SLRC: STRUCTURED LOW-RANK COUPLING FOR ONE-ROUND COMMUNICATION IN COOPERATIVE MULTI-AGENT REINFORCEMENT LEARNING

Anonymous authors
Paper under double-blind review

ABSTRACT

Efficient coordination under partial observability remains a central challenge in multi-agent reinforcement learning. Communication supplements local observations, while multi-round exchanges enable agents to refine their responses using updated peer information. However, these exchanges introduce sequential dependencies across rounds. Inspired by this refinement process, we introduce SLRC (**S**tructured **L**ow-**R**ank **C**oupling), a one-round communication framework for cooperative MARL. SLRC adopts a structured affine response model that separates agent-specific transformations from rank-r interaction dependencies. Each agent encodes its contribution into a compact packet of dimension $r^2 + r$. After one logical exchange, receivers aggregate available packets and solve an r-dimensional linear system to locally infer the outcome of iterative refinement within the learned response model. Our analysis establishes exact decoding of the model-defined coupled response and its equivalence to the limit of iterative refinement under contraction. Experiments show that SLRC achieves the highest performance among the evaluated methods across 13 Cooperative Navigation, Predator Prey, and SMACv2 scenarios and remains effective under field-of-view (FoV) constraints. Sparse-reward Hallway tasks reveal a limitation, while rank analyses characterize the trade-off between interaction capacity and message payload. SLRC offers a new perspective on cooperative MARL communication by shifting from repeated exchanges to structured interaction inference within one round.

1 INTRODUCTION

While Multi-Agent Reinforcement Learning (MARL) has witnessed significant progress in diverse domains such as unmanned aerial vehicle swarm control (Cui et al., 2019; Shi et al., 2025), traffic signal timing (Wang et al., 2020b), and autonomous driving (Kiran et al., 2021), eliciting efficient and reliable coordination remains a formidable challenge. Partial observability limits each agent's access to relevant team information and complicates cooperative decision-making under uncertainty (Papoudakis et al., 2019). Consequently, decisions based solely on local observations may fail to account for teammates' evolving behavior or consistently support the shared task objective (Xu et al., 2023).

To address these challenges, communication-based MARL incorporates information exchange during decentralized execution (Zhu et al., 2024). By augmenting local observations with messages from teammates, agents can make decisions using information beyond their individual sensory fields. Realizing this benefit depends on both what is communicated and how it is processed. Message content, such as encoded observations and latent intentions, provides additional context (Das et al., 2019; Kim et al., 2020), while receiver-side processing determines how agents incorporate this information and adjust their decision representations (Liu et al., 2020; Guan et al., 2022).

Communication mechanisms differ in exchange depth. Single-round communication (SRC) uses one exchange per decision step, whereas multi-round communication (MRC) allows agents to revise messages using information received in preceding rounds (Das et al., 2019; Wang et al., 2023). These revisions create response dependencies across agents, enabling updates to account for peer responses. Our study in Section 2 shows that additional rounds can improve coordination, although

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.

gains diminish with depth. However, each round depends on completion of preceding exchanges and processing, creating a sequential cost beyond message payload. This motivates our question:

Can senders encode the response dependencies driving iterative refinement, allowing receivers to infer the outcome of such refinement locally after a single logical exchange?

A possible direction is to communicate how an agent's response depends on its peers' adjustments during iterative refinement. By encoding these dependencies in a compact structured message, receivers can infer a self-consistent adjustment through a reduced solve without repeatedly requesting updated proposals. The conceptual example in Appendix A illustrates this idea more concretely.

To realize this idea, we propose **S**tructured **L**ow-**R**ank **C**oupling (SLRC), a communication framework with sender-side interaction encoding and receiver-side coupled inference. SLRC adopts a learned affine response model that separates agent-specific transformations from low-rank interaction dependencies, making self-consistency a linear condition. Each agent generates an intent representation, a positive diagonal operator, and two independent interaction factors. The rank parameter r controls interaction capacity, while separate factors allow asymmetric influences between agents. Each sender converts these quantities into an $r \times r$ coupling component and an r-dimensional response component, forming a packet of dimension $r^2 + r$. After one logical exchange, receivers aggregate the packets and solve an r-dimensional linear system to recover their model-defined collaborative updates locally. Auxiliary objectives guide intent learning and the structure of the interaction factors.

Our main contributions are summarized as follows. (1) We develop a structured rank-r response model and an interaction encoding that packages the information required for coupled decoding into $r^2 + r$ values, enabling local collaborative updates after one logical exchange. (2) We establish exact recovery of the model-defined coupled response given common aggregates and an invertible reduced system, and characterize its relation to iterative response refinement under a contraction condition. (3) We evaluate SLRC in communication-demanding cooperative settings with restricted teammate observations, examining coordination performance, the rank–payload trade-off, component ablations, and hyperparameter sensitivity. SLRC achieves the highest mean performance among the evaluated baselines in all 13 Cooperative Navigation (CN), Predator Prey (PP) (Lowe et al., 2017), and SMACv2 (Ellis et al., 2023) scenarios. Our code is accessible as an open-source repository[1].

2 Can MRC Improve Coordination?

Under the centralized training with decentralized execution (CTDE) paradigm, we use QMIX_COMM to examine the effect of communication depth. QMIX_COMM extends QMIX with Transformer-based message aggregation, where each block processes representations updated by the preceding block. When each block consumes newly updated peer representations, the block depth corresponds to the number of logical communication rounds. Detailed configurations are provided in Appendix B.

We evaluate this setup on `CN_10`, `PP_5_vs_2`, `PP_10_vs_4`, and `protoss_5_vs_5`. Detailed environment configurations are provided in Appendix G. Fig. 1 shows that increasing communication depth generally improves coordination performance across these tasks, while the gains become smaller at greater depths, indicating diminishing returns from additional rounds. These findings motivate modeling how peer responses influence subsequent updates. SLRC captures this dependency at the current decision step through a structured affine response model, whose self-consistent solution supports direct decoding and, under contraction, an iterative-limit interpretation.

3 Related Work

We focus on work most relevant to SLRC, with a broader discussion provided in Appendix C.

Communication Design. CommNet and DIAL established end-to-end learning of inter-agent communication (Sukhbaatar et al., 2016; Foerster et al., 2016). Later methods study when and with whom to communicate or how to reduce redundant information. IC3Net controls communication through individual gates (Singh et al., 2018); NDQ and IMAC introduce information-theoretic objectives for communication efficiency (Wang et al., 2019; 2020a); and I2C and SMS select partners

[1] https://anonymous.4open.science/r/SLRC-5046

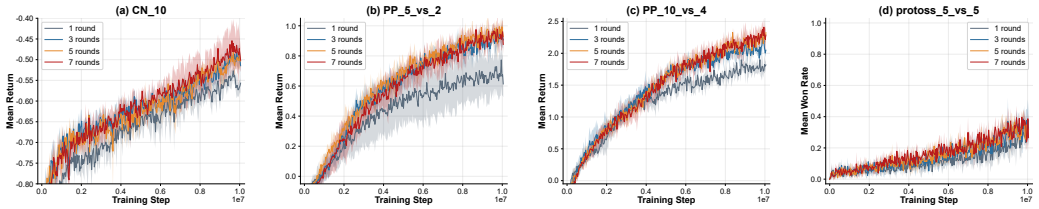


Figure 1: Performance profiles of QMIX_COMM with different communication depths.

according to inferred influence or Shapley message values (Ding et al., 2020; Xue et al., 2022). At the receiver, MASIA learns compact aggregated representations, while T2MAC performs evidence-driven message integration (Guan et al., 2022; Sun et al., 2024). SLRC instead designs message components for receiver-side low-rank coupled inference, rather than generic latent communication or selection.

Iterative and Graph-Based Communication. TarMAC enables multiple communication rounds, while AC2C conditionally invokes a second round to obtain two-hop information (Das et al., 2019; Wang et al., 2023). Graph-based methods propagate information through graph convolutions, communication graphs, or dynamic directed topologies (Jiang et al., 2018; Niu et al., 2021; Zhang et al., 2025). These approaches motivate SLRC's iterative-refinement perspective. SLRC models this refinement through a linear coupling, whose converged response can be recovered by its one-round decoder, without reproducing the nonlinear dynamics of these architectures.

Structured and Low-Rank Interaction Models. Existing work introduces structure into different MARL components. Learning Structured Communication focuses on hierarchical communication topologies (Sheng et al., 2022), while Deep Coordination Graphs use low-rank approximations to model pairwise payoffs in factored joint value functions (Böhmer et al., 2020). Low interaction rank has also been studied as a structural assumption for offline MARL (Zhan et al., 2025). These approaches structure communication topology, joint-value payoffs, or offline function classes. SLRC instead applies low-rank structure to cross-agent interactions for one-round collaborative inference.

4 Preliminaries

4.1 Decentralized Partially Observable Markov Decision Process

We consider fully cooperative MARL under partial observability, modeled as a Decentralized Partially Observable Markov Decision Process (Dec-POMDP) (Bernstein et al., 2002). A Dec-POMDP is defined by the tuple $\mathcal{D} = \langle \mathcal{N}, \mathcal{S}, \mathcal{A}, \Omega, \mathbb{O}, P, R, \gamma \rangle$, where $\mathcal{N} = \{1, \ldots, N\}$ is the set of agents, $\mathcal{S}$ is the global state space, $\mathcal{A} = \prod_{i \in \mathcal{N}} \mathcal{A}_i$ is the joint action space, and $\Omega = \prod_{i \in \mathcal{N}} \Omega_i$ is the joint observation space. At timestep t, the joint action $\mathbf{a}_t = (a_{1,t}, \ldots, a_{N,t})$ induces the next state according to $P(s_{t+1} \mid s_t, \mathbf{a}_t)$ and the joint observation according to $\mathbb{O}(\mathbf{o}_{t+1} \mid s_{t+1}, \mathbf{a}_t)$. All agents receive the shared reward $R(s_t, \mathbf{a}_t)$, and $\gamma \in [0, 1)$ is the discount factor.

Agent i maintains a local action-observation history $\tau_{i,t} = (o_{i,0}, a_{i,0}, \ldots, o_{i,t-1}, a_{i,t-1}, o_{i,t})$ and selects $a_{i,t} \sim \pi_i(\cdot \mid \tau_{i,t})$. The joint policy $\boldsymbol{\pi} = (\pi_1, \ldots, \pi_N)$ is optimized to maximize the discounted return $J(\boldsymbol{\pi}) = \mathbb{E}_{\boldsymbol{\pi}} \left[\sum_{t=0}^{\infty} \gamma^t R(s_t, \mathbf{a}_t) \right]$. Under CTDE, global states and joint trajectories are available during training, whereas each agent acts using only its local history during execution.

4.2 Communication in Dec-POMDP

Communication augments decentralized execution with a message space $\mathcal{P}$. Before selecting an action, agent i constructs a packet $p_{i,t} = \mu_i(\tau_{i,t}) \in \mathcal{P}$ and receives the packets $\{p_{j,t}\}_{j \in \mathcal{N}_{i,t}}$ from a permitted sender set $\mathcal{N}_{i,t} \subseteq \mathcal{N}$. The received packets are processed locally into an internal decision representation $\widetilde{m}_{i,t}$. The resulting action policy is $a_{i,t} \sim \pi_i(\cdot \mid \tau_{i,t}, \widetilde{m}_{i,t})$. The transmitted packet $p_{i,t}$ and internal representation $\widetilde{m}_{i,t}$ may have different dimensions, and only $p_{i,t}$ is transmitted. One exchange of the current packets constitutes one logical communication round. Subsequent aggregation and decoding are performed locally without further message exchange.

5 THEORETICAL ANALYSIS

SLRC formalizes MRC-inspired response refinement through a structured interaction model. At each step, the decoder produces an adjustment Δm_i, and the representation $m_i + \Delta m_i$ is used for action selection. The analysis defines this coupled response, establishes its exact recovery from compact packets after one logical exchange, and characterizes its relation to iterative response refinement.

5.1 STRUCTURED LOW-RANK COUPLING

At decision step t, MRC refinement is abstracted as $x^{(k+1)} = \mathcal{F}_t^{(k)}(x^{(k)})$, where $x^{(k)}$ denotes the joint response at stage k. SLRC models this dependency through the affine surrogate $\mathcal{T}(x) = a + Kx$, separating a baseline response from adjustment-dependent interactions and making self-consistency a linear condition. To enable compact encoding, it imposes $\mathrm{rank}(K) \le r$ and factorizes $K = PQ^\top$. An invertible block-diagonal matrix W parameterizes agent-specific scaling. Setting $f = -Wa$, $U = -WP$, and $V = Q$ yields $\mathcal{T}(x) = -W^{-1}f - W^{-1}UV^\top x$. The following definition formalizes this parameterization and its coupled response.

Definition 1. *Let d denote the representation dimension and r the rank parameter. At a decision step, agent i produces an intent representation $f_i \in \mathbb{R}^d$, a diagonal matrix $w_i \in \mathbb{R}^{d\times d}$ with strictly positive diagonal entries, and interaction factors $u_i, v_i \in \mathbb{R}^{d\times r}$. Define the stacked quantities:*

$$\begin{aligned} f &= [f_1^\top, \dots, f_N^\top]^\top, \quad W = \mathrm{diag}(w_1, \dots, w_N), \\ U &= [u_1^\top, \dots, u_N^\top]^\top, \quad V = [v_1^\top, \dots, v_N^\top]^\top. \end{aligned} \tag{1}$$

Here, $\mathrm{diag}(w_1, \dots, w_N)$ denotes the block-diagonal arrangement of the local matrices. These coefficients remain fixed when evaluating the response map. For a candidate joint adjustment $x = [x_1^\top, \dots, x_N^\top]^\top \in \mathbb{R}^{Nd}$,consider the mapping $\mathcal{T}$ defined by $\mathcal{T}(x) = -W^{-1}f - W^{-1}UV^\top x$. The self-consistent coupled response $\Delta = [\Delta m_1^\top, \dots, \Delta m_N^\top]^\top$ satisfies $\Delta = \mathcal{T}(\Delta)$, meaning that applying the response map again leaves the adjustment unchanged. This condition rearranges to $B\Delta = -f$, where $B = W + UV^\top$. If B is invertible, this system uniquely determines Δ.

The one-round decoder targets this self-consistent adjustment. The coefficient relating the adjustment of agent j to the response of agent i is $-w_i^{-1}u_i v_j^\top$. Here, $v_j^\top$ encodes the adjustment in the r-dimensional interaction space, u_i maps it to agent i, and w_i^{-1} applies agent-specific scaling. Appendix D.1 derives this agent-specific form and the coupled system $B\Delta = -f$. These factorized coefficients determine the affine response capacity formalized below.

Proposition 1. *Let $\mathcal{R}(x) = a + Kx$ be an affine response map on $\mathbb{R}^{Nd}$ with $\mathrm{rank}(K) \le r$. For any matrix W satisfying Definition 1, there exist $f \in \mathbb{R}^{Nd}$ and $U, V \in \mathbb{R}^{Nd\times r}$ such that $\mathcal{R}(x) = \mathcal{T}(x)$ for every $x \in \mathbb{R}^{Nd}$. If $V = U$, the factorized component $UV^\top$ is restricted to symmetric positive-semidefinite matrices with rank at most r.*

Proposition 1 shows that separate factors preserve arbitrary affine response dependencies of rank at most r, including asymmetric interactions. Its proof is provided in Appendix D.2.

5.2 EXACT DECODING FROM COMPACT PACKETS

The structure in Definition 1 enables decoding through an r-dimensional linear system. The sender-side packet and receiver-side decoding rule are defined below.

Definition 2. *To decode the response in Definition 1, agent i constructs two packet components:*

$$z_i = v_i^\top w_i^{-1} u_i \in \mathbb{R}^{r\times r}, \qquad b_i = v_i^\top w_i^{-1} f_i \in \mathbb{R}^r. \tag{2}$$

These components form the transmitted packet $p_i = [\mathrm{vec}(z_i)^\top, b_i^\top]^\top \in \mathbb{R}^{r^2+r}$, where $\mathrm{vec}(\cdot)$ vectorizes a matrix. After one logical exchange, each receiver aggregates the components from all N agents, including its own, to form $S = I_r + \sum_{j=1}^N z_j$ and $\bar{b} = \sum_{j=1}^N b_j$, where I_r is the $r \times r$ identity matrix. If S is invertible, the receiver computes the joint coupling coefficient $c = S^{-1}\bar{b} \in \mathbb{R}^r$ and decodes its adjustment $\Delta m_i = -w_i^{-1}f_i + w_i^{-1}u_i c$.

Appendix D.3 derives the packet components and decoding rule through the Woodbury identity. The derivation also establishes that $c = -V^\top\Delta$, linking the reduced coefficient to the coupled response.

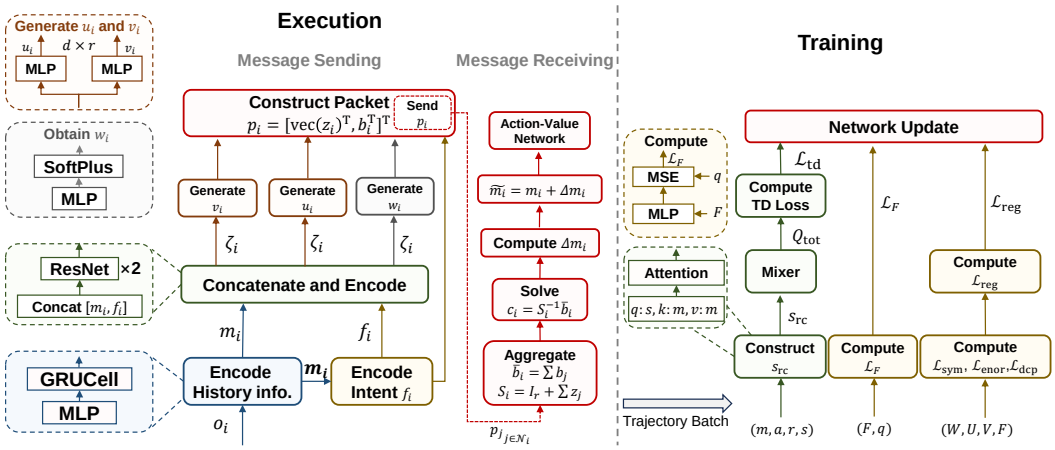


Figure 2: SLRC with structured packet encoding, coupled decoding, and centralized training.

Proposition 2. *Suppose all N receivers have the common aggregate quantities S and $\bar{b}$ in Definition 2, and S is invertible. Then B is invertible, and the decoded adjustments $\Delta = [\Delta m_1^\top, \ldots, \Delta m_N^\top]^\top$ form the unique solution of $B\Delta = -f$.*

The proof is provided in Appendix D.3. Under these conditions, one logical exchange followed by a local r-dimensional solve recovers the coupled response exactly.

5.3 Connection to Iterative Response Refinement

The structured response map also provides an iterative interpretation of the coupled response. With f, W, U, and V fixed at a decision step, its repeated application defines:

$$\Delta^{(k+1)} = \mathcal{T}(\Delta^{(k)}) = -W^{-1}f - W^{-1}UV^\top\Delta^{(k)}. \tag{3}$$

The matrix $A = W^{-1}UV^\top$ determines how errors propagate during refinement. The following proposition characterizes convergence of this process.

Proposition 3. *Suppose there exists $q \in [0, 1)$ such that $\|A\|_2 \le q$, where $\|\cdot\|_2$ denotes the matrix spectral norm. Then B and S are invertible. For any initialization $\Delta^{(0)} \in \mathbb{R}^{Nd}$, the response process in Eq. (3) converges to the unique coupled response $\Delta = -B^{-1}f$. Moreover, its error satisfies $\|\Delta^{(k)} - \Delta\|_2 \le q^k\|\Delta^{(0)} - \Delta\|_2$ for every $k \ge 0$. When the receivers have the common aggregates, this limit equals the response decoded by SLRC.*

The proof appears in Appendix D.4. The contraction condition establishes iterative-limit interpretation, while exact decoding in Proposition 2 requires only the common aggregates and invertibility of S.

6 Methods

Fig. 2 and Algorithm 2 in Appendix F illustrate the architecture and execution procedure, respectively. SLRC consists of sender-side structured packet encoding and receiver-side coupled decoding. Senders transform local intent and interaction factors into compact packets, while receivers aggregate the available packets and compute their collaborative representation updates locally without further message exchange. The complete framework is trained end-to-end under the CTDE paradigm.

6.1 Sender: Structured Interaction Encoding

During execution, agent i encodes its local action-observation history $\tau_{i,t}$ through a GRU to obtain the recurrent representation $m_{i,t}$. An intent encoder then maps it to $f_{i,t} = G_F(m_{i,t}) \in \mathbb{R}^d$. To encourage $f_{i,t}$ to retain decision-relevant information, an auxiliary value decoder G_{dec} predicts the individual action-value vector. The training objective of the encoder $G_F(\cdot)$ is:

$$\mathcal{L}_F = \mathbb{E}_{\mathcal{B}}\left[\|G_{\mathrm{dec}}(f_{i,t}) - \mathrm{sg}(\mathbf{q}_{i,t})\|_2^2\right]. \tag{4}$$

where the expectation is taken over samples from replay buffer $\mathcal{B}$, $\mathbf{q}_{i,t}$ denotes the action-value vector, and $\mathrm{sg}(\cdot)$ denotes stop-gradient. Subsequently, the recurrent representation $m_{i,t}$ and intent representation $f_{i,t}$ are then concatenated and processed by an encoding network to obtain $\zeta_{i,t} = G_E([m_{i,t}, f_{i,t}])$. Based on $\zeta_{i,t}$, three output heads generate the diagonal coefficients $d_{i,t} = \mathrm{softplus}(G_W(\zeta_{i,t})) + \epsilon \mathbf{1}_d \in \mathbb{R}^d$ and the interaction factors $u_{i,t} = G_U(\zeta_{i,t}) \in \mathbb{R}^{d\times r}$ and $v_{i,t} = G_V(\zeta_{i,t}) \in \mathbb{R}^{d\times r}$. The local transformation is defined as $w_{i,t} = \mathrm{Diag}(d_{i,t})$, where $\epsilon > 0$ ensures that its diagonal entries are strictly positive. The separate parameterizations of $u_{i,t}$ and $v_{i,t}$ preserve the ability to model asymmetric interactions, as characterized in Proposition 1.

Having generated the intent representation and interaction factors, agent i converts them into the components required for decoding. Transmitting $f_{i,t}$, diagonal entries of $w_{i,t}$, and factors $u_{i,t}$ and $v_{i,t}$ would require $2d + 2dr$ values. Instead, following Definition 2, the agent constructs $z_{i,t} = v_{i,t}^\top w_{i,t}^{-1} u_{i,t} \in \mathbb{R}^{r\times r}$ and $b_{i,t} = v_{i,t}^\top w_{i,t}^{-1} f_{i,t} \in \mathbb{R}^r$. The packet is $p_{i,t} = [\mathrm{vec}(z_{i,t})^\top, b_{i,t}^\top]^\top \in \mathbb{R}^{r^2+r}$. Therefore, its dimension is controlled by r and independent of representation dimension d. Proposition 2 establishes these components are sufficient to recover the coupled response.

During training, SLRC employs three auxiliary regularizers to guide the interaction factors. Let n_t denote the number of active agents at timestep t, and let $U^t, V^t \in \mathbb{R}^{n_t d\times r}$ stack their interaction factors. Define the reduced matrices $\Psi_u^t = (U^t)^\top U^t$, $\Psi_v^t = (V^t)^\top V^t$, and $\Omega^t = (V^t)^\top U^t$. A soft symmetry objective encourages reciprocal structure in the factorized interaction component:

$$\mathcal{L}_{\mathrm{sym}} = 4\left(\mathbb{E}_{t\sim\mathcal{B}}[n_t(n_t-1)]\right)^{-1} \mathbb{E}_{t\sim\mathcal{B}}\left[\mathrm{Tr}(\Psi_u^t \Psi_v^t) - \mathrm{Tr}\left((\Omega^t)^2\right)\right]. \tag{5}$$

Here, $\mathrm{Tr}(\cdot)$ denotes the matrix trace. Both expectations use the same sampled timesteps, and the loss is zero when the denominator vanishes. Given U^t, V^t, the trace formulation derived in Appendix E uses $r \times r$ matrices instead of the $n_t d \times n_t d$ interaction matrix, reducing intermediate storage from $\mathcal{O}(n_t^2 d^2)$ to $\mathcal{O}(r^2)$. The penalty encourages reciprocal interactions without enforcing exact symmetry.

$$\mathcal{L}_{\mathrm{enor}} = \mathbb{E}_{(i,t)\sim\mathcal{B}}\left[\left\|v_{i,t}^\top u_{i,t} - I_r\right\|_{\mathrm{F}}^2\right]. \tag{6}$$

Finally, SLRC encourages the agent-specific and interaction pathways to capture complementary directions, thereby reducing directional overlap between their learned features. Their features are defined as $s_{i,t}^{\mathrm{local}} = w_{i,t} f_{i,t}$ and $s_{i,t}^{\mathrm{inter}} = u_{i,t} \sum_{j=1}^N (v_{j,t})^\top f_{j,t}$. Their squared cosine similarity gives:

$$\mathcal{L}_{\mathrm{dcp}} = \mathbb{E}_{(i,t)\sim\mathcal{B}}\left[\mathrm{sim}_{\cos}^2\left(s_{i,t}^{\mathrm{local}}, s_{i,t}^{\mathrm{inter}}\right)\right], \tag{7}$$

where $\mathrm{sim}_{\cos}(\cdot,\cdot)$ denotes cosine similarity with numerical stabilization. The three objectives are combined as $\mathcal{L}_{\mathrm{reg}} = \lambda_{\mathrm{sym}}\mathcal{L}_{\mathrm{sym}} + \lambda_{\mathrm{enor}}\mathcal{L}_{\mathrm{enor}} + \lambda_{\mathrm{dcp}}\mathcal{L}_{\mathrm{dcp}}$. This regularization provides soft structural guidance for learning the factors while preserving flexibility in the learned interaction model, without explicitly enforcing the invertibility or contraction conditions used in the theoretical analysis. Through soft symmetry and channel-consistency constraints, SLRC encourages a coupling structure favorable to invertibility of the reduced coupling matrix S, as explained in Remark 1.

6.2 Receiver: Coupled Interaction Decoding

After one logical exchange, receiver i aggregates the packets available from its sender set $\mathcal{N}_{i,t}$ to form the reduced coupling matrix $S_{i,t} = I_r + \sum_{j\in\mathcal{N}_{i,t}} z_{j,t}$ and the aggregate response component $\bar{b}_{i,t} = \sum_{j\in\mathcal{N}_{i,t}} b_{j,t}$. Assuming that $S_{i,t}$ is nonsingular, receiver i computes the reduced coupling coefficient $c_{i,t}$ by solving $S_{i,t} c_{i,t} = \bar{b}_{i,t}$ and decodes its collaborative adjustment as follows:

$$\Delta m_{i,t} = -w_{i,t}^{-1} f_{i,t} + w_{i,t}^{-1} u_{i,t} c_{i,t}. \tag{8}$$

Agent i updates its recurrent representation using the decoded adjustment as $\widetilde{m}_{i,t} = m_{i,t} + \Delta m_{i,t}$. The local action-value network then produces $\mathbf{q}_{i,t} = G_Q(\widetilde{m}_{i,t})$ for action selection.

6.3 Trajectory-Based Credit Assignment

Under CTDE, SLRC employs a QMIX-style value-decomposition learner (Rashid et al., 2020). During centralized training, the learned projections W_Q, W_K, and W_V map the global state s_t and recurrent representation $m_{i,t}$ to the query, key, and value, respectively. For agent i, the compatibility

score is $(W_Q s_t)^\top W_K m_{i,t}$. Applying softmax normalization across agents yields the attention weight $\alpha_{i,t}$, with $\sum_{i=1}^{N} \alpha_{i,t} = 1$. The resulting state-conditioned team representation $s_{\mathrm{rc},t} = \sum_{i=1}^{N} \alpha_{i,t} W_V m_{i,t}$ is used to condition the centralized mixing network.

For the selected action $a_{i,t}$, let $q_{i,t} = [\mathbf{q}_{i,t}]_{a_{i,t}}$ denote its action value. The mixer computes the joint action value as $Q_{\mathrm{tot},t} = G_{\mathrm{mix}}(q_{1,t}, \ldots, q_{N,t}; s_{\mathrm{rc},t})$. Hypernetworks conditioned on $s_{\mathrm{rc},t}$ generate nonnegative mixing weights, imposing $\partial Q_{\mathrm{tot},t} / \partial q_{i,t} \geq 0$ for agent i. This monotonicity preserves consistency between centralized joint-value maximization and decentralized greedy action selection.

6.4 Overall Optimization Objective

Following the CTDE paradigm, SLRC minimizes a squared temporal-difference (TD) loss to optimize the parameters θ of the agent and mixing networks:

$$\mathcal{L}_{\mathrm{td}} = \mathbb{E}_{\mathcal{B}} \left[\left(y_t^{\mathrm{tot}} - Q_{\mathrm{tot}} \left(\boldsymbol{\tau}_t, \mathbf{a}_t, s_{\mathrm{rc},t}; \theta \right) \right)^2 \right]. \tag{9}$$

The target is computed using the target-network parameters θ^- as $y_t^{\mathrm{tot}} = r_t + \gamma \max_{\mathbf{a}'} Q_{\mathrm{tot}}(\boldsymbol{\tau}_{t+1}, \mathbf{a}', s_{\mathrm{rc},t+1}; \theta^-)$. The TD loss is jointly optimized with the intent-generation objective in Eq. (4) and the factor regularization objective $\mathcal{L}_{\mathrm{reg}}$ defined above:

$$\mathcal{L}_{\mathrm{tot}} = \mathcal{L}_{\mathrm{td}} + \lambda_F \mathcal{L}_F + \lambda_{\mathrm{reg}} \mathcal{L}_{\mathrm{reg}}. \tag{10}$$

Here, $\lambda_F > 0$ and $\lambda_{\mathrm{reg}} > 0$ balance the reinforcement-learning objective with the auxiliary signals for learning the intent representations and structured interaction factors.

7 Experiments

Our experiments evaluate SLRC's coordination performance against competitive baselines across standard, FoV-constrained, and sparse-reward settings, together with its message-payload efficiency, component contributions, and sensitivity to the rank parameter r and other key hyperparameters.

7.1 Experimental Setups

We evaluate SLRC on Hallway, PP, CN, and SMACv2. To construct communication-demanding settings, teammate-related information is removed from local observations, requiring agents to obtain information through communication. Environment configurations and FoV settings are provided in Appendix G. As shown in Fig. 9 in Appendix G.5, removing communication causes substantial performance degradation, confirming that these tasks provide tests of coordination. We compare SLRC with the multi-round communication methods TGCNet (Zhang et al., 2025), QMIX_COMM described in Appendix B, and MAGIC (Niu et al., 2021), the single-round communication methods T2MAC (Sun et al., 2024), SMS (Xue et al., 2022), and MASIA (Guan et al., 2022), and the no-communication baseline QMIX (Rashid et al., 2020). Hyperparameter settings for SLRC and all baselines are provided in Appendix H. All results are averaged over four runs with random seeds, and the shaded regions in the training curves indicate 95% confidence intervals.

7.2 Main Results

Table 2 and Fig. 3 show that SLRC achieves the highest mean performance among the methods in all 13 CN, PP, and SMACv2 scenarios. On `CN_4` and `CN_10`, SLRC obtains returns of -0.43 and -0.47, exceeding the strongest baselines by 0.09 and 0.03, respectively. On `PP_5_vs_2` and `PP_10_vs_4`, it achieves 1.35 and 3.29, compared with 0.91 and 2.06 for QMIX_COMM. Across the nine SMACv2 scenarios, SLRC surpasses the strongest baseline by 9.4–16.2 percentage points, demonstrating gains across races and team sizes. Fig. 11 further shows that SLRC coordinates multi-front advances, concentrated fire, formation adjustment, and target reassignment in `protoss_10_vs_10`. These behaviors demonstrate that the allied units maintain differentiated spatial roles while adapting their attack to the battle, rather than moving toward the same target. The analyses of representations after receiver-side message aggregation in Figs. 12 and 13, together with Fig. 4, reveal differences among the communication methods. TGCNet produces highly aligned, nearly rank-one representations, indicating representational homogenization, while QMIX_COMM maintains an intermediate level

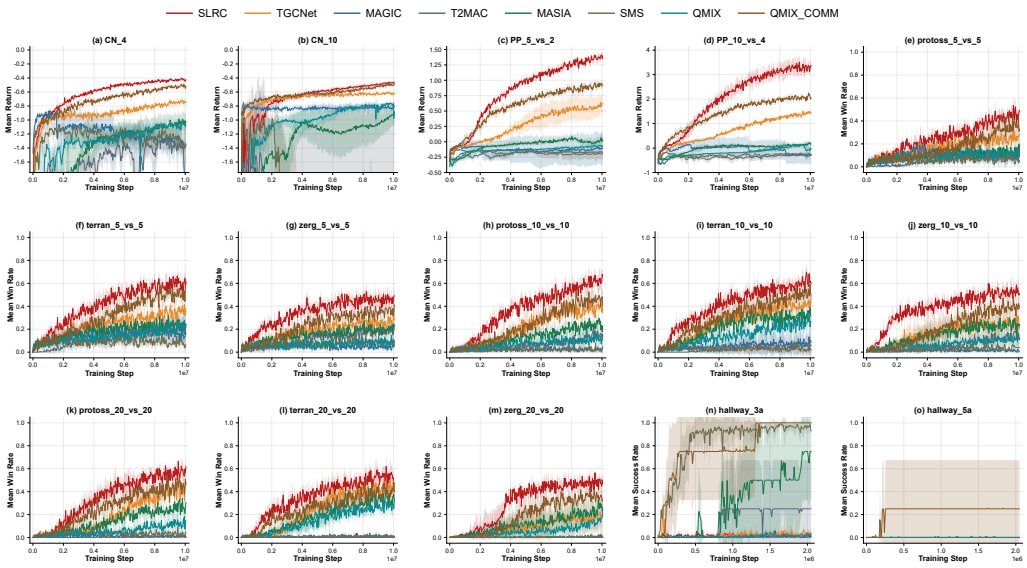


Figure 3: Overall performance across 15 cooperative scenarios. Panels (a)–(b) show Cooperative Navigation, (c)–(d) show Predator Prey, (e)–(m) show SMACv2, and panels (n)–(o) show Hallway.

of diversity. In contrast, SLRC exhibits lower mean similarity, greater pairwise dispersion, more negatively aligned pairs, and higher effective rank. When viewed alongside its superior performance in Table 2, this pattern indicates that SLRC coordinates agents while preserving task-dependent, multidirectional representations rather than enforcing a homogeneous shared representation.

Under FoV constraints, SLRC maintains its advantage over QMIX_COMM and TGCNet when communicating with visible neighbors, as shown in Fig. 14. SLRC also trains more efficiently. Table 3 reports the shortest wall-clock training time among communication-based methods, aided by one-round packet exchange and a reduced receiver-side solve. However, Hallway reveals a sparse-reward limitation: SLRC achieves success rates of 1.5% and 0.0% on `hallway_3a` and `hallway_5a`, compared with 100.0% and 25.0% for QMIX_COMM. Because rewards occur only when all agents reach their goals simultaneously, SLRC receives limited guidance for learning intent representations and interaction factors. Interaction decoding therefore does not resolve exploration under sparse feedback, motivating future work on auxiliary signals and exploration mechanisms.

7.3 Effect of Coupling Rank

The rank r controls interaction capacity and packet dimension $r^2 + r$. Fig. 15 and Table 4 show that, at $r = 4$ and $r = 8$, each agent transmits 20 and 72 values, respectively, in one exchange, whereas QMIX_COMM transmits 128 values per round over three rounds. On `CN_10`, `PP_10_vs_4`, and `protoss_10_vs_10`, SLRC obtains -0.57, 2.49, and 51.8% at $r = 4$, and -0.53, 2.96, and 50.9% at $r = 8$. QMIX_COMM obtains -0.50, 2.06, and 45.7% on these tasks. Both ranks outperform it on PP and Protoss while remaining competitive on CN, despite fewer exchanges and smaller packets.

As r increases, performance on `CN_10` improves from -0.57 at $r = 4$ to -0.45 at $r = 48$. However, `PP_10_vs_4` and `protoss_10_vs_10` peak at $r = 32$ with 3.29 and 61.9%, before decreasing to 3.03 and 56.7% at $r = 48$. Increasing interaction capacity thus yields diminishing returns and does not consistently improve coordination beyond a sufficient rank. Meanwhile, the packet dimension grows quadratically from 20 at $r = 4$ to 2352 at $r = 48$. This growth constitutes a limitation of high-rank SLRC configurations and motivates further compression of the $r \times r$ coupling component.

7.4 Ablation Studies

Fig. 16 evaluates the communication module, intent objective $\mathcal{L}_F$, and factor regularization objective $\mathcal{L}_{\text{reg}}$. Removing communication causes the largest degradation across all three tasks. Compared with

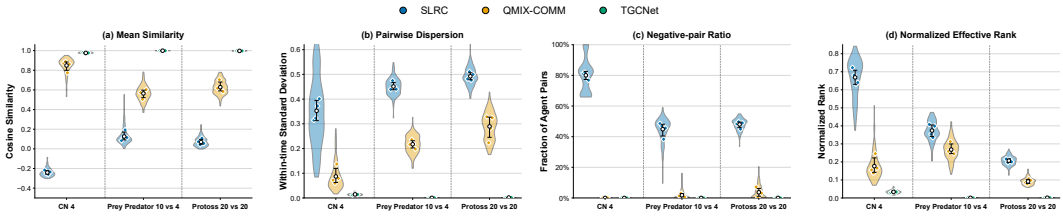


Figure 4: Representation diversity of $\widetilde{m}_i$ across CN 4, Prey Predator 10 vs 4, and Protoss 20 vs 20.

the no-communication variant, SLRC improves performance by 73.7% on `PP_10_vs_4` and 91.2% on `zerg_5_vs_5`. The visualizations in Figs. 18 and 17 further show that communication supports coordinated target assignment and encirclement in `PP_10_VS_4`. Among the auxiliary objectives, removing $\mathcal{L}_{\text{reg}}$ destabilizes training and reduces the score on `PP_10_vs_4` by 34.3% to 2.16 ± 1.89, confirming the importance of structural guidance for learning interaction factors. Removing $\mathcal{L}_F$ has a smaller effect, indicating that it mainly provides auxiliary supervision for the intent representation.

Fig. 19 examines the components of $\mathcal{L}_{\text{reg}}$. Removing any term causes moderate changes, suggesting partial redundancy among the regularizers. When used alone, $\mathcal{L}_{\text{sym}}$ and $\mathcal{L}_{\text{enor}}$ are more stable than $\mathcal{L}_{\text{dcp}}$ because they guide the algebraic relations between U and V. Specifically, $\mathcal{L}_{\text{sym}}$ encourages reciprocal coupling, while $\mathcal{L}_{\text{enor}}$ maintains compatible reduced channels. In contrast, $\mathcal{L}_{\text{dcp}}$ separates the local and interaction pathways. These results emphasize algebraic constraints between U and V for preserving the factorized structure, while $\mathcal{L}_{\text{dcp}}$ provides complementary pathway separation.

7.5 Hyperparameter Sensitivity

Fig. 20 and Table 5 examine SLRC's coefficients by varying one coefficient at a time while fixing the others. The coefficients λ_{sym}, λ_{enor}, and λ_{dcp} control the contributions of the three factor regularizers, while λ_{reg} determines their weight and λ_F controls intent supervision. SLRC remains stable over broad ranges of the coefficients. Both λ_{sym} and λ_{enor} support effective learning over $[0.01, 1.0]$, while the stable ranges of λ_{dcp} and λ_{reg} are $[0.1, 20.0]$ and $[0.05, 20.0]$, respectively. Performance varies moderately across these ranges in three environments. This robustness agrees with the ablation results, where removing one regularizer is less detrimental than removing their combined objective.

The intent coefficient λ_F also provides a broad reliable range of $[0.02, 1.0]$, although substantially larger values can destabilize training. As shown in (d), (i), and (n) of Fig. 20, setting λ_F to 5.0 or 10.0 reduces the win rate on `zerg_5_vs_5` to 11.2% and 5.7%, respectively. These results suggest that excessive intent supervision can interfere with RL learning. Overall, SLRC is robust across broad coefficient ranges, provided that the intent objective retains a moderate weight.

8 Conclusion and Future Work

Summary. We introduced SLRC, a one-round communication framework inspired by iterative refinement in MRC. SLRC models agent-specific transformations and cross-agent interactions through a structured rank-r coupling. Each agent replaces the direct transmission of $2d + 2dr$ values with a packet of dimension $r^2 + r$. After one logical exchange, receivers aggregate the available packets and solve an r-dimensional linear system locally to recover their collaborative updates. Our analysis establishes exact decoding and its relation to convergent iterative refinement. Experiments on CN, PP, and SMACv2 demonstrate strong performance across 13 scenarios, while rank analyses characterize the trade-off between interaction capacity and message payload. SLRC shifts communication design from repeated representation exchange toward structured interaction inference after a single exchange.

Limitations and Future Work. SLRC has two main limitations. First, its performance degrades on sparse-reward Hallway tasks, indicating that effective interaction learning remains difficult with limited task feedback. Second, the packet dimension $r^2 + r$ grows quadratically with r, increasing the communication cost of high-rank configurations. Future work will investigate auxiliary signals and exploration mechanisms for sparse-reward coordination, together with further factorization, sparsification, and compression of the r^2-dimensional coupling component.

AI USE STATEMENT

In this work, we used generative AI tools for grammar correction, occasional text rewriting, and data visualization. We did not use generative AI tools to generate scientific ideas, design experiments, analyze data, implement algorithms, or produce experimental results. The remaining required disclosure tasks were not applicable to this work. We also occasionally used generative AI tools to support bibliographic searches. All AI-assisted material was carefully reviewed and revised by the authors. We verified suggested references against their original sources and checked AI-assisted visualizations against the underlying experimental data. We take full responsibility for the final content of this work, including all text, claims, and artifacts produced with the aid of generative AI.

REPRODUCIBILITY STATEMENT

The SLRC execution and training procedures are summarized in Algorithm 2 and Appendix F, with source code and experimental configurations available at https://anonymous.4open.science/r/SLRC-5046. The derivations and proofs supporting the theoretical results are provided in Appendix D. Appendix G describes the environments, reward functions, and FoV settings, while Appendix H reports the implementation details and hyperparameters of SLRC and the baselines. Additional results, representation analyses, ablations, and hyperparameter studies are provided in Appendices I, L, and M. Together, these materials document the implementation and evaluation procedures required to reproduce our results.

REFERENCES

Daniel S Bernstein, Robert Givan, Neil Immerman, and Shlomo Zilberstein. The complexity of decentralized control of markov decision processes. *Mathematics of operations research*, 27(4): 819–840, 2002.

Wendelin Böhmer, Vitaly Kurin, and Shimon Whiteson. Deep coordination graphs. In *International Conference on Machine Learning*, pp. 980–991. PMLR, 2020.

Tianshu Chu, Sandeep Chinchali, and Sachin Katti. Multi-agent reinforcement learning for networked system control. *arXiv preprint arXiv:2004.01339*, 2020.

Jingjing Cui, Yuanwei Liu, and Arumugam Nallanathan. Multi-agent reinforcement learning-based resource allocation for uav networks. *IEEE Transactions on Wireless Communications*, 19(2): 729–743, 2019.

Abhishek Das, Théophile Gervet, Joshua Romoff, Dhruv Batra, Devi Parikh, Mike Rabbat, and Joelle Pineau. Tarmac: Targeted multi-agent communication. In *International Conference on machine learning*, pp. 1538–1546. PMLR, 2019.

Ziluo Ding, Tiejun Huang, and Zongqing Lu. Learning individually inferred communication for multi-agent cooperation. *Advances in neural information processing systems*, 33:22069–22079, 2020.

Benjamin Ellis, Jonathan Cook, Skander Moalla, Mikayel Samvelyan, Mingfei Sun, Anuj Mahajan, Jakob Foerster, and Shimon Whiteson. Smacv2: An improved benchmark for cooperative multi-agent reinforcement learning. *Advances in Neural Information Processing Systems*, 36:37567–37593, 2023.

Jakob Foerster, Ioannis Alexandros Assael, Nando De Freitas, and Shimon Whiteson. Learning to communicate with deep multi-agent reinforcement learning. *Advances in neural information processing systems*, 29, 2016.

Zihong Gao, Changsheng Qu, Yuanhui Hao, and Liangjun Ke. Communication in multiagent reinforcement learning via counterfactual message value. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 2025.

Zihong Gao, Hongjian Liang, Yuanhui Hao, Lei Hao, and Liangjun Ke. Grdc: A unified graph-driven framework for role discovery and communication in multi-agent reinforcement learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pp. 29441–29449, 2026.

Cong Guan, Feng Chen, Lei Yuan, Chenghe Wang, Hao Yin, Zongzhang Zhang, and Yang Yu. Efficient multi-agent communication via self-supervised information aggregation. *Advances in Neural Information Processing Systems*, 35:1020–1033, 2022.

Jiechuan Jiang, Chen Dun, Tiejun Huang, and Zongqing Lu. Graph convolutional reinforcement learning. *arXiv preprint arXiv:1810.09202*, 2018.

Woojun Kim, Jongeui Park, and Youngchul Sung. Communication in multi-agent reinforcement learning: Intention sharing. In *International conference on learning representations*, 2020.

B Ravi Kiran, Ibrahim Sobh, Victor Talpaert, Patrick Mannion, Ahmad A Al Sallab, Senthil Yogamani, and Patrick Pérez. Deep reinforcement learning for autonomous driving: A survey. *IEEE transactions on intelligent transportation systems*, 23(6):4909–4926, 2021.

Pei Li, Yongkang Zhang, Zhonglin Lv, Jiangjin Yin, et al. Caic: Congestion-aware intent communication for multi-agent reinforcement learning. In *Forty-Second Annual Conference on Uncertainty in Artificial Intelligence*, 2026.

Sheng Li, Jayesh K Gupta, Peter Morales, Ross Allen, and Mykel J Kochenderfer. Deep implicit coordination graphs for multi-agent reinforcement learning. *arXiv preprint arXiv:2006.11438*, 2020.

Yong Liu, Weixun Wang, Yujing Hu, Jianye Hao, Xingguo Chen, and Yang Gao. Multi-agent game abstraction via graph attention neural network. In *Proceedings of the AAAI conference on artificial intelligence*, volume 34, pp. 7211–7218, 2020.

Ryan Lowe, Yi Wu, Aviv Tamar, Jean Harb, Pieter Abbeel, and Igor Mordatch. Multi-agent actor-critic for mixed cooperative-competitive environments. *Neural Information Processing Systems (NIPS)*, 2017.

Hangyu Mao, Zhengchao Zhang, Zhen Xiao, Zhibo Gong, and Yan Ni. Learning agent communication under limited bandwidth by message pruning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pp. 5142–5149, 2020.

Yaru Niu, Rohan Paleja, and Matthew Gombolay. Multi-agent graph-attention communication and teaming. In *Proceedings of the 20th International Conference on Autonomous Agents and MultiAgent Systems*, pp. 964–973, 2021.

Georgios Papoudakis, Filippos Christianos, Arrasy Rahman, and Stefano V Albrecht. Dealing with non-stationarity in multi-agent deep reinforcement learning. *arXiv preprint arXiv:1906.04737*, 2019.

Tabish Rashid, Mikayel Samvelyan, Christian Schroeder De Witt, Gregory Farquhar, Jakob Foerster, and Shimon Whiteson. Monotonic value function factorisation for deep multi-agent reinforcement learning. *Journal of Machine Learning Research*, 21(178):1–51, 2020.

Junjie Sheng, Xiangfeng Wang, Bo Jin, Junchi Yan, Wenhao Li, Tsung-Hui Chang, Jun Wang, and Hongyuan Zha. Learning structured communication for multi-agent reinforcement learning. *Autonomous Agents and Multi-Agent Systems*, 36(2):50, 2022.

Rongye Shi, Xin Yu, Yandong Wang, Yongkai Tian, Zhenyu Liu, Wenjun Wu, Xiao-Ping Zhang, and Manuela M Veloso. Symmetry-informed marl: A decentralized and cooperative uav swarm control approach for communication coverage. *IEEE Transactions on Mobile Computing*, 2025.

Amanpreet Singh, Tushar Jain, and Sainbayar Sukhbaatar. Learning when to communicate at scale in multiagent cooperative and competitive tasks. *arXiv preprint arXiv:1812.09755*, 2018.

Sainbayar Sukhbaatar, Rob Fergus, et al. Learning multiagent communication with backpropagation. *Advances in neural information processing systems*, 29, 2016.

Chuxiong Sun, Zehua Zang, Jiabao Li, Jiangmeng Li, Xiao Xu, Rui Wang, and Changwen Zheng. T2mac: Targeted and trusted multi-agent communication through selective engagement and evidence-driven integration. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pp. 15154–15163, 2024.

Rundong Wang, Xu He, Runsheng Yu, Wei Qiu, Bo An, and Zinovi Rabinovich. Learning efficient multi-agent communication: An information bottleneck approach. In *International conference on machine learning*, pp. 9908–9918. PMLR, 2020a.

Tonghan Wang, Jianhao Wang, Chongyi Zheng, and Chongjie Zhang. Learning nearly decomposable value functions via communication minimization. *arXiv preprint arXiv:1910.05366*, 2019.

Xiaoqiang Wang, Liangjun Ke, Zhimin Qiao, and Xinghua Chai. Large-scale traffic signal control using a novel multiagent reinforcement learning. *IEEE transactions on cybernetics*, 51(1):174–187, 2020b.

Xuefeng Wang, Xinran Li, Jiawei Shao, and Jun Zhang. Ac2c: Adaptively controlled two-hop communication for multi-agent reinforcement learning. *arXiv preprint arXiv:2302.12515*, 2023.

Zhiwei Xu, Bin Zhang, Dapeng Li, Zeren Zhang, Guangchong Zhou, Hao Chen, and Guoliang Fan. Consensus learning for cooperative multi-agent reinforcement learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, pp. 11726–11734, 2023.

Di Xue, Lei Yuan, Zongzhang Zhang, and Yang Yu. Efficient multi-agent communication via shapley message value. In *Proceedings of the Thirty-First International Joint Conference on Artificial Intelligence, IJCAI-22*, pp. 578–584, 2022.

Wenhao Zhan, Scott Fujimoto, Zheqing Zhu, Jason Lee, Daniel Jiang, and Yonathan Efroni. Exploiting structure in offline multi-agent rl: The benefits of low interaction rank. In *International Conference on Learning Representations*, volume 2025, pp. 61648–61682, 2025.

Sai Qian Zhang, Qi Zhang, and Jieyu Lin. Efficient communication in multi-agent reinforcement learning via variance based control. *Advances in neural information processing systems*, 32, 2019.

Sai Qian Zhang, Qi Zhang, and Jieyu Lin. Succinct and robust multi-agent communication with temporal message control. *Advances in neural information processing systems*, 33:17271–17282, 2020.

Zhuohui Zhang, Bin He, Bin Cheng, and Gang Li. Bridging training and execution via dynamic directed graph-based communication in cooperative multi-agent systems. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pp. 23395–23403, 2025.

Changxi Zhu, Mehdi Dastani, and Shihan Wang. A survey of multi-agent deep reinforcement learning with communication. *Autonomous Agents and Multi-Agent Systems*, 38(1):4, 2024.

SLRC: Structured Low-Rank Coupling for One-Round Communication in Cooperative Multi-Agent Reinforcement Learning (Appendix)

CONTENTS

A Conceptual Motivation for SLRC **15**

B Implementation Details of QMIX_COMM **15**

C Related Work **16**

C.1 Learned Message Content, Selection, and Aggregation 16

C.2 Iterative and Graph-Based Communication 17

C.3 Structured and Low-Rank Interaction Models 17

D Additional Theoretical Analysis **18**

D.1 Derivation of the Structured Coupled Response 18

D.2 Proof of Proposition 1 19

D.3 Derivation and Exactness of Compact Decoding 19

D.4 Proof of Proposition 3 20

E Derivation of $\mathcal{L}_{\text{sym}}$ **21**

F Implementation Details of SLRC **22**

G Detailed Environment Configurations **22**

G.1 Hallway 22

G.2 Multi-Agent Particle Environment 24

G.2.1 Cooperative Navigation 24

G.2.2 Predator Prey 24

G.3 SMACv2 25

G.4 Field-of-View-Constrained Communication 26

G.5 Communication Necessity under Restricted Observations 26

H Algorithm Details for Baselines and SLRC **27**

I Additional Results and Analyses **28**

I.1 More Detailed Results 28

I.2 Representation Structure Analysis 29

I.3 Wall-Clock Training Time 31

J Analysis of Communication within Field of View **31**

K Effect of Coupling Rank **31**

L Ablation Studies **33**

M Hyperparameter Sensitivity **35**

A Conceptual Motivation for SLRC

Fig. 5 illustrates the intuition behind SLRC through a price-negotiation example. In conventional multi-round interaction, each participant communicates a current proposal and then revises it according to the response received from the other participant. As shown in Fig. 5(a), reaching an agreement therefore requires several sequential exchanges.

Fig. 5(b) illustrates a different communication strategy. Instead of transmitting only a current proposal, each participant communicates the proposal together with a compact description of how it would respond to further interaction. Once these descriptions have been exchanged, both participants can combine them and locally infer the same consistent outcome without transmitting every intermediate proposal.

This example captures the central motivation of SLRC. Each agent encodes its contribution to the collaborative response through the coupling component z_i and response component b_i. After one logical exchange, receivers aggregate these components and solve the resulting reduced interaction equation locally. The example is conceptual, while the exact packet construction and decoding rule are defined in Section 6 and analyzed in Section 5.

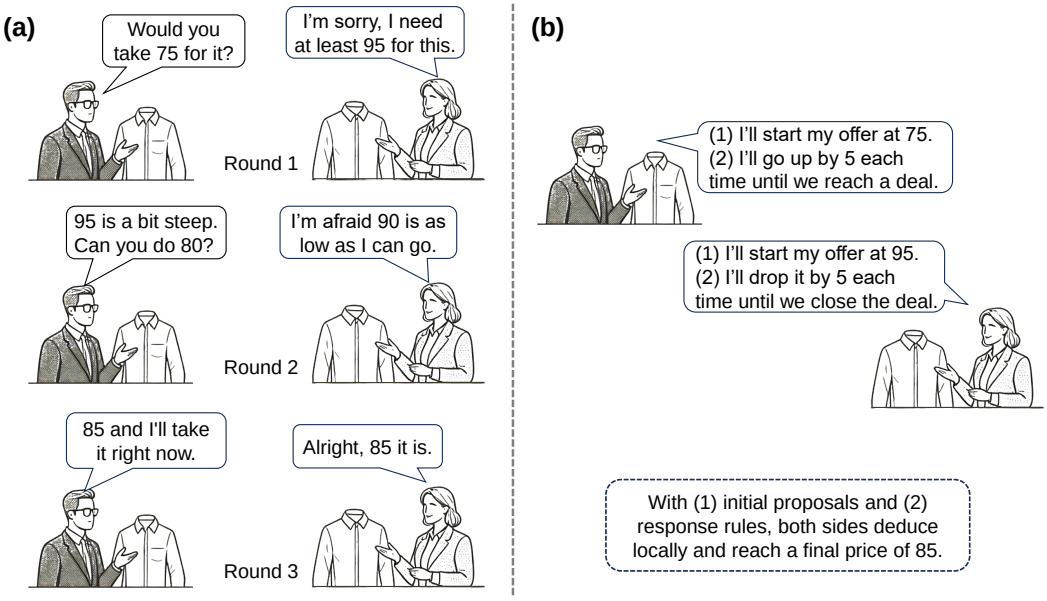


Figure 5: Conceptual comparison between multi-round interaction and one-round structured interaction inference. (a) Participants repeatedly exchange and revise their proposals before reaching an agreement. (b) Each participant communicates an initial proposal and its response rule once, allowing both sides to infer the agreed price locally. The initial listed price is 100.

B Implementation Details of QMIX_COMM

QMIX_COMM is a communication-augmented extension of QMIX that employs Transformer-based message aggregation during decentralized execution. Local inputs are first encoded into agent-wise communication representations and then processed by successive Transformer blocks. Each block operates on the representations updated by the preceding block, enabling agents to iteratively incorporate newly processed peer information. Under the execution setting considered in this paper, the number of Transformer blocks therefore corresponds to the number of logical communication rounds. A communication mask controls which agents can exchange information at each round.

In Section 2, QMIX_COMM is used to examine the empirical effect of communication depth on coordination performance. Configurations with different depths retain the same communication mechanism while varying the number of iterative refinement stages. QMIX_COMM is also included

Algorithm 1 QMIX_COMM

Require: Number of agents N, communication dimension d, discount factor γ, communication rounds N_r, maximum environment steps $T_{\max}$, and minibatch size B_{batch}

Initialize: Shared agent Q-network $Q_{\boldsymbol{\theta}}$, target network $Q_{\boldsymbol{\theta}^-}$, mixing network, target mixing network, replay buffer $\mathcal{B}$, and environment-step counter $n_{\text{env}} \leftarrow 0$
while $n_{\text{env}} < T_{\max}$ **do**
  Reset the environment and initialize $\{h_{i,0}\}_{i=1}^{N}$
  for $t = 1, \ldots, T$ **do**
    for each agent i **do**
      Receive the local policy input $x_{i,t}$
      Update $h_{i,t} \leftarrow \text{GRUCell}(x_{i,t}, h_{i,t-1})$
      Initialize $m_{i,t}^{(0)} \leftarrow G_{\text{enc}}(x_{i,t})$
    end for
    for $k = 1, \ldots, N_r$ **do**
      Exchange the current communication representations
      for each agent i **do**
        Update $m_{i,t}^{(k)} \leftarrow \mathcal{T}^{(k)}\left(m_{i,t}^{(k-1)}, \{m_{j,t}^{(k-1)}\}_{j \in \mathcal{N}_{i,t}}\right)$
      end for
    end for
    for each agent i **do**
      Compute $\mathbf{q}_{i,t} \leftarrow Q_{\boldsymbol{\theta}}\left(h_{i,t}, m_{i,t}^{(N_r)}\right)$
      Select $a_{i,t}$ from $\mathbf{q}_{i,t}$ using the behavior policy
    end for
    Execute the joint action $\mathbf{a}_t = (a_{1,t}, \ldots, a_{N,t})$
    Observe r_t, s_{t+1}, and the next local observations
    Store the transition in $\mathcal{B}$
    $n_{\text{env}} \leftarrow n_{\text{env}} + 1$
    if the episode terminates or $n_{\text{env}} \geq T_{\max}$ **then**
      break
    end if
  end for
  if $|\mathcal{B}| \geq B_{\text{batch}}$ **then**
    Sample a minibatch of trajectories from $\mathcal{B}$
    Unroll the agent and communication networks to compute individual utilities
    Compute $Q_{\text{tot},t} \leftarrow G_{\text{mix}}\left(q_{1,t}, \ldots, q_{N,t}; s_{\text{rc},t}\right)$
    Construct the TD(λ) target $y_t^{(\lambda)}$ using the target networks
    Compute $\mathcal{L}_{\text{td}} \leftarrow \mathbb{E}\left[\left(y_t^{(\lambda)} - Q_{\text{tot},t}\right)^2\right]$
    Update the agent, communication, and mixing networks by minimizing $\mathcal{L}_{\text{td}}$
    Periodically update the target networks
  end if
end while

as a multi-round baseline for comparison with SLRC in Section 7. The pseudocode is presented in Algorithm 1, and the implementation is also included in our open-source repository[2].

C Related Work

C.1 Learned Message Content, Selection, and Aggregation

Early differentiable communication frameworks established end-to-end learning of inter-agent communication, using continuous channels in CommNet and differentiable channels during DIAL training (Sukhbaatar et al., 2016; Foerster et al., 2016). Subsequent work has addressed complementary

[2] https://anonymous.4open.science/r/SLRC-5046

questions concerning what to transmit, when to communicate, whom to contact, and how to process received information. IC3Net learns when agents should communicate through individual gates, whereas TarMAC learns both message content and intended recipients (Singh et al., 2018; Das et al., 2019). NDQ introduces information-theoretic regularizers to encourage informative low-entropy messages and communication minimization, while IMAC combines an information-bottleneck objective with a learned scheduler under limited bandwidth (Wang et al., 2019; 2020a). VBC and TMC reduce communication overhead through variance-based and temporal message control, respectively (Zhang et al., 2019; 2020). Gated-ACML, I2C, and SMS prune communication according to learned message utility, inferred inter-agent influence, or Shapley message values, while CMVC uses counterfactual message values to select communication partners and weight received messages (Mao et al., 2020; Ding et al., 2020; Xue et al., 2022; Gao et al., 2025). At the receiver, MASIA learns a compact self-supervised representation from received messages, while T2MAC performs evidence-driven integration of observed and communicated information (Guan et al., 2022; Sun et al., 2024). Recent work further considers task-dependent communication structures and practical channel effects. GRDC constructs interaction graphs for role discovery and communication, whereas CAIC models queueing delays and message staleness in intent communication (Gao et al., 2026; Li et al., 2026).

SLRC focuses on a complementary question: how the transmitted message can support receiver-side collaborative inference. Each packet contains an $r \times r$ coupling component and an r-dimensional response component, whose aggregates define the reduced linear system used by each receiver. The message is therefore designed for computing the collaborative update rather than reconstructing each sender's full history.

C.2 Iterative and Graph-Based Communication

Multi-stage communication allows later messages to incorporate representations obtained from earlier exchanges. TarMAC explicitly introduces multiple pre-action communication rounds (Das et al., 2019), while AC2C conditionally requests a second round to obtain information from two-hop neighbors under range-constrained communication (Wang et al., 2023). Graph-based communication provides related mechanisms for propagating information between agents. DGN expands the receptive field through stacked graph-convolution layers (Jiang et al., 2018), NeurComm communicates representations between neighboring agents in networked control systems (Chu et al., 2020), MAGIC combines a learned communication graph with a graph-attention message processor (Niu et al., 2021), and TGCNet learns a dynamic directed communication graph with Transformer-based feature extraction (Zhang et al., 2025).

The number of logical communication rounds cannot be inferred from architectural depth alone; a new round is required only when a stage consumes peer representations produced by a preceding stage during decentralized execution. SLRC draws inspiration from the resulting iterative cross-agent refinement but does not claim to reproduce the nonlinear dynamics of these architectures. Its theoretical analysis instead concerns the repeated linear responses induced by SLRC's own learned interaction model and shows that, under the stated contraction condition, one-round decoding recovers their limiting update.

C.3 Structured and Low-Rank Interaction Models

Structured representations have been introduced into MARL at the levels of communication topology, value factorization, and function-class assumptions. Learning Structured Communication learns an adaptive hierarchical communication topology and routing mechanism (Sheng et al., 2022). Coordination-graph methods instead structure joint action values. Deep Coordination Graphs factor the joint value into pairwise payoffs, use low-rank payoff approximations, and perform local message passing for action maximization (Böhmer et al., 2020). Deep Implicit Coordination Graphs infer an observation-dependent coordination graph and use a graph neural network to reason about joint actions or values (Li et al., 2020). Low interaction rank has also been studied as a function-class assumption for computationally and statistically efficient offline MARL (Zhan et al., 2025).

These approaches apply structure to communication topology, joint-value interactions, or offline function classes. In contrast, SLRC structures the receiver-side interaction model through agent-specific positive diagonal operators and a factorized cross-agent coupling, whose factorization rank r controls the interaction capacity. This parameterization allows the aggregated message components

to define an $r \times r$ linear system after one logical exchange. SLRC therefore uses low-rank structure to compute collaborative representation updates, rather than to construct communication topologies, factor joint action values, or characterize offline learnability.

D Additional Theoretical Analysis

D.1 Derivation of the Structured Coupled Response

At a fixed decision step t, write the MRC response process as:

$$x^{(k+1)} = \mathcal{F}_t^{(k)}(x^{(k)}), \tag{11}$$

where $x^{(k)} \in \mathbb{R}^{Nd}$ is the joint response at stage k. The map is conditioned on the current agent histories, and the superscript allows different stages to use different transformations.

SLRC models this response dependency using a learned affine map $\mathcal{R}(x) = a + Kx$, where $a \in \mathbb{R}^{Nd}$ is independent of the current adjustment and $K \in \mathbb{R}^{Nd \times Nd}$ describes its influence on the next response. The affine form makes the self-consistency condition linear:

$$\Delta = a + K\Delta \quad \Longleftrightarrow \quad (I_{Nd} - K)\Delta = a. \tag{12}$$

To enable compact interaction encoding, SLRC restricts $\mathrm{rank}(K) \le r$. The affine and rank restrictions specify the SLRC model class. The following reparameterization is within this class.

By rank factorization, there exist $P, Q \in \mathbb{R}^{Nd \times r}$ such that:

$$K = PQ^\top. \tag{13}$$

If $\mathrm{rank}(K) < r$, the factors can be padded with zero columns. The product $Q^\top x$ aggregates the joint adjustment into an r-dimensional space, and P maps the aggregate back to the representation space.

For any matrix W satisfying Definition 1, introduce the parameters:

$$f = -Wa, \qquad U = -WP, \qquad V = Q. \tag{14}$$

Since W is invertible, these choices yield:

$$\begin{aligned} -W^{-1}f - W^{-1}UV^\top x &= a + PQ^\top x \\ &= a + Kx \\ &= \mathcal{R}(x). \end{aligned} \tag{15}$$

Thus, the structured map $\mathcal{T}(x) = -W^{-1}f - W^{-1}UV^\top x$ retains the chosen affine response while explicitly parameterizing agent-specific scaling through W.

For $x = [x_1^\top, \ldots, x_N^\top]^\top$, the stacked definition of V gives:

$$V^\top x = \sum_{j=1}^{N} v_j^\top x_j. \tag{16}$$

Because W^{-1} is block diagonal, the response of agent i is:

$$\mathcal{T}_i(x) = -w_i^{-1} f_i - w_i^{-1} u_i \sum_{j=1}^{N} v_j^\top x_j. \tag{17}$$

The coefficient mapping x_j to the response of agent i is therefore $-w_i^{-1} u_i v_j^\top$.

At self-consistency, $\Delta = \mathcal{T}(\Delta)$. Multiplying the agent-specific response equation by w_i and rearranging gives:

$$w_i \Delta m_i + u_i \sum_{j=1}^{N} v_j^\top \Delta m_j = -f_i. \tag{18}$$

Stacking these equations yields:

$$(W + UV^\top)\Delta = -f, \tag{19}$$

which is the coupled system in Definition 1.

Partitioning $B = W + UV^\top$ into $d \times d$ blocks gives:

$$[B]_{ii} = w_i + u_i v_i^\top, \qquad [B]_{ij} = u_i v_j^\top \quad \text{for } i \neq j. \tag{20}$$

The off-diagonal blocks describe cross-agent coupling, while $u_i v_i^\top$ augments the local block w_i. Separate factors allow $[B]_{ij} \neq [B]_{ji}^\top$. Furthermore:

$$\operatorname{rank}(UV^\top) \leq \min\{\operatorname{rank}(U), \operatorname{rank}(V)\} \leq r. \tag{21}$$

The rank constraint applies to the interaction component $UV^\top$, whereas the complete matrix B need not be low rank.

D.2 Proof of Proposition 1

Proof. Consider an affine response map $\mathcal{R}(x) = a + Kx$ with $\operatorname{rank}(K) \leq r$. For any W satisfying Definition 1, set:

$$f = -Wa, \qquad C = -WK. \tag{22}$$

Since W is invertible, multiplication by W preserves rank:

$$\operatorname{rank}(C) = \operatorname{rank}(K) \leq r. \tag{23}$$

If $C = 0$, choose $U = V = 0_{Nd \times r}$. Otherwise, write $s = \operatorname{rank}(C) \leq r$. A compact singular value decomposition gives:

$$C = L\Sigma R^\top, \tag{24}$$

where $L, R \in \mathbb{R}^{Nd \times s}$ have orthonormal columns and $\Sigma \in \mathbb{R}^{s \times s}$ has strictly positive diagonal entries. Define:

$$\begin{aligned} U &= \left[L\Sigma^{1/2},\, 0_{Nd \times (r-s)}\right], \\ V &= \left[R\Sigma^{1/2},\, 0_{Nd \times (r-s)}\right]. \end{aligned} \tag{25}$$

The zero blocks are omitted when $s = r$. In either case, the factors satisfy:

$$UV^\top = C = -WK. \tag{26}$$

Consequently:

$$\begin{aligned} \mathcal{T}(x) &= -W^{-1}f - W^{-1}UV^\top x \\ &= a + Kx \\ &= \mathcal{R}(x) \end{aligned} \tag{27}$$

for every $x \in \mathbb{R}^{Nd}$.

If $V = U$, the factorized component is $UU^\top$, which is symmetric. For every $y \in \mathbb{R}^{Nd}$:

$$y^\top UU^\top y = \|U^\top y\|_2^2 \geq 0. \tag{28}$$

Hence, $UU^\top$ is positive semidefinite and has rank at most r. Separate factors retain the same rank bound but need not produce a symmetric or positive-semidefinite interaction component. □

D.3 Derivation and Exactness of Compact Decoding

The following derivation identifies the compact packet components and proves Proposition 2 using the Sherman–Morrison–Woodbury identity.

Proof. From Definition 1, the coupled system is $B\Delta = -f$, where $B = W + UV^\top$. Define:

$$S = I_r + V^\top W^{-1} U, \qquad \bar{b} = V^\top W^{-1} f. \tag{29}$$

The strictly positive diagonal entries of each w_i make W invertible. The matrix determinant lemma gives:

$$\det(B) = \det(W + UV^\top) = \det(W)\det(S). \tag{30}$$

Thus, B is invertible if and only if S is invertible.

Under the assumed invertibility of S, the Woodbury identity gives:

$$B^{-1} = W^{-1} - W^{-1}US^{-1}V^\top W^{-1}. \tag{31}$$

Multiplying by $-f$ yields the unique coupled response:

$$\begin{aligned}\Delta &= -B^{-1}f \\ &= -W^{-1}f + W^{-1}US^{-1}V^\top W^{-1}f \\ &= -W^{-1}f + W^{-1}US^{-1}\bar{b}.\end{aligned} \tag{32}$$

This expression identifies the reduced coefficient $c = S^{-1}\bar{b}$, equivalently:

$$Sc = \bar{b}. \tag{33}$$

The decoded response therefore becomes:

$$\Delta = -W^{-1}f + W^{-1}Uc. \tag{34}$$

The block structure of W^{-1} gives:

$$\begin{aligned}V^\top W^{-1}U &= \sum_{j=1}^{N} v_j^\top w_j^{-1} u_j, \\ V^\top W^{-1}f &= \sum_{j=1}^{N} v_j^\top w_j^{-1} f_j.\end{aligned} \tag{35}$$

Each summand depends only on the coefficients of its sender. Defining:

$$z_j = v_j^\top w_j^{-1} u_j, \qquad b_j = v_j^\top w_j^{-1} f_j \tag{36}$$

gives the aggregate quantities:

$$S = I_r + \sum_{j=1}^{N} z_j, \qquad \bar{b} = \sum_{j=1}^{N} b_j. \tag{37}$$

Therefore, the packet $p_j = [\mathrm{vec}(z_j)^\top, b_j^\top]^\top \in \mathbb{R}^{r^2+r}$ contains all sender-dependent components required to form S and $\bar{b}$.

Taking the i-th block of Eq. (34) yields $\Delta m_i = -w_i^{-1} f_i + w_i^{-1} u_i c$, which recovers the agent-wise decoding rule in Definition 2. With common aggregate quantities, all receivers compute the same coefficient c and recover their respective blocks of the unique coupled response $\Delta = -B^{-1}f$.

Finally, the reduced coefficient also satisfies:

$$\begin{aligned}-V^\top \Delta &= \bar{b} - V^\top W^{-1} U c \\ &= \bar{b} - (S - I_r)c \\ &= c,\end{aligned} \tag{38}$$

where the last equality follows from $Sc = \bar{b}$. This establishes both the exactness of decoding and the interpretation $c = -V^\top \Delta$. □

D.4 Proof of Proposition 3

Proof. Recall $A = W^{-1}UV^\top$ and assume $\|A\|_2 \le q < 1$. If $(I_{Nd} + A)y = 0$ for some $y \in \mathbb{R}^{Nd}$, then $y = -Ay$, so:

$$\|y\|_2 = \|Ay\|_2 \le q\|y\|_2. \tag{39}$$

Since $q < 1$, this requires $y = 0$. Therefore, $I_{Nd} + A$ is invertible. Moreover:

$$B = W + UV^\top = W(I_{Nd} + A). \tag{40}$$

Since both factors are invertible, B is invertible. The determinant identity in Eq. (30) then implies that S is invertible.

Define the unique coupled response $\Delta = -B^{-1}f$. Multiplying $B\Delta = -f$ by W^{-1} gives:

$$\Delta = -W^{-1}f - A\Delta. \tag{41}$$

Subtracting this relation from Eq. (3) yields:

$$\Delta^{(k+1)} - \Delta = -A\left(\Delta^{(k)} - \Delta\right). \tag{42}$$

Consequently:

$$\left\|\Delta^{(k+1)} - \Delta\right\|_2 \leq q\left\|\Delta^{(k)} - \Delta\right\|_2. \tag{43}$$

Applying this inequality recursively gives:

$$\left\|\Delta^{(k)} - \Delta\right\|_2 \leq q^k\left\|\Delta^{(0)} - \Delta\right\|_2. \tag{44}$$

Because $q < 1$, the error converges to zero:

$$\lim_{k\to\infty}\Delta^{(k)} = \Delta = -B^{-1}f. \tag{45}$$

Since S is invertible, Proposition 2 ensures that receivers with the common aggregates recover this same response through compact decoding. □

E Derivation of $\mathcal{L}_{\text{sym}}$

At timestep t, let $U^t \in \mathbb{R}^{n_t d\times r}$ and $V^t \in \mathbb{R}^{n_t d\times r}$ stack the interaction factors of the active agents. The factorized interaction component is:

$$C^t = U^t {V^t}^\top. \tag{46}$$

Its transpose is ${C^t}^\top = V^t {U^t}^\top$. A soft symmetry penalty can therefore be written as:

$$\left\|C^t - {C^t}^\top\right\|_{\text{F}}^2 = \left\|U^t {V^t}^\top - V^t {U^t}^\top\right\|_{\text{F}}^2. \tag{47}$$

Directly constructing the matrices in Eq. (47) requires operations in the $n_t d$-dimensional stacked representation space. The same quantity can be evaluated using only $r \times r$ matrices. Expanding the Frobenius norm gives:

$$\begin{aligned}&\left\|U^t {V^t}^\top - V^t {U^t}^\top\right\|_{\text{F}}^2\\ &= \text{Tr}\left[\left(U^t {V^t}^\top - V^t {U^t}^\top\right)^\top\left(U^t {V^t}^\top - V^t {U^t}^\top\right)\right].\end{aligned} \tag{48}$$

Expanding the product and applying cyclic invariance of the trace yields:

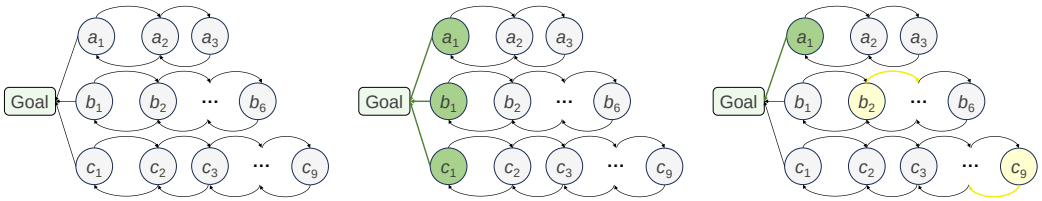
$$\begin{aligned}&\left\|U^t {V^t}^\top - V^t {U^t}^\top\right\|_{\text{F}}^2\\ &= 2\,\text{Tr}\left(U^{t\top} U^t V^{t\top} V^t\right) - 2\,\text{Tr}\left[\left(V^{t\top} U^t\right)^2\right].\end{aligned} \tag{49}$$

Define the reduced matrices:

$$\Psi_u^t = U^{t\top} U^t, \qquad \Psi_v^t = V^{t\top} V^t, \qquad \Omega^t = V^{t\top} U^t. \tag{50}$$

Substituting these definitions into Eq. (49) gives:

$$\left\|C^t - {C^t}^\top\right\|_{\text{F}}^2 = 2\left[\text{Tr}\left(\Psi_u^t \Psi_v^t\right) - \text{Tr}\left((\Omega^t)^2\right)\right]. \tag{51}$$



(a) Environment layout. (b) Successful coordination. (c) Failed coordination.

Figure 6: Hallway environment and representative outcomes. Agents succeed only when they reach their goals simultaneously. An early arrival by any agent results in failure.

For a sampled batch containing T valid timesteps, we normalize the accumulated symmetry penalty using the total count $\sum_{t=1}^{T} n_t(n_t-1)$:

$$\begin{aligned}\mathcal{L}_{\text{sym}} &= 2\left[\sum_{t=1}^{T} n_t(n_t-1)\right]^{-1} \sum_{t=1}^{T} \left\|C^t - (C^t)^\top\right\|_{\text{F}}^2 \\ &= 4\left[\sum_{t=1}^{T} n_t(n_t-1)\right]^{-1} \sum_{t=1}^{T} \left[\text{Tr}(\Psi_u^t \Psi_v^t) - \text{Tr}\left((\Omega^t)^2\right)\right].\end{aligned} \tag{52}$$

Dividing both the numerator and denominator by T yields the empirical-expectation form in Eq. (5). If the normalization denominator is zero, the loss is defined as zero. The penalty measures asymmetry of the complete factorized interaction matrix, including its diagonal blocks; the count-based normalization does not restrict the penalty to off-diagonal interactions.

The reduced matrices Ψ_u^t, Ψ_v^t, and Ω^t have dimension $r \times r$. Therefore, the implementation evaluates the symmetry objective without explicitly constructing the $n_t d \times n_t d$ interaction matrix. Since $\mathcal{L}_{\text{sym}}$ is optimized as a finite-weight penalty, it encourages reciprocal interactions without enforcing exact symmetry or requiring $U^t = V^t$.

Remark 1. At a fixed decision step, suppose the symmetry penalty vanishes for the same N agents used in decoding, so that $\mathcal{L}_{\text{sym}} = 0$, and the channel-consistency constraints hold exactly, with $v_i^\top u_i = I_r$ for every agent i. Let $C = UV^\top$. The zero symmetry penalty implies $C = C^\top$, while channel consistency gives $V^\top U = NI_r$. Consequently, $C^2 = U(V^\top U)V^\top = NC$. Since C is symmetric, every eigenvalue λ satisfies $\lambda^2 = N\lambda$, so its eigenvalues belong to $\{0, N\}$ and $C \succeq 0$. Together with $W \succ 0$, this yields $B = W + C \succ 0$. The determinant identity $\det(B) = \det(W)\det(S)$, where $S = I_r + V^\top W^{-1} U$, then implies that S is invertible. This result supports using the symmetry and channel-consistency losses as soft structural constraints that encourage invertibility of the reduced system.

F Implementation Details of SLRC

Algorithm 2 summarizes the decentralized data collection and centralized training procedures of SLRC. At each timestep, every agent encodes its local policy input and generates an intent representation together with the structured interaction factors. These quantities are converted into a packet of dimension $r^2 + r$ and transmitted through one logical exchange. Each receiver aggregates the packets available from its sender set, solves the resulting r-dimensional linear system, and computes its collaborative representation update locally. During centralized training, the agent and mixing networks are jointly optimized using the TD objective, the intent objective, and the three factor regularizers, while the target networks are updated periodically.

G Detailed Environment Configurations

G.1 Hallway

Hallway is a sparse-reward cooperative task used to evaluate communication under severe partial observability (Xue et al., 2022). A team of N agents moves independently along parallel one-

Algorithm 2 SLRC

Require: Number of agents N, representation dimension d, rank r, discount factor γ, maximum environment steps $T_{\max}$, minibatch size B_{batch}, and loss coefficients

Initialize: Agent network, mixing network, target networks, replay buffer $\mathcal{B}$, and environment-step counter $n_{\text{env}} \leftarrow 0$
while $n_{\text{env}} < T_{\max}$ **do**
  Reset the environment and initialize $\{m_{i,0}\}_{i=1}^{N}$
  for $t = 1, \dots, T$ **do**
    for each agent i **do**
      Update $m_{i,t} \leftarrow \text{GRUCell}(x_{i,t}, m_{i,t-1})$
      Generate $f_{i,t} \leftarrow G_F(m_{i,t})$
      Construct $\zeta_{i,t} \leftarrow G_E([m_{i,t}, f_{i,t}])$
      Compute $d_{i,t} \leftarrow \text{softplus}(G_W(\zeta_{i,t})) + \epsilon \mathbf{1}_d$
      Set $w_{i,t} \leftarrow \text{Diag}(d_{i,t})$
      Generate $u_{i,t} \leftarrow G_U(\zeta_{i,t})$ and $v_{i,t} \leftarrow G_V(\zeta_{i,t})$
      Compute $z_{i,t} \leftarrow v_{i,t}^{\top} w_{i,t}^{-1} u_{i,t}$
      Compute $b_{i,t} \leftarrow v_{i,t}^{\top} w_{i,t}^{-1} f_{i,t}$
      Form $p_{i,t} \leftarrow [\text{vec}(z_{i,t})^{\top}, b_{i,t}^{\top}]^{\top}$
    end for
    Exchange the packets $\{p_{i,t}\}_{i=1}^{N}$ once
    for each receiver i **do**
      Compute $S_{i,t} \leftarrow I_r + \sum_{j \in \mathcal{N}_{i,t}} z_{j,t}$
      Compute $\bar{b}_{i,t} \leftarrow \sum_{j \in \mathcal{N}_{i,t}} b_{j,t}$
      Solve $S_{i,t} c_{i,t} = \bar{b}_{i,t}$
      Decode $\Delta m_{i,t} \leftarrow -w_{i,t}^{-1} f_{i,t} + w_{i,t}^{-1} u_{i,t} c_{i,t}$
      Update $\widetilde{m}_{i,t} \leftarrow m_{i,t} + \Delta m_{i,t}$
      Compute $\mathbf{q}_{i,t} \leftarrow G_Q(\widetilde{m}_{i,t})$
      Select an action using the behavior policy
    end for
    Execute the joint action and store the transition in $\mathcal{B}$
    $n_{\text{env}} \leftarrow n_{\text{env}} + 1$
    if the episode terminates or $n_{\text{env}} \geq T_{\max}$ **then**
      break
    end if
  end for
  if $|\mathcal{B}| \geq B_{\text{batch}}$ **then**
    Sample a minibatch of trajectories from $\mathcal{B}$
    Unroll the agent network to obtain individual action values and interaction components
    Compute Q_{tot} using the centralized mixing network
    Compute $\mathcal{L}_{\text{td}}$ using the target networks
    Compute $\mathcal{L}_F$, $\mathcal{L}_{\text{sym}}$, $\mathcal{L}_{\text{enor}}$, and $\mathcal{L}_{\text{dcp}}$
    Compute $\mathcal{L}_{\text{reg}} \leftarrow \lambda_{\text{sym}} \mathcal{L}_{\text{sym}} + \lambda_{\text{enor}} \mathcal{L}_{\text{enor}} + \lambda_{\text{dcp}} \mathcal{L}_{\text{dcp}}$
    Compute $\mathcal{L}_{\text{tot}} \leftarrow \mathcal{L}_{\text{td}} + \lambda_F \mathcal{L}_F + \lambda_{\text{reg}} \mathcal{L}_{\text{reg}}$
    Update the agent and mixing networks by minimizing $\mathcal{L}_{\text{tot}}$
    Periodically update the target networks
  end if
end while

dimensional corridors. Agent i occupies position $p_{i,t} \in \{1, \dots, L_i\}$ at timestep t, where L_i denotes the length of its corridor. The agents are initialized at random positions and must coordinate their arrival at the designated goal states.

Each agent observes only its position and cannot access the positions or progress of its teammates, making effective communication essential for reliably coordinating their movements and reaching the goals simultaneously. The discrete action space consists of moving left, moving right, and remaining stationary. Fig. 6 illustrates the environment together with successful and failed outcomes.

Let g_i denote the goal position of agent i. The shared reward is defined as follows:

$$r_t = \begin{cases} 10, & \text{if } p_{i,t} = g_i \text{ for every } i \in \{1, \dots, N\}, \\ 0, & \text{otherwise.} \end{cases} \tag{53}$$

An episode terminates when any agent reaches its goal or when the horizon $H = \max_{i \in \{1,\dots,N\}} L_i + 10$ is reached. An agent arriving before its teammates therefore terminates the episode without a positive reward. This sparse feedback requires coordinated timing rather than independent progress toward the goals. We evaluate two Hallway configurations. The `hallway_3a` configuration contains three agents with corridor lengths $L_1 = 3$, $L_2 = 6$, and $L_3 = 9$. The `hallway_5a` configuration contains five agents with corridor lengths $L_1 = 3$, $L_2 = 6$, $L_3 = 6$, $L_4 = 9$, and $L_5 = 9$, increasing the difficulty of synchronizing their arrivals under partial observability. A reference implementation of the Hallway environment is available at https://github.com/DiXue98/SMS.

G.2 Multi-Agent Particle Environment

We evaluate SLRC on two Multi-Agent Particle Environment (MPE) tasks (Lowe et al., 2017): Cooperative Navigation (CN) and Predator Prey (PP). MPE is a two-dimensional particle world with discrete actions and continuous observations. Agents can move in four directions or remain stationary. Entities are randomly initialized each episode. The map size is 5, and episodes last at most 60 timesteps.

To emphasize communication under partial observability, cooperative agents cannot observe teammate-related information. In CN, each agent observes its own position and the two nearest landmarks. In PP, each predator observes its own position and the two nearest prey. Consequently, coordinated behavior requires information beyond an agent's local observation.

The environments are implemented using the MPE framework available at https://github.com/openai/multiagent-particle-envs. The pretrained prey policies in PP are obtained from https://github.com/uoe-agents/epymarl.

G.2.1 Cooperative Navigation

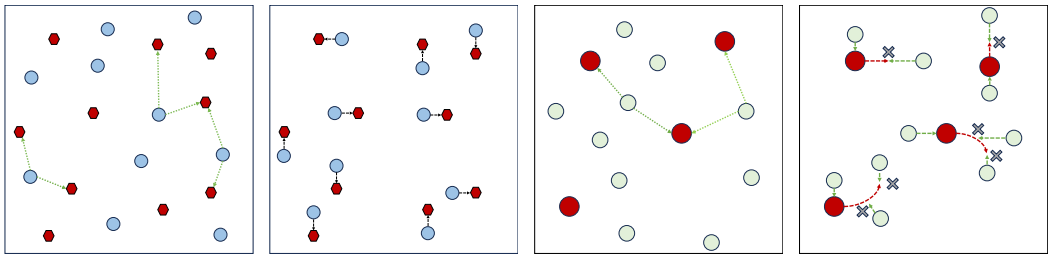
Cooperative Navigation requires a team of agents to cover a set of landmarks while avoiding collisions. The number of agents equals the number of landmarks, so effective coordination requires different agents to occupy different landmarks. The CN examples in Fig. 7 illustrate the environment and coordinated landmark coverage. Let $\mathcal{N}$ and $\mathcal{L}$ denote the sets of agents and landmarks, respectively. The positions of agent i and landmark ℓ at timestep t are denoted by $x_{i,t}$ and y_ℓ, and ρ_i denotes the radius of agent i. The shared reward is defined as follows:

$$r_t = -\sum_{\ell \in \mathcal{L}} \min_{i \in \mathcal{N}} \|x_{i,t} - y_\ell\|_2 - \sum_{\substack{i,j \in \mathcal{N} \\ i<j}} \mathbb{I}\left[\|x_{i,t} - x_{j,t}\|_2 < \rho_i + \rho_j\right]. \tag{54}$$

where $\mathcal{N}$ and $\mathcal{L}$ denote the agent and landmark sets, $x_{i,t}$ and y_ℓ denote their positions, ρ_i is the radius of agent i, and $\mathbb{I}[\cdot]$ is the indicator function. The first term encourages the team to cover all landmarks by penalizing the distance from each landmark to its nearest agent, while the second penalizes inter-agent collisions. We evaluate `CN_4` with four agents and four landmarks and `CN_10` with ten agents and ten landmarks. All agents have a radius of 0.15, and the larger configuration increases the difficulty of landmark assignment and collision avoidance.

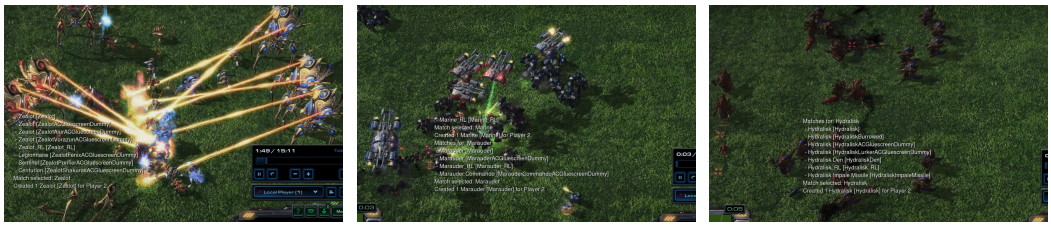
G.2.2 Predator Prey

Predator Prey requires predators to pursue and capture prey while avoiding inter-predator collisions. The prey follow a pretrained evasive policy. Predators have acceleration and radius values of 5 and 0.05, while prey have corresponding values of 7 and 0.03, respectively. The examples in Fig. 7 illustrate the environment and a coordinated pursuit strategy. The shared reward is defined as follows:



(a) CN overview. (b) Landmark coverage. (c) PP overview. (d) Coordinated pursuit.

Figure 7: MPE task configurations and representative coordinated behaviors. Panels (a)–(b) show Cooperative Navigation, while panels (c)–(d) show Predator Prey.



(a) Protoss. (b) Terran. (c) Zerg.

Figure 8: Representative 20-vs-20 SMACv2 scenarios for the three StarCraft II races.

$$
\begin{aligned}
r_t = & -\lambda_1 \sum_{i \in \mathcal{P}} \min_{j \in \mathcal{E}} \|x_{i,t} - y_{j,t}\|_2 + \lambda_2 \sum_{i \in \mathcal{P}} \sum_{j \in \mathcal{E}} \mathbb{I}\left[\|x_{i,t} - y_{j,t}\|_2 < \rho_i^{\mathrm{P}} + \rho_j^{\mathrm{E}}\right] \\
& - \lambda_3 \sum_{\substack{i,k \in \mathcal{P} \\ i<k}} \mathbb{I}\left[\|x_{i,t} - x_{k,t}\|_2 < \rho_i^{\mathrm{P}} + \rho_k^{\mathrm{P}}\right].
\end{aligned} \tag{55}
$$

where $\mathcal{P}$ and $\mathcal{E}$ denote the predator and prey sets, $x_{i,t}$ and $y_{j,t}$ denote their positions, and ρ_i^{P} and ρ_j^{E} denote their radii. The indicator function is denoted by $\mathbb{I}[\cdot]$, while $\lambda_1, \lambda_2, \lambda_3 > 0$ weight the pursuit, capture, and collision components. We set $\lambda_1 = 0.025$, $\lambda_2 = 5$, and $\lambda_3 = 0.25$ in all PP experiments. The first term encourages each predator to approach its nearest prey, the second rewards predator-prey contacts, and the third penalizes collisions between predators. We evaluate `PP_5_vs_2` with five predators and two prey and `PP_10_vs_4` with ten predators and four prey. The larger configuration increases the difficulty of target allocation and coordinated pursuit.

G.3 SMACv2

SMACv2 is a cooperative micromanagement benchmark based on StarCraft II (Ellis et al., 2023). Each agent controls an allied unit and selects from movement, attack, stop, and no-operation actions. An episode terminates when either army is eliminated or the horizon of 400 timesteps is reached.

To evaluate communication under partial observability, we remove teammate-related features from observations. Figs. 9 and 10 in Appendix G.5 show that this restriction increases dependence on exchanged information, enabling clearer evaluation. Following the SMACv2 recommendation for communication evaluation under extended partial observability (EPO), we set the enemy-observation probability to $p = 0$. Only the first agent detecting an enemy is guaranteed to observe it, while others must obtain the information through communication. Action masking excludes invalid actions. We evaluate Protoss, Terran, and Zerg at team sizes: 5-vs-5, 10-vs-10, and 20-vs-20, yielding nine scenarios with increasing team size and interaction complexity. 20-vs-20 battles are shown in Fig. 8.

All methods use the same SMACv2 team reward, ensuring that their performance differences arise from coordination and communication rather than distinct reward signals. At each timestep, the

reward measures progress against the enemy team through inflicted damage, unit eliminations, and the final battle outcome. Before normalization, it is computed as follows:

$$\widehat{r}_t = [D_t^{\text{enemy}} + 10K_t]_+ + 200I_t^{\text{win}}. \tag{56}$$

where D_t^{enemy} is the decrease in the enemies' total health and shields, K_t counts the enemy units eliminated at timestep t, and I_t^{win} indicates a team victory. The operator $[x]_+ = \max(x, 0)$ retains only nonnegative progress. Consequently, allied damage and deaths are not penalized, and defeat contributes no terminal bonus. The standard environment normalization accounts for enemy health, shields, elimination bonuses, and the victory bonus:

$$r_t = 20\,[H_{\text{max}}^{\text{enemy}} + 10n_{\text{enemy}} + 200]^{-1}\,\widehat{r}_t. \tag{57}$$

Here, n_{enemy} denotes the number of enemy units, and $H_{\text{max}}^{\text{enemy}}$ is their total maximum health and shields recorded when the environment initializes its normalization constant. This scaling preserves the relative contributions of damage, eliminations, and victory. The SMACv2 implementation is available at https://github.com/oxwhirl/smacv2.

G.4 Field-of-View-Constrained Communication

Under field-of-view (FoV) constraints, each receiver communicates only with teammates currently visible from its local observation range. Specifically, the sender set of agent i at timestep t is $\mathcal{N}_{i,t} = \{i\} \cup \{j \neq i : j \text{ is visible to } i\}$. The inclusion of i indicates that its locally generated packet remains available without transmission. Because visibility is receiver-dependent, the resulting communication graph may be directed. In the MPE tasks, including Cooperative Navigation and Predator Prey, visibility is determined by Euclidean distance with a fixed radius of 3 environment units. Accordingly, agent j is visible to agent i when $\|x_{i,t} - x_{j,t}\|_2 < 3$. In SMACv2, visibility varies with unit type and combat state. A living teammate j is visible to agent i when their Euclidean distance is smaller than the sight range of agent i. The sender set is therefore updated at every timestep according to current unit positions and survival status. For Protoss units, the sight ranges are 10 for Stalkers, 9 for Zealots, and 10 for Colossi. For Terran units, Marines, Marauders, and Medivacs use sight ranges of 9, 10, and 11, respectively. For Zerg units, the corresponding values are 8 for Zerglings, 8 for Banelings, and 9 for Hydralisks. Each receiver applies the same SLRC aggregation and decoding procedure to the packets available from its current sender set.

G.5 Communication Necessity under Restricted Observations

To construct communication-demanding evaluation settings, teammate-related features are excluded from local observations. In Hallway, this restriction follows directly from the environment definition. In CN, PP, and SMACv2, agents retain task-relevant information but cannot observe the states or progress of their teammates. This prevents policies from obtaining peer information without message exchange and allows the contribution of communication to be evaluated more clearly.

For enemy observations in SMACv2, we follow the recommended EPO configuration for communication evaluation and set the observation probability to $p = 0$ (Ellis et al., 2023). Once an enemy is detected, only the detecting agent is guaranteed to observe it when it is within sight range. The observations of other agents remain masked, making enemy information asymmetric across the team. Communication can convey information that is unavailable from the receivers' observations.

We first compare QMIX with QMIX_COMM under these restricted observations. QMIX_COMM augments QMIX with iterative Transformer-based message aggregation, as detailed in Appendix B. Fig. 9 shows that removing communication substantially degrades performance across the evaluated benchmarks and prevents effective coordination in several tasks. These results confirm that the restricted settings require agents to exploit information received from teammates.

We further examine how teammate-related observations affect the evaluation of communication in SMACv2. Fig. 10 repeats the comparison while retaining teammate features in local observations. In this setting, QMIX can use directly observed teammate states to support coordination, reducing the performance difference between the communication and no-communication variants. Comparing the two observation settings shows that excluding teammate features makes performance more dependent

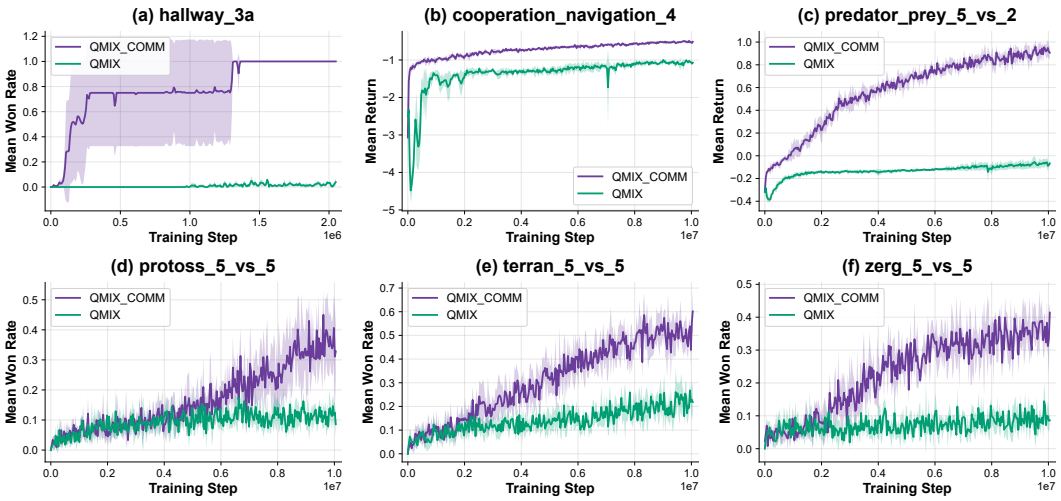


Figure 9: Performance with and without communication when teammate-related features are excluded from local observations.

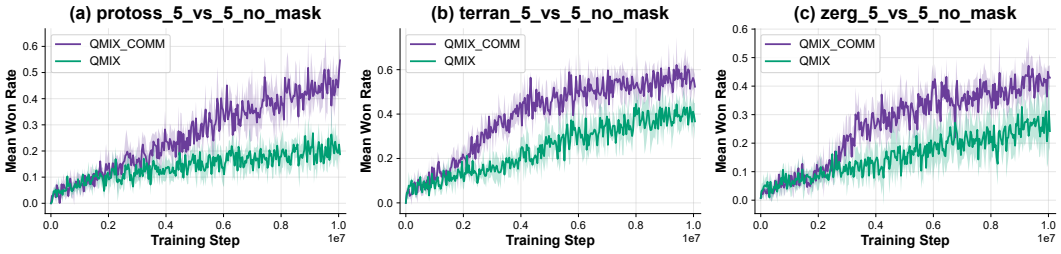


Figure 10: Performance with and without communication in SMACv2 when teammate-related features are retained in local observations.

on exchanged information and provides a clearer assessment of communication capabilities. We therefore use the restricted teammate observations in the subsequent SMACv2 experiments.

H Algorithm Details for Baselines and SLRC

All methods use the same environment configurations, observation restrictions, training budgets, random seeds, and evaluation protocol. Since the evaluated methods employ different learning frameworks and auxiliary objectives, their method-specific optimization and architectural settings follow the corresponding public implementations.

SLRC. SLRC uses the structured packet encoding and coupled decoding procedure described in Section 6. Its theoretical formulation is presented in Section 5, and the complete execution and training procedure is provided in Appendix F. The same default configuration is used across all environments, as summarized in Table 1. Rank, ablation, and sensitivity experiments change only the parameter or component under investigation.

QMIX and QMIX_COMM. QMIX (Rashid et al., 2020) provides the no-communication reference and uses the standard monotonic mixing architecture from EPyMARL. QMIX_COMM augments the same learner with Transformer-based message aggregation to examine iterative communication. It uses four attention heads and three sequential communication blocks, corresponding to three logical communication rounds. Its complete implementation is described in Appendix B. The QMIX implementation follows `https://github.com/uoe-agents/epymarl`.

Table 1: Default SLRC hyperparameters used across all environments. Controlled analyses modify only the parameter under investigation.

Training Configuration		SLRC Configuration	
Optimizer	Adam	Rank r	32
Learning rate	3×10^{-4}	Diagonal offset ϵ	10^{-4}
Discount factor γ	0.99	Packet dimension $r^2 + r$	1056
Batch size	32	Intent weight λ_F	0.2
Parallel environments	8	Regularization weight λ_{reg}	0.5
Replay-buffer size	5000	Symmetry weight λ_{sym}	0.05
Recurrent dimension	128	Channel-consistency weight λ_{enor}	0.05
Target-update interval	200	Pathway-separation weight λ_{dcp}	1

Table 2: Performance over the final 10% of training. The best result in each scenario is in bold.

Scenario	SLRC	TGCNet	MAGIC	T2MAC	MASIA	SMS	QMIX	QMIX_COMM
MPE								
CN_4	$\mathbf{-0.43 \pm 0.01}$	-0.75 ± 0.04	-1.37 ± 0.07	-1.38 ± 0.05	-1.05 ± 0.16	-1.31 ± 0.26	-1.08 ± 0.04	-0.52 ± 0.02
CN_10	$\mathbf{-0.47 \pm 0.02}$	-0.62 ± 0.02	-0.83 ± 0.07	-2.55 ± 1.95	-0.98 ± 0.21	-3.39 ± 0.67	-0.79 ± 0.03	-0.50 ± 0.03
PP_5_vs_2	$\mathbf{1.35 \pm 0.03}$	0.57 ± 0.20	-0.12 ± 0.31	-0.17 ± 0.03	0.00 ± 0.08	-0.20 ± 0.11	-0.07 ± 0.04	0.91 ± 0.02
PP_10_vs_4	$\mathbf{3.29 \pm 0.27}$	1.40 ± 0.07	-0.28 ± 0.42	-0.30 ± 0.04	0.14 ± 0.11	-0.25 ± 0.05	0.04 ± 0.18	2.06 ± 0.02
SMACv2 Win Rate (%)								
protoss_5_vs_5	$\mathbf{44.7 \pm 1.4}$	24.5 ± 1.1	9.6 ± 7.1	10.6 ± 1.6	13.1 ± 1.3	6.8 ± 1.7	11.3 ± 0.6	34.6 ± 9.7
protoss_10_vs_10	$\mathbf{61.9 \pm 4.1}$	38.6 ± 1.7	2.3 ± 2.3	14.9 ± 5.3	24.1 ± 1.4	2.4 ± 1.4	11.4 ± 2.1	45.7 ± 3.7
protoss_20_vs_20	$\mathbf{56.5 \pm 2.8}$	41.4 ± 3.9	0.9 ± 1.3	1.1 ± 1.0	26.4 ± 3.7	3.1 ± 0.2	12.2 ± 6.6	44.9 ± 3.9
terran_5_vs_5	$\mathbf{59.7 \pm 2.6}$	34.5 ± 1.2	16.3 ± 7.5	19.4 ± 11.8	24.2 ± 1.9	8.1 ± 2.3	21.0 ± 2.7	50.3 ± 5.3
terran_10_vs_10	$\mathbf{60.9 \pm 3.1}$	40.7 ± 2.0	8.0 ± 12.4	6.0 ± 8.9	31.8 ± 2.7	2.8 ± 1.7	25.9 ± 4.0	50.7 ± 7.4
terran_20_vs_20	$\mathbf{52.7 \pm 4.3}$	42.7 ± 2.6	0.3 ± 0.3	0.4 ± 0.3	33.7 ± 4.9	1.0 ± 0.2	27.1 ± 1.9	39.5 ± 4.0
zerg_5_vs_5	$\mathbf{45.7 \pm 1.4}$	26.0 ± 1.0	4.4 ± 1.3	18.5 ± 2.8	19.7 ± 0.6	8.1 ± 1.9	9.2 ± 1.1	35.9 ± 1.6
zerg_10_vs_10	$\mathbf{52.5 \pm 2.9}$	25.8 ± 10.5	1.2 ± 0.7	13.8 ± 5.4	23.9 ± 3.8	3.1 ± 1.3	11.3 ± 4.2	39.0 ± 2.1
zerg_20_vs_20	$\mathbf{47.9 \pm 2.1}$	17.1 ± 14.1	0.2 ± 0.2	1.2 ± 1.2	23.5 ± 6.5	0.4 ± 0.2	13.5 ± 6.7	36.0 ± 1.8
Hallway Success Rate (%)								
hallway_3a	1.5 ± 0.5	3.0 ± 2.2	0.5 ± 0.9	24.6 ± 49.3	65.8 ± 47.2	96.1 ± 6.4	2.0 ± 0.4	$\mathbf{100.0 \pm 0.0}$
hallway_5a	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	$\mathbf{25.0 \pm 50.0}$

Multi-Round Communication Baselines. TGCNet (Zhang et al., 2025) constructs dynamic directed communication graphs and applies three Transformer-based graph-processing layers. We use its official graph-coarsening and agent architectures from https://github.com/ZhuohuiZhang/TGCNet. MAGIC (Niu et al., 2021) learns directed communication graphs through a scheduler and processes exchanged messages using graph attention. We retain its original policy-based training framework and two-head graph-attention configuration from https://github.com/CORE-Robotics-Lab/MAGIC.

Single-Round Communication Baselines. T2MAC (Sun et al., 2024) performs selective engagement and evidence-based message integration. Its minimum uncertainty coefficient is set to 0.1, and its query, key, and value dimensions are set to 64, following https://github.com/ZangZehua/T2MAC. MASIA (Guan et al., 2022) uses self-supervised message aggregation with state reconstruction and two-step latent prediction enabled, following https://github.com/chenf-ai/MASIA. SMS (Xue et al., 2022) selects messages according to predicted Shapley message values. Its selector uses a hidden dimension of 32 with a message-dropout rate of 0.5, following https://github.com/DiXue98/SMS. The remaining baseline settings follow their recommended public configurations.

I Additional Results and Analyses

The complete training curves across all 15 scenarios are presented in Fig. 3 of the main text. This section reports the corresponding detailed results, followed by additional analyses of representation structure and wall-clock training time.

I.1 More Detailed Results

Table 2 reports performance over the final 10% of training, where SLRC achieves the highest mean performance among the methods in all 13 CN, PP, and SMACv2 scenarios. QMIX_COMM is



Figure 11: Keyframes of SLRC coordination in `protoss_10_vs_10`.

generally the strongest baseline, while the advantage of SLRC becomes pronounced in larger PP and SMACv2 scenarios, where more agents and interactions increase coordination difficulty. These results show that the one-round structured decoder maintains coordination across team sizes and task dynamics. Fig. 11 provides an example in `protoss_10_vs_10`. At 2 seconds, the green allied units are initialized at different positions on the map. Between 4 and 7 seconds, they advance from multiple directions and establish engagement fronts rather than converging along a single path. At 10 seconds, the units concentrate fire on local targets. From 14 to 18 seconds, they adjust their formation while maintaining distributed pressure. At 23 seconds, the surviving units reassign targets following enemy eliminations and refocus on the remaining enemies by 29 seconds. This behavior combines spatially differentiated movement with coordinated attacks, consistent with SLRC's objective of supporting complementary agent decisions through one structured exchange at each decision step.

The sparse-reward Hallway results reveal a limitation of SLRC. In `hallway_3a`, QMIX_COMM achieves a 100.0% success rate, followed by SMS at 96.1% and MASIA at 65.8%, whereas SLRC reaches 1.5%. In the challenging `hallway_5a`, QMIX_COMM obtains 25.0%, while SLRC and remaining methods achieve 0.0%. Hallway provides a reward only when all agents reach their goals simultaneously, offering no intermediate feedback for learning intent representations or interaction factors. SLRC improves how interaction information is exchanged and decoded, but does not address exploration problem created by sparse rewards. Incorporating auxiliary learning signals and exploration mechanisms for sparse-reward coordination is an important direction for future work.

I.2 Representation Structure Analysis

We compare the representations used for action selection after communication. Let $\widetilde{m}_{i,t} \in \mathbb{R}^d$ denote the post-communication representation for each method. For SLRC, $\widetilde{m}_{i,t} = m_{i,t} + \Delta m_{i,t}$, while for QMIX_COMM and TGCNet it denotes the representation produced by their communication modules. For agents i and j, the cosine similarity between their representations is defined as follows:

$$s_{i,j}^{(t)} = \widetilde{m}_{i,t}^{\top} \widetilde{m}_{j,t} \left[\max\{\|\widetilde{m}_{i,t}\|_2, \epsilon\} \max\{\|\widetilde{m}_{j,t}\|_2, \epsilon\} \right]^{-1}, \qquad \epsilon = 10^{-12}. \tag{58}$$

Only unordered agent pairs in $\mathcal{P} = \{(i,j) : 1 \le i < j \le N\}$ are included, where $P = |\mathcal{P}| = N(N-1)/2$. Their mean similarity at timestep t is $\mathrm{AvgSim}_t = P^{-1} \sum_{(i,j)\in\mathcal{P}} s_{i,j}^{(t)}$. Figs. 12 and 13 show how this quantity evolves throughout an episode. Solid lines report the mean across seeds, while shaded regions indicate 95% confidence intervals. TGCNet maintains pairwise similarities close to one across most scenarios, indicating strongly aligned agent representations. QMIX_COMM exhibits intermediate alignment, while SLRC generally produces lower similarities that vary with the task and episode stage. These temporal profiles show that SLRC preserves agent-specific variation throughout the interaction process rather than enforcing a uniformly shared representation.

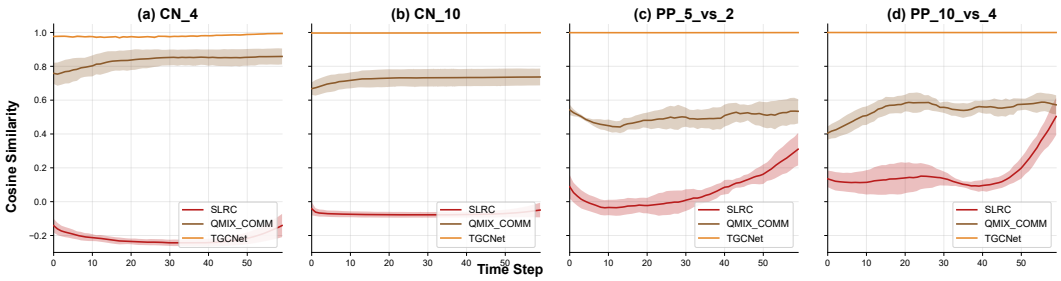


Figure 12: Temporal pairwise similarity profiles in CN and PP.

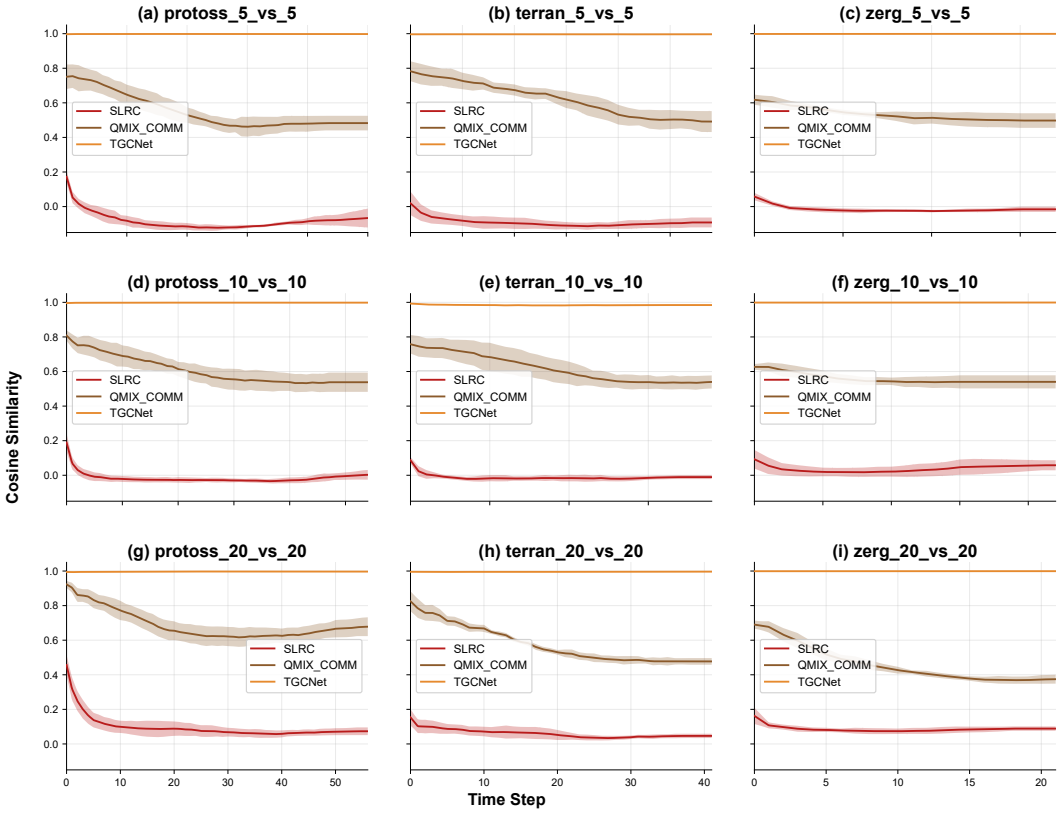


Figure 13: Temporal pairwise similarity profiles in SMACv2.

Beyond the temporal similarity profiles, Fig. 4 summarizes the post-communication representation structure using four metrics. Each metric is computed over the window $\mathcal{W} = \{20, \ldots, 40\}$, containing $T_{\mathcal{W}} = 21$ timesteps. The four metrics are defined as follows.

Mean similarity. The metric $\mathrm{MeanSim} = T_{\mathcal{W}}^{-1} \sum_{t \in \mathcal{W}} \mathrm{AvgSim}_t$ averages pairwise similarity over the time window. A larger value indicates stronger alignment among agent representations.

Pairwise dispersion. At timestep t, define $\sigma_t = \left[(P-1)^{-1} \sum_{(i,j) \in \mathcal{P}} (s_{i,j}^{(t)} - \mu_t)^2 \right]^{1/2}$ and $\mu_t = P^{-1} \sum_{(i,j) \in \mathcal{P}} s_{i,j}^{(t)}$. The metric $\mathrm{Dispersion} = T_{\mathcal{W}}^{-1} \sum_{t \in \mathcal{W}} \sigma_t$ averages within-timestep variation. A larger value indicates more heterogeneous relationships among agent pairs.

Negative-pair ratio. The metric $\mathrm{NegRatio} = (T_{\mathcal{W}} P)^{-1} \sum_{t \in \mathcal{W}} \sum_{(i,j) \in \mathcal{P}} \mathbb{I}[s_{i,j}^{(t)} < 0]$ measures the proportion of agent pairs with opposing representation directions. It describes directional differentiation rather than harmful communication.

Table 3: Wall-clock training time (hours) on two AMD EPYC 9654 CPUs and eight RTX 4090 GPUs. Green and red mark the fastest and slowest communication methods per scenario, while gray denotes communication-free QMIX.

Method	CN		PP		Protoss			Terran			Zerg		
	4	10	5v2	10v4	5v5	10v10	20v20	5v5	10v10	20v20	5v5	10v10	20v20
SLRC	**5.5**	**7.7**	**5.3**	**7.8**	**10.5**	**15.1**	**24.5**	**13.5**	**17.2**	**26.1**	**11.2**	**15.2**	**25.6**
TGCNet	12.5	17.9	10.4	19.2	21.3	24.1	32.2	24.3	32.6	42.1	19.3	23.3	34.4
MAGIC	11.1	13.6	12.3	16.1	16.8	18.8	29.9	15.3	18.9	28.7	14.8	15.5	29.7
T2MAC	23.4	78.6	22.4	96.7	41.2	79.5	216.2	52.2	86.6	204.5	55.5	88.7	218.7
MASIA	8.2	12.5	6.1	18.5	14.1	16.7	26.4	16.4	17.9	30.2	12.5	16.1	27.6
SMS	10.8	31.2	12.5	40.6	34.4	36.9	52.7	44.3	42.2	49.6	23.2	41.5	56.3
QMIX	4.1	6.8	4.7	7.2	9.2	12.6	22.3	12.1	14.9	24.1	24.8	12.3	21.5
QMIX_COMM	7.3	9.1	7.8	8.4	11.8	17.8	31.5	15.8	17.5	34.4	12.8	15.4	32.4

Normalized effective rank. Define the agent-similarity matrix as $G_t = [s_{i,j}^{(t)}]_{i,j=1}^{N}$. Given its eigenvalues $\{\lambda_{t,k}\}_{k=1}^{N}$, numerically negative values are clipped to zero and normalized as $p_{t,k} = \lambda_{t,k}(\sum_{\ell=1}^{N}\lambda_{t,\ell})^{-1}$. The effective rank at timestep t is $r_{\mathrm{eff},t} = \exp(-\sum_{k=1}^{N} p_{t,k}\log p_{t,k})$, and its trajectory-level normalized value is $\mathrm{NormRank} = T_{\mathcal{W}}^{-1}\sum_{t\in\mathcal{W}}(r_{\mathrm{eff},t}-1)(N-1)^{-1}$. A value near zero indicates concentration around one dominant direction, while a larger value indicates that more agent-wise directions contribute substantially.

In Fig. 4, violins show the distributions, diamonds denote seed means, and circles denote equally weighted means across seeds. The error bars report hierarchical-bootstrap 95% confidence intervals. Across all three scenarios, SLRC exhibits the lowest mean similarity and the highest pairwise dispersion, negative-pair ratio, and normalized effective rank. Its mean similarities are -0.241, 0.126, and 0.070 on `CN_4`, `PP_10_vs_4`, and `protoss_20_vs_20`, compared with 0.849, 0.564, and 0.628 for QMIX_COMM, while TGCNet remains close to one. SLRC obtains pairwise dispersions of 0.353, 0.451, and 0.492, negative-pair ratios of 80.0%, 44.9%, and 48.0%, and normalized effective ranks of 0.669, 0.374, and 0.207. The latter correspond to approximately 3.01, 4.36, and 4.93 effective directions. The variation from negative similarity in CN to low positive similarities in PP and SMACv2 indicates that the degree of differentiation depends on the task. Together, the temporal profiles and trajectory-level statistics show that SLRC maintains greater agent-specific and spectral diversity than QMIX_COMM and TGCNet. Combined with the task results, these findings demonstrate that strong coordination can coexist with differentiated agent representations.

I.3 Wall-Clock Training Time

Table 3 reports the wall-clock training time of SLRC and the baselines. SLRC requires the shortest training time among the communication-based methods in every scenario. Its streamlined architecture constructs compact packets once at each decision step and performs a single r-dimensional receiver-side solve, avoiding repeated communication blocks during network unrolling. QMIX is included as a communication-free reference and is distinguished from the communication methods in the table.

J Analysis of Communication within Field of View

Fig. 14 presents the performance profiles of SLRC, QMIX_COMM, and TGCNet under restricted field-of-view (FoV) communication, with specific FOV constraint configurations across diverse scenarios detailed in Appendix G.4. Empirically, even when subjected to such localized communication topologies, SLRC consistently demonstrates remarkable resilience and sustains its performance.

K Effect of Coupling Rank

The coupling rank r controls both the interaction capacity and the packet dimension $r^2 + r$. Fig. 15 and Table 4 evaluate its effect on `CN_10`, `PP_10_vs_4`, and `protoss_10_vs_10`. At $r = 4$, SLRC transmits only 20 values in one logical exchange, whereas QMIX_COMM transmits 128 values

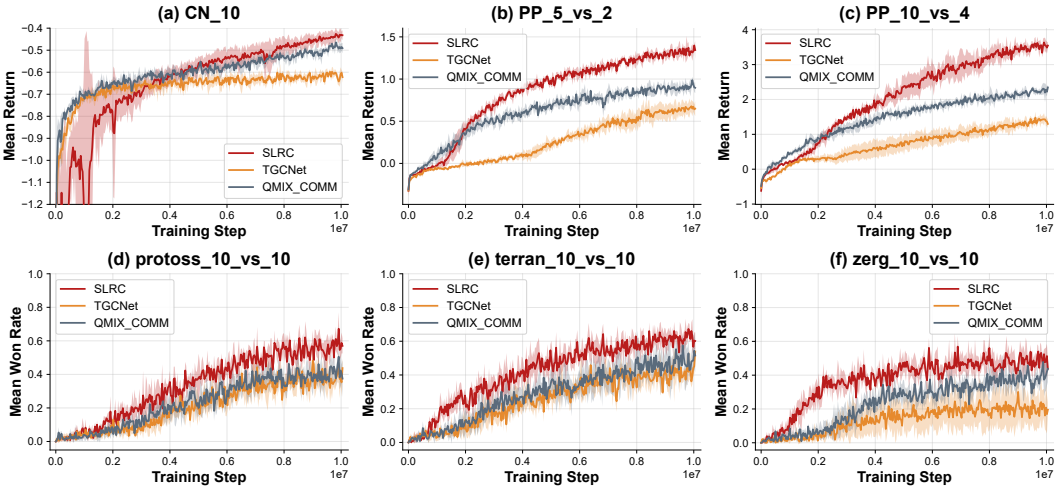


Figure 14: Performance comparison of SLRC, QMIX_COMM, and TGCNet under FoV constraint.

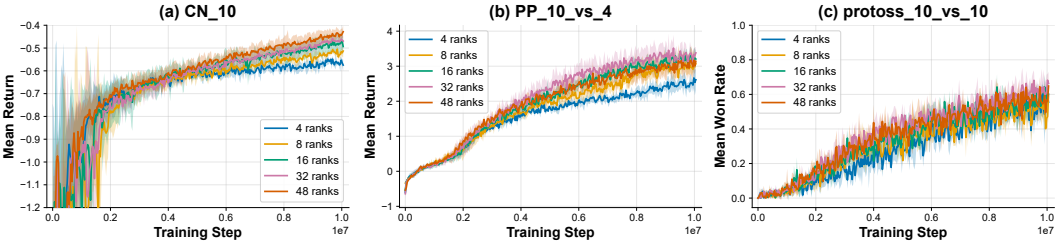


Figure 15: Impact of different ranks on performance.

Table 4: Performance, message dimension, and logical communication rounds across SLRC ranks and baselines. Zeros in the QMIX row indicate the absence of a communication module.

Method	Message Dimension	Logical Rounds	CN_10	PP_10_vs_4	protoss_10_vs_10 (%)
SLRC ($r = 4$)	20	1	-0.57 ± 0.01	2.49 ± 0.23	51.8 ± 4.1
SLRC ($r = 8$)	72	1	-0.53 ± 0.01	2.96 ± 0.15	50.9 ± 5.1
SLRC ($r = 16$)	272	1	-0.49 ± 0.05	3.18 ± 0.16	56.0 ± 2.9
SLRC ($r = 32$)	1056	1	-0.47 ± 0.02	$\mathbf{3.29 \pm 0.27}$	$\mathbf{61.9 \pm 4.1}$
SLRC ($r = 48$)	2352	1	$\mathbf{-0.45 \pm 0.03}$	3.03 ± 0.31	56.7 ± 2.6
TGCNet	128	3	-0.62 ± 0.02	1.40 ± 0.07	38.6 ± 1.7
QMIX_COMM	128	3	-0.50 ± 0.03	2.06 ± 0.02	45.7 ± 3.7
MAGIC	128	3	-0.83 ± 0.07	-0.28 ± 0.42	2.3 ± 2.3
T2MAC	128	1	-2.55 ± 1.95	-0.30 ± 0.04	14.9 ± 5.3
SMS	128	1	-3.39 ± 0.67	-0.25 ± 0.05	2.4 ± 1.4
MASIA	128	1	-0.98 ± 0.21	0.14 ± 0.11	24.1 ± 1.4
QMIX	0	0	-0.79 ± 0.03	0.04 ± 0.18	11.4 ± 2.1

per round over three rounds. Despite this compact communication, SLRC ($r = 4$) obtains -0.57, 2.49, and 51.8% on the three tasks, respectively. Compared with the strongest evaluated baseline, QMIX_COMM, this configuration is slightly weaker on CN_10 but performs better on PP_10_vs_4 and protoss_10_vs_10. These results show that a small rank can support strong coordination.

Increasing r generally improves performance until sufficient interaction capacity is reached. On CN_10, performance gradually improves from -0.57 at $r = 4$ to -0.45 at $r = 48$. On PP_10_vs_4, the gains from $r = 4$ to $r = 8$, 16, and 32 decrease from 0.47 to 0.22 and 0.11, before performance drops by 0.26 at $r = 48$. Similarly, protoss_10_vs_10 reaches its highest win rate of 61.9% at $r = 32$ and decreases by 5.2 percentage points at $r = 48$. These results indicate that increasing interaction capacity yields diminishing returns and does not necessarily improve coordination once a suitable rank is reached.

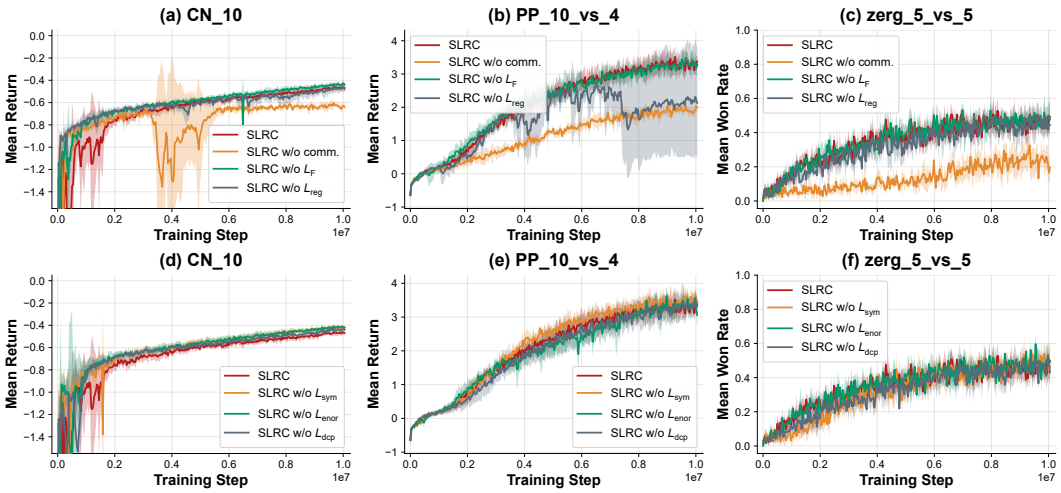


Figure 16: Ablation results for the communication module and auxiliary objectives across three tasks. Panels (a)–(c) compare SLRC with variants removing communication, $\mathcal{L}_F$, or $\mathcal{L}_{\text{reg}}$. Panels (d)–(f) compare SLRC with variants removing $\mathcal{L}_{\text{sym}}$, $\mathcal{L}_{\text{enor}}$, or $\mathcal{L}_{\text{dcp}}$.

Additionally, the packet dimension grows from 20 at $r = 4$ to 72, 272, 1056, and 2352 at $r = 8$, 16, 32, and 48, respectively. Small ranks therefore provide a favorable balance between coordination performance and message payload. However, this quadratic growth increases the communication cost of high-rank configurations, constituting a limitation of SLRC and motivating future work on further compression of the $r \times r$ coupling component.

L Ablation Studies

Fig. 16(a)–(c) evaluates the communication module, intent objective $\mathcal{L}_F$, and factor regularization objective $\mathcal{L}_{\text{reg}}$. The no-communication variant removes structured packet exchange and relies only on local recurrent representations, while the other variants remove the corresponding objectives from Eq. (10). Fig. 16(d)–(f) further evaluates the contributions of $\mathcal{L}_{\text{sym}}$, $\mathcal{L}_{\text{enor}}$, and $\mathcal{L}_{\text{dcp}}$.

Removing communication causes the largest performance degradation. Relative to full SLRC, the no-communication variant reduces performance by 33.5%, 42.4%, and 47.7% on `CN_10`, `PP_10_vs_4`, and `zerg_5_vs_5`, respectively. Because teammate-related information is excluded from local observations, agents without packet exchange must act solely from local information. The degradation confirms that the coordination gains primarily arise from inter-agent information exchange.

Figs. 17 and 18 provide a comparison of predator behavior with and without communication. Both variants begin with predators distributed across the environment at 0 seconds. In these visualizations, one second corresponds to one environment timestep, so the displayed times from 0 to 59 seconds correspond to timesteps 0 to 59. With SLRC, the predators use exchanged information to converge from multiple directions between 8 and 16 seconds. From 24 to 40 seconds, their trajectories separate into pursuit and interception roles, progressively restricting the prey's escape directions. By 48 seconds, they form a dense surround and maintain sustained containment through 59 seconds. Without communication, the predators approach targets independently at 8 seconds and form loose pursuit groups behind the prey from 16 to 32 seconds. Repeated trajectory following continues from 40 to 48 seconds, and no closed encirclement emerges by 59 seconds. This comparison illustrates how SLRC integrates complementary local information to support convergence, directional blocking, and persistent containment under limited observability.

The objective-level ablations first examine $\mathcal{L}_F$. Removing this objective causes only limited changes across the tasks. Since $\mathcal{L}_F$ encourages the intent representation $f_{i,t}$ to preserve information relevant to local action values before coupled decoding, the task loss still shapes this representation through end-to-end optimization. Its limited ablation effect therefore suggests that $\mathcal{L}_F$ provides auxiliary representation guidance rather than serving as the primary source of SLRC's coordination gains.

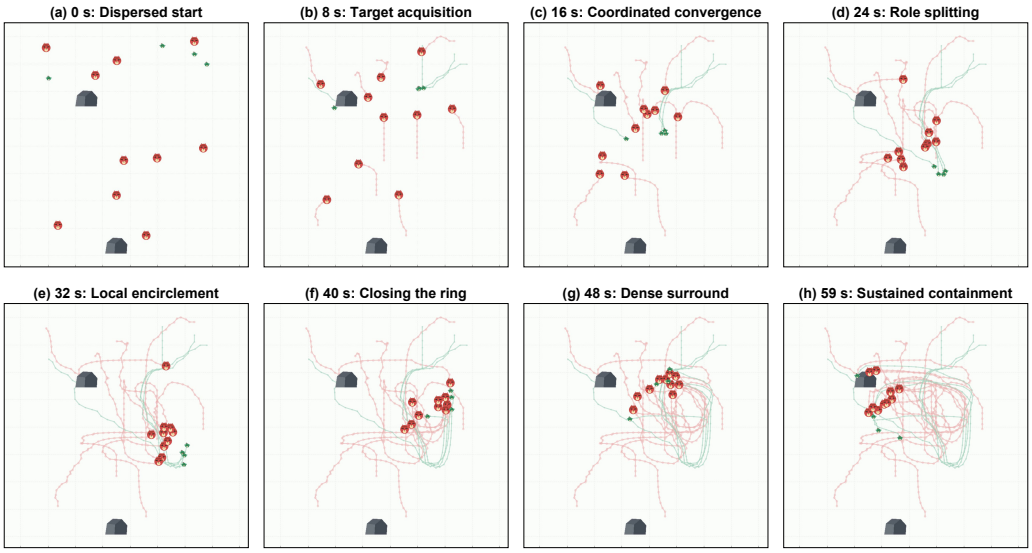


Figure 17: Predator-prey trajectories with SLRC communication on `PP_10_vs_4`. Red and green markers denote predators and prey, while black regions denote obstacles

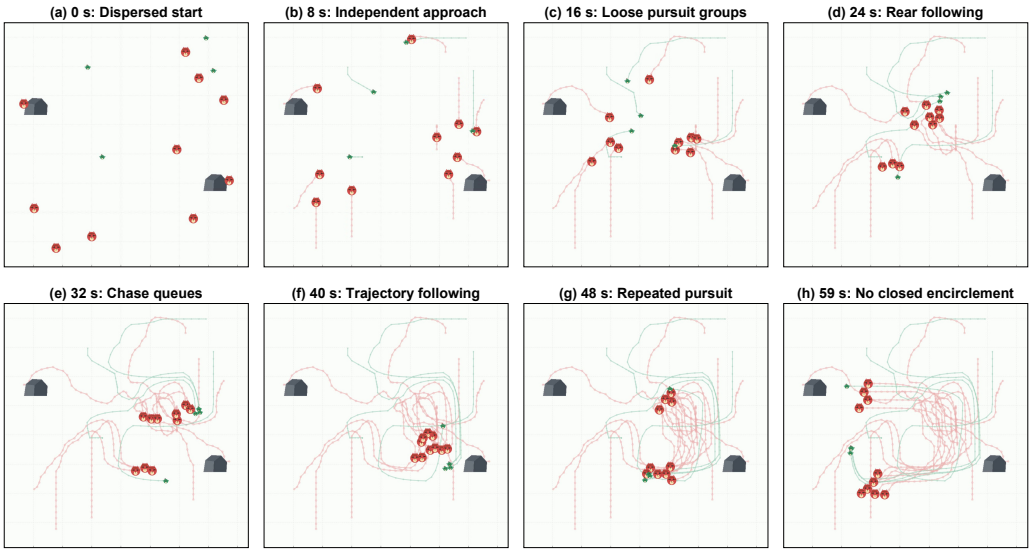


Figure 18: Predator-prey trajectories without communication on `PP_10_vs_4`. Red and green markers denote predators and prey, while black regions denote obstacles.

The remaining ablations examine $\mathcal{L}_{\text{reg}}$, which combines three forms of structural guidance. The symmetry term $\mathcal{L}_{\text{sym}}$ encourages reciprocal interactions, $\mathcal{L}_{\text{enor}}$ promotes compatible reduced interaction channels, and $\mathcal{L}_{\text{dcp}}$ reduces directional overlap between the agent-specific and interaction pathways. Removing their combined objective $\mathcal{L}_{\text{reg}}$ destabilizes training and reduces final performance, particularly on `CN_10` and `PP_10_vs_4`. On `PP_10_vs_4`, the final score decreases by 34.3% to 2.16 ± 1.89, indicating that jointly guiding the interaction factors improves optimization reliability. The component-level results in Figs. 16(d)–(f) and 19 provide complementary evidence. Removing any single regularizer causes smaller changes than removing $\mathcal{L}_{\text{reg}}$ entirely, while retaining only $\mathcal{L}_{\text{sym}}$ or $\mathcal{L}_{\text{enor}}$ produces more stable performance than retaining only $\mathcal{L}_{\text{dcp}}$. The former two terms directly guide algebraic relations between U and V, whereas $\mathcal{L}_{\text{dcp}}$ separates the agent-specific and interaction pathways. These results indicate that directly regularizing the algebraic relations

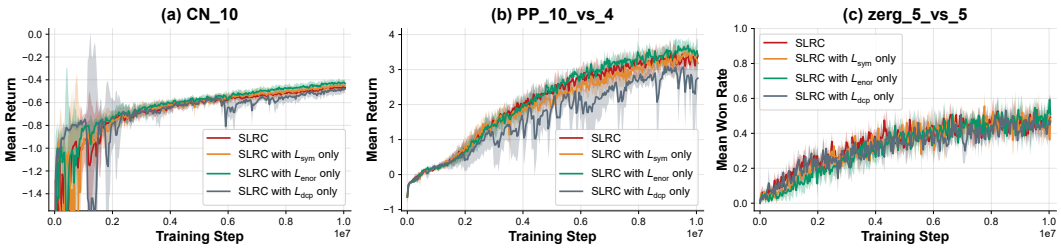


Figure 19: Single-regularizer ablations retaining only one component of $\mathcal{L}_{\text{reg}}$.

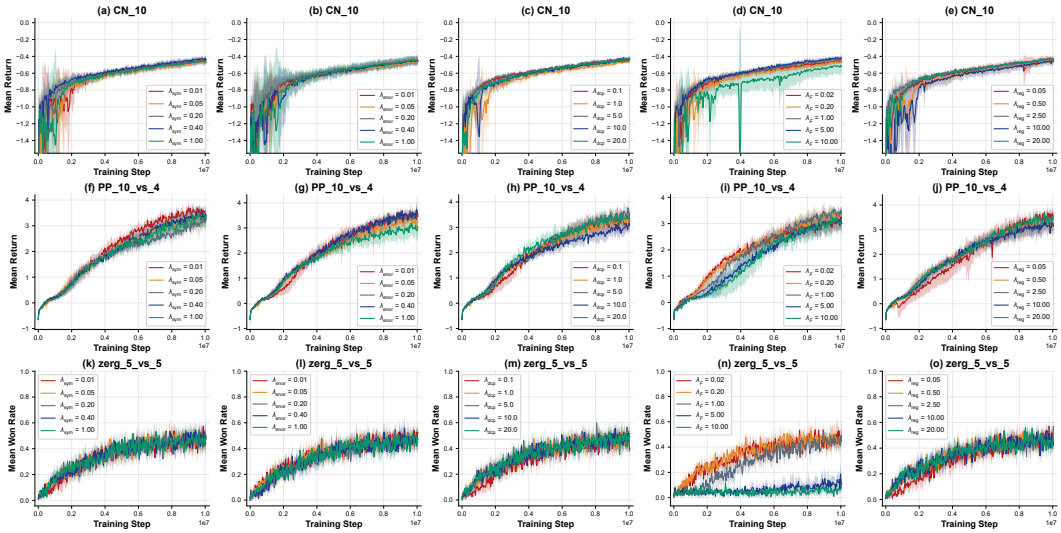


Figure 20: Sensitivity of SLRC to its five loss coefficients across three tasks.

Table 5: Stable ranges and corresponding performance intervals for SLRC. Each interval reports the minimum and maximum obtained while varying the specified coefficient.

Coefficient	Stable Range	CN_10	PP_10_vs_4	zerg_5_vs_5 (%)
λ_{sym}	$[0.01, 1.0]$	$[-0.47, -0.44]$	$[3.08, 3.53]$	$[45.7, 47.8]$
λ_{enor}	$[0.01, 1.0]$	$[-0.47, -0.45]$	$[2.97, 3.48]$	$[45.6, 47.6]$
λ_{dcp}	$[0.1, 20]$	$[-0.47, -0.43]$	$[2.98, 3.48]$	$[45.7, 47.6]$
λ_F	$[0.02, 1.0]$	$[-0.47, -0.43]$	$[3.04, 3.37]$	$[44.7, 46.0]$
λ_{reg}	$[0.05, 20.0]$	$[-0.47, -0.44]$	$[3.16, 3.52]$	$[44.9, 46.9]$

between U and V is more important for preserving the factorized interaction structure, while $\mathcal{L}_{\text{dcp}}$ provides complementary separation between the two representation pathways. Combining all three terms further improves optimization consistency.

M Hyperparameter Sensitivity

Fig. 20 and Table 5 evaluate the five loss coefficients of SLRC on CN_10, PP_10_vs_4, and zerg_5_vs_5. Each coefficient is varied independently while the remaining settings are fixed. The coefficients λ_{sym}, λ_{enor}, and λ_{dcp} determine the relative contributions of the factor regularizers. Their combined contribution is scaled by λ_{reg}, while λ_F controls the auxiliary supervision applied to the intent representation. SLRC remains stable across broad ranges of its factor-regularization coefficients. Both λ_{sym} and λ_{enor} remain effective over $[0.01, 1.0]$, while λ_{dcp} and λ_{reg} remain effective over $[0.1, 20.0]$ and $[0.05, 20.0]$, respectively. Table 5 shows that performance varies within limited intervals across these ranges. This robustness is consistent with the ablation results, where removing

one regularizer has a smaller effect than removing their combined objective. SLRC therefore does not require precise tuning of these four coefficients within the evaluated ranges.

Although the auxiliary coefficient λ_F becomes less stable at extreme values, it maintains stable performance over the broad range $[0.02, 1.0]$. As shown in panels (d), (i), and (n) of Fig. 20, increasing λ_F to 5.0 or 10.0 introduces training instability. On `zerg_5_vs_5`, these settings reduce the final win rates to 11.2% and 5.7%, respectively. This degradation suggests that excessively weighting the auxiliary intent objective can interfere with the task-driven TD objective and coupled representation learning. Overall, SLRC remains robust across broad coefficient ranges when the auxiliary objective is assigned a moderate weight.