LEARNING CROSS-CONTEXT TEAMMATE SIGNATURES FOR ZERO-SHOT COORDINATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Zero shot coordination studies how agents cooperate with previously unseen teammates. A key challenge is to determine what information in interaction history actually matters for coordination, since teammate behavior is often entangled with the situations encountered. We argue that teammate representations should preserve coordination relevant conditional behavior rather than identity differences alone. We propose FSIG, which learns functional signatures by comparing teammate action predictions under shared queries. Behavioral supervision grounds these predictions in observed actions, while a consistency objective encourages agreement across matched histories. The learned signature then conditions the ego policy for adaptive coordination with unseen teammates. Across nine tasks spanning classic Overcooked, OvercookedV2, and Level Based Foraging, FSIG achieves competitive coordination performance. Further analyses show that interaction history provides additional predictive information, the functional consistency objective improves agreement across histories, and learned signatures affect controller behavior during coordination.

1 INTRODUCTION

Zero shot coordination (ZSC) studies how agents can effectively cooperate with partners they have not encountered or jointly trained with before (Hu et al., 2020; Wang et al., 2024b). This capability is a fundamental component of open cooperative intelligence, with broad relevance to human AI collaboration, ad hoc teamwork, and dynamically formed multi agent teams (Strouse et al., 2021a; Rahman et al., 2024). However, when unfamiliar partners exhibit different strategies, preferences, and behavioral patterns, the challenge is not only to understand their behavior from limited interaction, but also to determine which information actually matters for the current coordination decision. This raises a more fundamental representation learning question: what should a teammate representation preserve in order to support effective coordination with unfamiliar partners?

A natural approach is to learn representations from interaction histories that capture partner identity, latent strategy type, or overall behavioral patterns (Grover et al., 2018b; Li et al., 2025b). Yet behavioral differences that distinguish two partners are not necessarily the differences that matter for coordination. As illustrated in Fig. 1, the same partner may induce different action predictions when viewed through different interaction histories, even under the same coordination query. For example, one history may suggest that the teammate will move aside and open the ego's path, whereas another may suggest that it will stay and keep the path blocked. Directly comparing the states or actions appearing in those histories does not reveal whether they imply the same coordination-relevant behavior. Conversely, two partners with different overall behavioral patterns may still call for the same complementary response in a coordination-critical situation. A useful teammate representation should therefore not preserve every behavioral difference that separates partners, but should emphasize conditional behavioral information that changes the ego's coordination response. In other words, rather than asking what type of partner this is, we are interested in how this partner will respond when coordination decisions actually matter.

Existing ZSC research approaches this problem from two main directions. One line constructs increasingly diverse training partners, exposing the ego to different skills, conventions, and behavioral preferences (Strouse et al., 2021a; Zhao et al., 2023b; Yu et al., 2023a). Such methods broaden what the ego has seen during training, but do not directly specify which behavioral differences of a

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

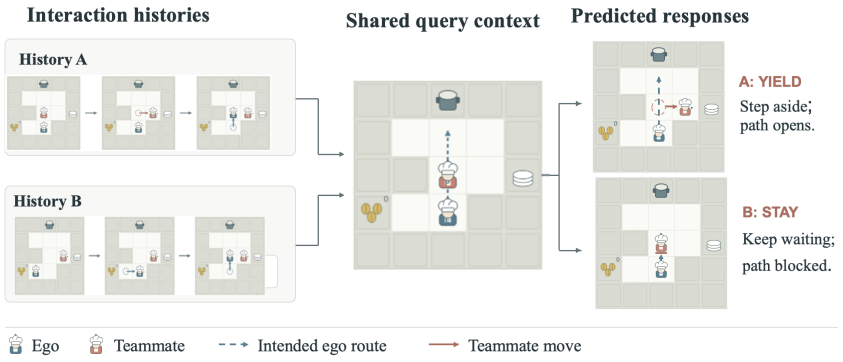


Figure 1: **Motivation for functional teammate representations.** Different interaction histories can imply different teammate responses under the same coordination context. FSIG uses shared queries to learn behavioral signatures that remain consistent across histories of the same teammate.

novel partner are relevant to the current coordination decision. A second line uses interaction history to predict partner behavior or infer latent strategies for online adaptation (Grover et al., 2018b; Papoudakis et al., 2021b; Yan et al., 2023a; Li et al., 2025b). However, partner diversity and behavior prediction mainly answer what has been encountered and what the partner may do. They do not directly answer which behavioral information will actually change the ego's coordination decision. Recent work on partner coverage and ZSC evaluation from a best response perspective offers a related insight. Behaviorally different partners do not necessarily require different cooperative responses (Rahman et al., 2024; Wang et al., 2024b).

Based on these observations, our central claim is that the goal of teammate representation should not be to distinguish who the partner is as finely as possible, but to characterize how the partner responds in coordination relevant situations. This means that a useful teammate representation does not need to fully recover partner identity, personality, or latent strategy. Instead, it should preserve the conditional behavioral information that changes coordination decisions. Two partners with different identities or overall behavioral styles need not be separated merely because they are different if they play the same coordination relevant functional role in the task. Conversely, the same partner may exhibit different coordination relevant responses across contexts. We call a teammate representation whose meaning is defined by such coordination relevant conditional behavior a functional signature.

Motivated by this view, we propose FSIG, which learns functional teammate representations through shared queries. The key idea is to provide different interaction histories with the same task situation as a common behavioral reference and compare what partner behavior those histories imply in that situation. Thus, even when two histories contain different states and actions, they can be compared by answering the same question: how would the partner respond here? Behavioral supervision anchors these responses in recorded partner behavior, while consistency across histories encourages stable functional information to be preserved across different experiences. The learned functional signature is then used for coordination, allowing the ego to adapt its behavior as partner evidence accumulates. In this way, FSIG explicitly separates understanding the partner from responding to the partner. Representation learning determines what teammate information should be preserved, while coordination learning determines how that information should be converted into complementary behavior.

We evaluate FSIG on nine tasks spanning classic Overcooked, OvercookedV2, and Level Based Foraging (Gessler et al., 2025; Papoudakis et al., 2021b). FSIG achieves competitive coordination performance across the benchmark suites and attains the highest mean return on several tasks. Beyond aggregate return, fixed query prediction, consistency analysis across histories, and signature interventions provide complementary evidence about the learned representation. Interaction history supplies additional predictive information about partner behavior, the functional consistency objective improves predictive agreement across distinct histories, and the controller's behavior is sensitive to information carried by the learned signatures. Together, these analyses distinguish the quality of the functional representation itself from its value in actual coordination.

2 Related Work

Zero-Shot Coordination. Zero-shot coordination (ZSC) studies how agents cooperate with partners not encountered during training (Hu et al., 2020). Self-play provides a natural starting point, but policies may overfit to arbitrary conventions and generalize poorly in cross-play. Other-Play mitigates this issue by exploiting symmetries in the environment (Hu et al., 2020). A complementary line of work improves generalization by broadening the partner distribution seen during training. Fictitious Co-Play trains against historical self-play checkpoints (Strouse et al., 2021a), while population-based approaches encourage diverse strategies, preferences, or incompatible cooperative behaviors (Zhao et al., 2023a; Yu et al., 2023b; Li et al., 2023b). Recent work further expands the training distribution through environment variation (Jha et al., 2025) or adaptive partner generation and selection (Ruhdorfer et al., 2026a). These methods primarily diversify the partners, conventions, or situations encountered during training. In contrast, FSIG focuses on how behavioral evidence from a teammate should be represented across different interaction histories, learning signatures whose behavioral readouts remain consistent under shared query contexts.

Teammate Modeling and Strategy Inference. Another line of work infers behaviorally relevant characteristics of unfamiliar teammates from interaction history. Prior methods encode trajectories into latent representations of policies, strategies, roles, or teammate behavior. Policy representations have been learned through cross-episode prediction and metric objectives (Grover et al., 2018a), character and mental-state modeling (Rabinowitz et al., 2018), and auxiliary reconstruction or action-prediction objectives under partial observability (Papoudakis et al., 2021a; Yan et al., 2023b). Other approaches adapt coordination through inferred contexts, strategy types, or latent policy clusters (Lou et al., 2023; Li et al., 2025a). Related representation-learning work also characterizes policies through behavioral similarity (Agarwal et al., 2021), while conditional neural processes motivate describing an object through predictions at common query inputs (Garnelo et al., 2018). FSIG differs by defining cross-history agreement directly at the behavioral readout: signatures inferred from different histories of the same teammate are evaluated under shared queries and encouraged to produce consistent action distributions, without directly forcing their latent representations to coincide. The learned signature is then frozen and used to condition the coordination policy, separating representation learning from downstream cooperative control.

3 Preliminaries

We study zero-shot coordination in a two-agent cooperative environment. An ego policy π^e controls one agent, while a teammate follows a fixed, possibly recurrent policy π^p within each episode. At time t, both agents select actions from their available observations; the environment executes the joint action and returns the shared reward r_t and subsequent observations. The ego's decision-time history is $h_t^e = ((o_k^e, a_{k-1}^e))_{k=0}^t$, with $a_{-1}^e = \mathrm{BOS}$ a beginning-of-episode token. Thus, the model receives ego observations and completed ego actions, not a teammate's action log, policy identity, hidden recurrent state, or future outcomes. Episode-start flags reset the online memories.

Training partners belong to $\mathcal{P}_{\mathrm{train}}$; evaluation uses $\mathcal{P}_{\mathrm{test}}$, whose policy-training sources are disjoint from those used for representation learning, ego training, and checkpoint selection. For a specified test-partner distribution ρ_{test}, the objective is

$$J(\pi^e) = \mathbb{E}_{\pi^p \sim \rho_{\mathrm{test}},\, \tau \sim (\pi^e, \pi^p)} \left[\sum_{t=0}^{T-1} r_t \right]. \tag{1}$$

Here r_t is the native team reward; training may additionally use reward shaping and discounting. Zero-shot execution permits history-dependent inference within an episode, but no parameter updates for a test partner and no memory transfer between test episodes. The formulation also applies when the environment observation is spatially complete: the teammate's policy and internal memory remain unavailable.

4 Functional Signatures for Adaptive Coordination

A teammate representation should preserve behavioral information that remains useful across the situations encountered during interaction. FSIG learns this information by evaluating different his-

tories of the same teammate under shared query contexts. The resulting functional signature summarizes a history through the teammate behavior it predicts, with comparability defined at the decoder output. We first train this representation using behavioral supervision and prediction consistency. We then freeze it and learn a recurrent controller that uses the signature together with the current observation to maximize team reward. Figure 2 summarizes the two stages.

4.1 Learning Functional Signatures through Shared Queries

Interaction histories entangle teammate behavior with the situations in which that behavior is observed. Comparing actions from different histories therefore does not isolate how the teammate would respond in a common situation. We instead compare the behavioral predictions induced by those histories at the same query observation. Let h denote a history available to the ego, q a query observation, and $\mathcal{A}^p$ the teammate action space. A history encoder E_θ produces a signature, whose behavioral readout is defined by a shared decoder D_ϕ,

$$s(h) = E_\theta(h), \qquad \Phi_{\theta,\phi}(h, q) = D_\phi\big(\cdot \mid q, s(h)\big) \in \Delta(\mathcal{A}^p). \tag{2}$$

Here $\Delta(\mathcal{A}^p)$ is the probability simplex over teammate actions. The functional meaning of $s(h)$ is specified by the map $q \mapsto \Phi_{\theta,\phi}(h, q)$, rather than by the coordinates of the signature alone. We write Φ when the parameters are clear. The encoder and decoder share an observation frontend, whose tied parameters are optimized jointly.

Training uses two source episodes $H_n^{(1)}$ and $H_n^{(2)}$ and a separate query episode Q_n. All three share the same teammate checkpoint, ego slot, and collector policy, which controls the ego during data acquisition. Each source supplies a prefix of length $t+1$ containing ego observations and completed ego actions. A decision $j_{n,t}$ sampled from Q_n supplies the ego observation $q_{n,t}$ and the teammate action $y_{n,t}$ taken at that decision. The resulting training set is

$$\mathcal{D} = \Big\{ \big(h_{n,t}^{(1)}, h_{n,t}^{(2)}, q_{n,t}, y_{n,t}\big) \Big\}_{n=1,\dots,N\,,\ t=0,\dots,T-1}. \tag{3}$$

The query index $j_{n,t}$ need not equal the source prefix index t. Both branches receive exactly the same query observation and target action. The query episode is excluded from both source histories, and the target action is used only for supervision. Checkpoint identities define the training groups and are never network inputs.

To express what the representation should preserve, let ρ be a specified distribution over query observations. We define the functional discrepancy between two histories as

$$d_\rho\big(h^{(1)}, h^{(2)}\big) = \mathbb{E}_{q\sim\rho}\Big[\mathrm{JS}\Big(\Phi(h^{(1)}, q), \Phi(h^{(2)}, q)\Big)\Big]. \tag{4}$$

This quantity measures disagreement between behavioral readouts on the support of ρ. It imposes no direct distance penalty on the latent vectors. During training, item n uses $\rho_n = T^{-1}\sum_{j=0}^{T-1} \delta_{o_j^{Q_n}}$, the empirical distribution of observations in Q_n, where δ_q is a point mass at q. We sample one query decision for each source prefix pair and keep these assignments fixed throughout pretraining.

Consistency alone admits identical predictions that disregard the recorded behavior. We therefore supervise both branches with the teammate action from the query episode. Writing $P_{n,t}^{(v)} = \Phi(h_{n,t}^{(v)}, q_{n,t})$, the prediction objective is

$$\mathcal{L}_{\mathrm{query}} = -\frac{1}{2NT}\sum_{n=1}^{N}\sum_{t=0}^{T-1}\sum_{v=1}^{2} \log P_{n,t}^{(v)}(y_{n,t}). \tag{5}$$

The corresponding consistency objective compares the two predictions at those same query observations,

$$\mathcal{L}_{\mathrm{func}} = \frac{1}{NT}\sum_{n=1}^{N}\sum_{t=0}^{T-1} \mathrm{JS}\Big(P_{n,t}^{(1)}, P_{n,t}^{(2)}\Big). \tag{6}$$

It is a fixed Monte Carlo realization of the average of $d_{\rho_n}(h_{n,t}^{(1)}, h_{n,t}^{(2)})$ over source prefix pairs. For distributions P and R, we use $M = (P+R)/2$ and $\mathrm{JS}(P,R) = \frac{1}{2}\,\mathrm{KL}(P\|M) + \frac{1}{2}\,\mathrm{KL}(R\|M)$. Both losses have unit weight,

$$\min_{\theta,\phi}\ \mathcal{L}_{\mathrm{FSIG}}, \qquad \mathcal{L}_{\mathrm{FSIG}} = \mathcal{L}_{\mathrm{query}} + \mathcal{L}_{\mathrm{func}}. \tag{7}$$

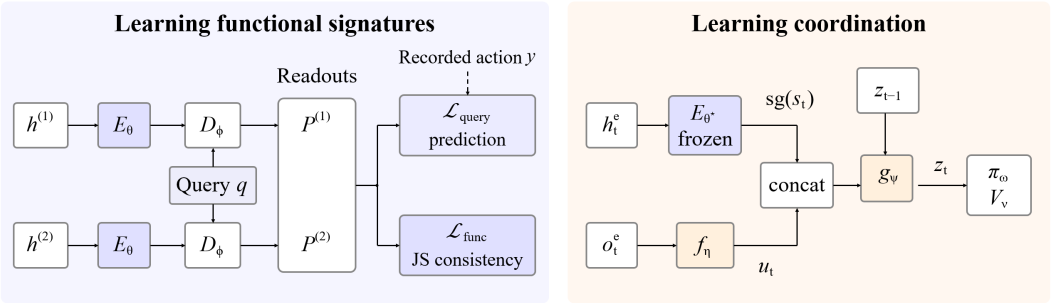


Figure 2: **Overview of FSIG. Left:** Two histories of the same teammate are evaluated under a shared query. Behavioral supervision grounds both readouts, while functional consistency aligns their predicted action distributions. **Right:** The pretrained encoder is frozen, and its signature conditions a recurrent controller optimized from team reward.

Gradients propagate through both history branches and the shared decoder. The prediction term grounds the readouts in observed actions, while the consistency term discourages disagreement across source histories under a shared query.

The comparison fixes the decoder's observed context, rather than the full latent environment state. Its scope is therefore the collected query distribution and the behavioral information recoverable from the histories. In particular, the objective encourages compatible predictions without requiring identical histories or establishing a unique latent description of the teammate. If query observations alone predict actions adequately, these losses do not by themselves guarantee that the decoder uses the signature. We assess history information through a predictor without history and through signature interventions in Section 5.3.

4.2 Adapting Coordination with Functional Signatures

Predicting teammate behavior does not specify which ego action maximizes team reward. The response also depends on the current observation and the preceding interaction. We therefore learn a coordination state that integrates the signature with task information. At decision time t, let o_t^e be the ego observation and h_t^e its available interaction history, defined in Section 3. The current joint action has not yet been taken. With the pretrained encoder parameters $\theta^\star$ fixed, we compute

$$s_t = E_{\theta^\star}(h_t^e), \qquad u_t = f_\eta(o_t^e). \tag{8}$$

A recurrent transition g_ψ updates the coordination state, which supplies the policy and value heads,

$$\begin{aligned} z_t &= g_\psi\big(z_{t-1}, \mathrm{concat}\big(u_t, \mathrm{sg}(s_t)\big)\big)\,, \\ \pi_t^e &= \pi_\omega(\,\cdot \mid z_t), \qquad V_t = V_\nu(z_t). \end{aligned} \tag{9}$$

Here concat concatenates its arguments and sg stops gradients. In our implementation, g_ψ is a gated recurrent unit (GRU). Both the actor and critic read z_t, and the signature enters before the recurrent update.

The controller receives the signature itself. It does not receive the decoder's action probabilities or a query episode. Behavioral supervision defines the representation during pretraining, while reward learning determines how that representation affects the ego policy. The coordination state can change as new evidence arrives within an episode. It is a learned recurrent state and does not encode an explicit search procedure or a supervised plan.

4.3 Optimization and Zero Shot Execution

We first optimize Eq. (7) on the fixed training corpus, obtaining encoder and decoder parameters $(\theta^\star, \phi^\star)$. We then freeze both and train the controller parameters $\Theta_C = (\eta, \psi, \omega, \nu)$ from fresh interactions with training partners. The controller objective is

$$\max_{\Theta_C} J_{\text{train}}(\Theta_C), \qquad J_{\text{train}}(\Theta_C) = \mathbb{E}_{\substack{\pi^p \sim \rho_{\text{train}} \\ \tau \sim (\pi_{\Theta_C}, \pi^p)}} \left[\sum_{t=0}^{T-1} \gamma^t r_t^{\text{train}} \right], \tag{10}$$

where ρ_{train} is a distribution over the training partner set $\mathcal{P}_{\text{train}}$, r_t^{train} is the shared training reward, and γ is the discount factor. We optimize the controller using recurrent proximal policy optimization (PPO) (Schulman et al., 2017). No FSIG auxiliary objective is applied in this stage. Freezing the representation preserves its learned behavioral mapping while the controller learns how to use it.

At execution with an unseen teammate, all model parameters remain fixed. The encoder and controller recurrent states reset at the start of each episode and update using the ongoing interaction. At each decision, the ego computes Eq. (8), updates Eq. (9), and selects an action from π_t^e. Subsequent observations become available only after the joint transition. This adaptation requires no teammate identity, separate query episode, or gradient update during deployment. Appendix A gives the data construction, architecture, optimization, and execution procedure.

5 Experiments

We evaluate whether FSIG coordinates with unseen partners and whether its functional signatures provide predictive information that the controller uses. We first compare coordination returns across complementary tasks, then examine fixed-query prediction, consistency across histories, and controller sensitivity.

5.1 Benchmarks and Evaluation

We evaluate FSIG on three complementary suites: classic Overcooked for cross-play across established layouts, OvercookedV2 for asymmetric-information and protocol-formation challenges, and Level-Based Foraging (LBF) as a structurally different ad hoc teamwork domain (Fig. 3) (Gessler et al., 2025; Papoudakis et al., 2021b). The main OvercookedV2 tasks are Grounded Coordination Simple (GCS), Demo Cook (Demo), and Test-Time Simple (TTS). Models are trained separately per layout and evaluated zero-shot with unseen partner policies; held-out partners are excluded from both training and checkpoint selection.

Baselines include population-based training (FCP, MEP), teammate modelling (LIAM, E3T), adaptive partner generation (UPD), symmetry-based coordination (OP), self-play (SP), and State Augmentation (SA) (Strouse et al., 2021b; Zhao et al., 2023b; Papoudakis et al., 2021b; Yan et al., 2023a; Ruhdorfer et al., 2026b; Hu et al., 2020; Gessler et al., 2025). Where evaluated, No-Signature is a matched control trained with the signature input fixed to zero. Results report native episodic return as mean $\pm$ sample SD over five independent training units. Matched confidence intervals, partner panels, training budgets, and additional protocol details are provided in Appendix B.

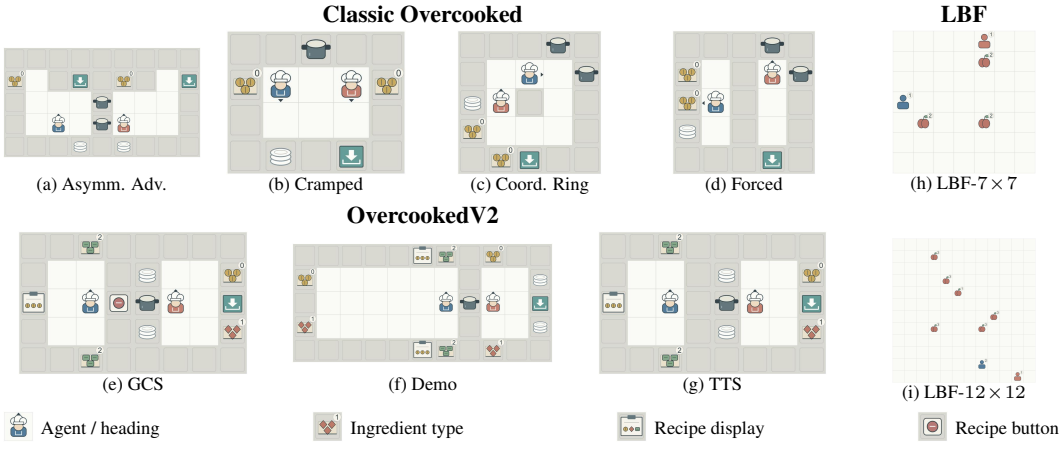


Figure 3: **Benchmark layouts.** State-derived schematic views of fixed reset states. Kitchen digits denote ingredient types; LBF digits denote levels. Kitchen layouts expose different spatial and information constraints; LBF varies grid size and food-loading requirements.

Table 1: **Classic Overcooked cross-play.** Native episodic team return (mean ± sample SD over five ego-training repetitions). Bold indicates the highest mean in each layout, not statistical significance.

Method	Asymm.	Cramped	Ring	Forced
SP/IPPO ref.	416.67 ± 16.37	201.67 ± 35.61	105.33 ± 49.46	60.00 ± 59.58
FCP	390.33 ± 67.31	$\mathbf{255.00 \pm 2.36}$	197.00 ± 41.84	$\mathbf{175.67 \pm 13.05}$
MEP	422.33 ± 46.08	189.33 ± 47.08	142.00 ± 32.02	147.33 ± 23.05
E3T	421.67 ± 44.33	234.33 ± 21.94	175.67 ± 50.15	156.67 ± 13.89
FSIG (ours)	$\mathbf{424.00 \pm 31.46}$	253.67 ± 5.82	$\mathbf{220.33 \pm 65.16}$	164.33 ± 15.57

Table 2: **Fixed heuristic-partner panel in OvercookedV2.** Native episodic team return (mean ± sample SD over five independent training units). SA uses population means over ten members; other rows use individual egos. No-Signature is the matched internal control defined in Sec. 5.1; bold marks the highest mean.

Method	GCS	Demo	TTS
FCP-16	28.986 ± 2.060	44.182 ± 10.419	30.188 ± 0.245
SP	-28.612 ± 26.011	23.279 ± 6.032	-23.200 ± 29.888
SA	-20.518 ± 6.115	46.191 ± 6.000	-9.152 ± 9.710
LIAM	30.475 ± 0.327	48.613 ± 11.866	$\mathbf{30.300 \pm 0.284}$
OP	-3.141 ± 28.991	42.826 ± 11.566	16.468 ± 31.097
No-Signature	29.506 ± 2.189	47.491 ± 8.864	28.428 ± 3.635
FSIG (ours)	$\mathbf{32.348 \pm 5.622}$	$\mathbf{49.065 \pm 5.892}$	30.005 ± 0.240

5.2 Unseen-Partner Coordination

Classic Overcooked. Each method is evaluated against six held-out partner lineages from SP, MEP, and E3T, across both player positions. FSIG achieves the highest mean return on Asymmetric Advantages and Coordination Ring, nearly matches FCP on Cramped Room (253.67 vs. 255.00), and ranks second on Forced Coordination (Table 1). Its advantage on Ring also persists across all three partner sources (Appendix B.2). The layouts impose distinct coordination demands: Asymmetric Advantages favors complementary roles, while Ring and Forced Coordination require adapting to partners' movement, timing, and handoff behavior (Li et al., 2023a). Overall, the results show that FSIG remains competitive across heterogeneous cross-play settings, with its strongest gains appearing where behavioral history can inform complementary responses.

OvercookedV2. On the common frozen heuristic-partner panel, FSIG has the highest mean on GCS and Demo (Table 2). TTS is closely grouped: LIAM, FCP-16, and FSIG score 30.300, 30.188, and 30.005, respectively. The benchmark distinguishes three information structures. GCS supplies a costly, grounded recipe-signalling channel; Demo requires signalling through ingredient placement without grounded communication feedback; TTS allows conventions to be inferred through successful-delivery feedback (Gessler et al., 2025). SA approaches FSIG on Demo (46.191 versus 49.065), but records negative population means on GCS (-20.518) and TTS (-9.152). Thus, broader self-play state coverage alone does not ensure strong cross-play on this partner panel. SA's relative performance varies across these communication structures, whereas FSIG remains competitive across all three. The aggregate returns do not identify the communication strategies learned by either method.

A partner's recent actions can therefore differ in informativeness across these tasks. The matched FSIG–No-Signature estimates are positive on all three, but their CIs include zero (Table 4). We examine predictive information and controller use separately below.

Level-Based Foraging. Each ego faces 24 held-out partners on LBF-7×7 and 18 on LBF-12×12, with 64 episodes per partner. FSIG has the highest mean return and food-collection rate in both settings (Table 3). Its matched gains are $+0.00994$ (95% CI $[0.00766, 0.01226]$) and $+0.01115$ ($[0.00636, 0.01621]$), with positive differences in all five repetitions on each task. LBF couples

Table 3: **Unseen-partner coordination in Level-Based Foraging.** Return is native per-agent episodic return, reported as mean ± sample SD over five ego-training repetitions; Food is the percentage collected by count. On LBF-7 × 7, equal food levels imply $\mathrm{Food}(\%) = 200 \times \mathrm{Return}$; this equivalence does not hold on LBF-12 × 12.

	LBF-7 × 7		**LBF-12 × 12**	
Method	**Return ↑**	**Food ↑**	**Return ↑**	**Food ↑**
SP-IPPO	0.3319 ± 0.0189	66.37	0.2439 ± 0.0092	52.09
FCP	0.4471 ± 0.0016	89.42	0.3897 ± 0.0040	79.42
LIAM	0.4342 ± 0.0055	86.84	0.3670 ± 0.0077	74.77
E3T	0.3493 ± 0.0317	69.85	0.3810 ± 0.0068	77.30
UPD	0.3810 ± 0.0139	76.20	0.3920 ± 0.0056	79.80
No-Signature	0.4504 ± 0.0021	90.08	0.3939 ± 0.0015	80.21
FSIG (ours)	$\mathbf{0.4604 \pm 0.0019}$	**92.07**	$\mathbf{0.4050 \pm 0.0034}$	**82.36**

Table 4: **Matched signature-conditioning effects.** FSIG minus No-Signature return with paired 95% CIs and positive differences out of five training pairs. Suites retain their native return scales; resampling details are in Appendix B.

Task	Mean difference	95% CI	Positive pairs
GCS	2.84150	$[-0.98450, 7.64300]$	3/5
Demo	1.57400	$[-10.15100, 11.46500]$	3/5
TTS	1.57700	$[-0.03900, 4.75300]$	3/5
LBF-7 × 7	0.00994	$[0.00766, 0.01226]$	5/5
LBF-12 × 12	0.01115	$[0.00636, 0.01621]$	5/5

target choice with co-location: successful loading requires sufficient combined agent levels adjacent to the food (Papoudakis et al., 2021b). In LBF-12 × 12, mean gains extend across learned (+0.00829), sequential (+0.01107), and level-aware (+0.01807) partner families. Both low-level- and high-level-food-preferring heuristics benefit, despite their contrasting target-selection conventions (Appendix B.3). These observations are consistent with history helping select complementary responses across heterogeneous partner behaviours.

5.3 Representation and Controller Use

Fixed-query prediction. We freeze the predictors and hold recorded query observations and teammate-action targets fixed while varying source-history length k. The Context-only comparator uses the same query inputs, targets, decoder family, optimizer schedule, and number of training repetitions; its inputs omit source-history conditioning. FSIG additionally receives the functional-consistency objective, whereas Context-only is trained on query cross-entropy (CE). We report

$$\Delta\mathrm{CE}(k) = \mathrm{CE}_{\mathrm{Context}} - \mathrm{CE}_{\mathrm{FSIG}}(k), \tag{11}$$

where positive values favour the history-conditioned predictor. The GCS/Demo probes use eight learned held-out partners; TTS uses 20 heuristic partners. Queries and partner panels remain fixed across k within each task.

At $k = 64$, mean $\Delta\mathrm{CE}$ is positive on GCS (0.0038 nats), Demo (0.0096), and TTS (0.0712; Fig. 4). GCS already benefits at short prefixes, whereas Demo improves as history grows. TTS changes from -0.0127 at $k = 16$ to $+0.0375$ at $k = 32$. The longer-history benefit on TTS is consistent with its interaction-dependent information structure. It is not a measurement of when a protocol forms. These within-task trends identify different history requirements for prediction, rather than a controlled ranking of task difficulty.

Consistency across distinct histories. We test $\mathcal{L}_{\mathrm{func}}$ using a held-out bank generated by a stochastic collector, with matched partner, query, and target but distinct complete source histories. Full and w/o-$\mathcal{L}_{\mathrm{func}}$ models score the same bank. On Ring, Full reduces same-partner predictive JS divergence

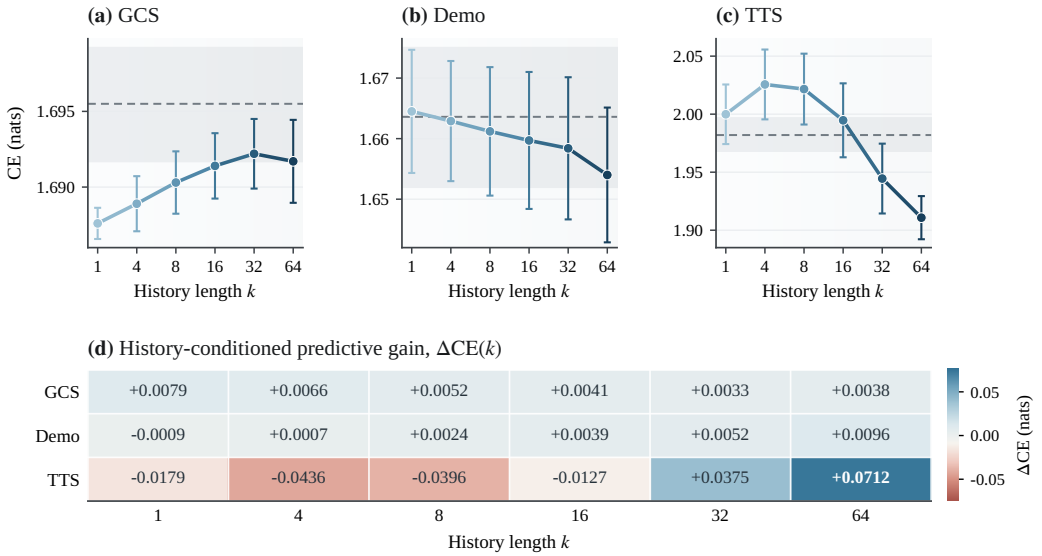


Figure 4: **Fixed-query teammate-action prediction.** (a–c) FSIG CE (solid) and Context-only CE (dashed) on identical query draws within each task. Error bars and grey bands show sample SD over five predictor training repetitions. (d) Mean $\Delta\mathrm{CE}(k)$ in nats; positive values indicate lower FSIG prediction error. GCS/Demo use learned partners and TTS uses heuristic partners. These descriptive curves vary source-history length, not online training time.

from 0.002046 to 0.001531 (difference -0.000515; 95% CI $[-0.000695, -0.000254]$), without an increase in mean query CE. Asymmetric Advantages has a smaller reduction whose interval includes zero. This supports improved held-out prediction consistency on Ring; downstream return effects remain layout dependent and statistically inconclusive (Appendix B.5).

Sensitivity to trajectory-aligned signatures. With the controller frozen, replacing trajectory-aligned FSIGs reduces return by 14.77 on GCS and 5.54 on Demo (Appendix B.6). On TTS, we further compare the native signature stream (Correct), an independent trajectory from the same teammate (Same), and a same-role trajectory from a different teammate (Different). Correct outperforms Same by 1.22 (95% CI $[0.15, 2.72]$) and Different by 1.03 ($[0.06, 2.50]$). These interventions indicate that the controller uses trajectory-aligned signature information. The absence of a Same–Different gap, however, means that this experiment does not separately identify teammate identity from current-trajectory context.

6 Conclusion

We introduced FSIG, a framework for learning teammate representations through behavioral predictions under shared query contexts. FSIG captures coordination-relevant information across different interaction histories and uses the resulting frozen functional signatures to condition a recurrent coordination policy. Across nine tasks spanning three benchmark suites, FSIG achieves competitive zero-shot coordination performance and the highest mean return on several tasks. Further analyses show that interaction history improves teammate-behavior prediction, functional consistency promotes agreement across distinct histories, and learned signatures directly influence controller behavior. Together, these results support functional signatures as a principled representation for adaptive coordination with previously unseen teammates.

AI USE STATEMENT

We used large language models to improve the clarity and organization of the manuscript, identify and summarize related work, and assist with research discussions, code development, and analysis of experimental results. The authors reviewed and revised the assisted content and take responsibility for the final manuscript, implementation, and reported findings.

REFERENCES

Rishabh Agarwal, Marlos C. Machado, Pablo Samuel Castro, and Marc G. Bellemare. Contrastive behavioral similarity embeddings for generalization in reinforcement learning. In *International Conference on Learning Representations*, 2021.

Marta Garnelo, Dan Rosenbaum, Christopher Maddison, Tiago Ramalho, David Saxton, Murray Shanahan, Yee Whye Teh, Danilo Rezende, and S. M. Ali Eslami. Conditional neural processes. In *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pp. 1704–1713. PMLR, 2018.

Tobias Gessler, Tin Dizdarevic, Ani Calinescu, Benjamin Ellis, Andrei Lupu, and Jakob Foerster. OvercookedV2: Rethinking Overcooked for zero-shot coordination. In *International Conference on Learning Representations*, 2025.

Aditya Grover, Maruan Al-Shedivat, Jayesh Gupta, Yuri Burda, and Harrison Edwards. Learning policy representations in multiagent systems. In *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pp. 1802–1811. PMLR, 2018a.

Aditya Grover, Maruan Al-Shedivat, Jayesh Gupta, Yuri Burda, and Harrison Edwards. Learning policy representations in multiagent systems. In *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pp. 1802–1811. PMLR, 2018b.

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*, pp. 4399–4410. PMLR, 2020.

Kunal Jha, Wilka Carvalho, Yancheng Liang, Simon Shaolei Du, Max Kleiman-Weiner, and Natasha Jaques. Cross-environment cooperation enables zero-shot multi-agent coordination. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 27198–27220. PMLR, 2025.

Benjamin Li, Shuyang Shi, Lucia Romero, Huao Li, Yaqi Xie, Woojun Kim, Stefanos Nikolaidis, Charles Lewis, Katia Sycara, and Simon Stepputtis. Adaptively coordinating with novel partners via learned latent strategies. In *Advances in Neural Information Processing Systems*, volume 38, 2025a. doi: 10.52202/085713-0047.

Benjamin Li, Shuyang Shi, Lucia Romero, Huao Li, Yaqi Xie, Woojun Kim, Stefanos Nikolaidis, Michael Lewis, Katia Sycara, and Simon Stepputtis. Adaptively coordinating with novel partners via learned latent strategies. In *Advances in Neural Information Processing Systems*, volume 38, 2025b. doi: 10.52202/085713-0047.

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*, pp. 20470–20484. PMLR, 2023a.

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*, pp. 20470–20484. PMLR, 2023b.

Xingzhou Lou, Jiaxian Guo, Junge Zhang, Jun Wang, Kaiqi Huang, and Yali Du. PECAN: Leveraging policy ensemble for context-aware zero-shot human-AI coordination. In *Proceedings of the 22nd International Conference on Autonomous Agents and Multiagent Systems*, pp. 679–688. International Foundation for Autonomous Agents and Multiagent Systems, 2023. doi: 10.5555/3545946.3598700.

Georgios Papoudakis, Filippos Christianos, and Stefano V. Albrecht. Agent modelling under partial observability for deep reinforcement learning. In *Advances in Neural Information Processing Systems*, volume 34, 2021a.

Georgios Papoudakis, Filippos Christianos, and Stefano V. Albrecht. Agent modelling under partial observability for deep reinforcement learning. In *Advances in Neural Information Processing Systems*, volume 34, 2021b.

Neil Rabinowitz, Frank Perbet, Francis Song, Chiyuan Zhang, S. M. Ali Eslami, and Matthew Botvinick. Machine theory of mind. In *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pp. 4218–4227. PMLR, 2018.

Muhammad Arrasy Rahman, Jiaxun Cui, and Peter Stone. Minimum coverage sets for training robust ad hoc teamwork agents. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pp. 17523–17530, 2024. doi: 10.1609/aaai.v38i16.29702.

Constantin Ruhdorfer, Matteo Bortoletto, Victor Oei, Anna Penzkofer, and Andreas Bulling. Unsupervised partner design enables robust ad-hoc teamwork. In *Proceedings of the 43rd International Conference on Machine Learning*, volume 306 of *Proceedings of Machine Learning Research*, 2026a.

Constantin Ruhdorfer, Matteo Bortoletto, Victor Oei, Anna Penzkofer, and Andreas Bulling. Unsupervised partner design enables robust ad-hoc teamwork. In *Proceedings of the 43rd International Conference on Machine Learning*, volume 306 of *Proceedings of Machine Learning Research*, 2026b.

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

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, pp. 14502–14515, 2021a.

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, pp. 14502–14515, 2021b.

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, 2024a. doi: 10.52202/079017-1501.

Xihui 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. Datasets and Benchmarks Track.

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, 2023a. doi: 10.52202/075280-0119.

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, 2023b. doi: 10.52202/075280-0119.

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 *International Conference on Learning Representations*, 2023a.

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 *International Conference on Learning Representations*, 2023b.

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

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

A Implementation Details

A.1 Source Histories and Query Construction

The training population contains policies obtained by self play and their intermediate and final checkpoints, following the population construction principle of fictitious co play (Strouse et al., 2021b). Each training item uses three distinct episode identities for $H_n^{(1)}$, $H_n^{(2)}$, and Q_n. They share the exact teammate checkpoint, physical ego slot, and collector policy. Distinct identities establish separate collection episodes without asserting statistical independence of the resulting trajectories.

For each source index t, the two encoders process prefixes of equal length. Each query index $j_{n,t}$ is sampled uniformly with replacement from the decisions in Q_n and fixed before representation training. The pair $(q_{n,t}, y_{n,t})$ consists of the ego observation and the recorded teammate action at that decision. Source and query indices are sampled without enforcing temporal alignment. At source time k, the encoder receives the current ego observation, previous ego action, and episode reset flag. A reset replaces the previous action with a beginning of sequence token. Recorded teammate actions, query targets, and teammate checkpoint identities are not encoder inputs.

A.2 Signature Parameterization and Training

Source observations and query observations use the same convolutional encoder with 64 output features, so θ and ϕ include tied frontend parameters. Source features are concatenated with a seven dimensional one hot encoding of the previous ego action, including the beginning of sequence token. A GRU with 128 hidden units processes this sequence. A linear layer maps its state to 64 outputs, split into two 32 dimensional blocks named $\mu(h)$ and $\ell(h) = \log \sigma(h)$. The second block is clipped to $[-5, 2]$. The signature used by both the decoder and controller is deterministic,

$$s(h) = \mathrm{concat}\big(\mu(h), \ell(h)\big) \in \mathbb{R}^{64}. \tag{12}$$

No latent variable is sampled. The names of the two blocks do not imply a posterior distribution or calibrated uncertainty. The decoder concatenates the encoded query observation and both blocks, then applies a multilayer perceptron with layer widths 128, 128, and 6. A softmax produces probabilities over the six primitive teammate actions. The final decoder layer has zero initial weights and biases.

The two encoder passes share parameters and receive gradients from Eq. (7). Neither branch is detached as a teacher. The query loss averages the two negative log likelihoods, and the functional loss uses the two categorical distributions on the same action support. Both terms are averaged over training items and source prefix times with unit weights. No latent distance, triplet objective, or negative pair from another teammate is added.

A.3 Controller and Parameter Freezing

A convolutional task encoder followed by layer normalization produces $u_t \in \mathbb{R}^{128}$. Concatenating the 64 dimensional signature gives a 192 dimensional input to the GRU transition g_ψ, whose state has dimension 128. There is no additional signature projection. The actor and critic share the task encoder and GRU, with separate output heads reading only z_t.

The added 64 rows of each GRU input kernel are initialized to zero. The other controller parameters retain their native initialization. This preserves the native controller's initial input to output mapping through the added signature pathway, while allowing those rows to learn during controller optimization. The signature is wrapped in a gradient stop, and all FSIG parameters are excluded from controller updates. The offline decoder is retained as part of the frozen model but is not required to choose online actions.

A.4 Reward Optimization and Evaluation

We train the controller with recurrent PPO. Its minimized objective is

$$\mathcal{L}_C = \mathcal{L}_{\text{clip}} + c_V \mathcal{L}_V - c_H \mathcal{H}(\pi^e), \qquad \Theta_{\text{FSIG}} \equiv \Theta^{\star}_{\text{FSIG}}, \tag{13}$$

where $\mathcal{L}_{\text{clip}}$ is the clipped policy loss, $\mathcal{L}_V$ is the clipped value loss, and $\mathcal{H}$ is policy entropy. The coefficients c_V and c_H weight the value and entropy terms. The shared training reward includes the benchmark's configured shaping component, which is annealed during training. A training partner is sampled uniformly at each episode boundary and remains fixed within that episode. Training covers both physical ego slots. Only ego transitions contribute to PPO, and the partner policies remain fixed. Controller learning adds no representation loss.

In the common benchmark evaluation interface, all parameters are fixed and both recurrent memories reset at every episode boundary. The encoder processes the current observation and previous ego action, after which the controller updates its state. Evaluation selects the action with maximum policy probability, resolving exact ties by the smallest action index. These action selection conventions specify the evaluated benchmark and are not part of the definition of FSIG. The query target from the current joint decision is never available before the ego acts.

A.5 Training and Execution Procedure

Algorithm 1. Learning and using functional signatures

Input Training partner pool, collection policy, and training budgets.
1. Collect matched episode triples and fix query indices as in Appendix A.1.
2. Initialize the history encoder and shared query decoder.
3. For each representation update, encode both source prefixes and evaluate them at the shared query observation.
4. Update both encoder and decoder parameters using Eq. (7). Repeat until the pretraining budget is exhausted.
5. Freeze all FSIG parameters and initialize the recurrent controller as in Appendix A.3.
6. Collect fresh rollouts with training partners using Eqs. (8) and (9).
7. Update only the controller using Eq. (13). Repeat rollout collection and updates until the controller budget is exhausted.
8. At deployment, freeze all parameters and reset both recurrent states at each episode boundary.
9. At each decision, infer s_t, update z_t, and select the ego action. Incorporate the next observation after the joint transition.

B Additional Experimental Details

B.1 Evaluation Panels and Uncertainty

Models are trained separately for each layout and evaluated with unseen partner policies within that layout. Results retain the native episodic return scale. For the two-agent LBF environments, return is the mean of the agents' native episodic returns. Shared rewards give $R = \frac{1}{2}\frac{\sum_{f\in C}\ell_f}{\sum_{f\in F}\ell_f}$, where C is the set of collected foods, F is the initial food set, and ℓ_f is the food level. Full collection gives $R = 0.5$. Food reports the percentage collected by count. Equal food levels on LBF-7 $\times$ 7 imply $\text{Food}(\%) = 200R$; heterogeneous levels on LBF-12 $\times$ 12 distinguish level-weighted return from collection percentage.

Partner panels. Classic cross-play uses six held-out lineages, balanced across SP, MEP, and E3T, in both physical roles with 64 episode keys per cell. The main OvercookedV2 coordination tables use 20 frozen heuristic partners, both roles, and 100 episodes per partner–role cell for each ego. SA evaluates ten members in each of five independently trained populations; other methods evaluate five independently trained egos. The GCS/Demo fixed-query probes and original stream interventions use learned held-out partners; the fixed-query probes contain eight learned SP checkpoints. The TTS probe and intervention use the 20 heuristic partners. Thus, GCS/Demo representation analyses

and main coordination results use different partner populations. LBF uses fixed panels of 24 and 18 partners on the two grids, respectively.

SA population aggregation and SP/IPPO reference. SA retains all ten final members of each population. Each member's score averages 100 episodes in each of 20 partner $\times$ two role cells; the population score is the equal mean of its ten member scores. The table reports the mean and sample SD of five population scores ($n = 5$). Across three layouts this comprises 600,000 episodes in 60,000 ten-episode shards. Evaluation uses a 400-step horizon, deterministic ego argmax, native heuristic actions, and recurrent-state resets each episode. Shared team reward is accumulated once. Every frozen final member contributes to the reported population mean. Methods share the heuristic evaluation panel; their training populations and budgets follow their respective recipes. The original SA evaluation uses cross-play within learned populations (Gessler et al., 2025). The classic SP/IPPO row provides an independent self-play reference under the same six-lineage, two-role, 64-episode evaluation as the other classic rows, with five ego-training repetitions.

Matched control. FSIG and No-Signature share the initial trainable controller parameters, seed namespace, optimizer, partner-sampling rule, and per-ego training budget. No-Signature receives a zero signature throughout training and evaluation. This comparison measures the effect of training with signature conditioning; frozen-controller stream interventions measure reliance on signatures during evaluation. Realized partner sequences depend on each policy's episode boundaries.

Confidence intervals. For the main heuristic OvercookedV2 comparison, we first average episodes, roles, and partners within each training repetition. The paired 95% percentile CI resamples the five repetition differences 20,000 times. LBF uses 10,000 bootstrap draws over matched training repetitions and matched episodes, with the held-out partner panel fixed. The original GCS/Demo intervention uses 10,000 hierarchical draws over egos, partners within sampled egos, and episodes within fixed roles. The TTS intervention and stochastic-collector consistency diagnostic use 20,000 draws over five paired repetition differences, conditional on the evaluated partner and history banks. Fixed-query error bars report sample SD. All confidence intervals are pointwise with unadjusted nominal coverage. External methods use independent ego-training seeds; their main-table rankings summarize observed mean returns.

The matched effects are reported in Table 4 of the main text.

B.2 Classic Partner-Source Decomposition

Evaluation-pool composition can change method rankings in zero-shot coordination (Wang et al., 2024a). Table 5 reports source decompositions for FCP, MEP, and E3T. The SP/IPPO reference appears at the aggregate level in Table 1. Each source contains two held-out partner lineages, equally weighted. On Ring, FSIG exceeds FCP across SP, MEP, and E3T partners. On Forced Coordination, the largest source-specific deficit relative to FCP occurs for SP partners. Source-level values summarize mean contrasts. Cross-play dispersion is largest on Ring among the reported FSIG classic-layout scores. Among the FCP, MEP, and E3T contrasts, FSIG versus MEP on Cramped Room has a 95% hierarchical-bootstrap interval excluding zero ($+64.33$, $[8.67, 139.00]$); the remaining intervals include zero. The bootstrap samples external ego repetitions as unpaired and preserves the shared evaluation-partner comparison. Repeated deterministic executions retain their original sampling identity.

B.3 LBF Partner Families and Target Selection

The LBF-12 $\times$ 12 panel contains five disjoint partner families (Table 6). Within each family, results average episodes, partner identities, and training repetitions with equal weights at each level. Weighting the five family means by their identity counts recovers the overall 18-partner result. The two level-aware rows are nested within that family and contribute once to the aggregate. Gains are positive for four families; the single human-proxy policy has a mean contrast of -0.00068. This row evaluates a policy proxy. Family summaries were assembled after the main return evaluation.

The low- and high-level heuristics choose the remaining food with the smallest and largest level, respectively, using food level as the selection criterion. Food and agent levels are observable; com-

Table 5: FSIG minus each baseline by classic partner source. All is the equally weighted source mean; no significance is encoded.

Layout	Baseline	SP	MEP	E3T	All
Asymm.	FCP	$+50.00$	-1.00	$+52.00$	$+33.67$
Asymm.	MEP	$+121.00$	$+3.00$	-119.00	$+1.67$
Asymm.	E3T	$+178.00$	-6.00	-165.00	$+2.33$
Cramped	FCP	$+3.00$	-8.00	$+1.00$	-1.33
Cramped	MEP	$+8.00$	$+91.00$	$+94.00$	$+64.33$
Cramped	E3T	$+10.00$	$+48.00$	$+0.00$	$+19.33$
Ring	FCP	$+29.00$	$+23.00$	$+18.00$	$+23.33$
Ring	MEP	$+153.00$	$+69.00$	$+13.00$	$+78.33$
Ring	E3T	$+101.00$	$+74.00$	-41.00	$+44.67$
Forced	FCP	-34.00	-1.00	$+1.00$	-11.33
Forced	MEP	$+64.00$	-5.00	-8.00	$+17.00$
Forced	E3T	$+31.00$	-10.00	$+2.00$	$+7.67$

Table 6: LBF-12×12 family and level-preference results. Differences use raw native return. Individual level preferences are already included in the level-aware family.

Family / partner	Identities	FSIG	No-Signature	Difference
level-aware	2	0.41141	0.39333	$+0.01807$
sequential	6	0.42136	0.41030	$+0.01107$
learned	7	0.42803	0.41974	$+0.00829$
human-proxy	1	0.29385	0.29453	-0.00068
other-heuristic	2	0.32484	0.30445	$+0.02039$
greedy_lowest_level	1	0.39432	0.37724	$+0.01708$
greedy_highest_level	1	0.42849	0.40943	$+0.01906$

bined levels of adjacent agents determine whether loading succeeds. These rules give contrasting target preferences under common loading constraints. FSIG's gains of $+0.01708$ and $+0.01906$ quantify coordination improvements under both preferences. Additional partner-level summaries give positive FSIG–No-Signature mean contrasts for 19 of 24 LBF-7×7 partners and 16 of 18 LBF-12×12 partners. Partner-IQM contrasts are $+0.01037$ and $+0.01072$; lower-tail CVaR_{20} contrasts are $+0.02133$ and $+0.01570$, respectively.

B.4 Fixed-Query Predictor Comparison

For each task, FSIG and Context-only are scored on the same query observations, teammate-action targets, and recorded query draws for every source prefix. The two source branches and all duplicate query draws are retained. GCS/Demo use eight learned held-out SP checkpoints in two roles; TTS uses 20 held-out heuristic policies in two roles, with 64 items per stratum and 400 recorded query draws per item. Source lengths are $1, 4, 8, 16, 32, 64$. The five FSIG models and five Context-only models use the same decoder family, Adam/clipping configuration, optimization schedule, and batch-order protocol, with separate initialization seeds. Context-only removes the source-history input and is trained on query CE. FSIG uses query CE plus functional JS. For a history-invariant predictor, same-query predictions coincide and the functional-consistency term is zero. This comparison scores separately trained predictors on shared queries.

All predictors remain frozen as k varies. The GCS/Demo analysis was assembled after the original evaluation, and TTS was added with the same metric and aggregation rule. Increasing k changes the available source prefix while leaving queries fixed. Source histories contain both teammate behavior and trajectory context. History-based partner prediction also appears in E3T and LIAM (Yan et al., 2023a; Papoudakis et al., 2021b).

Table 7: Held-out same-partner JS divergence. Intervals resample five paired checkpoint differences; lower JS indicates more consistent predictions across histories.

Layout	Full	w/o $\mathcal{L}_{\text{func}}$	Difference	95% CI
Asymm.	.000663	.000976	$-.000313$	$[-.000600, .000046]$
Ring	.001531	.002046	$-.000515$	$[-.000695, -.000254]$

TTS prediction error. At $k = 64$, FSIG reduces mean CE from 1.9824 for Context-only to 1.9112, a relative improvement of 0.0712 nats. The uniform six-action predictor provides an absolute reference of $\log 6 \approx 1.7918$ nats.

B.5 Held-Out Functional Consistency

We compare five paired Full and w/o-$\mathcal{L}_{\text{func}}$ FSIG checkpoints on Asymmetric Advantages and Ring. The stochastic collector samples actions from frozen training-pool SP logits, while held-out partners act greedily. Full and w/o models score the same source and query bank. Each layout contains 768 same-partner history pairs and 400 scored prefix positions per pair. All complete encoder-history pairs differ; the partner, physical slot, collector, query observation, and action target remain matched. Distinct prefixes account for 92.0% of positions on Asymmetric Advantages and 96.1% on Ring. The first prefix is identical by construction, and every position contributes to the reported means.

Full has lower JS in four of five Asymmetric repetitions and all five Ring repetitions (Table 7). Full minus w/o mean query CE is -0.002435 and -0.009253, respectively; both CE intervals include zero. Mean query accuracy changes by $+0.014616$ and $+0.006279$. Different-partner JS also decreases (-0.003737 and -0.001497), reducing $\text{JS}_{\text{diff}} - \text{JS}_{\text{same}}$ by 0.003424 and 0.000982. The lower cross-history JS therefore accompanies a smaller between-partner versus within-partner divergence gap.

The corresponding Full minus w/o control returns are -10.0 (95% CI $[-39.7, 16.0]$) on Asymmetric Advantages and $+29.0$ ($[-13.7, 65.3]$) on Ring. Layouts were selected using historical Full results; the diagnostic bank uses stochastic collection. Representation consistency and coordination returns are reported as separate outcomes, following the evaluation logic in prior coordination work (Zhao et al., 2023b).

B.6 Frozen-Controller Stream Interventions

The GCS/Demo analysis replaces the native FSIG stream with a stream from a real donor trajectory under a fixed controller. Native minus substituted return is 14.7734 (95% CI $[7.7343, 22.7396]$) on GCS and 5.5365 ($[1.2082, 10.1668]$) on Demo, with all five checkpoint effects positive. These evaluations use learned-partner panels. Substitution jointly changes partner and trajectory information and measures reliance on the native stream.

Three-condition TTS design. The TTS subset contains all 20 official held-out heuristic partners, both roles, five training repetitions, and the first 10 of 100 fixed episode ordinals per cell. It was fixed before observing the new returns. Each condition includes 2,000 target episodes of 400 steps. Same additionally collects 2,000 independent closed-loop source episodes with the same teammate policy and ego/FSIG checkpoint. At each target step, its same-time source signature is injected into the controller. Different uses a distinct policy selected by a fixed same-role circular donor map. The map aligns physical role and time index; semantic task state and recipe stage may differ across donor and target trajectories. Across conditions, target teammate, controller checkpoint, physical role, episode keys, environment and partner randomness, horizon, and greedy action rule are held fixed. Policies remain frozen, with closed-loop target states evolving under each condition's actions.

Correct minus Same is positive in four repetitions, Correct minus Different in three, and Same minus Different in one. The Same–Different mean contrast is -0.19, with its interval spanning zero. Correct averages 30.50 on this 10-episode subset and 30.005 on the full 100-episode grid.

Table 8: TTS intervention subset. Return means average partners, roles and episodes within repetitions, then weight the five repetitions equally.

Condition	Mean return	Repetition means
Correct	30.50	31.60, 30.70, 28.85, 30.75, 30.60
Same	29.28	27.55, 30.25, 27.40, 30.60, 30.60
Different	29.47	27.70, 30.10, 28.05, 30.90, 30.60
Contrast	**Difference**	**95% bootstrap CI**
Correct − Same	1.22	$[0.15, 2.72]$
Correct − Different	1.03	$[0.06, 2.50]$
Same − Different	−0.19	$[-0.45, 0.03]$

Inference across training repetitions. The inferential unit is the paired training repetition ($n = 5$). Paired-t 95% intervals are $[-0.866, 3.306]$ for Correct minus Same and $[-1.022, 3.082]$ for Correct minus Different. Exact two-sided sign-flip tests over all 2^5 assignments give $p = 0.125$ and $p = 0.250$, respectively. Omitting repetition 0 gives mean gains of 0.5125 and 0.3125. Table 8 reports the percentile-bootstrap intervals for the same paired differences.

Source-history overlap. All Same source keys differ from their target keys; 53 of 2,000 pairs have identical complete encoder inputs. Including all prespecified pairs gives Correct minus Same 1.220. Excluding the identical-input pairs gives a descriptive point estimate of 1.245. For those histories, signatures differ by at most 9.54×10^{-7}.

Evaluation hardware. Evaluating Correct on RTX 4090 GPUs, as used for Same and Different, yields the same five repetition means as the original RTX 5090 evaluation. Across 2,000 paired episodes with identical checkpoints and episode keys, per-step rewards, termination, and returns agree exactly; action sequences agree in 1,997 episodes. The Correct–Same and Correct–Different gains remain 1.22 and 1.03, with unchanged bootstrap intervals.

B.7 Training Budgets

Table 9 reports per-controller training budgets for FSIG and the matched No-Signature runs. The scope covers controller training; partner-population training, corpus collection, and evaluation fall outside these counts. FSIG additionally fits each signature model for 10 epochs and 720 optimizer updates, with five models per task. The reported accounting consists of environment steps and optimization counts; end-to-end GPU hours across external methods were not measured uniformly.

Table 9: Per-repetition FSIG controller-training budgets. Matched No-Signature uses the same budget where evaluated.

Suite	Environment steps	PPO updates	Optimizer steps
Classic Overcooked	$29,949,952$	457	$116,992$
GCS / Demo / TTS	$29,949,952$	457	$116,992$
LBF-7 × 7 / 12 × 12	$29,982,720$	915	$54,900$

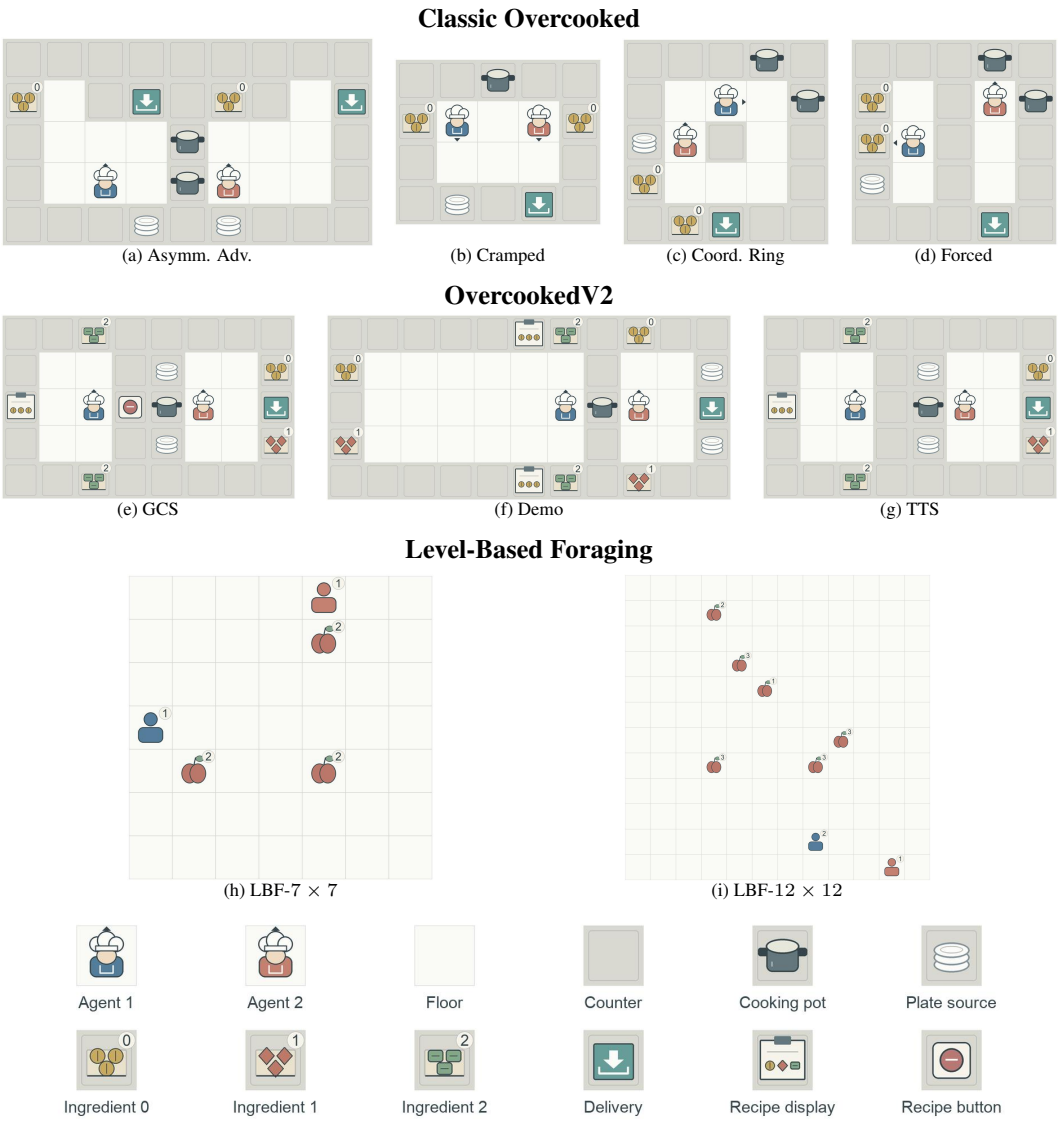


Figure 5: **Benchmark layouts.** Fixed reset states rendered with a shared visual key. Triangles indicate chef headings; kitchen digits identify ingredient types, and LBF digits indicate entity levels.