

ASP Term Project: Enhanced Approaches for NLMS and RLS Algorithms

Group 7

廖邦宸、黃少辰、謝育葦

Abstract

本次專題我們實作了 Least Mean Squares (LMS) 以及 Recursive Least Squares (RLS) 作為 Baseline Method，並延伸出兩種不同的變體，分別是 Decision Feedback Equalizer (DFE) 以及 Data Reuse (DR)，期望能加強 Baseline Method 並獲得更低的錯誤率。同時我們也探討了 DFE 與 DR 會遇到的”Mismatch”問題，並提出了以 Decision Mode 的星座圖為主，Training Mode 的 BER 為輔的效能評估手段；也探討了不同 Channel 的特性以及各個演算法在不同 Channel 的優劣；比較各個演算法的時間複雜度；以及使用 Grid Search 進行參數搜尋；最後我們綜合前述實驗，給出我們最終所選擇的演算法組合。

Keywords-NLMS、RLS、DFE、DR、Adaptive filter

1 Introduction

在現代通訊系統中，適應性通道等化器 (Adaptive Channel Equalizer) 的主要目的是補償信號在傳輸過程中由於多路徑效應和高斯白噪音 (AWGN) 所引起的訊號失真。等化器通過 Adaptive Filter 技術，根據接收信號來恢復原始數據，這在無線通道傳輸中尤為重要。

而其中適應性濾波器演算法中最有名的例子便是 LMS 演算法。但是 LMS 具有收斂不穩定的缺點。為了改進這個問題，進一步產生了 NLMS 的演算法。經過少許的變動，便能夠有效的改進 LMS 所面臨的收斂穩定性的問題。而我們探討了 DFE [1] 的作法，DFE 相對於 LMS 和 RLS，主要的優勢在於其對 Inter-Symbol Interference (ISI) 的處理能力。DFE 結合了前饋濾波器 Feedforward Filter (FF) 和反饋濾波器 Feedback Filter (FB)，

能更有效地消除由過去符號引起的干擾。而 Data Reuse [2; 3] 則能夠讓原本沒有收斂的部分收斂得更完整。

我們將聚焦於幾種適應濾波器演算法的實現與比較，包括：1.NLMS 2.RLS 3.DFE-NLMS 4.DFE-RLS 5.Data Reuse NLMS 6.Data Reuse RLS 將以上的幾種方法應用在 3 種不同的通道中，分別為：Static Channel、Quasi-Static Channel 以及 Time-Varying Channel。比較訊號在 Training Mode 的 BER (與 Desired Signal 比較)、收斂速度以及 Decision Mode 的訊號通過濾波器等化後的星座圖 (集中至 QPSK 的四個星座點) 來比較各個方法在 3 個通道中的優劣勢。

2 Overview of the Problem Set

下圖 1 為我們此次實做的題目，我們得到的訊號為經過通道與 AWGN 污染後的 Training sequence 以及測試資料，必須透過 Adaptive Equalizer 以及 Desired Signal (未受污染的 Training sequence) 來訓練我們的濾波器係數並在 Decision Device 讓接收的訊號通過訓練好的 Adaptive Equalizer 將訊號等化並解調。

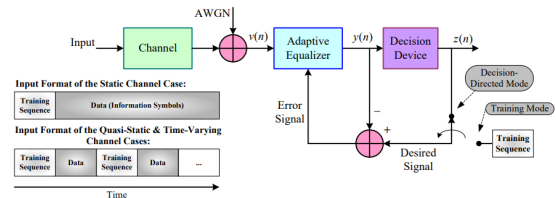


Figure 1: Problem Set

而通道分為 3 種，Static Channel、Quasi-Static Channel 以及 Time-Varying Channel。Static Channel 模擬了較為穩定的通道，資料給定的方式為 1000 個 training symbol 以及 200000 個 decision symbol；Quasi-Static Channel 以及 Time-Varying

Channel 為不穩定的通道，因此訓練的時間較短，資料給定的方式分別為 200 個 training symbol 以及 1000 個 decision symbol 和 50 個 training symbol 以及 400 個 decision symbol。

3 Method

NLMS NLMS 是 Least Mean Squares (LMS) 演算法的改進版，其目的是透過自適應調整濾波器係數，最小化誤差平方和。NLMS 相較於 LMS 的主要改進是引入了一個正規化因子，用來調整 Step Size 大小，使更新速度與輸入信號的功率相適應，避免因輸入信號能量變化而導致的收斂性問題。

RLS RLS 是一種基於最小化加權誤差平方和的自適應濾波演算法，能以遞歸方式更新濾波器係數。它考慮了過去所有的數據點，並對較新數據給予更大的權重（使用遺忘因子 λ ）。RLS 的性能優於 LMS 和 NLMS，但計算複雜度較高。

DFE 下圖 2 為 DFE 運作流程，並以 DFE-NLMS 的 Training Mode 來講解。DFE 最重要的功能便是消除過去訊號的干擾來讓預測更加準確。其中 FFF 為 feed forward filter，FBF 為 feedback filter。FFF 內的運作方式為利用 NLMS 更新 filter。FBF 內的係數也是通過 NLMS 去更新的。圖中的 Sign 的方塊我們利用 Soft Decision (這邊是讓 $y(n)$ 通過 Hyperbolic Tangent 函數) 得到 $y(n)$ 的判決結果。將這個結果延遲一段時間後送入 FBF 的 filter 中，用於消除過去的訊號造成的干擾。其中要注意的是如果 FBF 的判斷出錯，則錯誤會越變越大，因此才有前面的通過 Soft Decision 來縮小誤差。

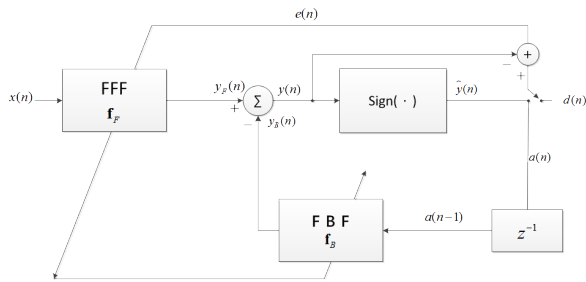


Figure 2: Structure of DFE

接著是架構圖裡各個訊號的說明。第一個是 $y_F(n)$ 表示的是通過 FFF (NLMS) 的 filter 後更新出來的結果。接著 $y_B(n)$ 代表的是過去訊號的猜測結果經過一個延遲後通過 FBF 所算出的結果。最後的 $y(n)$ 便是 $y_F(n) - y_B(n)$ (扣掉過去訊號的預測) 的最終答

案。 $e(n)$ 為 error 訊號，通過 $y(n)$ 與 Desired Signal ($d(n)$) 相減得出，並用於更新 FFF 以及 FBF。

接續之 Decision Mode 也是相同的作法，只是不再更新 FFF 以及 FBF 內的係數，最後 Decision Mode 所算出的 $y(n)$ 就是我們最後的預測結果。

而 DFE-RLS 的做法基本與 DFE-NLMS 的做法相同，只是 FFF 以及 FBF 在更新時利用的是 RLS 的演算法去更新。

Data Reuse Data Reuse 是一種重複利用訓練資料的方法，雖然訓練資料都屬於同一批樣本，但通過當下的 weights 狀態，得到的更新結果是不同的。Data Reuse 的優勢為經過多次遍歷，充分挖掘資料的潛在價值從而提升 Adaptive Equalizer 的效能。然而如果訓練時同一份資料使用太多次，可能會有 overfitting 的風險，讓 filter 只偏好於同一份或同一類型的資料，若對一個 overfitting 的模型輸入同一份訓練資料，會得到比上一輪更好的表現，但輸入新資料，則會得到較差的結果。

在實作的部分，我們會先找參數代入去看 filter 的訓練表現，且在每個 block 的 training sequence 會重複被用到好幾次來訓練，下圖 3 是整個 Data Reuse 的流程圖。我們首先選一組參數代入，依序輸入每個 block 的訓練資料，用訓練的結果去更新 filter 的參數。而更新參數的方法選擇用 NLMS 或 RLS，然後每個 block 都會用同樣的訓練資料重複去訓練，來幫助模型收斂。

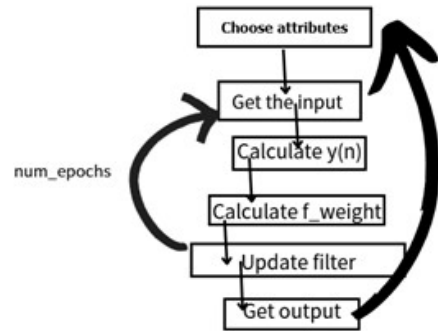


Figure 3: Structure of DR

4 Experiments

Overall BER Performance Comparison

我們實作了 NLMS、RLS、NLMS-DFE、RLS-DFE、DR-NLMS、DR-RLS 等六種演算法，我們透過每個 block 的 training sequence 計算模型在 training mode 下的平均 BER，並且只將 training sequence 的後 80% 納入計

算，避免尚未收斂的模型干擾 BER 的估計結果。下表為各個演算法在 Channel 1 (static)、Channel 2 (qstatic)、Channel 3 (varying) 下的位元錯誤率 (BER)；表 1 為高 SNR、表 2 為低 SNR。其中 **Best**、**Second Best** 分別以綠色和黃色表示。

Table 1: Overall BER of high SNR channel case.

Methods	Channel 1	Channel 2	Channel 3
NLMS	0.000625	0.020281	0.1086
RLS	0	0.0052031	0.03015
NLMS-DFE	0	0.010203	0.11132
RLS-DFE	0	0.0084062	0.0882
DR-NLMS	0.001	0.0045	0.0131
DR-RLS	0.002	0.0097	0.02272

Table 2: Overall BER of low SNR channel case.

Methods	Channel 1	Channel 2	Channel 3
NLMS	0.07125	0.069594	0.098775
RLS	0.059375	0.067891	0.07995
NLMS-DFE	0.06625	0.067828	0.099025
RLS-DFE	0.0625	0.065094	0.13005
DR-NLMS	0.06	0.05935	0.04576
DR-RLS	0.065	0.0607	0.044

Discussion of Convergence Speed 我們比較了 NLMS 與 RLS 的收斂速度，以下皆以 Channel 1 high SNR 的情況進行實驗。下圖 5 為實驗結果，可以看到 RLS 的收斂速度較 NLMS 快、RLS-DFE 的收斂速度較 NLMS-DFE 快。並且在經過大約 20% 的 training sequence 後，模型整體大致收斂。而 Data Reuse 由於會使用較大的 filter length，因此收斂時間必定慢於 NLMS 與 RLS。

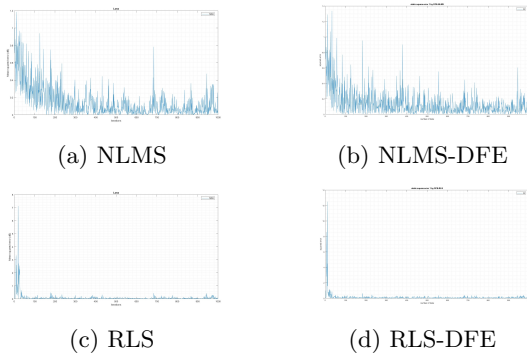


Figure 4: NLMS 和 RLS 之收斂速度比較。

Constellation Map of Decision Mode 實驗中我們發現，透過 training sequence 估計的 BER 與實際 Decision Mode 等化後的信號，兩者之間並非完全正相關。我們將此現象稱為”Mismatch”，此現象在加上 DFE 或是 Data Reuse 後尤為明顯。因此，我們下面繪製了 Adaptive Equalizer 所輸出的訊號 $y(n)$ ，分別在 training mode 和 decision mode 下的星座圖，來作為 BER 以外的第二個評估手段。確保模型在 training mode 和 decision mode (也就是實際資料) 下，都能正確地將訊號聚集於 QPSK 的四個星座點附近。

下圖 5 首先展示了 NLMS 與 RLS 在 Channel 1 high SNR 下進行實驗的結果。左半邊為 Training Mode；右半邊為 Decision Mode。不論是 NLMS 或是 RLS，假使其在 Training Mode 能順利將信號聚集於 QPSK 的四個星座點附近，該演算法在 Decision Mode 亦能將信號順利分離，我們稱此為”No-Mismatch”。亦即，當此演算法為”No-Mismatch”，我們可以藉由在 Training Mode 所估計的 BER 來推斷其在 Decision Mode 的 BER 的好壞，兩者應具有正相關性。

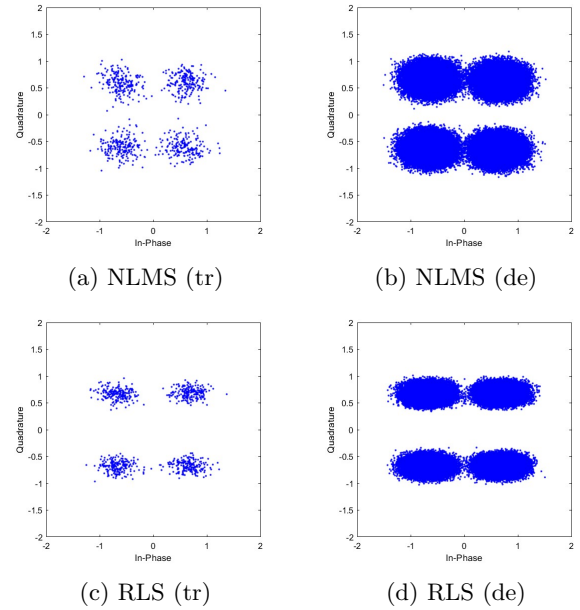


Figure 5: NLMS 和 RLS Training Mode 與 Decision Mode 之星座圖對比。

接著我們展示了發生”Mismatch”的情形，也就是 Training Mode 的 BER 並不能直接決定演算法的好壞，必須加上 Decision Mode 的星座圖做為評估手段，以下我們比較兩種演算法在 Decision Mode 下等化後信號的星座圖，下圖 6 分別是 DR-NLMS 與 DR-RLS 在 Channel 3 high SNR 進行實驗

的結果。根據表 1，DR-NLMS 的在 Training Mode 的 BER 低於 DR-RLS，但從 Decision Mode 的星座圖來看，DR-RLS 略為好於 DR-NLMS。因此，我們會以星座圖為主，BER 為輔來做為評估手段。以此情況為例，我們會選擇 DR-RLS 作為 Channel 3 high SNR 所使用的演算法。

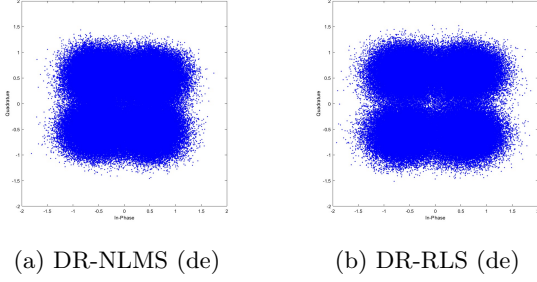


Figure 6: DR-NLMS 和 DR-RLS Decision Mode 之星座圖對比。

Discussion Of Algorithms In Different Channels 我們以三個 Channels 的特性來分析何種演算法較為適合。以 NLMS、RLS、DFE 來說，我們目標是找到一個盡可能大的 filter 長度，並且要在 training sequence 結束前能夠收斂。NLMS 收斂速度較慢，適用於靜態、半靜態通道 (Channel 1, 2)；RLS 收斂速度較快，更能適用於動態通道 (Channel 3)；DFE base 的方法能夠進一步降低 BER 的表現，但會增加複雜度與運算時間。至於 DR base 的方法，同樣會增加複雜度與運算時間，並且 DFE 與 DR 皆有可能有“Mismatch”的情形發生。

Parameters Searching Method 在調整模型參數時，我們使用了網格搜索 (Grid Search) 的技巧。亦即，先決定一個要搜索的參數範圍，然後以 for 迴圈去遍歷所有的可能的參數，最後選擇 BER 最低的那組參數。下圖 7 是 RLS 在 Channel 2 high SNR 下進行 Grid Search 的結果，由於 RLS 有兩個參數需要搜索，分別是 λ (遺忘係數) 和 L (filter 長度)，我們會產生一張 2D 的結果。圖中 BER 越小的越接近深藍色，越大的越接近淺黃色，可以看到最小的 BER 位於 $\lambda = 1$ 與 $L = 9$ 的位置，且在此圖中是“Local Minimum”，因此此參數為擁有最低 BER 之最佳參數。

值得注意的是，在實驗中我們發現，對於 Channel 1、2，RLS 的 λ 最佳參數為 1，亦即模型會記住所有歷史數據、而 Channel 3 RLS 的 λ 參數的最佳參數為 0.96。此結果符合我們對通道情形的預期，非時變通道 RLS 可以

透過記憶所有的歷史訊息來降低 BER；而時變通道則需要遺忘太過久遠的資訊，才能夠獲得較好的估計結果。

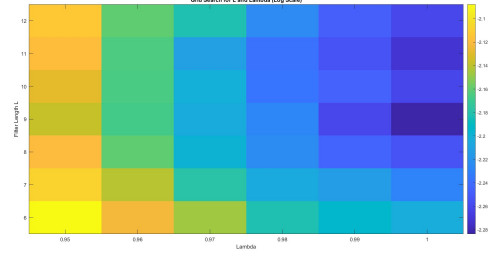


Figure 7: Grid Search of RLS.

Time Complexity Comparison 下表 3 為六種演算法之時間複雜度的比較，其中 L 代表 filter 長度、 N 代表 training sequence 的長度、 R 代表 Data Reuse 的次數。從 per update 的時間複雜度來看，NLMS 的複雜度比 RLS 低，加上 DFE 後複雜度會增加兩倍，因為 DFE 需要兩個 filter (feedforward 和 feedback) 進行計算；而 DR base 的演算法 (以 DR-NLMS 為例) 以 per update 的角度來看與 NLMS 相同，皆為 $\mathcal{O}(L)$ ，但從 entire training 的角度來看，會比 NLMS 多了 R 倍。

Table 3: Time Complexity of Different Algorithms.

Method	$\mathcal{O}(n)$ per update	$\mathcal{O}(n)$ entire training
NLMS	$\mathcal{O}(L)$	$\mathcal{O}(N \times L)$
RLS	$\mathcal{O}(L^2)$	$\mathcal{O}(N \times L^2)$
NLMS-DFE	$\mathcal{O}(2L)$	$\mathcal{O}(N \times 2L)$
RLS-DFE	$\mathcal{O}((2L)^2)$	$\mathcal{O}(N \times (2L)^2)$
DR-NLMS	$\mathcal{O}(L)$	$\mathcal{O}(R \times N \times L)$
DR-RLS	$\mathcal{O}(L^2)$	$\mathcal{O}(R \times N \times L^2)$

The Final Method Selection 綜上所述，我們決定各個 Channel 適用的演算法之準則為 Decision Mode 星座圖 > Training Mode BER > Time Complexity。下表是我們最終選擇的演算法以及其對應的 Channel。

Table 4: The Final Algorithm Selection.

Type of Channels	Type of Algorithms
Channel 1 high	RLS
Channel 1 low	RLS-DFE
Channel 2 high	RLS
Channel 2 low	RLS-DFE
Channel 3 high	DR-RLS
Channel 3 low	RLS

5 Work Distribution

下表為本次的分工表，總共分爲五個階段進行：Stage 1，共同討論 channel 的特性及可行之演算法 -> Stage 2，分配實作之演算法 -> Stage 3，整合三人的程式碼並討論結果 -> Stage 4，簡報製作以及上台報告 -> Stage 5，書面報告製作。

Table 5: 分工表

	廖邦宸	黃少辰	謝育葦
Stage 1	o	o	o
Stage 2	baseline	DFE	DR
Stage 3	o	o	o
Stage 4	o	o	o
Stage 5	o	o	o

6 Conclusion

在此次專題中，我們分別研究了 Decision Feedback Equalizer (DFE) 和 Data Reuse (DR) 兩種方法，DFE 主要的功能是消除過去訊號的干擾來讓預測更加準確，而 DR 是利用相同的資料重複訓練 Adaptive Equalizer。我們將 NLMS、RLS、DFE-NLMS、DFE-RLS、DR-NLMS、DR-RLS 實作於給定的三種 channel 上 (包含 high SNR 和 low SNR)，分別觀察每種方法在六組不同資料的 BER 以及複雜度等各種表現，並用 Grid Search 找出可得到最低 BER 的參數組合。最後，我們透過觀察測試資料通過 Adaptive Equalizer 的輸出在星座圖上的表現，以及參考訓練資料在 BER 的表現，我們最終選擇 RLS、RLS-DFE、DR-RLS 作為我們的主要方法。

References

- [1] F. Lounoughi and M. Djendi, “A decision feedforward channel equalization system based pseudo affine projection (DFE-PAP) algorithm,” *Appl. Acoust.*, vol. 214, no. 109693, p. 109693, 2023.
- [2] C. Paleologu, J. Benesty, and S. Ciochina, “Data-reuse recursive least-squares algorithms,” *IEEE Signal Process. Lett.*, vol. 29, pp. 752–756, 2022.
- [3] W. Gao, J. Chen, and C. Richard, “Theoretical analysis of the performance of the

data-reuse RLS algorithm,” *IEEE Trans. Circuits Syst. II Express Briefs*, vol. 71, no. 1, pp. 490–494, 2024.

- [4] P. M. Clarkson, *Optimal and Adaptive Signal Processing*. Boca Raton, FL: CRC, 1993.
- [5] S. Haykin, *Adaptive Filter Theory*. 3rd Ed., Prentice-Hall, 1996.