

臺北市立大學資訊科學系
碩士論文

指導教授：黃志鵬 博士

模糊推論結合線性二次調節器應用於

兩輪平衡車之智慧型演算法設計

Intelligent Algorithm Design by Using Fuzzy
Inference with Linear Quadratic Regulator on
Two-Wheeled Self-Balancing Vehicle

研究生：顏瑋良 撰

中 華 民 國 一 百 一 十 年 八 月

謝誌

碩士，是為了精進自己的能力與邁入社會的墊腳石，指導教授黃志鵬老師教導我的除了論文上的方向，更教導了如何看待事情的更多面向，把原本沒有想到的問題，都給點了出來，讓我明白不論是論文或是其他事情，需要看待的不只表面上的數據，還有其背後的原理與解釋，多方面的去細想，深思熟慮後才能使其更加完善。

感謝洪瑞鍾教授與林建洲教授擔任畢業論文口試委員，給予了許多論文上的建議，也謝謝系辦助教的幫忙與叮嚀瑣事的流程。最後感謝家人背後默默的支持與付出，讓我可以沒有後顧之憂，專心的在學業上努力奮鬥。

能在學業這短暫的路上，遇到各位是學生的運氣，老師們的諄諄教誨，使我受益良多，學生會銘記在心，期許能在未來的路上發光發熱。

II 模糊推論結合線性二次調節器應用於兩輪平衡車之智慧型演算法設計

中文摘要

本篇論文主要探討兩輪自體平衡車的平衡問題。由於其功耗和可操控性優勢，它具有很多潛力，也引起許多研究人員用來分析可使其平衡的控制器性能。本次研究中，我們討論了 LQR 控制器演算法的設計，並加入模糊控制器，使兩輪自體平衡車的 LQR 增益值隨著位移、速度、車體角度和車體角速度的不同而改變，並在 Matlab-Simulink 仿真中模擬執行，最後使用 Arduino 所組裝的兩輪自體平衡車完成實機平衡。該控制器設計主要是測試並分析位移、速度、車體角度和車體角速度的模糊邏輯控制的隸屬函數，使兩輪自體平衡車更快的達到穩定平衡、減少最大超越量與總誤差量，並且也能有效地抑制干擾。

IV 模糊推論結合線性二次調節器應用於兩輪平衡車之智慧型演算法設計

Abstract

The main purpose of this paper is to discuss the balance problem of a two-wheel self-balanced vehicle. Due to its power consumption and the advantages of controllability, it brings many potential opportunities and has encouraged many researchers to analyze the performance of the controller which allows it to remain balanced. In this study, we discussed the design of the LQR controller algorithm. We added a fuzzy controller to make the LQR gain value of the two-wheel self-balanced vehicle change with the difference of position, speed, vehicle body angle and vehicle body angular velocity and then execute the simulation in Matlab-Simulink. Finally, we use the two-wheeled self-balancing vehicle assembled by Arduino to complete the real machine balance. The design of the controller is mainly to test and analyze the membership function of the fuzzy logic control of position, speed, vehicle body angle and vehicle body angular velocity so that the two-wheeled balance vehicle can reach a stable balance faster, and thus suppress the interference effectively.

VI 模糊推論結合線性二次調節器應用於兩輪平衡車之智慧型演算法設計

目次

謝誌.....	I
中文摘要.....	III
Abstract.....	V
目次.....	VII
圖目次.....	IX
表目次.....	XI
第一章 緒論.....	1
第一節 研究背景.....	1
第二節 研究目的.....	2
第三節 論文架構.....	3
第二章 文獻探討.....	5
第一節 兩輪自體平衡車.....	5
第二節 線性二次型調節器.....	7
第三節 模糊推論.....	9
第四節 線性二次型調節器結合模糊推論.....	11
第三章 研究方法.....	13
第一節 使用平台介紹.....	13
第二節 機器人系統模型.....	18
第三節 LQR 控制器設計.....	21
第四節 模糊規則設計.....	23
第五節 LQR 模糊控制器(FLQR).....	30
第四章 研究結果與討論.....	32
第一節 模擬性能比較.....	32
第二節 不同模糊規則數模擬.....	35
第三節 實機測試.....	38
第五章 結論與建議.....	43

VIII 模糊推論結合線性二次調節器應用於兩輪平衡車之智慧型演算法設計

參考文獻.....	45
附錄.....	49

圖目次

圖 3-1 兩輪平衡車架構	13
圖 3-2 Arduino Leonardo 開發版	13
圖 3-3 MPU6050 陀螺儀感測器	14
圖 3-4 42 步進馬達	14
圖 3-5 A4988 馬達驅動器	15
圖 3-6 兩輪自體平衡車坐標系統	16
圖 3-7 4 個狀態與 5 個狀態的輪子角度比較圖	22
圖 3-8 位移輸入	24
圖 3-9 速度輸入	24
圖 3-10 位移輸出	24
圖 3-11 速度輸出	24
圖 3-12 $\theta = -20$	25
圖 3-13 $\theta = -10$	25
圖 3-14 $F = 20$	26
圖 3-15 $F = 10$	26
圖 3-16 車體角度輸入	27
圖 3-17 車體角速度輸入	27
圖 3-18 車體角度輸出	27
圖 3-19 車體角速度輸出	27
圖 3-20 $\psi = 20$, $\psi = 10$	28
圖 3-21 $\psi = -20$, $\psi = -10$	28
圖 3-22 $\psi = 20$, $\psi = 10$	29
圖 3-23 $\psi = 20$, $\psi = -10$	29
圖 3-24 FLQR 控制器	30
圖 4-1 車體傾斜弧度動態響應圖	32
圖 4-2 位移的動態響應圖	32

圖 4-3 車體傾斜角速度動態響應圖	33
圖 4-4 速度動態響應圖	33
圖 4-5 初始車體角度 0.37 時 LQR 與 7FLQR 比較圖	35
圖 4-6 初始車體角時 0.37 5FLQR 與 7FLQR 比較圖	36
圖 4-7 初始車體角度 0.37 時 9FLQR 與 7FLQR 比較圖	37
圖 4-8 LQR 實機平衡圖	38
圖 4-9 7FLQR 實機平衡圖	39
圖 4-10 LQR 於 5 秒時受到 50 脈衝的力	40
圖 4-11 7FLQR 於 5 秒時受到 50 脈衝的力	41

表目次

表 2-1 兩輪自體平衡車文獻探討	6
表 2-2 LQR 控制文獻探討	8
表 2-3 模糊控制文獻探討	10
表 3-1 系統參數	17
表 3-2 調整 k 值的影響[1].....	22
表 3-3 增益值的上下界範圍	23
表 3-4 模糊規則表	29
表 4-1 LQR 與 FLQR 性能比較	34
表 4-2 LQR 與 7FLQR 性能比較	35
表 4-3 5FLQR 與 7FLQR 性能比較	36
表 4-4 9FLQR 與 7FLQR 性能比較	37

XII 模糊推論結合線性二次調節器應用於兩輪平衡車之智慧型演算法設計

第一章 緒論

第一節 研究背景

近年來，越來越多的人使用機器人來改善生活或幫助人類解決問題。而輪型機器人相較於雙腳型機器人行走較快，就靈活性和成本以及控制、操作性都是輪型較為優異。而兩輪自體平衡車的優勢有轉彎半徑小，幾乎為零，可在較小空間範圍內使用，不論是在人群擁擠的都市裡行走或是狹窄巷道難以讓救難人員進入的地區，都可使用兩輪機器人來行進。

1996 年，Yun-Su Ha 提出了一種倒立擺式的移動機器人，他們使用二維倒立擺的簡化模型成功地實現了平衡和移動控制[3]。2001 年，Kamen, D 與他的 Deka 研發公司(DEKA Research and Development Corp)團隊研發的電動人力運輸車賽格威(Segway) [31]，其速度和方向可由駕駛人轉移重心或是車把去做控制。

兩輪自體平衡車的研究不斷的有新的控制理論，目的就是能在平衡方面能夠有更好的表現，本文也將使用兩輪自體平衡車的數學模型進行平衡控制的討論。

第二節 研究目的

控制理論中，傳統的 PID 方法雖然可以控制車體角度讓平衡車保持平衡，但因為 PID 是單一輸入單一輸出系統(SISO)，所以只能保持平衡車直立卻不能控制平衡車輪子移動的位置。而 LQR 方法卻是可以同時控制多輸入多輸出系統(MIMO)，所以透過 LQR 方法可以運算出所有狀態的增益值讓平衡車保持平衡且能夠控制平衡車輪子移動的位置。但是平衡車會有外力的干擾，例如地板摩擦力的不同，上下坡，風阻等。所以加入了模糊推論系統去補償 LQR 的增益值，讓 4 個增益值隨著位移、速度、車體角度和車體角速度的不同而改變，使平衡車的上升時間、安定時間、最大超越量以及總誤差量可以有較佳的成果。

第三節 論文架構

第一章為緒論，說明研究背景與目的。第二章為文獻探討，了解其中理論與使用方法。接著第三章為研究方法，有關兩輪自體平衡車的數學模型與控制理論將在此章做說明，第四章為研究結果與討論，說明仿真模擬與實機結果，第五章為結論與建議。

4 模糊推論結合線性二次調節器應用於兩輪平衡車之智慧型演算法設計

第二章 文獻探討

第一節 兩輪自體平衡車

兩輪自體平衡車也可以當作是可移動的倒立擺模型，擁有 2 個輪子與車體，輪子與地面接觸，而車體須避免觸碰到地面，主要目的為讓平衡車直立平衡且保持位置不改變。由於其動態非線性的特性，所以在探討非線性系統與控制領域中有許多的研究。

2002 年，Grasser, F 等人利用牛頓法推導出兩輪自體平衡車系統的動力學方程式並使用極點配置法(Pole Placement)成功使其平衡[5]。2008 年 Yamamoto, Y 對 Lego Mindstorms NXT 設計出數學模型和運動方程式，並使用 LQR 控制器成功穩定平衡車體[6]。2015 年 Jose, A 比較了傳統 PID 與 LQR 在倒立擺上的性能研究，結果顯示使用 LQR 控制器可以有效減少過衝和安定時間且響應更加穩[7]。2016 年 Oróstica, R 針對三種不同的控制器進行仿真比較，LQR 和整數階 PID 和分數階 PID(FOPID)，並使用 PSO 調整參數以穩定手推車上的倒立擺，當考慮位置跟蹤時，性能最好的是 LQR[8]。2016 年汪榮章使用了 PID 與 Fuzzy 控制車體角度，並加入了 PI 控制器來控制位置[1]。而 LQR 方法卻是可以同時控制多輸入多輸出系統(MIMO)，所以透過 LQR 方法可以運算出所有狀態的增益值讓平衡車保持平衡且能夠控制平衡車輪子移動的位置。

以上方法都可使平衡車穩定，但大多是建立在平衡車數學模型的基礎上，而建立數學模型中一般需要進行線性化處理，無法精準的反應實際的平衡車狀態，因此許多研究提出控制方法減少誤差並穩定平衡。而這些控制器已被引入許多平台，例如 Arduino 或是 LEGO Mindstorms 套件等，這兩種套件一般人都可以輕易取得，本論文也將使用 Arduino 組裝的兩輪自體平衡車並採用 Yamamoto, Y[6]的數學模型做研究，在本研究中，Matlab 將用來分析線性二次型調節器結合模糊邏輯控制以穩定系統的平台。表 2-1 為兩輪自體平衡車的相關研究。

表 2-1 兩輪自體平衡車文獻探討

作者	年份	研究內容
林哲英	2015	利用模糊推論結合基因演算法在 Lego NXT 上達到平衡，且與 LQR 比較後上升時間、最大超越量及總誤差量都取得較佳的結果。[2]
Sung, H.-C	2015	提出一種在單輪和兩輪上的幾種控制方法，通過拉格朗日建模獲得運動方程式後，測試了線性和非線性控制器如 PID，LQR 和滑動模式控制，當使用 LQR 時，為了減少車輪位置的穩態誤差而添加了 PID 來控制，改善了系統的性能。[9]
Cao, J.-R	2016	利用兩個歸屬函數車體角度與車體角速度所組成的 T-S 模糊模型於二輪自體平衡車並使用基因演算法尋找狀態回饋增益值，結果與 LQR 相比較，達到穩定平衡的所需時間減少，平衡前的震盪也有效降低。[10]

第二節 線性二次型調節器

現代控制理論中，線性二次型調節器(Linear Quadratic Regulator,LQR)是一種基於狀態空間表達式的最優控制，但需要線性化的系統狀態，其設計是根據數學上精確的性能標準來表達性能規格，然後得出控制增益的矩陣方程式的控制器，可以從兩個方面去作取捨，分別是平衡和損失成本，若給定線性時變系統模型的狀態方程(狀態維度為 n ，輸入維度為 r)。

$$\dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{u} \quad (1)$$

\mathbf{A} 為 $n \times n$ 的系統矩陣、 \mathbf{B} 為 $n \times r$ 的輸入矩陣，都是由系統本身的參數推導而成，而關於狀態向量 \mathbf{x} 和輸入向量 \mathbf{u} 的性能指標函數如下

$$J = \frac{1}{2} \int_0^{\infty} [\mathbf{x}^T \mathbf{Q} \mathbf{x} + \mathbf{u}^T \mathbf{R} \mathbf{u}] dt \quad (2)$$

為了實現最佳控制，將算出反饋矩陣 k 值使性能指標函數 J 最小化。

$$\mathbf{u}(t) = -\mathbf{R}^{-1} \mathbf{B}^T \mathbf{P} \mathbf{x}(t) = -\mathbf{K} \mathbf{x}(t) \quad (3)$$

\mathbf{P} 為代數 Riccati 方程式的解

$$\mathbf{P} \mathbf{A} + \mathbf{A}^T \mathbf{P} - \mathbf{P} \mathbf{B} \mathbf{R}^{-1} \mathbf{B}^T \mathbf{P} = -\mathbf{Q} \quad (4)$$

最佳控制取決於 \mathbf{Q} 和 \mathbf{R} 矩陣，其中 \mathbf{Q} 和 \mathbf{R} 分別表示狀態矩陣和系統輸入的加權矩陣， \mathbf{Q} 為半正定矩陣， \mathbf{R} 為正定矩陣。

但是 \mathbf{Q} 和 \mathbf{R} 沒有通用的方法來調整這些參數，對於調整到有較好的性能，上升時間(Rise Time)、超越量(Overshoot)已及安定時間(Settling Time)需要較長的時間，通常使用仿真試驗和反覆試驗法來驗證正確的參數。

2005 年 Kim, Y 等人[11]使用了 3-DOF 的建模方法並開發了線性二次調節器(LQR)控制器來平衡兩輪機器人，此機器人系統中最重要的控制變量是傾斜角，因此透過反覆試驗法調整權重，以實現更好的性能。2009 年，Wongsathan, C 和 Sirima, C 在倒立擺系統中將基因演算法(GA)應用於線性二次調節器(LQR)中的權重矩陣設計，具有良好的結果，也提到了未來的設計思路可以擴展到模糊邏輯等控制演算法[12]。2017 年 Varghese, E. S 等人將 LQR 的增益值添加到 PID 控制

中，經過反覆試驗調整到最佳控制，證明了性能相對優於單獨使用 PID[13]。2019 年 Chang, Z 等人設計了一個由極點配置和 LQR 組成的綜合控制器 PLQR，將跟蹤誤差添加到平面倒立擺系統中，以實現軌跡跟踪控制，仿真結果表明 PLQR 的跟蹤角頻率大於 LQR 的軌跡[14]。

根據以上文獻，可以得知 LQR 能夠穩定兩輪自體平衡車，但是 LQR 結合其他方法的強健性及穩定性較高，許多研究人員也不斷研究 LQR 的控制方法，因此有關於 LQR 的應用與非線性控制器將在下一節探討。表 2-2 為 LQR 結合其他應用的相關研究。

表 2-2 LQR 控制文獻探討

作者	年份	研究內容
Zhao, J、 Ruan, X	2008	研究了雙輪立式自平衡機器人的結構和數學模型，並在 Matlab 上設計了最優控制器 LQR，完成仿真與實際實驗，驗證了系統建模和所設計控制器的有效性和合理性，其中的參數是使用反覆測試法求得。[15]
Fang, J	2014	基於 LQR 控制方法，採用粒子群演算法優化 LQR 控制器的參數矩陣，提高了系統穩定性，取得了良好的控制效果，具有較高的應用價值。[16]
Banerjee, R、 Pal, A	2018	結合了 LQR 與卡爾曼濾波器也稱為線性二次高斯控制器(LQG)，結果可有效消除雜訊。[17]

第三節 模糊推論

模糊邏輯理論是由 Zadeh·L·A 教授[18]於 1965 年提出，一開始並不是作為一種控制方法，他認為不需要精確的數值輸入，但卻具有高度自適應的控制能力。如果可以將反饋控制器設計為可以接受不精確，有程度上的多或少，有雜訊的輸入，則它們將更有效，更輕鬆地實現。但是由於小型計算機能力不足，這種設定理論的方法直到 1970 年代才應用於控制系統。

儘管模糊邏輯的基本概念是在 1960 年代提出的，但 FLC 已在工業應用中得到了廣泛應用，1973 年 Zadeh 把模糊邏輯應用於控制領域[19]，後年 1975 年 Mamdani·E·H 成功將模糊控制器應用於鍋爐與蒸汽機的控制[20]，1985 年 Takagi, T 及 Sugeno, M 提出 T-S 模糊推論系統[21]，與 Mamdani 的差別在於後論使用輸入數學函式或常數做為替代，計算效率大幅提高，開啟了模糊技術的應用前景。

2011 年，Wu, J 設計了極點配置狀態反饋控制器和模糊邏輯控制器，比較仿真結果後模糊控制器獲得了較好的動態性能，且具有較好的理論和應用研究價值[22]。2016 年汪榮章提出了車體平衡、載物與速度三種模糊控制器用來調控二輪自體平衡車在不同狀態的平衡機制，使兩輪自體平衡車在不同狀態當中的轉換更加穩定，並且在平衡狀態時降低晃動的幅度[1]。

2020 年 Al-Isawi, M 等人提出了一種用於自動駕駛無人機（UAV）的縱向控制器的設計。是一種基於智能演算法的雙迴路（內外迴路）控制，內部反饋環路控制器是線性二次調節器（LQR），可提供穩定的（自適應）穩定性。外環控制器基於 Fuzzy-PID 演算法提供參考信號追蹤，將控制飛機的位置和速度，並在仿真實驗中得到驗證，結果與 Fuzzy PID 相比安定時間，上升時間，最大超越量都得到了改善[23]。

表 2-3 為模糊控制的相關應用研究。

表 2-3 模糊控制文獻探討

作者	年份	研究內容
Wu, J	2012	分析了具有模糊 PD 控制方法的兩輪自體平衡車，結果表明其演算法可以成功實現對兩輪自體平衡車的自平衡控制。[24]
Bharali, J 、 Buragohain, M	2016	設計用於汽車主動懸架系統的三種類型的主動控制器，分別是基於 LQR 的模糊控制器，模糊 PID 控制器和線性二次控制器 (LQR)，仿真結果表明，在減少安定時間和最大超越量方面模糊 LQR 控制器的主動懸架系統具有更好的性能表現。[25]
Hazem, Z 、 Fotuhi, M. J 、 Bingül, Z	2020	設計了基於模糊的線性二次調節器 (FLQR) 和線性二次高斯 (FLQG) 控制器開發用於雙鏈旋轉倒立擺 (DLRIP) 系統的穩定性控制，在內部和外部干擾下的強健性分析，比較了控制器的動態響應，仿真後顯示相比經典 LQR 與 LQG 控制器，FLQR 和 FLQG 的動態響應表現較佳。[26]

第四節 線性二次型調節器結合模糊推論

由於兩輪自體平衡車屬於非線性系統，雖然已經通過動力學計算出線性化系統，但是其內部摩擦力的干擾，抑或是外部阻力的影響，都會使系統性能降低，因此本文加入了模糊推論系統去補償 LQR 的增益值，使其上升時間、最大超越量以及總誤差量在受到干擾的情況下，也可以解決在多輸入多輸出情況下對精確數學模型的過分依賴和模糊控制器的複雜處理問題，而呈現較佳的成果。

2018 年 Chantarachit, S 利用 LQR-Fuzzy 控制器設計和仿真具有雙飛輪模型的獨輪車機器人，主要結構控制是 LQR 控制器，並使用 Fuzzy 控制器來調節閉環控制器增益，採用 4 個輸入俯仰 (pitch)、偏擺 (yaw)、翻滾(roll)及輪子位移 (wheel position)和 1 個輸出來調整 LQR 控制器的增益，仿真結果顯示與傳統的 LQR 相比，FLQR 可以減少大約 50% 的最大超越量[27]。2019 年 Kudinov, Y 等人分析了線性二次 LQR 和 FLQR 控制器。結果顯示，在相同條件下，兩個控制器都沒有超過最大超越量和靜態誤差，上升時間也快，但是 FLQR 控制器具有更高的性能[28]。2019 年 Devi, P. S 和 Santhi, R. V 在多區域負載頻率控制 LFC 模型加入電力需求響應(DR)控制迴路 (稱為 LFC-DR)通常由標準控制器 (如 PI、PD、PID 控制器)調整，因為它們具有恆定的增益。因此，它們無法針對各種運行條件和各種負載變化獲得理想的動態性能。因此提出了 FLQR 控制傳統的 LFC-DR。還考慮了 DR-CDL 的影響，即 (需求響應通信延遲延遲) 在設計中的影響。仿真結果顯示，所提出的控制器可以保證整個閉環 LFC-DR 系統的穩定性，從而有效地提高了系統的動態性能，並且在不同的操作場景下優於傳統的控制[29]。

2020 年 Adamu, J. K 等人提出了一種用於雙旋轉倒立擺(DRIP)系統的 FLQR 控制器的設計與 PID-LQR 控制器進行了比較。仿真結果顯示，所提出的 FLQR 控制器與 PID-LQR 控制器相比有更好的性能，特別是在有干擾的情況下，可以被認為是控制高度非線性、不穩定與傳統的 LQR 相比，FLQR 可以減少大約 50 % 的最大超越量、非最小相位和欠驅動機械系統選擇[30]。2021 年 Jeon, T 和 Paek, I 設計了一種基於模糊控制演算法的線性二次調節器(FLQR)，用於可變速可變螺距風力發電機。仿真結果顯示，與 PI 控制相比，FLQR 控制的塔架振動減少了

12.50%，功率偏差減少了 38.93%。因此 FLQR 控制可提高風力發電機的功率性能和結構穩定性[4]。

根據以上研究，FLQR 可以有效的減少最大超越量，而兩輪自體平衡車最重要的就是在馬達轉速允許的情況下，維持著直立平衡，當傾斜角度超過車體傾斜角度所能承受的範圍之外，就會無法平衡而導致翻覆，所以最大超越量的減少對於兩輪自體平衡車的控制是重要指標之一。

第三章 研究方法

第一節 使用平台介紹

本研究將使用慧淨電子所開發的 Arduino Leonardo 所組裝出的兩輪自體平衡車[32]，其架構如圖 3-1

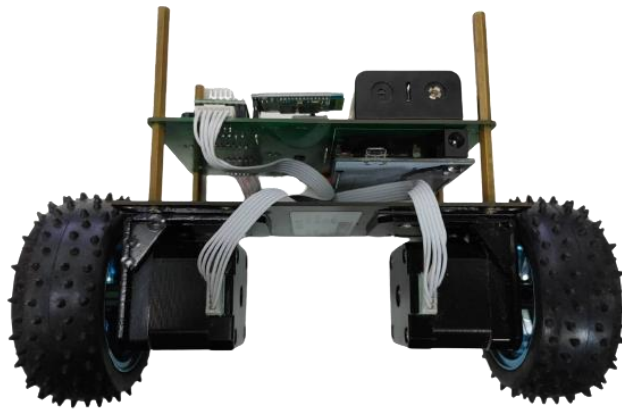


圖 3-1 兩輪平衡車架構

兩輪自體平衡車主要硬件為 Arduino Leonardo 開發版、感測器與馬達，透過感測器取得目前角度，而 Arduino Leonardo 開發版計算出所需輸出，最後驅動器驅動馬達轉動，達到平衡的效果。Arduino Leonardo 開發版是一種微型控制器，只需要接 USB 到電腦，再使用 Arduino 的開源軟體平台，可以隨時編譯，也可以跟其他微控制器一樣透過取得輸入而控制各種電子設備的輸出。



圖 3-2 Arduino Leonardo 開發版

MPU6050 陀螺儀感測器如圖 3-3，可以測量 ± 2000 度/秒的三軸角速度變化，取樣速率約 200 次/秒，可以有效且即時的取得當前角度，本次研究將使用 200 次/秒的取樣速率做計算。



圖 3-3 MPU6050 陀螺儀感測器

馬達為 42 步進馬達如圖 3-4，透過脈衝的輸入即可實現高精準度的定位，對於穩定平衡有良好的控制作用。

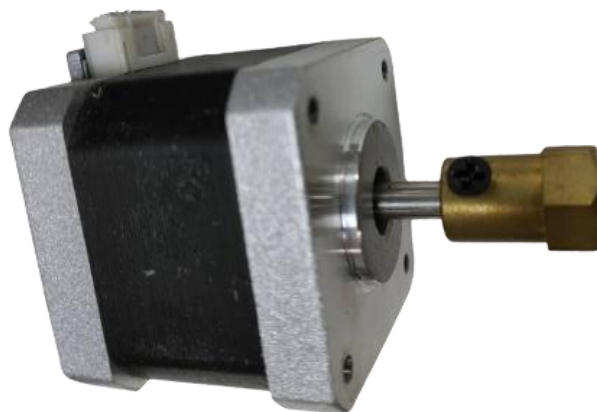


圖 3-4 42 步進馬達

馬達驅動使用了 A4988 馬達驅動器如圖 3-5，輸出單位為脈衝個數，全步進模式旋轉一圈需 200 個脈衝，所以一個脈衝可旋轉 1.8 度，可再細分為半步進、1/4、1/8 和 1/16 步進模式。藉由 Arduino Leonardo 版的輸出來調整平衡所需要的脈衝，本次研究將使用全步進模式。



圖 3-5 A4988 馬達驅動器

因此利用感測器，可以將讀取的數據轉化成輪子角度、輪子角速度、車體傾斜角度及車體傾斜角速度四個數值當作系統的輸出，兩輪自體平衡車的坐標系統在圖 3-6 呈現，其中 θ 是輪子角度， ψ 是車體傾斜角度。

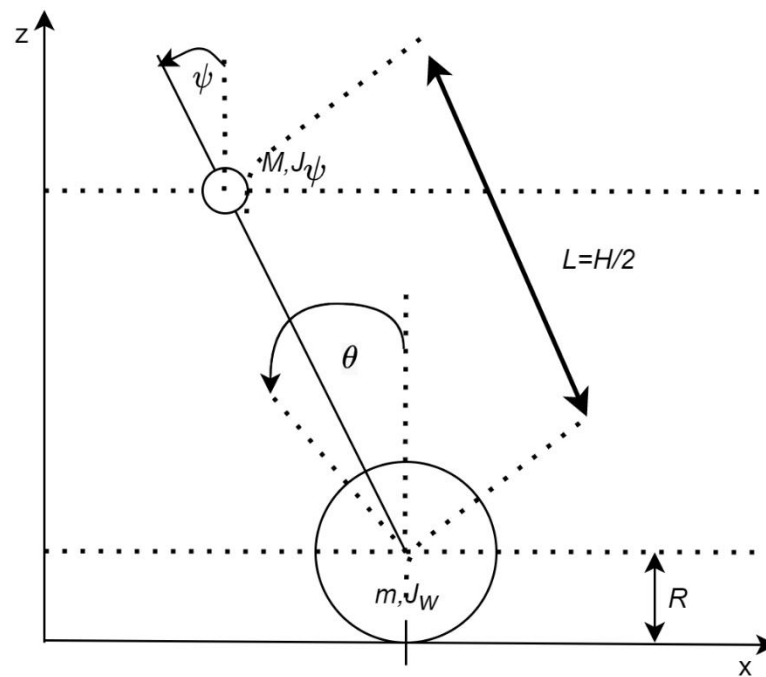


圖 3-6 兩輪自體平衡車坐標系統

系統模型相關參數為手動測量與參考自[2]，馬達參考自[33]的 17HS4602 型如表 3-1。

表 3-1 系統參數

名稱	代號	參數
輪子重量	m	0.05 (kg)
車體重量	M	1 (kg)
輪子半徑	R	0.04 (m)
車體深度	D	0.083 (m)
車體高度	h	0.154 (m)
車體寬度	W	0.148 (m)
輪軸與車體質心的距離	L	0.077 (m)
輪子的轉動慣量	J_w	$4 \times 10^{-5} \text{ (kgm}^2\text{)}$
車體的轉動慣量	J_ψ	$2 \times 10^{-3} \text{ (kgm}^2\text{)}$
馬達的轉動慣量	J_d	$1 \times 10^{-5} \text{ (kgm}^2\text{)}$
馬達的電阻	R_m	4.35 (Ω)
馬達的 back EMF 常數	K_b	$0.2647 \left(\text{V} \frac{\text{sec}}{\text{rad}} \right)$
馬達的扭力常數	K_t	0.2647 (Nm/A)
重力	g	$9.81 \left(\frac{\text{m}}{\text{sec}^2} \right)$
齒輪比	n	1
車體與馬達的摩擦係數	f_m	0.0022
輪子與地面的摩擦係數	f_w	0

第二節 機器人系統模型

本文的平衡車數學動態模型參考自 Yamamoto, Y[6] 的方法進行推導，兩輪自體平衡車所受的力可分為平移的動能(輪子和車體)、旋轉的動能(輪子、車體與馬達)和位能，拉格朗日量 L 的定義為動能 T 減去位能 U (5)，拉格朗日方程式為(6)(7)。

$$L = T - U \quad (5)$$

$$F_\theta = \frac{d}{dt} \left(\frac{\partial L}{\partial \dot{\theta}} \right) - \frac{\partial L}{\partial \theta} \quad (6)$$

$$F_\psi = \frac{d}{dt} \left(\frac{\partial L}{\partial \dot{\psi}} \right) - \frac{\partial L}{\partial \psi} \quad (7)$$

根據拉格朗日方程式推導出下列(8)式與(9)式

$$F_\theta = [(2m + M)R^2 + 2J_w + 2n^2J_m]\ddot{\theta} \quad (8)$$

$$+ (MLR\cos\psi - 2n^2J_m)\ddot{\psi} - MLR\dot{\psi}^2\sin\psi$$

$$F_\psi = (MLR\cos\psi - 2n^2J_m)\ddot{\theta} + (ML^2 + J_\psi + 2n^2J_m)\ddot{\psi} \quad (9)$$

$$- MgL\sin\psi - ML^2\dot{\theta}^2\sin\psi\cos\psi$$

式子(8)(9)可以在平衡點處線性化方程式，使車體傾角 ψ 趨近於零度($\sin\psi \rightarrow \psi$, $\cos\psi \rightarrow 1$)，並且忽略二次項，動力學方程式更改如下

$$F_\theta = [(2m + M)R^2 + 2J_w + 2n^2J_m]\ddot{\theta} + (MLR - 2n^2J_m)\ddot{\psi} \quad (10)$$

$$F_\psi = (MLR - 2n^2J_m)\ddot{\theta} + (ML^2 + J_\psi + 2n^2J_m)\ddot{\psi} - MgL\psi \quad (11)$$

接下來計算馬達的輸出與摩擦力，輪子會受到馬達的力矩和輪子磨擦力的影響，車體會受到相同馬達力矩的影響但是受力方向相反，因此馬達的廣義力如下

$$F_\theta = 2nK_t i + 2f_m(\dot{\psi} - \dot{\theta}) - f_w\dot{\theta} \quad (12)$$

$$F_\psi = -2nK_t i - 2f_m(\dot{\psi} - \dot{\theta}) \quad (13)$$

因為上下坡跟地板摩擦力不同所需的輸出動力不一樣，因此通過改變 PWM 脈衝的佔空比來控制馬達的轉速，所以需要評估馬達的電流 i 與電壓 v 之間的關係。如果馬達內部摩擦與電感忽略不計，電流 i 與左右馬達 v_l 、 v_r 的方程式如下：

$$\dot{i} = \frac{v + K_b(\dot{\psi} - \dot{\theta})}{R_m} \quad (14)$$

$$F_\theta = \alpha(v_l + v_r) - 2(\beta + f_w)\dot{\theta} + 2\beta\dot{\psi} \quad (15)$$

$$F_\psi = -\alpha(v_l + v_r) + 2\dot{\theta} - 2\beta\dot{\psi} \quad (16)$$

$$\alpha = \frac{nK_t}{R_m}, \beta = \frac{nK_tK_b}{R_m} + f_m \quad (17)$$

基於現代控制理論導出狀態方程式如下

$$E \begin{bmatrix} \ddot{\theta} \\ \ddot{\psi} \end{bmatrix} + I \begin{bmatrix} \dot{\theta} \\ \dot{\psi} \end{bmatrix} + G \begin{bmatrix} \theta \\ \psi \end{bmatrix} = H \begin{bmatrix} v_l \\ v_r \end{bmatrix} \quad (18)$$

$$E = \begin{bmatrix} (2m + M)R^2 + 2J_w + 2n^2J_m & MLR - 2n^2J_m \\ MLR - 2n^2J_m & ML^2 + J_\psi + 2n^2J_m \end{bmatrix} \quad (19)$$

$$= \begin{bmatrix} E_{11} & E_{12} \\ E_{21} & E_{22} \end{bmatrix}$$

$$I = -2 \begin{bmatrix} \beta + f_w & -\beta \\ -\beta & \beta \end{bmatrix} \quad (20)$$

$$G = \begin{bmatrix} 0 & 0 \\ 0 & -MgL \end{bmatrix} = \begin{bmatrix} G_{11} & G_{12} \\ G_{21} & G_{22} \end{bmatrix} \quad (21)$$

$$H = \begin{bmatrix} \alpha & \alpha \\ -\alpha & -\alpha \end{bmatrix} \quad (22)$$

於是將以下變量 \mathbf{x} 視為狀態空間，其中有 4 個狀態。 \mathbf{u} 為左右馬達輸入。

$$\mathbf{x} = [\theta \quad \psi \quad \dot{\theta} \quad \dot{\psi}]^T, \mathbf{u} = [v_l \quad v_r]^T \quad (23)$$

因此可以從方程式(18)推導出兩輪自體平衡車的狀態方程式如下

$$\dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{u} \quad (24)$$

$$\mathbf{A} = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & A_{32} & A_{33} & A_{34} \\ 0 & A_{42} & A_{43} & A_{44} \end{bmatrix}, \mathbf{B} = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ B_3 & B_3 \\ B_4 & B_4 \end{bmatrix} \quad (25)$$

$$A_{32} = \frac{G_{22}E_{12}}{\det(E)}, A_{42} = \frac{-G_{22}E_{11}}{\det(E)} \quad (26)$$

$$A_{33} = \frac{-2[(\beta + f_w)E_{22} + \beta E_{12}]}{\det(E)} \quad (27)$$

$$A_{43} = \frac{2[(\beta + f_w)E_{12} + \beta E_{11}]}{\det(E)} \quad (28)$$

$$A_{34} = \frac{2\beta[E_{22} + E_{12}]}{\det(E)} \quad (29)$$

$$A_{44} = \frac{-2\beta[E_{11} + E_{12}]}{\det(E)} \quad (30)$$

$$B_3 = \frac{\alpha[E_{22} + E_{12}]}{\det(E)} \quad (31)$$

$$B_4 = \frac{-\alpha[E_{11} + E_{12}]}{\det(E)} \quad (32)$$

$$\det(E) = E_{11}E_{22} - E_{12}^2 \quad (33)$$

計算出近似平衡點的線性系統模型如下

$$\begin{bmatrix} \dot{\theta} \\ \dot{\psi} \\ \ddot{\theta} \\ \ddot{\psi} \end{bmatrix} = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & -429.8 & -78.48 & 78.48 \\ 0 & 261.3 & 35.15 & -35.15 \end{bmatrix} \begin{bmatrix} \theta \\ \psi \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 148.9 & 148.9 \\ -66.7 & -66.7 \end{bmatrix} \begin{bmatrix} v_l \\ v_r \end{bmatrix} \quad (34)$$

第三節 LQR 控制器設計

經過計算系統的可控性，可以得知系統是可控制的，利用 LQR 方法獲得狀態回饋增益值，並轉換為空間矩陣狀態。因此令輸入 $u = -Kx$ ， $\dot{x} = (A - BK)x$ ， $K = [k1 \ k2 \ k3 \ k4]$ 。

但是 LQR 仍存在一些問題需要修正，在穩定車身傾斜角度和輪子角度的同時，可以穩定車身而不會產生穩態誤差，但輪子角度可能會有穩態誤差。為了消除位移的穩態誤差使平衡車可以回到初始位置如圖 3-7，所以在控制迴路中添加積分器 $k5$ ，狀態向量 x 也會新增輪子角度積分的變量 $\int \theta dt$ ，因此狀態向量 $x = [\theta \ \psi \ \dot{\theta} \ \dot{\psi} \ \int \theta dt]^T$ ，於是增益回饋變成下列式子：

$$K = [k1 \ k2 \ k3 \ k4 \ k5] \quad (35)$$

$k1$ = 輪子角度的增益值

$k2$ = 車體角度的增益值

$k3$ = 輪子角速度的增益值

$k4$ = 車體角速度的增益值

$k5$ = 輪子角度積分的增益值

式子(34)的線性系統模型更改為

$$\begin{bmatrix} \dot{\theta} \\ \dot{\psi} \\ \ddot{\theta} \\ \ddot{\psi} \\ \theta \end{bmatrix} = \begin{bmatrix} 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & -429.8 & -78.48 & 78.48 & 0 \\ 0 & 261.3 & 35.15 & -35.15 & 0 \\ 1 & 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} \theta \\ \psi \\ \dot{\theta} \\ \dot{\psi} \\ \int \theta dt \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 148.9 & 148.9 \\ -66.7 & -66.7 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} v_l \\ v_r \end{bmatrix} \quad (36)$$

Q 加權矩陣中，車身角度的性能較為重要，因此 Q_{22} 的值將會大於其他值，從反覆試驗法求出車體角度的 Q 調整為 1000，選擇如下

$$Q = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1000 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}, R = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \quad (37)$$

然後，本研究使用了 MATLAB 函數 $K = \text{lqr}(A, B, Q, R)$ 來計算式子 (3) (4) 以
求出增益值。運行後得到狀態反饋矩陣為

$$K = [-1.0596 \quad -34.1132 \quad -1.1094 \quad -3.2127 \quad -0.5099] \quad (38)$$

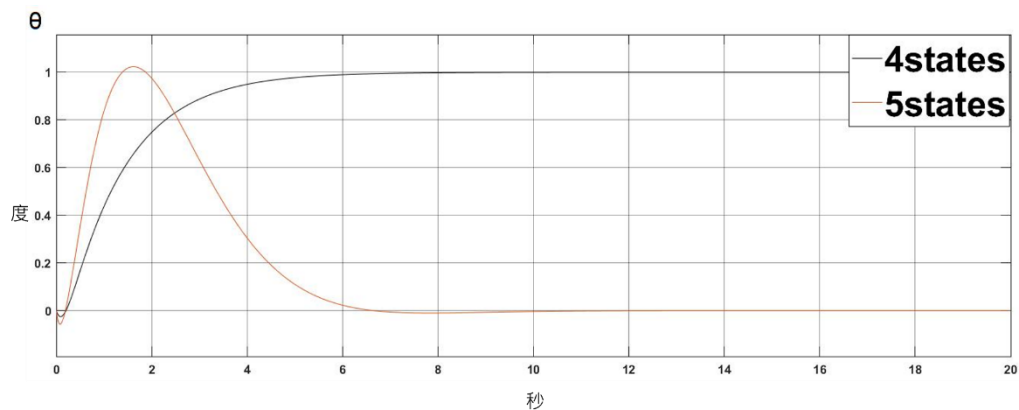


圖 3-7 4 個狀態與 5 個狀態的輪子角度比較圖

但是以上的最優增益是在無干擾且平衡角度近似 0 度的情況下，但平衡車移動時的車體角度與阻力卻是會變動的。而從 [1] 論文中，可以得知增益值不同將會對上升時間，最大超越量及上升時間產生影響如表 3-2，因此將會加入模糊規則來因為車體角度、車體角速度、輪子角度和輪子角速度的不同，而增加或減少增益值。

表 3-2 調整 k 值的影響 [1]

調整方式	Rise Time	Overshoot	Settling Time
k_p 調高	下降	上升	微幅上升
k_i 調高	微幅下降	上升	上升
k_d 調高	微幅下降	下降	下降

第四節 模糊規則設計

根據[1]的研究中，可以得知調整增益值的影響，於是利用反覆試驗法找出合適的增益值上下限，使控制輸入向量的結果不會超過馬達上下限(1080 度/秒)，令控制輸入向量的數值保持在邊界範圍內，得到的上下界範圍如表 3-3

表 3-3 增益值的上下界範圍

調整之變數	上界值	下界值
k_1 輪子角度增益值	-0.14	-4.2
k_2 車體角度增益值	-15	-49
k_3 輪子角速度增益值	-1.1	-1.5
k_4 車體角速度增益值	-2.1	-4.2
k_5 輪子角度積分增益值	-5.099	-5.099

對於輪子角度的模糊規則，本研究將使用三角形歸屬函數的 Mamdani 模糊模型以調節控制器的反饋增益。位移與速度是根據輪子角度和輪子的半徑所計算，輸入為位移和速度，輸出為位移 u_e 和速度 u_f 。模糊集的寬度範圍是經過表 3-3 找出增益值上下限確定的。對於兩個輸入，位移範圍為 $[-0.5, 0.5]$ ，速度範圍為 $[-0.5, 0.5]$ ，位移輸出範圍為 $[-3.1404, 3.1404]$ ，速度輸出範圍為 $[-0.3906, 0.3906]$ ，輸入的模糊隸屬函數如圖 3-8、圖 3-9，輸出如圖 3-10、圖 3-11 所示。

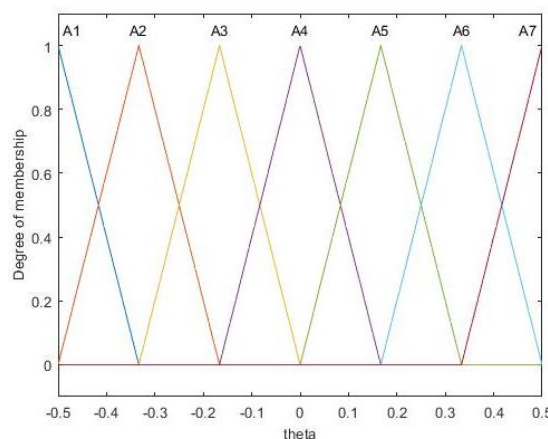


圖 3-8 位移輸入

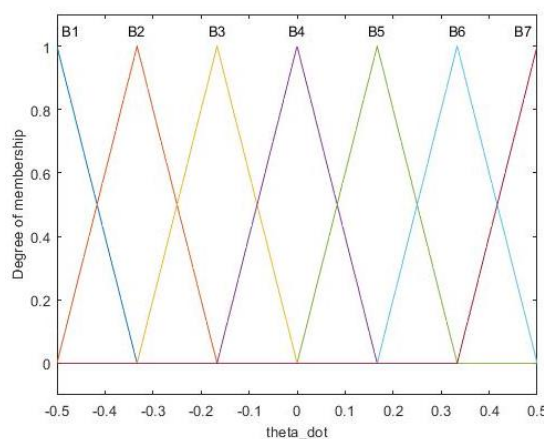


圖 3-9 速度輸入

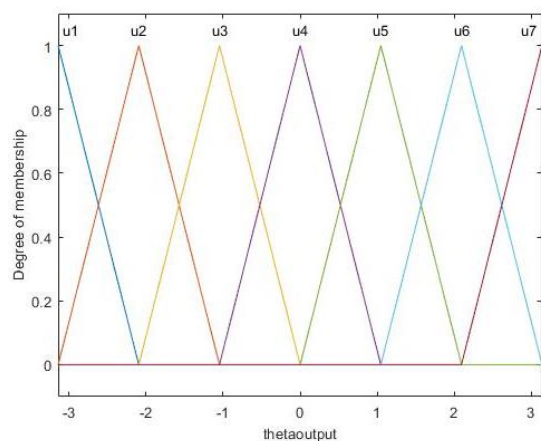


圖 3-10 位移輸出

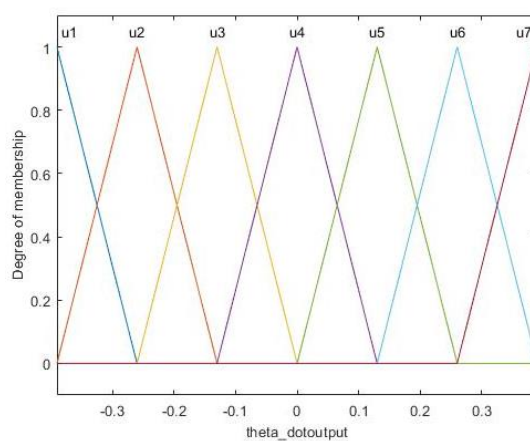


圖 3-11 速度輸出

在模糊隸屬函數有 7 個時，位移與速度將被劃分為 0， ± 0.1667 ， ± 0.333 ， ± 0.5 ，位移單位為公尺，速度為公尺/秒。接著將計算輸出的值，模糊規則表示如下：

$$\text{If } \theta \text{ is } A_i \text{ then } u = u_e \quad (39)$$

$$\text{If } \dot{\theta} \text{ is } B_j \text{ then } u = u_f$$

A_i 為位移的模糊集， $i \in \{1, 2, 3, 4, 5, 6, 7\}$

B_j 為速度的模糊集， $j \in \{1, 2, 3, 4, 5, 6, 7\}$

u_e 為位移的模糊規則輸出， $e \in \{1, 2, 3, 4, 5, 6, 7\}$

u_f 為速度的模糊規則輸出， $f \in \{1, 2, 3, 4, 5, 6, 7\}$

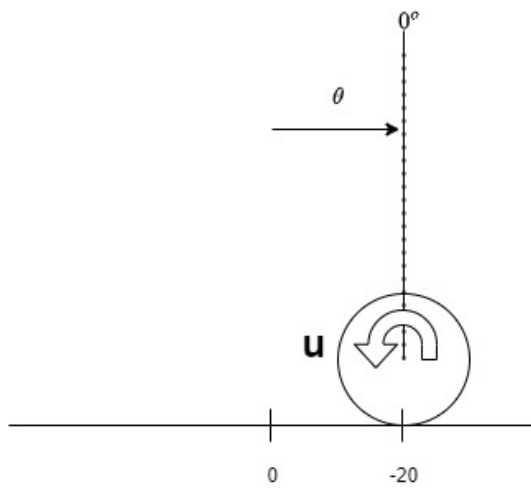


圖 3-12 $\theta = -20$

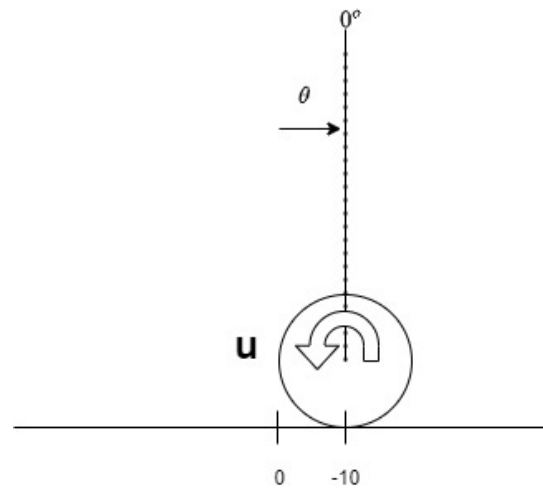


圖 3-13 $\theta = -10$

假設初始位移為-20 時，車體傾斜角度為 0 如圖 3-12，輪子為了回到原點，將會產生位移與速度。當車體傾斜角度同樣為 0，但初始位移為-10 時為如圖 3-13，車體一樣會往原點的方向前進，但會比初始位移-20 度更快到達原點，所需要的馬達輸出也較小，位移與速度也比較小，因此當位移偏移量較大時，輪子角度的增益值應隨著偏移大小去改變。

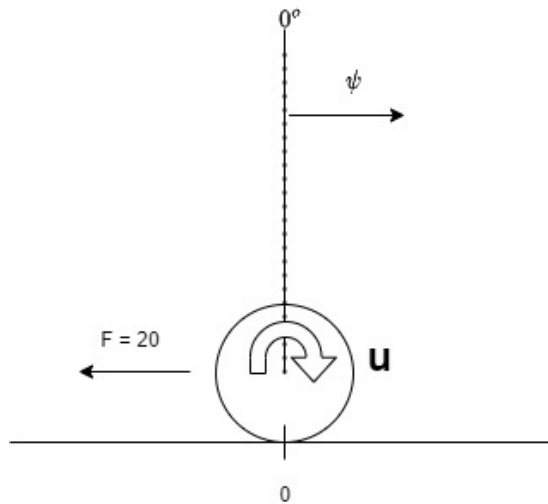


圖 3-14 $F = 20$

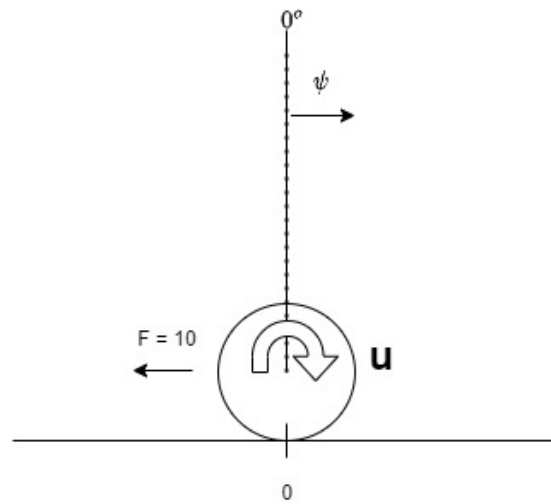


圖 3-15 $F = 10$

當初始位移為 0，車體傾斜角度為 0，當受到外力 F 為 20 如圖 3-14 與外力為 10 如圖 3-15 比較下，外力 20 的位移與速度將比 10 來的高，相對的車體向反方向的傾斜角度也較高。

因此當位移和速度較大的情況下，可以說明兩輪自體平衡車的受力(例如風)較大，才會使輪子移動幅度加大，這種情況下應該使輪子角度與輪子角速度的增益值隨著誤差大小做補償，使輪子角度可以有效降低最大位移量且平衡。

對於車體傾斜角度的模糊規則，輸入為車體角度與車體角速度，輸出為車體角度 u_g 與車體角速度 u_h 。模糊集的寬度範圍是同樣是根據表 3-3 確定的。對於兩個輸入，車體角度範圍為 $[-21\ 21]$ ，車體角速度範圍為 $[-14\ 14]$ ，車體角度輸出範圍為 $[-14.8868\ 14.8868]$ ，車體角速度輸出範圍為 $[-1.3073\ 1.3073]$ ，輸入和輸出的模糊隸屬函數如下圖所示：

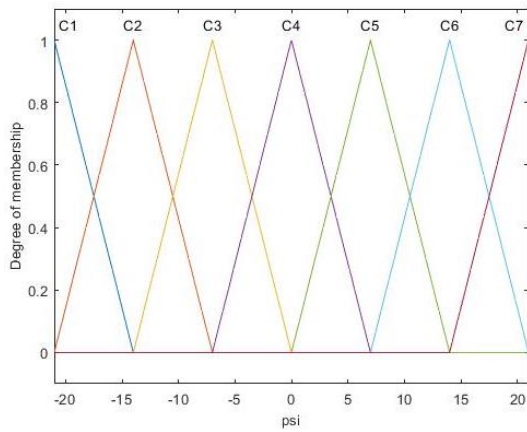


圖 3-16 車體角度輸入

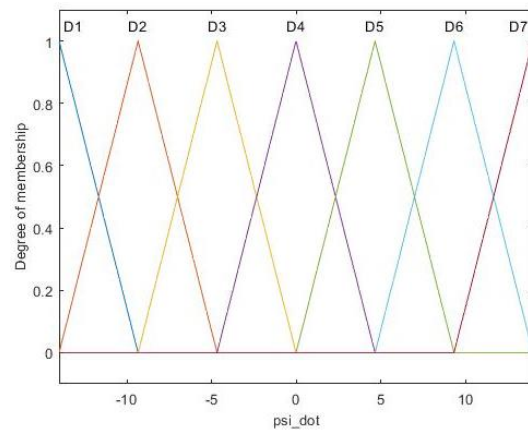


圖 3-17 車體角速度輸入

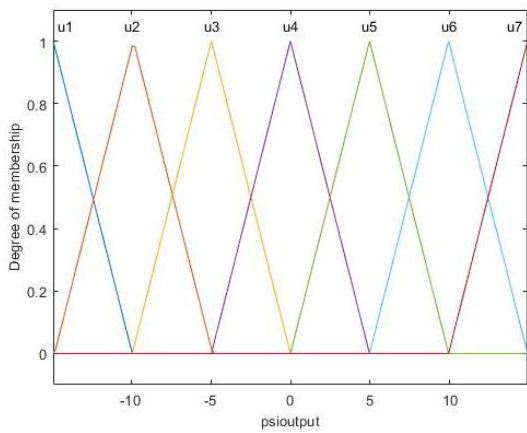


圖 3-18 車體角度輸出

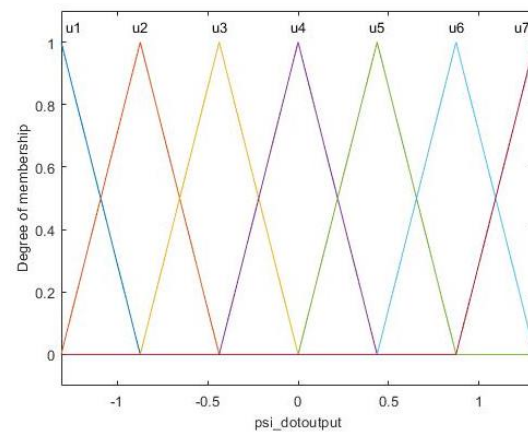


圖 3-19 車體角速度輸出

在模糊隸屬函數有 7 個時，角度將被劃分為 $0, \pm 7, \pm 14, \pm 21$ ，單位為角度，角速度將被劃分為 $0, \pm 4.668, \pm 9.334, \pm 14$ ，單位為弧度/秒。接著將計算輸出的值，模糊規則表示如下：

$$\text{If } \psi \text{ is } C_l \text{ then } u = u_g \quad (40)$$

$$\text{If } \dot{\psi} \text{ is } D_m \text{ then } u = u_h$$

C_l 是車體角度的模糊集， $l \in \{1, 2, 3, 4, 5, 6, 7\}$

D_m 是車體角速度的模糊集， $m \in \{1, 2, 3, 4, 5, 6, 7\}$

u_g 是車體角度的模糊規則輸出， $g \in \{1, 2, 3, 4, 5, 6, 7\}$

u_h 是車體角速度的模糊規則輸出， $h \in \{1, 2, 3, 4, 5, 6, 7\}$

根據狀態向量 x ，可以得知目前的輪子角度、車體角度、輪子角速度、車體角速度、和輪子角度積分，而平衡車因為有向前向後走的移動平衡方式，根據第 (3) 式 $u = -kx$ 的回授增益控制方式，因此狀態向量 x 的正負狀態會影響控制輸入 u 的正負數值，從而決定平衡車該往前或往後行駛。

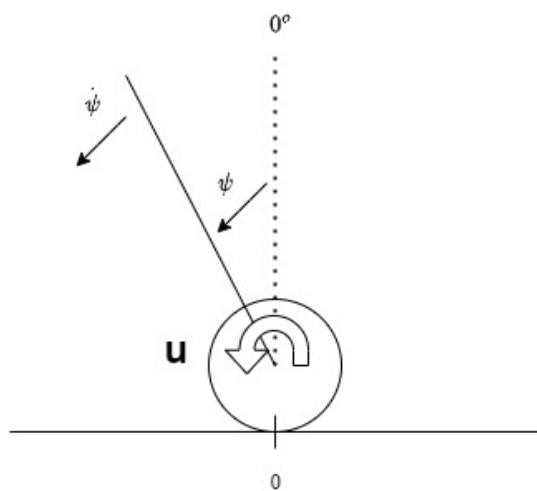


圖 3-20 $\psi = 20, \dot{\psi} = 10$

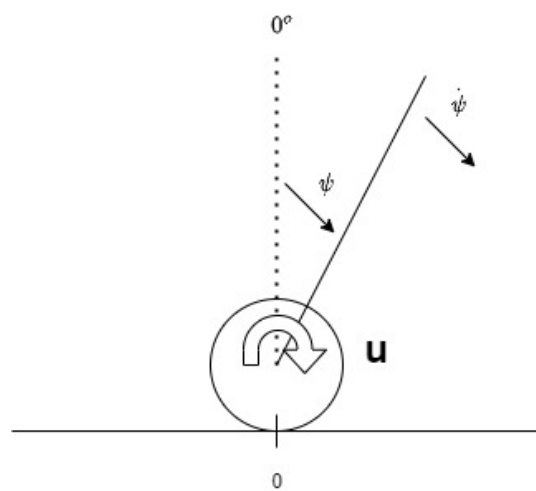
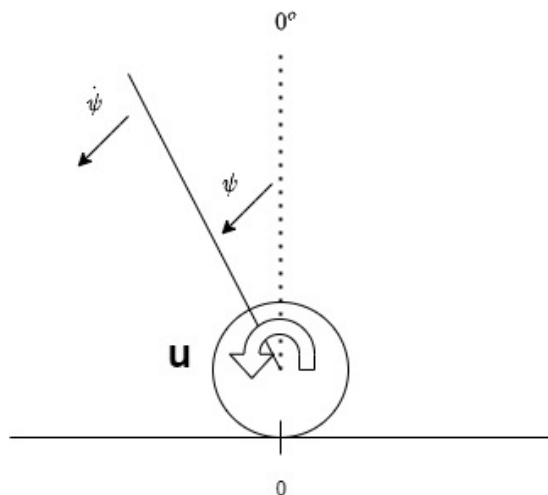
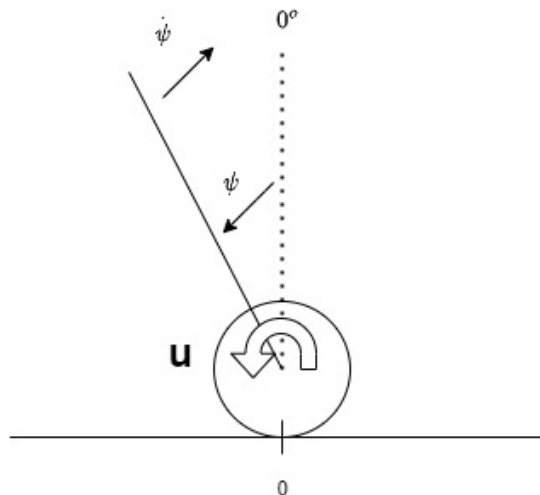


圖 3-21 $\psi = -20, \dot{\psi} = -10$

當車體傾斜角度為 20 度且車體傾斜角速度為 10 弧度/秒如圖 3-20 與車體傾斜角度為-20 度且車體角速度為-10 弧度/秒如圖 3-21，兩者所產生的 u 是相同的數值但是卻相反方向性的關係。

圖 3-22 $\psi = 20$, $\dot{\psi} = 10$ 圖 3-23 $\psi = 20$, $\dot{\psi} = -10$

而當車體傾斜角度相同時，車體傾斜角速度不同時，兩者產生的 u 的數值也會不同。例如:當車體傾斜度為 20 度且車體傾斜角速度為 10 弧度/秒如圖 3-22 與車體傾斜度為 20 度且車體角速度為-10 弧度/秒如圖 3-23，兩者所產生的 u 是同方向性關係，但是輸出的數值卻會因為車體傾斜角速度的不同而改變。

表 3-4 為 4 個輸入的模糊規則，由 Degree1 到 Degree7 依序為負大、負中、負小、零、正小、正中、正大。

表 3-4 模糊規則表

	Degree1	Degree2	Degree3	Degree4	Degree5	Degree6	Degree7
θ	A1	A2	A3	A4	A5	A6	A7
$\dot{\theta}$	B1	B2	B3	B4	B5	B6	B7
ψ	C1	C2	C3	C4	C5	C6	C7
$\dot{\psi}$	D1	D2	D3	D4	D5	D6	D7

第五節 LQR 模糊控制器(FLQR)

FLQR 結合了 LQR 與模糊邏輯控制器的優點，LQR 控制器可以穩定平衡且保持位置，模糊控制可以應對系統的非線性，因此設計了一種方法來補償輸出增益值，根據模擬結果，動態響應將比單獨使用 LQR 有更好的性能。FLQR 滿足了動態響應中的最大超越量和穩態誤差及上升時間小的要求。圖 3-24 描述了輪子角度、輪子角速度、車輪傾斜角度和車輪傾斜角速度控制的 FLQR 控制器。

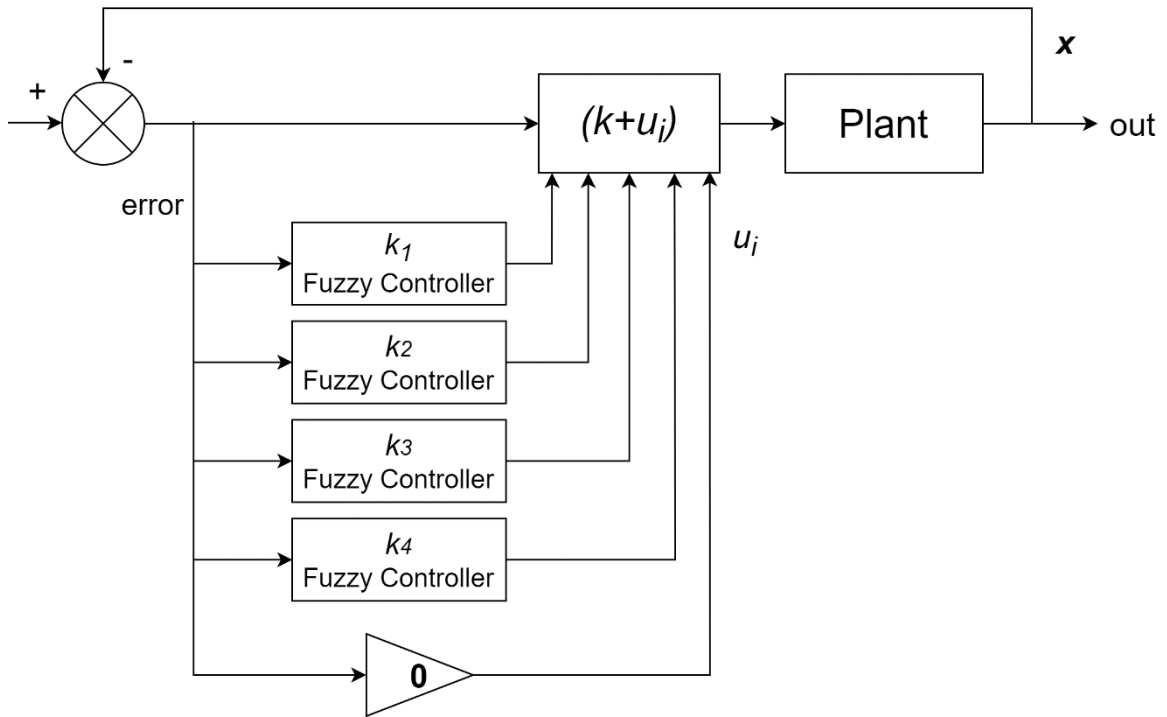


圖 3-24 FLQR 控制器

FLQR 將應用在兩輪自體平衡車的數學模型，模糊控制器中有 4 個輸入與 4 個輸出，輸入分別為位移、速度、車體傾斜角度和車體傾斜角速度，速度與車體傾斜角速度誤差是通過採樣間格來計算，利用模糊控制器規則來補償狀態反饋增益 $K = [k_1 \ k_2 \ k_3 \ k_4]$ 。

FLQR 是指找到狀態 $x = [\theta \ \psi \ \dot{\theta} \ \dot{\psi} \ \int \theta dt]^T$ 反饋的 5 個 LQR 增益值，輪子角度 k_1 、輪子角速度 k_3 、車體傾斜角度 k_2 、車體傾斜角速度 k_4 和輪子角度積分 k_5 後，並根據模糊控制器規則的輸出補償原本 LQR 的 $[k_1 \ k_2 \ k_3 \ k_4]$ 以滿足不同的要求，當位移、速度、車體傾斜角度和車體傾斜角速度不同時能獲得不同的增益值，使兩輪自體平衡車產生良好的平衡性能，輸出的 FLQR 模糊規則如下：

$$\text{If } z \text{ is } E_i \text{ then } \mathbf{u} = -(k + u_i)\mathbf{x} \quad (41)$$

z 為 5 個輸入位移、車體角度、速度、車體角速度、輪子角速度

E_i 是歸屬函數， $i \in \{1, 2, \dots, j\}$

a_i 為輸入角度參數

O_i 為輸出範圍參數

$$u_i = \sum_{i=1}^j E_i(z) O_i \quad (42)$$

若模糊規則數為 7 個時，歸屬函數如下

$$\begin{aligned} E_1(z) &= \begin{cases} 1, & z < a_1 \\ \frac{z - a_2}{a_1 - a_2}, & a_1 \leq z \leq a_2 \\ 0, & z > a_2 \end{cases} \\ E_2, E_3, \dots, E_6(z) &= \begin{cases} \frac{z - a_{k+1}}{a_k - a_{k+1}}, & a_k \leq z \leq a_{k+1} \\ \frac{z - a_{k-1}}{a_k - a_{k-1}}, & a_{k-1} \leq z \leq a_k \\ 0, & z > a_{k+1} \text{ or } z < a_{k-1} \end{cases} \\ E_7(z) &= \begin{cases} 1, & z > a_7 \\ \frac{z - a_6}{a_7 - a_6}, & a_6 \leq z \leq a_7 \\ 0, & z < a_6 \end{cases} \end{aligned} \quad (43)$$

若誤差都為 0 則代表模糊輸出 $u_i = 0$ ，控制輸入 $\mathbf{u} = -K\mathbf{x}$ 。

第四章 研究結果與討論

第一節 模擬性能比較

根據增益值的上下限界值，可以給定模糊控制輸出的調整範圍，但是當處於臨界點時，系統較為不穩定，因此將速度的模糊控制輸出乘上 0.8，車體角速度的模糊控制輸出乘上 0.5，而位移和車體角度的模糊控制輸出乘上 0.3。若給定一個相同的推力，仿真模擬的結果如圖 4-1、圖 4-2、圖 4-3、圖 4-4 所示。

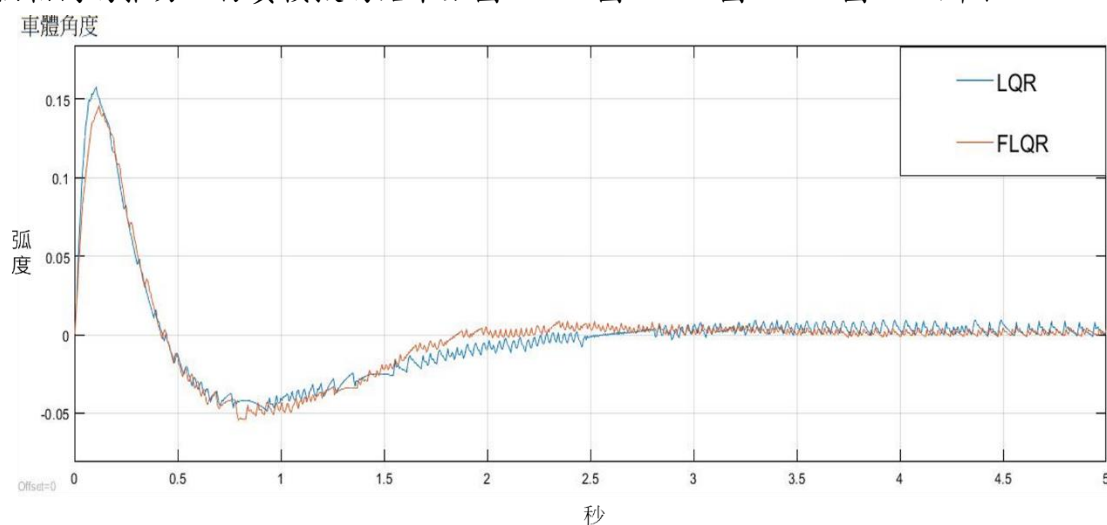


圖 4-1 車體傾斜弧度動態響應圖

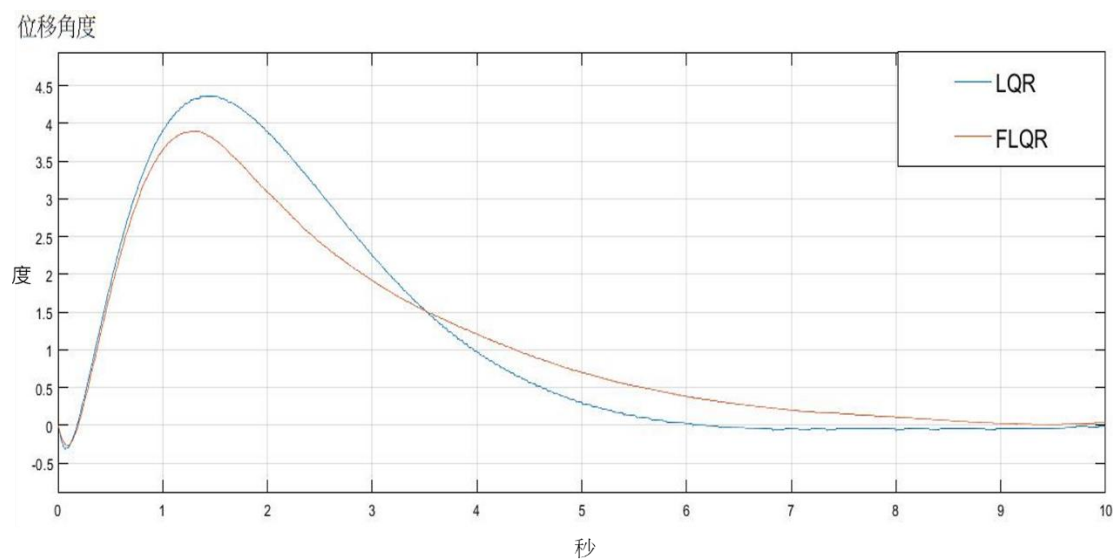


圖 4-2 位移的動態響應圖

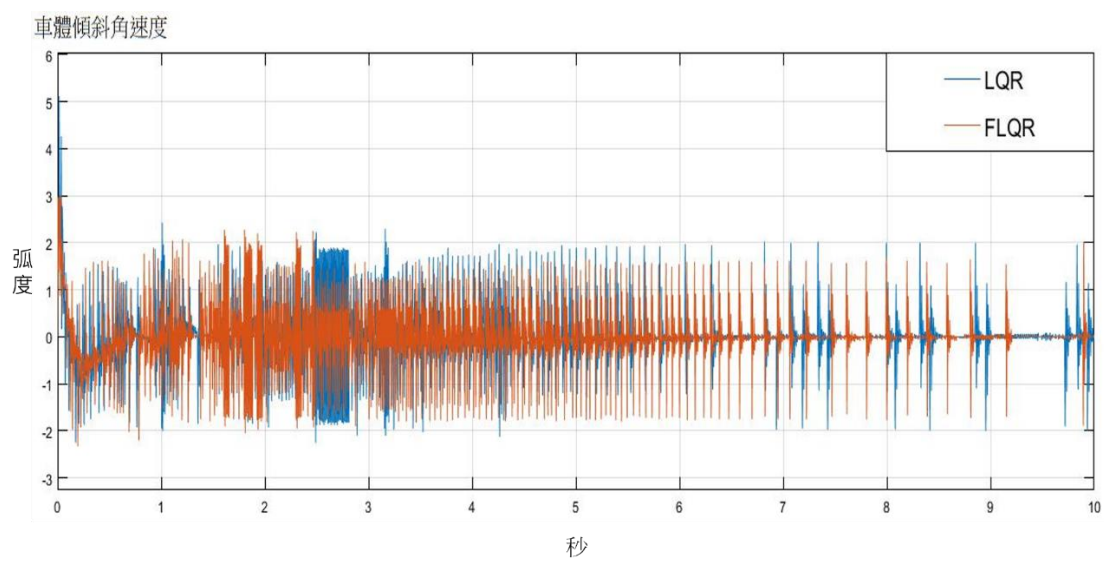


圖 4-3 車體傾斜角速度動態響應圖

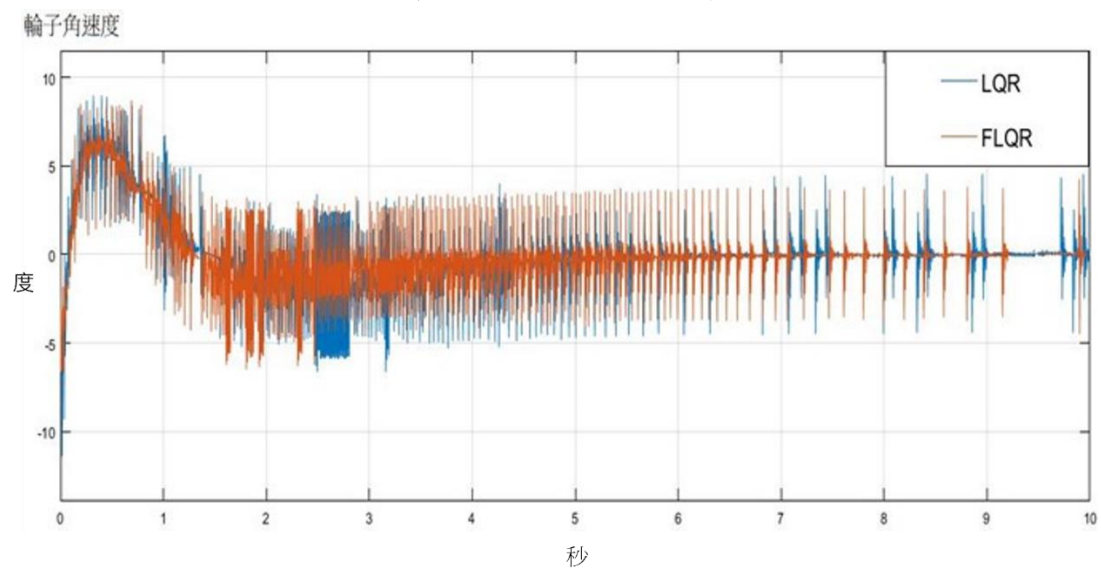


圖 4-4 速度動態響應圖

從圖 4-1、圖 4-2。可以得知兩種控制演算法都可以有效保持平衡，而 FLQR 的最大超越量、安定時間和最大位移量都較為優異，表 4-1 為 LQR 與 FLQR 的性能比較。

表 4-1 LQR 與 FLQR 性能比較

	LQR	FLQR
上升時間	49.326	25.121
安定時間	2.7	2
最大超越量	0.1576	0.1456
總誤差量	0.2349	0.2287

第二節 不同模糊規則數模擬

經過模擬可以得知 FLQR 的性能較為優異，使用的模糊規則數為 7 個，因此將模擬不同的模糊規則數，探討其中的差異性。

當初始傾斜角度設定為 0.37(弧度)，模糊規則數為 7 個時(7FLQR)，與原本 LQR 車體角度的模擬如下圖所示

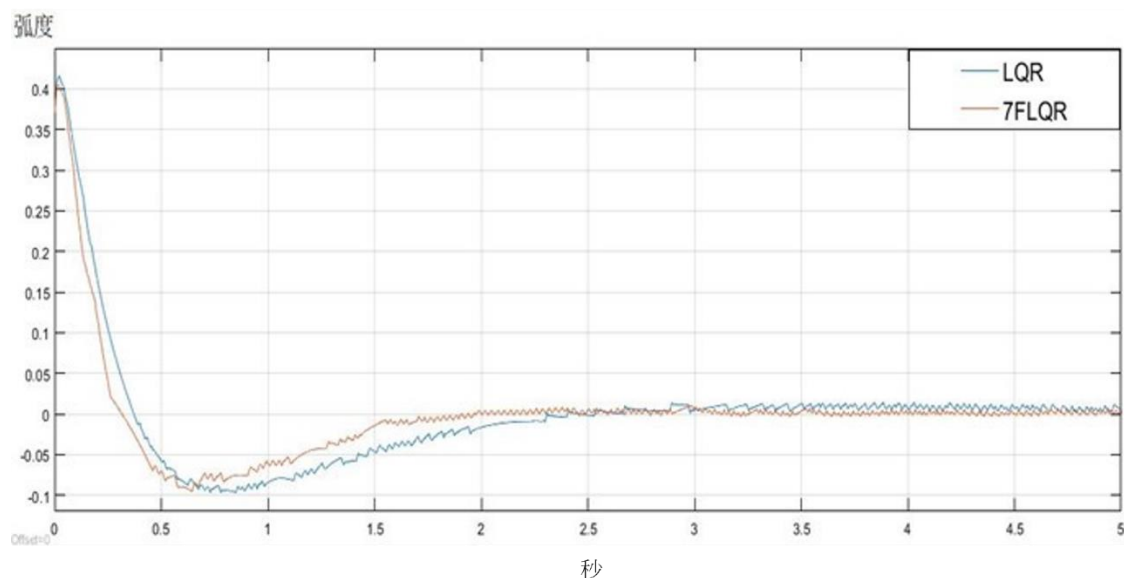


圖 4-5 初始車體角度 0.37 時 LQR 與 7FLQR 比較圖

表 4-2 LQR 與 7FLQR 性能比較

	LQR	7FLQR
最大超越量	0.4156	0.4039
總誤差量	0.5854	0.5233
安定時間	2.5	2

從模擬可以得知 7FLQR 較 LQR 的方法更快達到穩定平衡，且最大超越量與總誤差量也有著較佳的結果。

當模糊規則數為降為 5 個時(5FLQR)，與模糊規則數 7 個(7FLQR)的比較圖如下

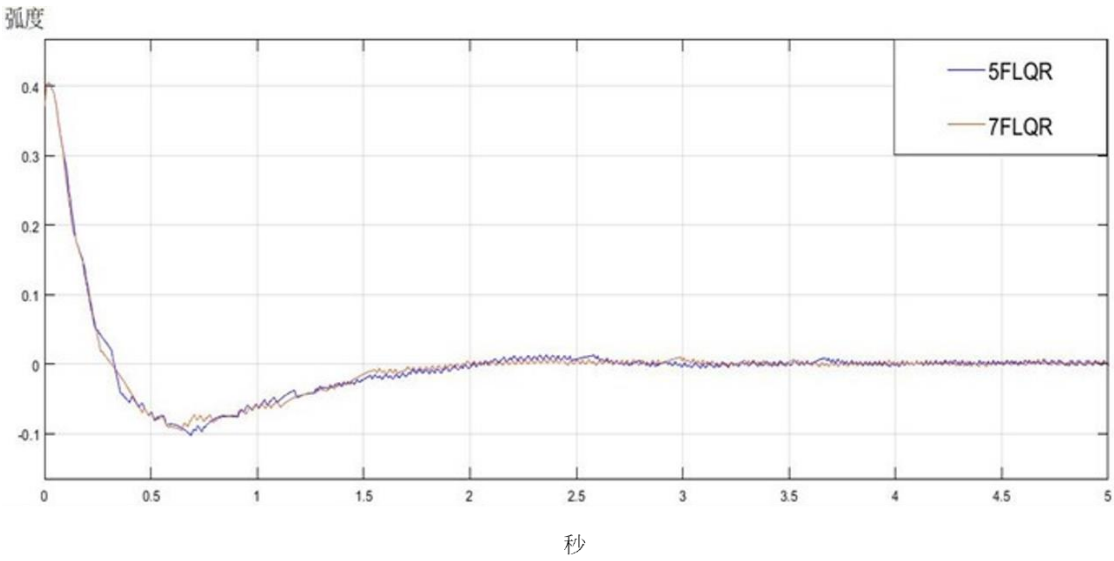


圖 4-6 初始車體角時 0.37 5FLQR 與 7FLQR 比較圖

表 4-3 5FLQR 與 7FLQR 性能比較

	5FLQR	7FLQR
最大超越量	0.4041	0.4039
總誤差量	0.5290	0.5233
安定時間	2	2

從模擬可以得知規則 5 與規則 7 都可以穩定平衡，且最大超越量、總誤差量差異不大，但規則 7 較規則 5 的性能好一點，而安定時間則是同樣在 2 秒處。

當模糊規則數為升為 9 個時(9FLQR)，與模糊規則數 7 個(7FLQR)的比較圖如下

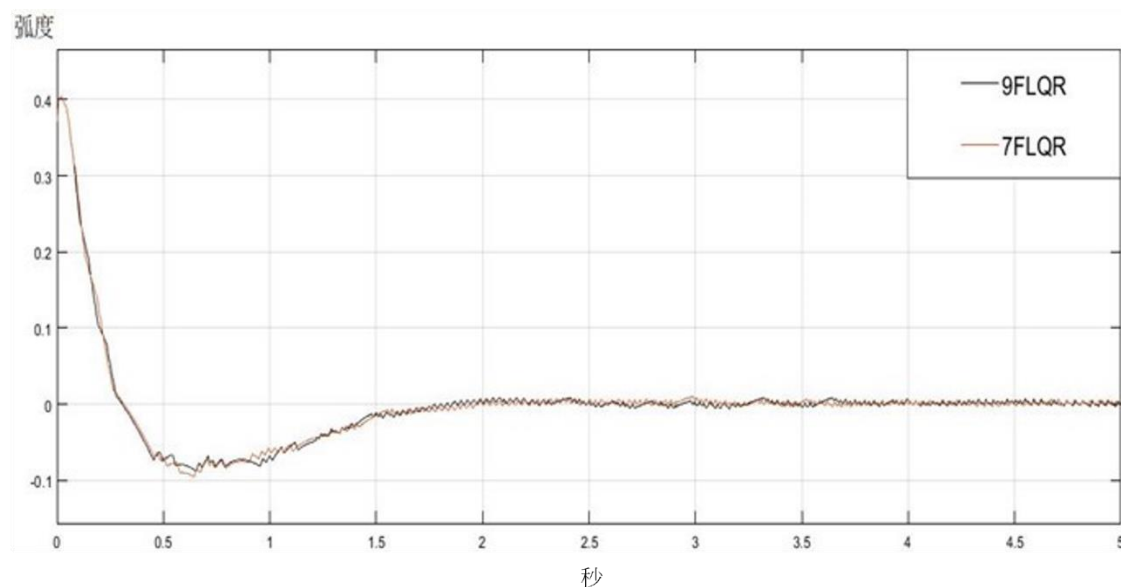


圖 4-7 初始車體角度 0.37 時 9FLQR 與 7FLQR 比較圖

表 4-4 9FLQR 與 7FLQR 性能比較

	9FLQR	7FLQR
最大超越量	0.4038	0.4039
總誤差量	0.5213	0.5233
安定時間	2	2

從模擬可以得知規則 9 與規則 7 都可以穩定平衡，且最大超越量、總誤差量差異不大，但規則 9 較規則 7 的性能好一點，而安定時間則是同樣在 2 秒處。

根據以上結果，可以知道規則數的不同，對最大超越量、總誤差量和安定時間的影響並不是很大，規則 5 改成規則 7 的效能有些微增加，規則 7 改成規則 9 的效能也有些微增加，但幾乎無差異，而若規則數越多，計算量也會增加，因此實機將採用 7 個模糊規則。

第三節 實機測試

將本文研究的控制理論在 Arduino 中編譯實現，並寫入 Arduino Leonardo 版以達成兩輪自體平衡車的實機平衡，並利用 Matlab Arduino Support package 去取得 Arduino Leonardo 版上連接線所傳送的角度數據，最後將數據繪製成圖示。

感測器的取樣速率約為 200 次/秒，x 軸為取樣次數，y 軸為車體傾斜角度，單位為角度，LQR 的實機平衡測試結果如下圖所示：

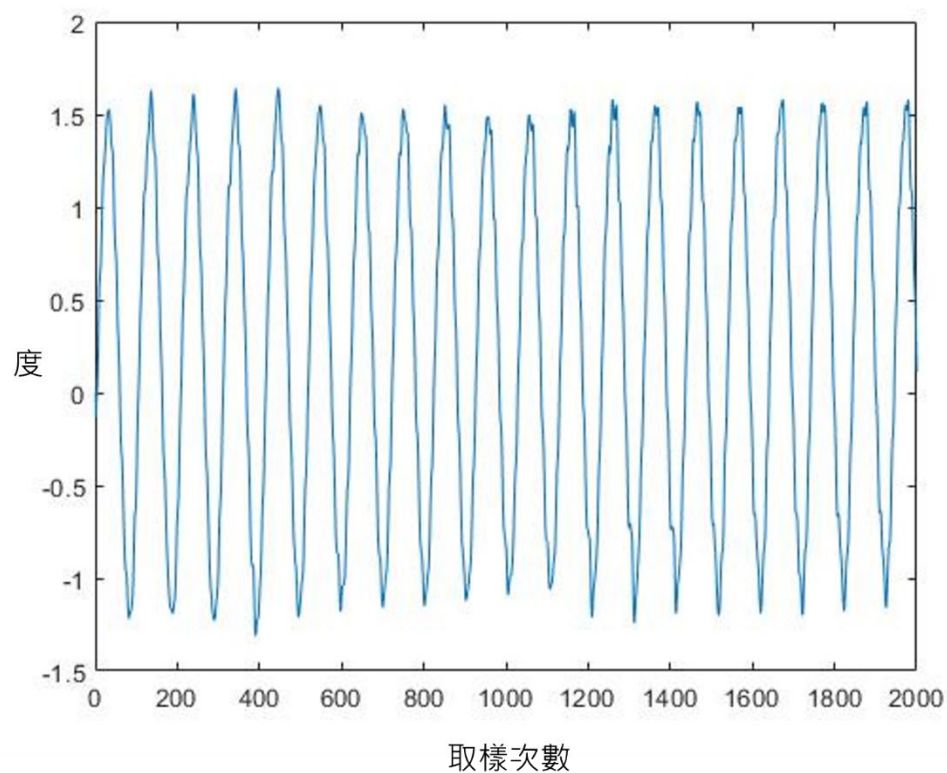


圖 4-8 LQR 實機平衡圖

經過測試，LQR 可以穩定平衡超過 10 分鐘以上，振幅約為 1.2~1.7 度之間。

FLQR 控制演算法部分，不同規則數的影響不大，因此本實驗使用 7 個規則數，7FLQR 實機平衡測試結果如下圖所示：

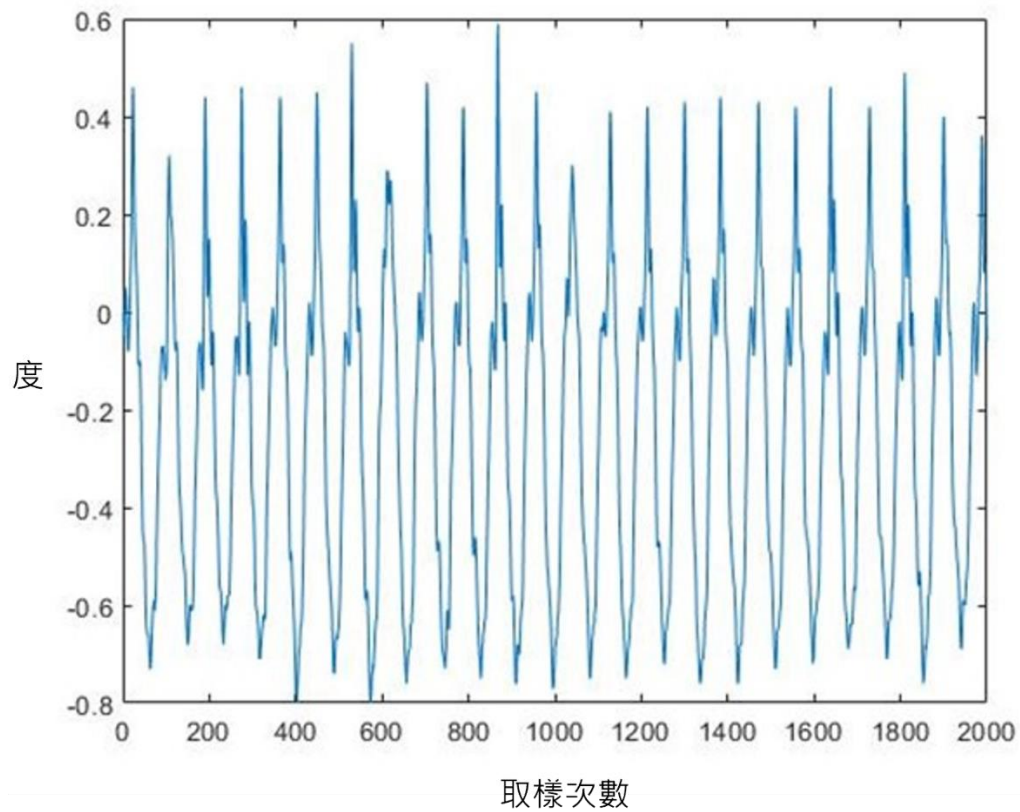


圖 4-9 7FLQR 實機平衡圖

經過測試，7FLQR 可以穩定平衡超過 10 分鐘以上，且振幅約為 0.3~0.8 之間，明顯比 LQR 的振幅來的小，因此可以得知 FLQR 平衡時的總誤差量性能較佳。

為了測試最大超越量的差異，使實機受到相同的外力 F ，因此編譯啟動後的 5 秒時(1000 取樣次數)會給與 LQR 與 FLQR 相同 50 脈衝的推力，LQR 所產生的動態響應圖如下

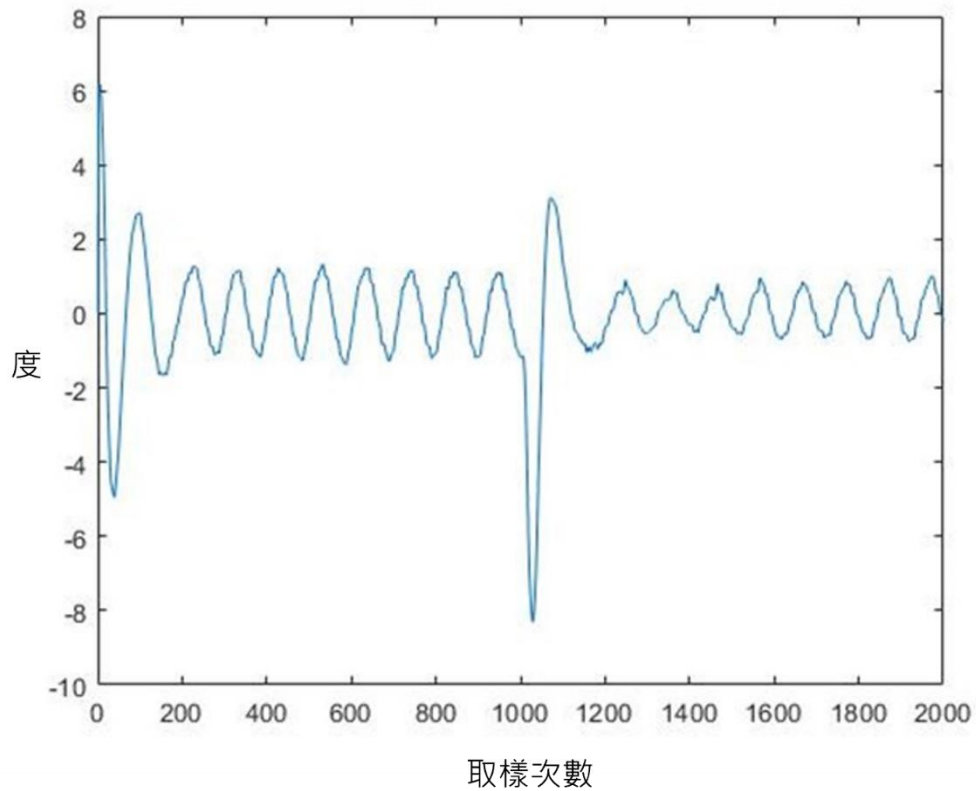


圖 4-10 LQR 於 5 秒時受到 50 脈衝的力

可以看到在取樣次數 1000 時發生劇烈震盪，最大超越量為 8.3。

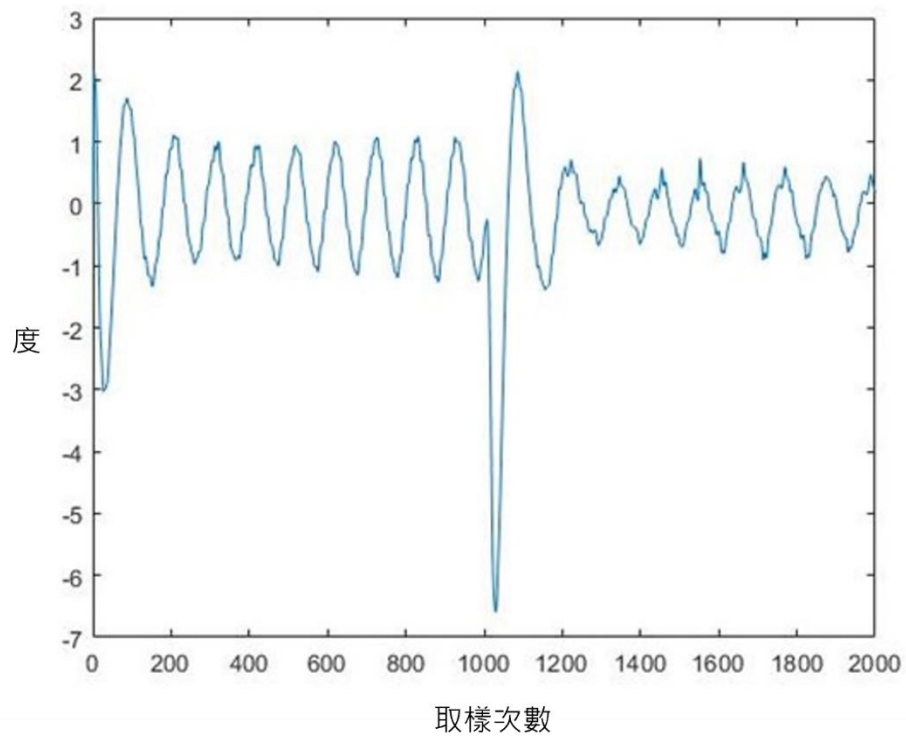


圖 4-11 7FLQR 於 5 秒時受到 50 脈衝的力

同樣在取樣次數 1000 時發生劇烈震盪，最大超越量為 6.6 明顯小於 LQR，因此可以得知 7FLQR 的最大超越量性能較佳。

第五章 結論與建議

本文探討了 FLQR 控制器演算法的設計，使兩輪自體平衡車的輪子角度、輪子角速度、車體角度和車體角速度因為不同的誤差輸入，而有一定程度上的變化輸出，用以補償原本 LQR 的增益值，使 Arduino 的兩輪自體平衡車保持直立穩定。最後在 Matlab Simulink 仿真模擬並在 Arduino 的實機完成穩定平衡，實驗結果顯示本文的 FLQR 控制方法在動態響應上的上升時間、安定時間、最大超越量以及總誤差量與 LQR 方法相比都有較佳的結果。

未來希望可嘗試使用其他演算法尋找 Q 、 R 矩陣的數值與增益值的上下界範圍，而在模糊控制的輸入也可以嘗試使用不同的輸入，例如輪子加速度或是車體加速度等。

參考文獻

一、中文參考文獻

- [1] 汪榮章(2017)，利用模糊推論於二輪自體平衡車系統之智慧型演算法設計，中華民國，臺北，臺北市立大學資訊科學系碩士班碩士論文。
- [2] 林哲英(2015)，利用模糊推論結合基因演算法應用於LEGO Mindstorms NXT兩輪自體平衡機器人，中華民國，臺北，臺北市立大學資訊科學系碩士班碩士論文。

二、英文參考文獻

- [3] Ha, Y.-S. (1996), *Trajectory tracking control for navigation of the inverse pendulum type self-contained mobile robot*. Robotics and autonomous systems, 17(1-2), pp. 65-80.
- [4] Jeon, T., Paek, I. (2021), *Design and Verification of the LQR Controller Based on Fuzzy Logic for Large Wind Turbine*. Energies, 14(1), pp. 230.
- [5] Grasser, F., D'arrigo, A., Colombi, S., Rufer, A. C. (2002), *JOE: a mobile, inverted pendulum*. IEEE Transactions on industrial electronics, 49(1), pp. 107-114.
- [6] Yamamoto, Y. (2008), *NXTway-gs (self-balancing two-wheeled robot) Controller Design*. The Math Works.
- [7] Jose, A., Augustine, C., Malola, S. M., Chacko, K. (2015), *Performance study of PID controller and LQR technique for inverted pendulum*. World Journal of Engineering and Technology, 3(02), pp. 76-81.
- [8] Oróstica, R., Duarte-Mermoud, M. A., Jáuregui, C. (2016), *Stabilization of inverted pendulum using LQR, PID and fractional order PID controllers: A simulated study*, IEEE International Conference on Automatica (ICA-ACCA), Curico, Chile, IEEE, pp. 1-7.
- [9] Sung, H.-C. (2015), *Balancing Robot Control and Implementation*, Texas A & M University.
- [10] Cao, J.-R., Huang, C.-P., Hung, J.-C. (2016), *Stabilizing controller design using fuzzy TS model on two wheeled self-balancing vehicle*, International Conference on Advanced Materials for Science and Engineering (ICAMSE), IEEE, pp. 520-523.
- [11] Kim, Y., Kim, S. H., Kwak, Y. K. (2005), *Dynamic analysis of a nonholonomic two-wheeled inverted pendulum robot*. Journal of Intelligent and Robotic Systems, 44(1), pp. 25-46.
- [12] Wongsathan, C., Sirima, C. (2009), *Application of GA to design LQR controller for an inverted pendulum system*, International Conference on Robotics and Biomimetics, IEEE, pp. 951-954.
- [13] Varghese, E. S., Vincent, A. K., Bagyaveereswaran, V. (2017), *Optimal control of*

- inverted pendulum system using PID controller, LQR and MPC, IOP Conference Series: Materials Science and Engineering*, IOP Publishing, pp. 052007.
- [14] Chang, Z., Chu, H., Shao, Y., Zhang, X. (2019), *A Synthetic LQR Control for The Planar Inverted Pendulum*, *Chinese Automation Congress (CAC)*, IEEE, pp. 3583-3587.
- [15] Zhao, J., Ruan, X. (2008), *The LQR control and design of dual-wheel upright self-balance robot*. *Intelligent Control and Automation*, pp. 4864-4869.
- [16] Fang, J. (2014), *The LQR controller design of two-wheeled self-balancing robot based on the particle swarm optimization algorithm*. *Mathematical Problems in Engineering*, 12, pp. 1-6.
- [17] Banerjee, R., Pal, A. (2018), *Stabilization of Inverted Pendulum on Cart Based on LQG Optimal Control*, *International Conference on Circuits and Systems in Digital Enterprise Technology (ICCSDET)*, Kottayam, India, IEEE, pp. 1-4.
- [18] Zadeh, L. A. (1965), *Fuzzy sets*. *Information and Control*, 8, pp. 338-353.
- [19] Zadeh, L. A. (1973), *Outline of a new approach to the analysis of complex systems and decision processes*. *IEEE Transactions on systems, Man, and Cybernetics*(1), pp. 28-44.
- [20] Mamdani, E. H., Assilian, S. (1975), *An experiment in linguistic synthesis with a fuzzy logic controller*. *International journal of man-machine studies*, 7(1), pp. 1-13.
- [21] Takagi, T., Sugeno, M. (1985), *Fuzzy identification of systems and its applications to modeling and control*. *IEEE transactions on systems, man, and cybernetics*(1), pp. 116-132.
- [22] Wu, J., Zhang, W. (2011), *Design of fuzzy logic controller for two-wheeled self-balancing robot*, *International Forum on Strategic Technology (IFOST)*, IEEE, pp. 1266-1270.
- [23] Al-Isawi, M. M., Attiya, A. J., ADOGHE, J. O. (2020), *UAV Control Based on Dual LQR and Fuzzy-PID Controller*. *Al-Khwarizmi Engineering Journal*, 16(3), pp. 43-53.
- [24] Wu, J., Zhang, W., Wang, S. (2012), *A two-wheeled self-balancing robot with the fuzzy PD control method*. *Mathematical Problems in Engineering*.
- [25] Bharali, J., Buragohain, M. (2016), *Design and performance analysis of fuzzy LQR; fuzzy PID and LQR controller for active suspension system using 3 degree of freedom quarter car model*, *International Conference on Power Electronics, Intelligent Control and Energy Systems (ICPEICES)*, Delhi, India, IEEE, pp. 1-6.
- [26] Hazem, Z. B., Fotuhi, M. J., Bingül, Z. (2020), *Development of a Fuzzy-LQR and Fuzzy-LQG stability control for a double link rotary inverted pendulum*. *Journal of the Franklin Institute*, 357(15), pp. 10529-10556.
- [27] Chantarachit, S. (2018), *Design and simulate of LQR-Fuzzy controller for unicycle robot with double flywheels*, *MATEC Web of Conferences*, EDP Sciences, pp. 02001.
- [28] Kudinov, Y., Pashchenko, F., Kelina, A., Vasutin, D., Duvanov, E., Pashchenko, A. (2019), *Analysis of Control System Models with Conventional LQR and Fuzzy LQR Controller*. *Procedia Computer Science*, 150, pp. 737-742.

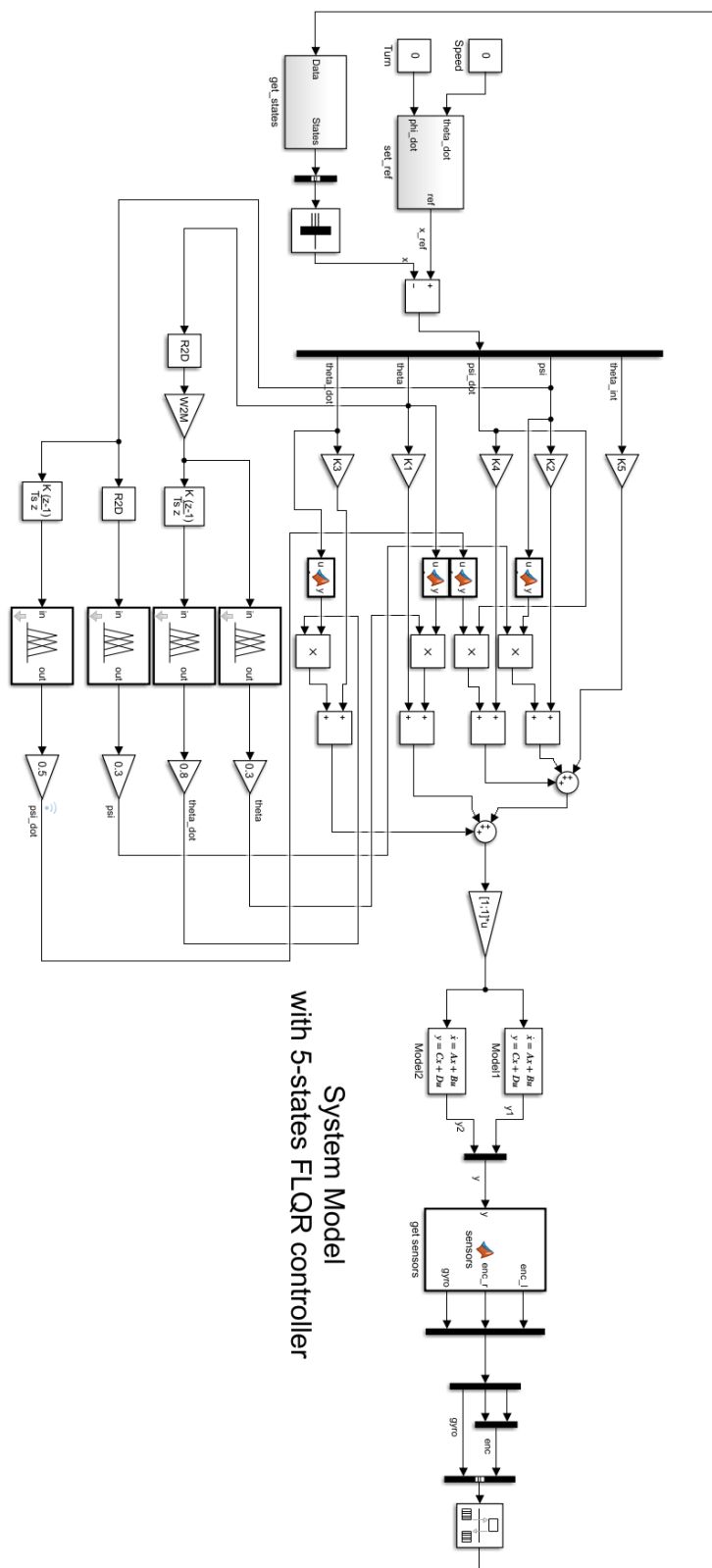
- [29] Devi, P. S., Santhi, R. V. (2019), *Introducing LQR-fuzzy for a dynamic multi area LFC-DR model*. International Journal of Electrical & Computer Engineering (2088-8708), 9(2), pp. 861-874.
- [30] Adamu, J. K., Hamza, M. F., Isa, A. I. (2020), *Performance Comparisons Of Hybrid Fuzzy-LQR And Hybrid PID-LQR Controllers On Stabilizing Double Rotary Inverted Pendulum*. Journal of Applied Materials and Technology, 1(2), pp. 71-80.

三、 網頁參考文獻

- [31] Segway。線上檢索日期: 2020 年 12 月 10 日。網址: <http://www.segway.com/>
- [32] 慧淨電子。線上檢索日期: 2020 年 12 月 10 日。網址: <http://www.hjmccu.com/>
- [33] MotionKing。線上檢索日期: 2020 年 12 月 10 日。網址: http://www.svaltera.ua/catalogs/knowledge-base/brands/motionking/HB_Stepper_Motor_E.pdf

附錄

使用 Matlab Simulink 建置的兩輪自體平衡車模型

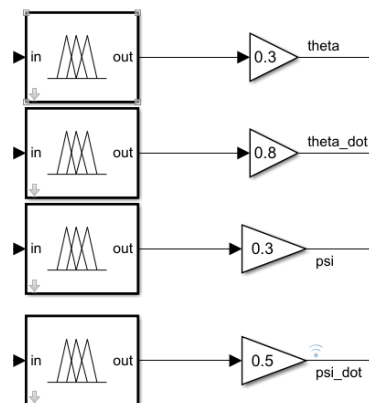


System Model
with 5-states FLQR controller

將感測器數值轉成 5 個輸入值



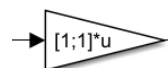
4 個輸入的模糊規則與常數加權



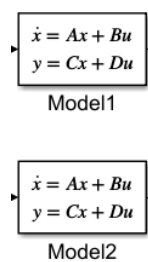
5 個輸入的 LQR 增益值



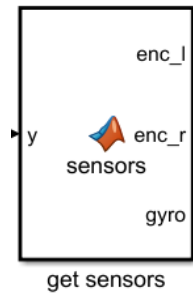
將輸出轉換成左右馬達輸出



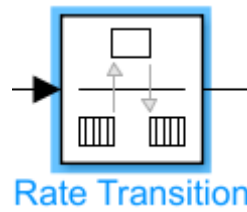
兩輪平衡車的狀態方程式



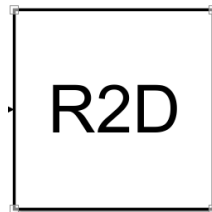
取得感測器數值



實機操作時，感測器讀取為離散值而非連續數值，因此轉成離散



將弧度轉換成度



將度轉換成位移(公尺)

