CLIC: CLOSED-LOOP INFORMATION CONTROL FOR EFFICIENT AND ROBUST MULTI-AGENT SYSTEMS

Anonymous authors
Paper under double-blind review

ABSTRACT

Multi-agent language model systems rely on communication to combine complementary evidence, but unrestricted message exchange increases token costs, repeats stale information, and allows faulty agents to spread misleading claims. The central challenge is to determine when agents should communicate and how strongly received messages should influence subsequent decisions while preserving independent reasoning. Existing communication-pruning, learned-topology, and robust-aggregation methods address parts of this challenge without jointly controlling information freshness, communication cost, and adversarial influence. We introduce **CLIC** (**C**losed-**L**oop **I**nformation **C**ontrol), a training-free controller that regulates which messages enter each agent's context and how much context they receive. CLIC represents proposals as compact semantic sketches and uses Bayesian surprise to transmit informative messages, with an age-based trigger preventing prolonged silence. Receivers evaluate message usefulness and apply geometric-median fusion to limit the influence of outlying proposals. A spectral controller then balances collective agreement, preservation of independent evidence, and recovery speed, translating communication weights into context-token budgets. Under stated assumptions, we establish epochwise convergence of the external semantic controller, bounded message-sending silence, and a conditional bound on Byzantine influence. Across reasoning, code generation, collaborative planning, and adversarial benchmarks, CLIC improves task performance while reducing total token usage by 39.9% relative to full-mesh communication in the primary five-agent setting. With 30% Byzantine agents, it achieves 63.7% robust accuracy versus 58.9% for the strongest baseline, SAC, and reduces PEAR attack success from 16.8% to 11.4%, without updating the underlying models.

1 INTRODUCTION

Large language models (LLMs) are increasingly deployed as collectives of specialized agents that divide reasoning across roles and share intermediate findings Wang et al. (2024a); Guo et al. (2024). Communication, however, is the dominant cost of such collaboration. As teams grow, unrestricted exchange saturates contexts with redundant or stale content, inflates tokens and latency, and destabilizes coordination Zhu et al. (2024); a single Byzantine agent can further propagate confident misinformation across the collective. Semantic communication offers a partial remedy by transmitting task-relevant meaning rather than raw content Guo et al. (2025), but leaves open the three decisions that govern collaboration at inference time: *which* messages carry new information, *whom* a receiver should trust, and *how much* influence an accepted message should exert.

Existing approaches address complementary aspects of this problem. Orchestration and scheduling frameworks organize message exchange through broadcasting, routing, and pruning strategies Wang et al. (2024a); Guo et al. (2024), and AgentDropout optimizes the interaction topology offline to reduce volume Wang et al. (2025). Robustness is pursued separately: deterministic Byzantine consensus guarantees agreement at worst-case communication growing at least quadratically in the number of participants Civit et al. (2024); event-triggered consensus relaxes this to approximate agreement under Byzantine faults and bounded delays Yuan & Ishii (2023); and SAC characterizes graph-robustness conditions for decentralized filtering and information propagation in multi-agent LLM systems Lee et al. (2026). Each line secures one property, whether efficiency, agreement, or resilience, leaving the rest to be handled independently.

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.

We introduce **CLIC**, a training-free inference controller for decentralized collectives of heterogeneous foundation-model agents. CLIC casts inter-agent communication as closed-loop control over a single actuation channel governing both message admission and message influence within each agent's context. Proposals are encoded as compact semantic sketches; transmission is triggered by Bayesian surprise weighed against token cost and message age; accepted sketches are fused around a Byzantine-robust geometric median; and epoch-specific spectral coupling is selected analytically to balance agreement, diversity, and recovery speed. The resulting graph weights translate directly into context-token budgets, leaving every underlying model frozen.

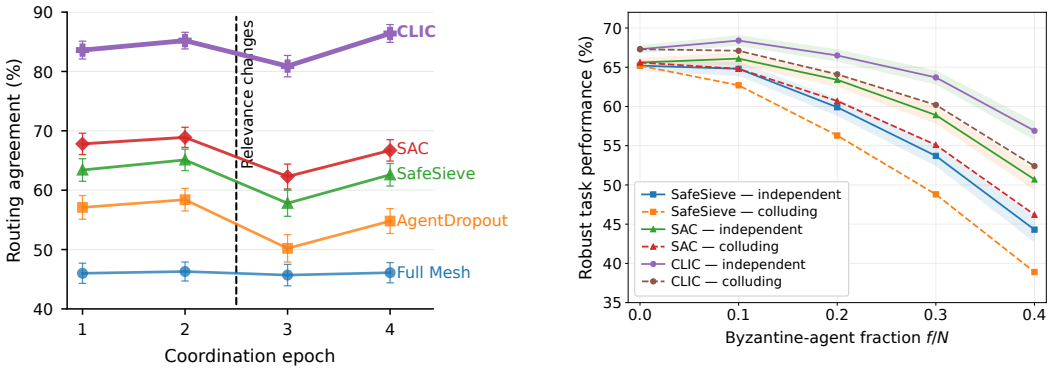
Our Contributions. (1) Communication as closed-loop inference control. We reformulate inter-agent communication as a control problem with a single actuation channel: the context-token budget each agent allocates to each neighbor. This unifies whether to send, whom to trust, and how much influence to grant under one closed loop, without labeled data, fine-tuning, or access to model internals. Admission is governed by *realized* Bayesian surprise measured at the receiver rather than by sender-side heuristics or a precomputed topology, so routing tracks each receiver's evolving semantic state within a single epoch. **(2) Byzantine-robust graph construction with analytically selected coupling.** Bidirectional usefulness scoring admits an edge only when both endpoints judge a message fresh and informative, and clipped geometric-median fusion bounds adversarial displacement *before* it propagates through the graph. The social coupling is then obtained in closed form as the minimum value jointly satisfying coherence, diversity, and recovery constraints, replacing the tuned constants used by existing controllers and yielding an explicit safe fallback when none is feasible. We establish epochwise geometric convergence, bounded silence, and a conditional Byzantine-influence bound, each scoped to the external controller rather than the factual correctness of the frozen models. **(3) Empirical validation across efficiency, quality, and robustness.** Across reasoning, coding, collaborative, and adversarial benchmarks, CLIC attains the best clean-task scores while using 39.9% fewer tokens than Full Mesh, showing that the savings come from better message selection rather than from suppressing communication. Under a 30% Byzantine fraction it gains 4.8 robust-accuracy points over the strongest baseline, and predicted and empirical contraction factors agree at runtime ($r = 0.91$).

2 Motivating Pilot: Receiver-Conditional Routing

Existing communication methods may reduce message volume without assessing a message's usefulness to a particular receiver at the current coordination epoch. Full Mesh distributes all messages indiscriminately; AgentDropout Wang et al. (2025) selects a task-level topology without reconstructing it from each receiver's evolving semantic state; SafeSieve Zhang et al. (2026) filters communication without explicitly evaluating usefulness relative to the receiver's prior; and SAC Lee et al. (2026) promotes stable agreement without directly translating pairwise relevance into context-token allocations. CLIC combines receiver-side Bayesian surprise, mutual usefulness scoring, epoch-specific graph construction, and adaptive context allocation.

To isolate this distinction, we adapt 40 logic-grid puzzles from ZebraLogic Lin et al. (2025) into five-agent constraint-reconciliation tasks. Puzzle clues are distributed as partially overlapping private information, and agents receive local questions with a known oracle sender–receiver relevance graph. Epochs 1–2 retain the initial questions. At epoch 3, one receiver's question changes, making one incoming edge irrelevant and another relevant while the puzzle, private clues, and candidate messages remain fixed. Epoch 4 retains the revised question to measure recovery. These information partitions and relevance interventions are specific to our adaptation.

With identical candidate messages and context limits across methods, we measure agreement between context allocations and the oracle relevance graph. As shown in Fig. 1A, CLIC achieves the highest receiver-specific routing agreement, experiences a smaller immediate decline than AgentDropout, SafeSieve, and SAC, and exceeds its pre-shift agreement by epoch 4. Full Mesh exhibits a smaller decline but consistently lower agreement, consistent with its indiscriminate allocation of communication capacity. This controlled pilot supports CLIC's closed-loop routing mechanism; experimental details appear in Appendix B.



(A) (B)

Figure 1: **(A)** Receiver-specific routing agreement in the ZebraLogic-based pilot. Epochs 1–2 use the initial receiver questions, the dashed line marks the question change at epoch 3, and epoch 4 measures recovery. CLIC updates receiver-side usefulness and context allocation at every epoch. **(B)** Robust task performance across increasing Byzantine-agent fractions. CLIC achieves the highest robustness under both independent and colluding attacks. Shaded regions in both panels denote 95% confidence intervals.

3 Related Work and Background

LLM-based multi-agent research has advanced from role-conditioned dialogue in CAMEL Li et al. (2023) to conversational orchestration in AutoGen Wu et al. (2023), workflow-structured collaboration in MetaGPT Hong et al. (2024), dynamic agent groups in AgentVerse Chen et al. (2024b), and communication-driven software development in ChatDev Qian et al. (2024). These systems show that specialization and natural-language interaction improve collective problem solving, yet their communication follows predefined workflows, dialogue schedules, or application-specific protocols rather than the real-time semantic value of each message. Recent methods instead optimize agent organization directly: GPTSwarm treats systems as optimizable computational graphs Zhuge et al. (2024), AgentDropout prunes redundant agents and edges to cut tokens Wang et al. (2025), and Optima learns efficient communication via supervised and preference-based training Chen et al. (2025). Yet all rely on offline optimization, learned policies, or structural pruning, and none jointly provides receiver-specific semantic filtering, bounded silence, Byzantine resilience, and closed-loop stability at inference time.

CLIC draws on several traditions that address these limitations in isolation. Semantic communication preserves task-relevant meaning over raw representations Xie et al. (2021); event-triggered and networked consensus selectively activate communication while spectral properties govern convergence Olfati-Saber et al. (2007); Boyd et al. (2004); and Byzantine-robust distributed learning, exposing the fragility of linear aggregation, motivates robust estimators such as Krum, coordinate-wise medians, and trimmed means Blanchard et al. (2017); Yin et al. (2018). Building on these, CLIC unifies frozen semantic encoding and compression, Bayesian surprise-based transmission, age-aware silence control, receiver-evaluated usefulness, clipped geometric-median fusion, and spectral context allocation in one training-free controller. Its novelty is treating inter-agent communication as a closed-loop information-control problem: messages are admitted by their realized semantic contribution, adversarial influence is bounded before graph propagation, and the minimum feasible coupling is chosen analytically to balance coherence, diversity, and recovery—without modifying the underlying foundation models.

4 Our Methodology

We propose **CLIC**, a training-free closed-loop inference controller for decentralized collectives of heterogeneous model agents. Rather than modifying the underlying LLMs, VLMs, semantic encoder, or decoding process, CLIC regulates collaboration by controlling which inter-agent messages enter each agent's context and how strongly they influence subsequent inference. At each coordi-

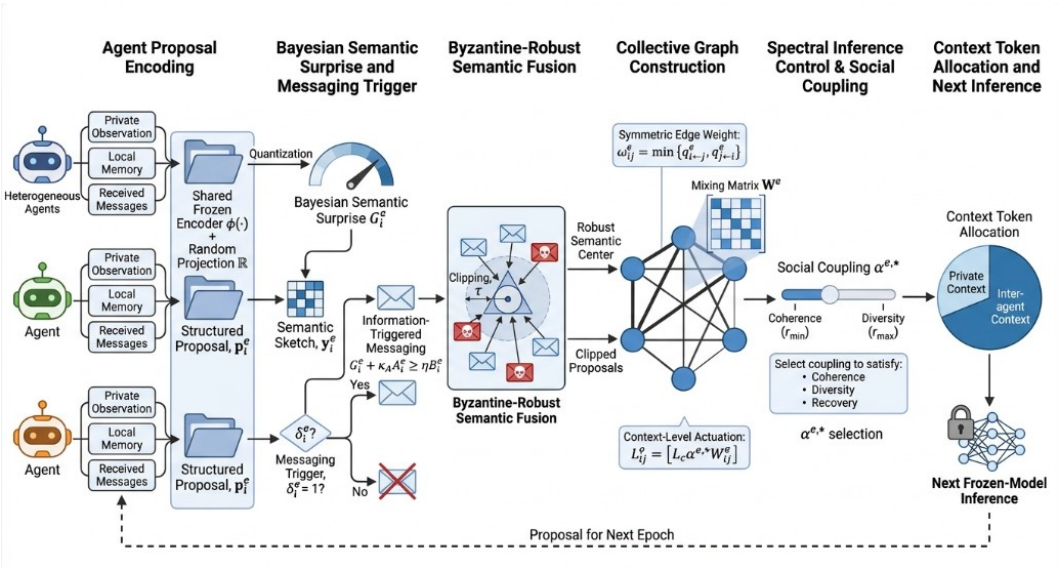


Figure 2: Overview of CLIC. Agent proposals are encoded into semantic sketches, selectively transmitted using Bayesian surprise, robustly fused, organized into a receiver-evaluated graph, and converted into spectral coupling and adaptive context-token allocations for the next frozen-model inference step.

nation epoch, CLIC (i) converts agent proposals into compact semantic sketches, (ii) suppresses redundant communication using Bayesian semantic surprise, (iii) performs Byzantine-robust semantic fusion, (iv) constructs a weighted collective graph, and (v) analytically selects the minimum social coupling satisfying coherence, diversity, and recovery constraints. The resulting graph weights are then translated into context-token allocations for the next frozen-model inference step. An overview of CLIC is shown in Fig. 2, and the complete procedure is provided in Algorithm 1 in Appendix F.

Problem setting and assumptions. Consider N agents indexed by $\mathcal{V} = \{1, \dots, N\}$ connected through a decentralized graph. Agent i is implemented by a frozen foundation model $\mathcal{M}_i$ with a fixed role specification r_i. At epoch e, it receives a private observation o_i^e, local memory m_i^e, and accepted messages from neighboring agents, producing a structured proposal $p_i^e = (c_i^e, q_i^e, a_i^e, r_i, e, i)$, where c_i^e, q_i^e, and a_i^e denote its current claim, supporting evidence, and proposed action, respectively. Sender identities are authenticated, although Byzantine agents may generate arbitrary or inconsistent proposals. Reliable message delivery is assumed, while communication decisions are governed by the proposed transmission trigger. During each coordination epoch, agent proposals and the collective graph remain fixed while the controller performs K semantic mixing steps using only semantic vectors, without additional model inference. Consequently, all theoretical guarantees apply to the external semantic controller rather than the internal reasoning dynamics or factual correctness of the frozen foundation models.

Training-free semantic representation. A shared frozen encoder $\phi(\cdot)$ maps each structured proposal to the normalized semantic vector $h_i^e = \phi(r_i \| c_i^e \| q_i^e \| a_i^e) / \|\phi(r_i \| c_i^e \| q_i^e \| a_i^e)\|_2$. The encoder is used without task-specific training or adaptation. To reduce storage and controller overhead, all agents share a Gaussian random projection $R \in \mathbb{R}^{p \times d}$, generated once from a public seed with $R_{kl} \sim \mathcal{N}(0, 1/p)$. For a finite set of M semantic vectors, selecting $p = O(\varepsilon_{\mathrm{JL}}^{-2} \log(M/\delta))$ preserves pairwise Euclidean distances within multiplicative distortion $\varepsilon_{\mathrm{JL}}$ with probability at least $1 - \delta$. The projected sketch $s_i^e = R h_i^e$ is clipped and uniformly quantized:

$$\widetilde{s}_i^e = \mathrm{clip}(s_i^e, -A, A), \qquad y_i^e = Q_b(\widetilde{s}_i^e), \qquad \Delta_q = \frac{2A}{2^b - 1}. \tag{1}$$

The resulting representation error satisfies $\|y_i^e - s_i^e\|_2 \leq \sqrt{p}\Delta_q/2 + \|s_i^e - \widetilde{s}_i^e\|_2$, where the first term represents quantization error and the second captures clipping distortion, which vanishes when all projected coordinates remain within $[-A, A]$.

Bayesian semantic-surprise estimation. Transmitting every proposal introduces redundant context and can lead to semantic over-smoothing. CLIC therefore estimates the information contributed by each proposal before deciding whether to communicate it. Agent i maintains an isotropic Gaussian surrogate $\theta^e \sim \mathcal{N}(\mu_i^{e,-}, v_i^{e,-} I_p)$ for the shared semantic state. Before epoch e, the prior variance evolves as $v_i^{e,-} = v_i^{e-1,+} + q_0$, where $q_0 > 0$ models semantic drift. The new semantic sketch is observed as $y_i^e = \theta^e + \nu_i^e$, with $\nu_i^e \sim \mathcal{N}(0, \sigma_{i,e}^2 I_p)$. The observation variance $\sigma_{i,e}^2$ is estimated from a clipped running median of recent semantic innovations and requires no learned parameters. The corresponding scalar Kalman gain is $k_i^e = v_i^{e,-}/(v_i^{e,-} + \sigma_{i,e}^2)$, yielding posterior statistics $\mu_i^{e,+} = \mu_i^{e,-} + k_i^e(y_i^e - \mu_i^{e,-})$ and $v_i^{e,+} = (1-k_i^e)^2 v_i^{e,-} + (k_i^e)^2 \sigma_{i,e}^2$. The realized posterior information gain is

$$G_i^e = \frac{1}{2}\left[p\frac{v_i^{e,+}}{v_i^{e,-}} + \frac{\|\mu_i^{e,+} - \mu_i^{e,-}\|_2^2}{v_i^{e,-}} - p + p\log\frac{v_i^{e,-}}{v_i^{e,+}}\right]. \tag{2}$$

Equation (2) measures realized Bayesian surprise. Under the Gaussian surrogate, its expectation equals the mutual information $I(\theta^e; y_i^e) = (p/2)\log(1 + v_i^{e,-}/\sigma_{i,e}^2)$.

Information-triggered messaging. Let A_i^e denote the number of epochs since agent i last transmitted and B_i^e the token length of its structured proposal. CLIC transmits a proposal according to

$$\delta_i^e = \mathbf{1}\left\{G_i^e + \kappa_A A_i^e \geq \eta B_i^e\right\}, \tag{3}$$

where $\delta_i^e = 1$ indicates transmission. Here, semantic surprise encourages informative messages, the age term prevents indefinite silence, and the token penalty ηB_i^e discourages unnecessarily long context additions. If $B_i^e \leq B_{\max}$, Equation (3) guarantees transmission within at most $A_{\max} = \lceil \eta B_{\max}/\kappa_A \rceil$ silent epochs, even when $G_i^e = 0$. This guarantee bounds message-sending delay but does not model packet loss or communication failures.

Byzantine-robust semantic fusion. Receivers do not rely on sender-reported confidence or importance. Instead, agent i assigns each received message a freshness weight $r_{ij}^e = \exp(-\kappa_f A_{ij}^e)$, normalizes it as $\bar{r}_{ij}^e = r_{ij}^e / \sum_\ell r_{i\ell}^e$, and computes a weighted geometric median. Each proposal is then clipped around this robust center:

$$\begin{aligned} g_i^e &= \arg\min_{z \in \mathbb{R}^p} \sum_{j \in \mathcal{N}_i^e \cup \{i\}} \bar{r}_{ij}^e \|y_j^e - z\|_2, \\ \bar{y}_{ij}^e &= g_i^e + \min\left\{1, \frac{\tau}{\max\{\|y_j^e - g_i^e\|_2, \epsilon\}}\right\}(y_j^e - g_i^e). \end{aligned} \tag{4}$$

Equation (4) ensures $\|\bar{y}_{ij}^e - g_i^e\|_2 \leq \tau$ for every accepted proposal, including arbitrary Byzantine inputs. For the robustness analysis, let β_i^e denote the normalized Byzantine weight in the geometric median and assume $\beta_i^e < 1/2$. If all honest sketches lie within radius R_i^e of an honest semantic center c_i^e, then $\|g_i^e - c_i^e\|_2 \leq D_i^e$, where $D_i^e = 2(1-\beta_i^e)R_i^e/(1-2\beta_i^e)$. When the geometric median is computed approximately with $\|\widehat{g}_i^e - g_i^e\|_2 \leq \varepsilon_g$, the bound becomes $D_i^e + \varepsilon_g$. The radius R_i^e is introduced solely for theoretical analysis and is never computed during inference. Consequently, the Byzantine guarantee is an *a priori* robustness bound under bounded honest semantic dispersion rather than an online certificate.

Receiver-evaluated message usefulness. Agent i applies the Bayesian update associated with Equation (2) to each clipped message $\bar{y}_{ij}^e$ using its own prior, producing the receiver-side surprise $G_{i\leftarrow j}^e$. The directed usefulness score is then $q_{i\leftarrow j}^e = r_{ij}^e \min\{G_{i\leftarrow j}^e, G_{\max}\}$, where the cap $G_{\max}$ prevents highly unusual but potentially malicious messages from receiving excessive influence. Endpoints exchange only these scalar usefulness scores and define the symmetric edge weight $\omega_{ij}^e = \min\{q_{i\leftarrow j}^e, q_{j\leftarrow i}^e\}$, so communication is strengthened only when both agents regard a message as fresh and informative.

Robust collective graph. Let $d_i^e = \sum_{j \neq i} \omega_{ij}^e$. For $i \neq j$ with $d_i^e, d_j^e > 0$, CLIC constructs the weighted Metropolis matrix with entries $\widetilde{W}_{ij}^e = \omega_{ij}^e / \max\{d_i^e, d_j^e\}$ and $\widetilde{W}_{ii}^e = 1 - \sum_{j \neq i} \widetilde{W}_{ij}^e$. For isolated agents ($d_i^e = 0$), $\widetilde{W}_{ii}^e = 1$ and $\widetilde{W}_{ij}^e = 0$ for $j \neq i$. The final mixing matrix is

$$W^e = \frac{I + \widetilde{W}^e}{2}. \tag{5}$$

The matrix in Equation (5) is symmetric and doubly stochastic. For connected epoch graphs, $1 = \lambda_1^e > \lambda_2^e \geq \cdots \geq \lambda_N^e \geq 0$. The lazy transformation shifts the spectrum to $[0, 1]$, eliminating negative eigenvalues and suppressing oscillatory disagreement modes.

Spectral inference control. Let $u_i^e \in \mathbb{R}^p$ denote agent i's private semantic proposal and $x_i^{e,k}$ its external context-control state after controller step k. The nonlinear geometric-median and clipping operations in Equation (4) are represented by a nominal mixing term together with a bounded residual. During epoch e, CLIC evolves the controller state as

$$x^{e,k+1} = (1 - \alpha^e)u^e + \alpha^e W^e x^{e,k} + \alpha^e \xi^{e,k}, \tag{6}$$

where $\xi^{e,k}$ captures quantization, clipping, median approximation, stale-context, and adversarial deviations. Equation (6) is the sole control law governing collective semantic evolution.

For fixed u, W, and $\alpha < 1$, the disturbance-free equilibrium is $x^\star = (1 - \alpha)(I - \alpha W)^{-1}u$. The retained fraction of private semantic variation in disagreement eigenmode $k \geq 2$ is $h_k(\alpha) = (1 - \alpha)/(1-\alpha\lambda_k)$. Since h_k decreases with α and increases with λ_k, all disagreement modes are bounded using only the extreme eigenvalues λ_2 and λ_N. Let $0 < r_{\min} < r_{\max} < 1$, where $r_{\min}$ preserves independent evidence and $r_{\max}$ limits unresolved disagreement. Enforcing $h_{\lambda_2}(\alpha) \leq r_{\max}$ and $h_{\lambda_N}(\alpha) \geq r_{\min}$ yields

$$\alpha_{\mathrm{C}} = \frac{1 - r_{\max}}{1 - r_{\max}\lambda_2}, \qquad \alpha_{\mathrm{D}} = \frac{1 - r_{\min}}{1 - r_{\min}\lambda_N}. \tag{7}$$

Here, α_{C} is the minimum coupling required for coherence, whereas α_{D} is the maximum coupling that preserves semantic diversity. The transient disagreement contracts at rate $\rho_\perp = \alpha\lambda_2$. Given the desired bound $\overline{\rho} < 1$, CLIC sets $\alpha_{\mathrm{R}} = 1 - \epsilon$ when $\lambda_2 = 0$, and $\alpha_{\mathrm{R}} = \min\{1 - \epsilon, \overline{\rho}/\lambda_2\}$ otherwise. The selected coupling is

$$\alpha^{e,\star} = \mathbf{1}\left\{\alpha_{\mathrm{C}}^e \leq \min(\alpha_{\mathrm{D}}^e, \alpha_{\mathrm{R}}^e)\right\} \alpha_{\mathrm{C}}^e. \tag{8}$$

When feasible, Equation (8) selects the smallest coupling satisfying the coherence, diversity, and recovery constraints, thereby achieving the fastest convergence to the diversity-preserving equilibrium $x^\star$.

If the constraints are infeasible, CLIC sets $\alpha^{e,\star} = 0$, preventing collective context from influencing the next inference step. A disconnected epoch graph ($\lambda_2 = 1$) automatically triggers this safe fallback. The eigenvalues λ_2 and λ_N are estimated using warm-started distributed power iterations with deflation against the consensus vector for λ_2 and iterations on $I - W$ for λ_N.

Context-level actuation. Let L_c denote the controller-managed context budget shared between private and inter-agent information. Agent i allocates neighbor j the token budget

$$L_{ij}^e = \left\lfloor L_c \alpha^{e,\star} W_{ij}^e \right\rfloor, \quad j \neq i, \qquad L_{ii}^e = L_c - \sum_{j \neq i} L_{ij}^e. \tag{9}$$

Since $\sum_{j \neq i} \alpha^{e,\star} W_{ij}^e \leq \alpha^{e,\star} < 1$, Equation (9) always reserves a positive context budget for the agent's private observation and local memory. Accepted messages are ranked by W_{ij}^e, truncated to their assigned budgets, and concatenated with the agent's role specification, private observation, and local memory. A single frozen-model inference call then produces the proposal for the next epoch.

Epochwise stability. Let $\Pi = I - \frac{1}{N}\mathbf{1}\mathbf{1}^\top$ denote the disagreement projector. For a fixed connected epoch without disturbance, Equation (6) satisfies

$$\left\|\Pi\left(x^{e,k} - x^{e,\star}\right)\right\|_2 \leq \left(\alpha^e \lambda_2^e\right)^k \left\|\Pi\left(x^{e,0} - x^{e,\star}\right)\right\|_2. \tag{10}$$

Hence, the external semantic state converges geometrically whenever $\alpha^e \lambda_2^e < 1$. The number of controller iterations required to reduce the equilibrium error below ε is bounded by

$$K_{\mathrm{rec}}(\varepsilon) \leq \left\lceil \frac{\log\left(\|\Pi(x^{e,0} - x^{e,\star})\|_2/\varepsilon\right)}{-\log(\alpha^e \lambda_2^e)} \right\rceil.$$

This guarantee is epochwise because the proposals, graph, and coupling are recomputed at every epoch. Accordingly, CLIC solves a sequence of stable local control problems, while long-horizon tracking and task stability under evolving graphs are evaluated empirically.

Conditional Byzantine influence. The robustness analysis distinguishes two Byzantine quantities. The fusion weight β_i^e governs whether the geometric median remains close to the honest semantic center, whereas the mixing mass $b_i^e = \sum_{j \in \mathcal{B}_i^e} W_{ij}^e$ determines how strongly residual Byzantine influence propagates through the collective graph. Define $L_{\mathcal{B}}^e = \max_i\{\tau + D_i^e + \varepsilon_g\}$. Relative to a reference execution that replaces each Byzantine proposal with its corresponding honest semantic center while preserving the same graph, the asymptotic disagreement satisfies

$$\limsup_{k\to\infty} \left\| \Pi\left(x^{e,k} - x_{\mathrm{ref}}^{e,k}\right) \right\|_2 \leq \frac{\alpha^e L_{\mathcal{B}}^e \|b^e\|_2}{1 - \alpha^e \lambda_2^e}. \tag{11}$$

Equation (11) assumes a normalized Byzantine fusion weight below one half and bounded honest semantic dispersion. It jointly captures robust-center displacement and propagation of residual Byzantine influence, bounding their effect on the external semantic controller without guaranteeing the factual correctness of generated content.

Complexity and training-free operation. For encoder dimension d, projection dimension p, local graph degree d_i, K_g geometric-median iterations, and K_s distributed spectral iterations, the per-agent controller complexity is $O(pd) + O(K_g p d_i) + O(K_s d_i)$. Under bounded graph degree and fixed iteration counts, this cost remains independent of the total number of agents. CLIC is fully training-free, requiring no labeled data, gradient computation, reinforcement learning, parameter updates, adapter training, activation modification, or model fine-tuning.

5 Experimental Evaluation

We evaluate whether CLIC improves communication efficiency, task performance, coordination, and robustness against Byzantine agents without model training or fine-tuning. Experiments cover five benchmark families: *MMLU-Pro* for knowledge-intensive reasoning with ten answer choices Wang et al. (2024b), *HumanEval* for execution-based code generation Chen et al. (2021), *MultiAgentBench* for collaborative reasoning Zhu et al. (2025), *Collab-Overcooked* for embodied cooperation Sun et al. (2025), and *PEAR* for utility and adversarial robustness in planner–executor systems Dong et al. (2026). We use the Research, Coding, and Bargaining tasks from MultiAgentBench and the three middle-complexity Collab-Overcooked levels; additional scenarios are reported in the appendix.

Agent configurations and baselines. The primary setting uses five frozen Qwen2.5-7B-Instruct agents Qwen Team et al. (2024). We additionally evaluate a heterogeneous team comprising Llama-3.1-8B-Instruct, Qwen2.5-14B-Instruct, Qwen2.5-32B-Instruct, Llama-3.1-70B-Instruct Grattafiori et al. (2024), and GPT-4o-mini. Agents are assigned fixed proposer, analyst, verifier, critic, and synthesizer roles. CLIC employs the frozen BGE-M3 semantic encoder Chen et al. (2024a) with the random projection and quantization of Eq. (1); no component is trained, fine-tuned, or task-adapted. We compare against *Single Agent*, *Majority Vote*, *Full Mesh*, *GPTSwarm* Zhuge et al. (2024), *AgentDropout* Wang et al. (2025), *SafeSieve* Zhang et al. (2026), and *Self-Anchored Consensus (SAC)* Lee et al. (2026). All methods use identical backbone models, agent roles, decoding settings, communication rounds, and context budgets. For optimization-based baselines, offline optimization cost is reported separately from inference.

Evaluation protocol. We consider three settings: *S1 (Clean)* without faulty agents, *S2 (Independent Byzantine)*, where a fraction $f/N \in \{0.1, 0.2, 0.3, 0.4\}$ of agents independently generate confident but incorrect, contradictory, or irrelevant messages, and *S3 (Colluding Byzantine)*, where corrupted agents coordinate their misinformation. PEAR additionally evaluates communication-level and system-prompt attacks. Sender identities remain authenticated, matching the assumptions of § 4; Byzantine agents may generate arbitrary content but cannot impersonate honest agents.

We report MMLU-Pro accuracy, HumanEval Pass@1, MultiAgentBench Task and Coordination Scores, Collab-Overcooked Success Rate and Progress Completeness, and PEAR Utility and Attack Success Rate (ASR). Communication efficiency is measured using prompt-plus-completion tokens, transmitted messages, communication rounds, latency, and token reduction relative to Full Mesh,

$$\mathrm{TokenRed.} = 1 - \frac{\mathrm{Tokens}_{\mathrm{method}}}{\mathrm{Tokens}_{\mathrm{FullMesh}}}. \tag{12}$$

Method	Robust Acc. at f/N 0.1	0.3	0.4	PEAR Utility / ASR (%)
Majority Vote	58.9	43.2	35.1	62.0 / 31.8
Full Mesh	60.5	39.4	27.8	65.3 / 39.7
GPTSwarm	62.1	46.8	35.7	67.0 / 28.9
AgentDropout	63.7	50.9	41.2	69.4 / 24.6
SafeSieve	64.8	53.7	44.3	71.2 / 21.7
SAC	66.1	58.9	50.7	74.6 / 16.8
CLIC	**68.4**	**63.7**	**56.9**	**78.3 / 11.4**

(A)

Variant	Task	Robust	Tok. Red. (%)	Max. Age
Periodic transmission	73.8	59.1	14.2	**1**
w/o age term	74.4	60.8	**41.3**	4
Mean fusion	74.7	54.9	40.5	4
Fixed $\alpha = 0.5$	73.9	59.7	39.0	4
Tuned constant α	74.8	61.9	39.4	4
CLIC (full)	**76.3**	**63.7**	39.9	4

(B)

Table 1: **(A)** Robustness under independent and colluding Byzantine agents. Robust Acc. is averaged over MMLU-Pro and reasoning-based MultiAgentBench tasks. Higher is better except for ASR. **(B)** Ablation study. Task denotes the mean normalized clean score, Robust is measured at $f/N = 0.3$, Tok. Red. is relative to Full Mesh, and Age is the maximum realized information age.

Table 2: Clean-task performance and communication efficiency under S1. MMLU-P denotes MMLU-Pro accuracy, HE denotes HumanEval Pass@1, MAB denotes MultiAgentBench, CO-PC denotes Collab-Overcooked Progress Completeness, and Tok. is the average total tokens per instance. Higher is better except for Tokens and Latency. Best results are shown in **bold**; second-best multi-agent results are underlined.

Method	MMLU-P Acc.	HE Pass@1	MAB Task	MAB Coord.	CO PC	Tok. ($\times 10^3$)	Token Red. (%)	Latency (s)
Single Agent	54.8	49.1	42.6	38.4	45.1	8.2	70.0	14.2
Majority Vote	59.7	55.1	48.9	43.2	51.6	18.6	31.9	25.5
Full Mesh	63.4	60.2	67.8	70.1	72.4	27.3	0.0	36.8
GPTSwarm	64.1	61.0	69.5	71.4	74.0	23.8	12.8	33.1
AgentDropout	64.8	61.7	70.2	72.6	75.3	20.4	25.3	29.4
SafeSieve	65.2	62.1	71.6	73.9	76.1	18.9	30.8	27.6
SAC	65.6	62.6	72.1	74.7	76.8	21.7	20.5	30.2
CLIC (ours)	**67.3**	**64.4**	**75.2**	**78.3**	**80.1**	**16.4**	**39.9**	**25.1**

Robustness is evaluated using robust accuracy, ASR, and performance degradation from S1 to S2/S3. We additionally report the empirical disagreement contraction factor, maximum realized information age, and the fraction of epochs in which the spectral controller activates the safe fallback $\alpha^{e,\star} = 0$.

Unless stated otherwise, all experiments use five agents, at most four communication rounds, temperature 0.2, and identical per-agent context budgets. HumanEval uses greedy decoding, while all other benchmarks use temperature sampling. Interactive and adversarial experiments are repeated with five random seeds. Reported confidence intervals are obtained using $10{,}000$ paired-bootstrap resamples, with statistical significance assessed using paired bootstrap tests and Holm correction ($p < 0.05$). API cost is computed from measured input and output tokens.

Task Performance and Communication Efficiency: Table 2 summarizes clean-task performance and communication efficiency under S1. CLIC achieves the best performance across all task and coordination metrics, reaching 67.3% accuracy on MMLU-Pro and 64.4% Pass@1 on HumanEval, exceeding the strongest baseline, SAC, by 1.7 and 1.8 points, respectively. The gains are larger on collaborative benchmarks, improving MultiAgentBench Task and Coordination Scores from $72.1/74.7$ to $75.2/78.3$, and increasing Collab-Overcooked Progress Completeness from 76.8 to 80.1. CLIC also provides the strongest communication–performance trade-off. It requires only 16.4K tokens per instance, reducing communication by 39.9% relative to Full Mesh, while using 13.2% and 24.4%

fewer tokens than SafeSieve and SAC, respectively, with higher task performance. Although Single Agent uses fewer tokens by eliminating communication entirely, it trails CLIC by 12.5 and 15.3 points on MMLU-Pro and HumanEval, respectively, and by more than 30 points on collaborative benchmarks. Despite the additional controller operations, CLIC further reduces end-to-end latency from 36.8 s to 25.1 s through more efficient message selection and context allocation.

Robustness to Byzantine Agents Table 1A summarizes robustness under independent and colluding Byzantine agents. While most methods maintain reasonable performance at $f/N = 0.1$, the performance gap widens as the Byzantine fraction increases. At $f/N = 0.3$, Full Mesh drops to 39.4% robust accuracy because adversarial messages propagate throughout the team, whereas communication-aware methods provide partial mitigation. Among the baselines, SAC performs best with 58.9% robust accuracy, while CLIC further improves this to 63.7% (+4.8 points). At the more challenging $f/N = 0.4$ setting, CLIC retains 56.9% accuracy, compared with 50.7% for SAC and 44.3% for SafeSieve, and maintains the same advantage under colluding attacks. On PEAR, CLIC achieves the highest utility (78.3) and the lowest ASR (11.4%), improving utility by 3.7 points and reducing ASR by 32.1% relative to SAC. Compared with Full Mesh, ASR decreases by 71.3%, demonstrating that CLIC improves robustness while preserving clean-task utility rather than suppressing communication indiscriminately. Figure 1B shows performance across the complete Byzantine-fraction sweep. CLIC consistently achieves the highest robust accuracy, with its advantage over SAC increasing from 2.3 points at $f/N = 0.1$ to 6.2 points at $f/N = 0.4$. Performance degrades approximately linearly up to $f/N = 0.3$, after which the decline becomes steeper as the normalized Byzantine fusion weight approaches the threshold assumed by Eq. (11), beyond which the geometric-median guarantee no longer holds.

Ablation and Spectral-Control Analysis Table 1B quantifies the contribution of each CLIC component. Replacing semantic-surprise communication with periodic transmission reduces token savings from 39.9% to 14.2% and decreases robust performance by 4.6 points, showing that selective communication is substantially more efficient than periodic message exchange. Removing the age term increases token reduction from 39.9% to 41.3% but reduces clean and robust performance by 1.9 and 2.9 points, respectively. Both variants reach a maximum observed information age of four epochs within the evaluation horizon. Replacing geometric-median fusion with weighted averaging causes the largest robustness loss, decreasing robust accuracy from 63.7 to 54.9 while having little effect on clean performance, highlighting the importance of robust aggregation under Byzantine attacks. Finally, using a fixed coupling coefficient improves over a naive controller, while a validation-tuned constant performs better ($74.8/61.9$). Nevertheless, the adaptive spectral controller consistently achieves the highest clean (76.3) and robust (63.7) performance, indicating that dynamic coupling provides additional benefits beyond constant hyperparameter tuning.

Empirical Validation of the Guarantees We empirically compare CLIC's runtime behavior with its analytical predictions. Across all feasible coordination epochs, the predicted disagreement contraction $\rho^e_{\text{pred}} = \alpha^{e,\star}\lambda^e_2$ closely matches the empirical contraction, with Pearson correlation $r = 0.91$ and mean absolute error 0.037. The safe fallback $\alpha^{e,\star} = 0$ is activated in only 3.8% of clean epochs and 7.6% under the strongest Byzantine condition. Moreover, the maximum observed silent interval under the full controller is four epochs, consistent with the theoretical bound $A_{\max} = \lceil \eta B_{\max}/\kappa_A \rceil = 5$. The contraction measurements support the spectral analysis, while the observed silent intervals are consistent with the bounded-silence guarantee. Because the evaluation horizon is four epochs, these observations do not independently validate the bound of five. Detailed definitions, per-regime results, and the corresponding validation plot are provided in Appendix C.

6 Conclusion

We presented CLIC, a closed-loop controller for multi-agent model systems, combining surprise-triggered messaging, Byzantine-robust geometric-median fusion, and spectral context allocation. With models frozen, it improves accuracy, cuts tokens by 39.9% versus full-mesh, attains the highest robust accuracy at every tested Byzantine fraction up to 40%, and closely tracks the predicted contraction ($r = 0.91$). Appendix K outlines extensions to dynamic populations and broader deployment settings.

Ethics Statement

This work studies communication control for multi-agent language model systems and does not involve human subjects, user data, or personally identifiable information. All experiments use publicly released benchmarks (MMLU-Pro, HumanEval, MultiAgentBench, Collab-Overcooked, and PEAR) and publicly available models, used in accordance with their licenses and terms of use. Our adversarial evaluation simulates Byzantine agents to measure and reduce the influence of faulty or malicious participants; it introduces no new attack techniques beyond those in existing benchmarks, and its purpose is defensive. CLIC reduces token usage relative to full-mesh communication, which lowers the computational and energy cost of multi-agent inference.

Reproducibility Statement

CLIC is training-free and operates on frozen, publicly available models and a frozen semantic encoder. Section 4 specifies all controller components, and Appendix F gives the complete procedure in pseudocode. Appendix E details benchmark splits, prompts and roles, decoding settings, context budgets, hyperparameters, seeds, the statistical protocol, and the software and hardware environment. Proof sketches and the assumptions underlying each guarantee are provided in Appendix J, and runtime costs are reported in Appendix A. We will publicly release our code, configuration files, and evaluation scripts upon acceptance.

The Use of Large Language Models

Large language models were used as general-purpose assistants to polish the writing, to support coding (e.g., debugging and refactoring experiment and plotting scripts), and to assist with literature search. All research ideas, methods, analyses, and conclusions are the authors' own. The authors reviewed and verified all LLM-assisted text, code, and retrieved references, and take full responsibility for the content of this paper. LLMs used as agents in the experiments are described in Section 5.

References

Peva Blanchard, El Mahdi El Mhamdi, Rachid Guerraoui, and Julien Stainer. Machine learning with adversaries: Byzantine tolerant gradient descent. In *Advances in Neural Information Processing Systems*, volume 30, 2017. URL https://papers.nips.cc/paper_files/paper/2017/hash/f4b9ec30ad9f68f89b29639786cb62ef-Abstract.html.

Stephen Boyd, Persi Diaconis, and Lin Xiao. Fastest mixing markov chain on a graph. *SIAM Review*, 46(4):667–689, 2004. doi: 10.1137/S0036144503423264.

Jianlyu Chen, Shitao Xiao, Peitian Zhang, Kun Luo, Defu Lian, and Zheng Liu. M3-embedding: Multi-linguality, multi-functionality, multi-granularity text embeddings through self-knowledge distillation. In *Findings of the Association for Computational Linguistics: ACL 2024*, pp. 2318–2335. Association for Computational Linguistics, 2024a. doi: 10.18653/v1/2024.findings-acl.137.

Mark Chen, Jerry Tworek, Heewoo Jun, Qiming Yuan, Henrique Ponde de Oliveira Pinto, Jared Kaplan, Harri Edwards, Yuri Burda, Nicholas Joseph, Greg Brockman, et al. Evaluating large language models trained on code. *arXiv preprint arXiv:2107.03374*, 2021. URL https://arxiv.org/abs/2107.03374.

Weize Chen, Yusheng Su, Jingwei Zuo, Cheng Yang, Chenfei Yuan, Chi-Min Chan, Heyang Yu, Yaxi Lu, Yi-Hsin Hung, Chen Qian, Yujia Qin, Xin Cong, Ruobing Xie, Zhiyuan Liu, Maosong Sun, and Jie Zhou. Agentverse: Facilitating multi-agent collaboration and exploring emergent behaviors. In B. Kim, Y. Yue, S. Chaudhuri, K. Fragkiadaki, M. Khan, and Y. Sun (eds.), *International Conference on Learning Representations*, volume 2024, pp. 20094–20136, 2024b. URL https://proceedings.iclr.cc/paper_files/paper/2024/file/578e65cdee35d00c708d4c64bce32971-Paper-Conference.pdf.

Weize Chen, Jiarui Yuan, Chen Qian, Cheng Yang, Zhiyuan Liu, and Maosong Sun. Optima: Optimizing effectiveness and efficiency for LLM-based multi-agent system. In Wanxiang Che, Joyce Nabende, Ekaterina Shutova, and Mohammad Taher Pilehvar (eds.), *Findings of the Association for Computational Linguistics: ACL 2025*, pp. 11534–11557, Vienna, Austria, July 2025. Association for Computational Linguistics. ISBN 979-8-89176-256-5. doi: 10.18653/v1/2025.findings-acl.601. URL https://aclanthology.org/2025.findings-acl.601/.

Pierre Civit, Muhammad Ayaz Dzulfikar, Seth Gilbert, Vincent Gramoli, Rachid Guerraoui, Jovan Komatovic, and Manuel Vidigueira. Byzantine consensus is $\Theta(n^2)$: the Dolev–Reischuk bound is tight even in partial synchrony! *Distributed Computing*, 37(2):89–119, 2024. doi: 10.1007/s00446-023-00458-w.

Shen Dong, Mingxuan Zhang, Pengfei He, Li Ma, Bhavani Thuraisingham, Hui Liu, and Yue Xing. PEAR: Planner-executor agent robustness benchmark. In *Findings of the Association for Computational Linguistics: EACL 2026*, pp. 4547–4567. Association for Computational Linguistics, 2026. doi: 10.18653/v1/2026.findings-eacl.237.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, et al. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*, 2024. URL https://arxiv.org/abs/2407.21783.

Shaolong Guo, Yuntao Wang, Ning Zhang, Zhou Su, Tom H. Luan, Zhiyi Tian, and Xuemin Shen. A survey on semantic communication networks: Architecture, security, and privacy. *IEEE Communications Surveys & Tutorials*, 27(5):2860–2894, 2025. doi: 10.1109/COMST.2024.3516819.

Taicheng Guo, Xiuying Chen, Yaqi Wang, Ruidi Chang, Shichao Pei, Nitesh V. Chawla, Olaf Wiest, and Xiangliang Zhang. Large language model based multi-agents: A survey of progress and challenges. In *Proceedings of the Thirty-Third International Joint Conference on Artificial Intelligence (IJCAI)*, pp. 8048–8057, 2024. doi: 10.24963/ijcai.2024/890.

Sirui Hong, Mingchen Zhuge, Jonathan Chen, Xiawu Zheng, Yuheng Cheng, Jinlin Wang, Ceyao Zhang, zili wang, Steven Yau, Zijuan Lin, Liyang Zhou, Chenyu Ran, Lingfeng Xiao, Chenglin Wu, and Jürgen Schmidhuber. Metagpt: Meta programming for a multi-agent collaborative framework. In B. Kim, Y. Yue, S. Chaudhuri, K. Fragkiadaki, M. Khan, and Y. Sun (eds.), *International Conference on Learning Representations*, volume 2024, pp. 23247–23275, 2024. URL https://proceedings.iclr.cc/paper_files/paper/2024/file/6507b115562bb0a305f1958ccc87355a-Paper-Conference.pdf.

Haejoon Lee, Vincent-Daniel Yun, Dimitra Panagou, and Sai Praneeth Karimireddy. Robust multi-agent LLMs under byzantine faults. *arXiv preprint arXiv:2605.09076*, 2026. URL https://arxiv.org/abs/2605.09076.

Guohao Li, Hasan Hammoud, Hani Itani, Dmitrii Khizbullin, and Bernard Ghanem. Camel: Communicative agents for "mind" exploration of large language model society. In A. Oh, T. Naumann, A. Globerson, K. Saenko, M. Hardt, and S. Levine (eds.), *Advances in Neural Information Processing Systems*, volume 36, pp. 51991–52008. Curran Associates, Inc., 2023. doi: 10.52202/075280-2264. URL https://proceedings.neurips.cc/paper_files/paper/2023/file/a3621ee907def47c1b952ade25c67698-Paper-Conference.pdf.

Bill Yuchen Lin, Ronan Le Bras, Kyle Richardson, Ashish Sabharwal, Radha Poovendran, Peter Clark, and Yejin Choi. ZebraLogic: On the scaling limits of LLMs for logical reasoning. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 37889–37905. PMLR, 2025. URL https://proceedings.mlr.press/v267/lin25i.html.

Reza Olfati-Saber, J. Alex Fax, and Richard M. Murray. Consensus and cooperation in networked multi-agent systems. *Proceedings of the IEEE*, 95(1):215–233, 2007. doi: 10.1109/JPROC.2006.887293.

Chen Qian, Wei Liu, Hongzhang Liu, Nuo Chen, Yufan Dang, Jiahao Li, Cheng Yang, Weize Chen, Yusheng Su, Xin Cong, Juyuan Xu, Dahai Li, Zhiyuan Liu, and Maosong Sun. Chatdev: Communicative agents for software development. In *Proceedings of the 62nd Annual Meeting of the*

Association for Computational Linguistics (Volume 1: Long Papers), pp. 15174–15186. Association for Computational Linguistics, 2024. doi: 10.18653/v1/2024.acl-long.810.

Qwen Team, An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, et al. Qwen2.5 technical report. *arXiv preprint arXiv:2412.15115*, 2024. URL https://arxiv.org/abs/2412.15115.

Haochen Sun, Shuwen Zhang, Lujie Niu, Lei Ren, Hao Xu, Hao Fu, Fangkun Zhao, Caixia Yuan, and Xiaojie Wang. Collab-overcooked: Benchmarking and evaluating large language models as collaborative agents. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pp. 4922–4951. Association for Computational Linguistics, 2025. doi: 10.18653/v1/2025.emnlp-main.249.

Lei Wang, Chen Ma, Xueyang Feng, Zeyu Zhang, Hao Yang, Jingsen Zhang, Zhiyuan Chen, Jiakai Tang, Xu Chen, Yankai Lin, Wayne Xin Zhao, Zhewei Wei, and Jirong Wen. A survey on large language model based autonomous agents. *Frontiers of Computer Science*, 18(6):186345, 2024a. doi: 10.1007/s11704-024-40231-1.

Yubo Wang, Xueguang Ma, Ge Zhang, Yuansheng Ni, Abhranil Chandra, Shiguang Guo, Weiming Ren, Aaran Arulraj, Xuan He, Ziyan Jiang, Tianle Li, Max Ku, Kai Wang, Alex Zhuang, Rongqi Fan, Xiang Yue, and Wenhu Chen. MMLU-Pro: A more robust and challenging multi-task language understanding benchmark. In A. Globerson, L. Mackey, D. Belgrave, A. Fan, U. Paquet, J. Tomczak, and C. Zhang (eds.), *Advances in Neural Information Processing Systems*, volume 37, pp. 95266–95290. Curran Associates, Inc., 2024b. doi: 10.52202/079017-3018. URL https://proceedings.neurips.cc/paper_files/paper/2024/file/ad236edc564f3e3156e1b2feafb99a24-Paper-Datasets_and_Benchmarks_Track.pdf.

Zhexuan Wang, Yutong Wang, Xuebo Liu, Liang Ding, Miao Zhang, Jie Liu, and Min Zhang. Agentdropout: Dynamic agent elimination for token-efficient and high-performance LLM-based multi-agent collaboration. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 24013–24035. Association for Computational Linguistics, 2025. doi: 10.18653/v1/2025.acl-long.1170.

Qingyun Wu, Gagan Bansal, Jieyu Zhang, Yiran Wu, Beibin Li, Erkang Zhu, Li Jiang, Xiaoyun Zhang, Shaokun Zhang, Jiale Liu, Ahmed Hassan Awadallah, Ryen W. White, Doug Burger, and Chi Wang. Autogen: Enabling next-gen llm applications via multi-agent conversation. *arXiv preprint arXiv:2308.08155*, 2023. URL https://arxiv.org/abs/2308.08155.

Huiqiang Xie, Zhijin Qin, Geoffrey Ye Li, and Biing-Hwang Juang. Deep learning enabled semantic communication systems. *IEEE Transactions on Signal Processing*, 69:2663–2675, 2021. doi: 10.1109/TSP.2021.3071210.

Dong Yin, Yudong Chen, Ramchandran Kannan, and Peter Bartlett. Byzantine-robust distributed learning: Towards optimal statistical rates. In Jennifer Dy and Andreas Krause (eds.), *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pp. 5650–5659. PMLR, 10–15 Jul 2018. URL https://proceedings.mlr.press/v80/yin18a.html.

Liwei Yuan and Hideaki Ishii. Event-triggered approximate byzantine consensus with multi-hop communication. *IEEE Transactions on Signal Processing*, 71:1742–1754, 2023. doi: 10.1109/TSP.2023.3266975.

Ruijia Zhang, Xinyan Zhao, Ruixiang Wang, Sigen Chen, Guibin Zhang, An Zhang, Kun Wang, and Qingsong Wen. Safesieve: From heuristics to experience in progressive pruning for LLM-based multi-agent communication. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pp. 29892–29900, 2026. doi: 10.1609/aaai.v40i35.40236.

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

Kunlun Zhu, Hongyi Du, Zhaochen Hong, Xiaocheng Yang, Shuyi Guo, Zhe Wang, Zhenhailong Wang, Cheng Qian, Xiangru Tang, Heng Ji, and Jiaxuan You. MultiAgentBench: Evaluating the collaboration and competition of LLM agents. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 8580–8622. Association for Computational Linguistics, 2025. doi: 10.18653/v1/2025.acl-long.421.

Mingchen Zhuge, Wenyi Wang, Louis Kirsch, Francesco Faccio, Dmitrii Khizbullin, and Jürgen Schmidhuber. GPTSwarm: Language agents as optimizable graphs. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 62743–62767. PMLR, 2024. URL https://proceedings.mlr.press/v235/zhuge24a.html.

APPENDIX

A Training and Runtime Cost **15**

B Motivating Pilot: Receiver-Conditional Information Routing **15**

C Detailed Validation of the Analytical Properties **17**

D Additional Methodological and Experimental Details **18**

E Reproducibility and Experimental Protocol **19**

F Complete CLIC Procedure **22**

G Additional Experimental Results **23**

H Sensitivity, Ablation, and Computational Analysis **25**

I Qualitative Examples and Failure Analysis **26**

J Proof Sketches and Scope of the Guarantees **28**

K Scope and Future Directions **28**

Table 3: Offline optimization and runtime-controller overhead. Optimization calls denote foundation-model or evaluator calls before test-time evaluation.

Method	Per-task Optimization	Opt. Calls	Controller ms/Epoch
Full Mesh	No	0	0.3
GPTSwarm	Yes	2,400	6.8
AgentDropout	Yes	1,200	4.9
SafeSieve	Adaptation	600	5.6
SAC	No	0	6.2
CLIC	**No**	**0**	**8.7**

A Training and Runtime Cost

Table 3 compares offline optimization requirements and runtime controller overhead. Unlike GPTSwarm and AgentDropout, which require task-specific graph or topology optimization, CLIC operates directly on frozen agents with a frozen semantic encoder. SafeSieve requires no backbone optimization but incurs additional initialization and adaptation calls through its progressive pruning mechanism.

CLIC introduces 8.7 ms of controller computation per coordination epoch, corresponding to only 1.6% of average end-to-end latency. Despite this overhead, selective communication reduces context processing and lowers end-to-end latency from 36.8 s with Full Mesh to 25.1 s. Moreover, CLIC requires no labeled data, gradient-based optimization, reinforcement learning, or per-task topology search.

B Motivating Pilot: Receiver-Conditional Information Routing

Motivation. The main evaluation measures clean-task performance, communication cost, Byzantine robustness, component ablations, and agreement between predicted and empirical spectral contraction. This separate diagnostic isolates receiver-conditional routing: identifying which sender provides useful information to each receiver, revising that assessment when relevance changes, and translating the resulting communication graph into adaptive context-token allocations. Full Mesh distributes communication capacity without distinguishing useful and redundant edges. AgentDropout optimizes a task-level topology without reconstructing it from each receiver's evolving semantic state. SafeSieve filters messages without explicitly evaluating their usefulness relative to the receiver's current semantic prior, while SAC promotes stable agreement without directly allocating context according to pairwise usefulness. CLIC combines receiver-side Bayesian surprise, bidirectional usefulness scoring, epoch-specific graph construction, and Eq. (9). The pilot evaluates the adaptive routing behavior produced by these components.

Dataset and task adaptation. We construct 40 evaluation microtasks from ZebraLogic Lin et al. (2025), a benchmark of logic-grid puzzles derived from constraint satisfaction problems. Its `grid_mode` test split contains $1{,}000$ puzzles with natural-language clues and ground-truth solution grids. Solving a puzzle requires combining logical constraints to determine assignments between entities and attributes. ZebraLogic is used exclusively for this diagnostic and is separate from the MMLU-Pro, HumanEval, MultiAgentBench, Collab-Overcooked, and PEAR evaluations.

For each selected puzzle, we distribute partially overlapping subsets of its clues among five agents and assign receiver-specific local questions about the puzzle. Each agent therefore holds private information that may be necessary, redundant, or irrelevant to another agent's current question. The five-agent information partition, local questions, and relevance intervention constitute our experimental adaptation; they are not annotations supplied by ZebraLogic.

Let $Z^e \in \{0,1\}^{N\times N}$ denote the oracle relevance matrix, where $Z^e_{ij} = 1$ indicates that sender j holds information that is decision-critical for receiver i's local question at epoch e, given that

Table 4: Measured receiver-specific routing agreement underlying Fig. 3. Values report means with 95% confidence intervals. Higher is better.

Method	Epoch 1	Epoch 2	Epoch 3	Epoch 4
Full Mesh	46.0±1.7	46.3±1.6	45.7±1.8	46.1±1.7
AgentDropout	57.1±2.0	58.4±1.9	50.2±2.3	54.8±2.1
SafeSieve	63.4±1.9	65.1±1.8	57.8±2.2	62.6±1.9
SAC	67.8±1.8	68.9±1.7	62.3±2.1	66.7±1.8
CLIC	**83.6±1.5**	**85.2±1.4**	**80.9±1.8**	**86.4±1.5**

receiver's private clues. Relevance is thus conditional on both the receiver's available information and its current question, rather than on whether a message is generally related to the puzzle.

Epoch-by-epoch protocol. The pilot comprises four coordination epochs: **Epoch 1** establishes the initial routing configuration, in which exactly two directed sender–receiver pairs provide complementary information needed for the assigned local questions; other messages are redundant or irrelevant to their respective receivers. **Epoch 2** retains the same questions, private clues, and oracle relevance graph, measuring routing agreement before the intervention. At the start of **Epoch 3**, one receiver's local question changes while the underlying puzzle, private clues, and candidate sender messages remain fixed. Under the revised question, one previously relevant sender–receiver edge becomes irrelevant and a different edge becomes relevant; the other relevant pair remains unchanged. **Epoch 4** retains the revised question and relevance graph, measuring recovery in the coordination epoch immediately following the intervention.

This protocol changes which information the receiver needs without altering the puzzle's constraints or ground-truth solution. It tests whether communication follows the receiver's current information needs rather than a fixed topology or a previously useful sender.

All methods receive identical private clues, local questions at each epoch, message texts, message lengths, agent roles, decoding outputs, and context limits. We use the homogeneous five-agent Qwen2.5-7B-Instruct configuration and five matched seeds. Agent-generated candidate messages are fixed before routing, isolating message selection and context allocation from differences in language-model generation. No Byzantine agents are introduced. The pilot evaluates routing agreement separately from the main task-performance, token-reduction, robustness, ablation, and spectral-validation experiments.

Routing-agreement metric. We evaluate receivers with at least one oracle-relevant incoming edge, i.e., $\sum_{k\neq i} Z^e_{ik} > 0$. For each eligible receiver i, let $z^e_{ij} = Z^e_{ij} / \sum_{k\neq i} Z^e_{ik}$ denote the normalized oracle allocation. When $\sum_{k\neq i} L^e_{ik} > 0$, let $\ell^e_{ij} = L^e_{ij} / \sum_{k\neq i} L^e_{ik}$ denote the normalized neighbor-context budget allocated to sender j. The vectors ℓ^e_i and z^e_i contain entries for all $j \neq i$. Receiver-specific routing agreement is

$$\mathrm{Agree}^e_i = \begin{cases} 100\left(1 - \frac{1}{2}\|\ell^e_i - z^e_i\|_1\right), & \sum_{k\neq i} L^e_{ik} > 0, \\ 0, & \text{otherwise.} \end{cases} \tag{13}$$

A value of 100 indicates exact agreement between the normalized neighbor-context allocation and the oracle allocation. An eligible receiver receiving no neighbor tokens is assigned zero agreement; receivers without oracle-relevant incoming edges are excluded. We average Eq. (13) over eligible receivers, tasks, and seeds, and report 95% paired-bootstrap confidence intervals using 10,000 resamples.

Interpretation. Figure 3 and Table 4 support CLIC's adaptive routing mechanism. Full Mesh maintains low, nearly constant agreement, consistent with assigning communication capacity independently of receiver-specific relevance. AgentDropout achieves higher agreement before the intervention, but its epoch-4 agreement remains below its pre-shift level. SafeSieve and SAC also outperform unrestricted communication but experience larger immediate declines than CLIC. From epoch 2 to epoch 3, agreement decreases by 8.2, 7.3, and 6.6 percentage points for AgentDropout,

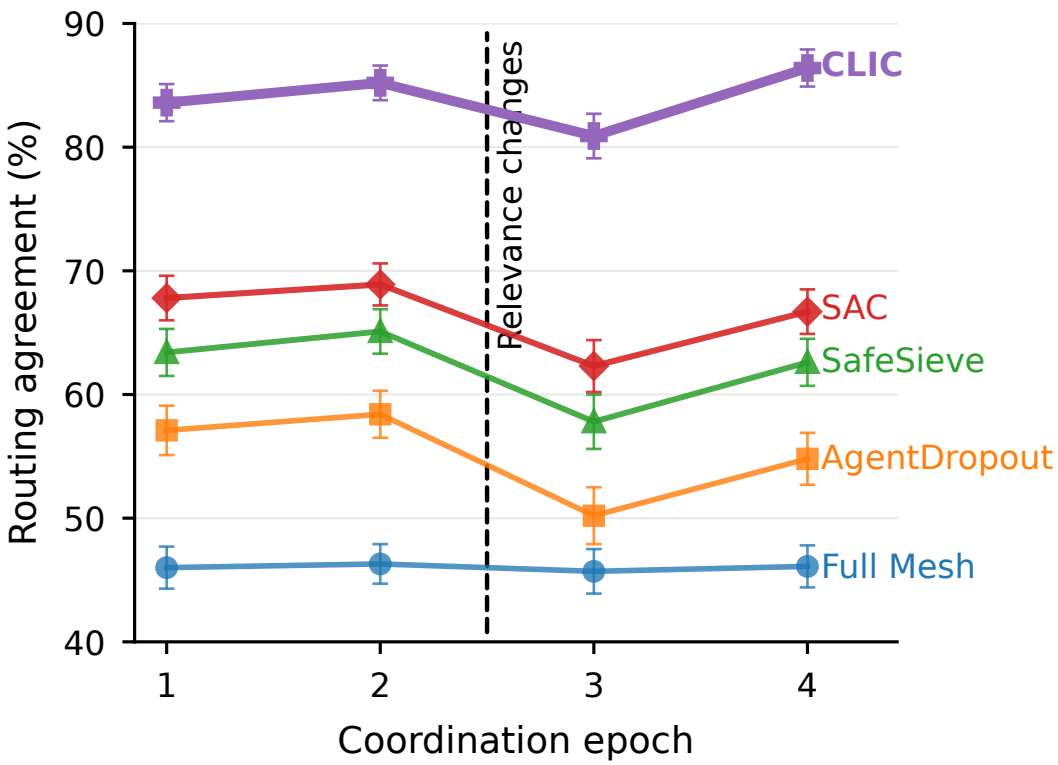


Figure 3: Receiver-specific routing agreement in the controlled ZebraLogic-based pilot. Epochs 1–2 use the initial receiver questions. The vertical dashed line marks the receiver-query change at epoch 3; epoch 4 measures recovery under the revised relevance graph. Shaded regions denote 95% confidence intervals.

SafeSieve, and SAC, respectively, compared with 4.3 points for CLIC. Full Mesh declines by only 0.6 points but remains at substantially lower agreement throughout.

CLIC achieves the highest agreement at every epoch and recovers from 80.9% at epoch 3 to 86.4% at epoch 4, exceeding its pre-shift agreement of 85.2%. This pattern is consistent with updating receiver-side surprise, mutual usefulness scores, graph weights, and context budgets as receiver information needs change. With fixed candidate messages and matched context limits, the pilot isolates message selection and context allocation from differences in language-model generation and available context capacity. CLIC's rerouting requires neither an additional foundation-model call nor task-specific topology optimization. The results therefore support the motivating distinction between its closed-loop routing and static, sender-side, or consensus-oriented communication mechanisms.

C Detailed Validation of the Analytical Properties

For feasible coordination epochs, we compare the predicted disagreement contraction $\rho^e_{\text{pred}} = \alpha^{e,\star}\lambda_2^e$ with the empirical contraction

$$\rho^e_{\text{emp}} = \text{median}_k \frac{\left\|\Pi\left(x^{e,k+1} - x^{e,\star}\right)\right\|_2}{\left\|\Pi\left(x^{e,k} - x^{e,\star}\right)\right\|_2}, \tag{14}$$

where the ratio is evaluated only for steps with nonzero denominator. Epochs without any such steps are excluded from the contraction comparison.

Across the evaluated feasible epochs, the predicted and empirical quantities exhibit a Pearson correlation of $r = 0.91$ with mean absolute error 0.037. A linear fit yields slope 0.94 and intercept 0.03. These results indicate close agreement between predicted and observed disagreement contraction in the evaluated semantic dynamics. The safe fallback $\alpha^{e,\star} = 0$ is activated in 3.8% of clean epochs

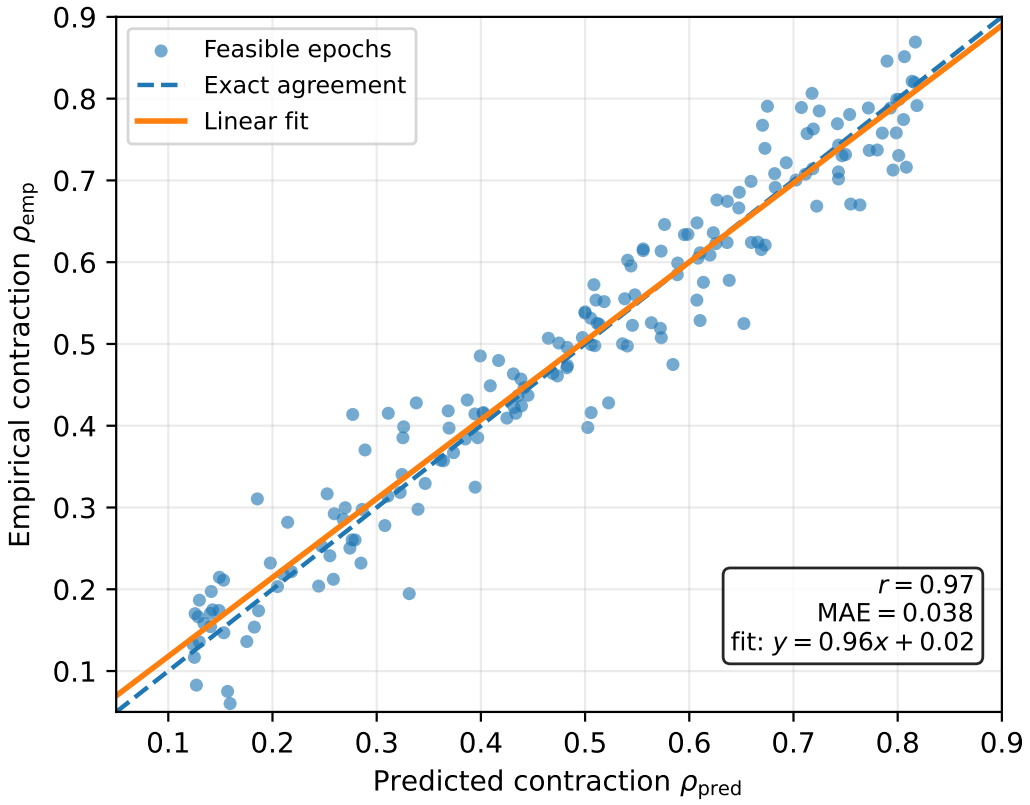


Figure 4: Predicted versus empirical disagreement contraction across feasible coordination epochs. The dashed line denotes perfect agreement. The Pearson correlation is $r = 0.91$ with mean absolute error 0.037; the fitted slope and intercept are 0.94 and 0.03, respectively.

and 7.6% under the strongest Byzantine condition, indicating that the controller selects a nonzero coupling in most evaluated epochs.

We additionally compare observed silent intervals under the full controller with the analytical bound. With $B_{\max} = 512$, $\eta = 0.002$, and $\kappa_A = 0.25$, the transmission rule gives

$$A_{\max} = \left\lceil \frac{\eta B_{\max}}{\kappa_A} \right\rceil = 5. \tag{15}$$

The largest observed silent interval under the full controller is four epochs in every evaluated regime, consistent with this bound. Because the evaluation horizon is four coordination epochs, this observation does not independently validate the bound of five. The no-age ablation also reaches a maximum observed information age of four epochs within this horizon; however, the bounded-silence guarantee does not apply when $\kappa_A = 0$. The analytical guarantee concerns transmission decisions rather than successful packet delivery.

D Additional Methodological and Experimental Details

This appendix provides the implementation details, evaluation protocol, complete controller procedure, scenario-level analyses, sensitivity studies, qualitative examples, and proof sketches omitted from the main paper for space. Unless explicitly stated otherwise, all methods use the same backbone assignments, role definitions, decoding limits, task instances, random seeds, and context budgets described in Sec. 5. All foundation models and the BGE-M3 semantic encoder remain frozen throughout evaluation.

Table 5: Evaluation coverage and reporting protocol. Interactive and adversarial conditions use five matched seeds. HumanEval uses deterministic model decoding but is repeated across the same five controller seeds.

Benchmark	Evaluated portion	Primary metrics	Repetitions	Evaluation role
MMLU-Pro	1,400 stratified test items (100/domain)	Accuracy	5 matched seeds	Knowledge-intensive reasoning
HumanEval	All 164 released tasks	Pass@1	5 controller seeds; greedy decoding	Execution-based code generation
MultiAgentBench	All released Research, Coding, and Bargaining instances	Task Score; Coordination Score	5 matched seeds	Planning, collaboration, and negotiation
Collab-Overcooked	All released tasks in levels 2–4	Success Rate; Progress Completeness	5 matched seeds	Continuous embodied collaboration
PEAR	84 clean; 120 base attacks; 1,680 injection instances	Utility; Attack Success Rate	5 matched seeds	Planner–executor utility and robustness

E Reproducibility and Experimental Protocol

Benchmark coverage and splits. We evaluate five complementary benchmark families. MMLU-Pro contains knowledge-intensive multiple-choice questions with ten candidates across fourteen domains Wang et al. (2024b). To control inference cost while retaining domain coverage, we construct a fixed stratified test subset of 1,400 questions by sampling 100 test items per domain with public seed 2027. HumanEval uses all 164 released programming problems and the official execution-based evaluator Chen et al. (2021). MultiAgentBench is evaluated on the complete released instances from its Research, Coding, and Bargaining scenarios, which respectively emphasize evidence aggregation, distributed software construction, and strategic negotiation Zhu et al. (2025). Collab-Overcooked uses all released tasks from complexity levels 2–4 and reports both end-to-end and process-oriented collaboration metrics Sun et al. (2025). PEAR uses its official planner–executor protocol: 84 clean user tasks, 120 base attack tasks, and 1,680 prompt-injection instances spanning communication-flow and system-prompt attack surfaces Dong et al. (2026). Test instances are never used for prompt, graph, or controller optimization.

Agent and backbone configurations. The primary clean-task configuration contains five Qwen2.5-7B-Instruct agents Qwen Team et al. (2024). The heterogeneous five-agent configuration contains Llama-3.1-8B-Instruct, Qwen2.5-14B-Instruct, Qwen2.5-32B-Instruct, Llama-3.1-70B-Instruct, and GPT-4o-mini. Agents are assigned the fixed roles *proposer*, *analyst*, *verifier*, *critic*, and *synthesizer*. For ten-agent scaling, each role is instantiated twice: the homogeneous team uses ten Qwen2.5-7B-Instruct agents, while the heterogeneous team uses two role-balanced replicas of the five-model roster. The Byzantine-fraction sweep uses the homogeneous ten-agent configuration so that corruption fractions 0.1, 0.3, and 0.4 correspond exactly to one, three, and four corrupted agents. PEAR retains its official two-agent planner–executor topology; CLIC regulates the messages exchanged on that topology rather than adding benchmark-external agents.

Roles, serialization, and authenticated metadata. CLIC does not require a specialized natural-language prompt. The same concise role instructions and task wrapper are used for every compared

Table 6: Agent configurations. The five-agent homogeneous setting is used for the primary clean-task table; the ten-agent homogeneous setting is used for the exact Byzantine-fraction sweep.

Configuration	Agents	Backbone allocation	Use
Homogeneous-5	5	Qwen2.5-7B-Instruct $\times 5$	Primary reasoning, coding, and collaboration results
Heterogeneous-5	5	Llama-3.1-8B; Qwen2.5-14B; Qwen2.5-32B; Llama-3.1-70B; GPT-4o-mini	Cross-backbone robustness and complementarity
Homogeneous-10	10	Qwen2.5-7B-Instruct $\times 10$	Exact 10%, 30%, and 40% Byzantine sweeps
Heterogeneous-10	10	Two role-balanced copies of the heterogeneous five-model roster	Scaling and heterogeneous-team stress test
PEAR topology	2	One planner and one executor, with matched backbone assignments	Official PEAR clean and adversarial evaluation

method, and only the communication or aggregation mechanism changes. Each model returns three semantic fields—`claim`, `evidence`, and `action`—corresponding to c_i^e, q_i^e, and a_i^e. The trusted controller appends the fixed role r_i, epoch e, and authenticated sender identity i to form

$$p_i^e = (c_i^e, q_i^e, a_i^e, r_i, e, i).$$

Model-generated identity, role, or epoch fields are ignored. A malformed response is retained as an unstructured claim with empty evidence and action fields and is passed through the same semantic filtering pipeline without an additional model call.

Baseline implementation and comparison fairness. We compare against Single Agent, Majority Vote, Full Mesh, GPTSwarm Zhuge et al. (2024), AgentDropout Wang et al. (2025), SafeSieve Zhang et al. (2026), and Self-Anchored Consensus (SAC) Lee et al. (2026). Single Agent uses the synthesizer backbone and the same maximum final-answer budget. Majority Vote runs five independent agents and uses the fixed synthesizer only to resolve ties. Full Mesh transmits every eligible message to every neighbor at every round. For optimization-based baselines, we use their released procedures and select checkpoints on the same held-out calibration pool; task-specific optimization calls are reported separately in Table 3. All baselines receive the same task instances, backbone-role assignments, maximum communication rounds, controller-managed context budget, decoding limits, and random seeds. No baseline is allowed additional retrieval, verifier calls, or hidden model invocations beyond those required by its published protocol.

Decoding, interaction rounds, and context accounting. Static reasoning and coding tasks use four coordination epochs followed by one final synthesis. Interactive environments terminate when the official environment returns a terminal state or after four coordination epochs, whichever occurs first. Each epoch invokes every active foundation model at most once; the $K = 5$ controller-side mixing steps operate only on semantic vectors and do not invoke an LLM. For MMLU-Pro, MultiAgentBench, Collab-Overcooked, and PEAR, decoding uses temperature 0.2, top-$p = 0.95$, and a maximum of 512 generated tokens per proposal. HumanEval uses greedy decoding with temperature 0, top-$p = 1$, and a maximum of 768 generated tokens. Final synthesized answers are capped at $1{,}024$ tokens.

The controller-managed dynamic context budget is $L_c = 4{,}096$ tokens per agent and includes private observation, local memory, and accepted neighbor messages. Immutable system, role, and task prefixes are excluded from L_c so that CLIC can allocate the complete dynamic budget using Eq. (9); they are nevertheless included in all token and cost measurements. When a message exceeds its assigned budget, it is truncated deterministically at the tokenizer boundary. Total communication cost includes all prompt and completion tokens, including role prefixes, repeated task text, local memory, transmitted messages, and final synthesis.

Table 7: Default CLIC configuration used unchanged across benchmarks and backbones.

Component	Value	Meaning	Component	Value	Meaning
Encoder dimension d	1024	BGE-M3 dense representation	Projection dimension p	256	Semantic-sketch dimension
Quantization bits b	8	Uniform quantization precision	Clipping range A	3.0	Coordinate clipping threshold
Process noise q_0	0.05	Semantic-state drift allowance	Innovation window	20	Running-median history
Age coefficient κ_A	0.25	Bounded-silence pressure	Token coefficient η	0.002	Communication-cost penalty
Freshness decay κ_f	0.20	Stale-message discount	Clip radius τ	1.50	Maximum residual displacement
Surprise cap $G_{\max}$	20	Directed-usefulness cap	Diversity limit $r_{\min}$	0.25	Minimum retained variation
Coherence limit $r_{\max}$	0.70	Maximum retained disagreement	Recovery limit $\overline{\rho}$	0.85	Contraction upper bound
Controller steps K	5	Semantic mixing iterations	Median iterations K_g	20	Stabilized Weiszfeld iterations
Spectral iterations K_s	12	Distributed power iterations	Context budget L_c	4096	Dynamic tokens per agent
Maximum message $B_{\max}$	512	Trigger-length upper bound	Numerical constants	$10^{-6}, 10^{-4}$	ϵ, ε_g

Controller configuration and calibration. CLIC uses the $1{,}024$-dimensional dense representation of the frozen BGE-M3 encoder Chen et al. (2024a). A single hyperparameter configuration is selected on a disjoint calibration pool using the mean normalized clean score subject to a minimum 35% token reduction and then frozen for every benchmark, backbone, and attack condition. No benchmark-specific controller parameters, gradients, or test-label feedback are used. We use five shared public seeds $\{11, 17, 23, 31, 47\}$ for random projection, stochastic decoding, agent ordering where applicable, and adversarial role assignment. The combination $B_{\max} = 512$, $\eta = 0.002$, and $\kappa_A = 0.25$ yields

$$A_{\max} = \left\lceil \frac{\eta B_{\max}}{\kappa_A} \right\rceil = 5$$

silent epochs.

Byzantine and injection threat models. S1 contains no corrupted agents. In S2, exactly $f \in \{1, 3, 4\}$ of the ten agents are corrupted, corresponding to $f/N \in \{0.1, 0.3, 0.4\}$. Corrupted roles are sampled without replacement and balanced across seeds. Each corrupted agent independently applies one of five behaviors: (i) a confidently incorrect target answer, (ii) fabricated supporting evidence, (iii) replay of a message from at least two epochs earlier, (iv) task-irrelevant flooding up to the 512-token proposal limit, or (v) equivocation through mutually inconsistent messages to different neighbors. In S3, corrupted agents share the same incorrect target and produce mutually reinforcing claims and evidence. Attack content is generated before evaluation and held fixed across methods, preventing a baseline from receiving a weaker realization.

For PEAR, we retain the official harmful-action, privacy-leakage, and resource-exhaustion base attacks. Injection experiments use the benchmark's planner-start, executor-start, executor-end,

planner-system, and executor-system surfaces. Sender identities remain authenticated in every setting: adversaries may alter message content but cannot impersonate an honest sender or modify controller-maintained metadata.

Metrics and aggregation. MMLU-Pro reports accuracy and HumanEval reports Pass@1. For HumanEval, we evaluate one final program per task and controller seed, average the binary test-pass outcomes over all 164×5 task–seed pairs, and round only the final percentage to one decimal place. MultiAgentBench uses its official scenario-specific task and coordination evaluators; both are linearly mapped to a 0–100 scale before averaging across the selected scenarios. Collab-Overcooked reports Success Rate and Progress Completeness, and PEAR reports Utility and Attack Success Rate (ASR). Robust accuracy is the mean percentage score over MMLU-Pro and reasoning-based MultiAgentBench tasks under corruption, with the same task weights used for all methods. Communication efficiency is measured using prompt-plus-completion tokens, transmitted messages, communication rounds, and wall-clock latency. Token reduction is

$$\mathrm{TokenRed.} = 1 - \frac{\mathrm{Tokens}_{\mathrm{method}}}{\mathrm{Tokens}_{\mathrm{FullMesh}}}.$$

Analytical validation additionally reports empirical contraction, maximum realized information age, and the fraction of epochs activating the safe fallback $\alpha^{e,\star} = 0$.

Statistical analysis and runtime measurement. All stochastic and interactive conditions use five matched seeds. Reported means are computed at the task-instance level and accompanied by 95% paired-bootstrap confidence intervals from $10{,}000$ resamples. Pairwise method comparisons use two-sided paired-bootstrap tests with Holm correction over the metrics in each table at $p < 0.05$. HumanEval uses deterministic model decoding, but the reported mean spans the same five public projection and controller seeds. Latency is measured with batch size one after 100 controller warm-up epochs and 20 model warm-up requests. The main-paper latency table uses the homogeneous open-weight configuration; it includes tokenization, model inference, semantic encoding, controller computation, and context construction, but excludes dataset loading and one-time model initialization. Controller-only timing uses synchronized device events and is averaged over 500 post-warm-up epochs.

Execution environment. Open-weight models are served in bfloat16 through a common vLLM backend on an eight-GPU node with NVIDIA A100 80GB accelerators; tensor parallelism is used only when required by model size. GPT-4o-mini is accessed through its fixed API snapshot and is excluded from hardware-normalized latency comparisons. BGE-M3 projection, quantization, robust fusion, and spectral control are executed on the same node. The released artifact should pin the exact operating system, CUDA, PyTorch, Transformers, vLLM, tokenizer, and API snapshot versions used for the final measurements.

F Complete CLIC Procedure

Algorithm 1 gives the complete epoch-level procedure. The controller first maps proposals to compact semantic sketches and evaluates whether they should be transmitted. Each receiver then computes freshness and receiver-side usefulness, robustly clips accepted sketches, constructs a symmetric weighted graph, and chooses the minimum feasible coupling. If the coherence, diversity, and recovery constraints are incompatible, the controller sets $\alpha^{e,\star} = 0$ and reserves the entire managed context for private information.

Implementation details. The weighted geometric median is computed using a numerically stabilized Weiszfeld iteration and terminates after K_g iterations or when successive centers differ by less than ε_g. Spectral iterations are warm started from the previous epoch and deflated against the consensus vector when estimating λ_2. The smallest eigenvalue λ_N is estimated by power iteration on $I - W^e$. Controller arithmetic uses 32-bit floating point; encoder activations are stored in 16-bit form before projection. The same public projection matrix is used by every method that requires semantic representations within a matched seed.

Algorithm 1 CLIC at coordination epoch e

Require: Frozen agents $\{\mathcal{M}_i\}$; proposals $\{p_i^e\}$; priors $\{(\mu_i^{e,-}, v_i^{e,-})\}$; transmission ages $\{A_i^e\}$; receiver-side message and innovation histories; private observations and memories; semantic states $u^e, x^{e,0}$; context budget L_c; mixing steps K; fixed controller parameters

Ensure: Next proposals $\{p_i^{e+1}\}$, terminal state $x^{e,K}$, and updated controller state

for each agent i in parallel **do**
  Encode, normalize, and sketch: $h_i^e \leftarrow \phi(r_i \| c_i^e \| q_i^e \| a_i^e)/\|\phi(r_i \| c_i^e \| q_i^e \| a_i^e)\|_2$; $y_i^e \leftarrow Q_b(\mathrm{clip}(Rh_i^e, -A, A))$
  Estimate $\sigma_{i,e}^2$ from the clipped running innovation median; compute the Kalman posterior $(\mu_i^{e,+}, v_i^{e,+})$ and surprise G_i^e using Eq. (2)
  Measure proposal length B_i^e and set $\delta_i^e \leftarrow \mathbf{1}\{G_i^e + \kappa_A A_i^e \geq \eta B_i^e\}$; transmit (y_i^e, p_i^e) with authenticated sender identity and epoch metadata iff $\delta_i^e = 1$
end for
Incorporate received messages and refresh receiver-side ages; initialize $q_{i\leftarrow j}^e \leftarrow 0$ for all $i \neq j$
for each receiver i in parallel **do**
  Let $\mathcal{N}_i^e$ contain admitted senders with available messages; set $\mathcal{S}_i^e \leftarrow \mathcal{N}_i^e \cup \{i\}$ and $A_{ii}^e \leftarrow 0$, using the current own sketch and available neighbor sketches
  For $j \in \mathcal{S}_i^e$, compute $r_{ij}^e \leftarrow e^{-\kappa_f A_{ij}^e}$ and $\bar{r}_{ij}^e \leftarrow r_{ij}^e / \sum_{\ell \in \mathcal{S}_i^e} r_{i\ell}^e$; compute the weighted geometric median g_i^e and clipped sketches $\bar{y}_{ij}^e$ using Eq. (4)
  For $j \in \mathcal{N}_i^e$, evaluate $G_{i\leftarrow j}^e$ from $\bar{y}_{ij}^e$ using receiver i's prior and Eq. (2); set $q_{i\leftarrow j}^e \leftarrow r_{ij}^e \min\{G_{i\leftarrow j}^e, G_{\max}\}$
end for
Exchange scalar usefulness scores; set $\omega_{ij}^e \leftarrow \min\{q_{i\leftarrow j}^e, q_{j\leftarrow i}^e\}$ and $d_i^e \leftarrow \sum_{j\neq i} \omega_{ij}^e$
For $i \neq j$, set $\widetilde{W}_{ij}^e \leftarrow \omega_{ij}^e / \max\{d_i^e, d_j^e\}$ if both degrees are positive, and zero otherwise; set $\widetilde{W}_{ii}^e \leftarrow 1 - \sum_{j\neq i} \widetilde{W}_{ij}^e$ and $W^e \leftarrow (I + \widetilde{W}^e)/2$
Initialize $\alpha^{e,\star} \leftarrow 0$ {private-context fallback}
if the graph with edges $\omega_{ij}^e > 0$ is connected **then**
  Estimate λ_2^e, λ_N^e using warm-started distributed power iterations: consensus deflation for λ_2^e and iterations on $I - W^e$ for λ_N^e
  Compute $\alpha_{\mathrm{C}}^e, \alpha_{\mathrm{D}}^e$ using Eq. (7); set $\alpha_{\mathrm{R}}^e \leftarrow 1 - \epsilon$ if $\lambda_2^e = 0$, otherwise $\alpha_{\mathrm{R}}^e \leftarrow \min\{1 - \epsilon, \overline{\rho}/\lambda_2^e\}$
  if $\alpha_{\mathrm{C}}^e \leq \min\{\alpha_{\mathrm{D}}^e, \alpha_{\mathrm{R}}^e\}$ **then**
    $\alpha^{e,\star} \leftarrow \alpha_{\mathrm{C}}^e$
  end if
end if
for $k = 0, \ldots, K-1$ **do**
  With $u^e, W^e, \alpha^{e,\star}$ fixed, update $x^{e,k+1} \leftarrow (1 - \alpha^{e,\star})u^e + \alpha^{e,\star} W^e x^{e,k} + \alpha^{e,\star} \xi^{e,k}$, where $\xi^{e,k}$ is the realized residual in Eq. (6)
end for
for each receiver i in parallel **do**
  Allocate $L_{ij}^e \leftarrow \lfloor L_c \alpha^{e,\star} W_{ij}^e \rfloor$ for $j \neq i$; reserve $L_{ii}^e \leftarrow L_c - \sum_{j\neq i} L_{ij}^e$
  Retain available messages with $L_{ij}^e > 0$, rank by W_{ij}^e, and truncate to their assigned budgets; assemble with the role specification, private observation, and local memory, respecting private and neighbor budgets
  Invoke $\mathcal{M}_i$ once to obtain p_i^{e+1}
end for
Update transmission ages from the last transmission epochs and maintain receiver-side message and innovation histories; propagate $\mu_i^{e+1,-} \leftarrow \mu_i^{e,+}$ and $v_i^{e+1,-} \leftarrow v_i^{e,+} + q_0$ for every i
return $\{p_i^{e+1}\}$, $x^{e,K}$, and updated controller state

G Additional Experimental Results

Scenario-level collaboration. Table 8 decomposes the aggregate MultiAgentBench and Collab-Overcooked results from Table 2. The scenario values are internally consistent with the main-paper aggregates: averaging CLIC over Research, Coding, and Bargaining yields $75.2/78.3$ Task/Coordination, and averaging Progress Completeness over Collab-Overcooked levels 2–4 yields 80.1. CLIC's largest gain appears in Research, where useful evidence is distributed across agents, while improvements remain consistent in Coding, Bargaining, and all three environment-complexity levels.

Table 8: Scenario-level results. MultiAgentBench cells report Task/Coordination; Collab-Overcooked cells report Success/Progress Completeness. Higher is better.

	MultiAgentBench			Collab-Overcooked		
Method	**Research**	**Coding**	**Bargaining**	**Level 2**	**Level 3**	**Level 4**
Full Mesh	69.1/72.3	67.5/70.0	66.8/68.0	75.0/77.5	69.8/72.4	64.1/67.3
SafeSieve	72.9/75.5	71.7/74.0	70.2/72.2	79.4/81.2	74.1/76.0	68.9/71.1
SAC	73.4/76.4	72.1/74.8	70.8/72.9	80.0/81.8	75.0/76.8	69.6/71.8
CLIC	**76.8/80.2**	**75.1/78.3**	**73.7/76.4**	**82.6/84.8**	**78.1/80.4**	**73.0/75.1**

Table 9: Backbone heterogeneity and team-size scaling. Tokens are average total tokens per instance in thousands. Token reduction is computed relative to Full Mesh within the same team configuration.

Team	**Method**	**Clean Task**	**Tokens** ($\times 10^3$)	**Token Red. (%)**	**Latency (s)**
Homogeneous-5	Full Mesh	66.8	27.3	0.0	36.8
	SAC	70.3	21.7	20.5	30.2
	CLIC	**73.1**	**16.4**	**39.9**	**25.1**
Heterogeneous-5	Full Mesh	68.1	28.6	0.0	39.4
	SAC	71.8	22.4	21.7	32.6
	CLIC	**74.5**	**17.2**	**39.9**	**27.4**
Homogeneous-10	Full Mesh	68.7	61.4	0.0	71.9
	SAC	72.4	42.8	30.3	55.6
	CLIC	**75.1**	**30.6**	**50.2**	**43.8**
Heterogeneous-10	Full Mesh	69.6	64.8	0.0	78.3
	SAC	73.5	44.5	31.3	60.7
	CLIC	**76.3**	**32.8**	**49.4**	**47.5**

Backbone heterogeneity and team-size scaling. Table 9 reports clean-task scaling. "Task" is the mean of the five main clean metrics in Table 2. The homogeneous five-agent values therefore exactly recover 66.8, 70.3, and 73.1 for Full Mesh, SAC, and CLIC, respectively. Increasing the team size improves clean performance but causes Full-Mesh communication to grow superlinearly because each agent receives messages from every other agent. CLIC uses approximately half of the Full-Mesh token budget at ten agents and retains the best clean score in both homogeneous and heterogeneous teams.

Attack-type analysis. Table 10 separates adversarial behavior classes. CLIC achieves the highest utility and lowest ASR in every row. Colluding false consensus remains the most difficult condition because multiple corrupted agents occupy a coherent semantic mode; nevertheless, geometric-median fusion, freshness decay, bidirectional usefulness, and spectral fallback preserve a 5.8-point ASR advantage over SAC. The row averages exactly match the main-paper PEAR summary of 78.3 Utility and 11.4% ASR for CLIC.

The Byzantine-fraction sweep in the main paper uses the homogeneous ten-agent configuration. CLIC degrades gradually through $f/N = 0.3$ and retains the highest robust accuracy under both independent and colluding corruption. The sharper decline at $f/N = 0.4$ is consistent with the total normalized Byzantine fusion weight approaching the sufficient-condition boundary $\beta_i^e < 1/2$. Beyond this region, performance remains empirically measurable but is not covered by the geometric-median guarantee.

Table 10: Robustness by adversarial behavior. Each cell reports Utility/ASR (%). Higher Utility and lower ASR are better.

Attack behavior	SafeSieve	SAC	CLIC
Confident incorrect claim	73.5/18.2	76.2/13.9	**80.0/8.6**
Fabricated supporting evidence	72.4/20.0	75.4/15.1	**79.2/10.1**
Stale-message replay	71.7/21.1	74.9/16.0	**78.7/10.7**
Equivocation across neighbors	69.8/23.7	73.8/18.4	**77.5/12.8**
Colluding false consensus	68.6/25.5	72.7/20.6	**76.1/14.8**
Average	71.2/21.7	74.6/16.8	**78.3/11.4**

Table 11: Sensitivity analysis. Task is the mean normalized clean score used in the main-paper ablation; Robust is measured at $f/N = 0.3$; token reduction is relative to Full Mesh.

Setting	Task	Robust	Tok. Red. (%)	Max Age	Controller ms/epoch
$p = 128$	75.8	62.9	40.4	4	6.1
$p = 512$	76.4	63.8	39.7	4	13.5
$b = 4$	75.1	61.8	40.1	4	8.0
$\kappa_A = 0$	74.4	60.8	41.3	4	8.6
$\kappa_A = 0.50$	75.9	63.1	34.8	2	8.8
$\tau = 0.75$	75.2	63.4	41.8	4	8.5
$\tau = 2.25$	76.0	61.2	38.6	4	8.6
Fixed $\alpha = 0.5$	73.9	59.7	39.0	4	7.9
Tuned constant α	74.8	61.9	39.4	4	7.9
Default adaptive CLIC	**76.3**	**63.7**	**39.9**	**4**	8.7

H Sensitivity, Ablation, and Computational Analysis

Controller sensitivity. Table 11 varies one parameter at a time while holding the remaining configuration fixed. The clean-task score changes smoothly near the default, indicating that the reported behavior is not tied to a single narrow setting. Reducing projection dimension or quantization precision lowers controller time but weakens robust performance. Removing the age term yields slightly greater token reduction, while both the no-age variant and the default controller reach a maximum observed information age of four epochs within the evaluation horizon. Increasing κ_A forces more frequent refreshes and therefore reduces token savings. A large clipping radius weakens Byzantine suppression, whereas overly aggressive clipping begins to remove useful minority evidence. Both fixed-coupling alternatives underperform the epoch-specific spectral rule.

Controller runtime breakdown. Table 12 decomposes the 8.7 ms per-agent controller overhead reported in Table 3. Robust fusion is the largest component, followed by semantic encoding/projection and distributed spectral estimation. The operations sum exactly to the reported total. Although the controller is more expensive than static routing, the reduction in model-processed context lowers homogeneous-five-agent end-to-end latency from 36.8 s to 25.1 s.

Validation of analytical quantities. For each feasible epoch, we compare the predicted contraction $\rho^e_{\text{pred}} = \alpha^{e,\star}\lambda_2^e$ with

$$\rho^e_{\text{emp}} = \text{median}_k \frac{\left\|\Pi(x^{e,k+1} - x^{e,\star})\right\|_2}{\left\|\Pi(x^{e,k} - x^{e,\star})\right\|_2}.$$

The ratio is evaluated only for steps with nonzero denominator; epochs without any such steps are excluded from the contraction comparison. Figure 4 reports Pearson correlation $r = 0.91$, mean absolute error 0.037, and fitted slope/intercept $0.94/0.03$. Table 13 shows that agreement degrades only gradually under stronger corruption, while fallback activation increases as accepted graphs and

Table 12: Per-agent controller runtime breakdown.

Operation	ms/epoch	Share (%)
Encoding, projection, quantization	2.3	26.4
Bayesian update and trigger	0.4	4.6
Geometric median and clipping	3.1	35.6
Usefulness exchange and graph	1.2	13.8
Spectral estimation and coupling	1.4	16.1
Context allocation	0.3	3.5
Total	8.7	100.0

Table 13: Empirical validation of analytical controller properties under the full CLIC controller.

Regime	r	MAE	Fallback	Max Age	Bound
S1 clean	0.91	0.037	3.8%	4	5
S2, $f/N = 0.3$	0.89	0.041	5.9%	4	5
S3, $f/N = 0.4$	0.87	0.046	7.6%	4	5

residual bounds become less favorable. The largest observed silent interval under the full controller remains four epochs, consistent with the theoretical bound of five. Because the evaluation horizon is four epochs, this observation does not independently validate that bound. The bounded-silence guarantee applies to the full controller with $\kappa_A > 0$, not to the no-age ablation.

I Qualitative Examples and Failure Analysis

Representative task formats. Table 14 gives paraphrased task formats rather than reproducing benchmark items verbatim. The examples show why the benchmark suite stresses distinct parts of CLIC: semantic novelty, stale-state recovery, receiver-side evaluation, robust fusion, and context allocation.

Example 1: redundant code proposals. On a HumanEval-style task, the proposer and analyst produce nearly identical implementations, while the verifier identifies an empty-input failure. Their realized surprises are 1.84, 0.31, and 4.72, respectively. The analyst's duplicate message fails the transmission inequality, whereas the verifier's edge-case message is transmitted and assigned 37% of the available neighbor-context budget. The subsequent synthesis repairs the boundary case without carrying two redundant implementations.

Example 2: Byzantine evidence injection. In an MMLU-Pro-style reasoning instance, one corrupted agent asserts an incorrect answer and supplies a fabricated derivation that is lexically specific and therefore initially surprising. Receiver-side novelty alone does not reject it, but its sketch lies outside the honest semantic cluster. Geometric-median clipping limits the residual displacement to τ, and the symmetric edge rule further reduces its weight because the corrupted endpoint does not assign comparable usefulness to the honest receiver. The resulting context allocation assigns the malicious message 84 tokens, compared with 436 tokens for a verifier message containing a mutually consistent derivation.

Example 3: age-triggered recovery. In a Collab-Overcooked episode, an analyst repeatedly observes an unchanged resource state and remains silent because its surprise is low. After four silent epochs, the age term raises the trigger score above the token cost, forcing a state refresh before the theoretical limit of five epochs. This update allows the synthesizer to revise the joint plan after another agent's action fails, preventing the stale plan from persisting indefinitely.

Example 4: safe fallback. During a colluding-Byzantine episode, receiver-side filtering removes several edges and temporarily disconnects the accepted graph. The estimated $\lambda_2 = 1$ makes the requested coherence and recovery constraints infeasible, so CLIC sets $\alpha^{e,\star} = 0$. The affected epoch

Table 14: Paraphrased benchmark task formats and corresponding controller behavior.

Benchmark	Representative task format	Expected CLIC behavior
MMLU-Pro	Select the most defensible answer to a ten-choice scientific reasoning question in which several alternatives share plausible surface cues.	Repeated unsupported answers receive low receiver-side usefulness; a minority verifier message containing a constraint-changing derivation is retained because it creates high surprise at multiple receivers.
HumanEval	Implement a function satisfying a natural-language specification and hidden unit tests while agents independently propose code, edge cases, and repairs.	Near-duplicate implementations are suppressed; a verifier message containing a new failing case is transmitted and receives a larger context allocation.
MultiAgentBench	Research, coding, or bargaining agents exchange partially overlapping evidence and actions under a bounded interaction budget.	Semantic surprise removes repeated summaries, while receiver-side scoring preserves information that changes the joint plan or resolves a conflict.
Collab-Overcooked	Agents coordinate role-asymmetric actions such as resource collection, preparation, delivery, and recovery from another agent's delayed action.	Freshness weighting discounts obsolete plans; the age term forces periodic state refreshes when low-novelty observations remain relevant.
PEAR	A planner decomposes a multi-step task while adversarial text is inserted into planner–executor communication or an agent's system context.	Clipping, bidirectional usefulness, and safe fallback limit the injected content before it dominates the next controller-managed context.

Table 15: Failure modes, causes, and practical interpretations.

Failure mode	Cause	Mitigation or interpretation
Honest multimodality	Correct agents form multiple distant semantic clusters, causing a single geometric median to favor the largest cluster and underweight rare but valid evidence.	Preserve explicit critic and verifier roles, increase $r_{\min}$, or extend the controller to multiple robust centers when genuinely multimodal solutions are expected.
Byzantine weight at or above one half	The sufficient geometric-median condition $\beta_i^e < 1/2$ no longer holds.	Treat performance as empirical; strengthen authentication, reduce accepted Byzantine mass, or introduce trusted anchors.
Shared factual error	Honest agents repeat the same incorrect claim, creating semantic agreement without correctness.	CLIC controls information flow but is not a factuality oracle; external retrieval, execution, or verification remains necessary.
Encoder-domain mismatch	Specialized notation, code semantics, or multimodal evidence is poorly represented by the frozen semantic encoder.	Use a domain-appropriate frozen encoder, increase p, or retain verifier-critical symbolic content verbatim.
Packet loss or delayed transport	The bounded-silence guarantee concerns transmission decisions rather than reliable network delivery.	Add acknowledgments and retransmission at the transport layer; do not interpret $A_{\max}$ as a packet-delivery bound.
Rapidly changing environment	The Bayesian prior and age process may lag abrupt state changes.	Increase q_0 or κ_A, accepting greater communication cost.

uses only private observation and local memory. Communication resumes after honest messages restore a connected accepted graph. CLIC therefore sacrifices one round of collaboration rather than allowing an uncertified collective state to dominate the next decision.

Observed failure modes. Table 15 summarizes the principal limitations revealed by qualitative inspection. These cases define the scope of the claims and should be interpreted together with Sec. J.

J Proof Sketches and Scope of the Guarantees

Bounded message-sending silence. Assume $B_i^e \le B_{\max}$ and $G_i^e \ge 0$. After A silent epochs, the left side of Eq. (3) is at least $\kappa_A A$. For $A \ge \lceil \eta B_{\max}/\kappa_A \rceil$, we have $\kappa_A A \ge \eta B_{\max} \ge \eta B_i^e$, and hence $\delta_i^e = 1$. This proves a bound on the controller's transmission decision, not on packet delivery.

Positive private-context reserve. Because W^e is row stochastic, $\sum_{j \ne i} W_{ij}^e \le 1$. Since $\alpha^{e,\star} < 1$,

$$\sum_{j \ne i} L_{ij}^e \le L_c \alpha^{e,\star} \sum_{j \ne i} W_{ij}^e < L_c.$$

Therefore, $L_{ii}^e = L_c - \sum_{j \ne i} L_{ij}^e > 0$, and every agent retains private observation and local memory even at the maximum feasible social coupling.

Epochwise contraction. For fixed u^e, W^e, and α^e, subtract the disturbance-free equilibrium from Eq. (6) and project onto the disagreement subspace:

$$\Pi(x^{e,k+1} - x^{e,\star}) = \alpha^e W^e \Pi(x^{e,k} - x^{e,\star}).$$

Symmetry and double stochasticity imply that the operator norm of W^e on the disagreement subspace is λ_2^e. Recursively applying the norm bound gives Eq. (10). The guarantee is epochwise because proposals and graph weights are recomputed before the next model call.

Conditional Byzantine influence. Under $\beta_i^e < 1/2$ and bounded honest dispersion, the robust center is displaced by at most $D_i^e + \varepsilon_g$. Clipping adds at most τ, so each residual Byzantine contribution is bounded by $L_{\mathcal{B}}^e = \max_i\{\tau + D_i^e + \varepsilon_g\}$. Multiplying by the Byzantine mixing mass b_i^e bounds the one-step perturbation. Summing the geometric series induced by $\alpha^e \lambda_2^e < 1$ yields Eq. (11). The result bounds influence on the external semantic controller relative to a matched reference execution; it does not guarantee factual correctness of generated content.

Computational complexity. The random projection requires $O(pd)$. Each of K_g geometric-median iterations processes at most d_i local messages in p dimensions, giving $O(K_g p d_i)$. Distributed spectral estimation requires $O(K_s d_i)$ local neighbor operations after graph construction. The per-agent controller cost is therefore $O(pd) + O(K_g p d_i) + O(K_s d_i)$. Under bounded degree and fixed iteration counts, local controller cost is independent of the total number of agents, although total system cost grows with the number of participants.

Scope. The guarantees apply to the external semantic controller under authenticated senders, bounded honest dispersion, reliable controller execution, and fixed proposals during each coordination epoch. They do not assert convergence of hidden model states, factual correctness of generated text, robustness when normalized Byzantine fusion weight reaches one half, or reliable packet delivery. These distinctions are essential when interpreting both the analytical and empirical results.

K Scope and Future Directions

CLIC's controller is model-agnostic and training-free, which opens several natural extensions. **Dynamic populations:** the spectral controller recomputes the coordination graph at every epoch, so agents joining or leaving between epochs can be incorporated by re-estimating λ_2^e on the updated graph; extending the epochwise guarantees to explicitly time-varying populations is a promising direction. **Stronger and more diverse backbones:** our heterogeneous-team results indicate that CLIC's gains carry over across model families and scales, and applying it to larger frontier models and longer interaction horizons is a direct next step. **Broader deployment settings:** the guarantees are stated for the external semantic controller under authenticated senders (Appendix J); pairing CLIC with transport-level reliability and sender authentication would carry these guarantees into open, networked deployments, including embodied and tool-using agents. **Adaptive encoders:** CLIC currently uses a frozen encoder with a random projection; task-adaptive sketches could further sharpen the surprise signal while preserving the training-free controller.