Exposing Fragility in Multi-Agent Reinforcement Learning through Formal Specifications and Risk-Aware Analysis

Qingpei Li[1] and Xin Qin[2]

***Abstract*— Robust and adaptable behavior is critical for multi-agent reinforcement learning (MARL) systems deployed in dynamic and unpredictable environments. However, common evaluation practices, such as reporting the mean episodic reward, often fail to reveal coordination fragilities that can undermine reliability in practice. This paper introduces a lightweight evaluation framework that combines Signal Temporal Logic (STL) monitoring with Conditional Value-at-Risk (CVaR) analysis to expose coordination pathologies in MARL policies. Using the Steakhouse cooking simulator, we specify interpretable temporal properties of collaboration—such as fairness, activeness, and conflict resolution—and complement them with risk-sensitive performance metrics. Our experiments show that policies with similar average returns can diverge significantly in terms of robustness, role allocation, and tail-risk fragility. By bridging formal specification with empirical MARL evaluation, our framework contributes to ongoing efforts in benchmarking robustness, interpretability, and safety in multi-agent systems.**

I. INTRODUCTION

Multi-agent reinforcement learning (MARL) has become a promising approach for coordinating autonomous agents in shared, dynamic environments such as collaborative robotics, autonomous driving, and interactive games. However, for real-world deployment, MARL policies must not only be high-performing on average, but also robust, adaptable, and interpretable under diverse and unpredictable conditions [14], [3]. Conventional evaluation practices—dominated by mean episodic reward—fall short of capturing these requirements, as they often obscure coordination failures, unfair workload distributions, or catastrophic breakdowns in worst-case scenarios. To address this gap, we propose an evaluation framework that integrates formal specification and risk-aware metrics to provide a more complete picture of MARL robustness, including asymmetric role specialization, persistent conflicts, or brittle worst-case performance.

Recent MARL benchmarks largely emphasize sample efficiency and asymptotic averages [9] [14] while overlooking robustness and interpretability [3] [15], precisely where real systems fail: rarely, abruptly, and expensively. We address this gap by coupling temporal-logic monitoring, which exposes behavioral asymmetries, with risk-sensitive evaluation that quantifies the severity of tail failures. This alignment with verification and risk analysis moves the evaluation toward benchmarks that reflect a safety-critical deployment.

*This work was not supported by any organization

[1]Qingpei Li is with Department of Computer Science, University of Southern California, Los Angeles, CA, USA qingpeil@usc.edu

[2]Qin Xin is with the Department of Computer Science, California State University Long Beach, Long Beach, CA, USA xin.qin@csulb.edu

Our experiments systematically analyze these dimensions in a cooperative cooking environment [7] by introducing four key evaluation axes: (1) conflict behavior, measured through STL-monitored interaction patterns; (2) activeness, ensuring agents remain actively engaged; (3) robustness under worst-case outcomes; and (4) workload balance and role specialization, revealing emergent imbalances. In particular, we evaluate RL-RL pairs, where both agents are trained reinforcement learning policies, to examine the capabilities and limitations of fully learned coordination. We also examine greedy agents, which operate based on hand-coded heuristics. At each step, they select a medium-level action (e.g., picking up meat, washing a plate) and then choose a concrete motion toward that goal—optionally perturbed with Boltzmann noise. To avoid deadlocks, the agent detects no-progress states and executes a random unblocking move. This design maximizes immediate reward while offering a simple and interpretable baseline for cooperative behavior. For instance, we find that while RL-RL pairs achieve competitive average rewards, they exhibit high fragility and unintended role polarization, with one agent often relegated to low-reward sub-tasks. In contrast, greedy agents avoid conflicts, but suffer from inactivity.

These insights highlight the necessity of multi-faceted evaluation: Conditional Value at Risk (CVaR)[8], a metric that captures expected performance in the worst-case outcomes beyond a specified percentile, quantifies fragility under adverse conditions, while our STL-based evaluation uncovers latent asymmetries, both crucial for deploying agents in real-world settings where reliability and fairness are paramount.

II. RELATED WORK

Cooperative multi-agent reinforcement learning (MARL) has largely emphasized maximizing shared rewards under the centralized training and decentralized execution (CTDE) paradigm [9], [6]. Although this approach often improves mean task performance, it can mask important dimensions such as robustness and fairness: naive reward design may produce inactive or "lazy" agents and uneven reward distributions that degrade long-term cooperation [9], [15].

Formal methods have been explored as one remedy. Signal Temporal Logic (STL) provides an expressive way to formalize temporal and safety requirements, and has been used to diagnose and enforce behaviors in RL [1], [10]. More recently, several works have embedded STL directly into the training loop: specifications are converted into robustness-based rewards or used to construct safety

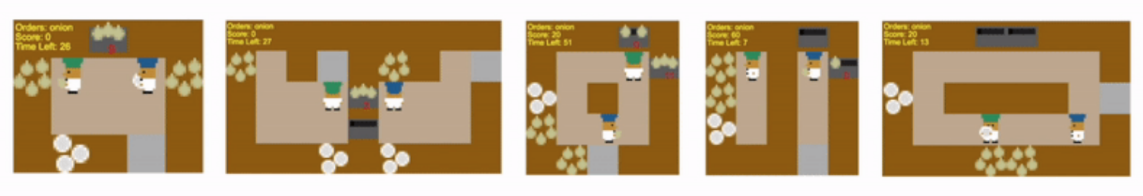


Fig. 1: Different map layouts in the Overcooked environment [4].

shields that constrain learning, improving both performance and safety during policy acquisition [13], [11], [5]. These training-time uses of STL contrast with approaches that treat logic as a post-hoc analytic tool.

Risk-sensitive evaluation offers another complementary perspective. Metrics such as Conditional Value-at-Risk (CVaR) highlight tail behavior and worst-case outcomes that average-return statistics miss, and have been shown useful for quantifying fragility in reinforcement learning systems [8]. Finally, emergent role specialization—where agents implicitly assume asymmetric responsibilities without explicit coordination—poses another challenge: specialization can boost efficiency, but also hide unfairness and brittleness from conventional metrics [2].

Our work brings these strands together by using STL not to shape training, but as a runtime, black-box monitoring instrument for coordination properties (conflict, activeness, fairness), and by coupling these logical checks with CVaR-based risk analysis to surface latent fragilities in MARL policies.

By bridging formal specification with risk-sensitive evaluation, we provide a diagnostic tool to reveal hidden fragilities essential for deploying MARL in safety-critical applications. We now demonstrate these ideas in the Steakhouse environment, a gridworld designed for studying cooperative cooking tasks.

III. THE STEAKHOUSE ENVIRONMENT

The Steakhouse environment [7] is a multi-agent gridworld simulator inspired by *Overcooked* [4], designed to study complex coordination in cooperative cooking tasks. Agents control chefs with six discrete actions (movement, interaction, or idle) and must collaborate to complete sequential subtasks (ingredient collection, cooking, and dish delivery) in variable kitchen layouts, receiving joint rewards for successful deliveries. This setup tests simultaneous strategy coordination, motion planning, and emergent role assignment without predefined roles or communication channels.

It provides an ideal testbed for analyzing MARL policies, as shared rewards during training often lead to emergent asymmetries and specialization. We leverage its structure to investigate coordination behaviors through temporal logic specifications and risk metrics, evaluating policies as black boxes without internal access.

A. Risk analysis for reward

Conditional Value at Risk (CVaR) [8] is a coherent risk measure that evaluates the expected loss in the worst-case scenarios beyond a given confidence level (e.g. the 95th percentile). To complement logical property check, we also quantify coordination consistency using Conditional Value-at-Risk at confidence level $\alpha = 0.1$ for the reward variable:

$$\text{CVaR}_{0.1}(-R) = \mathbb{E}\left[-R \mid -R \geq q_{0.1}\right] \tag{1}$$

where $q_{0.1}$ is the 90th percentile $1 - 0.1$ of the negative episodic reward distribution, $\mathbb{E}$ is the expectation operator, and R is the episodic reward. CVaR captures the expected performance in the worst 10% of cases, revealing fragile coordination behaviors that may remain hidden when only considering average reward.

a) Agent pair configurations: We evaluate agent coordination across three pair configurations. In the RL-RL setting, both agents are independently trained via Proximal Policy Optimization (PPO) [12] with shared layouts and reward functions. In the Greedy-Greedy setup, both agents follow a deterministic, hand-coded policy that always chooses the nearest subtask (ingredient collection, cooking, and dish delivery) contributing to delivery. The RL-Greedy configuration pairs an RL-trained agent with a fixed greedy partner, highlighting the adaptability and role preference of the learned policy.

Each pair is evaluated over 100 episodes in two kitchen layouts. We log all agent positions, actions, rewards, and environment events at every timestep.

B. Behavior analysis with STL

Rather than relying solely on aggregate rewards based on whether a dish is successfully delivered, we incorporate Signal Temporal Logic (STL) specifications [10] to capture nuanced coordination behaviors that evolve over time.

Agents remain on a grid and move in four cardinal directions: up, down, left, and right. We define a "conflict" as one agent trailing behind the other without performing any useful interactions. An agent is considered to be in conflict at timestep t if it faces the same direction as its partner, is located on the same row or column in that facing direction, is within a Manhattan distance of 2 in that direction, and performs no environment interaction (e.g., `INTERACT` is not used). This is formally captured by the STL specification $\varphi_{\text{conflict}}$ in Table I, which identifies conflict intervals lasting longer than 5 steps.

For liveness, we verify that each agent maintains sufficient activity by requiring that no agent remains idle for more than 10 consecutive steps. Here, $(t : t + 10)$ denotes the interval over which the idle condition holds, as formalized by $\varphi_{\text{activeness}}$ in Table I.

TABLE I: Formal specifications of coordination properties.

Property	Specification
$\varphi_{\text{conflict}}$	$\neg\mathbb{F}\Big(\mathbb{G}_{[0,5]}(\text{same\_dir} \wedge \text{in\_front\_and\_close} \wedge \text{no\_interaction})\Big)$
$\varphi_{\text{activeness}}$	$\mathbb{G}_{[0,T]}\big(\neg\text{idle}_i(t:t+10)\big)$
$\varphi_{\text{fair\_reward}}$	$\Big(\frac{R_0}{R_0+R_1} > 0.4\Big) \wedge \Big(\frac{R_1}{R_0+R_1} > 0.4\Big)$

C. Role attribution

To understand how coordination strategies emerge across agent configurations, we analyze both the distribution of shaped rewards and the division of task responsibilities between agents. Although the Steakhouse environment provides only a shared reward for delivery, intermediate shaped rewards (e.g., for placing ingredients or handling dishes) enable attribution of individual contributions during execution.

Each agent's action is labeled as either a preparation or delivery task. Preparation includes actions related to fetching ingredients, placing them into pots, and interacting with cooking stations. Delivery includes actions involving collecting cooked soup and delivering it to the serving counter.

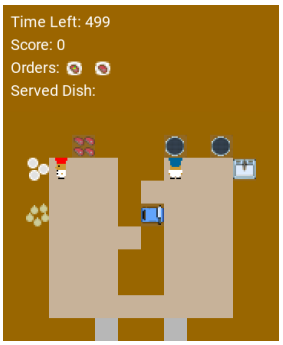
We measure task role asymmetry by computing the proportion of delivery and preparation actions taken by each agent over an episode. This allows us to determine whether responsibilities are shared symmetrically or skewed toward a specific agent. These annotations are computed automatically from action logs.

In parallel, we compare reward distribution under two reward schemes defined in [7]: sparse rewards, which grant a fixed 100-point bonus for each successful dish delivery to both agents, and shaped rewards, which assign smaller bonuses (e.g., 10 points) exclusively to the agent performing subtasks such as chopping, washing, or plating. To evaluate whether both agents contribute meaningfully, we use the STL property $\varphi_{\text{fair\_reward}}$ (Table I), which requires each agent to earn at least 40% of the total shaped rewards, ensuring that no single agent dominates the cooperative effort.

By combining role labeling and reward tracking, we can assess whether agents not only coordinate effectively but also participate equitably.

IV. Experimental Evaluation

We evaluate each agent configuration over 100 episodes in two Steakhouse layouts: `hallway` and `ring`. The two layouts are customized overcooked layouts, each contains two agents inside. The `hallway` layout splits the kitchen into two partially separated rooms. One side contains key preparation stations such as ingredients (onions O and meat M) and chopping boards (B), while the other houses grills (G), dishes (D), water sinks (W), and delivery counters (S). Narrow passageways connect the two sides, requiring agents to divide responsibilities across rooms and coordinate hand-offs of partially completed tasks. Unlike the corridor layout, congestion is less severe, but effective workload



(a) Map layout of the `hallway` environment.

```
{
    "grid":  """XXXXXXXXXXX
              XXXMXXXGXGXX
              XD1  XX2  WX
              XX   X    XX
              X0   XB   XX
              XX    X   XX
              XX   XX   XX
              XX   XX   XX
              XX        XX
              XXXXSXXSXXXX""",
    "start_all_orders" : [
        { "ingredients" : ["meat"]},
        { "ingredients" : ["chicken"]}
    ],
    "order_list": ['steak_dish','steak_onion_dish'],
    "cook_time": 10,
    "delivery_reward": 20,
    "in_order_delivery_reward": 100,
    "chicken_time": 100,
    "steak_time": 60,
    "rew_shaping_params": None,
}
```

(b) Customization of map layouts.

Fig. 2: We used the environment from [7], which leveraged the encoding provided by Overcooked to modify and tailor the map, resulting in customized layouts.

allocation and role specialization are essential for timely dish delivery. The `ring` layout is essentially a long narrow corridor. Ingredients, dishes, chopping boards, and grills are all positioned along the central strip, with walls tightly constraining movement. Because the space only allows two agents to pass side-by-side in limited regions, navigation becomes a major coordination challenge. Agents frequently need to resolve conflicts at choke points and manage access to shared stations, making motion coordination critical.

The agent is responsible for chopping onions on the chopping board, cooking meat on the grill, washing plates in the sink when none are clean, assembling the prepared ingredients on plates, and delivering the completed dishes.

The `hallway` layout presents less congestion than a corridor but requires effective workload allocation and role specialization, as agents must divide tasks across rooms and coordinate timely hand-offs; for example, agents can save time by cooperating to pass ingredients through the middle counter. In contrast, the `ring` layout is highly congested, making navigation, conflict resolution, and motion coordination critical.

Evaluation focuses on four coordination aspects: consistency under failure (CVaR), conflict behavior, activeness, and balance in task and reward distribution. Our results demonstrate that average reward alone is insufficient to assess effective cooperation, and that STL-based runtime analysis reveals critical behavioral asymmetries.

A. Risk performance analysis

To measure performance stability, we compute the Conditional Value-at-Risk at confidence level $\alpha = 0.1$, defined in equation 1.

Table II presents both average reward and CVaR of reward across configurations.

Greedy-Greedy pairs are fully deterministic, consistently achieving 200 points with no performance variance. RL-Greedy pairs exhibit the highest average returns while maintaining strong worst-case consistency ($\text{CVaR}_{0.1}$ = 300). In

Agent Pair	Hallway Layout		Ring Layout	
	Mean	$\text{CVaR}_{0.1}$	Mean	$\text{CVaR}_{0.1}$
Greedy-Greedy	200.0	200.0	200.0	200.0
RL-Greedy	341.0	240.0	372.4	280.0
RL-RL	288.4	0.0	305.0	0.0

TABLE II: Average episode reward and worst-case performance ($\text{CVaR}_{0.1}$).

Agent Pair	Layout	Conflict Avoidance	Activeness	Reward Balance
Greedy-Greedy	Hallway	41%	10.0%	49.0%
	Ring	57%	12.5%	62.5%
RL-Greedy	Hallway	58%	18.5%	20.0%
	Ring	65%	21.2%	18.0%
RL-RL	Hallway	65%	26.0%	20.0%
	Ring	70%	26.4%	20.5%

TABLE III: Behavioral metrics across agent pairs and layouts. Conflict Avoidance shows episodes satisfying $\varphi_{\text{conflict}}$, Activeness reports $\varphi_{\text{activeness}}$ compliance, and Reward Balance indicates $\varphi_{\text{balance\_reward}}$ satisfaction.

contrast, RL-RL pairs suffer from severe instability, with frequent failures leading to zero reward in the worst 10% of episodes despite comparable mean performance.

B. Behavioral metrics analysis

We evaluate agent pairs across three key metrics: conflict avoidance, activeness, and reward balance. Conflict intervals are defined as five or more consecutive steps where one agent trails another without environment interaction (STL $\varphi_{\text{conflict}}$). Activeness ($\varphi_{\text{activeness}}$) identifies episodes where agents avoid 10+ step inactivity periods. Reward balance ($\varphi_{\text{fair\_reward}}$) requires both agents to receive $> 40\%$ of shaped rewards.

The key findings highlight distinct behavioral patterns between RL-RL and greedy agent pairs. RL-RL pairs demonstrate the least conflict, achieving 65–70% success rates, which indicates superior motion coordination. Additionally, RL-trained agents maintain higher activeness levels (18.5–26.4%) compared to greedy pairs (10–12.5%). However, while greedy pairs exhibit a more equitable reward distribution (49–62.5%), RL-based pairs show stronger role specialization, with reward balance skewed toward 18–20.5%. These metrics reveal a trade-off: RL agents achieve better coordination and sustained activity but at the expense of reward equality, suggesting the emergence of specialized roles that are not fully captured by individual reward metrics.

C. Role attribution analysis

Next, we examine the division of task roles—specifically, how often each agent performs preparation tasks (e.g., collecting onions) versus delivery tasks (e.g., serving soup). Table IV summarizes the action distribution for each agent type.

These results support the finding that RL agents—especially when paired with greedy agents—tend to take on helper roles, performing preparation while relying on their partner to complete deliveries. This leads to uneven shaped reward distributions, despite shared objective functions.

Layout	Agent Pair	Role	Agent 0	Agent 1	Biased?
Hallway	Greedy-Greedy	Delivery	50%	50%	No
		Prep	42%	48%	No
	RL-Greedy	Delivery	33%	67%	Yes
		Prep	70%	30%	Yes
	RL-RL	Delivery	42%	58%	No
		Prep	60%	40%	No
Ring	Greedy-Greedy	Delivery	50%	50%	No
		Prep	50%	50%	No
	RL-Greedy	Delivery	26%	74%	Yes
		Prep	78%	22%	Yes
	RL-RL	Delivery	40%	60%	No
		Prep	54%	46%	No

TABLE IV: Proportion of delivery and preparation actions by each agent. In RL-Greedy pairs, the RL agent is agent 0 and the greedy agent is agent 1.

D. Cross-Layout Comparison

Layout complexity amplifies these differences. In the hallway layout, agents benefit from spatial separation, and role specialization improves throughput. However, in the congested two-room layout, choke points intensify conflicts, magnifying the consequences of poor coordination. RL-Greedy pairs adapt best to these challenges, maintaining both high mean performance and non-zero CVaR values. RL-RL pairs, however, collapse under congestion, as simultaneous pursuit of delivery tasks often leads to deadlock. These cross-layout findings suggest that robustness is highly environment-dependent, and evaluation frameworks must consider how structural constraints shape coordination dynamics.

V. Conclusion

We propose a Multi-Agent Reinforcement Learning (MARL) evaluation framework that integrates Conditional Value-at-Risk (CVaR) analysis with Signal Temporal Logic (STL) monitoring. Unlike conventional metrics that focus on mean episodic reward, our approach explicitly captures robustness, behavioral asymmetries, and fairness properties during execution. Through experiments, we showed that policies with similar average returns can differ substantially in their fragility, role allocation, and coordination balance. In future work, we plan to expand the evaluation to additional environments, more training seeds, and diverse perturbations. These extensions will help determine the generality of the fragilities observed and move toward standardized benchmarks for multi-agent robustness in dynamic environments. Ultimately, we aim to apply this framework to real-world robotic testbeds, where interpretable and risk-sensitive evaluation is crucial for safety-critical deployment.

References

[1] Aksaray, D., Jones, A., Kong, Z., Schwager, M., Belta, C.: Q-learning for robust satisfaction of signal temporal logic specifications. In: 2016 IEEE 55th Conference on Decision and Control (CDC). pp. 6565–6570. IEEE (2016)

[2] Baker, B., Kanitscheider, I., Markov, T., Wu, Y., Powell, G., McGrew, B., Mordatch, I.: Emergent tool use from multi-agent autocurricula. In: International conference on learning representations (2019)

[3] Bukharin, A., Li, Y., Yu, Y., Zhang, Q., Chen, Z., Zuo, S., Zhang, C., Zhang, S., Zhao, T.: Robust multi-agent reinforcement learning via adversarial regularization: Theoretical foundation and stable algorithms. Advances in Neural Information Processing Systems **36**, 68121–68133 (2023)

[4] Carroll, M., Shah, R., Ho, M.K., Griffiths, T., Seshia, S., Abbeel, P., Dragan, A.: On the utility of learning about humans for human-ai coordination. Advances in neural information processing systems **32** (2019)

[5] Hammond, L., Abate, A., Gutierrez, J., Wooldridge, M.: Multi-agent reinforcement learning with temporal logic specifications. arXiv preprint arXiv:2102.00582 (2021)

[6] Hsu, H.L., Wang, W., Pajic, M., Xu, P.: Randomized exploration in cooperative multi-agent reinforcement learning. arXiv preprint arXiv:2404.10728 (2024)

[7] Hsu, Y.C., Defranco, M., Patel, R., Nikolaidis, S.: Integrating field of view in human-aware collaborative planning. arXiv preprint arXiv:2505.14805 (2025)

[8] Lindemann, L., Jiang, L., Matni, N., Pappas, G.J.: Risk of stochastic systems for temporal logic specifications. ACM Transactions on Embedded Computing Systems **22**(3), 1–31 (2023)

[9] Lowe, R., Wu, Y.I., Tamar, A., Harb, J., Pieter Abbeel, O., Mordatch, I.: Multi-agent actor-critic for mixed cooperative-competitive environments. Advances in neural information processing systems **30** (2017)

[10] Maler, O., Nickovic, D.: Monitoring temporal properties of continuous signals. In: International Symposium on Formal Techniques in Real-Time and Fault-Tolerant Systems. pp. 152–166. Springer (2004)

[11] Ritz, F., Phan, T., Müller, R., Gabor, T., Sedlmeier, A., Zeller, M., Wieghardt, J., Schmid, R., Sauer, H., Klein, C., et al.: Specification aware multi-agent reinforcement learning. In: International Conference on Agents and Artificial Intelligence. pp. 3–21. Springer (2021)

[12] Schulman, J., Wolski, F., Dhariwal, P., Radford, A., Klimov, O.: Proximal policy optimization algorithms. arXiv preprint arXiv:1707.06347 (2017)

[13] Wang, J., Yang, S., An, Z., Han, S., Zhang, Z., Mangharam, R., Ma, M., Miao, F.: Multi-agent reinforcement learning guided by signal temporal logic specifications. arXiv preprint arXiv:2306.06808 (2023)

[14] Zang, Y., He, J., Li, K., Fu, H., Fu, Q., Xing, J., Cheng, J.: Automatic grouping for efficient cooperative multi-agent reinforcement learning. Advances in Neural Information Processing Systems **36**, 46105–46121 (2023)

[15] Zimmer, M., Glanois, C., Siddique, U., Weng, P.: Learning fair policies in decentralized cooperative multi-agent reinforcement learning. In: International conference on machine learning. pp. 12967–12978. PMLR (2021)