

Gridworld 問題求解報告

(Solve Gridworld Problem using Value Iteration)

1. 問題背景

本專案的目標是使用「價值迭代演算法 (Value Iteration Algorithm)」來解決「Gridworld (網格世界) 問題」，並透過 Flask 與 HTML/CSS/JS 打造一個互動式網頁應用，讓使用者可以：

- 設定「起點 (S)」、「終點 (E)」與「障礙物 (X)」
- 計算每個格子的「價值函數 $V(s)$ 」與「最佳策略 (Optimal Policy)」
- 動畫化展示代理人從起點依照策略走到終點的過程
- 同時呈現四個視覺化網格，輔助學習與理解

Gridworld 是強化學習中經典的範例，透過離散化空間模擬智慧體在環境中做出最佳決策的過程。

2. 解法方法與技術架構

2.1 價值迭代演算法 (Value Iteration)

本系統使用 Value Iteration 來計算每個狀態的最大期望回報：

- 每個非終點狀態給予固定負回報（例如 -1），鼓勵智慧體快速抵達終點。
- 當前狀態 s 選擇一個動作 a ，到達下個狀態 s' 。
- 終點狀態的 $V(s)$ 固定為 0。

2.2 使用者互動與 UI

使用者可透過點擊格子來設定：

- ☒ 起點 (S)：顯示為綠色
- ☒ 終點 (E)：顯示為紅色
- ☒ 障礙物 (X)：顯示為灰色，最多設置 $n-2$ 個

選完後會即時觸發後端運算，前端則更新策略與動畫。

2.3 技術架構說明

/ 專案根目錄

—— app.py	# Flask 後端，處理 API 與價值計算
—— templates/	
—— index.html	# 前端 HTML 主頁，包含互動邏輯與動畫
—— static/	
—— styles.css	# 前端樣式檔案

3. 視覺化呈現內容

本專案提供「四個視覺化網格」供使用者理解強化學習決策流程：

網格編號	類型	功能說明
①	初始網格 (Initial Grid)	讓使用者設定起點 S、終點 E 與障礙物 X
②	價值函數網格 (Value Function)	顯示每個格子的 $V(s)$ 數值，反映從該格出發的期望回報
③	最佳策略網格 (Optimal Policy)	以箭頭顯示每個格子的最佳行動方向 ($\uparrow \downarrow \leftarrow \rightarrow$)
④	最佳策略動畫 (Optimal Path Animation)	動畫展示代理人依照策略從 S 移動到 E 的過程

4. 執行流程與互動說明

1. 選擇 Grid 大小 ($5 \times 5 \sim 9 \times 9$)

2. 使用者透過點擊設定起點、終點與障礙物
 3. 系統即時呼叫 Flask API，執行 Value Iteration
 4. 前端更新三種資訊：V(s)、Policy、動畫路徑
 5. 使用者可點擊「播放最佳策略動畫」觀看代理人決策過程
-

5. 結果展示與優勢

- ✓ 每次設定都能即時刷新價值與策略結果，操作直觀
 - ✓ 四個網格清楚對應四種資訊，有助於理解強化學習決策邏輯
 - ✓ 最佳策略動畫具體展示智慧體行為，強化概念學習
-

6. 延伸改進方向

- 加入 Q-learning / SARSA 等線上學習方法
 - 支援動態回報設定（不同格子不同 R 值）
 - 加入「探索與利用」機制，模擬不確定性
 - 增加牆壁、陷阱等更多地圖元素，提高複雜度
-

7. 結語

本專案結合「強化學習理論」與「Web 前後端整合」，以互動方式呈現 Value Iteration 在 Gridworld 中的應用。透過四種網格視覺化與動畫模擬，不僅提升了學習效率，也增進了對 AI 決策過程的理解。