StylePlan: Style-Conditioned Intent Planning for Zero-Shot Coordination

Anonymous Author(s)
Affiliation
Address
email

Abstract

Zero-shot coordination requires agents to cooperate with unfamiliar partners while adapting through interaction. Effective partner modeling must retain recurring behavioral patterns and respond to a partner's latest actions. We introduce **StylePlan**, a partner-conditioned controller that couples slow behavioral memory with fast predictive intent. An exponential moving average of behavioral fingerprints captures persistent style, while a recent-history encoder predicts near-future partner actions. A style-conditioned prior places the slow summary in the intent space, and a learned gate uses its disagreement with the fast estimate to construct the policy context. Slow style conditions the actor's recurrent transition, and the fused context modulates its action features. Action prediction and Gaussian prior matching train the partner representations jointly with the coordination policy. Experiments on Overcooked V2 demonstrate that StylePlan achieves the highest average cross-play return with unseen partners and the highest post-switch return among the evaluated methods. Structural ablations further highlight the distinct contributions of slow behavioral memory, fast predictive intent, and conflict-aware arbitration. Code is available at `https://github.com/MIKE-TECH-CO/StylePlan`.

1 Introduction

Zero-shot coordination (ZSC) studies how an agent can cooperate effectively with previously unseen partners without online fine-tuning. This problem appears in many cooperative multi-agent settings where coordination must emerge under decentralized execution, partial observability, and uncertainty about a teammate's behavior. Successful cooperation requires more than task competence: the ego agent must infer how the current partner tends to coordinate and what it is likely to do next. Partners may differ in skill, risk preference, division-of-labor convention, responsiveness, and local tactical habits. These differences can substantially change which action is optimal at a given moment. Consequently, policies trained through self-play or with a narrow set of partners often achieve strong in-distribution performance but fail to generalize to novel partners whose conventions differ from those observed during training [Carroll et al., 2019, Hu et al., 2020, Strouse et al., 2021].

A common strategy is to improve partner diversity during training. Population-based approaches construct partner pools from independently trained self-play agents, historical checkpoints, or diversity-promoting objectives [Strouse et al., 2021, Lupu et al., 2021, Zhao et al., 2023, Yu et al., 2023, Yan et al., 2023, Li et al., 2023, Wang et al., 2024a, Nguyen et al., 2024, Hui et al., 2026]. Recent benchmarks and extensions sharpen this question by stressing evaluation diversity, generalized Overcooked layouts, noisy game specification, and multi-team coordination [Wang et al., 2024b, Ruhdorfer et al., 2024, Gessler et al., 2025, Anwar et al., 2024, Abderezaei et al., 2025]. Context-conditioned approaches give the ego policy access to behavior history, inferred strategy types, or latent teammate representations. Online coordination places two temporal demands on this information: the ego must retain recurring behavioral patterns while responding to a partner's latest actions. A partner can

Submitted to 40th Conference on Neural Information Processing Systems (NeurIPS 2026). Do not distribute.

preserve its usual movement and interaction patterns while changing its immediate action sequence. The controller therefore needs both persistent evidence and a prediction of what the partner will do next.

We address these demands with two coupled representations. A slow style state summarizes recurring patterns in observed actions and movement. A fast intent representation predicts near-future partner actions from recent interactions and current task context. To connect the two, we learn a style-conditioned prior in the intent space. Comparing its mean with the fast estimate gives the controller an explicit measure of how recent evidence relates to its accumulated behavioral summary.

StylePlan implements this structure through behavioral memory, predictive intent, and conflict-aware arbitration. A learned gate uses the disagreement between the fast estimate and the style-conditioned prior to weight their contributions to the policy context. Slow style enters the actor's recurrent transition directly, while the fused context modulates its action features. Action prediction and latent prior matching train the partner representations jointly with the coordination policy.

Our contributions are:

- We introduce a two-timescale perspective on partner modeling for zero-shot coordination, distinguishing persistent behavioral style from short-horizon predictive intent.
- We propose StylePlan, which integrates slow behavioral memory and fast predictive intent through a style-conditioned prior and conflict-aware arbitration to guide partner-conditioned decision-making.
- We evaluate StylePlan on Overcooked V2 through unseen-partner cross-play, within-episode partner switches, and structural ablations. Results demonstrate superior average cross-play and post-switch returns compared with the evaluated baselines.

2 Related Work

Population-based zero-shot coordination. A major line of work improves ZSC by exposing the ego agent to diverse training partners. Fictitious Co-Play constructs partner diversity from independently trained self-play agents and historical checkpoints [Strouse et al., 2021]. Other-Play uses symmetry-aware training [Hu et al., 2020], while trajectory-diversity, maximum-entropy population training, human-bias priors, end-to-end partner generation, and open-ended co-play broaden the partner distribution [Lupu et al., 2021, Zhao et al., 2023, Yu et al., 2023, Yan et al., 2023, Li et al., 2023]. Recent methods and benchmarks further study meta-task partners, predictable training pools, evolving games, noisy specifications, and generalized ZSC evaluation [Wang et al., 2024a, Nguyen et al., 2024, Wang et al., 2024b, Anwar et al., 2024, Hui et al., 2026]. These methods substantially improve robustness over self-play, but their main mechanism is to broaden partner exposure rather than to structure what should be inferred online about a partner.

Partner modeling and ad hoc teamwork. A complementary line of work models other agents directly. Agent modeling studies how to infer actions, goals, plans, policy types, or hidden states of other agents [Albrecht and Stone, 2018]. This perspective is central to ad hoc teamwork, where an agent must coordinate with previously unknown teammates [Stone et al., 2010, Mirsky et al., 2023, Zhang et al., 2025, Chan et al., 2026]. Recent zero-shot collaboration work increasingly emphasizes latent partner strategies, theory-of-mind adaptation, and cross-environment generalization [Li et al., 2025, Ni et al., 2026, Jha et al., 2025]. These approaches move beyond purely population-based robustness by adapting to the current teammate. However, the resulting partner representation is often a single embedding, a discrete type, or a static context. The key distinction is not whether partner information is used, but how it is structured. StylePlan structures partner information by separating slow style from fast intent.

Conditional and hierarchical coordination. Another direction improves coordination by imposing structure on the policy itself. Context-conditioned policies condition on partner embeddings, task identifiers, role labels, or environment-level context. Hierarchical approaches decompose coordination into high-level subgoal selection and low-level control, while recent Overcooked-style language-agent benchmarks study explicit communication and collaborative planning [Sun et al., 2025]. In cooperative MARL, centralized training with decentralized execution (CTDE) and MAPPO-style

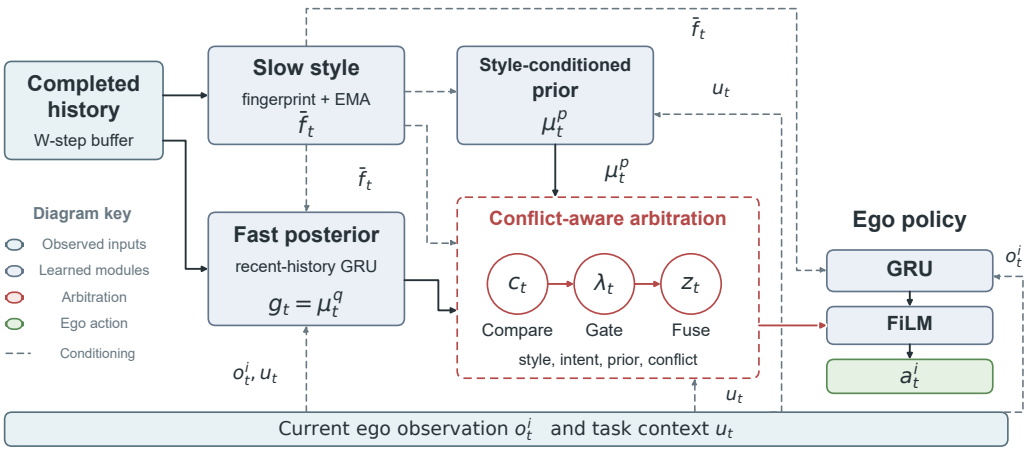

Figure 1: StylePlan's online information flow. Slow behavioral memory conditions the fast posterior and the intent prior. A learned gate uses their projected disagreement to weight separate intent channels for fusion. The fused context modulates policy features through FiLM, while slow style enters the actor's GRU directly.

actor-critic methods provide strong backbones for partially observable cooperative tasks [Yu et al., 2022]. StylePlan is compatible with these backbones, but differs by introducing a structured partner context rather than a single latent teammate vector.

3 Problem Formulation

We study coordination between an ego agent i and a partner j in a cooperative Markov game. At timestep t, the agents receive observations o_t^i, o_t^j, choose actions a_t^i, a_t^j, and obtain a shared reward r_t as the environment transitions from s_t to s_{t+1}. The ego's interaction history $\mathcal{H}_t^i$ contains its observations through time t and observable partner actions and positions from completed steps before t. We denote current task features by u_t. In Overcooked, these features comprise the ego's full-grid observation, normalized player positions, and episode time; partner actions enter the history after each joint transition.

The ego policy $\pi_\theta(a_t^i \mid \mathcal{H}_t^i)$ is trained with a frozen population Π_{train} and evaluated with partners in a disjoint population Π_{test}. The zero-shot coordination objective is

$$J_{\text{ZSC}}(\theta) = \mathbb{E}_{\pi_j \sim p_{\text{test}}} \mathbb{E}_{\tau \sim (\pi_\theta, \pi_j)} \left[\sum_{t=0}^{T-1} \gamma^t r_t \right]. \tag{1}$$

Network parameters remain fixed at evaluation, while the ego's memory evolves with interaction. A partner switch replaces π_j within an episode and tests adaptation to a change in the behavior generating the observed history.

4 Method

StylePlan couples two timescales of partner evidence. A slow style state accumulates behavioral statistics, and a fast intent representation predicts near-future partner actions. A style-conditioned prior maps the slow state and current task context into the intent space. The disagreement between its mean and the fast prediction enters a learned gate, producing a partner context for control. Slow style shapes the actor's recurrent transition; the fused context modulates its action features. Figure 1 summarizes these dependencies.

4.1 Slow Behavioral Memory

Let $\mathcal{W}^j_{t,W}$ contain the last W completed interaction steps available to the ego. We summarize partner behavior using action frequencies, action entropy and switching, movement speed, and spatial occupancy statistics. Validity masks determine which actions and positions contribute to each statistic. The fingerprint ϕ_t, instantaneous encoding f_t, and slow memory $\bar{f}_t$ are

$$\begin{aligned} \phi_t &= \Phi(\mathcal{W}^j_{t,W}), & f_t &= \tanh(E_\varphi(\mathrm{Norm}(\phi_t)))\,, \\ \bar{f}_t &= \rho \bar{f}_{t-1} + (1-\rho) f_t, & \bar{f}_t &\in \mathbb{R}^{d_f}. \end{aligned} \tag{2}$$

Before the first completed transition, $f_t = 0$. The window aggregates observed behavior, and the exponential moving average carries that evidence across decisions. The retention coefficient ρ determines the memory timescale. This slow state conditions both the partner predictor and the ego controller.

4.2 Fast Predictive Intent

A history GRU encodes the most recent $L < W$ completed steps as $e_t = E_{\mathrm{hist}}(\mathcal{W}^j_{t,L})$. Each step contains the ego observation, an observed partner-action encoding, partner position, and action and position visibility indicators. At each decision, the encoder scans valid steps in the short window from a zero initial state. The intent network combines e_t with an encoding of the current ego observation, the slow style, and task context:

$$(\mu^q_t, \ell^q_t) = I_\psi\big([e_t; E_o(o^i_t); \bar{f}_t; u_t]\big)\,. \tag{3}$$

These outputs parameterize a diagonal Gaussian posterior over the predictive latent,

$$q^g_\psi(g \mid \mathcal{H}^i_t, \bar{f}_t) = \mathcal{N}(\mu^q_t, \mathrm{diag}(\exp(\ell^q_t)))\,, \qquad g_t = \mu^q_t \in \mathbb{R}^{d_g}. \tag{4}$$

The mean g_t supplies the fast signal for control. Linear decoding heads produce partner-action distributions

$$q^a_{\psi,k}(\cdot \mid g_t) = \mathrm{softmax}(W_k g_t + b_k), \qquad k = 0, \ldots, H. \tag{5}$$

Here H is the maximum prediction offset: $H = 5$ gives six targets, from a^j_t through a^j_{t+5}. Predicting these actions trains g_t to summarize short-term behavioral evidence. Both training and execution use the posterior mean in the decoder and controller.

4.3 Style-Conditioned Prior and Arbitration

The prior predicts an intent distribution from the slow state and current task context:

$$p_\omega(g \mid \bar{f}_t, u_t) = \mathcal{N}(\mu^p_t, \mathrm{diag}(\exp(\ell^p_t)))\,, \qquad (\mu^p_t, \ell^p_t) = P_\omega([\bar{f}_t; u_t]). \tag{6}$$

The posterior additionally incorporates recent interaction evidence through e_t. Thus μ^p_t provides a style-conditioned reference for the fast estimate g_t. Gaussian matching during training couples their latent spaces, while their means provide the two predictions used by the controller.

We project the two means into a common comparison space, $x_t = W_p \mu^p_t$ and $y_t = W_g g_t$, and compute their cosine disagreement:

$$c_t = 1 - \frac{\langle x_t, y_t \rangle}{\|x_t\|_2 \, \|y_t\|_2 + \epsilon}. \tag{7}$$

A scalar gate conditions on this disagreement together with style, fast intent, and task context:

$$\lambda_t = \mathrm{sigmoid}\big(w^\top_\lambda [\bar{f}_t; g_t; c_t; u_t] + b_\lambda\big)\,, \tag{8}$$

$$z_t = F_\zeta\big([\bar{f}_t; \lambda_t g_t; (1-\lambda_t)\mu^p_t; c_t]\big)\,. \tag{9}$$

The fusion network receives the gated intent vectors as separate input blocks, preserving their distinct sources. The slow state and disagreement remain explicit inputs alongside them. This makes the weighting of recent evidence and the style-conditioned prediction part of the learned control pathway.

4.4 Partner-Conditioned Recurrent Control

Partner information enters the actor at two points. Slow style modifies the carried recurrent state before the GRU update, and the fused context applies a residual FiLM transformation [Perez et al., 2018] to the resulting features:

$$h_t^i = \mathrm{GRU}\big(\eta(o_t^i), \tanh(h_{t-1}^i + W_f \bar{f}_t)\big)\,, \tag{10}$$

$$\tilde{h}_t^i = (1 + a_\zeta(z_t)) \odot h_t^i + b_\zeta(z_t), \tag{11}$$

$$\pi_\theta(\cdot \mid \mathcal{H}_t^i) = \mathrm{softmax}(W_\pi \tilde{h}_t^i + b_\pi). \tag{12}$$

The bounded style injection changes the recurrent transition, carrying persistent partner information into memory. FiLM exposes the current fused context directly to action selection. During centralized training, the critic $V_\nu(\chi_t, h_t^i, z_t)$ receives the joint observation $\chi_t = [o_t^i; o_t^j]$ in addition to the actor state and partner context.

5 Optimization

We collect interactions with a frozen partner population to pretrain the partner model, then jointly optimize the partner model and ego actor–critic. Training combines action prediction, latent prior matching, temporal regularization, and the clipped PPO objective [Schulman et al., 2017, Yu et al., 2022].

Predictive supervision. Let $\mathcal{B}$ index prediction origins in a rollout batch. The mask $M_{t,k}$ selects collected action targets within the same episode and uninterrupted partner segment as their origin. The prediction loss is

$$\mathcal{L}_{\mathrm{pred}} = -\frac{1}{Z_{\mathcal{B}}} \sum_{t \in \mathcal{B}} \sum_{k=0}^{H} M_{t,k} \log q_{\psi,k}^a(a_{t+k}^j \mid g_t). \tag{13}$$

The reduction constant $Z_{\mathcal{B}}$ specifies averaging over prediction origins or valid targets; Table 4 gives the exact reduction for each training setting. Targets are assembled from collected partner actions and paired with predictions computed from each origin's pre-action history.

Coupling the two timescales. We match the posterior to the style-conditioned prior and regularize changes in slow memory:

$$\mathcal{L}_{\mathrm{prior}} = \mathbb{E}_t\Big[\mathrm{KL}\Big(q_\psi^g(\cdot \mid \mathcal{H}_t^i, \bar{f}_t) \,\|\, p_\omega(\cdot \mid \bar{f}_t, u_t)\Big)\Big]\,, \tag{14}$$

$$\mathcal{L}_{\mathrm{smooth}} = \mathbb{E}_t\left\|\bar{f}_t - \bar{f}_{t-1}\right\|_2^2. \tag{15}$$

Action prediction supplies behavioral supervision to the fast representation; prior matching trains a style-based reference in the same latent space. Gradients pass through both Gaussian distributions, linking the predictive and style pathways.

Joint policy learning. Let $r_t^\theta = \pi_\theta(a_t^i \mid \mathcal{H}_t^i)/\pi_{\mathrm{old}}(a_t^i \mid \mathcal{H}_t^i)$ be the policy likelihood ratio and $\hat{A}_t$ the generalized advantage estimate. We optimize

$$\mathcal{L}_{\mathrm{PPO}} = -\mathbb{E}_t\Big[\min\Big(r_t^\theta \hat{A}_t, \mathrm{clip}(r_t^\theta, 1-\epsilon_\pi, 1+\epsilon_\pi)\hat{A}_t\Big)\Big]\,. \tag{16}$$

The total objective is

$$\begin{aligned}\mathcal{L}_{\mathrm{total}} = \mathcal{L}_{\mathrm{PPO}} + c_v\mathcal{L}_V - c_e\mathcal{H}(\pi_\theta) \\ + m(p)\left(\alpha_I\mathcal{L}_{\mathrm{pred}} + \alpha_P\mathcal{L}_{\mathrm{prior}} + \alpha_T\mathcal{L}_{\mathrm{smooth}}\right),\end{aligned} \tag{17}$$

where $\mathcal{L}_V$ is clipped value regression, $\mathcal{H}$ is policy entropy, and $m(p)$ controls the auxiliary-loss schedule over training progress p. Pretraining optimizes the auxiliary terms; joint PPO updates the actor–critic and its partner-conditioning modules together. Algorithm 1 gives the decision loop, and Appendix A.2 lists training settings.

Table 1: Cross-play return on six Overcooked V2 layouts at the 8M training endpoint. Avg. XP is the unweighted mean over the six layouts. Bold marks the highest mean in each column.

Method	Cramped	Asymm.	Ring	Circuit	Forced	Grounded	Avg. XP
MAPPO-SP	63.61 ± 7.29	124.31 ± 35.48	32.75 ± 37.73	7.57 ± 7.85	43.93 ± 16.24	$\mathbf{131.64 \pm 48.83}$	67.30 ± 12.30
PBT	65.60 ± 6.36	151.55 ± 22.70	95.29 ± 47.87	1.65 ± 1.78	41.92 ± 22.24	100.00 ± 25.00	76.00 ± 10.50
FCP	151.45 ± 33.13	159.94 ± 7.58	133.12 ± 28.76	10.00 ± 2.62	111.03 ± 4.33	76.81 ± 1.33	107.06 ± 7.47
MEP	166.95 ± 15.84	209.50 ± 12.84	132.98 ± 31.35	3.13 ± 2.14	39.65 ± 5.65	95.00 ± 18.00	107.87 ± 6.99
E3T	165.91 ± 8.95	174.55 ± 26.21	89.39 ± 59.00	12.84 ± 18.95	42.14 ± 25.25	112.85 ± 49.07	99.61 ± 14.58
GAMMA	157.40 ± 9.99	187.85 ± 30.26	63.30 ± 60.86	6.00 ± 8.99	31.07 ± 17.45	90.00 ± 32.00	89.27 ± 13.05
TrajeDi	22.51 ± 11.28	27.59 ± 2.67	10.23 ± 18.46	0.00 ± 0.00	0.13 ± 0.28	10.00 ± 12.00	11.74 ± 4.15
COLE	138.79 ± 6.93	138.17 ± 44.47	28.70 ± 6.74	12.96 ± 8.48	23.22 ± 19.95	85.00 ± 29.00	71.14 ± 9.69
HSP	153.64 ± 7.44	189.30 ± 26.69	54.76 ± 48.02	9.78 ± 13.33	7.96 ± 4.02	90.00 ± 27.00	84.24 ± 10.54
IPPO	192.01 ± 2.26	90.77 ± 12.27	47.39 ± 22.67	20.28 ± 9.70	38.02 ± 33.05	110.00 ± 36.00	83.08 ± 9.36
CBPR	112.59 ± 18.48	159.60 ± 2.11	$\mathbf{161.86 \pm 11.60}$	12.69 ± 1.77	84.47 ± 10.56	100.00 ± 19.00	105.20 ± 5.15
StylePlan	$\mathbf{196.20 \pm 5.23}$	$\mathbf{212.67 \pm 2.86}$	141.71 ± 4.14	$\mathbf{21.71 \pm 3.45}$	$\mathbf{145.89 \pm 1.57}$	77.79 ± 0.62	$\mathbf{132.66 \pm 1.37}$

6 Experimental Setup

Environments and baselines. We evaluate on six Overcooked V2 layouts: cramped-room, asymmetric-advantages, coordination-ring, counter-circuit, forced-coordination, and grounded-coordination-ring [Gessler et al., 2025]. These layouts cover narrow passages, asymmetric task access, circular traffic, counter passing, and constrained division of labor. We compare StylePlan with 11 baselines: MAPPO-SP, PBT, FCP, MEP, E3T, GAMMA, TrajeDi, COLE, HSP, IPPO, and CBPR. Their method categories are listed in Table 2.

Evaluation protocol. The baseline comparisons use fixed checkpoints at 8M training steps and a pool of 16 frozen test partners disjoint from the training partners. Cross-play evaluates 32 episodes per partner in each agent position. Static episodes last 400 steps; dynamic episodes replace the partner at step 200. Table 1 reports layout-wise XP across all six layouts. Table 2 reports the five-layout evaluation on cramped-room, asymmetric-advantages, coordination-ring, counter-circuit, and forced-coordination, aggregated over three shared training seeds and both agent positions. All evaluations keep network parameters fixed while updating the ego's online state.

Metrics. XP measures cross-play return with unseen partners. ZSC is the full-episode cooperative return in the five-layout evaluation. Post-switch return is the cumulative reward over the 200 steps following partner replacement. Recovery is the number of steps required for the reward window to return to its pre-switch level, averaged over valid episodes in which recovery is observed. The ablation study additionally reports normalized performance drop from the pre-switch score to the lowest post-switch window. Higher returns, shorter recovery, and smaller drop indicate stronger coordination.

The ego policy is a GRU actor–critic trained with Adam and clipped PPO updates [Cho et al., 2014, Kingma and Ba, 2015, Schulman et al., 2017]. Structural ablations evaluate 64 episodes per partner on three layouts.

7 Results and Analysis

7.1 Cross-Play with Unseen Partners

Table 1 compares cross-play performance across six layouts. StylePlan attains the highest average XP, 132.66 ± 1.37, exceeding MEP, the strongest baseline by this average, by 23.0%. It has the highest mean return on Cramped, Asymm., Circuit, and Forced. On Forced, StylePlan reaches 145.89 ± 1.57, a 31.4% increase over FCP's 111.03 ± 4.33. CBPR and MAPPO-SP attain the largest returns on Ring and Grounded, respectively.

The comparison connects partner diversity with interaction-conditioned control. FCP and MEP obtain higher average cross-play returns than MAPPO-SP and PBT, while StylePlan adds an explicit response to the current partner's observed behavior. Its slow memory summarizes recurring patterns, its fast predictor encodes recent action evidence, and the arbitration module routes these signals into

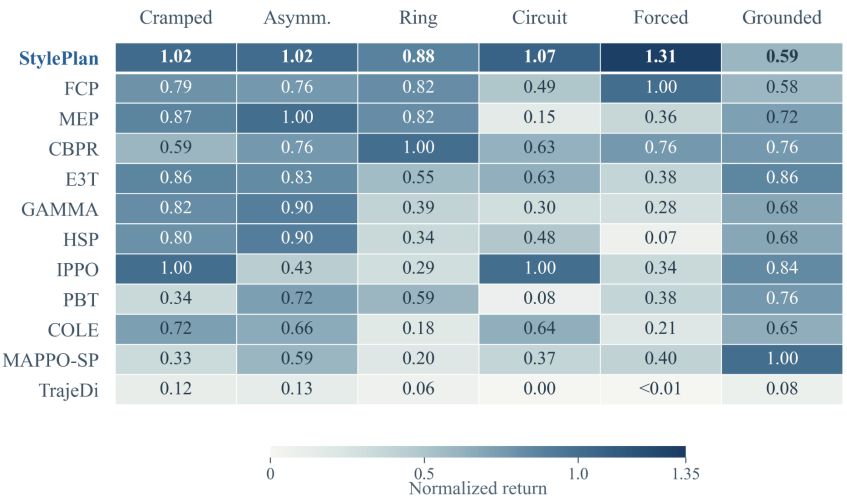

Figure 2: Normalized cross-play performance across six Overcooked V2 layouts. Returns are divided by the strongest baseline mean on each layout. StylePlan obtains the highest return on four layouts, with a 31.4% improvement on Forced.

action selection. This architecture combines behavioral coverage from training partners with context that evolves during coordination.

Figure 2 shows the normalized cross-play performance across six Overcooked V2 layouts. StylePlan achieves the highest returns on four layouts, with its largest improvement of 31.4% on Forced. This advantage highlights the value of modeling partner behavior at multiple timescales. The slow branch captures persistent behavioral tendencies, while the fast branch tracks recent behavioral cues. Conflict-aware arbitration integrates these complementary signals to guide action selection, enabling the ego policy to maintain consistent coordination while adapting to ongoing interactions. The pronounced improvement on Forced particularly demonstrates the benefit of this design when successful cooperation requires closely aligned behaviors.

7.2 Coordination after Partner Changes

Table 2 compares static coordination and partner-switch adaptation across five layouts. StylePlan achieves the highest mean ZSC and post-switch returns of 142.80 ± 0.89 and 67.20 ± 0.72, outperforming FCP by 23.9% and 13.9%, respectively. Its post-switch return also exceeds MEP's 57.46 ± 2.88. While FCP and MEP emphasize partner diversity during training, StylePlan additionally adapts to observed partner behavior. Slow behavioral memory captures persistent patterns, while fast predictive intent tracks recent actions. The style-conditioned prior and learned gate integrate these signals, allowing the policy to adapt as evidence about a new partner accumulates.

StylePlan's observed recovery time is 46.17 ± 7.40 steps, which is 24.6%, 21.7%, and 26.9% shorter than CBPR, IPPO, and MEP, respectively. Although MAPPO-SP and FCP recover earlier, StylePlan achieves the highest post-switch return, demonstrating its ability to sustain effective coordination after partner replacement.

7.3 Ablation Study

Table 2 compares static coordination and partner-switch adaptation across five layouts. StylePlan achieves the highest mean ZSC and post-switch returns of 142.80 ± 0.89 and 67.20 ± 0.72, outperforming FCP by 23.9% and 13.9%, respectively. Its post-switch return also exceeds MEP's 57.46 ± 2.88, demonstrating strong coordination when partners change within an episode.

While FCP and MEP emphasize partner diversity during training, StylePlan additionally adapts its policy based on observed partner behavior. Slow behavioral memory captures persistent patterns, while fast predictive intent incorporates recent actions. The style-conditioned prior and learned

Table 2: Static and partner-switch evaluation on five common layouts with fixed 8M checkpoints. Higher ZSC and post-switch return are better; lower recovery is better. Recovery averages valid episodes with observed recovery, and an em dash denotes an unavailable statistic. Bold marks the best mean in each metric.

Method	Method category	ZSC ↑	Post-switch ↑	Recovery ↓
MAPPO-SP	Single-policy self-play	53.74 ± 6.19	24.10 ± 3.28	$\mathbf{38.27 \pm 5.74}$
PBT	Population self-play	76.50 ± 6.82	35.40 ± 2.71	45.92 ± 6.89
FCP	Frozen partner population	115.23 ± 4.13	59.02 ± 1.75	43.18 ± 1.37
MEP	Maximum-entropy population	113.87 ± 5.88	57.46 ± 2.88	63.19 ± 17.89
TrajeDi	Trajectory diversity	9.31 ± 0.87	3.51 ± 0.35	—
HSP	Hidden-utility self-play	84.37 ± 16.38	42.68 ± 8.75	76.65 ± 11.50
COLE	Cooperative open-ended learning	70.03 ± 13.17	35.96 ± 6.68	65.90 ± 6.60
E3T	Partner-action prediction	93.70 ± 15.85	45.32 ± 7.70	53.98 ± 8.30
GAMMA	Generative partner population	94.43 ± 17.74	47.94 ± 9.23	62.83 ± 9.42
IPPO	Independent PPO	73.27 ± 1.51	37.70 ± 2.14	58.95 ± 0.12
CBPR	Bayesian policy selection	106.87 ± 7.13	50.04 ± 3.18	61.23 ± 9.89
StylePlan	Slow-style/fast-intent arbitration	$\mathbf{142.80 \pm 0.89}$	$\mathbf{67.20 \pm 0.72}$	46.17 ± 7.40

Table 3: Descriptive structural ablations on three layouts using the R setting at fixed $8{,}388{,}608$-step checkpoints. Entries are mean $\pm$ sample SD across three training seeds, after within-seed equal-layout averaging. Bold marks the best displayed mean.

Variant	XP ↑	Post ↑	Rec. ↓
Full StylePlan	$\mathbf{183.64 \pm 1.64}$	85.89 ± 2.20	35.91 ± 0.69
No partner model	170.65 ± 1.43	79.93 ± 1.16	36.04 ± 1.11
Style only	170.07 ± 1.56	78.70 ± 0.75	37.12 ± 2.05
Intent only	183.36 ± 3.27	$\mathbf{88.59 \pm 2.36}$	35.29 ± 1.31
No arbitration	180.86 ± 4.29	84.75 ± 2.45	36.65 ± 0.95
No style	183.50 ± 4.00	87.04 ± 1.12	$\mathbf{34.15 \pm 0.67}$
No intent	182.02 ± 2.60	86.27 ± 1.39	35.82 ± 0.58

Post denotes return during the 200 steps after partner replacement. Rec. is conditional on valid episodes with observed recovery; coverage and censoring are reported separately.

gate integrate these signals, allowing the policy context to evolve as evidence about a new partner accumulates. This supports effective coordination following partner switches.

StylePlan achieves an observed recovery time of 46.17 ± 7.40 steps, which is 24.6%, 21.7%, and 26.9% shorter than CBPR, IPPO, and MEP, respectively. Although MAPPO-SP and FCP recover earlier, at 38.27 and 43.18 steps, StylePlan achieves the highest post-switch return. These results highlight the complementary nature of recovery speed and cumulative post-switch performance, demonstrating StylePlan's ability to sustain effective coordination after partner replacement.

8 Conclusion

We introduced StylePlan, a partner-conditioned controller that integrates slow behavioral memory and fast predictive intent through a style-conditioned prior and learned arbitration. The resulting context guides recurrent control and evolves through interaction with unfamiliar partners. Across five Overcooked V2 layouts, StylePlan achieves the highest mean static coordination and post-switch returns among the evaluated methods. These results highlight the promise of combining persistent behavioral summaries with recent interaction evidence, offering a practical framework for partner-aware zero-shot coordination.

References

Ava Abderezaei, Chi-Hui Lin, Joseph Miceli, Naren Sivagnanadasan, Stéphane Aroca-Ouellette, Jake Brawer, and Alessandro Roncone. Towards zero-shot coordination between teams of agents: The N-XPlay framework. *arXiv preprint arXiv:2506.17560*, 2025. doi: 10.48550/arXiv.2506.17560.

Stefano V. Albrecht and Peter Stone. Autonomous agents modelling other agents: A comprehensive survey and open problems. *Artificial Intelligence*, 258:66–95, 2018. doi: 10.1016/j.artint.2018.01.002.

Usman Anwar, Ashish Pandian, Jia Wan, David Krueger, and Jakob Foerster. Noisy zero-shot coordination: Breaking the common knowledge assumption in zero-shot coordination games. *arXiv preprint arXiv:2411.04976*, 2024. doi: 10.48550/arXiv.2411.04976.

Micah Carroll, Rohin Shah, Mark K. Ho, Thomas L. Griffiths, Sanjit A. Seshia, Pieter Abbeel, and Anca Dragan. On the utility of learning about humans for human-ai coordination. In *Advances in Neural Information Processing Systems*, volume 32, 2019.

Hohei Chan, Xinzhi Zhang, Antao Xiang, Weinan Zhang, and Mengchen Zhao. PADiff: Predictive and adaptive diffusion policies for ad hoc teamwork. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pages 19943–19951, 2026. doi: 10.1609/aaai.v40i24.39078.

Kyunghyun Cho, Bart van Merriënboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. Learning phrase representations using RNN encoder–decoder for statistical machine translation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing*, pages 1724–1734, 2014. doi: 10.3115/v1/D14-1179.

Tobias Gessler, Tin Dizdarević, Anisoara Calinescu, Benjamin Ellis, Andrei Lupu, and Jakob Nicolaus Foerster. OvercookedV2: Rethinking overcooked for zero-shot coordination. In *Proceedings of the International Conference on Learning Representations*, 2025.

Hengyuan Hu, Adam Lerer, Alex Peysakhovich, and Jakob Foerster. "Other-Play" for zero-shot coordination. In *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pages 4399–4410. PMLR, 2020.

Bingyu Hui, Lebin Yu, Quanming Yao, Yunpeng Qu, Xudong Zhang, and Jian Wang. Efficient reinforcement learning for zero-shot coordination in evolving games. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pages 22110–22118, 2026. doi: 10.1609/aaai.v40i26.39366.

Kunal Jha, Wilka Carvalho, Yancheng Liang, Simon S. Du, Max Kleiman-Weiner, and Natasha Jaques. Cross-environment cooperation enables zero-shot multi-agent coordination. *arXiv preprint arXiv:2504.12714*, 2025. doi: 10.48550/arXiv.2504.12714.

Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *Proceedings of the International Conference on Learning Representations*, 2015.

Benjamin Li, Shuyang Shi, Lucia Romero, Huao Li, Yaqi Xie, Woojun Kim, Stefanos Nikolaidis, Michael Lewis, Katia Sycara, and Simon Stepputtis. Modeling latent partner strategies for adaptive zero-shot human-agent collaboration. *arXiv preprint arXiv:2507.05244*, 2025. doi: 10.48550/arXiv.2507.05244.

Yang Li, Shao Zhang, Jichen Sun, Yali Du, Ying Wen, Xinbing Wang, and Wei Pan. Cooperative open-ended learning framework for zero-shot coordination. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pages 20470–20484. PMLR, 2023.

Andrei Lupu, Brandon Cui, Hengyuan Hu, and Jakob Foerster. Trajectory diversity for zero-shot coordination. In *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pages 7204–7213. PMLR, 2021.

Reuth Mirsky, Ignacio Carlucho, Arrasy Rahman, Elliot Fosong, William Macke, Mohan Sridharan, Peter Stone, and Stefano V. Albrecht. A survey of ad hoc teamwork research. In *Multi-Agent Systems: 19th European Conference, EUMAS 2022, and 5th International Conference, AT 2022, Düsseldorf, Germany, September 14–16, 2022, Revised Selected Papers*, pages 275–293. Springer, 2023. doi: 10.1007/978-3-031-20614-6_16.

Dung Nguyen, Hung Le, Kien Do, Sunil Gupta, Svetha Venkatesh, and Truyen Tran. Diversifying training pool predictability for zero-shot coordination: A theory of mind approach. In *Proceedings of the Thirty-Third International Joint Conference on Artificial Intelligence*, pages 166–174, 2024. doi: 10.24963/ijcai.2024/19.

Andrew Ni, Simon Stepputtis, Stefanos Nikolaidis, Michael Lewis, Katia P. Sycara, and Woojun Kim. Theory of mind guided strategy adaptation for zero-shot coordination. *arXiv preprint arXiv:2602.12458*, 2026. doi: 10.48550/arXiv.2602.12458.

Ethan Perez, Florian Strub, Harm de Vries, Vincent Dumoulin, and Aaron Courville. FiLM: Visual reasoning with a general conditioning layer. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 32, 2018.

Constantin Ruhdorfer, Matteo Bortoletto, Anna Penzkofer, and Andreas Bulling. The overcooked generalisation challenge. *arXiv preprint arXiv:2406.17949*, 2024. doi: 10.48550/arXiv.2406.17949.

John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*, 2017.

Peter Stone, Gal A. Kaminka, Sarit Kraus, and Jeffrey S. Rosenschein. Ad hoc autonomous agent teams: Collaboration without pre-coordination. In *Proceedings of the Twenty-Fourth AAAI Conference on Artificial Intelligence*, pages 1504–1509, 2010.

DJ Strouse, Kevin McKee, Matt Botvinick, Edward Hughes, and Richard Everett. Collaborating with humans without human data. In *Advances in Neural Information Processing Systems*, volume 34, pages 14502–14515, 2021.

Haochen Sun, Shuwen Zhang, Lujie Niu, Lei Ren, Hao Xu, Hao Fu, Fangkun Zhao, Caixia Yuan, and Xiaojie Wang. Collab-overcooked: Benchmarking and evaluating large language models as collaborative agents. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 4922–4951, 2025. URL `https://aclanthology.org/2025.emnlp-main.249/`.

Haoming Wang, Zhaoming Tian, Yunpeng Song, Xiangliang Zhang, Zhongmin Cai, Jun Wang, and Yali Du. Beyond single stationary policies: Meta-task players as naturally superior collaborators. In *Advances in Neural Information Processing Systems*, volume 37, 2024a.

Xihuai Wang, Shao Zhang, Wenhao Zhang, Wentao Dong, Jingxiao Chen, Ying Wen, and Weinan Zhang. ZSC-Eval: An evaluation toolkit and benchmark for multi-agent zero-shot coordination. In *Advances in Neural Information Processing Systems*, volume 37, 2024b.

Xue Yan, Jiaxian Guo, Xingzhou Lou, Jun Wang, Haifeng Zhang, and Yali Du. An efficient end-to-end training approach for zero-shot human-ai coordination. In *Advances in Neural Information Processing Systems*, volume 36, 2023.

Chao Yu, Akash Velu, Eugene Vinitsky, Jiaxuan Gao, Yu Wang, Alexandre M. Bayen, and Yi Wu. The surprising effectiveness of PPO in cooperative multi-agent games. In *Advances in Neural Information Processing Systems*, volume 35, pages 24611–24624, 2022.

Chao Yu, Jiaxuan Gao, Weilin Liu, Botian Xu, Hao Tang, Jiaqi Yang, Yu Wang, and Yi Wu. Learning zero-shot cooperation with humans, assuming humans are biased. In *Proceedings of the International Conference on Learning Representations*, 2023.

Xinzhi Zhang, Hohei Chan, Deheng Ye, Yi Cai, and Mengchen Zhao. Ad hoc teamwork via offline goal-based decision transformers. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*. PMLR, 2025. URL `https://openreview.net/forum?id=tl3FlgWScA`.

Rui Zhao, Jinming Song, Yufeng Yuan, Haifeng Hu, Yang Gao, Yi Wu, Zhongqian Sun, and Yang Wei. Maximum entropy population-based training for zero-shot human-ai coordination. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, pages 6145–6153, 2023. doi: 10.1609/aaai.v37i5.25758.

A Additional Method Details

A.1 Online Coordination

Algorithm 1 Online coordination with StylePlan

Require: Learned parameters, normalization statistics, windows W, L, retention ρ

Initialize history buffers, $\bar{f}_{-1} = 0$, and $h^i_{-1} = 0$ at episode start
for each timestep t **do**
  Receive ego observation o^i_t and task features u_t
  Compute masked fingerprint ϕ_t from the last W completed steps
  Encode f_t and update slow memory $\bar{f}_t$ using Eq. (2)
  Encode the last L steps and infer posterior mean g_t
  Compute prior mean μ^p_t from $[\bar{f}_t; u_t]$
  Compute disagreement c_t, gate λ_t, and context z_t
  Update h^i_t, apply FiLM, and sample $a^i_t \sim \pi_\theta(\cdot \mid \mathcal{H}^i_t)$
  Execute the joint transition and append observable partner evidence
end for

The history GRU starts from zero for each short-window encoding; the actor GRU carries state across decisions. Episode boundaries reset both history buffers and the style and actor memories. Within-episode partner switches retain the ego's online state. Running normalization statistics are estimated between training batches and fixed during rollout collection, recurrent replay, and evaluation.

A.2 Architecture and Training Settings

The style window is $W = 50$, the intent window is $L = 10$, and the EMA retention is $\rho = 0.99$. Style and intent have dimensions $d_f = 32$ and $d_g = 16$; recurrent states and fused context have dimension 64. The prediction decoder has $H + 1 = 6$ action heads. Posterior and prior log variances are clipped to $[-8, 4]$. Cosine disagreement uses $\epsilon = 10^{-8}$ and evaluates each vector norm as $\sqrt{\|x\|_2^2 + 10^{-12}}$, with the result clipped to $[0, 2]$.

Table 4 specifies three optimization settings for the same architecture. Let $N_M = \sum_{t \in \mathcal{B}} \sum_{k=0}^{H} M_{t,k}$ and let $p = b/(N-1)$ be joint PPO progress at batch $b \in \{0, \ldots, N-1\}$. Arrows in the table denote linear schedules over p.

Table 4: Optimization settings. Reward scaling applies once to the collected training reward before GAE.

Setting	V0	R	RA
Prediction normalizer $Z_{\mathcal{B}}$	$\lvert\mathcal{B}\rvert$	$\lvert\mathcal{B}\rvert$	$\max(N_M, 1)$
Prior weight α_P	0.01	0.01	0.001
Auxiliary multiplier $m(p)$	1	1	$\min(1, p/0.1)$
Training reward scale	1	0.05	0.05
Learning rate	3×10^{-4}	$3\times10^{-4} \to 3\times10^{-5}$	$3\times10^{-4} \to 3\times10^{-5}$
Entropy coefficient c_e	0.01	$0.03 \to 0.01$	$0.03 \to 0.01$
Adam state after pretraining	Retain	Reset	Reset

All settings use $\alpha_I = 0.1$, $\alpha_T = 0.01$, $c_v = 0.5$, discount $\gamma = 0.99$, GAE parameter 0.95, PPO and value-clipping thresholds 0.2, and maximum gradient norm 0.5. Auxiliary pretraining sets $\alpha_I = 1$ and $m(p) = 1$. The optimization uses the prediction, Gaussian KL, temporal, and actor–critic terms in Eq. (17). R/RA reset Adam's moments and step count once after full-model pretraining while retaining network parameters, normalization statistics, and online memories.

Training uses 16 environments, rollouts of 64 steps, two optimization epochs per pretraining batch, and eight per joint PPO batch. The collected Overcooked training reward is the sparse team reward plus 0.1 times the sum of both agents' shaping rewards. Value targets share the reward scale in Table 4; evaluation reports native sparse team returns. Recurrent PPO replay starts from the saved rollout-

start memory and recomputes features under the updated parameters with the batch's normalization statistics.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The abstract and introduction describe the main contributions of the paper and contextualize these against previous contributions in the literature.

   Guidelines:

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: We discuss limitations in the discussion section.

   Guidelines:

   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [Yes]

Justification: The paper provides explicit assumptions for its theoretical result and includes a full proof in the Appendix. Specifically, Theorem 1 is stated under the assumption of at most k-order interactions and temporal dependencies of at most H steps, and a constructive inductive proof is provided with both the base case and inductive step.

Guidelines:

- The answer [N/A] means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

Answer: [Yes]

Justification: We provide precise algorithm descriptions and specific details for obtaining efficient implementations in the Appendix.

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
- If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
  (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
  (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
  (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
  (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [Yes]

   Justification: We provide an anonymous link to the code in the abstract.

   Guidelines:

   - The answer [N/A] means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
   - The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
   - At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
   - Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

   Answer: [Yes]

   Justification: We describe all implementation details and hyperparameters in the Appendix.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [Yes]

   Justification: The mean of the test win rate is plotted as the bold line with 95% bootstrapped confidence intervals shown as shaded areas.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).

- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
- If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: We report the computational resources available of experiments in the Appendix.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: This work conforms with the NeurIPS Code of Ethics.

   Guidelines:

   - The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [No]

    Justification: As this is a theoretical paper in nature, the societal impacts of this paper are somewhat limited in scope.

    Guidelines:

    - The answer [N/A] means that there is no societal impact of the work performed.
    - If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.

- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

    Answer: [No]

    Justification: Our Reinforcement Learning algorithm improvements use standard benchmarks without sensitive data or deployment modules, with safeguards planned for future applications.

    Guidelines:

    - The answer [N/A] means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: Open-source code and environments are all listed and relevant authors are cited when comparing to their benchmarks.

    Guidelines:

    - The answer [N/A] means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.

- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [No]

    Justification: No new assets introduced.

    Guidelines:

    - The answer [N/A] means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [No]

    Justification: We do not do any research with human subjects or crowdsourcing.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [No]

    Justification: We do not do any research with human subjects or crowdsourcing.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.

- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

    Answer: [No]

    Justification: LLMs were only used for minor text polishing (e.g., grammar fixes), with no involvement in core algorithmic development or technical analysis.

    Guidelines:

    - The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
    - Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.