

國立交通大學

電機與控制工程學系

碩士論文

異質性雙軸網路系統的同步控制策略

**The Synchronized Motion Controller Design in Remote  
Control Systems With Different Time Delays**

研究生：林軒正

指導教授：徐保羅 博士

中華民國一百零一年年七月

# 異質性雙軸網路系統的同步控制策略

## The Synchronized Motion Controller Design in Remote Control Systems With Different Time Delays

研究生：林軒正

Student : Hsuan-Cheng Lin

指導教授：徐保羅 博士

Advisor : Dr. Pau-Lo Hsu



A Thesis

Submitted to Institute of Electrical and Control Engineering  
College of Electrical Engineering  
National Chiao-Tung University  
in Partial Fulfillment of the Requirements  
for the Degree of Master  
in  
Electrical and Control Engineering  
July 2012  
Hsinchu, Taiwan, Republic of China

中華民國一百零一年年七月

# 異質網路之遠端同步控制系統設計

研究生：林軒正

指導教授：徐保羅 博士

國立交通大學 電控工程研究所

## 摘要

在網路控制系統中，時間延遲與資料遺失對系統的穩定性有很大的影響，本研究結合時間延遲完全補償策略(PDC)與最小平方估測法處理網路延遲的問題。PDC將延遲的影響簡化為原閉迴路系統串接一時間延遲，避免系統不穩定；而最小平方估測用既有資料產生估測項，避免空白取樣發生時，資料沒有更新使誤差增加，由實驗得到，加入最小平方估測後，當突發性的大延遲發生時可避免突然增加的控制誤差量。

本研究建立異質性網路雙軸系統並進行循圓實驗，雙軸延遲不同造成兩軸在循圓的結果為橢圓，藉由量測網路傳輸延遲並建立雙軸的等待機制使其同步，將真圓度由 0.157 改善到 0.013。延遲測量實驗結果顯示傳輸延遲會變動，本論文進一步加入即時補償機制可達成即時同步調整。

將交叉耦合控制器(CCC)與PDC結合並應用於雙軸異質性網路控制系統，克服CCC雙軸誤差計算上的不匹配之後，能夠進一步改善循圓的輪廓誤差。進行新竹到台中的遠端控制實驗並以上述同步方法改善，真圓度可降低為 1/10，輪廓誤差降為原本的 1/4。

關鍵字：網路控制系統，估測，交叉耦合控制，同步

# **The Synchronized Motion Controller Design in Remote Control Systems With Different Time Delays**

**Student : Hsuan-Cheng Lin**

**Advisor : Dr. Pau-Lo Hsu**

**Institute of Electrical Control Engineering  
National Chiao-Tung University**

## **ABSTRACT**

The data dropout and time delay in the network control system (NCS) have a great effect in control performance. In this Thesis, we combine both the perfect delay compensation (PDC) and the least square estimation (LSE) to recover the dropout data caused by the network-induced time delay. In this study, experiments were constructed through NCS with different time delays. As the command is circular, the contour becomes elliptical due to the different delays in two NCS, one is CAN and another is Ethernet. The round trip time (RTT) between two network nodes was measured and furthermore, the faster NCS is designed to wait for the slower NCS to result in the synchronized motion. The roundness is then significantly improved from 0.157 to 0.013. As the time delay is varied in the commercial Ethernet, the adaptive compensation based on the real measured RTT was adopted to match the real time delay. The cross-coupled control (CCC) was further applied and experiments between Hsinchu and Taichung show that roundness and IAE for the 2-axis motion control system can be improved by 90% and 25%, respectively, in the roundness and IAE.

**Keywords:** networked control systems, estimation, CCC, synchronization.

## 誌謝

能夠完成本論文，要感謝在這段歲月所有幫助我，支持我的人，首先要感謝指導教授徐保羅 博士，不厭其煩地教導，不時導正我的研究方向與方法，在我屢遭瓶頸與挫折時，適時地鼓勵我，引領我至正確的道路，順利取得碩士學位。在此，獻上我最誠摯的敬意與感謝。

感謝博士班賴建良學長、謝鎮州學長黃煒生學長的給我的意見與幫助以及實驗室的同學釗甫、政宏、思翰，學弟志翰、琮昇、信佑、振文、誌緯、晟傑在學業上的切磋與指教，生活上的分享與幫助。

最後感謝我的父母，姊妹與女友給我的支持幫助與鼓勵，在我徬徨無助時給我希望並讓我更加堅強，謝謝你們的付出與關懷，將此榮耀喜悅與你們一同分享。



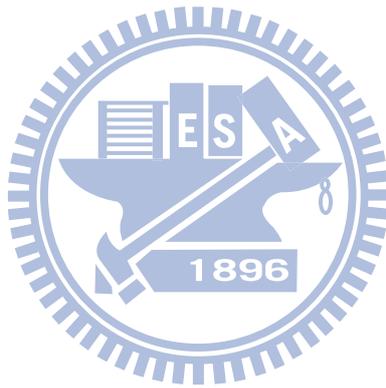
# 目 錄

異質性雙軸網路系統的同步控制策略.....	1
摘要.....	I
ABSTRACT.....	II
表 目 錄.....	VI
圖 目 錄.....	VII
第一章 緒論.....	1
1-1 研究動機.....	1
1-2 研究背景與發展概況.....	1
網路傳輸所造成的延遲(time delay).....	2
網路引起的訊息遺失(data dropout).....	2
訊息排程(network scheduling).....	2
節點間的時脈不同步(jitter).....	3
取樣週期(sampling period).....	3
QoS(quality of service)[15].....	4
1-3 問題描述.....	4
1-4 研究方法.....	4
1-5 論文架構.....	5
第二章 CAN、Ethernet 與實驗系統架構.....	6
2-1 CAN (Controller Area Network)[16].....	6
CAN 網路協定簡介.....	6
2-2 乙太網路(Ethernet)介紹.....	7
2-3 整合 CAN 與 Ethernet 的網路控制系統.....	9
2-3-1 網路實驗平台.....	9
2-3-2 TCP/IP 與 CAN 封包轉換.....	10
第三章 網路控制系統.....	13
3-1 網路控制系統簡介.....	13
3-2 取樣週期、時間延遲、資料遺失探討.....	14
3-2-1 時間延遲.....	14
3-2-2 資料遺失.....	16
3-3 Model-free Perfect Delay Compensation scheme[18].....	17
3-3-1 PDC 架構介紹.....	17
3-3-2 PDC 模擬與實驗結果.....	19
3-4 資料估測.....	20
3-4-1 網路控制系統的封包傳送.....	20
3-4-2 資料遺失與 One-delay 估測.....	22
3-4-3 最小平方法估測.....	23

3-4-4	PDC 架構的命令估測分析 .....	24
3-4-5	PDC 架構結合最小平方估測實驗 .....	25
第四章	雙軸同動協調 .....	28
4-1	雙軸的網路控制系統 .....	28
4-2	雙軸同步 .....	29
4-2-1	等待同步機制 .....	29
4-2-2	等待同步機制模擬 .....	30
4-3	交叉耦合控制器用於網路遠端控制系統 .....	31
4-3-1	交叉耦合控制器(cross-coupled controller , CCC) .....	31
4-3-2	交叉耦合控制器圓路徑(circular Contour) .....	31
4-3-3	交叉耦合控制器結合 PDC 應用於網路控制 .....	32
4-4	結合的雙軸協調機制 .....	36
4-4-1	CCC 異質性網路循圓 .....	36
4-4-2	結合架構 .....	36
第五章	異質性網路循圓實驗結果 .....	40
5-1	異質性網路誤差改善流程 .....	40
5-2	實驗結果 .....	41
5-2-1	Kp 調整 .....	41
5-2-2	各軸加入 PDC .....	42
5-2-3	測量 RTT，延遲等待協調雙軸 .....	44
5-2-4	Cross Coupled Control .....	46
5-3	遠端控制實驗 .....	49
5-4	小結 .....	51
第六章	結論與未來發展 .....	52
6-1	結論 .....	52
6-2	未來發展 .....	53
參考文獻	.....	54

## 表 目 錄

表 4.1 CCC 加入前後模擬 .....	38
表 5.1：各延遲時間同步比較 .....	44
表 5.2：CCC 加入前後比較(無同步) .....	46
表 5.3：CCC 加入前後比較(同步後) .....	48
表 5.4：遠端實驗結果 .....	50



## 圖目錄

圖 2.1 實驗系統架構圖 .....	10
圖 2.2 Server 架構圖 .....	11
圖 2.3 封包轉換圖 .....	12
圖 3.1 網路控制系統方塊圖 .....	14
圖 3.2 延遲時間對系統影響圖(delay time = 200 ms) .....	15
圖 3.3 延遲時間對系統影響圖(delay time = 1000 ms) .....	15
圖 3.4 (a)事件觸發(event-trigger)示意圖 (b)時間觸發(time-trigger)示意圖 .....	16
圖 3.5 資料遺失示意圖 .....	17
圖 3.6 PDC 應用於網路控制系統 .....	18
圖 3.7 網路延遲 400ms, PDC 加入前後的 step response .....	19
圖 3.8 網路延遲 400ms, PDC 加入前後的 sine response .....	20
圖 3.9 時間延遲測量實例 (a)RTT 測量架構 (b)不同延遲時間造成的傳輸情形 .....	22
圖 3.10 交大內部網路到校外網路的 RTT 時間 .....	22
圖 3.11 One-delay 處理時間延遲的響應 .....	23
圖 3.12 閉迴路網路控制系統 .....	24
圖 3.13 加入 PDC 的網路控制系統 .....	25
圖 3.14 延遲時間對馬達位置控制影響實驗圖(One-delay 估測) .....	26
圖 3.15 遲時間對馬達位置控制影響實驗圖(最小平方估測) .....	27
圖 4.1 不同時間延遲的雙軸命令與響應結果圖 .....	28
圖 4.2 異質性網路雙軸實驗架構 .....	29
圖 4.3 等待同步機制 .....	29
圖 4.4 (a) 等待同步機制加入前後的 XY 響應圖 .....	30
圖 4.5 可變增益交叉耦合控制器結構圖 .....	31
圖 4.6 圓路徑之輪廓誤差 .....	32
圖 4.7 交叉耦合控制器的運動控制系統架構 .....	33
圖 4.8 補償位置提前到增益之前並加上網路傳輸 .....	34
圖 4.9 CCC+PDC 的遠端控制系統 .....	35
圖 4.10 根據雙軸延遲差距補償的 CCC 架構 .....	37
圖 4.11 CCC 加入前後循圓模擬圖 .....	38
圖 4.12: 同步+CCC 模擬圖(x30) (a)延遲 50ms (b)延遲 100ms (c)延遲 200ms (d)延遲 400ms .....	39
圖 5.1 改善流程圖 .....	40

圖 5.2(a) $K_{px} = K_{py} = 0.0001$ (b) $K_{px} = 0.0001, K_{py} = 0.00023$ (c) $K_{px} = K_{py} = 0.00023$ .....	42
圖 5.3 加 PDC 前後 X 軸響應 .....	43
圖 5.4 加 PDC 前後 Y 軸響應 .....	43
圖 5.5 加 PDC 前後循圓結果 .....	43
圖 5.6: 同步前後比較圖 .....	44
圖 5.7: 一天不同時刻新竹到台中傳輸延遲時間紀錄 .....	45
圖 5.8: 固定補償與即時補償結果比較 (a) 固定補償 (b) 即時補償 .....	45
圖 5.9: CCC 加入前後比較(無同步) (a) 延遲時間差 50ms (b) 延遲時間差 100ms (c) 延遲時間差 200ms .....	47
圖 5.10: 同步後, CCC 加入前後比較(x30) (a) 延遲時間差 50ms (b) 延遲時間差 100ms (c) 延遲時間差 200ms .....	49
圖 5.11: 遠端實驗架構圖 .....	49
圖 5.12: 台中 $\longleftrightarrow$ 新竹延遲時間 .....	50
圖 5.13: 遠端實驗結果 (a) 兩軸加入 PDC (b) 兩軸同步 (c) 兩軸同步+CCC	51



# 第一章 緒論

## 1-1 研究動機

近年來將運算、通訊和控制單元，透過網路作整合，以方便系統的建立與管理，進而提高系統的整合度與便利性，乃是控制系統的一大趨勢。也因為網路化的日益普及化，各種應用於不同領域的網路協定也因應而生，常見於工業應用的協定如：WorldFIP、Profibus、SERCOS、CAN 等；其他如 Ethernet、無線網路等協定，也廣泛使用於各種應用上，特別是無線網路已是未來網路通訊的發展重點。

利用網路做為運算、通訊和控制單元的媒介，可以在遙遠的距離內控制與監控特定的系統，如果使用無線網路，更可以節省佈線的複雜度與成本，且具有高機動性，所以網路控制系統具有節省配線、系統擴充性增加、減少安裝和維護成本、方便監控等優點。但也有許多所要面臨的問題，首先便是網路傳輸所帶來的影響(time delay)、網路引起的訊息傳遞延遲與訊息遺失(data dropout)、訊息排程、節點間的時脈不同步。這些種種問題，任何一個均可能讓控制效能降低，甚至讓受控體變成不可控的狀況，如何去克服網路傳輸所引起的問題並且發揮遠端網路控制的優點，變成為研究的重點之一。

另外有鑑於網路的種類越來越多，各種通訊協定更是不斷地推陳出新的情況下，不同網路的使用便會成為一種必然的現象，因此我們將討論一個雙軸的網路控制系統當中，當兩軸所使用的網路有差異時，如何去對兩軸協調與同步，達到良好的循環結果。

## 1-2 研究背景與發展概況

隨著網路的蓬勃發展，加上近年來最熱門的領域-無線網路的研究，讓許多學者與研究人員紛紛投入網路通訊這領域，讓網路概念的延伸與應用處

處可見，更可以將網路的概念應用在控制理論上，因此網路控制系統的想法便因應而生，而針對網路控制系統所相關的研究課題與問題也紛紛提出[1]。一般網路控制系統，雖然有網路化的優點，但往往需要犧牲一點控制效能，以下針對一些足以影響控制系統效能的問題，整理後可歸類成下列幾項：

- 網路傳輸所造成的延遲(time delay)

一般的理想控制系統，訊號在傳輸時，其傳輸時所花費的時間極小，通常可以忽略不計，但當控制系統利用網路做為傳輸介面時，由於網路先天上的頻寬限制與網路通訊協定限制下，造成訊號有一段不可忽略的傳輸延遲時間[2,3]。所以在系統的等效模型中，需加入一個時間延遲項，而此時間延遲會造成系統的 phase margin 下降，導致系統不穩定，現在在此方面，已有相關理論解決延遲時間對系統造成的不良影響[4,5]。

- 網路引起的訊息遺失(data dropout)

網路在傳輸的過程中，有可能會發生封包遺失的情形，一般來說，大部分的網路通訊協定都有防止與解決封包遺失的機制，通常是避免資料封包碰撞(collision)以及封包遺失後等候節點再一次重傳，資料封包的遺失，會造成控制系統的命令遺失，等候節點再一次重傳的時間會增加延遲時間，有人提出使用泰勒展開法來估測遺失的命令，使得系統不會因為命令遺失而造成系統控制效果不佳[6]。

- 訊息排程(network scheduling)

在網路排程方面，有人提出 adjustable priority scheme 的排程方法 [7,8]，藉由累積 CAN 網路上訊息一段時間的傳輸狀況後，逐漸調整訊息優先權，經由此方法改變網路的排成可以有效的提昇網路傳輸的效率。

- 節點間的時脈不同步(jitter)

在系統取樣週期較小的架構下，時脈漂移造成節點間的時脈不同步，將導致大量的系統資料遺失量，由此可知，時脈同步對於網路系統是不可忽略的因素。關於網路時脈同步的方式，可由系統架構、網路協定等層面解決[9]，以系統架構來說，可利用額外的硬體接線作為同步線路，以此作為節點執行的基準；在網路協定層面，以CAN bus 為例，透過同步物件的傳送，提供整體網路的基準時脈。這兩類作法各有其優缺點，前者為獨立於原本網路架構之外的作法，可免除網路傳輸的干擾，且經由硬體線路可達到極高的準確度，但正因為是獨立於網路之外，所以必須在系統建構時加以考量，若要應用於已建立的系統之上，就必須重新規劃系統架構；後者則是基於原先的網路協定所衍生的方式，主要優勢在於可經由軟體規範達成而不需改變系統架構，所以，實做上的可行性較高且提高系統維護的便利性，但由於是透過網路封包傳遞的方式，傳輸狀況的影響就無法忽略，傳輸速率、延遲時間和封包碰撞等等因素，都可能影響其效果而必須納入考量，以CAN bus 為例，由於封包碰撞可由CAN 的重傳機制避免，而將主要考量放在傳輸延遲上，因此，賦予同步物件極高的網路優先權，降低傳輸的等待時間，便於同步物件順利傳送，以達到較好的同步效果[10]。

- 取樣週期(sampling period)

在數位控制系統下，控制效能會隨著取樣週期變小而變好，但將網路加入控制系統當中，取樣週期則不可以小於某個極限值，如果小於該極限值，反而控制效能會下降[11]，主要原因是因為當取樣週期變小的情況下，所要傳輸的資料量變多，而當網路的頻寬不足以傳輸這些資料量時，便會造成網路塞車、負載量增大，所需要傳輸的資料，必須等待

網路允許通過，方可傳輸，造成傳輸時間延遲，甚至資料損失，導致控制效能下降。有人提出利用動態調整取樣頻率的方法[12,13]，當網路負載大時，取樣週期下降，減少傳輸資料量，有人提出以動態調整取樣頻率為基礎，其適應性控制器的設計[14]。

- QoS(quality of service)[15]

此因素會直接影響到網路控制系統的控制效能，其特性主要受訊息傳輸延遲時間、訊息遺失率、訊息來源的正確性所影響，並非一個可量化的參數。

### 1-3 問題描述

1. 在網路控制系統當中，當網路傳輸延遲超過系統取樣週期，控制效能便受到影響，嚴重時系統會不穩定。
2. 資料封包的遺失，造成控制系統命令的遺失。
3. 在遠端控制系統進行雙軸以上的網路控制，並且兩軸是使用特性不同的網路時，在循軌跡時便發生兩軸不同步。

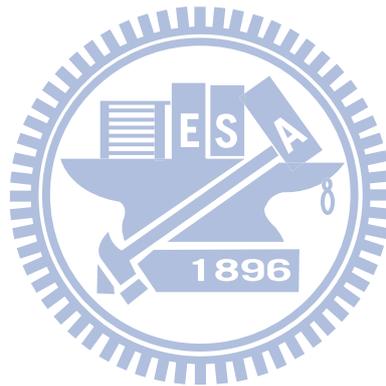
### 1-4 研究方法

1. 在時間延遲的處理方面，以 Perfect delay compensation 將時間延遲對系統的影響減低為單純時間延遲，避免系統因網路延遲影響而發散。使用最小平方估測取代 one-delay 處理資料遺失問題，利用先前數筆的資料來做多項式的逼近，估測數值。
2. 在改善資料封包遺失造成控制命令遺失方面，以最小平方估測取代 one-delay 法，以較多先前資料進行 N 次方多項式曲線逼近，能夠有效改善 one-delay 直接沿用前筆資料而估測誤差較大的問題。
3. 在改善異質性網路循軌跡的同動上，分別測量兩軸 RTT 時間，並加入

延遲等待的 buffer 讓兩軸在時間上同步，接著利用 CCC 進一步改善輪廓誤差。

## 1-5 論文架構

本論文共分六章，第一章說明研究動機、目的及研究方法。第二章介紹 CAN、Ethernet 與 TCP/IP 的協定。第三章介紹 PDC 與最小平方估測解決網路延遲對系統的影響。第四章介紹異質性網路的雙軸網路控制系統。第五章為實驗結果與討論。第六章則為本論文的結論以及未來的發展方向。



## 第二章 CAN、Ethernet 與實驗系統架構

### 2-1 CAN (Controller Area Network)[16]

#### CAN網路協定簡介

CAN 全名為 Controller Area Network，起源於 1980 年代後期，由德國 Robert Bosch 公司所制定的一種具有高度安全且支援即時分散式控制的通訊協定，最高傳輸速度可達 1M bit per second (bps)。到 1990 年代其通訊協定逐漸成型，相關的 IC 元件紛紛被推出，才被廣泛地應用於工業界。CAN 原先是應用在汽車電子系統，用來連接汽車內防鎖死煞車系統(ABS)或是引擎控制與感測單元等的電子元件，以簡化車內複雜的硬體配線；但由於 CAN 提供可靠快速的資料傳遞，適合用在即時系統(real-time system)且價格低廉，因此，發展至今也被廣泛運用在各類的控制系統上。目前 CAN 已成為國際標準規格 (ISO11898)，CAN 在場域匯流排 (field bus) 的應用已有 Honeywell 的 SDS 及 Allen-Bradley 的 DeviceNet。

CAN 根據其傳輸協定，具有以下特點：

- 資料訊息具有優先權 (priority)
- 優先權的仲裁 (arbitration) 為非破壞性
- 保證延遲時間
- 彈性的架構
- 採用廣播的方式 (multicast)，並藉由傳輸的訊息做時序同步的動作
- 任一節點皆可主動發出訊息 (multimaster)
- 未傳送成功的訊息會自動重新傳送
- 錯誤檢查

- 區分暫時性錯誤和永久性錯誤並且自動關閉有問題的節點

## 2-2 乙太網路(Ethernet)介紹

乙太網路(Ethernet)源自於 Xerox 公司的區域網路系統，一開始速度只有 2.49Mbps，僅在 Xerox 公司內部使用，隨後由 DEC、Intel 和 Xerox 三家公司共同參與標準之改進與擴展，並且發表了 Ethernet Version 2(EV2)規格，將網路頻寬提升到 10Mbps，之後由 IEEE 根據 EV2 的內容，在 1983 年通過了 802.3 CSMA/CD 規格，從此成為使用最廣泛的區域網路標準之一。

乙太網路使用 CSMA/CD(載波感應多重存取及衝突偵測)方式來存取介質上面的資料，在 CSMA/CD 裡，某一站要送資料前必須檢查是否別有別的站正在傳送資料，假如沒有其他站要傳送資料，則這一監聽站就可以將資料傳送到網路上面。在乙太網路上面可能會遇到多站同時要傳遞資料的情況，此情況會造成資料的碰撞。因此所有的站必須持續監聽網路及偵測出任何可能的碰撞。若產生碰撞，所有站放棄所接收到的資料，原先傳送的各站退後等待一段時間再重送。為了降低再次碰撞的可能，每個傳送站各自產生一個亂數來決定要等待多久時間才能傳送。

乙太網路在實際使用上可以為匯流排或是星形方式連接，所有乙太網路的組態在邏輯上是一種匯流排。每個訊框(Frame)可傳送資料到網路線上的各站，但只有被定址到的那個站才能讀到訊框。在實現上目前使用最多的實現方式為 10BASE-T(雙絞線乙太網路)，10BASE-T 是使用星形方式連接各站，速度為 10Mbps，最長傳輸距離為 100 公尺。除了 10BASE-T 之外，還有 10BASE5(粗纜乙太網路)和 10BASE2(細纜乙太網路)，上述兩種均使用匯流排傳輸方式做連接。

目前使用上比較普遍的是快速乙太網路(Fast Ethernet)，快速乙太網路是比乙太網路更新一點的版本，最高速率為 100Mbps 跟乙太網路比較起來，在訊框上沒有做任何改變，唯一改變的是傳輸速度和碰撞區間的大小，傳輸速度提高 10 倍，因此碰撞區間也降低 10 倍。快速乙太網路和 10BASE-T 一樣採用星狀連接方式，其規範定義三種實體介質：100BASE-T4、100BASE-XT 和 100BASE-XF。

至於 Gigabit 網路速度 1000 Mbps，就是 Gigabit Ethernet，只是 Gigabit Ethernet 的網路線就需要更加的精良。當傳輸速度增加時，線材的電磁效應相互干擾會增強，因此在網路線的製作時就得需要特別注意線材的質料以及內部線蕊之間的纏繞情況配置等，以使電子流之間的電磁干擾降到最小，才能使傳輸速度提升到應有的 Gigabit。因此現在高速網路的傳輸介質逐漸的被光纖所取代。

## TCP/IP 通訊協定[17]

TCP/IP 主要是用來連結網路上的電腦主機，作為網路傳輸資料的標準協定，是一套完整的通訊協定，他的名稱來自於其中兩個最重要的協定：傳輸控制協定(transmission control protocol, TCP)及網際網路協定(internet protocol, IP)，除了這兩種協定外，TCP/IP 還包含了其他協定，但最重要的就是 TCP 和 IP 兩種協定。

TCP/IP 是一種階層式的協定，這裡所謂的階層式是指每一個較高的協定是由一個或多個較低分層協定所支援。這裡我們主要介紹網路層的網際網路協定(IP)和傳輸層的傳輸控制層協定(TCP)。

### (a) 網際網路協定(IP)

IP 是 TCP/IP 使用的傳輸機制，是一種非可靠性、非預接式的資料封包協定，只提供盡量傳送的服務。所謂盡量傳送是指 IP 沒有提供錯誤檢查

或追蹤，IP 假設他的底層是不可靠的而盡力將資料傳送到目的地，但不保證是否傳到目的地。IP 以資料封包方式傳遞，而各個資料封包分別傳送，資料封包在傳輸過程可以經過不同的路徑，可以不按順序到達，也可以被重複，當資料封包達目的時，也沒有方法將資料封包調整回原來的順序。然而 IP 的限制不應被視為缺點，IP 提供骨幹傳輸的功能，且讓使用者自由加入所需要的功能，因而獲得較高的效率。

### (b) 傳輸控制協定(TCP)

TCP 提供完整的傳輸服務給所需的應用程式，在應用層與網路層之間提供應用程式和網路運作的中介服務。TCP 在傳輸層中提供了流量控制和錯誤控制的機制，TCP 使用滑動窗口(sliding window)的協定來做流量控制，使用回應封包、計時(time-out)、重送的機制來做錯誤控制。

TCP 為應用程式提供連線的機制，應用程式能夠以資料流方式傳送訊息給傳輸層，在傳送端的傳輸層負責與接收者建立一個連線後，將資料拆成可傳送的單元並給予序號，再將傳送單元一個一個傳送出去。接收端的傳輸層等待這些資料的到來，然後加以檢查，並將沒有錯誤的資料以資料流的方式傳給接收端的應用程式，等全部資料完成後，TCP 要關掉剛剛使用的連線。TCP 是一種連線導向，具可靠的傳輸協定。TCP 把 IP 的服務加入連接導向及可靠性這兩種特點。

## 2-3 整合CAN與Ethernet的網路控制系統

### 2-3-1 網路實驗平台

圖 2.1 為這次實驗的系統架構圖，此架構由三個部份所組成，第一個部份為 TI 的 F2812 DSP 和 Panasonic AC 伺服馬達所組成，馬達工作模式調整為速度模式，第二部份為一台電腦和 USB CAN 所組成的 Server 端，Server 端是連結 CAN 網路和乙太網路的溝通橋樑，具有閘道器(gateway)的功用，並且具有

簡單的管理功能，第三部份為負責網路控制的電腦。DSP 跟 Server 端的 USBCAN 之間的聯繫是透過 CAN 來聯結，Server 端和負責遠端控制的 Client 端中間透過乙太網路來聯結，資料傳輸的協定則是使用 TCP/IP。整個控制迴路為位置迴路控制。

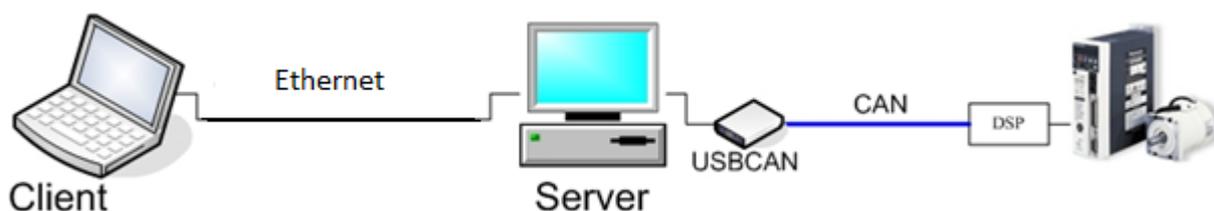


圖 2.1 實驗系統架構圖

由於本實驗系統是透過網路來控制，DSP 端採用時間觸發方式，而 Client 則是採用事件觸發的方式，DSP 每隔固定的時間會將馬達 Encoder 讀到的資料回傳給 Client，當 Client 偵測是否接收到 DSP 傳過來的回授訊號，接收到訊號才經由控制器計算出命令回傳給 DSP。

### 2-3-2 TCP/IP與CAN封包轉換

由於 DSP 上面只有 CAN 介面，而遠端控制是使用無線網路透過 TCP/IP 通訊協定來傳遞資料，兩種不同協定的網路無法直接做溝通，所以我們中間需要透過閘道器來轉換兩種不同型式的封包。

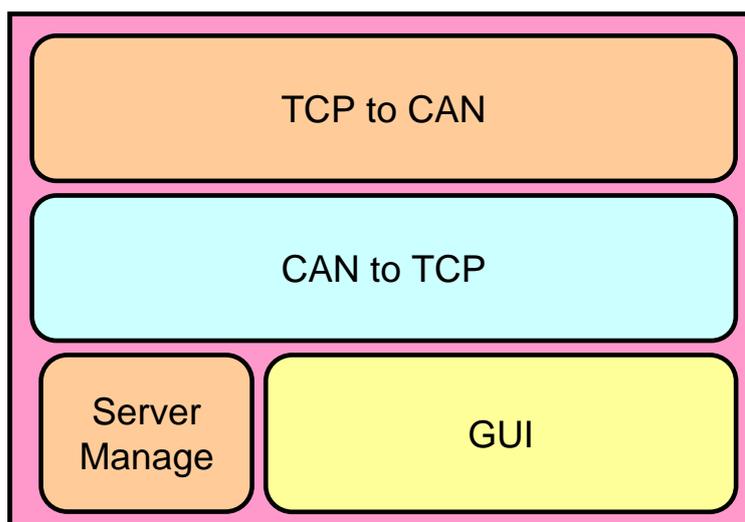


圖 2.2 Server 架構圖

圖 2.2 為 Server 的架構圖，Server 主要分成四個部份，第一部份是由 CAN 網路接受 CAN 封包，然後將該封包轉成 TCP/IP 格式的封包透過 Ethernet，將封包送到遠端的控制端，第二部份為由 Ethernet 接收到 TCP/IP 封包，轉成 CAN 的封包然後經由 CAN 網路將資料傳給 DSP。第三部份為 Server 管理程式，Server 一次僅允許一台 Client 連結到 Server，以防止在做網路控制時，還有其他 Client 端進來干擾，嚴重將使系統不穩定，當已經有一台 Client 已經建立好連線，此時若有其他 Client 嘗試連線，Server 會拒絕該 Client 的連線並且給予警告的訊息。第四部份則為使用者介面。

當有資料要從 DSP 傳送到 Client 的控制端，首先將訊息種類(Type)及傳送資料(Data)的訊息格式設定好，設定好訊息格式之後將他放到 CAN 封包的 Data 部份透過 CAN 網路傳到 Server，此步驟如圖 2.3 的(1)。當 Server 接收到 CAN 送過來的封包之後，將整個 CAN 封包放到 TCP 封包的 Data 部份，再利用 Ethernet 傳送到 Client 端，此步驟如圖 2.3 的(2)。Client 收到由無線網路送過來的封包後，先從 TCP 封包 Data 部份取出 CAN 的封包，再由 CAN 封包 Data 部份取出使用者自己定義的資料格式，最後再分析資料格式即可完成 DSP 傳送到 Client 的工作，如圖 2.3 的(3)。照著圖 9 中(1)→(2)→(3)的順序即可將訊息完整的傳送到 Client 控制端。

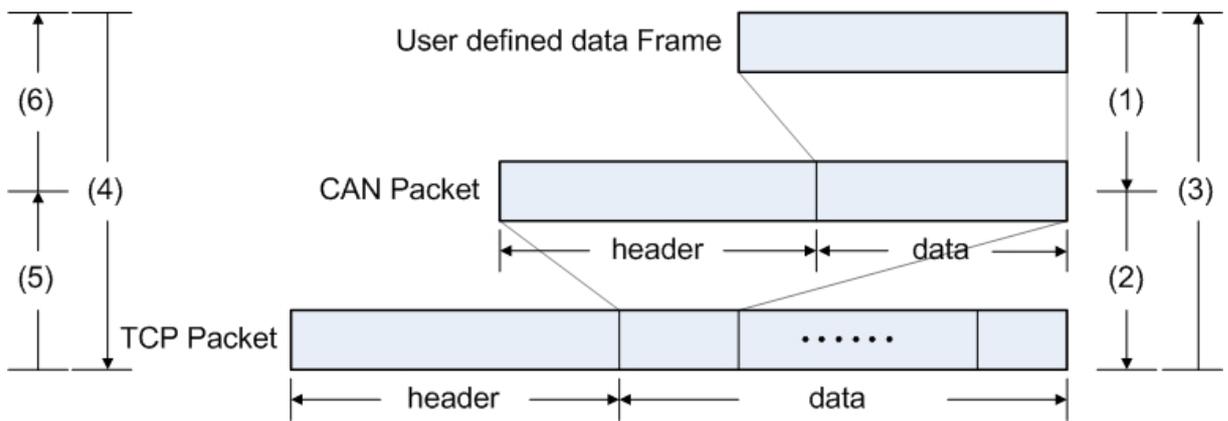


圖 2.3 封包轉換圖

若要將資料由 Client 控制端傳送到 DSP，先將要傳送的訊息種類(Type)和傳送資料(Data)訊息格式設定好，再放到 CAN 封包 Data 部份，最後把 CAN 封包整個放到 TCP 封包的 Data 部份，利用無線網路將封包傳送出去，如圖 2.3 的(4)。Server 收到封包之後，從 TCP 封包 Data 部份取出 CAN 封包，將 CAN 封包透過 CAN 網路將封包傳送到 DSP，如圖 2.3 的(5)。DSP 收到 CAN 封包後，將訊息格式由 CAN 封包 Data 部份取出，再進行解析，如圖 2.3 的(6)。只要照著圖 2.3 中(4)→(5)→(6)的訊息即可完成 Client 控制端傳送訊息到 DSP 的動作。

## 第三章 網路控制系統

### 3-1 網路控制系統簡介

回授控制系統之中的控制迴路及回授迴路，以網路來做傳輸媒介，則將此類控制系統稱之為網路控制系統(networked control systems, NCSs)，所使用的網路可以是有線網路如 Ethernet，無線網路如 IEEE802.11，或是工業用網路如 CAN。透過網路來做控制，有許多優點，例如：

- (1) 節省配線：一般的控制系統其配線複雜，不同訊號、元件、受控端各別需要不同線路傳輸、連結，但網路控制系統只需備妥網路環境，所有資訊封裝在封包內傳輸，即可將不同的受控端連結在一起，網路上各個元件不需在個別配線回到控制端，就可以將資訊傳回主控端，達到配線簡化、效率化。
- (2) 模組化：每個連接在網路上的控制端或是受控端都可以被視為物件，在一個系統裡面要增減一個物件，只需在軟體做設定即可。
- (3) 故障易於診斷及排除：網路化的元件，可在軟體上設定自我診斷的功能；當元件故障，元件能夠透過網路回傳故障訊息給主控台，無需個別元件做檢查，提高維修時的效率與便利性。
- (4) 遠端控制：增加控制的距離，可以使操作人員遠離危險環境，在遠端即可控制機器。

由上面可以知道，網路控制系統可以為我們帶來許多便利，但是將控制系統網路化之後，也帶來幾個缺點，例如時間延遲(delay time)，資料遺失(data dropout)，封包碰撞，取樣週期降低，網路排程(network scheduling)等問題，這些問題沒有處理好，輕則降低系統效能，重則整個系統產生不穩定或是無法控制，這些問題都是網路控制上的重要課題。

## 3-2 取樣週期、時間延遲、資料遺失探討

### 3-2-1 時間延遲

圖 3.11 為一般網路控制方塊圖，在控制端與受控端之間的命令和回授訊號的傳輸是透過各種網路來傳輸，由網路來當作傳輸媒介就會產生延遲時間(delay time)。網路控制所產生的時間延遲分成兩類，一種為命令延遲時間(command time delay)是由控制端將命令傳到受控端的所產生的延遲時間，另外一種則是回授延遲時間( feedback time delay) ，回授延遲時間是指受控端將感測器的資料傳回控制端所產生的延遲時間。

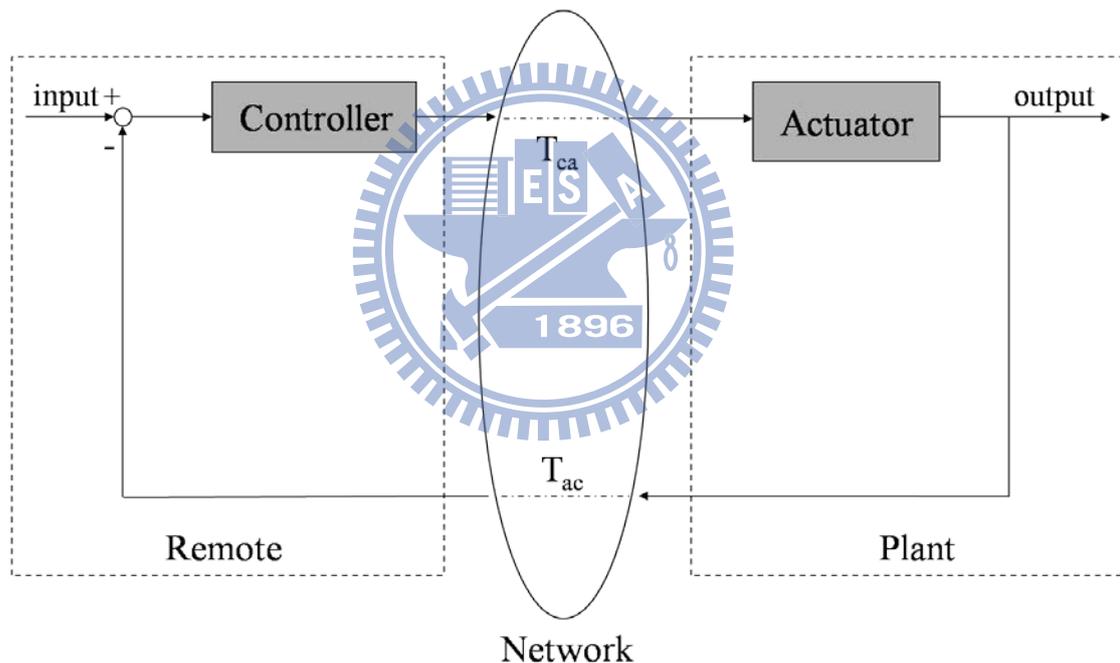


圖 3.1 網路控制系統方塊圖

延遲時間若小於系統的取樣週期，則我們可以將延遲時間忽略掉，但是當延遲時間大於取樣週期，此時延遲時間對整個控制系統的影響，輕則會降低系統的效能，嚴重則會造成整個控制系統的不穩定。圖 3.22 和圖 3.3 分別為實際的網路控制系統在取樣頻率為  $20\text{ ms}$  時的響應圖，兩張圖中虛線部份是不經過網路控制的系統跑來的響應，實線部份則是經過網路來做控制的響應圖，圖 3.22

的延遲時間為  $200\text{ ms}$ ，此時響應圖與未受網路控制做比較，可以看出因延遲時間造成的系統震盪，若我們再將時間延遲加大到  $1000\text{ ms}$ ，整個網路控制系統已經發散不受控制。

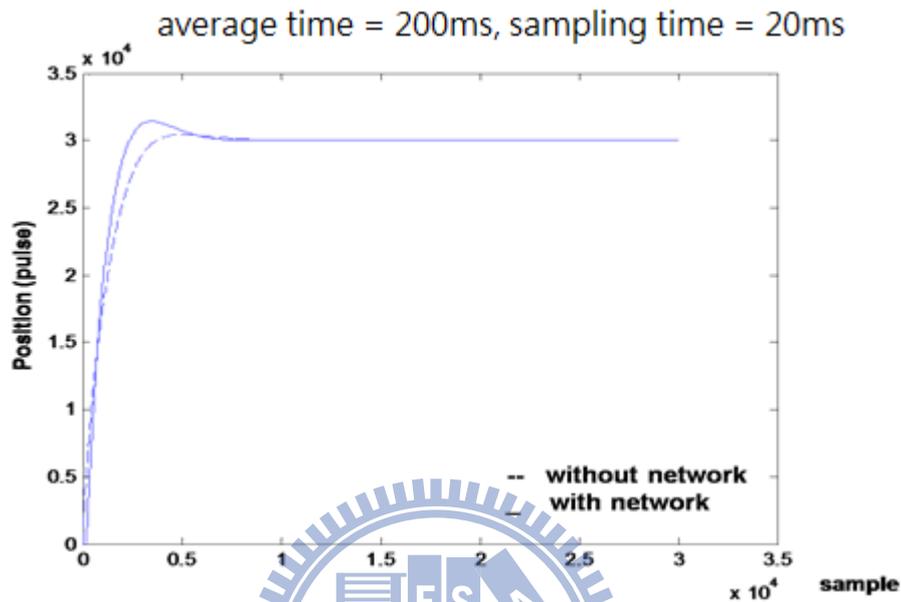


圖 3.2 延遲時間對系統影響圖(delay time = 200 ms)

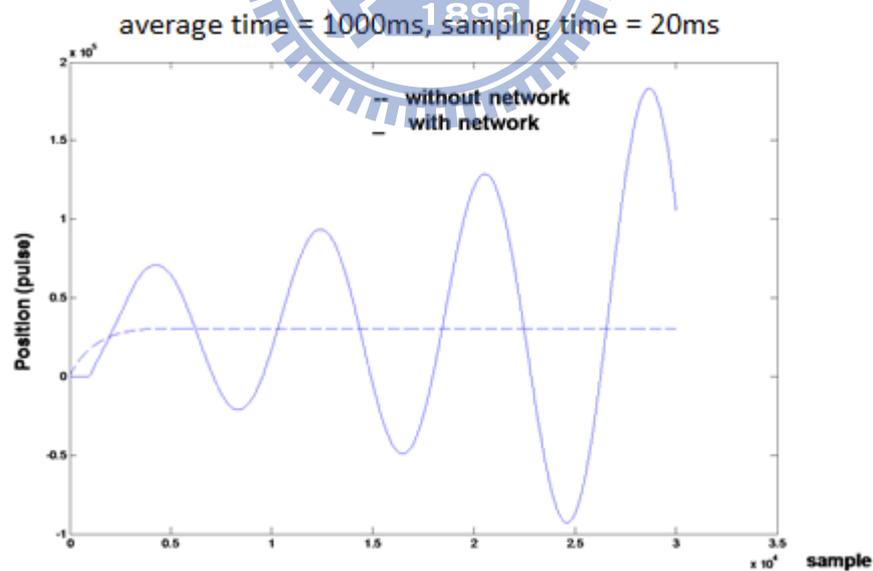


圖 3.3 延遲時間對系統影響圖(delay time = 1000 ms)

由上面兩張圖可以看出，在網路控制系統若不將延遲時間做妥善處理，則會對網路控制系統造成無法控制和降低效能。

### 3-2-2 資料遺失

一般傳統的網路控制系統其回授訊號的傳遞多是採用事件觸發(event-trigger)，即受控端根據感測器的資料是否傳回控制端做為觸發依據，所以可視為是每間隔一時間延遲，受控端回傳感測器資料到控制端，如圖 3.14(a)所示，其優點是架構簡單、無須多次回授訊號給控制端處理，以避免網路壅塞，所以多半用在低精確度控制或開迴路控制，但其缺點是無法及時回授感測器資料，應用於需要精密控制的領域上，如運動控制、CNC 伺服控制會有取樣點數不夠多，讓誤差過大，導致控制效能低落的問題。為了解決此問題，本研究採用時間觸發(time-trigger)，每一固定取樣週期回傳回授訊號給控制端做處理，如圖 3.14(b)所示，相較於事件觸發，具有高取樣頻率且閉迴路即時控制的優點。

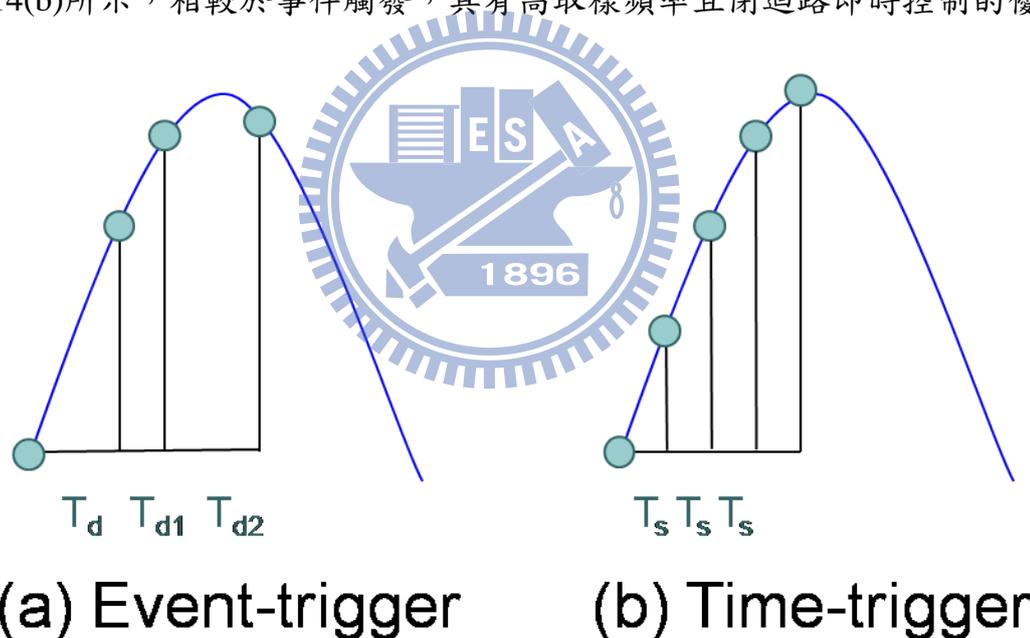


圖 3.4 (a)事件觸發(event-trigger)示意圖 (b)時間觸發(time-trigger)示意圖

當在固定的取樣週期下，因為時間延遲遠大於取樣週期，導致無感測器資料可被接收，此現象我們定義為資料遺失(data dropout)，如圖 3.5 所示。

當網路控制系統應用於精密控制，如運動控制時，需要高取樣頻率以達到高精確度，若使用的網路環境為低時間延遲的工業用網路如 CAN，則少有時間

延遲遠大於取樣週期的現象發生，如同閉迴路控制一般，回授訊號可以被及時回授處理，使輸出誤差減少，改善系統的控制效能。

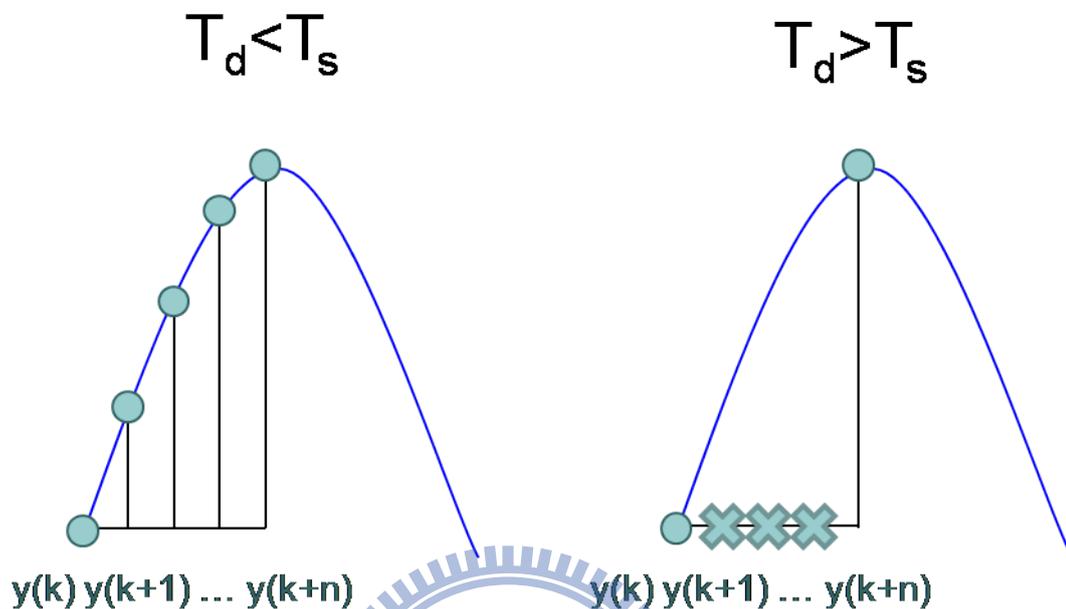


圖 3.5 資料遺失示意圖

### 3-3 Model-free Perfect Delay Compensation scheme[18]

#### 3-3-1 PDC架構介紹

大部分的網路控制系統在設計控制器時，都假設網路延遲時間為一固定項，但是實際上網路延遲時間會變動，造成相位的落差以及系統的不穩定。因此，本研究使用了延遲時間完全補償機制 Perfect delay compensation(PDC)，來對延遲時間造成的問題進行處理。PDC 在使用時不需要鑑別系統模型，甚至不需要知道網路延遲時間，就能夠將網路延遲時間對系統的影響轉變為一單純的時間延遲，相當有效且方便。

圖 3.6 是 PDC 應用於一般網路控制系統的架構圖，在此推導 PDC 的結果。

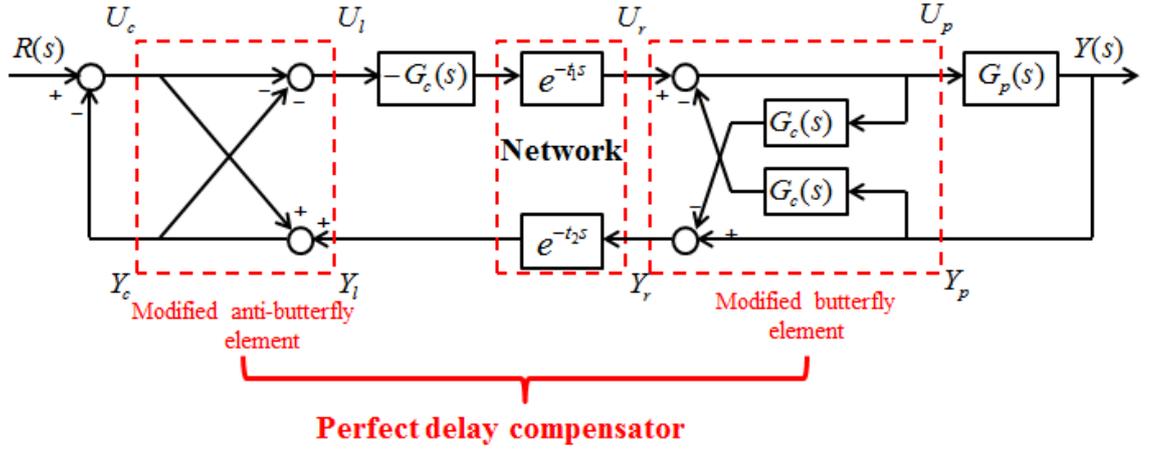


圖 3.6 PDC 應用於網路控制系統

首先，從圖 3.6 得知，modified butter element 的輸出結果可以被表示成

$$U_p(s) = U_r(s) - G_c(s)Y_p(s); Y_r(s) = -G_c(s)U_p(s) + Y_p(s) \quad (3-1)$$

接著， $U_r$  到  $Y_r$  的轉移函數可以表示成為

$$\frac{Y_r}{U_r} = \frac{-G_c(s)U_p(s) + Y_p(s)}{U_p(s) + G_c(s)Y_p(s)} = \frac{G_p(s) - G_c(s)}{1 + G_c(s)G_p(s)} \quad (3-2)$$

因為  $\frac{Y_p(s)}{U_p(s)} = G_p(s)$ ，故左側  $U_c$  到  $Y_c$  的轉移函數可表示為

$$\frac{Y_c(s)}{U_c(s)} = \frac{(1 + G_c(s)G_p(s)) + G_c(s)(G_p(s) - G_c(s)) \cdot e^{-(t_1+t_2)s}}{(1 + G_c(s)G_p(s)) - G_c(s)(G_p(s) - G_c(s)) \cdot e^{-(t_1+t_2)s}} \quad (3-3)$$

而系統的開迴路轉移函數可被表示為

$$\frac{Y_c(s)}{R(s)} = \frac{(1 + G_c(s)G_p(s)) - G_c(s)(G_p(s) - G_c(s)) \cdot e^{-(t_1+t_2)s}}{2 \cdot (1 + G_c(s)G_p(s))} \quad (3-4)$$

網路控制系統的閉迴路轉移函數則為

$$G_{NCS}(s) = \frac{Y(s)}{R(s)} = \frac{G_c(s)G_p(s) \cdot e^{-t_1 s}}{1 + G_c(s)G_p(s)} = G_o(s) \cdot e^{-t_1 s} \quad (3-5)$$

由(3-5)式得知，最後的結果原本無網路的閉迴路轉移函數與 $t_1$ 的延遲項相乘， $G_c$ 為原本系統的控制器可以直接使用，不需要因為加入網路的關係而另外作修。而回授延遲時間 $t_2$ 不影響輸出響應。

### 3-3-2 PDC模擬與實驗結果

在此小節用模擬與實驗來印證 PDC 的推導結果，首先我們假設命令延遲時間等於回授延遲時間， $t_1 = t_2 = (RTT / 2)$ ，系統鑑別結果得到

$$G_p = \frac{10^4(0.058s + 3.221)}{s(0.0001s^2 + 0.019s + 1)}$$

， $G_c(s)$  為 PI controller，其中  $K_p = 0.0013$ ，

$$K_i = 0.00000001$$

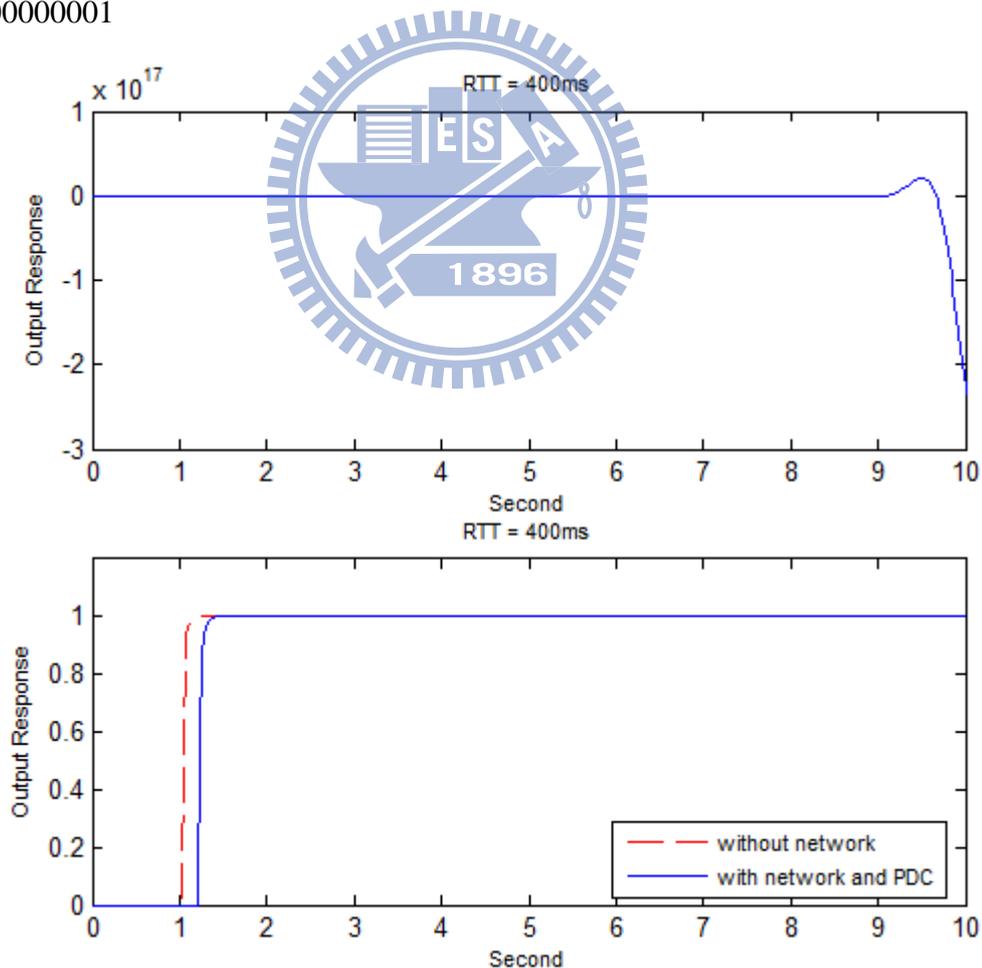


圖 3.7 網路延遲 400ms，PDC 加入前後的 step response

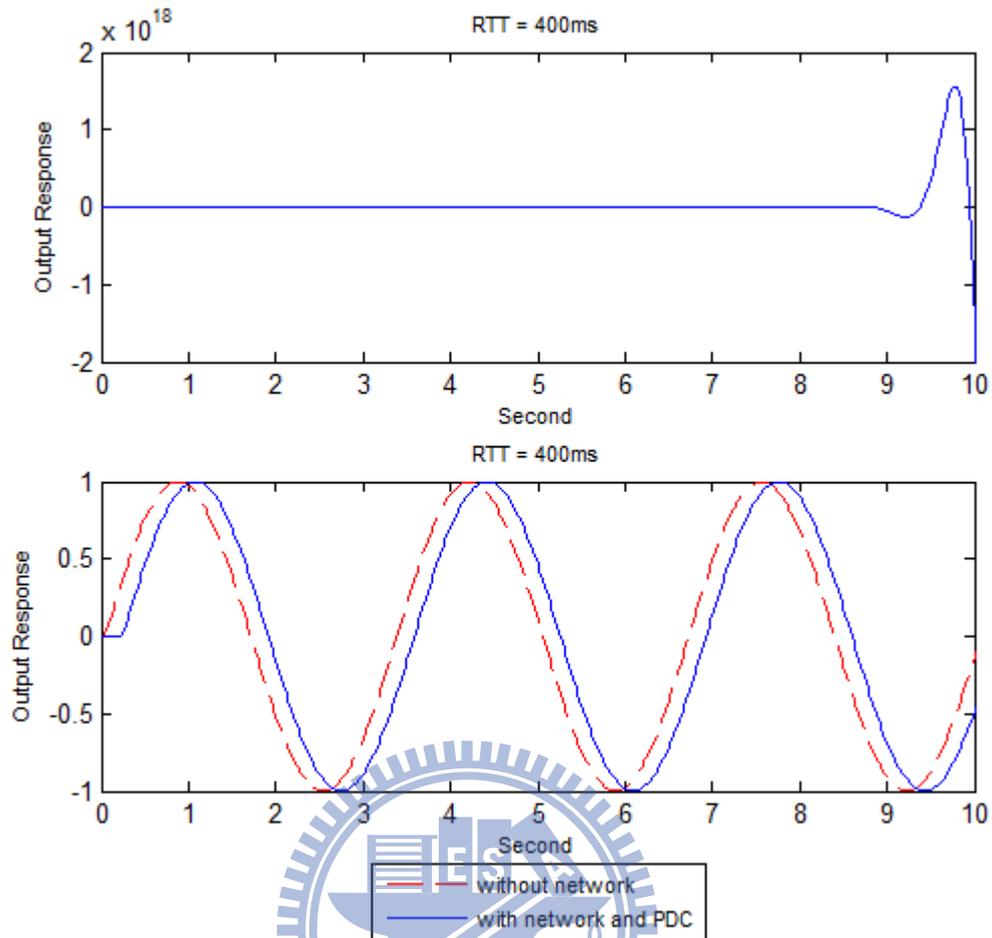


圖 3.8 網路延遲 400ms，PDC 加入前後的 sine response

從模擬以及實驗的結果得到，加了 PDC 的網路控制系統的響應為

$$G_o(s) \cdot e^{-t_1 s}。$$

## 3-4 資料估測

### 3-4-1 網路控制系統的封包傳送

為了了解網路控制系統封包傳送的情形，我們對 RTT 時間作測量並歸納各種延遲時間變化會造成的情形，網路控制系統的封包傳送如圖 3.9(a)所示。首先，由時間觸發的執行端將馬達資料傳到控制端，假設執行端到控制端的網路延遲為  $t_2$ ；事件觸發的控制端將收到的馬達資料處理後，得到控制訊號並傳回

執行端，假設控制端到執行端的網路延遲為  $t_1$ ，執行端藉由紀錄封包送出與最後回來的時間，即可得到  $RTT = t_1 + t_2$ ，接著以取樣時間 20ms 的訊息傳輸為例，並歸納各種網路延遲造成的封包傳出情形，如圖 3.9(b)所示。

(1) 正常傳輸(Normal transmission)：

當網路延遲小於取樣週期，延遲不對系統造成影響。如圖 3.9(b)第 1 筆。

(2) 空白取樣(Vacant sampling)：

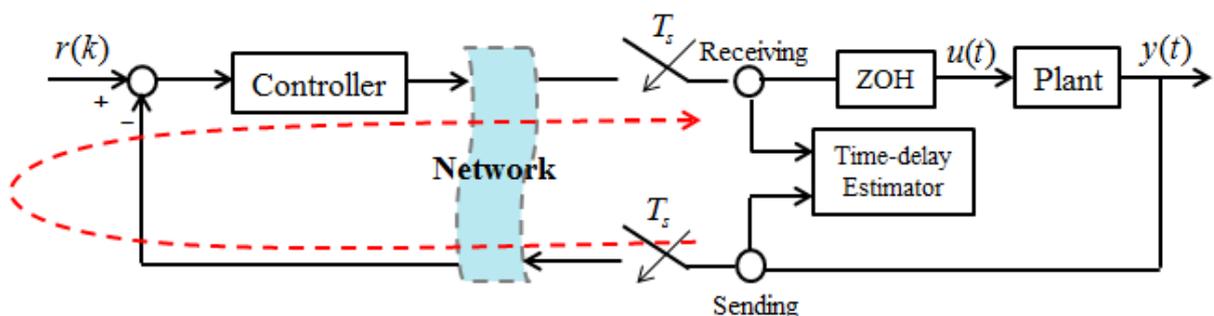
當資料傳輸的延遲時間超過了取樣週期，造成某些取樣週期內沒有收到資料(如圖 3.9(b)第 2、3 筆，時間 20~60ms 處)，空白取樣會造成響應的誤差增加，必要需加入資料估測改善。

(3) 資料覆蓋(Data rejection)：

當一個取樣週期內，接收到兩筆以上的資料，最後接收到的資料會覆蓋前面的。如圖 3.9(b)60~80ms 處，第二筆與第三筆同時到達，最後會使用第三筆的資料。

(4) 延遲傳輸(Delayed transmission)：

當網路延遲超過取樣週期，且變動不大，能夠在每一取樣週期內接收到一筆資料，稱作延遲傳輸。超過取樣週期時間的網路延遲會對系統的穩定性造成影響，可由史密斯估測器或 PDC 來解決。



(a)

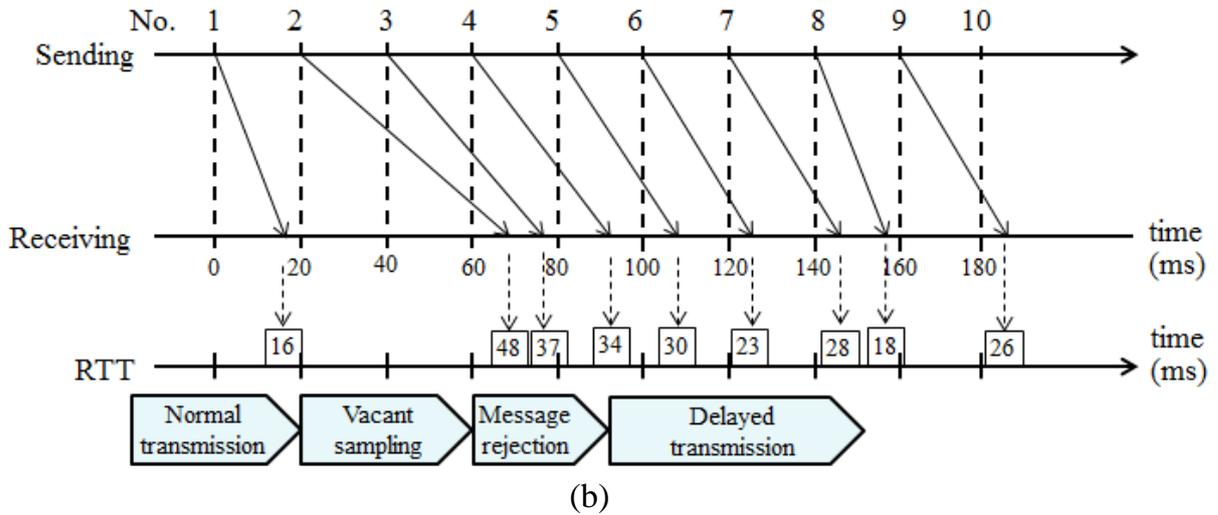


圖 3.9 時間延遲測量實例 (a)RTT 測量架構 (b)不同延遲時間造成的傳輸情形

### 3-4-2 資料遺失與One-delay估測

圖 3.10 為交大 816 研究室透過網路與校外住處連接而量測到的 RTT 時間，由圖中可以觀察到資料遺失重傳造成的延遲時間突然大幅增加情形，在原本的實驗架構當中，資料遺失情況發生時，直接沿用前一筆資料，因此循軌跡的結果會如圖 3.11 所示，當較大延遲時間發生時，產生連續的資料遺失，使用 One-delay 法一直沿用前筆資料響應結果與命令相差很多，之後發展到雙軸也會有不同動的情形。因此，欲使用最小平方估測來改善此種情形。

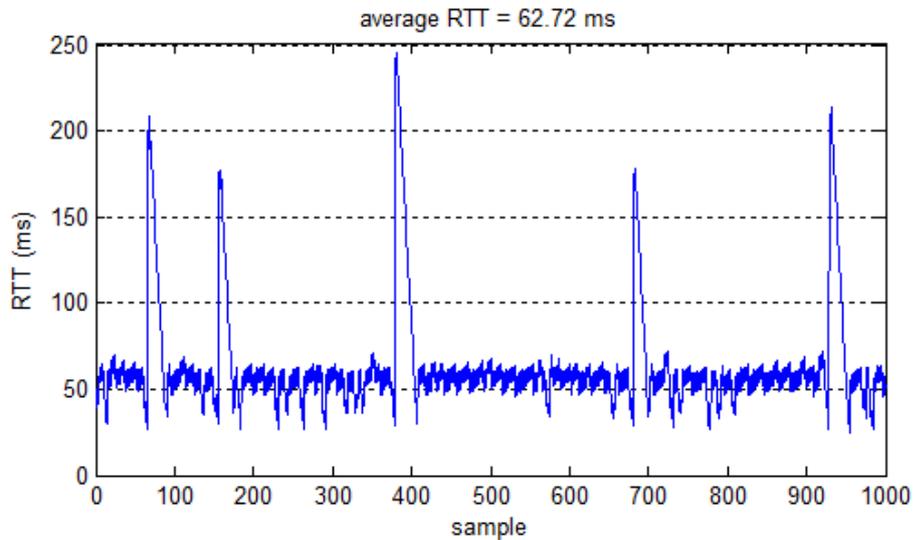


圖 3.10 交大內部網路到校外網路的 RTT 時間

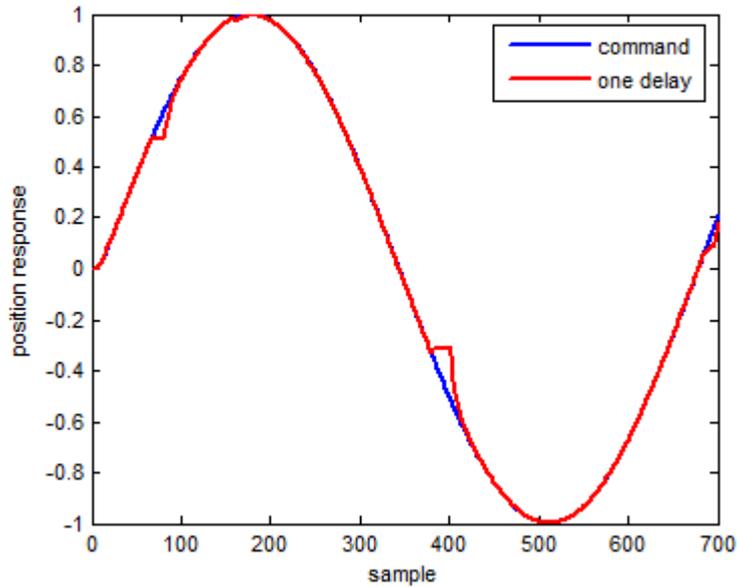


圖 3.11 One-delay 處理時間延遲的響應

### 3-4-3 最小平方法估測

最小平方法則是以多項式曲線的方式來估測，假設有  $M$  個已知數值，欲得到第  $M+1$  個的數值，藉由前  $M$  筆已知數值與  $N$  次方多項式曲線來做逼近，該  $N$  次方多項式曲線與  $M$  個已知數值的平方距離為最小，則該  $N$  次方多項式曲線為其他未知數值所估測出的集合，此為最小平方法的原理。而因為要實現於軟體上，所以以公式來實現此估測法：

$$x[k] = c_0 + c_1k + c_2k^2 + \dots + c_Nk^N \quad (3-6)$$

$$\begin{bmatrix} x[1] \\ x[2] \\ \vdots \\ x[M] \end{bmatrix} = \begin{bmatrix} 1 & 1 & \dots & 1 \\ 2^0 & 2^1 & \dots & 2^N \\ \vdots & \vdots & \ddots & \vdots \\ M^0 & M^1 & \dots & M^N \end{bmatrix} \begin{bmatrix} c_0 \\ c_1 \\ \vdots \\ c_N \end{bmatrix} = X = A \cdot C \quad (3-7)$$

$$C = (A^T A)^{-1} A^T X \quad (3-8)$$

$x[1]$ 、 $x[2]$ 、...、 $x[M]$  為  $M$  筆已知數值，(3-6) 式為  $N$  次方多項式曲線，由 (3-7) 式可知矩陣  $A$  為已知，矩陣  $X$  為  $M$  筆已知數值所構成的矩陣亦為已知，

根據(3-8)式可得矩陣  $C$ ，則所欲估測的下筆數值  $x[M+1]$  由下式可知，

$$\begin{aligned}
 x[M+1] &= c_0 + c_1(M+1) + c_2(M+1)^2 + \dots + c_N(M+1)^N \\
 &= [(M+1) \quad (M+1)^2 \quad \dots \quad (M+1)^N] \cdot C \\
 &= [(M+1) \quad (M+1)^2 \quad \dots \quad (M+1)^N] \cdot (A^T A)^{-1} A^T \\
 &\equiv LSE(M, N) \cdot X
 \end{aligned} \tag{3-9}$$

故根據(3-9)式，只要有  $M$  筆已知數值與欲逼近  $N$  次方多項式，可估測出所希望的估測值。而本研究是以前三筆資料來做二次曲線的逼近，因此，由(3-9)式

$$LSE(3,2) = [1 \quad -3 \quad 3] \tag{3-10}$$

在實現時，用  $LSE(3,2)$  乘上前三個紀錄的值，即可得到估測項。

### 3-4-4 PDC架構的命令估測分析

估測得到的效果好壞與訊號可否估測相關，以一般的閉迴路網路控制系統如圖 3.12，控制命令是  $R(s) - Y(s)$  後的誤差，為不易估測的訊號，估測此訊號並無意義，且估測誤差大[19]。

本研究的網路控制系統，為了避免網路延遲造成的系統不穩定，加了 PDC 的架構，因為 client 端為事件觸發，沒有空取樣的問題，因此估測是由時間觸發的 DSP 端來進行命令估測，故在此討論加入 PDC 架構後的網路控制架構使用命令估測的可行性。

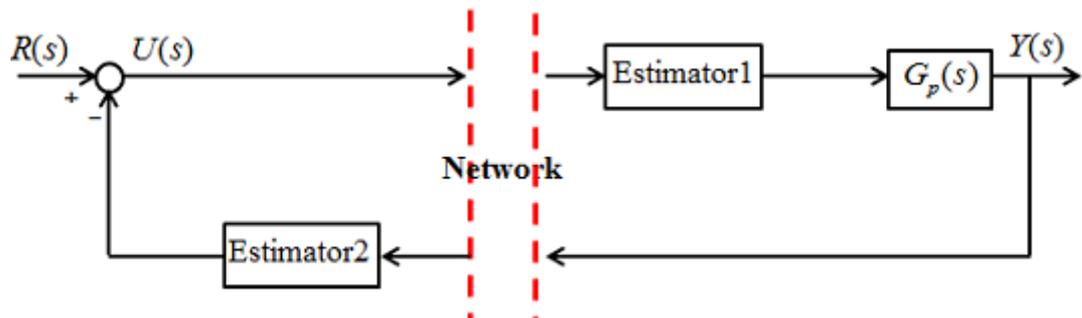


圖 3.12 閉迴路網路控制系統

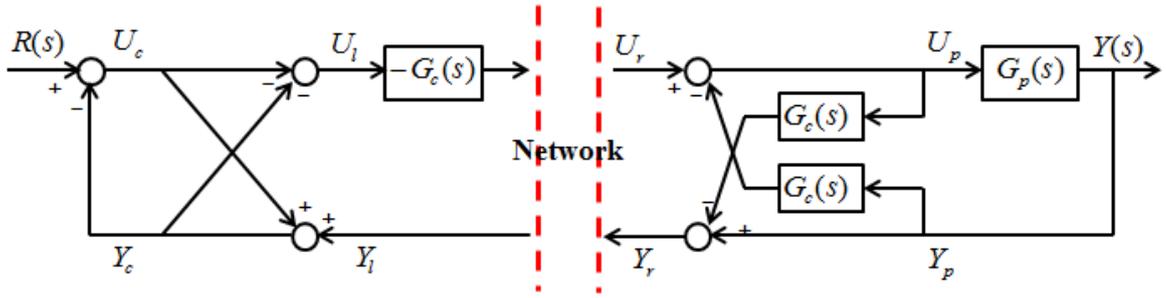


圖 3.13 加入 PDC 的網路控制系統

當遺失發生時，需要進行  $U_r$  的估測，由圖 3.13 左側可以推得

$$U_c = R(s) - Y_c \quad (3-11)$$

$$U_l = -U_c - Y_c = -(R(s) - Y_c) + Y_c = -R(s) \quad (3-12)$$

$$U_r = -G_c(s) \cdot U_l = G_c(s) \cdot R(s) \quad (3-13)$$

因此我們得知，加入 PDC 架構以後，所估測的命令  $U_r$  為一可估測的訊號，而不是誤差訊號，當資料遺失發生時，可透過估測此訊號來改善不好的響應。

### 3-4-5 PDC 架構結合最小平方估測實驗

圖 3.14 是當發生資料重傳時使用 One-delay 法，馬達的響應結果。我們可以觀察到重傳造成的大延遲導致連續多筆的資料遺失，由於使用 One-delay，在這段期間持續沿用最後一筆到達的資料，造成了響應的不平滑，由圖 3.14 中間的誤差結果圖也可以觀察到當延遲發生時追跡誤差的增加。

為了改善此情形，將 PDC 結合最小平方估測使用，PDC 可避免超過取樣週期的網路延遲造成系統的不穩定；另一方面最小平方估測用來取代 One-delay 法，改善當連續遺失發生時，響應不平滑的情形。實驗結果如圖 3.15 所示，用最小平方估測代替 One-delay 法可以改善延遲時間增加造成的響應不平順的問題。

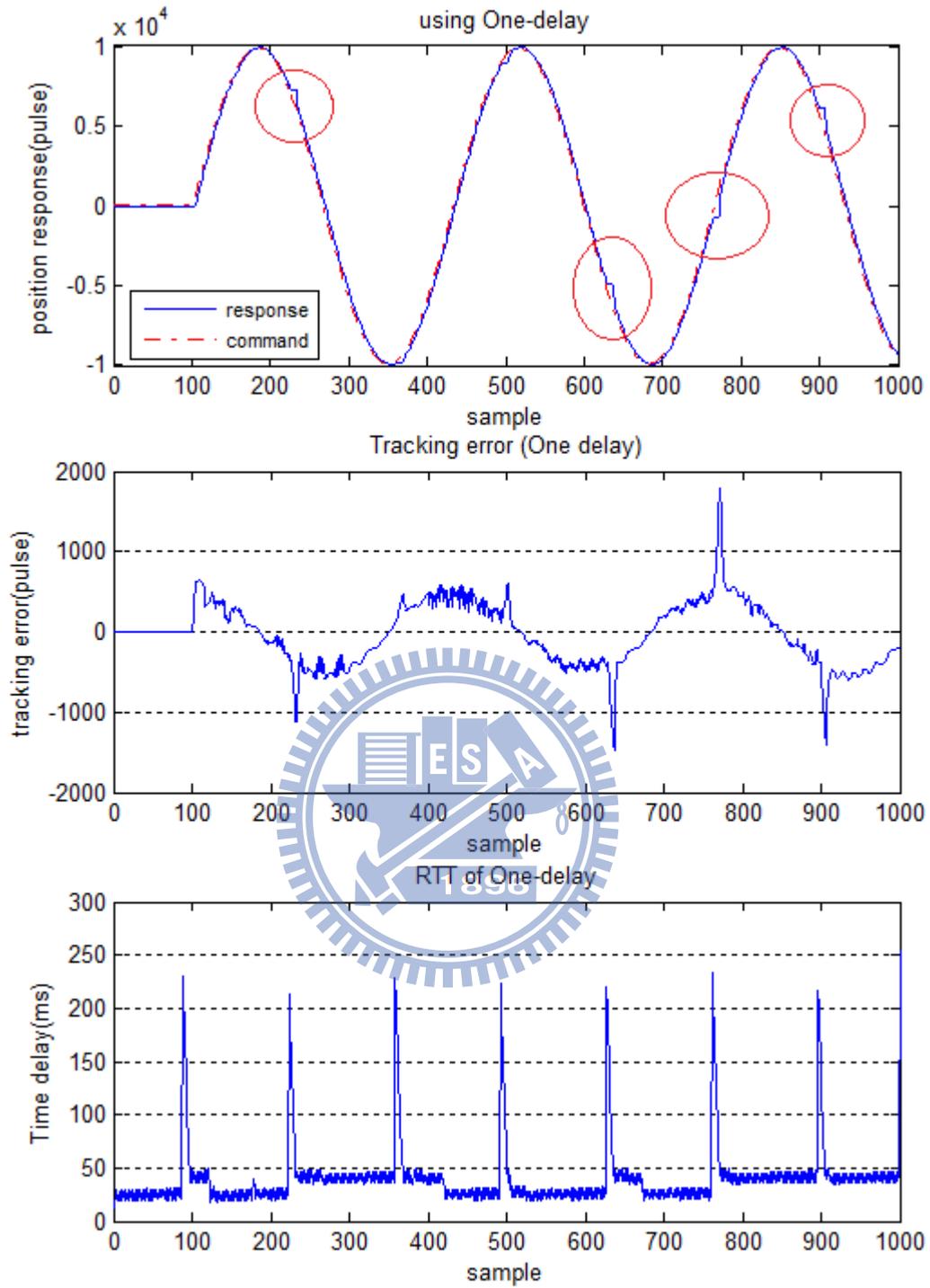


圖 3.14 延遲時間對馬達位置控制影響實驗圖(One-delay 估測)

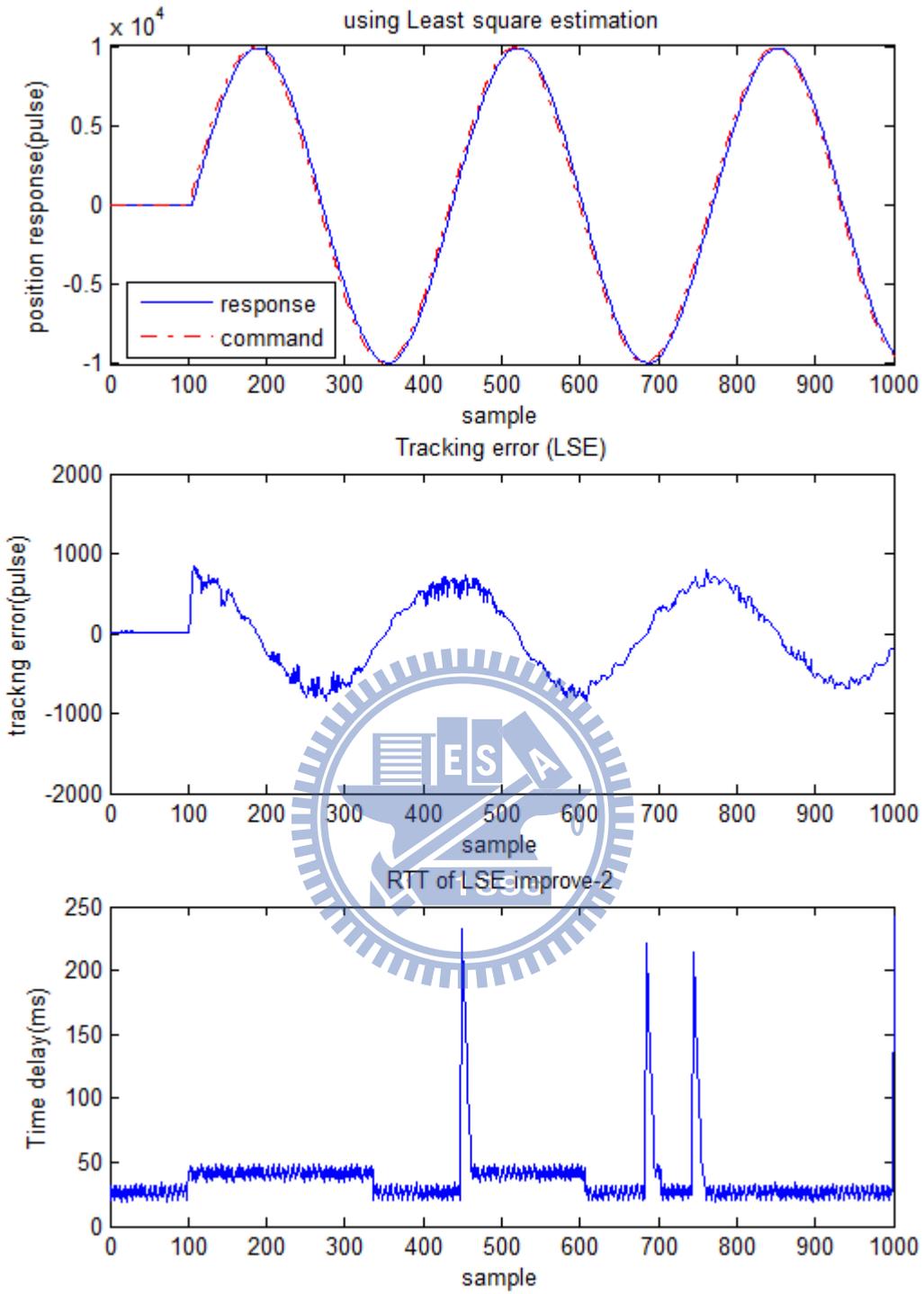


圖 3.15 遲時間對馬達位置控制影響實驗圖(最小平方估測)

## 第四章 雙軸同動協調

### 4-1 雙軸的網路控制系統

當網路控制系統從單軸拓展到雙軸時，兩軸在網路延遲時間可能不同，原因是雙軸使用了不同網路、使用了容易受到環境影響的無線網路而使兩軸傳輸延遲時間不同、兩軸網路通過的網域、路由器的數目不同...等。而當我們使用異質性的網路在循軌跡時，兩軸傳輸延遲時間不同會造成循軌跡的結果與命令不同，如圖 4.1 所示 XY 的循圓命令因為兩軸的延遲時間不同，響應變成了橢圓。

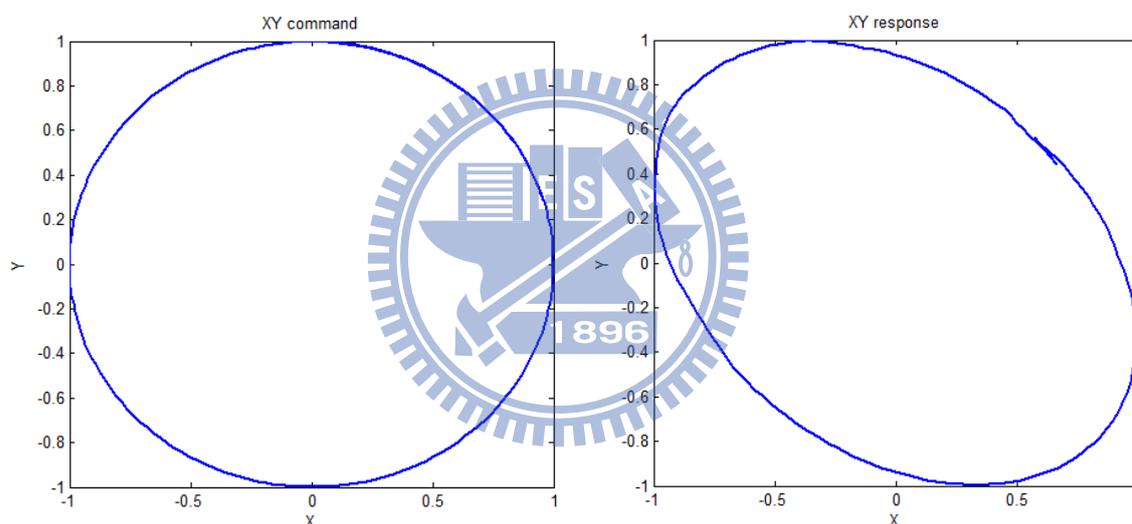


圖 4.1 不同時間延遲的雙軸命令與響應結果圖

而本研究的異質性網路雙軸實驗架構如圖 4.2，異質性的雙軸網路其中一軸為 Ethernet + CAN 的混和網路，而另一軸是單純的 CAN 網路。其中 Ethernet + CAN 的網路由於 Ethernet 有與外部連接，會發生網路壅塞產生延遲，而進行 Ethernet 與 CAN 的封包轉換也會額外增加延遲。因此 Ethernet + CAN 比單純使用 CAN 網路的延遲時間還要長。

在本章將介紹雙軸的協調機制，其主要包含兩部分，以測量到的延遲時間

來進行協調的等待同步機制與用輪廓誤差來 CCC 架構，使用協調機制後，便能夠克服異質性網路造成的不同動與誤差。

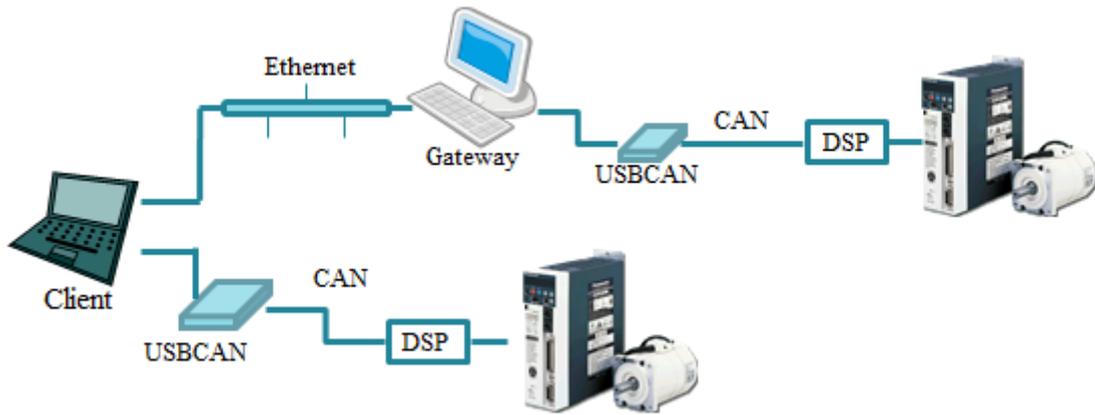


圖 4.2 異質性網路雙軸實驗架構

## 4-2 雙軸同步

### 4-2-1 等待同步機制

造成雙軸不匹配的主要原因就是兩軸的延遲時間不同，因此在兩軸都各別測量出網路的延遲時間，根據延遲時間差距來進行調整，如圖 4.3。

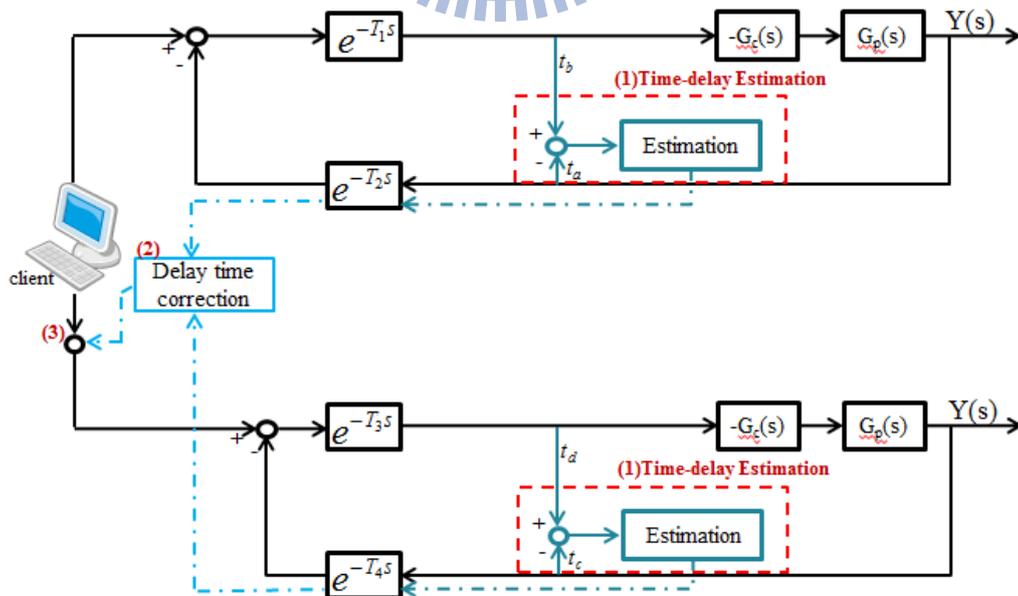


圖 4.3 等待同步機制

等待同步機制主要分成三個步驟， $T_1$  為在 Ethernet+CAN 的時間延遲， $T_2$  為在

CAN 網路的時間延遲，配合圖 4.3 來看：

1. 兩軸個別的延遲時間測量。  
→經過測量得到  $T_1=102\text{ms}$ 、 $T_3=2\text{ms}$ 。
2. 測量結果回傳到 Client 端，計算出時間差，換算成需等待的命令數。  
→ $(102-2)/10=5$  (取樣週期  $T_s=10\text{ms}$ )，因此 X 軸慢了 10 個命令
3. 根據得到的計算結果，修改延遲時間較小的軸的命令，配合延遲大的軸。  
→Client 送 no.500 給 X 軸，送 no.(500-10) 給 Y 軸

#### 4-2-2 等待同步機制模擬

實驗設定：X 軸延遲 102ms，Y 軸延遲 2ms，取樣週期 10ms

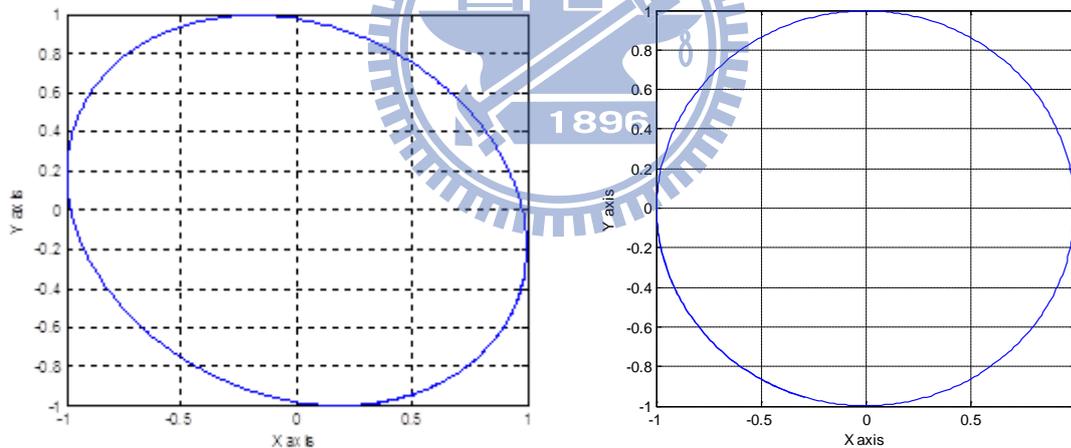


圖 4.4 (a) 等待同步機制加入前後的 XY 響應圖

橢圓與圓以真圓度最能夠比較出其差異，真圓度直徑法：最大直徑與最小直徑之差

{ 加入前：0.3584  
加入後：0.0134

加入等待同步機制消除了延遲造成的差異，大幅改善了真圓度，將軌跡由橢圓改善成圓。

## 4-3 交叉耦合控制器用於遠端網路控制系統

### 4-3-1 交叉耦合控制器(cross-coupled controller , CCC)

交叉耦合控制器是由 Koren 所提出[20]，主要目的不是改善各軸的追蹤誤差，而是協調各軸的位置誤差以消除兩軸以上的輪廓誤差。交叉耦合控制器並不直接改變個軸的運動控制迴路，而是在軸與軸的控制迴路間加上補償器，藉由補償器產生適當的迴授信號至各軸而使得各軸的動態響應能夠匹配。之後 Koren and Lo 又提出一種新的交叉耦合控制架構，稱為可變增益交叉耦合控制器[21,22]，控制器架構如圖 4.5 所示。之所以稱為可變增益的原因為控制器會依照不同的軌跡形式去調整輪廓誤差增益模組增益值( $C_x, C_y$ )。

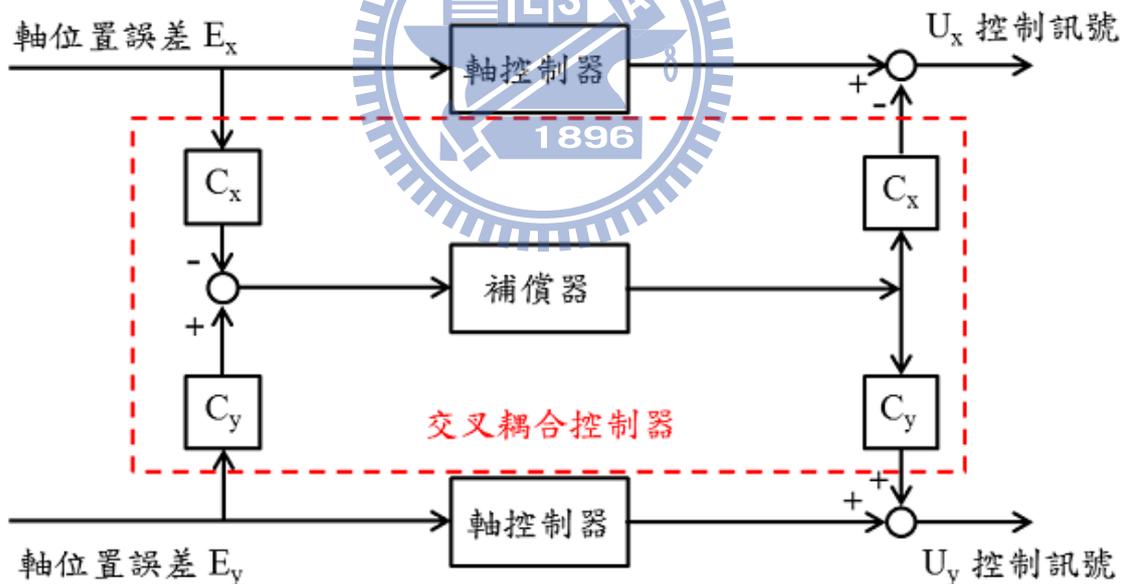


圖 4.5 可變增益交叉耦合控制器結構圖

### 4-3-2 交叉耦合控制器圓路徑(circular Contour)

由圖 4.6 的圓路徑幾何形狀可知道其輪廓誤差  $\varepsilon$  為：

$$\varepsilon = \sqrt{(P_x - X_0)^2 + (P_y - Y_0)^2} - R$$

其中刀具實際位置 $(P_x, P_y)$ 可以由軸位置誤差 $(E_x, E_y)$ 與命令軌跡來表示，即：

$$P_x = R \sin \theta + X_0 - E_x$$

$$P_y = R \sin \theta + Y_0 + E_y$$

可得

$$\varepsilon = \sqrt{(R \sin \theta - E_x)^2 - (R \cos \theta + E_y)^2} - R$$

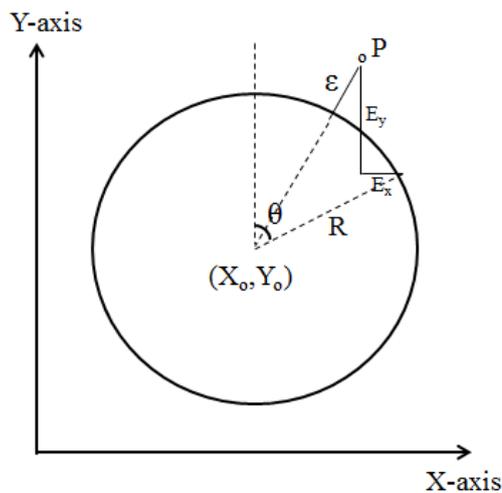
將上式以泰勒展開式展開，並且假設輪廓誤差遠小於位置誤差，且軸位置誤差遠小於圓半徑。輪廓誤差可寫成：

$$\varepsilon = \left( \cos \theta + \frac{E_y}{2R} \right) E_y - \left( \sin \theta - \frac{E_x}{2R} \right) E_x$$

其中可變增益交叉耦合控制的輪廓誤差模組增益值為：

$$C_x = \sin \theta - \frac{E_x}{2R}$$

$$C_y = \cos \theta + \frac{E_y}{2R}$$



$R$ ：圓半徑  
 $R$ ：實際位置  
 $E_x, E_y$ ：位置誤差  
 $\varepsilon$ ：輪廓誤差  
 $(X_0, Y_0)$ ：圓中心座標

圖 4.6 圓路徑之輪廓誤差

### 4-3-3 交叉耦合控制器結合PDC應用於網路控制

在實驗架構中，位置回授透過網路傳回控制端，位置命令由控制端透過網路向 XY 兩軸發出，因此，將 CCC 的補償值改為由位置命令加入，因此將傳統的交叉耦合控制系統架構(圖 4.7)稍作修改，將加入補償值的位置提前到位置增益之前[23]，如圖 4.8 所示。由於是網路控制，為了避免網路傳輸延遲造成系統的不穩定，把 PDC 加入圖 4.8 的架構當中，然而 PDC 的 Butterfly element 卻會影響位置回授訊號，因此我們把受到 PDC butterfly element 影響到的位置回授訊號與 CCC 要用的位置回授訊號分開，最後得到一系統架構，能夠防止網路傳輸延遲時間造成的系統不穩定，另外還有 CCC 協調雙軸減少誤差的功能，如圖 4.9 所示。

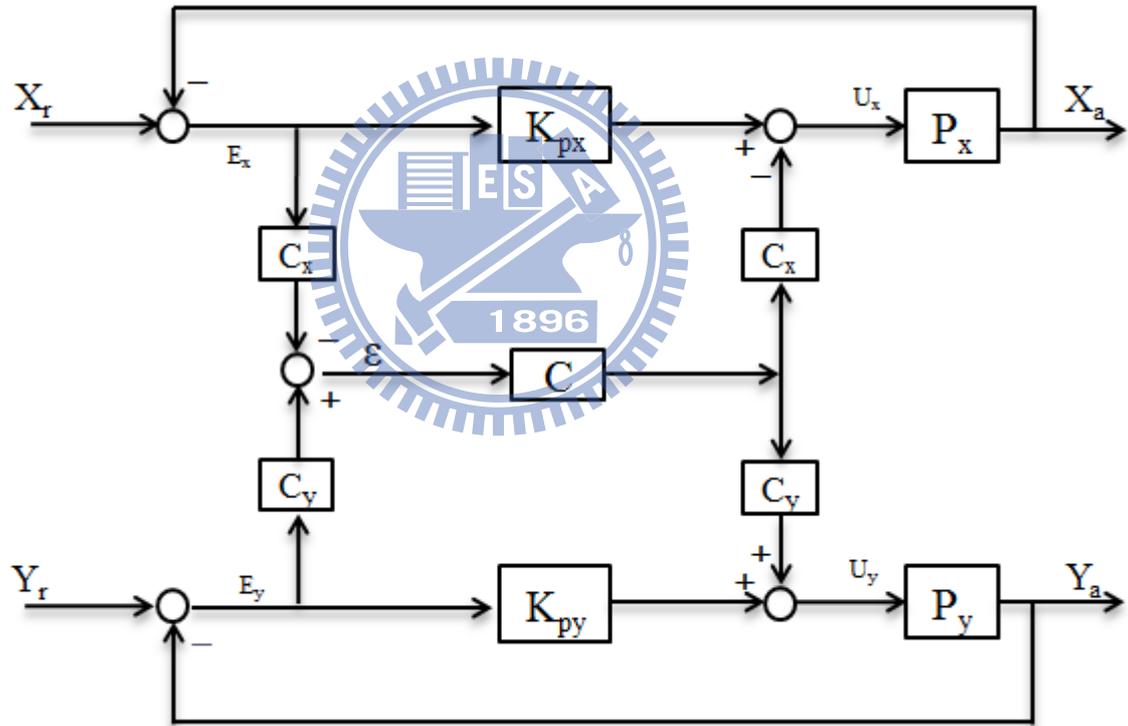


圖 4.7 交叉耦合控制器的運動控制系統架構

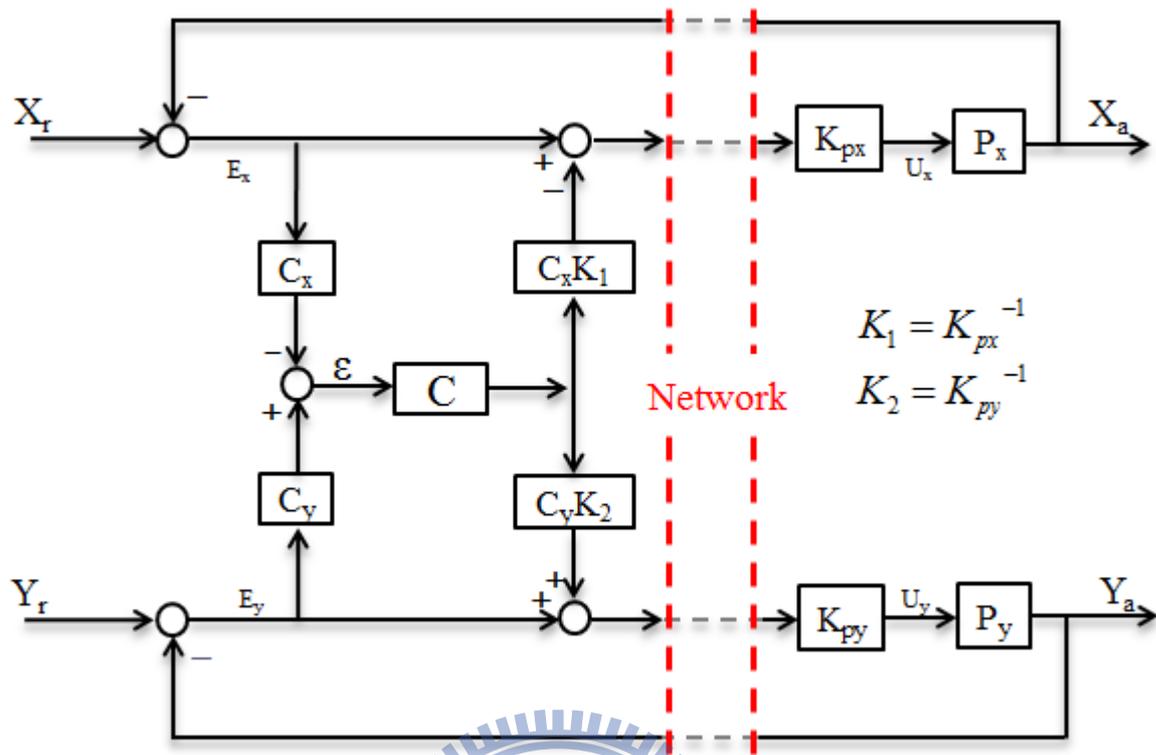


圖 4.8 補償位置提前到增益之前並加上網路傳輸



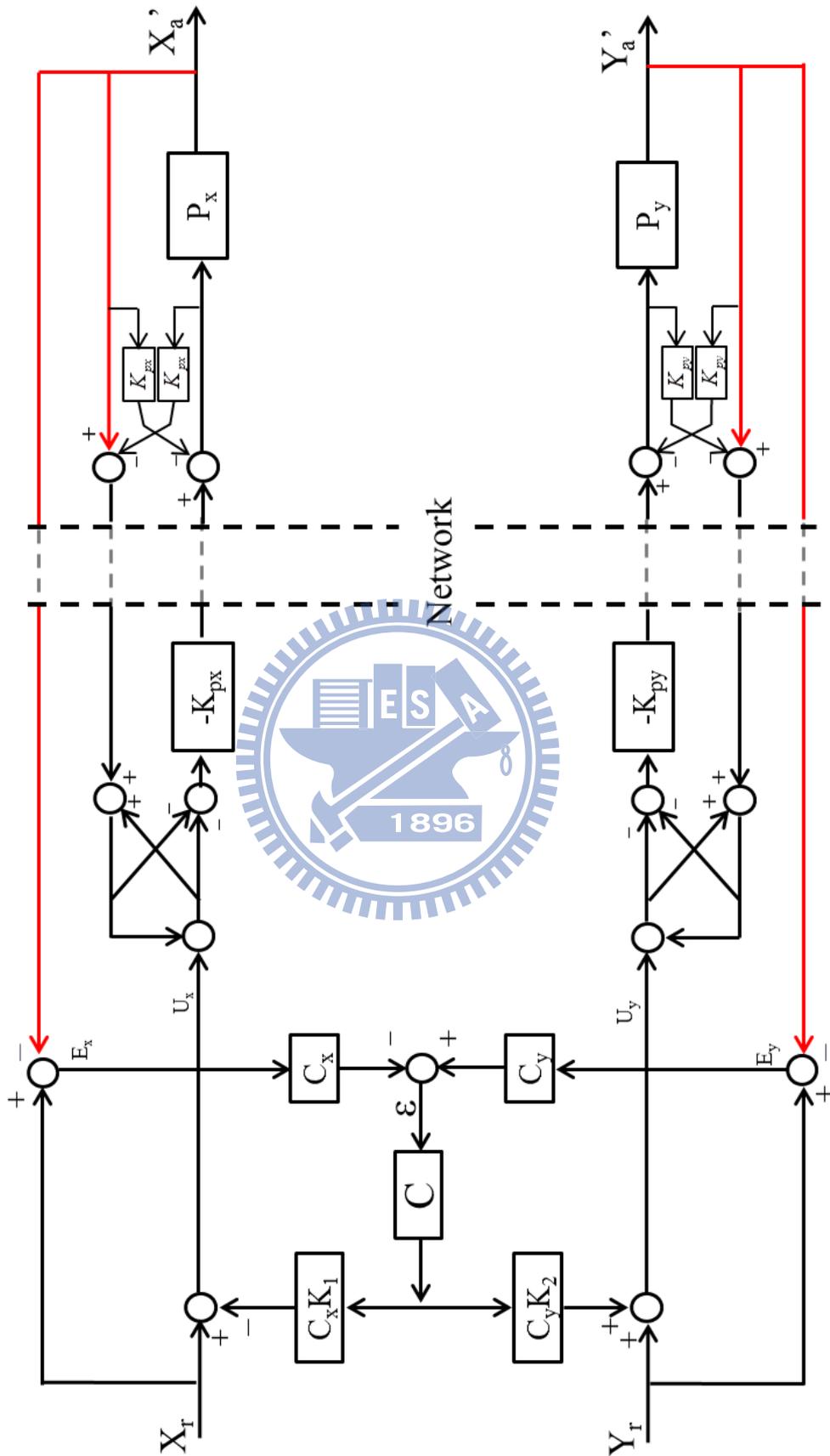


圖 4.9 CCC+PDC 的遠端控制系統

## 4-4 結合的雙軸協調機制

在異質性網路系統，由於兩軸的傳輸、回授延遲不一樣，而兩軸在特性上也會有些微差異，因此圖 4.9 中的架構加上等待同步機制對延遲進行處理，而 CCC 部分則對兩軸差異補償。

### 4-4-1 CCC 異質性網路循環

4-2 根據延遲進行協調的等待同步機制可以消除延遲時間差造成的橢圓，當兩軸特性不一樣，等待同步機制並沒有辦法處理；因此加入 CCC，CCC 能夠透過計算得到輪廓誤差，對兩軸進行協調。

但是在異質性網路兩軸延遲不同情況下使用 CCC，會因為回授延遲不同，導致 CCC 計算輪廓誤差上的錯誤，為避免此情況發生，透過測量到的 RTT 資訊匹配回授的位置資訊得到正確輪廓誤差。

### 4-4-2 結合架構

以 4.10 架構討論 CCC 在異質網路下計算得到的輪廓誤差加入 PDC 且 XY 軸同步後得到

$$X_a' = X_a e^{-t_1 s}$$

$$Y_a' = Y_a e^{-t_1 s}$$

其中  $X_a, Y_a$  為無網路的閉迴路輸出

$$E_x' = X_r e^{-2t_1 s} - (X_a') e^{-t_1 s} = E_x e^{-2t_1 s}$$

$$E_y' = Y_r e^{-(t_1+t_2)s} - (Y_a') e^{-t_2 s} = E_y e^{-(t_1+t_2)s}$$

加入延遲項使  $E_x'$  與  $E_y'$  在時間上匹配

$$E_y e^{-(t_1+t_2)s} \cdot e^{-(t_1-t_2)s} = E_y e^{-2t_1 s}$$

並得到  $2t_1$  前的輪廓誤差

$$\varepsilon' = (\cos \theta \cdot e^{-2t_1 s} + \frac{E_y e^{-2t_1 s}}{2R}) E_y e^{-2t_1 s} - (\sin \theta \cdot e^{-2t_1 s} - \frac{E_x e^{-2t_1 s}}{2R}) E_x e^{-2t_1 s}$$

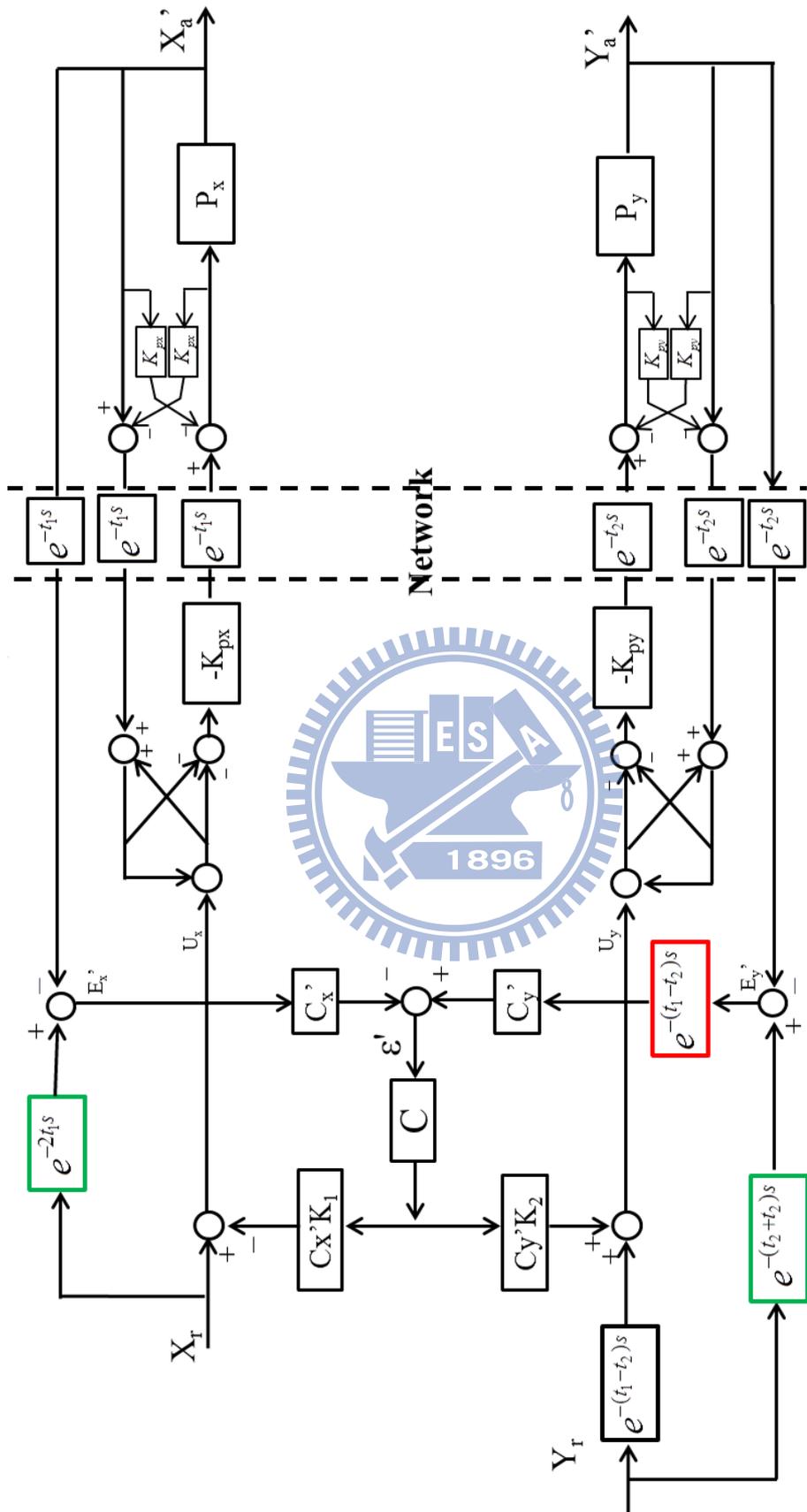


圖 4.10 根據雙軸延遲差距補償的 CCC 架構

以 X 軸延遲 50ms，Y 軸延遲 2ms 進行模擬，比較 XY 軸延遲同步後，加入 CCC 前後的改善。

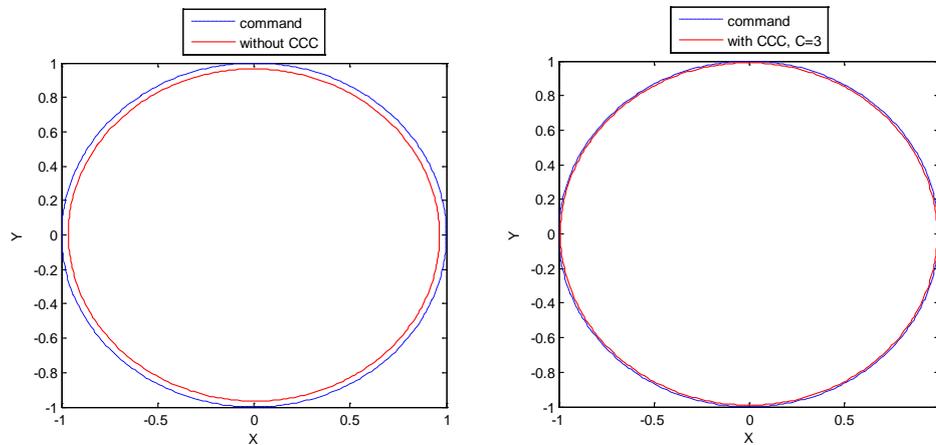


圖 4.11 CCC 加入前後循圓模擬圖

在異質性網路情況下 CCC 得到的輪廓誤差受到延遲影響，計算得到  $2t_1$  前的誤差，由表 4.1 的模擬結果，當延遲達 200ms，同步加上 CCC 真圓度為 0.0037 比雙軸同步 0.0018 還差，原因是過大的延遲造成錯誤補償，因此在使用時必須注意延遲時間，當  $t_1$  過大會影響 CCC 效果。

表 4.1 CCC 加入前後模擬

延遲		真圓度	IAE
50ms	同步	0.0022	0.3064
	同步+CCC	0.0011	0.1392
100ms	同步	0.0022	0.3064
	同步+CCC	0.0018	0.1404

200ms	同步	0.0022	0.3064
	同步+CCC	0.0037	0.1530
400ms	同步	0.0022	0.3064
	同步+CCC	0.0057	0.2290

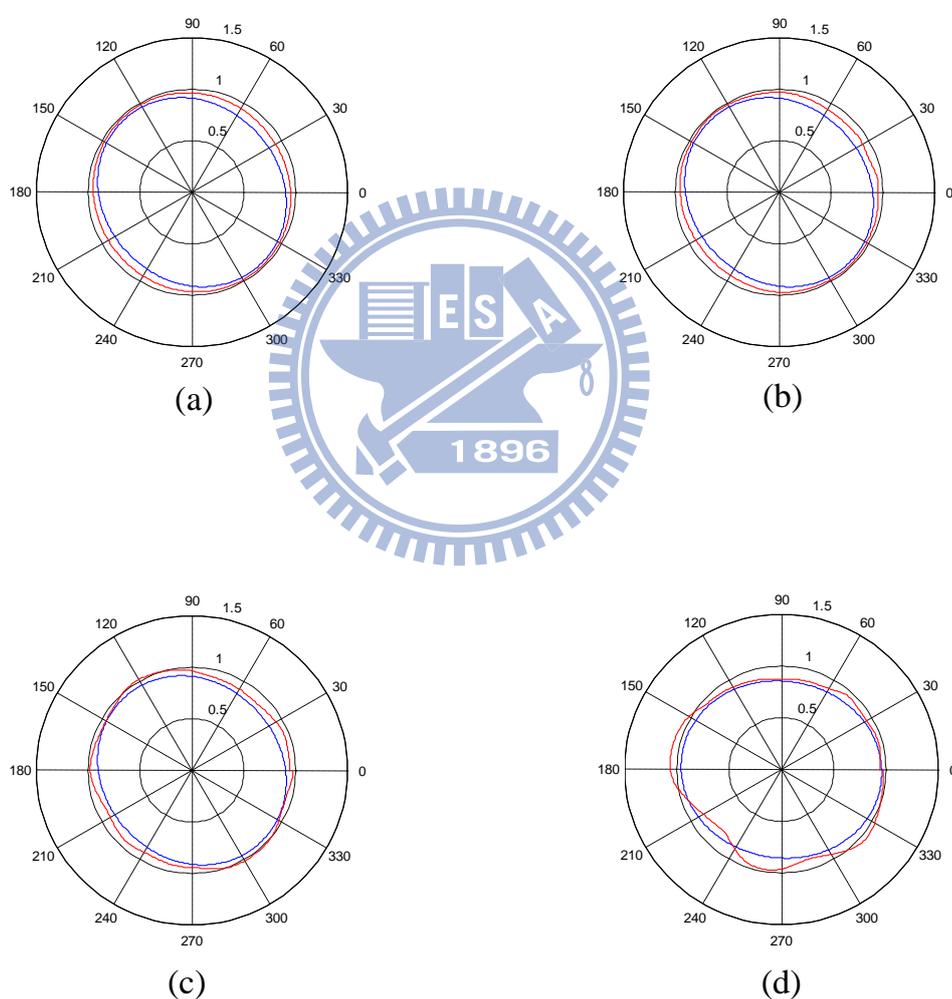


圖 4.12: 同步+CCC 模擬圖(x30) (a)延遲 50ms (b)延遲 100ms (c)延遲 200ms (d)延遲 400ms

## 第五章 異質性網路循圓實驗結果

在第五章我們將第四章的方法進行實作，分別對延遲時間以及兩軸差異進行補償，觀察其累積誤差，最大誤差與真圓度，並對不同延遲時間狀況進行分析比較其結果。

### 5-1 異質性網路誤差改善流程

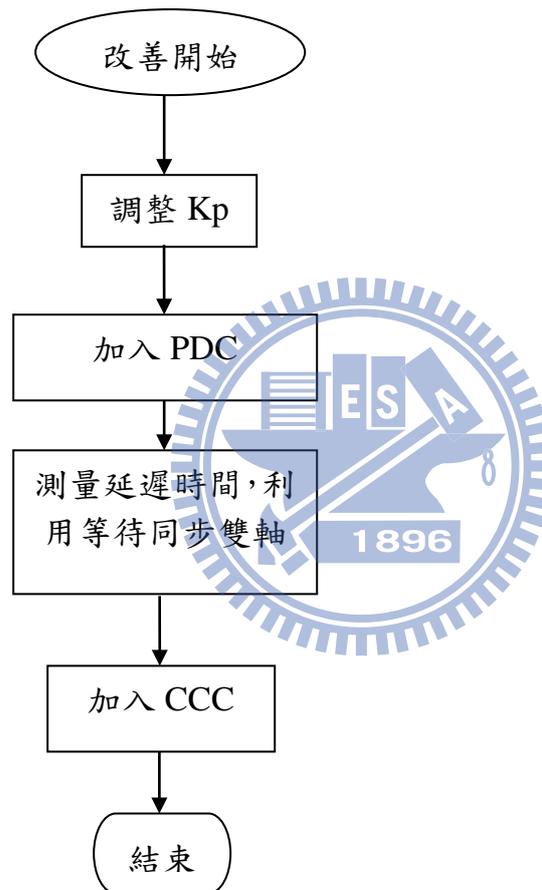


圖 5.1 改善流程圖

造成異質性網路循圓為橢圓的原因最主要是傳輸與回授延遲時間不同，因此最優先得到兩軸延遲資訊，得到 RTT 時間後，我們假設命令延遲等於回授延遲等於一半的 RTT 時間，經過計算得到兩軸相差命令個數，以延遲時間短的軸(實驗中為 Y 軸，2ms 延遲)等待延遲時間長的軸(X 軸，108ms) 進一步改善循圓結果，並以真圓度來判斷改善的程度，最後再以 CCC 改善輪廓誤差。

## 5-2 實驗結果

峰值為 30000 pulse 的 sine、cosine 命令，X 軸延遲 108ms，Y 軸延遲 2ms，速度為 57pulse/ms，初始兩軸，圖中的單位皆為  $\frac{\text{pulse}}{30000}$  而比較的基準為輪廓誤差的 integral absolute error(IAE)，真圓度

輪廓誤差： $\varepsilon = \sqrt{(P_x - X_0)^2 + (P_y - Y_0)^2} - R$ （實際位置到圓心的距離 - 半徑）

$$IAE = \int_{t_0}^{t_f} |\varepsilon| dt, \text{ or } \sum_{k=k_0}^{k=k_f} |\varepsilon_k|$$

真圓度直徑法：最大直徑與最小直徑之差

### 5-2-1 Kp調整

進行網路實驗前 Kp 沒調整好造成的問題有

1. Kp 太小(0.0001)，響應不佳
2. 兩軸特性差太多(以 X 軸 Kp = 0.0001, Y 軸 Kp = 0.00023 為例子)，造成橢圓。

在進行網路循圓之前必須確定各軸響應跟得上命令，需要先進行無網路循圓。從圖 2(a) Kp = 0.0001，系統不夠快跟不上命令，造成響應較命令圓小，需要較大的 Kp 來進行循圓。而在調整 Kp 時須注意兩軸需一致，不一致時會出現圖 2(b)的橢圓，因此需要調整適當 Kp 值讓兩軸特性相近，來確保造成誤差的變因是來自於網路。

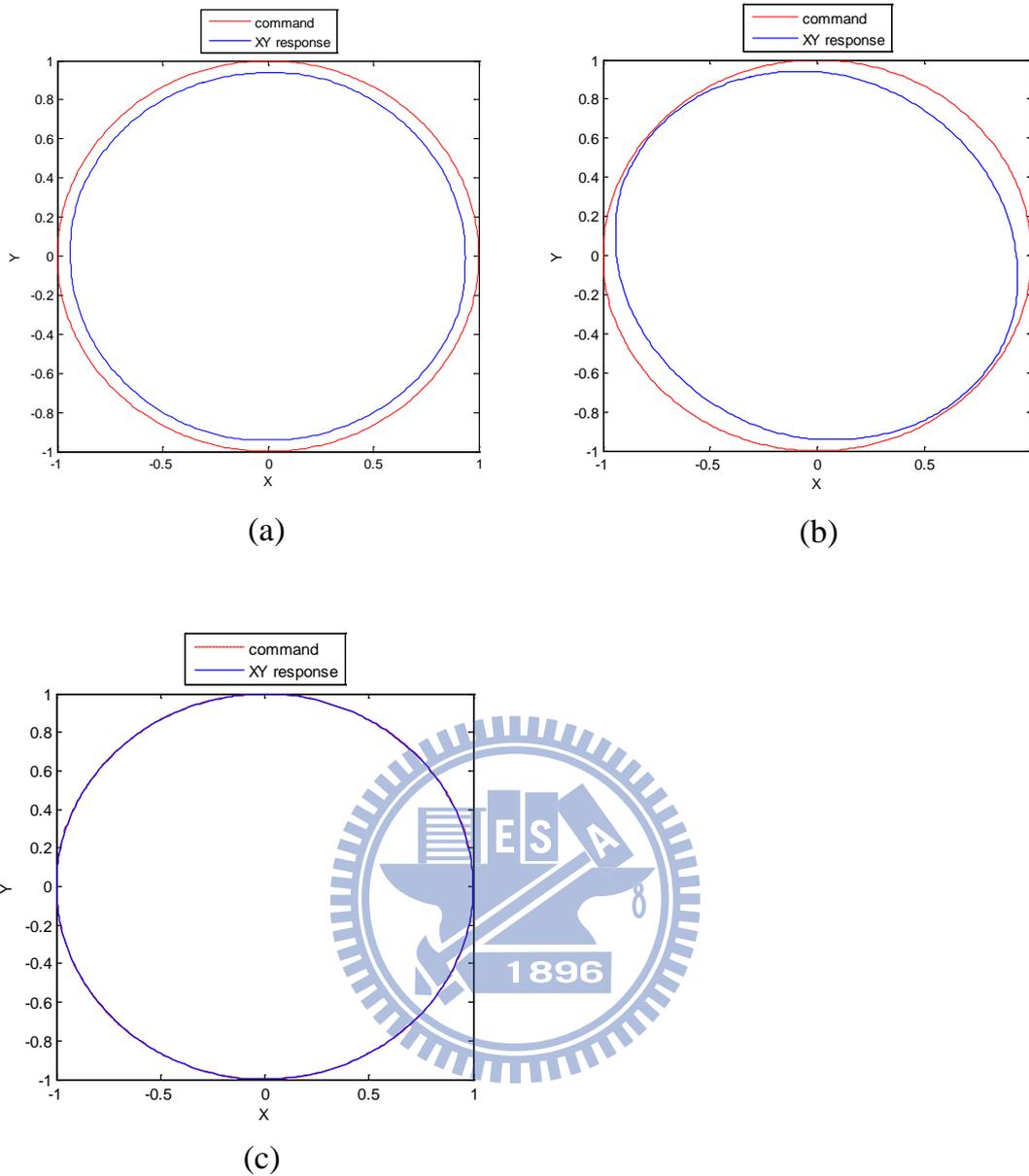


圖 5.2(a)  $K_{px} = K_{py} = 0.0001$  (b)  $K_{px} = 0.0001, K_{py} = 0.00023$  (c)  $K_{px} = K_{py} = 0.00023$

### 5-2-2 各軸加入PDC

沒加 PDC 的 X 軸受到延遲影響，響應跟不上命令，無法與 Y 軸配合畫出一圓，因此需要加入 PDC 改善延遲所造成的影響。

加入 PDC 降低延遲時間對系統影響，將延遲對系統的影響轉為單純時間延

遲，X 軸會是延遲 100ms 的 sine 結果，由於並無針對兩軸時間差改善，X 軸落後 100ms 造成了橢圓的結果，針對兩軸時間差進行下一步的改善同步。

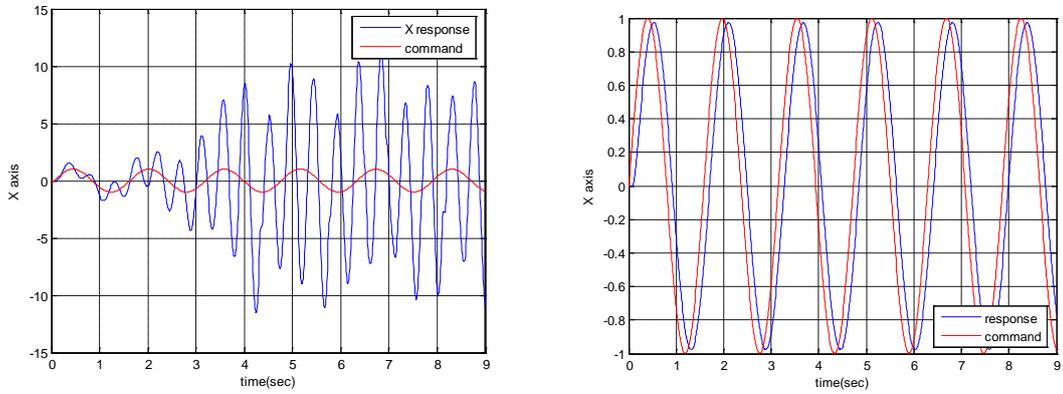


圖 5.3 加 PDC 前後 X 軸響應

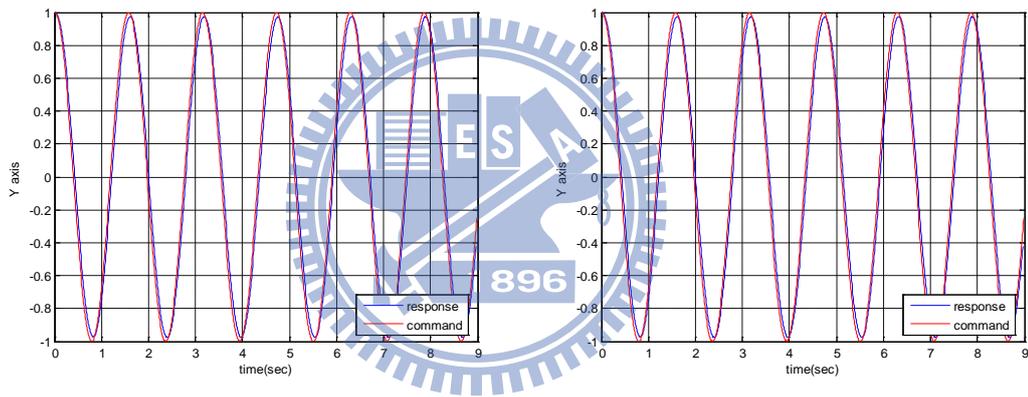


圖 5.4 加 PDC 前後 Y 軸響應

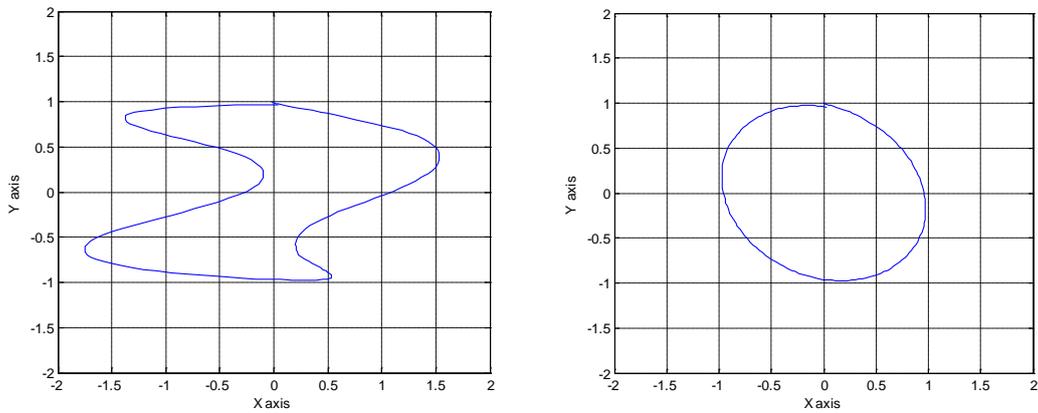


圖 5.5 加 PDC 前後循圓結果

### 5-2-3 測量RTT，延遲等待協調雙軸

根據測量到的 RTT 時間，能夠推得命令延遲，並得到 XY 軸的相差命令個數，如此一來變能夠利用加入 delay buffer 的方式來讓 Y 軸(2ms)等待 X 軸 (108ms)，消除延遲差異，將橢圓改善為圓。

表 5.1：各延遲時間同步比較

延遲時間差	改善法	真圓度	IAE
50ms	PDC	0.1571	8.1770
	PDC+同步	0.0125	0.8869
100ms	PDC	0.3836	20.0876
	PDC+同步	0.0126	0.9087
200ms	PDC	0.7549	39.6935
	PDC+同步	0.0126	0.9072

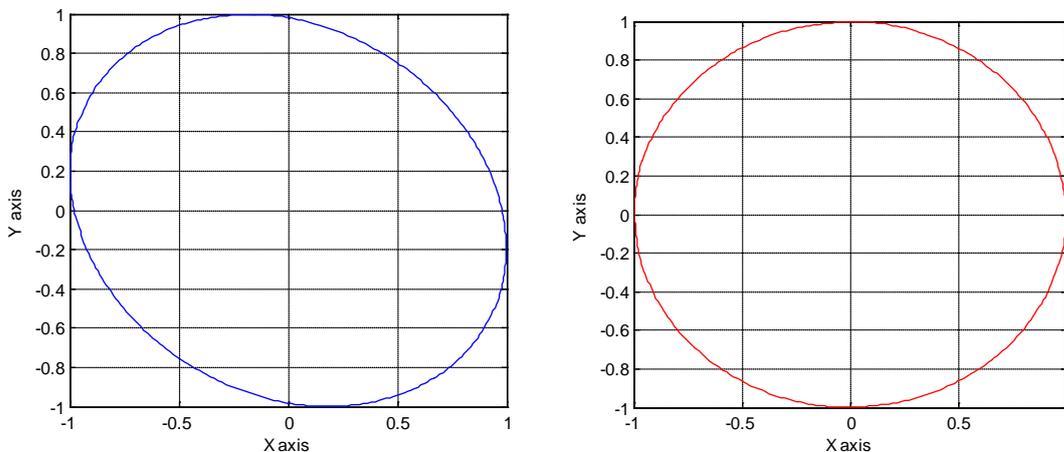


圖 5.6：同步前後比較圖

## 遠端即時量測補償

藉由測量一天各個時刻的延遲時間，得知延遲時間是會變動的(如圖 7)，因此加入即時測量的補償機制來因應。

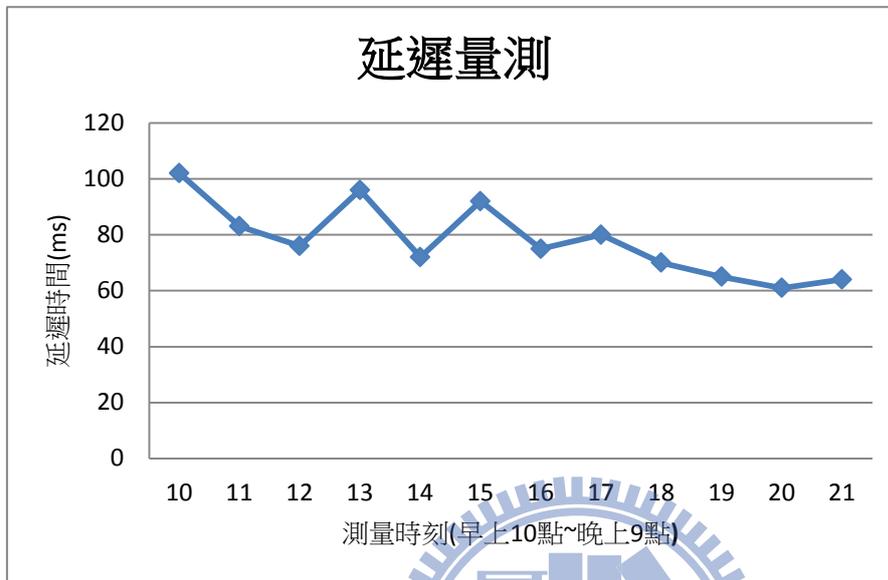


圖 5.7：一天不同時刻新竹到台中傳輸延遲時間紀錄

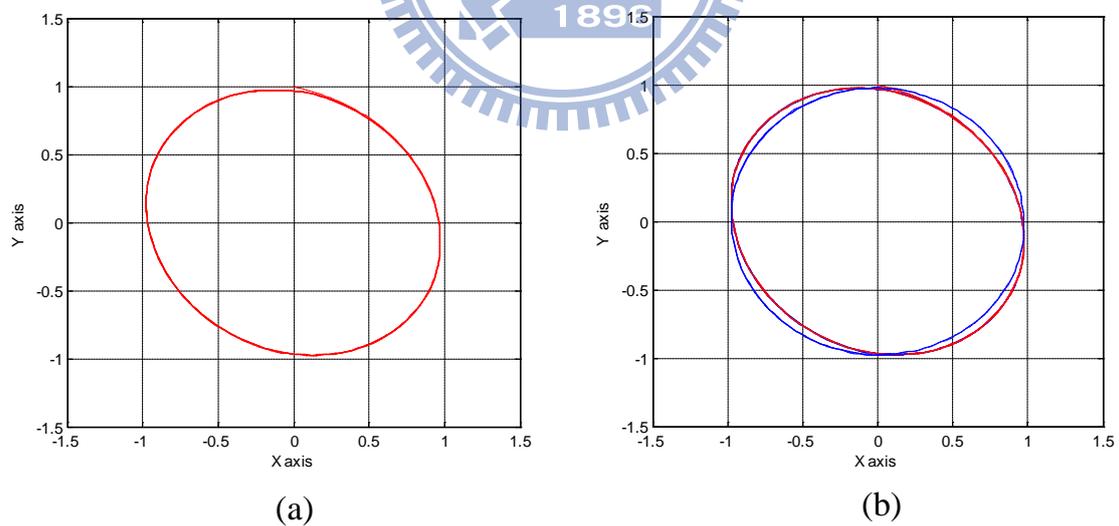


圖 5.8：固定補償與即時補償結果比較 (a)固定補償 (b)即時補償

由實驗來驗證即時補償的效果：在實驗中 X 軸傳輸延遲設定為 100ms，Y 軸補償時間為 40ms，由於等待時間不夠，因此結果仍為橢圓，右圖為即時補償啟動後，系統將補償時間改變，改善原本不夠的補償量。

## 5-2-4 Cross Coupled Control

CCC 能夠透過計算輪廓誤差來得到兩軸補償量，進一步讓兩軸協調同步，在此討論兩部分，1.同步前直接使用 CCC 讓兩軸協調 2.兩軸同步後使用 CCC 的改善。

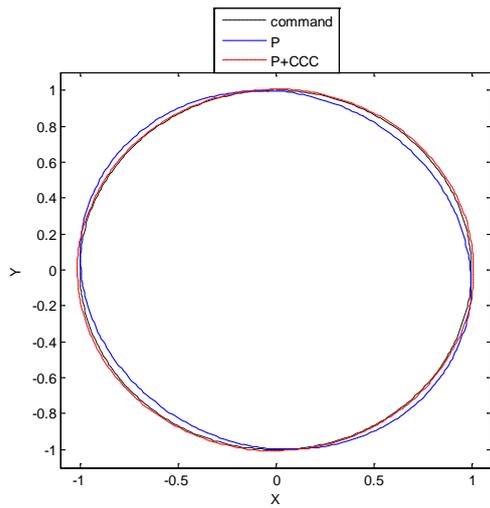
### 同步前使用 CCC：

在兩軸無同步狀態下使用 CCC，位置回授資訊沒有根據 RTT 來做匹配，CCC 的輪廓誤差計算、補償都會受到延遲差異所影響。

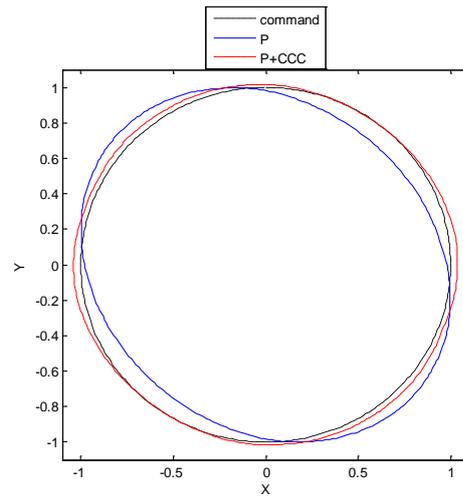
表 5.2：CCC 加入前後比較(無同步)

延遲時間差	改善法	真圓度	IAE
50ms	PDC	0.1571	8.1770
	PDC+CCC	0.0178	2.6362
100ms	PDC	0.3836	20.0876
	PDC+CCC	0.0827	7.322
200ms	PDC	0.7549	39.6935
	PDC+CCC	0.5652	23.5652

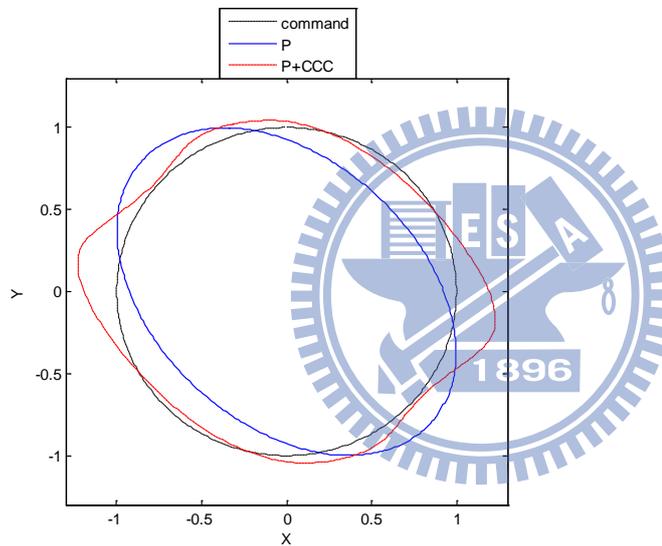
沒同步直接使用 CCC 從表 2 來看真圓度及 IAE 都有明顯改善，但觀察 200ms 的循圓結果圖得到，結果已經是一個不規則的形狀，主要原因就是同時受到輪廓誤差不匹配與延遲補償的雙重影響，而兩軸同步方法的真圓度及 IAE 都優於使用在同步前使用 CCC，因此在有兩軸 RTT 資訊的情況下，直接使用 5-2-3 的兩軸同步法較適合。



(a)



(b)



(c)

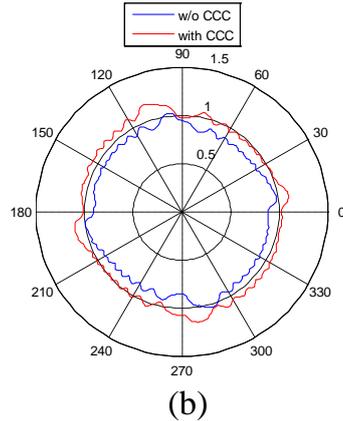
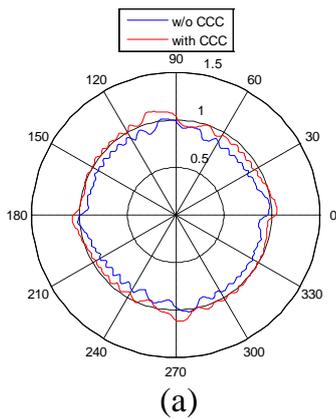
圖 5.9: CCC 加入前後比較(無同步) (a)延遲時間差 50ms (b)延遲時間差 100ms  
(c)延遲時間差 200ms

### 同步後使用 CCC :

將傳輸延遲不同的因素做考量並得到正確的輪廓誤差後，可在兩軸同步之後使用 CCC 並且進一步的改善輪廓誤差。

表 5.3 : CCC 加入前後比較(同步後)

延遲時間差	改善法	真圓度	IAE
50ms	PDC	0.1571	8.1770
	PDC+同步	0.0125	0.8869
	PDC+同步+CCC	0.0127	0.3667
100ms	PDC	0.3836	20.0876
	PDC+同步	0.0126	0.9087
	PDC+同步+CCC	0.0146	0.5583
200ms	PDC	0.7549	39.6935
	PDC+同步	0.0126	0.9072
	PDC+同步+CCC	0.0155	0.9249



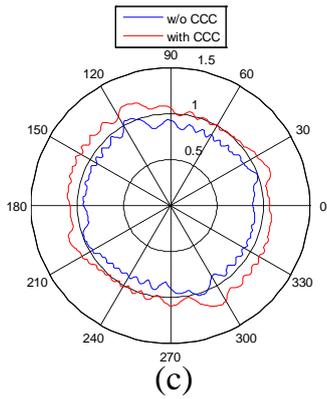


圖 5.10：同步後，CCC 加入前後比較(x30) (a)延遲時間差 50ms  
(b)延遲時間差 100ms (c)延遲時間差 200ms

由實驗結果得知，CCC 在誤差計算上做匹配，但是仍然會受到延遲的影響，在延遲>200ms 情況下加入 CCC 後 IAE 與真圓度都變差，應避免在此情況下使用。

### 5-3 遠端控制實驗

前面的實驗的 X 軸延遲時間是以加入 Buffer 得到，為了證明此方法在實際的 Ethernet 也能夠發揮作用，建立了一遠端系統，如圖所示

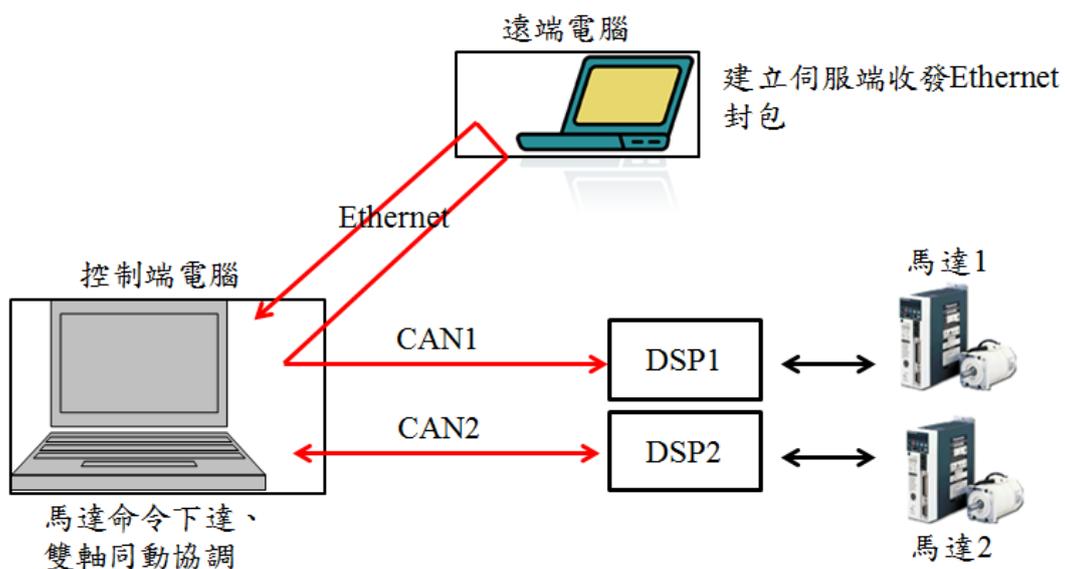


圖 5.11：遠端實驗架構圖

將遠端電腦設置在台中，測量其延遲時間並根據上述改善方法進行雙軸的同動循圓，進行遠端實驗時，實驗架構調整成圖 5.11，主控端 X 軸的傳輸量與處理量增加為原先三倍，使用原本的取樣週期 10ms 進行實驗 buffer 很容易塞爆而馬達不受控制，因此遠端實驗以取樣週期 20ms 來進行。

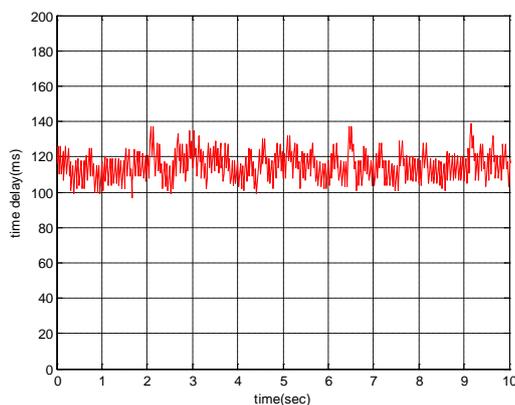


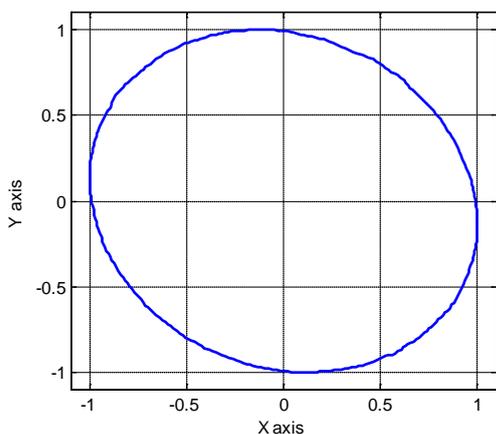
圖 5.12：台中↔新竹延遲時間

**實驗結果：**

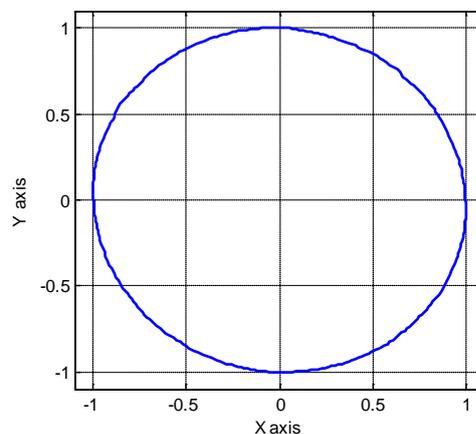
在遠端控制實驗中，使用先前方法確實能夠改善真圓度及 IAE。

表 5.4：遠端實驗結果

改善法	真圓度	IAE
PDC	0.3914	10.0096
PDC+同步	0.0382	3.9947
PDC+同步+CCC	0.0257	2.5540



(a)



(b)

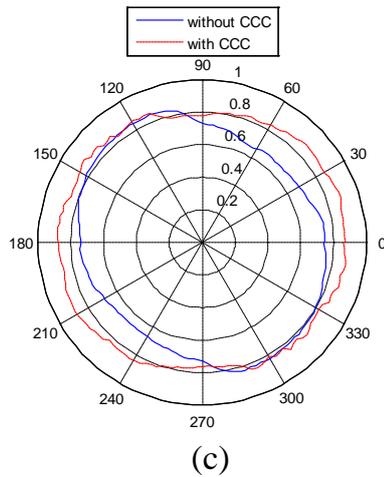


圖 5.13：遠端實驗結果 (a)兩軸加入 PDC (b)兩軸同步  
(c)兩軸同步+CCC

## 5-4 小結

1. CCC 在 50ms 延遲情況下效果明顯，當延遲增加到 200ms 則不適用。
2. 比較同步與 CCC 兩種方法，同步法改善程度較直接使用 CCC 佳，且不受延遲增加影響。
3. 延遲小於 200ms 的情況下，同步加上 CCC 能夠進一步改善輪廓誤差；延遲大於 200ms，CCC 加入會讓真圓度及輪廓誤差增加。
4. 在網路實作上，連線到台中的遠端電腦，平均延遲 115ms，用 delay buffer 與 CCC 都在真圓度以及 IAE 上都能夠改善。

# 第六章 結論與未來發展

## 6-1 結論

本研究對異質性的雙軸網路控制系統進行探討，其中討論延遲時間對網路系統穩定性的影響，資料遺失處理，測量延遲時間方法、雙軸同步等實驗，一步步處理延遲造成的系統問題，並且讓兩軸馬達在傳輸延遲不同的網路系統下達到同步的循圓結果，完成從新竹到台中之間的遠端控制實驗，以下為三點結論：

1. 探討網路延遲時間、資料遺失對系統的影響：

當網路延遲時間 $>$ 取樣週期時，產生系統不穩定，此部分以加入 PDC 將系統的輸出轉為原先無網路的閉迴路轉移函數與延遲項的相乘；而突發性的大延遲則是利用最小平方估測取代 one-delay，消除突然增加的 1000 pulse 誤差。

2. 異質性網路雙軸控制系統的同步

在異質性網路系統當中，由於兩軸使用不同的網路作為傳輸媒介，造成在命令執行上的不同步，此部分以 RTT 的即時測量得到兩軸的傳輸延遲，在延遲較小的軸上加入等待的 buffer 來同步。此方法不受傳輸延遲長短的限制，並且能夠及時調整達到同步。在兩軸延遲時間差距為 100ms 的情況下，雙軸使用 PDC 真圓度為 0.3836，同步加入後可改善為 0.0126，循圓結果也會由橢圓改善為圓。另外加入即時補償機制，當延遲改變，能夠即時偵測並自動同步。

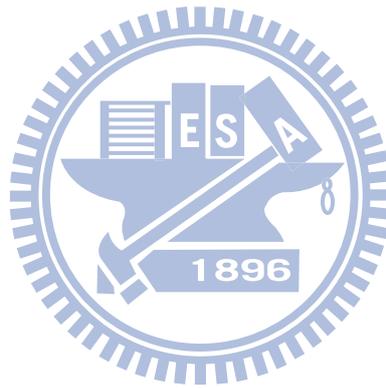
3. CCC 應用於異質性雙軸網路控制系統：

CCC 補償值改由位置控制迴路加入，結合包含 PDC 的雙軸網路控制系統架構，並在輪廓誤差計算上考慮到兩軸不同，匹配兩軸回授

資訊得到正確輪廓誤差，在傳輸延遲 100ms 的情況，把 IAE 從 0.9087 改善到 0.5583

## 6-2 未來發展

本論文中的網路都是有線網路，未來可替換成無線網路，設置方便、易擴充且高便捷的特性都能夠使論文的同步法不只用在循圓上面，而在實際應用上再更上一層樓。但是無線傳輸品質隨環境不同而有很大的差異，因此，克服不同環境下無線的傳輸問題並讓各端點達到同步可成為未來更深入探討的一個主題。



## 參考文獻

- [1] J. Y. Yu, S. M. Yu, and H. Q. Wang, "Survey on the Performance Analysis of Networked Control Systems," *IEEE International Conference on Systems, Man and Cybernetics*, Volume 6, pp.5068–5073, 10-13 Oct. 2004.
- [2] P. L. Tang, and C. W. de Silva, "Compensation for Transmission Delays in an Ethernet-Based Control Network Using Variable-Horizon Predictive Control," *IEEE Transaction on Control System Technology*, Volume 14, No.4, pp.707-718, Jul. 2006.
- [3] J. Nilson, B. Bermhardsson, and Wittermark, "Stochastic Analysis of Control of Real Time Systems with Random Time Delays," *Automatica*, Volume 34, Issue 1, pp.57-64, B.1998.
- [4] 鄭景文, "動態網路控制系統之時間延遲分析," 國立交通大學, 碩士論文, 中華民國 95 年
- [5] P. Sourdille and A. O'wyer, "A New Modified Smith Predictor Design", *Proceedings of the 1st international symposium on Information and communication technologies*, Vol. 49, pp. 385-390, Sep. 2003.
- [6] C. C. Hsieh, P. L. Hsu and B. C. Wang, "The Motion Message Estimator in Real-Time Network Control Systems", *Conference of the IEEE Industrial Electronics Society*, Nov. 2006.
- [7] S. H. Hong, "Scheduling Algorithm of Data Sampling Times in the Integrated Communication and Control Systems," *IEEE Transactions on Control Systems Technology*, Volume 3, Issue 2, pp.225-231, Jun 1995.
- [8] 薛伊婷, "以硬體實現主動可調式CAN 網路排程系統", 國立交通大學, 碩士論文, 中華民國 93 年7 月.

- [9] P. Marti, G. Fohler, K. Ramamritham, and J. M. Fuertes, "Jitter Compensation for Real-Time Control Systems," *22nd IEEE Real-Time Systems Symposium*, London, UK, December, 2001.
- [10] 鄧元銘, "CAN 網路同步應用協定之設計與實現", 國立交通大學, 碩士論文, 中華民國 96 年 7 月.
- [11] F. L. Lian, "Analysis, Design, Modeling, and Control of Networked Control Systems." Ph.D thesis, University of Michigan, May 2001.
- [12] Z. Li and M. Y. Chow, "Adaptive Multiple Sampling Rate Scheduling of Real-time Networked Supervisory Control System – Part I", *IEEE Industrial Electronics, IECON 2006 - 32nd Annual Conference*, pp. 4604 – 4609, Nov. 2006.
- [13] Z. Li and M. Y. Chow, "Adaptive Multiple Sampling Rate Scheduling of Real-time Networked Supervisory Control System – Part II", *IEEE Industrial Electronics, IECON 2006 - 32nd Annual Conference*, pp. 4615 – 4620, Nov. 2006.
- [14] A. Antunes, P. Pedreiras, L. Almeida, and A. Mota, "Dynamic Rate and Control Adaptation in Networked Control Systems", *5th IEEE International Conference on Industrial Informatics*, Volume 2, pp.841-846, Jun. 2007.
- [15] X. D. Ren, S. B. Li, Z. Wang, M. Z. Yuan, and Y. X. Sun, "A QoS Management Scheme for Paralleled Networked Control Systems with CAN Bus," *IEEE Industrial Electronics, IECON 2003 - 29th Annual Conference*, Volume 1, pp.842-847, 2-6 Nov. 2003.
- [16] "CAN Specification Version 2.0", *BOSCH*, 1991.
- [17] 吳燦銘, 鄭苑鳳, "TCP/IP 概論", 五南出版社, 2002.
- [18] Chien-Liang Lai, "Networked Control Systems Design with the Time Delay Compensation". 國立交通大學, 博士論文, 中華民國 100 年 5 月

- [19] 張哲維, ”無線網路控制系統之訊息估測與網路流量監控”, 國立交通大學, 碩士論文, 中華民國 98 年 7 月
- [20] Y. Koren, “Cross-coupled biaxial computer for manufacturing systems,” *ASME Trans. J. Dyn. Syst., Meas. Contr.*, vol. 102, no. 4, pp. 265-272, 1980.
- [21] Y. Koren and C. C. Lo, “Variable gain cross-coupling control,” *CIRP Proc. Manufacturing Systems*, vol. 40, pp. 371-374, 1991.
- [22] Y. Koren and C. C. Lo, ”Advance controller for feed drivers,” *Annals of the CIRP*, vol. 41, pp. 689-698, 1992.
- [23] 葉釗甫, “CNC剛性攻牙製程之先進運動控制器設計”, 國立交通大學, 碩士論文, 中華民國 100 年 7 月

