Activation Steering Guided Constraint Generation for Robotic Manipulation

Samarth Agrawal
Columbia University
ssa2206@columbia.edu

Aakash Aanegola
Columbia University
aa5506@columbia.edu

Danelle Tuchman
Columbia University
dst2161@columbia.edu

I. Problem Overview

Autonomous robot architectures comprised of multiple "foundation models" are receiving considerable interest. Such systems offer the promise of decoding high-level natural language instructions with visual context and inferring how to execute them. As these systems grow more complex, it becomes increasingly essential to understand what drives their behavior, especially in under specified task settings with multiple possible interpretations. Before autonomy can be granted in any meaningful way, developers must ensure the safety and reliability of any task-decoding system.

ReKep: Relational Keypoint Constraints [1] offers a promising framework for multi-modal AI architectures to communicate and encode their preferences. A ReKep instance is a Python function $f : \mathbb{R}^{K\times 3} \rightarrow \mathbb{R}$ mapping an array of 3D locations ("keypoints") to numerical cost. In a very concrete sense, the loss landscape over keypoints is a human-understandable reflection of the model's values. What if such constraints were broadened to more abstract concepts such as fairness and safety? As demonstrated by vision-language query models such as ViperGPT [2], code generation is readily able to capture the commonsense knowledge and beliefs of LLMs. Thus zero shot prompting should be sufficient to evoke path and goal constraints related to broader alignment notions ("divide the muffins fairly", "be extra gentle with fragile objects"). Nevertheless, monitoring and quantifying the interpretability, safety, and efficiency of model-imposed constraints remains an open challenge. Fortunately, there is a wide body of work on the robustness and value alignment Large Language Models (LLMs). In this project, we propose leveraging activation steering techniques, a recent discovery made by Mechanistic Interpretability researchers at Anthropic, to test the constraint generation process. We aim to activate abstract concepts within the model during inference and assess the influence on the resulting ReKep instances.

II. Related Work

Prior works have demonstrated the ability of LLMs to generate task plans, infer spatial constraints, and reason about affordances from multimodal inputs. These methods, however, often struggle with issues of safety, reliability, and interpretability, which are critical considerations before the widespread deployment of human-facing robots. ReKep [1] introduces a framework for encoding robotic manipulation tasks as relational key point constraints, leveraging vision-language models to specify constraints. By structuring tasks as a sequence of optimizable constraints in SE(3), ReKep provides a flexible and efficient means of generating task plans. Still, it remains susceptible to the limitations of its underlying LLM, which can lead to unintended behavior due to hallucinated constraints [3]. Extensive prior work has explored LLMs for robotic task planning and execution [1, 2, 4, 5]. [6] presents a multi-layer LLM approach for enhancing robotic task execution, emphasizing structured planning. More recently, multimodal LLMs have been applied to end-to-end robotic manipulation [7], incorporating vision and language priors for task inference. However, these approaches largely rely on LLMs' intrinsic reasoning capabilities, making them prone to hallucinations that may lead to unsafe or suboptimal robotic behaviors.

Experiments like [8] demonstrate the inability of LLMs to act morally in a scenario that offers less utility. A new trend in the AI safety and interpretability research camp is the use of activation steering [9] to override user prompts and "trigger" learned LLM features to explicitly ensure constrained outputs. This approach leverages Sparse Auto-Encoders (SAEs) [10] to identify latent LLM features associated with high- and low-level concepts. By selectively modifying activations at specific layers, it becomes possible to steer the model's outputs, effectively amplifyingh or suppressing concepts during inference. To the best of our knowledge, there remains a gap in current literature that applies this methodology to robotic manipulation planning.

III. Proposed Method and Experiments

Our approach extends ReKep by diving deeper into the Vision Language model and constraint generation. We propose using activation steering to influence the LLM's internal representations during constraint generation. Activation steering helps us to extract and apply abstract concepts that may bias the model towards producing more structured, interpretable, and safer constraints. These vectors can be derived from Sparse Autoencoder (SAE) features or manually identified activation patterns with desirable constraint properties.

Our method consists of three key stages that we detail below.

A. Constraint Generation with Activation Steering

We begin by implementing a constraint generation pipeline using an open-source LLM (e.g., LLaMa). Activation steering will be applied during this process to encourage the generation of semantically meaningful keypoint constraints.

B. Optimization and Integration into ReKep

The modified constraints will be incorporated into the hierarchical optimization pipeline used in ReKep, where they guide the selection of end-effector poses in SE(3). We aim to analyze how activation-steered constraints impact the optimization process and whether they lead to more interpretable/predictable robotic behavior that satisfies overarching prespecified goals.

C. Experimental Validation

We will evaluate our method in a simulated environment (potentially OmniGibson), comparing standard ReKep-generated constraints with those modified via activation steering. Our metrics will capture baseline performance, but we'll also have to construct additional metrics that measure not-so-obvious costs like safety and reliability. We hope to explore different categories of robotic manipulation, including multi-stage tasks and reactive behaviors similar to the experiments conducted in ReKep.

Additionally, if time permits, we will extend our method to real-world robotic deployments to validate its feasibility in the real-world setting. By integrating activation steering with ReKep, we aim to improve the transparency and reliability of constraint-based robotic manipulation while preserving the flexibility and scalability of LLM-driven constraint generation. This is also an attempt to align AI interpretability work with the field of robotics to ensure that advancements in both spaces can drive future AI research.

IV. Milestones and Timelines

Recreate ReKep in novel setting: Our extension requires two major modifications to ReKep. Firstly, we must transition to an open source Vision Language model (original work used GPT-4o) such as LLaMa or DeepSeek for which Mechanistic Interpretability methods are possible. Secondly, we plan to work primarily in a simulated environment, with integration of a real robot arm as a tentative stretch goal. Simulation opens up a far greater diversity of testable scenarios, as many situations (such as those involving humans) are unsuitable for real world evaluation. Identifying software that enables flexible object configurations will be a hefty component of the actual project labor. Current options under consideration are PyBullet, Webots, and NVIDIA Isaac Sim, but these are very tentative. Naturally, we will also have to translate the inverse kinematics and path solving/control methods to whatever setup we decide on.

Literature Review on LLM Safety: In parallel to this effort will be an extensive literature review on LLM guardrails and feature activation. Our team will identify which multi-modal Vision Language models are suitable for activation steering, leveraging existing tools like NeuronPedia and collections of existing features and open source SAEs. This will provide us with a vocabulary of abstract concepts that we can amplify or diminish within the model. *We hope to have our steering literature review and ReKep reproduction completed by mid-March.*

Defining Scenarios and Experiments: with these two preliminaries completed, we will have access to a list of extracted features as well as the ability to impose relational keypoint constraints on models in simulation. By the end of March, we hope to turn this into a concrete list of experiments and a quantitative evaluation framework. What kind of behavioral scenarios, steering vectors, and perceptual information we will be using should be the subject of our Milestone report in March 27th.

Integrate Steering Vectors into constraint generation: April will be entirely focused on execution and tabulating results. We will use the extracted activation vectors to augment keypoint constraints and validate their effect on interpretability and behavior in simulation. If time permits, extend the implementation from sim to a real robot system; as a member of the A2R Barnard Robotics Labs, Danelle has access to a 7 jointed robotic arm and a quadruped robot. The arm can traverse in a figure eight while bolted to the floor. The quadruped has a full range of motion, which would allow for motion testing through utilizing constraints from our model.

V. Team Background

A. Aakash Aanegola

Aakash is a first-year master's student with experience in perception and planning. He is interested in AI safety and interpretability and seeks to bridge his two interests through this project. With a strong background in deep learning research, robotics, and graph-based learning, he aims to develop more transparent and reliable AI-driven robotic systems.

B. Danelle

Danelle is a senior at Columbia College majoring in Computer Science on the Intelligent Systems track. She is interested in applying ML towards improving robotic motion. Her classwork on Computational Aspects of Robotics, Natural Language Processing, Robotics Studio, AI, and research in the Accessible and Accelerated Robotics lab (A2R) relates to this project's goal of utilizing LLM's to generate safe and intentional constraints for robotic motion.

C. Sammy

Sammy is a first-year master's student in Computer Science, with significant deep learning research experience across multiple domains. As a member of the Columbia AI Alignment Club (CAIAC), he has prior exposure to Mechanistic Interpretability research papers. His coursework in Applied Computer Vision, Robotics, and Cognitive Science relates to his project focus on integrating the steering vectors and Physics simulation with the actual task descriptions and evaluation methods for the project.

References

[1] Wenlong Huang, Chen Wang, Yunzhu Li, Ruohan Zhang, and Li Fei-Fei. Rekep: Spatio-temporal reasoning of relational keypoint constraints for robotic manipulation. *arXiv preprint arXiv:2409.01652*, 2024.

[2] Dídac Surís, Sachit Menon, and Carl Vondrick. Vipergpt: Visual inference via python execution for reasoning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 11888–11898, 2023.

[3] Lei Huang, Weijiang Yu, Weitao Ma, Weihong Zhong, Zhangyin Feng, Haotian Wang, Qianglong Chen, Weihua Peng, Xiaocheng Feng, Bing Qin, and Ting Liu. A survey on hallucination in large language models: Principles, taxonomy, challenges, and open questions. *ACM Trans. Inf. Syst.*, 43(2), January 2025. ISSN 1046-8188. doi: 10.1145/3703155. URL https://doi.org/10.1145/3703155.

[4] Mingjie Pan, Jiyao Zhang, Tianshu Wu, Yinghao Zhao, Wenlong Gao, and Hao Dong. Omnimanip: Towards general robotic manipulation via object-centric interaction primitives as spatial constraints. 2025. URL https://api.semanticscholar.org/CorpusID:275342381.

[5] Wenlong Huang, Chen Wang, Ruohan Zhang, Yunzhu Li, Jiajun Wu, and Li Fei-Fei. Voxposer: Composable 3d value maps for robotic manipulation with language models. *ArXiv*, abs/2307.05973, 2023. URL https://api.semanticscholar.org/CorpusID:259837330.

[6] Zhirong Luan, Yujun Lai, Rundong Huang, Shuanghao Bai, Yuedi Zhang, Haoran Zhang, and Qian Wang. Enhancing robot task planning and execution through multi-layer large language models. *Sensors (Basel, Switzerland)*, 24, 2024. URL https://api.semanticscholar.org/CorpusID:268341391.

[7] Jiaming Liu, Chenxuan Li, Guanqun Wang, Lily Lee, Kaichen Zhou, Sixiang Chen, Chuyan Xiong, Jiaxin Ge, Renrui Zhang, and Shanghang Zhang. Self-corrected multimodal large language model for end-to-end robot manipulation. *ArXiv*, abs/2405.17418, 2024. URL https://api.semanticscholar.org/CorpusID:270063644.

[8] Dan Hendrycks, Mantas Mazeika, Andy Zou, Sahil Patel, Christine Zhu, Jesus Navarro, Dawn Song, Bo Li, and Jacob Steinhardt. What would jiminy cricket do? towards agents that behave morally. *CoRR*, abs/2110.13136, 2021. URL https://arxiv.org/abs/2110.13136.

[9] Alexander Matt Turner, Lisa Thiergart, Gavin Leech, David S. Udell, Juan J. Vazquez, Ulisse Mini, and Monte Stuart MacDiarmid. Steering language models with activation engineering. 2023. URL https://api.semanticscholar.org/CorpusID:261049449.

[10] Trenton Bricken, Adly Templeton, Joshua Batson, Brian Chen, Adam Jermyn, Tom Conerly, Nick Turner, Cem Anil, Carson Denison, Amanda Askell, Robert Lasenby, Yifan Wu, Shauna Kravec, Nicholas Schiefer, Tim Maxwell, Nicholas Joseph, Zac Hatfield-Dodds, Alex Tamkin, Karina Nguyen, Brayden McLean, Josiah E Burke, Tristan Hume, Shan Carter, Tom Henighan, and Christopher Olah. Towards monosemanticity: Decomposing language models with dictionary learning. *Transformer Circuits Thread*, 2023. https://transformer-circuits.pub/2023/monosemantic-features/index.html.