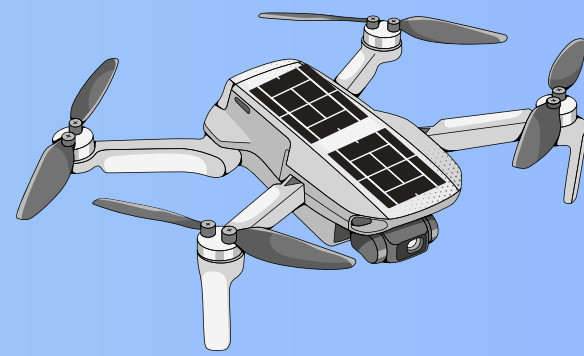


# 分數多代理深度強化學習結合能量虛擬佇列 於行動邊緣運算之 AoI 與能量最佳化



成員：張皓閔 陳鈺憲 李語桐 指導教授：劉建興

## 摘要

隨著自駕車、無人機與智慧城市等應用快速發展，系統對「即時」且「新鮮」的資訊需求愈加重要。然而，傳統僅依賴網路延遲的評估方式已不足以反映資料的實際價值。為此，「資訊年齡（Age of Information, AoI）」成為衡量資料新鮮度的關鍵指標。然而在行動邊緣運算（MEC）環境中，由於裝置資源有限，要同時維持低 AoI 與能量消耗平衡是一項極具挑戰性的問題。

本研究提出一套基於非同步分數多代理深度強化學習的決策架構，並結合能量虛擬佇列機制，用以處理 MEC 系統中的雙目標最佳化：降低 AoI 與提升能量使用效率。我們建立能量收穫與能量消耗模型，確保系統在有限能源下仍能長期運作。

以救災或軍用無人機為應用場景，本研究模擬無人機利用太陽能進行飛行、攝影與計算，並根據任務負載與剩餘能量決定是否將運算卸載至邊緣伺服器（如指揮車）。實驗結果顯示，本方法能有效降低 AoI 並同時節省能量，證明其在資源受限環境中的可行性與效能提升潛力。

## 實驗方法

我們參考了[1]的做法，模型使用 D3QN 來決定任務的卸載位置（離散動作），使用 PPO 來決定任務的等待時間（連續動作），形成一個混合動作的雙層強化學習架構。並從[2]中提取靈感將能量虛擬佇列 Z 與瞬時漂移  $\Delta Z$  則作為能量穩定控制信號，將能量長期限制融入 reward 中，確保代理在降低 AoI 的過程中不會消耗過多能量，讓系統能在長時間運作中保持穩定且高效。

## 系統架構

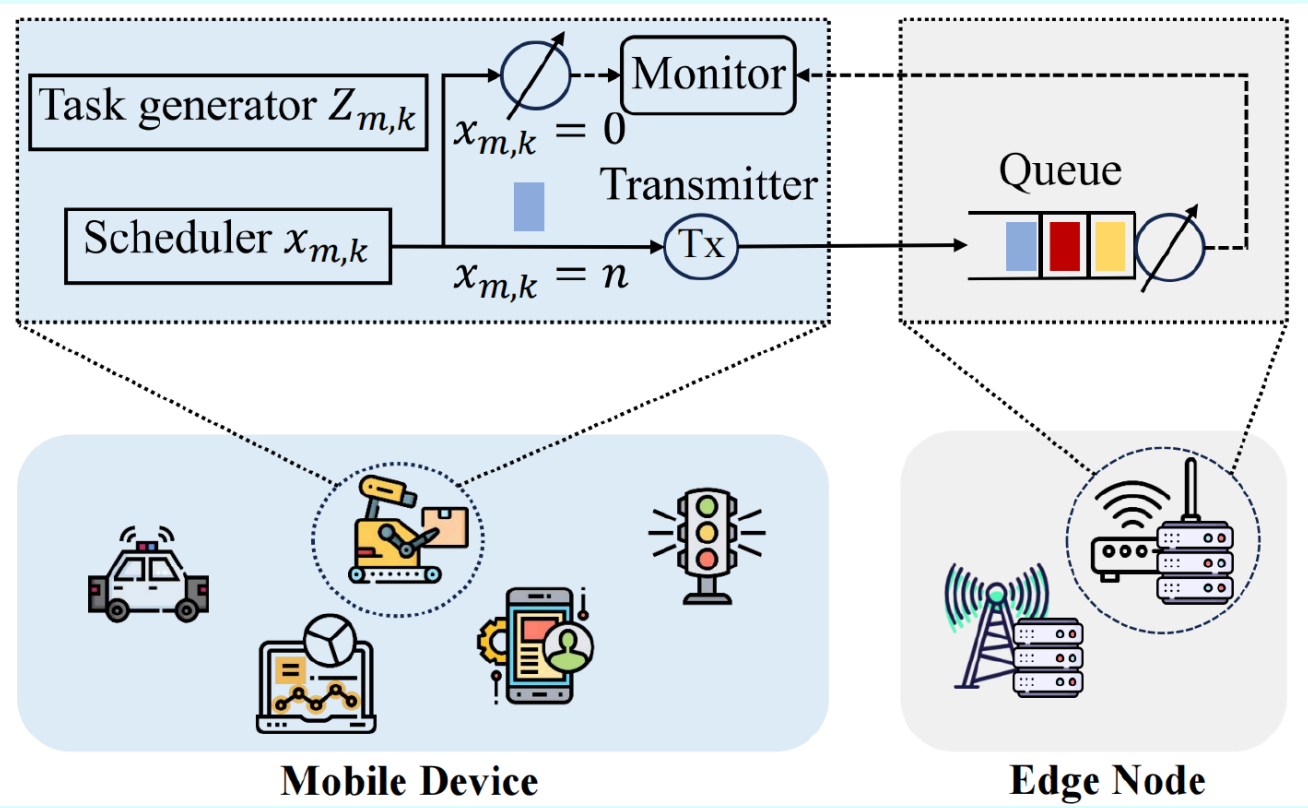


圖1。圖示為一個行動邊緣運算（MEC）系統，其中包含一個行動裝置  $m \in M$  和一個邊緣伺服器  $n \in N$ ，其中不同行動裝置卸載的任務用不同的顏色表示。

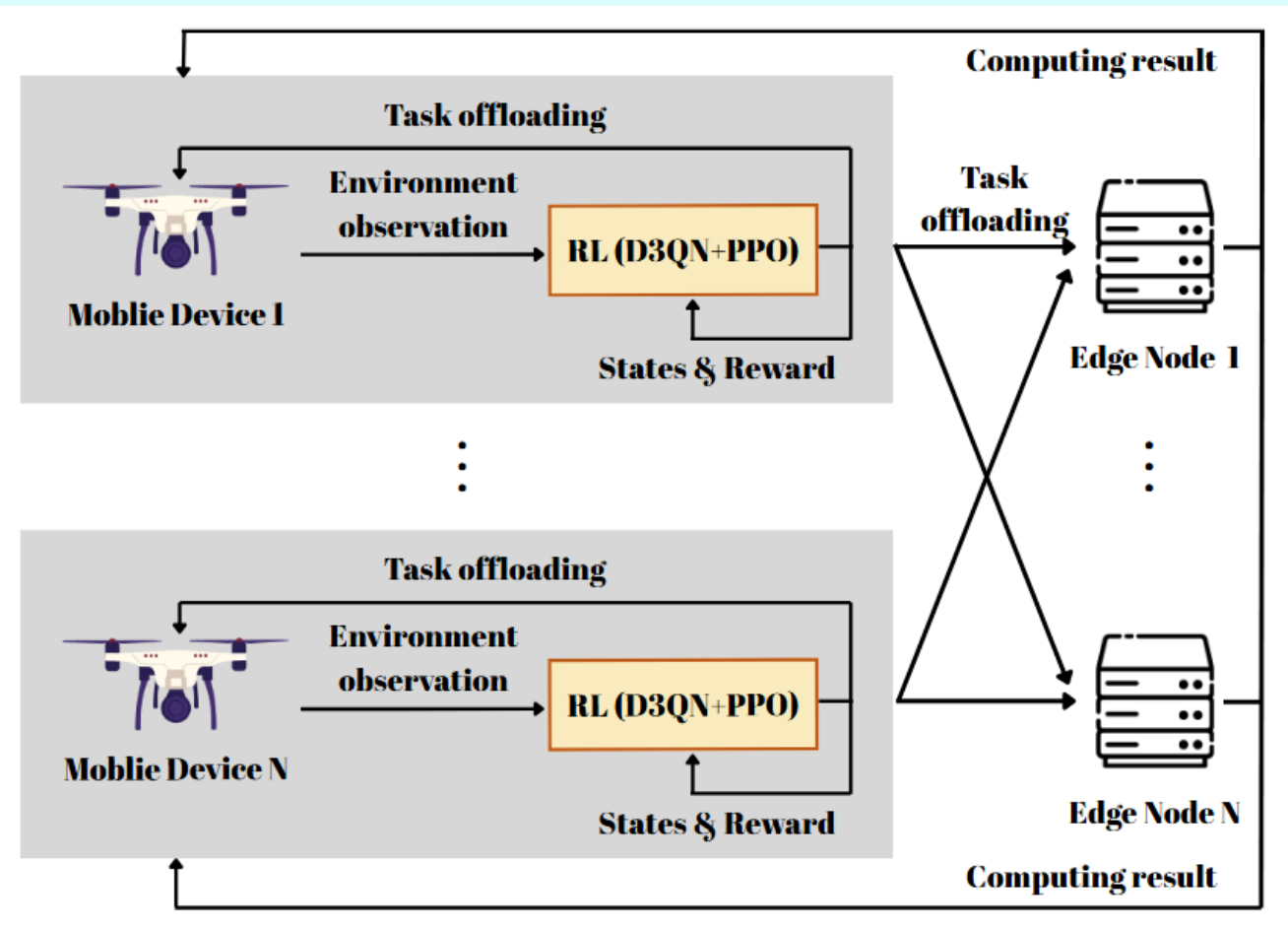


圖2。展示多行動裝置於 MEC 環境中，透過本地強化學習模組（D3QN+PPO）決定任務卸載至不同邊緣伺服器，並接收運算結果以更新後續決策。

## 參考資料

[1] Asynchronous fractional multi-agent deep reinforcement learning for age-minimal mobile edge computing, May 2025.

[2] Minimizing Version Age of Information in Resource-Constrained Multi-Source Systems via Lyapunov and Learning Based Scheduling.

Minimizing Version Age of Information in Resource-Constrained Multi-Source Systems via Lyapunov and Learning Based Scheduling

Asynchronous Fractional Multi-Agent Deep Reinforcement Learning for Age-Minimal Mobile Edge Computing

## 實驗設計

### 瞬時AoI的定義

$$\Delta_m(t) = t - T_m(t), \quad \forall m \in \mathcal{M}, t \geq 0$$

### 平均AoI的定義

$$\Delta_m^{(ave)} \triangleq \liminf_{K \rightarrow \infty} \frac{\sum_{k=0}^K \mathbb{E}[A(Y_{m,k}, Z_{m,k+1}, Y_{m,k+1})]}{\sum_{k=0}^K \mathbb{E}[Y_{m,k} + Z_{m,k+1}]}$$

AoI 為『分數型』目標：  
將『長期平均 AoI』定義為分數型形式 (Fractional Form)，  
分子『AoI 累積面積』除以 分母『總時間』

AoI

### 本地運算能耗

$$E_{local} = k f^2 C$$

k：晶片能號常數  
f：CPU 頻率(Hz)  
C：計算所需週期

### 傳輸能耗

$$E_{tx} = e_{bit} \times B$$

$e_{bit}$ ：每 bit 傳輸能量(J/bit)  
B：任務大小 (bit)

### 能量收穫

$$E_{harvest} \sim U(h_{min}, h_{max}) \quad \text{均勻分布(J)}$$

### 能量虛擬佇列

$$Z(t+1) = \max(Z(t) + E_{use} - E_{harvest}, 0)$$

使用虛擬佇列來記錄能量的虧損

Energy

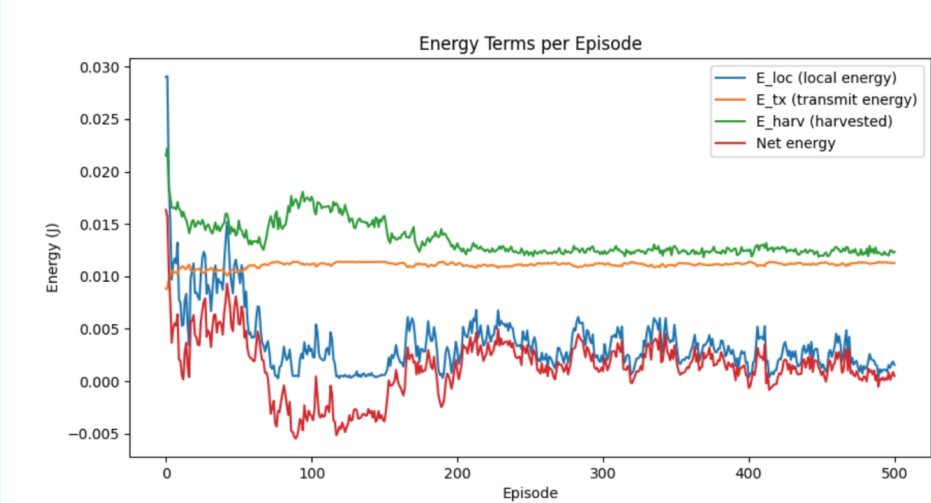
### reward設計

$$R = -(AoI + \lambda_1 \Delta Z + \lambda_2 Z)$$

虛擬佇列的功能是將能量約束轉換成佇列穩定問題，  
Z代表虛擬佇列、  
 $\Delta Z$  代表與前次虛擬佇列的漂移量

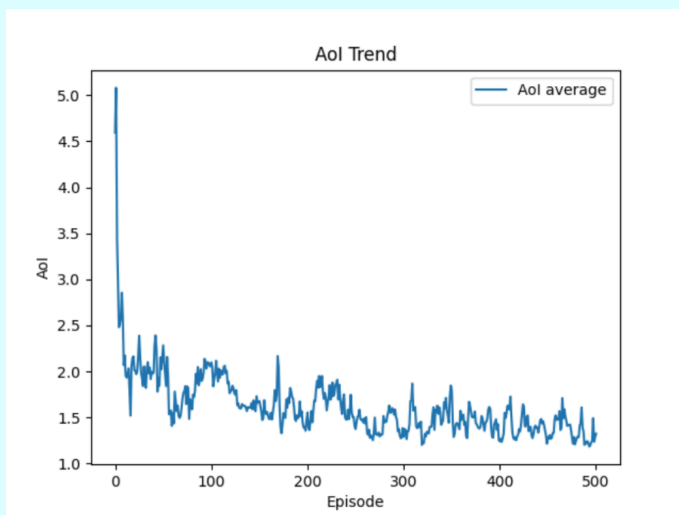
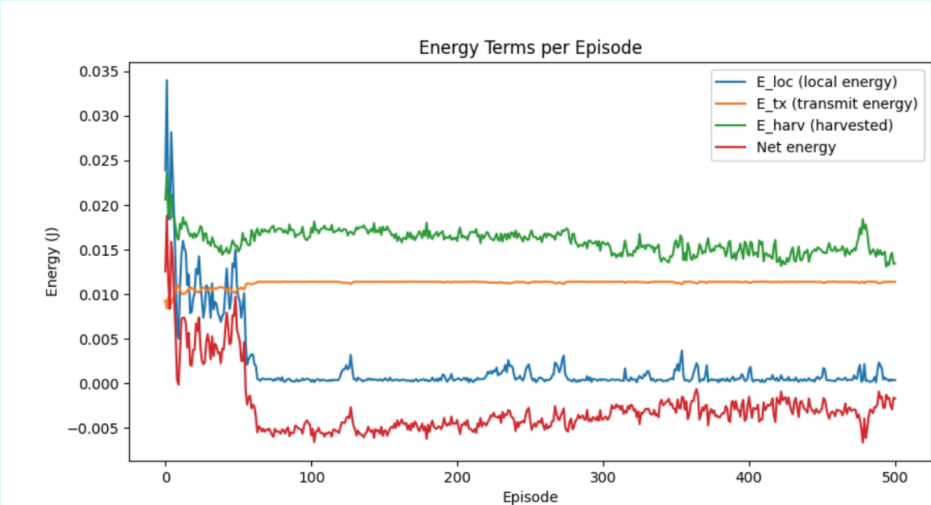
## 實驗結果

無虛擬佇列懲罰  
( $\lambda_1 = 0$ )

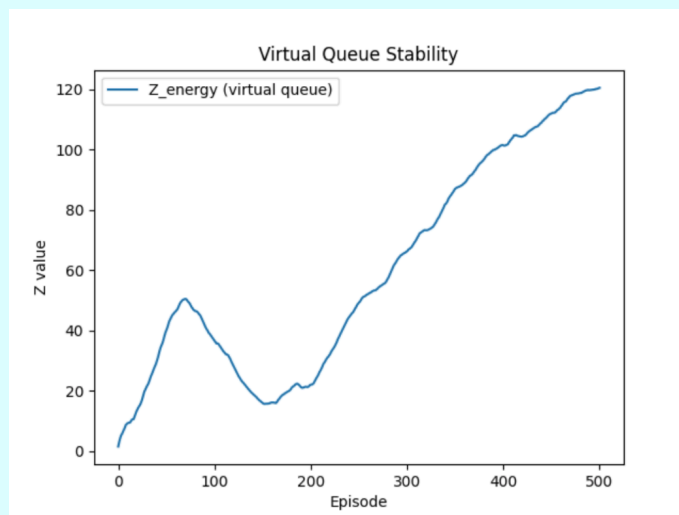
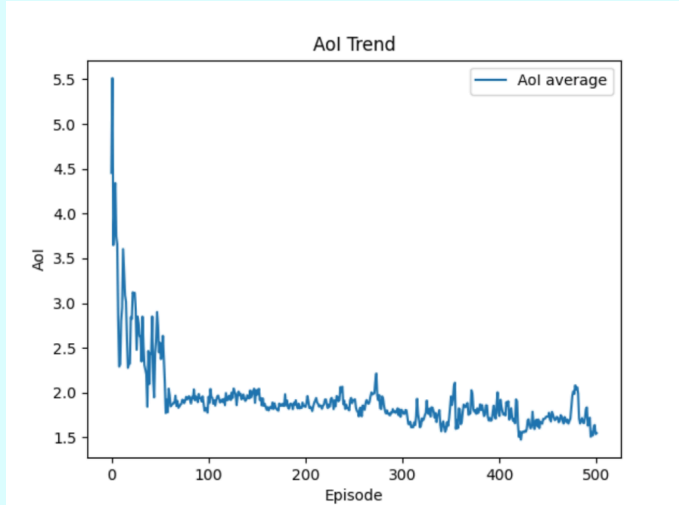


各 Episode 的平均能量  
包括本地運算、傳輸、收穫與淨變化

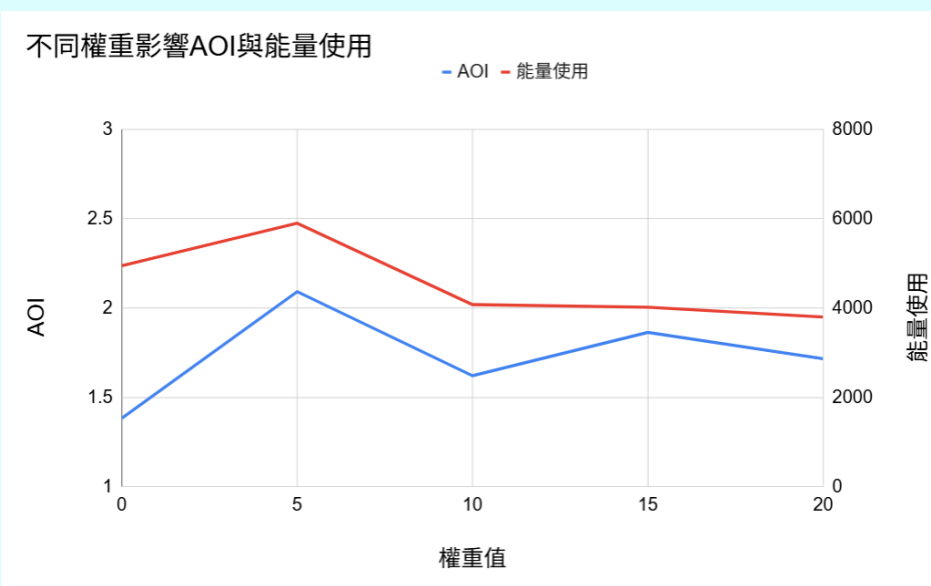
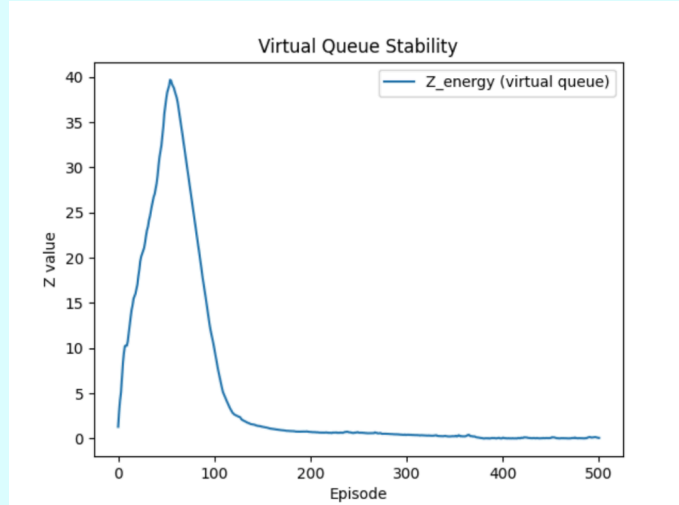
啟用虛擬佇列懲罰



平均 AoI 隨 Episode 的變化



能量虛擬佇列隨 Episode 的變化



左圖為AoI和總能量使用在不同權重下後100回合訓練的平均值。系統在權重 10~20 之間能達到能量與 AoI 的平衡，能量使用逐步下降，但 AoI 仍維持良好表現。

能量使用量從 4946 (焦耳)降到 3796 (焦耳)，降低約 23%。

結論：適度調整權重可以讓系統在資訊新鮮度（AoI）與能量消耗之間找到最佳平衡點，使系統能長期穩定運作。