MUTE: RETURN-PRESERVING COMMUNICATION UNLEARNING FOR EFFICIENT MULTI-AGENT COORDINATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Inter-agent communication is critical for coordinating Multi-Agent Reinforcement Learning (MARL) agents under partial observability to perform effectively in cooperative games; however, real-world bandwidth constraints demand sparse interactions. Prior approaches primarily address this trade-off by optimizing information-theoretic surrogates. We argue that these statistical proxies are fundamentally misaligned with the true objective: a message can be highly informative yet irrelevant to the joint return of the task. In this work, we propose **M**essage **U**nlearning for **T**argeted **E**fficiency (**MUTE**), a framework that views communication reduction as a value-guided machine unlearning problem. MUTE rigorously quantifies the *Counterfactual Message Value* using an attention-based estimator, and systematically unlearns the transmission of low-value messages from a policy trained without any communication constraints. This is achieved through a dual-objective mechanism that enforces communication sparsity while preserving the return of the original joint policy. We derive a theoretical upper bound on the performance gap induced by this sparsification, guaranteeing controlled return degradation. We also empirically evaluate MUTE on various complex multi-agent environments, achieving 80% to 90% bandwidth reduction while maintaining performance comparable to state-of-the-art baselines.

1 INTRODUCTION

Multi-Agent Reinforcement Learning (MARL) has emerged as a powerful paradigm for solving complex coordination problems, ranging from autonomous vehicles (Dinneweth et al., 2022) to micromanagement in StarCraft II (Samvelyan et al., 2019). To address the challenges of non-stationarity and scalability, the Centralized Training with Decentralized Execution (CTDE) (Kraemer & Banerjee, 2016) framework has become the standard, exemplified by value-based methods (Rashid et al., 2020; Yuan et al., 2022) and actor-critic approaches (Yu et al., 2022; Lowe et al., 2017; Foerster et al., 2016). However, in environments characterized by partial observability and policy-induced non-stationarity, decentralized execution often falters. Communication emerges as a pivotal solution to this challenge (Foerster et al., 2016; Sukhbaatar et al., 2016), empowering agents to transcend the limitations of local perception by exchanging diverse signals—ranging from raw observations (Singh et al.; Guan et al., 2022) and latent intentions (Yuan et al., 2022) to encoded experiences (Sun et al., 2024). This continuous information exchange serves to stabilize the learning process against environmental non-stationarity, granting agents a clearer understanding of the global state (Yuan et al., 2022; Wang et al., 2019). While unrestricted communication can mitigate partial observability, it is costly in real-world deployments. Beyond bandwidth scarcity (Wang et al., 2020a; Zhang et al., 2024), each transmission dissipates energy from an agent's limited onboard battery (Pottie & Kaiser, 2000), and frequent communication exposes agents to a higher risk of spoofing or adversarial attacks (Sun et al., 2023). Reducing communication is therefore beneficial even when bandwidth itself is not the bottleneck. Prior works have primarily addressed these costs by optimizing information-theoretic (Wang et al., 2020a; 2019; Yuan et al., 2022) auxiliary objectives to reduce communication. For instance, IMAC (Wang et al., 2020a) enforces communication sparsity via the Information Bottleneck principle, explicitly *minimizing* the mutual information between observations and messages, which implicitly assumes that high-entropy messages are redun-

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.

dant. NDQ (Wang et al., 2019) and MAIC (Yuan et al., 2022) prioritize messages by *maximizing* the mutual information between the message and the recipient's action selection. This promotes messages that alter behavior, regardless of whether that alteration leads to higher returns. By optimizing for statistical significance without directly targeting the joint return, these methods risk prioritizing messages that are merely informative over those that are truly return-critical.

In this work, we reframe communication reduction as a return-optimization problem, directly learning which messages are critical for maximizing team performance. Although SMS (Xue et al., 2022) shares this objective, it utilizes Shapley values (Shapley et al., 1953) to estimate each message's marginal contribution to the expected return. However, exact Shapley value computation requires evaluating all possible message subsets, resulting in exponential computational complexity. Consequently, SMS is forced to rely on sampling approximations that exhibit high-variance, and restrict the value function to a linear decomposition (Wang et al., 2020b), which severely limits its applicability to complex tasks requiring non-monotonic coordination. In contrast, we propose a more tractable approach to estimate the marginal contribution of a message via *Counterfactual Message Value* (CMV), which captures the difference in the joint action-value function between two counterfactual scenarios: when the message is transmitted versus when it is masked. We train a Message-Value Estimator (MVE) to approximate the CMV, and apply the MVE to identify the subset of redundant messages whose removal does not degrade the expected joint return relative to a policy with unrestricted communication.

To efficiently remove the identified redundancies without destabilizing learned coordination, we propose *Message Unlearning for Targeted Efficiency* (MUTE). Inspired by machine unlearning (Cao & Yang, 2015; Golatkar et al., 2020; Nguyen et al., 2025), which seeks to erase the influence of specific training data from trained models, we view communication reduction as a targeted unlearning task that progressively *unlearns* the transmission of low-value messages identified by our estimator. MUTE aims to *reduce communication*, not to retrain the underlying policy, and applies to any pre-trained value-based communicative policy. In this work, we start from a fully communicative policy because training it avoids the unstable joint optimization of return and communication cost (Figure 1). To ensure reliability, we derive a theoretical upper bound on the performance gap induced by unlearning, guaranteeing that any return degradation due to discarding messages is rigorously bounded within a controllable margin. We further design a dual-objective loss consisting of (i) a sparsity term that penalizes low-value messages and (ii) an anchoring term that prevents policy collapse by regularizing the agents' behavior toward the original policy.

We evaluate MUTE on diverse benchmarks, including Hallway (Wang et al., 2019), StarCraft Multi-Agent Challenge (SMAC) (Samvelyan et al., 2019), Traffic Junction (TJ) (Singh et al.), SMAC-Communication (Wang et al., 2019) and SMACv2 (Ellis et al., 2023). Empirical results demonstrate that MUTE achieves performance comparable to state-of-the-art approaches with up to 90% communication reduction.

2 Related Work

Multi-agent Communication In the paradigm of CTDE, agents learn to coordinate without explicit communication. Value-based methods like QMIX (Rashid et al., 2020) have shown success by factoring the joint value function, while MAPPO (Yu et al., 2022) has demonstrated strong performance using a centralized critic approach. However, because these methods rely solely on implicit coordination derived from training history, they often struggle under partial observability, where agents lack the local information necessary to infer the global state.

To overcome this informational bottleneck, explicit communication protocols have been introduced. Frameworks such as TarMAC (Das et al., 2019) and MAIC (Yuan et al., 2022) utilize soft attention mechanisms to weigh incoming messages, allowing agents to selectively focus on relevant information from their teammates. Building

Category	Method
No Communication	QMIX
	MAPPO
Unrestricted	MASIA
Mutual Information	NDQ
	IMAC
Gating	IC3Net
Attention	TarMAC
	MAIC
Shapley Value	SMS
Unlearning (ours)	**MUTE**

Table 1: **Comparison of Multi-agent Communication baselines.**

on this, T2MAC (Sun et al., 2024) further optimizes efficiency by learning to explicitly target specific recipients rather than broadcasting to all. MASIA (Guan et al., 2022) extends these ideas by integrating unrestricted communication into a value-based framework. Although these methods significantly improve coordination, they often ignore bandwidth constraints, leading to redundant information processing and high communication overhead.

Several works have been proposed to mitigate these costs. IC3Net (Singh et al.) employs a gating mechanism that allows agents to learn when to block communication based on local context. Information-theoretic approaches like NDQ (Wang et al., 2019) and IMAC (Wang et al., 2020a) leverage the Information Bottleneck principle to compress messages by minimizing mutual information while preserving value prediction. Similarly, SMS (Xue et al., 2022) adopts a game-theoretic perspective, estimating the Shapley value of messages to prune those with low marginal contribution. Distinct from these methods, which often struggle to balance sparsity and performance during training, our MUTE framework introduces a novel *unlearning* paradigm: we initialize with a policy without any communication constraints and progressively prune redundant channels via behavioral anchoring, ensuring efficiency without sacrificing cooperative returns.

3 Problem Formulation

We consider a fully cooperative multi-agent communication task modeled as a Decentralized Partially Observable Markov Decision Process (Dec-POMDP), defined by the tuple $\mathcal{G} = \langle \mathcal{N}, \mathcal{S}, \mathcal{A}, P, r, \Omega, O, \gamma, \mathcal{M} \rangle$. $\mathcal{N} = \{1, \ldots, N\}$ denotes the set of N agents. $\mathcal{S}$ represents the global state of the environment. $\mathcal{A}$, Ω, and $\mathcal{M}$ represent the action, observation, and message spaces, respectively, while O is the observation function. We use τ_i^t to denote the local action-observation history of agent i up to time t, $\tau_i^t = (o_i^0, a_i^0, \ldots, a_i^{t-1}, o_i^t)$.

We adopt a broadcast communication protocol. At each timestep t, each agent i generates a message m_i^t using a learnable message generator f_ψ, conditioned on its local action-observation history: $m_i^t = f_\psi(\tau_i^t)$. Under the broadcast setting, this message is transmitted identically to all other agents. Consequently, agent i receives a set of messages from its teammates, denoted as $\mathbf{m}_{-i}^t = \{m_j^t\}_{j \in \mathcal{N} \setminus \{i\}}$. After the communication phase, each agent selects an action $a_i \in \mathcal{A}$. The agent's policy $\pi(a_i | \tau_i^t, \mathbf{m}_{-i}^t)$ is conditioned on its local action-observation history τ_i^t and the set of received messages.

The joint action $\mathbf{a} = \langle a_1, \ldots, a_n \rangle$ triggers a transition to the next state $s' \sim P(\cdot|s, \mathbf{a})$ and yields a global reward $r(s, \mathbf{a})$. The objective is to learn a joint policy $\boldsymbol{\pi}$ that maximizes the expected discounted return $J(\boldsymbol{\pi}) = \mathbb{E}\left[\sum_{t=0}^{\infty} \gamma^t r_t\right] = \mathbb{E}_{\boldsymbol{\tau}, \mathbf{a}}[Q_{\text{tot}}(\boldsymbol{\tau}, \mathbf{a}; \theta)]$. In value decomposition methods (Rashid et al., 2020), the joint action-value function Q_{tot}, parametrized by $\boldsymbol{\theta}$, is composed of individual utility functions $Q_i(\tau_i, a_i; \theta)$, which determine the decentralized policies.

We adopt the CTDE paradigm. In the centralized RL training phase, the network is trained end-to-end by minimizing the standard Temporal Difference (TD) error:

$$\mathcal{L}(\theta) = \mathbb{E}_{\mathcal{D}}\left[\left(y^{\text{tot}} - Q_{\text{tot}}(\boldsymbol{\tau}, \mathbf{a}; \theta)\right)^2\right] \tag{1}$$

where $y^{\text{tot}} = r + \gamma \max_{\mathbf{a}} Q_{\text{tot}}(\boldsymbol{\tau}', \mathbf{a}; \theta^-)$ is the target value, computed using the target network parameters θ^-, and τ' represents the action-observation history at state s'. We denote the converged joint action-value function as Q_{tot}^* and the converged individual utility functions as Q_i^*. For simplicity, we omit the time index t hereafter.

4 Method

4.1 Revisiting Communication Reduction: A Return-Preserving Perspective

As summarized in Table 1, existing approaches to communication reduction largely rely on penalizing message generation during policy learning. Whether through Gating (Singh et al.), Attention (Das et al., 2019), or Mutual Information regularization (Wang et al., 2019; 2020a), these methods fundamentally treat communication as a cost to be balanced against environmental returns. This formulation forces agents to simultaneously learn *how* to cooperate (maximize return) and *when* to

remain silent (minimize communication cost), often leading to unstable convergence or suboptimal policies where agents sacrifice critical coordination to minimize penalties (Karten et al., 2023; Wei et al., 2025).

To empirically verify whether simultaneously optimizing task performance and minimizing communication cost compromises coordination, we conducted a preliminary experiment on the SMAC-Communication (`1o_2r_vs_4r`) (Wang et al., 2019) scenario, as shown in Figure 1, where agents were trained with varying communication rates for 1M steps. The results reveal that the agents' performance degrades significantly as the communication rate decreases. This empirical evidence supports our hypothesis that indiscriminate communication reduction compromises coordination. *This suggests that a better approach may be to first train the policy under unrestricted communication bandwidth, and then unlearn unnecessary communication to reach the bandwidth requirements.*

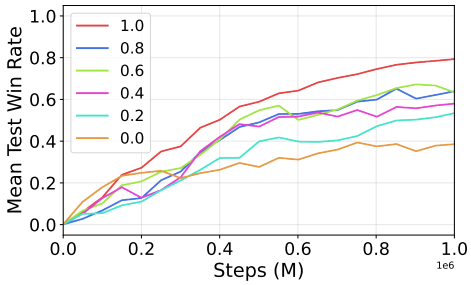


Figure 1: Impact of communication reduction on performance during MARL training. The performance degradation highlights the compromise in coordination when communication is restricted indiscriminately.

Moreover, as discussed in Section 1, by optimizing information-theoretic objectives without directly targeting the joint return, these methods (Wang et al., 2019; Yuan et al., 2022; Wang et al., 2020a) risk prioritizing messages that are merely informative over those that are truly return-critical. For example, consider agent j whose local history τ_j alone is sufficient to select the optimal action a_j^*. Suppose sender i transmits a misleading message m_{bad}. Influenced by this message, agent j instead executes a suboptimal action a_{bad}, yielding a low reward $r_{\min}$. In this case, the message strongly drives the action, producing high mutual information $\big(I(A_j; M_i) \gg 0\big)$ even as the return degrades. To address the aforementioned challenges, we formulate communication reduction as a *redundancy elimination* problem. We posit that a fully communicative expert policy naturally contains a subset of return-critical messages sufficient to maintain (near-)optimal performance, mixed with redundant messages. Therefore, instead of balancing competing objectives from scratch, our approach aims to *identify and discard* these redundant messages while preserving the original returns.

Formally, our goal is to learn a sparse communication policy $\boldsymbol{\pi}$ that minimizes total communication overhead while rigorously bounding return gap:

$$\min_{\boldsymbol{\pi}} \mathbb{E}\left[\sum_{t=0}^{T}\sum_{i=1}^{n} C(m_i^t)\right] \quad \text{s.t.} \quad J(\boldsymbol{\pi}^*) - J(\boldsymbol{\pi}) \leq \epsilon \tag{2}$$

where $C(m_i^t) \geq 0$ represents the message cost, $\boldsymbol{\pi}^*$ is the policy derived from the converged critic Q_{tot}^* (i.e., greedy w.r.t Q_{tot}^*), serving as the performance upper bound, and $J(\boldsymbol{\pi})$ represents the expected discounted return under $\boldsymbol{\pi}$.

4.2 Counterfactual Message Value Estimation

To solve the constrained optimization in Equation (2), we must strictly distinguish between essential and redundant communication. However, directly measuring the impact of a single message on the total return $J(\boldsymbol{\pi})$ is intractable due to the long time horizon. In value-based MARL, the joint action-value function Q_{tot} serves as a reliable approximation of the expected future return. Leveraging this, we introduce *Counterfactual Message Value* (CMV) to measure the value of message m_i by comparing the global Q-value with and without that message. This concept mirrors the counterfactual advantage function in COMA (Foerster et al., 2018); however, instead of estimating the advantage of action, CMV measures the *message's advantage*.

Definition 4.1 (Counterfactual Message Value)**.** The counterfactual value of broadcast message m_i is the difference in expected return caused by masking that specific message for all receivers.

$$\Delta Q_i(\boldsymbol{\tau}, \mathbf{m}) = Q_{\text{tot}}^*(\boldsymbol{\tau}, \mathbf{a}) - Q_{\text{tot}}^*(\boldsymbol{\tau}, \tilde{\mathbf{a}}) \tag{3}$$

where $\mathbf{a} \sim \boldsymbol{\pi}^*(\cdot \mid \boldsymbol{\tau}, \mathbf{m})$ is the actual joint action sampled from the optimal policy, and $\tilde{\mathbf{a}}$ is the counterfactual joint action where the actions of all teammates are resampled under the masked message:

$$\tilde{\mathbf{a}} = (a_i, \tilde{\mathbf{a}}_{-i}) \quad \text{with} \quad \tilde{\mathbf{a}}_{-i} \sim \boldsymbol{\pi}^*_{-i}(\cdot \mid \boldsymbol{\tau}_{-i}, \mathbf{m} \setminus \{m_i\}) \tag{4}$$

Here, a_i denotes the fixed action of the sender, and $\tilde{\mathbf{a}}_{-i}$ denotes the resampled actions of all other agents given that m_i is masked. For notational convenience, we denote the values of these induced joint actions as $Q^*_{\text{tot}}(\boldsymbol{\tau}, \mathbf{m}) \triangleq Q^*_{\text{tot}}(\boldsymbol{\tau}, \mathbf{a})$ and $Q^*_{\text{tot}}(\boldsymbol{\tau}, \mathbf{m} \setminus \{m_i\}) \triangleq Q^*_{\text{tot}}(\boldsymbol{\tau}, \tilde{\mathbf{a}})$. Intuitively, ΔQ_i measures the return gap caused by silencing agent i. If $\Delta Q_i \approx 0$, this specific signal is redundant given the current context and can be pruned without degrading performance. Conversely, a large positive ΔQ_i indicates a return-critical signal that must be preserved.

However, calculating the exact counterfactual baseline requires marginalizing over the receiver's policy and repeatedly querying the centralized critic Q^*_{tot} for every communication channel for both masked and unmasked scenarios. Therefore, estimating the CMV of all messages in a single communication round requires $O(N(N-1))$ forward passes. This computational burden is prohibitively expensive, particularly for the downstream unlearning task, which demands efficient value estimation.

To address this inefficiency, we design the *Message Value Estimator* (MVE) using an attention mechanism. Crucially, this formulation allows us to predict the marginal values for all agents simultaneously in a single forward pass. The choice of an attention architecture is motivated by two key properties. First, the CMV must be *permutation-invariant*, treating the incoming messages as an unordered set rather than a fixed sequence. Second, the CMV is *context-dependent*; for instance, if multiple agents broadcast identical information, the marginal value of any single message diminishes, as masking it does not result in information loss. The self-attention mechanism naturally satisfies both requirements. During the message value learning phase, the objective is:

$$\mathcal{L}_{\text{MVE}}(\phi) = \frac{1}{N} \sum_{i=1}^{N} \left(V_\phi(\mathbf{m})_i - y_i\right)^2 \tag{5}$$

where $V_\phi(\mathbf{m})$ is the MVE parameterized by ϕ, $V_\phi : \mathcal{M} \to \mathbb{R}^N$. The target y_i represents the marginal contribution of agent i's message, calculated using Equation (3). The MVE takes the joint set of all messages $\mathbf{m}$ as input. Because each message is generated directly from an agent's local observation history ($\mathbf{m} = f_\psi(\boldsymbol{\tau})$), $\mathbf{m}$ inherently encapsulates this vital state information. We verified this via an ablation study of a state-and-message variant, $V_\phi(s, \mathbf{m})$, which showed no performance gains and higher communication rates compared to our message-only model (full details are provided in the Appendix G).

4.3 Communication Unlearning

To improve communication efficiency, we define the communication cost $C(\mathbf{m})$ in Equation (2) as the L_1 norm of low-value messages, yielding the sparsity loss:

$$\mathcal{L}_{\text{sparse}}(\psi) = \sum_{m \in \mathcal{M}_{\text{red}}} \|m\|_1 \tag{6}$$

where $\mathcal{M}_{\text{red}} = \{m_i \mid V_\phi(\mathbf{m})_i \leq \lambda \mu_{\text{MVE}}\}$ denotes the set of messages whose estimated CMV falls below the adaptive sparsity threshold. Here, λ is a scaling hyperparameter, and μ_{MVE} represents the exponential moving average of MVE predictions. Rather than removing these messages outright, the L_1 penalty applies a continuous gradient pressure on their generators, gradually attenuating low-value channels while leaving the optimization free to revive any message whose CMV rises above λ as the policy evolves.

Moreover, we approximate the return gap using *Behavioral Anchoring*, which minimizes the divergence between the current policy Q_i and the frozen baseline Q^*_i:

$$\mathcal{L}_{\text{anchor}}(\theta, \psi) = \sum_{i=1}^{N} \|Q^*_i(\tau_i, a_i, \mathbf{m}_{-i}) - Q_i(\tau_i, a_i, \mathbf{m}_{-i}; \theta)\|_2^2 \tag{7}$$

By anchoring the individual value estimates rather than the global Q_{tot}, we ensure that each agent preserves its specific role and local policy stability, preventing the decentralized policies from collapsing as the communication channels are pruned.

Finally, we substitute the sparsity cost and the anchoring constraint into the Lagrangian objective. This yields the final unlearning loss, optimized over both the policy parameters θ and the message generator parameters ψ:

$$\mathcal{L}(\theta, \psi) = \mathcal{L}_{\text{sparse}}(\psi) + \beta \mathcal{L}_{\text{anchor}}(\theta, \psi) \tag{8}$$

A subtle case arises when two messages carry the same useful information. In such a case, the CMV of each message may be near zero, since masking either one alone causes little or no return loss. We show in Appendix F that the joint objective $\mathcal{L}$ optimizes toward keeping exactly one of the two messages, without jointly suppressing useful information. The complete training procedure for MUTE is summarized in Algorithm 1 in Appendix B.

To rigorously justify this intuition, we now provide a theoretical analysis quantifying the maximum performance degradation induced by our unlearning process.

Theorem 4.2 (Performance Bound of Unlearning)**.** *Let Q^*_{tot} and Q_{tot} denote the joint value functions of the converged policy $\boldsymbol{\pi}^*$ and the unlearned policy $\boldsymbol{\pi}$, respectively. Under Assumption A.1, and assuming MVE has a bounded error $|V_\phi(m) - \Delta Q(m)| \leq \delta$, alongside an action-value approximation error bounded by $\|Q^*_{tot} - Q_{tot}\| \leq \epsilon$, the performance gap is bounded by:*

$$J(\boldsymbol{\pi}^*) - J(\boldsymbol{\pi}) \leq \frac{1}{1-\gamma} \mathbb{E}\left[2\epsilon + |\mathcal{M}_{red}|(\lambda + \delta) + L \sum_{m_i \in \mathcal{M}_{red}} \|m_i\|^2 \right] \tag{9}$$

where γ is the discount factor, λ is the sparsity threshold, and L is the smoothness constant.

See Appendix A for the detailed proof. The bounds presented in Theorem 4.2 rely on the assumption of bounded estimation errors, a standard analytical practice in MARL (Kuba et al., 2022; Chen et al., 2024). To verify that these theoretical assumptions hold in practice, we tracked the empirical $\|Q^*_{\text{tot}} - Q_{\text{tot}}\|_\infty$ and the MVE loss during training, as detailed in Appendix I. This empirical evidence confirms that our objective enforces the practical conditions necessary to maintain these bounds.

5 Experiments

Our experimental design is guided by the following three research questions: **RQ1 (Communication Efficiency):** Can MUTE significantly reduce communication bandwidth without degrading the joint return? **RQ2 (Selectivity):** Does MUTE correctly identify and prune redundant messages? **RQ3 (Mechanism):** Are the MVE and Behavioral Anchoring components necessary for stable unlearning?

5.1 Experimental Setup

We implement our proposed algorithm on top of the MASIA (Guan et al., 2022) framework, which we augment with an MLP-based message generator. We select MASIA as our backbone because it is a value-based CTDE framework that supports unrestricted communication channels. This structural flexibility makes it an ideal baseline for evaluating the effectiveness of our algorithm. MUTE itself is not tied to MASIA; any value-based communicative policy can serve as its starting point.

Benchmarks. We evaluate MUTE on five benchmarks: Hallway (Wang et al., 2019), StarCraft Multi-Agent Challenge (SMAC) (Samvelyan et al., 2019)[1], Traffic Junction (TJ) (Singh et al.), SMAC-Communication (Wang et al., 2019) and SMACv2 (Ellis et al., 2023). Hallway is a diagnostic grid-world environment requiring precise temporal coordination, where agents must navigate to designated positions simultaneously to succeed. Traffic Junction (Singh et al.) simulates a traffic control scenario where agents travel along intersecting routes to reach destinations. It challenges agents to negotiate right-of-way and avoid collisions under limited field-of-view, making communication essential for resolving congestion. SMAC and SMAC-Communication serve as standard testbeds for cooperative micromanagement, featuring high-dimensional combat scenarios in StarCraft II with heterogeneous units. SMACv2 extends this by introducing stochastic start positions and random unit distributions, designed to prevent strategy memorization and test the agents' ability to generalize to unseen tactical configurations.

[1] We use StarCraft II version SC2.4.6.2.69232.

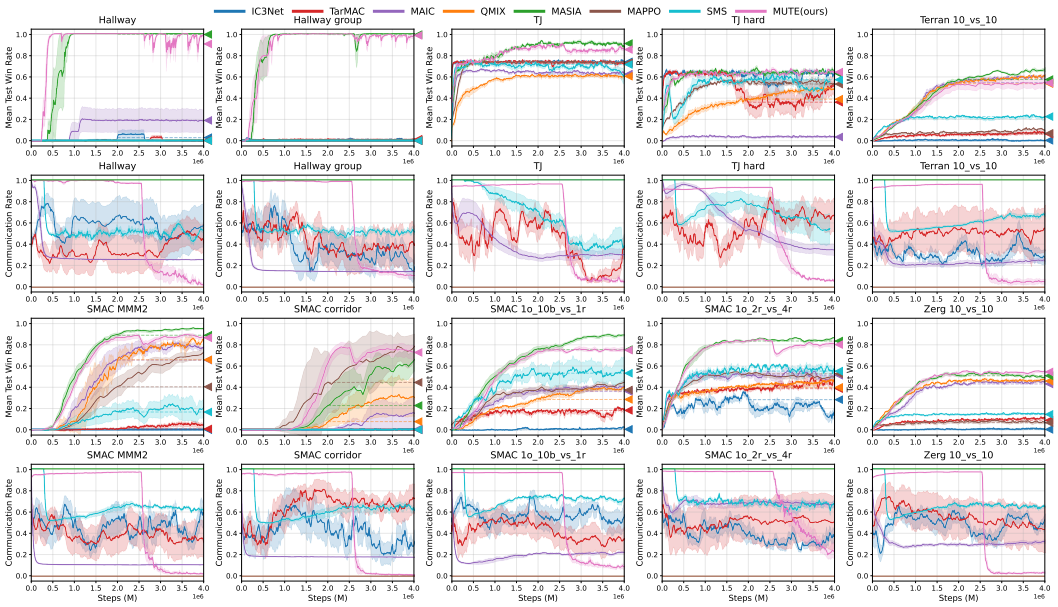


Figure 2: Comparison of performance and communication efficiency against baselines across 10 benchmarks. Plots display mean test win rates and communication rates over 4 million steps. **MUTE follows a three-stage curriculum: MARL pre-training (0–2M steps), MVE training (2–2.5M steps), and communication unlearning (2.5–4M steps).** Triangles on the right axis mark the win rate at 2M steps, serving as a baseline for policy quality before the MVE and unlearning phases commence.

Baselines. To validate the efficacy of MUTE, we compare it against a comprehensive suite of baselines ranging from standard value decomposition methods to advanced communication protocols. We include QMIX (Rashid et al., 2020) and MAPPO (Yu et al., 2022) as representative baselines *without* explicit communication. To evaluate the impact of restricted message exchange, we benchmark against TarMAC (Das et al., 2019), MAIC (Yuan et al., 2022), MASIA (Guan et al., 2022), IC3Net (Singh et al.) and SMS (Xue et al., 2022), which are representative MARL communication frameworks.

MUTE[2] undergoes a three-stage training process: pre-training, MVE training, and communication unlearning, taking 2M, 0.5M, and 1.5M steps, respectively. For a fair comparison, all baseline models are also trained for 4M steps using their original implementations. This budget is more than sufficient to guarantee convergence, as most baselines were shown to converge within 2M steps in their original publications. For evaluation, all results are reported with a 95% confidence interval across 5 random seeds.

5.2 RQ1: Communication Efficiency

We evaluate the communication efficiency of MUTE by comparing its converged communication rate and performance preservation against baselines across all benchmarks. As illustrated in Figure 2, MUTE consistently achieves the lowest communication overhead among all communication-enabled methods, while maintaining competitive win rates. To rigorously quantify the performance-communication trade-off, we conduct a normalized performance-communication analysis in Appendix L. By normalizing the performance with $W_{\text{norm}} = \frac{W - W_{\text{base}}}{W_{\text{full}} - W_{\text{base}} + \epsilon}$, we map the results onto a unified 2D plane. The resulting scatter plots confirm that MUTE robustly dominates the communication efficiency: it is the only method that reliably achieves maximal normalized performance ($W_{\text{norm}} \geq 1.0$) while maintaining the lowest communication bandwidth.

In the Hallway and Hallway Group environments, only MUTE and MASIA achieve a 100% win rate. However, while MASIA requires full communication, MUTE successfully unlearns redundant messages, reducing communication by 99% in Hallway and 90% in Hallway Group.

[2]Code available at `https://anonymous.4open.science/r/mute-2540/`

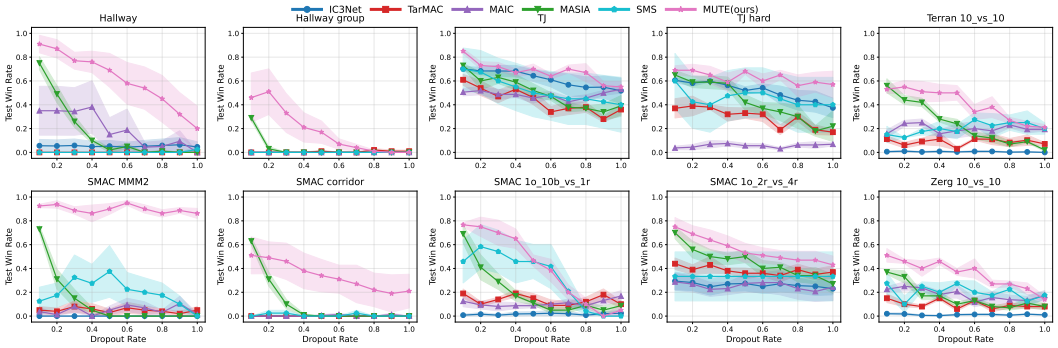


Figure 3: Comparison of test win rate under varying message dropout rates against baselines across 10 benchmarks.

In the TJ and TJ Hard scenarios, MUTE exhibits a similar pattern of efficiency. As shown in Figure 2, MUTE aggressively unlearns redundant messages, stabilizing at a communication rate of approximately 5% while matching the optimal win rates in TJ Hard and achieving performance comparable to MASIA in TJ.

MUTE also demonstrates superior communication efficiency in SMAC for high-dimensional combat scenarios compared to all baselines. In the MMM2 map, MUTE matches the optimal win rates 94% of the full-communication baseline MASIA at 2M steps, yet it has less than $<$5% communication usage, whereas methods like TarMAC and SMS plateau at 40–60% bandwidth. This advantage is even more critical in the hard-exploration Corridor map. MUTE achieves a high win rate of 73% at 4M steps, comparable to MAPPO (77%). However, it is important to note that MAPPO requires the full 4M steps of MARL training to reach this level, whereas MUTE essentially freezes its policy updates after 2M steps. As indicated by the triangles at the right side of the plot in Figure 2, MAPPO's win rate is only 42% when trained with 2M steps. More importantly, MUTE sustains this performance with near-zero ($\approx$1%) communication, which is far more efficient compared to other communicative baselines.

Similarly, in the 1o_10b_vs_1r scenario, MUTE achieves a win rate comparable to MASIA while maintaining a communication rate of 10%. In the 1o_2r_vs_4r scenario, MUTE matches the optimal performance of the full-communication baseline MASIA but differentiates itself by reducing communication overhead to $\approx$20%.

In the Zerg 10_vs_10 environment, MUTE outperforms all baselines, achieving a higher win rate 55% than even the full-communication method MASIA 49%, while simultaneously dropping its communication rate to near-zero (2%) after the unlearning phase. In the Terran 10_vs_10 scenario, MUTE remains competitive at the 2M step mark, comparable to the communication-heavy baselines, while utilizing less than 5% of the available bandwidth. However, other baselines achieve higher performance at 4M steps. We attribute this to the nature of SMACv2, which highlights stochastic start positions and random unit distributions rather than explicit coordination. This environmental randomness allows non-communicative frameworks to achieve higher performance than in other environments, thereby diminishing the relative efficiency of communication-based methods.

5.3 RQ2: Selectivity

To address RQ2, we compare the win rate of MUTE across varying message dropout rates (from 10% to 100%) on 10 benchmarks against five communication-enabled baselines (see Figure 3). To quantify this robustness across the entire spectrum of communication scarcity, we calculate the Area Under the Curve (AUC) for the win rate relative to the dropout rate. Table 2 presents the summary of these AUC scores, where a higher value indicates superior resilience to information loss. As quantified in Table 2, MUTE achieves the highest AUC across all 10 benchmarks, demonstrating that the reserved messages are truly performance-preserving. In MMM2, MUTE achieves an AUC of 0.808, which is over $4\times$ higher than the best-performing baseline, SMS (0.191). Similarly, in Hallway, MUTE's AUC (0.554) dwarfs the runner-up MAIC (0.178). The visual trends in Figure 3 reveal distinct decay patterns between MUTE and the baselines: Although methods like MASIA start with a similar win rate at the beginning, they exhibit a steep negative slope. For instance,

Environment	IC3Net	TarMAC	MAIC	MASIA	SMS	MUTE (ours)
Hallway	0.048	0.000	0.178	0.131	0.001	0.554
Hallway Grp	0.000	0.005	0.000	0.018	0.000	0.157
10b_vs_1r	0.015	0.121	0.093	0.161	0.294	0.366
2r_vs_4r	0.236	0.341	0.227	0.402	0.300	0.498
MMM2	0.000	0.044	0.047	0.088	0.191	0.808
Corridor	0.000	0.000	0.003	0.074	0.008	0.302
Terran 10v10	0.005	0.079	0.181	0.209	0.185	0.364
TJ	0.558	0.386	0.433	0.446	0.468	0.609
TJ Hard	0.464	0.269	0.051	0.381	0.405	0.564
Zerg 10v10	0.011	0.087	0.162	0.139	0.178	0.319

Table 2: AUC Summary: Area Under the Dropout-WinRate Curve. Green and Red highlights indicate the highest and lowest values, respectively.

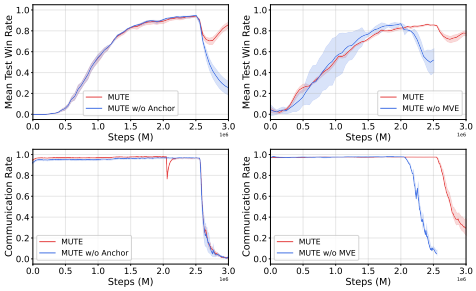


Figure 4: Ablation study of *w/o Anchoring* on SMAC MMM2 (left) and *w/o MVE* on 1o_10b_vs_1r (right).

in MMM2, Hallway, Corridor and 1o_10b_vs_1r, their win rates crash significantly as soon as the dropout rate exceeds 20–30%, suggesting a high dependency on full communication. In contrast, MUTE exhibits a gentle decay curve. Notably, in MMM2, MUTE maintains a near-optimal win rate (≈90%) even as the dropout rate approaches 80%.

5.4 RQ3: Ablation Studies

Finally, for RQ3, we verify the necessity of our dual-objective mechanism by ablating key components. We evaluate two variants of MUTE: **(1) w/o Anchoring ($\beta = 0$):** We minimize only the sparsity loss $\mathcal{L}_{\text{sparse}}$ without the anchoring constraint. **(2) w/o MVE:** We replace the value-guided redundancy set $\mathcal{M}_{\text{red}}$ with a random selection of the same sparsity.

In Figure 4, *w/o Anchoring* suffers a catastrophic collapse immediately upon triggering the unlearning process. As the communication rate drops, the test win rate plummets from ≈95% to below 30%. This confirms that $\mathcal{L}_{\text{anchor}}$ is essential for safe unlearning. Without the anchoring constraint forcing, the policy fails to internalize the coordination logic, leading to immediate failure when the communication channel is restricted. The *w/o MVE* variant was pre-trained for 2M steps and 0.5M steps for unlearning; while MUTE trained for extra 0.5M steps for MVE, that causes the curve to shift by 0.5M steps. *w/o MVE* fails to maintain performance as sparsity increases. While MUTE maintains a win rate of ≈80% even after the communication rate drops significantly, the win rate of *w/o MVE* crashes to ≈50%. This validates the Value-Awareness of the MVE module. Randomly removing messages—even at the same sparsity level—inevitably cuts critical information. MVE ensures that only redundant messages are unlearned, preserving the critical coordination signals.

We present additional ablation experiments on hyperparameters (e.g., λ, β, and the norm constraints of $\mathcal{L}_{\text{sparse}}$) in Appendix H. Furthermore, Appendix M provides a budget-sensitivity analysis of our three-stage training schedule, confirming that our allocation was not chosen arbitrarily, but rather optimized to prevent premature unlearning under a fixed 4M-step budget.

Relation to Learned Empty Messages. A natural question is whether, without unlearning, standard MARL agents can simply learn to output empty messages if provided with gating mechanisms or direct communication penalties. As we detail in Appendix J, standard MARL struggles to reduce communication bandwidth purely through environmental reward feedback due to weakly attributable team rewards. Empirical comparisons under identical training budgets demonstrate that such baselines either fail to meaningfully reduce communication or suffer severe coordination degradation, confirming the necessity of MUTE's unlearning approach.

Computational Cost and Scalability. A comprehensive profiling of MUTE's computational overhead—including training wall-clock time, inference FLOPs, and scalability across agent counts and hyperparameters—is provided in Appendix K.

6 Conclusion

We propose **MUTE**, a theoretically grounded, value-based unlearning framework that retains near-optimal SMAC performance with 80–90% less communication. Limitations are discussed in Appendix N.

AI USE STATEMENT

In this work, we used generative AI tools for assisting with code implementation. We have not used generative AI tools for any other tasks with required disclosure (including developing theoretical models or conceptual frameworks, formulating or proving mathematical claims, proposing hypotheses, designing experiments, or interpreting results), and tasks such as generating synthetic datasets, translation, and qualitative thematic analysis are not applicable to this work. Additionally, we used generative AI tools for tasks with recommended disclosure, including editing the manuscript to improve readability, formatting LaTeX tables and figures, and writing evaluation and plotting scripts. We have reviewed all AI-assisted work: all AI-assisted code was thoroughly reviewed and verified for correctness across multiple benchmark runs by the authors, and all text edits were manually checked to ensure fidelity to our original claims. We take full responsibility for the final content of this work, including text, claims, and artifacts produced with the aid of generative AI.

REFERENCES

Yinzhi Cao and Junfeng Yang. Towards making systems forget with machine unlearning. In *2015 IEEE symposium on security and privacy*, pp. 463–480. IEEE, 2015.

Jingdi Chen, Tian Lan, and Carlee Joe-Wong. Rgmcomm: Return gap minimization via discrete communications in multi-agent reinforcement learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pp. 17327–17336, 2024.

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.

Joris Dinneweth, Abderrahmane Boubezoul, René Mandiau, and Stéphane Espié. Multi-agent reinforcement learning for autonomous vehicles: A survey. *Autonomous Intelligent Systems*, 2(1):27, 2022.

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.

Jakob Foerster, Gregory Farquhar, Triantafyllos Afouras, Nantas Nardelli, and Shimon Whiteson. Counterfactual multi-agent policy gradients. In *Proceedings of the AAAI conference on artificial intelligence*, volume 32, 2018.

Aditya Golatkar, Alessandro Achille, and Stefano Soatto. Eternal sunshine of the spotless net: Selective forgetting in deep networks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 9304–9312, 2020.

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.

Sham Kakade and John Langford. Approximately optimal approximate reinforcement learning. In *Proceedings of the nineteenth international conference on machine learning*, pp. 267–274, 2002.

Seth Karten, Mycal Tucker, Siva Kailas, and Katia Sycara. Towards true lossless sparse communication in multi-agent systems. In *2023 ieee international conference on robotics and automation (icra)*, pp. 7191–7197. IEEE, 2023.

Landon Kraemer and Bikramjit Banerjee. Multi-agent reinforcement learning as a rehearsal for decentralized planning. *Neurocomputing*, 190:82–94, 2016.

Jakub Grudzien Kuba, Ruiqing Chen, Muning Wen, Ying Wen, Fanglei Sun, Jun Wang, and Yaodong Yang. Trust region policy optimisation in multi-agent reinforcement learning. In *International Conference on Learning Representations*, 2022.

Ryan Lowe, YI WU, Aviv Tamar, Jean Harb, OpenAI Pieter Abbeel, and Igor Mordatch. Multi-agent actor-critic for mixed cooperative-competitive environments. In I. Guyon, U. Von Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett (eds.), *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017. URL https://proceedings.neurips.cc/paper_files/paper/2017/file/68a9750337a418a86fe06c1991a1d64c-Paper.pdf.

Thanh Tam Nguyen, Thanh Trung Huynh, Zhao Ren, Phi Le Nguyen, Alan Wee-Chung Liew, Hongzhi Yin, and Quoc Viet Hung Nguyen. A survey of machine unlearning. *ACM Transactions on Intelligent Systems and Technology*, 16(5):1–46, 2025.

Gregory J Pottie and William J Kaiser. Wireless integrated network sensors. *Communications of the ACM*, 43(5):51–58, 2000.

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.

Mikayel Samvelyan, Tabish Rashid, Christian Schroeder de Witt, Gregory Farquhar, Nantas Nardelli, Tim G. J. Rudner, Chia-Man Hung, Philip H. S. Torr, Jakob Foerster, and Shimon Whiteson. The starcraft multi-agent challenge. In *Proceedings of the 18th International Conference on Autonomous Agents and MultiAgent Systems*, AAMAS '19, pp. 2186–2188, Richland, SC, 2019. International Foundation for Autonomous Agents and Multiagent Systems. ISBN 9781450363099.

Lloyd S Shapley et al. A value for n-person games. 1953.

Amanpreet Singh, Tushar Jain, and Sainbayar Sukhbaatar. Learning when to communicate at scale in multiagent cooperative and competitive tasks. In *International Conference on Learning Representations*.

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.

Yanchao Sun, Ruijie Zheng, Parisa Hassanzadeh, Yongyuan Liang, Soheil Feizi, Sumitra Ganesh, and Furong Huang. Certifiably robust policy learning against adversarial multi-agent communication. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=dCOL0inGl3e.

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.

Yihan Wang, Beining Han, Tonghan Wang, Heng Dong, and Chongjie Zhang. Dop: Off-policy multi-agent decomposed policy gradients. In *International conference on learning representations*, 2020b.

Xinghai Wei, Tingting Yuan, Jie Yuan, Dongxiao Liu, and Xiaoming Fu. Rescom: Reward-shaped curriculum for efficient multi-agent communication learning. In *Proceedings of the 24th International Conference on Autonomous Agents and Multiagent Systems*, pp. 2153–2161, 2025.

Di Xue, Lei Yuan, Zongzhang Zhang, and Yang Yu. Efficient multi-agent communication via shapley message value. In *IJCAI*, pp. 578–584, 2022.

Chao Yu, Akash Velu, Eugene Vinitsky, Jiaxuan Gao, Yu Wang, Alexandre Bayen, and Yi Wu. The surprising effectiveness of ppo in cooperative multi-agent games. *Advances in neural information processing systems*, 35:24611–24624, 2022.

Lei Yuan, Jianhao Wang, Fuxiang Zhang, Chenghe Wang, Zongzhang Zhang, Yang Yu, and Chongjie Zhang. Multi-agent incentive communication via decentralized teammate modeling. In *Proceedings of the AAAI conference on artificial intelligence*, volume 36, pp. 9466–9474, 2022.

Ruiqi Zhang, Jing Hou, Florian Walter, Shangding Gu, Jiayi Guan, Florian Röhrbein, Yali Du, Panpan Cai, Guang Chen, and Alois Knoll. Multi-agent reinforcement learning for autonomous driving: A survey. *arXiv preprint arXiv:2408.09675*, 2024.

Appendix Table of Contents

CONTENTS

1 Introduction **1**

2 Related Work **2**

3 Problem Formulation **3**

4 Method **3**

4.1 Revisiting Communication Reduction: A Return-Preserving Perspective 3

4.2 Counterfactual Message Value Estimation 4

4.3 Communication Unlearning 5

5 Experiments **6**

5.1 Experimental Setup 6

5.2 RQ1: Communication Efficiency 7

5.3 RQ2: Selectivity 8

5.4 RQ3: Ablation Studies 9

6 Conclusion **9**

A Proof of Theorem 4.2 **15**

B Complete Algorithm for MUTE **17**

C Environments Details **17**

C.1 StarCraft Multi-Agent Challenge (SMAC) 18

C.2 SMACv2 18

C.3 Hallway 18

C.4 Traffic Junction 19

D Model Details **19**

E Training Details **19**

F Resolving Duplicate Messages Through Joint Optimization **19**

G Ablation Study: State-Aware Message Value Estimator **20**

H Hyperparameter Ablation Study **21**

I Empirical Validation of Bounded Estimation Errors **22**

J Relation to Learned Empty Messages **23**

K Computational Cost and Scalability **24**

L Normalized Performance–Communication Tradeoff **25**

M Ablation on Training Budget Allocation **25**

N Limitations and Future Work **27**

A PROOF OF THEOREM 4.2

Theorem 4.2 (Performance Bound of Unlearning)**.** *Let Q^*_{tot} and Q_{tot} denote the joint value functions of the converged policy $\boldsymbol{\pi}^*$ and the unlearned policy $\boldsymbol{\pi}$, respectively. Under Assumption A.1, and assuming MVE has a bounded error $|V_\phi(m) - \Delta Q(m)| \leq \delta$, alongside an action-value approximation error bounded by $\|Q^*_{tot} - Q_{tot}\| \leq \epsilon$, the performance gap is bounded by:*

$$J(\boldsymbol{\pi}^*) - J(\boldsymbol{\pi}) \leq \frac{1}{1-\gamma}\mathbb{E}\left[2\epsilon + |\mathcal{M}_{red}|(\lambda+\delta) + L\sum_{m_i \in \mathcal{M}_{red}} \|m_i\|^2\right] \tag{9}$$

where γ is the discount factor, λ is the sparsity threshold, and L is the smoothness constant.

Proof. We analyze the performance gap using the **Performance Difference Lemma** (Kakade & Langford, 2002).

$$\begin{aligned}
&J(\boldsymbol{\pi}^*) - J(\boldsymbol{\pi}) = \frac{1}{1-\gamma}\,\mathbb{E}_{(\boldsymbol{\tau},\mathbf{a})\sim\boldsymbol{\pi}^*}\left[A_{\boldsymbol{\pi}}(\boldsymbol{\tau},\mathbf{a})\right] \\
&= \frac{1}{1-\gamma}\,\mathbb{E}_{(\boldsymbol{\tau},\mathbf{a})\sim\boldsymbol{\pi}^*}\left[Q_{\text{tot}}(\boldsymbol{\tau},\mathbf{a}) - V(\boldsymbol{\tau})\right] \\
&\leq \frac{1}{1-\gamma}\,\mathbb{E}_{(\boldsymbol{\tau},\mathbf{a})\sim\boldsymbol{\pi}^*}\left[Q^*_{\text{tot}}(\boldsymbol{\tau},\mathbf{a}) + \epsilon - V(\boldsymbol{\tau})\right] \\
&= \frac{1}{1-\gamma}\,\mathbb{E}_{(\boldsymbol{\tau},\mathbf{a})\sim\boldsymbol{\pi}^*}\left[Q^*_{\text{tot}}(\boldsymbol{\tau},\mathbf{a}) + \epsilon - \mathbb{E}_{\mathbf{a}'\sim\boldsymbol{\pi}(\cdot|\boldsymbol{\tau})}\left[Q_{\text{tot}}(\boldsymbol{\tau},\mathbf{a}')\right]\right] \\
&= \frac{1}{1-\gamma}\,\mathbb{E}_{\substack{(\boldsymbol{\tau},\mathbf{a})\sim\boldsymbol{\pi}^* \\ \mathbf{a}'\sim\boldsymbol{\pi}(\cdot|\boldsymbol{\tau})}}\left[Q^*_{\text{tot}}(\boldsymbol{\tau},\mathbf{m}) - Q_{\text{tot}}(\boldsymbol{\tau},\tilde{\mathbf{m}}) + \epsilon\right]
\end{aligned} \tag{10}$$

Let $\Delta\bar{Q} = Q^*_{\text{tot}}(\boldsymbol{\tau},\mathbf{m}) - Q_{\text{tot}}(\boldsymbol{\tau},\tilde{\mathbf{m}})$. We introduce an intermediate term $Q^*_{\text{tot}}(\boldsymbol{\tau},\tilde{\mathbf{m}})$ to decompose the gap:

$$\Delta\bar{Q} = \underbrace{Q^*_{\text{tot}}(\boldsymbol{\tau},\mathbf{m}) - Q^*_{\text{tot}}(\boldsymbol{\tau},\tilde{\mathbf{m}})}_{\text{(A) Information Loss}} + \underbrace{Q^*_{\text{tot}}(\boldsymbol{\tau},\tilde{\mathbf{m}}) - Q_{\text{tot}}(\boldsymbol{\tau},\tilde{\mathbf{m}})}_{\text{(B) Approximation Error}} \tag{11}$$

Bounding Term (B) [Approximation Error]: This term represents the inability of Q_{tot} to perfectly mimic Q^*_{tot} under the sparse context, which the Anchoring Loss minimizes. Given the assumption $\|Q^*_{\text{tot}} - Q_{\text{tot}}\| \leq \epsilon$, we have: $\mathbb{E}[\text{Term B}] \leq \epsilon$.

Bounding Term (A) [Information Loss]: This term captures the value lost by masking messages. Let $K = |\mathcal{M}_{\text{red}}|$ be the total number of suppressed messages. Without loss of generality, we index the messages in this subset as $\mathcal{M}_{\text{red}} = \{m_1, \ldots, m_K\}$. Term A can be exactly expanded via a telescoping sum:

$$\begin{aligned}
\text{Term (A)} &= Q^*_{\text{tot}}(\boldsymbol{\tau},\mathbf{m}) - Q^*_{\text{tot}}(\boldsymbol{\tau},\mathbf{m}\setminus\mathcal{M}_{\text{red}}) \\
&= \sum_{j=1}^{K}\left[Q^*_{\text{tot}}(\boldsymbol{\tau},\mathbf{m}\setminus\mathcal{M}_{<j}) - Q^*_{\text{tot}}(\boldsymbol{\tau},\mathbf{m}\setminus\mathcal{M}_{\leq j})\right] \\
&= \sum_{j=1}^{K}\Delta Q_j(\boldsymbol{\tau},\mathbf{m}\setminus\mathcal{M}_{<j})
\end{aligned} \tag{12}$$

where $\mathcal{M}_{<j} = \{m_1, \ldots, m_{j-1}\}$ denotes the subset of the first $j-1$ messages removed (with $\mathcal{M}_{<1} = \emptyset$). We can relate this to the base individual marginal drops $\Delta Q_j(\boldsymbol{\tau},\mathbf{m})$ by isolating an explicit interaction term $\mathcal{I}(\mathcal{M}_{\text{red}})$:

$$\text{Term (A)} = \sum_{j=1}^{K}\Delta Q_j(\boldsymbol{\tau},\mathbf{m}) + \mathcal{I}(\mathcal{M}_{\text{red}}) \tag{13}$$

where $\mathcal{I}(\mathcal{M}_{\text{red}}) = \sum_{j=1}^{K}\left[\Delta Q_j(\boldsymbol{\tau},\mathbf{m}\setminus\mathcal{M}_{<j}) - \Delta Q_j(\boldsymbol{\tau},\mathbf{m})\right]$ represents the additive information loss caused by redundant message interactions.

Since $\Delta Q_j \leq \lambda + \delta$, applying Lemma A.2 yields:

$$\text{Term (A)} \leq |\mathcal{M}_{\text{red}}|(\lambda+\delta) + L\sum_{m_i\in\mathcal{M}_{\text{red}}}\|m_i\|^2 \tag{14}$$

Substituting this interaction bound yields the final performance gap:

$$J(\boldsymbol{\pi}^*) - J(\boldsymbol{\pi}) \leq \frac{1}{1-\gamma}\mathbb{E}\left[2\epsilon + |\mathcal{M}_{\text{red}}|(\lambda+\delta) + L\sum_{m_i\in\mathcal{M}_{\text{red}}}\|m_i\|^2\right] \tag{15}$$

□

Assumption A.1 (L-Smoothness of the Value Function)**.** Let $\mathbf{m} = [m_1^\top, m_2^\top, \ldots, m_N^\top]^\top \in \mathbb{R}^{N \times d_{\text{obs}}}$ be the concatenated continuous message vector of all agents. We assume the expected joint action-value function, $f(\mathbf{m}) := Q^*_{\text{tot}}(\boldsymbol{\tau}, \mathbf{m})$, is continuously differentiable and L-smooth with respect to the continuous message embedding $\mathbf{m}$. This implies its gradient $\nabla f(\mathbf{m})$ is L-Lipschitz continuous. A fundamental property of L-smooth functions is that they can be bounded by a quadratic Taylor expansion. For any perturbation vector $\mathbf{v}$:

$$f(\mathbf{m} + \mathbf{v}) = f(\mathbf{m}) + \langle \nabla f(\mathbf{m}), \mathbf{v} \rangle + R(\mathbf{v}) \tag{16}$$

where the non-linear remainder term is strictly bounded by $|R(\mathbf{v})| \leq \frac{L}{2}\|\mathbf{v}\|^2$.

Lemma A.2 (Bound on the Interaction Term)**.** *Let $\mathcal{M}_{red}$ be the set of redundant messages to be masked. The interaction term is bounded by:*

$$\mathcal{I}(\mathcal{M}_{red}) \leq L \sum_{m_i \in \mathcal{M}_{red}} \|m_i\|^2 \tag{17}$$

Proof. Masking a single message m_i is equivalent to applying a perturbation vector $\mathbf{v}_i$, where the i-th block of the vector is $-m_i$ and all other blocks are $\mathbf{0}$. Note that the squared norm of this perturbation is exactly the squared norm of the message: $\|\mathbf{v}_i\|^2 = \|m_i\|^2$. Masking all messages in $\mathcal{M}_{\text{red}}$ simultaneously corresponds to the joint perturbation $\mathbf{v}_{\text{red}} = \sum_{m_i \in \mathcal{M}_{\text{red}}} \mathbf{v}_i$.

The interaction term is defined:

$$\mathcal{I}(\mathcal{M}_{\text{red}}) = \Big(Q^*_{\text{tot}}(\boldsymbol{\tau}, \mathbf{m}) - Q^*_{\text{tot}}(\boldsymbol{\tau}, \mathbf{m} \setminus \mathcal{M}_{\text{red}}) \Big) - \sum_{j=1}^{K} \Delta Q_j(\boldsymbol{\tau}, \mathbf{m}) \tag{18}$$

Applying the Taylor expansion to both the joint perturbation $\mathbf{v}_{\text{red}}$ and the individual perturbations $\mathbf{v}_i$ yields:

$$\mathcal{I}(\mathcal{M}_{\text{red}}) = \Big(- \langle \nabla f(\mathbf{m}), \mathbf{v}_{\text{red}} \rangle - R(\mathbf{v}_{\text{red}}) \Big) - \sum_{m_i \in \mathcal{M}_{\text{red}}} \Big(- \langle \nabla f(\mathbf{m}), \mathbf{v}_i \rangle - R(\mathbf{v}_i) \Big) \tag{19}$$

Because $\langle \nabla f(\mathbf{m}), \mathbf{v}_{\text{red}} \rangle = \sum_{m_i \in \mathcal{M}_{\text{red}}} \langle \nabla f(\mathbf{m}), \mathbf{v}_i \rangle$ by linearity,

$$\mathcal{I}(\mathcal{M}_{\text{red}}) = -R(\mathbf{v}_{\text{red}}) + \sum_{m_i \in \mathcal{M}_{\text{red}}} R(\mathbf{v}_i) \tag{20}$$

We apply the triangle inequality to bound the absolute value of the interaction term:

$$|\mathcal{I}(\mathcal{M}_{\text{red}})| \leq |R(\mathbf{v}_{\text{red}})| + \sum_{m_i \in \mathcal{M}_{\text{red}}} |R(\mathbf{v}_i)| \leq \frac{L}{2}\|\mathbf{v}_{\text{red}}\|^2 + \sum_{m_i \in \mathcal{M}_{\text{red}}} \frac{L}{2}\|m_i\|^2 \tag{21}$$

Crucially, because the individual perturbations $\mathbf{v}_i$ modify disjoint, orthogonal blocks in the concatenated vector, the cross-terms in the squared norm are zero ($\langle \mathbf{v}_i, \mathbf{v}_j \rangle = 0$ for $i \neq j$). Thus, the squared norm of the joint perturbation is the sum of the squared individual norms:

$$\|\mathbf{v}_{\text{red}}\|^2 = \Big\| \sum_{m_i \in \mathcal{M}_{\text{red}}} \mathbf{v}_i \Big\|^2 = \sum_{m_i \in \mathcal{M}_{\text{red}}} \|\mathbf{v}_i\|^2 = \sum_{m_i \in \mathcal{M}_{\text{red}}} \|m_i\|^2 \tag{22}$$

Substituting this equivalence back into our bound yields the final result:

$$|\mathcal{I}(\mathcal{M}_{\text{red}})| \leq \frac{L}{2} \sum_{m_i \in \mathcal{M}_{\text{red}}} \|m_i\|^2 + \frac{L}{2} \sum_{m_i \in \mathcal{M}_{\text{red}}} \|m_i\|^2 = L \sum_{m_i \in \mathcal{M}_{\text{red}}} \|m_i\|^2 \tag{23}$$

□

B Complete Algorithm for MUTE

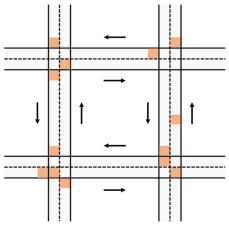
Algorithm 1 MUTE: Message Unlearning for Targeted Efficiency

Input: Pre-trained converged Policy $\boldsymbol{\pi}^*$ and Critic Q^*_{tot}, Replay Buffer $\mathcal{D}$.
Hyperparameters: Sparsity threshold λ, Anchor coefficient β.
Initialize: Message Value Estimator V_ϕ, Policy $\boldsymbol{\pi}_{\theta,\psi}$ (copy from $\boldsymbol{\pi}^*$).

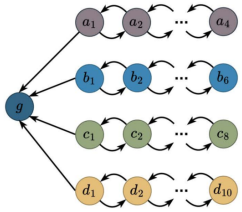
Phase 2: Counterfactual Message Value Estimation
for iteration $\leftarrow 1$ **to** K_{mve} **do**
  Sample batch $(\boldsymbol{\tau}, \mathbf{m}, \mathbf{a}, r, \boldsymbol{\tau}')$ from $\mathcal{D}$.
  for each agent i **do**
    // Counterfactual resampling: mask m_i and resample teammates
    $\tilde{\mathbf{a}}_{-i} \sim \boldsymbol{\pi}^*_{-i}(\cdot \mid \boldsymbol{\tau}_{-i}, \mathbf{m} \setminus \{m_i\})$.
    $y_i \leftarrow Q^*_{\text{tot}}(\boldsymbol{\tau}, \mathbf{a}) - Q^*_{\text{tot}}(\boldsymbol{\tau}, a_i, \tilde{\mathbf{a}}_{-i})$.
    $\hat{v}_i \leftarrow V_\phi(m_i)$.
  end for
  Take a gradient step to update ϕ by minimizing $\mathcal{L}_{\text{MVE}} = \frac{1}{N}\sum_{i=1}^{N}(\hat{v}_i - \text{sg}[y_i])^2$.
end for

Phase 3: Value-Guided Message Unlearning
for episode $\leftarrow 1$ **to** M_{train} **do**
  Collect trajectory $\boldsymbol{\tau}$ using current policy $\boldsymbol{\pi}_{\theta,\psi}$.
  for each timestep t in batch **do**
    Generate messages $\mathbf{m}^t = f_\psi(\boldsymbol{\tau}^t)$ and estimates $\hat{\mathbf{v}}^t = V_\phi(\mathbf{m}^t)$.
    Identify redundant set: $\mathcal{M}_{\text{red}} = \{m_i^t \mid \hat{v}_i^t \le \lambda\}$.
    1. Sparsity Loss: $\mathcal{L}_{\text{sparse}} = \sum_{m \in \mathcal{M}_{\text{red}}} \|m\|_1$.
    2. Anchor Loss:
      $\mathcal{L}_{\text{anchor}} = \sum_{i=1}^{N} \|Q_i^*(\tau_i^t, a_i^t, \mathbf{m}_{-i}^t) - Q_i(\tau_i^t, a_i^t, \mathbf{m}_{-i}^t; \theta)\|_2^2$.
    Take a gradient step to update θ, ψ by minimizing $\mathcal{L}(\theta, \psi) = \mathcal{L}_{\text{sparse}} + \beta\mathcal{L}_{\text{anchor}}$.
  end for
end for

C Environments Details



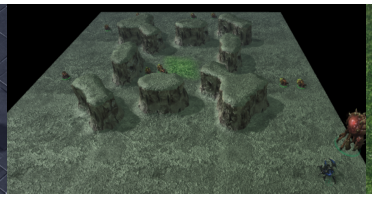
(a) Traffic Junction



(b) Hallway



(c) SMAC



(d) SMAC-Communication



(e) SMACv2

Figure 5: Experimental environments used in this work.

We evaluate our method across four diverse cooperative multi-agent domains with varying coordination requirements and observability constraints.

C.1 StarCraft Multi-Agent Challenge (SMAC)

SMAC (Samvelyan et al., 2019) and SMAC-Communication (Wang et al., 2019) provides a set of cooperative micromanagement scenarios in StarCraft II where agents must coordinate to defeat enemy units controlled by the built-in game AI. Each agent controls a single unit with local observations limited to a sight range of 9 units, including ally and enemy health, positions, and unit types within view. Agents can move in four cardinal directions, stop, or attack enemies within range (shoot range of 6 units). The team receives a shared reward based on damage dealt and eliminating enemies, with a bonus reward of 200 for winning the battle.

We evaluate on the following scenarios:

- **MMM2**: A heterogeneous battle featuring 1 Medivac, 2 Marauders, and 7 Marines against the same enemy composition. The Medivac provides healing support but cannot attack, requiring agents to learn to protect it while coordinating offensive actions. This scenario demands role-based coordination between healers and combat units.
- **Corridor**: A super-hard scenario where 6 Zealots must defeat 24 Zerglings in a narrow corridor. The confined space creates complex movement dependencies, and agents must coordinate their positions to avoid blocking each other while maximizing damage output.
- **1o_10b_vs_1r**: An asymmetric micromanagement task where 1 Overseer (detector unit without attack capability) and 10 Banelings (suicide bomber units) must defeat a single Roach. Banelings deal splash damage upon death, requiring precise timing and positioning to maximize damage while avoiding friendly fire.
- **1o_2r_vs_4r**: 1 Overseer and 2 Roaches versus 4 enemy Roaches. The numerical disadvantage requires optimal focus-fire strategies and kiting maneuvers to overcome the enemy's superior firepower.

C.2 SMACv2

SMACv2 (Ellis et al., 2023) extends SMAC with procedurally generated scenarios featuring randomized unit compositions, starting positions, and team configurations. Unlike the fixed scenarios in SMAC, SMACv2 evaluates generalization capabilities by randomizing both ally and enemy team compositions at the start of each episode. Units are sampled from weighted distributions, and starting positions follow a "surrounded and reflect" distribution where teams may start in encircled configurations.

We evaluate on two faction-specific scenarios:

- **Terran 10v10**: 10 allied Terran units versus 10 enemy units, with unit types sampled from Marines (45%), Marauders (45%), and Medivacs (10%). Marines provide ranged DPS, Marauders offer tankier ranged support with bonus damage against armored units, and Medivacs heal nearby allies.
- **Zerg 10v10**: 10 allied Zerg units versus 10 enemies, sampled from Zerglings (45%), Hydralisks (45%), and Banelings (10%). Zerglings are fast melee units, Hydralisks provide ranged attacks, and Banelings are explosive suicide units requiring careful positioning.

C.3 Hallway

The Hallway environment (Wang et al., 2019) is a coordination benchmark where n agents navigate one-dimensional corridors of varying lengths. Each agent i occupies a discrete position in a hallway of length L_i and can choose from three actions: stay, move left (decrease position), or move right (increase position). The team receives a reward only when *all* agents simultaneously reach position 0. If any agent reaches position 0 while others have not, the episode terminates with zero reward. Each agent observes only its own position, requiring implicit coordination through communication.

- **Hallway (4 agents)**: Four agents with corridor lengths $[4, 6, 8, 10]$. Agents must learn to synchronize their movements despite different travel distances, with faster agents learning to wait for slower ones. The episode limit is set to $\max(L_i) + 10 = 20$ steps.

- **Hallway-Group (7 agents)**: Seven agents divided into two groups: Group 0 (3 agents, corridor lengths $[3, 5, 7]$) and Group 1 (4 agents, corridor lengths $[4, 6, 8, 10]$). Groups must reach position 0 *sequentially*—if both groups complete simultaneously, a penalty is incurred and the episode continues. This creates a hierarchical coordination problem where agents must first coordinate within their group, then coordinate the timing between groups.

C.4 Traffic Junction

Traffic Junction (Singh et al.) simulates a multi-agent traffic coordination problem where autonomous vehicles must navigate through intersections without collisions. Cars spawn probabilistically at entry points and follow predetermined routes through the junction. Each car can either accelerate (GAS: move forward one cell) or brake (BRAKE: stay in place). Agents receive a timestep penalty of -0.01 per step while active and a collision penalty of -10 when two cars occupy the same cell. The episode succeeds when all cars reach their destinations without collisions.

- **Traffic Junction (Medium)**: A 14×14 grid with a 4-way intersection and 10 cars. Cars spawn with probability in range $[0.05, 0.2]$ and must navigate through the central junction. Each car has zero vision (observes only its own state), requiring communication to avoid collisions. The episode limit is 40 timesteps, and success requires learning implicit right-of-way protocols.
- **Traffic Junction (Hard)**: An 18×18 grid featuring a more complex road network with 8 possible routes and 20 cars. The spawning rate is lower ($[0.02, 0.05]$) but the increased number of agents and longer routes (episode limit of 80 timesteps) create more challenging coordination requirements. The grid contains multiple intersections, requiring agents to handle cascading traffic flow decisions.

D Model Details

We adopt the same backbone architecture as MASIA (Guan et al., 2022). The primary distinction is the inclusion of a message generator f_ψ at the sender side. This generator is parameterized as a single Linear layer mapping from d_{obs} to d_{obs} with no non-linear activation.

E Training Details

We detail the specific hyperparameters used for the Message Value Estimator (MVE) training and the subsequent unlearning phase in Table 3. Across both phases, network parameters are updated using the Adam optimizer. For the MVE (Phase 2), we utilize a multi-head attention mechanism ($H = 4$) to capture complex agent interactions, training for K_{mve} steps to ensure accurate counterfactual value estimation. In the unlearning phase (Phase 3), the sparsity threshold λ controls the aggressiveness of message pruning, while the anchoring coefficient β stabilizes the policy by penalizing deviations from the teacher's original Q-values.

Table 3: Training Hyperparameters for Phase 2 (MVE) and Phase 3 (Unlearning)

Phase 2: Context-Aware MVE			**Phase 3: Value-Conditional Unlearning**		
Sym.	**Parameter**	**Value**	**Sym.**	**Parameter**	**Value**
K_{mve}	Training steps	500,000	M_{unlearn}	Training steps	1,500,000
d_{mve}	MVE hidden dim	64	λ	Sparsity threshold	1.0
H	Attention heads	4	β	Anchoring weight	10
η_{mve}	MVE Learning rate	0.0005	τ_{ema}	EMA momentum	0.01

F Resolving Duplicate Messages Through Joint Optimization

A subtle case with the CMV in Equation (3) is that two messages carrying identical information would each receive a marginal value of zero: masking either alone causes no return loss, since the

other still carries the same information. A naive thresholding rule would then place both in $\mathcal{M}_{\text{red}}$ and suppress them jointly, eliminating the information entirely. We argue that the joint objective $\mathcal{L}$ optimizes toward keeping exactly one of the two messages, without jointly suppressing useful information.

Consider two messages m_i and m_k carrying identical information, and examine three configurations of the unlearning dynamics. (i) Both messages active: $\mathcal{L}_{\text{anchor}}$ is small but $\mathcal{L}_{\text{sparse}}$ is large, since both fall below λ and are penalized by Equation (6). (ii) Both messages suppressed: $\mathcal{L}_{\text{sparse}}$ vanishes, but the shared information is lost, the individual Q-values diverge from the frozen Q_i^*, and $\mathcal{L}_{\text{anchor}}$ grows as a result. (iii) Exactly one message active: $\mathcal{L}_{\text{anchor}}$ remains small because the information is still transmitted through the surviving channel, and $\mathcal{L}_{\text{sparse}}$ is reduced relative to (i) because only one message is being penalized. Configuration (iii) is therefore the minimizer of $\mathcal{L}$ over the three.

The unlearning phase of MUTE naturally navigates the network toward this minimizer. Driven by the opposing continuous gradients—where the sparsity loss pushes away from keeping both messages, and the behavioral anchoring loss penalizes the performance degradation of suppressing both—the optimization process dynamically adjusts the message generator and receiver policies. This continuous adjustment naturally breaks the initial symmetry between the identical messages, allowing the system to smoothly converge to configuration (iii) and eliminate the redundant message without sacrificing task performance.

G Ablation Study: State-Aware Message Value Estimator

Table 4: Comparison of Win Rate and Communication Rate between the original message-only MUTE and an ablation variant using both state and messages.

	Win Rate		Comm Rate	
Environment	**MUTE (ours)**	**MUTE state+msg**	**MUTE (ours)**	**MUTE state+msg**
`smac_1o_10b_vs_1r`	0.77 ± 0.02	0.75 ± 0.04	0.06 ± 0.01	0.05 ± 0.01
`hallway`	0.97 ± 0.09	0.96 ± 0.12	0.02 ± 0.02	0.14 ± 0.06
`smac_1o_2r_vs_4r`	0.81 ± 0.03	0.80 ± 0.02	0.23 ± 0.07	0.34 ± 0.25

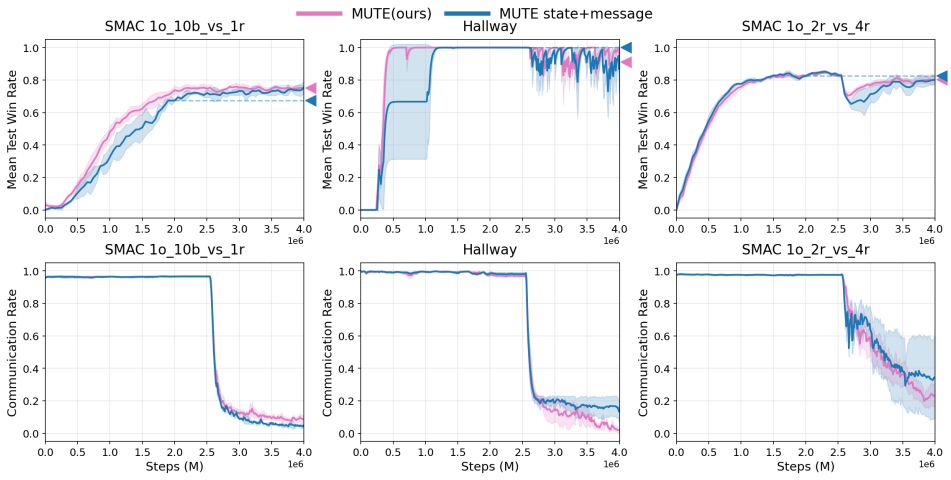


Figure 6: Training curves of Win Rate and Communication Rate between the original message-only MUTE and an ablation variant using both state and messages.

In the main text, we design the Message Value Estimator (MVE) to evaluate the marginal contribution of a message using only the joint set of messages, $V_\phi(\mathbf{m})$. While message importance may depend on the state in addition to the other agents' messages, we hypothesize that the joint message

set inherently encapsulates sufficient environmental context because each message is generated from an agent's local observation history.

To directly test this and validate our design choice, we implemented an additional variant of MUTE in which the MVE takes both the global state and the joint messages as input, denoted as $V_\phi(s, \mathbf{m})$. We compared this state-aware variant against our original message-only MVE across three environments. The results for both task performance (Win Rate) and communication bandwidth (Comm Rate) are presented in Table 4 and Figure 6.

As shown in the table and the figure, the two variants exhibit very similar task performance across all evaluated environments. Furthermore, the original message-only MVE generally achieves a better communication bandwidth trade-off. Specifically, on `hallway`, the original MVE learns substantially faster and reaches a much lower final communication rate. Similarly, on `1o_2r_vs_4r`, the `state+message` variant requires clearly higher communication after unlearning. While the state-aware variant achieves a slightly lower communication rate on `1o_10b_vs_1r`, the overall results demonstrate that adding explicit state information provides negligible benefits while often degrading unlearning efficiency. This empirical evidence strongly supports our choice of a message-only MVE formulation.

H Hyperparameter Ablation Study

In this section, we provide the detailed hyperparameter ablation study of MUTE across 10 benchmarks.

Table 5: Test Win Rate of MUTE using different values of λ across 10 benchmarks.

Method	Hallway	Hallway Grp	TJ	TJ Hard	Terran 10_vs_10	MMM2	Corridor	1o_10b_vs_1r	1o_2r_vs_4r	Zerg 10_vs_10
MUTE	1.00 ± 0.10	1.00 ± 0.08	0.84 ± 0.03	0.65 ± 0.06	0.57 ± 0.07	0.86 ± 0.05	0.72 ± 0.27	0.74 ± 0.04	0.81 ± 0.09	0.55 ± 0.03
$\lambda = 0.1 \cdot \mu_{MVE}$	0.65 ± 0.31	0.92 ± 0.06	0.83 ± 0.05	0.64 ± 0.04	0.51 ± 0.07	0.84 ± 0.03	0.54 ± 0.23	0.76 ± 0.02	0.73 ± 0.11	0.52 ± 0.03
$\lambda = 0.5 \cdot \mu_{MVE}$	0.62 ± 0.35	0.86 ± 0.13	0.81 ± 0.06	0.61 ± 0.04	0.51 ± 0.10	0.87 ± 0.05	0.50 ± 0.22	0.70 ± 0.05	0.64 ± 0.10	0.50 ± 0.04
$\lambda = 2.0 \cdot \mu_{MVE}$	0.55 ± 0.35	0.59 ± 0.10	0.76 ± 0.07	0.59 ± 0.05	0.48 ± 0.07	0.80 ± 0.03	0.49 ± 0.22	0.39 ± 0.15	0.57 ± 0.11	0.54 ± 0.03

Table 6: Communication Rate of MUTE using different values of λ across 10 benchmarks.

Method	Hallway	Hallway Grp	TJ	TJ Hard	Terran 10_vs_10	MMM2	Corridor	1o_10b_vs_1r	1o_2r_vs_4r	Zerg 10_vs_10
MUTE	0.03 ± 0.06	0.10 ± 0.10	0.02 ± 0.01	0.05 ± 0.02	0.05 ± 0.07	0.02 ± 0.01	0.01 ± 0.02	0.08 ± 0.05	0.23 ± 0.20	0.03 ± 0.02
$\lambda = 0.1 \cdot \mu_{MVE}$	0.38 ± 0.28	0.15 ± 0.08	0.09 ± 0.02	0.30 ± 0.26	0.28 ± 0.24	0.18 ± 0.19	0.20 ± 0.24	0.20 ± 0.20	0.24 ± 0.16	0.15 ± 0.10
$\lambda = 0.5 \cdot \mu_{MVE}$	0.26 ± 0.27	0.14 ± 0.04	0.02 ± 0.03	0.14 ± 0.08	0.19 ± 0.23	0.08 ± 0.11	0.03 ± 0.04	0.07 ± 0.07	0.11 ± 0.09	0.02 ± 0.02
$\lambda = 2.0 \cdot \mu_{MVE}$	0.16 ± 0.18	0.13 ± 0.09	0.01 ± 0.05	0.05 ± 0.05	0.00 ± 0.00	0.01 ± 0.01	0.02 ± 0.03	0.00 ± 0.00	0.07 ± 0.10	0.00 ± 0.00

Impact of λ in $\mathcal{M}_{\text{red}}$. We evaluate the effect of the threshold multiplier λ by comparing values of 0.1, 0.5, and 2.0 against the MUTE default of 1.0. A larger λ imposes a stricter penalty on communication, whereas a smaller λ results in a less restrictive bandwidth when selecting low-value message candidates for unlearning. Here, the threshold is scaled by μ_{MVE}, which represents the moving average of the MVE predictions during training (i.e., the expected message value). As shown in Table 5 and Table 6, utilizing a relaxed threshold ($\lambda = 0.1 \cdot \mu_{\text{MVE}}$ or $0.5 \cdot \mu_{\text{MVE}}$) yields the highest communication rates across the benchmarks. Conversely, an overly strict threshold ($\lambda = 2.0 \cdot \mu_{\text{MVE}}$) aggressively reduces the communication rate to the lowest level—even reaching 0.0 in three environments—but this severely degrades task performance. The default setting ($\lambda = 1.0$) strikes the optimal balance, successfully minimizing communication bandwidth while preserving the highest overall task performance across the benchmarks.

Table 7: Test Win Rate of MUTE using different values of β across 10 benchmarks.

Method	Hallway	Hallway Grp	TJ	TJ Hard	Terran 10_vs_10	MMM2	Corridor	1o_10b_vs_1r	1o_2r_vs_4r	Zerg 10_vs_10
MUTE	1.00 ± 0.10	1.00 ± 0.08	0.84 ± 0.03	0.65 ± 0.06	0.57 ± 0.07	0.86 ± 0.05	0.72 ± 0.27	0.74 ± 0.04	0.81 ± 0.09	0.55 ± 0.03
$\beta = 1$	0.49 ± 0.34	0.97 ± 0.03	0.81 ± 0.03	0.64 ± 0.06	0.53 ± 0.14	0.86 ± 0.04	0.57 ± 0.24	0.73 ± 0.03	0.78 ± 0.03	0.51 ± 0.05
$\beta = 50$	0.80 ± 0.31	1.00 ± 0.00	0.85 ± 0.04	0.53 ± 0.07	0.54 ± 0.09	0.88 ± 0.04	0.58 ± 0.23	0.79 ± 0.06	0.81 ± 0.04	0.52 ± 0.03

Table 8: Communication Rate of MUTE using different values of β across 10 benchmarks.

Method	Hallway	Hallway Grp	TJ	TJ Hard	Terran 10_vs_10	MMM2	Corridor	1o_10b_vs_1r	1o_2r_vs_4r	Zerg 10_vs_10
MUTE	0.03 ± 0.06	0.10 ± 0.10	0.02 ± 0.01	0.05 ± 0.02	0.05 ± 0.07	0.02 ± 0.01	0.01 ± 0.02	0.08 ± 0.05	0.23 ± 0.20	0.03 ± 0.02
$\beta = 1$	0.18 ± 0.08	0.11 ± 0.06	0.03 ± 0.01	0.04 ± 0.04	0.01 ± 0.01	0.02 ± 0.00	0.02 ± 0.03	0.02 ± 0.01	0.06 ± 0.02	0.03 ± 0.02
$\beta = 50$	0.30 ± 0.18	0.18 ± 0.06	0.07 ± 0.04	0.10 ± 0.04	0.06 ± 0.04	0.06 ± 0.04	0.04 ± 0.03	0.15 ± 0.03	0.22 ± 0.06	0.09 ± 0.01

Sensitivity to β. We ablate the hyperparameter β, which controls the weight of the behavioral anchoring during the unlearning phase, comparing the MUTE default of $\beta = 10$ against extreme values of 1 and 50. As shown in Table 7 and Table 8, setting β to a small value ($\beta = 1$) results in a severe drop in task performance (Win Rate) across the majority of environments. This indicates that when β is too weak, the model suffers from catastrophic forgetting, failing to preserve essential cooperative behaviors as messages are aggressively removed. Conversely, setting β to an overly large value ($\beta = 50$) makes the policy too rigid. While it recovers much of the task performance compared to $\beta = 1$, it resists unlearning, yielding the worst (highest) communication rates across nearly all benchmarks. The default setting ($\beta = 10$) successfully navigates this trade-off, providing an optimal level of anchoring to preserve high win rates while still allowing the model to safely and efficiently reduce communication bandwidth.

Table 9: Test Win Rate of MUTE with L_1 and L_2 norms for $\mathcal{L}_{\text{sparse}}$ across 10 benchmarks.

Method	Hallway	Hallway Grp	TJ	TJ Hard	Terran 10_vs_10	MMM2	Corridor	1o_10b_vs_1r	1o_2r_vs_4r	Zerg 10_vs_10
MUTE	1.00 ± 0.10	1.00 ± 0.08	0.84 ± 0.03	0.65 ± 0.06	0.57 ± 0.07	0.86 ± 0.05	0.72 ± 0.27	0.74 ± 0.04	0.81 ± 0.09	0.55 ± 0.03
L_2 norm	0.54 ± 0.35	0.80 ± 0.31	0.85 ± 0.03	0.57 ± 0.06	0.54 ± 0.07	0.82 ± 0.08	0.56 ± 0.23	0.65 ± 0.10	0.68 ± 0.11	0.52 ± 0.04

Table 10: Communication Rate of MUTE with L_1 and L_2 norms for $\mathcal{L}_{\text{sparse}}$ across 10 benchmarks.

Method	Hallway	Hallway Grp	TJ	TJ Hard	Terran 10_vs_10	MMM2	Corridor	1o_10b_vs_1r	1o_2r_vs_4r	Zerg 10_vs_10
MUTE	0.03 ± 0.06	0.10 ± 0.10	0.02 ± 0.01	0.05 ± 0.02	0.05 ± 0.07	0.02 ± 0.01	0.01 ± 0.02	0.08 ± 0.05	0.23 ± 0.20	0.03 ± 0.02
L_2 norm	0.31 ± 0.15	0.13 ± 0.08	0.04 ± 0.03	0.05 ± 0.03	0.05 ± 0.06	0.03 ± 0.00	0.02 ± 0.01	0.03 ± 0.02	0.22 ± 0.25	0.01 ± 0.01

Impact of the Norm Constraint in $\mathcal{L}_{\text{sparse}}$. We ablate the choice of regularization norm used in the sparsity objective, comparing MUTE's default L_1 against an L_2 norm. As shown in Table 9 and Table 10 , substituting the L_1 norm with an L_2 norm results in a degradation of task performance (Win Rate) across 9 out of 10 environments. Mathematically, the L_1 norm is well-suited for this objective because it naturally induces exact sparsity, effectively driving the values of redundant messages strictly to zero. In contrast, the L_2 norm tends to distribute the penalty, uniformly shrinking message values without discretely eliminating them. Consequently, the L_2 variant struggles to effectively sparsify communication in several environments (e.g., Hallway, where the communication rate balloons to 0.31). While the L_2 norm does achieve slightly lower communication rates on a few specific maps (such as `1o_10b_vs_1r` and `Zerg 10_vs_10`), it does so indiscriminately, sacrificing massive amounts of task performance. This confirms that the L_1 norm is essential for safely pruning the communication channel without destroying cooperative coordination.

I Empirical Validation of Bounded Estimation Errors

In Theorem 4.2, the performance bound of the unlearned policy relies on two key assumptions: the action-value approximation error is bounded by ϵ, and the Message Value Estimator (MVE) error is bounded by δ. To demonstrate that these theoretical assumptions hold in practice, we tracked the empirical max-norm deviation ($\|Q^*_{\text{tot}} - Q_{\text{tot}}\|_\infty$) and the MVE loss throughout the unlearning phase.

As summarized in Table 11, the empirical errors remain exceptionally tight across all evaluated environments. The max-norm deviation, which corresponds to ϵ, stays consistently below 1.0, ranging from 0.15 on `MMM2` to 0.48 on `1o_2r_vs_4r`. Similarly, the MVE loss, which relates to the estimation error δ, remains low and stable across the benchmarks. Furthermore, the sparsity loss is consistently near zero, indicating stable convergence during the unlearning process.

These empirical results strongly validate our theoretical assumptions. They confirm that the value estimations do not diverge wildly when messages are unlearned, and that the anchoring objective ($\mathcal{L}_{\text{anchor}}$) successfully enforces the bounded errors required to guarantee safe unlearning.

Table 11: Summary of $\|Q^*_{\text{tot}} - Q_{\text{tot}}\|_\infty$, MVE loss, and Sparsity loss per episode across 10 environments during the unlearning phase.

Env	$\|Q^*_{\text{tot}} - Q_{\text{tot}}\|_\infty$	MVE Loss	Sparsity Loss
1o_10b_vs_1r	0.46 ± 0.18	0.52 ± 0.06	0.04 ± 0.04
1o_2r_vs_4r	0.48 ± 0.06	0.38 ± 0.03	0.01 ± 0.01
MMM2	0.15 ± 0.03	0.18 ± 0.09	0.02 ± 0.01
corridor	0.39 ± 0.11	0.39 ± 0.51	0.06 ± 0.04
hallway	0.15 ± 0.05	0.41 ± 0.05	0.01 ± 0.01
hallway_group	0.44 ± 0.05	0.41 ± 0.15	0.01 ± 0.00
terran_10v10	0.43 ± 0.17	0.37 ± 0.13	0.02 ± 0.01
tj_hard	0.40 ± 0.24	0.17 ± 0.09	0.00 ± 0.00
tj_medium	0.46 ± 0.25	0.06 ± 0.06	0.00 ± 0.00
zerg_10v10	0.43 ± 0.15	0.26 ± 0.11	0.03 ± 0.01

J Relation to Learned Empty Messages

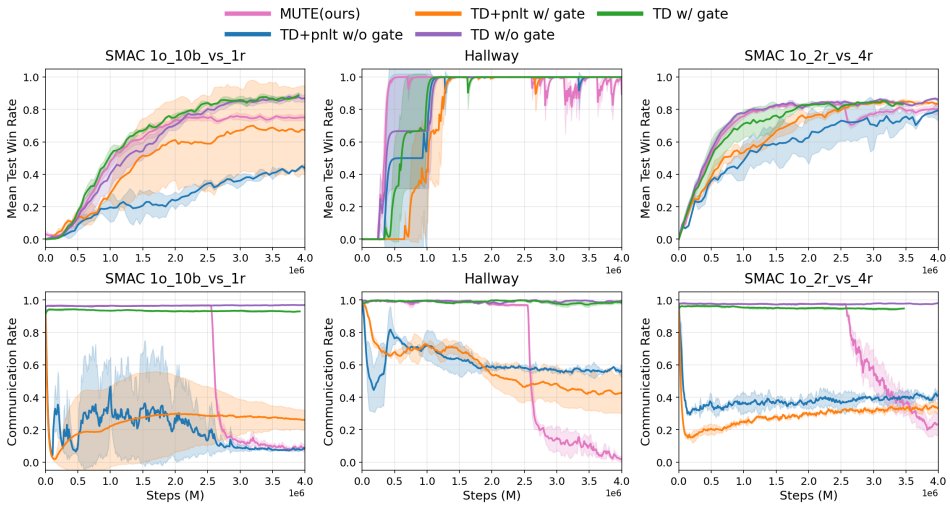


Figure 7: Training curves of Win Rate and Communication Rate between the MUTE and

Table 12: Combined Win Rate and Communication Rate (mean $\pm$ CI) for MUTE vs. Baseline Methods across Different Environments.

	Test Win Rate (%)					Communication Rate (%)				
Environment	**MUTE (ours)**	**TD+pnlt w/o gate**	**TD+pnlt w/ gate**	**TD w/o gate**	**TD w/ gate**	**MUTE (ours)**	**TD+pnlt w/o gate**	**TD+pnlt w/ gate**	**TD w/o gate**	**TD w/ gate**
smac_1o_10b_vs_1r	0.77 ± 0.02	0.44 ± 0.02	0.67 ± 0.28	0.87 ± 0.02	0.89 ± 0.01	0.06 ± 0.01	0.09 ± 0.00	0.26 ± 0.06	0.97 ± 0.00	0.93 ± 0.00
hallway	0.97 ± 0.09	1.00 ± 0.00	1.00 ± 0.00	1.00 ± 0.00	1.00 ± 0.00	0.02 ± 0.02	0.56 ± 0.04	0.43 ± 0.13	1.00 ± 0.00	0.98 ± 0.01
smac_1o_2r_vs_4r	0.81 ± 0.03	0.79 ± 0.07	0.83 ± 0.00	0.86 ± 0.01	0.83 ± 0.01	0.23 ± 0.07	0.41 ± 0.03	0.34 ± 0.03	0.98 ± 0.00	0.94 ± 0.01

A natural question arises: instead of our proposed unlearning framework, could standard RL simply learn to output an empty message (e.g., an all-zero token or via a learned gating mechanism) to reduce communication overhead? In multi-agent settings heavily reliant on shared, delayed team rewards, the benefit of silencing one specific message is weakly attributable. Consequently, standard RL struggles to reliably discover *selective silence* purely through environmental reward feedback, even when silencing is explicitly provided as an available action.

To verify this, Table 12 compares MUTE against standard RL baselines augmented with communication gating (`w/ gate`) and direct communication penalties in the loss function (`+pnlt`). Notably, while the baseline variants allocate their entire 4M-step budget to standard RL training, MUTE achieves its results under the exact same budget by partitioning it into 2M steps for RL pre-training, 0.5M steps for MVE training, and 1.5M steps for value-guided unlearning.

As the empirical results illustrate, simply providing an empty-message option (`TD w/ gate`) does not reduce the communication bandwidth; both `TD w/ gate` and `TD w/o gate` keep the communication rate near 100%. Conversely, while adding a direct penalty (`TD+pnlt w/o gate` and `TD+pnlt w/ gate`) forces a reduction in communication, the suppression is less targeted. This untargeted penalization degrades coordination and results in slower convergence. In contrast, MUTE's counterfactual unlearning allows it to maintain highly competitive win rates while achieving the lowest overall communication rate.

K Computational Cost and Scalability

Table 13: Comprehensive computational profiling. (a) compares baseline training and inference costs. (b), (c), and (d) ablate the computational cost of evaluating Eq. (3) in `traffic_junction` across different hyperparameters.

(a) Computational profiling in SMAC 1o_10b_vs_1r.

Method	Parameters	Training Time (hrs)	Inference FLOPs (M)
QMIX	85,160	6.07 ± 0.91	0.79
IC3Net	47,690	5.23 ± 1.23	0.35
TarMAC	53,930	5.42 ± 0.75	1.11
MAIC	114,607	15.6 ± 3.56	2.48
MASIA	176,029	7.82 ± 2.33	1.36
MAPPO	85,200	2.03 ± 0.15	0.26
SMS	42,524	8.45 ± 1.27	0.84
MUTE	185,149	8.09 ± 0.88	1.55

(b) Cost per timestep across agent numbers (N).

N	Time (ms)	MFLOPs	Mean Test Win Rate
2	3.077 ± 0.03	23.6	1.000 ± 0.05
3	5.405 ± 0.05	47.7	0.975 ± 0.04
5	8.053 ± 0.05	123.2	0.975 ± 0.04
7	8.802 ± 0.01	238.7	0.875 ± 0.10
10	14.589 ± 0.09	497.2	0.725 ± 0.07
15	27.791 ± 0.04	1195.9	0.850 ± 0.05
20	30.471 ± 0.11	2301.6	0.625 ± 0.08

(c) Cost per timestep across message dimensions (d_{msg}).

d_{msg}	Time (ms)	MFLOPs	Mean Test Win Rate
4	10.90 ± 0.25	458.17	0.85 ± 0.02
8	14.80 ± 0.91	497.23	0.95 ± 0.03
16	15.17 ± 3.14	575.35	0.96 ± 0.01
32	16.73 ± 1.14	731.58	0.90 ± 0.05
64	20.00 ± 2.56	1044.04	0.68 ± 0.03

(d) Cost across number of sampled experiences (N_{exp}).

N_{exp}	Time (s)	GFLOPs	Mean Test Win Rate
4	0.4346	2.49	0.78 ± 0.01
8	0.4667	4.97	0.88 ± 0.01
16	0.5772	9.94	0.90 ± 0.01
32	0.5632	19.89	0.73 ± 0.01
64	0.6156	39.78	0.88 ± 0.02

Because MUTE introduces an explicit Message Value Estimator (MVE) and a dedicated unlearning phase, it is critical to evaluate its computational overhead. All computational profiling and timing evaluations were conducted on an L40S GPU. Table 13(a) compares the training wall-clock time and inference FLOPs of MUTE against established baselines in the SMAC benchmark. While MUTE introduces additional parameters (185K) to support the MVE, its total training time (8.09 hours) remains highly competitive. It trains significantly faster than heavy communication frameworks like MAIC (15.6 hours) and is on par with MASIA and SMS. During inference, MUTE operates with 1.55M FLOPs per step, confirming that the value-guided unlearning process does not impose prohibitive forward-pass bottlenecks compared to other state-of-the-art multi-agent methods.

To understand how the counterfactual value estimation scales in denser environments, we ablate the cost of evaluating Equation (3) across a varying number of agents (N) in `traffic_junction`. As shown in Table 13(b), the computational time and MFLOPs scale as the environment grows from 2 to 20 agents. However, even at $N = 20$, the evaluation time remains practically efficient at approximately 30 ms per timestep. The mean test win rate slightly decays at extreme agent counts, which reflects the inherent environmental difficulty of large-scale traffic coordination rather than a computational bottleneck. Finally, Table 13(c) and (d) analyze the scalability of MUTE with

respect to its internal hyperparameters: message dimension (d_{msg}) and the number of sampled experiences (N_{exp}). Increasing d_{msg} yields a sub-linear growth in evaluation time, but we observe that excessively large message spaces (e.g., $d_{\text{msg}} = 64$) cause a drop in the win rate, likely due to the increased difficulty of accurately estimating counterfactual values in high-dimensional continuous spaces. Similarly, sweeping N_{exp} demonstrates that GFLOPs scale strictly linearly with the sample size. A moderate configuration (e.g., $N_{\text{exp}} = 16$) provides an optimal balance, yielding peak coordination performance without incurring unnecessary computational drag.

L Normalized Performance–Communication Tradeoff

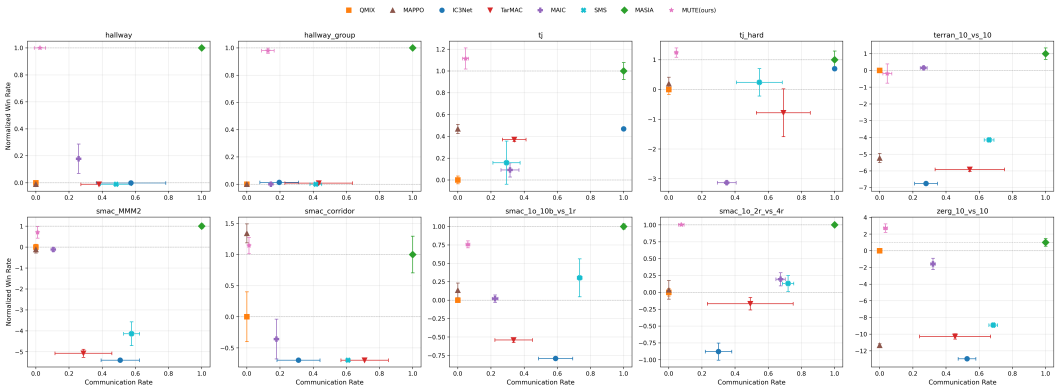


Figure 8: Comparison of performance and communication

To explicitly evaluate the tradeoff between coordination performance and communication bandwidth, we project the results of all evaluated methods onto a normalized performance–communication plane.

Specifically, for each environment, we calculate the normalized win rate (W_{norm}) as follows:

$$W_{\text{norm}} = \frac{W - W_{\text{base}}}{W_{\text{full}} - W_{\text{base}} + \epsilon} \tag{24}$$

where W is the raw win rate of the evaluated method, W_{base} is the win rate of the no-communication baseline (QMIX), W_{full} is the win rate of the full-communication baseline (MASIA), and ϵ is a small constant added for numerical stability to prevent division by zero.

Figure 8 visualizes this normalized win rate against the final communication rate for 10 diverse environments. In this 2D plane, the ideal operating region is the top-left corner, which corresponds to matching or exceeding full-communication performance ($W_{\text{norm}} \geq 1.0$) while incurring minimal communication overhead (Communication Rate ≈ 0.0).

As the scatter plots clearly demonstrate, MUTE (represented by the pink star) consistently dominates this ideal top-left region across the benchmarks. While the full-communication baseline MASIA anchors the top-right ($W_{\text{norm}} = 1.0$, Communication Rate $= 1.0$), MUTE matches or occasionally exceeds this performance (e.g., in *smac_corridor* and *tj_hard*) while reducing communication by over 90%.

Furthermore, the plots highlight the failure modes of existing communication-reduction baselines. Methods such as IC3Net, TarMAC, and SMS frequently suffer from severe drops in normalized win rates (dropping well below 0.0, indicating worse performance than the no-communication QMIX baseline) or fail to meaningfully reduce the communication rate. MUTE's value-guided counterfactual unlearning is the only approach that robustly preserves absolute coordination capabilities while successfully isolating and eliminating redundant messages.

M Ablation on Training Budget Allocation

To ensure that our reported three-stage training schedule was not chosen arbitrarily, we performed a budget-sensitivity ablation on the *smac_1o_10b_vs_1r* environment. MUTE operates on a fixed total

training budget of 4M environment steps, which must be partitioned across three distinct phases: (1) RL pre-training of the dense expert policy, (2) training the Message Value Estimator (MVE), and (3) value-guided unlearning.

In this ablation, we compare our default schedule (2M / 0.5M / 1.5M) against several alternative allocations while keeping the 4M total step budget strictly constant.

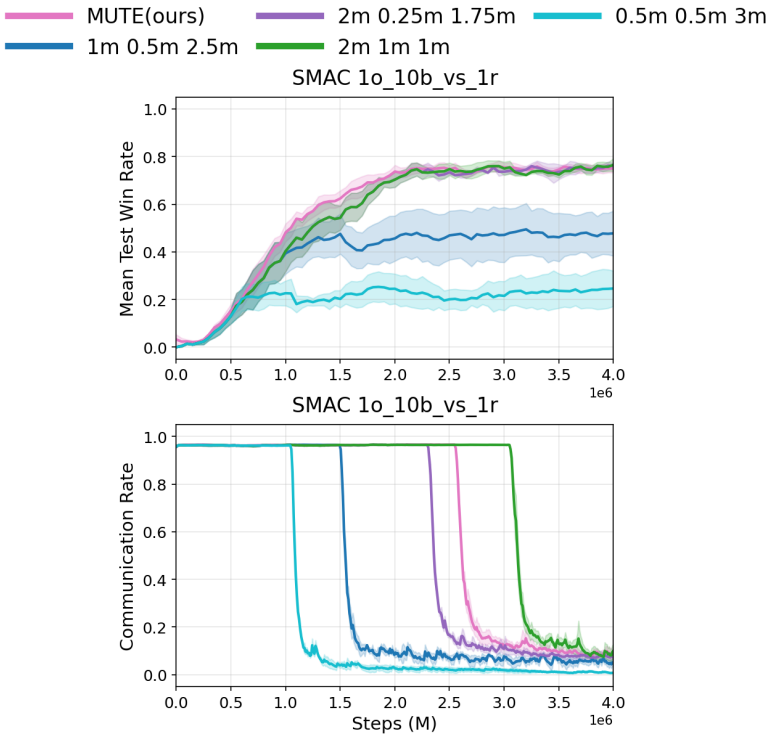


Figure 9: Training curves of Mean Test Win Rate and Communication Rate for different budget allocations in *smac_1o_10b_vs_1r*. The sharp drops in communication rate correspond to the initiation of the unlearning phase.

Table 14: Final Win Rate and Communication Rate (mean $\pm$ CI) under different training budget allocations in *smac_1o_10b_vs_1r*. Allocations are denoted in millions of steps as (Pre-training / MVE / Unlearning).

Allocation Schedule (M Steps)	**Mean Test Win Rate (%)**	**Communication Rate (%)**
2.0 / 0.50 / 1.50 (MUTE default)	0.77 ± 0.02	0.06 ± 0.01
1.0 / 0.50 / 2.50	0.48 ± 0.09	0.05 ± 0.02
2.0 / 0.25 / 1.75	0.76 ± 0.01	0.09 ± 0.01
2.0 / 1.00 / 1.00	0.77 ± 0.03	0.10 ± 0.03
0.5 / 0.50 / 3.00	0.25 ± 0.07	0.01 ± 0.00

As shown in Table 14 and Figure 9, the results demonstrate a clear and consistent trend regarding the onset of the unlearning phase. Triggering unlearning too early (e.g., the 0.5M / 0.5M / 3M or 1M / 0.5M / 2.5M schedules) substantially degrades the final coordination performance. This indicates that the dense expert policy has not yet sufficiently converged; attempting to isolate and unlearn redundant messages before the foundational communication strategy is stable leads to catastrophic performance drops.

In contrast, schedules that allocate a full 2M steps to initial pre-training perform much better overall. Among these stable configurations, altering the ratio of MVE training to unlearning (e.g., 2M / 0.25M / 1.75M or 2M / 1M / 1M) yields similar win rates, but slightly inferior communication

reduction. Our empirically chosen default allocation (2M / 0.5M / 1.5M) strikes the optimal balance: it achieves one of the strongest final win rates while simultaneously reaching a highly efficient, near-zero final communication rate.

N LIMITATIONS AND FUTURE WORK

MUTE uses CMV as a tractable local estimate of a message's marginal contribution to the joint return. This design avoids the exponential cost of subset-based attribution methods such as Shapley value estimation, but it does not explicitly enumerate all possible message subsets. As a result, higher-order interactions among messages may not be fully captured by the MVE alone. In MUTE, this limitation is mitigated by behavioral anchoring, which provides feedback during unlearning when suppressing candidate messages would alter the behavior of the full-communication expert. Nevertheless, extending CMV to subset-aware or interaction-aware estimation could provide stronger theoretical guarantees and is an important direction for future work.