Multi-Agent Reinforcement Learning on Stochastic Game

Cortinovis Nicola, Lucas Marta
June 29, 2025

Introduction

This report focuses on the implementation of the simplified football game introduced by Michael L. Littman [1], modeled as a zero-sum stochastic game. The learning algorithm used to determine the agents' behavior is the belief-based joint action learner. All parameters are kept consistent with those used in the original paper. We train an agent that applies the belief-based update rule both against a random opponent and against another belief-based agent. Our GitHub code implementation is available at [2].

1 Problem setting

In this simplified football game, two agents, A and B, compete on a 4×5 grid that represents the playing field. Each side has a goal post located just outside the grid. Ball possession is indicated by a circle and is assigned randomly at the start of each game, while the starting positions of the agents are fixed and symmetrical. Five moves are generally allowed: North, South, East, West and Stand. In our implementation, moves that would take the agent outside the grid are considered illegal, except when such moves would result in scoring. If a player attempts to move into a square occupied by the opponent, the move is blocked and possession transfers to the opponent. When both players try to move into each other's square simultaneously, neither moves; possession is resolved based on the action execution order. A player wins by reaching the opponent's goal area while in possession of the ball, receiving a reward of $+1$, while the opponent receives -1. The game then resets to the initial configuration, shown in the figure below.

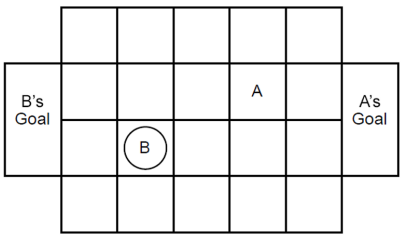

Figure 1: Simplified soccer game, initial state

2 Belief-based joint action learning algorithm

Belief-based joint action learning equips an agent with an internal model of its opponent's strategy, allowing it to operate in a multi-agent setting. For each state s, the agent maintains a belief distribution $B(s)$ over the opponent's possible actions, updated empirically from observed behavior. The agent maintains a joint-action Q-table $Q(s, a, o)$, where a is its own action and o is the opponent's. Action selection is guided by the expected utility under the current belief:

$$V(s') = \max_{a \in Act} \sum_{o \in Act_{\text{opp}}} B(s')(o) \cdot Q(s', a, o)$$

Upon observing a transition, the Q-value is updated via temporal-difference learning:

$$Q_{t+1}(s, a, o) \leftarrow (1 - \alpha) Q_t(s, a, o) + \alpha \left[r + \gamma V(s') \right]$$

By continually refining both its value estimates and its belief model of the opponent, the agent learns a policy that best responds to the opponent's observed behavior.

3 Experiments

We perform two experiments: first, we train a belief-based agent against a random opponent, and then against another belief-based agent. We adopt the following parameters:

$$\gamma = 0.9 \qquad \varepsilon = 0.2$$
$$\alpha_0 = 1 \qquad \alpha_{t+1} = \alpha_t \cdot 10^{\log_{10}\left(\frac{0.01}{10^6}\right)}$$

Both training phases are conducted over 10^6 steps, while the testing phase is limited to 10^5 steps. During testing, in each game step there is a probability $1 - \gamma = 0.1$ of terminating prematurely in a draw (0 reward for both players) to simulate the effect of the discounting factor.

4 Results

Let RA be a random agent and BBA be a belief based agent. To indicate its training opponent we use BBA(r) for random agent and BBA(b) for belief based agent. We tested the agents in the following scenarios:

$$BBA(r) \text{ vs } RA \qquad BBA(b) \text{ vs } BBA(b)$$
$$BBA(r) \text{ vs } BBA(b) \qquad RA \text{ vs } BBA(b)$$

To mitigate the effect of randomness in the game dynamics, we performed five independent training and testing runs. The results reported in the following tables represent the average across the testing runs.

Experiment A vs B	A Wins (%)	B Wins (%)	Draw (%)
BBA(r) vs RA	46.04	0.35	53.81
BBA(b) vs BBA(b)	29.33	30.03	40.23
BBA(r) vs BBA(b)	10.17	40.82	49.01
RA vs BBA(b)	0.42	45.06	54.52

Table 1: Win rates and draw percentages

Experiment A vs B	Games	Avg length	Std length
BBA(r) vs RA	17,334	5.77	4.53
BBA(b) vs BBA(b)	20,945	4.78	2.68
BBA(r) vs BBA(b)	20,886	5.11	3.83
RA vs BBA(b)	16,914	5.92	4.70

Table 2: Average number of games and their length

5 Conclusions

Our findings validate that agents based on belief can effectively learn policies within multi-agent settings via joint action learning. When pitted against a random agent (RA), the belief-based agent (BBA(r)) recorded a significantly elevated win rate, showcasing its capacity to outclass a simpler agent. When a belief-based agent (BBA(b)) is matched against another BBA(b), the win rates became more equitable, accompanied by a marked rise in draw results—underscoring the development of competitive and symmetric strategies. As expected, BBA(r) exhibits subpar performance against BBA(b), indicating that strategies acquired against less formidable opponents do not transfer effectively to more tactical challengers. Finally, the RA is consistently outperformed by BBA(b), securing nearly no victories. In summary, these results confirm the significance of integrating belief modeling into reinforcement learning agents within competitive environments.

References

[1] Markov games as a framework for multi-agent reinforcement learning, Michael L. Littman, 1994

[2] gitub.com/NicolaCortinovis/MAS_2025