LLM-Coordinated Multi-Agent Planning: Balancing Performance and Bandwidth Constraints

Xue Li* Xiaonan Song*
Scholar42, InfiniPouch LLC
San Francisco, CA, USA, 94158
{xueli,ssong}@scholar42.com

Dowson Liu*
RapidsDB, Inc.
Lakewood, WA, USA, 98499
dowson.liu@rapidsdb.com

Abstract

This paper studies whether large language models (LLMs) can improve coordination in multi-agent systems when communication bandwidth is the primary bottleneck. We introduce a controlled Grid World benchmark in which five heterogeneous agents must collect typed resources under strict per-step communication budgets. We compare a Gemma-centered coordination pipeline using `gemma-4-26b-a4b-it` *against three non-LLM baselines: fair direct broadcast, a tabular Q learned protocol, and a tiny DQN learned protocol. Across 50 seeds and six low-bandwidth settings, the Gemma-based ranked coordination protocol achieves higher task completion than the learned baselines. At 200 bits per step, Gemma completes 86% of episodes, while tiny DQN completes 6%, tabular Q completes 2%, and direct broadcast completes none. The LLM protocol also uses much shorter transmitted messages, reaching 0.285 per-step average transmitted message tokens at 200 bits per step and 3148 average transmitted bits per episode. These results suggest that LLMs can act as compact, high-level coordinators in bandwidth-constrained multi-agent planning, but also expose important limitations: performance saturates beyond moderate bandwidth, and the current evidence is from a controlled benchmark rather than a physical deployment.*

1. Introduction

Multi-agent systems are increasingly deployed in complex environments, yet their efficacy often hinges on the ability to coordinate effectively, a capability frequently challenged by limitations in communication bandwidth. These bandwidth constraints can severely restrict the amount of information agents can exchange, thereby hindering their ability to reach consensus, optimize resource allocation, and adapt to dynamic conditions. This paper investigates the potential of large language models (LLMs) to serve as high-level coordinators within multi-agent systems, specifically targeting scenarios where communication bandwidth is a critical bottleneck.

Our work is motivated by the observation that LLMs can transform verbose, local observations into compact, task-level directives. In a communication-limited multi-agent system, this ability may be more valuable than raw message capacity: agents do not need to transmit every observation if a coordinator can select a small set of useful assignments. We therefore study a central LLM coordinator that receives compact state summaries and returns ranked target preferences for each agent. The resulting messages are short, structured, and budgeted by a per-step bandwidth constraint.

All results in this paper are generated from the benchmark and protocols described below. We construct a transparent Grid World task, define explicit baselines, and report the evaluation protocol used to produce the current results.

2. Related Work

Prior work on learned multi-agent communication studies how agents can induce useful message protocols from task rewards, including differentiable inter-agent communication and targeted communication mechanisms [4, 5]. Our work differs in that the communication policy is not learned end-to-end from scratch: an LLM-centered pipeline maps compact state summaries to ranked symbolic targets, and a deterministic resolver converts those rankings into executable assignments. This places the approach closer to semantic and selective communication under bandwidth constraints [10] than to fully emergent message learning. The broader motivation is also connected to LLM-based reasoning and human-robot interaction [8, 11], but our evaluation focuses narrowly on controlled multi-agent planning rather than open-ended dialogue or physical robot deployment.

*Equal contribution.

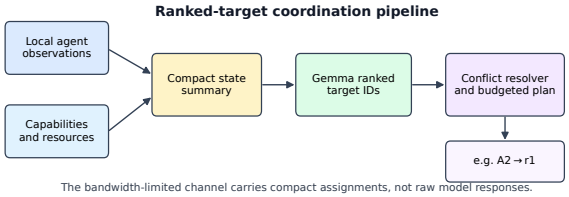

Figure 1. Gemma ranked-target coordination pipeline. The model receives compact task state and returns ranked symbolic target identifiers; a deterministic resolver converts them into compact budgeted assignments.

3. Methods

We evaluate LLM-coordinated multi-agent planning under bandwidth constraints using one LLM strategy and three non-LLM baselines: fair direct broadcast, tabular Q communication, and tiny DQN communication.

3.1. Communication Strategies

Gemma ranked LLM protocol: The evaluated LLM strategy is a pipeline rather than a bare model call. The coordinator receives compact observations, known resources, per-agent capabilities, and a list of valid target identifiers. Gemma returns ranked target identifiers for each agent. A deterministic resolver then removes invalid targets, resolves duplicate first-choice conflicts, and transmits compact messages such as `A2->r1`. Previously transmitted plans are reused until replanning is needed, so many decision steps transmit no new message. We evaluate this protocol with Google's Gemma-family model `gemma-4-26b-a4b-it`; we treat the full suffix as a provider model-variant identifier and avoid architectural assumptions beyond its instruction-tuned usage in this benchmark. Model-call failures are recorded explicitly; in the full evaluation reported below, all Gemma calls succeeded.

Tiny DQN protocol: This learned baseline uses a small deep Q-network to choose compact communication actions in the same Grid World environment. It provides a lightweight neural learned-protocol comparison point without requiring a large multi-agent training system.

Tabular Q protocol: This learned baseline uses a tabular Q policy trained on the same task family. It is intentionally lightweight and interpretable, but its state abstraction is less expressive than the DQN baseline.

Fair direct broadcast: The direct baseline fairly allocates the per-step bandwidth budget to agents and transmits local observations directly. This approach is simple but produces high attempted communication overhead and cannot reason globally about which information is most useful. Related communication bottlenecks are common in Internet of Things and sensor-network systems [2, 3, 7].

3.2. Training Details

The tabular and DQN baselines are fixed before evaluation and do not update online during test episodes. The tabular Q protocol is trained for 20,000 episodes on the same v2-lite Grid World configuration used in evaluation, using seed 0, a 500-bit training bandwidth, learning rate 0.2, discount 0.95, and epsilon annealed from 0.35 to 0.03. The tiny DQN protocol is trained for 100,000 episodes on the same task family, also with seed 0 and 500-bit training bandwidth, using a 128-dimensional hidden layer, replay buffer size 50,000, batch size 256, and a training update every 32 environment steps. The final evaluation loads fixed checkpoints for both baselines. These baselines are intentionally lightweight; they are useful comparisons against compact learned communication, but they should not be interpreted as exhaustive state-of-the-art multi-agent reinforcement learning systems.

3.3. Experimental Setup

The experimental setup involves simulating a multi-agent system operating in a defined environment. The environment is designed to present tasks that require coordinated planning and execution. Agents have limited sensing capabilities and must communicate to share information and coordinate their actions.

Environment and Task: We use a 20 by 20 Grid World with five agents, twelve resources, and random obstacles. Agents have observation radius 3 and a maximum of 90 steps. Resources are typed as food, metal, or energy, and agents have heterogeneous capabilities: some can collect only one resource type, while others can collect multiple types. An episode is successful when the team collects at least 80% of resources, i.e., 10 of 12 resources. Bandwidth constraints are imposed as per-step bit budgets, limiting how much target information can be transmitted. This reflects realistic scenarios where communication resources are limited [1].

Agents: Each agent is equipped with basic sensing and actuation capabilities. Agents can observe their local environment, move within the grid, and exchange messages with other agents or the central LLM, depending on the evaluated strategy.

Evaluation Protocol: We evaluate six bandwidth levels: 50, 100, 150, 200, 300, and 500 bits per step. Each strategy is evaluated on the same 50 random seeds at each bandwidth level, with the same environment generator, success criterion, horizon, action rules, and bandwidth accounting. The formal comparison therefore contains 1200 episodes: 900 non-LLM baseline episodes and 300 Gemma LLM episodes.

3.4. Evaluation Metrics

The performance of each communication strategy is evaluated based on the following metrics:

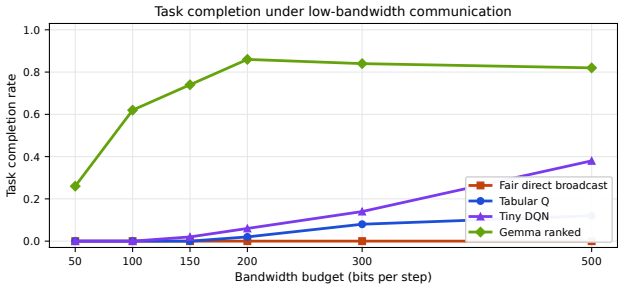

Figure 2. Task completion as a function of per-step bandwidth budget. Gemma reaches high completion in the low-bandwidth regime, while the learned baselines improve more gradually and fair direct broadcast remains bottlenecked.

Task Completion Rate: The percentage of tasks successfully completed by the multi-agent system within a specified time frame. This measures the effectiveness of the communication strategy in enabling coordinated task execution.

Message Length: The mean token count of transmitted messages, averaged across decision steps in an episode. This is not the full LLM response length: it measures the compact messages that enter the bandwidth-limited channel after parsing and conflict resolution. Plan-reuse steps transmit no message and therefore contribute zero. Shorter message lengths indicate more efficient communication.

Communication Overhead: The total amount of data transmitted between agents during task execution. This measures the overall communication burden on the network. Lower communication overhead indicates better utilization of bandwidth resources. The importance of efficient communication protocols in networked systems is further highlighted in [12].

By analyzing these strategies under the same simulator, seed set, and bandwidth grid, we aim to quantify the tradeoffs between task performance and communication cost in bandwidth-constrained multi-agent planning.

4. Results

The experimental results provide a quantitative assessment of the LLM-coordinated multi-agent system under varying bandwidth constraints, comparing its performance against learned protocols and direct communication methods. The core metrics analyzed include task completion rate, average message length, and communication overhead.

Figure 2 and Table 1 show that the Gemma ranked protocol achieves the best performance among the evaluated strategies in the low-bandwidth regime. At 50 bits per step, neither learned baseline completes any episode, while Gemma completes 26% of episodes and collects 8.26 resources on average. At 200 bits per step, Gemma reaches

Table 1. Task completion and resources across bandwidth budgets (50-seed average).

Strategy	50	100	150	200	300	500
Direct (comp.)	0.000	0.000	0.000	0.000	0.000	0.000
Direct (res.)	2.00	2.00	2.00	2.00	2.00	2.20
Tab-Q (comp.)	0.000	0.000	0.000	0.020	0.080	0.120
Tab-Q (res.)	3.40	4.48	4.84	5.54	5.88	6.58
Tiny-DQN (comp.)	0.000	0.000	0.020	0.060	0.140	0.380
Tiny-DQN (res.)	3.40	5.34	6.06	6.82	7.70	8.66
Gemma (comp.)	**0.260**	**0.620**	**0.740**	**0.860**	**0.840**	**0.820**
Gemma (res.)	**8.26**	**9.24**	**9.48**	**9.82**	**9.82**	**9.72**

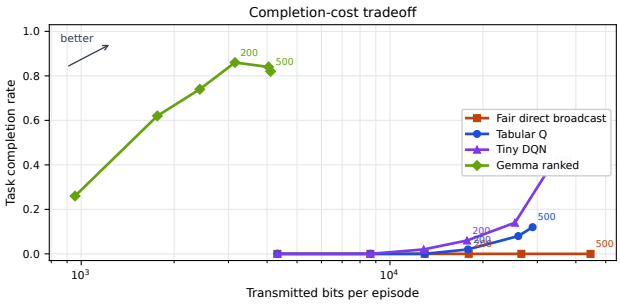

Figure 3. Completion-cost tradeoff across all evaluated bandwidths. Each marker is one strategy at one bandwidth budget, with 200-bit and 500-bit points annotated. Better protocols move toward the upper-left: higher completion with fewer transmitted bits.

86% completion, compared with 6% for tiny DQN, 2% for tabular Q, and 0% for direct broadcast. Using Wilson intervals over 50 seeds, Gemma's 200-bit completion rate has an approximate 95% confidence interval of [0.738, 0.930], while tiny DQN's best observed completion rate at 500 bits has interval [0.259, 0.518]. The improvement is therefore not confined to task-completion thresholding: Gemma also collects at least 3.00 more resources than tiny DQN at 200 bits per step.

The learned baselines improve as bandwidth increases, with tiny DQN outperforming tabular Q in this benchmark. At 500 bits per step, tiny DQN reaches 38% completion and collects 8.66 resources on average. However, Gemma remains ahead at 82% completion and 9.72 resources. Direct broadcast remains weak across the evaluated range because fair bandwidth allocation does not solve the core coordination problem: agents still spend bandwidth on low-level information that is not globally prioritized.

Figure 3 summarizes the core efficiency result: Gemma occupies the high-completion, low-communication region of the tradeoff space, whereas direct broadcast uses substantially more transmitted bits without solving the coordination task. Figure 4 and Table 2 further show that Gemma's advantage is not obtained by using more transmitted bandwidth. At 200 bits per step, Gemma transmits 3148.0 bits per episode on average, far below direct broadcast and both

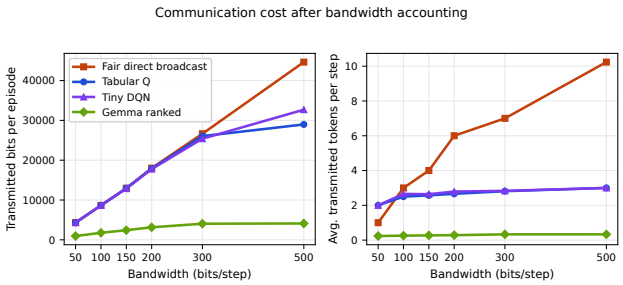

Figure 4. Communication cost after bandwidth accounting. Gemma transmits far fewer bits and tokens because the bandwidth-limited channel carries compact target assignments rather than verbose local observations or raw model responses.

Table 2. Representative communication metrics at 200 and 500 bits per step (50-seed average).

Strategy	B/step	Completion	Tok/step	Bits/ep	Res.
Direct	200	0.000	6.000	18000.0	2.00
Tab-Q	200	0.020	2.662	17892.0	5.54
Tiny-DQN	200	0.060	2.791	17768.0	6.82
Gemma	200	**0.860**	**0.285**	**3148.0**	**9.82**
Direct	500	0.000	10.236	44640.0	2.20
Tab-Q	500	0.120	3.000	28978.9	6.58
Tiny-DQN	500	0.380	3.000	32682.6	8.66
Gemma	500	**0.820**	**0.331**	**4105.6**	**9.72**

Table 3. Gemma inference diagnostics across the formal evaluation.

Metric	Value
Episodes	300
LLM calls	4967
LLM call success rate	1.000
Invalid target ids	8
Duplicate preferences	0
Empty with candidates	495
Conflicts resolved	5382

learned protocols; at 500 bits, it transmits 4105.6 bits. This occurs because Gemma emits compact target assignments rather than verbose state descriptions. The result is consistent with the broader observation that adaptive compression and selective communication are important for distributed systems under constrained bandwidth [6, 9].

5. Discussion

The main result is that an LLM-based coordinator, embedded in a ranked-target coordination pipeline, can act as a compact coordinator in a constrained multi-agent environment. The effect is strongest at low and moderate bandwidths: Gemma rises from 26% completion at 50 bits per step to 86% at 200 bits per step. Beyond 200 bits per step, completion enters a plateau: 84% at 300 bits and 82% at 500 bits. Because these differences are small relative to the 50-seed uncertainty, we do not interpret the 200, 300, and 500-bit settings as a strict ranking. Instead, the current benchmark suggests that the LLM protocol quickly reaches a regime where planning quality and environment stochasticity become more important than message capacity.

Table 3 provides additional evidence that the result is not affected by model-call failures. All 4967 model calls completed successfully. The parser observed only eight invalid target identifiers and no duplicate preferences across the entire Gemma run. The large number of resolved conflicts indicates that agents often prefer overlapping resources, and that deterministic conflict resolution is an important part of converting ranked LLM output into executable multi-agent assignments.

The current study has limitations. The environment is a controlled Grid World rather than a physical deployment. The learned baselines are lightweight and stronger multi-agent reinforcement learning baselines may narrow the gap. The prompt and ranked-target interface are benchmark-specific. Future work should test additional model scales, stronger learned protocols, decentralized variants, and richer partially observable tasks. The convergence of LLMs and robotic systems remains a promising direction for human-robot interaction and task execution [11], but careful benchmarking is needed before extrapolating from Grid World to real-world deployment.

6. Conclusion

This paper evaluates LLM-coordinated multi-agent planning under explicit communication budgets in a controlled Grid World benchmark. A Gemma-based ranked coordination protocol outperforms fair direct broadcast, tabular Q, and tiny DQN baselines across six low-bandwidth settings and 50 seeds. The strongest practical result occurs around 200 bits per step, where Gemma reaches 86% task completion with compact transmitted assignments. These findings support the hypothesis that LLMs can serve as useful high-level coordinators under bandwidth constraints, not by sending more information, but by selecting and compressing the information that matters most.

These results are benchmark evidence rather than a final real-world claim. The next stage of this work is to strengthen the paper with additional model comparisons, harder environments, and stronger learned-protocol baselines.

References

[1] Ian F. Akyildiz, Weilian Su, Yogesh Sankarasubramaniam, and Erdal Cayirci. Wireless sensor networks: a survey. *Computer Networks*, 38(4):393–422, 2002. 2

[2] Luigi Atzori, Antonio Iera, and Giacomo Morabito. The internet of things: A survey. *Computer Networks*, 54(15): 2787–2805, 2010. 2

[3] Cüneyt Bayılmış, M. Ali Ebleme, Ünal Çavuşoğlu, Kerem Küçük, and Abdullah Sevin. A survey on communication protocols and performance evaluations for internet of things. *Digital Communications and Networks*, 8(6):1094–1104, 2022. 2

[4] Abhishek Das, Théophile Gervet, Joshua Romoff, Dhruv Batra, Devi Parikh, Mike Rabbat, and Joelle Pineau. Tarmac: Targeted multi-agent communication. In *Proceedings of the 36th International Conference on Machine Learning*, pages 1538–1546. Proceedings of Machine Learning Research, 2019. arXiv:1810.11187. 1

[5] Jakob Foerster, Ioannis Alexandros Assael, Nando de Freitas, and Shimon Whiteson. Learning to communicate with deep multi-agent reinforcement learning. In *Advances in Neural Information Processing Systems*, pages 2137–2145, 2016. 1

[6] Dailan He, Ziming Yang, Weikun Peng, Rui Ma, Hongwei Qin, and Yan Wang. Elic: Efficient learned image compression with unevenly grouped space-channel contextual adaptive coding. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5708–5717, 2022. 4

[7] Wendi Rabiner Heinzelman, Anantha Chandrakasan, and Hari Balakrishnan. Energy-efficient communication protocol for wireless microsensor networks. In *Proceedings of the 33rd Annual Hawaii International Conference on System Sciences*, 2000. 2

[8] Takeshi Kojima, Shixiang Shane Gu, Machel Reid, Yutaka Matsuo, and Yusuke Iwasawa. Large language models are zero-shot reasoners, 2022. 1

[9] Peng Luo, F. Richard Yu, Jianyong Chen, Jianqiang Li, and Victor C. M. Leung. A novel adaptive gradient compression scheme: Reducing the communication overhead for distributed deep learning in the internet of things. *IEEE Internet of Things Journal*, 8(14):11476–11486, 2021. 4

[10] Guangming Shi, Yong Xiao, Yingyu Li, and Xuemei Xie. From semantic communication to semantic-aware networking: Model, architecture, and open problems. *IEEE Communications Magazine*, 59(8):44–50, 2021. 1

[11] Ceng Zhang, Junxin Chen, Jiatong Li, Yanhong Peng, and Zebing Mao. Large language models for human–robot interaction: A review. *Biomimetic Intelligence and Robotics*, 3 (4):100131, 2023. 1, 4

[12] Lei Zou, Zidong Wang, Jun Hu, Yurong Liu, and Xiaohui Liu. Communication-protocol-based analysis and synthesis of networked systems: Progress, prospects and challenges. *International Journal of Systems Science*, 52(14): 3013–3034, 2021. 3