LNAI 14271

Huayong Yang · Honghai Liu ·
Jun Zou · Zhouping Yin ·
Lianqing Liu · Geng Yang ·
Xiaoping Ouyang · Zhiyong Wang (Eds.)

Intelligent Robotics and Applications

16th International Conference, ICIRA 2023
Hangzhou, China, July 5–7, 2023
Proceedings, Part V

5 Part V

Springer

MOREMEDIA

Lecture Notes in Computer Science

Lecture Notes in Artificial Intelligence **14271**

Founding Editor

Jörg Siekmann

Series Editors

Randy Goebel, *University of Alberta, Edmonton, Canada*
Wolfgang Wahlster, *DFKI, Berlin, Germany*
Zhi-Hua Zhou, *Nanjing University, Nanjing, China*

The series Lecture Notes in Artificial Intelligence (LNAI) was established in 1988 as a topical subseries of LNCS devoted to artificial intelligence.

The series publishes state-of-the-art research results at a high level. As with the LNCS mother series, the mission of the series is to serve the international R & D community by providing an invaluable service, mainly focused on the publication of conference and workshop proceedings and postproceedings.

Huayong Yang · Honghai Liu · Jun Zou ·
Zhouping Yin · Lianqing Liu · Geng Yang ·
Xiaoping Ouyang · Zhiyong Wang
Editors

Intelligent Robotics and Applications

16th International Conference, ICIRA 2023
Hangzhou, China, July 5–7, 2023
Proceedings, Part V

Springer

Editors
Huayong Yang
Zhejiang University
Hangzhou, China

Jun Zou
Zhejiang University
Hangzhou, China

Lianqing Liu
Shenyang Institute of Automation
Shenyang, Liaoning, China

Xiaoping Ouyang
Zhejiang University
Hangzhou, China

Honghai Liu
Harbin Institute of Technology
Shenzhen, China

Zhouping Yin
Huazhong University of Science
and Technology
Wuhan, China

Geng Yang
Zhejiang University
Hangzhou, China

Zhiyong Wang
Harbin Institute of Technology
Shenzhen, China

ISSN 0302-9743 ISSN 1611-3349 (electronic)
Lecture Notes in Artificial Intelligence
ISBN 978-981-99-6494-9 ISBN 978-981-99-6495-6 (eBook)
https://doi.org/10.1007/978-981-99-6495-6

LNCS Sublibrary: SL7 – Artificial Intelligence

This Springer imprint is published by the registered company Springer Nature Singapore Pte Ltd.
The registered company address is: 152 Beach Road, #21-01/04 Gateway East, Singapore 189721, Singapore

Paper in this product is recyclable.

Preface

With the theme "Smart Robotics for Sustainable Society", the 16th International Conference on Intelligent Robotics and Applications (ICIRA 2023) was held in Hangzhou, China, July 5–7, 2023, and designed to encourage advancement in the field of robotics, automation, mechatronics, and applications. It aimed to promote top-level research and globalize quality research in general, making discussions and presentations more internationally competitive and focusing on the latest outstanding achievements, future trends, and demands.

ICIRA 2023 was organized and hosted by Zhejiang University, co-hosted by Harbin Institute of Technology, Huazhong University of Science and Technology, Chinese Academy of Sciences, and Shanghai Jiao Tong University, co-organized by State Key Laboratory of Fluid Power and Mechatronic Systems, State Key Laboratory of Robotics and System, State Key Laboratory of Digital Manufacturing Equipment and Technology, State Key Laboratory of Mechanical System and Vibration, State Key Laboratory of Robotics, and School of Mechanical Engineering of Zhejiang University. Also, ICIRA 2023 was technically co-sponsored by Springer. On this occasion, ICIRA 2023 was a successful event after the COVID-19 pandemic. It attracted more than 630 submissions, and the Program Committee undertook a rigorous review process for selecting the most deserving research for publication. The Advisory Committee gave advice for the conference program. Also, they help to organize special sections for ICIRA 2023. Finally, a total of 431 papers were selected for publication in 9 volumes of Springer's Lecture Note in Artificial Intelligence. For the review process, single-blind peer review was used. Each review took around 2–3 weeks, and each submission received at least 2 reviews and 1 meta-review.

In ICIRA 2023, 12 distinguished plenary speakers delivered their outstanding research works in various fields of robotics. Participants gave a total of 214 oral presentations and 197 poster presentations, enjoying this excellent opportunity to share their latest research findings. Here, we would like to express our sincere appreciation to all the authors, participants, and distinguished plenary and keynote speakers. Special thanks are also extended to all members of the Organizing Committee, all reviewers for peer-review, all staffs of the conference affairs group, and all volunteers for their diligent work.

July 2023

Huayong Yang
Honghai Liu
Jun Zou
Zhouping Yin
Lianqing Liu
Geng Yang
Xiaoping Ouyang
Zhiyong Wang

Organization

Conference Chair

Huayong Yang	Zhejiang University, China

Honorary Chairs

Youlun Xiong	Huazhong University of Science and Technology, China
Han Ding	Huazhong University of Science and Technology, China

General Chairs

Honghai Liu	Harbin Institute of Technology, China
Jun Zou	Zhejiang University, China
Zhouping Yin	Huazhong University of Science and Technology, China
Lianqing Liu	Chinese Academy of Sciences, China

Program Chairs

Geng Yang	Zhejiang University, China
Li Jiang	Harbin Institute of Technology, China
Guoying Gu	Shanghai Jiao Tong University, China
Xinyu Wu	Chinese Academy of Sciences, China

Award Committee Chair

Yong Lei	Zhejiang University, China

Publication Chairs

Xiaoping Ouyang	Zhejiang University, China
Zhiyong Wang	Harbin Institute of Technology, China

Regional Chairs

Zhiyong Chen	University of Newcastle, Australia
Naoyuki Kubota	Tokyo Metropolitan University, Japan
Zhaojie Ju	University of Portsmouth, UK
Eric Perreault	Northeastern University, USA
Peter Xu	University of Auckland, New Zealand
Simon Yang	University of Guelph, Canada
Houxiang Zhang	Norwegian University of Science and Technology, Norway
Duanling Li	Beijing University of Posts and Telecommunications, China

Advisory Committee

Jorge Angeles	McGill University, Canada
Tamio Arai	University of Tokyo, Japan
Hegao Cai	Harbin Institute of Technology, China
Tianyou Chai	Northeastern University, China
Jiansheng Dai	King's College London, UK
Zongquan Deng	Harbin Institute of Technology, China
Han Ding	Huazhong University of Science and Technology, China
Xilun Ding	Beihang University, China
Baoyan Duan	Xidian University, China
Xisheng Feng	Shenyang Institute of Automation, Chinese Academy of Sciences, China
Toshio Fukuda	Nagoya University, Japan
Jianda Han	Nankai University, China
Qiang Huang	Beijing Institute of Technology, China
Oussama Khatib	Stanford University, USA
Yinan Lai	National Natural Science Foundation of China, China
Jangmyung Lee	Pusan National University, Korea
Zhongqin Lin	Shanghai Jiao Tong University, China

Hong Liu	Harbin Institute of Technology, China
Honghai Liu	University of Portsmouth, UK
Shugen Ma	Ritsumeikan University, Japan
Daokui Qu	Siasun Robot and Automation Co., Ltd., China
Min Tan	Institute of Automation, Chinese Academy of Sciences, China
Kevin Warwick	Coventry University, UK
Guobiao Wang	National Natural Science Foundation of China, China
Tianmiao Wang	Beihang University, China
Tianran Wang	Shenyang Institute of Automation, Chinese Academy of Sciences, China
Yuechao Wang	Shenyang Institute of Automation, Chinese Academy of Sciences, China
Bogdan M. Wilamowski	Auburn University, USA
Ming Xie	Nanyang Technological University, Singapore
Yangsheng Xu	Chinese University of Hong Kong, China
Huayong Yang	Zhejiang University, China
Jie Zhao	Harbin Institute of Technology, China
Nanning Zheng	Xi'an Jiaotong University, China
Xiangyang Zhu	Shanghai Jiao Tong University, China

KGGPT: Empowering Robots with OpenAI's ChatGPT and Knowledge Graph

Zonghao Mu[1,2], Wenyu Zhao[1,2], Yue Yin[1,2], Xiangming Xi[1,2], Wei Song[1,2](✉), Jianjun Gu[1,2], and Shiqiang Zhu[1,2]

[1] Research Center for Intelligent Robotics, Research Institute of Interdisciplinary Innovation, Zhejiang Lab, Hangzhou 311100, China
[2] Zhejiang Engineering Research Center for Intelligent Robotics, Hangzhou 311100, China
{muzonghao,wyuzhao,yinyue,xxm21,weisong,jgu,zhusq}@zhejianglab.com

Abstract. This paper presents a study on using knowledge graph with ChatGP for robotics applications, called KGGPT. Traditional planning methods for robot tasks based on structured data and sequential actions, such as rosplan, have limitations such as limited data range and lack of flexibility to modify behaviors based on user feedback. Recent research has focused on combining AI planning with large language models (LLMs) to overcome these limitations, but generated text may not always be consistent with real-world physics and the robot skills to perform physical actions. To address these challenges, we propose KGGPT, a system that incorporates prior knowledge to enable ChatGPT for a variety of robotic tasks. KGGPT extracts relevant knowledge from the knowledge graph, generates a semantic description of the knowledge, and connects it to ChatGPT. The gap between the knowledge of ChatGPT and actual service environments is addressed by using the knowledge graph to model robot skills, task rules, and environmental constraints. The output is a behavior tree based on robot skills. We evaluate our method in an office setting and show that it outperforms traditional PDDL planning and a separate ChatGPT planning scheme. Additionally, our system reduces programming effort for applications when new task requirements arise. This research has the potential to significantly advance the field of robotics.

Keywords: ChatGPT · Knowledge graph · AI planning

1 Introduction

Robotic task planning is an important aspect of artificial intelligence with a wide range of applications in industry and services. However, traditional planning approaches based on structured data and sequential operations (e.g. Rosplan

Supported by "Pioneer" and "Leading Goose" R&D Program of Zhejiang (2022C01130) and Key Research Project of Zhejiang Lab (No. G2021NB0AL03).
Z. Mu and W. Zhao—Contribute equally to this work.

H. Yang et al. (Eds.): ICIRA 2023, LNAI 14271, pp. 340–351, 2023.
https://doi.org/10.1007/978-981-99-6495-6_29

[2]) have several limitations, including limited data range and lack of flexibility to modify behaviors based on user feedback. Recent researches [1,14] has focused on overcoming these limits with LLMs can flexibly generate lists of key actions required to complete tasks, opening up the possibility of building general-purpose robot intelligence systems. However, the text generated by large language models may not always be consistent with real-world physics and the robot skills to perform physical actions. In this paper, we propose a system called KGGPT to bridge the gap between generated text and actual task execution. As shown in Fig. 1, the entire process of KGGPT can be divided into three stages:

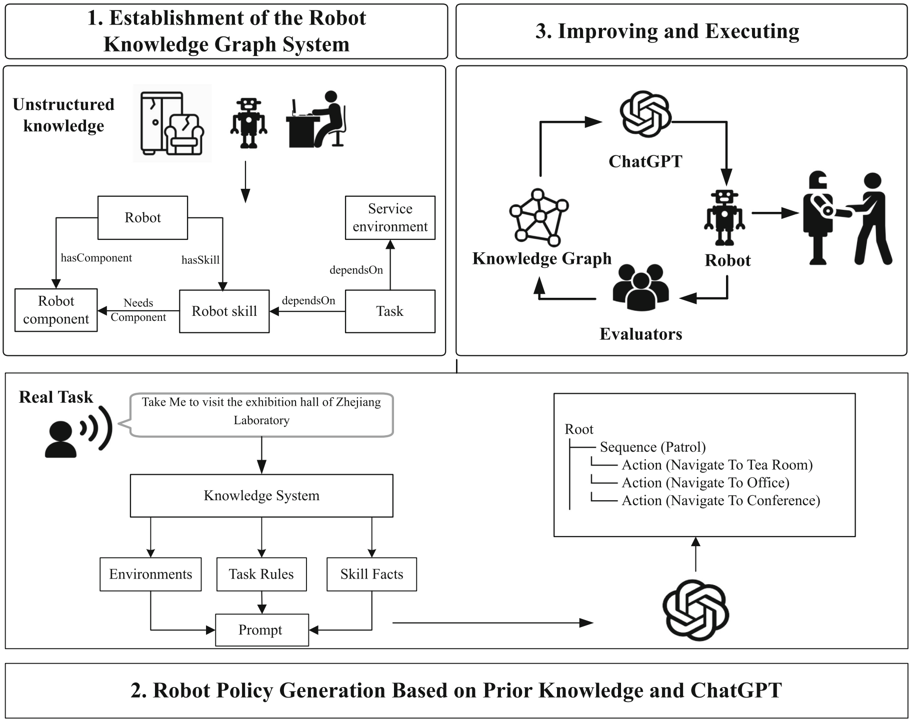

Fig. 1. Overview of KGGPT

- Establishment of the robot knowledge graph system: The system organizes unstructured knowledge into a knowledge graph, including robot, robot component, robot skills, service environment, and task.
- Robot Policy Generation Based on Prior Knowledge and ChatGPT: Convert the task-oriented knowledge extracted from the knowledge graph into appropriate semantic hints, enabling ChatGPT to plan tasks effectively and efficiently. The output is a behavior tree with robot skills as nodes.
- Improvement and execution: During the test process, according to the feedback of the applicable person, modify the task rules in the knowledge graph, so as to improve the planning effect until the requirements are met.

Therefore, KGGPT can make full use of the robot skills and task rules to constrain the task strategy generated by ChatGPT, thus handling various real tasks.

The contributions of this paper mainly include the following three aspects:

– Development of a knowledge graph, including robot, robot component, robot skills, service environment, and task.
– Provide a new way to design general-purpose robot task planning.
– Provision of a scheme to continuously improve the execution logic by updating the knowledge graph.

In summary, our study proposes KGGPT, a system for robotic task planning that combines ChatGPT with knowledge graph. Our experimental results demonstrate the effectiveness of KGGPT in solving robotic tasks, making a significant contribution to the field of robotics.

2 Related Work

2.1 Large Language Models for Robotic Tasks

In recent years, several attempts have been made to incorporate LLMs into robotic systems. For example, [1] uses LLMs to compute value functions that rank the best action types in a robot-specific library using free-form text commands. Other studies have explored the use of LLMs in zero-shot advanced robotic task planning [5,7,14]. These approaches rely on a prompt structure containing predefined functions, behaviors, and examples to guide the answers generated by the model.

However, the pre-training knowledge induced in LLMs can lead to spurious associations between goals and steps, as robotic systems require a deeper understanding of real-world physics, environments, and the skills to perform physical actions. To address this, [8] introduces a Neural Symbolic Program Planner that extracts program planning knowledge from LLMs by incorporating commonsense cues. Additionally, [17] allows users to interactively optimize and correct the plan, rather than modifying the prompt from scratch to generate another zero-shot response. These approaches rely on prior knowledge to modify the planning of the LLMs.

Overall, the use of prior knowledge and commonsense cues can improve the accuracy and effectiveness of LLM-based planning, providing a new way to design general-purpose robot task planning.

2.2 Knowledge Graph and Robotic Task Planning

In the field of robotics, prior knowledge can be expressed through knowledge graph, and there have been many studies. Research on knowledge graphs such as RoboEarth [18], KnowRob [15], and RoboBrain [12] has yielded outstanding results, highlighting the growing importance of knowledge methodologies.

Therefore, it is very common to use knowledge graph to organize the information required for robot task planning. Previous research has attempted to use knowledge graph to store the underlying task planning and semantic representation of robots, including object affordances [16], robot action trees [11], and dynamic capabilities [9]. This knowledge can then guide planning modules, such as the Model of PDDL structure [6] and neural networks [3], in planning new tasks. Recent experiments that combine task planning and knowledge graph have demonstrated the effectiveness of incorporating knowledge in robotic tasks [4,10].

In the task planning process, it is critical to understand what the task needs to do and what the robot is capable of doing. It seems feasible to infer from the robot's components whether it can perform the task. But it is tedious to infer robot tasks directly from components because components vary between robots. To address this problem, modeling tasks as a set of basic skills to be performed and reasoning about the availability of the modeling [19] or potential representations of these skills [13] has proven successful in integrating planning algorithms and generalizing to various task domains. In this study, an ontology approach to modeling robot tasks, skills, and components is used to enable the task planning module to understand the robot's capabilities better and thus improve the effectiveness of task planning.

3 Method

The workflow of KGGPT consists of three stages, 1) Establishment of Robotic Knowledge Graph system, 2) Robot Policy Generation Based on Prior Knowledge and ChatGPT, and 3) Improving and Executing.

3.1 Establishment of Robotic Knowledge Graph System

When planning for a task, the task planning system needs to understand not only what it needs to do in the servicing environment but also what it is capable of doing. Therefore, we use a task-oriented ontology, including a multi-layered framework: robot, robot component, robot skills, service environment, and task to guide task planning with the related knowledge. As shown in Fig. 2, in our system, the completion of robot tasks depends on the robot skills and the service environment. Task class semantic descriptions are independent of the robot class and are adapted to robot component classes through robot skills classes. The following sections provide a detailed description of the ontology of the robot, robot component, robot skills, service environment, and task.

Robot and Robot Component. The robot ontology defines the attributes of the robot, such as name, dimensions, and so on. The robot component ontology is defined to describe all that a robot contains, including hardware and software. Robot components are represented in the knowledge graph as instances of component classes to capture their component types and specific robots.

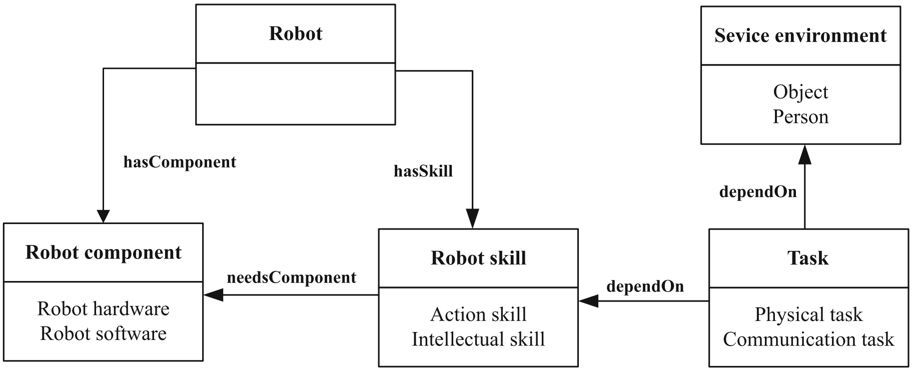

Fig. 2. The Ontology of KGGPT

Robot Skills. Robots and their components are specific, and different robots may require different components to perform the same task. There are many limitations if each component is a necessary element in the task planning process. Thus, robot skills are needed to serve as a bridge between robot components and tasks. Robot skills represent the ability of a robot to perform certain tasks. The ontology of robot skills is crucial. The ability of a robot to perform a task in a given situation depends on its skills. A task requires a set of skills, while a set of components gives the robot certain skills that help us to plan tasks more flexibly without the limitations of the robot software and hardware.

Service Environment. The service environment ontology is designed to describe people, objects, and their relationships in the scene, to provide support for task planning. In the process of task execution, the robot not only needs to perceive the items in the environment but also needs to identify the service object. For example, for service robots, map information is very important in the process of completing tasks. Map information is required for many robot actions, such as moving, manipulating objects, and finding objects. Therefore, we semantically describe the space information in the map, mainly including the positional relationship between locations and the coordinates of the locations.

Task. The tasks that robots can complete are mainly divided into two categories: communication tasks and physical tasks. A communication task is a task in which two or more agents exchange information. Physical tasks include the tasks in which a Physical Agent affects some physical object, such as Interacting, Manipulating, Navigating, and so on.

The task ontology semantically describes the relevant content and execution rules of robot tasks. In our knowledge system, task ontology has a mutual relationship with robot ontology to define and reflect the skills information required in the task. In addition to skills, tasks also depend on the service environment. The dependency relationship is specified using the object attribute "dependOn".

By using these relationships in knowledge reasoning, complex knowledge can be generated and provided to determine which robots can complete specific tasks.

In addition, the knowledge graph also includes logical rules for task execution. These rules enhance the ability of the task planner to derive the corresponding optimal plan when determining that a specific robot can complete a task. Once it is determined that a robot has the skills to complete a task, this information will be extracted and used to help the task planning system.

3.2 Robot Policy Generation Based on Prior Knowledge and ChatGPT

To plan a robot task, KGGPT needs to extract relevant knowledge from the knowledge graph and input it into ChatGPT. This requires converting structured knowledge into a suitable semantic description to facilitate ChatGPT's understanding and planning. KGGPT extracts the required knowledge from the three types of knowledge related to the robot, task, and service environment mentioned above, and semantically describes them in the form of facts and rules.

The robot knowledge graph provides the semantic description, code specification, and robot components on which the skills of the robot are based, which are stated as known facts. Robot skills has their own limitations, such as their scope of use and necessary components required, which are described as rules. The task knowledge graph provides the semantic description of the task, the list of parameters required for task execution, and the task dependencies that define the prerequisite tasks required for execution. The rules mainly impose common constraints on the completion of tasks. The service environment knowledge includes various objects and their attributes that robots can visually perceive in the service environment.

The output of ChatGPT task planning is generally semantics or code. Semantic methods are not suitable for robots to understand, while code methods are difficult for humans to understand, and it is not easy to handle tasks that require parallel operations. Behavior trees are a common method for robot task planning, which is easy for users to understand and can perform tasks. Therefore, KGGPT requires ChatGPT to output skills as behavior tree nodes, enabling the system to plan tasks effectively and efficiently, ensuring that the robot can understand and complete tasks smoothly.

Figure 3 provides an example of the notification policy generation process in the office scenario, demonstrating the skills to generate robot task processes and content. In this scenario, the user commands the robot to notification Sam that he can take leave tomorrow, which falls under the notification task type. For the notification task, the robot must have two skills: search for people and communicate. The system extracts the semantics of the notification task from the knowledge graph, along with the general rules of tasks and skills. Then, based on the skills requirements, the system searches for component information, evaluates the feasibility of the task and sends it to ChatGPT.

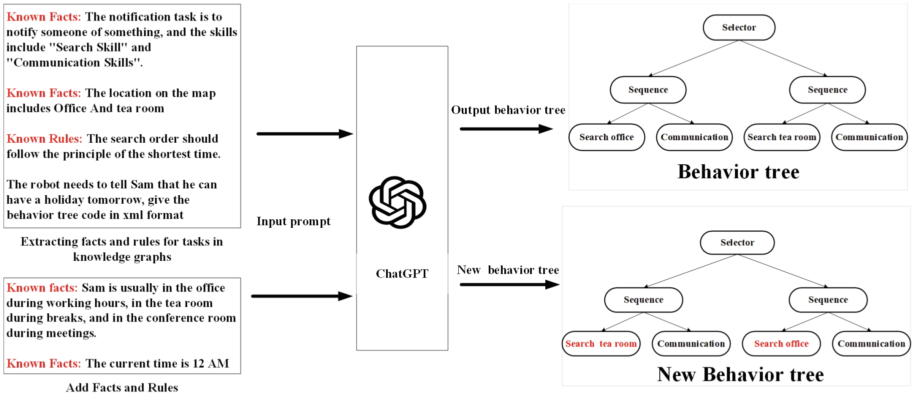

Fig. 3. Notification policy generation process

3.3 Improving and Executing

During actual testing, various practical problems may arise, but unlike traditional task processes that require recoding, KGGPT can adapt to these problems by changing the knowledge in the knowledge graph and generating new solutions accordingly. As shown in Fig. 3, we add two known facts: Sam is usually in the office during working hours, in the tea room during breaks, and in the conference room during meetings, while the current time is 10 AM. Using this information, the regenerated behavior tree will search the office first and the tea room last. This approach can be used as a strategy by designers to improve the efficiency and effectiveness of task planning. In the next chapter, we conduct method tests and comparisons to evaluate the performance of KGGPT in handling robotic tasks.

4 Experiments and Results

To demonstrate the effectiveness of our proposed strategy for improving robotic systems, we evaluate it on real robots, as shown in Fig. 4. To evaluate the effectiveness of our method, we design three different types of tasks: patrolling, notification and delivery. We compare our method with rosplan-based planning methods (which is a ROS package that includes PDDL-based planning methods) and proposals generated directly based on ChatGPT (input task description and given the robot's API), all of which are under the same task information conditions. We evaluate these methods in terms of success rate and time required to complete the task.

4.1 Patrol Task

We design a patrol task to evaluate the effectiveness of our proposed method, which requires the robot to individually visit all specified points on the map

Fig. 4. The robot in our system is equipped with a mobile base, two 6-DOF arms, a depth camera, and a speaker.

within a given time limit for the task to be considered successful. To make the task more challenging, we randomize the initial position of the robot, which means that the robot has to prioritize the remaining points for patrolling according to the current situation.

Table 1. Patrol task success rate and average time

Project	Approach	Success Rate	Avg. Time (s)
Patrol	Our approach	100%	55
	rosplan	100%	72
	Only ChatGPT	100%	75

In the patrol task, the general facts of the task include its current location, and the coordinates of the marked points, and our method obtains the main rule from the knowledge graph is to complete the patrol as quickly as possible, and the shortest path principle can be considered. Our proposed method exploits common sense in large language models and combines the positions of multiple markers with the robot's current position, as shown in Fig. 5. In contrast, the solutions provided by rosplan and ChatGPT patrol according to the order of the given points on the map, regardless of the location of the robot. This is because rosplan does not have the function of planning the navigation order according to the length of the path, but only in a predetermined order. ChatGPT, on the other hand, does not emphasize time and only treats it as a secondary factor. As shown in Table 1, although all tasks are completed within the specified time, it is clear that our method outperforms other solutions in terms of time efficiency.

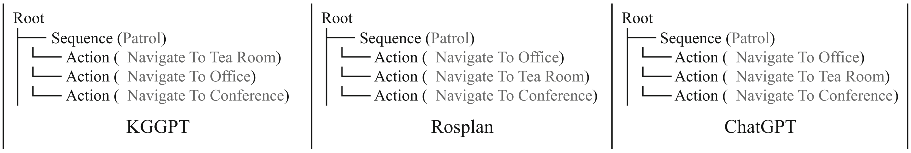

Fig. 5. The nearest point of the robot's real-time position is the tea room, which we use as the starting point of the patrol, and give priority to unvisited marked points according to the shortest path principle.

4.2 Notification Task

Another task we designed to evaluate our proposed strategy is the notification task, in which a robot is assigned to notify a specific individual of a specific event within a specified time frame. To add to the difficulty of the task, we do not provide the exact location of the person, only possible locations. Therefore, the robot must adjust its search strategy according to the current situation.

Table 2. Notification task success rate and average time

Project	Approach	Success Rate	Avg. Time (s)
Notification	Our approach	100%	38
	rosplan	100%	58
	Only ChatGPT	70%	35

In this task, the known facts of the task include the possible location of the human being, such as in the office, tea room, or conference room, and the time when the task starts. What our method obtains from the knowledge graph includes the people are usually in the office during working hours and in the tea room during breaks.

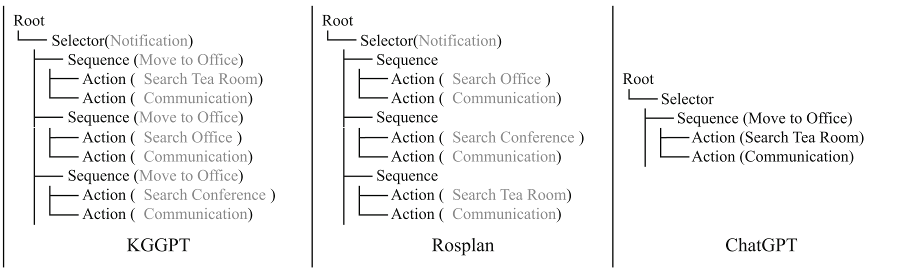

Fig. 6. The current time is 10:00 am, our scenario starts the search in the office, and if the person cannot be found, continues the search in other rooms.

Our method prioritizes search locations according to the current time, as shown in Fig. 6. On the other hand, the solution provided by Rosplan searches each room in turn and does not take into account the most likely location of the person. As a result, the search takes a long time, and some tasks may fail. The solution given by ChatGPT only searches the most likely room. If no one is found in that room, it will not search in other areas. As shown in Table 2, our proposed method outperforms other solutions in terms of time efficiency and task completion rate.

4.3 Delivery Tasks

The last task we designed to evaluate our proposed policy was the delivery task, in which the robot was required to transport 4 drinking glasses from the office to the tea room within a specified time limit. To make the task more challenging, we randomly assign the weight of each item between 0–500 g, and the weight of each delivery of the robot is limited to 500 g. Since our robot does not have dexterous hands, human assistance is required for the grasping part.

Table 3. Delivery task success rate and average time

Project	Approach	Success Rate	Avg. Time (s)
Delivery	Our approach	100%	138
	rosplan	0%	521
	Only ChatGPT	80%	228

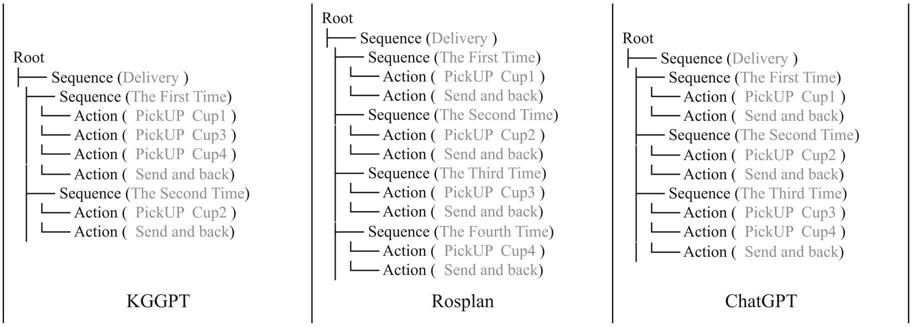

Fig. 7. In this example the office has the following cups: Cup No. 1 (230 g), Cup No. 2 (430 g), Cup No. 3 (130 g), Cup No. 4 (30 g). Our method delivered cups 1, 3, and 4 for the first time, and cup 2 for the second time, significantly outperforming the other methods

In this task, the known facts include the weight of each drinking glass and the total mass of each delivered item, which cannot exceed 500 g. Our method extracts the rule required to complete delivery tasks from the knowledge graph is that the robot can carry more than one cup at a time and should carry as much weight as possible. To complete the delivery task, KGGPT employs a greedy algorithm to select the delivery method with the least number of times, as shown in Fig. 7. In contrast, rosplan does not have its own plan, and the water glasses are delivered one by one, which leads to exceeding the time limit. ChatGPT delivers the drinking glasses in the given order until the weight limit of 500g is reached. As shown in Table 3, our method outperforms both of these methods in terms of success rate and time required to complete the task.

5 Conclusion

This paper presents KGGPT, a system that uses a knowledge graph to link LLMs with actual robotic tasks. LLMs act as the robot's brain and plan tasks into a behavior tree composed of multiple skills nodes based on the robot and actual scene information. The gap between the knowledge of ChatGPT and actual service environments is addressed by using the knowledge graph to model robot skills, task rules, and environmental constraints. Experimental results demonstrate that KGGPT outperforms conventional task planning algorithms and pure ChatGPT planning effects in handling robot tasks. Our design can inspire the entire robotics industry and open a new path toward general-purpose robotic intelligence. In the future, we hope to explore the use of ChatGPT for multimodal information understanding and task planning to achieve general robot intelligent operation.

References

1. Brohan, A., et al.: Do as i can, not as i say: grounding language in robotic affordances. In: Conference on Robot Learning, pp. 287–318. PMLR (2023)
2. Cashmore, M., et al.: ROSPlan: planning in the robot operating system. In: Proceedings of the International Conference on Automated Planning and Scheduling, vol. 25, pp. 333–341 (2015)
3. Daruna, A., Nair, L., Liu, W., Chernova, S.: Towards robust one-shot task execution using knowledge graph embeddings. In: 2021 IEEE International Conference on Robotics and Automation (ICRA), pp. 11118–11124. IEEE (2021)
4. Hanheide, M., et al.: Robot task planning and explanation in open and uncertain worlds. Artif. Intell. **247**, 119–150 (2017)
5. Huang, W., Abbeel, P., Pathak, D., Mordatch, I.: Language models as zero-shot planners: extracting actionable knowledge for embodied agents. In: International Conference on Machine Learning, pp. 9118–9147. PMLR (2022)
6. Kootbally, Z., Schlenoff, C., Lawler, C., Kramer, T., Gupta, S.K.: Towards robust assembly with knowledge representation for the planning domain definition language (PDDL). Robot. Comput.-Integr. Manuf. **33**, 42–55 (2015)
7. Liang, J., et al.: Code as policies: language model programs for embodied control. arXiv preprint arXiv:2209.07753 (2022)
8. Lu, Y., et al.: Neuro-symbolic procedural planning with commonsense prompting. arXiv preprint arXiv:2206.02928 (2022)

9. Munawar, A., et al.: MaestROB: a robotics framework for integrated orchestration of low-level control and high-level reasoning. In: 2018 IEEE International Conference on Robotics and Automation (ICRA), pp. 527–534. IEEE (2018)
10. Nyga, D., et al.: Grounding robot plans from natural language instructions with incomplete world knowledge. In: Conference on Robot Learning, pp. 714–723. PMLR (2018)
11. Puig, X., Ra, K., Boben, M., Li, J., Wang, T., Fidler, S., Torralba, A.: VirtualHome: simulating household activities via programs. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 8494–8502 (2018)
12. Saxena, A., et al.: RoboBrain: large-scale knowledge engine for robots. arXiv preprint arXiv:1412.0691 (2014)
13. Silver, T., Athalye, A., Tenenbaum, J.B., Lozano-Perez, T., Kaelbling, L.P.: Learning neuro-symbolic skills for bilevel planning. arXiv preprint arXiv:2206.10680 (2022)
14. Singh, I., et al.: ProgPrompt: generating situated robot task plans using large language models. arXiv preprint arXiv:2209.11302 (2022)
15. Tenorth, M., Beetz, M.: KNOWROB-knowledge processing for autonomous personal robots. In: 2009 IEEE/RSJ International Conference on Intelligent Robots and Systems, pp. 4261–4266. IEEE (2009)
16. Varadarajan, K.M., Vincze, M.: AfRob: the affordance network ontology for robots. In: 2012 IEEE/RSJ International Conference on Intelligent Robots and Systems, pp. 1343–1350. IEEE (2012)
17. Vemprala, S., Bonatti, R., Bucker, A., Kapoor, A.: ChatGPT for robotics: design principles and model abilities (2023)
18. Waibel, M., et al.: Roboearth. IEEE Robot. Autom. Mag. **18**(2), 69–82 (2011)
19. Xu, D., Mandlekar, A., Martín-Martín, R., Zhu, Y., Savarese, S., Fei-Fei, L.: Deep affordance foresight: planning through what can be done in the future. In: 2021 IEEE International Conference on Robotics and Automation (ICRA), pp. 6206–6213. IEEE (2021)