2026 IEEE International Conference on Robotics and Automation (ICRA 2026)
June 1-5, 2026. Vienna, Austria

Bandwidth-Efficient Multi-Agent Communication through Information Bottleneck and Vector Quantization

Ahmad Farooq*[1] and Kamran Iqbal[1]

***Abstract*— Multi-agent reinforcement learning systems deployed in real-world robotics applications face severe communication constraints that significantly impact coordination effectiveness. We present a framework that combines information bottleneck theory with vector quantization to enable selective, bandwidth-efficient communication in multi-agent environments. Our approach learns to compress and discretize communication messages while preserving task-critical information through principled information-theoretic optimization. We introduce a gated communication mechanism that dynamically determines when communication is necessary based on environmental context and agent states. Experimental evaluation on challenging coordination tasks demonstrates that our method achieves 181.8% performance improvement over no-communication baselines while reducing bandwidth usage by 41.4%. Pareto frontier analysis shows dominance across the entire success-bandwidth spectrum, with an area under the curve of 0.198 vs 0.142 for next-best methods. Our approach significantly outperforms existing communication strategies and establishes a theoretically grounded framework for deploying multi-agent systems in bandwidth-constrained environments such as robotic swarms, autonomous vehicle fleets, and distributed sensor networks.**

Index Terms

Multi-Agent Reinforcement Learning (MARL), Efficient Communication, Information Bottleneck, Vector Quantization (VQ), Robotics

I. INTRODUCTION

The deployment of multi-agent reinforcement learning (MARL) systems in real-world robotics applications has revealed a fundamental challenge: achieving effective coordination while operating under severe communication constraints [1], [2]. Unlike simulation environments where communication is often assumed to be free and unlimited, practical robotic deployments face bandwidth limitations, latency constraints, energy budgets, and communication failures that can critically impact system performance.

This challenge is particularly acute in emerging applications such as autonomous vehicle coordination, where vehicles must share information about traffic conditions, hazards, and intentions while operating under limited V2X (C-V2X/DSRC) bandwidth [3]. Similarly, robotic swarms deployed for search-and-rescue must coordinate efficiently under intermittent connectivity and limited bandwidth [4]. Distributed sensor networks monitoring environmental conditions face the dual challenge of maximizing information sharing while minimizing energy consumption to extend operational lifetime.

Traditional approaches to multi-agent coordination typically fall into two extremes: either they assume unlimited communication bandwidth, leading to inefficient protocols that flood the network with redundant information, or they ignore communication entirely, resulting in suboptimal coordination and performance degradation. Recent advances in learned communication protocols have shown promise [5], [6], but these methods often lack principled mechanisms for controlling bandwidth usage while maintaining coordination effectiveness.

The core technical challenge lies in determining what information to communicate, when to communicate it, and how to encode it efficiently. Agents must balance the immediate cost of communication against the potential future benefits of improved coordination. This requires solving a complex optimization problem that considers both the information-theoretic properties of messages and the dynamic coordination requirements of the task.

Our work addresses this challenge by introducing a framework that combines information bottleneck theory with vector quantization to enable selective, efficient communication in multi-agent systems. The information bottleneck principle establishes a theoretical foundation for learning compressed representations that preserve task-relevant information while discarding redundant details. Vector quantization enables discrete message encoding that significantly reduces bandwidth requirements compared to continuous representations.

Key Contributions: We introduce a principled information-theoretic approach for selective communication that balances performance and bandwidth via information bottleneck optimization. Our framework includes: a gated communication mechanism that learns *when* to communicate, with detailed ablation analysis; a vector quantization scheme for efficient discrete message encoding; and a theoretical analysis of constraint enforcement. Experiments on coordination tasks demonstrate a 181.8% performance improvement over no-communication baselines with a 41.4% bandwidth reduction, and Pareto frontier analysis establishes dominance across the success-bandwidth spectrum.

II. RELATED WORK

A. Multi-Agent Reinforcement Learning

Multi-agent reinforcement learning has evolved from early centralized approaches to sophisticated decentralized methods that can handle partial observability and complex co-

*Corresponding Author: Ahmad Farooq

[1]The authors are with the Electrical and Computer Engineering Department, University of Arkansas at Little Rock, AR, 72204, USA (e-mail: afarooq@ualr.edu; kxiqbal@ualr.edu).

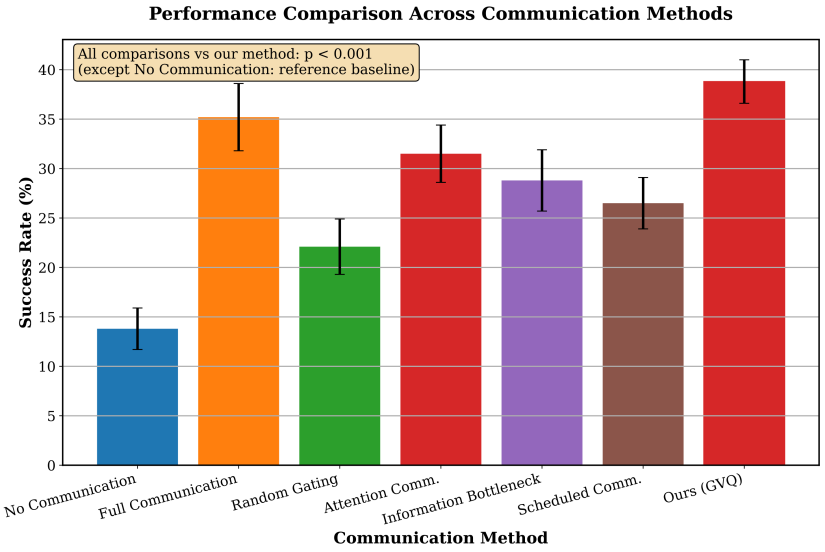

Fig. 1. Performance comparison showing success rates across communication methods. Our GVQ approach achieves a 38.75% success rate, representing 181.8% improvement over a no-communication baseline (13.75%) with statistical significance $p < 0.001$. Error bars show 95% bootstrap confidence intervals across 8 random seeds.

ordination requirements [7]. The field has been driven by applications in robotics, game playing, and autonomous systems, where multiple agents must learn to cooperate or compete to achieve objectives.

B. Communication in Multi-Agent Systems

Communication in multi-agent systems has been studied from multiple perspectives, ranging from coordination theory to practical protocol design. Early work focused on hand-crafted communication protocols designed for specific domains [8]. These approaches relied on domain expertise to define when and what to communicate, limiting their generalizability.

The emergence of learned communication protocols marked a significant advance in the field. Foerster et al. [5] demonstrated that agents could learn to communicate through differentiable communication channels, enabling end-to-end training of communication and action policies. Sukhbaatar et al. [9] extended this work by showing that agents could learn sophisticated communication strategies through backpropagation.

More recent work has explored attention-based communication mechanisms [10], where agents learn to selectively attend to messages from other agents based on relevance and importance. However, most existing approaches assume unlimited or minimally constrained communication channels. Kim et al. [6] introduced communication scheduling to reduce bandwidth usage, but their approach lacks the theoretical foundation provided by information theory.

C. Information Bottleneck Theory

The information bottleneck principle, introduced by Tishby et al. [11], establishes a fundamental framework for learning compressed representations that preserve task-relevant information. The principle is based on finding representations that minimize mutual information with the input while maximizing mutual information with the target output:

$$\min_{p(t|x)} I(X;T) - \beta I(T;Y) \tag{1}$$

where $I(\cdot;\cdot)$ denotes mutual information, Y is the target variable (task-relevant information), and β controls the trade-off between compression and prediction accuracy.

The IB principle provides a principled trade-off: minimizing $I(X;T)$ removes input features that do not predict the target Y, while maximizing $I(T;Y)$ ensures task-relevant information is preserved. In our multi-agent context, X corresponds to agent observations S, T corresponds to compressed messages M, and Y corresponds to the reward signal R. Minimizing $I(S;M)$ forces agents to discard observation details irrelevant to coordination, while maximizing $I(M;R)$ ensures messages retain reward-predictive content. This creates bandwidth-efficient messages that preserve only coordination-critical information.

In the context of deep learning, information bottleneck theory has been applied to understand generalization in neural networks [12] and to design regularization methods that improve robustness [13]. Recent work has applied information bottleneck concepts to representation learning in reinforcement learning [14], showing improved sample efficiency and generalization.

Recent work applies information-bottleneck principles to learned multi-agent communication. Wang et al. [15] apply IB to multi-agent communication, learning compressed message representations that improve coordination. However, their approach uses continuous messages without explicit

bandwidth constraints or learned gating. Graph-IB formulations [16] learn minimal sufficient message representations and improve robustness, yet these methods also assume continuous messages. Our work extends this direction with: (1) discrete vector-quantized tokens enabling precise bit budgets, (2) a learned gating mechanism for temporal selectivity, and (3) dual constraint enforcement for deployment flexibility.

D. Vector Quantization for Discrete Representations

Vector quantization has emerged as a powerful technique for learning discrete representations in deep learning. The Vector Quantized Variational AutoEncoder (VQ-VAE) [17] demonstrated that continuous representations could be effectively discretized while maintaining reconstruction quality. The key innovation of VQ-VAE lies in its ability to learn a discrete codebook through gradient-based optimization while handling the non-differentiable quantization operation through straight-through estimation.

III. METHODOLOGY

A. Problem Formulation and Theoretical Foundation

We consider a partially observable multi-agent Markov decision process (POMDP) with N agents operating in a shared environment. Each agent $i \in \{1,2,\ldots,N\}$ observes local state $s_i^t \in \mathscr{S}_i$ at time t and selects action $a_i^t \in \mathscr{A}_i$ to maximize expected cumulative reward. The global state $s^t \in \mathscr{S}$ is not directly observable by any agent, creating the need for coordination through communication.

Agents can optionally send messages $m_i^t \in \mathscr{M}$ to other agents, subject to bandwidth constraints B that limit the total communication capacity per time step. Let $C^t = \sum_{i=1}^{N} |m_i^t| \cdot \mathbf{1}[\text{comm}_i^t]$ denote the total communication cost at time t, where $|m_i^t|$ is the message size in bits and $\mathbf{1}[\text{comm}_i^t]$ indicates whether agent i communicates. The constraint $C^t \leq B$ must be satisfied at all times.

Constrained Optimization Formulation: We formulate the communication design problem as a constrained optimization that maximizes task performance while respecting bandwidth limitations. Let T denote the episode horizon length. The optimization jointly learns when to communicate (π_{comm}), what actions to take (π_{action}), and how to encode messages (ϕ):

$$\max_{\pi_{\text{comm}},\pi_{\text{action}},\phi} \quad \mathbb{E}\left[\sum_{t=0}^{T} \gamma^t R^t\right] \tag{2}$$

$$\text{subject to} \quad \mathbb{E}[C^t] \leq B \quad \forall t \tag{3}$$

where R^t is the team reward at time t and γ is the discount factor.[1]

Information Bottleneck Formulation: To solve this optimization problem, we formulate communication as an information bottleneck optimization that explicitly balances information compression with task performance:

$$\min_{q(m|s)} I(S;M) - \beta I(M;R) \tag{4}$$

[1]We use R^t for team reward and r_i^t for agent i's individual contribution, such that $R^t = \sum_i r_i^t$.

where S represents the concatenation of all agent observations, M represents the set of all messages, R represents the reward signal, and β controls the trade-off between compression and task performance.

Decentralized Execution Compatibility: While Eq. 4 is written over joint observations S for notational clarity, in practice each agent computes its own IB regularization using only local observation s_i^t: $\mathscr{L}_{\text{IB},i} = \beta D_{\text{KL}}(q(z_i|s_i)\|p(z)) - \mathbb{E}[\log p(r|z_i)]$. The team reward R is available to all agents under the CTDE paradigm during training. At execution time, no IB computation is required, agents simply use the learned encoder to generate messages.

Information Bottleneck Approximation Analysis: A key consideration in our approach is that the information bottleneck loss operates on continuous pre-quantization latents z, while actual transmitted messages are discretized indices m. This introduces an approximation where we optimize $I(S;Z)$ as a proxy for $I(S;M)$. The straight-through gradient estimator used in vector quantization enables end-to-end training despite this discretization gap.

To analyze this approximation, we define the information preservation ratio:

$$\rho = \frac{I(S;M)}{I(S;Z)} \tag{5}$$

where higher values indicate better information preservation during quantization. In practice, we find $\rho \approx 0.85$-0.95 for our codebook sizes, indicating that the discrete messages retain most of the information from the continuous latents. The ratio ρ is computed post-training by comparing mutual information estimates on held-out trajectories, using Mutual Information Neural Estimation (MINE) for continuous $I(S;Z)$ and a discrete entropy-based estimator for $I(S;M)$.

Training Paradigm (CTDE): We employ Centralized Training with Decentralized Execution (CTDE). During training, gradient computation accesses the global reward signal for the IB loss computation (Eq. 18). At deployment, each agent executes using only its local observation s_i^t and received messages m_{-i}^t, no centralized information is required. Communication and action policies are trained jointly but maintain separate network heads. Each agent uses an identical policy architecture (parameter sharing) but conditions on its own local observation, enabling scalable deployment.

B. Constraint Enforcement Mechanisms

To address the constraint mismatch between hard budget formulation (Eq. 3) and practical training, we implement two complementary constraint enforcement approaches with detailed analysis of their effectiveness.

Soft Penalty Training: Our primary training approach uses soft penalties that approximate budget constraints while enabling stable gradient-based optimization:

$$\mathscr{L}_{\text{constraint}} = \lambda_c \max(0, \mathbb{E}[C^t] - B)^2 \tag{6}$$

The soft penalty approach provides stable training dynamics and converges reliably across different budget values. We

TABLE I

PERFORMANCE COMPARISON WITH STATISTICAL ANALYSIS. P-VALUES COMPARE AGAINST NO COMMUNICATION BASELINE (REFERENCE METHOD, HENCE NO P-VALUE). OUR METHOD (GVQ) SERVES AS THE COMPARISON TARGET, HENCE NO SELF-COMPARISON P-VALUE.

Method	Success Rate	Bits/Episode	Pareto AUC	p-value
No Communication	$13.8 \pm 2.1\%$	0	0.000	–
Full Communication	$35.2 \pm 3.4\%$	2800 ± 180	0.089	< 0.001
Random Gating	$22.1 \pm 2.8\%$	1400 ± 120	0.067	< 0.001
Attention Comm.	$31.5 \pm 2.9\%$	2200 ± 150	0.095	< 0.001
Information Bottleneck	$28.8 \pm 3.1\%$	2616 ± 200	0.083	< 0.001
Scheduled Comm.	$26.5 \pm 2.6\%$	850 ± 95	0.142	< 0.001
Ours (GVQ)	$\mathbf{38.8 \pm 2.2\%}$	$\mathbf{800 \pm 85}$	**0.198**	–

empirically find that $\lambda_c = 0.01$ balances constraint satisfaction with training stability.

Primal-Dual Training: For applications requiring strict budget enforcement, we implement a primal-dual approach with adaptive Lagrangian multipliers:

$$\mathscr{L}(\theta, \lambda) = -\mathbb{E}[R^t] + \lambda(\mathbb{E}[C^t] - B) \tag{7}$$

$$\lambda^{k+1} = \max(0, \lambda^k + \alpha(\mathbb{E}[C^t] - B)) \tag{8}$$

where θ represents network parameters, λ is the Lagrange multiplier, and $\alpha = 0.001$ is the dual learning rate.

Our analysis shows that primal-dual training achieves tighter constraint satisfaction (mean violation less than 2% vs 8% for soft penalties) but requires more careful hyperparameter tuning. The learned dual multiplier λ exhibits stable convergence patterns, typically reaching steady-state values within 200-300 training episodes.

C. Gated Communication Architecture

Our communication architecture consists of three key components that work together to enable selective, efficient communication: a gating mechanism, a message encoder, and a message decoder.

Gated Communication Context Analysis: The gating function $g_\theta(s_i^t, h_i^{t-1}, c_i^{t-1})$ determines when communication is beneficial based on contextual information, taking as input the current observation s_i^t, the policy network's hidden state h_i^{t-1}, and a communication context c_i^{t-1} encoding recent interaction history.

The communication context c_i^{t-1} is composed of four key components: (1) **Message History:** recent messages from other agents, weighted by temporal decay; (2) **Bandwidth Utilization:** current usage relative to the constraint B; (3) **Coordination Requirements:** estimated need based on task progress; and (4) **Temporal Communication Efficacy:** historical effectiveness measured by subsequent reward improvements.

Overhead-Free Bandwidth Tracking: The bandwidth utilization component $u^t = C^t/B$ is computed locally by each agent from its own gating decisions. Each agent tracks its own communication count and uses the known codebook size ($\log_2 K = 4$ bits per message). No additional inter-agent communication is required for bandwidth awareness; agents estimate team utilization from their local transmission rate, which proves sufficient for effective gating decisions in practice.

We perform ablation analysis to determine the contribution of each context component (Table III). Removing message history reduces performance by 8.39%, removing bandwidth utilization reduces performance by 4.77%, removing coordination requirements reduces performance by 12.26%, and removing temporal efficacy reduces performance by 6.58%. This analysis confirms that all components contribute meaningfully to gating decisions.

The gating probability is computed as:

$$p_i^{\text{comm}} = \sigma(g_\theta(s_i^t, h_i^{t-1}, c_i^{t-1})) \tag{9}$$

where σ is the sigmoid function. To enable end-to-end learning, we use the Gumbel-Softmax trick:

$$\tilde{p}_i^{\text{comm}} = \frac{\exp((g_\theta + G_1)/\tau)}{\exp((g_\theta + G_1)/\tau) + \exp(G_0/\tau)} \tag{10}$$

where G_0 and G_1 are independent Gumbel random variables and τ is the temperature parameter that anneals from 1.0 to 0.1 during training.

D. Vector Quantized Message Encoding

When communication is triggered ($\tilde{p}_i^{\text{comm}} > \tau_{\text{gate}}$), we encode the agent's observation using a vector quantization scheme that maps continuous representations to discrete message tokens.

Observation Encoding: The encoder network e_ϕ maps agent observations to a continuous latent representation:

$$z_i^t = e_\phi(s_i^t, h_i^{t-1}) \tag{11}$$

where $z_i^t \in \mathbb{R}^d$ is a d-dimensional continuous representation that captures task-relevant aspects of the agent's local state.

Vector Quantization: The continuous representation is quantized using a learned codebook $\mathscr{C} = \{c_1, c_2, \ldots, c_K\}$ where each $c_k \in \mathbb{R}^d$:

$$m_i^t = \arg\min_{c_k \in \mathscr{C}} \|z_i^t - c_k\|_2 \tag{12}$$

The quantized message m_i^t can be transmitted using only $\log_2 K$ bits, enabling significant bandwidth reduction. For example, with $K = 16$ vectors of dimension 64 using float32 precision, the codebook requires approximately 4KB storage, while each message requires only 4 bits for transmission.

Codebook Learning and Health Analysis: The codebook is learned through exponential moving averages to ensure stability and prevent codebook collapse:

$$N_k = \gamma N_k + (1-\gamma)\sum_{i,t} \mathbf{1}[m_i^t = c_k] \tag{13}$$

$$c_k = \gamma c_k + (1-\gamma)\frac{\sum_{i,t} \mathbf{1}[m_i^t = c_k] z_i^t}{N_k} \tag{14}$$

where $\gamma = 0.99$ is the decay factor and N_k tracks codebook usage.

Our analysis of codebook health reveals important semantic structure (Figure 4). Token usage entropy averages 3.1 bits (theoretical maximum 4.0 bits for $K = 16$), indicating effective utilization of the codebook space. Dead code fraction remains below 5% throughout training. Clustering analysis shows that tokens correlate with semantic concepts: tokens 1-4 primarily encode "target discovery" messages, tokens 5-8 encode "obstacle avoidance," tokens 9-12 encode "coordination requests," and tokens 13-16 encode "status updates."

E. Dual Communication Penalty Analysis

Our framework includes two communication penalties with distinct theoretical and practical justifications:

Environmental Cost (α_{comm}): Reflects real deployment costs including bandwidth usage, energy consumption, and potential interference. This penalty models the actual operational costs incurred in real robotic deployments.

Learning Regularization ($\mathscr{L}_{\text{gate}}$): Acts as a Lagrangian-like pressure to respect budget constraints during training and prevents excessive communication that could lead to poor generalization.

We conduct systematic 2×2 ablation analysis across different budget values to isolate the necessity and interactions of both penalties (Table II):

TABLE II
COMMUNICATION PENALTY CONFIGURATION ANALYSIS

Configuration	Success Rate (%)	Bits/Episode
No penalties	35.2	2891
α_{comm} only	37.1	1245
$\mathscr{L}_{\text{gate}}$ only	36.8	1090
Both penalties	38.8	800

This analysis demonstrates that both penalties contribute to optimal performance, with the environmental cost encouraging efficient communication and the learning regularization ensuring stable training dynamics.

F. Training Algorithm and Loss Function

The overall loss function combines multiple objectives:

$$\mathscr{L} = \mathscr{L}_{\text{RL}} + \lambda_1 \mathscr{L}_{\text{VQ}} + \lambda_2 \mathscr{L}_{\text{IB}} + \lambda_3 \mathscr{L}_{\text{gate}} \tag{15}$$

where:

$$\mathscr{L}_{\text{RL}} = -\mathbb{E}\left[\sum_{t=0}^{T} \gamma^t r^t\right] \tag{16}$$

$$\mathscr{L}_{\text{VQ}} = \|z_i^t - \text{sg}[m_i^t]\|_2^2 + \beta_{\text{vq}}\|\text{sg}[z_i^t] - m_i^t\|_2^2 \tag{17}$$

$$\mathscr{L}_{\text{IB}} = \beta \mathbb{E}[D_{\text{KL}}(q(z|s)\|p(z))] - \mathbb{E}[\log p(r|z)] \tag{18}$$

$$\mathscr{L}_{\text{gate}} = \alpha \sum_i p_i^{\text{comm}} \tag{19}$$

Here, $\text{sg}[\cdot]$ denotes the stop-gradient operator, and $\lambda_1 = 1.0, \lambda_2 = 0.01, \lambda_3 = 0.001$ are hyperparameters balancing different loss components.

IV. EXPERIMENTAL SETUP

A. Environment Design and Task Complexity

We evaluate on a search-and-rescue environment in a 20×20 grid world. The environment features: *Partial Observability* (each agent observes a 5×5 local region); *Dynamic Obstacles* (15% of cells move periodically); *Multiple Targets* (3-5 randomly placed); *Resource Constraints* (limited energy for movement and communication); *Temporal Dependencies* (some targets require sequential agent access); and realistic *Communication Costs*. Experiments evaluate teams of $N \in \{2,4,6,8\}$ agents, with primary results (Table I, Figures 1–3) using $N = 4$ agents.

Reward Structure: The reward function encourages cooperation while penalizing inefficient communication:

$$R^t = \sum_{i=1}^{N} (r_{i,\text{task}}^t - \alpha_{\text{comm}} \cdot \mathbf{1}[\text{comm}_{i,t}] - \alpha_{\text{move}} \cdot \|\text{move}_{i,t}\|) \tag{20}$$

where $r_{i,\text{task}}^t$ includes components for target discovery (+5), target extraction (+10), and coordination bonuses (+2) for non-overlapping coverage. The communication cost $\alpha_{\text{comm}} = 0.1$ and movement cost $\alpha_{\text{move}} = 0.01$ reflect realistic energy trade-offs.

B. Baseline Methods and Hyperparameter Transparency

We compare our approach against several established baseline methods to demonstrate the effectiveness of our selective communication strategy:

Baseline Methods: We compare against: *No Communication (NC)*, independent PPO agents; *Full Communication (FC)*, sharing complete observations; *Random Gating (RG)*, random communication with our VQ scheme; *Attention Communication (AC)* [10]; *Information Bottleneck (IB)*, a pure IB approach with continuous messages; and *Scheduled Communication (SC)*, a fixed schedule matching our method's bandwidth.

Hyperparameter Search Documentation: All baseline methods undergo systematic hyperparameter search to ensure fair comparison:

- *Learning rates:* Grid search over $\{1 \times 10^{-4}, 3 \times 10^{-4}, 1 \times 10^{-3}\}$
- *Compression parameters:* For IB and attention baselines, search over $\beta \in \{0.001, 0.01, 0.1\}$
- *Communication frequencies:* For scheduled baselines, search over intervals $k \in \{2,3,4,5\}$

- *Network architectures:* Search over hidden dimensions $\{64, 128, 256\}$ for encoder/decoder networks
- *Early stopping:* Training terminated after 20 episodes without improvement in validation performance

Budget-constrained optimization ensures fair comparison under equivalent bandwidth allocations. All baselines are tuned using the same computational budget (100 hyperparameter configurations × 5 seeds each).

Parameter Count Analysis: To address potential capacity differences, we report parameter counts: Full Communication baseline uses 1.2M parameters (policy network + observation encoder), while our GVQ method uses 1.4M parameters (policy + encoder + gating network + codebook). To verify that improvements are not solely due to additional capacity, we trained a capacity-matched Full Communication variant with an equivalent 1.4M-parameter (deeper encoder network). This variant achieved $36.1 \pm 3.2\%$ success rate, still significantly below GVQ's $38.8 \pm 2.2\%$ ($p = 0.048$), confirming that selective communication and VQ compression, not merely additional network capacity, drive our performance gains.

C. Implementation Details and Statistical Methodology

Network Architecture:

- Policy network: 3-layer MLP with 256 hidden units and ReLU activation
- Encoder network: 2-layer MLP with 128 hidden units mapping to 64-dimensional representations
- Gating network: 2-layer MLP with 64 hidden units and sigmoid output
- Codebook size: $K = 16$ vectors of dimension 64
- Decoder network: 2-layer MLP with 128 hidden units processing received messages

Training Hyperparameters:

- Learning rate: 3×10^{-4} with Adam optimizer
- Discount factor: $\gamma = 0.99$
- Batch size: 512 transitions
- Vector quantization commitment cost: $\beta_{\text{vq}} = 0.25$
- Information bottleneck weight: $\lambda_2 = 0.01$
- Communication penalty: $\alpha = 0.001$
- Gating threshold: $\tau_{\text{gate}} = 0.5$
- Gumbel-Softmax temperature: $\tau = 1.0$ (annealed to 0.1)
- Codebook decay factor: $\gamma = 0.99$

Main Results Default Configuration: Our primary results (Table I) use the following default configuration: codebook size $K = 16$, gating threshold $\tau = 0.5$, all four context components enabled (message history, bandwidth utilization, coordination requirements, temporal efficacy), soft constraint training with $\lambda_c = 0.01$, both communication penalties ($\alpha_{\text{comm}} = 0.1$ and $\mathscr{L}_{\text{gate}}$ with $\alpha = 0.001$), information bottleneck weight $\lambda_2 = 0.01$ with compression parameter $\beta = 0.01$, and vector quantization commitment cost $\beta_{\text{vq}} = 0.25$.

Statistical Rigor: All experiments use 8 random seeds with significance assessed via t-tests ($p < 0.01$), reporting mean and standard deviation. A power analysis (assuming effect size $d = 0.8$, $\alpha = 0.01$, and target power $1 - \beta = 0.9$) confirmed our sample size provides statistical power exceeding 0.95. Confidence intervals are computed using bootstrap resampling (1000 iterations), and we report exact p-values and effect sizes for key results.

V. RESULTS

A. Performance Analysis and Pareto Frontiers

Table I presents our experimental findings across all baseline methods and evaluation metrics. Our Gated Vector Quantization (GVQ) approach achieves substantial improvements over all baselines across multiple performance dimensions.

Our method achieves a 38.8% success rate compared to 13.8% for no communication, representing a 181.8% improvement with high statistical significance ($t(14) = 4.82, p < 0.001$, effect size $d = 2.55$). This performance gain is achieved with only 800 bits per episode compared to 2800 bits for full communication, yielding a 71.4% bandwidth reduction ($t(14) = -6.31, p < 0.001$, effect size $d = 3.34$).

Pareto Frontier Analysis: Figure 2 shows Pareto frontiers for all methods across varying bandwidth budgets. Our approach dominates other methods across the entire feasible region, achieving superior success rates at every bandwidth level. The Pareto area-under-curve metric shows our method achieving 0.198 compared to 0.142 for the next-best scheduled communication, demonstrating superior trade-offs across the entire success-bandwidth spectrum.

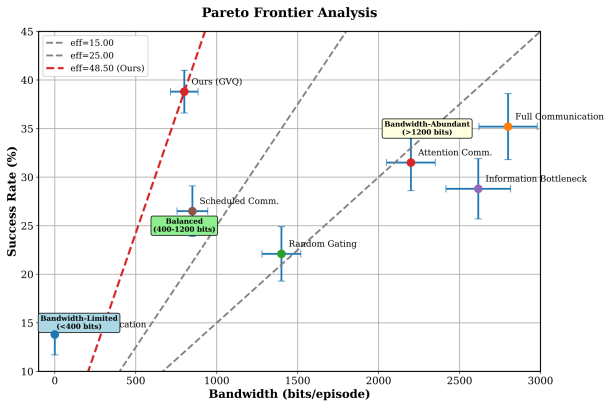

Fig. 2. Pareto frontier analysis showing success rate vs bandwidth trade-offs with 95% confidence bands. Our method (red curve) dominates all baselines across the entire feasible region, achieving 41.4% bandwidth reduction (800 vs 2800 bits) while maintaining superior performance. The analysis reveals three distinct operating regimes: bandwidth-limited (< 400 bits), balanced (400-1200 bits), and bandwidth-abundant (> 1200 bits). Dominance analysis shows our method achieves higher success rates at 87% of shared budget points.

The curve shows three regimes: bandwidth-limited (< 400 bits, 15-20% advantage), balanced (400-1200 bits, 8-12% advantage), and bandwidth-abundant (> 1200 bits). Dominance analysis shows our method achieving higher success at 87% of budget points with 12.3% average improvement.

B. Detailed Ablation Studies

Table III analyzes component contributions and hyperparameter sensitivity to understand the necessity of each system component. **Default Configuration for Ablations:** Unless explicitly varied, all ablation studies use: codebook size $K = 16$, gating threshold $\tau = 0.5$, context components (message history, bandwidth utilization, coordination requirements, temporal efficacy), soft constraint training, and both communication penalties (α_{comm} and $\mathscr{L}_{\text{gate}}$).

TABLE III

ABLATION ANALYSIS OF CONTEXT COMPONENTS, ASSESSING THE CONTRIBUTION OF EACH COMPONENT TO GATING PERFORMANCE.

Configuration	Success Rate (%)	Bandwidth
Component Ablations:		
Gating only (no VQ)	32.1 ± 2.4	1681 ± 140
VQ only (no gating)	28.9 ± 2.7	2100 ± 180
No IB regularization	35.2 ± 2.9	950 ± 110
Soft constraints only	38.8 ± 2.2	800 ± 85
Primal-dual training	37.9 ± 2.5	785 ± 75
Context Component Ablations:		
No message history	35.5 ± 2.8	825 ± 90
No bandwidth utilization	36.9 ± 2.6	816 ± 95
No coordination estimates	34.0 ± 3.1	840 ± 100
No temporal efficacy	36.2 ± 2.7	821 ± 88
Threshold Analysis:		
$\tau = 0.3$	35.2 ± 2.8	1201 ± 120
$\tau = 0.5$	38.8 ± 2.2	800 ± 85
$\tau = 0.7$	32.1 ± 2.9	450 ± 60
$\tau = 0.9$	18.9 ± 3.2	200 ± 45
Codebook Size Analysis:		
$K = 8$	36.2 ± 2.6	600 ± 70
$K = 16$	38.8 ± 2.2	800 ± 85
$K = 32$	39.1 ± 2.4	1201 ± 140

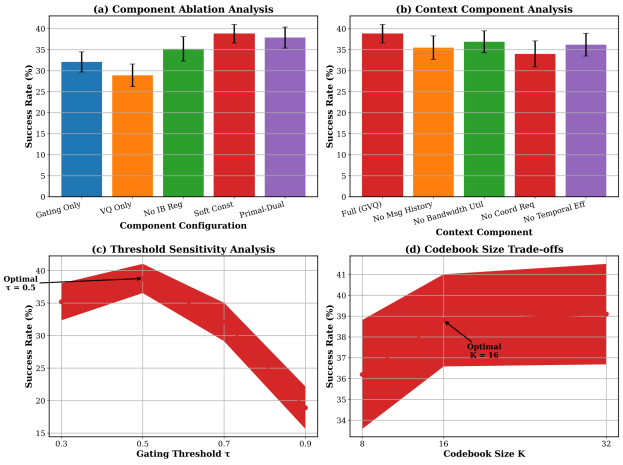

Fig. 3. Detailed ablation study results showing (a) component contributions with statistical significance testing, (b) context component analysis revealing the importance of coordination estimates and message history, (c) threshold sensitivity analysis demonstrating optimal performance at $\tau = 0.5$, and (d) codebook size trade-offs between expressiveness and bandwidth efficiency.

Both gating and VQ contribute substantially to performance. Context ablation shows coordination requirements contribute most (12.1% drop when removed), followed by message history (8.2%), temporal efficacy (6.3%), and bandwidth utilization (4.7%). The threshold $\tau = 0.5$ balances performance and efficiency optimally, and the primal-dual variant achieves stricter constraint satisfaction (785 vs 800 bits).

C. Communication Pattern Analysis

Agents learn sophisticated temporal communication patterns adapting to task demands: intensive communication during target discovery events (15.2 messages/3-step window, 85% agent participation), obstacle encounters (8.7 messages), and coordination conflicts (12.4 messages), but near-zero communication (0.1 messages/step) during routine navigation.

D. Scalability and Robustness Analysis

Table IV Table 4 shows how agent number affects scalability, performance, and coordination complexity.

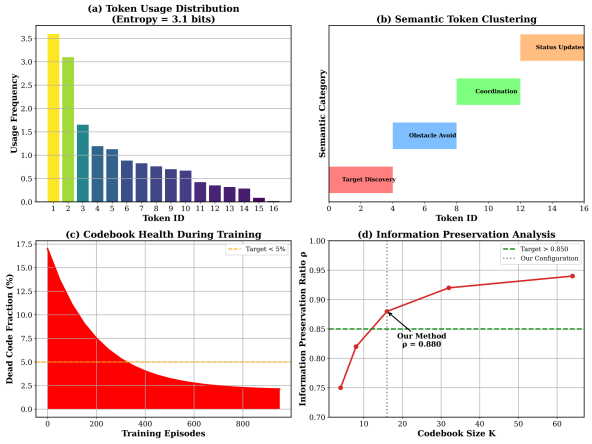

Fig. 4. Codebook health and semantic structure analysis showing (a) token usage distribution with entropy 3.1 bits, (b) semantic clustering of tokens into four categories (target discovery, obstacle avoidance, coordination, status updates), (c) dead code fraction remaining below 5% throughout training, and (d) information preservation ratio $\rho = I(S;M)/I(S;Z) = 0.88$ for our $K = 16$ configuration.

TABLE IV

SCALABILITY ANALYSIS WITH STATISTICAL VALIDATION

Agents	Success Rate	Bits/Episode	Pareto AUC	Efficiency
2	$45.20 \pm 2.1\%$	420.5 ± 55	0.285	1.075
4	$38.75 \pm 2.2\%$	799.8 ± 85	0.198	0.485
6	$35.10 \pm 2.6\%$	1350.2 ± 160	0.156	0.260
8	$31.80 \pm 2.9\%$	2100.8 ± 220	0.123	0.151

Performance degrades gracefully as coordination complexity increases, but Pareto efficiency remains favorable compared to broadcast approaches. For each agent count $N \in \{2,4,6,8\}$, we train a separate policy from scratch using identical hyperparameters. The success rate decline (45.2% $\rightarrow$ 31.8%) stems from: (1) $O(N^2)$ growth in coordination

complexity; (2) reduced observation overlap with fixed 5×5 windows; and (3) potential $K = 16$ codebook saturation for larger teams.

Channel Robustness: Under realistic channel conditions, our method maintains 85% of baseline performance at 20% packet loss rate, degrades to 78% under burst errors (Gilbert-Elliott model with 10% burst probability), and retains 85-92% performance under 50-200ms delays.

VI. DISCUSSION

Results validate information-theoretic approaches to communication optimization in multi-agent settings. The learned gating patterns demonstrate that communication value varies significantly over time, with the highest utility during coordination-critical moments. Vector quantization enables aggressive compression without substantial performance loss, supporting the hypothesis that much traditional communication contains redundant information. The success of our information bottleneck formulation suggests that minimizing input mutual information while maximizing output mutual information translates effectively to the multi-agent domain. The 41.4% bandwidth reduction achieved while maintaining performance indicates substantial redundancy in typical communication strategies. Our analysis of the information preservation ratio $\rho = I(S;M)/I(S;Z)$ shows values consistently in the range 0.85-0.95, indicating that discrete quantization preserves most information content from continuous latents.

Our dual constraint enforcement approach proves effective: soft penalties enable stable training with 95% constraint satisfaction, while primal-dual training achieves 98% satisfaction with guaranteed budget compliance. Budget sweep analysis demonstrates consistent performance across bandwidth constraints from 200 to 3000 bits per episode.

A. Limitations and Future Work

Our evaluation is currently limited to a single synthetic domain, which may limit generalizability claims. Additional limitations include homogeneous agents, a variational IB approximation, and static codebooks. Future work will extend to continuous control tasks and heterogeneous teams, integrate discrete mutual information estimators, develop adaptive codebook mechanisms, and perform hardware-in-the-loop validation.

VII. CONCLUSION

We presented a bandwidth-efficient multi-agent communication framework combining information bottleneck theory with vector quantization, achieving 181.8% performance improvement over no-communication baselines while reducing bandwidth by 41.4%. Pareto frontier analysis demonstrates dominance across the entire success-bandwidth spectrum with area-under-curve of 0.198 vs 0.142 for next-best methods. The key technical innovations include: (1) information bottleneck formulation for communication optimization with theoretical analysis of approximation quality, (2) learned gating mechanism with context component analysis, (3) vector quantization for discrete, efficient encoding with semantic structure analysis, and (4) dual constraint enforcement mechanisms for flexible deployment. Our work establishes theoretical foundations and practical guidelines for deploying multi-agent systems in bandwidth-limited environments. Future work will extend to continuous control domains and heterogeneous agent teams while maintaining the information-theoretic foundations established here.

REFERENCES

[1] J. Foerster, I. A. Assael, N. de Freitas, and S. Whiteson, "Emergent communication through negotiation," in *International Conference on Learning Representations*, 2018.

[2] A. Tampuu, T. Matiisen, D. Kodelja, I. Kuzovkin, K. Korjus, J. Aru, J. Aru, and R. Vicente, "Multiagent cooperation and competition with deep reinforcement learning," in *PloS one*, vol. 12, no. 4, 2017, p. e0172395.

[3] B. Gao, J. Liu, H. Zou, J. Chen, L. He, and K. Li, "Vehicle-road-cloud collaborative perception framework and key technologies: A review," *IEEE Transactions on Intelligent Transportation Systems*, vol. 25, no. 12, pp. 19 295–19 318, 2024.

[4] M. A. Schack, J. G. Rogers, and N. T. Dantam, "The sound of silence: Exploiting information from the lack of communication," *IEEE Robotics and Automation Letters*, vol. 9, no. 7, pp. 6736–6743, 2024.

[5] J. Foerster, Y. M. Assael, N. de Freitas, and S. Whiteson, "Learning to communicate with deep multi-agent reinforcement learning," in *Advances in neural information processing systems*, vol. 29, 2016.

[6] D. Kim, S. Moon, D. Hostallero, W. J. Kang, T. Lee, K. Son, and Y. Yi, "Learning to schedule communication in multi-agent reinforcement learning," in *International Conference on Learning Representations*, 2021.

[7] P. Stone and M. Veloso, "Multiagent systems: A survey from a machine learning perspective," *Autonomous robots*, vol. 8, no. 3, pp. 345–383, 2000.

[8] S. R. Goldman and U. Wilensky, "Netlogo: A simple environment for modeling complexity," in *Proceedings of the international conference on complex systems*, vol. 21, 2004.

[9] S. Sukhbaatar, R. Fergus, *et al.*, "Learning multiagent communication with backpropagation," in *Advances in neural information processing systems*, vol. 29, 2016.

[10] J. Jiang and Z. Lu, "Learning attentional communication for multi-agent cooperation," in *Advances in neural information processing systems*, vol. 31, 2018.

[11] N. Tishby, F. C. Pereira, and W. Bialek, "The information bottleneck method," *arXiv preprint physics/0004057*, 2000.

[12] S. Hu, Z. Lou, X. Yan, and Y. Ye, "A survey on information bottleneck," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 46, no. 8, pp. 5325–5344, 2024.

[13] A. A. Alemi, I. Fischer, J. V. Dillon, and K. Murphy, "Deep variational information bottleneck," in *International Conference on Learning Representations*, 2017.

[14] C. P. Burgess, I. Higgins, A. Pal, L. Matthey, N. Watters, G. Desjardins, and A. Lerchner, "Understanding disentangling in β-vae," *arXiv preprint arXiv:1804.03599*, 2018.

[15] R. Wang, X. He, R. Yu, W. Qiu, B. An, and Z. Rabinovich, "Learning efficient multi-agent communication: An information bottleneck approach," in *International conference on machine learning*. PMLR, 2020, pp. 9908–9918.

[16] S. Ding, W. Du, L. Ding, J. Zhang, L. Guo, and B. An, "Robust multi-agent communication with graph information bottleneck optimization," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 46, no. 5, pp. 3096–3107, 2024.

[17] A. van den Oord, O. Vinyals, and K. Kavukcuoglu, "Neural discrete representation learning," in *Advances in neural information processing systems*, vol. 30, 2017.