

## 摘要

本研究透過強化學習，開發四軸無人機之高度控制系統，為其載具提供在未知環境中執行任務時的高適應性和卓越的控制能力。強化學習可以解決高複雜性的控制的問題，經由接收環境反饋並持續學習，逐漸優化控制系統以達到目標。四軸無人機具有操作簡單、成本低廉、機動性高和穩定等特性，且具有廣泛的應用範圍，可以取代許多高風險的工作。目前，大多數四軸無人機使用比例積分微分控制器，但調整其參數需要累積的經驗且耗時。模糊邏輯控制器更符合人類邏輯，可以處理不確定的控制問題，但設計規則庫也需要大量的時間成本，且在面對複雜環境時可能會有侷限性。以上兩種控制器無法在不同環境下實現最佳性能，例如高度差異較大或有外部干擾等情況。本研究主要著重於四軸無人機的高度控制器，使用近端策略最佳化算法作為四軸無人機高度控制器，並在不同高度及外在干擾下，使用 Ziegler-Nichols 法則調整之比例積分微分控制器和模糊邏輯控制器進行比較及分析，以選擇最具適應不同環境之控制系統。

**關鍵字：**四軸無人機、具適應性能之控制系統、近端策略最佳化、強化學習、無人機暫態及穩態響應分析、隨機干擾環境

## Abstract

This study presents the development of a robust altitude control system for quadcopters using reinforcement learning. The system aims to enhance adaptability and precise control capabilities, particularly in unfamiliar environments. Reinforcement learning is leveraged to address the complexities of control, enabling continuous learning and adaptation based on different environmental feedback. Quadcopters are valued for their simplicity, cost-effectiveness, maneuverability, and stability, making them versatile tools across numerous applications. However, the popular use of Proportional-Integral-Derivative (PID) controller in these drones presents limitations, requiring time-consuming parameter adjustments based on empirical experience. Fuzzy Logic Controllers (FLCs) align more closely with human logic and handle uncertain control problems but necessitate substantial time investments in rule-base design and may have constraints in complex environments. Neither PID controllers nor FLCs can consistently deliver optimal performance in varying environments, such as those with significant altitude variations or flight environment with random disturbances. This research focuses on optimizing quadcopter altitude control using the Proximal Policy Optimization (PPO) algorithm. It conducts a comparative analysis between PPO, Ziegler-Nichols-tuned PID controller, and FLC under various altitude conditions and flight environment with random disturbances.

**Keywords :** Quadcopter, PID Controller, Fuzzy Logic Controller, Reinforcement Learning Controller, Proximal Policy Optimization Algorithm, Flight Environment with Random Disturbance

## 誌謝

本專題研究中，我們深感榮幸能夠受到恩師劉俊宏博士的悉心指導與支持。在過去的三年中，給予我們許多的指導與幫助，無論是在論文投稿方面還是專業領域的探討，恩師的建議都讓我們受益良多。在假日時也會抽空與我們討論。您的悉心指導讓我們了解了不同的控制系統，也讓我們更有方向的去解決遇到的問題，在研究的過程中更有信心。在參與研究的這段時間中，恩師的付出無疑是我們成長過程中最寶貴的財富，在此向劉老師至以誠摯的謝意與最高的敬意。同時，我們也要感謝在大三下學期指導我們的恩師朱力民博士，不僅點出了我們研究報告的不足之處，還給予了寶貴的建議，並在我們參與研討會報名等方面提供了很大的協助。朱老師的悉心指導讓我們深受啟發，我們由衷地感謝朱老師的支持與鼓勵。也感謝學長們的引導與協助，從專題的初期設計到最後的呈現，都毫不保留地與我們分享經驗，在遇到困難與瓶頸時，總是與我們討論，找到解決問題的方法，每一次的討論交流，都讓我們獲得了新的視角和思考方式，幫助我們成長，我們衷心的感謝學長們的付出與支持。最後非常感謝同組的夥伴們這一年來的互合作，在過去的一年中，我們彼此合作，在困難與逆境中相互照應，共同克服了許多困難，相互扶持，不斷成長。也特別感謝我們的家人在我們的求學過程中給予了無限的支持與鼓勵，讓我們能夠專注於學業，堅持到專題研究的完成。最後感謝，幫助過我們的所有人，我們將帶著獲得的專業知識和祝福，踏入未來的挑戰，繼續努力前行。

王昱健、葉舜斌 謹誌於  
國立臺東大學  
綠色與資訊科技學士學位學程  
中華民國一一二年六

# 壹、前言

## 一、研究源起

近年來，無人機在眾多領域中嶄露頭角，全球的無人機市場呈現快速成長的趨勢，越來越多的企業紛紛投資無人機市場，無人機的市場規模預估在 2025 年增長到 450 億美元[1]，不論是消費級市場的空拍機、穿越機，或是工業中常見的電塔檢修、監控量測、農業中的噴灑農藥與生長監控、物流中的貨物配送、建築物的安全檢測與 3D 建模以及消防救災中的環境監控與搜救等，透過四軸無人機進入人類難以抵達或是高度風險、高重複性之任務，來取代人力執行的風險，已經是未來趨勢。其廣泛的應用範圍也預示著無人機在未來的商業發展中巨大的潛力。四軸無人機具有高度穩定性、靈活性及結構簡單等特點。然而，要讓四軸無人機能夠在多種不同的複雜環境中發揮作用，建立穩定的控制系統就顯得相當重要，控制系統不但是四軸無人機最基本的環節，也是重要的核心技術之一。不同的控制器擁有不同的特性，如反應速度較佳可能控制精準度就較低，反之亦然。為了使四軸無人機可以在多種複雜環境下使用，高適應性的精確控制系統就顯得相當重要。

現今市面上的四軸無人機通常使用比例-積分-微分 (Proportional-Integral-Derivative, PID) 算法作為姿態控制器。PID 控制器是一種經典的控制算法，簡單且實現成本低，在多數情況下，PID 控制器可以滿足四軸無人機控制的需求，然而 PID 算法對環境變化並不敏銳，因此無法適應各類環境，需根據不同環境調整參數，但一開始其參數難以尋找。常見找尋 PID 參數的方式有，利用 Ziegler-Nichols 法則[2]，粒子群最佳化 (Particle Swarm Optimization, PSO)[3] 等方法，都可用來搜尋或調整較適當之 PID 參數於控制四軸無人機，但以上調整參數之方式都需經由經驗法則及適當調整演算法來獲得最佳化後之 PID 參數，若環境突然改變，需要持續花費大量的時間搜尋最佳化之參數。模糊邏輯控制器 (Fuzzy Logic Control, FLC)[4] 也是為常見的四軸無人機控制器之一，具有處理環境不確定因素的能力，透過將模糊的輸入轉換成清晰的輸出，以制定適當的控制策略。然而模糊邏輯控制器的性能往往受到模糊規則設計和參數調整的影響。模糊規則的制定需要基於專家知識，而參數的調整可能需要多次實驗和調整，在面對不同飛行環境時需要重新調整參數以確保穩定的飛行，整個過程耗時且複雜。

強化學習已經成為引人關注的熱門領域，在多個不同領域中都取得了顯著的進展，其中包括自然語言處理、遊戲控制、自駕車、機器人控制、金融等多個領域。最著名的強化學習應用就是第一個打敗人類圍棋世界冠軍的電腦 AlphaGo[5]，它的發表不僅讓眾人認知到強化學習強大的能力，也讓人意識到強化學習在解決高度複雜問題的潛力。強化學習具有適應性和高效性的特點，在訓練的過程中不需透過人工進行參數調整。強化學習能夠根據環境的變化自動學習調整控制策略，使其在應對各種複雜且具有不確定性的環境時，依然能夠確保控制的穩定性，且目前國內的文獻較少關於四軸無人機強化學習控制系統之研究。因此，本研究使用強化學習應用於高度控制的領域中，以實現更優越的飛行性能。

## 二、研究目的

強化學習近年來的發展與技術快速進步，促使各類領域將強化學習用來解決

各種問題，這一進展也為四軸無人機的控制問題提供了不同的解決途徑。當前常見的傳統 PID 控制方法[6]，在飛行前需要事先對控制器進行參數設定，在面對複雜的環境時，往往會受到外部干擾導致飛行誤差，難以獲得最佳的控制效果；而模糊邏輯控制器[4]雖然能夠透過環境中的一些不確定因素制定策略，但是模糊規則與參數需要事先設計，在遇到未知的或複雜的環境時，可能會存在一定的局限性，無法達到最佳的控制效果。強化學習則是透過四軸無人機與環境中進行互動學習交換情報，藉此獲取最佳的控制策略，從而對高度控制的效果進行優化，針對不確定且有干擾的環境中，強化學習具備更優秀的適應能力。因此本研究著重於探討使用強化學習中的近端策略最佳化算法 (Proximal Policy Optimization, PPO)[7]做為四軸無人機高度控制器，並在無干擾環境中使用 PID 控制器、模糊邏輯控制器與本研究 PPO 控制器，比較三種控制器在不同目標高度中的暫態、穩態響應等性能表現；再以高適應能力為主要目標，在飛行環境中加入隨機陣風干擾，針對連續不同目標高度且進行訓練，使用 PID 控制器作為對照組，與本研究 PPO 控制器在不同連續高度中飛行訓練，觀察兩控制器之暫態、穩態響應與性能指標，經由比較分析選出最優之四軸無人機高度控制器。

## 貳、文獻探討

### 一、四軸無人機之動力學

四軸無人機本身為不容易控制且不穩定系統，其具有三個平移自由度和三個旋轉自由度。其中兩個螺旋槳以順時針方向旋轉，而另外兩個以逆時針方向旋轉，如圖 1 所示。其動力學通常透過兩個座標係來描述：地球座標系(E)和機身座標系(B)，如圖 1。假設機身座標系的原點位於四軸無人機的重心，慣性測量單元(IMU)所在的位置。由於四軸無人機框架在運動過程中不斷加速及旋轉，因此機身座標系是非慣性的。假設地球固定參考系表現為慣性。指定四軸飛行器的地理位置和重力方向需要考慮地球坐標系。機身固定參考系承載旋翼推力和四軸無人機的角速度。

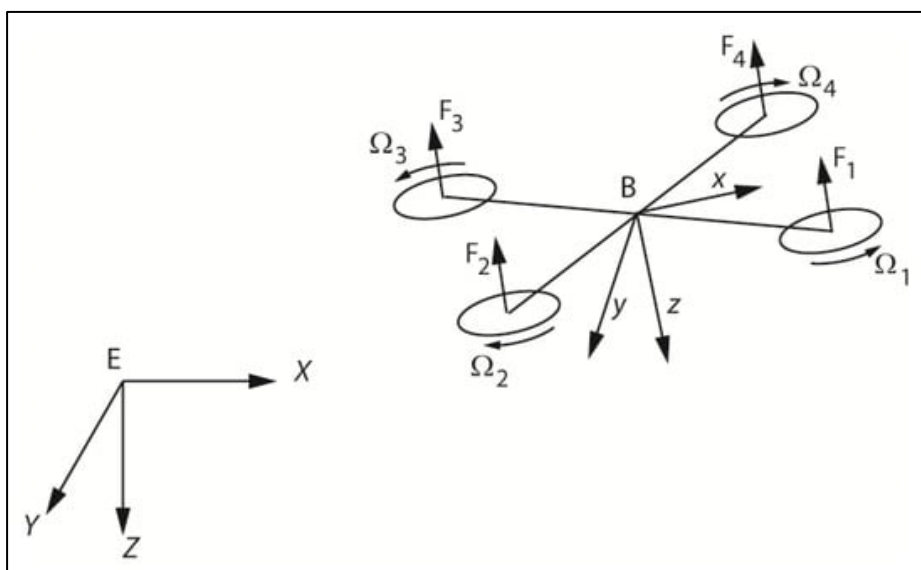


圖 1、四軸無人機的動力學。E(X,Y,Z):地球座標系；B(x,y,z):機身座標系； $\Omega_i$ :旋翼轉速； $F_i$ :旋翼升力[8]

四軸無人機的飛行運動方式可以分解為平移運動和旋轉運動。在空間中的方向由滾轉、俯仰和偏航轉動表示；當無人機在空中並且其機體輕微傾斜時，會產生沿 X 和 Y 軸的平移運動，分別以 x 和 y 的米為單位測量。為了便於穩定控制器的設計，通常根據機體配置選擇一個作動器解耦矩陣[9]。對於結構為'X'型四軸無人機中的解耦矩陣如式 1，它將四個平方速度參考值  $\Omega_1^2$ 、 $\Omega_2^2$ 、 $\Omega_3^2$  及  $\Omega_4^2$  映射到升力和穩定力矩[9]。

$$\begin{pmatrix} F_Z \\ L \\ M \\ N \end{pmatrix} = \begin{pmatrix} -k_F & -k_F & -k_F & -k_F \\ -\frac{1}{\sqrt{2}}lk_F & -\frac{1}{\sqrt{2}}lk_F & \frac{1}{\sqrt{2}}lk_F & \frac{1}{\sqrt{2}}lk_F \\ \frac{1}{\sqrt{2}}lk_F & -\frac{1}{\sqrt{2}}lk_F & -\frac{1}{\sqrt{2}}lk_F & \frac{1}{\sqrt{2}}lk_F \\ k_M & -k_M & k_M & -k_M \end{pmatrix} \begin{pmatrix} \Omega_1^2 \\ \Omega_2^2 \\ \Omega_3^2 \\ \Omega_4^2 \end{pmatrix} \quad \text{式 1}$$

其中  $F_Z$  是沿著機體 z 維度的升力，L 是滾轉力矩，M 是俯仰力矩，而 N 是偏航力矩。控制系統透過解耦矩陣計算速度參考值的平方，該矩陣是從式 1 的映射的反向，這樣解耦的操作變數  $u_{1,d}$ 、 $u_{2,d}$ 、 $u_{3,d}$  及  $u_{4,d}$  就對應到升力和穩定力矩。式 2 中  $u_1 \approx F_Z$  負責沿 z 軸的上升或下降， $u_{2,d} \approx L$ 、 $u_{3,d} \approx M$ ，以及  $u_{4,d} \approx N$ ，則分別負責滾轉、俯仰和偏航[9]。藉由式 1 是線性的特性，控制系統可以通過設置  $u_i = \Omega_i$ ， $i=1 \cdots 4$ ，直接驅動馬達參考值。解耦操作變數  $u_{d,i}$  和直接操作變數  $u_i$  之間關係反比於式 1，並變為式 2。

$$\begin{pmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \end{pmatrix} = \begin{pmatrix} -\frac{1}{4k_F} & -\frac{1}{2\sqrt{2}lk_F} & \frac{1}{2\sqrt{2}lk_F} & \frac{1}{4k_M} \\ -\frac{1}{4k_F} & -\frac{1}{2\sqrt{2}lk_F} & -\frac{1}{2\sqrt{2}lk_F} & -\frac{1}{4k_M} \\ -\frac{1}{4k_F} & \frac{1}{2\sqrt{2}lk_F} & -\frac{1}{2\sqrt{2}lk_F} & \frac{1}{4k_M} \\ -\frac{1}{4k_F} & \frac{1}{2\sqrt{2}lk_F} & \frac{1}{2\sqrt{2}lk_F} & -\frac{1}{4k_M} \end{pmatrix} \begin{pmatrix} u_{1,d} \\ u_{2,d} \\ u_{3,d} \\ u_{4,d} \end{pmatrix} \quad \text{式 2}$$

牛頓-歐拉方程 (Newton-Euler equation) 可以描述四軸無人機的動力學，該方程處理了四軸無人機上的力和力矩。式 3 描述了載具的平移運動[8]，而式 4 描述其旋轉運動[8]。

$$m \begin{pmatrix} \ddot{x} \\ \ddot{y} \\ \ddot{z} \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ mg \end{pmatrix} + R_{EB} \begin{pmatrix} 0 \\ 0 \\ F_Z \end{pmatrix} \quad \text{式 3}$$

$$\begin{bmatrix} L \\ M \\ N \end{bmatrix} = I\dot{\omega} + \omega \times I\omega \quad \text{式 4}$$

以下為四軸無人機完整的數學模型，如下所示[8, 9]:

$$\ddot{x} = \frac{1}{m} \left[ F_z (c_\phi c_\varphi s_\theta + s_\phi s_\varphi) \right] \quad \text{式 5}$$

$$\ddot{y} = \frac{1}{m} \left[ F_z (s_\theta c_\phi s_\varphi - s_\phi c_\varphi) \right] \quad \text{式 6}$$

$$\ddot{z} = g + \frac{F_z}{m} (c_\theta c_\phi) \quad \text{式 7}$$

$$\dot{\phi} = p + r \cdot c_\phi t_\theta + q \cdot s_\phi t_\theta \quad \text{式 8}$$

$$\dot{\theta} = q \cdot c_\phi - r \cdot s_\phi \quad \text{式 9}$$

$$\dot{\psi} = r \cdot \frac{c_\phi}{t_\theta} + q \cdot \frac{s_\phi}{c_\theta} \quad \text{式 10}$$

$$\dot{p} = \frac{1}{I_x} (L + I_y \cdot qr - I_z \cdot qr) \quad \text{式 11}$$

$$\dot{q} = \frac{1}{I_y} (M - I_x \cdot pr + I_z \cdot pr) \quad \text{式 12}$$

$$\dot{r} = \frac{1}{I_z} (N - I_x \cdot pq + I_y \cdot pq) \quad \text{式 13}$$

最後，藉由狀態向量  $x_s = (x, y, z, \dot{x}, \dot{y}, \dot{z}, \phi, \theta, \psi, p, q, r)^T$ ，其中  $(x, y, z)$  為慣性座標系中位置狀態， $(\dot{x}, \dot{y}, \dot{z})$  為速度狀態， $(\phi, \theta, \psi)$  為姿態狀態，分別為橫滾角 (roll)、俯仰角 (pitch) 及偏航角 (yaw)，及  $(p, q, r)$  姿態之角速度狀態，再配合四個控制輸入  $(F_z, L, M, N)^T$  描述四軸無人機的完整之動態模型。式 5 至式 13 為非線性系統，存在三角函數和狀態變數之間的乘法，但其函數有平滑之特性，使其近似於線性系統。

## 二、比例-積分-微分控制

PID 控制器是四軸無人機最基礎且常見的控制系統[6]，因其簡單且容易快速調整的特性應用在許多不同的領域。PID 控制器(圖 2)包括了三個不同的參數：

比例控制項  $K_p$ 、積分控制項  $K_i$  與微分控制項  $K_d$ ，比例控制項主要與控制誤差與無人機穩態誤差有關，積分控制項主要與累積誤差有關，微分控制項主要與誤差變化率與預測未來誤差有關。

- (一) 比例控制項(P):比例控制項利用當前的狀態值與目標值互相比較，將兩者之間的誤差值乘於比例增益常數  $K_p$ ，依比例的方式對誤差進行修正，使得控制器的輸出能夠隨著誤差變化而相應調整。如果  $K_p$  數值設定過大，可能會造成系統的不穩定導致系統震盪無法收斂，反之若  $K_p$  數值設定過小，系統的反應變慢，將無法迅速修正干擾所造成的誤差。
- (二) 積分控制項(I):積分控制項透過過去的誤差值，將過去累積的誤差值乘上一個積分增益常數  $K_i$ ，以消除系統的穩態誤差，加快系統的收斂速度。但若  $K_i$  數值設定過大，可能會導致系統出現過衝或震盪的情況。
- (三) 微分控制項(D):微分控制項計算誤差的變化率，即誤差的微分乘上微分增益常數  $K_d$ ，用於預測系統未來誤差的變化趨勢，避免系統因誤差變化過快而產生過度響應，提高系統的收斂速度與穩定性。為了避免受到高頻雜訊的影響，通常在微分控制項還會加上一個低通濾波器，以平滑控制信號，避免不必要的振盪。

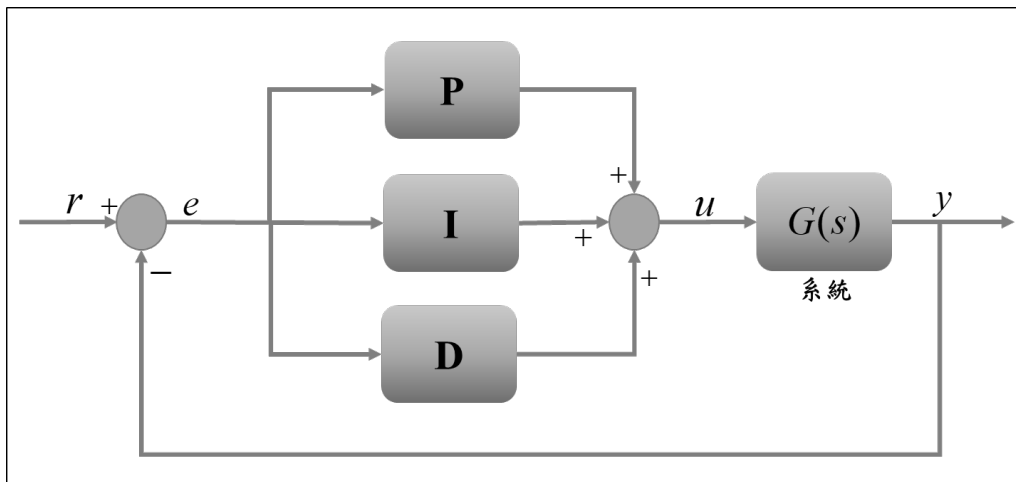


圖 2、PID 控制系統流程圖[2]

$$u(t) = K_p e(t) + K_i \int_0^t e(\tau) d\tau + K_d \frac{d}{dt} e(t) \quad \text{式 14}$$

PID 控制(式 14)在大多數情況下的表現足夠優異，但  $K_p$ 、 $K_i$  及  $K_d$  參數皆為定值，無法實時根據不同的飛行環境進行即時的改動。因此，當系統在環境變化與外部干擾較多的環境時，PID 控制系統可能較難達到穩定控制。本研究採用 ZN 法則 (Ziegler–Nichols method) 找尋一組 PID 參考值，再依據系統飛行狀態進行調整以獲得最佳值。

### 三、 模糊邏輯控制理論

本研究之第二種控制為模糊邏輯控制器[4]，其控制器較 PID 控制器需要花費更多時間在設計上，但此控制系統是基於模糊邏輯所設計出來的[10]，可以處理模糊、不確定和非線性系統，且較符合人類的邏輯且直觀，模糊控制系統包含



量化(模糊化)、控制器(專家系統)、解模糊三個主要的部份，透過量化將感測器獲取到的數據轉換為模糊邏輯控制器所能讀取之資料；而控制器也稱為專家系統，其中包含了資料庫、規則庫和推理引擎，量化後的數據經過專家系統進行判斷並權重調整，再將結果輸出並解模糊化，使訊號轉為機器可以判別的值。以下是幾個比較重要的名詞解釋：

- (一) **模糊化 (Fuzzification)**:這是將感測器資料轉換為模糊集的過程。模糊集使用隸屬函數來表示元素的隸屬度。在這裡，你可能會使三角形、梯形等隸屬函數，如，來將感測器資料轉化為隸屬度值。
- (二) **規則庫 (Rule Base)**:規則庫是模糊控制器中的重要組成部分，其中包含一系列的模糊規則，每個模糊規則將輸入的模糊集映射到輸出的模糊集，這些規則可以通過人類專家知識來制定。例如"如果誤差大而誤差變化小，則增加推力"。
- (三) **推理引擎 (Inference Engine)**:推理引擎會根據輸入的模糊規則來進行推理，並產生模糊輸出。常見的推理方法為最小最大法 (Max-Min)或最大平均法 (Max-Avg)，以合併不同規則的輸出。
- (四) **解模糊化 (Defuzzification)**:解模糊化是將模糊輸出轉換為實際控制信號的過程。常見的方法有：重心法 (Centroid Method)、最大值法 (Max Method)等，以獲得一個清晰的輸出值。而本研究使用重心法[10]，如式 15。

$$\text{COG} = \frac{\int_a^b x \cdot \mu(x) dx}{\int_a^b \mu(x) dx} \quad \text{式 15}$$

若需要反應優異的模糊邏輯控制器，則必須以詳細的定義輸入和輸出，並建立完整的資料庫和語言規則庫[11]。本研究的規則庫不僅參考高度的誤差( $e_z$ )計算，也使用了高度的誤差微分( $d_z$ )來做為混合運算，如此一來模糊邏輯控制器可以更精確的控制無人機，如目前無人機高度低於目標值且瞬時速度是往上，此時若有兩種輸入參考，控制系統就會給予合適的向上推力，避免過衝情況產生。

#### 四、強化學習

強化學習[12]是機器學習中很重要的一個算法，在近年來已經廣泛應用在控制系統、遊戲、機器人、醫療等眾多領域中，在控制系統方面，強化學習被用於設計智能控制器，讓系統能夠在不確定的環境中自主調整以實現更佳的適應能力。強化學習的工作原理如圖 3 所示，主要有以下五個重點。其核心理念為在不同環境中，通過觀察當前狀態，代理持續地嘗試並評估不同的動作，同時接收來自環境的獎勵，以學習如何最大化累積的回報，不斷迭代優化其行動策略，以實現更優的性能。馬可夫決策過程 (Markov Decision Process, MDP)[13]是強化學習領域中的一個核心數學框架，被廣泛運用於解決涉及隨機性和可控制性的序列決策問題。在 MDP 中，將代理與環境的互動過程進行建模，代理根據其採用的策略，在不同的狀態下做出決策，而環境則根據代理的動作，提供新的狀態以及即時的獎勵回饋，依據當前狀態和獎勵更新動作策略以實現策略學習。MDP 主要由狀態集合 (S, State set)、動作集合 (A, Action set)、狀態轉移函數 (P, State transition function)、獎勵函數 (R, Reward function)和折扣因子 ( $\gamma$ , Discount factor)組成。狀態集合涵蓋了代理在任務中可能出現的所有狀態；動作集合則包含了代

理可以選擇執行的所有動作；狀態轉移函數代表代理在特定狀態中執行一個動作選擇，下一個新狀態發生的機率分佈，當代理在狀態  $s$  執行動作  $a$  後，下一個狀態為  $s'$  的機率，表示為  $P(s'|s,a)$ ；獎勵函數則是指代理在執行某個動作並轉移到新狀態  $s'$  時，獲得的即時獎勵，表示為  $R(s,a)$ ；折扣因子  $\gamma$  的值介於 0 和 1 之間，用於對未來的獎勵進行折扣，適當的平衡即時獎勵與累積獎勵。強化學習與傳統監督式學習的區別在於，強化學習不依賴於人為提供正確的輸入/輸出，也不需要精確的校正來優化，強化學習更注重於在實時環境中進行規劃學習最佳策略[14]，這使其能夠解決傳統線性控制方法難以處理的高維度複雜環境。同時，強化學習在處理非線性系統與動態變化等方面表現出色，能夠自動調整控制策略以保證持久性的控制效果。在強化學習領域中，包含非常多種常見的演算法，例如最基本的策略梯度 (Policy Gradient) 和 Q-learning 演算法，以及結合深度學習的 Deep Q Network (DQN) 等等。其中 PPO 演算法通過近端方法對策略進行優化，確保策略更新幅度適中，克服過度探索和收斂速度緩慢問題，且適用於連續與離散動作空間，具有高效且穩定等優點。因此，本研究選擇了 PPO 演算法作為主要研究對象，探討其作為四軸無人機控制器中的效能表現。

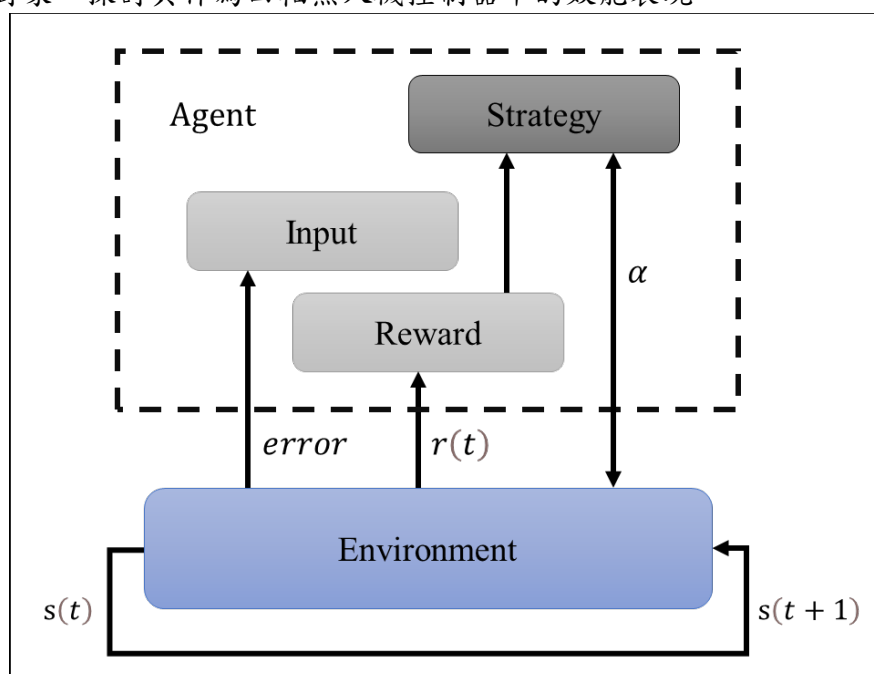


圖 3、強化學習架構圖[12]

## 五、近端策略最佳化

PPO(式 16)是強化學習演算法中較新且效率較佳的演算法[7]，主要針對策略梯度 (Policy Gradient) 方法進行改進[15]，在傳統策略梯度算法中，存在過度的策略探索和收斂效率不佳等問題，PPO 算法利用重要性採樣的方法進行學習，重要性採樣是一種用於衡量不同概率分佈之間差異的技術。在 PPO 算法中，重要性採樣的目的是估計在新策略下採取特定動作的機率，相對於舊策略下的機率，這個變化有多大。在策略更新的過程中，因為新的策略可能會引入動作概率的變化，這可能會對估計的獎勵值和優勢函數產生影響。通過應用重要性採樣，我們可以對這些變化進行校正，以便在策略更新時更準確地估計期望的回報。確保每次策略更新的幅度都在合理的範圍，從而提高系統的穩定性和效能。PPO 算法主要是

透過將期望總獎勵最大化為目標來學習尋找最佳的策略，使代理在環境中能夠有效的朝著預期的目標前進。為了找出系統的最佳的策略，PPO 算法透過與環境互相迭代交換資訊。如圖 4，利用過往的數據資料集成一個訓練集，作為經驗，並採用 CLIP (Conservative Loss Improvement for Policy)[7]方法來限制策略更新的幅度。CLIP 方法是一種用於控制策略更新幅度的方法，藉由重要性採樣來衡量每次策略更新後的動作機率與更新前的動作機率間的分佈差異，限制策略分佈的變化幅度，再將分佈差異乘上優勢函數 (Advantage Function) 計算出每次策略更新可能的回報，有效避免策略大幅度更新可能造成的負面影響，維持系統的穩定性。在文獻中，PPO 在具有挑戰性的複雜環境中表現優於其他強化學習方法。

$$J_{PPO2}^{\theta^k}(\theta) \approx \sum_{s_t, a_t} \min \left( \frac{p_{\theta}(a_t | s_t)}{p_{\theta^k}(a_t | s_t)} A^{\theta^k}(s_t, a_t), \text{clip}\left(\frac{p_{\theta}(a_t | s_t)}{p_{\theta^k}(a_t | s_t)}, 1-\epsilon, 1+\epsilon\right) A^{\theta^k}(s_t, a_t) \right) \quad \text{式 16}$$

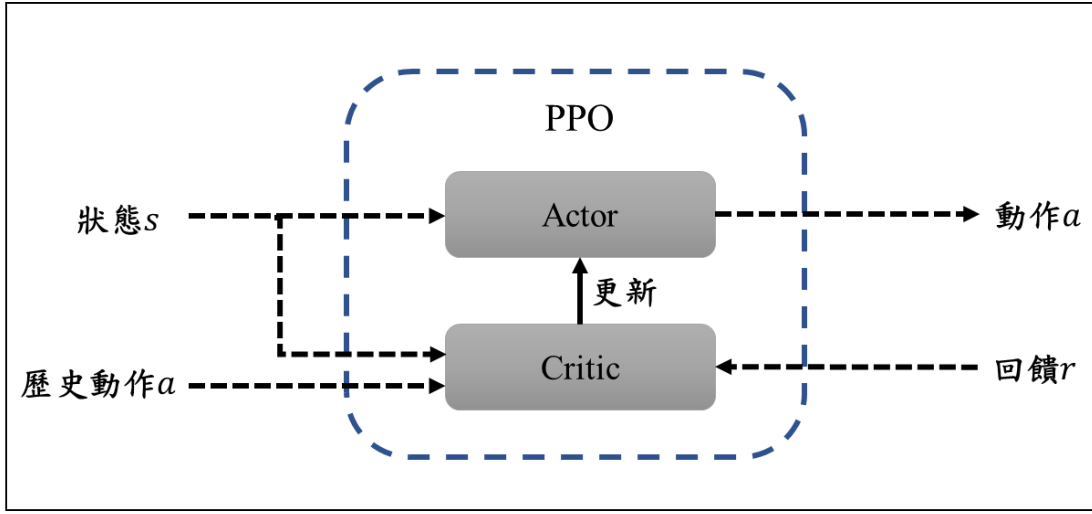


圖 4、PPO 算法架構圖[16]

策略梯度 (Policy Gradient) 用於強化學習中的優化算法，其目的是通過調整優化代理的策略 (Policy) 來最大化期望的總獎勵。在 PPO 中，政策梯度通常是通過計算優勢函數 (Advantage Function) 的梯度來實現的。優勢函數是一個用來衡量一個行動相對於其他行動的優勢程度的函數，其計算方式通常涉及當前狀態的價值函數和行動價值函數[7]，如式 17。 $\pi(a|s)/\pi_{old}(a|s)$  是新政策  $\pi$  和舊政策  $\pi_{old}$  之間的機率比。策略梯度不易確定每一步更新的大小。如果每步過小，則學習出來的策略會花上許多不必要的等待時間，若每步太大則會造成訓練出來的策略無法收斂。PPO 與 TRPO 皆為策略梯度法，但相較起來 TRPO 每次更新策略時，都需要解決一個凸規劃問題，並且需要計算二階導數，計算量較大，使用一種簡單的梯度下降算法的 PPO 更容易實現參數調整，及避免 TRPO 中過大的梯度方差。

$$L_{\pi_{old}}(\pi) = E_{s,a} [\gamma_{\pi}(s,a) \cdot A^{\pi_{old}}(s,a)] + \eta(\pi_{old}) \quad \text{式 17}$$

PPO 使用一種融合了策略梯度和價值函數的網路架構，主要由 Actor 網路和 Critic 網路[17]組成。Actor 網路主要負責學習如何決定在給定狀態下該採取什麼動作並更新策略以最大化期望回報；而 Critic 網路負責評估 Actor 網路策略選擇的優劣程度，依據觀察到的狀態，將其轉換為動作機率分佈，進而實現強化學習。

Actor-Critic 網路架構可以有效地處理連續動作空間的問題，有效解決了策略梯度算法中出現的梯度變化問題，提高了 PPO 學習的穩定性及收斂速度。本研究構建了 Actor-Critic 網路架構並使用 PPO 做為策略梯度的方法。

- Actor: 假設狀態  $s$  的觀察值為  $o$ ，動作  $a$  的機率為  $\pi(a|o, \theta)$ ，其中  $\theta$  是 Actor 網路的權重參數， $P$  表示概率分佈。Actor 的目標是學習一個策略  $\pi$ ，使得在給定狀態下選擇的行動可以最大化期望回報，如式 18[17]。
- Critic: 假設狀態  $s$  的觀察值為  $o$ ，對於該狀態下的動作  $a$ ，其價值函數為  $V(s, \theta)$ ，其中  $\theta$  是 Critic 網路的權重參數， $R_t$  表示從時間  $t$  開始的期望回報。Critic 的目標是學習一個價值函數，使得該狀態下的行動可以得到最大化的期望回報，如式 19[17]。

$$\pi(a | o, \theta) = P(a | o, \theta) \quad \text{式 18}$$

$$V(s, \theta) = E[R_t | s_t = s, a_t = a] \quad \text{式 19}$$

獎勵函數 (Reward function)[18] 主要用來評估控制器行動的好壞的函數。該函式定義了在每個時間步策略的表現，並依據好壞產生獎勵或懲罰，依照這些獎勵來指引控制器往更優的策略學習。在 PPO 算法中，代理通過與環境互動來獲取狀態和獎勵，然後根據策略網路產生動作決策，鼓勵控制器保持期望高度，並且對於過衝或墜落進行懲罰。本研究基於以上規則進行獎勵函數設計，並於研究中實驗證實此獎勵函數可以協助代理在訓練中找到並收斂到最佳策略。

## 六、系統暫態及穩態響應

本研究使用系統暫態與穩態響應做為評判不同控制器在不同環境中表現的基準。暫態響應[19]是指一個系統在經歷變化後的短暫響應，通常是指系統在時間  $t=0$  之後，隨著初始條件或者外界刺激的改變，在這段時間內，系統輸出還未達到穩定狀態所產生的臨時性響應。穩態響應則代表系統在長時間內的狀態行為，主要反應出系統的穩定性和控制性能，當一個系統的暫態響應消失之後，系統的狀態保持穩定不再發生變化，系統的輸出逐漸趨於一個固定的值或穩定的狀態，是評判控制系統效能好壞的重要指標。本研究根據系統的暫態及穩態響應來對控制系統的評估量化，用來分析系統在面對環境的性能與合適程度。

因此，暫態及穩態之間的差異在於它們測量的時間範圍和控制系統的性能方面，如圖 5。上升時間關注控制系統輸出從初始值到達最終穩定值的速度；安定時間關注系統從暫態狀態進入穩態並保持在一定誤差範圍內的速度。圖一為系統暫態及穩態響應圖，控制系統經 MATLAB 量化後，若欲評斷該系統在環境中飛行的優劣程度，可以透過系統暫態及穩態響應指標與其他控制系統進行比較，了解每個控制系統在不同狀態的優劣。暫態是指一段短時間，如系統受到暫時干擾，而這段處於平衡狀態前就稱為「暫態響應」，而穩定後就稱為「穩態響應」。下列是與暫態穩態響應相關的五種名詞與定義：

(一) 上升時間 (Rise time): 系統由最終目標的 10% 上升至 90% 所需的時間。

- (二) 安定時間 (Settling time):系統震盪幅度小於最終目標特定百分比(通常為 5%~2%)所需時間。
- (三) 過衝率 (Percentage overshoot):表示超過目標值的最大百分比。
- (四) 延遲時間 (Delay time):系統響應第一次到達目標值一半所需的時間。
- (五) 穩態誤差 (Steady-state Error):指系統隨著時間達到無窮大(即響應達到穩態)時，系統輸入與輸出之間的差值極限，實際輸出值與期望輸出值的偏差。

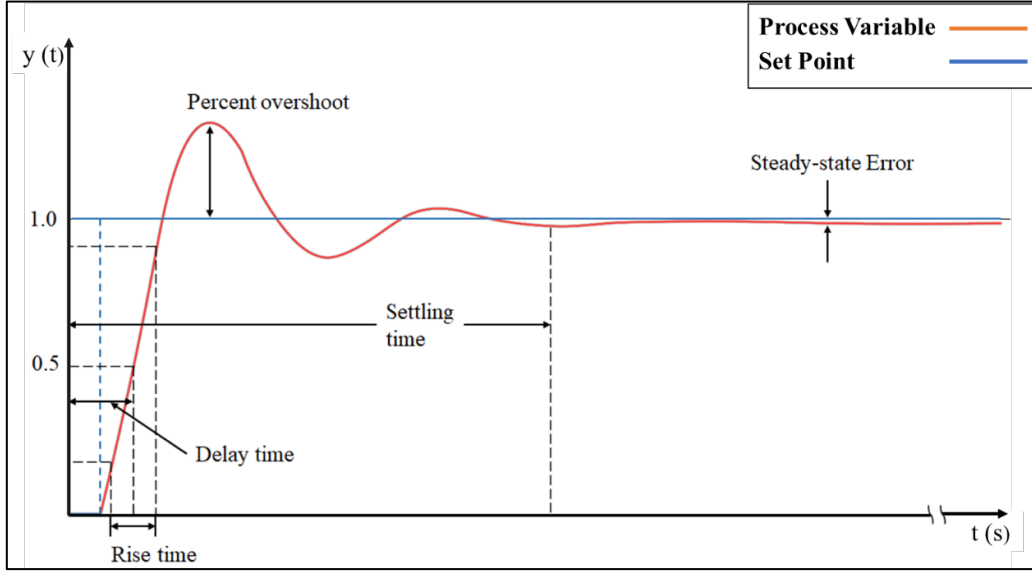


圖 5、系統暫態及穩態響應圖[19]

## 七、性能指標

本研究藉由性能指標[20]的量化以更客觀的判斷各控制器的優劣勢，不同性能指標判斷不同情況下控制之能力。我們使用三種誤差積分準則作為性能指標，分別為平方誤差積分 (Integral Squared Error, ISE)(式 20)、絕對誤差積分 (Integral of Absolute Error, IAE)(式 21)以及時間絕對誤差積分 (Integral Time Absolute Error, ITAE)(式 22)。ISE 用於控制器需要快速達到穩定狀態並抑制最大誤差的情況，主要針對控制系統的反應速度和最大誤差的抑制，但對於微小誤差較不敏感；IAE 與 ISE 相反，其著重於系統的穩定性並抑制小誤差，主要用來判斷在穩態時之誤差；ITAE 比起 IAE 多乘於時間，更著重於調整時間，負責評估控制系統在動態響應時的性能表現，與控制系統的快速穩定與調整能力有很大的關聯。

$$ISE = \int_0^{\infty} [e(t)]^2 dt \quad \text{式 20}$$

$$IAE = \int_0^{\infty} |e(t)| dt \quad \text{式 21}$$

$$ITAE = \int_0^{\infty} t[e(t)] dt \quad \text{式 22}$$

## 參、研究方法及步驟

### 一、模擬環境及四軸無人機硬體

Parrot Mambo mini drone (以下稱 Parrot) 是一台小型四軸無人機，適用於室內及戶外，大多用於教育。該載具配備超聲波感測器、高度計、慣性測量單元 (IMU) 及鏡頭。使用 Parrot 的飛行模型作為本研究四軸無人機模擬機型，並利用 MATLAB/Simulink 模組化及具有官方無人機相關模組優勢做為軟體在環 (Software-In-the-Loop, SIL) 模擬及控制系統開發之軟體。本研究系統採樣時間維  $T=0.005$  (s)。

透過 MATLAB/Simulink 修改官方四軸無人機模擬之模型[21]，針對高度控制器設計 PID 算法及強化學習演算法，並使用 Simulink 進行模擬，模擬模組大致上分成六個區塊，如圖 6 所示，說明如下：

- (a) 命令 (Command): 設定四軸無人機飛行目標位置。
- (b) 飛行控制系統 (Flight Control System): 控制飛機姿態及位移之控制器，本研究主要研究之高度控制器位於此區塊。
- (c) 四軸無人機飛行模型 (Multicopter Model): 藉由四軸無人機動力學建立屬於 parrot 之模型，將飛行控制系統傳入模型以計算出四軸無人機位置及角速度等物理量。
- (d) 感測器模組 (Sensor Model): 接收環境及飛行模型輸出之訊號以模擬實際飛行時，感測器接收之訊號，本研究訊號接有加入雜訊 (noise) 以盡可能模擬真實環境感測器會出現之誤差。
- (e) 環境模組 (Environment Model): 模擬大自然之環境，如溫度、氣壓及地球磁場等相關數據，以貼近真實環境。

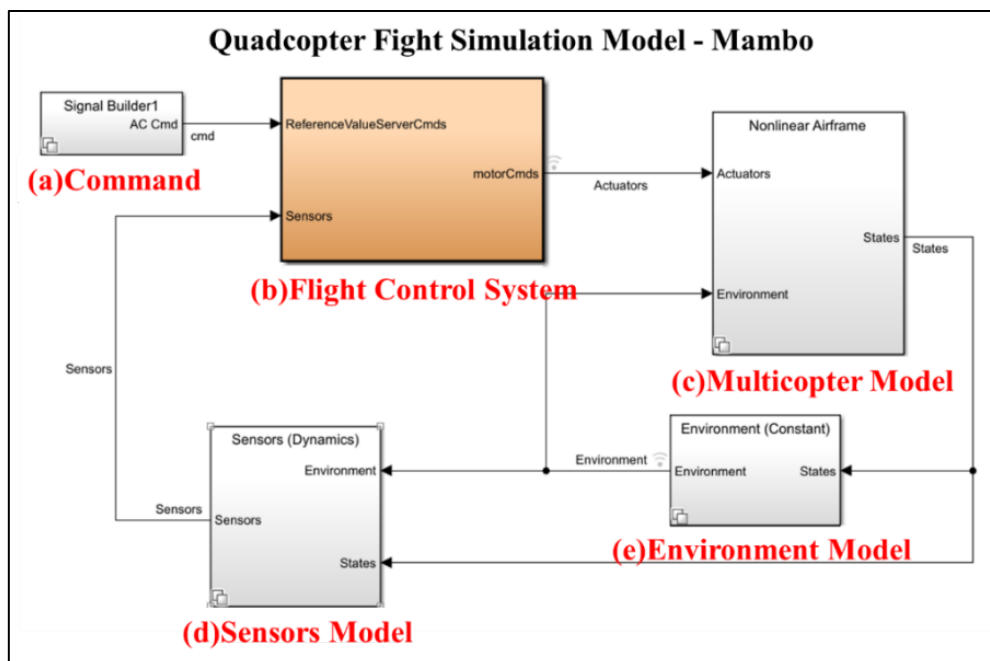


圖 6、模擬環境建構圖[包含(a)命令、(b)飛行控制器、(c)四軸無人機飛行模型、(d)感測器、(e)環境][22]



## 二、 比例-積分-微分-高度控制器

(一) **透過 ZN 法則調整 PID 參數:**本研究中，設定了 1 公尺的目標高度，進行四軸無人機高度控制器的 PID 值調整。使用 ZN 法則，幫助我們系統性的找尋合適之 PID 參數，調整規則參照表 1。首先將積分增益(I)與微分增益(D)值設為 0，然後調整比例增益值(P)，使控制器輸出保持穩定的震盪。此時的比例增益值(P)稱為  $K_u$ ，根據震盪週期  $T_u$  計算，得出臨界增益值  $K_u$ ，從而調整出一組 ZN-PID 值。

(二) **人工優化 ZN-PID 參數:**ZN 法則雖然能夠快速調整出一組值，但其特性包括響應激烈及不穩定等問題，因此，本研究透過表 2 中規則進行人工調整，調整過後值分別為  $K_p=0.57$ 、 $K_i=0.01$  及  $K_d=0.27$ [22]。

表 1、ZN 法則調整規則[2, 23]

Control Type	$K_p$	$K_i$	$K_d$
P	$0.5K_u$	/	/
PI	$0.45K_u$		
PID	$0.6K_u$	$1.2K_u/T_u$	$0.075K_uT_u$

表 2、暫態穩態響應與 PID 關係表[23, 24]

	上升時間 (Rise Time)	過衝 (Overshoot)	穩定時間 (Settling time)	穩態誤差 (Steady-State Error)	穩定性 (Stability)
P 增加	減少	增加	小幅增加	減少	降低
I 增加	小幅減少	增加	增加	大幅減少	降低
D 增加	小幅減少	減少	減少	小幅變化	提升

## 三、 模糊邏輯之高度控制器

本研究使用與目標高度誤差( $e_z$ )及高度誤差微分( $d_z$ )設置規則庫，混合運算使模糊邏輯高度控制器可以針對不同情況下皆有合適之規則，更精確的輸出控制訊號。模糊邏輯高度控制器設置步驟如下[4]：

(一) **模糊化:**定義模糊語言並設定其數量，將系統控制的輸入數值分配給具歸屬函數的模糊集。本研究將模糊語言定義如

(二) 表3，分別為:NB (Negative Big)、NM (Negative Medium)、NS (Negative Small)、ZE (Zero)、PS (Positive Small)、PM (Positive Medium)、PB (Positive Big)， $e_z$ 的模糊語言為小寫， $d_z$ 和最後輸出F為大寫。

(三) **各語言變數資料庫:**訂定歸屬函數的範圍值，範圍值越精準控制效果會越顯著。本研究目標為室內四軸無人機，故z軸最大誤差量( $e_z$ )設定成 $\pm 1.5$ 公尺，而誤差積分( $d_z$ )經觀察飛行狀況後設定成 $\pm 2.5$ 每秒公尺，輸出最大設

定成 $\pm 2.5$ 牛頓，以最大正負值去分成7個模糊語言，其中NB及PB採用梯形歸屬函數，其餘皆採用三角型歸屬函數。

(四) **模糊規則庫**:使用If-Then方式建立，並參照

(五) 表3所定義之規則來描述系統輸入及輸出的關係[23]，如If  $e_z$  is nb and  $d_z$  is NB then F is PB。圖7為本研究之Mamdani模糊控制3D圖。

(六) **模糊推理引擎**:本研究模糊推理引擎採用Mamdani推理系統。And method採min；Or method採max。

(七) **重心法解模糊化**:解模糊化採用Center of Gravity (COG)重心法，如式15，其中 $\mu(x)$ 為模糊規則庫的歸屬函數。

表 3、高度誤差與 z 軸速度之模糊規則表[23, 25]

$d_z$ \ $e_z$	nb	nm	ns	ze	ps	pm	pb
NB	PB	PB	PB	PB	PM	PS	ZE
NM	PB	PB	PB	PM	PS	ZE	NS
NS	PB	PB	PM	PS	ZE	NS	NM
ZE	PB	PM	PS	ZE	NS	NM	NB
PS	PM	PS	ZE	NS	NM	NB	NB
PM	PS	ZE	NS	NM	NB	NB	NB
PB	ZE	NS	NM	NB	NB	NB	NB

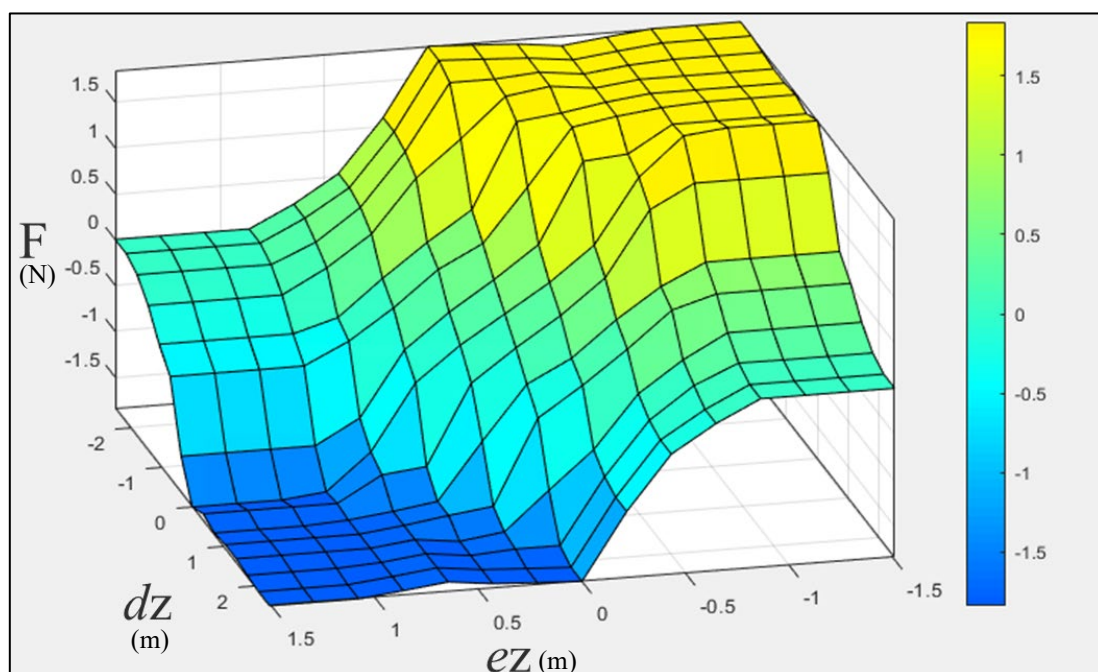


圖 7、Mamdani 模糊控制 3D 面[23, 25]



#### 四、近端策略最佳化算法之高度控制器

(一) **建構 PPO 高度控制器:**本研究透過 Simulink 設計四軸無人機 PPO 高度控制器，並整合環境系統。強化學習高度控制器(圖 8)由一個強化學習模塊及三個輸入組成，分別是(a)狀態輸入、(b)獎勵函數及(c)重製條件。狀態輸入包括機體當前高度、加速度等數據；獎勵函數依照誤差高度給予不同懲罰；重製條件如墜機、飛出指定範圍，會重新訓練以節省這些訓練時間，也使訓練更容易收斂。最終這些數據被透過 Actor-Critic 網路層級 PPO 算法計算後輸出符合當時高度所需要力的大小。本研究以目標高度 1 公尺且無外在干擾環境中進行 PPO 訓練。

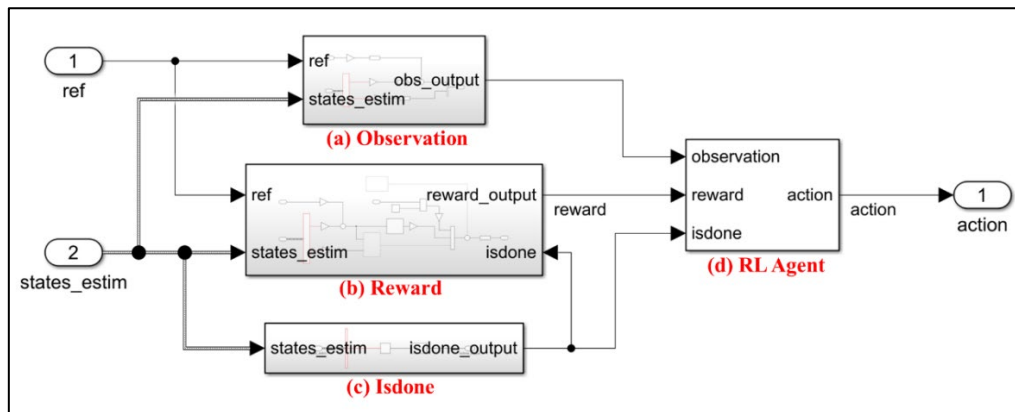


圖 8、PPO 控制器結構圖[(a)四軸無人機狀態輸入、(b)獎勵函數、(c)重製條件、(d)強化學習模塊][26]

(二) **設定 PPO 超參數及 Actor-Critic 網路層設計:**使用 MATLAB 設計 Actor-Critic 網路層並調整了 PPO 超參數，Actor 網路(圖 9)使用觀察模塊的數據作為輸入[22]，通過完全連接層和 ReLU 激勵函數進行非線性轉換，以提取數據重要特徵並生成不同動作的機率分布，最終輸出無人機的動作。Critic 網路(圖 10)也使用觀察模塊的數據[22]，評估 Actor 的策略效果，並輸出價值函數估計值。超參數(表 4)包括學習率、折扣因子和訓練樣本數等[22]，這些超參數影響學習速率和效果。

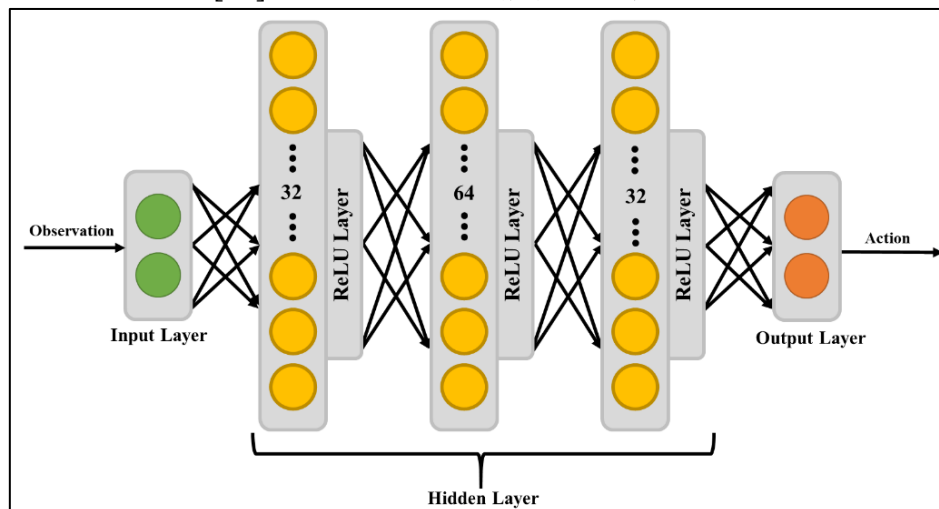


圖 9、Actor 網路層結構圖[22]

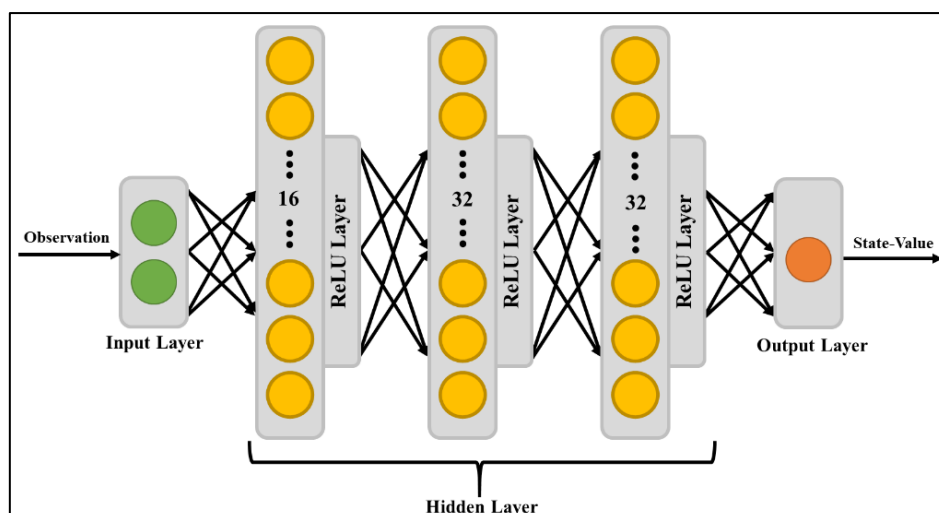


圖 10、Critic 網路層結構圖[22]

表 4、本研究所設置相關之超參數[22]

熵損失 (Entropy Loss Weight)	0.1	訓練總步數 (Max Episodes)	8000
代理步數 (Max Steps Per Episode)	2000	優勢估計器平滑因子 (GAE Factor)	0.95
代理採樣時間 (Sample Time)	0.005	演員-學習率 (Actor Learnrate)	6e-06
批量大小 (Mini BatchSize)	20	評論家-學習率 (Critic Learnrate)	2e-05

## 五、 隨機干擾

除了不同目標高度，本研究也特別關注了外部干擾對高度控制器的影響，透過在在模擬環境中引入外部隨機干擾[27]，模擬不穩定自然風的影響，以更符合實際飛行情況。在 Z 軸方向上引入了持續 1 秒的隨機大小的力，並每秒改變力的大小和方向，並使用 ITAE、IAE 和 ISE 作為評估指標。我們的設計旨在將這些干擾力轉化為對四軸無人機造成的位移。這個模擬模型被加入到圖 6(e)中，每秒位移的大小如圖 11 所示。

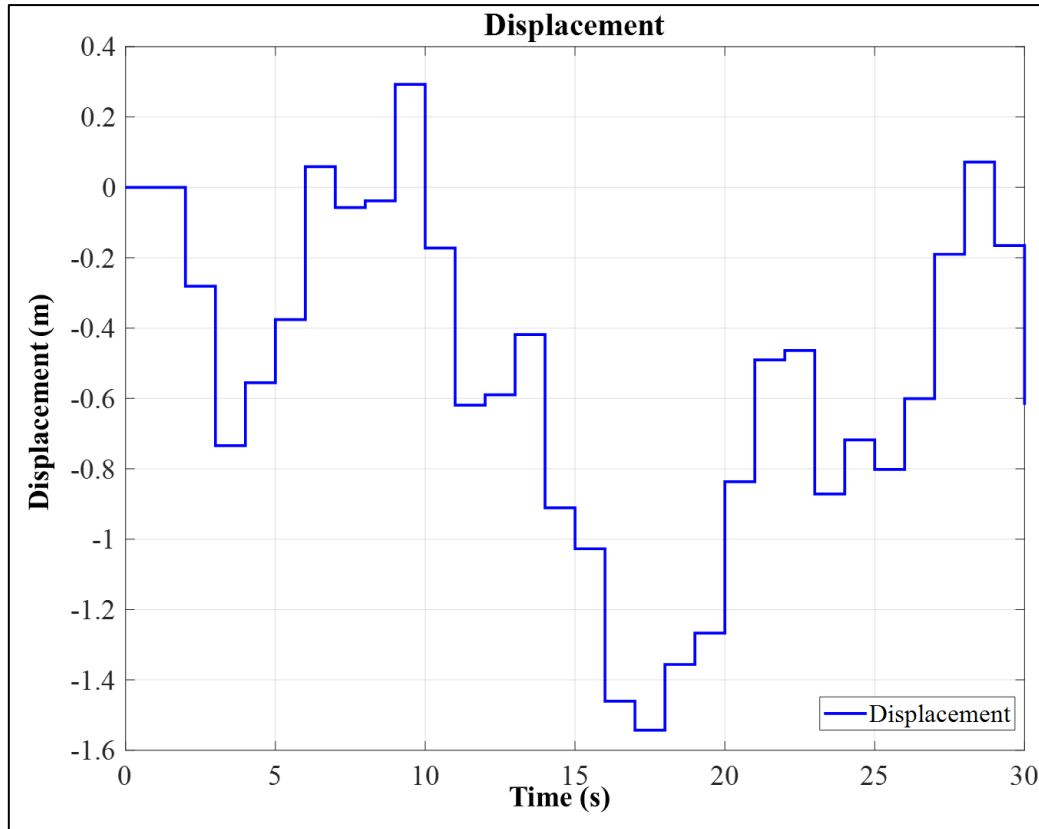


圖 11、本研究干擾轉換對四軸無人機造成之高度位移影響[26]

## 肆、結果與討論

本研究實驗結果如圖 12 圖 14 所示，主要表示在無加入干擾的環境中，分別以 0.1、1.0、3.0 公尺作為目標高度，運用上升時間、安定時間、過衝、IAE 四種指標，比較 ZN-PID、FLC 與本研究開發之 PPO 控制器飛行性能比較，三種控制器皆以 1.0 公尺作為設計及訓練的基準。由圖 12 可以看出，在目標高度 0.1 公尺時，ZN-PID 有明顯的過衝且經過 5 秒後還未完全收斂至目標高度；FLC 與 PPO 控制器皆於 0.5 秒內成功收斂至目標高度；根據

表 5 呈現，PPO 控制器不管在上升時間、安定時間、過衝與 IAE 均略優於 FLC，因此在 0.1 公尺的部分以 PPO 控制器表現最佳。將目標高度提高到 1.0 公尺時，可以看到 ZN-PID 產生過衝且到最後依然留有一段穩態誤差，無法穩定收斂；根據表 5 呈現，FLC 控制器雖然在過衝的部分表現優異，但 PPO 控制器在其餘指標的表現上皆優於 FLC，且只有微幅的過衝 1.2%。當我們把目標高度提高到 3.0 公尺時，可以看出 ZN-PID 上升速度緩慢，且收斂問題也沒有解決；而 FLC 雖然幾乎沒有過衝，只有微小的 0.0878%，但其耗時近 2 秒才完成收斂，上升時間緩辦；PPO 控制器上升時間僅花費 0.65 秒，且只有微幅過衝 3.8%。根據

表 5 呈現進行綜合比較看出，ZN-PID 無論在哪一個高度，皆存在一定的穩態誤差，由此可知透過 ZN 法則無法調整至最佳化之 PID 參數；FLC 雖然在飛行

的過程中過衝極小，但隨著目標高度的上升，可以看出 FLC 在上升時的飛行效率降低很多；PPO 控制器在上升時間擁有絕對的優勢，控制器在穩態飛行時的表現也為三種控制器中最佳的，且都只有微小過衝。由以上結果可知 PPO 控制器具備優秀之性能以作為四軸無人機之高度控制器。

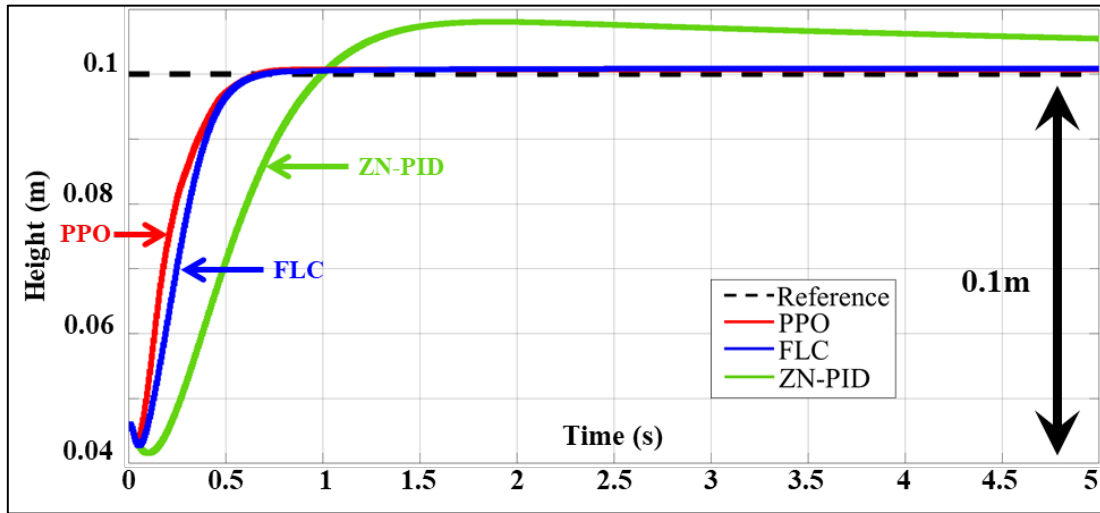


圖 12、三個控制器在目標高度 0.1 公尺下的 5 秒飛行的暫態和穩態響應[28]

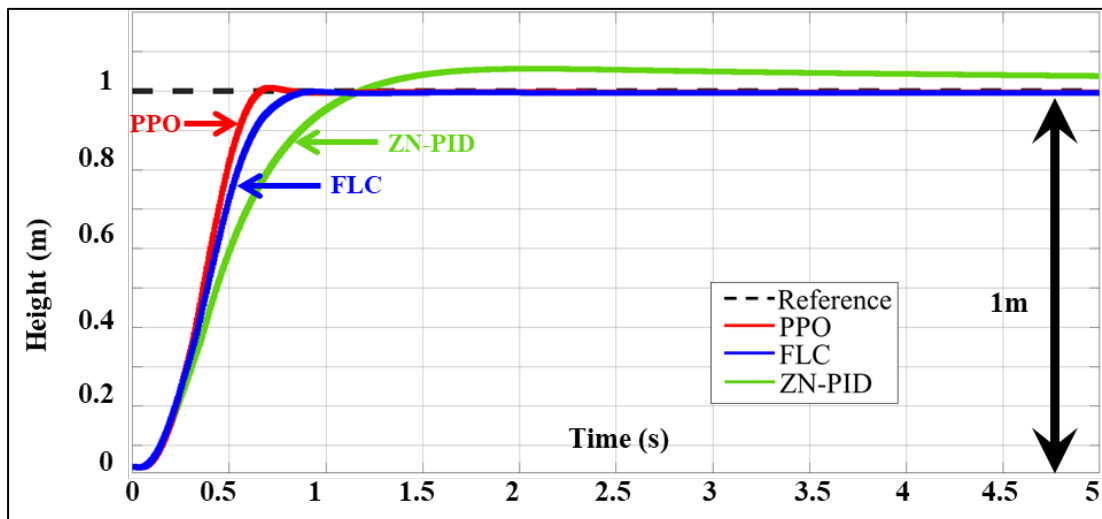


圖 13、三個控制器在目標高度 1.0 公尺下的 5 秒飛行的暫態和穩態響應[28]

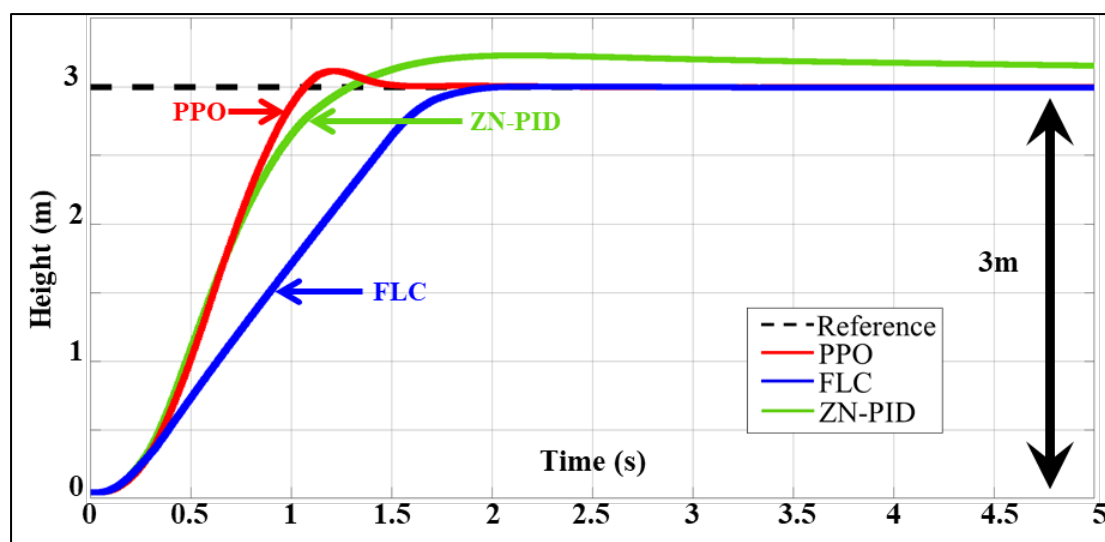


圖 14、三個控制器在目標高度 3.0 公尺下的 5 秒飛行的暫態和穩態響應[28]

表 5、目標高度為 1、0.1 和 3 公尺下各控制器的數值性能評估 (紅色粗體代表最佳控制器數值性能)[28]

Height (m)	1.0			0.1			3.0		
Controller System Type	ZN-PID	FLC	PPO	ZN-PID	Fuzzy	PPO	ZN-PID	Fuzzy	PPO
Rise Time (s)	0.7207	0.4851	<b>0.4010</b>	0.7565	0.3933	<b>0.3573</b>	0.7576	1.2493	<b>0.6491</b>
Settling Time (s)(5%)	2.9789	0.7029	<b>0.5935</b>	—	0.4635	<b>0.4564</b>	—	1.6444	<b>0.9889</b>
Overshoot (%)	5.6669	<b>0</b>	1.2237	8.0624	0.8482	<b>0.7252</b>	7.6294	<b>0.0878</b>	3.8419
IAE	0.6454	0.4055	<b>0.3746</b>	0.0564	0.0188	<b>0.0165</b>	2.5534	2.7154	<b>1.8612</b>

觀察在不同高度下各控制器之性能後，比較在有無外部干擾下飛行結果，從圖 15 中可以觀察出，在模擬無干擾環境下，PPO 控制器在上升及下降過程中都沒有過度震盪的情況，飛行保持相當平穩；相反，ZN-PID 則出現明顯的過度震盪現象。根據表 6 的數據，PPO 控制器上升到目標高度 1 公尺，比 ZN-PID 控制器節省了 3.28% 的時間。當從 1 公尺上升到 2.5 公尺時，雖然 ZN-PID 控制器的時間略少於 PPO 控制器 0.29%，但 ZN-PID 仍然存在明顯的過度震盪問題，且無法在 10 秒內達到穩態。IAE 性能指標表明 PPO 控制器的穩態效果優於 ZN-PID 控制器，在過度震盪方面，PPO 控制器只有 1% (0.01 公尺)，相較於 ZN-PID 的 22% (0.22 公尺) 減少了 95.45%。這再次證明了 PPO 控制器具有適應不同高度的能力。

在圖 16 中，我們引入了外部隨機干擾，其位移每秒變化範圍為 0.3 公尺至負 1.5 公尺。從圖 16 中可以看出，當受到外部干擾時，PPO 控制器的振幅明顯小於 ZN-PID。根據表 6 中的各性能指標，PPO 控制器的表現優於 ZN-PID 控制器，以 ITAE 作為評估動態響應，PPO 改善了 23.62%。在穩態性能方面，使用 IAE 和 ISE 指標進行評估，可以看到 PPO 控制器分別改善了 24.22% 和 14.29%。綜合以上結果，我們可以得出結論，PPO 控制器在面對外部環境干擾時，具有卓越的抗干擾性能，適用於四軸無人機的高度控制。

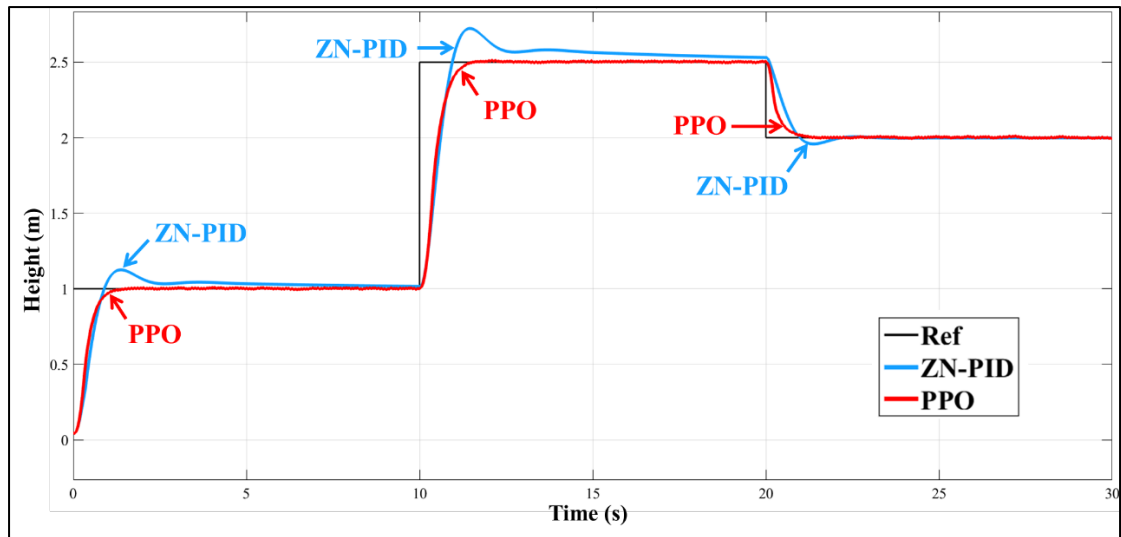


圖 15、PPO 及 ZN-PID 無干擾時飛行圖(高度分別為 1.0 公尺、2.5 公尺及 2.0 公尺)[26]

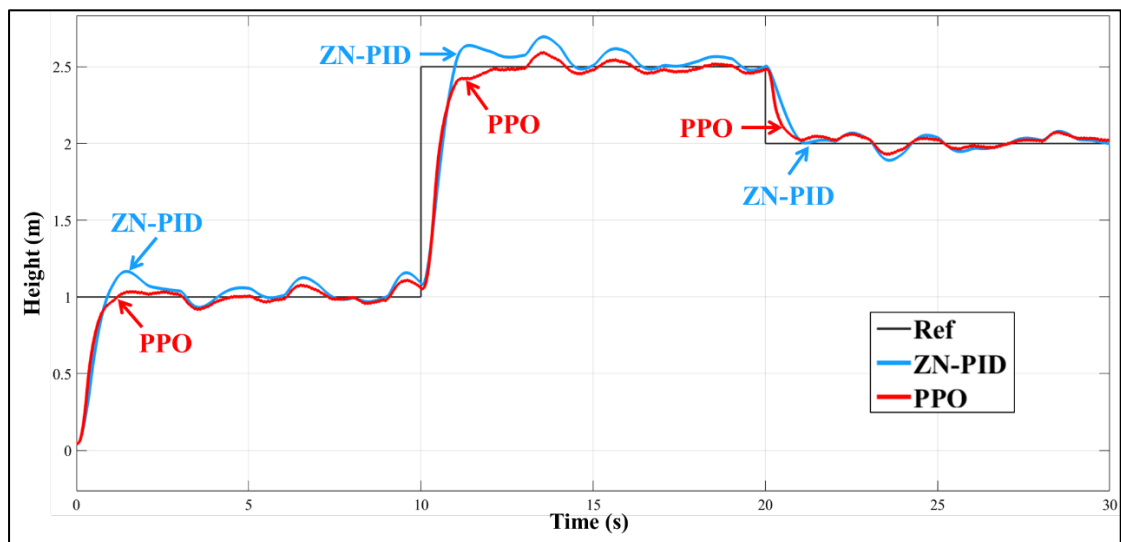


圖 16、PPO 及 ZN-PID 在具有干擾環境時飛行圖，如圖 11 之干擾(高度分別為 1.0 公尺、2.5 公尺及 2.0 公尺)[26]

表 6、ZN-PID 與 PPO 高度控制器於有無干擾之性能比較(紅色粗體代表最佳控制器數值性能)[26]

<b>Without Disturbance</b>			
	Benchmark ZN-PID	PPO	Improvement (%)
ITAE (m · s)	4817.47	<b>2426.32</b>	49.63 %
IAE (m)	2.45	<b>1.37</b>	44.08 %
ISE (m <sup>2</sup> )	1.29	<b>1.07</b>	17.05 %
1.0 m Rise Time (s)	0.61	<b>0.59</b>	3.28 %
2.5 m Rise Time (s)	<b>10.47</b>	10.5	-0.29 %
Peak (m)	0.22	<b>0.01</b>	95.45 %
<b>With Disturbance</b>			
ITAE (m · s)	6501.74	<b>4965.88</b>	23.62 %
IAE (m)	2.89	<b>2.19</b>	24.22 %
ISE (m <sup>2</sup> )	1.33	<b>1.14</b>	14.29 %
1.0 m Rise Time (s)	0.61	<b>0.59</b>	3.28 %
2.5 m Rise Time (s)	<b>10.49</b>	10.53	-0.38 %
Peak (m)	0.19	<b>0.09</b>	52.63 %

## 伍、結論

本研究利用 MATLAB 與 Simulink 結合 Parrot Mambo mini drone，研究 ZN-PID 控制器、FLC 以及 PPO 控制器在無干擾環境中面對不同高度的適應能力。實驗結果表明，ZN-PID 在飛行過程中有明顯的暫態與穩態誤差，且收斂速度緩慢；FLC 明顯優於 ZN-PID 控制器，跟 PPO 控制器相比，雖然 FLC 過衝非常微小，但是其在面對高度的情況下，控制器從上升到收斂時間明顯比 PPO 控制器慢。本研究再針對有無隨機干擾環境，依據三種不同目標高度測試 ZN-PID 控制器與 PPO 高度控制器之適應能力。經過觀察，我們可以得出結論，PPO 控制器在動態響應和穩態響應方面均表現優於 ZN-PID 控制器。因此，在外部干擾的飛行環境中，PPO 控制器展現出更優越的適應性和控制能力，使其適用於四軸無人機的高度控制。

在實驗前期，PID 控制器需要花費較長時間調適參數，即便使用 ZN 等系統性調整法，仍然需要依靠經驗法則進行微調；FLC 在前期規則庫的設計與參數調整也需要耗費大量時間與經驗，才能得到優秀的控制系統；相較之下，PPO 控制器雖然也需要設置超參數，但藉由電腦訓練調整，可以省下許多的時間，而研究結果可以看出 PPO 控制器不論在快速上升能力、過衝量或是收斂速度都達成一定之水準且優於另外兩款控制器，能夠更有效率的執行無人機的高精度控制任務。綜合以上結論，本研究認為 PPO 控制器因其優秀的暫態與穩態表現，以及優秀的抗干擾能力，最適合做為四軸無人機之高度控器。

## 陸、未來展望

未來，我們將繼續研究設計四軸無人機之 PPO 姿態控制系統，經由不斷優化 PPO 算法，提高無人機在姿態控制中的性能。這將使四軸無人機在應對各種複雜環境時都能夠保持良好的飛行狀態，透過將 PPO 姿態控制器應用於實際飛行中，觀察並驗證 PPO 姿態控制器具有更優越的自適應控制能力，能夠應對不同干擾的複雜環境。我們還計劃將 PPO 姿態控制器進行實際環境應用，透過引入不同的最佳化演算法，實現 PPO 姿態控制器在三維空間環境中的路徑最佳化，並加入激光雷達的自動避障系統，結合 PPO 姿態控制器的穩定度與高適應性，設計出在不同環境皆可擁有良好飛行性能之四軸無人機自動駕駛系統，並且將這一技術實際應用於巡航、維修檢測、地形偵查等各類不同領域中。