

國立臺灣大學工學院機械工程學系

碩士論文

Department of Mechanical Engineering

College of Engineering

National Taiwan University

Master Thesis



以多重性能偏移特性辨識與校準複雜系統參數之方法  
Identification and Calibration of Complex Model  
Parameters via Multiple Performance Deviations

李旻憲

Min-Hsien Lee

指導教授：詹魁元博士

Advisor: Kuei-Yuan Chan, Ph.D.

中華民國 107 年 8 月

August, 2018



---

## 誌謝

---

研究所的這兩年受到許多人的幫助，使我能夠順利度過中間的困難和瓶頸，事非經過不知難，最後能完成這本碩士論文，我心中充滿了感謝。

謝謝我的指導教授詹魁元老師耐心的指導和協助，不論是研究還是做人處事方面，我都從老師學習到很多，未來的日子我也會把老師的教誨放在心上。謝謝吳文方老師和劉霆老師擔任我的口試委員，兩位老師給予的建議和指點，使我的論文的內容能更加完善。

謝謝實驗室的同仁，SoLab 就像一個大家庭，大家總是熱心的互相幫助，謝謝學長姐們，彥智、子頡、米約瑟、柏伸、岳羿、柏安、柏伸、冠霖、右均、顯主、盈樺、欣怡、冠龍，自己也成為學長時，更能體會學長姐們的經驗傳承有多重要。謝謝同屆的夥伴們，穎寬、峻廷、世哲，我們共同度過了許多難關，互相扶持走到了最後一哩路。謝謝學弟妹們，柏宇、員成、心婷，實驗室有你們歡樂許多。謝謝助理期璟、柏賢幫忙處理許多實驗室事務。

謝謝家人的支持和陪伴，你們是我堅持下去的動力來源，完成這階段性任務後，我要踏上另一段旅程了。



---

## 摘要

---

參數數值無法確定是影響系統性能及可靠度的主要原因之一，本研究建立辨識校準參數的流程，以確認運行系統之各參數數值。然而校準參數在複雜系統應用上可能會遇到問題有 (1) 參數過多造成校準困難，(2) 參數校準準確率不足，以及 (3) 參數校準結果信心水準不足。本研究藉由主因素分析找出系統的重要參數，降低複雜系統的分析難度，根據系統性能偏移，以類神經網路校準參數數值，再利用多個根據不同性能偏移以類神經網路校準參數的結果，以決策樹提升校準準確率，並以信賴區間評估參數的校準結果。研究以一車輛動態測試的工程案例作為演示，車輛參數校準方均根誤差最小可達 0.136%，本研究所提出之方法可有效校準偏移之參數，並提供校準複雜系統參數的完整分析流程。

**關鍵字:** 複雜系統分析、參數不確定因素、參數數值估計、主因素分析、類神經網路、決策樹



---

# Abstract

---

Parameter uncertainty plays an important role in system performance and robustness. This research builds up a procedure for calibrating deviated parameters. However, there may be difficulties applying parameter calibration in complex system, namely (1) computation inefficiency due to a large number of parameters, (2) inaccuracy in parameter calibration, and (3) low confidence in calibration result. This research selects important parameters by main effect analysis and uses the neural network to calibrate parameters via performance deviation. After getting calibration results via different performance deviation, we use the decision tree to increase the accuracy of calibration and evaluate the result by applying confidence interval. The method is demonstrated in an engineering case: vehicle dynamic test, the minimum mean square error of calibration is 0.136%.

**Keywords:** complex system analysis, parameter uncertainty, parameter calibration, main effect analysis, neural network, decision tree



---

# 目錄

---

誌謝 .....	i
摘要 .....	ii
Abstract .....	iii
目錄 .....	iv
圖目錄 .....	vii
表目錄 .....	ix
符號列表 .....	xi
第一章 緒論 .....	1
1.1 研究背景 .....	1
1.2 研究動機與目的 .....	2
1.2.1 研究動機 .....	2
1.2.2 研究目的 .....	3
1.3 論文架構 .....	4
第二章 文獻回顧 .....	6
2.1 參數估計方法 .....	6
2.2 參數估計問題 .....	12
2.3 小結 .....	13



第三章 校準方法概念 .....	15
3.1 校準參數方法一：正向校準 .....	15
3.2 校準參數方法二：逆向校準 .....	18
3.3 參數校準檢驗方法 .....	20
第四章 研究方法 .....	23
4.1 方法流程 .....	23
4.2 找出重要參數 .....	24
4.3 建立訓練集、驗證集、測試集 .....	28
4.4 建立正向校準模型 .....	30
4.5 建立逆向校準模型 .....	33
4.6 提升參數校準準確率 .....	36
4.7 校準參數 .....	40
第五章 案例探討 .....	42
5.1 車輛工程案例 .....	42
5.1.1 找出重要參數 .....	45
5.1.2 建立訓練集、驗證集、測試集 .....	48
5.1.3 建立正向校準模型 .....	48
5.1.4 建立逆向校準模型 .....	52
5.1.5 校準參數 .....	55
5.1.6 結果與討論 .....	59
第六章 結論 .....	62
6.1 研究貢獻 .....	62
6.2 未來展望 .....	63



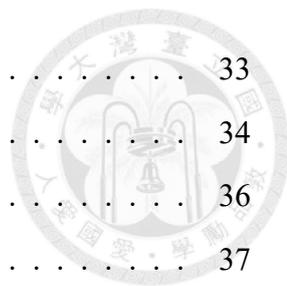


---

# 圖目錄

---

1.1	汽車系統示意圖 . . . . .	2
1.2	物理模型與真實系統 . . . . .	3
1.3	論文架構圖 . . . . .	4
2.1	系統參數估計流程 . . . . .	7
2.2	神經網路結構示意圖 . . . . .	11
2.3	決策樹示意圖 . . . . .	12
3.1	$E_1$ 分布圖 . . . . .	17
3.2	參數分布圖 . . . . .	19
3.3	$p_{m_1}$ 搜尋範圍 . . . . .	20
3.4	$p_{m_2}, p_{m_3}$ 搜尋範圍 . . . . .	20
3.5	信賴區間 . . . . .	21
3.6	$p_{m_1}$ 搜尋範圍 . . . . .	22
3.7	$p_{m_2}$ 搜尋範圍 . . . . .	22
3.8	$p_{m_3}$ 搜尋範圍 . . . . .	22
4.1	主流程圖 . . . . .	24
4.2	主因素分析示意圖 1 . . . . .	25
4.3	主因素分析示意圖 2 . . . . .	26
4.4	神經網路架構示意圖 1 . . . . .	30
4.5	預測結果示意圖 . . . . .	32



4.6	$E_{\text{set}}$ 接受域和拒絕域	33
4.7	神經網路架構示意圖 2	34
4.8	使用更多訓練資料結果示意圖	36
4.9	差異性小的性能輸出示意圖	37
4.10	差異性大的性能輸出示意圖	38
4.11	使用不同性能輸出結果示意圖	38
4.12	使用更多測試方法示意圖	39
4.13	使用更多測試方法結果示意圖	40
4.14	校準參數流程	40
5.1	Matlab Simulink 示意圖	42
5.2	車輛模型示意圖	43
5.3	性能輸出計算流程	43
5.4	動力計算	44
5.5	動力分配	44
5.6	性能輸出計算	45
5.7	測試方法	45
5.8	新測試方法	46
5.9	原 $\mathbf{y}_{m_1}^1$	49
5.10	移除重複值後的 $\mathbf{y}_{m_1}^1$	49
5.11	$E_{\text{set}}$ 分布圖	52
5.12	逆向校準誤差分布	54
5.13	位於接受域的參數分布	57
5.14	$p_{m_5}, p_{m_6}$ 接受域分布	58
5.15	$p_{m_5}$ 接受域分布	59
5.16	使用更多訓練資料結果	60
5.17	使用更多測試方法結果	61



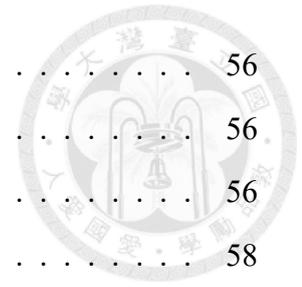
---

## 表目錄

---

3.1	單邊信賴區間 z 值表 . . . . .	21
4.1	設計變數分布 . . . . .	27
4.2	主因素分析結果 . . . . .	27
4.3	校準模型預測結果 (MSE=0.0414) . . . . .	27
4.4	校準模型預測結果 (MSE=0.0023) . . . . .	28
4.5	設計變數分布 . . . . .	35
5.1	量測輸出 . . . . .	46
5.2	主因分析參數 . . . . .	47
5.3	主因素排名 . . . . .	47
5.4	參數設定值 . . . . .	48
5.5	參數標準化結果 . . . . .	48
5.6	剩下性能輸出數值的數量 . . . . .	50
5.7	保留模型數量 . . . . .	51
5.8	接受域 . . . . .	52
5.9	驗證集單一性能輸出方均根誤差 . . . . .	53
5.10	驗證集決策樹方均根誤差 . . . . .	54
5.11	驗證集誤差 . . . . .	55
5.12	接受域搜尋範圍 . . . . .	55
5.13	拒絕域搜尋範圍 . . . . .	55
5.14	參數實際值 . . . . .	55

5.15	參數預測值 . . . . .	56
5.16	參數搜尋範圍 . . . . .	56
5.17	更新參數校準結果 . . . . .	56
5.18	參數搜尋範圍 . . . . .	58
5.19	訓練集、驗證集、測試集方均根誤差 . . . . .	59
5.20	測試集單一性能輸出方均根誤差 . . . . .	61
5.21	測試集決策樹方均根誤差 . . . . .	61





## 符號列表

$E_{\text{set}}$	估計性能輸出之誤差值
$f_e(\mathbf{p}_e; \mathbf{x})$	真實系統函數
$f_m(\mathbf{p}_m; \mathbf{x})$	物理模型函數
$\hat{f}_k^{dt}(\mathbf{p}_{\text{set}})$	以保留的物理模型參數估計值 $\mathbf{p}_{\text{set}}$ 預測第 $k$ 個參數 $\hat{\mathbf{p}}_{m_k}$ 的決策樹模型函數
$\hat{f}_{i,j}^{fwd}(\mathbf{p}_m)$	以參數 $\mathbf{p}_m$ 預測第 $i$ 個性能輸出第 $j$ 個值 $\hat{y}_{m_{i,j}}$ 的正向校準模型函數
$\hat{f}_{i,k}^{inv}(\mathbf{y}_{m_i})$	以第 $i$ 個性能輸出 $\mathbf{y}_{m_i}$ 預測第 $k$ 個參數 $\hat{p}_{m_k}$ 的逆向校準模型函數
$n_e$	真實系統參數數量
$n_f$	保留正向模型數量
$n_i$	保留逆向模型數量
$n_m$	重要參數數量
$n_p$	物理模型參數數量
$n_s$	取樣數量
$n_{\text{set}}$	參數組合數量
$n_{\text{test}}$	測試集數量
$n_{\text{train}}$	訓練集數量
$n_{\text{val}}$	驗證集數量
$n_x$	測試方式數量
$n_y$	性能輸出數量
$\mathbf{p}_e$	真實系統所有參數
$p_{e_k}$	真實系統第 $k$ 個參數



$\mathbf{p}_m$	物理模型所有參數
$\mathbf{p}_m^r$	第 $r$ 次實驗物理模型參數設定
$\hat{\mathbf{p}}_m$	物理模型所有參數估計值
$p_{m_k}$	物理模型第 $k$ 個參數
$\hat{p}_{m_k}$	物理模型第 $k$ 個參數估計值
$R_{i,j}$	物理模型第 $i$ 個性能輸出的第 $j$ 項之標準化常數
$\mathbf{R}_{\text{set}}$	保留的物理模型性能輸出值之標準化常數
$R_{\text{set},i}$	保留的物理模型性能輸出值之標準化常數第 $i$ 項
$\mathbf{x}$	測試方式
$\mathbf{y}_{e_i}$	真實系統第 $i$ 個性能輸出
$y_{e_i,j}$	真實系統第 $i$ 個性能輸出的第 $j$ 項
$\hat{y}_{e_i,j}$	真實系統第 $i$ 個性能輸出的第 $j$ 項估計值
$\mathbf{y}_{m_i}$	物理模型第 $i$ 個性能輸出
$y_{m_i,j}$	物理模型第 $i$ 個性能輸出的第 $j$ 項
$\hat{y}_{m_i,j}$	物理模型第 $i$ 個性能輸出的第 $j$ 項估計值
$y_{m_i,j}^{k+}$	第 $k$ 個參數高水準第 $i$ 個性能輸出的第 $j$ 項之值
$y_{m_i,j}^{k-}$	第 $k$ 個參數低水準第 $i$ 個性能輸出的第 $j$ 項之值
$\bar{y}_{m_i,j}$	參數為原設定值時第 $i$ 個性能輸出的第 $j$ 項之值
$\mathbf{y}_{m_i,j}^r$	第 $r$ 次虛擬實驗，物理模型第 $i$ 個性能輸出的第 $j$ 項估計值
$\mathbf{y}_{\text{set}}$	保留的物理模型性能輸出值
$\hat{\mathbf{y}}_{\text{set}}$	保留的物理模型性能輸出估計值
$y_{\text{set},i}$	保留的物理模型性能輸出值第 $i$ 項之值
$\hat{y}_{\text{set},i}$	保留的物理模型性能輸出估計值第 $i$ 項之值
$\varepsilon_{\text{max}_k}$	驗證集中預測第 $k$ 個參數 $p_{m_k}$ 的誤差最大值
$\varepsilon_{s_k}$	驗證集中預測第 $k$ 個參數 $p_{m_k}$ 的誤差標準差



---

# 緒論

---

## 1.1 研究背景

分析模型是工程領域了解系統特性的重要工具，利用物理、數學原理建立和系統相近的物理模型，例如汽車動態模型、飛機流場模型等，藉由這些模型的模擬結果了解系統特性，可以減少製造測試用的原型與實驗成本，是開發工程系統時常見的方法 [1] [2] [3]。

在複雜系統的分析上，模型更是可以快速的了解系統行為、評估系統性能偏差原因的方法。以汽車這一個典型的複雜系統為例，我們會建立汽車動態模型模擬汽車系統，如圖1.1，這些系統又可分為車體系統、傳動系統、煞車系統、轉向系統等主要子系統，而這些子系統下又有更多的細部子系統，例如傳動系統下還有引擎系統、變速箱系統。這些細部子系統的參數眾多，函數複雜，因此不論是要了解各細部子系統間的交互作用或是掌握其參數數值，都是艱鉅的任務。



圖 1.1: 汽車系統示意圖

## 1.2 研究動機與目的

### 1.2.1 研究動機

在真實世界的複雜系統運行時更受許多不確定因素的影響，使其性能發生偏移而不如原先期望，常見的不確定因素有：

1. 製造過程不確定：元件在製造時受加工精度、組裝偏差影響，而和原設計有差異，例如：輪胎的鋼絲因為組裝偏差有不同的張力。
2. 材料性質不確定：特定材料的性質因材料的來源，處理的過程不同而有所差異，例如：輪胎的橡膠彈性係數會因為原材料批號不同而有差異。
3. 環境參數不確定：假設的環境參數因環境發生變異，與實際情形不同，例如：輪胎在道路乾燥和濕滑時有不同的的滾動阻力。

這些不確定因素會造成系統參數偏移原本的設定值，系統參數偏移又會導致實際運行時系統性能發生偏移，造成與原本期望結果的差異。當系統的性能表現和原本期望的結果不同時，我們通常藉由調整系統參數使系統的性能表現和期望結果一致，而系統參數經過適當的的調整，亦能穩定系統的性能表現，提升系統

的可靠度。因此不論是為了改善系統的性能表現，或是提升系統的可靠度，正確掌握系統參數都是重要的一環。

然而真實世界的系統中有許多參數不易量測，例如車輛的轉動慣量、輪胎的滾動阻力，若尚需考慮參數的量測誤差以及系統性能的量測誤差，系統性能輸出偏移和系統參數偏移之實際關係將非常難以得知。由於我們多數時間僅能靠系統的性能輸出推估系統的參數值，建立物理模型是常見的替代作法。

系統與物理模型的關係如圖1.2，假設系統有  $n_p$  個參數，全共為  $p_{e_1}, p_{e_2}, \dots, p_{e_{n_p}}$ ，可表示為  $\mathbf{p}_e$ ；以及  $n_y$  個性能輸出， $\mathbf{y}_{e_1}, \mathbf{y}_{e_2}, \dots, \mathbf{y}_{e_{n_y}}$ ，可表示為  $\mathbf{y}_e$ ，經由測試方式  $\mathbf{x}$ ，每個性能輸出產生  $n_x$  個值。而物理模型有  $n_p$  個參數， $p_{m_1}, p_{m_2}, \dots, p_{m_{n_p}}$ ，可表示為  $\mathbf{p}_m$ ；以及  $n_y$  個性能輸出， $\mathbf{y}_{m_1}, \mathbf{y}_{m_2}, \dots, \mathbf{y}_{m_{n_y}}$ ，可表示為  $\mathbf{y}_m$ ，亦經由測試方式  $\mathbf{x}$ ，每個性能輸出產生  $n_x$  個值，其中下標  $e$  代表 experiment、 $m$  代表 model、 $p$  代表 parameter、 $y$  代表性能輸出、 $x$  代表測試方式。

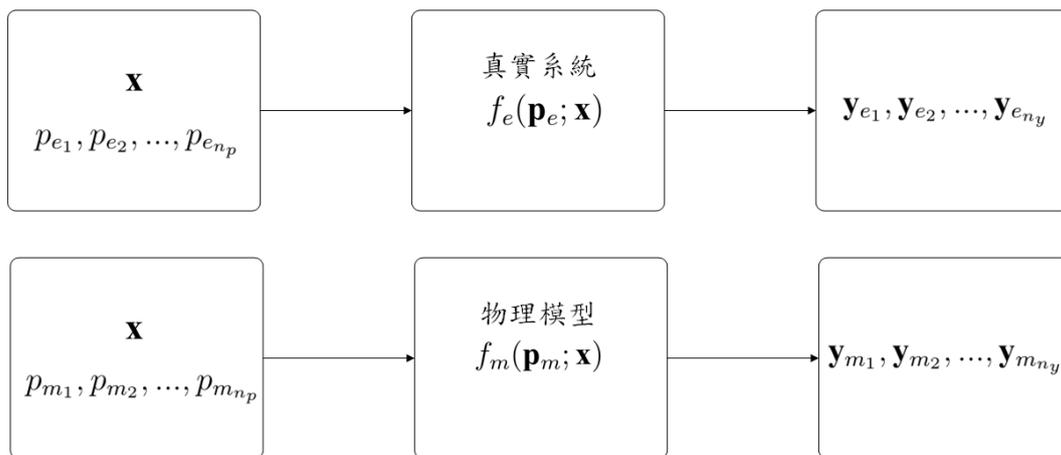


圖 1.2: 物理模型與真實系統

以物理模型模擬系統，雖可掌握各細部子系統的函數，但子系統函數之間的交互作用複雜，參數眾多，造成難以直接藉由性能輸出，以數學方程式回推參數數值，此外，系統的性能偏差可能存在多種潛在的參數組合 [4]，眾多的參數、函數複雜的交互作用、多種潛在的參數組合都增加了掌握參數的困難。

## 1.2.2 研究目的

綜合上述討論結果，校準系統參數的準確率及效率仍有問題尚待解決，本研究的重點有：

1. 針對複雜系統的眾多參數，找出對系統性能影響較大的重要參數，將問題聚焦到校準重要參數上。
2. 針對參數校準的困難之處，提出解決方法，以提升參數校準的準確率及效率。
3. 針對參數校準結果，建立檢驗校準結果的方法。
4. 針對複雜系統參數偏移辨識與校準，建立完整的分析流程。

本論文建立一套分析方法，協助工程師經由系統輸出有效推估系統參數，進而了解實際影響系統輸出偏差的主要不確定因素。

### 1.3 論文架構

本研究共分為六個章節，架構如圖1.3所示：

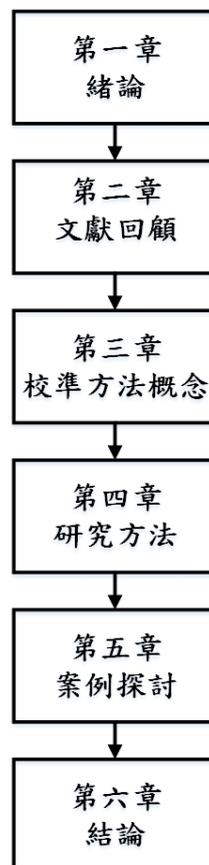


圖 1.3: 論文架構圖



1. **第一章 緒論**  
介紹研究主題，並敘述研究的動機與目的。
2. **第二章 文獻回顧**  
探討複雜系統參數校準的相關方法與文獻。
3. **第三章 校準方法概念**  
說明參數校準方法概念，以及如何檢驗校準結果。
4. **第四章 研究方法**  
說明參數校準的方法，並詳細介紹其細節。
5. **第五章 案例探討**  
利用一車輛工程案例探討研究方法的成效。
6. **第六章 結論**  
整理本研究之研究成果與討論，並敘述後續研究的方向。



---

## 文獻回顧

---

### 2.1 參數估計方法

無論是實驗量測或是物理模型，我們都是希望找到真實系統的現象，而兩者之間又有一定的差距，暫且將這差距定義為誤差，若誤差太大我們認為物理模型模擬結果和實驗結果不一致，因此重新調整模擬參數設定，直至誤差小於一某可接受範圍，則我們認為物理模型有足夠代表性重現系統的行為。圖2.1說明我們如何以物理模型估算系統的實驗輸出，取代實際的實驗，並校準偏移參數的方法概念。在圖2.1中，我們以物理模型估算系統輸出，並和系統實驗輸出進行比較，若誤差小於一某可接受範圍，則取得估計參數，否則更新估計參數，重新估算系統輸出。

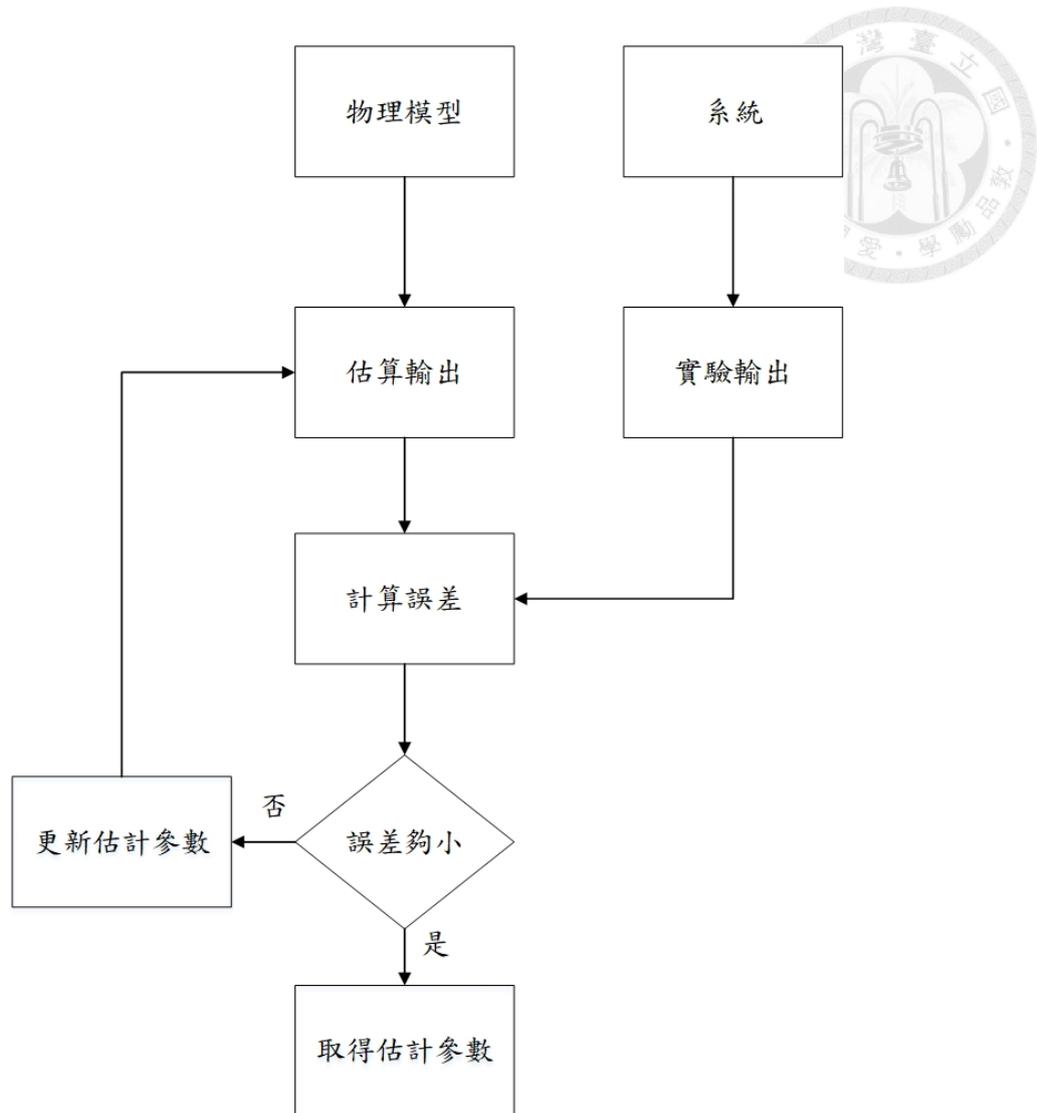


圖 2.1: 系統參數估計流程

針對以物理模型估算實驗輸出，匹配實際實驗輸出，進行參數估計的方法，常見的方法有：最大似然法、貝氏推估、神經網路、決策樹，本節將介紹這些方法的原理及優缺點。

### 1. Maximum Likelihood Methods 最大似然法 [5]

最大似然法以機率的概念找出最可重現實驗結果的參數組合，似然函數可表示為式2.1， $y$  為量測值， $\theta$  為欲估計之參數，

藉由找出能最大化似然函數的  $\hat{\theta}$  如式2.2，似然函數有最大值的必要條件為式2.3，以此法達成估計參數  $\theta$ ，若假設的似然函數與真實情形不符，則無法獲得準確結果。

$$L(y|\theta) \tag{2.1}$$



$$L(\mathbf{y}|\boldsymbol{\theta})|_{\boldsymbol{\theta}=\hat{\boldsymbol{\theta}}} \quad (2.2)$$

$$\frac{\partial}{\partial \boldsymbol{\theta}} L(\mathbf{y}|\boldsymbol{\theta})|_{\boldsymbol{\theta}=\hat{\boldsymbol{\theta}}} = 0 \quad (2.3)$$

以估計  $\mathbf{y}$  的平均值  $\mu$  和標準差  $\sigma^2$  為例，假設  $\mathbf{y}$  為高斯分布 (Gaussian Distribution)，則  $\mathbf{y}$  之似然函數如式2.4，對  $\mathbf{y}$  進行  $m$  次量測，藉由最大化似然函數，求  $\mathbf{y}$  的平均值和標準差  $\mu, \sigma^2$ 。

$$L(\mathbf{y}|\mu, \sigma^2) = \left( \frac{1}{2\pi\sigma^2} \right)^{\frac{m}{2}} e^{-\frac{\sum_{i=1}^m (y_i - \mu)^2}{2\sigma^2}} \quad (2.4)$$

為滿足似然函數最大值的必要條件，將  $L(\mathbf{y}|\mu, \sigma^2)$  對  $\mu, \sigma^2$  進行偏微分，其結果等同於將  $\ln[L(\mathbf{y} | \mu, \sigma^2)]$  對  $\mu, \sigma^2$  進行偏微分。將  $\ln[L(\mathbf{y} | \mu, \sigma^2)]$  對  $\mu$  偏微分如式2.5，可推得  $\hat{\mu}$  如式2.6，而將  $\ln[L(\mathbf{y}|\mu, \sigma^2)]$  對  $\sigma^2$  偏微分如式2.7，可推得  $\hat{\sigma}^2$  如式2.8。

$$\frac{\partial}{\partial \mu} \ln[L(\mathbf{y}|\mu, \sigma^2)] = 0 \quad (2.5)$$

$$\hat{\mu} = \bar{y} = \sum_{i=1}^m y_i / m \quad (2.6)$$

$$\frac{\partial}{\partial \sigma^2} \ln[L(\mathbf{y}|\mu, \sigma^2)] = 0 \quad (2.7)$$

$$\hat{\sigma}^2 = \sum_{i=1}^m \frac{(y_i - \hat{\mu})^2}{m} \quad (2.8)$$

## 2. Bayesian Estimation 貝氏推估

貝式推估結合似然率和條件機率的概念，若有 A 和 B 兩事件，則 B 事件發生的情況下，A 事件發生的機率為  $p(A|B)$  如式2.9，而 A 事件發生的情況下，B 事件發生的機率為  $p(B|A)$  如式2.10。

由於  $p(A \cap B) = p(B \cap A)$  如式2.11，我們可將  $p(A|B)$  改寫如式2.12。我們稱  $p(A)$  為假設的先驗機率， $p(A|B)$  為後驗機率，為對  $p(A)$  的進一步了解，而  $p(B|A)$  為假設的似然率，因此貝氏推估亦需考量似然函數假設的正確性，此外，先驗機率的正確性也會影響貝氏推估的結果 [6]。

$$p(A|B) = \frac{p(A \cap B)}{p(B)} \quad (2.9)$$

$$p(B|A) = \frac{p(B \cap A)}{p(A)} \quad (2.10)$$

$$p(A \cap B) = p(B \cap A) = p(A|B)p(B) = p(B|A)p(A) \quad (2.11)$$

$$p(A|B) = \frac{p(B|A)p(A)}{p(B)} \quad (2.12)$$

現欲推估參數  $\theta$  原先的機率分布為  $p(\theta)$ ，此為  $\theta$  的先驗機率分布。今已知  $p(\mathbf{y})$  後，可更新其機率分布為  $p(\theta|\mathbf{y})$  如式2.13，此為  $\theta$  的後驗機率分布，以此方法估計  $\theta$  之值。

$$p(\theta|\mathbf{y}) = \frac{p(\mathbf{y}|\theta)p(\theta)}{p(\mathbf{y})} \quad (2.13)$$

貝氏推估應用於參數推估的案例：2015 年，Green 等人 [7]，以貝氏推估估計震盪器 (oscillator) 的參數數值。

### 3. Neural Network 神經網路

近年由於電腦運算力的提升，機器學習成為熱門的工具，其中神經網路 (Neural Network) 被廣泛的使用，其優點有不必對系統有全面的了解，且能產生複雜模型 [8]，具強擬和能力，但須避免過擬和的情形發生。

神經網路的構成由輸入層 (input layer)，一或多個隱藏層 (hidden layer) 及輸出層 (ouput layer) 構成，如圖2.2，輸入層為輸入神經網路的參數，輸出層

為神經網路輸出結果，隱藏層具有多個神經元，每個神經元具有激勵函數 (activation function)，目的是增加神經網路的非線型，提升神經網路的擬和能力，常使用的激勵函數如式2.14

$$F(x) = \frac{1}{1 + e^{-x}} \quad (2.14)$$

由於參數估計屬於回歸問題，因此輸出層神經元的激勵函數為式 2.15。

$$F(x) = x \quad (2.15)$$

若第  $i-1$  層有  $n_{i-1}$  個神經元，則第  $i$  層的第  $j$  個神經元之值為式 2.16，再經過激勵函數處理即為最終輸出值。

$$x_{i,j} = \sum_{k=1}^{n_{i-1}} w_{i,j} \times x_{i-1,k} + b_{i,j} \quad (2.16)$$

而輸出層之誤差利用反向傳播 (back propagation)，回傳至隱藏層，調整隱藏層神經元的權重  $w_{i,j}, b_{i,j}$ ，反向傳播的概念可參見 [9]。神經網路於參數推估的應用：

1990 年 Narendra 提出運用神經網路進行參數推估的方法 [10]，2014 年 Wei 等人以神經網路預測電池之剩餘電量 [11]，2017 年 Boda 等人以神經網路預測車輛的轉角值 [12]。

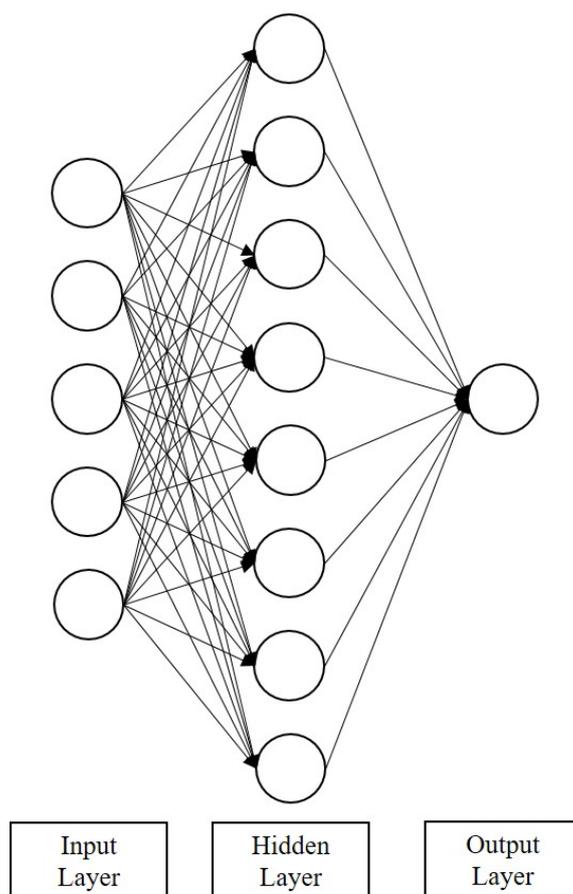


圖 2.2: 神經網路結構示意圖

#### 4. Decision Tree 決策樹

決策樹的基本演算法為：根據實驗數據特徵訂定指標，依據指標將實驗數據分為數個子集合，在子集合中重複此過程，其流程示意圖如圖2.3，雖然決策樹的複雜程度不如神經網路，但相較於神經網路，能產生較容易理解的規則：以  $x$  之值作為分類的依據。以決策樹推估系統性能的案例如：2007 年，Geoffrey 等人以決策樹推估系統消耗電量 [13]。

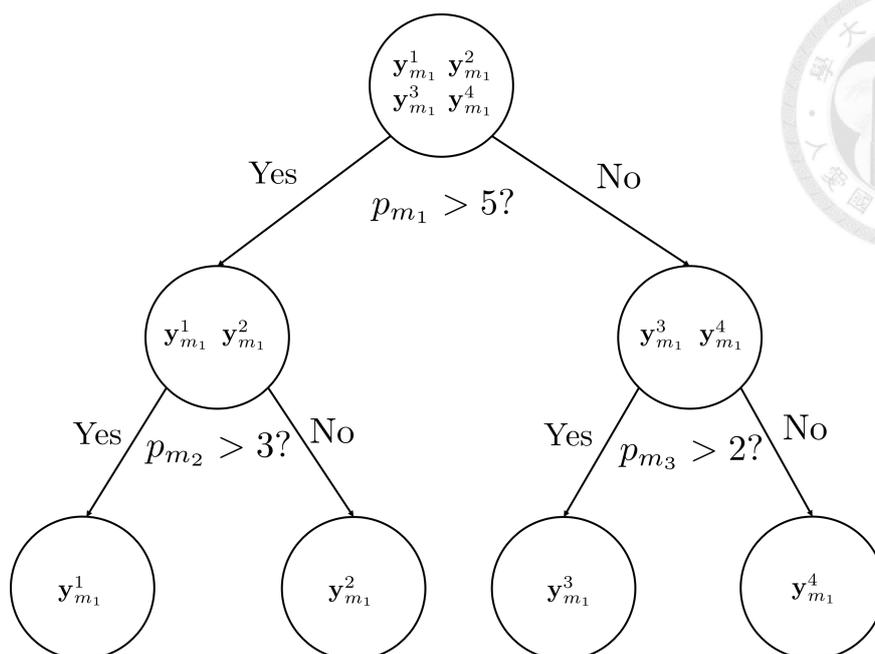


圖 2.3: 決策樹示意圖

總結以上討論，貝氏推估、最大似然法皆需考量似然函數假設的正確性，若假設的似然函數與真實情形不符，則無法獲得準確結果，此外，先驗資訊的準確率也會影響貝氏推估的結果。神經網路由於能產成複雜的模型，具強大的擬和能力，因此相當適合分析複雜系統，但須避免過擬和的現象，而決策樹雖無神經網路強大的擬和能力，但較神經網路易產生容易理解的規則。

## 2.2 參數估計問題

無論使用何種推估性能輸出的方法，例如：最大似然法、貝氏推估、神經網路、決策樹，目的皆是最小化實際性能輸出和估算性能輸出之誤差，以進行參數估計，在這樣推估的過程中，常見的問題有：

1. 無法找到全域最佳解：參數校準結果為區域最佳解。
2. 參數有多組可能解：有多組參數對應相似的性能輸出。
3. 參數校準準確率不足：參數校準結果和實際參數數值不符。

最小化實際性能輸出和估算性能輸出之誤差的過程中，實際性能輸出和估算性能輸出之誤差即為目標函數，由於目標函數可能為非凸 (nonconvex) 函數，最小

化目標函數的過程中，可能找到目標函數的區域最佳解 [14] [15]，為找到全域最佳解，可使用多個隨機的起始點，以避免解為區域最佳解，但隨著參數數量增加，搜尋範圍變得更大，使用多個起始點的方法效率低落且不切實際 [16]。此外，若有多組參數對應相似的性能輸出，也就是參數有多組解，如何判斷參數是否為可能的解，更提升了參數推估的困難性 [4]。本研究將針對以上問題，在第三章說明如何提升搜尋參數解的效率及檢驗其可信度的方法。

本研究亦重視參數校準準確率的問題，為提升校準參數數值的準確率，Bayarri [6]、Gábor [17] 等人提出以精確的先驗資訊提升參數校準的準確度，然而在推論參數前，大多數情形難以獲得準確的先驗資訊。對應此困難，Arendt [18]、Conti [19] 等人建議使用多個測試實驗，提升參數校準準確率，本研究將在第四章說明完整的研究方法及提升參數校準準確率的策略。

## 2.3 小結

根據文獻內容，針對參數校準的問題，尚有以下問題需要解決：

1. 如何有效率的校準多個參數：

需校準之參數有  $p_{m_1}, p_{m_2}, \dots, p_{m_{n_p}}$ ，在眾多參數中，應設法找出應優先校準的參數。

2. 如何有效率的搜尋參數的多組解：

當有多組參數的估算性能輸出  $\hat{y}_m$  和實際性能輸出  $y_m$  相似，這些具有相似性能輸出的參數，都有可能是實際的參數，需以有效率的方法找到所有可能的參數。

3. 如何檢驗參數校準的結果：

即判斷估算性能輸出  $\hat{y}_m$  和實際性能輸出  $y_m$  相似的方法，需訂定估算性能輸出和實際性能輸出可接受的誤差範圍，以評估校準參數結果。

4. 如何提升參數校準的準確率：

需降低  $\hat{p}_m$  和  $p_m$  的誤差，以更精確的掌握系統參數的實際狀況。

本研究將建立藉由性能偏移，校準參數的完整分析流程，並對以上問題提出  
解決方法。





---

## 校準方法概念

---

本章先定義校準實驗的意義，接著說明正向校準參數和逆向校準參數的概念，分析其優缺點和可互補之處，以及如何提升參數校準的效率，最後探討如何以假設檢定的方法，進一步檢驗參數校準的準確性。

若以實際實驗作為校準實驗，進行參數校準會有以下缺點：

1. 需考慮參數以及系統性能的量測誤差：參數及系統性能的量測結果與實際情形不符。
2. 系統性能輸出偏移和系統參數偏移之實際關係不明：參數和系統性能的量測誤差造成難以判斷系統性能偏移的原因。
3. 較高的時間、金錢成本：使用硬體及耗材進行實際實驗需付出額外成本。

因此本研究以物理模型的模擬結果作為虛擬實驗結果，替代實際實驗，並根據虛擬實驗結果，建立校準參數的方法，當虛擬實驗的性能表現不如期望時，將其原因視為參數偏移造成，並校準參數。

### 3.1 校準參數方法一：正向校準

即使以物理模型的虛擬實驗結果取代實際實驗，受限於時間因素，我們仍無法進行無限次虛擬實驗，只能以有限次數的虛擬實驗，估計在不同參數設定下，

對應的虛擬實驗結果。我們以估計值  $\hat{\mathbf{y}}_m$ ，匹配虛擬實驗結果  $\mathbf{y}_m$ ，藉由最小化  $\hat{\mathbf{y}}_m$  和  $\mathbf{y}_m$  之誤差，進行參數校準，本研究中將此方法稱為正向校準參數。

以  $\mathbf{y}_m$  其中一性能輸出  $\mathbf{y}_{m_1}$  進行正向校準為例， $\mathbf{y}_{m_1} = [y_{m_1,1}, y_{m_1,2}, \dots, y_{m_1,n_x}]$ ，由於測試方式  $\mathbf{x}$  皆相同，我們將  $y_{m_1,1}, y_{m_1,2}, \dots, y_{m_1,n_x}$  表示為參數  $\mathbf{p}_m$  的函數， $\mathbf{y}_{m_1}$  和  $\mathbf{p}_m$  的關係如式3.1，藉由找出  $\mathbf{y}_{m_1}$  和  $\mathbf{p}_m$  的關係，也就是估算  $f_{1,1}(\mathbf{p}_m), f_{1,2}(\mathbf{p}_m), \dots, f_{1,n_x}(\mathbf{p}_m)$ ，即可獲得性能輸出估計值  $\hat{y}_{m_1,1}, \hat{y}_{m_1,2}, \dots, \hat{y}_{m_1,n_x}$ 。

$$\begin{aligned} y_{m_1,1} &= f_{1,1}(\mathbf{p}_m) \\ y_{m_1,2} &= f_{1,2}(\mathbf{p}_m) \\ &\vdots \\ y_{m_1,n_x} &= f_{1,n_x}(\mathbf{p}_m) \end{aligned} \quad (3.1)$$

求得性能輸出的估計值  $\hat{\mathbf{y}}_{m_1}$  後，我們計算估計值和虛擬實驗結果的誤差，以評估兩者相似程度。定義性能輸出  $\hat{\mathbf{y}}_{m_1}$  和  $\mathbf{y}_{m_1}$  之誤差  $E_1$  如式3.2

$$E_1 = \sum_{i=1}^{n_x} (\hat{y}_{m_1,i} - y_{m_1,i})^2 \quad (3.2)$$

為了找出有較小性能誤差  $E_1$  對應的  $\hat{\mathbf{p}}_m$ ，我們搜尋許多不同的  $\hat{\mathbf{p}}_m$ ，估算這些  $\hat{\mathbf{p}}_m$  對應的性能輸出  $\hat{\mathbf{y}}_{m_1}$ ，並計算性能輸出誤差  $E_1$  的數值。搜尋許多  $\hat{\mathbf{p}}_m$  後，我們得到多個  $E_1$  的數值，分別對應不同的  $\hat{\mathbf{y}}_{m_1}$ ，假設  $E_1$  的數值分布如圖3.1，較小的  $E_1$  數值代表當參數值為其對應的  $\hat{\mathbf{p}}_m$  時， $\hat{\mathbf{y}}_{m_1}$  和  $\mathbf{y}_{m_1}$  相似，則我們將  $\hat{\mathbf{p}}_m$  視為可能的參數值，以此方法進行正向校準參數。

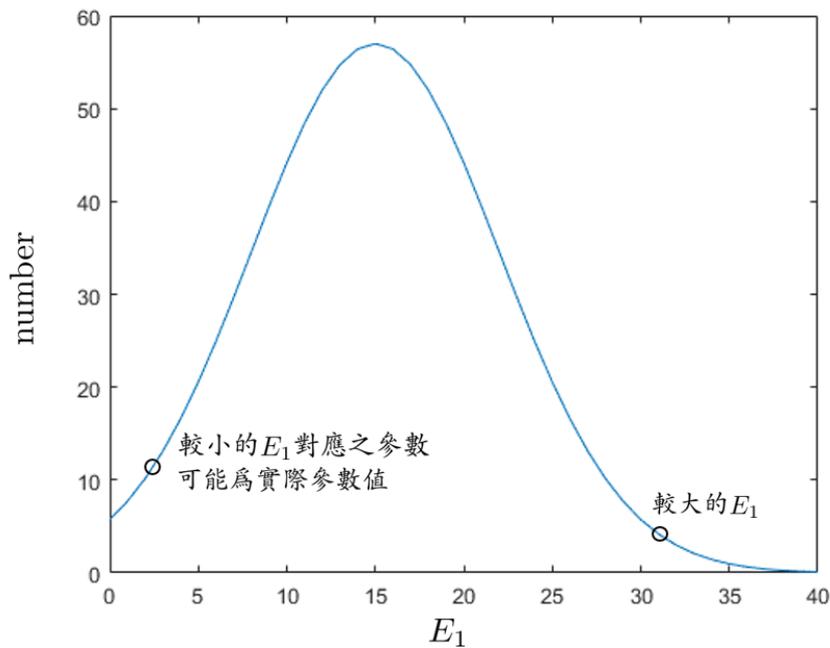


圖 3.1:  $E_1$  分布圖

然而正向校準會面臨的問題是，隨著參數數量  $n_p$  增加， $\mathbf{p}_m$  參數組合數量呈指數成長 [20]，參數組合數量  $n_{\text{set}}$  如式 3.3，其中  $c$  為參數解析度，代表搜尋時不同  $\hat{\mathbf{p}}_m$  的差值，參數的差值越小，解析度越高，可更精確的校準參數，但搜尋的參數也越多，需花費較多計算時間。

$$n_{\text{set}} \propto c^{n_p} \quad (3.3)$$

此外，若有多組參數  $\hat{\mathbf{p}}_m$  可產生相似  $\mathbf{y}_{m1}$  的性能輸出，使用最佳化方法時容易陷入區域最佳解，若想要找出所有可能的  $\hat{\mathbf{p}}_m$ ，在不確定是否有多組解的情形下，需藉由以多個不同起始點進行搜尋，效率差且耗費計算資源，

綜合以上討論，正向校準參數尚有以下問題需要解決，以提升參數校準的效率：

1. 判斷參數是否有多組解：若參數為唯一解，則可不必搜尋多組解。
2. 決定搜尋參數的初始值：以接近實際參數值作為搜尋的初始值。
3. 縮減搜尋參數的範圍：根據搜尋的初始值決定對應的搜尋範圍。



## 3.2 校準參數方法二：逆向校準

為解決正向校準參數的困難之處，我們需提供一可靠之初始值，縮減正向校準的搜尋範圍，並需判斷該系統輸出是否有多組參數解之可能，承前章節之符號，我們令物理模型函數及其輸出為  $\mathbf{y}_m = f_m(\mathbf{p}_m; \mathbf{x})$ ，由於測試方式  $\mathbf{x}$  皆相同，我們藉由求反函數  $\mathbf{p}_m = f^{-1}(\mathbf{y}_m)$  推估參數  $\hat{\mathbf{p}}_m$ ，這種以反函數推估參數  $\hat{\mathbf{p}}_m$  的方法，在本研究中稱為逆向校準方法。

以  $\mathbf{y}_m$  其中一性能輸出  $\mathbf{y}_{m_1}$  進行逆向校準為例，我們將  $\mathbf{p}_m$  表示為式3.4，藉由求得  $f_1^{-1}(\mathbf{y}_{m_1}), f_2^{-1}(\mathbf{y}_{m_1}), \dots, f_{n_p}^{-1}(\mathbf{y}_{m_1})$ ，即可推估  $\hat{\mathbf{p}}_m$ 。

$$\begin{aligned} p_{m_1} &= f_1^{-1}(\mathbf{y}_{m_1}) \\ p_{m_2} &= f_2^{-1}(\mathbf{y}_{m_1}) \\ &\vdots \\ p_{m_{n_p}} &= f_{n_p}^{-1}(\mathbf{y}_{m_1}) \end{aligned} \tag{3.4}$$

由於對應特定性能輸出的參數可能有多組解，無法保證能以逆向校準方法準確求得  $\hat{p}_{m_1}, \hat{p}_{m_2}, \dots, \hat{p}_{m_{n_p}}$ 。但配合正向校準方法，逆向校準參數可解決以下問題：

### 1. 判斷參數是否有多組解

若逆向校準的準確率高，可推測參數應為唯一解，若逆向校準參數準確低，可能原因為：1. 參數有多組解、2. 逆向校準本身誤差

假設一物理模型共有 3 個參數，分別為  $p_{m_1}, p_{m_2}, p_{m_3}$ ，以該物理模型進行一次虛擬實驗，得性能輸出  $\mathbf{y}_{m_1}$ ，以正向校準方法估計  $p_{m_1}, p_{m_2}, p_{m_3}$  在不同設定值下，對應的性能輸出  $\hat{\mathbf{y}}_{m_1}$ ，這些不同  $p_{m_1}, p_{m_2}, p_{m_3}$  的設定值中，有部分可產生和  $\mathbf{y}_{m_1}$  相似的性能輸出，為可能的參數設定值。我們將可能的參數設定值對應的  $p_{m_1}, p_{m_2}, p_{m_3}$  分別畫出分布圖，假設分布圖如圖3.2，可見  $p_{m_1}$  有較小的分布範圍， $p_{m_1}$  可能的數值分布於 0.94 至 0.96， $p_{m_2}, p_{m_3}$  有較大的分布範圍， $p_{m_2}$  可能的數值分布於 1.0 至 1.2， $p_{m_3}$  可能的數值分布於 0.8 至 1.0，由此推論， $p_{m_1}$  應為唯一解， $p_{m_2}, p_{m_3}$  應有多組解。

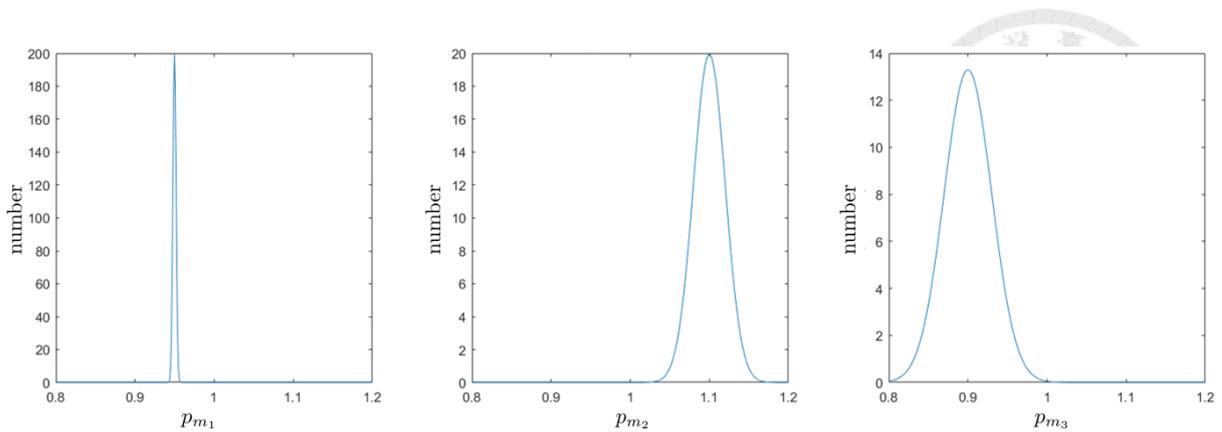


圖 3.2: 參數分布圖

若以逆向校準參數方法能準確校準  $p_{m_1}$ ，可得知  $p_{m_1}$  應為唯一解，若  $p_{m_1}$  逆向校準的準確率低，而以正向校準方式檢驗， $p_{m_1}$  的分布如圖3.2，顯示  $p_{m_1}$  應為唯一解，則可判斷準確率低的原因為逆向校準本身誤差，此時應設法提升逆向校準結果。

而  $p_{m_2}, p_{m_3}$  由於有多組解，將導致逆向校準方法無法準確校準參數，若  $p_{m_2}, p_{m_3}$  逆向校準的準確率低，而以正向校準方式檢驗結果  $p_{m_2}, p_{m_3}$  的分布如圖3.2，顯示  $p_{m_2}, p_{m_3}$  有多組解，則可判斷準確率低原因可能為  $p_{m_2}, p_{m_3}$  有多組解。

## 2. 提供正向校準初始值及搜尋範圍

逆向校準方法藉由反函數推估  $\hat{\mathbf{p}}_m$ ，作為正向校準參數搜尋的初始值，但由於不知道  $\hat{\mathbf{p}}_m$  和  $\mathbf{p}_m$  的誤差大小，正向校準方法的參數搜尋範圍未知。本研究建議依據逆向校準之準確率，決定正向校準時參數  $p_{m_1}, p_{m_2}, \dots, p_{m_{n_p}}$  的搜尋範圍。

$\hat{p}_{m_1} \hat{p}_{m_2} \hat{p}_{m_3}$  假設  $p_{m_1}$  逆向校準準確率高，代表逆向校準  $p_{m_1}$  大多數情形接近真實值，因此將會有較小的搜尋範圍，如圖3.3，假設逆向校準  $p_{m_2}, p_{m_3}$  準確率低，則  $p_{m_2}, p_{m_3}$  會有較大的搜尋範圍，如圖3.4。

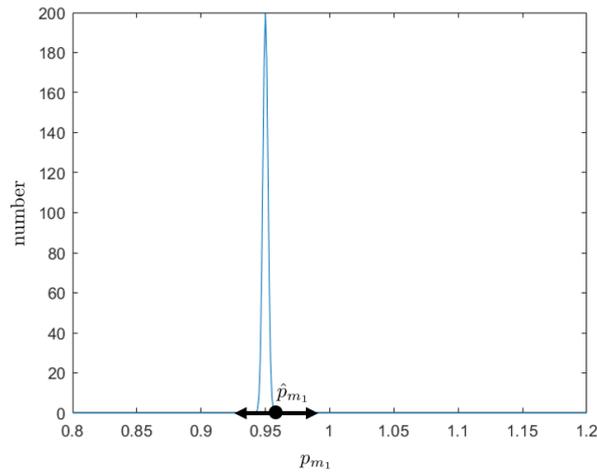


圖 3.3:  $p_{m_1}$  搜尋範圍

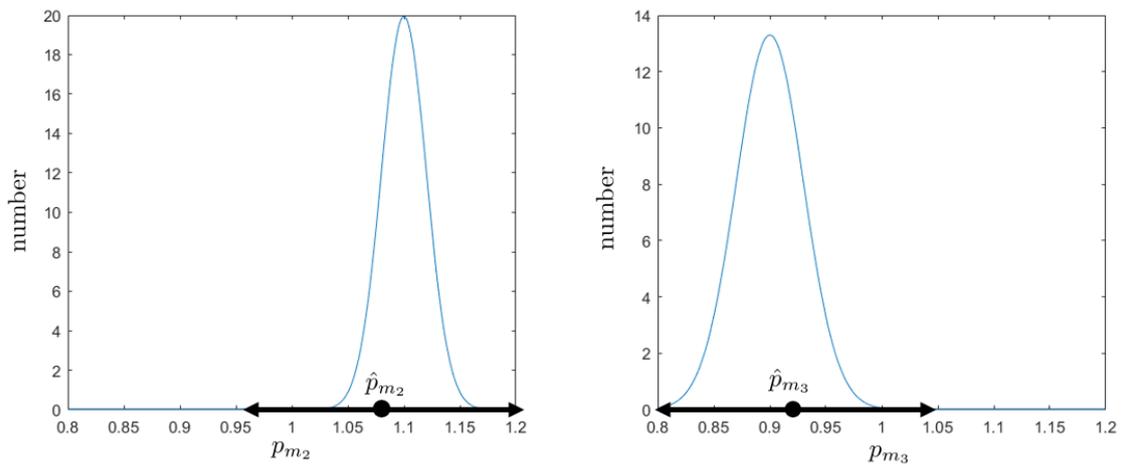


圖 3.4:  $p_{m_2}, p_{m_3}$  搜尋範圍

### 3.3 參數校準檢驗方法

逆向校準雖然可不經由疊代運算，以反函數快速推估參數，但參數是否準確未知，本研究建議配合正向校準方法，以假設檢定判斷是否接受逆向校準的結果。

以  $y_m$  其中一性能輸出  $y_{m_1}$  檢驗逆向校準結果為例，以正向校準方法估計逆向校準所得  $\hat{\mathbf{p}}_m$  對應之性能輸出  $y_{m_1}$ ，若  $\hat{\mathbf{p}}_m \equiv \mathbf{p}_m$ ，則性能輸出誤差  $E_1$  值應為 0，但由於需考量正向校準的性能輸出估計誤差，故性能輸出誤差  $E_1$  不為 0。假設正

向校準的估計誤差為常態分布，我們隨機取樣  $n_s$  組  $\mathbf{p}_m$ ，計算  $n_s$  組性能輸出誤差  $E_1$ ，其平均值為  $\bar{E}_1$ ，標準差為  $s_1$ ，我們希望估算性能輸出和實際性能輸出盡可能相似，也就是較小的性能輸出誤差  $E_1$ ，因此接受性能輸出誤差  $E_1$  位於接受域的  $\hat{\mathbf{p}}_m$ ，為決定接受域大小，首先須決定顯著水準  $\alpha$ ，一般使用  $\alpha=0.05$ ，對應單邊的 95% 信賴區間，常使用的  $\alpha$  及對應的  $z_\alpha$  值如表 3.1，

表 3.1: 單邊信賴區間  $z$  值表

信賴區間	$\alpha$	$z_\alpha$
90%	0.1	1.282
95%	0.05	1.645
99%	0.001	2.326

若  $E_1$  滿足式 3.5，代表性能輸出誤差  $E_1$  位於接受域，信任校準結果，若否則代表  $E_1$  位於拒絕域，不信任校準結果，如圖 3.5。

$$E_1 < \bar{E}_1 + z_\alpha \times s_1 \quad (3.5)$$

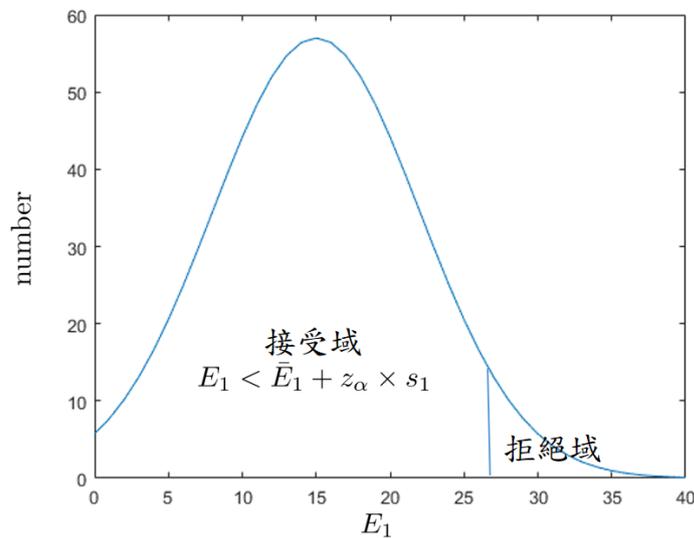


圖 3.5: 信賴區間

若信任逆向校準結果，代表  $\hat{\mathbf{p}}_m$  應相當接近  $\mathbf{p}_m$ ，此時只須搜尋  $\hat{\mathbf{p}}_m$  附近之參數即可，若不信任逆向校準結果，代表  $\hat{\mathbf{p}}_m$  可能和  $\mathbf{p}_m$  誤差較大，則應擴大搜尋範圍，如圖 3.6、3.7、3.8。

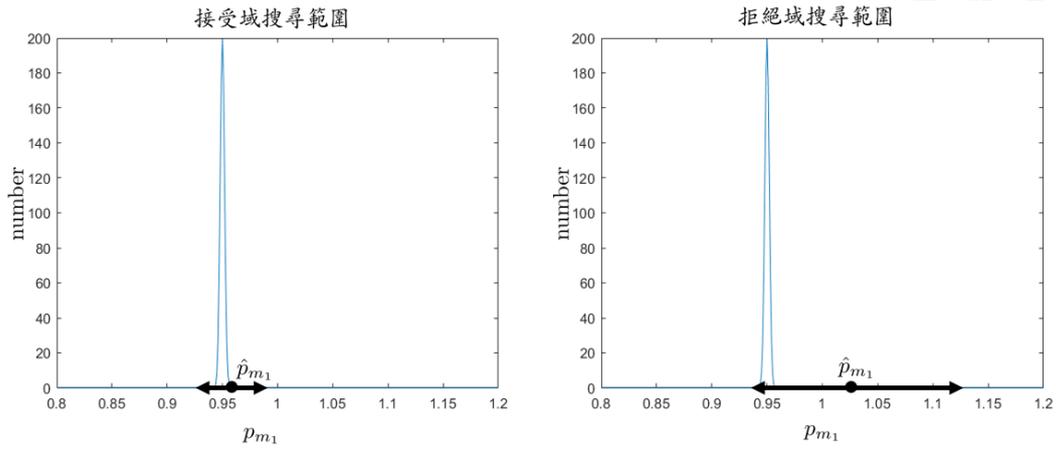


圖 3.6:  $p_{m_1}$  搜尋範圍

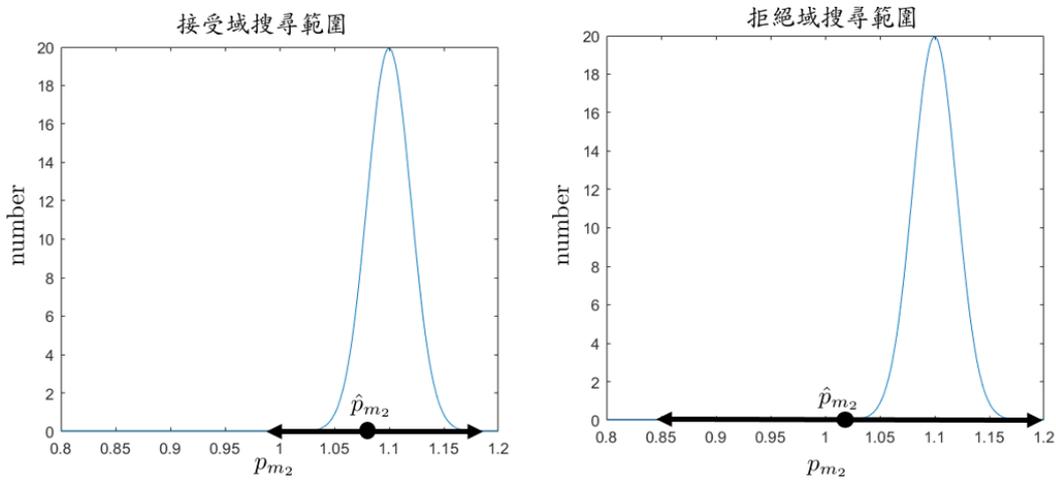


圖 3.7:  $p_{m_2}$  搜尋範圍

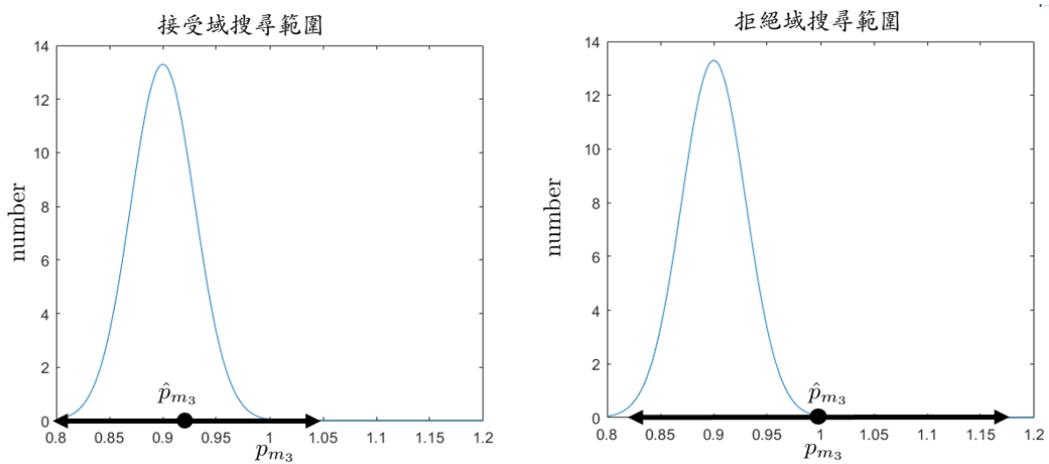


圖 3.8:  $p_{m_3}$  搜尋範圍



## 第四章

---

# 研究方法

---

本章延續第三章內容，結合正、逆向校準參數方法進行參數校準，首先介紹參數校準的整體流程，接著分別說明各步驟的細部執行方法，以及使用多項性能輸出提升校準參數準確率的方法。

## 4.1 方法流程

本研究以神經網路建立正、逆向校準模型，進行參數校準，流程如圖4.1

### 1. 找出重要參數：

評估參數對系統性能之影響，找出對系統性能影響較大的重要參數，忽略其他參數的影響，以降低分析的複雜度，提升參數校準的效率。

### 2. 建立訓練集、驗證集、測試集：

決定重要參數的分布情形，以物理模型進行虛擬實驗，建立訓練集、驗證集、測試集，避免校準模型發生過度擬合的情形，提升參數校準的準確率。

### 3. 建立正向校準模型：

根據訓練集建立正向校準模型，正向校準模型可依據重要參數數值，估計性能輸出，藉由最小化估計性能輸出和實際性能輸出誤差，回推參數數值，並評估校準結果的準確率。



#### 4. 建立逆向校準模型：

根據訓練集建立逆向校準模型，逆向模型可依據性能輸出，直接推估重要參數數值，並提供正向校準參數時參數的初始值和搜尋範圍，提升校準的效率。

#### 5. 校準參數：

根據虛擬實驗結果，以正、逆向校準模型進行參數校準。



圖 4.1: 主流程圖

## 4.2 找出重要參數

同時回推所有參數是艱鉅的任務，本研究以主因素分析找出對性能輸出影響大的重要參數，並忽略其他參數的影響，降低分析複雜系統的困難度，由於物理模型可能有多項性能輸出，本研究建議對在意之性能輸出進行主因素分析，以汽車模型為例，假設性能輸出有扭力、馬達轉速、油耗表現，而我們最在意的是汽車的油耗表現，則對油耗表現進行主因素分析。



### 1. 主因素分析：

主因素分析進行方法為：為參數設定兩水準，分別為高水準和低水準，藉由分析參數在不同水準下性能輸出的表現，找出重要參數。若參數  $p_{m_k}$  無明確資訊，建議設定高水準為  $p_{m_k} \times 110\%$ ，低水準為  $p_{m_k} \times 90\%$ 。

若在意之性能輸出為第  $i$  個性能輸出的第  $j$  項  $y_{m_i,j}$ ，當所有參數為原設定值時，物理模型  $f_m(\mathbf{x}, \mathbf{p}_m)$  模擬結果  $y_{m_i,j}$  之值為  $\bar{y}_{m_i,j}$ ，當參數  $p_{m_k}$  為高水準時， $y_{m_i,j}$  之值為  $y_{m_i,j}^{k+}$ ，參數  $p_{m_k}$  為低水準時， $y_{m_i,j}$  之值為  $y_{m_i,j}^{k-}$ ，如圖4.2。

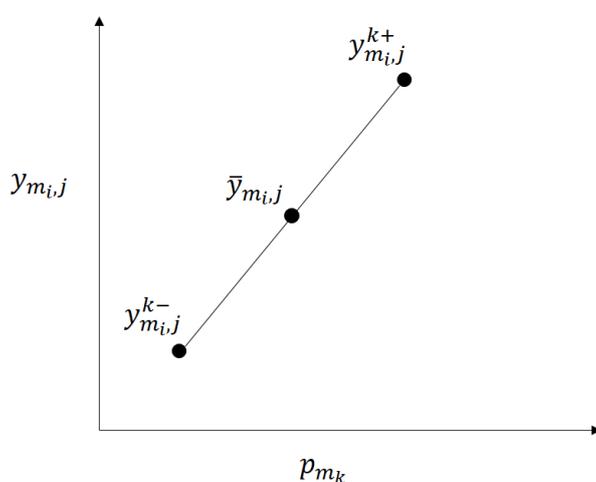


圖 4.2: 主因素分析示意圖 1

依序對所有參數進行主因素分析，如圖4.3，參數  $p_{m_k}$  之主因素為  $y_{m_i,j}^{k+} - y_{m_i,j}^{k-}$ ，主因素絕對值較大值，代表該參數偏移對性能輸出影響較大，為重要參數，最後共從  $n_p$  個參數中選出  $n_m$  個重要參數。

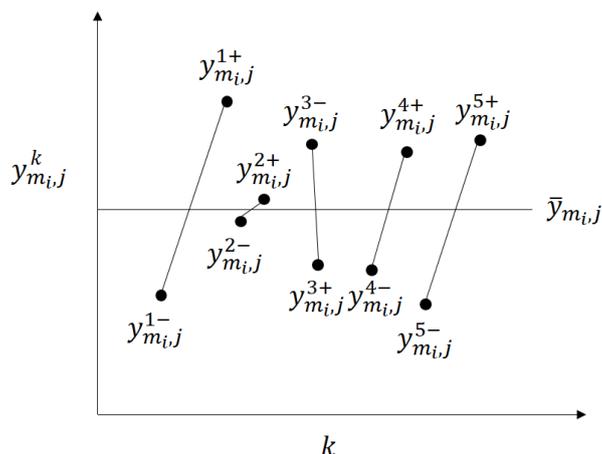


圖 4.3: 主因素分析示意圖 2

## 2. 重要參數分類：

‘使用模型校準以識別複雜系統參數數值之方法’ [20]，建議將重要參數進一步分為：直接量測參數及模型校準參數。

有些重要參數易於量測，易於量測之重要參數可直接量測，不易量測之重要參數則以模型校準，然而，重要參數間的交互作用未知，若忽略易於量測之重要參數的影響，可能造成無法校準不易量測之重要參數，因此本研究建議進行虛擬實驗時，仍需考量易於量測的重要參數偏移對性能輸出之影響，但在校準時可以量測結果替代校準結果，進一步提升校準準確率。

本研究建議仍可使用量測方式獲得直接量測參數數值，但校準參數時，需考慮所有重要參數偏移之影響，並以一數學案例說明原因。

假設一簡單系統，性能輸出  $y_{m_1}$  有 2 個值，分別為  $y_{m_1,1}$  及  $y_{m_1,2}$ ，系統有三個參數，分別為  $p_{m_1}$ 、 $p_{m_2}$ 、 $p_{m_3}$ 。

$y_{m_1,1}$  及  $y_{m_1,2}$  分別為：

$$y_{m_1,1} = 2p_{m_1} + (p_{m_2} + 1) + 0.005p_{m_3} \quad (4.1a)$$

$$y_{m_1,2} = 4p_{m_1} + (p_{m_2} + 1)^2 + 0.005p_{m_3} \quad (4.1b)$$

假設  $p_{m_1}$ 、 $p_{m_2}$ 、 $p_{m_3}$  原設計點為 1.0，受不確定因素影響，最小值為 0.9，最大值為 1.1，分布為均勻分布 (uniform distribution)，如表 4.1



表 4.1: 設計變數分布

設計變數	分布類型	原設計點	最小值	最大值
$p_{m_1}$	均勻分布	1.0	0.9	1.1
$p_{m_2}$	均勻分布	1.0	0.9	1.1
$p_{m_3}$	均勻分布	1.0	0.9	1.1

假設  $y_{m_{1,2}}$  為關心的性能輸出，進行主因素分析後，結果如表 5.3

表 4.2: 主因素分析結果

參數	主因素
$p_{m_1}$	0.805
$p_{m_2}$	0.805
$p_{m_3}$	0.005

假設  $p_{m_2}$  易於量測，而  $p_{m_3}$  之影響可以忽略，依據 [20] 之作法則僅需校準  $p_{m_1}$ ，將  $p_{m_1}$  設定為最小值 0.9，最大值 1.1 之均勻分布， $p_{m_2}$ 、 $p_{m_3}$  設定為原設計點 1，隨機產生 50 筆資料建立逆向校準模型，另將  $p_{m_1}$ 、 $p_{m_2}$ 、 $p_{m_3}$  設定為最小值 0.9，最大值 1.1 之均勻分布，隨機產生 10 筆資料， $\hat{p}_{m_1}$  之值。

校準模型預測結果如表 4.3，

表 4.3: 校準模型預測結果 (MSE=0.0414)

資料編號	$p_{m_1}$	$\hat{p}_{m_1}$
1	0.9222	0.9001
2	1.0472	1.0092
3	1.0530	1.0931
4	0.9320	0.9470
5	1.0008	1.0265
6	1.0695	1.0194
7	1.0538	1.1177
8	0.9684	0.8991
9	0.9874	1.0242
10	1.0416	1.0403

雖然  $p_{m_2}$  可直接量測，但結果顯示若不考慮  $p_{m_2}$  之變動，則無法準確  $\hat{p}_{m_1}$ 。

現在將  $p_{m_1}$ 、 $p_{m_2}$  設定為最小值 0.9，最大值 1.1 之均勻分布， $p_{m_3}$  設定為原設計點 1 隨機產生 50 筆資料建立逆向校準模型， $\hat{p}_{m_1}$  之值。

預測結果如表 4.4，可準確  $\hat{p}_{m_1}$ ，結果顯示即使忽略  $p_{m_3}$  變動之影響，仍可準確  $\hat{p}_{m_1}$ ，但若忽略重要參數  $p_{m_2}$  之影響，則無法準確  $\hat{p}_{m_1}$ ，因此應考量所有重要參數偏移造成性能輸出之影響。

表 4.4: 校準模型預測結果 (MSE=0.0023)

資料編號	$p_{m_1}$	$\hat{p}_{m_1}$
1	0.9222	0.9212
2	1.0472	1.0465
3	1.0530	1.0561
4	0.9320	0.9316
5	1.0008	1.0045
6	1.0695	1.0664
7	1.0538	1.0505
8	0.9684	0.9679
9	0.9874	0.9866
10	1.0416	1.0390

### 4.3 建立訓練集、驗證集、測試集

為避免校準模型過度擬合訓練集資料，需使用驗證集資料檢驗校準模型準確率，並對校準模型做出調整，最後使用模型完全測試集資料評估模型預測未見資料的準確率，以提升校準參數的準確率，建立訓練集、驗證集、測試集方法如下。

#### 1. 設定重要參數分布情形：

建立訓練集時，若只能進行較少的實驗次數，建議以直交表設定重要參數分布情形，若允許較多次的實驗次數，建議以蒙地卡羅法隨機產生重要參數分布情形，而建立驗證集及測試集時皆使用蒙地卡羅法隨機產生重要參數分布情形。

#### 2. 建立訓練集、驗證集、測試集：

依據重要參數分布情形，將重要參數分別設定為  $\mathbf{p}_m^1, \mathbf{p}_m^2, \dots, \mathbf{p}_m^{n_{\text{train}}}$ ，以物理模型  $f_m(\mathbf{p}_m; \mathbf{x})$  進行  $n_{\text{train}}$  次虛擬實驗，產生  $n_{\text{train}}$  組實驗數據，作為訓練集。



每組虛擬實驗數據共有  $n_y$  個性能輸出。

第 1 組為： $\mathbf{y}_{m_1}^1, \mathbf{y}_{m_2}^1, \dots, \mathbf{y}_{m_{n_y}}^1$

第 2 組為： $\mathbf{y}_{m_1}^2, \mathbf{y}_{m_2}^2, \dots, \mathbf{y}_{m_{n_y}}^2$

第  $n_{\text{train}}$  組為： $\mathbf{y}_{m_1}^{n_{\text{train}}}, \mathbf{y}_{m_2}^{n_{\text{train}}}, \dots, \mathbf{y}_{m_{n_y}}^{n_{\text{train}}}$

接下來分別進行  $n_{\text{val}}$  次虛擬實驗，作為驗證集，及  $n_{\text{test}}$  次虛擬實驗，做為測試集。 $n_{\text{train}}, n_{\text{val}}, n_{\text{test}}$  建議數量為 2 : 1 : 1。

### 3. 模型預測能力評估：

- 訓練集：

根據訓練集的資料，以神經網路建立預測模型，由於神經網路能產生複雜的模型，因此容易發生過擬合的現象，過擬合的原因為在訓練過程中，模型過度擬合訓練集的資料，過擬合造成模型對於訓練資料的預測能力表現良好，但對於未見過的資料預測能力表現得很差，而對未見過資料的預測能力稱為模型的泛化能力，

- 驗證集：

為了避免過擬合的現象，使用驗證集檢驗模型的泛化能力，並做出調整是必要的。雖然驗證集的資料並未用來訓練模型，但利用驗證集檢驗模型泛化能力，做出相應調整，使驗證集和訓練集的預測準確率更為接近時，亦有可能對驗證集產生些微的過擬合現象。

- 測試集：

為了客觀的評估模型泛化能力，最後會使用測試集進行檢驗，測試集在訓練模型以及調整模型的過程中皆未曾使用，僅用於評估最終模型的泛化能力。一般而言，雖然藉由調整模型，使驗證集和訓練集的預測準確率更為接近，相較於訓練集的準確率，驗證集的準確率仍然較差，而測試集之準確率又略遜驗證集的準確率，但差距通常會小於驗證集和測試集的差距，若模型的泛化能力佳，驗證集的準確率亦有可能高於測試集。



## 4.4 建立正向校準模型

正向校準模型可推估參數對應的性能輸出，藉由尋找性能輸出和實際性能輸出相似的參數，達成校準參數的目的，若有多個參數性能輸出和實際性能輸出相似，也可使用正向校準模型找出所有可能的參數組合。

### 1. 建立模型：

由於以物理模型進行虛擬實驗需花費較久時間，本研究以神經網路建立正向校準模型，快速推估物理模型  $y_{m_1}, y_{m_2}, \dots, y_{m_{n_y}}$  等性能輸出。

以推估性能輸出  $y_{m_1}$  為例，以  $\mathbf{p}_m$  作為輸入層， $y_{m_1,1}$  作為輸出層，神經網路架構如圖4.4

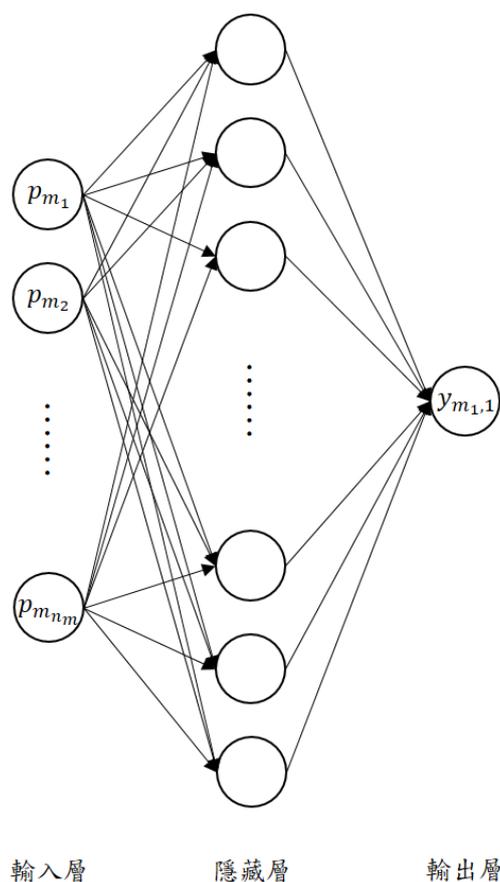


圖 4.4: 神經網路架構示意圖 1

接著分別以  $y_{m_1,2}, \dots, y_{m_1,n_x}$  作為輸出層，以神經網路建立  $n_x$  個正向校準模

型。



$$\hat{y}_{m1,1} = \hat{f}_{1,1}^{fwd}(\mathbf{p}_m) \quad (4.2a)$$

$$\hat{y}_{m1,2} = \hat{f}_{1,2}^{fwd}(\mathbf{p}_m) \quad (4.2b)$$

$$\hat{y}_{m1,n_x} = \hat{f}_{1,n_x}^{fwd}(\mathbf{p}_m) \quad (4.2c)$$

仍以  $\mathbf{p}_m$  作為輸入層，對  $\mathbf{y}_{m2}, \dots, \mathbf{y}_{m_{n_y}}$  以相同作法建立正向校準模型：

$$\hat{y}_{m2,1} = \hat{f}_{2,1}^{fwd}(\mathbf{p}_m) \quad (4.3a)$$

$$\hat{y}_{m2,2} = \hat{f}_{2,2}^{fwd}(\mathbf{p}_m) \quad (4.3b)$$

$$\hat{y}_{m2,n_x} = \hat{f}_{2,n_x}^{fwd}(\mathbf{p}_m) \quad (4.3c)$$

$$\hat{y}_{m_{n_y},1} = \hat{f}_{n_y,1}^{fwd}(\mathbf{p}_m) \quad (4.4a)$$

$$\hat{y}_{m_{n_y},2} = \hat{f}_{n_y,2}^{fwd}(\mathbf{p}_m) \quad (4.4b)$$

$$\hat{y}_{m_{n_y},n_x} = \hat{f}_{n_y,n_x}^{fwd}(\mathbf{p}_m) \quad (4.4c)$$

最後共建立  $n_x \times n_y$  個正向校準模型。

## 2. 保留預測準確率高的模型：

由於建立之模型不一定皆能準確推估性能輸出，我們以驗證集檢驗模型，只保留可準確推估性能輸出的正向校準模型，如圖4.5。

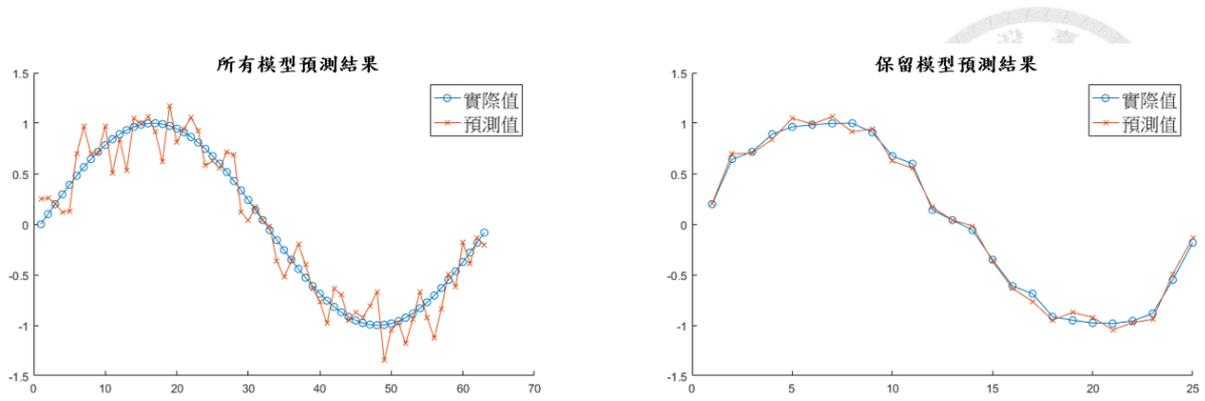


圖 4.5: 預測結果示意圖

為比較模型的準確率，需先將模型預測結果的誤差標準化，定義模型  $\hat{f}_{i,j}^{fwd}(\mathbf{p}_m)$  的標準化常數為  $R_{i,j}$ ，由於訓練集資料筆數較多，以訓練集資料計算之  $R_{i,j}$  較具代表性，因此根據訓練集  $n_{\text{train}}$  筆資料計算  $R_{i,j}$  的數值，如式4.5，而模型  $\hat{f}_{i,j}^{fwd}(\mathbf{p}_m)$  在驗證集之準確率，則根據驗證集  $n_{\text{val}}$  資料計算，如式4.6。

$$R_{i,j} = \sum_{r=1}^{n_{\text{train}}} (y_{m_{i,j}}^r - \bar{y}_{m_{i,j}})^2 / n_{\text{train}} \quad (4.5)$$

$$1 - \frac{\sum_{r=1}^{n_{\text{val}}} (y_{m_{i,j}}^r - \hat{y}_{m_{i,j}}^r)^2 / n_{\text{val}}}{R_{i,j}} \quad (4.6)$$

最後保留  $n_f$  個正向校準模型，分別推估  $n_f$  個性能輸出數值，並將  $n_f$  個性能輸出數值改寫為式4.7，各性能輸出對應的標準化常數改寫為式4.8。

$$\hat{\mathbf{y}}_{\text{set}} = \begin{bmatrix} \hat{y}_{\text{set},1} & \hat{y}_{\text{set},2} & \cdots & \hat{y}_{\text{set},n_f} \end{bmatrix}_{1 \times n_f} \quad (4.7)$$

$$\mathbf{R}_{\text{set}} = \begin{bmatrix} R_{\text{set},1} & R_{\text{set},2} & \cdots & R_{\text{set},n_f} \end{bmatrix}_{1 \times n_f} \quad (4.8)$$

藉由  $n_f$  個準確率高的正向校準模型，可預測在特定參數設定下的性能輸出，並計算和實際性能輸出的誤差，以性能輸出誤差進行參數校準，定義性能輸

出誤差為  $E_{\text{set}}$ ，可表示為式4.9。

$$E_{\text{set}} = \sum_{i=1}^{n_f} \frac{(y_{\text{set},i} - \hat{y}_{\text{set},i})^2}{R_{\text{set},i}} \quad (4.9)$$



### 3. 建立接受域與拒絕域

計算驗證集  $n_{\text{val}}$  筆資料， $E_{\text{set}}$  的平均值  $\bar{E}_{\text{set}}$  及標準差  $s_{\text{set}}$ ，首先決定顯著水準  $\alpha$ ，一般使用  $\alpha=0.05$ ，對應單邊的 95% 信賴區間，建立  $\bar{E}_{\text{set}}$  的接受域與拒絕域，若  $E_{\text{set}}$  滿足式4.10，代表性能輸出誤差  $E_{\text{set}}$  位於接受域，信任校準結果，若否則代表  $E_{\text{set}}$  位於拒絕域，不信任校準結果，如圖4.6。

$$E_{\text{set}} < \bar{E}_{\text{set}} + z_{\alpha} \times s_{\text{set}} \quad (4.10)$$

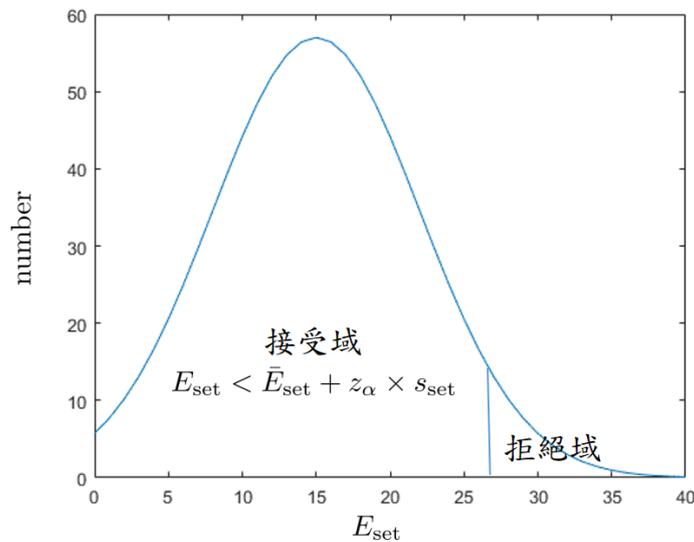


圖 4.6:  $E_{\text{set}}$  接受域和拒絕域

## 4.5 建立逆向校準模型

逆向校準模型可根據實際性能輸出，直接推估參數數值，若有多個參數性能輸出和實際性能輸出相似，雖然逆向校準模型無法找到所有可能參數，但可提供正向校準模型參數搜尋的初始值和搜尋範圍，提升校準參數的效率。



1. 建立模型：

和建立正向校準模型一樣，以神經網路建立逆向校準模型，但輸入層和輸出層有所改變，以  $\mathbf{y}_{m_1}$  校準  $p_{m_1}$  為例，以  $\mathbf{y}_{m_1}$  作為輸入層， $p_{m_1}$  做為輸出層，神經網路架構如圖4.7

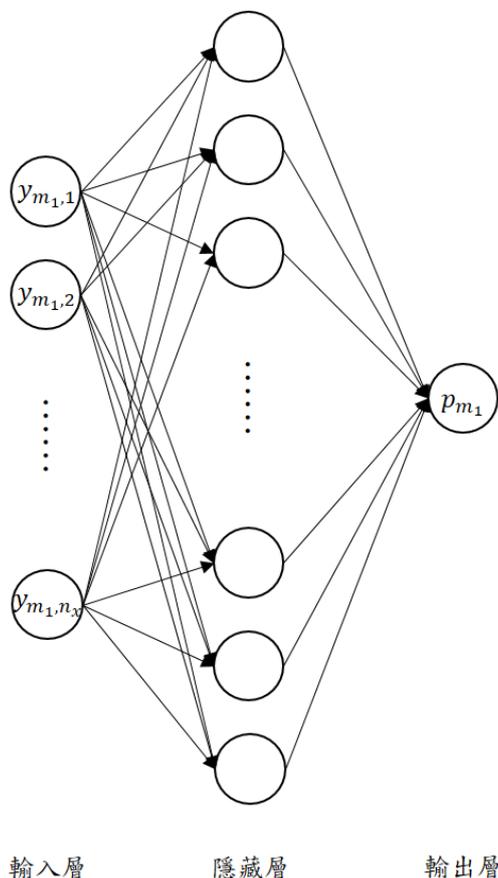


圖 4.7: 神經網路架構示意圖 2

接著分別以  $p_{m_2}, \dots, p_{m_{n_m}}$  做為輸出層，以神經網路建立  $n_m$  個逆向校準模型。

$$\hat{p}_{m_1} = \hat{f}_{1,1}^{inv}(\mathbf{y}_{m_1}) \quad (4.11a)$$

$$\hat{p}_{m_2} = \hat{f}_{1,2}^{inv}(\mathbf{y}_{m_1}) \quad (4.11b)$$

$$\hat{p}_{m_{n_m}} = \hat{f}_{1,n_m}^{inv}(\mathbf{y}_{m_1}) \quad (4.11c)$$

再以  $\mathbf{y}_{m_2}, \mathbf{y}_{m_3}, \dots, \mathbf{y}_{m_{n_y}}$  作為輸入層，分別以  $p_{m_1}, p_{m_2}, \dots, p_{m_{n_m}}$  做為輸出層，用相同的作法建立逆向校準模型，最後共建立  $n_m \times n_y$  個逆向校準模型。



$$\hat{p}_{m_1} = \hat{f}_{2,1}^{inv}(\mathbf{y}_{m_2}) \quad (4.12a)$$

$$\hat{p}_{m_2} = \hat{f}_{2,2}^{inv}(\mathbf{y}_{m_2}) \quad (4.12b)$$

$$\hat{p}_{m_{n_m}} = \hat{f}_{2,n_m}^{inv}(\mathbf{y}_{m_2}) \quad (4.12c)$$

$$\hat{p}_{m_1} = \hat{f}_{n_y,1}^{inv}(\mathbf{y}_{m_{n_y}}) \quad (4.13a)$$

$$\hat{p}_{m_2} = \hat{f}_{n_y,2}^{inv}(\mathbf{y}_{m_{n_y}}) \quad (4.13b)$$

$$\hat{p}_{m_{n_m}} = \hat{f}_{n_y,n_m}^{inv}(\mathbf{y}_{m_{n_y}}) \quad (4.13c)$$

## 2. 建立決策樹模型：

建立之逆向校準模型，在不同參數區間可能有不同準確率，假設  $\hat{f}_{1,1}^{inv}(\mathbf{y}_{m_1})$  在  $p_{m_1} > 1$  時有較高的準確率， $\hat{f}_{2,1}^{inv}(\mathbf{y}_{m_2})$  在  $p_{m_1} < 1$  時有較高的準確率，如表4.5，故選擇適合的模型可提升參數校準的準確率。

表 4.5: 設計變數分布

	$\hat{f}_{1,1}^{inv}(\mathbf{y}_{m_1})$	$\hat{f}_{2,1}^{inv}(\mathbf{y}_{m_2})$
$p_{m_1} > 1$	高準確率	低準確率
$p_{m_1} < 1$	低準確率	高準確率

本研究根據各逆向校準模型結果，以決策樹模型判斷逆向校準模型準確的區間，進一步提升校準準確率，一共建立  $n_m$  個決策樹模型。

$$\hat{p}_{m_1} = \hat{f}_1^{dt}(\hat{\mathbf{p}}_{\text{set}}) \quad (4.14a)$$

$$\hat{p}_{m_2} = \hat{f}_2^{dt}(\hat{\mathbf{p}}_{\text{set}}) \quad (4.14b)$$

$$\hat{p}_{m_{n_m}} = \hat{f}_{n_m}^{dt}(\hat{\mathbf{p}}_{\text{set}}) \quad (4.14c)$$



## 4.6 提升參數校準準確率

為了提升參數校準準確率，本研究建議使用三種方法提升參數校準準確率：

1. 使用更多訓練資料
2. 使用不同性能輸出
3. 使用更多測試方法

### 1. 取得更多訓練：

對參數進行更多次取樣並進行虛擬實驗，使訓練集資料更充足，以提升參數校準的準確率。以  $y_{m_1}$  校準參數為例，其學習曲線如圖4.8，隨著訓練資料數量增加，雖然預測訓練集參數的準確率略為下降，但預測驗證集的參數準確率從 0.7 上升至 0.78，可見取得更多訓練資料可提升參數校準的準確率。

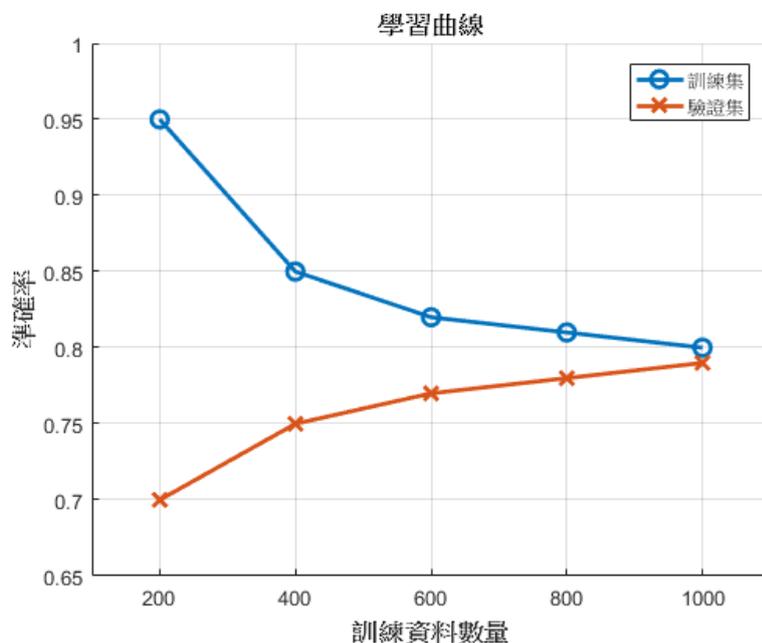


圖 4.8: 使用更多訓練資料結果示意圖

### 2. 使用不同性能輸出：

不同性能輸出對於參數偏移的影響不同，選擇適當的性能校準偏移的參數也是一個方法，在不同參數設定下進行虛擬實驗，性能輸出可能相似，造成難

以校準參數偏移量，如圖4.9， $y_{m_1}^1, y_{m_1}^2$  相似度高，難以分辨差異，造成參數校準困難

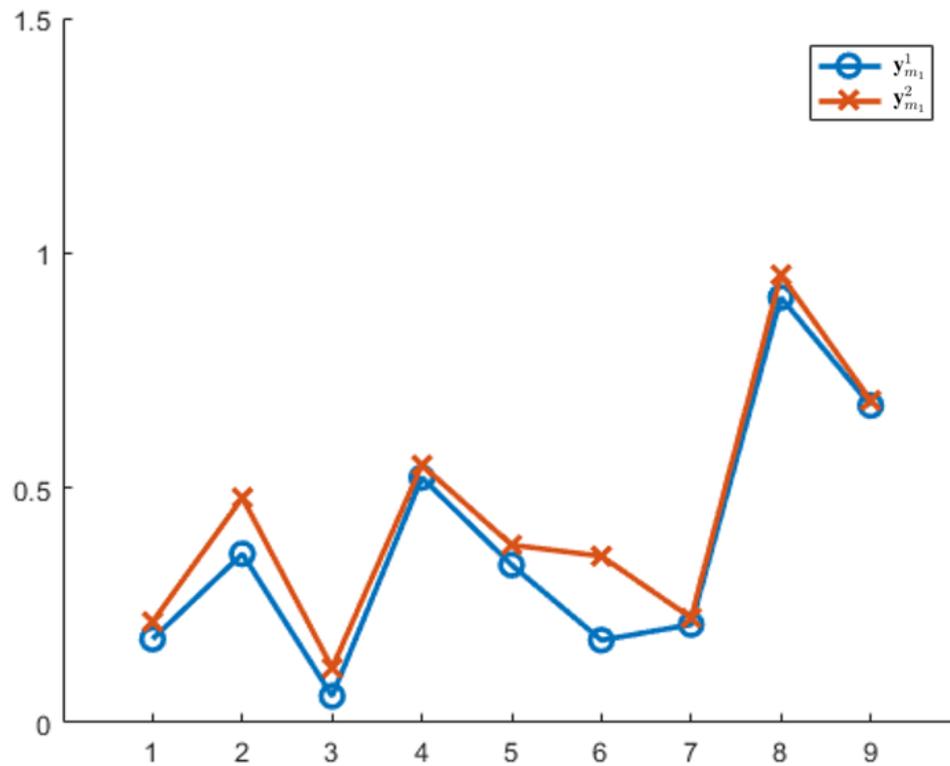


圖 4.9: 差異性小的性能輸出示意圖

此時建議以不同性能輸出校準參數，如圖4.10， $y_{m_2}^1, y_{m_2}^2$  相似度低，較易分辨差異，有利於校準參數。

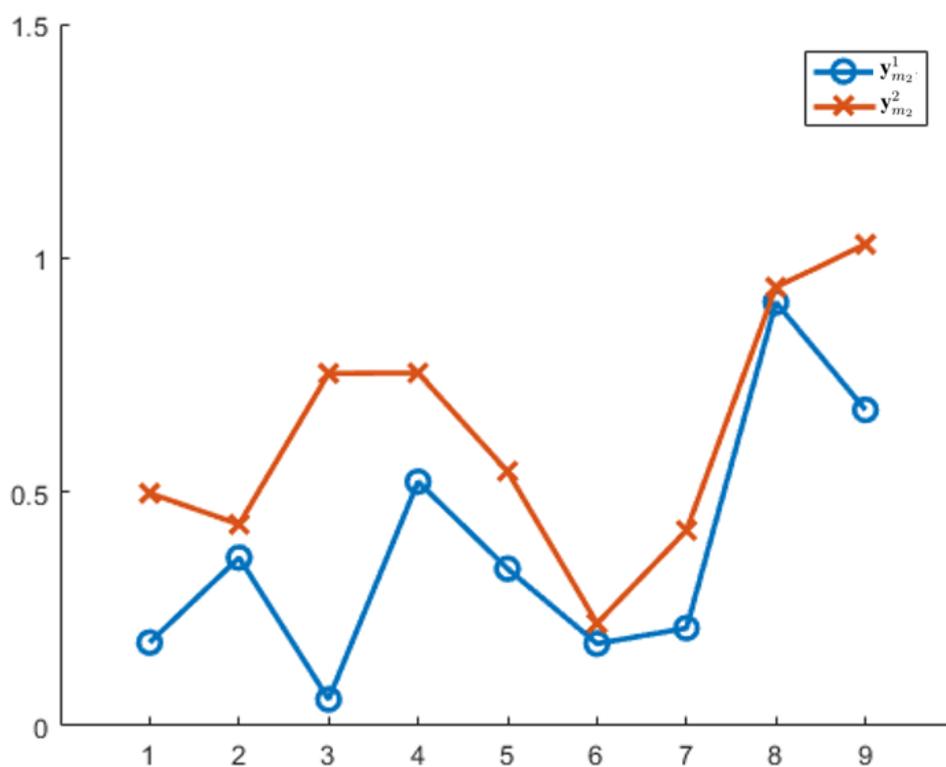


圖 4.10: 差異性大的性能輸出示意圖

現選擇以  $y_{m_2}$  校準參數，其學習曲線如圖4.11，可見相較於圖4.8，以  $y_{m_2}$  校準驗證集的參數有較高的準確率，從 0.78 上升至 0.85。

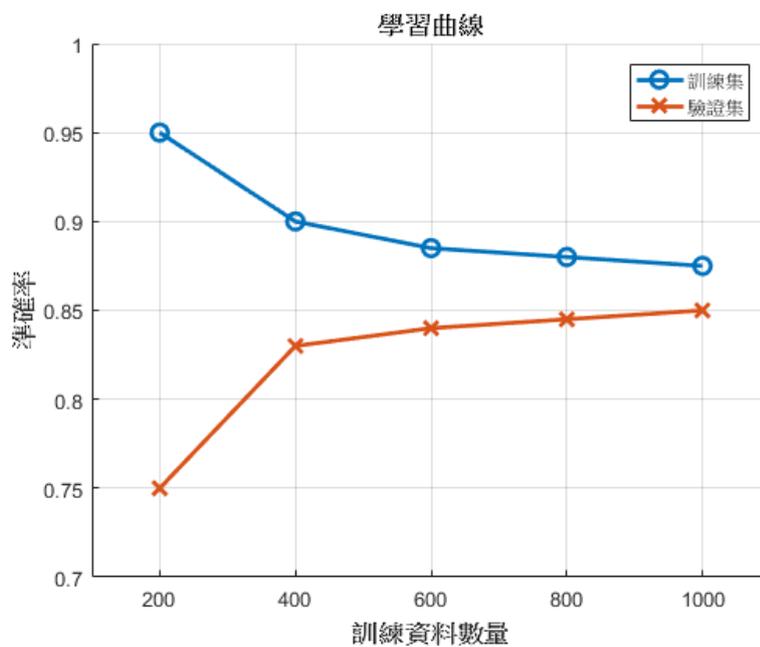


圖 4.11: 使用不同性能輸出結果示意圖

### 3. 使用更多測試方法：

亦可藉由進行更多測試提高參數校準準確率，系統參數相同的偏移，對於不同的測試方法，可能造成不同的性能偏移幅度。同樣使用  $y_{m_2}$  校準參數，但使用更多測試方式，可見圖4.12相較於圖4.10，更容易分辨性能輸出  $y_{m_2}$  的差異，有利提升參數校準的準確性。

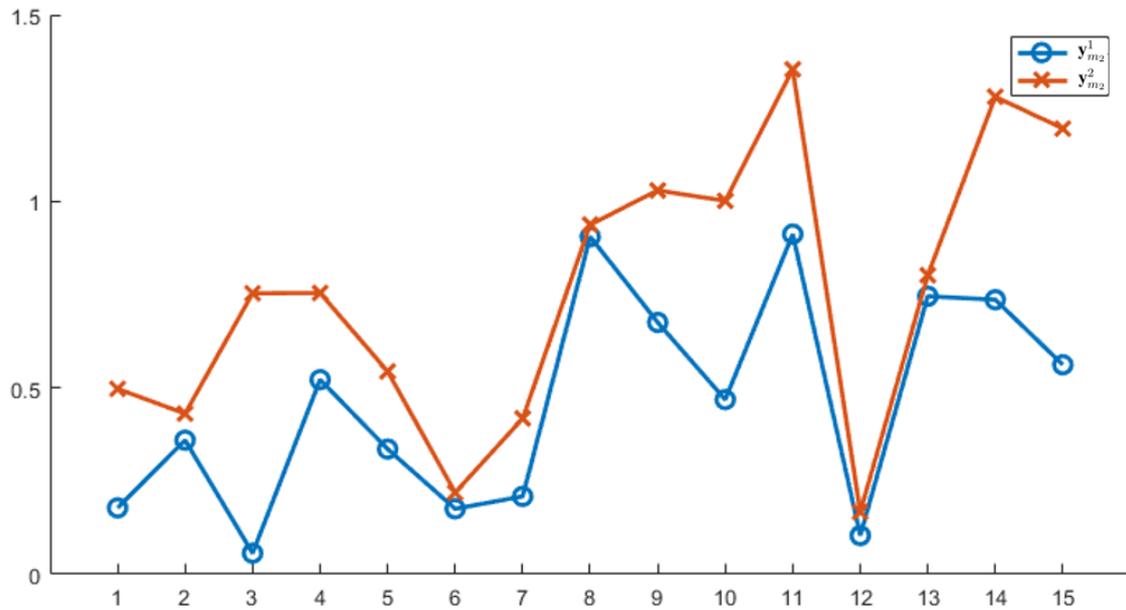


圖 4.12: 使用更多測試方法示意圖

因此使用更多的測試方法，有可能提升校準參數的準確率，如圖4.13，隨著增加測試方法，預測驗證集參數的準確率有所提升。

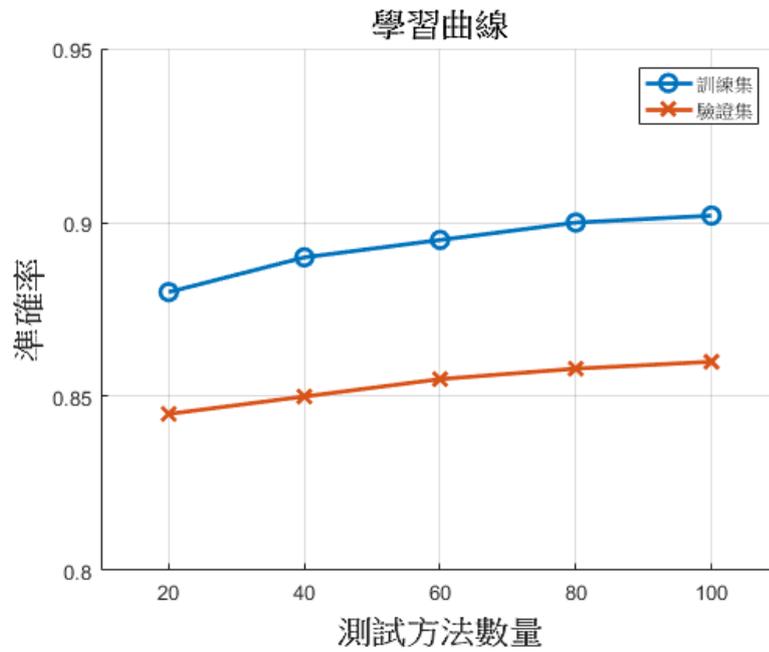


圖 4.13: 使用更多測試方法結果示意圖

## 4.7 校準參數

建立完正、逆向校準模型後，即可校準參數，校準流程如圖4.14。

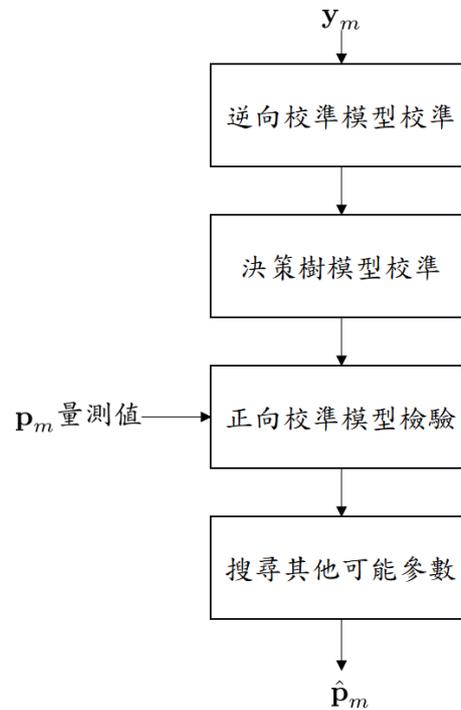


圖 4.14: 校準參數流程



1. 逆向校準模型校準：

根據性能輸出  $y_m$ ，以逆向校準模型校準參數，得  $\hat{\mathbf{p}}_{\text{set}}$ 。

2. 決策樹模型校準：

根據逆向校準模型結果  $\hat{\mathbf{p}}_{\text{set}}$ ，以決策樹模型再次校準參數得  $\hat{\mathbf{p}}_m$ ，並定義驗證集中校準  $p_{m_1}, p_{m_2}, \dots, p_{m_{n_m}}$  誤差的標準差為  $\varepsilon_{s_1}, \varepsilon_{s_2}, \dots, \varepsilon_{s_{n_m}}$ ；校準  $p_{m_1}, p_{m_2}, \dots, p_{m_{n_m}}$  誤差的最大值為  $\varepsilon_{\max_1}, \varepsilon_{\max_2}, \varepsilon_{\max_{n_m}}$ 。

3. 正向校準模型檢驗

根據決策樹模型校準結果  $\hat{\mathbf{p}}_m$ ，以正向校準模型估計對應之性能輸出  $\hat{y}_{\text{set}}$ ，若有參數可準確量測，可用參數量測結果取代校準結果，並計算  $\hat{y}_{\text{set}}$  和  $y_{\text{set}}$  之誤差  $E_{\text{set}}$ ，評估是否接受校準結果  $\hat{\mathbf{p}}_m$ 。

4. 搜尋其他可能參數

若信任校準結果，則在搜尋範圍內尋找位於接受域的參數，搜尋範圍根據兩倍校準誤差的標準差  $\varepsilon_{s_1}, \varepsilon_{s_2}, \dots, \varepsilon_{s_{n_m}}$  如式4.15

$$\begin{aligned}
 & [\hat{p}_{m_1} - 2\varepsilon_{s_1}, \hat{p}_{m_1} + 2\varepsilon_{s_1}] \\
 & [\hat{p}_{m_2} - 2\varepsilon_{s_2}, \hat{p}_{m_2} + 2\varepsilon_{s_2}] \\
 & \vdots \\
 & [\hat{p}_{m_{n_m}} - 2\varepsilon_{s_{n_m}}, \hat{p}_{m_{n_m}} + 2\varepsilon_{s_{n_m}}]
 \end{aligned} \tag{4.15}$$

若若不接受  $\hat{\mathbf{p}}_m$ ，則不信任校準結果，在較大的搜尋範圍內尋找位於接受域的參數，搜尋範圍根據校準誤差的最大值  $\varepsilon_{\max_1}, \varepsilon_{\max_2}, \varepsilon_{\max_{n_m}}$  如式4.16

$$\begin{aligned}
 & [\hat{p}_{m_1} - \varepsilon_{\max_1}, \hat{p}_{m_1} + \varepsilon_{\max_1}] \\
 & [\hat{p}_{m_2} - \varepsilon_{\max_2}, \hat{p}_{m_2} + \varepsilon_{\max_2}] \\
 & \vdots \\
 & [\hat{p}_{m_{n_m}} - \varepsilon_{\max_{n_m}}, \hat{p}_{m_{n_m}} + \varepsilon_{\max_{n_m}}]
 \end{aligned} \tag{4.16}$$

## 案例探討

本章將以一車輛工程案例，執行第四章的研究方法，從以主因素分析找出重要參數、以物理模型進行虛擬實驗，至以性能輸出進行參數校準。

### 5.1 車輛工程案例

本研究在 Matlab 環境下，以一車輛模型模擬車輛動態，輸入為測試方式及車輛參數，以 Matlab Simulink 計算車輛的性能輸出，如圖5.1。



圖 5.1: Matlab Simulink 示意圖

車輛模型為一油電混合車，如圖5.2，車輛平時以電池驅動，以前軸馬達和後軸馬達行駛，且當車輛煞車時，可藉由煞車回充為電池充電。當電力不足時會以引擎使用燃料驅動前軸輔助馬達行駛。

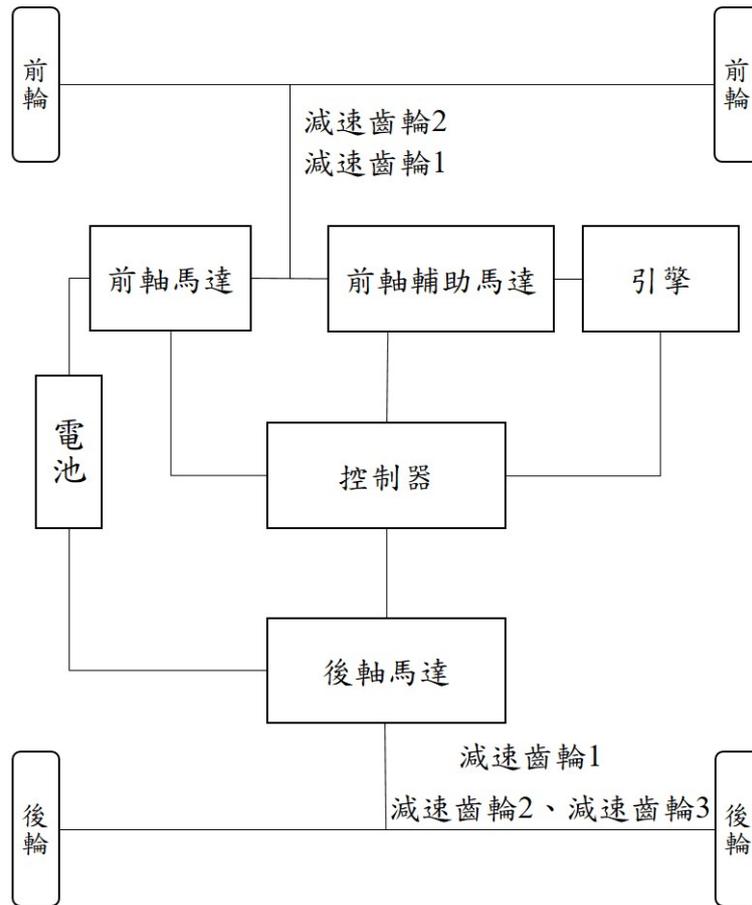


圖 5.2: 車輛模型示意圖

車輛性能輸出計算流程如圖5.3，首先根據車輛動態，計算車輛所需動力，如圖5.4，計算完所需車輛動力後，以控制器進行動力分配如圖5.5，根據動力分配結果，計算對應的馬達扭矩，馬達耗能、車輛實際速度等性能輸出，如圖5.6。

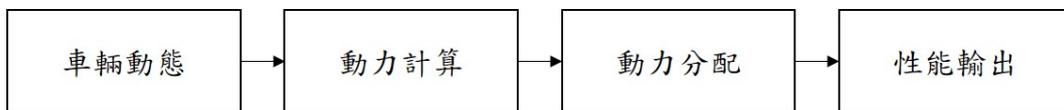


圖 5.3: 性能輸出計算流程

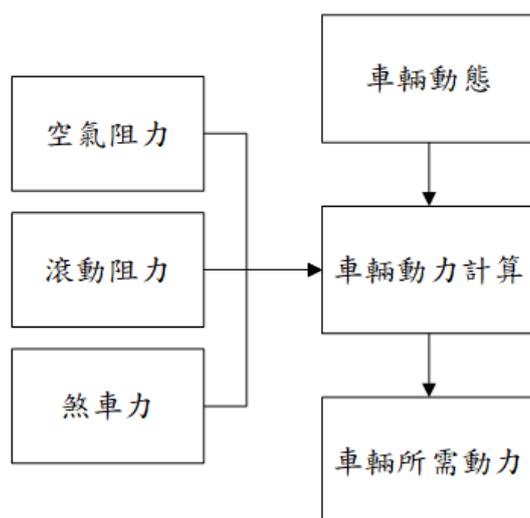


圖 5.4: 動力計算

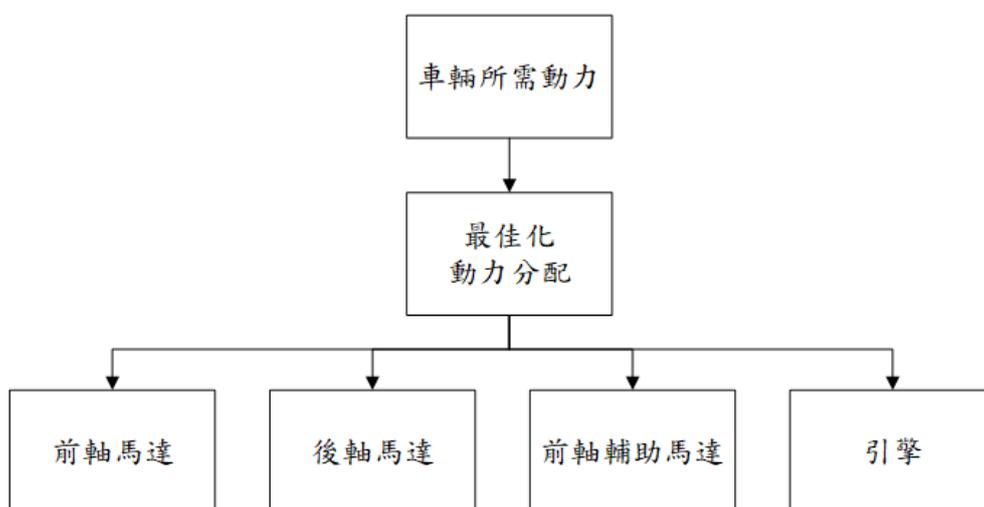


圖 5.5: 動力分配

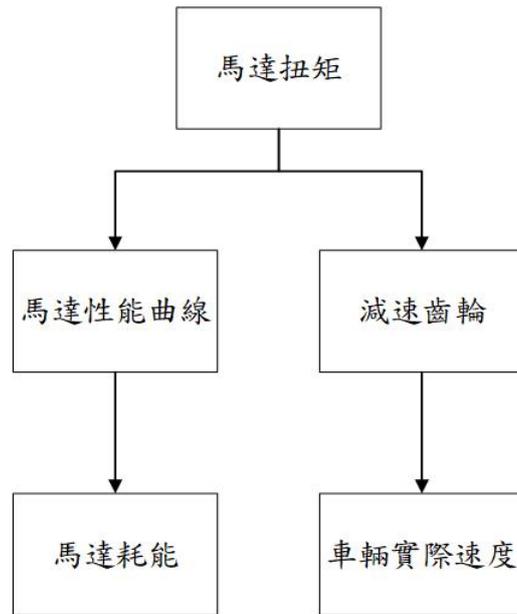


圖 5.6: 性能輸出計算

### 5.1.1 找出重要參數

New European Driving Cycle(NEDC Cycle) 為一常見測試車輛耗能的方式，測試方式如圖5.7，可見 0 秒至 800 秒的測試方式相同的，為加快模擬的速度，本研究將重複的測試方式去除，簡化後的測試方式如圖5.8。

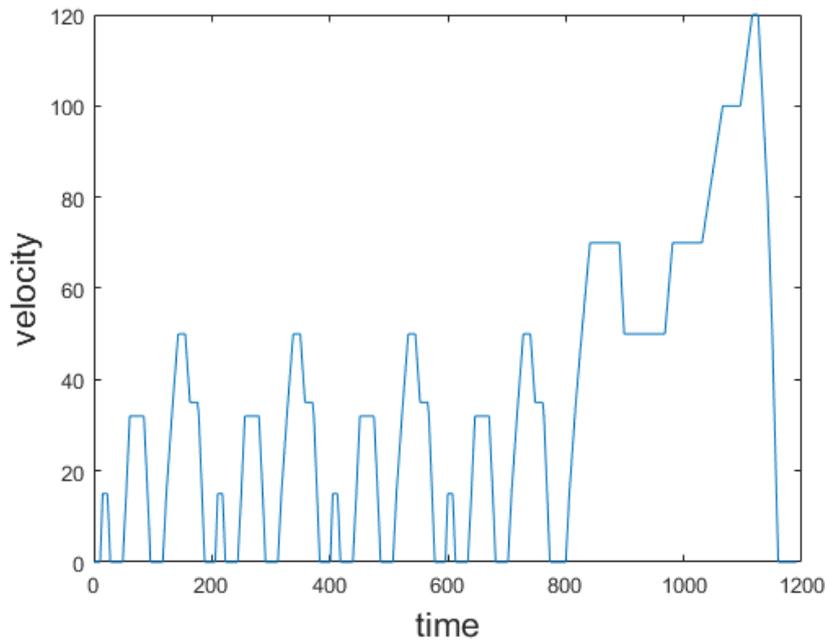


圖 5.7: 測試方法

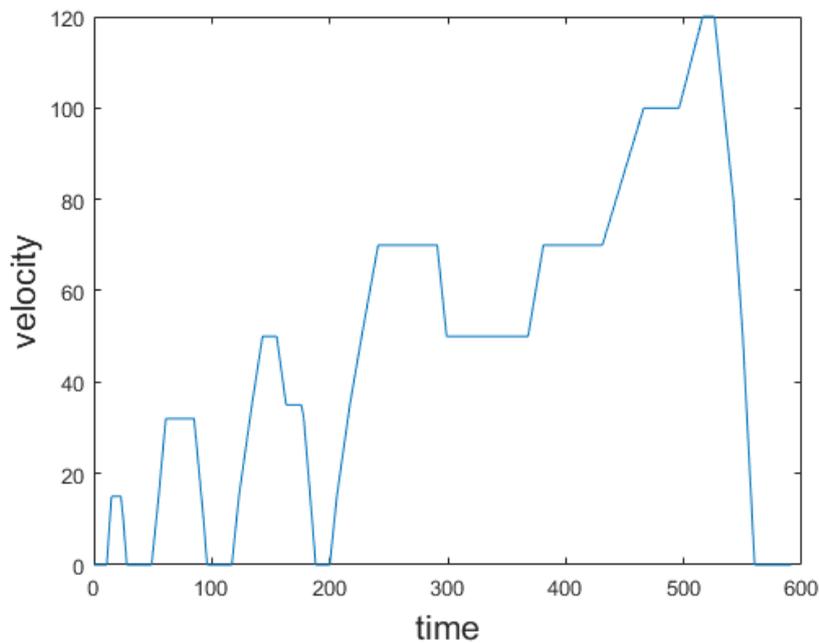


圖 5.8: 新測試方法

由於模擬過程車輛是在電池充滿電的狀況下行駛，故並未使用引擎和前軸輔助馬達，僅使用前軸馬達、後軸馬達能耗表現以及電池總耗電量校準參數，如表5.1，在此工程案例我們最關心的性能輸出是電池耗電量，故對電池耗電量進行主因素分析，由於模型中部分參數為二維矩陣，矩陣內有多個數值，例如：馬達性能曲線，難以進行主因素分析，因此並未對所有參數進行主因素分析，本研究對汽車物理模型中 22 個參數進行主因素分析，如表??。

表 5.1: 量測輸出

性能輸出	替代符號
前軸馬達消耗能量	$y_{m_1}$
後軸馬達消耗能量	$y_{m_2}$
前軸馬達煞車回充能量	$y_{m_3}$
後軸馬達煞車回充能量	$y_{m_4}$
電池總耗電量	$y_{m_5}$



表 5.2: 主因分析參數

參數編號	參數物理意義
1	前軸馬達控制扭力上限
2	前軸馬達控制扭力下限
3	後軸馬達控制扭力上限
4	後軸馬達控制扭力下限
5	後軸變速箱換檔速度
6	前軸馬達齒比 1
7	前軸馬達齒比 2
8	後軸馬達齒比 1
9	後軸馬達齒比 2
10	後軸馬達齒比 3
11	最低充電速限
12	最低充電電量
13	離合器控制扭力上限
14	離合器控制扭力下限
15	車重
16	輪胎滾動阻力
17	輪胎半徑
18	最大煞車力
19	風阻係數
20	車輛正投影面積
21	齒輪軸轉動慣量
22	齒輪阻尼係數

進行主因素分析後，主因素不為 0 之參數共有 12 個，如表 5.3，由於實際車輛的馬達齒比參數、最低充電速限應不會變動，故僅分析車重、風阻係數、車輛正投影面積、輪胎滾動阻力、輪胎半徑、最大煞車力等 6 個參數。

表 5.3: 主因素排名

主因素排名	參數物理意義	主因素
1	車重	0.7245
2	風阻係數	0.6833
3	車輛正投影面積	0.6833
4	輪胎滾動阻力	0.5326
5	輪胎半徑	0.1242
6	前軸馬達齒比 2	0.1075
7	後軸馬達齒比 1	0.0904
8	前軸馬達齒比 1	0.0453
9	後軸馬達齒比 3	0.0452
10	後軸馬達齒比 2	0.0171
11	最大煞車力	-0.0087
12	最低充電速限	0.0015



### 5.1.2 建立訓練集、驗證集、測試集

假設參數最小值為原設定值 0.8 倍，最大值為原設定值 1.2 倍，且為均勻分布，如表5.4，隨機產生 2000 組參數，以物理模型進行模擬，獲得 2000 筆虛擬實驗數據，其中 1000 筆作為訓練集、500 筆作為驗證集、500 筆作為測試集。

表 5.4: 參數設定值

參數物理意義	替代符號	原設定值	最小值	最大值
車重	$p_{m_1}$	2230	1858	2676
輪胎滾動阻力	$p_{m_2}$	0.010	0.008	0.012
輪胎半徑	$p_{m_3}$	0.370	0.296	0.444
最大煞車力	$p_{m_4}$	15000	12500	18000
風阻係數	$p_{m_5}$	0.370	0.296	0.444
車輛正向投影面積	$p_{m_6}$	2.790	2.232	3.348

為比較各參數校準結果，需先將參數數值標準化，標準化方法為將參數數值除以參數原設定值，參數標準化結果如表5.5

表 5.5: 參數標準化結果

參數物理意義	替代符號	原設定值	最小值	最大值
車重	$p_{m_1}$	1.0	0.8	1.2
輪胎滾動阻力	$p_{m_2}$	1.0	0.8	1.2
輪胎半徑	$p_{m_3}$	1.0	0.8	1.2
最大煞車力	$p_{m_4}$	1.0	0.8	1.2
風阻係數	$p_{m_5}$	1.0	0.8	1.2
車輛正向投影面積	$p_{m_6}$	1.0	0.8	1.2

### 5.1.3 建立正向校準模型

#### 1. 移除性能輸出重複值

每個性能輸出有 593 個值，但我們不必為每個值都建立模型，可去除重複值以節省計算量，以  $\mathbf{y}_{m_1}$  為例，若  $y_{m_1,1}^1 = y_{m_1,2}^1, y_{m_1,1}^2 = y_{m_1,2}^2, \dots, y_{m_1,1}^{2000} = y_{m_1,2}^{2000}$ ，也就是 2000 筆資料皆滿足  $y_{m_1,1} = y_{m_1,2}$  的條件，則只需要預測  $y_{m_1,1}$  即可。原  $\mathbf{y}_{m_1}^1$  如圖 5.9，移除重複值後如圖 5.10  $y_{m_1,j}^1$

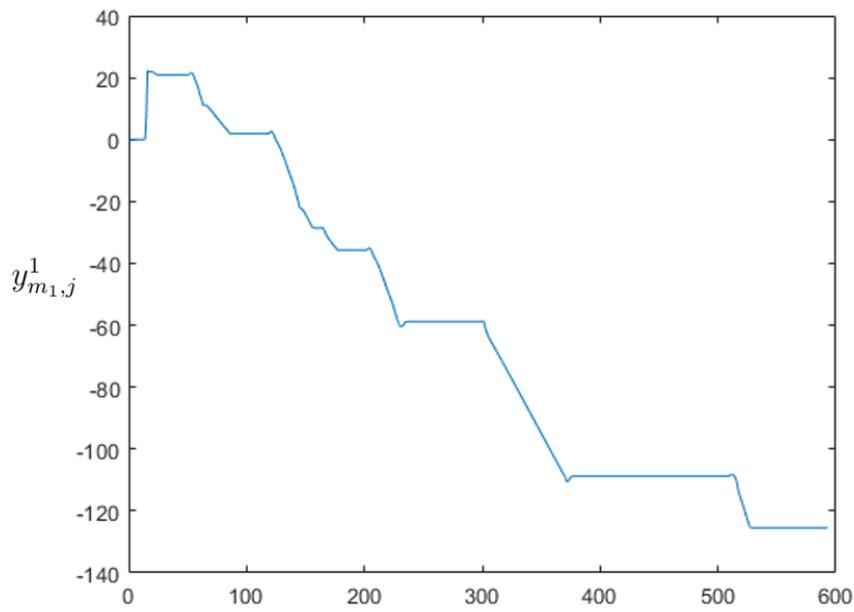


圖 5.9: 原  $y_{m_1}^1$

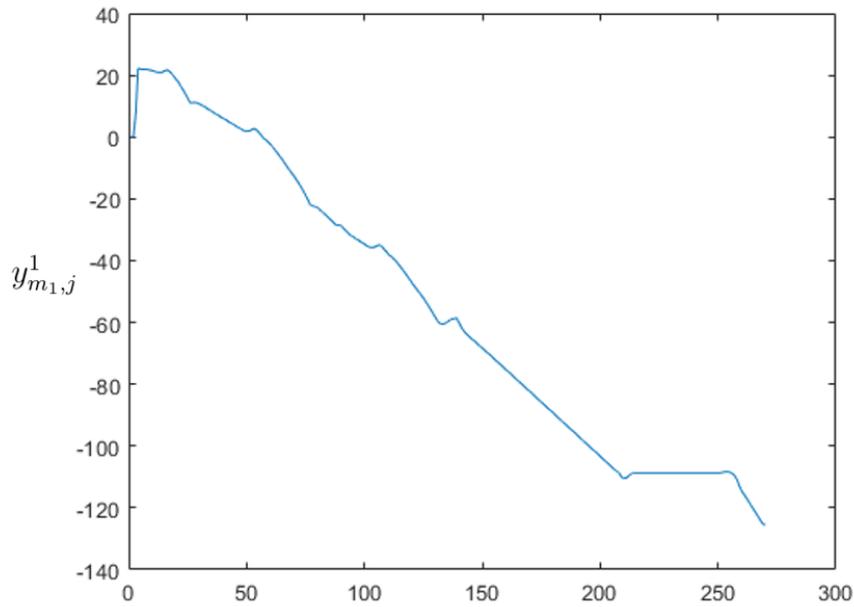


圖 5.10: 移除重複值後的  $y_{m_1}^1$

依序移除各項性能輸出重複值後，各項性能輸出保留數值的數量，如表5.6



表 5.6: 剩下性能輸出數值的數量

	原數量	保留數量
$\mathbf{y}_{m_1}$	593	253
$\mathbf{y}_{m_2}$	593	275
$\mathbf{y}_{m_3}$	593	78
$\mathbf{y}_{m_4}$	593	78
$\mathbf{y}_{m_5}$	593	488

## 2. 建立正向校準模型

以神經網路建立正向校準模型，使用 scikit-learn(version 0.19.0) 之套件 MLPRegressor，以 1 層隱藏層、50 個神經元進行訓練，詳細設定參數如下：

MLPRegressor(hidden\_layer\_sizes=(50,), max\_iter=2000,  
alpha=0.0,activation='relu',solver='lbfgs',tol=0.0001)

根據訓練集 1000 筆資料，以此參數設定依序建立正向校準模型如：

$$\begin{aligned} & \hat{f}_{1,1}^{fwd}(\mathbf{p}_m), \hat{f}_{1,2}^{fwd}(\mathbf{p}_m), \dots, \hat{f}_{1,253}^{fwd}(\mathbf{p}_m) \\ & \hat{f}_{2,1}^{fwd}(\mathbf{p}_m), \hat{f}_{2,2}^{fwd}(\mathbf{p}_m), \dots, \hat{f}_{2,275}^{fwd}(\mathbf{p}_m) \\ & \hat{f}_{3,1}^{fwd}(\mathbf{p}_m), \hat{f}_{3,2}^{fwd}(\mathbf{p}_m), \dots, \hat{f}_{3,78}^{fwd}(\mathbf{p}_m) \\ & \hat{f}_{4,1}^{fwd}(\mathbf{p}_m), \hat{f}_{4,2}^{fwd}(\mathbf{p}_m), \dots, \hat{f}_{4,78}^{fwd}(\mathbf{p}_m) \\ & \hat{f}_{5,1}^{fwd}(\mathbf{p}_m), \hat{f}_{5,2}^{fwd}(\mathbf{p}_m), \dots, \hat{f}_{5,488}^{fwd}(\mathbf{p}_m) \end{aligned}$$

我們僅保留在驗證集準確率達 99% 之模型，模型  $\hat{f}_{i,j}^{fwd}(\mathbf{p}_m)$  的標準化常數  $R_{i,j}$  如式 5.1，且  $n_{\text{train}} = 1000$ ，模型  $\hat{f}_{i,j}^{fwd}(\mathbf{p}_m)$  之準確率如式 5.2，且  $n_{\text{val}} = 500$ ，各性能輸出保留模型的數量如表 5.7。

$$R_{i,j} = \sum_{r=1}^{n_{\text{train}}} (y_{m_i,j}^r - \bar{y}_{m_i,j})^2 / n_{\text{train}} \quad (5.1)$$

$$1 - \frac{\sum_{r=1}^{n_{\text{val}}} (y_{m_i,j}^r - \hat{y}_{m_i,j}^r)^2 / n_{\text{val}}}{R_{i,j}} \quad (5.2)$$



表 5.7: 保留模型數量

性能輸出	訓練模型數量	保留模型數量
$\mathbf{y}_{m_1}$	253	35
$\mathbf{y}_{m_2}$	275	189
$\mathbf{y}_{m_3}$	78	14
$\mathbf{y}_{m_4}$	78	28
$\mathbf{y}_{m_5}$	488	412

最後共保留 678 個模型，將其改寫為式5.3，模型對應的標準化常數改寫為式5.4

$$\hat{\mathbf{y}}_{\text{set}} = \begin{bmatrix} \hat{y}_{\text{set},1} & \hat{y}_{\text{set},2} & \cdots & \hat{y}_{\text{set},678} \end{bmatrix}_{1 \times 678} \quad (5.3)$$

$$\mathbf{R}_{\text{set}} = \begin{bmatrix} R_{\text{set},1} & R_{\text{set},2} & \cdots & R_{\text{set},678} \end{bmatrix}_{1 \times 678} \quad (5.4)$$

則性能輸出估計誤差  $E_{\text{set}}$  可表示為式5.5。

$$E_{\text{set}} = \sum_{i=1}^{678} \frac{(y_{\text{set},i} - \hat{y}_{\text{set},i})^2}{R_{\text{set},i}} \quad (5.5)$$

### 3. 建立接受域與拒絕域

針對驗證集 500 筆資料，畫出  $E_{\text{set}}$  的分布圖如圖5.11。

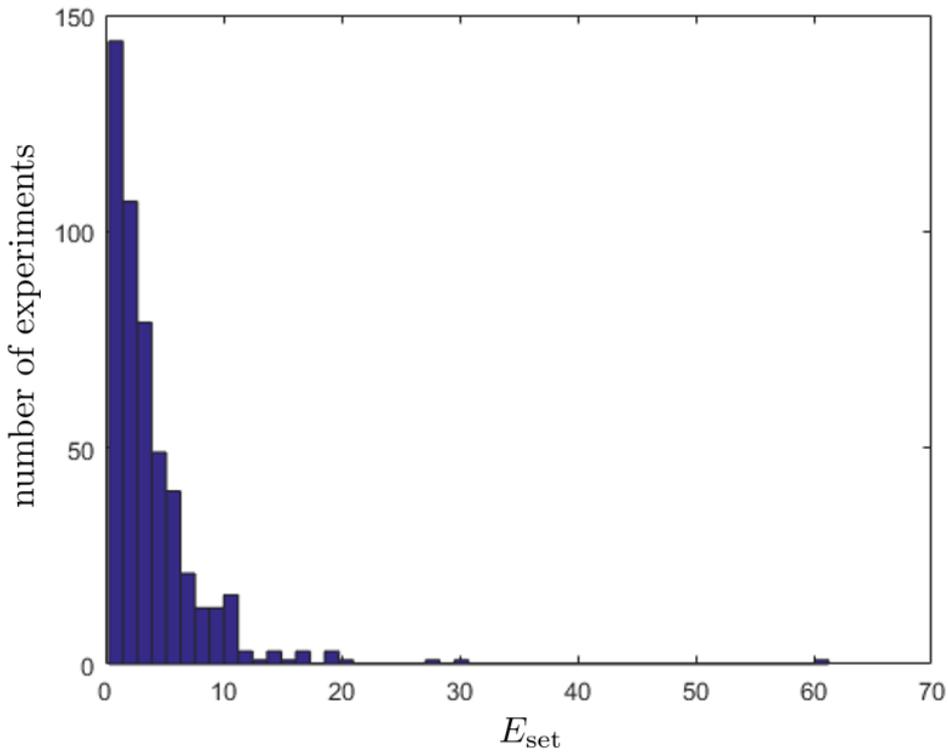


圖 5.11:  $E_{set}$  分布圖

$E_{set}$  平均值、標準差及接受域如表5.8，以 95% 信賴區間， $\alpha = 0.05$ ， $z_{0.05}=1.645$  建立接受域範圍。

表 5.8: 接受域

平均值	標準差	接受域
7.3203	4.4852	$E_{set} < 11.3153$

## 5.1.4 建立逆向校準模型

### 1. 建立逆向模型

定義模型  $\hat{p}_{m_k} = \hat{f}_{i,k}^{inv}(\mathbf{y}_{m_i})$  以 Neural Network 建立逆向校準模型，使用 scikit-learn(version 0.19.0) 之套件 MLPRegressor，以 1 層隱藏層、50 個神經元進行訓練，詳細設定參數如下：

```
MLPRegressor(hidden_layer_sizes=(50,), max_iter=2000,
alpha=0.0,activation='relu',solver='lbfgs',tol=0.00001)
```



根據訓練集 1000 筆資料，以此設定參數依序建立逆向校準模型如：

$$\begin{aligned} & \hat{f}_{1,1}^{inv}(\mathbf{y}_{m_1}), \hat{f}_{1,2}^{inv}(\mathbf{y}_{m_1}), \dots, \hat{f}_{1,6}^{inv}(\mathbf{y}_{m_1}) \\ & \hat{f}_{2,1}^{inv}(\mathbf{y}_{m_2}), \hat{f}_{2,2}^{inv}(\mathbf{y}_{m_2}), \dots, \hat{f}_{2,6}^{inv}(\mathbf{y}_{m_2}) \\ & \hat{f}_{3,1}^{inv}(\mathbf{y}_{m_3}), \hat{f}_{3,2}^{inv}(\mathbf{y}_{m_3}), \dots, \hat{f}_{3,6}^{inv}(\mathbf{y}_{m_3}) \\ & \hat{f}_{4,1}^{inv}(\mathbf{y}_{m_4}), \hat{f}_{4,2}^{inv}(\mathbf{y}_{m_4}), \dots, \hat{f}_{4,6}^{inv}(\mathbf{y}_{m_4}) \\ & \hat{f}_{5,1}^{inv}(\mathbf{y}_{m_5}), \hat{f}_{5,2}^{inv}(\mathbf{y}_{m_5}), \dots, \hat{f}_{5,6}^{inv}(\mathbf{y}_{m_5}) \end{aligned}$$

方均根誤差如表 5.9。

表 5.9: 驗證集單一性能輸出方均根誤差

	$p_{m_1}$	$p_{m_2}$	$p_{m_3}$	$p_{m_4}$	$p_{m_5}$	$p_{m_6}$
$\mathbf{y}_{m_1}$	0.00193	0.00352	0.00344	0.14000	0.08798	0.09108
$\mathbf{y}_{m_2}$	0.00343	0.00559	0.00236	0.12938	0.08771	0.08581
$\mathbf{y}_{m_3}$	0.00472	0.00838	0.00363	0.00864	0.08198	0.08469
$\mathbf{y}_{m_4}$	0.00502	0.01138	0.00346	0.01055	0.08871	0.08584
$\mathbf{y}_{m_5}$	0.00487	0.00627	0.00836	0.02131	0.08499	0.08548

將一共 30 個校準值表示為： $\hat{\mathbf{p}}_{\text{set}} = [\hat{p}_1, \hat{p}_2, \dots, \hat{p}_{30}]$ ，並以決策樹提高校準準確率。

使用 scikit-learn(version 0.19.0) 之套件 GradientBoostingRegressor，設定參數為：

GradientBoostingRegressor(n\_estimators=2000,max\_depth=3, random\_state=0)

根據訓練集 1000 筆資料，以此參數設定建立決策樹校準模型：

$$\hat{f}_1^{dt}(\hat{\mathbf{p}}_{\text{set}}), \hat{f}_2^{dt}(\hat{\mathbf{p}}_{\text{set}}), \dots, \hat{f}_9^{dt}(\hat{\mathbf{p}}_{\text{set}})$$

驗證集中，決策樹校準方均根誤差如表 5.10，可見並非所有參數誤差皆降低， $p_{m_1}, \dots, p_{m_4}$  的誤差皆比原最小誤差小，顯示結合多個模型校準結果可提升準確率。而可能有多組解，誤差較大的參數  $p_{m_5}, p_{m_6}$ ，則有過擬合的現象，誤差反而增加。

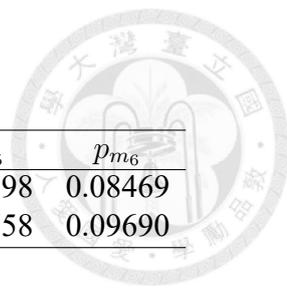


表 5.10: 驗證集決策樹方均根誤差

	$p_{m_1}$	$p_{m_2}$	$p_{m_3}$	$p_{m_4}$	$p_{m_5}$	$p_{m_6}$
原最小誤差	0.00193	0.00352	0.00236	0.00864	0.08198	0.08469
決策樹誤差	0.00190	0.00307	0.00224	0.00691	0.09758	0.09690

## 2. 決定正向校準搜尋範圍

驗證集的參數逆向校準誤差分布如圖5.12

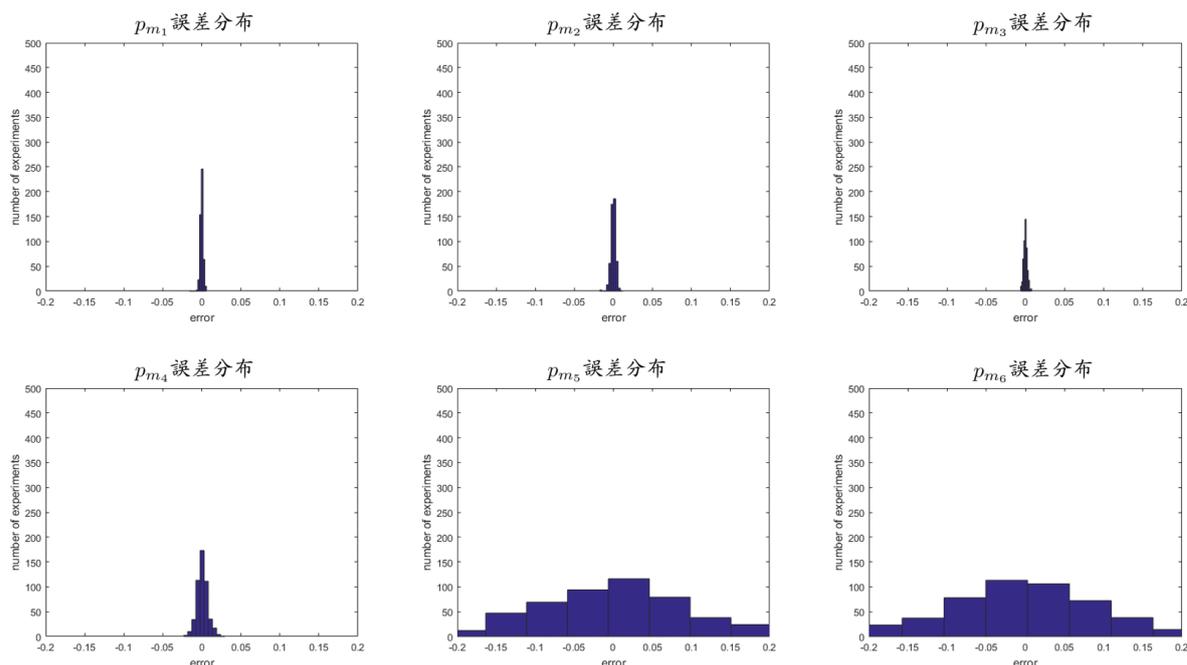


圖 5.12: 逆向校準誤差分布

各參數逆向校準誤差平均值、標準差、最大誤差如表5.11，誤差小的參數如  $p_{m_1}, \dots, p_{m_4}$ ，正向校準時會有較小的搜尋範圍，誤差大的參數如  $p_{m_5}, p_{m_6}$ ，正向校準時應有較大的搜尋範圍。若  $\hat{p}_m$  位於正向校準接受域，代表  $\hat{p}_m$  接近  $p_m$ ，則根據各參數的誤差標準差，決定正向校準的參數搜尋範圍，以 0.005 做為最小單位，範圍為兩倍誤差標準差，若  $\hat{p}_m$  位於正向校準拒絕域，代表  $\hat{p}_m$  和  $p_m$  差距較大，則根據各參數之最大誤差，作為正向校準的參數搜尋範圍，以 0.005 做為最小單位，範圍為最大誤差，由於  $p_{m_5}, p_{m_6}$  誤差較大，可能有多組解，接受域和拒絕域的搜尋範圍皆設為 0.8-1.2，各參數搜尋範圍如表5.12、5.13



表 5.11: 驗證集誤差

參數	平均誤差	誤差標準差	最大誤差
$p_{m_1}$	0.00008	0.00190	0.01617
$p_{m_2}$	0.00009	0.00307	0.01777
$p_{m_3}$	0.00003	0.00224	0.00788
$p_{m_4}$	0.00083	0.00696	0.02948
$p_{m_5}$	0.00020	0.09758	0.26835
$p_{m_6}$	0.00022	0.09690	0.26993

表 5.12: 接受域搜尋範圍

參數	搜尋範圍
$p_{m_1}$	$[\hat{p}_{m_1} - 0.005, \hat{p}_{m_1} + 0.005]$
$p_{m_2}$	$[\hat{p}_{m_2} - 0.010, \hat{p}_{m_2} + 0.010]$
$p_{m_3}$	$[\hat{p}_{m_3} - 0.005, \hat{p}_{m_3} + 0.005]$
$p_{m_4}$	$[\hat{p}_{m_4} - 0.015, \hat{p}_{m_4} + 0.015]$
$p_{m_5}$	$[\hat{p}_{m_5} - 0.190, \hat{p}_{m_5} + 0.190]$
$p_{m_6}$	$[\hat{p}_{m_6} - 0.190, \hat{p}_{m_6} + 0.190]$

表 5.13: 拒絕域搜尋範圍

參數	搜尋範圍
$p_{m_1}$	$[\hat{p}_{m_1} - 0.020, \hat{p}_{m_1} + 0.020]$
$p_{m_2}$	$[\hat{p}_{m_2} - 0.020, \hat{p}_{m_2} + 0.020]$
$p_{m_3}$	$[\hat{p}_{m_3} - 0.010, \hat{p}_{m_3} + 0.010]$
$p_{m_4}$	$[\hat{p}_{m_4} - 0.030, \hat{p}_{m_4} + 0.030]$
$p_{m_5}$	$[\hat{p}_{m_5} - 0.270, \hat{p}_{m_5} + 0.270]$
$p_{m_6}$	$[\hat{p}_{m_6} - 0.270, \hat{p}_{m_6} + 0.270]$

### 5.1.5 校準參數

以測試集資料中一組資料作為參數校準範例， $\mathbf{p}_m$  之實際值如表 5.14

表 5.14: 參數實際值

$p_{m_1}$	$p_{m_2}$	$p_{m_3}$	$p_{m_4}$	$p_{m_5}$	$p_{m_6}$
0.9190	1.0331	0.8050	0.8168	1.1793	1.0101

綜合逆向模型及決策樹模型校準結果如式 5.15



表 5.15: 參數預測值

$\hat{p}_{m_1}$	$\hat{p}_{m_2}$	$\hat{p}_{m_3}$	$\hat{p}_{m_4}$	$\hat{p}_{m_5}$	$\hat{p}_{m_6}$
0.9200	1.0371	0.8028	0.8065	1.0313	1.0679

以正向校準計算其  $E_{\text{set}}$  之值為 50.9765，大於接受域臨界值 11.3153，因此不信任  $\hat{\mathbf{p}}_m$ ，以拒絕域搜尋範圍進行搜尋，搜尋範圍如表 5.16。

表 5.16: 參數搜尋範圍

參數	搜尋範圍
$p_{m_1}$	[0.9000,0.9400]
$p_{m_2}$	[1.0171,1.0517]
$p_{m_3}$	[0.7928,0.8128]
$p_{m_4}$	[0.7765,0.8365]
$p_{m_5}$	[0.8000,1.2000]
$p_{m_6}$	[0.8000,1.2000]

在搜尋範圍內隨機搜尋 10000 組參數，共有 744 組參數  $E_{\text{set}}$  位於接受域， $E_{\text{set}}$  最小值為 3.565，並以  $E_{\text{set}}$  最小值對應之參數更新  $\hat{\mathbf{p}}_m$ 。

表 5.17: 更新參數校準結果

	$p_{m_1}$	$p_{m_2}$	$p_{m_3}$	$p_{m_4}$	$p_{m_5}$	$p_{m_6}$
實際值	0.9190	1.0331	0.8050	0.8168	1.1793	1.0101
第一次預測值	0.9200	1.0371	0.8028	0.8065	1.0313	1.0679
第二次預測值	0.9202	1.0540	0.8027	0.8132	1.1825	1.0171

$p_{m_1} \dots p_{m_4}$  第一次和第二次校準結果都相當接近實際值，而  $p_{m_5}, p_{m_6}$  經過第二次校準後更接近實際值。畫出 744 組位於接受域的參數分布圖如圖 5.13，可見  $p_{m_1} \dots p_{m_4}$  有分布範圍較小， $p_{m_5}, p_{m_6}$  的分布範圍較大，證實有多個參數組合可產生相似的性能輸出。

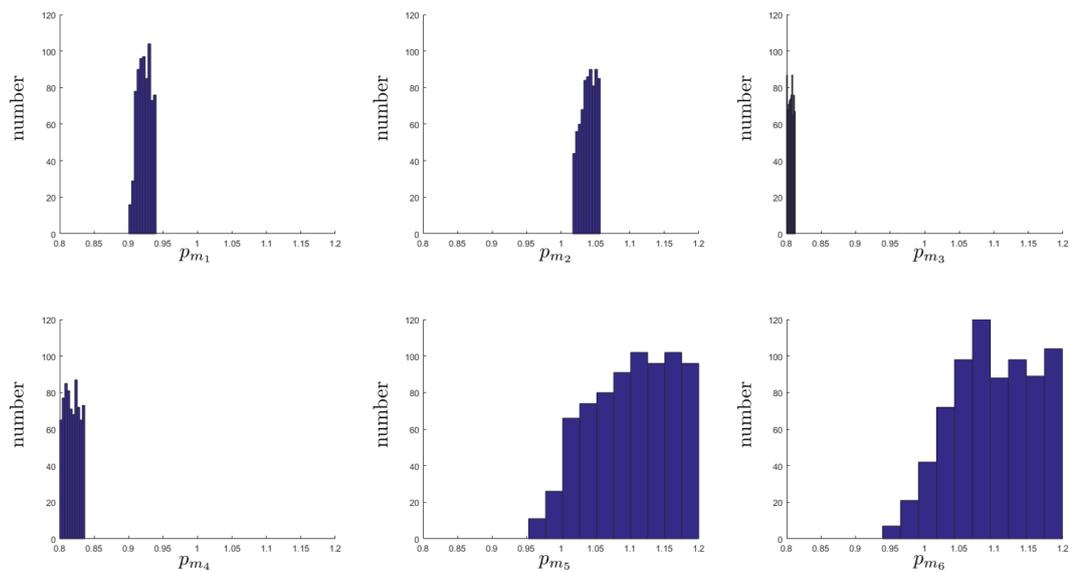


圖 5.13: 位於接受域的參數分布

$p_{m_5}, p_{m_6}$  分別為風阻係數、車輛正向投影面積，這兩個參數會影響車輛所受空氣阻力大小，本研究的車輛模型車輛動力計算方法如式5.6，而空氣阻力大小如式5.7。

$$\text{車輛動力} = \text{馬達驅動力} - \text{滾動阻力} - \text{空氣阻力} - \text{煞車力} \quad (5.6)$$

$$\text{空氣阻力} \propto \text{空氣密度} \times \text{風阻係數} \times \text{車輛正向投影面積} \times \text{車速}^2 \quad (5.7)$$

風阻係數  $p_{m_5}$  數值為 1.1793，車輛正向投影面積  $p_{m_6}$  數值為 1.0101，根據式5.7， $p_{m_5}, p_{m_6}$  應滿足式5.8。

$$p_{m_5} \times p_{m_6} = 1.1793 \times 1.0101 = 1.1912 \quad (5.8)$$

畫出  $p_{m_5}, p_{m_6}$  位於接受域的分布圖，如圖5.14，可發現  $p_{m_5}, p_{m_6}$  接受域的分布範圍和符合模型的理論，正向校準參數的結果合理。

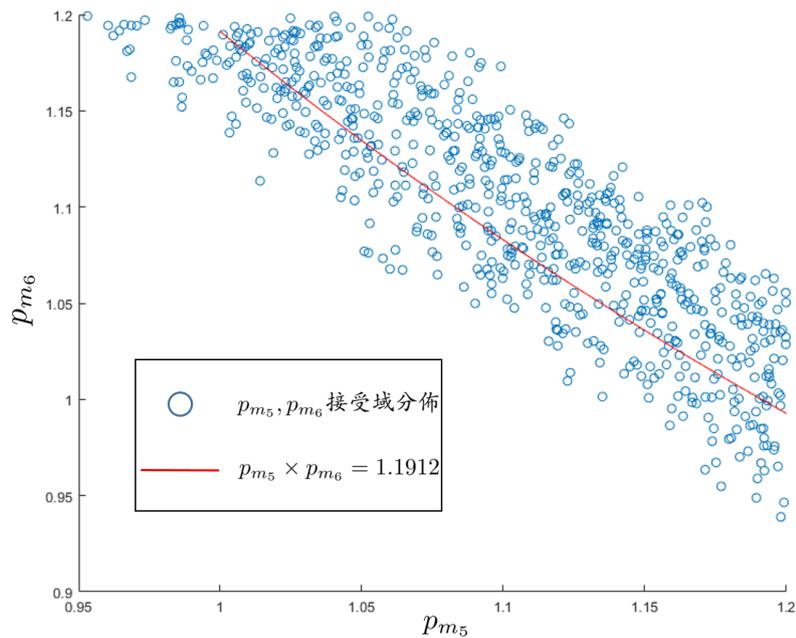


圖 5.14:  $p_{m_5}, p_{m_6}$  接受域分布

若有參數容易量測且準確率高，可使用量測結果取代逆向校準的參數值， $p_{m_1}, \dots, p_{m_4}$  逆向校準準確率高風阻係數  $p_{m_5}$  逆向校準的準確率低，且不易量測，車輛正向投影面積  $p_{m_6}$  逆向校準的準確率低，但易於量測，可用  $p_{m_6}$  量測結果取代  $p_{m_6}$  逆向校準結果， $p_{m_1}, \dots, p_{m_4}$  則使用原逆向校準結果，則僅需搜尋  $p_{m_5}$  位於接受域的參數值分布。假設  $p_{m_6}$  的量測結果和實際值相同，重新以正向校準方法，搜尋  $p_{m_5}$  位於接受域的參數值分布，搜尋範圍如表 5.18，結果如圖 5.15，可見  $p_{m_5}$  位於接受域的分佈範圍較原先縮小，從 0.95-1.20 縮小為 1.17-1.20，而  $p_{m_5}$  實際值為 1.1793，藉由量測  $p_{m_6}$ ，可更準確的校準  $p_{m_5}$ 。

表 5.18: 參數搜尋範圍

參數	搜尋範圍
$p_{m_1}$	[0.9200,0.9200]
$p_{m_2}$	[1.0371,1.0371]
$p_{m_3}$	[0.8028,0.8028]
$p_{m_4}$	[0.8065,0.8065]
$p_{m_5}$	[0.8000,1.2000]
$p_{m_6}$	[1.0101,1.0101]

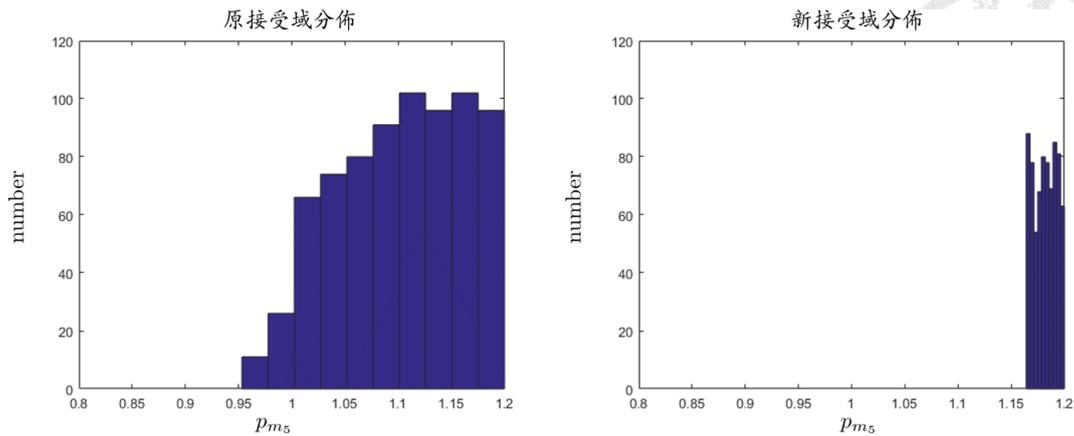


圖 5.15:  $p_{m_5}$  接受域分佈

### 5.1.6 結果與討論

訓練集、驗證集和測試集逆向校準方均根誤差如表5.19，由於  $p_{m_5}, p_{m_6}$  有多組解，因此無法準確校準，雖然在訓練集的誤差很小，但在驗證集和測試集的誤差很大，顯示  $p_{m_5}, p_{m_6}$  有過擬合訓練集的情形。 $p_{m_1}, \dots, p_{m_4}$  方均根誤差約 0.002 至 0.008，誤差略大於訓練集，但誤差仍相當的小，可準確校準參數，而  $p_{m_1}, \dots, p_{m_6}$  驗證集和測試集的方均根誤差則相當接近，代表逆向校準模型並未過度擬合驗證集。

表 5.19: 訓練集、驗證集、測試集方均根誤差

	$p_{m_1}$	$p_{m_2}$	$p_{m_3}$	$p_{m_4}$	$p_{m_5}$	$p_{m_6}$
訓練集	0.00031	0.00031	0.00031	0.00031	0.00085	0.00092
驗證集	0.00190	0.00307	0.00224	0.00691	0.09758	0.09690
測試集	0.00158	0.00308	0.00217	0.00819	0.10013	0.10004

車重  $p_{m_1}$ 、輪胎滾動阻力  $p_{m_2}$ 、輪胎半徑  $p_{m_3}$ 、最大煞車力  $p_{m_4}$  皆有相當小的校準誤差，方均根誤差約 0.002 至 0.008，其中最大煞車力  $p_{m_4}$  的主因素最小，但也有相當小的校準誤差，代表即使主因素較小，若無多組解的情形，仍能準確校準參數。

逆向校準方法可提供正向校準參數的搜尋範圍，提升校準的效率，根據逆向校準提供的搜尋範圍，隨機搜尋 10000 組參數，測試集 500 筆資料中，平均有 2420 組參數位於接受域內，若不根據逆向校準的搜尋範圍，在 0.8-1.2 的範圍內隨機搜尋 10000 組參數，測試集 500 筆資料中，平均只有 14 組位於接受域內，顯示

逆向校準提供的參數搜尋範圍有效提升找到正確參數的效率。風阻係數  $p_{m5}$ 、車輛正向投影面積  $p_{m6}$  由於有多組解，無法以逆向校準方法準確校準，但能以正向校準方法找出可能解的分布範圍，且正向校準的結果和模型理論相似，風阻係數和車輛正向投影面積的乘積為一定值。

本研究建議使用三種方法提升校準準確率：1. 使用更多訓練資料、2. 使用更多測試方法、3. 使用不同性能輸出，在此工程案例都有提升參數校準準確率的作用。

### 1. 使用更多訓練資料

橫軸% 代表使用所有訓練資料的百分比，100% 代表使用所有訓練資料，隨著訓練資料增加，測試集參數校準誤差有下降的趨勢，如圖5.16，顯示使用更多訓練資料有助提升參數校準準確率。

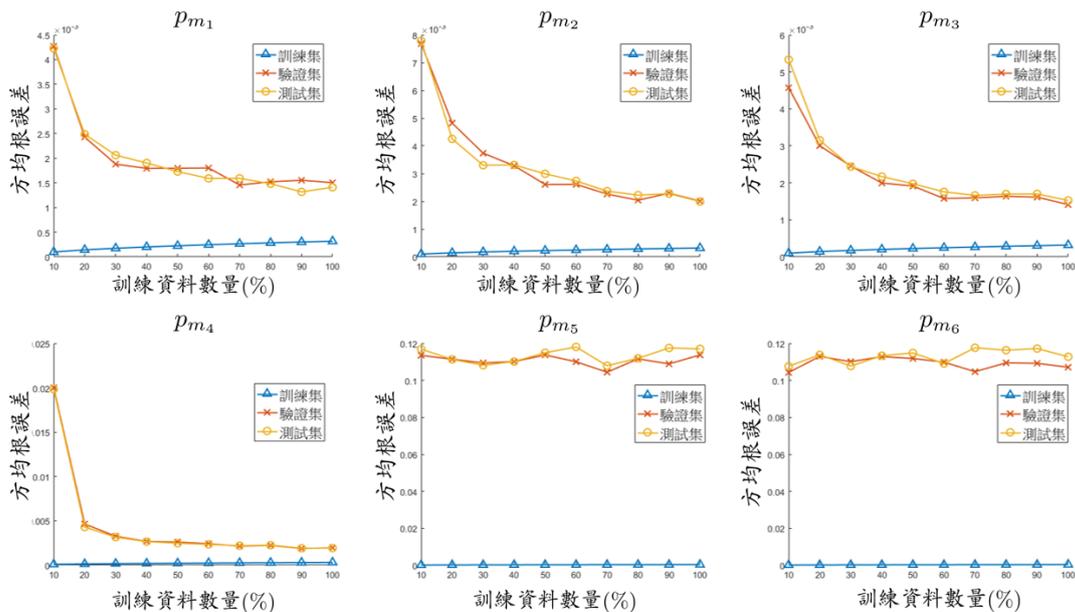


圖 5.16: 使用更多訓練資料結果

### 2. 使用更多測試方法

橫軸% 代表使用所有測試方法的百分比，100% 代表使用所有測試方法，隨著測試方法增加，測試集參數校準誤差有下降的趨勢，如圖5.16，雖不如使用更多訓練資料，但使用更多測試方法的確有助提升參數校準準確率。

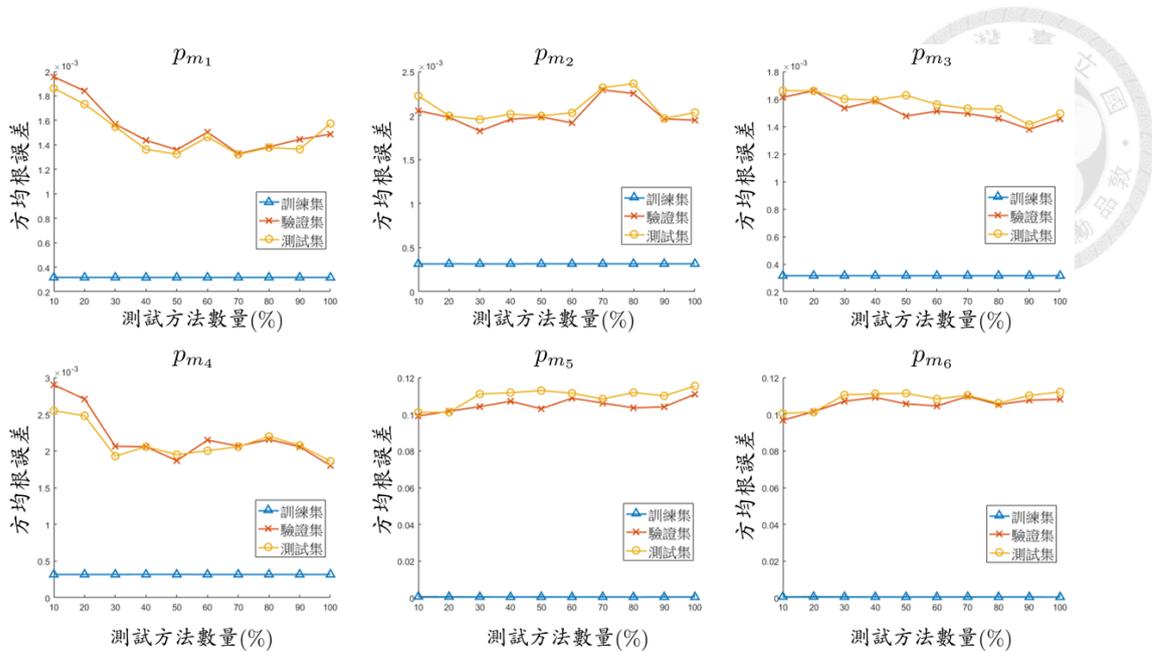


圖 5.17: 使用更多測試方法結果

### 3. 使用不同性能輸出

使用不同性能輸出進行校準，測試集的方均根誤差如表5.20，可見無法僅用單一性能輸出獲得最小的誤差，若再使用決策樹進行校準，部分參數可進一步提升校準準確率，誤差比各性能輸出的最小誤差更小，如表5.21，顯示使用不同性能輸出可提升參數校準準確率。

表 5.20: 測試集單一性能輸出方均根誤差

	$p_{m_1}$	$p_{m_2}$	$p_{m_3}$	$p_{m_4}$	$p_{m_5}$	$p_{m_6}$
$y_{m_1}$	0.00136	0.00342	0.00320	0.13261	0.09475	0.09399
$y_{m_2}$	0.00346	0.00530	0.00242	0.13250	0.08954	0.08879
$y_{m_3}$	0.00474	0.00904	0.00389	0.00941	0.08896	0.08973
$y_{m_4}$	0.00524	0.01209	0.00365	0.01244	0.09081	0.09195
$y_{m_5}$	0.00519	0.00634	0.00783	0.02136	0.08712	0.08537

表 5.21: 測試集決策樹方均根誤差

	$p_{m_1}$	$p_{m_2}$	$p_{m_3}$	$p_{m_4}$	$p_{m_5}$	$p_{m_6}$
原最小誤差	0.00136	0.00342	0.00242	0.00941	0.08712	0.08537
決策樹誤差	0.00158	0.00308	0.00217	0.00819	0.10013	0.10004



## 第六章

---

# 結論

---

## 6.1 研究貢獻

本研究提出一套複雜系統參數校準方法，藉由主因素分析辨識重要參數根據性能輸出偏移，以多個性能輸出校準偏移之參數。本論文主要的貢獻為：

### 1. 建構校準複雜系統參數的流程

本研究為複雜系統參數校準提出一套建議的分析方法，將參數校準的問題聚焦到重要參數校準上，並結合正向校準、逆向校準方法，藉由性能輸出偏移校準偏移的參數。

### 2. 建構逆向校準參數方法

逆向校準參數方法可根據性能輸出，不透過疊代及最佳化演算法，迅速校準可能偏移的參數，並提供正向校準參數搜尋的初始值及搜尋範圍。

### 3. 建構正向校準參數方法

正向校準參數方法可評估參數對性能輸出之影響，處理參數多組解的問題，並以信賴區間評估參數校準的準確率。

### 4. 提出提升校準準確率之方法

本研究提出提升校準準確率之方法，當校準準確率不足時，可藉由使用更多訓練資料、使用不同性能輸出、使用更多測試方式等三種方法提升校準準確

率，若有參數可藉由量測獲得準確數值，亦能提升其他參數校準準確率。



## 6.2 未來展望

本研究的案例探討仍有不足之處，未來研究方向可朝改善以下兩點進行，使研究整體更加完善：

### 1. 以實際實驗驗證研究方法

本研究並未以實際實驗數據驗證研究方法，以實際實驗數據驗證時，尚須考慮物理模型和系統的模型誤差以及實驗的量測誤差，研究方法需要進行修正。

### 2. 建立即時校準方法

本研究案例探討的汽車性能測試方式為 NEDC Cycle，未來應建立即時校準方法，如此不必使用特定測試方式，使系統在一般情況運行時即可進行參數校準。

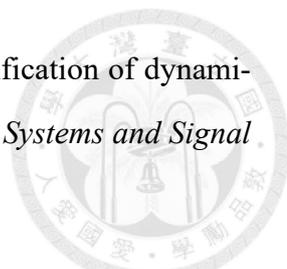


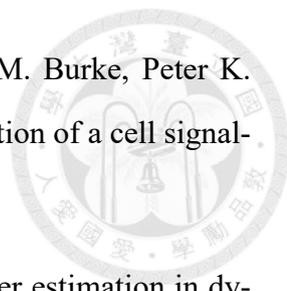
---

## 參考文獻

---

- [1] Andrew H. Briggs, Milton C. Weinstein, Elisabeth A. L. Fenwick, Jonathan Karnon, Mark J. Sculpher, and A. David Paltiel. Model parameter estimation and uncertainty: A report of the ispor-smdm modeling good research practices task force-6. *Value in Health*, 15(6):835–842, September 2012.
- [2] Giulio Reina, Matilde Paiano, and Jose-Luis Blanco-Claraco. Vehicle parameter estimation using a model-based estimator. *Mechanical Systems and Signal Processing*, 87:227 – 241, 2017. Signal Processing and Control challenges for Smart Vehicles.
- [3] J. C. Jensen, D. H. Chang, and E. A. Lee. A model-based design methodology for cyber-physical systems. In *2011 7th International Wireless Communications and Mobile Computing Conference*, pages 1666–1671, July 2011.
- [4] Richard C. Aster, Brian Borchers, and Clifford H. Thurber. Chapter one - introduction. In Richard C. Aster, Brian Borchers, and Clifford H. Thurber, editors, *Parameter Estimation and Inverse Problems (Second Edition)*, pages 1 – 23. Academic Press, Boston, second edition edition, 2013.
- [5] J.L. Crassidis and J.L. Junkins. *Optimal Estimation of Dynamic Systems*. Chapman & Hall/CRC Applied Mathematics & Nonlinear Science. Taylor & Francis, 2004.
- [6] F. Liu, M. Bayarri, and J. Berger. Modularization in bayesian analysis, with emphasis on analysis of computer models. *Bayesian Analysis*, 4(1):119–150, 2009.

- 
- [7] P.L. Green, E.J. Cross, and K. Worden. Bayesian system identification of dynamical systems using highly informative training data. *Mechanical Systems and Signal Processing*, 56-57:109 – 122, 2015.
- [8] R. Isermann and M. Münchhof. *Identification of Dynamic Systems: An Introduction with Applications*. Advanced Textbooks in Control and Signal Processing Series. Springer Berlin Heidelberg, 2010.
- [9] David E. Rumelhart, Geoffrey E. Hinton, and Ronald J. Williams. Learning representations by back-propagating errors. *Nature*, 323:533, October 1986.
- [10] K. S. Narendra and K. Parthasarathy. Identification and control of dynamical systems using neural networks. *IEEE Transactions on Neural Networks*, 1(1):4–27, Mar 1990.
- [11] Wei He, Nicholas Williard, Chaochao Chen, and Michael Pecht. State of charge estimation for li-ion batteries using neural network modeling and unscented kalman filter-based error cancellation. 2014.
- [12] Leandro Vargas-Melendez, Beatriz L. Boada, Maria Jesus L. Boada, Antonio Gauchia, and Vicente Diaz. Sensor fusion based on an integrated neural network and probability density function (pdf) dual kalman filter for on-line estimation of vehicle parameters and states. *Sensors (Basel, Switzerland)*, 17(PMC5469340):987, April 2017.
- [13] Geoffrey K.F. Tso and Kelvin K.W. Yau. Predicting electricity energy consumption: A comparison of regression analysis, decision tree and neural networks. *Energy*, 32(9):1761 – 1768, 2007.
- [14] Costas Kravaris, Juergen Hahn, and Yunfei Chu. Advances and selected recent developments in state and parameter estimation. *Computers & Chemical Engineering*, 51:111–123, April 2013.
- [15] L. Ljung and T. Chen. Convexity issues in system identification. In *2013 10th IEEE International Conference on Control and Automation (ICCA)*, pages 1–9, 2012.

- 
- [16] Kyoung Ae Kim, Sabrina L. Spencer, John G. Albeck, John M. Burke, Peter K. Sorger, Suzanne Gaudet, and Do Hyun Kim. Systematic calibration of a cell signaling network model. *BMC Bioinformatics*, 11(1):202, Apr 2010.
- [17] Attila Gábor and Julio R. Banga. Robust and efficient parameter estimation in dynamic models of biological systems. *BMC Systems Biology*, 9(1):74, Oct 2015.
- [18] P. Arendt, D. Apley, W. Chen, D. Lamb, and D. Gorsich. Improving identifiability in model calibration using multiple responses. *Journal of Mechanical Design*, 134(10), 2012.
- [19] Stefano Conti and Anthony O' Hagan. Bayesian emulation of complex multi-output and dynamic computer models. *Journal of Statistical Planning and Inference*, 140(3):640–651, March 2010.
- [20] 林岳羿. 使用模型校準以識別複雜系統參數數值之方法. Master's thesis, 臺灣大學, 2016.