

FAI Final Project Report

B10705009 資工三邱一新

June 13, 2025

1 Methods I Have Tried

在這次 Final Project 之前，我本身對德州撲克完全沒有經驗，因此是從零開始學習與嘗試策略的。初期我採用最簡單的 Rule-Based 方法，逐步引入 Monte Carlo Simulation 與 Probability Player 等方法。以下是我嘗試過的方法說明：

1.1 Monte Carlo Simulation

要評估手牌的強度，理想上可以對所有可能的未來公牌與對手手牌進行窮舉，但在德州撲克中這樣的組合數非常大，在 Preflop 時的所有組合就有 $C_2^{50} \times C_5^{48} = 1225 \times 1712304 = 2097572400$ 種，因此不切實際。為了克服這個問題，我使用了 Monte Carlo Simulation：

每次決策時，我會進行 2000 次模擬，每次從剩餘的牌堆中隨機抽出對手的手牌與未翻開的公牌，然後比較雙方的手牌，計算我方獲勝的比例。這個比例就作為目前手牌的勝率估計值，作為後續決策的依據。

1.2 Decision Agent

此策略是根據 Monte Carlo 模擬出的勝率以及 Pot Odds 進行決策。

$$\text{Pot Odds} = \frac{\text{Call Amount}}{\text{Pot Amount} + \text{Call Amount}}$$

根據撲克理論，若手牌的勝率高於 Pot Odds，則 Call 是期望值為正的行為。基於這個原則，我設計如下策略：

- 若勝率高於 0.65，則視為強牌並進行 Raise，且勝率越高 Raise 金額越大；
 - 若勝率高於 0.8，則 Raise 最大數量
 - 若勝率介於 0.75 ~ 0.8，則 Raise 最小數量加 100
 - 若勝率介於 0.7 ~ 0.75，則 Raise 最小數量加 50

– 若勝率介於 0.65 ~0.7，則 Raise 最小數量

- 其他情況則根據勝率與 Pot Odds 的比較，判斷是否 Call 或 Fold。

下圖為模擬 Preflop 階段各種起手牌的勝率分布，可以觀察到大多數手牌的勝率落在 0.5 附近，而勝率高於 0.65 的情況較為稀少，因此我將它作為 Raise 的標準。

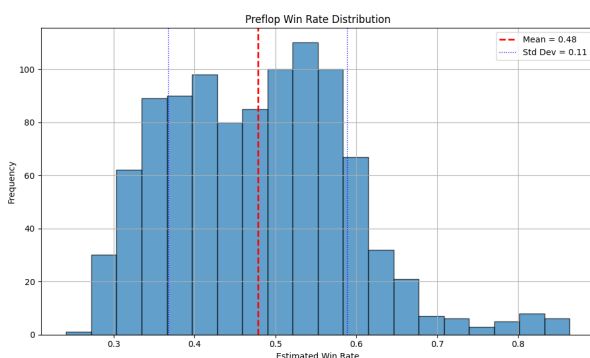


Figure 1: Preflop Win Rate Distribution

1.3 Probability Agent

在這個方法中，我不僅考慮手牌的勝率，更進一步評估每種行動對 Stack 數量的期望影響。使用以下方法計算期望 Stack 數量：

1. **Fold** : Current Stack Amount ;
2. **Call, Raise** : $\text{Win Rate} \times \text{Win Stack} + (1 - \text{Win Rate}) \times \text{Lose Stack}$;

針對 Raise 行動，我分成多種 Raise 金額進行模擬，並在對手會 Call 的假設之下估計 Win Stack。選擇上述期望值最高的行動當作最佳行動。

這個策略與 1.2 最大的差別是，它考慮了輸錢的潛在損失，因此在面對高額投入時會更謹慎，而不是單憑勝率決定行動，更能避免高風險情況下過度樂觀的判斷。

2 Tips I Have Tried

除了上述的方法，我也針對某些特殊情況設計策略，以提高獲勝率或避免不必要的風險。

2.1 Preflop Table

在 Preflop 階段，我參考網路上已經統整好的 **Preflop Table**，根據經驗法則將手牌進行強弱分類，根據起手牌的類型做行動。希望能讓 Preflop 的第一個行動更合理。

2.2 Detect Win Condition and Fold

因本專案為二人對戰 (Heads-Up) 且進行 20 局 (Rounds)，勝利條件為比賽結束時我方的籌碼數量高於對手。因此，若在某個時刻我方已經擁有遠高於初始籌碼的數量，即使之後每一局都棄牌 (Fold)，只要扣除大小盲的損失後仍可獲勝，即可視為「必勝狀態」。

我的方法是在做決策之前判斷現在的 Stack 是否為「必勝狀態」，若已經必勝，則我的 AI Player 總是 Fold，以零風險的方式結束比賽。

2.3 Compute Minimal Raise Amount

延續 2.2 的邏輯，我觀察到並不需要使用極大額度的 Raise 才能獲得籌碼優勢。若能算出「讓對手輸掉必要籌碼」的最低 Raise 數量，即可用最小風險達成目標。

因此，我算出達到「必勝狀態」所需的最少 Raise Amount 當作我的最大 Raise Amount，避免再投入過多籌碼承擔風險。

2.4 Bluff in River with Blocker

在進入 River 階段時，所有公牌皆已公開，雙方的手牌潛力幾乎已定。此時若我方手牌較弱，但能判斷對手較難擁有某些強牌組合，即可考慮利用「Blocker」進行 Bluff，藉此代表那些強牌並施壓對手棄牌。在我的實作中，Bluff 行為是 All In，不是上述提過較小的 Max Raise Amount，這樣做的目的是施壓對手。

我實作了以下幾種 Blocker 類型，用於輔助判斷何時適合 Bluff：

1. **Flush Blocker**：當我手上持有某花色的大牌，而公牌已有四張相同花色，代表我阻斷了對手湊成同花的可能性，可嘗試代表同花進行 Bluff。
2. **Straight Blocker**：若公牌出現四張連號牌，而我手中握有構成順子所需的邊張（如持有 T，在公共牌為 6-7-8-9 時），則有機會阻止對手擁有順子，可藉此代表順子進攻。
3. **Set Blocker**：當我持有與公共牌相同 Rank 的牌（例如場上有一張 8，我手上也有一張 8），則對手擁有三條的機率降低，可代表 Set 施壓對手。
4. **Overpair Blocker**：若我手中為大牌對子，且高於所有公牌（如持有 QQ，場上為 T-high），我可能阻止對手擁有更高對子或更強手牌，可假裝擁有 Overpair 進行進攻。
5. **Top Pair Blocker**：若我手上的一張牌與公牌的最高 Rank 相同（例如場上最大牌為 K，我手持 K），我降低了對手也持有 Top Pair 的可能性，有時可代表 Top Pair 進行 Bluff 或 Thin Value Bet。

我的 Agent 在做判斷時，除了有 Blocker 外，當手牌勝率低於 0.2 且 stack 小於 1000 時，我才會 Bluff，希望能起威嚇作用，讓對手 Fold，進而逆轉局勢。

3 Experiments

經由上述可知，加入 Detect Win Condition and Fold 和 Compute Minimal Raise Amount 這兩個方法不會讓勝率下降，因為它們只是排除了比較差的情況，所以我的每個 Agent 都有使用這兩種策略。

以下的實驗我比較 Decision Agent(DA) 和 Probability Agent (PA) 的優劣，以及加入 Bluff in River with Blocker(B) 和 Preflop Table(PT) 觀察其對勝率的影響。

我讓每個 Agent 跟 Baselines 進行 50 次對戰，每次對戰的最大輪數為 20，起始 Stack 為 1000，小盲為 5，和此專案的遊戲環境相同，測試出的勝率如下圖所示：

Table 1: Win Rate on Baseline For Different Method

Baseline	DA	PA	DA & PT	PA & PT	DA & PT & B	PA & PT & B
Baseline 1	0.94	0.98	0.90	0.92	0.76	0.88
Baseline 2	0.86	0.86	0.78	0.90	0.82	0.86
Baseline 3	0.64	0.98	0.66	0.76	0.7	0.82
Baseline 4	0.54	0.64	0.50	0.64	0.42	0.68
Baseline 5	0.38	0.62	0.38	0.32	0.3	0.42
Baseline 6	0.42	0.58	0.40	0.40	0.42	0.48
Baseline 7	0.30	0.50	0.20	0.44	0.28	0.44

根據以上實驗我發現：

- Probability Agent 綜合來看優於 Decision Agent
- Preflop Table 的幫助有限，甚至拉低勝率
- Bluff 對某些 Baseline 雖有小幅度進步，但整體還是更不穩定

我最後決定使用的 Agent 為未使用 Preflop Table 和 Bluff 的純 Probability Agent。

4 Conclusion And Discussion

整體而言，以下幾點為本次實作的發現與討論：

- **期望值決策架構具較高穩定性**：實驗中可以觀察到，Probability Agent 在各 Baseline 上的勝率普遍高於 Decision Agent，顯示考慮風險的決策方式確實更穩定、實用。
- **Preflop Table 效果不如預期**：儘管加入經驗法則所設計的起手牌表應能幫助前期決策，但實驗中反而在部分對局中拖累了表現，可能原因在於 table 設計與 Agent 的其它策略相性不合。
- **Blocker Bluff 風險高**：雖然理論上有機會反敗為勝，但在勝率偏低時過於激進的 Bluff 有可能導致額外損失，若能針對對手行為進一步調整觸發條件，可以降低不必要的 Bluff 機率。