

Unity3D 結合 AI 實務

專題編號：113-CSIE-S020

執行期限：112 年第 1 學期至 113 年第 1 學期

指導教授：劉傳銘

專題參與人員：110590008 韓正勤

110590038 呂凱達

110590039 張高偉

一、摘要

本計畫為在 PC 平台上使用 Unity3D[1] 遊戲引擎製作一款人工智慧結合戰爭模擬的策略遊戲，其中士兵以強化學習進行訓練，我們研究如何調整獎勵、參數、方法，來優化智能體的行為。

關鍵字：ML-Agent、MA-POCA、self-play、Raycast。

二、緣由與目的

近年來人工智慧愈來愈普遍，應用也越來越簡單，但是 AI 在電子遊戲中還沒有太多的發展，且傳統的遊戲智能體開發往往依賴於程式邏輯，導致在應對動態遊戲環境時缺乏靈活性和適應性。隨著遊戲複雜性的增加，遊戲開發者面臨著智能體行為難以透過程式邏輯去實現的挑戰，因此我們想嘗試 AI 在遊戲方面的應用。

遊戲玩法方面，基本上參考於市面上的許多策略遊戲，包括戰地風雲、全面戰爭模擬器等，結合以上玩法，再加入關卡和 AI 士兵。

此外，我們還希望為戰爭模擬遊戲的 AI 開發提供一些建議，所以做了一項比較疏密的研究，並為未來更複雜的 AI driven 遊戲奠定基礎[3]。

三、研究範圍

本研究主要聚焦於智能體在 Unity 環境中的行為模式與優化。

此次專案透過強化學習使智能體可

以在多樣化環境下進行有效的決策和學習機制，我們將探索智能體如何在複雜的遊戲場景中，利用環境反饋不斷優化其行為策略，進而實現更加複雜的任務。

關於疏密研究的部分，我們評估 MA-POCA 演算法在稀疏和密集獎勵環境中的表現差異[3]，理解不同獎勵結構如何影響 AI 的學習過程和行為。

四、使用技術方法

本專案採用 ML-Agents[2] 和 MA-POCA 演算法、Unity 等技術，以提升智能體的行為表現，以下是本專案的技術細節。

(一) ML-Agents

ML-Agents 是 Unity 提供的一個開源工具包，我們使用它建立智能體，以便在各種遊戲環境中進行自動化的學習和決策。我們考慮給予智能體獎勵的參數有攻擊獎勵、受傷懲罰，空揮懲罰等等，各個職業都有相對應的獎勵倍率。

此外，有個參數會隨著訓練時間不斷變化，它會與我們各個獎勵相乘，隨著時間的推移，個人獎勵會越來越少、團隊的獎勵會越來越多。這樣的獎勵機制促使智能體團隊合作，激勵它們不僅僅依賴個人的獎勵，還需學會評估如何讓團隊的得利益最大化，我們 model 的 input 有視野、血量、冷卻時間等訊息；output 有移動、旋轉、攻擊等動作。

(二) MA-POCA

MA-POCA 是一個 ML-Agents 提供的一種演算法，這種演算法採用了集中式學習與分散式執行。一個中心裁判（即

一個神經網路)負責評估所有代理的狀態，進而對其表現進行打分，我們透過修改 yaml 檔中的參數來進行調整，其中比較重要的有 learning_rate_schedule 、hidden_units 、normalize 等等。

在疏密獎勵的研究中[3]，我們比較 1v1 、3v3 和 5v5 的不同團隊規模的場景，評估了 MA-POCA 的表現。我們分析了累積獎勵、ELO 評分、Episode 長度和 Entropy 等關鍵指標，全面評估演算法性能。

(三) Unity

Unity 是一款遊戲引擎，能幫助我們實現遊戲中的物理模擬和邏輯運算。我們利用 Unity 提供的 RayCast 工具(射線技術)來獲取視野輸入。RayCast 能夠偵測環境中的訊息，例如物體之間的距離、方向或是否存在障礙物，並將這些偵測到的資訊傳遞給模型，作為模型進行決策的依據。

五、架構流程

圖 1 是我們訓練的架構流程，每當 episode 開始，會持續判斷是否其中一方全滅，如果沒有則會給予士兵中戰鬥時個人獎勵，當一方全滅時則會進入判斷這回合的勝負，依照勝負給予雙方團隊的獎勵，並進入下一回合。

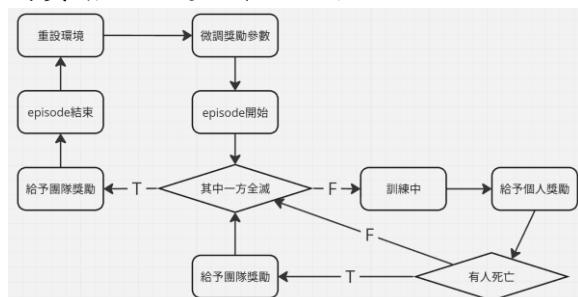


圖 1. 訓練架構

圖 2 是我們的遊戲流程，我們有為遊戲製作簡易的 UI 介面，方便玩家操作。

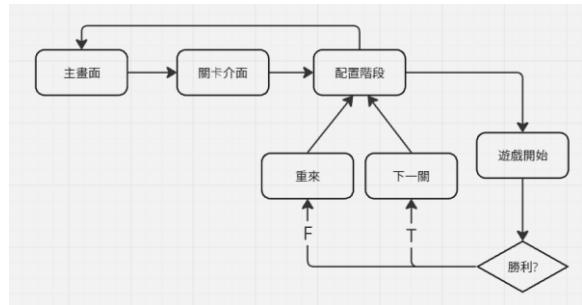


圖 2. 遊戲架構

六、工具說明

Unity3D: 遊戲引擎

ML-Agents: Unity 的 ML 插件

Visual Studio: Unity 指定 IDE

Git: 版本控制、開發

Tensorboard: 訓練數據分析

七、實驗結果

(一) 最後訓練出來的成果有符合觀賞上與數據上的預期

(二) 最後訓練出來的士兵模型能夠良好的適應遊戲中的環境

(三) 產出有關疏密性獎勵的研究論文

八、參考文獻

- [1] Unity 3D(2021.3) <https://docs.unity3d.com/ScriptReference> (viewed on the Oct, 2024)
- [2] ML Agent <https://github.com/Unity-Technologies/ml-agents> (viewed on the Oct, 2024)
- [3] Cheng-Chin Han, Kai-Ta Lu, Gao-Wei Chang “[A Comparative Study of MA-POCA in Sparse and Dense Reward on Battle Simulator Environments](#)”, Unpublished, 2024-10-08