Think, Reflect, Create: Metacognitive Learning for Zero-Shot Robotic Planning with LLMs

Anonymous ACL submission

Abstract

While large language models (LLMs) have shown great potential across various domains, their applications in robotics remain largely limited to static, prompt-based behaviors and still face challenges in handling complex tasks under zero-shot or few-shot settings. Inspired by human metacognitive learning and creative problem-solving, we address this limitation by exploring a fundamental research question: *Can LLMs be empowered with metacognitive capabilities to reason, reflect, and create, thereby enhancing their ability to perform robotic tasks with minimal demonstrations?* In this paper, we present an early-stage framework that integrates metacognitive learning into LLM-powered multi-robot collaboration. The proposed framework equips the LLM-powered robotic agents with a skill decomposition and self-reflection mechanism that identifies modular skills from prior tasks, reflects on failures in unseen task scenarios, and synthesizes effective new solutions. Experimental results show that our metacognitive-learning-empowered LLM framework significantly outperforms existing baselines. Moreover, we observe that the framework is capable of generating solutions that differ from the ground truth yet still successfully complete the tasks. These exciting findings support our hypothesis that metacognitive learning can foster creativity in robotic planning.

1 Introduction

In recent years, large language models (LLMs) have emerged as powerful reasoning engines capable of performing complex planning, decision-making, and knowledge-intensive tasks across various application domains. These successes have sparked growing interest in applying LLMs to robotic operations, enabling robots to understand instructions, generate executable action sequences, and generalize across diverse and novel scenarios (Wang et al., 2024; Liu et al., 2024; Jin et al., 2024; Tan et al., 2024; Cheng et al., 2024; Chen et al., 2025). Recent research on LLM-powered robotic operations can be broadly categorized into three main directions: 1) generating robot plans by prompting LLMs with task instructions (Mandi et al., 2024), 2) enabling embodied reasoning by integrating LLMs with multimodal perception systems (Aissi et al., 2025), and 3) synthesizing robot control codes from natural language commands (Liang et al., 2023). While these approaches demonstrate the potential of LLMs to advance robotic operations, most remain limited to static prompting, which restricts their performance in complex tasks in zero-shot or few-shot settings.

Inspired by human metacognitive learning and its impact on enabling creative skills (Hargrove, 2013; Schuster et al., 2020), we address this limitation by exploring a fundamental question: *Can LLMs be endowed with metacognitive capabilities to reason, reflect, and create, thereby enhancing their ability to perform robotic tasks with minimal demonstrations?* In this paper, we present an early-stage framework that integrates metacognitive learning into LLM-powered multi-robot collaboration. The proposed system equips the LLM with a skill decomposition and self-reflection mechanism that identifies modular skills from prior tasks, reflects on failures in novel scenarios, and synthesizes effective new solutions. Our work in this paper has three key contributions: (1) To the best of our knowledge, this is the first work to explore integrating metacognitive learning into LLM-equipped robot operations, to support both reliable performance and creative problem-solving; (2) We propose a metacognitive learning framework that enables the LLM-powered robotic agents decompose modular skills, reflect on task failures, and synthesize effective new solutions; and (3) We validate our framework on the RoCo benchmark, where it significantly outperforms baselines and sometimes generates successful solutions that deviate from

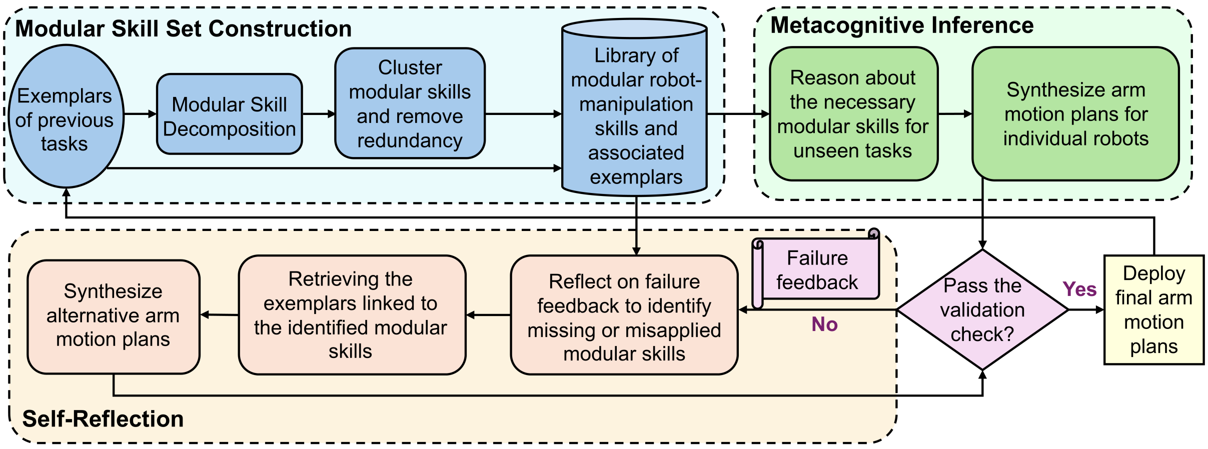

Figure 1: Overview of our proposed metacognitive learning module.

the ground truth, which supports the hypothesis that metacognitive learning can foster reliable and creative robotic planning.

The rest of the paper is organized as follows. In Section 2, we will present the problem formulation. In Section 3, we will introduce our proposed metacognitive learning module. In Section 4, we will show the experiment results, followed by conclusions with discussion on limitations in Section 5.

2 Problem Formulation

We consider a cooperative multi-agent manipulation setting, where multiple LLM-powered robot agents collaborate to complete tasks over a finite time horizon. Each agent operates within its own observation space and needs to coordinate with other agents to achieve a shared task objective. At each time step, each agent n receives a prompt $p_t^n = f_n(g_n, o_t^n, r_t^n)$ and outputs π_n, where π_n denotes the resulting arm motion plan, g_n represents the agent-specific task description that includes goals and constraints, $o_t^n \in \Omega_n$ is its current observation, and r_t^n is the metacognition-informed input.

The generation of effective r_t^n is guided by our proposed metacognitive learning module, which empowers the LLM to decompose prior task completions into modular skills, synthesize arm motion plans $\{\pi_n\}_{n=1}^N$ for unseen task scenarios, reflect on planning generation failures, and iteratively produce effective and potentially creative solutions. In our current implementation, we adopted the validation mechanism in (Mandi et al., 2024) to detect planning generation failures and trigger the self-reflection process.

3 Methodology

The proposed metacognitive learning module is illustrated in Fig. 1. As shown in the figure, the module comprises three key components: (1) modular skill set construction, (2) metacognitive inference, and (3) self-reflection.

In the *modular skill set construction* component, the LLM is guided by the metacognition-informed input r_t^n to construct a library of exemplars from previously completed tasks. Each exemplar maps identified robot manipulation skills to a successful task execution and includes a representative scene along with a one-shot demonstration of an effective action plan. The LLM then extracts fine-grained modular skills from each exemplar, clusters similar skills to reduce redundancy, and organizes them into a reusable library of transferable modular robot manipulation skills and their associated exemplars.

In the *metacognitive inference* component, based on the library of modular robot manipulation skills and associated exemplars, the task description g_n, and the current observation o_t^n, the LLM is guided by r_t^n to reason about the necessary modular skills for the new task. Using the identified skills and their associated exemplars, the LLM synthesizes arm motion plans π_n for the robot agent n.

The *self-reflection* component is activated when the arm motion plans synthesized during metacognitive inference do not pass the validation check. It guides the LLM to reflect on failure feedback to identify modular skills that are missing or need refinement. Based on these insights, the LLM retrieves the corresponding exemplars and synthesizes revised arm motion plans.

Equipped with the proposed metacognitive learning module, the LLM adaptively generates reliable

Table 1: Performance Comparison between Our Framework and Baselines.

		Move Rope	Arrange Cabinet	Make Sandwich
Central Plan	Task Success Rate	0.50 ± 0.11	0.90 ± 0.07	0.96 ± 0.04
	Environment Steps, Replan Attempts	2.3, 3.9	4.0, 2.7	8.8, 1.2
RoCo+GPT-4	Task Success Rate	0.65 ± 0.11	0.75 ± 0.10	0.80 ± 0.08
	Environment Steps, Replan Attempts	2.5, 3.1	4.7, 2.0	10.2, 1.7
Our framework	Task Success Rate	0.76 ± 0.10	0.95 ± 0.05	0.95 ± 0.05
	Environment Steps, Replan Attempts	2.0, 2.4	4.0, 1.7	9.4, 1.8

and potentially creative arm motion plans, which are subsequently used to update the library of prior task exemplars.

4 Experiments

We validate the performance of the proposed metacognitive learning module in enabling LLM-enabled robotic agents to complete complex multi-robot collaboration tasks under zero-shot settings. In this initial stage of work, we conduct experiments using the RoCo benchmark (Mandi et al., 2024), focusing on three challenging tasks: *Move Rope*, *Arrange Cabinet*, and *Make Sandwich*. In each task, LLM-powered robot agents coordinate via structured textual prompts that include metacognition-informed inputs and execute actions in environments with obstacles and spatial constraints. An embedded validation mechanism detects planning failures, such as those caused by collisions or inverse kinematics (IK) infeasibility, and triggers the self-reflection process. All experiments are conducted on a machine with an NVIDIA A100 GPU to support efficient inference using the LLaMA 3.1-70B model. We note that LLaMA is adopted instead of GPT-4 in our framework to align with our long-term objective of developing an open-source LLM framework that can be widely adopted by the robotics and AI research communities.

Baselines and Performance Metrics

We compare our framework against two baselines. (1) **Central Plan**: an oracle LLM-based planner with access to the full environment state, task description, and capabilities of all robots. It generates a joint centralized plan without accounting for information asymmetry. (2) **RoCo+GPT-4**: the state-of-the-art multi-robot collaboration framework proposed in (Mandi et al., 2024), which uses GPT-4 but does not incorporate metacognition-informed input. To ensure a fair performance comparison, we follow RoCoBench (Mandi et al., 2024) and use the same evaluation metrics: (1) *Task Success Rate*, which measures the percentage of successful task completions within a fixed number of rounds (we also use over 20 rounds in our experiements); (2) *Environment Steps*, defined as the average number of steps taken in successful runs; and (3) *Replan Attempts*, which refers to the average number of replan attempts across all runs.

Experiment Findings

Reliability Enhancement: The evaluation results comparing our proposed framework with the two baselines across the three tasks are shown in Table 1. In the *Move Rope* task, which is the most challenging task in the experiments, our framework achieves a success rate of 0.76, representing a 17% improvement over RoCo+GPT-4 and a 26% gain over Central Plan. It also reduces environment steps in successful runs to 2.0, compared to 2.5 for RoCo+GPT-4 and 2.3 for Central Plan, and lowers replan attempts to 2.4, compared to 3.1 for RoCo+GPT-4 and 3.9 for Central Plan. These results indicate more effective coordination and faster convergence under challenging spatial constraints. For the *Arrange Cabinet* task, our framework achieves a success rate of 0.95, outperforming RoCo by 20% and the Central Plan by 5%. Additionally, it requires the same number of environment steps as the Central Plan while requiring fewer replan attempts than both baselines, suggesting improved planning robustness and reduced reliance on corrective execution. For the *Make Sandwich* task, which involves long-horizon planning and strict stacking constraints, our framework achieves performance comparable to Central Plan. In comparison to the RoCo+GPT-4 baseline, it achieves a 15% higher success rate and reduces the required environment steps, highlighting its ability to generalize to structurally complex tasks with minimal planning overhead, despite a slight increase in replan attempts. These results validate the effectiveness of our framework in advancing the capabilities of LLM-powered robot agents for completing complex tasks under zero-shot settings. They suggest that the proposed metacognitive learning module enables LLM-equipped robot agents to adaptively and proactively reason about and reflect on spatial, temporal, and structural challenges.

Creativity Cultivation: In the experiments, we observe that our framework is capable of generating solutions that differ from the ground truth yet still successfully complete the tasks. These exciting findings support our hypothesis that metacognitive learning can cultivate creativity in robotic planning. Due to space limitations, we describe one representative case. In the *Move Rope* task, two robot agents collaboratively grasp the ends of a rope, maneuver it over an obstacle wall, and place it into a designated groove. In the ground-truth plan, the robots grasp the two ends of the rope to complete the task. In contrast, after initial plan generation failures due to collision or IK infeasibility, our framework generates an alternative motion plan in which one robot grasps the rope slightly inward instead of at the end. Although this alternative plan deviates from the ground truth, it effectively shortens the trajectory, improves the spatial separation between the robot arms, and significantly reduces the risk of collision and IK errors.

Table 2: Reflection success results by our framework.

	Move Rope	Arrange Cabinet	Make Sandwich
Reflection Success Rate	0.44	1.00	0.70

Metacognitive Self-Reflection Analysis: As shown in Fig. 1, the self-reflection component is one of the three key elements of our proposed metacognitive learning module. To evaluate its practical impact, we introduce an additional metric in our experiments, the *reflection success rate*, defined as the proportion of successful plan regenerations among all reflection attempts within task rounds that ultimately succeeded. The results, summarized in Table 2. As shown in the table, in the *Arrange Cabinet* task, the framework achieves a perfect reflection success rate of 100%, indicating that every initial failure was successfully recovered through self-reflection. In the *Make Sandwich* task, which involves long-horizon dependencies and stacking constraints, the framework recovers from 70% of failed plans via self-reflection. Even in the most challenging *Move Rope* task, characterized by tight spatial coordination, the framework achieves a 44% reflection success rate, underscoring the role of self-reflection in enabling meaningful plan recovery under physical constraints. These findings highlight the critical role of the self-reflection component within our metacognitive learning module. They also show that the proposed framework not only supports LLM-powered agents in initial plan generation but also enables them to proactively reason about and reflect on execution failures, providing the essential capability of reliable recovery and adaptation in zero-shot settings.

5 Conclusions

In this paper, we introduced an early-stage framework to explore the research question: Can LLMs be empowered with metacognitive capabilities to reason, reflect, and create, thereby enhancing their ability to perform robotic tasks with minimal demonstrations? Our proposed framework integrates metacognitive learning into LLM-powered multi-robot collaboration, inspired by human reflective problem-solving processes. By equipping LLM-driven robotic agents with a metacognitive learning module, our framework enables effective reasoning, self-reflection on failures, and creative synthesis of novel solutions in zero-shot robotic planning scenarios. Experimental results demonstrate that our framework outperforms state-of-the-art methods and can generate innovative task solutions that differ from the provided ground-truth plans. These findings highlight the potential of metacognitive strategies to significantly enhance the adaptability, reliability, and creativity of robotic systems powered by LLMs.

Limitations

While this early-stage work shows exciting results on exploiting metacognitive learning methodology to advance LLM-based robot operations. Several limitations remain. For example, the current metacognitive self-reflection focuses primarily on identifying missing or misapplied skills but does not yet support finer-grained failure modes such as multi-agent coordination errors and long-horizon dependency mistakes. Extending the system to reason hierarchically about such failures remains an open challenge.

References

Mohamed Salim Aissi, Clémence Grislain, Mohamed Chetouani, Olivier Sigaud, Laure Soulier, and Nicolas Thome. 2025. Viper: Visual perception and explainable reasoning for sequential decision-making. *arXiv preprint arXiv:2503.15108*.

Zixuan Chen, Jing Huo, Yangtao Chen, and Yang Gao. 2025. Robohorizon: An llm-assisted multi-view world model for long-horizon robotic manipulation. *arXiv preprint arXiv:2501.06605*.

Guangran Cheng, Chuheng Zhang, Wenzhe Cai, Li Zhao, Changyin Sun, and Jiang Bian. 2024. Empowering large language models on robotic manipulation with affordance prompting. *arXiv preprint arXiv:2404.11027*.

Ryan A Hargrove. 2013. Assessing the long-term impact of a metacognitive approach to creative skill development. *International Journal of Technology and Design Education*, 23:489–517.

Yixiang Jin, Dingzhe Li, Jun Shi, Peng Hao, Fuchun Sun, Jianwei Zhang, Bin Fang, and 1 others. 2024. Robotgpt: Robot manipulation learning from chatgpt. *IEEE Robotics and Automation Letters*, 9(3):2543–2550.

Jacky Liang, Wenlong Huang, Fei Xia, Peng Xu, Karol Hausman, Brian Ichter, Pete Florence, and Andy Zeng. 2023. Code as policies: Language model programs for embodied control. In *2023 IEEE International Conference on Robotics and Automation (ICRA)*, pages 9493–9500. IEEE.

Haokun Liu, Yaonan Zhu, Kenji Kato, Atsushi Tsukahara, Izumi Kondo, Tadayoshi Aoyama, and Yasuhisa Hasegawa. 2024. Enhancing the llm-based robot manipulation through human-robot collaboration. *IEEE Robotics and Automation Letters*.

Zhao Mandi, Shreeya Jain, and Shuran Song. 2024. Roco: Dialectic multi-robot collaboration with large language models. In *2024 IEEE International Conference on Robotics and Automation (ICRA)*, pages 286–299. IEEE.

Corinna Schuster, Ferdinand Stebner, Detlev Leutner, and Joachim Wirth. 2020. Transfer of metacognitive skills in self-regulated learning: an experimental training study. *Metacognition and Learning*, 15(3):455–477.

Runjia Tan, Shanhe Lou, Yanxin Zhou, and Chen Lv. 2024. Multi-modal llm-enabled long-horizon skill learning for robotic manipulation. In *2024 IEEE International Conference on Cybernetics and Intelligent Systems (CIS) and IEEE International Conference on Robotics, Automation and Mechatronics (RAM)*, pages 14–19. IEEE.

Jiaqi Wang, Enze Shi, Huawen Hu, Chong Ma, Yiheng Liu, Xuhui Wang, Yincheng Yao, Xuan Liu, Bao Ge, and Shu Zhang. 2024. Large language models for robotics: Opportunities, challenges, and perspectives. *Journal of Automation and Intelligence*.