Belief-Aware Multi-Agent RL Agent for Partially Observable Battle Arena Game

Hanzhang Yin

Dec 13, 2025

1 Introduction

Battle Arena is an $n \times n$ partially observable multi-agent grid combat game with latent agent strengths and global combat broadcasts. We develop CUSTOM5, a belief- and state-abstraction-based agent with bias–risk-shaped tabular Q-learning, and evaluate it in mixed-policy and ablation settings with careful analysis.

2 Uncertainty in the Environment

We treat everything outside our chosen agent (other agents, the grid, and the game rules) as part of the environment.

Outcome uncertainty. Even with complete information, transitions are stochastic because moves are simultaneous and collisions depend on other agents' actions; tie-breaking and the weakest-versus-strongest exception add additional randomness.

State (partial observability). Each agent observes only a local radius-2 window; opponents outside this window are unobserved. While the pre-/post-combat broadcast layers provide global information about *which* IDs fought and survived—and implicitly the *locations of combats* on the observation grid—they do not reveal the current positions of all agents. Thus, the full board configuration remains only partially observed.

Model uncertainty. Key parameters of the environment are latent: strengths (including our own) are never observed directly and must be inferred from repeated combats. This creates uncertainty about win probabilities and therefore about the effective transition dynamics.

Interaction uncertainty. Opponents follow unknown policies (and may themselves learn), so the environment from our agent's perspective can be non-stationary. The effective transition kernel depends on the evolving behaviors of other agents, not just on the game rules.

3 Method

3.1 Problem Formulation

From the learner's point of view the game is a partially observable Markov decision process (POMDP). The hidden state s_t encodes the full grid (positions and alive flags for all agents), the latent strength permutation, and the remaining horizon. At each timestep the learner receives an observation o_t consisting of: (i) a map layer with IDs and walls inside a radius-2 field of view, and (ii) two global broadcast layers that mark which IDs took part in combat (pre-combat) and which survived (post-combat).

Actions $a_t \in \{\text{up}, \text{right}, \text{down}, \text{left}\}$ are chosen simultaneously by all agents; transitions implement the movement and combat rules, including the weakest-beats-strongest exception. Rewards combine survival and kill bonuses, with weaker agents receiving more reward per timestep. The objective is to maximize the **expected discounted return** $\mathbb{E}[\sum_{t=0}^{T} \gamma^t r_t]$. Because strengths and many opponents are hidden, the policy must act on beliefs over latent variables summarised into a compact state representation.

3.2 Custom5 Policy

CUSTOM5[1] augments tabular Q-learning with three environment-specific components: (i) combat-based strength beliefs from the broadcast layers, (ii) a relational state abstraction focused on one target enemy, and (iii) a bias–risk scoring rule that shapes Q-values into actions.

[1]NOTE: Design iterations and lessons learned are in Appendix C.

3.2.1 Combat-Based Strength Beliefs

Global and pairwise posteriors. From the broadcast layers we recover three ID sets at each step: participants pre_ids, survivors $post\_ids$, and $dead\_ids = pre\_ids \setminus post\_ids$. For every ID i we maintain global win/loss counts (w_i, ℓ_i) and update

$$w_i \leftarrow w_i + \mathbf{1}\{i \in post\_ids\}, \qquad \ell_i \leftarrow \ell_i + \mathbf{1}\{i \in dead\_ids\}.$$

We interpret these as Beta–Bernoulli posteriors with prior $\text{Beta}(1, 1)$,

$$p_i = \frac{1 + w_i}{2 + w_i + \ell_i},$$

a smoothed estimate of the long-run probability that i wins random fights.

When there is exactly one 1-v-1 combat (two IDs in pre_ids, one in $post\_ids$), we also update a pairwise win table W_{ij}, incrementing W_{ij} when i defeats j. For pairs with at least three observed fights we estimate

$$P(i \text{ beats } j) = \frac{1 + W_{ij}}{2 + W_{ij} + W_{ji}},$$

which design to hopefully better captures the weakest-beats-strongest rule than marginal win-rates alone.

Strength buckets and hotspot. For the learner we compute coarse *strength buckets* from its empirical win-rate (weak / medium / strong), used to scale aggressiveness and exploration. We also track a moving average of recent combat locations from the pre-combat layer and treat the resulting centroid as a "hotspot"; when strong and "hungry" (many steps since last combat), the agent is biased toward this hotspot if no enemies are visible.

3.2.2 Relational State Abstraction

The raw observation tensor is too large for tabular RL, so CUSTOM5 maps each observation to a discrete state s.

We first locate the learner by searching for its ID in the map layer and, if necessary, falling back to the post-combat layer when the map was overwritten by combat. If the learner is not found, we use a terminal state $s = \texttt{"dead"}$.

Otherwise, let (r, c) be the learner's position. We compute a *centre bucket* from the Manhattan distance to the board centre and a *self-strength bucket* from its win-rate. We then build an enemy mask by taking all IDs in the map layer, removing the learner and any IDs in $dead\_ids$ to filter out "ghost" enemies.

If no enemies are visible, we use

$$s = (\texttt{"no\_enemy"}, \texttt{center}, \texttt{self\_bucket}).$$

If at least one enemy is visible, we select a single *target* and encode its relation to the learner. For each candidate enemy j at (r_j, c_j) we compute the Manhattan distance d_j, a strength gap

$$\Delta_j = \begin{cases} 2\big(P(\text{self beats } j) - 0.5\big) & \text{if enough pairwise data,} \\ p_{\text{self}} - p_j & \text{otherwise,} \end{cases}$$

and an exploration bonus $\text{UCB}_j \propto 1/\sqrt{\#\text{ fights involving } j}$. The chosen target maximizes $\Delta_j + \text{UCB}_j - \lambda_d d_j$ for a fixed distance penalty $\lambda_d > 0$.

Let $(r^*, c^*, \Delta^*, d^*)$ denote the chosen target. We derive:

$$\texttt{row\_dir} = \text{sign}(r^* - r), \qquad \texttt{col\_dir} = \text{sign}(c^* - c),$$

$$\texttt{dist\_bucket} = \begin{cases} 0 & d^* \le 1, \\ 1 & 1 < d^* \le 2, \\ 2 & d^* > 2, \end{cases} \qquad \texttt{relation} = \begin{cases} 1 & \Delta^* > \tau, \\ -1 & \Delta^* < -\tau, \\ 0 & |\Delta^*| \le \tau, \end{cases}$$

for a small margin $\tau > 0$. We also bucket the number of visible enemies into $\{1, 2, \ge 3\}$.

The final state is

$$s = (\texttt{row\_dir}, \texttt{col\_dir}, \texttt{dist\_bucket}, \texttt{relation}, \texttt{center}, \texttt{enemy\_count}, \texttt{self\_bucket}).$$

This relational encoding should preserve local geometry and relative strength while keeping the state space small enough for tabular methods.

3.2.3 Learning and Action Selection

Tabular Q-learning. We maintain a tabular $Q(s, a)$ over all states and four actions. After each step we update

$$Q(s_t, a_t) \leftarrow Q(s_t, a_t) + \alpha\Big(r_t + \gamma \max_{a'} Q(s_{t+1}, a') - Q(s_t, a_t)\Big),$$

with discount factor $\gamma = 0.985$ and learning rate $\alpha = 0.05$. If $s_{t+1} =$ `"dead"` we drop the bootstrap term.

Bias–risk scoring. To turn Q into actions we compute, for each legal action a,

$$\text{score}(s, a) = \lambda_b(s)\,\text{bias}(s, a) - \lambda_r(s)\,\text{risk}(s, a) + \lambda_Q\big(Q(s, a) - \bar{Q}(s)\big),$$

where $\bar{Q}(s) = \frac{1}{4}\sum_{a'} Q(s, a')$ and $\lambda_Q = 0.25$.

The *bias* term encodes preferences:

- With no enemies visible, strong and "hungry" agents bias toward the hotspot or the centre; weak agents bias slightly outward.
- With a target visible, we compute a scalar attack score from $(\Delta^*, \texttt{relation}, \texttt{dist\_bucket}, \texttt{enemy\_count})$ and turn it into directional preferences either to move closer (chase) or further (flee).

The *risk* term penalizes:

- illegal actions (stepping into walls or off the board);
- stepping directly onto a visible enemy;
- ending adjacent to multiple enemies when not clearly strongest; and
- moving closer to a cluster of enemies even when currently strongest.

Both bias and risk are scaled by functions of the agent's global/local strength buckets and a hunger variable h (steps since last combat), so strong, hungry agents are more willing to take risks than weak ones.

Finally we act ϵ-greedily with respect to the scores, restricting to legal moves. The exploration rate ϵ decays with the episode index and is further modulated by the global strength bucket, but is always clipped to $[0.05, 0.40]$.

4 Experiments and Results

4.1 Experimental Setup

We report results in three settings. In all settings, each *session* is a single long run of the environment under a fixed random seed, and we summarize performance using only the *tail* of the run to reduce transient effects.

(1) General mixed-policy evaluation. We evaluate one instance of CUSTOM5 together with the five provided baseline policies (Aggressive, Coward, Random, Scanner, Zombie). We run 20 independent sessions with seeds 0–19. Each session contains 800 episodes. For each policy, we compute average return and win rate using only the last 200 episodes (i.e., episodes 600–799) within that session, and then aggregate across all sessions.

(2) Mixed-policy ablation To isolate the contribution of key components of CUSTOM5, we compare four variants: CUSTOM5, CUSTOM5-NOQ, CUSTOM5-NOHEURISTIC, and CUSTOM5-NOBELIEF. We run 3 seeds $(3, 12, 42)$. For each seed, we run one session per variant (thus 3×4 sessions total). Each session contains 1000 episodes, and we compute metrics using only the last 400 episodes (episodes 600–999).

(3) Custom-only arena. We further evaluate the four custom variants against each other without baselines: CUSTOM5, CUSTOM5-NOQ, CUSTOM5-NOHEURISTIC, and CUSTOM5-NOBELIEF. We run 5 independent sessions with seeds $(1, 2, 3, 4, 42)$. Each session contains 1000 episodes, and we compute metrics using the last 400 episodes.

Metrics. Return is the cumulative episodic reward defined by the environment, and win rate is the fraction of episodes in which the policy is the last surviving agent. We report per-policy means over the evaluation windows described above, and when multiple seeds/sessions are used we aggregate across them. We also report average episode length (steps per episode) computed over the same evaluation windows.

Policy	Strength S	Avg. return	Win rate (%)
CUSTOM5	overall	6.14	69.8
	1 (weakest)	9.40	63.4
	2	4.21	66.0
	3	3.19	63.2
	4	4.54	58.7
	5	7.77	89.4
	6 (strongest)	10.84	91.5
Aggressive	overall	3.30	23.2
Coward	overall	3.48	50.8
Random	overall	2.58	22.5
Scanner	overall	3.33	19.7
Zombie	overall	3.77	57.7

Table 1: Mixed-policy environment: overall metrics and a breakdown of CUSTOM5 by strength rank S (1 = weakest).

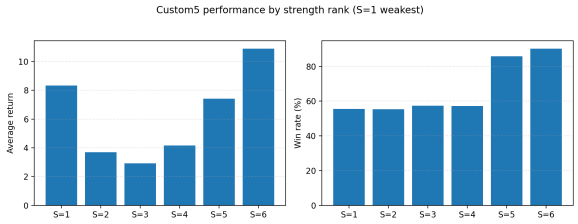
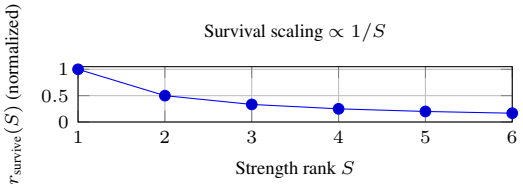


Figure 1: CUSTOM5 performance by strength rank. Returns are U-shaped: $S = 1$ is boosted by inverse-strength survival scaling, while $S = 6$ benefits from frequent wins and survival.



4.2 Overall Performance vs Baselines

Table 1 summarizes the mixed-policy evaluation and reports both overall metrics and a breakdown of CUSTOM5 by strength rank S. Figure 1 visualizes the same breakdown. Average return is *U-shaped* across strength: performance is strongest at the extremes ($S = 1$ and $S \in \{5, 6\}$), while mid-strength ranks are lower.

A small part of this pattern is induced by the environment's reward scaling, where the per-step survival reward is inversely proportional to strength,

$$r_{\text{survive}}(S) \propto \frac{1}{S} \cdot \mathbf{1}\{\text{alive}\}.$$

More importantly, the breakdown highlights the behavior of CUSTOM5 across regimes: when weak, it relies on conservative risk-aware movement and the combat-broadcast belief model to avoid unfavorable engagements; when strong, it becomes decisively exploitative via targeted chasing and hotspot-driven navigation. Overall, the results suggest that combining belief-based strength estimation with a compact relational state and bias–risk-shaped action scoring yields robust performance across the full strength spectrum.

4.3 Ablation: Mixed-Policy Environment

We next examine the ablation experiment where each custom policy variant plays against the five baselines with strengths re-sampled each session. We run three seeds (3, 12, 42), each with 1000 episodes per variant, and report averages over the last 400 episodes across all seeds. Table 2 shows the global summary.

Policy	Avg. return	Win rate (%)
CUSTOM5	9.60	72.4
CUSTOM5-NOQ	9.35	72.7
CUSTOM5-NOHEURISTIC	7.04	73.3
CUSTOM5-NOBELIEF	6.21	78.1
Best baseline (Zombie)	3.79	51.5

Table 2: Ablation in the mixed-policy environment; results averaged over three seeds with varying strength assignments.

All four custom variants substantially outperform the baselines in average return. The full CUSTOM5 agent and the heuristic-only variant (CUSTOM5-NOQ) achieve similar returns (≈ 9.5), indicating that geometry-based bias/risk rules carry most of the performance. Removing heuristics (CUSTOM5-NOHEURISTIC) causes a noticeable drop in return to 7.04, while removing beliefs (CUSTOM5-NOBELIEF) reduces return further to 6.21. Interestingly, win rates are comparable across variants (72–78%), suggesting that the primary benefit of Q-learning and beliefs is in accumulating higher rewards rather than simply surviving longer.

4.4 Ablation: Custom-Only Arena

To better understand the interaction between learning, heuristics, and beliefs when strength is not always maximal, we let only the four custom agents play against each other with strengths $S \in \{1, \dots, 4\}$ re-sampled each session. The global results appear in Table 3.

Policy	Avg. return	Win rate (%)
CUSTOM5	3.33	92.6
CUSTOM5-NOQ	2.82	78.0
CUSTOM5-NOHEURISTIC	2.61	60.2
CUSTOM5-NOBELIEF	1.71	75.5

Table 3: Ablation in the custom-only arena; strengths range over $S = 1, \dots, 4$.

Here differences between the variants are clearer. CUSTOM5 substantially outperforms the others, with about 18% higher average return than CUSTOM5-NOQ and a win rate roughly 15 percentage points higher. Removing heuristics hurts the most: CUSTOM5-NOHEURISTIC has the lowest win rate and similar return to CUSTOM5-NOQ, showing that bias/risk geometry is crucial when the agent is not guaranteed to be strongest. Removing beliefs (CUSTOM5-NOBELIEF) gives respectable win rates but the lowest return, suggesting that belief information mainly helps convert favourable positions into extra kills and survival reward.

Conditioning on strength rank confirms this picture. When $S = 1$ (weakest), CUSTOM5 achieves average return 6.52 and win rate 97%, whereas CUSTOM5-NOHEURISTIC and CUSTOM5-NOBELIEF achieve returns around 3.7 and win rates 57% and 65%, respectively. When $S = 2$–4, the full agent remains consistently ahead but with narrower gaps, consistent with Q-learning and beliefs mainly providing robustness in unfavourable situations.

5 Discussion and "What If" Analysis

Role of learning vs heuristics. In the mixed-policy ablation environment, geometry-based heuristics carry most of the performance: CUSTOM5-NOQ achieves similar returns to the full agent. However, removing heuristics or beliefs consistently reduces returns while win rates remain comparable, suggesting that Q-learning and beliefs primarily help accumulate higher rewards rather than just survive. In the custom-only arena, where agents sometimes start weak, Q-learning and beliefs become more important: the full CUSTOM5 agent clearly outperforms its ablations, particularly when starting at low strengths. This matches the intuition that fixed heuristics can exploit strength, but learned values and beliefs help compensate for weakness.

Non-learning vs learning opponents. Against fixed baselines the environment is effectively stationary, and Q-learning behaves as a standard model-free method and typically stabilizes empirically under our state abstraction. When other agents also learn, the effective transition dynamics change over time. Our relatively high learning rate and persistent exploration allow CUSTOM5 to track these changes, though a more principled multi-agent RL method might adapt more quickly.

Fog-of-war. If the environment were fully observable there would be less need for beliefs and heuristics: one could plan directly on the full grid using model-based RL or MCTS. Our design deliberately throws away global information; under full observability this would be wasteful. Conversely, if the field of view were even smaller, the combat broadcasts and belief model would become even more critical because direct observations of enemies would be rare.

Episode length. If episodes were much longer in a sense, survival reward would dominate and agents would have more time to adapt to each other within a single game. Our current bias/risk rules might then need to be more conservative. If episodes were much shorter, the benefit of building persistent beliefs would shrink, and a simpler heuristic chaser might suffice.

Only battling copies of our own agent. If we only battle identical copies of CUSTOM5, the game becomes approximately symmetric. In this setting, strength beliefs provide less separation signal (opponents behave similarly), and performance would be driven more by stochastic symmetry breaking (early random encounters) and by how well the policy avoids mutual "coin-flip" collisions when strengths are close. We would expect longer, more cautious games at mid strengths, and faster eliminations when strength gaps are large.

6 Conclusion

We model Battle Arena as a complicated POMDP. CUSTOM5 combines combat-based strength beliefs, a compact relational state abstraction, and bias–risk-shaped tabular Q-learning, outperforming the provided baselines and remaining strong across strength ranks. Ablations suggest heuristics deliver most of the performance, while belief estimation and learning add robustness, especially when the agent is not already the strongest.

A Additional Referential Figures

The following figures provide additional details and visualizations of the results presented in the main *4.2 Overall Performance vs Baselines* experiment. They are included for reference and to offer a more comprehensive understanding of the performance and dynamics of our agent.

A.1 Overall summary

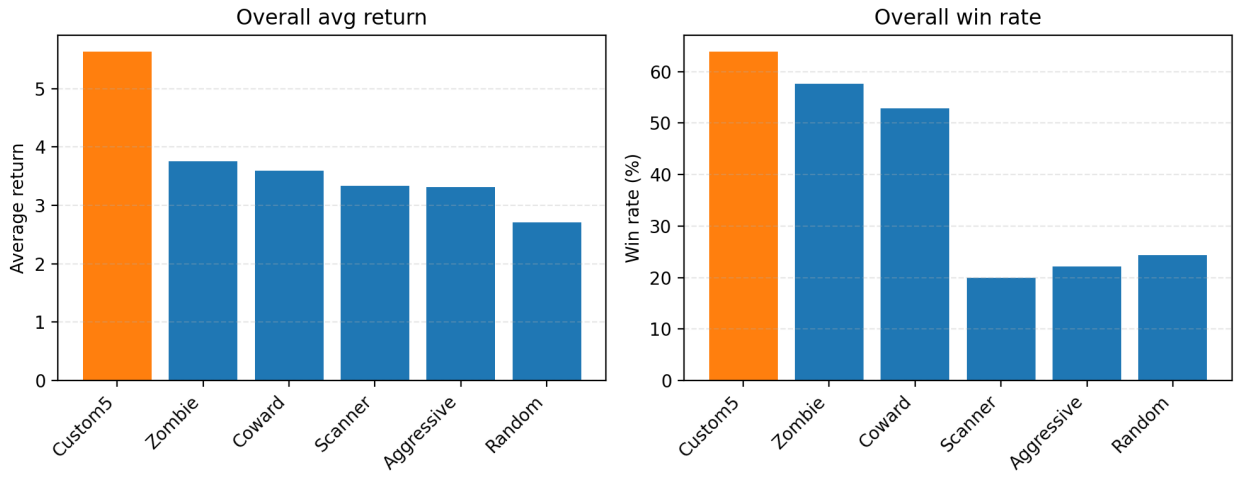


Figure 2: Overall average return and win rate across policies in the mixed-policy evaluation.

A.2 Performance by strength rank

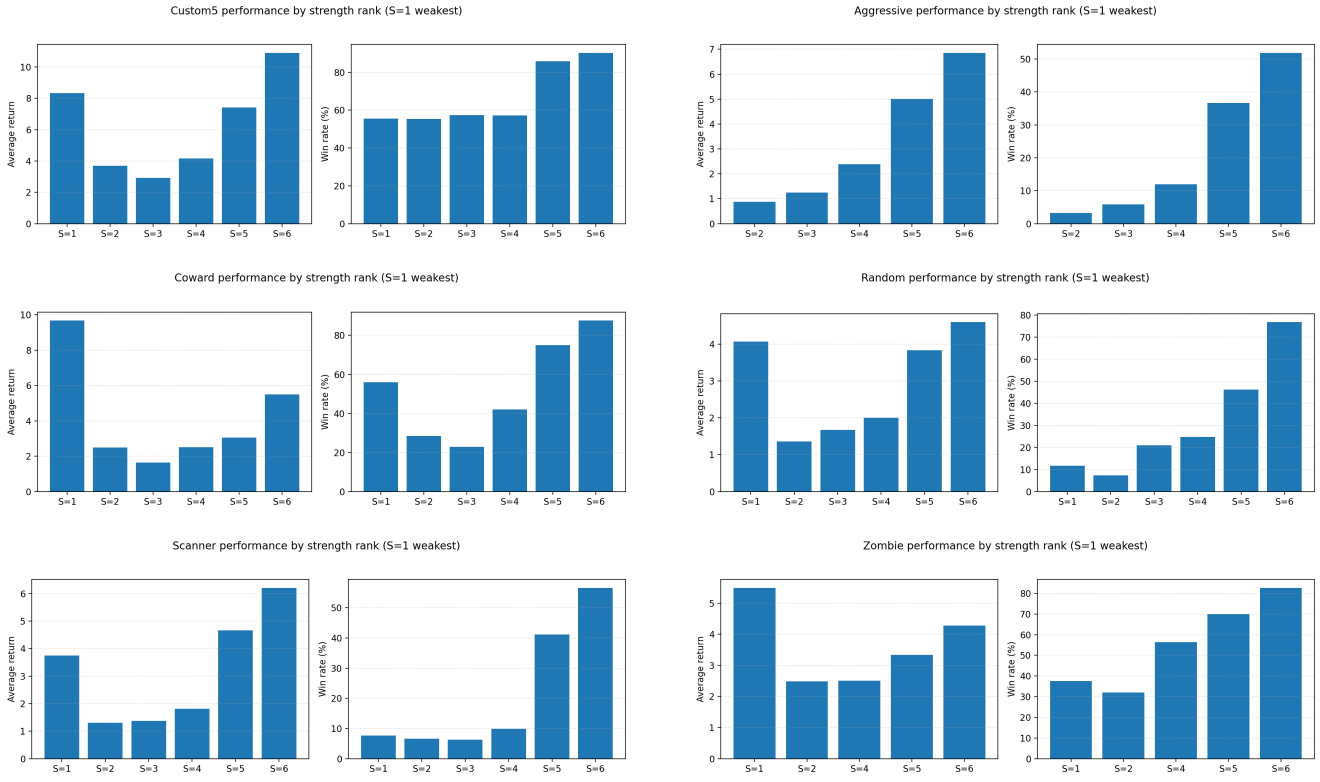


Figure 3: Average return and win rate broken down by starting strength rank ($S = 1$ weakest) for each policy.

A.3 Training dynamics

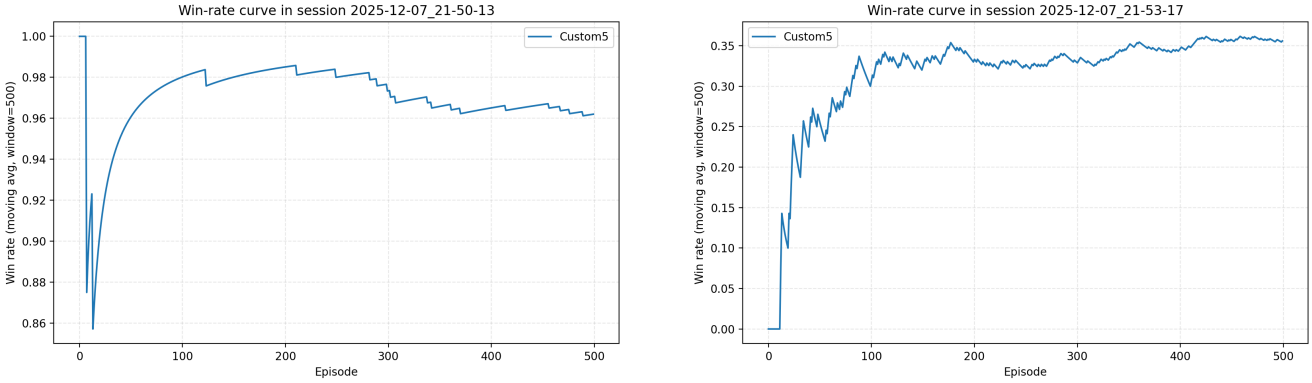


Figure 4: Representative win-rate learning curves (moving average; window size as shown in the plots) for CUSTOM5 across two sessions.

A.4 Episode length distribution

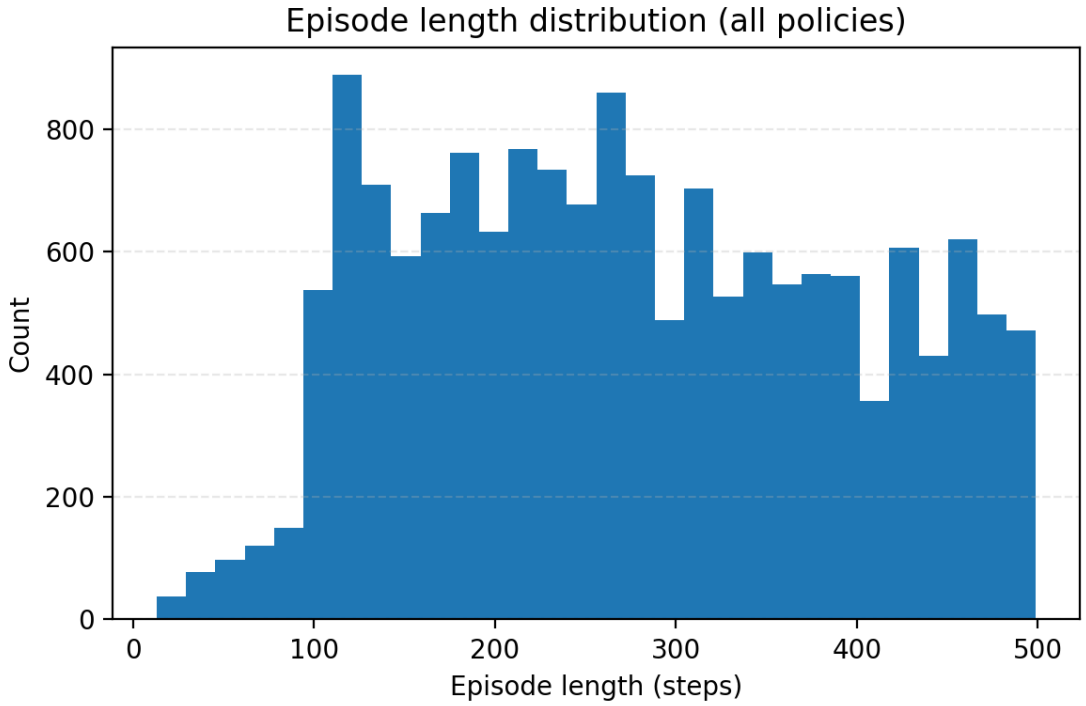


Figure 5: Histogram of episode lengths (steps) aggregated across policies.

B Interpretation of Additional Referential Figures

This section provides qualitative interpretation of the supplementary plots in Appendix A. The goal is to contextualize the main quantitative tables in the body and to highlight observable patterns that support the design choices in CUSTOM5.

B.1 Overall mixed-policy summary (Figure 2)

Figure 2 compares policies aggregated over the mixed-policy evaluation. Two trends stand out. First, CUSTOM5 achieves the highest average return and the highest win rate among all compared policies, indicating that it successfully balances survival and combat rewards under partial observability. Second, the best baseline (typically ZOMBIE) is competitive but consistently below CUSTOM5, suggesting that even a fixed observation-blind movement pattern can achieve reasonable performance when advantaged by strength.

B.2 Strength-conditioning effects (Figure 3)

Figure 3 breaks performance down by starting strength rank S. This breakdown is useful because it separates (a) effects due purely to the environment (e.g., the reward scaling with strength) from (b) effects due to policy behavior.

Custom5. The CUSTOM5 panel shows two qualitatively different regimes. When starting weak ($S = 1$–3), win rate is moderate but stable, indicating that the agent is able to avoid obviously losing engagements despite limited information. This is consistent with the design: the broadcast-driven belief model and risk penalties discourage suicidal moves, while the target selection prefers fights that are both nearby and plausibly favorable. When starting strong ($S = 5$–6), win rate increases sharply, consistent with the policy becoming more exploitative: chasing is triggered more often, hotspot navigation reduces search time, and risk penalties are relatively less restrictive when the agent is confident.

Baselines. The baseline panels illustrate why a belief- and risk-aware agent helps:

- AGGRESSIVE tends to perform poorly when weak but improves strongly with rank, reflecting that indiscriminate chasing is only safe when the agent is already advantaged.
- COWARD Notably, COWARD achieves 54% win rate at S=1, the second-highest among all policies, suggesting its survival-oriented approach is particularly effective for weak agents, consistent with prioritizing survival, but it does not reliably convert advantage into eliminations.
- RANDOM shows general improvement with rank (though not strictly monotone), but remains dominated by more structured policies.
- SCANNER is sensitive to rank and tends to underperform at low ranks, consistent with spending effort on information gathering without sufficiently strong mechanisms to exploit it.
- ZOMBIE is also sensitive to rank and tends to underperform at low ranks, consistent with its design that follows a fixed cyclic movement pattern without observing the environment, making combat encounters purely incidental.

Overall, the rank-conditioned plots support the interpretation that CUSTOM5's advantage is not only that it benefits from being strong, but that it also retains reasonable performance when weak by leveraging combat broadcasts (beliefs) and conservative risk-aware movement.

B.3 Training dynamics (Figure 4)

Figure 4 shows representative win-rate learning curves for CUSTOM5 (moving average). Both curves exhibit a common qualitative pattern: a rapid early increase followed by a plateau, indicating that the policy learns a stable set of preferences relatively quickly under the chosen state abstraction and shaping.

The curves differ substantially in their absolute levels, which is expected across sessions because (i) the mix of opponents and strength assignments changes the difficulty of the learning problem, and (ii) multi-agent interaction can make learning non-stationary (opponents' behaviors induce different effective transition dynamics). Importantly, the absence of long-term collapse suggests that the combination of (a) conservative risk penalties and (b) persistent exploration is sufficient to avoid catastrophic degradation, even when the environment is not strictly stationary.

B.4 Episode length distribution (Figure 5)

Figure 5 plots the distribution of episode lengths (steps) aggregated across policies. The distribution is broad, spanning from short episodes (early eliminations) to long episodes that approach the horizon. This shape is consistent with the environment's mechanics: some games end quickly due to early collisions, while others last much longer when agents avoid direct contact or when eliminations occur slowly.

From the perspective of CUSTOM5, this distribution is desirable rather than pathological: the policy does not systematically terminate episodes immediately (which would indicate overly risky behavior), nor does it systematically drag episodes to the maximum length (which could indicate excessive avoidance). Instead, it supports the interpretation that CUSTOM5 is capable of both survival-oriented play when uncertain/weak and decisive engagement when confident/strong, matching the intended bias–risk design.

C Design Iterations and Lessons Learned

We iterated through five policy designs (Custom $\rightarrow$ Custom2 $\rightarrow$ Custom3 $\rightarrow$ Custom4 $\rightarrow$ Custom5) before converging on the final agent. The early versions already leveraged the broadcast pre-/post-combat layers to maintain per-ID Beta win-rate posteriors and used a simple relational state around the nearest visible enemy (direction, distance bucket, and a coarse strength relation). However, the initial agent relied almost entirely on ϵ-greedy tabular Q-learning, which was slow to stabilize and often produced fragile behavior. Custom2 introduced explicit chase/retreat heuristics for "very close + clearly stronger/weaker" cases, which reduced catastrophic deaths but remained short-sighted. Custom3 expanded the state with an enemy-count bucket and added a flee heuristic for close multi-enemy situations, and also switched to globally shared tables for wins/losses/Q to accumulate information across episodes and agents; this improved robustness but still struggled with observation noise and inconsistent local geometry. Custom4 focused on practical stability: it avoided redundant belief updates via a broadcast signature, refined strength bucketing thresholds, and made the heuristics more conditional on global strength and opponent uncertainty (e.g., different engagement probabilities when data on an enemy is scarce). Finally, Custom5 unified these ideas into a belief–planning–control pipeline as described in *Method* (Section 3). Additionally, it also fixes two observation quirks mentioned in EdSTEM discussion, and introduces a "hunger" mechanism (steps since last combat) to modulate aggressiveness. Across iterations, the main lesson was that tabular RL alone is insufficient under partial observability; the broadcast signal (beliefs), compact relational abstraction, and safety shaping heuristics are essential for stable and competitive behavior.

D Acknowledgment

I want to share my deepest gratitude to Professor Littman, the course TAs, and classmates for feedback and support throughout the project and the journey of the course. Their guidance and encouragement have been invaluable.

E LLM Usage

I used ChatGPT for minor language editing and final proofreading on minor typos. All modeling, derivations, code, and experiments are my own.