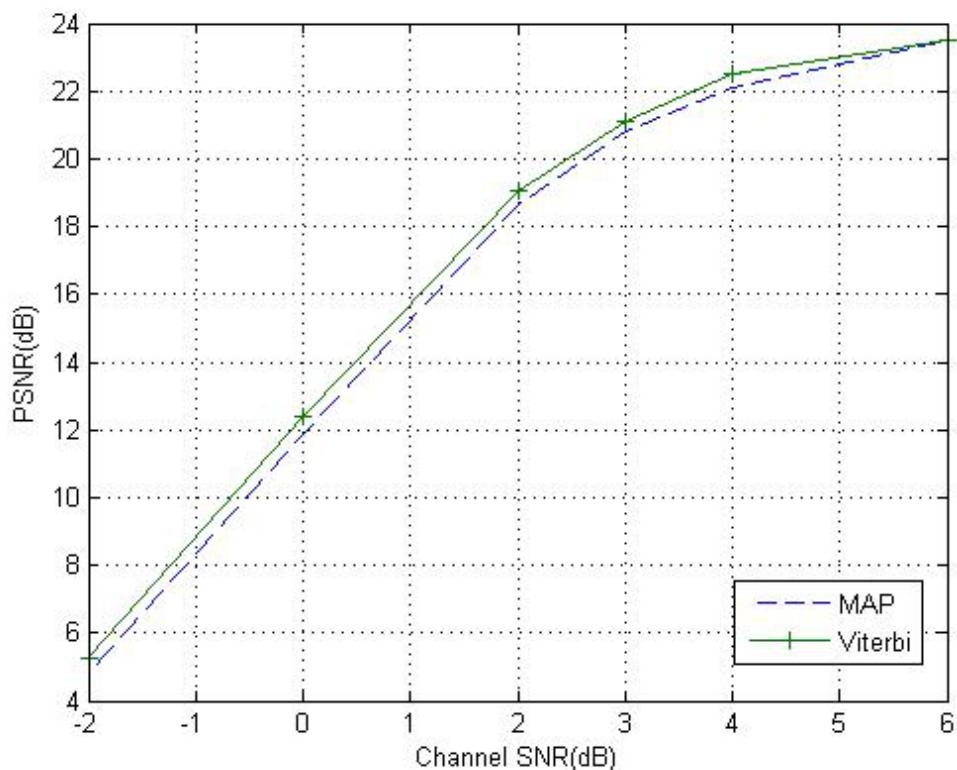


圖 5.3 $RVLC_1$ 之不同索引檢測方式比較

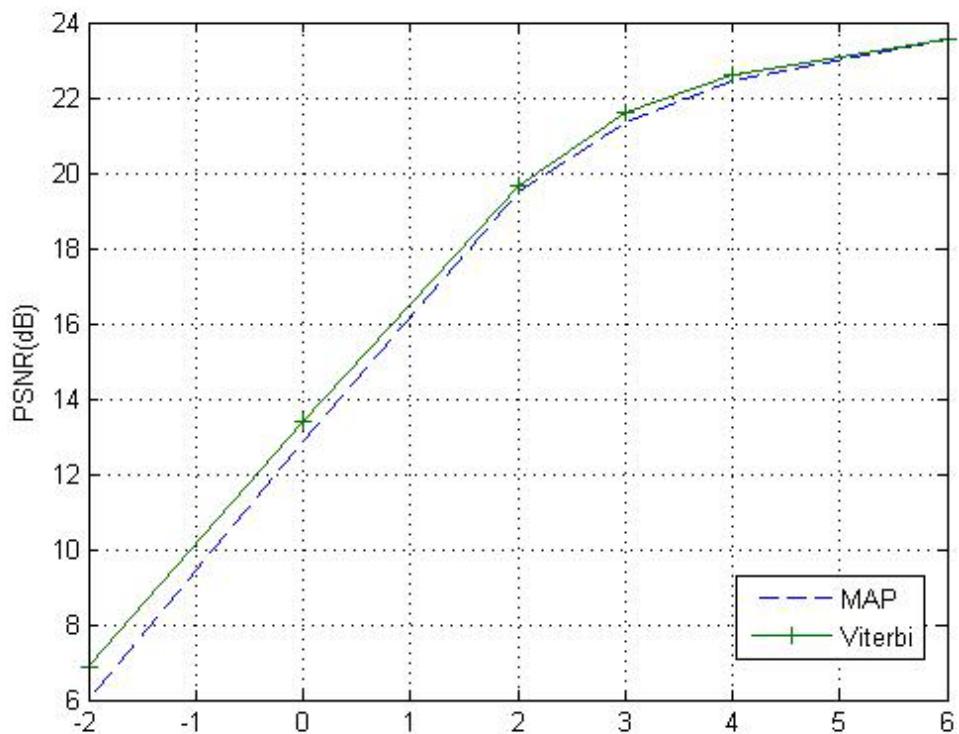


圖 5.4 $RVLC_2$ 之不同索引檢測方式比較

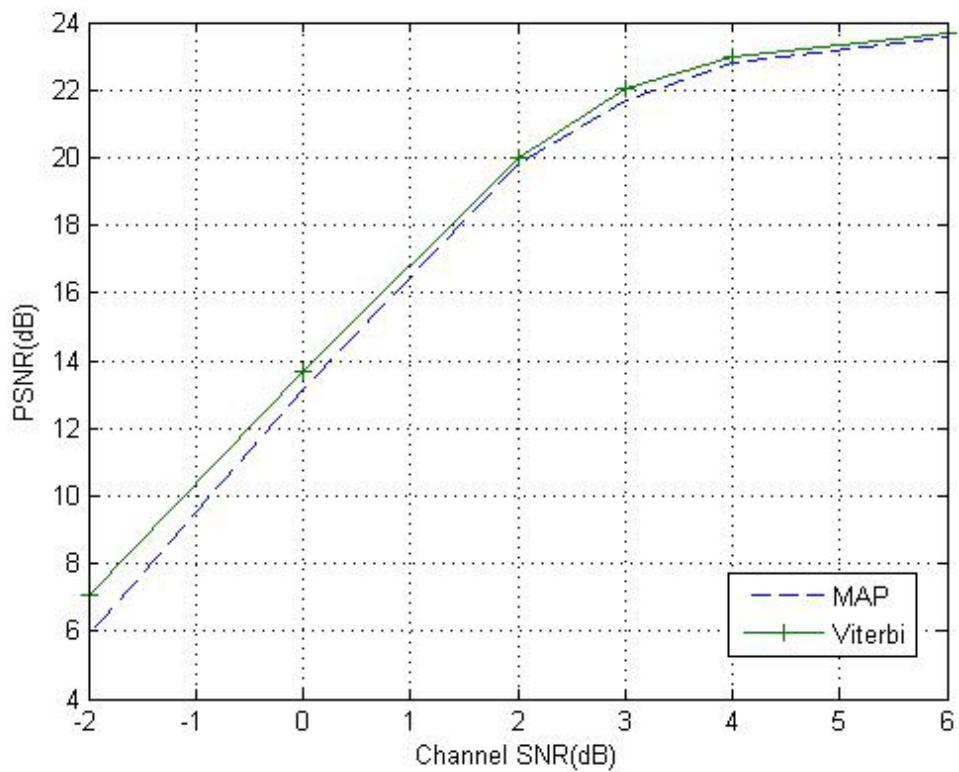
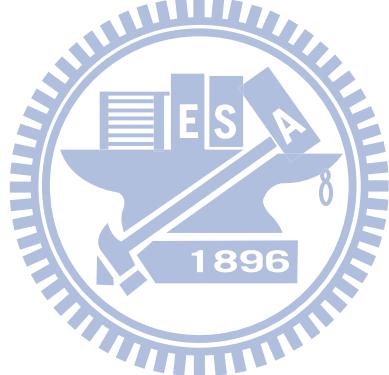


圖 5.5 $RVLC_3$ 之不同索引檢測方式比較

分 析：整體而言，三種可變長度碼使用維特比演算法的效能皆較 MAP 來得好，而兩種估測方法的效果在通道環境差時 (-2 、 0 dB) 較通道環境好 (2 、 3 、 4 、 6 dB) 差異更大，大約有 1 dB 的改進。我們的推測是通道環境好時，索引後驗機率的計算較為可靠，使得兩種估測方法較無太大的差別。但通道環境差時，使用 MAP 估測的索引組合之位元長度不等於 N 的錯誤可能性較大。反之，採用維特比演算法則可保證找出一條符合 N 與 T 中機率最大的索引組合。



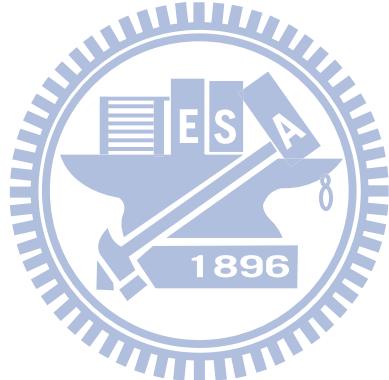
第六章 結論與未來展望

有鑑於國人對提升醫療品質的需求，居家式的醫療照護在未來將日趨普及，對於人體生理訊號重建品質的要求也越來越高。本論文提出一個可變長度碼的軟性訊源解碼機制，同時推導基於解碼效能而設計的雙向性可變長度碼，並將其運用於心電圖的訊號壓縮。

首先我們針對個別患者之心電圖進行向量量化，利用碼向量索引間的機率分佈不均產生一組霍夫曼碼，接著據此發展基於最小碼字長度與具有錯誤更正能力的非對稱性 RVLC。本論文中採用的軟性訊源解碼機制，主要是利用個別封包的索引與位元數目為已知的條件，發展出索引層級的籬柵圖，並以此配合通道的軟性資訊與一階事前相關性進行解碼。至於非對稱性 RVLC 碼的索引指定，是將訊源解碼效能納入設計的考量，逐次置換以尋找索引與碼字間最佳的對應關係，以期能增進整體傳輸系統的強健性。

現今行動通訊的日漸蓬勃，各種多媒體資料如音樂、影像…等等的傳輸將是熱門的議題，各式各樣的壓縮方法也如雨後春筍般地出現，雖然彼此間方法有其不同之處，但雙向性可變長度碼已逐漸納入各個主流規格，因此其解碼機制的設計便成為一個重要的研究方向。可變長度解碼的問題主要還是在於索引同步的控制，RVLC 提供了一

個編碼機制，使得在解碼時能夠進行前向與後向性的即時解碼，但至今仍然未見有利用其雙向解碼能力的軟性輸出解碼機制。本論文中所採用的解碼演算法受限於籬柵結構的複雜性而執行遞迴運算，使得其解碼效率在即時性的運用上面臨一定的考驗。除此之外，在我們模擬的實驗環境下，僅考慮無記憶性的白高斯通道，並無法代表真實多變的通訊環境。因此未來如何設計一套專屬於 RVLC 的即時性雙向軟性訊源解碼機制，並將其應用在實際的通訊環境下，將是一個重大的挑戰。



參考文獻

- [1] 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.
- [2] R. Thobaben and J. Kliewer, “Low-Complexity Iterative Joint Source–Channel Decoding for Variable-Length Encoded Markov Sources,” *IEEE Trans. Commun.*, vol. 53, no. 12, pp. 2054–2046, Dec. 2005.
- [3] J. Kliewer and R. Thobaben, “Iterative Joint Source–Channel Decoding of Variable-Length Codes Using Residual Source Redundancy,” *IEEE Trans. Wireless Commun.*, vol. 4, no. 3, pp. 919–929, 2005.
- [4] Y. Takishima, M. Wada, and H. Murakami, “Reversible Variable Length Codes,” *IEEE Trans. Commun.*, vol. 43, pp. 158–162, Feb.–Apr. 1995.
- [5] C. W. Tsai and J. L. Wu, “On Construct the Huffman–Code–Based Reversible Variable–Length Codes,” *IEEE Trans. Commun.*, vol. 49, pp. 1506–1509, Sept. 2001.
- [6] Ksenija Lakovic and John Villasenor, “On Design of Error–Correcting Reversible Variable Length Codes,” *IEEE Commun. Lett.*, vol. 6, no. 8, pp. 337–339, Aug. 2002.
- [7] Kenneth Zeger and Allen Sersho, “Pseudo–Gary Coding,” *IEEE Trans. Commun.*, vol. 38, no. 12, Dec. 1990.
- [8] Susanna Kaiser and Maja Bystrom, “Soft Decoding of Variable–Length Codes,” in *Proc. IEEE Int. Conf. Commun.*, 2000, pp. 1203–1207.
- [9] Khlid Sayood, Hasan H. Otu, and Nejat Demir, “Joint Source/Channel Coding for Variable Length Codes,” *IEEE Trans. Commun.*, vol. 48, no. 5, May. 2000.

- [10] V. buttigieg and P. G. Farrell, "A maximum a-posteriori(MAP) decoding algorithm for variable-length error-correcting codes," in *Codes and cyphers:Cryptography and coding IV*, Essex, England, The Institute of Mathematics and its Application, pp. 103–119, 1995.
- [11] R. Bauer and J. Hagenauer, "Symbol by symbol MAP decoding of variable length codes," accepted for *3rd ITG Conference Source and Channel Coding*, Munich, Germany, Jan. 17–19. 2000.
- [12] N. Götz, "Optimization of Bit Mappings for Iterative Source-Channel Coding," in *Proc. of International Symposium on Turbo Codes and Related Topics*, (Brest, France), pp. 255 – 258, Sept. 2003.
- [13] Linde, Y. , Buzo, A. , & Gray , R. (1980). An algorithm for vector quantization design. *IEEE Transactions On Communications*, 28(1), 84-94.
- [13] 古偉裕，「臨床心電圖入門」，眾文圖書股份有限公司
- [14] 何宗穎，「可攜式即時心電圖監控系統」，慈濟大學碩士論文，民國九十六年。
- [15] 戴玲玲，「基於疊代解碼效能的可變長度碼設計」，國立交通大學碩士論文，民國九十九年。
- [16] 葉遠呈，「基於額外資訊相關性的合併訊源通道編碼機制」，國立交通大學碩士論文，民國九十九年。