# Everyone Contributes! Incentivizing Strategic Cooperation in Multi-LLM Systems via Sequential Public Goods Games

Yunhao Liang
The University of Hong Kong
Hong Kong, China
yunhao8@hku.hk

Yuan Qu
The University of Hong Kong
Hong Kong, China
yuanqu@hku.hk

Jingyuan Yang
George Mason University
Fairfax, VA, USA
jyang53@gmu.edu

Shaochong Lin
The University of Hong Kong
Hong Kong, China
shaoclin@hku.hk

Zuo-Jun Max Shen
The University of Hong Kong
Hong Kong, China
maxshen@hku.hk

## ABSTRACT

Coordinating multiple large language models (LLMs) to solve complex tasks collaboratively poses a fundamental trade-off between computational costs and collective performance when compared with individual models. We introduce a novel, game-theoretically grounded reinforcement learning (RL) framework, the Multi-Agent Cooperation Sequential Public Goods Game (MAC-SPGG), to systematically incentivize cooperation in multi-LLM ensembles. In MAC-SPGG, LLM agents move in sequence, observing predecessors' outputs and updating beliefs to condition their own contributions. By redesigning the public-goods reward, effortful contributions become the unique Subgame Perfect Nash Equilibrium (SPNE), which eliminates free-riding under traditional SPGG or PGG. Its sequential protocol replaces costly round-based information exchanges with a streamlined decision flow, cutting communication overhead while retaining strategic depth. We prove the existence and uniqueness of the SPNE under realistic parameters and empirically demonstrate that MAC-SPGG-trained ensembles outperform single-agent baselines, chain-of-thought prompting, and other cooperative methods, achieving comparable performance to large-scale models across various tasks, including reasoning, math, code generation, and NLP. Our results highlight the power of structured, incentive-aligned MAC-SPGG cooperation for scalable and robust multi-agent language generation. Appendix and code can be found at https://github.com/YunhaoLiang/MAC-SPGG.

## KEYWORDS

Multi-LLM Collaboration; Public Goods Games; Reinforcement Learning

**ACM Reference Format:**
Yunhao Liang, Yuan Qu, Jingyuan Yang, Shaochong Lin, and Zuo-Jun Max Shen. 2026. Everyone Contributes! Incentivizing Strategic Cooperation in Multi-LLM Systems via Sequential Public Goods Games. In *Proc. of the 25th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2026), Paphos, Cyprus, May 25 – 29, 2026*, IFAAMAS, 19 pages. https://doi.org/10.65109/SJPO2377



*Proc. of the 25th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2026), C. Amato, L. Dennis, V. Mascardi, J. Thangarajah (eds.), May 25 – 29, 2026, Paphos, Cyprus.* © 2026 International Foundation for Autonomous Agents and Multiagent Systems (www.ifaamas.org). https://doi.org/10.65109/SJPO2377

## 1 INTRODUCTION

Recent advancements in large language models (LLMs) have demonstrated impressive capabilities across various reasoning and decision-making tasks, especially within multi-agent scenarios. Emerging research [59] explores diverse interaction paradigms among multiple LLMs, from competitive debating and strategic reasoning [14, 19, 31, 41, 67] to cooperative decision-making and collaborative problem-solving [9, 33, 38, 39, 49, 65]. Multi-LLM ensembles are promising because they combine complementary reasoning strategies, diversify knowledge sources, and enhance robustness and accuracy compared to single-model systems.

However, achieving these benefits crucially depends on effectively coordinating the ensemble, especially from the perspective of information sharing. Existing frameworks predominantly rely on two communication strategies: simultaneous and sequential. In the simultaneous setting, LLMs act independently and concurrently, requiring a central coordinator to aggregate outputs. This single-point bottleneck raises communication costs and limits dynamic, information-driven interaction within the ensemble [30, 68]. Conversely, sequential communication enables information sharing among agents, allowing each model to condition its action on preceding outputs. However, without careful strategic design, unrestricted sequential information exchange accumulated among all agents can lead to significant communication overhead and computational complexity [8, 40, 42].

Hence, a critical challenge arises: *how can we achieve high-performance multi-LLM ensembles while reducing communication and computational overhead?* Inspired by game theory, where all the players contribute rationally with only a common knowledge of the game rules, we adopt the idea of Public Goods Game (PGG) for multi-LLM ensemble learning. Here, our goal is to introduce a *public-goods-inspired incentive mechanism* tailored to multi-LLM coordination, rather than to resolve the classical public goods problem in its general economic sense. PGG is a canonical paradigm extensively examined in economics and behavioral sciences [4, 25], which characterizes scenarios where individuals contribute to a collective good, balancing private costs against shared public benefits. Prominent real-world examples include crowdfunding platforms [5], open-source collaborations [26, 58], and public infrastructure funded by taxation [17].

Building upon this paradigm, we propose the two-phase game-theoretical reinforcement learning (RL) framework, *Multi-Agent*

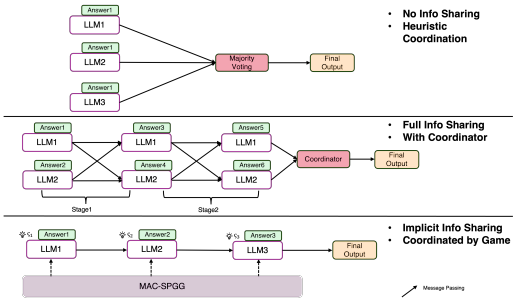


**Figure 1: Comparison of coordination mechanisms across LLM-based multi-agent systems.**

*Cooperation Sequential Public Goods Game (MAC-SPGG)*, as a theoretical foundation for systematically coordinating multi-LLM ensembles. While SPGG is established in the game-theory literature [4, 4, 27], its implications for LLM ensembles remain underexplored. Our MAC-SPGG explicitly models sequential decision-making, where agents observe their predecessors' contributions before acting—a scenario that naturally aligns with multi-LLM frameworks, such as cascading prompting [69] and iterative refinement [10]. Different from existing coordinator-based multi-LLM ensembles [19, 39], which typically vary in their approaches to dynamically sharing information during simultaneous agent contributions but still require a central coordinator to aggregate and evaluate the final outputs, our MAC-SPGG enables each model to evaluate prior contributions sequentially. See Figure 1. By incentivizing the sequential coordination process, MAC-SPGG eliminates the need for a central coordinator, substantially reduces associated costs, and enhances collaboration among agents. In this paper, we formally develop and validate the effectiveness of our SPGG-based multi-LLM coordination mechanism.

In our framework, we prove that a unique *Subgame Perfect Nash Equilibrium (SPNE)* can be found under reasonable conditions in the inference phase, the SPGG part. Each agent conditions its strategy on observed actions, revising its beliefs to maximize expected utility. By adjusting the incentives in traditional PGG, the equilibrium shifts from free-riding to positively cooperative participation. Our theoretically guaranteed equilibrium behaviors are largely absent from existing debate-, voting-, or heuristic-based coordination methods [11, 13, 19, 39]. Empirically, our sequential protocol can reduce communication overhead relative to some other baseline methods, as shown by token usage comparisons in our experiments.

Theoretically, to address the lack of theoretical grounding in current multi-agent LLM systems, we introduce a game-theoretic framework based on the SPGG. Prior coordination methods—such as debate, voting, or heuristic prompting—often exhibit promising empirical performance but remain theoretically underexplored, offering no formal guarantees of equilibrium behavior [11, 13, 19, 39]. In contrast, the SPGG-based coordination mechanism enables us to rigorously analyze the strategic dynamics of multi-agent collaboration in LLM-based systems. Specifically, we prove the existence and uniqueness of an SPNE under reasonable conditions, ensuring that each agent contributes meaningfully and that the group collectively achieves the desired outcome without central coordination. We also provide comparative statics and numerical verification, confirming the Pareto Optimality.

In experiments, MAC-SPGG robustly directs multi-LLM ensembles toward cooperative equilibria, consistently outperforming single-agent baselines, Chain-of-Thought (CoT) prompting [61], and other cooperative frameworks across four diverse tasks, including code generation (HumanEval), factual knowledge (MMLU), mathematical reasoning (GSM8K), and natural language understanding (SummEval). We evaluate our framework under two information-sharing regimes, *Partial Observation (PO)* and *Full Observation (FO)*, to explore how inter-agent information transparency affects performance.

Our key contributions are summarized as follows:

- We propose a theoretically grounded MAC-SPGG framework for structured multi-LLM cooperation. The existence and uniqueness of the SPNE provide theoretical foundations for equilibrium-driven cooperation.
- We empirically test MAC-SPGG across varied tasks and ablation tests, which consistently outperforms other single-agent and cooperative benchmarks. We find some strategic insights for future multi-agent communication protocols, where optimal information sharing is context-dependent, and minimal transparency may yield superior outcomes.

## 2 RELATED WORK

Our work synthesizes insights from multi-agent collaboration and mechanism design in LLM systems.

**Multi-Agent Collaboration with LLMs.** Recent research extensively explores frameworks enabling effective collaboration among multiple LLM agents, aimed at addressing complex cognitive and decision-making tasks [22, 38, 70]. A prominent paradigm involves mimicking human collaborative dynamics through explicit "role-playing" mechanisms, where LLM agents are assigned specialized functions corresponding to organizational roles [33], while Chen et al. [13] explore multi-agent collaboration via prompting-based interactions. Alternative frameworks further enrich multi-agent collaboration through voting and consensus mechanisms [39, 48, 60], collective reasoning or discussion-based methodologies [9], and structured agentic debate approaches [19, 41], aiming at enhancing factual accuracy and logical consistency.

Prevalent multi-LLM collaboration frameworks lack theoretical grounding and offer no guarantees of convergence, stability, or cooperation. Our MAC-SPGG framework introduces PGG-inspired incentives to enable collaboration via utility-aligned rewards and structured inter-agent reasoning.

**Mechanism Design and Game Theory in LLMs.** Integrating mechanism design and game-theoretic insights into multi-agent LLM systems is increasingly investigated.

LLM's rationality has been primarily tested. Mao et al. [44] rigorously evaluated LLM strategic behaviors across game-theoretic scenarios, while Pan et al. [47] showed that Bayesian reasoning frameworks encourage cooperative strategies in repeated games among LLM agents, demonstrating cooperative behaviors under suitable incentives in structured games like Public Goods Games (PGGs) [55]. Recent work further introduces structured game-theoretic workflows to improve LLMs' strategic rationality in both complete- and

incomplete-information games [34]. Some empirical studies also indicate LLMs exhibit rational behaviors in strategic settings, emphasizing historical context in shaping interactions [1, 6, 24, 43].

While prior work utilizes game-theoretic tasks to evaluate LLM rationality, the integration of game theory and LLM research has not been thoroughly investigated. Recent studies have developed tailored incentive mechanisms, such as token auctions, promoting collaboration among agents [21]. Cheng et al. [14] embedded games to enhance the intrinsic reasoning capabilities of LLMs, demonstrating significant performance improvements across various reasoning benchmarks. Methods like multi-stakeholder alignment significantly enhance LLM output alignment in value-conflict environments [54].

We propose a new multi-agent collaboration framework grounded in the strategic structure of the SPGG, which demonstrates strong empirical and theoretical effectiveness across diverse tasks.

## 3 METHOD

In this section, we first introduce the fundamental formulation of our MAC-SPGG design, the inference phase in our framework. We then propose the crucial reward structure, followed by the theoretical guarantee of MAC-SPGG. Lastly, we describe the MAC-SPGG learning framework and its optimization phase, as shown in Figure 2. The training process is concluded in Algorithm 1, and a comprehensive notation table is summarized in Appendix A.

### 3.1 MAC-SPGG Formulation

To model multi-agent collaboration among $n$ LLM agents performing a shared textual task $q$, we assume that the cooperation process follows a finite-horizon, sequential, and decentralized setting. Each agent $i$ sequentially provides exactly one *contribution* $\tau_i$ toward the final collective outcome,

$$\tau_i = T_i(h_i, q). \tag{1}$$

Here, the function $T_i$ represents the LLM base model of agent $i$, while $h_i$ represents the historical information that is observable to agent $i$. We name the observable history and task information $(h_i, q)$ as *local knowledge*, where all participants make their own contributions based on it.

For the history $h_i$, we have two modes of observations under the MAC-SPGG framework: (1) **Partial Observation (PO):** The agent $i$ can observe only the contribution from the immediately preceding agent (if any), $h_i^{PO} = \{\tau_{i-1}\}$, and (2) **Full Observation (FO):** The agent $i$ can observe all contributions made by previous agents, $h_i^{FO} = \{\tau_1, \tau_2, \ldots, \tau_{i-1}\}$.

In the PO schema, agent $i$ only observes the immediate predecessor's contribution $\tau_{i-1}$, following the SPGG [4, 28] setting, which is similar to the sense of Markov decision process. In contrast, the agents under the FO regime have full access to the complete history of prior contributions. Both types of observation settings exist in multi-agent LLM studies [19, 62]. Although the coordinator-free mechanism of MAC-SPGG saves computation resources, the FO mode would consume more tokens than the PO mode. Such a difference in information availability and resource usage will lead to distinct comprehensibility in various types of tasks in our experiment.[1]

After all agents have committed their contribution, the contribution $\tau_i$ of each agent will be evaluated by a task-specific metric, the *score*, $c_i(\tau_i, q)$, and a model-related metric, the *cost*, $\ell_i(\tau_i, q, T_i)$. The score indicates the performance of the contribution, which is evaluated by a given task-specific function $\mathcal{E}$, $c_i = \mathcal{E}(\tau_i, q)$. For instance, in multiple-choice tasks, the score represents the accuracy of the test; in more complex tasks, such as a generation task, the score is evaluated by a fine-tuned evaluator; see training details in Appendix D. We denote the score by $c_i(\tau_i, q)$ to show its relevance to $\tau_i$ and $q$. For the cost part, under the usage of LLM, the number of consumed tokens would be a straightforward measure of cost, and different base models $T$ will lead to various levels of token usage.

We denote the final task score used to assess success by $C(\vec{\tau}, q)$, defined as the last agent's contribution:

$$C(\vec{\tau}, q) = c_n(\tau_n, q). \tag{2}$$

The task $q$ succeeds if the final score $C(\vec{\tau}, q)$ surpasses a predefined threshold $B(q)$. Our objective is to maximize the final score $C$, rather than merely exceeding the threshold $B$ on task $q$, by efficiently utilizing LLM agents to collaborate on the shared task.

Remark 1 (Cumulative Effect). Although other agents' contribution is not on the surface of $c_n(\tau_n, q)$, we still denote the final score $C$ as a function of all the contributions $\vec{\tau}$ due to the cumulative effect of the MAC-SPGG. Unlike PGG, where the final performance is calculated by summing up all the contributions, the nature of multi-agent LLM tasks and prompting necessitates a summary step instead of concatenating the AI-generated content (AIGC) directly. In ECON [68] or other coordinator-based frameworks, a summary agent in the last step would absorb all the others' outputs and generate the final answer. In our MAC-SPGG framework, predecessors' outputs have already been embedded into the sequential process. For instance, if we are under the FO mode, where $c_n = T_n(h_n, q)$, $h_n$ contains all the previous $\tau_i$ information. If we are under the PO mode, we can regard the final score as

$$\begin{aligned} C(\vec{\tau}, q) =& c_n\left(T_n(\tau_{n-1}, q), q\right) \\ =& c_n\left(T_n\left(T_{n-1}(\tau_{n-2}, q), q\right), q\right) \quad \cdots \\ =& c_n\left(T_n\left(T_{n-1}(\cdots(T_1(q), q)\cdots, q), q\right)\right). \end{aligned}$$

In such a context, the impact of each contribution $\tau_i$ on the final score is not explicit, but in an iterative way.

### 3.2 Reward Structure and Equilibrium

Our reward design is rooted in the classic threshold public goods game (TPGG) [4, 28, 29, 37] rather than an ad-hoc construction. In a TPGG, each player $i$ privately chooses a contribution $\chi_i \geq 0$, and the public good is provided only if the total contributions $\sum_{j=1}^{n} \chi_j$ reach a provision point $B > 0$; otherwise, the project fails. A canonical expected utility can be written as

$$U_i = -\ell_i(\chi_i) + \frac{\rho}{n} \sum_j \chi_j - \mathbf{1}\left(\sum_{j=1}^{n} \chi_j < B\right) \cdot P,$$

[1] When $h_i = \emptyset$, agents act independently on $q$, reducing to a simultaneous-move PGG [3, 57]. This "no-observation" regime parallels ECON [68] but is incompatible with the sequential, coordinator-free MAC-SPGG.

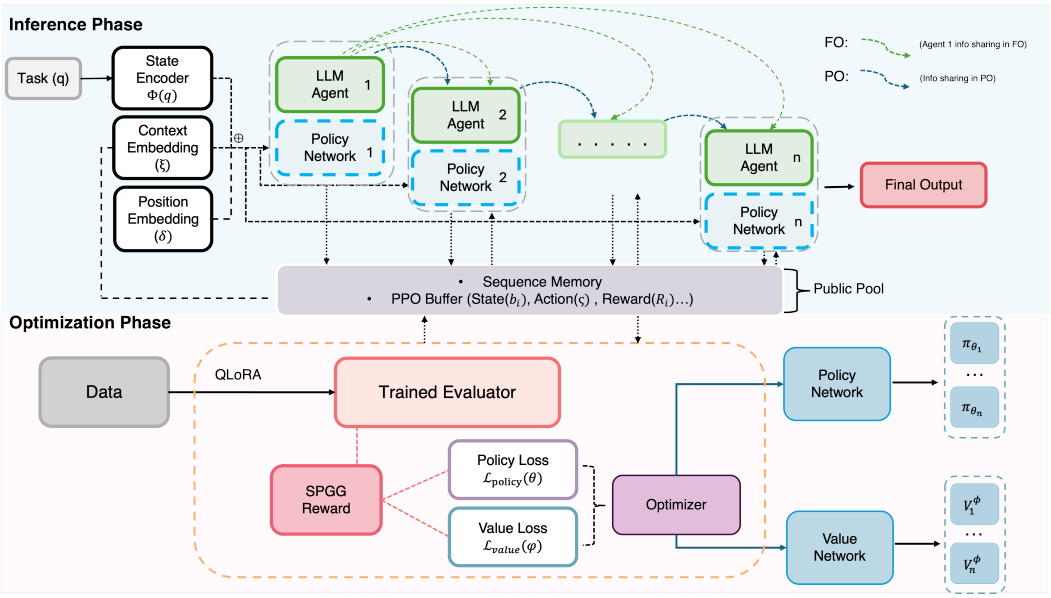


**Figure 2: MAC-SPGG Framework. Top: *The Inference Phase*, where LLM agents act in sequence, conditioned on (Partial/Full) observation regimes. Bottom: *The Optimization Phase*, where SPGG rewards drive PPO updates for policy and value networks.**

where $\ell_i(\chi_i)$ is the private cost of contribution, the second term represents the equal-sharing return from the successfully provided public good, and the last term captures the outcome when the threshold is not met (e.g., refund, no-refund, or penalty depending on the mechanism variant).

*Notation mapping.* To connect this formulation to our multi-LLM collaboration setting, we establish the following correspondence:

$$\chi_i \leftrightarrow \tau_i, \quad \sum_j \chi_j \leftrightarrow C(\vec{\tau}, q), \quad B \leftrightarrow B(q), \quad \ell_i(\chi_i) \leftrightarrow \ell_i(\tau_i, q, T_i).$$

Thus, each agent's textual output $\tau_i$ is viewed as its effective contribution, and the evaluator's final score $C(\vec{\tau}, q)$ represents the collective provision level.

Since the classical public goods game admits a Nash equilibrium in which all players optimally choose zero contribution, we introduce a **cooperation-incentive term** to break this degenerate equilibrium and encourage positive participation. This addition preserves the threshold structure of the original TPGG while explicitly promoting cooperative behavior among sequential agents.

Definition 1 (Reward with Cooperation Incentive). *The reward for agent $i \in \{1, \ldots, n\}$ in the MAC-SPGG is defined as a threshold-PGG payoff augmented by an explicit cooperation incentive:*

$$R_i = \underbrace{\left[ -\ell_i(\tau_i, q, T_i) + \frac{\rho}{n} C(\vec{\tau}, q) - P\,\mathbf{1}(C(\vec{\tau}, q) < B(q)) \right]}_{\text{classical threshold-PGG payoff}} + \underbrace{\gamma_c \frac{c_i(\tau_i, q)}{B(q)} C(\vec{\tau}, q)}_{\text{cooperation incentive}}.$$

Here, $B(q)$ serves as the provision point: the final score $C(\vec{\tau}, q)$ must exceed $B(q)$ for the shared reward to materialize. The $\gamma_c$-weighted cooperation term extends the standard group return by linking an agent's own contribution $c_i$ to the realized group performance $C(\vec{\tau}, q)$, thus aligning individual incentives with collective success. Consequently, the MAC-SPGG reward remains a threshold-PGG structure augmented by an explicit cooperation incentive, rather than a redesigned objective.

Assumption 1 (Score Assumption). *For each agent $i \in \{1, \ldots, n\}$, the individual score $c_i$ is positive, bounded, and finite:*

$$c_i \in [c_{\min}, c_{\max}], \qquad \text{where} \quad 0 < c_{\min} \le c_{\max} < \infty.$$

*The upper bound $c_{\max}$ is defined as*

$$c_{\max} \equiv \sup_{\vec{\tau}} \left\{ c_n\big(\tau_n(\tau_{n-1}(\cdots(\tau_1(q), q)\cdots), q), q\big) \right\},$$

*and may exceed the task-specific threshold, i.e., $c_{\max} \ge B(q)$.*

This assumption reflects an empirical property of large language models (LLMs): when prompted with a question, they do not produce null responses. Hence, each agent contributes at least a minimal amount to the collective output, represented by the positive lower bound $c_{\min}$.

Assumption 2 (Cost Assumption). *For each agent $i \in \{1, \ldots, n\}$, the individual cost of producing an output with quality level $c_i$ is denoted by $\ell_i(c_i)$. The cost function is strictly convex and twice continuously differentiable over the feasible range $[c_{\min}, c_{\max}]$, representing increasing marginal effort as output quality improves.*

*Formally,*

$$\ell_i''(c_i) > 0 \quad \text{and} \quad \ell_i'(c_i) > 0, \qquad \forall\, c_i \in [c_{\min}, c_{\max}].$$

The function $\ell_i(\cdot)$ captures the computational and cognitive resources (explicit inference cost, attention span, or token usage)

required for the agent's model $T_i$ to generate higher-quality contributions, consistent with empirical scaling observations in large language models [15, 35].

The assumptions underlying our framework are not arbitrary, but rather grounded in a long line of research on public goods games and mechanism design, which involves payments or taxation. These mechanisms aim to internalize externalities, either positive or negative, so that individually rational agents collectively achieve socially efficient outcomes, which are historically developed in economics and network routing theory [4, 29, 50, 51].

In the context of network routing, for instance, taxation mechanisms penalize overuse of congested links, thereby steering self-interested agents toward equilibria that align with global efficiency. Analogously, public-goods mechanisms reward contributions that benefit others, ensuring cooperation is individually rational.

Building on these classical foundations, we extend the incentive-alignment logic into the LLM era.

Large language models (LLMs) differ from human or classical agents in several structural aspects. Given the properties of LLM, they cannot produce a null response when prompted with a question; in other words, if we abstract a prompt as a task, there always exists a minimal level of effort or cost, denoted as $c_{min}$, which corresponds directly to Assumption 1.

The computational cost of large language models is typically strictly convex, as the inference complexity of Transformers scales quadratically with respect to the context length, i.e., $O(n^2 \cdot d)$, where $n$ denotes the number of tokens and $d$ the hidden dimension. Consequently, the overall cost of LLM inference exhibits a strictly convex relationship with token usage, consistent with Assumption 2.

Although LLMs are not perfectly rational agents in the classical economic sense, prior studies have demonstrated that they can exhibit quasi-rational behavior under well-structured environments and incentive signals [34, 55]. Consequently, the adherence of LLMs to the proposed reward structure in our framework is enforced through a combination of prompt engineering and reward-based training. These mechanisms effectively guide model behavior toward equilibrium-consistent strategies, thereby supporting the practical validity of the proposed framework.

Therefore, our MAC-SPGG framework is not a departure from classical public-goods reasoning, but rather a modern instantiation of it—recasting established cooperative mechanisms within the computational substrate of large language models.

Given the mathematical support from Assumptions 1 and 2, we have

Theorem 1 (Equilibrium). *Under a reasonable cooperation coefficient $\gamma_c$ and failure penalty $P$, where*

$$\rho > n \cdot \max_i \ell_i'(c_{\max}),$$

$$\gamma_c > \max_{k=2,\dots,n} \frac{\ell_k'(c_{\max}) \cdot B(q) - \rho/n}{c_{\min}/B(q)}, \text{ and}$$

$$P > \left( \max_i \left\{ \ell_i'(c_{\max}) \right\} + \gamma_c \frac{c_{\max}}{B(q)} + \frac{\rho}{n} \right) \cdot (c_{\max} - c_{\min}),$$

*there exists a joint strategy $\mathbf{c}_i^* = (c_1^*, \dots, c_n^*)$ that constitutes a **unique** Subgame Perfect Nash Equilibrium (SPNE),*

$$\mathbf{c}_i^* \in \arg\max_{\vec{c}} \{ \textit{SPNE under } R_i \},$$

*where every agent $i \in \{1, \dots, n\}$ contributes positively, $c_i^* > 0$, and the overall task would succeed $C(\vec{\tau}, q) \geq B(q)$.*

Theorem 1 demonstrates the existence and uniqueness of the SPNE under our MAC-SPGG framework, which ensures the rationality of LLM agents. Under SPNE, each agent contributes positively to cooperation, thereby operating within the successful provision region $C(\vec{\tau}, q) \geq B(q)$.

To examine the sensitivity of equilibrium behavior and welfare outcomes to incentive parameters in our theoretical MAC-SPGG framework, we conduct a comparative statics analysis. via the *envelope theorem* [7].

Theorem 2 (Comparative Statics of Welfare). *Let the total welfare under the MAC-SPGG equilibrium be defined as*

$$W(\gamma_c, \rho, B) = \sum_{i=1}^{n} R_i(c_i^*; \gamma_c, \rho, B),$$

*where we denote $c_i(\tau_i, q)$ as the effective contribution level and $c_i^*$ as its equilibrium realization, and $R_i$ is as defined in Definition 1. Then, under any equilibrium with $c_i^* \geq 0$ and $C(\vec{\tau}, q) \geq B(q)$, the following comparative-statics relationships hold:*

$$\frac{\partial W}{\partial \gamma_c} > 0, \qquad \frac{\partial W}{\partial \rho} > 0, \qquad \frac{\partial W}{\partial B} < 0.$$

These monotonic results indicate that stronger cooperation incentives ($\gamma_c$) and larger public-good sharing rates ($\rho$) jointly enhance system-level welfare, while a higher task threshold ($B$) increases collective difficulty and reduces achievable welfare. Detailed derivations of Theorems 1 and 2 are provided in Appendix B, with numerical verification in Appendix C.

### 3.3 RL as a Meta-Control Framework

To operationalize the theoretical SPGG formulation, we instantiate each agent's generation function $\mathcal{G}_i$ through a two-phase process illustrated in Figure 2. At the Inference Phase, a foundational language model $T_i$ produces textual outputs guided by the MAC-SPGG mechanism. At the Optimization Phase, a reinforcement learning (RL) based meta-policy $\pi_{\theta_i}$ is trained to synthesize high-level strategic configurations from belief representations, enabling adaptive and coordinated behaviors among agents. Each agent employs an independent Proximal Policy Optimization (PPO) learner [53] under the synergy-aligned reward function defined in Definition 1, translating the SPGG's theoretical payoff formulation into actionable feedback for RL.

The generation process for agent $i$ is cast as a hierarchical control problem given a prompt of the task $q$. First, the agent constructs an enhanced belief state vector $b_i = [\Phi(q); \xi_i; \delta_i]$ by concatenating a task embedding $\Phi(q)$, context features $\xi_i$ containing historical performance and environmental information, and a position embedding $\delta_i$. This belief $b_i$ informs the agent's meta-policy $\pi_{\theta_i}$, which generates a generative configuration vector, $\vec{\zeta}_i \sim \pi_{\theta_i}(\cdot|b_i)$. As a result, this vector serves as a local policy to direct the global collaboration. Finally, the LLM produces the agent's contribution $\tau_i$ as Eq. (1) under this strategic guidance $T_i(q, h_i|\vec{\zeta}_i)$.

We train each agent's meta-policy $\pi_{\theta_i}$ using a decentralized actor–critic framework based on PPO. The synergy-aligned reward $R_i$

**Algorithm 1** MAC-SPGG Framework

**Require:** Initial prompt $q$; base models $\{T_i\}_{i=1}^n$; evaluator $\mathcal{E}$; game parameters $\rho, \gamma, P, B(q)$; max episodes $T_{\max}$; early stopping thresholds $R_{\text{th}}, C_{\text{target}}, \epsilon$

**Ensure:** Optimized policy and value function parameters $\{\theta_i^*, \phi_i^*\}_{i=1}^n$

Initialize $\{\theta_i, \phi_i\}_{i=1}^n$, encoder $\theta_\Phi$, buffer $\mathcal{D}$, and history $\mathcal{H}$
**for** episode $t = 1$ to $T_{\max}$ **do**
  Reset $\mathcal{D} \leftarrow \emptyset, \mathcal{H} \leftarrow \emptyset$
  **for** agent $i = 1$ to $n$ **do** ▷ Sequential rollout
    Extract task embedding $\Phi(q)$, context features $\xi_i$ and position embedding $\delta_i$
    Construct $b_i \leftarrow [\Phi(q); \xi_i; \delta_i]$
    Sample configuration $\vec{\zeta}_i \sim \pi_{\theta_i}(\cdot \mid b_i)$
    Generate output $\tau_i \leftarrow T_i(q, h_i \mid \vec{\zeta}_i)$
    Store $(b_i, \vec{\zeta}_i, \tau_i)$ in $\mathcal{D}$, update $\mathcal{H} \leftarrow \mathcal{H} \oplus \tau_i$
  **end for**
  **for** agent $i = 1$ to $n$ **do** ▷ Reward computation
    Evaluate quality $c_i \leftarrow \mathcal{E}(\tau_i, q)$
    Compute reward $R_i$, advantage $A_i = R_i - V^{\phi_i}(b_i)$
    Store $(R_i, A_i)$ in $\mathcal{D}$
  **end for**
  **Final integration:** obtain overall output $\tau_n$ and evaluate final score

$$C_t \leftarrow \mathcal{E}_{\text{final}}(\tau_n, q), \quad R_t^{\text{final}} \leftarrow R_n$$

  **for** agent $i = 1$ to $n$ **do** ▷ PPO update
    Update $\theta_i$, $\phi_i$ via gradient descent on $-\mathcal{L}_{\text{PPO}}(\theta_i)$
  **end for**
  **if** $R_t^{\text{final}} \geq R_{\text{th}}$ and $C_t \geq C_{\text{target}}$ and $|R_t^{\text{final}} - R_{t-1}^{\text{final}}| \leq \epsilon$ and $|C_t - C_{t-1}| \leq \epsilon$ **then**
    **break** ▷ Early stopping based on final integrator's score
  **end if**
**end for**
**return** $\{\theta_i^*, \phi_i^*\}_{i=1}^n$

defined in Definition 1 serves as the optimization target, linking the SPGG's game-theoretic payoff structure to reinforcement signals.

In each episode, agent $i$ observes its belief state $b_i$, samples a continuous configuration vector $\vec{\zeta}_i$, and generates a textual contribution through its base LLM. The policy network is updated using the immediately observable part of $R_i$.

The value function over belief states is defined as:

$$V_i^\phi(b_i) = \mathbb{E}_{\pi_{\theta_i}}[R_i | b_i], \tag{3}$$

where $R_i$ is the total episodic reward in Definition 1. We estimate the advantage via the standard Generalized Advantage Estimation (GAE) [52]. Since the MAC-SPGG environment is a one-step episodic setting, the next-state term $V_i^\phi(b_{i+1})$ corresponds to the terminal value and vanishes ($V_i^\phi(b_{i+1}) = 0$), thus GAE naturally reduces to the one-step advantage:

$$A(b_i, \vec{\zeta}_i) = R_i - V_i^\phi(b_i). \tag{4}$$

where $V_i^\phi(b_{i+1})$ is the terminal value, typically set to zero under the one-step assumption.

Hence, each agent's PPO objective is defined in the standard maximization form for clarity; in practice, gradient descent is performed on its negated loss during implementation. Formally,

$$\mathcal{L}_{\text{PPO}}(\theta_i) = \mathbb{E}_{b_i, \vec{\zeta}_i}\Big[\min\big(R(\theta_i) \cdot A(b_i, \vec{\zeta}_i),\ \text{clip}_\varepsilon(R(\theta_i)) \cdot A(b_i, \vec{\zeta}_i)\big)\Big] - \lambda_{\text{value}} \cdot \big(V_i^\phi(b_i) - R_i\big)^2, \tag{5}$$

where the importance-sampling ratio is defined as

$$R(\theta_i) = \frac{\pi_{\theta_i}(\vec{\zeta}_i \mid b_i)}{\pi_{\theta_{\text{old}},i}(\vec{\zeta}_i \mid b_i)}. \tag{6}$$

Here, $R(\theta_i)$ represents the ratio between current and previous policies, $A(b_i, \vec{\zeta}_i)$ is the estimated advantage, and $V_i^\phi(b_i)$ denotes the learned value function. The coefficient $\lambda_{\text{value}}$ controls the contribution of the value loss in the overall objective.

To ensure efficient optimization and convergence, we adopt an early stopping mechanism aligned with the theoretical success criterion of the MAC-SPGG framework. Specifically, training is terminated once two external conditions are jointly satisfied. First, the final integrator's reward $R_n$ exceeds a predefined threshold, $R_n \geq R_{\text{th}}$. Second, the evaluator-assessed collective score of the final output, $C_t = \mathcal{E}_{\text{final}}(\tau_n, q)$, meets or surpasses a target value, $C_t \geq C_{\text{target}}$. These criteria ensure that early stopping depends on the overall task success rather than intermediate averages, in accordance with the theoretical formulation in Definition 1. To guarantee convergence stability, we further require both the final reward and the collective score to remain within a small tolerance across consecutive episodes, $|R_t^{\text{final}} - R_{t-1}^{\text{final}}| \leq \epsilon$ and $|C_t - C_{t-1}| \leq \epsilon$. This mechanism halts training only after the cooperative system achieves sustained task-level improvement and stabilizes in both reward and quality.

## 4 EXPERIMENT

This section outlines the experimental setup, reports effectiveness performance comparisons with various benchmarks, sequential ordering effect analysis, and presents ablation studies on base model combinations and heterogeneity.

### 4.1 Datasets

We evaluate our workflow on four standard benchmarks spanning diverse capabilities: *HumanEval* [12] for code generation (Python tasks with unit-test evaluation), *MMLU* [32] for general knowledge and reasoning (57 subjects across STEM and humanities), *GSM8K* [16] for multi-step arithmetic problem solving (grade-school math word problems), and *SummEval* [23] for natural language understanding (human-annotated summaries rated on coherence, consistency, fluency, and relevance). For *SummEval*, we train a reinforcement learning-based evaluator aligned with human-centric metrics; see Appendix D.1.

### 4.2 Comparison Methods

We compare MAC-SPGG against several widely adopted strong baselines: (1) ***Zero-shot*** *CoT prompting* [36]: Directly asks the model to reason step-by-step without any examples. (2) ***Few-shot*** *CoT*

| System Category | Configuration | #Params | HumanEval | MMLU | GSM8K | SummEval (Avg) |
|---|---|---|---|---|---|---|
| Zero-Shot COT Single-Agent | SmolLM2-1.7B-Instruct | 1.7 | 24.4 (-49.38) | 29 (-46) | 45 (-50) | 4.607 (-0.12) |
| | Llama3.1-8B-Instruct | 8 | 59.76 (-14.02) | 57 (-18) | 88 (-7) | 4.638 (-0.09) |
| | Qwen3-8B | 8 | 64.63 (-9.15) | 66 (-9) | 89 (-6) | 4.677 (-0.05) |
| Few-Shot COT Single-Agent | SmolLM2-1.7B | 1.7 | 29.9 (-43.88) | 41 (-34) | 52 (-43) | – |
| | Llama3.1-8B | 8 | 72.6 (-1.18) | 70 (-5) | 90 (-5) | – |
| | Qwen3-8B | 8 | 72.0 (-1.78) | 67 (-8) | 92 (-3) | – |
| Multi-Agent Baselines | Majority Voting | 17.7 | – | 71 (-4) | 84 (-11) | – |
| | Multi-Agent Debate | 17.7 | – | 66 (-9) | 86 (-9) | – |
| | CAMEL | 16 | 48.78 (-24.99) | 42 (-33) | 88 (-7) | – |
| | ECON | 25.7 | 70.73 (-3.05) | 64 (-11) | 89 (-6) | 4.590 (-0.14) |
| MAC-SPGG Framework (Ours) | MAC-SPGG (PO) | 17.7 | 67.07 (-6.71) | **75 (-)** | **95 (-)** | 4.449 (-0.28) |
| | MAC-SPGG (FO) | 17.7 | **73.78 (-)** | 69 (-6) | 93 (-2) | **4.728 (-)** |

***Note.*** **"-" indicates not applicable, e.g., voting-based methods cannot generate coherent outputs for HumanEval or SummEval.** ***Ordering used in both PO and FO settings:*** **Smol → LLaMA → Qwen.**

**Table 1: Performance on four benchmarks with delta (in parentheses) relative to the best MAC-SPGG setup. Metrics: HumanEval in Pass@1 (%), MMLU and GSM8K in accuracy (%), and SummEval in the averaged evaluator-predicted human score (0–5).**

*prompting* [61]: Provides a few worked-out examples to guide the model's step-by-step reasoning. (3) ***Majority Voting****-based multi-agent ensemble* [39]: Multiple independent agents generate answers in parallel, and the final output is selected via majority vote or other aggregation strategies. (4) ***Multi-Agent Debate****-style prompting* [19]: Agents engage in argumentation or critique each other's outputs before converging on a final decision. (5) ***CAMEL****-style role-based collaboration* [38]: Agents are assigned distinct roles (e.g., user, assistant, critic) to simulate structured dialogues. (6) ***ECON*** [68]: Agents act independently without observing each other, controlled and manipulated by one coordinator.

## 4.3 MAC-SPGG Setups

In our experiments, we instantiate the MAC-SPGG framework using three sequentially interacting language models. The training details are provided in Appendix D.2. We focus primarily on training and evaluating *small-scale language models* under the MAC-SPGG setting. As heterogeneous model integration has been shown to enhance multi-agent reasoning and strategic capabilities [48, 56], we specifically employ Qwen3-8B [63], SmolLM2-1.7B [2], and LLaMA 3.1-8B [20] to effectively exploit model heterogeneity.

## 4.4 Main Results

We show the performance of each method across four representative evaluation tasks spanning code generation (HumanEval), factual knowledge (MMLU), mathematical reasoning (GSM8K), and natural language understanding (SummEval) in Table 1. The MAC-SPGG, under both PO and FO regimes, consistently outperforms most single-agent and multi-agent baselines, particularly excelling on complex reasoning tasks such as GSM8K and MMLU. To provide reference points for upper-bound performance, we include GPT-3.5 Turbo [66], GPT-4-0613 [46], and Qwen2.5-72B-Instruct [63] in a zero-shot setting, without fine-tuning. We find that our MAC-SPGG achieves competitive performance with significantly fewer total parameters. Details could be found in Section 4.5. These results highlight the effectiveness of our cooperative mechanism in MAC-SPGG: by strategically leveraging multiple smaller models and incentivizing collaboration through game-theoretic design, our framework achieves strong performance with substantially fewer parameters. For a detailed case study, we refer readers to Appendix E.

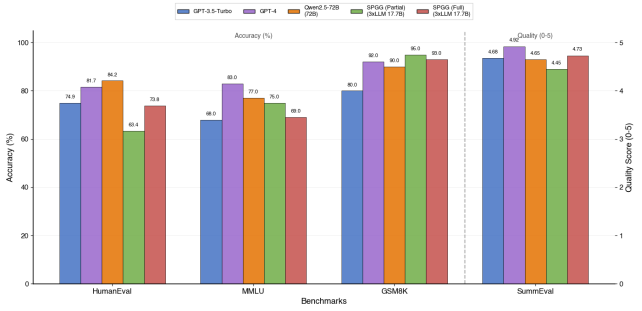


**Figure 3: Performance comparison across four benchmarks: HumanEval, MMLU, GSM8K, and SummEval. MAC-SPGG (ours) achieves competitive performance with significantly fewer total parameters.**

## 4.5 Comparison with Large LLMs

To further assess the efficiency of MAC-SPGG parameters, Figure 3 compares its performance with strong proprietary models, including GPT-3.5-Turbo [66], GPT-4-0613 [46], and Qwen2.5-72B-Instruct [63]. Despite comprising only three smaller LLMs totaling 17.7B parameters, MAC-SPGG achieves performance comparable to or even exceeding these large-scale systems on certain benchmarks.

## 4.6 Agent Sequential Ordering Effects

From Table 2, we observe three insights: (i) *Sequencing matters*: under PO, `LLaMA` → `Smol` → `Qwen` attains the highest MMLU accuracy (78%), while `Smol` → `LLaMA` → `Qwen` leads on GSM8K (95%), indicating task-dependent optima shaped by task complexity and model capabilities. (ii) *Avoid "poor" summarizer*: performance often degrades when ending with the smallest model, as the last agent bears greater responsibility in cumulative decision-making and, under PO, has limited backward correction, constraining its ability to refine complex outputs. (iii) *More context is not always better*: FO's full access does not guarantee superior results, as PO can outperform FO when excess information introduces redundancy or distractions, echoing a "less is more" effect. Together, these findings highlight the nuanced effects of agent ordering and offer actionable guidance for multi-agent design.

| Obs. | Agent Order | MMLU | GSM8K |
|---|---|---|---|
| PO | Qwen → LLaMA → Smol | 56 | 66 |
| | Qwen → Smol → LLaMA | 74 | 91 |
| | Smol → Qwen → LLaMA | 76 | 91 |
| | LLaMA → Smol → Qwen | **78** | 93 |
| | LLaMA → Qwen → Smol | 48 | 71 |
| | Smol → LLaMA → Qwen | 75 | **95** |
| FO | Qwen → LLaMA → Smol | 49 | 61 |
| | Qwen → Smol → LLaMA | **77** | 90 |
| | Smol → Qwen → LLaMA | 76 | 90 |
| | LLaMA → Smol → Qwen | 72 | **96** |
| | LLaMA → Qwen → Smol | 44 | 72 |
| | Smol → LLaMA → Qwen | 69 | 93 |

**Table 2: Ablation Study of Agent Ordering under Partial Observation (PO) and Full Observation (FO) Settings.**

## 4.7 Efficiency Analysis

We also conducted a cost efficiency analysis by comparing the token usage per task across different collaboration frameworks, as shown in Figure 4. Token counts include both input and output tokens used in each episode. The results indicate that MAC-SPGG consistently achieves lower token consumption in both PO and FO settings compared to other baselines. Specifically, the MAC-SPGG mechanism under PO achieves the lowest token usage, demonstrating significant efficiency gains. This reduction in tokens highlights the economic advantage of MAC-SPGG, as it effectively leverages structured collaboration, minimizing communication overhead while maintaining or improving task performance.

## 4.8 Ablation Study

To understand the effectiveness of the MAC-SPGG mechanism and the role of agent heterogeneity, we conducted an ablation study presented in Table 3. First, enabling the MAC-SPGG mechanism consistently improves performance across both PO and FO settings, which highlights the efficacy of our framework. Second, we chose to employ three Qwen models in our experiments due to their consistently superior performance across all evaluated benchmarks. Using the strongest available model ensures that our observed results accurately reflect the capabilities and potential benefits of the MAC-SPGG framework, without introducing confounding factors that may be introduced by model heterogeneity.

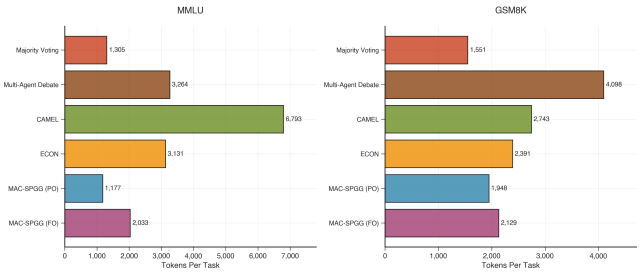


**Figure 4: Token usage per task across different collaboration frameworks. MAC-SPGG significantly reduces token consumption under both Full and Partial observation settings.**

| Obs. | Agents | Het. | SPGG | MMLU | GSM8K |
|---|---|---|---|---|---|
| PO | LLaMA + Smol + Qwen | ✓ | ✓ | **78** | 93 |
| | LLaMA + Smol + Qwen | ✓ | | 72 | 79 |
| | Qwen×3 | | ✓ | **78** | **94** |
| | Qwen×3 | | | 71 | 77 |
| FO | LLaMA + Smol + Qwen | ✓ | ✓ | 72 | **96** |
| | LLaMA + Smol + Qwen | ✓ | | 71 | 77 |
| | Qwen×3 | | ✓ | **80** | 95 |
| | Qwen×3 | | | 68 | 74 |

**Table 3: Ablation study on MAC-SPGG mechanism and agent heterogeneity (accuracy %).**

## 5 CONCLUSION

This paper presents a principled framework for structured cooperation among LLM-based agents, grounded in the theory of Sequential Public Goods Games (SPGG). By embedding incentive-compatible mechanisms into the agent interaction protocol, our approach enables conditional cooperation, belief propagation, and sequential adaptation. These capabilities are rarely addressed in existing multi-agent LLM systems. Through empirical evaluations, we show that MAC-SPGG not only improves performance across diverse tasks and observation regimes but also enhances cost efficiency by minimizing redundant communication.

More broadly, this work advances the methodological foundation for aligning autonomous language agents through economic incentives and strategic reasoning. Rather than relying on ad-hoc coordination heuristics or static voting rules, MAC-SPGG formalizes collaboration as a dynamic process shaped by information flow and strategic interdependence.

Our findings invite further exploration into mechanism design for large-scale multi-agent LLM systems. We believe this work takes an essential step toward scalable, mechanism-grounded, and adaptive cooperation among foundation models.

# REFERENCES


[1] Elif Akata, Lion Schulz, Julian Coda-Forno, Seong Joon Oh, Matthias Bethge, and Eric Schulz. 2023. Playing repeated games with Large Language Models. *CoRR* abs/2305.16867 (2023).

[2] Loubna Ben Allal, Anton Lozhkov, Elie Bakouch, Gabriel Mart'in Bl'azquez, Guilherme Penedo, Lewis Tunstall, Andrés Marafioti, Hynek Kydl'ivcek, Agust'in Piqueres Lajar'in, Vaibhav Srivastav, and et al. 2025. SmolLM2: When Smol Goes Big - Data-Centric Training of a Small Language Model. *ArXiv* abs/2502.02737 (2025). https://api.semanticscholar.org/CorpusID:276116722

[3] James Andreoni. 1988. Why free ride?: Strategies and learning in public goods experiments. *Journal of Public Economics* 37 (1988), 291–304. https://api.semanticscholar.org/CorpusID:17935915

[4] Chowdhury Mohammad Sakib Anwar and Konstantinos Georgalos. 2023. Position uncertainty in a sequential public goods game: an experiment. *Experimental Economics* (2023). https://api.semanticscholar.org/CorpusID:260351178

[5] Paul Belleflamme, Thomas Lambert, and Armin Schwienbacher. 2013. Crowdfunding: Tapping the Right Crowd. *Entrepreneurship & Finance eJournal* (2013). https://api.semanticscholar.org/CorpusID:2461588

[6] Philip Brookins and Jason Debacker. 2023. Playing Games With GPT: What Can We Learn About a Large Language Model From Canonical Strategic Games? *SSRN Electronic Journal* (2023). https://api.semanticscholar.org/CorpusID:259714625

[7] Michael Carter. 2001. *Foundations of mathematical economics*. MIT press.

[8] Mert Cemri, Melissa Z. Pan, Shuyi Yang, Lakshya A. Agrawal, Bhavya Chopra, Rishabh Tiwari, Kurt Keutzer, Aditya G. Parameswaran, Dan Klein, Kannan Ramchandran, and et al. 2025. Why Do Multi-Agent LLM Systems Fail? *CoRR* abs/2503.13657 (2025).

[9] Justin Chih-Yao Chen, Swarnadeep Saha, and Mohit Bansal. 2024. ReConcile: Round-Table Conference Improves Reasoning via Consensus among Diverse LLMs. In *ACL (1)*. Association for Computational Linguistics, 7066–7085.

[10] Justin Chih-Yao Chen, Archiki Prasad, Swarnadeep Saha, Elias Stengel-Eskin, and Mohit Bansal. 2024. Magicore: Multi-agent, iterative, coarse-to-fine refinement for reasoning. *arXiv preprint arXiv:2409.12147* (2024).

[11] Justin Chih-Yao Chen, Swarnadeep Saha, and Mohit Bansal. 2023. ReConcile: Round-Table Conference Improves Reasoning via Consensus among Diverse LLMs. *ArXiv* abs/2309.13007 (2023). https://api.semanticscholar.org/CorpusID:262217323

[12] Mark Chen, Jerry Tworek, Heewoo Jun, Qiming Yuan, Henrique Pondé de Oliveira Pinto, Jared Kaplan, Harri Edwards, Yuri Burda, Nicholas Joseph, Greg Brockman, and et al. 2021. Evaluating Large Language Models Trained on Code. *CoRR* abs/2107.03374 (2021).

[13] Weize Chen, Yusheng Su, Jingwei Zuo, Cheng Yang, Chenfei Yuan, Cheng Qian, Chi-Min Chan, Yujia Qin, Ya-Ting Lu, Ruobing Xie, and et al. 2023. AgentVerse: Facilitating Multi-Agent Collaboration and Exploring Emergent Behaviors in Agents. *ArXiv* abs/2308.10848 (2023). https://api.semanticscholar.org/CorpusID:261048935

[14] Pengyu Cheng, Tianhao Hu, Han Xu, Zhisong Zhang, Yong Dai, Lei Han, Nan Du, and Xiaolong Li. 2024. Self-playing Adversarial Language Game Enhances LLM Reasoning. In *NeurIPS*.

[15] Aakanksha Chowdhery, Sharan Narang, Jacob Devlin, Maarten Bosma, Gaurav Mishra, Adam Roberts, Paul Barham, Hyung Won Chung, Charles Sutton, Sebastian Gehrmann, and et al. 2022. PaLM: Scaling Language Modeling with Pathways. *ArXiv* abs/2204.02311 (2022). https://api.semanticscholar.org/CorpusID:247951931

[16] Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, and et al. 2021. Training Verifiers to Solve Math Word Problems. *CoRR* abs/2110.14168 (2021).

[17] Sara Connolly and Alistair Munro. 1999. Economics of the Public Sector. https://api.semanticscholar.org/CorpusID:152587074

[18] Tim Dettmers, Artidoro Pagnoni, Ari Holtzman, and Luke Zettlemoyer. 2023. QLoRA: Efficient Finetuning of Quantized LLMs. *ArXiv* abs/2305.14314 (2023). https://api.semanticscholar.org/CorpusID:258841328

[19] Yilun Du, Shuang Li, Antonio Torralba, Joshua B. Tenenbaum, and Igor Mordatch. 2024. Improving Factuality and Reasoning in Language Models through Multiagent Debate. In *ICML*. OpenReview.net.

[20] Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, and et al. 2024. The Llama 3 Herd of Models. *ArXiv* abs/2407.21783 (2024). https://api.semanticscholar.org/CorpusID:271571434

[21] Paul Dütting, Vahab Mirrokni, Renato Paes Leme, Haifeng Xu, and Song Zuo. 2024. Mechanism Design for Large Language Models. In *WWW*. ACM, 144–155.

[22] Andrew Estornell and Yang Liu. 2024. Multi-LLM Debate: Framework, Principals, and Interventions. In *NeurIPS*.

[23] Alexander R. Fabbri, Wojciech Kryscinski, Bryan McCann, Caiming Xiong, Richard Socher, and Dragomir R. Radev. 2021. SummEval: Re-evaluating Summarization Evaluation. *Trans. Assoc. Comput. Linguistics* 9 (2021), 391–409.

[24] Caoyun Fan, Jindou Chen, Yaohui Jin, and Hao He. 2024. Can Large Language Models Serve as Rational Players in Game Theory? A Systematic Analysis. In *AAAI*. AAAI Press, 17960–17967.

[25] Ernst Fehr and Simon Gächter. 2002. Altruistic punishment in humans. *Nature* 415, 6868 (2002), 137–140. https://doi.org/10.1038/415137a

[26] Andrea Forte and Amy Bruckman. 2005. Why Do People Write for Wikipedia? Incentives to Contribute to Open-Content Publishing. https://api.semanticscholar.org/CorpusID:18268963

[27] Simon Gächter, Daniele Nosenzo, Elke Renner, and Martin Sefton. 2010. Sequential vs. simultaneous contributions to public goods: Experimental evidence. *Journal of Public Economics* 94, 7-8 (2010), 515–522.

[28] Andrea Gallice and Ignacio Monzón. 2018. Cooperation in Social Dilemmas Through Position Uncertainty. *ERN: Non-Cooperative Games (Topic)* (2018). https://api.semanticscholar.org/CorpusID:51852661

[29] Joel M. Guttman, Leif Danziger, and Robert McClelland. 2007. SEQUENTIAL CONTRIBUTIONS TO PUBLIC GOODS. https://api.semanticscholar.org/CorpusID:14784313

[30] Lewis Hammond, Alan Chan, Jesse Clifton, Jason Hoelscher-Obermaier, Akbir Khan, Euan McLean, Chandler Smith, Wolfram Barfuss, Jakob Foerster, Tom'avs Gavenvciak, and et al. 2025. Multi-Agent Risks from Advanced AI. *ArXiv* abs/2502.14143 (2025). https://api.semanticscholar.org/CorpusID:276482614

[31] Zhitao He, Pengfei Cao, Yubo Chen, Kang Liu, Ruopeng Li, Mengshu Sun, and Jun Zhao. 2023. LEGO: A Multi-agent Collaborative Framework with Role-playing and Iterative Feedback for Causality Explanation Generation. In *EMNLP (Findings)*. Association for Computational Linguistics, 9142–9163.

[32] Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. 2021. Measuring Massive Multitask Language Understanding. In *ICLR*. OpenReview.net.

[33] Sirui Hong, Mingchen Zhuge, Jiaqi Chen, Xiawu Zheng, Yuheng Cheng, Ceyao Zhang, Jinlin Wang, Zili Wang, Steven Ka Shing Yau, Zijuan Lin, Liyang Zhou, Chenyu Ran, Lingfeng Xiao, Chenglin Wu, and Jürgen Schmidhuber. 2024. MetaGPT: Meta Programming for A Multi-Agent Collaborative Framework. In *Proceedings of the International Conference on Learning Representations (ICLR)*. https://github.com/geekan/MetaGPT Published as a conference paper at ICLR 2024.

[34] Wenyue Hua, Ollie Liu, Lingyao Li, Alfonso Amayuelas, Julie Chen, Lucas Jiang, Mingyu Jin, Lizhou Fan, Fei Sun, William Wang, and et al. 2024. Game-theoretic LLM: Agent Workflow for Negotiation Games. *CoRR* abs/2411.05990 (2024).

[35] Jared Kaplan, Sam McCandlish, T. J. Henighan, Tom B. Brown, Benjamin Chess, Rewon Child, Scott Gray, Alec Radford, Jeff Wu, and Dario Amodei. 2020. Scaling Laws for Neural Language Models. *ArXiv* abs/2001.08361 (2020). https://api.semanticscholar.org/CorpusID:210861095

[36] Takeshi Kojima, Shixiang Shane Gu, Machel Reid, Yutaka Matsuo, and Yusuke Iwasawa. 2022. Large Language Models are Zero-Shot Reasoners. In *NeurIPS*.

[37] John O. Ledyard. 1994. Public Goods: A Survey of Experimental Research. *Public Economics* (1994), 111–194. https://api.semanticscholar.org/CorpusID:214607050

[38] Guohao Li, Hasan Hammoud, Hani Itani, Dmitrii Khizbullin, and Bernard Ghanem. 2023. CAMEL: Communicative Agents for "Mind" Exploration of Large Language Model Society. In *NeurIPS*.

[39] Junyou Li, Qin Zhang, Yangbin Yu, Qiang Fu, and Deheng Ye. 2024. More Agents Is All You Need. *Trans. Mach. Learn. Res.* 2024 (2024).

[40] Yuxuan Li, Aoi Naito, and Hirokazu Shirado. 2025. Assessing Collective Reasoning in Multi-Agent LLMs via Hidden Profile Tasks. *CoRR* abs/2505.11556 (2025).

[41] Tian Liang, Zhiwei He, Wenxiang Jiao, Xing Wang, Yan Wang, Rui Wang, Yujiu Yang, Shuming Shi, and Zhaopeng Tu. 2024. Encouraging Divergent Thinking in Large Language Models through Multi-Agent Debate. In *EMNLP*. Association for Computational Linguistics, 17889–17904.

[42] Wei Liu, Chenxi Wang, Yifei Wang, Zihao Xie, Rennai Qiu, Yufan Dang, Zhuoyun Du, Weize Chen, Cheng Yang, and Chen Qian. 2024. Autonomous Agents for Collaborative Task under Information Asymmetry. In *NeurIPS*.

[43] Nunzio Lorè and Babak Heydari. 2023. Strategic Behavior of Large Language Models: Game Structure vs. Contextual Framing. *CoRR* abs/2309.05898 (2023).

[44] Shaoguang Mao, Yuzhe Cai, Yan Xia, Wenshan Wu, Xun Wang, Fengyi Wang, Qiang Guan, Tao Ge, and Furu Wei. 2025. ALYMPICS: LLM Agents Meet Game Theory. In *COLING*. Association for Computational Linguistics, 2845–2866.

[45] Paul Milgrom and Chris Shannon. 1994. Monotone comparative statics. *Econometrica* 62, 1 (1994), 157–180. https://doi.org/10.2307/2951479

[46] OpenAI. 2023. GPT-4 Technical Report. *CoRR* abs/2303.08774 (2023).

[47] Dingwen Pan, Weilong Chen, Jian Shi, Chenye Wu, Dan Wang, Choong Seon Hong, and Zhu Han. 2025. Cooperation and Decision-Making of LLM Agents in Bayesian-Informed Infinitely Repeated Games. In *CISS*. IEEE, 1–6.

[48] Chanwoo Park, Seungju Han, Xingzhi Guo, Asuman E. Ozdaglar, Kaiqing Zhang, and Joo-Kyung Kim. 2025. MAPoRL: Multi-Agent Post-Co-Training for Collaborative Large Language Models with Reinforcement Learning. *CoRR* abs/2502.18439 (2025).

[49] Yinzhu Quan and Zefang Liu. 2024. InvAgent: A Large Language Model based Multi-Agent System for Inventory Management in Supply Chains. *CoRR* abs/2407.11384 (2024).

[50] Tim Roughgarden. 2002. The price of anarchy is independent of the network topology. In *Symposium on the Theory of Computing*. https://api.semanticscholar.

org/CorpusID:7826193
[51] Tim Roughgarden. 2005. Selfish routing and the price of anarchy. https://api.semanticscholar.org/CorpusID:31821998
[52] John Schulman, Philipp Moritz, Sergey Levine, Michael I Jordan, and Pieter Abbeel. 2016. High-Dimensional Continuous Control Using Generalized Advantage Estimation. In *International Conference on Learning Representations (ICLR)*. https://arxiv.org/abs/1506.02438
[53] John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. 2017. Proximal Policy Optimization Algorithms. *CoRR* abs/1707.06347 (2017).
[54] Bilgehan Sel, Priya Shanmugasundaram, Mohammad Kachuee, Kun Zhou, Ruoxi Jia, and Ming Jin. 2024. Skin-in-the-Game: Decision Making via Multi-Stakeholder Alignment in LLMs. In *ACL (1)*. Association for Computational Linguistics, 13921–13959.
[55] Karthik Sreedhar, Alice Cai, Jenny Ma, Jeffrey V. Nickerson, and Lydia B. Chilton. 2025. Simulating Cooperative Prosocial Behavior with Multi-Agent LLMs: Evidence and Mechanisms for AI Agents to Inform Policy Decisions. In *Proceedings of the 30th International Conference on Intelligent User Interfaces (IUI '25)*. ACM, 1–15. https://doi.org/10.1145/3708359.3712149
[56] Vighnesh Subramaniam, Yilun Du, Joshua B. Tenenbaum, Antonio Torralba, Shuang Li, and Igor Mordatch. 2025. Multi-Agent Finetuning: Self-Improvement with Diverse Reasoning Chains. In *Proceedings of the International Conference on Learning Representations (ICLR)*. OpenReview.net.
[57] Guido Suurmond, Otto H. Swank, and Bauke Visser. 2004. On the bad reputation of reputational concerns. *Journal of Public Economics* 88, 12 (2004), 2817–2838. https://doi.org/10.1016/j.jpubeco.2003.10.004
[58] Jean Tirole and Josh Lerner. 2002. Some Simple Economics of Open Source. *IO: Firm Structure* (2002). https://api.semanticscholar.org/CorpusID:219722756
[59] Khanh-Tung Tran, Dung Dao, Minh-Duong Nguyen, Quoc-Viet Pham, Barry O'Sullivan, and Hoang D. Nguyen. 2025. Multi-Agent Collaboration Mechanisms: A Survey of LLMs. *CoRR* abs/2501.06322 (2025).
[60] Xuezhi Wang, Jason Wei, Dale Schuurmans, Quoc V. Le, Ed H. Chi, Sharan Narang, Aakanksha Chowdhery, and Denny Zhou. 2023. Self-Consistency Improves Chain of Thought Reasoning in Language Models. In *ICLR*. OpenReview.net.
[61] Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Ed H. Chi, F. Xia, Quoc Le, and Denny Zhou. 2022. Chain of Thought Prompting Elicits Reasoning in Large Language Models. *ArXiv* abs/2201.11903 (2022). https://api.semanticscholar.org/CorpusID:246411621
[62] Qingyun Wu, Gagan Bansal, Jieyu Zhang, Yiran Wu, Beibin Li, Erkang (Eric) Zhu, Li Jiang, Xiaoyun Zhang, Shaokun Zhang, Jiale Liu, Ahmed Hassan Awadallah, Ryen W. White, Doug Burger, and Chi Wang. 2023. AutoGen: Enabling Next-Gen LLM Applications via Multi-Agent Conversation. https://api.semanticscholar.org/CorpusID:263611068
[63] An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, and et al. 2025. Qwen3 Technical Report. *ArXiv* abs/2505.09388 (2025). https://api.semanticscholar.org/CorpusID:278602855
[64] An Yang, Baosong Yang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Zhou, Chengpeng Li, Chengyuan Li, Dayiheng Liu, Fei Huang, and et al. 2024. Qwen2 Technical Report. *CoRR* abs/2407.10671 (2024).
[65] Huaiyuan Yao, Longchao Da, Vishnu Nandam, Justin Turnau, Zhiwei Liu, Linsey Pang, and Hua Wei. 2024. CoMAL: Collaborative Multi-Agent Large Language Models for Mixed-Autonomy Traffic. *CoRR* abs/2410.14368 (2024).
[66] Junjie Ye, Xuanting Chen, Nuo Xu, Can Zu, Zekai Shao, Shichun Liu, Yuhan Cui, Zeyang Zhou, Chao Gong, Yang Shen, and et al. 2023. A Comprehensive Capability Analysis of GPT-3 and GPT-3.5 Series Models. *CoRR* abs/2303.10420 (2023).
[67] Xie Yi, Zhanke Zhou, Chentao Cao, Qiyu Niu, Tongliang Liu, and Bo Han. 2025. From Debate to Equilibrium: Belief-Driven Multi-Agent LLM Reasoning via Bayesian Nash Equilibrium. In *Proceedings of the 42nd International Conference on Machine Learning (ICML)*. https://api.semanticscholar.org/CorpusID:279260557
[68] Xie Yi, Zhanke Zhou, Chentao Cao, Qiyu Niu, Tongliang Liu, and Bo Han. 2025. From Debate to Equilibrium: Belief-Driven Multi-Agent LLM Reasoning via Bayesian Nash Equilibrium. *CoRR* abs/2506.08292 (2025).
[69] Chenrui Zhang, Lin Liu, Chuyuan Wang, Xiao Sun, Hongyu Wang, Jinpeng Wang, and Mingchen Cai. 2024. Prefer: Prompt ensemble learning via feedback-reflect-refine. In *Proceedings of the AAAI conference on artificial intelligence*, Vol. 38. 19525–19532.
[70] Wanjia Zhao, Mert Yüksekgönül, Shirley Wu, and James Zou. 2025. SiriuS: Self-improving Multi-agent Systems via Bootstrapped Reasoning. *CoRR* abs/2502.04780 (2025).