

國立交通大學

電機與控制工程學系

碩士論文

基於啾聲信號調變的演算法用於
單輸出多輸出通道的盲等化

A Chirp Modulation Based Algorithm for
SIMO Blind Channel Equalization



研究生：游江龍

指導教授：林清安 教授

中華民國九十四年七月

基於啾聲調變的演算法用於單輸入多輸出通道的盲等化

學生：游江龍

指導教授：林清安教授

國立交通大學電機與控制工程學系（研究所）碩士班

摘 要

我們提出了一個基於啾聲信號調變的盲等化適應性演算法，我們的方法使用一個改良的目標函數同時兼顧消除符碼間干擾以及提升信號增益，並且得出一個最佳的加權因子在符碼間干擾和信號增益之間取得平衡，以改善原始基於啾聲信號調變的演算法在多輸出通道性能不佳的缺點。模擬結果顯示我們方法的性能在單輸出和多輸出通道都優於原始的適應性演算法（特別是在多輸出的情況），而其穩態時的性能在多輸出通道的情況下亦與另一個基於啾聲信號調變的批次演算法相近。

A Chirp Modulation Based Algorithm for SIMO Blind Channel Equalization

student : Chiang-Lung Yu

Advisors : Dr. Ching-An Lin

Department (Institute) of Electrical and Control Engineering
National Chiao Tung University

ABSTRACT



We propose an improved adaptive algorithm for blind channel equalization based on chirp modulation. Our method utilizes an improved objective function that removes ISI and amplifies signal gain simultaneously. An optimal weighting factor is also derived for the balance between ISI and signal gain. It turns out that our method exhibits better performance since the original chirp modulation based adaptive algorithm considers only ISI removal and thus fails to improve performance in SIMO channel. The simulation results confirm the superiority of proposed algorithm over the original one in both SISO and SIMO channels. The steady state performance of our method is also close to another chirp modulation based batch algorithm in SIMO case.

誌 謝

首先要感謝我的指導教授林清安老師，老師的指導讓我培養了正確的研究態度以及方法，適時的指引也讓我避免許多錯誤的嘗試，得以在學生生涯的最後一年順利的完成碩士論文。

另外也要感謝 606 實驗室的益生學長和建賢學長，他們總是不吝於給予課業、生活上的經驗以及幫助(還有好吃的水果)，讓我在進入研究所之後很快的進入狀況而不至於跌跌撞撞。

最後要感謝我的父母以及女友盈盈，有他們在我身邊為我加油打氣、分憂解勞，是我能夠一直向前的原動力。



目錄

中文摘要	i
英文摘要	ii
誌謝	iii
目錄	iv
表目錄	vii
圖目錄	viii
1 序論	1
2 以啾聲調變信號實現通道的盲等化	3
2.1 啾聲信號的特性	3
2.2 系統模型	5
2.2.1 單輸入多輸出通道	5
2.2.2 基於啾聲信號調變的系統架構	7
2.3 基礎假設	8



2.4	通道的盲等化	9
2.4.1	符碼間干擾	10
2.4.2	以啣聲信號調變/解調後的相位特性	11
2.4.3	調變/解變非同步的情況	12
2.4.4	盲等化成立的充要條件	13
3	基於啣聲信號調變的演算法	16
3.1	批次演算法	16
3.2	最陡坡降法	19
3.3	多輸入通道的性能分析	20
3.4	改良的演算法	21
3.4.1	雜訊統計特性的分析	21
3.4.2	改良的目標函數	23
3.4.3	加權因子 γ	24
3.4.4	演算法流程	26
3.4.5	穩定性分析	27
4	模擬結果	31
4.1	多路徑通道參數	31
4.2	單輸出及多輸出通道的性能比較	33
4.2.1	單輸入單輸出通道的性能比較	33
4.2.2	單輸入多輸出通道的性能比較	34
4.2.3	模擬結果比較	35

4.3	不同 SNR 的性能比較	37
4.4	加權因子 γ 值對 SINR 的影響	39
4.5	錯誤發生率的比較	42
5	結論	44
	參考文獻	46



表目錄

4.1 多路徑通道參數	31
4.2 離散時間單輸出通道響應	33
4.3 離散時間多輸出通道響應	35



圖目錄

2.1	等效多輸出通道	6
2.2	基頻等效模型	8
4.1	多路徑通道響應	32
4.2	單輸入單輸出通道 SNR=20dB 的性能比較	36
4.3	單輸入多輸出通道 SNR=20dB 的性能比較	36
4.4	不同 SNR 改良的演算法的性能比較	37
4.5	不同 SNR 改良的演算法與批次演算法的性能比較	38
4.6	SNR=20dB 時不同 γ 值的性能比較	40
4.7	SNR=15dB 時不同 γ 值的性能比較	40
4.8	SNR=10dB 時不同 γ 值的性能比較	41
4.9	SNR=5dB 時不同 γ 值的性能比較	41
4.10	BPSK 的位元錯誤發生率比較	43
4.11	PAM4 的符碼錯誤發生率比較	43

第 1 章 序論

在許多無線通訊的應用之中，其傳輸環境往往是高失真、高雜訊甚至經常是時變的，為了對抗如此惡劣的傳輸環境，一個常見的手段是週期性地傳送已知的訓練符碼 (training symbols)，藉由傳送已知的訓練符碼以及接收端所接收到的訊號即可判別傳輸的通道，或是直接等化 (equalize) 傳送端信號，以此來彌補傳輸過程所帶來的失真以及適應通道的變化。

隨著通訊技術的進步，對於資料傳輸率 (data rate) 的要求越來越高，然而在有限的頻寬和傳輸功率的限制之下，週期性地傳送訓練符碼卻成了提高資料傳輸率的障礙，同時也浪費了不必要的功率在不具有真正資訊的訓練符碼上。因此如果能夠不依賴訓練符碼的幫助而僅僅由接收端失真的訊號來判別通訊通道或等化通道輸出，就可以有效地提高資料傳輸率和節省傳輸功率，而盲判別、盲等化等等不需要訓練符碼的盲技巧就是因應這樣的需求所產生的新技術。

盲判別、盲等化的概念最早是由 Godard [1] 所提出，而盲技巧真正廣泛地被研究、發展是在 L. Tong 提出基於二階統計特性 (second order statistics) 的方法 [2] 之後。隨後發展的許多基於二階統計的方法如子空間法 [3] 等等，其共通特性是對於傳輸通道的基本假設要求相當嚴苛，因此大大地限制了其應用範圍。而最近盲技巧的研究方向逐漸轉為在前級對傳送信號作前處理 (pre-processing)—在傳送端以特殊的方式

改變傳送信號的振幅或是相位來產生一些特殊性質，如此一來即使沒有訓練符碼的幫助，仍然可以由傳送端刻意引入信號中的特性來幫助接收器進行盲判別或盲等化。由於使用前處理的方法比二階統計特性的方法多了來自傳送端產生的特性，所以通常可以獲得較好的性能，對通道的限制與假設也大幅地降低，也因此可以適用於較具一般性的情形。

本論文改良了一種基於前處理的盲等化演算法 [4]，[4] 使用啾聲信號 (chirp signal) 來調變傳送信號，並利用調變後信號的特殊相位來實現通道的盲等化。然而我們發現原作者提出的適應性演算法有性能上的缺陷，因此在常見的多輸出通道下性能並不理想。本文分析了該演算法無法提高性能的原因，並藉此改良 [4] 的演算法使其在單輸入多輸出的系統亦能得到相當良好的性能。

本論文章節組織如下：

1. 第二章

介紹如何利用啾聲調變信號來盲等化通道。

2. 第三章

介紹並分析現有基於啾聲信號調變的演算法，接著提出改良的演算法。

3. 第四章

以實際模擬結果驗證改良後的演算法的性能，並與原始的演算法作比較。

4. 第五章

整合三章分析以及第四章模擬的結果並且作出結論。



第 2 章 以啣聲調變信號實現通道的盲等化

在這一章將會建立我們所使用的系統模型、基本假設，並且介紹 [4] 如何以啣聲信號調變的特性來實現通道盲等化的理論基礎，以利之後的分析及推演。

2.1 啣聲信號的特性

在介紹如何以啣聲調變信號來實現通道的盲等化之前，有必要先了解啣聲信號的一些特性。啣聲信號具有以下特殊的型式：

$$c(k) = e^{j2\pi f k^l} \quad (2.1)$$

其中 f 、 l 都是設計參數，為了簡化分析這裡將選擇 $l = 2$ ，而 $c(k)$ 即為啣聲信號並且具有以下幾個特殊的性質：

1. 週期性

連續時間 (continuous time) 的啣聲信號本身並不具備週期特性，觀察其型式也可以發現其相位的變化會隨著時間而變得越來越快；但是離散時間 (discrete time) 的啣聲信號則具有週期性而且週期與參數 f 有密切的關係。為了得出啣聲信號週期性與參數 f 之間的關係，我們先假設其週期為 P ，因此週期性的啣聲信號可以

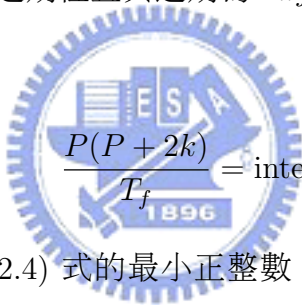
表示為：

$$\begin{aligned}c(k) &= c(k + P) && \forall k \\e^{j2\pi f k^2} &= e^{j2\pi f (k+P)^2} && \forall k \\&= e^{j2\pi f (k^2 + 2kP + P^2)} && \forall k\end{aligned}\quad (2.2)$$

觀察 (2.2) 式可知，若 (2.2) 式要對所有的時間 k 成立 (也就是具備週期性)，必須滿足下式：

$$f(2kP + P^2) = \text{integer} \quad \forall k \quad (2.3)$$

上式利用到指數函數的週期性且其週期為 $2\pi j$ 。若進一步選擇 $T_f = 1/f$ ，上式可再改寫為：


$$\frac{P(P + 2k)}{T_f} = \text{integer} \quad \forall k \quad (2.4)$$

對於所有 k 值皆滿足 (2.4) 式的最小正整數 P 即為啾聲信號的週期，如果選擇 $T_f = 1/f$ 為一質數 (prime number)，則週期 P 就會等於 T_f 。因為 (2.4) 式中分子右邊的項 $P + 2k$ 是一個隨著 k 變動的值，其範圍可以從 $-\infty$ 到 ∞ ，因此 (2.4) 式如果要對所有的 k 都能被整除， P 就必須等於 T_f 的倍數，其最小的正整數解就是 $P = T_f$ 。

2. 等模數

$$\begin{aligned}|c(k)| &= |e^{j2\pi f k^2}| \\&= 1 \quad \forall k\end{aligned}\quad (2.5)$$

由於這個性質，信號經過啾聲信號調變後其模數 (modulus) 不會改變。

3. 非均勻相位增量

$$\begin{aligned}\frac{d}{dk}(\angle c(k)) &= \frac{d}{dk}(\angle e^{j2\pi f k^2}) \\ &= 4\pi f k\end{aligned}\quad (2.6)$$

由上式可知，啾聲信號的相位增量 (phase increment) 會隨著時間線性地增加。因此也有人把啾聲信號調變稱為—線性頻率調變 (linear frequency modulation)。

綜合以上三個殊特的性質我們可以知道，啾聲信號調變只改變信號的相位而不改變振幅，而且其相位的變化具有週期性，並且在週期內相位的變化是線性的。基於這幾點使得啾聲信號非常適合作為一個調變信號，而性質 1、3 更是通道盲等化能否實現的關鍵。



2.2 系統模型

在這一節之中會建立我們所使用的單輸入多輸出通道模型，以及介紹基於啾聲訊號調變的系統架構。

2.2.1 單輸入多輸出通道

考慮傳輸符碼 $s(k)$ 經由一個長度為 M 的有限脈衝響應 (finite impulse response) 通道來傳輸，並且在通道輸出作過取樣 (over sampling) 或是使用多接收器，則在通道輸出端會等效於有多個子輸出，如圖 2.1，而每一個子輸出 $x^i(k)$ 都可以表示為：

$$x^i(k) = \sum_{n=0}^{M-1} h^i(n)s(k-n) + n^i(k)\quad (2.7)$$

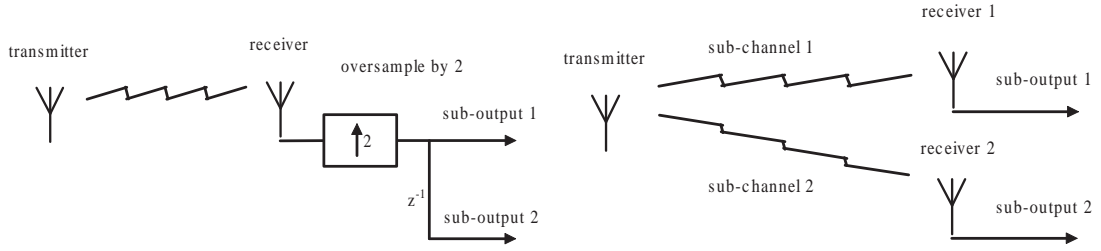


圖 2.1: 等效多輸出通道

其中 $h^i(0), \dots, h^i(M-1)$ 為第 i 個等效子通道的係數， $n^i(k)$ 為第 i 個等效子通道附加的雜訊。接著我們將 (2.7) 式旋積的型式改寫為向量的型式：

$$\mathbf{x}^i(k) = \mathbf{h}^i \mathbf{s}(k) + n^i(k) \quad (2.8)$$

$$\mathbf{h}^i = [h^i(0) \ h^i(1) \ \dots \ h^i(M-1)] \quad (2.9)$$

$$\mathbf{s}(k) = [s(k) \ s(k-1) \ \dots \ s(k-M+1)]^T \quad (2.10)$$

如果在各個子通道的輸出都堆疊 Q 筆 $x^i(k)$ ，我們可以把堆疊後的子輸出 $\mathbf{x}^i(k)$ 表示如下：

$$\mathbf{x}^i(k) = \begin{bmatrix} x^i(k) \\ x^i(k-1) \\ \vdots \\ x^i(k-Q+1) \end{bmatrix} = \mathbf{H}^i \mathbf{s}(k) + \mathbf{n}^i(k) \quad (2.11)$$

其中 \mathbf{H}^i 為第 i 個等效子通道的旋積矩陣 (convolution matrix)，其型式如下：

$$\mathbf{H}^i = \begin{bmatrix} \mathbf{h}^i & 0 & \dots & \dots & 0 \\ 0 & \mathbf{h}^i & 0 & \dots & 0 \\ \vdots & & \ddots & \ddots & \vdots \\ 0 & \dots & \dots & 0 & \mathbf{h}^i \end{bmatrix} \in \mathcal{C}^{Q \times M+Q-1} \quad (2.12)$$

$$\underline{\mathbf{s}}(k) = [s(k) \ s(k-1) \ \cdots \ s(k-(M+Q-2))]^T \quad (2.13)$$

$$\underline{\mathbf{n}}^i(k) = [n(k) \ n(k-1) \ \cdots \ n(k-Q+1)]^T \quad (2.14)$$

最後再把所有子通道的輸出 $\underline{\mathbf{x}}^0(k), \dots, \underline{\mathbf{x}}^{L-1}(k)$ 堆疊在一起，也就是：

$$\underline{\mathbf{x}}(k) = \begin{bmatrix} \underline{\mathbf{x}}^0(k) \\ \underline{\mathbf{x}}^1(k) \\ \vdots \\ \underline{\mathbf{x}}^{L-1}(k) \end{bmatrix} = \underline{\mathbf{H}}\underline{\mathbf{s}}(k) + \underline{\mathbf{n}}(k) \quad (2.15)$$

其中 L 為等效子通道的數目。

$$\underline{\mathbf{H}} = \begin{bmatrix} \underline{\mathbf{H}}^0(k) \\ \underline{\mathbf{H}}^1(k) \\ \vdots \\ \underline{\mathbf{H}}^{L-1}(k) \end{bmatrix} \in \mathcal{C}^{LQ \times M+Q-1} \quad (2.16)$$

$$\underline{\mathbf{n}}(k) = [\underline{\mathbf{n}}^0(k)^T \ \underline{\mathbf{n}}^1(k)^T \ \cdots \ \underline{\mathbf{n}}^{L-1}(k)^T]^T \in \mathcal{C}^{LQ} \quad (2.17)$$

(2.15) 式即為我們所使用的單輸入多輸出通道模型。

2.2.2 基於啾聲信號調變的系統架構

建立起單輸入多輸出的通道模型後，接著介紹 [4] 中基於啾聲信號調變的系統架構。

$b(k)$ 表示帶有資訊的資訊符碼 (information symbols) 並以啾聲信號 $c(k)$ 加以調變得到傳輸符碼 $s(k)$ 。也就是：

$$s(k) = b(k)c(k) \quad (2.18)$$

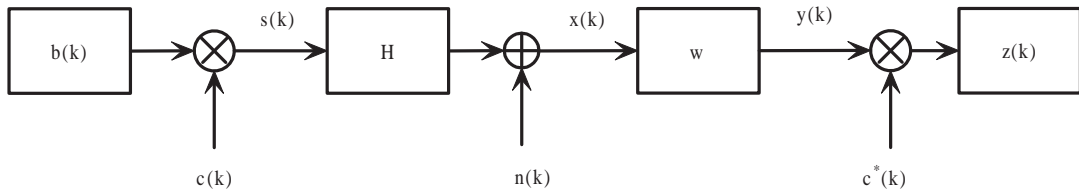


圖 2.2: 基頻等效模型

接著如同 (2.15) 式般，傳輸符碼 $s(k)$ 經由一個單輸入多輸出的通道傳輸，接著在通道輸出端以等化器等化堆疊後的 $\mathbf{x}(k)$ ，而等化器的輸出 $y(k)$ 可以表示為：

$$y(k) = \mathbf{w}^T \mathbf{x}(k) \quad (2.19)$$

其中 $\mathbf{w} \in \mathcal{C}^{LQ}$ 為等化器的係數， Q 為等化器的時間長度 (temporal length)，和每一個子通道輸出所堆疊的數目相同。最後利用 $c^*(k)$ 解調等化器輸出 $y(k)$ 得到系統最後的輸出 $z(k)$ ：

$$z(k) = c^*(k)y(k) \quad (2.20)$$

以上關係可以化成一個基頻等效模型如圖 2.2 所示，此即為基於啾聲信號調變的盲等化架構。

2.3 基礎假設

以下是本文會用到的基礎假設：

1. $b(k)$ 為一個廣義靜止 (wide sense stationary) 的隨機程序，並且為獨立同分佈 (independent, identically distributed)，其期望值為 0，變異數為 σ_s^2 ，而且 $b(k)$ 必須限制為實數。

2. $n(k)$ 為白色高斯雜訊 (white Gaussian noise)，其期望值為 0，變異數為 σ_n^2 ，並且與 $s(k)$ 無關 (independent)。

依據第一個假設可以求得 $s(k)$ 的統計特性如下：

$$\begin{aligned} E\{s(k)\} &= E\{b(k)c(k)\} \\ &= E\{b(k)\}c(k) \\ &= 0 \end{aligned} \tag{2.21}$$

$$\begin{aligned} E\{s(k)s^*(l)\} &= E\{b(k)b^*(l)\}c(k)c^*(l) \\ &= \sigma_b^2\delta(k-l)c(k)c^*(l) \\ &= \sigma_b^2\delta(k-l)|c(k)|^2 \\ &= \sigma_b^2\delta(k-l) \end{aligned} \tag{2.22}$$

由於啾聲調變信號 $c(k)$ 的特殊性質，使得 $s(k)$ 具有和 $b(k)$ 相同的期望值和自相關函數 (auto-correlation function)，從 (2.22) 式亦可以發現 $s(k)$ 也是一個獨立同分佈廣義靜止的隨機程序。

2.4 通道的盲等化

通道盲等化的目的是希望在沒有訓練符碼的幫助下依然能夠等化通道輸出，使得等化器後輸出盡量地接近原始的傳輸符碼。而如何利用啾聲信號的特殊性質來實現通道的盲等化將會在這一節說明。

2.4.1 符碼間干擾

除了通道雜訊的影響之外，另一個造成信號嚴重失真的原因往往是通道存在記憶 (memory)，這使得接收到的訊號會與現在以及過去的輸入有關，就如同 (2.7) 式一般。因此在通道存在記憶的情況下，接收到的訊號之中除了要等化的信號，還會有所謂的“符碼間干擾” (inter-symbol interference)，以 (2.7) 式為例並且暫時不考慮雜訊：

$$x^i(k) = \underbrace{h^i(0)s(k)}_{\text{信號}} + \sum_{n=1}^{M-1} \underbrace{h^i(n)s(k-n)}_{\text{符碼間干擾}} \quad (2.23)$$

在上式之中我們選擇 $h^i(0)s(k)$ 為要等化的信號，其它的符碼就會以符碼間干擾的型式存在，使得接收到的訊號產生失真 i.e. $x^i(k) \neq h^i(0)s(k)$ 。而等化的目的就是尋找適當的等化器係數 \mathbf{w} 來消除符碼間干擾的影響。

接著考慮盲等化架構中的等化器輸出 $y(k)$ ，暫時忽略雜訊的影響並將 (2.15) 代入 (2.19) 式：



$$\begin{aligned} y(k) &= \mathbf{w}^T \mathbf{x}(k) \\ &= \mathbf{w}^T \mathbf{H} \mathbf{s}(k) \\ &= \mathbf{g}^T \mathbf{s}(k) \end{aligned} \quad (2.24)$$

其中 $\mathbf{g} = (\mathbf{H}^T \mathbf{w}) \in \mathcal{C}^{M+Q-1}$ 為通道加等化器的整體響應。觀察上式可以發現，如果能夠調整 \mathbf{w} 使得通道加等化器的整體響應 \mathbf{g} 只存在一個不為 0 的元素，也就是：

$$\begin{cases} g(m_0) \neq 0 \\ g(m) = 0, \quad \forall m \neq m_0 \end{cases} \quad (2.25)$$

如此一來 $y(k) = g(m_0)s(k - m_0)$ ，等化器輸出 $y(k)$ 只與傳輸符碼 $s(k - m_0)$ 一項有關，通道存在記憶所造成的影響——符碼間干擾就可以被消除掉而達到等化的目的。

2.4.2 以啣聲信號調變/解調後的相位特性

為了分析以啣聲信號調變/解調後的相位特性，我們將 (2.24) 式展開如下：

$$\begin{aligned}
 y(k) &= \underline{\mathbf{g}}^T \underline{\mathbf{s}}(k) \\
 &= g(0)b(k)e^{j2\pi f k^2} + g(1)b(k-1)e^{j2\pi f (k-1)^2} \\
 &\quad + \cdots + g(M+Q-2)b(k-(M+Q-2))e^{j2\pi f (k-(M+Q-2))^2}
 \end{aligned} \tag{2.26}$$

根據上式，以 $c^*(k)$ 加以解調後的輸出可以寫成：

$$\begin{aligned}
 z(k) &= c^*(k)y(k) \\
 &= e^{-j2\pi f k^2} y(k) \\
 &= \underbrace{g(0)b(k)}_{\text{unmodulated}} + g(1)b(k-1)e^{j2\pi f (-2k+1)} + \cdots \\
 &\quad + g(q)b(k-q)e^{j2\pi f (-2kq+q^2)}
 \end{aligned} \tag{2.27}$$

觀察 (2.27) 式可以發現一個現象，經由 $c^*(k)$ 解調後只有第一項 $g(0)b(k)$ 是未調變的 (unmodulated)，其它項都帶有改變相位的指數項。

接著回想之前提到的限制— $b(k)$ 必須為實數，在這個限制之下如果能夠調整等化器係數 $\underline{\mathbf{w}}$ 使得通道加等化器的整體響應 $\underline{\mathbf{g}}$ 也是實數的，則 (2.27) 式就只有第一項 $g(0)b(k)$ 會是實數，其它項都因為指數項的關係而一定是複數。此時如果使得 $z(k)$ 變成實數，是否表示 (2.27) 式中帶有指數項的部分都不存在？也就是 $z(k) = g(0)b(k)$ ？在這個情況下整體響應 $\underline{\mathbf{g}}$ 滿足下式：

$$\begin{cases} g(0) & \neq 0 \\ g(m) & = 0, \quad \forall m \neq 0 \end{cases} \tag{2.28}$$

明顯地上式與不存在符碼間干擾的條件 (2.25) 式一致，但 $z(k)$ 為實數是否暗示 $z(k) = g(0)b(k)$ 仍需要檢視 (2.27) 式的關係。

2.4.3 調變/解變非同步的情況

在討論 $z(k)$ 為實數與 $z(k) = g(0)b(k)$ 之間的關聯性之前，我們先放寬一點限制來討論更為一般性的情況。

考慮調變與解調的時機不一定同步 (synchronous)， $c^*(k)$ 與 $c(k)$ 差了一個時間差 m ，因此 (2.27) 式應改寫成：

$$z(k) = c^*(k - m)y(k) \quad (2.29)$$

如果我們選擇啣聲信號 $c(k)$ 的頻率 $f = 1/T_f$ ，並且 T_f 是一個質數，則啣聲信號會是一個週期性函數其週期恰等於 T_f 。對於任何的同步誤差 m 我們都可以將它表示成 $m_0 + dT_f$ 的型式，其中 d 為整數而 m_0 為小於啣聲信號週期 T_f 的一個正整數，因此可以將 (2.29) 式改寫為：

$$\begin{aligned} z(k) &= c^*(k - m_0 - dT_f)y(k) \\ &= c^*(k - m_0)y(k) \end{aligned} \quad (2.30)$$

$$= \sum_{q=0}^{M+Q-2} g(q)b(k - q)e^{j2\pi f[(k-q)^2 - (k-m_0)^2]} \quad (2.31)$$

根據 (2.31) 式，現在 $z(k)$ 只有未調變項 (unmodulated term) 的情況應寫為：

$$z(k) = g(m_0)b(k - m_0) \quad (2.32)$$

而在 $b(k)$ 為實數的限制下， $z_I(k) = 0$ 可以表示如下：

$$z_I(k) = \sum_{q=0}^{M+Q-2} b(k-q) \times [g_R(q) \sin(2\pi\theta(q)) + g_I(q) \cos(2\pi\theta(q))] = 0 \quad (2.33)$$

其中 $\theta(q) = -2k(q - m_0)f + (q^2 - m_0^2)f$ 。值得注意的是當 $q = m_0$ 時， $\theta(q)$ 將會等於 0，所以：

$$\begin{cases} \sin(2\pi\theta(m_0)) = 0 \\ \cos(2\pi\theta(m_0)) = 1 \end{cases} \quad (2.34)$$

最後將上式代入 (2.33) 式可以把 $z_I(k) = 0$ 表示為：

$$z_I(k) = b(k - m_0)g_I(m_0) + \sum_{q=0, q \neq m_0}^{M+Q-2} b(k-q)[g_R(q) \sin(2\pi\theta(q)) + g_I(q) \cos(2\pi\theta(q))] = 0 \quad (2.35)$$

2.4.4 盲等化成立的充要條件

首先觀察 (2.35) 式，等號的右邊可以視為 $b(k)$, $b(k-1)$, \dots , $b(k-Q+2)$ 的線性組合，根據之前的假設， $b(k)$ 是一個獨立同分佈的序列，依照統計獨立強於線性獨立的特性 [5] [6]，如果 $b(k)$, $b(k-1)$, \dots , $b(k-Q+2)$ 為統計獨立則對於足夠長的觀察時間，下列的向量將會是線性獨立 (但反之未必然)。

$$\begin{bmatrix} b(k) \\ b(k-1) \\ \vdots \\ b(k-Q+2) \end{bmatrix} \begin{bmatrix} b(k+1) \\ b(k) \\ \vdots \\ b(k-Q+3) \end{bmatrix} \begin{bmatrix} b(k+2) \\ b(k+1) \\ \vdots \\ b(k-Q+4) \end{bmatrix} \dots \quad (2.36)$$

因此對於所有的 k 都使得 $z_I(k) = 0$ 的情況下表示 (2.35) 式中 $b(k), b(k-1), \dots, b(k-Q+2)$ 的係數全為 0，也就是：

$$\begin{cases} g_I(m_0) = 0 \\ g_R(q) \sin(2\pi\theta(q)) + g_I(q) \cos(2\pi\theta(q)) = 0 \quad \forall q \neq m_0 \end{cases} \quad (2.37)$$

而上式成立是否就暗示 (2.25) 式成立？這個答案在 $\sin(2\pi\theta(q)), \cos(2\pi\theta(q)) \forall q \in (0, \dots, M+Q-2)$ 是線性獨立的情況下是肯定的，因為在這樣的情況下 (2.37) 式只存在無用解 (trivial solution)：

$$\begin{cases} g_I(m_0) = 0 \\ g_R(q) = g_I(q) = 0 \quad \forall q \neq m_0 \end{cases} \quad (2.38)$$

這個時候整體響應 $\underline{\mathbf{g}}$ 不但只有一個不為零的元素，而且還是實數 ($g(m_0) = g_R(m_0)$)，因此暗示了 (2.25) 式。

最後的條件剩下 $\sin(2\pi\theta(q)), \cos(2\pi\theta(q)) \forall q \in (0, \dots, M+Q-2)$ 必須為線性獨立，因為 $\theta(q) = -2k(q - m_0)f + (q^2 - m_0^2)f$ ，其中的 f 為啣聲信號的設計參數，我們可以藉由適當地選擇 f 的值來使得 $\sin(2\pi\theta(q)), \cos(2\pi\theta(q)) \forall q \in (0, \dots, M+Q-2)$ 為線性獨立。而 f 的選擇可以參考 [4] 中的定理，引用如下：

定理 1

若且唯若當 f 滿足以下條件時，(2.37) 式只存在無用解：

$$f \neq \frac{n}{4(q - m_0)} \quad 0 < |q - m_0| \leq M + Q - 2, \quad n \in (0, \pm 1, \pm 2, \dots) \quad (2.39)$$

綜合以上關係可以得到一個結論，在 f 使滿足 (2.39) 式並且 $b(k)$ 為實數的情況下，只要 $z_I(k) = 0$ ，通道加等化器的整體響應 $\underline{\mathbf{g}}$ 就只會有一個不為 0 的實數元素

$g_R(m_0)$ 而滿足 (2.25) 式，因此達到消除符碼間干擾的目的。所以在上述條件成立下，只要調整等化器係數 \mathbf{w} 使得 $z(k)$ 為實數就能夠盲等化通道輸出 $\mathbf{x}(k)$ ，而且完全不需要訓練符碼的幫助，也不需要得知 $b(k)$ 的二階統計特性如 σ_b^2 ，這有別於其它基於二階統計特性的演算法，如 [3] 等等。



第 3 章 基於啣聲信號調變的演算法

本章首先介紹兩種基於啣聲信號調變的演算法，在分析其性能上的缺點之後將提出一個可適用於單輸入多輸出通道，而且能夠追蹤快速變化通道的適應性演算法。

3.1 批次演算法

現在我們已經知道要達到等化通道的目的可以利用使 $z_I(k) = 0$ 這個手段來完成，為了方便接下來的分析，我們把 $z(k)$ 分解為實部與虛部的和，根據 (2.30) 式 $z(k)$ 可以寫成：

$$\begin{aligned} z(k) &= c^*(k - m_0)y(k) \\ &= \underbrace{[c_R(k - m_0)y_R(k) + c_I(k - m_0)y_I(k)]}_{z_R(k)} + j \underbrace{[c_R(k - m_0)y_I(k) - c_I(k - m_0)y_R(k)]}_{z_I(k)} \end{aligned} \quad (3.1)$$

如果 $z(k)$ 為實數則表示：

$$\begin{aligned} z_I(k) &= c_R(k - m_0)y_I(k) - c_I(k - m_0)y_R(k) \\ &= 0 \end{aligned} \quad (3.2)$$

接著將 (2.19) 式代入上式可以再改寫為：

$$-(\underline{\mathbf{w}}_R^T \underline{\mathbf{x}}_R(k) - \underline{\mathbf{w}}_I^T \underline{\mathbf{x}}_I(k)) c_I(k - m_0) + (\underline{\mathbf{w}}_R^T \underline{\mathbf{x}}_I(k) + \underline{\mathbf{w}}_I^T \underline{\mathbf{x}}_R(k)) c_R(k - m_0) = 0 \quad (3.3)$$

最後把等化器係數的部分 $\underline{\mathbf{w}}_R$ 、 $\underline{\mathbf{w}}_I$ 提出來並且排成矩陣的型式：

$$[c_R(k - m_0) \underline{\mathbf{x}}_I^T(k) - c_I(k - m_0) \underline{\mathbf{x}}_R^T \quad c_I(k - m_0) \underline{\mathbf{x}}_I^T + c_R(k - m_0) \underline{\mathbf{x}}_R^T(k)] \begin{bmatrix} \underline{\mathbf{w}}_R \\ \underline{\mathbf{w}}_I \end{bmatrix} = 0 \quad (3.4)$$

為了簡化表示，我們定義：

$$\underline{\mathbf{u}}_I(k) = [c_R(k - m_0) \underline{\mathbf{x}}_I^T(k) - c_I(k - m_0) \underline{\mathbf{x}}_R^T \quad (3.5)$$

$$c_I(k - m_0) \underline{\mathbf{x}}_I^T + c_R(k - m_0) \underline{\mathbf{x}}_R^T(k)]^T \in \mathcal{C}^{2LQ}$$

以及

$$\underline{\mathbf{v}} = \begin{bmatrix} \underline{\mathbf{w}}_R \\ \underline{\mathbf{w}}_I \end{bmatrix} \in \mathcal{C}^{2LQ} \quad (3.6)$$

接著將 (3.5)、(3.6) 代入 (3.4) 式後，可以得到一個相當簡單的型式：

$$\underline{\mathbf{u}}_I^T(k) \underline{\mathbf{v}} = 0 \quad (3.7)$$

藉由求解 (3.7) 式這個線性方程式即可達到盲等化通道的目的。

而這一節要介紹的批次演算法 (batch algorithm) 便是一次堆疊大量的 $\underline{\mathbf{u}}_I(k)$ 後再來求解如下的線性方程式：

$$\underline{\mathbf{U}}_I \underline{\mathbf{v}} = \underline{\mathbf{0}} \quad (3.8)$$

其中

$$\mathbf{U}_I = [\mathbf{u}_I(0) \ \mathbf{u}_I(1) \ \cdots \ \mathbf{u}_I(N-1)]^T \quad (3.9)$$

而 N 則是每一次堆疊 $\mathbf{u}_I(k)$ 的數目。

明顯地解 (3.8) 式這個線性方程式是在尋找 \mathbf{U}_I 的零空間向量，然而雜訊 $\mathbf{n}(k)$ 存在時會使得 \mathbf{U}_I 為滿秩 (full rank) 而無法找到非零的零空間向量，故只能尋找最小平方解 (least square solution)。所以在考慮雜訊存在的情況下，批次演算法應該改為尋找能夠最小化 $\|\mathbf{U}_I \mathbf{v}\|^2$ 的解，也就是：

$$\text{Minimize: } \|\mathbf{U}_I \mathbf{v}\|^2 = \mathbf{v}^T \mathbf{R}_{\mathbf{U}_I} \mathbf{v} \quad \mathbf{R}_{\mathbf{U}_I} = \mathbf{U}_I^T \mathbf{U}_I \quad (3.10)$$

$$\text{Subject to: } \|\mathbf{v}\|^2 = 1 \quad (3.11)$$

其中 $\|\mathbf{v}\|^2 = 1$ 的限制是為了要避免得到無用解，也就是 $\mathbf{v} = \mathbf{0}$ 的情況。而一般要得到最小平方解可以利用對 $\mathbf{R}_{\mathbf{U}_I}$ 進行奇異值分解 (singular value decomposition)，並且找出對應到最小奇異值的奇異向量即為最小化 (3.10) 式的解。

一般來說，要求 $\mathbf{U}_I \mathbf{v} = 0$ 的最小平方解，堆疊的數目 N 至少要大於等於 \mathbf{v} 的維度 ($2LQ$)。如果以等化器長度 $Q = 7$ ，接收器數目 $L = 4$ 為例，就需要堆疊至少 56 筆以上的 $\mathbf{u}_I(k)$ ，如果要得到滿意的性能更往往需要堆疊 200 筆以上，而接下來還需要計算 $\mathbf{R}_{\mathbf{U}_I} = \mathbf{U}_I^T \mathbf{U}_I$ 以及 $\mathbf{R}_{\mathbf{U}_I}$ 的奇異值分解，其計算量相當可觀。

由於過高的計算複雜度以及需要暫存大量的資料，使得批次演算法不適合用於低複雜度的硬體以及要求低延遲的即時通訊環境上，並考慮無線通訊通道本質上是迅速變化的，若不能即時 (或頻繁) 地更新等化器的係數，則盲等化的輸出將無法完全地追蹤通道的變化，這使得每 N 筆資料才能更新一次等化器係數的批次演算法不適合用於快速變化的通道。

3.2 最陡坡降法

接下來介紹另一個基於啾聲信號調變的演算法—最陡坡降法 (steepest descent method)。

在推導最陡坡降法之前需要先定義其目標函數 (objective function) J ，根據 (3.10) 式可以知道應該選擇目標函數如下：

$$J = \|\underline{\mathbf{U}}_I \underline{\mathbf{v}}\|^2 = \underline{\mathbf{v}}^T \underline{\mathbf{R}}_{\underline{\mathbf{U}}_I} \underline{\mathbf{v}} \quad (3.12)$$

將上式對 $\underline{\mathbf{v}}(k)$ 作偏微分可以得到其梯度函數：

$$\frac{\partial J(\underline{\mathbf{v}}(k))}{\partial \underline{\mathbf{v}}(k)} = 2\underline{\mathbf{R}}_{\underline{\mathbf{U}}_I} \underline{\mathbf{v}} \quad (3.13)$$

接著將梯度函數代入最陡坡降法的疊代公式：

$$\begin{aligned} \underline{\mathbf{v}}(k+1) &= \underline{\mathbf{v}}(k) - \frac{1}{2} \mu \frac{\partial J(\underline{\mathbf{v}}(k))}{\partial \underline{\mathbf{v}}(k)} \\ &= \underline{\mathbf{v}}(k) - \mu \underline{\mathbf{R}}_{\underline{\mathbf{U}}_I} \underline{\mathbf{v}}(k) \end{aligned} \quad (3.14)$$

為了減少一部份的計算量以及避免堆疊大量的向量 $\underline{\mathbf{u}}_I(k)$ ，可以將上式中的 $\underline{\mathbf{R}}_{\underline{\mathbf{U}}_I}$ 改以指數加權 (exponentially weighted) 的方式來更新如下：

$$\underline{\mathbf{R}}_{\underline{\mathbf{U}}_I}^k = \beta \underline{\mathbf{R}}_{\underline{\mathbf{U}}_I}^{k-1} + (1 - \beta) \underline{\mathbf{u}}_I(k) \underline{\mathbf{u}}_I^T(k) \quad (3.15)$$

其中 β 為遺忘因子 (forgetting factor)，其值在 $0 \sim 1$ 之間。

如果要更進一步減少計算量，還可以用瞬時誤差 $\|\underline{\mathbf{u}}_I(k) \underline{\mathbf{v}}(k)\|^2$ 來取代 $\|\underline{\mathbf{U}}_I \underline{\mathbf{v}}\|^2$ ，因此最陡坡降法可以再簡化成：

$$\underline{\mathbf{v}}(k+1) = \underline{\mathbf{v}}(k) - \mu \underline{\mathbf{u}}_I(k) \underline{\mathbf{u}}_I^T(k) \underline{\mathbf{v}}(k) \quad (3.16)$$


再根據 (3.7) 式， $z_I(k) = \mathbf{u}_I^T(k)\mathbf{v}(k)$ 並將之代入上式即可得到運算複雜度極低的最小均方 (least mean square) 演算法：

$$\mathbf{v}(k+1) = \mathbf{v}(k) - \mu z_I(k)\mathbf{u}_I(k) \quad (3.17)$$

然而經由實驗後發現，(3.14) 式的最陡坡降法與 (3.17) 式的最小均方法在多輸出 ($L \geq 2$) 通道的性能都不理想，我們將在下一節分析其性能不佳的原因。

3.3 多輸入通道的性能分析

為了分析演算法的性能，我們需要一個客觀的性能準則——輸出信號干涉雜訊比 (output signal to interference noise ratio, SINR)，其定義如下：



$$\text{SINR} = \frac{\sigma_s^2}{\sigma_i^2 + \sigma_n^2} \quad (3.18)$$

其中 σ_s^2 、 σ_i^2 和 σ_n^2 分別是信號、符碼間干擾與雜訊的功率。

由 SINR 的定義可以看出，一個演算法性能的好壞，取決於能否在消除符碼間干擾之外，還能提高信號的功率以及抑制雜訊的影響。基於這一點來看最陡坡降法，演算法本身的目的只是降低 $|z_I(k)|^2$ 使其儘量滿足 (2.38) 式，也就是只有消除符碼間干擾的部分，但是在能否提高信號的功率或是抑制雜訊上並無保證。

從另一方面來看，通道等化的目的是找一組 \mathbf{w} 使得整體響應 \mathbf{g} 只存在一個不為零的元素，而把 \mathbf{w} 與 \mathbf{g} 關連起來的是通道矩陣 \mathbf{H} ，也就是 $\mathbf{g} = \mathbf{H}^T \mathbf{w}$ 。這一個線性方程式的解存不存在決定於 \mathbf{H}^T 的維度，根據 (2.16) 式， \mathbf{H}^T 的維度為 $(M+Q-1) \times LQ$ ，在使用過取樣或者使用多接收器的情況下 ($L \geq 2$)，這個時候 $(M+Q-1) < LQ$ ，所以 $\mathbf{g} = \mathbf{H}^T \mathbf{w}$ 這個線性方程式會有無限多組解。因此即使利用 $z_I(k) = 0$ 使得符碼間干

擾的部分等於 0，信號增益 $g_R(m_0)$ 的值仍然有無限多種可能，我們甚至可以找到無限多組 $\underline{\mathbf{w}}$ 使得 $z_I(k) = 0$ 消除符碼間干擾，但是信號的增益 $g_R(m_0) = 0$ 。

因為以上理由，最陡坡降法使用在單輸入多輸出通道時，往往演算法已經收斂到 $z_I(k) = 0$ 的解，信號的增益 $g_R(m_0)$ 卻仍然無法提升甚至停留在很低的值，所以 SINR 通常不高。

3.4 改良的演算法

這一節將分析雜訊的統計特性來找出抑制雜訊的方法，以及得出一個改良過的目標函數來提升信號增益 $g_R(m_0)$ ，接著將抑制雜訊、提高信號增益的方法結合起來得到一個可適用於多輸出通道的適應性演算法，最後驗證改良演算法的穩定性。

3.4.1 雜訊統計特性的分析

首先討論通道輸出端存在雜訊 $\underline{\mathbf{n}}(k)$ 的情況下，對於 $z(k)$ 的統計特性會有什麼影響，根據 (2.29) 式， $z(k)$ 的變異數可以表示如下：

$$\begin{aligned} E\{z(k)z^*(k)\} &= E\{c^*(k - m_0)y(k)c(k - m_0)y^*(k)\} \\ &= E\{y(k)y^*(k)\} \end{aligned} \quad (3.19)$$

將 (2.19)、(2.29) 式代入上式：

$$= E\{[\underline{\mathbf{w}}^T \underline{\mathbf{H}}\underline{\mathbf{s}}(k) + \underline{\mathbf{w}}^T \underline{\mathbf{n}}(k)][\underline{\mathbf{w}}^T \underline{\mathbf{H}}\underline{\mathbf{s}}(k) + \underline{\mathbf{w}}^T \underline{\mathbf{n}}(k)]^*\} \quad (3.20)$$

根據 $n(k)$ 與 $s(k)$ 無關的假設，可以再改寫如下：

$$\begin{aligned}
 &= E\{\underline{\mathbf{w}}^T \underline{\mathbf{H}} \underline{\mathbf{s}}(k) \underline{\mathbf{s}}^H(k) \underline{\mathbf{H}}^H \underline{\mathbf{w}}^*\} + E\{\underline{\mathbf{w}}^T \underline{\mathbf{n}}(k) \underline{\mathbf{n}}^H(k) \underline{\mathbf{w}}^*\} \\
 &= \underline{\mathbf{g}}^H E\{\underline{\mathbf{s}}(k) \underline{\mathbf{s}}^H(k)\} \underline{\mathbf{g}} + \underline{\mathbf{w}}^T E\{\underline{\mathbf{n}}(k) \underline{\mathbf{n}}^H(k)\} \underline{\mathbf{w}}^* \\
 &= \underline{\mathbf{g}}^H \underline{\mathbf{R}}_s \underline{\mathbf{g}} + \underline{\mathbf{w}}^T \underline{\mathbf{R}}_n \underline{\mathbf{w}}^*
 \end{aligned} \tag{3.21}$$

根據 (2.22) 式以及 $n(k)$ 為白色雜訊的假設， $\underline{\mathbf{s}}(k)$ 與 $\underline{\mathbf{n}}(k)$ 的自相關矩陣 (autocorrelation matrix) 可以表示如下：

$$\underline{\mathbf{R}}_s = \sigma_s^2 \underline{\mathbf{I}} \quad \underline{\mathbf{R}}_n = \sigma_n^2 \underline{\mathbf{I}} \tag{3.22}$$

因此 (3.21) 式可以化簡為：

$$E\{z(k)z^*(k)\} = \sigma_s^2 \|\underline{\mathbf{g}}\|^2 + \sigma_n^2 \|\underline{\mathbf{w}}\|^2 \tag{3.23}$$

(3.23) 式表示在 $z(k)$ 的功率之中， $\sigma_s^2 \|\underline{\mathbf{g}}\|^2$ 是信號加上符碼間干擾的所造成的貢獻，而 $\sigma_n^2 \|\underline{\mathbf{w}}\|^2$ 則是表示雜訊能量造成的貢獻。乍看之下 (3.23) 式前後兩項都與 $\underline{\mathbf{w}}$ 有關而無法獨立地調整其中一項來達到抑制雜訊的目的，但是如果我們在演算法裡加上 $\|\underline{\mathbf{w}}\|^2 = 1$ 的限制，則 $\sigma_n^2 \|\underline{\mathbf{w}}\|^2$ 這一項就會變成一個常數。在這個限制之下如果調整 $\underline{\mathbf{w}}$ 使得 $E\{|z|^2(k)\}$ 變大就表示 $\|\underline{\mathbf{g}}\|^2$ 會相對應的變大，由於 $\sigma_n^2 \|\underline{\mathbf{w}}\|^2$ 仍為常數，所以雜訊所造成的影響就會相對地變小，因此達到抑制雜訊的目的。

雖然在 $\underline{\mathbf{v}}(k)$ 為單模 (unit norm) 的限制之下放大 $E\{|z|^2(k)\}$ 可以抑制雜訊，但是我們知道：

$$\begin{aligned}
 E\{z(k)z^*(k)\} &= E\{(z_R(k) + jz_I(k))(z_R(k) - jz_I(k))\} \\
 &= E\{z_R^2(k)\} + E\{z_I^2(k)\}
 \end{aligned} \tag{3.24}$$

由於 $E\{z_I^2(k)\}$ 表示符碼間干擾所造成的貢獻，所以應該只放大 $E\{z_R^2(k)\}$ 的部分以避免放大符碼間干擾。

3.4.2 改良的目標函數

最陡坡降法的缺點在於其目標函數僅最小化符碼間干擾，而無法提高信號增益 $g_R(m_0)$ 以及抑制雜訊，為了彌補這個缺點有必要改良其目標函數以提供改良的演算法使用。

從上一節的分析我們知道放大 $z_R^2(k)$ 有助於抑制雜訊，而 $z_R(k)$ 可以表示如下：

$$\begin{aligned} z_R(k) &= c_R(k - m_0)y_R(k) + c_I(k - m_0)y_I(k) \\ &= \mathbf{u}_R^T(k)\mathbf{v} \end{aligned} \quad (3.25)$$

其中

$$\begin{aligned} \mathbf{u}_R(k) &= [c_I(k - m_0)\mathbf{x}_I^T(k) + c_R(k - m_0)\mathbf{x}_R^T(k) \\ &\quad c_I(k - m_0)\mathbf{x}_R^T(k) - c_R(k - m_0)\mathbf{x}_I^T(k)]^T \in \mathcal{C}^{2LQ} \end{aligned} \quad (3.26)$$

而 \mathbf{v} 與之前的定義 (3.6) 式相同。

由於 $z_R(k) = \mathbf{u}_R^T(k)\mathbf{v}$ 、 $z_I(k) = \mathbf{u}_I^T(k)\mathbf{v}$ ，兩者都與 \mathbf{v} 有關，所以可以輕易地將 $z_R(k)$ 併入原有最小化 $z_I^2(k)$ 目標函數之中。我們可以利用最小化 $-\|\mathbf{u}_R^T(k)\mathbf{v}\|^2$ 來達到最大化 $\|\mathbf{u}_R^T(k)\mathbf{v}\|^2$ 的目的，也就是：

$$\text{Minimize: } -\|\mathbf{u}_R^T(k)\mathbf{v}\|^2 = -\mathbf{v}^T\mathbf{u}_R(k)\mathbf{u}_R^T(k)\mathbf{v} \quad (3.27)$$

接著利用一個大於 0 的加權因子 (weighting factor) γ 把 (3.27) 式合併到原有的目標函

數 (3.12) 式之中，得到新的目標函數 J_{new} ：

$$\begin{aligned} J_{\text{new}} &= \mathbf{v}^T \underline{\mathbf{R}}_{\mathbf{u}_I} \mathbf{v} - \gamma \mathbf{v}^T \mathbf{u}_R(k) \mathbf{u}_R^T(k) \mathbf{v} \\ &= \mathbf{v}^T \underline{\mathbf{R}}_{\mathbf{u}_I} \mathbf{v} - \gamma z_R^2(k) \end{aligned} \quad (3.28)$$

現在我們可以藉由最小化新的目標函數 J_{new} 同時地最小化 $z_I^2(k)$ 以及最大化 $z_R^2(k)$ 。值得注意的是，最大化 $z_R^2(k)$ 雖然是在抑制雜訊的目的下得出的手段，但是隨著最小化 $z_I^2(k)$ 的過程漸漸地消除符碼間干擾，會使得 $z(k) \simeq z_R(k) \simeq g_R(m_0)b(k - m_0)$ ，在這個情況下最大化 $z_R^2(k)$ 不但可以抑制雜訊還可以同時提高信號的增益 $g_R(m_0)$ 。而要在抑制雜訊、消除符碼間干擾以及提高信號增益之間取得平衡，則需要仰賴合適的加權因子 γ 。



3.4.3 加權因子 γ

到目前為止我們知道要消除符碼間干擾可以利用最小化 $z_I^2(k)$ ，要抑制雜訊及提高信號增益可以利用最大化 $z_R^2(k)$ ，而如何適當地選擇加權因子 γ 來取得之間的平衡將會在這一小節說明。

首先觀察目標函數 J_{new} 中的 $\underline{\mathbf{R}}_{\mathbf{u}_I}$ ，根據 Rayleigh 不等式我們知道對於任何單模的 \mathbf{v} ，二次式 (quadratic form) $\mathbf{v}^T \underline{\mathbf{R}}_{\mathbf{u}_I} \mathbf{v}$ 必定滿足以下關係：

$$\lambda_{\max} \geq \mathbf{v}^T \underline{\mathbf{R}}_{\mathbf{u}_I} \mathbf{v} \geq \lambda_{\min} \quad (3.29)$$

其中 λ_{\max} 與 λ_{\min} 分別為 $\underline{\mathbf{R}}_{\mathbf{u}_I}$ 的最大與最小特徵值 (eigenvalue)，而此時對應到最小特徵值的特徵向量 (eigenvector) 即為最小化 $\mathbf{v}^T \underline{\mathbf{R}}_{\mathbf{u}_I} \mathbf{v}$ 的解；換言之，尋找最小化 $\mathbf{v}^T \underline{\mathbf{R}}_{\mathbf{u}_I} \mathbf{v}$ 的解就是在尋找對應到最小特徵值的特徵向量。

根據 (3.29) 式，新的目標函數 J_{new} 的極值範圍可以表示如下：

$$\lambda_{\max} - \gamma z_R^2(k) \geq J_{\text{new}} \geq \lambda_{\min} - \gamma z_R^2(k) \quad (3.30)$$

由上式可以發現，因為加入了 $-\gamma z_R^2(k)$ 這一項，使得最小化 J_{new} 的解不一定能夠最小化 $\mathbf{v}^T \mathbf{R}_{\mathbf{u}_I} \mathbf{v}$ ，這使得抑制了雜訊以及提高信號增益的同時，符碼間干擾卻有可能增加，但是如果我們選擇：

$$\gamma = \frac{\lambda_{\min}}{z_R^2(k)} \quad (3.31)$$

將 (3.31) 式代入新的目標函數 (3.28) 式後，現在最小化 J_{new} 的問題會變成：

$$\text{Minimize: } J_{\text{new}} = \mathbf{v}^T \mathbf{R}_{\mathbf{u}_I} \mathbf{v} - \frac{\lambda_{\min}}{z_R^2(k)} z_R^2(k) = \mathbf{v}^T \mathbf{R}_{\mathbf{u}_I} \mathbf{v} - \lambda_{\min} \quad (3.32)$$

由上式可以得知，最小化 $\mathbf{v}^T \mathbf{R}_{\mathbf{u}_I} \mathbf{v}$ 與 λ_{\min} 之間差距的解，即為最小化 $\mathbf{v}^T \mathbf{R}_{\mathbf{u}_I} \mathbf{v}$ 的解，如此一來就可以在不增加多餘符碼間干擾的情況下來最大化 $z_R^2(k)$ 。值得注意的是此時 J_{\min} 的極小值恰等於 0，這個關係之後會用於演算法的穩定性分析上。

由於 $z_R^2(k)$ 是一個會變動的值，所以實際上我們選擇：

$$\gamma = \frac{\lambda_{\min}}{E\{z_R^2(k)\}} \quad (3.33)$$

而 $\mathbf{R}_{\mathbf{u}_I}$ 以及 $z_R(k)$ 都是接收端輸出的資料，所以可以估計出 $\mathbf{R}_{\mathbf{u}_I}$ 的最小特徵值 λ_{\min} 以及 $E\{z_R^2(k)\}$ 來決定 γ 的值。

以上的分析是建立在穩態的基礎上 (J_{new} 收斂到極小值時分析 γ 最好的值)，不過在通道等化剛開始的時候符碼間干擾 ($|z_I(k)|^2$) 往往很大，所以要快速提升 SINR 可以在演算法剛開始時使用較大的 γ 值先確實地提升 $g_R(m_0)$ ，而隨著符碼間干擾漸漸消除

掉再讓 γ 值漸漸地趨近 $\lambda_{\min}/E\{z_R^2(k)\}$ ，如此一來 SINR 會上升的比較快。為了因應這樣的需求，我們可以用下列指數加權的方式來更新 γ 值：

$$\gamma^k = \alpha\gamma^{k-1} + (1 - \alpha)\frac{\lambda_{\min}}{E\{z_R^2(k)\}} \quad (3.34)$$

其中 $0 < \alpha < 1$ 為遺忘因子，用來決定 γ 從較大的初始值 γ^0 趨近於終值 $\lambda_{\min}/E\{z_R^2(k)\}$ 的速度。

3.4.4 演算法流程

現在我們已經有改良的目標函數 J_{new} 以及適當的加權因子 γ ，接下來要將這些合併起來得到我們提出的適應性演算法。

首先將 J_{new} 對 \mathbf{v} 作偏微分得到梯度函數：

$$\begin{aligned} \frac{\partial J_{\text{new}}(\mathbf{v}(k))}{\partial \mathbf{v}(k)} &= \frac{\partial}{\partial \mathbf{v}(k)} (\mathbf{v}^T \mathbf{R}_{\mathbf{u}_I} \mathbf{v} - \gamma \mathbf{v}^T \mathbf{u}_R(k) \mathbf{u}_R^T(k) \mathbf{v}) \\ &= 2\mathbf{R}_{\mathbf{u}_I} \mathbf{v} - 2\gamma \mathbf{u}_R(k) \mathbf{u}_R^T(k) \mathbf{v} \end{aligned} \quad (3.35)$$

把 (3.35) 式帶入最陡坡降法的疊代公式：

$$\begin{aligned} \mathbf{v}(k+1) &= \mathbf{v}(k) - \frac{1}{2}\mu \frac{\partial J(\mathbf{v}(k))}{\partial \mathbf{v}(k)} \\ &= \mathbf{v}(k) - \mu [\mathbf{R}_{\mathbf{u}_I} - \gamma^k \mathbf{u}_R(k) \mathbf{u}_R^T(k)] \mathbf{v}(k) \end{aligned} \quad (3.36)$$

(3.36) 式即為我們提出的改良演算法，演算法流程如下：

演算法流程：

step 1

$$\mathbf{u}_I(k) = [c_R(k - m_0) \mathbf{x}_I^T(k) - c_I(k - m_0) \mathbf{x}_R^T(k) \quad c_I(k - m_0) \mathbf{x}_I^T(k) + c_R(k - m_0) \mathbf{x}_R^T(k)]^T$$

step 2

$$\underline{\mathbf{u}}_R(k) = [c_I(k - m_0)\underline{\mathbf{x}}_I^\top(k) + c_R(k - m_0)\underline{\mathbf{x}}_R^\top(k) \quad c_I(k - m_0)\underline{\mathbf{x}}_R^\top(k) - c_R(k - m_0)\underline{\mathbf{x}}_I^\top(k)]^\top$$

step 3

$$\underline{\mathbf{R}}_{\mathbf{u}_I}^k = \beta \underline{\mathbf{R}}_{\mathbf{u}_I}^{k-1} + (1 - \beta)\underline{\mathbf{u}}_I(k)\underline{\mathbf{u}}_I^\top(k)$$

step 4

$$\gamma^k = \alpha\gamma^{k-1} + (1 - \alpha)\lambda_{\min}/E\{z_R^2(k)\}$$

step 5

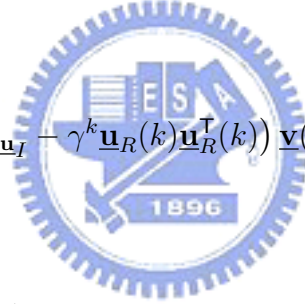
$$\underline{\mathbf{v}}(k) = \underline{\mathbf{v}}(k)/\|\underline{\mathbf{v}}(k)\|$$

step 6

$$\underline{\mathbf{v}}(k + 1) = \underline{\mathbf{v}}(k) - \mu (\underline{\mathbf{R}}_{\mathbf{u}_I} - \gamma^k \underline{\mathbf{u}}_R(k)\underline{\mathbf{u}}_R^\top(k)) \underline{\mathbf{v}}(k)$$

step 7

$$k = k + 1 \quad \text{goto step 1}$$



3.4.5 穩定性分析

在這一節的最後要驗證我們改良的演算法的穩定性，以保證 $\underline{\mathbf{v}}(k)$ 確實會收斂到最小化目標函數 J_{new} 的解。

首先考慮以下的疊代公式：

$$\underline{\mathbf{v}}(k + 1) = \underline{\mathbf{v}}(k) - \mu \underline{\mathbf{R}}_{\mathbf{u}_I} \underline{\mathbf{v}}(k) \tag{3.37}$$

對上式等號兩邊都減去 $\underline{\mathbf{v}}(k)$ 的最佳化解 $\underline{\mathbf{v}}_o$:

$$\underline{\mathbf{v}}(k+1) - \underline{\mathbf{v}}_o = \underline{\mathbf{v}}(k) - \underline{\mathbf{v}}_o - \mu \underline{\mathbf{R}}_{\underline{\mathbf{u}}_I} \underline{\mathbf{v}}(k) + (\mu \underline{\mathbf{R}}_{\underline{\mathbf{u}}_I} \underline{\mathbf{v}}_o - \mu \underline{\mathbf{R}}_{\underline{\mathbf{u}}_I} \underline{\mathbf{v}}_o) \quad (3.38)$$

接著定義 $\underline{\mathbf{v}}(k)$ 與最佳解 $\underline{\mathbf{v}}_o$ 之間的誤差 $\underline{\mathbf{e}}(k) = \underline{\mathbf{v}}(k) - \underline{\mathbf{v}}_o$, 上式可以寫為 :

$$\begin{aligned} \underline{\mathbf{e}}(k+1) &= \underline{\mathbf{e}}(k) - \mu \underline{\mathbf{R}}_{\underline{\mathbf{u}}_I} \underline{\mathbf{e}}(k) - \mu \underline{\mathbf{R}}_{\underline{\mathbf{u}}_I} \underline{\mathbf{v}}_o \\ &= (\underline{\mathbf{I}} - \mu \underline{\mathbf{R}}_{\underline{\mathbf{u}}_I}) \underline{\mathbf{e}}(k) - \mu \underline{\mathbf{R}}_{\underline{\mathbf{u}}_I} \underline{\mathbf{v}}_o \end{aligned} \quad (3.39)$$

因為 $\underline{\mathbf{v}}_o$ 為 $\underline{\mathbf{R}}_{\underline{\mathbf{u}}_I}$ 的零空間向量所以 $\underline{\mathbf{R}}_{\underline{\mathbf{u}}_I} \underline{\mathbf{v}}_o = \underline{\mathbf{0}}$, 因此上式可以簡化成 :

$$\underline{\mathbf{e}}(k+1) = (\underline{\mathbf{I}} - \mu \underline{\mathbf{R}}_{\underline{\mathbf{u}}_I}) \underline{\mathbf{e}}(k) \quad (3.40)$$

如果 (3.40) 式要收斂, $\underline{\mathbf{e}}(k)$ 就要隨著 k 變大而趨近於 0 , 根據 [7] , (3.40) 式要收斂的條件是 :

$$0 < \mu < \frac{2}{\lambda_{\max}} \quad (3.41)$$

其中 λ_{\max} 為 $\underline{\mathbf{R}}_{\underline{\mathbf{u}}_I}$ 的最大特徵值。

接下來回頭考慮我們提出的演算法, 同樣地對 (3.36) 式等號兩邊減去最小化 J_{new} 的最佳解 $\hat{\underline{\mathbf{v}}}_o$:

$$\underline{\mathbf{v}}(k+1) - \hat{\underline{\mathbf{v}}}_o = \underline{\mathbf{v}}(k) - \hat{\underline{\mathbf{v}}}_o - \mu (\underline{\mathbf{R}}_{\underline{\mathbf{u}}_I} - \gamma^k \underline{\mathbf{u}}_R(k) \underline{\mathbf{u}}_R^T(k)) \underline{\mathbf{v}}(k) \quad (3.42)$$

同樣定義 $\hat{\underline{\mathbf{e}}}_o(k) = \underline{\mathbf{v}}(k) - \hat{\underline{\mathbf{v}}}_o$ 並且代入上式 :

$$\begin{aligned} \hat{\underline{\mathbf{e}}}_o(k+1) &= \hat{\underline{\mathbf{e}}}_o(k) - \mu (\underline{\mathbf{R}}_{\underline{\mathbf{u}}_I} - \gamma^k \underline{\mathbf{u}}_R(k) \underline{\mathbf{u}}_R^T(k)) \hat{\underline{\mathbf{e}}}_o(k) \\ &\quad - \mu (\underline{\mathbf{R}}_{\underline{\mathbf{u}}_I} - \gamma^k \underline{\mathbf{u}}_R(k) \underline{\mathbf{u}}_R^T(k)) \hat{\underline{\mathbf{v}}}_o \end{aligned} \quad (3.43)$$

因為 $\hat{\mathbf{v}}_o$ 為最小化 J_{new} 的解，在 $\gamma = \lambda_{\min}/E\{z_R^2(k)\}$ 的情況下 J_{\min} 的極小值恰為 0，所以 $(\mathbf{R}_{\mathbf{u}_I} - \gamma \mathbf{u}_R(k) \mathbf{u}_R^T(k)) \hat{\mathbf{v}}_o = \mathbf{0}$ ，並且根據 (3.15) 式當 $k \rightarrow \infty$ 時 $\gamma^k \rightarrow \gamma$ ，因此 (3.43) 式可以化簡如下：

$$\hat{\mathbf{e}}(k+1) = [\mathbf{I} - \mu (\mathbf{R}_{\mathbf{u}_I} - \gamma \mathbf{u}_R(k) \mathbf{u}_R^T(k))] \hat{\mathbf{e}}(k) \quad (3.44)$$

比較 (3.40) 與 (3.44) 式可以發現，(3.44) 式只多了 $\gamma \mathbf{u}_R(k) \mathbf{u}_R^T(k)$ 這一項，如果可以證明 $\mathbf{R}_{\mathbf{u}_I}$ 的最大特徵值 λ_{\max} 一定大於等於 $\mathbf{R}_{\mathbf{u}_I} - \gamma \mathbf{u}_R(k) \mathbf{u}_R^T(k)$ 的最大特徵值 $\hat{\lambda}_{\max}$ ，那麼只要選擇 μ 滿足 (3.41) 式，就一定可以使得 (3.44) 式收斂，因為：

$$\mu < \frac{2}{\lambda_{\max}} \leq \frac{2}{\hat{\lambda}_{\max}} \quad (3.45)$$

所以 μ 也會滿足使 (3.44) 收斂的條件—— $\mu < 2/\hat{\lambda}_{\max}$ 。

定理 2.

對於任何對稱矩陣 \mathbf{A} 以及向量 \mathbf{b} ，只要常數 $a > 0$ 則一定保證 \mathbf{A} 的最大特徵值 λ_{\max} 一定大於等於 $\mathbf{A} - a \mathbf{b} \mathbf{b}^T$ 的最大特徵值 $\hat{\lambda}_{\max}$ 。

證明：

根據 Rayleigh 不等式我們知道：

$$\lambda_{\max} \geq \underline{\mathbf{p}}^T \mathbf{A} \underline{\mathbf{p}} \quad (3.46)$$

等號發生的條件是 $\underline{\mathbf{p}}$ 恰為對應到 λ_{\max} 的特徵向量 $\underline{\mathbf{p}}_{\max}$ 。同理，當 $\underline{\mathbf{q}}$ 恰為對應 $\mathbf{A} - a \mathbf{b} \mathbf{b}^T$ 最大特徵值 $\hat{\lambda}_{\max}$ 的特徵向量 $\underline{\mathbf{q}}_{\max}$ 時：

$$\begin{aligned} \hat{\lambda}_{\max} &= \underline{\mathbf{q}}_{\max}^T (\mathbf{A} - a \mathbf{b} \mathbf{b}^T) \underline{\mathbf{q}}_{\max} \\ &= \underline{\mathbf{q}}_{\max}^T \mathbf{A} \underline{\mathbf{q}}_{\max} - a |\mathbf{b}^T \underline{\mathbf{q}}_{\max}|^2 \end{aligned} \quad (3.47)$$

因為 $\underline{\mathbf{q}}_{\max}$ 不一定等於 $\underline{\mathbf{p}}_{\max}$ ，所以 $\underline{\mathbf{q}}_{\max}^T \mathbf{A} \underline{\mathbf{q}}_{\max} \leq \underline{\mathbf{p}}_{\max}^T \mathbf{A} \underline{\mathbf{p}}_{\max} = \lambda_{\max}$ ，將之代入 (3.47) 式可以得到：

$$\begin{aligned}\hat{\lambda}_{\max} &\leq \lambda_{\max} - a|\underline{\mathbf{b}}^T \underline{\mathbf{q}}_{\max}|^2 \\ \Leftrightarrow \lambda_{\max} &\geq \hat{\lambda}_{\max} + a|\underline{\mathbf{b}}^T \underline{\mathbf{q}}_{\max}|^2\end{aligned}\quad (3.48)$$

因為 $a|\underline{\mathbf{b}}^T \underline{\mathbf{q}}_{\max}|^2 \geq 0$ ，故定理 2. 得證。

從定理 2. 可以知道，由於 γ^k 恆為正值所以只要選擇 $\mu < 2/\lambda_{\max}$ 就可以保證我們的改良演算法也會收斂。



第 4 章 模擬結果

這一章將以實際模擬的結果來比較現有的演算法和我們提出的演算法的性能。

4.1 多路徑通道參數

我們使用一個具有三條路徑的通道 $h(t)$ 來進行本章的模擬實驗， $h(t)$ 給定如下：

$$h(t) = [a_1 e^{j2\pi f_c \tau_1} p(t - \tau_1, b) + a_2 e^{j2\pi f_c \tau_2} p(t - \tau_2, b) + a_3 e^{j2\pi f_c \tau_3} p(t - \tau_3, b)] W_{8T} \quad (4.1)$$

其中 $p(t, b)$ 為升餘弦函數 (raised cosine function)，其 roll-off 因子 $b = 0.11$ ，載波頻率 $f_c = 900$ MHz， W_{8T} 為一個觀察 8 個符碼週期的窗型函數而符碼週期 $T = 1\mu\text{s}$ ， a_i 與 τ_i 分別為第 i 條路徑的路徑增益和路徑延遲，數值如表 4.1：

	$i = 1$	$i = 2$	$i = 3$
a_i	0.2	0.4	0.3
τ_i	3.0000	3.7271	4.0122

表 4.1: 多路徑通道參數

$h(t)$ 的實部與虛部響應如下圖：

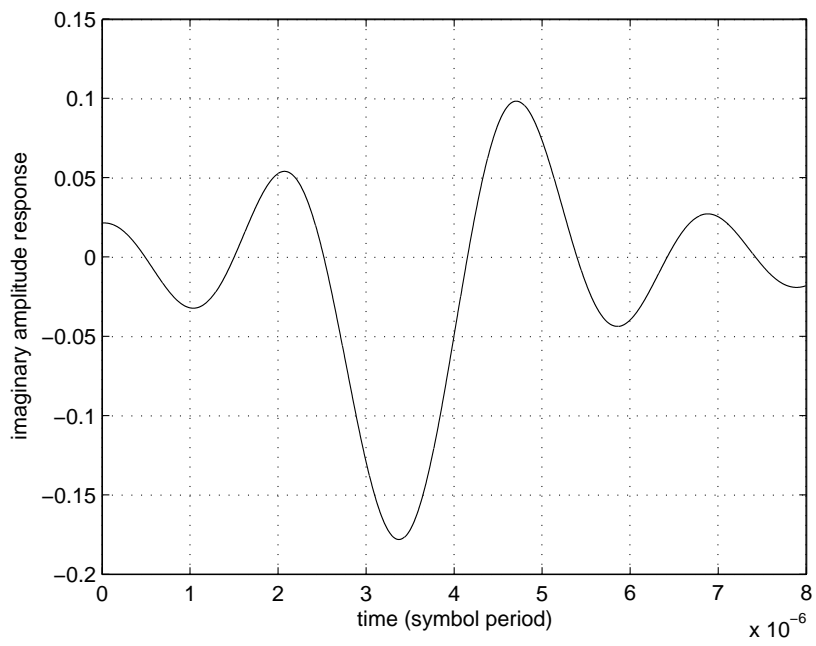
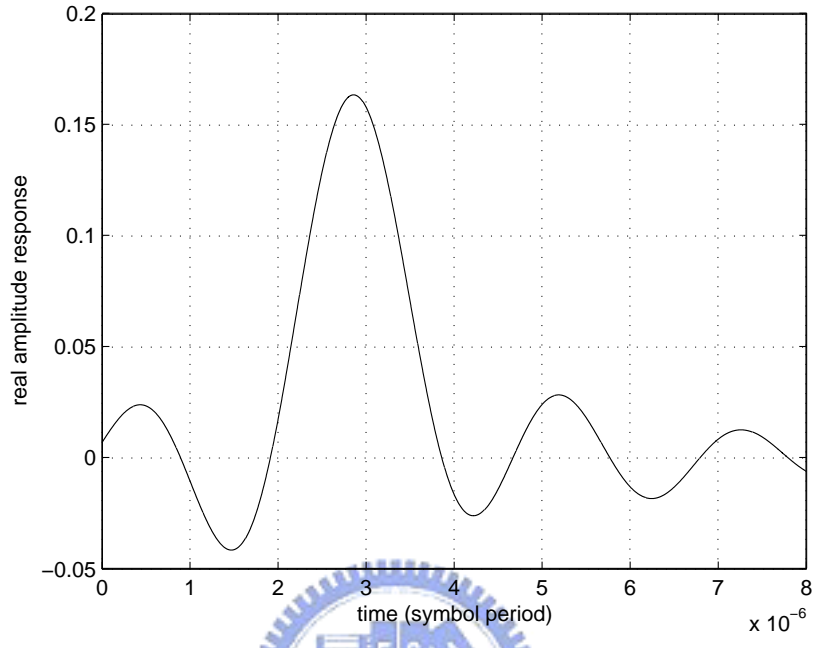


圖 4.1: 多路徑通道響應

4.2 單輸出及多輸出通道的性能比較

這一節我們比較在單輸出通道 ($L = 1$) 以及多輸出通道 ($L \geq 2$) 下的各種演算法性能，來驗證我們改良的演算法確實比最陡坡降法更能在多輸出通道有效地提高 SINR。SINR 的估計方法將使用下式：

$$\text{SINR} = \frac{1}{R} \sum_{r=1}^R \frac{\sigma_b^2 \max\{|g_r(q)|^2\}}{\sigma_b^2 \left[\sum_{q=0}^{M+Q-2} |g_r(q)|^2 - \max\{|g_r(q)|^2\} \right] + \sigma_n^2 \sum_{i=0}^{L-1} \sum_{q=0}^{Q-1} |w_r^i(q)|^2} \quad (4.2)$$

其中 R 為蒙地卡羅數， g_r 、 w_r 分別是第 r 次蒙地卡羅的整體響應和等化器係數。

4.2.1 單輸入單輸出通道的性能比較

在不使用過取樣的情況下 ($L = 1$)，以符碼週期 ($1\mu\text{s}$) 為取樣週期來取樣 $h(t)$ 可以得到離散時間的單輸出通道響應，響應係數如表 4.2。

$h(0)$	$0.007 - 0.021j$
$h(1)$	$-0.010 + 0.031j$
$h(2)$	$0.017 - 0.052j$
$h(3)$	$0.158 + 0.128j$
$h(4)$	$-0.016 + 0.049j$
$h(5)$	$0.024 - 0.074j$
$h(6)$	$-0.012 + 0.039j$
$h(7)$	$0.008 - 0.025j$

表 4.2: 離散時間單輸出通道響應

在這個情況下子通道輸出的數目 $L = 1$ ，通道長度 $M = 8$ ，我們選擇等化器長度 $Q = 7$ 、啣聲信號的設計參數 $1/f = T_f = 17 > M + Q - 2$ 以滿足定理 1. 的要求，而傳送的資訊符碼 $b(k)$ 為二元的序列 (± 1)。

我們提出的演算法使用遺忘因子 $\alpha = 0.95$ 、 $\beta = 0.98$ 、 $\mu = 2$ 以及加權因子 $\gamma = \lambda_{\min}/E\{z_R^2(k)\} = 0.05$ ，加權因子的初始值 $\gamma^0 = 1$ ；最陡坡降法使用 $\beta = 0.98$ ， $\mu = 2$ ；而批次演算法堆疊的資料數目 $N = 200$ 。以上演算法的等化延遲 m_0 都為 7 ($M + Q - 2$ 的中間值)，換言之我們將使得等化並且解調完的輸出 $z(k) = g_R(m_0)b(k - m_0)$ 。分別針對三個演算法在通道輸出 SNR = 20dB 的情況進行 100 次蒙地卡羅 ($R = 100$)，演算法的性能比較如圖 4.2。

4.2.2 單輸入多輸出通道的性能比較

接著以過取樣 4 倍的方式為例來產生一個等效的多輸出通道，使用 $1/4$ 符碼週期 ($0.25\mu s$) 為取樣週期過取樣 $h(t)$ 可以得到 4 個等效子通道，等效子通道的響應係數如表 4.3。

此時子輸出的數目 $L = 4$ ，通道的長度 $M = 8$ ，改良的演算法使用遺忘因子 $\alpha = 0.95$ 、 $\beta = 0.98$ 、 $\mu = 2$ 以及加權因子 $\gamma = \lambda_{\min}/E\{z_R^2(k)\} = 0.002$ ，加權因子的初始值 $\gamma^0 = 1$ ；最陡坡降法使用 $\beta = 0.98$ ， $\mu = 2$ ；而批次演算法堆疊的資料數目 $N = 200$ 。以上演算法的等化延遲 m_0 都為 7，針對三個演算法在通道輸出 SNR = 20dB 的情況進行 100 次蒙地卡羅，演算法性能比較如圖 4.3。

	$i = 0$	$i = 1$	$i = 2$	$i = 3$
$h^i(0)$	$0.007 - 0.021j$	$0.020 - 0.016j$	$0.023 + 0.000j$	$0.012 + 0.020j$
$h^i(1)$	$-0.010 + 0.031j$	$-0.030 + 0.025j$	$-0.041 + 0.000j$	$-0.025 - 0.032j$
$h^i(2)$	$0.017 - 0.052j$	$0.074 - 0.045j$	$0.128 - 0.005j$	$0.160 + 0.060j$
$h^i(3)$	$0.158 + 0.128j$	$0.124 + 0.172j$	$0.071 + 0.172j$	$0.018 + 0.126j$
$h^i(4)$	$-0.016 + 0.049j$	$-0.025 - 0.029j$	$-0.014 - 0.083j$	$0.007 - 0.098j$
$h^i(5)$	$0.024 - 0.074j$	$0.028 - 0.028j$	$0.018 + 0.016j$	$0.001 + 0.041j$
$h^i(6)$	$-0.012 + 0.039j$	$-0.018 + 0.018j$	$-0.013 - 0.007j$	$-0.002 - 0.024j$
$h^i(7)$	$0.008 - 0.025j$	$0.012 - 0.012j$	$0.009 + 0.004j$	$0.001 + 0.017j$

表 4.3: 離散時間多輸出通道響應

4.2.3 模擬結果比較

由圖 4.2、圖 4.3 可以知道，最陡坡降法雖然可以適用於單輸出 ($L = 1$) 的通道，但是用於多輸出通道卻完全無法地提升 SINR。藉由修改目標函數以及選擇最佳的加權因子 γ 的幫助，改良的演算法在單輸出的情況下，SINR 的上升速度以及穩態的 SINR 值都遠比原來的最陡坡降法來得好；而在多輸出通道的情況下亦可以有效的提升 SINR，甚至在穩態時改良演算法的 SINR 還可以略高於批次演算法。

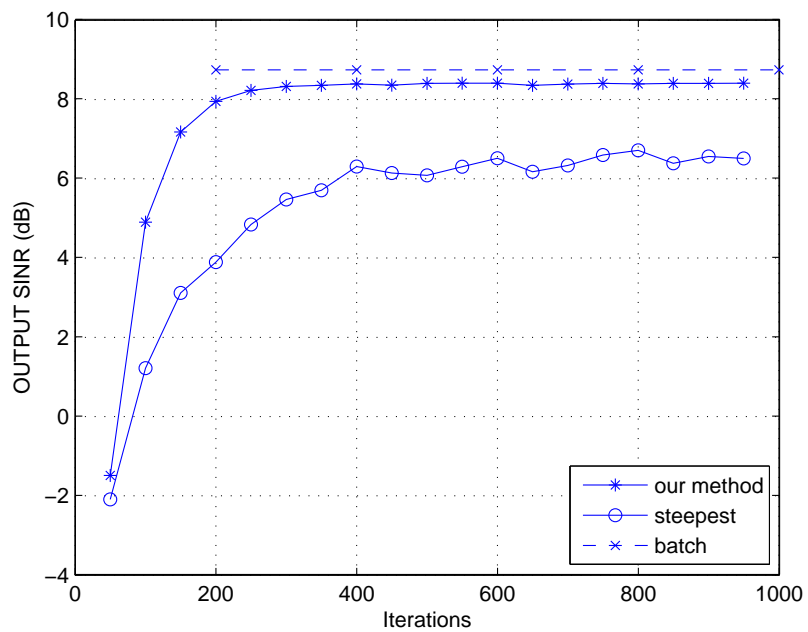


圖 4.2: 單輸入單輸出通道 SNR=20dB 的性能比較

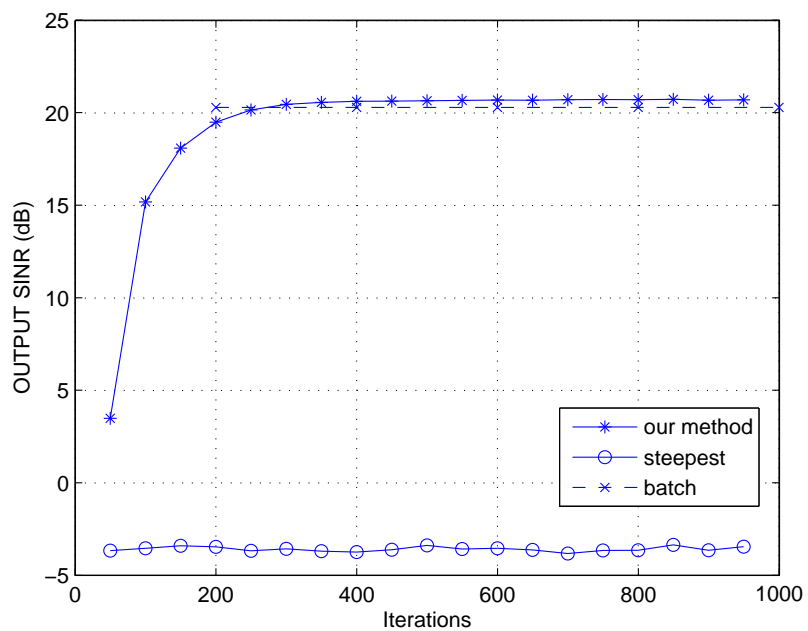


圖 4.3: 單輸入多輸出通道 SNR=20dB 的性能比較

4.3 不同 SNR 的性能比較

這一節要比較在多輸出通道 ($L = 4$) 的情況下，通道輸出信號雜訊比 SNR 對於等化器輸出 SINR 的影響。為了考慮較具一般性的多輸出通道的情況，無法在多輸出通道獲得良好的性能最陡坡降法將不列入比較的範圍內。

首先我們使用多輸出通道的響應係數 (表 4.3) 以及 4.2.2 節的演算法參數進行 100 次蒙地卡羅實驗，來比較不同 SNR 時我們改良的演算法的收斂速度以及穩態的 SINR，實驗結果如圖 4.4。由圖 4.4 可以發現即使在雜訊強度很大的情況下，改良演算法的 SINR 仍然可以很快地提升到穩態，並且穩態的 SINR 可以高過通道輸出端的 SNR (約 1 到 2dB)，這表示改良的演算法在消除符碼干擾的同時也抑制了一部分的雜訊影響，如此才有可能使得等化器輸出端的 SINR 高於通道輸出端的 SNR。

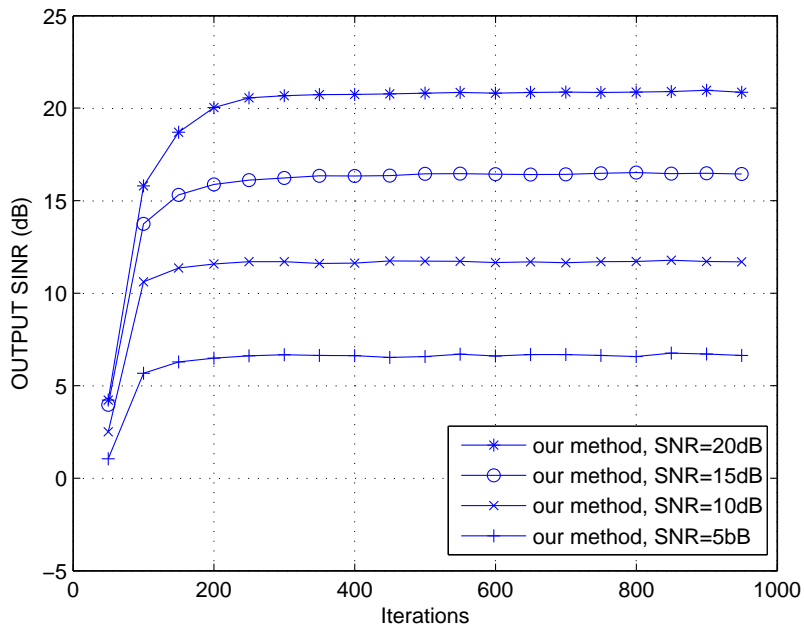


圖 4.4: 不同 SNR 改良的演算法的性能比較

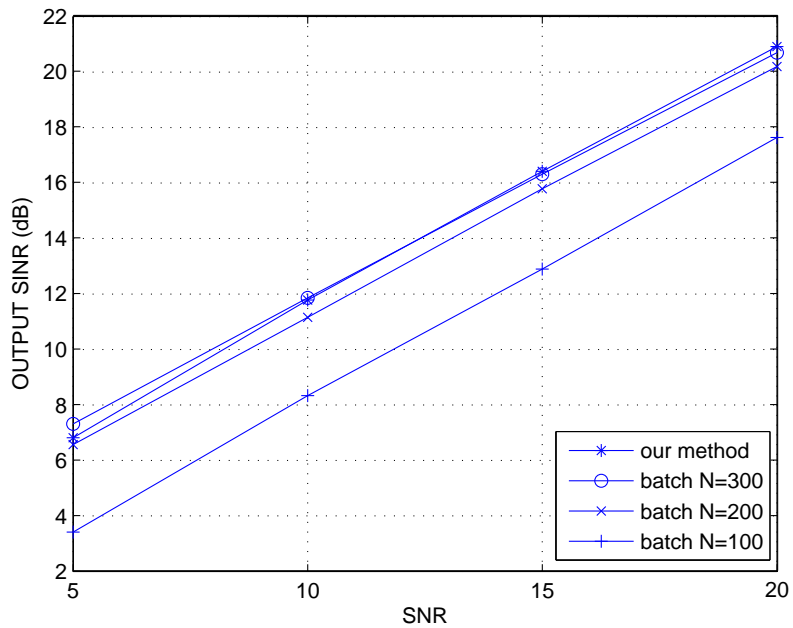


圖 4.5: 不同 SNR 改良的演算法與批次演算法的性能比較

接下來比較在不同 SNR 的情況下，改良的演算法的穩態 SINR 與堆疊不同數目資料的批次演算法 SINR，同樣以表 4.3 的通道係數以及 4.2.2 節的參數進行 100 次蒙地卡羅的實驗結果如圖 4.5。從圖 4.4 可以知道，改良的演算法需要 150 到 250 次的疊代來到達穩態，和圖 4.5 比較後發現堆疊了相同數量 (150 到 250) 資料的批次演算法，其 SINR 尚低於我們改良的演算法，批次演算法需要每次堆疊到 300 筆資料其性能才會接近我們改良的演算法。

4.4 加權因子 γ 值對 SINR 的影響

在之前的分析我們知道穩態時最佳的 γ 值應該等於 $\lambda_{\min}/E\{z_R^2(k)\}$ ，在這一節我們將比較不同的 γ 值對 SINR 所造成的影響。

我們使用表 4.3 的多輸出通道係數以及 4.2.2 來進行實驗，分別在 SNR = 20dB、15dB、10dB、5dB 的情況下選擇 γ 等於 10 倍($\lambda_{\min}/E\{z_R^2(k)\}$)、 $\lambda_{\min}/E\{z_R^2(k)\}$ 以及 0.1 倍($\lambda_{\min}/E\{z_R^2(k)\}$) 來進行 100 次蒙地卡羅實驗，以觀察 γ 值對改良演算法 SINR 的影響。

從圖 4.7、圖 4.8 和圖 4.9 可以看出，當 γ 值為 1/10 的 ($\lambda_{\min}/E\{z_R^2(k)\}$) 時，其穩態的 SINR 會慢慢的降低，並且在 SNR 低的時候 SINR 降低的越快。因為如果選擇過小的 γ 值，其性質會接近於無法有效提升 SINR 的最陡坡降法，所以當 γ^k 慢慢接近終值時演算法就不再能夠抑制雜訊以及提高信號增益，因此 SINR 會漸漸往下掉。

如果 γ 選擇為 10 倍的 ($\lambda_{\min}/E\{z_R^2(k)\}$) 時可以發現其穩態的 SINR 會變低但沒有往下掉趨勢，原因是當選擇大於 $\lambda_{\min}/E\{z_R^2(k)\}$ 太多的 γ 值時，最小化改良的目標函數 J_{new} 的解變得不能最小化穩態的符碼間干擾 (參考 (3.32) 式)，因此符碼間干擾上升則等化器輸出的 SINR 就會隨著下降；而夠大的 γ 值可以提升信號增益並抑制雜訊所以穩態的 SINR 不會往下掉。

至於在圖 4.6 中選擇過小的 γ 值時演算法性能沒有什麼變化的原因是，在 SNR = 20dB 的情況下雜訊的影響已經非常小，所以到達穩態後等化器的係數就幾乎不會因為雜訊的影響而有所變動，因此穩態的 SINR 可以維持一定水平。

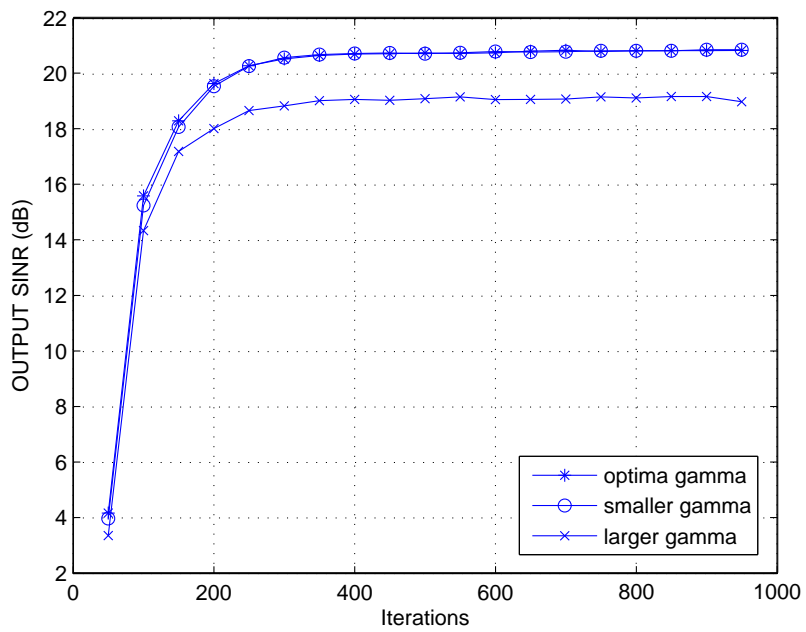


圖 4.6: SNR=20dB 時不同 γ 值的性能比較

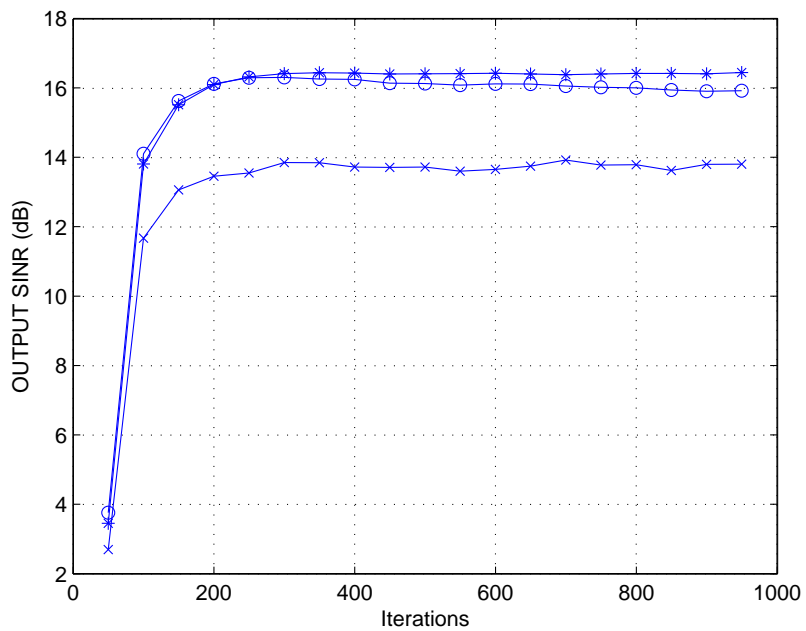


圖 4.7: SNR=15dB 時不同 γ 值的性能比較

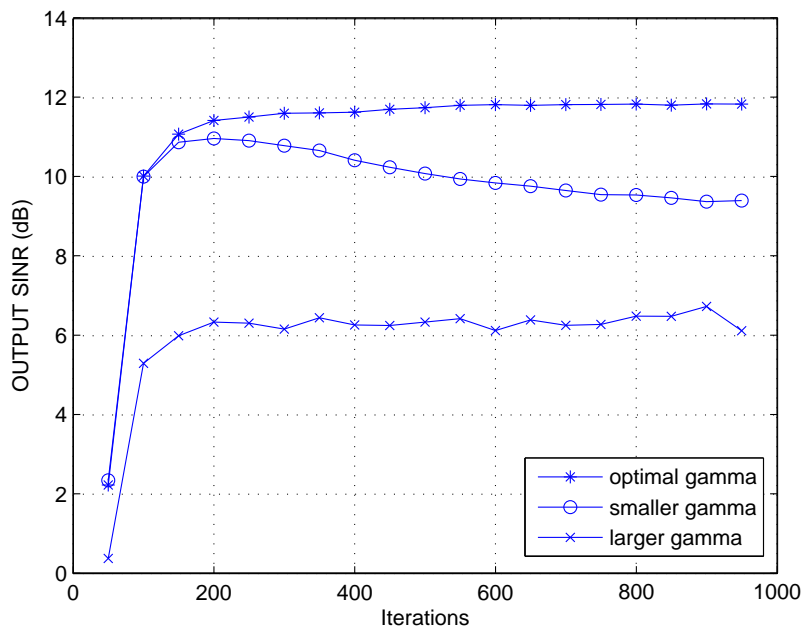


圖 4.8: SNR=10dB 時不同 γ 值的性能比較

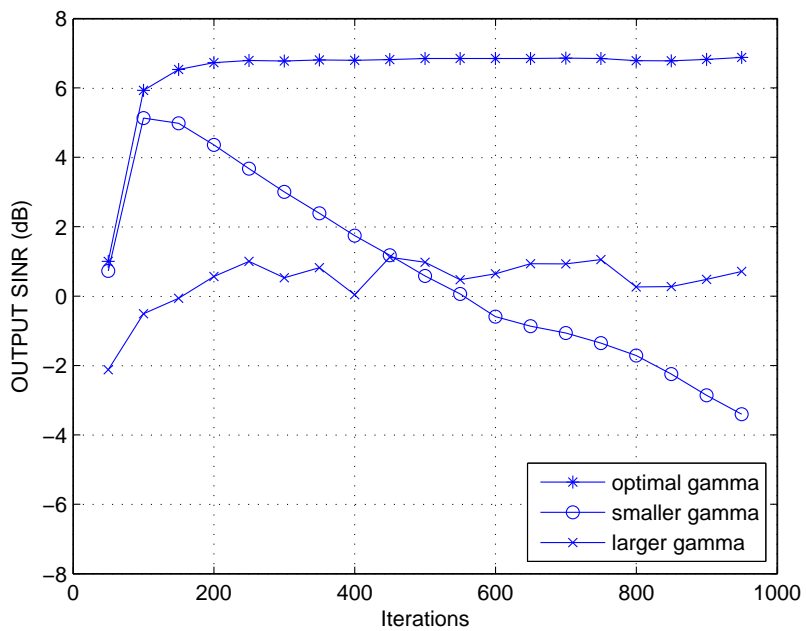


圖 4.9: SNR=5dB 時不同 γ 值的性能比較

4.5 錯誤發生率的比較

本章的最後一節將模擬等化之後決策的錯誤發生率。同樣使用表 4.3 的多輸出通道響應係數以及 4.2.2 節的參數，首先傳送二元序列 (± 1) 的資訊符碼 $b(k)$ 進行 100 次蒙地卡羅實驗，每次則等化 100,000 筆 $b(k)$ ，而決策的方法是採用最小距離 (minimal distance)，模擬得出的位元錯誤發生率如圖 4.10。

在相同的通道係數以及演算法參數下傳送 PAM4 (pulse amplitude modulation) 的資訊符碼 $b(k) \in \{\pm 3/\sqrt{5}, \pm 1/\sqrt{5}\}$ ，比例常數 $1/\sqrt{5}$ 使得 $E\{b^2(k)\}$ 等於 1。PAM4 的實驗同樣也是進行 100 次，每次等化 100,1000 筆 $b(k)$ ，模擬得出的符碼發生率如圖 4.11。

觀察圖 4.10、圖 4.11 可以發現在這兩個實驗之中，我們改良的演算法在 SNR 極低 ($\text{SNR} < 5$) 的情況下其錯誤發生率都略高於批次演算法堆疊數目 $N = 300$ 的錯誤發生率，而在 $\text{SNR} > 5\text{dB}$ 左右改良演算法的錯誤發生率開始低於 $N = 300$ 的批次演算法，類似的現象也出現在圖 4.5 之中，在 $\text{SNR} < 5\text{dB}$ 時改良的演算法的 SINR 略低於堆疊 300 筆資料的批次演算法，之後改良的演算法性能開始超過批次演算法。

造成這個現象可能的原因是我們改良的演算法是一個不斷疊代的適應性演算法，當等化器係數受到雜訊的干擾而產生較大變動時會需要幾次的疊代才會回到原來的穩態，在回到穩態的過程期間等化的輸出都會因此而受到影響，如果在雜訊較強的情況下有可能在回到穩態的過程中產生數個錯誤，造成錯誤發生率上升；反之批次演算法是堆疊數筆資料後再一次求解等化器係數，得出來的等化器係數對於堆疊的 N 資料平均上來說是最好的，也可以視為等化器把雜訊的影響平均地分佈到整個堆疊的期間，因此不較不會有錯誤連續發生的機會。

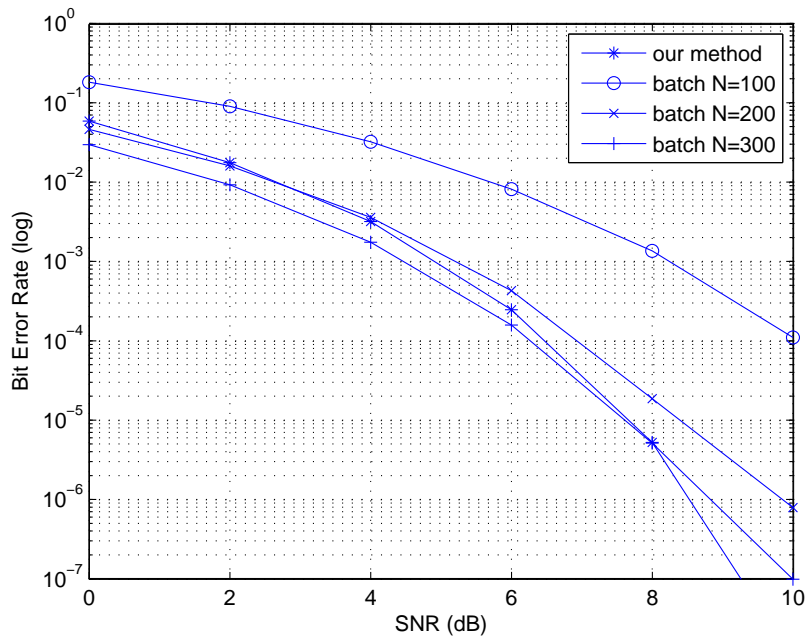


圖 4.10: BPSK 的位元錯誤發生率比較

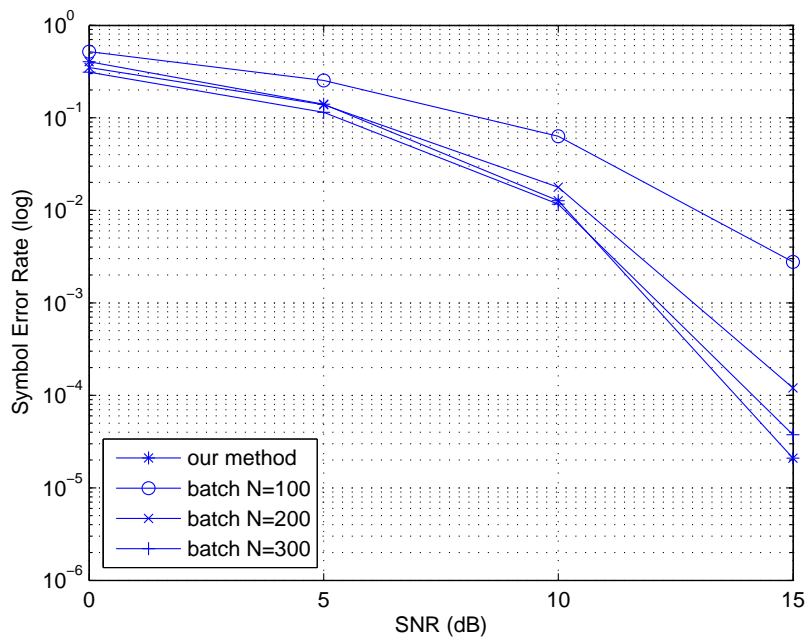


圖 4.11: PAM4 的符碼錯誤發生率比較

第 5 章 結論

我們藉由分析 [4] 提出的演算法，發現其提出的適應性演算法無法應用於多輸出通道的原因，接著以此為基礎分別針對抑制雜訊、提高信號增益兩方面的分析來改良目標函數，以克服原有演算法無法在多輸出通道提升 SINR 的缺點，並且得出加權因子 γ 最佳的選擇方式來取得消除符碼間干擾、抑制雜訊和提高信號增益之間的平衡以發揮演算法的最佳性能，最後再以 3.4.5 的相關證明得到一個簡單的方式來選擇 μ 值確保演算法的穩定性。

而模擬結果顯示了改良的演算法確實可以在多輸出通道有效地且快速地提升 SINR，而且其穩態的性能與批次演算法的性能不相上下，因此克服了原有的適應性演算法無法在多輸出通道提升 SINR 的缺點。模擬結果也說明了較大、較小的 γ 值都會造成改良的演算法性能下降，也證實了我們提出的 γ 選擇方式對改良演算法的性能確實有幫助。SNR 比較的實驗也顯示改良的演算法即使在信號雜訊比極低 (5dB) 的情況下仍然可以發揮作用，使得 SINR 高於通道輸出信號雜訊比，並且收斂速度不因高雜訊的影響而變慢，因此增進了改良演算法的實用性。最後錯誤發生率的實驗也與 SNR 的實驗結果吻合，證明我們改良的演算法性能的確達到了改善性能的目的。

參考文獻

- [1] D. N. Godard, “Self-recovering equalization and carrier tracking in two-dememsional data communication systems,” *IEEE Trans. Communications*, vol. 28, pp. 1867–1875, Nov 1980.
- [2] L. Tong, G. Xu, and T. Kailath, “Blind identification and equalization based on second-order statistics: A time domain approach,” *IEEE Trans. Information Theory*, vol. 40, pp. 340–350, Mar 1994.
- [3] E. Moulines, P. D. J.-F. Cardoso, and S. Mayrargue, “Subspace methods for the blind identification of multichannel fir filters,” *IEEE Trans. Signal Processing*, vol. 43, pp. 516–525, Feb 1995.
- [4] A. G. Orozco-Lugo and D. C. McLernon, “Blind channel equalization using chirp modulating signals,” *IEEE Trans. Signal Processing*, vol. 52, pp. 1364–1375, May 2004.
- [5] K. S. Narendra and A. M. Annaswamy, *Stable Adaptive Systems*, ch. 7. Prentice Hall, 1989.

- [6] K. S. Narendra and A. M. Annaswamy, "Persistent excitation of adaptive systems," *International Journal of Control*, vol. 45, pp. 127–160, 1987.
- [7] B. Farhang-Boroujeny, *Adaptive Filters: Theory and Applications*, pp. 122–123. John Wiley & Sons, 1999.

