Kinematic-aware Prompting for Generalizable Articulated Object Manipulation with LLMs

Wenke Xia[1,2,*,†], Dong Wang[2,*], Xincheng Pang[1], Zhigang Wang[2], Bin Zhao[2,3], Di Hu[1,‡], Xuelong Li[2,4,‡]

Abstract— Generalizable articulated object manipulation is essential for home-assistant robots. Recent efforts focus on imitation learning from demonstrations or reinforcement learning in simulation, however, due to the prohibitive costs of real-world data collection and precise object simulation, it still remains challenging for these works to achieve broad adaptability across diverse articulated objects. Recently, many works have tried to utilize the strong in-context learning ability of Large Language Models (LLMs) to achieve generalizable robotic manipulation, but most of these researches focus on high-level task planning, sidelining low-level robotic control. In this work, building on the idea that the kinematic structure of the object determines how we can manipulate it, we propose a kinematic-aware prompting framework that prompts LLMs with kinematic knowledge of objects to generate low-level motion trajectory waypoints, supporting various object manipulation. To effectively prompt LLMs with the kinematic structure of different objects, we design a unified kinematic knowledge parser, which represents various articulated objects as a unified textual description containing kinematic joints and contact location. Building upon this unified description, a kinematic-aware planner model is proposed to generate precise 3D manipulation waypoints via a designed kinematic-aware chain-of-thoughts prompting method. Our evaluation spanned 48 instances across 16 distinct categories, revealing that our framework not only outperforms traditional methods on 8 seen categories but also shows a powerful zero-shot capability for 8 unseen articulated object categories with only 17 demonstrations. Moreover, the real-world experiments on 7 different object categories prove our framework's adaptability in practical scenarios. Code is released at https://github.com/GeWu-Lab/LLM_articulated_object_manipulation.

I. INTRODUCTION

Generalizable articulated object manipulation is imperative for building intelligent and multi-functional robots. However, due to the considerable heterogeneity in the kinematic structures of objects, the manipulation policy might vary drastically across different object instances and categories. To ensure consistent performance in automated tasks within intricate real-world scenarios, prior works on generalizable object manipulation have been devoted to imitation learning from demonstrations [1], [2] and reinforcement learning in simulation [3], [4]. As shown in Figure 1(a), these approaches consistently require substantial amounts of robotic data. To bolster the manipulation generalization, recent works [5], [6] have made notable advancements in curating extensive robotic datasets. However, the diversity of

[1]Gaoling School of Artificial Intelligence, Renmin University of China
[2]Shanghai Artificial Intelligence Laboratory
[3]Northwestern Polytechnical University
[4]Institute of Artificial Intelligence, China Telecom Corp Ltd
*Equal contribution,†Work is done during internship at Shanghai Artificial Intelligence Laboratory, ‡Corresponding author
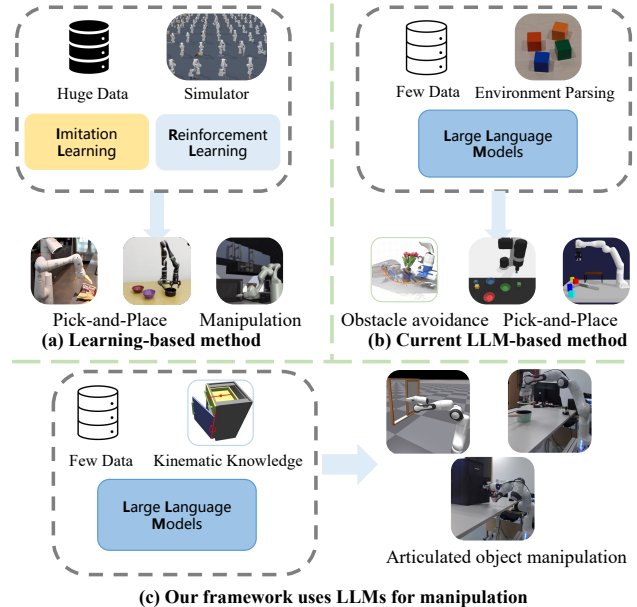
Fig. 1: As depicted in (a), traditional learning-based methods rely on vast datasets for broad manipulation tasks. Recent studies in (b) harness LLMs to reduce data reliance, but primarily apply to elementary challenges like obstacle avoidance, and pick-and-place. In contrast, our framework, highlighted in (c), achieves zero-shot articulated object manipulation with the kinematic-aware prompting method.

scenarios covered in these datasets is still limited. To achieve generalization across various scenarios, there would be a prohibitive cost of accumulating more data. Thus, contemporary studies strived to mitigate this excessive data dependency through strategies such as efficient architectures [2], [7], redefined training objectives [1], [8], and data augmentation techniques [9], [10]. Nevertheless, the generalizability of these solutions remains limited, especially in novel scenarios that contain unseen object states.

In pursuit of developing a generalizable object manipulation policy with reduced data reliance, recent works incorporated rich world knowledge within LLMs to promote policy learning [11], [12]. Drawing insights from the reasoning capabilities of LLMs [13], [14], as shown in Figure 1(b), some contemporary studies [15], [16] successfully used LLMs to predict reasonable action sequences for object manipulation with a parsed textual representation of environments. However, such a coarse representation limits these works to simple tasks like pick-place and rearrangement planning. Although Huang et al. [17] introduced a 3D voxel map to capture intricate environment details and tackle challenges

like obstacle avoidance, there is still a notable gap in handling sophisticated articulated object manipulation tasks.

In this work, we delve into the problem of harnessing LLMs for generalizable articulated object manipulation, recognizing that the rich world knowledge inherent in LLMs is adept at providing reasonable manipulation understanding of various articulated objects. For instance, when presented with the instruction "open the cabinet", LLMs can provide a systematic approach: 1) Locate the handle or knob, 2) Determine the hinge direction, and 3) Either push or pull based on the hinge type. However, to fully leverage the rich world knowledge within LLMs for precise articulated object manipulation, we still confront the critical challenge of converting these abstract manipulation commonsense into precise low-level robotic control.

To tackle the aforementioned challenge, we propose a kinematic-aware prompting framework. This framework is designed to extract the kinematic knowledge of various objects and prompt LLMs to generate low-level motion trajectory waypoints for object manipulations as shown in Figure 1(c). The idea behind this method is that the kinematic structure of an object determines how we can manipulate it. Therefore, we first propose a unified kinematic knowledge parser, which represents the various articulated objects as a unified textual description with the kinematic joints and contact location. Harnessing this unified description, a kinematic-aware planner is proposed to generate precise 3D manipulation waypoints for articulated object manipulation via a kinematic-aware chain-of-thought prompting. Concretely, it initially prompts LLMs to generate an abstract textual manipulation sequence under the kinematic structure guidance. Subsequently, it takes the generated kinematic-guided textual manipulation sequence as inputs, and outputs 3D manipulation trajectory waypoints via in-context learning for precise robotic control. With this kinematic-aware hierarchical prompting, our framework can effectively utilize LLMs to understand various object kinematic structures to achieve generalizable articulated object manipulation.

To validate the efficacy of our framework, we conduct exhaustive experiments on 48 objects across 16 categories in Isaac Gym [18] simulator and extend our method to real-world scenarios. The results prove that our framework could generalize across seen/unseen object instances and categories in a zero-shot context. Moreover, the real-world experiments prove our framework's ability to extend its generalization to practical scenarios.

The main contributions of our work are as follows:

- We propose the Kinematic-aware prompt framework, aiming for generalizable articulated object manipulation across novel instances and categories with minimal robotic data requirements.
- We design the unified kinematic knowledge parser and kinematic-aware planner components, utilizing the kinematic knowledge to prompt LLMs to generate precise 3D manipulation trajectory waypoints.
- We evaluate our method on 48 instances across 16 categories. The results prove our framework exhibits

zero-shot ability for articulated object manipulation. The real-world experiments also prove our framework's generalization to practical scenarios.

II. RELATED WORKS

A. Policy Learning for object manipulation

Object manipulation policy learning methods have primarily focused on imitation learning from demonstrations [1], [5], [19] and reinforcement learning [3] in simulations. To devise a practical manipulation policy for specific scenarios, these methods [5]–[7], [20] often rely on numerous demonstrations or episodes. To reduce the cost of data collection, some researchers [9], [21]–[23] employ generative models to augment the limited robotic data for robotic policy training. In addition to innovations in dataset augmentation, recent works [1], [2] seek to bolster the learning efficacy of models. Leveraging the transformer architecture, numerous studies have showcased efficient policy learning in a limited demonstration dataset [2], [24]. Moreover, Jia et al. [1] incorporates the idea of hierarchical reinforcement learning with imitation learning for generalizable policy learning. However, these studies still face challenges in unseen scenarios when training with a limited dataset. In this work, we propose a kinematic-aware prompting framework that guides LLMs to generate low-level motion trajectory waypoints with the object kinematic knowledge, thereby facilitating a more generalized approach to articulated object manipulation with minimal reliance on robotic demonstrations.

B. Large Language Models for Robotics

Motivated by the rich world knowledge exhibited by LLMs, recent literature has explored the integration of LLMs with robotics across various domains [25]–[27]. To enable robots to adapt to complex real-world scenarios, many works [28]–[38] focus on task planning and the decomposition of complex instructions. Although these studies exhibit superior planning ability to decompose complex unseen instructions into subgoals, inevitably, they still depend on a pre-trained skill library for the fulfillment of subgoals. However, this dependence poses challenges due to the scarcity of extensive robotic datasets for learning various skills. To improve this skill acquisition process, some researchers [39], [40] employ LLMs for reward designing. Moreover, Mirchandani et al. [15] encodes actions into separate tokens and leverages LLMs to generate corresponding token sequences for robotic control through in-context learning. While, Huang et al. [17] and Zhao et al. [16] construct the environment information to prompt LLMs to produce action sequences for manipulation. However, these works primarily focus on elementary manipulation tasks such as obstacle avoidance and pick-and-place tasks, exhibiting shortcomings in the manipulation of complex articulated objects. To harness the full potential of LLMs for articulated object manipulation, we extract object kinematic knowledge to prompt LLMs to generate precise 3D manipulation waypoints, and achieve zero-shot manipulation for articulated objects across novel instances and categories.
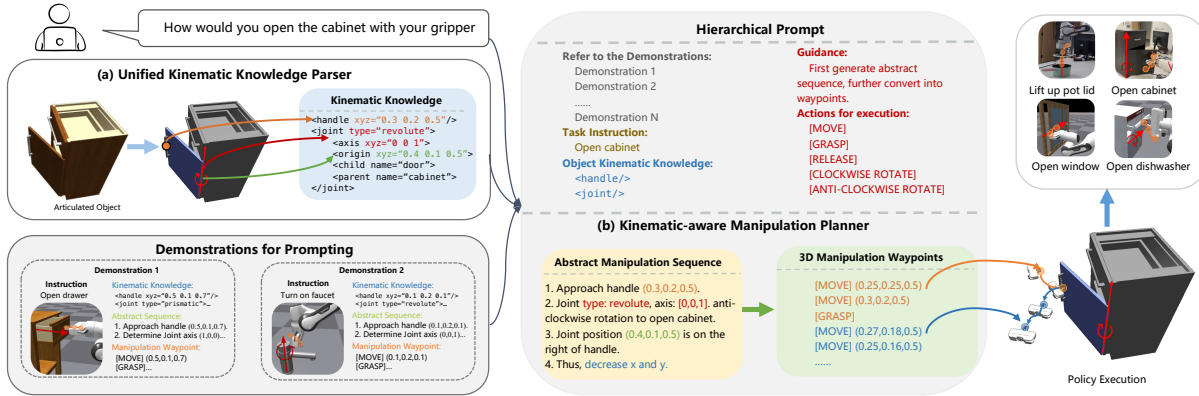
Fig. 2: We first propose the Unified Kinematic Knowledge Parser component to grasp the object's kinematic structure as a kinematic knowledge description for LLMs as shown in (a). Based on the description, we construct a kinematic-aware hierarchical prompt, which is used in the Kinematic-aware Manipulation Planner component to guide LLMs to generate an abstract textual manipulation sequence, and 3D manipulation waypoints for generalizable articulated object manipulation in (b). Distinct colors assigned to numbers represent the properties of the different kinematic structure components.

III. METHOD

We aim to solve generalizable articulated object manipulation problems that require kinematic and geometric reasoning of objects to generate precise manipulation policy. As shown in Figure 2, the proposed kinematic-aware prompting framework is composed of two modules: Unified Kinematic Knowledge Parser and Kinematic-aware Manipulation Planner. Given the manipulation instruction $\mathbf{I}$, the unified kinematic knowledge parser (Sec. III-A) extracts the kinematic structure of various articulated objects $\mathbf{O}$ and formats it as a unified kinematic knowledge description $\mathbf{K}$. Then, the kinematic-aware manipulation planner (Sec. III-B) incorporates this kinematic description with few manipulation demonstrations to prompt LLMs to generate a sequence of 3D manipulation waypoints $\mathbf{W} = \{w_1, w_2, .., w_n\}$ for manipulation. Finally, the manipulation policy is executed via waypoints following a traditional motion planning algorithm.

A. Unified Kinematic Knowledge Parser

The manipulation policy of an articulated object is mostly determined by its kinematic structure [41], [42], and the strong complex reasoning capability of LLMs presents a promising pathway to general kinematic structure understanding. For manipulating objects with different kinematic structures, a unified and effective kinematic knowledge description is essential for LLMs to understand various articulated objects and subsequently generate manipulation policy. Thus, as depicted in Figure 2(a), we propose the unified kinematic knowledge parser component to represent the articulated object as a unified textual kinematic description for kinematic structure understanding. Concretely, the unified kinematic knowledge parser consists of two distinct steps.

First, we detect and segment the geometric and kinematic structures of a given articulated object via an off-the-shelf model. In the simulator, we could directly extract this information. While, in the real world, we rely on existing perception models [41], [43] for joint estimation to obtain this information. The output of this step consists of geometric-linked parts, kinematic joints between parts, and a contact point for the manipulation. These part segments, joint properties, and the contact point represent the kinematic structure of this object and determine how to manipulate it.

Inspired by the remarkable aptitude of LLMs in parsing structured textual data [38], ultimately, we translate the kinematic joint structure and contact location into a unified structured .xml format. Using the proposed unified kinematic knowledge parser, the complex kinematic structure of different articulated objects can be easily understood by LLMs via this code-like textual kinematic knowledge description $\mathbf{K}$.

B. Kinematic-aware Manipulation Planner

To enable LLMs to generate precise 3D waypoints for articulated object manipulation, we propose a hierarchical prompting method named kinematic-aware planner component, to prompt the LLMs with the unified kinematic knowledge description $\mathbf{K}$ and the manipulation instruction $\mathbf{I}$. Concretely, the hierarchical prompting method is composed of manipulation sequence planning and manipulation way-points generation, achieving abstract textual manipulation planning to precise manipulation waypoints conversion via chain-of-thoughts prompting.

Manipulation Sequence Planning. As shown in Figure 2, given an articulated object $\mathbf{O}$, a manipulation command specified by textual instruction $\mathbf{I}$, and its kinematic description $\mathbf{K}$ from unified kinematic knowledge parser. First, the manipulation instruction $\mathbf{I}$ and kinematic description $\mathbf{K}$ are concatenated as the header of the prompting. Then, we prompt LLMs with this concatenated text to produce an abstract textual manipulation sequence $\mathbf{A}$.

As demonstrated in the textual manipulation sequence of Figure 2(b), our hierarchical prompt first promotes LLMs to generate an abstract textual manipulation sequence. To generate a kinematic-feasible manipulation sequence, we prompt the LLMs to pay attention to the referred kinematic
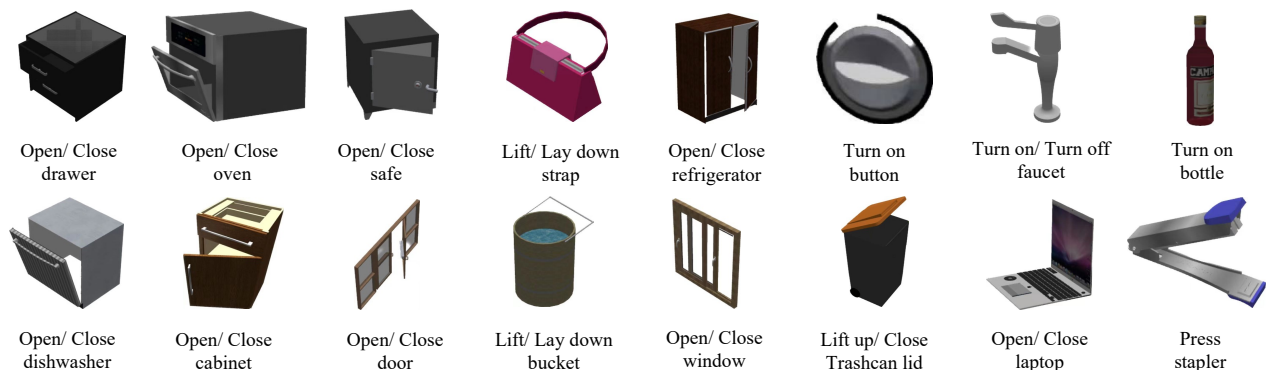
Fig. 3: The illustration of the articulated objects used in our experiments. Each of these entities corresponds to either a singular or a pair of manipulation instructions.

components in kinematic representation **K** such as handle and joint. To align the abstract manipulation plan with the concrete object kinematic knowledge, we make LLMs copy the corresponding properties (*e.g.,* coordinate and joint orientation) of referred kinematic components into the textual manipulation sequence. With this explicit alignment between kinematic knowledge and abstract manipulation sequences, we could incorporate the rich world knowledge of LLMs into 3D spatial reasoning for articulated object manipulation.

Manipulation Waypoints Generation. Following the generated textual manipulation sequence, the kinematic-aware planner then produces a sequence of 3D waypoints for precise robotic control. To apply the generated waypoints for various manipulation tasks, we provide five basic actions for LLMs to control the end-effector as follows: (1) move: move the gripper to the target position. (2) grasp: close the gripper. (3) release: open the gripper. (4) clockwise rotate: clockwise rotate the gripper by 30 degrees. (5) anti-clockwise rotate: anti-clockwise rotate the gripper by 30 degrees. As shown in Figure 2(b), LLMs could understand the 3D spatial information and generate waypoints with the formatted commands referring to the textual manipulation sequence via in-context learning. Further, utilizing an operational space controller, we can transform the output of the LLMs into low-level robotic control actions. With these explicit prompting between abstract manipulation sequences to precise 3D manipulation waypoints, the LLMs show a strong 3D spatial reasoning capability to generate reasonable 3D waypoints to manipulate various articulated objects.

IV. EXPERIMENTS

A. Experiment Setting

In this work, we propose the kinematic-aware prompting method for zero-shot articulated object manipulation. To comprehensively evaluate the generalization capability of our framework, we first conduct experiments within the Isaac Gym simulator [18], utilizing 48 distinct object instances across 16 types of articulated objects from the PartNet-Mobility dataset [44]. As shown in Figure 3, our evaluation dataset contains a broad spectrum of commonplace articulated objects, which covers the diversity of manipulation

policies and articulated structures. To enhance the scale of our evaluation data, we devised two opposite instructions for many object categories, like open/close window, open/close oven, lift/lay down bucket, turn on button, etc. In experiments, we provide the performance for all these manipulation instructions, and the order is consistent with Figure 3.

To guide LLMs in generating 3D manipulation waypoints with the object kinematic knowledge, we collect 17 3D manipulation waypoints demonstrations across 8 distinct object categories for in-context learning in the kinematic-aware manipulation planner module. These demonstrations cover open/close drawers, open/close ovens, open/close safes, lift/lay down straps, open/close refrigerators, turn on buttons, turn on/turn off faucets, and turn on bottles. To further comprehensively measure the performance of different methods, we divide the dataset into two subsets. The first subset comprises objects from eight categories of provided demonstrations, but with diverse poses and instances. The second is devoted to object categories unseen from the demonstrations, which is more challenging for the LLMs' reasoning capability and commonsense. During the evaluation, we randomly place the object in a reachable position for the robotic arm.

In each experiment, we employ a simple operational space controller to follow generated 3D waypoints to manipulate objects in simulation, and each object category is evaluated thrice by randomly selected object instances, constrained by the API cost limitation. The results are reported by the Average Success Rate (ASR) of manipulation for assessing overall performance.

B. Comparison Experiments

We compare our method with other approaches:

- **LLM2Skill:** We implement LLM2Skill baseline as a variant of Code as Policy [38]. We predefined 18 action primitives that could finish both the demonstrated and novel instructions. Here, LLMs would determine the suitable action primitive given the detailed object kinematic knowledge.
- **LLM2Waypoints:** We implement this method as a naive attempt to directly output manipulation waypoints for articulated object manipulation without considering the kinematic knowledge.

	Seen Categories with Unseen Instances and Poses							
Methods	drawer	oven	safe	strap	refrigerator	button	faucet	bottle
LLM2Skill	100%/ 100%	66.6%/ 100%	33.3%/ 66.6%	0%/ 0%	66.6%/ 33.3%	100%	33.3%/ 33.3%	100%
LLM2Waypoints	100%/ 100%	100%/ 100%	100%/ 33.3%	33.3%/ 33.3%	100%/ 100%	100%	33.3%/ 33.3%	100%
BC	33.3%/ 33.3%	33.3%/ 33.3%	33.3%/ 66.6%	33.3%/ 66.6%	33.3%/ 66.6%	100%	33.3%/ 0%	33.3%
Ours	100%/ 100%	100%/ 100%	100%/ 100%	100%/ 100%	66.6%/ 100%	100%	66.6%/ 66.6%	66.6%

TABLE I: Evaluation results on seen categories objects. We use / to differentiate the model's performance on different manipulation commands for the same object.

	Unseen Categories with different instances and poses							
Methods	dishwasher	cabinet	door	bucket	window	trashcan	laptop	stapler
LLM2Skill	33.3%/ 33.3%	0%/ 33.3%	0%/ 0%	0%/0%	33.3%/ 0%	0% / 0%	0%/ 0%	0%
LLM2Waypoints	33.3%/ 0%	33.3%/ 33.3%	0%/ 33.3%	33.3%/ 0%	0%/ 0%	33.3%/ 0%	0%/ 0%	33.3%
BC	33.3%/ 33.3%	0%/33.3%	0%/ 33.3%	0%/ 0%	0%/ 0%	0%/ 33.3%	0%/ 33.3%	100%
Ours	66.6%/ 100%	66.6%/ 66.6%	66.6%/ 66.6%	66.6%/ 100%	66.6%/ 66.6%	100%/ 33.3%	66.6%/ 100%	100%

TABLE II: Evaluation results on unseen categories objects. We use / to differentiate the model's performance on different manipulation commands for the same object.

- **Behavior Cloning.** We train a language-conditioned behavior cloning algorithm on the demonstrations, leveraging the structure of Decision Transformer [24].

We systematically evaluate these methods on the divided two subsets, and the results are as follows:

Results on unseen instances and pose: We evaluate the methods on seen categories, but with different poses and instances. As illustrated in Table I, most LLM-based methods were able to exhibit considerable performance on these familiar categories, drawing strength from their robust in-context learning capabilities and a wealth of inherent commonsense knowledge. Conversely, it is challenging for learning-based methods to generalize to previously unseen instances, primarily due to the lack of demonstration data.

Notably, we discerned that similar manipulation policies might be applicable across diverse instances, allowing the LLM2Skill method to demonstrate appreciable performance on these relatively easy categories such as drawers, and buttons. However, when faced the variations within the kinematic structures, LLM2Skill fails due to the inability to craft novel trajectories for different kinematic structures like lifting straps and turning on faucets. Meanwhile, our method could still generalize to these object instances with a more flexible manipulation policy benefiting from the comprehension of kinematic knowledge.

Results on unseen categories: We extended our evaluation to objects within unseen categories. As shown in Table II, LLMs could easily generalize to prismatic articulated objects like kitchen pots, given that the manipulation trajectory is a simple straightforward linear path. Conversely, when manipulating revolute articulated objects, these baseline models exhibit a notable decline in the average success rate. This decline is due to that objects with revolute joints require more complex trajectories to manipulate. For example, when attempting to open a door, the generated trajectory must account for both the radius and angle to align with the object's kinematic structure. Otherwise, it would get stuck due to force constraint issues. Nonetheless, leveraging the

comprehension of the object's kinematic knowledge provided by our unified kinematic knowledge parser component, our method is adept at manipulating these revolute objects.

Despite equipping the LLM2Skill method with action candidates tailored to these novel categories, it still fails due to the lack of 3D reasoning capabilities essential for versatile articulated object manipulation.

Although the behavior cloning method shows some success on a few tasks, we find that these successes are the result of random movements in the environment. Such unpredictability is not usable for real-world applications, given the inherent risks. However, by representing the action as a sequence of manipulation waypoints, our method could combine with traditional motion planners to ensure safety and applicability in real-world scenarios.

C. Ablation Experiments

In the ablation experiments, we evaluate the effectiveness of our kinematic-aware prompt and waypoints generation method. Concretely, the model without the kinematic-aware planner component would directly utilize the unified kinematic description to generate manipulation waypoints without the hierarchical prompt. The model without waypoints would use the predefined action list as LLM2Skill but with our kinematic-aware prompting framework. We follow the setting in comparison experiments and demonstrate the results on unseen categories.

As shown in Table III, the method with both components achieves the best performance, which proves the effectiveness of each component. Concretely, in experiments, we find that the model without the kinematic-aware planner is more likely to generate trajectories that do not correspond to the task instructions. However, the model without waypoint generation could generate reasonable manipulation sequences but fail in manipulating complex objects such as laptop and bucket, due to the lack of flexibility. Combining with both these modules, our method could fully understand the kinematic structure of objects for generalizable articulated

Unseen Categories with different instances and poses								
Method	dishwasher	cabinet	door	bucket	window	trashcan	laptop	stapler
W/o. kinematic-planer planner	33.3%/ 66.6%	0%/ 33.3%	33.3%/ 33.3%	33.3%/ 33.3%	33.3%/ 0%	0%/ 0%	0%/ 0%	0%
W/o. waypoints	33.3%/ 66.6%	0%/ 0%	0%/ 33.3%	0%/33.3%	33.3%/ 33.3%	66.6% / 0%	33.3%/ 33.3%	66.6%
Ours	66.6%/ 100%	33.3%/ 66.6%	66.6%/ 66.6%	66.6%/ 100%	66.6%/ 66.6%	100%/ 33.3%	66.6%/ 100%	100%

TABLE III: The ablation study results. We use / to differentiate the model's performance on different manipulation commands for the same object.

	Unseen Instance		Unseen Categories		
Method	safe	refrigerator	dishwasher	cabinet	bucket
GPT-4	66.6%/ 66.6%	66.6%/ 100%	66.6%/ 100%	66.6%/ 66.6%	66.6%/ 100%
GPT-3.5-turbo	0%/ 0%	33.3%/ 33.3%	0%/ 0%	33.3%/ 33.3%	0%/ 0%

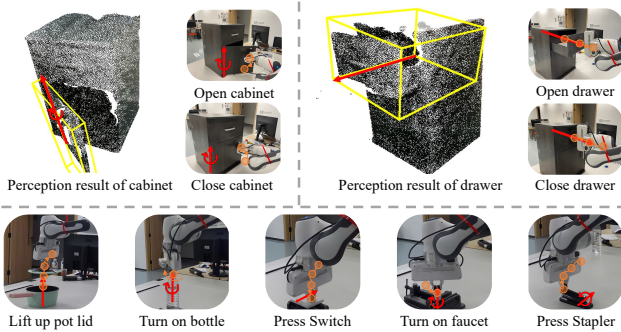TABLE IV: Abalation study on different GPT models with 7 demonstrations as prompt.

Fig. 4: Real-world experiments: we generate 3D manipulation waypoints with our framework for real-world object manipulation. The joint information of cabinet and drawer is estimated by the perception model, while the others are provided manually.

object manipulation.

We compare the performance of different Large Language Models. Due to the token limitation of GPT3.5, we only provide 9 demonstrations across 4 categories, which contain open/close safe, lift/lay down strap, open/close refrigerator, and open/close oven. As shown in Table IV, we observe that GPT-4 is better at following the prompt to reason the spatial information, while GPT-3.5-turbo always fails to understand the relationship between kinematic knowledge and manipulation waypoints, and provide wrong manipulation waypoints.

D. Real-world Experiment

To demonstrate the effectiveness of our framework in practical scenarios, we conducted experiments with a Franka Panda robot arm in the real world. To convert the kinematic structure of the manipulation object into texture format with our unified kinematic knowledge parser, we first combine Grounding-DINO [45] and Segment-anything [43] to accurately segment the target object. We incorporate the GAPartNet [41] as our backbone to detect actionable parts and capture joint information.

As shown in Figure 4, we evaluate our framework on 7 distinct objects. For cabinet and drawer categories, we utilize the perception model to obtain the contact point and joint information, while for other categories, we manually provide this information due to the limitation of perception models. Further, we use LLMs to generate 3D manipulation waypoints with our kinematic-aware prompting framework given the joint and affordance information. The results prove that our framework is capable of generalizing to practical scenarios without any additional demonstrations collected in the real world.

V. CONCLUSIONS AND LIMITATIONS

In this work, we propose a kinematic-aware prompting framework to utilize the rich world knowledge inherent in LLMs for generalizable articulated object manipulation. Based on the idea that the kinematic structure of an object determines the manipulation policy on it, this framework prompts LLMs with kinematic knowledge of objects to generate low-level motion trajectory waypoints for various object manipulations. Concretely, we build the unified kinematic knowledge parser and kinematic-aware planner, to empower LLMs to understand various object kinematic structures for generalizable articulated object manipulation via in-context learning. We evaluate our method on 48 instances across 16 categories, and the results prove our method could generalize across unseen instances and categories with only 17 demonstrations for prompting. The real-world experiments also prove our framework's generalization to practical scenarios.

Limitations. Provided with accurate object kinematic knowledge, our framework could achieve generalizable articulated object manipulation. However, its application in the real world is constrained by the capability of existing perception models. It is a promising way to combine the visual foundation models with LLMs for more challenging real-world scenarios. Further, the capability of LLMs in mathematical reasoning and spatial comprehension remains constrained, which would sometimes result in inaccuracy for manipulation waypoint generation. Thus, methods aimed at enhancing mathematical comprehension can be leveraged to augment the efficacy of our object manipulation framework.

VI. ACKNOWLEDGEMENT

REFERENCES

[1] Z. Jia, F. Liu, V. Thumuluri, L. Chen, Z. Huang, and H. Su, "Chain-of-thought predictive control," *arXiv preprint arXiv:2304.00776*, 2023.

[2] M. Shridhar, L. Manuelli, and D. Fox, "Perceiver-actor: A multi-task transformer for robotic manipulation," in *Conference on Robot Learning*. PMLR, 2023, pp. 785–799.

[3] Y. Geng, B. An, H. Geng, Y. Chen, Y. Yang, and H. Dong, "Rlafford: End-to-end affordance learning for robotic manipulation," in *2023 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2023, pp. 5880–5886.

[4] J. Gu, F. Xiang, X. Li, Z. Ling, X. Liu, T. Mu, Y. Tang, S. Tao, X. Wei, Y. Yao *et al.*, "Maniskill2: A unified benchmark for generalizable manipulation skills," *arXiv preprint arXiv:2302.04659*, 2023.

[5] A. Brohan, N. Brown, J. Carbajal, Y. Chebotar, J. Dabis, C. Finn, K. Gopalakrishnan, K. Hausman, A. Herzog, J. Hsu *et al.*, "Rt-1: Robotics transformer for real-world control at scale," *arXiv preprint arXiv:2212.06817*, 2022.

[6] A. Brohan, N. Brown, J. Carbajal, Y. Chebotar, X. Chen, K. Choromanski, T. Ding, D. Driess, A. Dubey, C. Finn *et al.*, "Rt-2: Vision-language-action models transfer web knowledge to robotic control," *arXiv preprint arXiv:2307.15818*, 2023.

[7] O. Mees, L. Hermann, and W. Burgard, "What matters in language conditioned robotic imitation learning over unstructured data," *IEEE Robotics and Automation Letters*, vol. 7, no. 4, pp. 11205–11212, 2022.

[8] L. X. Shi, A. Sharma, T. Z. Zhao, and C. Finn, "Waypoint-based imitation learning for robotic manipulation," *arXiv preprint arXiv:2307.14326*, 2023.

[9] H. Bharadhwaj, J. Vakil, M. Sharma, A. Gupta, S. Tulsiani, and V. Kumar, "Roboagent: Towards sample efficient robot manipulation with semantic augmentations and action chunking," *arxiv*, 2023.

[10] A. Zhou, M. J. Kim, L. Wang, P. Florence, and C. Finn, "Nerf in the palm of your hand: Corrective augmentation for robotics via novel-view synthesis," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 17907–17917.

[11] J. Liu, Z. Li, S. Calinon, and F. Chen, "Softgpt: Learn goal-oriented soft object manipulation skills by generative pre-trained heterogeneous graph transformer," *arXiv preprint arXiv:2306.12677*, 2023.

[12] C. Tang, D. Huang, W. Ge, W. Liu, and H. Zhang, "Graspgpt: Leveraging semantic knowledge from a large language model for task-oriented grasping," *arXiv preprint arXiv:2307.13204*, 2023.

[13] J. Wei, X. Wang, D. Schuurmans, M. Bosma, F. Xia, E. Chi, Q. V. Le, D. Zhou *et al.*, "Chain-of-thought prompting elicits reasoning in large language models," *Advances in Neural Information Processing Systems*, vol. 35, pp. 24824–24837, 2022.

[14] J. Wei, Y. Tay, R. Bommasani, C. Raffel, B. Zoph, S. Borgeaud, D. Yogatama, M. Bosma, D. Zhou, D. Metzler *et al.*, "Emergent abilities of large language models," *arXiv preprint arXiv:2206.07682*, 2022.

[15] S. Mirchandani, F. Xia, P. Florence, B. Ichter, D. Driess, M. G. Arenas, K. Rao, D. Sadigh, and A. Zeng, "Large language models as general pattern machines," *arXiv preprint arXiv:2307.04721*, 2023.

[16] Z. Mandi, S. Jain, and S. Song, "Roco: Dialectic multi-robot collaboration with large language models," *arXiv preprint arXiv:2307.04738*, 2023.

[17] W. Huang, C. Wang, R. Zhang, Y. Li, J. Wu, and L. Fei-Fei, "Voxposer: Composable 3d value maps for robotic manipulation with language models," *arXiv preprint arXiv:2307.05973*, 2023.

[18] V. Makoviychuk, L. Wawrzyniak, Y. Guo, M. Lu, K. Storey, M. Macklin, D. Hoeller, N. Rudin, A. Allshire, A. Handa *et al.*, "Isaac gym: High performance gpu-based physics simulation for robot learning," *arXiv preprint arXiv:2108.10470*, 2021.

[19] X. Zhang, D. Wang, S. Han, W. Li, B. Zhao, Z. Wang, X. Duan, C. Fang, X. Li, and J. He, "Affordance-driven next-best-view planning for robotic grasping," *arXiv preprint arXiv:2309.09556*, 2023.

[20] S. Nair, E. Mitchell, K. Chen, S. Savarese, C. Finn *et al.*, "Learning language-conditioned robot behavior from offline data and crowd-sourced annotation," in *Conference on Robot Learning*. PMLR, 2022, pp. 1303–1315.

[21] K. Rao, C. Harris, A. Irpan, S. Levine, J. Ibarz, and M. Khansari, "Rl-cyclegan: Reinforcement learning aware simulation-to-real," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 11157–11166.

[22] Z. Mandi, H. Bharadhwaj, V. Moens, S. Song, A. Rajeswaran, and V. Kumar, "Cacti: A framework for scalable multi-task multi-scene visual imitation learning," *arXiv preprint arXiv:2212.05711*, 2022.

[23] Z. Chen, S. Kiami, A. Gupta, and V. Kumar, "Genaug: Retargeting behaviors to unseen situations via generative augmentation," *arXiv preprint arXiv:2302.06671*, 2023.

[24] L. Chen, K. Lu, A. Rajeswaran, K. Lee, A. Grover, M. Laskin, P. Abbeel, A. Srinivas, and I. Mordatch, "Decision transformer: Reinforcement learning via sequence modeling," *Advances in neural information processing systems*, vol. 34, pp. 15084–15097, 2021.

[25] A. Z. Ren, A. Dixit, A. Bodrova, S. Singh, S. Tu, N. Brown, P. Xu, L. Takayama, F. Xia, J. Varley *et al.*, "Robots that ask for help: Uncertainty alignment for large language model planners," *arXiv preprint arXiv:2307.01928*, 2023.

[26] W. Huang, F. Xia, T. Xiao, H. Chan, J. Liang, P. Florence, A. Zeng, J. Tompson, I. Mordatch, Y. Chebotar *et al.*, "Inner monologue: Embodied reasoning through planning with language models," *arXiv preprint arXiv:2207.05608*, 2022.

[27] G. Zhou, Y. Hong, and Q. Wu, "Navgpt: Explicit reasoning in vision-and-language navigation with large language models," *arXiv preprint arXiv:2305.16986*, 2023.

[28] M. Ahn, A. Brohan, N. Brown, Y. Chebotar, O. Cortes, B. David, C. Finn, C. Fu, K. Gopalakrishnan, K. Hausman *et al.*, "Do as i can, not as i say: Grounding language in robotic affordances," *arXiv preprint arXiv:2204.01691*, 2022.

[29] J. Wu, R. Antonova, A. Kan, M. Lepert, A. Zeng, S. Song, J. Bohg, S. Rusinkiewicz, and T. Funkhouser, "Tidybot: Personalized robot assistance with large language models," *arXiv preprint arXiv:2305.05658*, 2023.

[30] B. Chen, F. Xia, B. Ichter, K. Rao, K. Gopalakrishnan, M. S. Ryoo, A. Stone, and D. Kappler, "Open-vocabulary queryable scene representations for real world planning," in *2023 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2023, pp. 11509–11522.

[31] P. A. Jansen, "Visually-grounded planning without vision: Language models infer detailed plans from high-level instructions," *arXiv preprint arXiv:2009.14259*, 2020.

[32] Z. Wu, Z. Wang, X. Xu, J. Lu, and H. Yan, "Embodied task planning with large language models," *arXiv preprint arXiv:2307.01848*, 2023.

[33] X. Zhu, Y. Chen, H. Tian, C. Tao, W. Su, C. Yang, G. Huang, B. Li, L. Lu, X. Wang *et al.*, "Ghost in the minecraft: Generally capable agents for open-world enviroments via large language models with text-based knowledge and memory," *arXiv preprint arXiv:2305.17144*, 2023.

[34] H. Yuan, C. Zhang, H. Wang, F. Xie, P. Cai, H. Dong, and Z. Lu, "Plan4mc: Skill reinforcement learning and planning for open-world minecraft tasks," *arXiv preprint arXiv:2303.16563*, 2023.

[35] D. Driess, F. Xia, M. S. Sajjadi, C. Lynch, A. Chowdhery, B. Ichter, A. Wahid, J. Tompson, Q. Vuong, T. Yu *et al.*, "Palm-e: An embodied multimodal language model," *arXiv preprint arXiv:2303.03378*, 2023.

[36] S. Vemprala, R. Bonatti, A. Bucker, and A. Kapoor, "Chatgpt for robotics: Design principles and model abilities," *Microsoft Auton. Syst. Robot. Res*, vol. 2, p. 20, 2023.

[37] I. Singh, V. Blukis, A. Mousavian, A. Goyal, D. Xu, J. Tremblay, D. Fox, J. Thomason, and A. Garg, "Progprompt: Generating situated robot task plans using large language models," in *2023 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2023, pp. 11523–11530.

[38] J. Liang, W. Huang, F. Xia, P. Xu, K. Hausman, B. Ichter, P. Florence, and A. Zeng, "Code as policies: Language model programs for embodied control," in *2023 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2023, pp. 9493–9500.

[39] W. Yu, N. Gileadi, C. Fu, S. Kirmani, K.-H. Lee, M. G. Arenas, H.-T. L. Chiang, T. Erez, L. Hasenclever, J. Humplik *et al.*, "Language to rewards for robotic skill synthesis," *arXiv preprint arXiv:2306.08647*, 2023.

[40] N. Di Palo, A. Byravan, L. Hasenclever, M. Wulfmeier, N. Heess, and M. Riedmiller, "Towards a unified agent with foundation models," in *Workshop on Reincarnating Reinforcement Learning at ICLR 2023*, 2023.

[41] H. Geng, H. Xu, C. Zhao, C. Xu, L. Yi, S. Huang, and H. Wang, "Gapartnet: Cross-category domain-generalizable object perception and manipulation via generalizable and actionable parts," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 7081–7091.

[42] L. Liu, W. Xu, H. Fu, S. Qian, Q. Yu, Y. Han, and C. Lu, "Akb-48: A real-world articulated object knowledge base," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 14 809–14 818.

[43] A. Kirillov, E. Mintun, N. Ravi, H. Mao, C. Rolland, L. Gustafson, T. Xiao, S. Whitehead, A. C. Berg, W.-Y. Lo *et al.*, "Segment anything," *arXiv preprint arXiv:2304.02643*, 2023.

[44] K. Mo, S. Zhu, A. X. Chang, L. Yi, S. Tripathi, L. J. Guibas, and H. Su, "PartNet: A large-scale benchmark for fine-grained and hierarchical part-level 3D object understanding," in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019.

[45] S. Liu, Z. Zeng, T. Ren, F. Li, H. Zhang, J. Yang, C. Li, J. Yang, H. Su, J. Zhu *et al.*, "Grounding dino: Marrying dino with grounded pre-training for open-set object detection," *arXiv preprint arXiv:2303.05499*, 2023.