Verifiably Following Complex Robot Instructions with Foundation Models

Benedict Quartey[†*], Eric Rosen[*], Stefanie Tellex, George Konidaris
Department of Computer Science, Brown University

***Abstract*— When instructing robots, users want to flexibly express constraints, refer to arbitrary landmarks, and verify robot behavior, while robots must disambiguate instructions into specifications and ground instruction referents in the real world. To address this problem, we propose Language Instruction grounding for Motion Planning (LIMP), an approach that enables robots to verifiably follow complex, open-ended instructions in real-world environments without prebuilt semantic maps. LIMP constructs a symbolic instruction representation that reveals the robot's alignment with an instructor's intended motives and affords the synthesis of correct-by-construction robot behaviors. We conduct a large-scale evaluation of LIMP on 150 instructions across five real-world environments, demonstrating its versatility and ease of deployment in diverse, unstructured domains. LIMP performs comparably to state-of-the-art baselines on standard open-vocabulary tasks and additionally achieves a 79% success rate on complex spatiotemporal instructions, significantly outperforming baselines that only reach 38%.** [1]

I. Introduction

Robots need a rich understanding of natural language to be instructable by non-experts in unstructured environments. People, on the other hand, need to be able to verify that a robot has understood a given instruction and will act appropriately. Achieving these objectives, however, is challenging as natural language instructions often feature ambiguous phrasing, intricate spatiotemporal constraints, and unique referents. To illustrate, consider the instruction shown in Figure 1: *"Bring the green plush toy to the whiteboard in front of it, watch out for the robot in front of the toy"*. Solving such a task requires a robot to ground open-vocabulary referents, follow temporal constraints, and disambiguate objects using spatial descriptions. Foundation models [1], [2] offer a path to achieving such complex long-horizon goals; however, existing approaches for robot instruction following have largely focused on navigation [3], [4], [5], [6], [7]. These methods, broadly classified under object goal navigation [8], enable navigation to instances of an object category but are limited in their ability to localize spatial references and disambiguate object instances based on descriptive language. Other works [9], [10], [11] extend instruction following to mobile manipulation but are limited to tasks with simple temporal constraints expressed in unambiguous language. Moreover, existing efforts typically rely on Large Language Models (LLMs) as complete planners, bypassing intermediate symbolic representations that could provide verification of correctness before execution. Alternative approaches leveraging code-writing LLMs [5], [6], [12] are susceptible to errors in generated code, which may lead to unsafe robot behaviors. Mapping natural language to specification languages like temporal logic [13] provides a robust framework for language disambiguation, handling complex temporal constraints, and behavior verification. However, prior works along this line require prebuilt semantic maps with discrete sets of prespecified referents/landmarks from which instructions can be constructed [7], [14], [15].

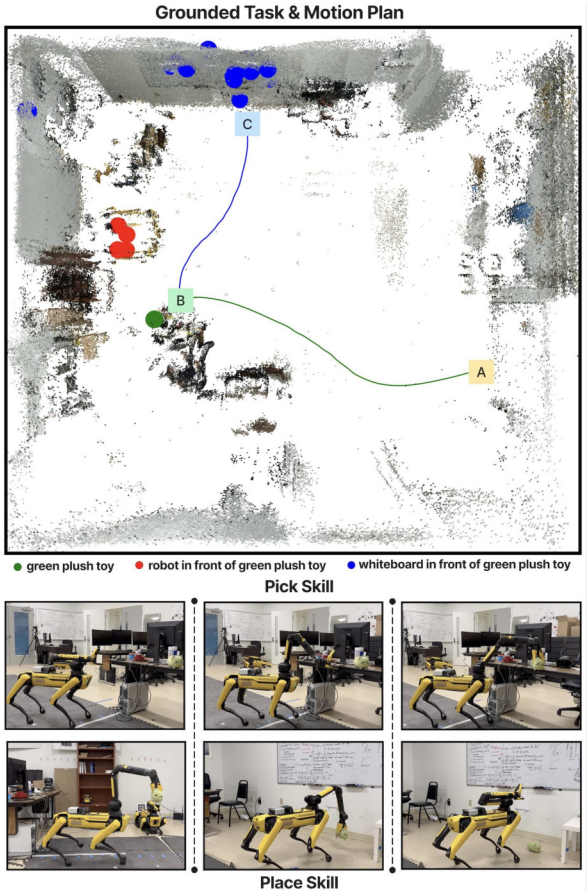

Fig. 1: Our approach executing the instruction *"Bring the green plush toy to the whiteboard in front of it, watch out for the robot in front of the toy"*. The robot dynamically detects and grounds open-vocabulary referents with spatial constraints to construct an instruction-specific semantic map, then synthesizes a task and motion plan to solve the task. In this example, the robot navigates from its start location (yellow, A), to the green plush toy (green, B), executes a pick skill then navigates to the whiteboard (blue, C), and executes a place skill. Note that the robot has no prior semantic knowledge of the environment.

We propose *Language Instruction grounding for Motion Planning (LIMP)*, a method that leverages foundation

[*]Equal Contribution
[†]Corresponding Author (*Email*: benedict_quartey@brown.edu)
[1]See supplementary materials and demo videos at robotlimp.github.io

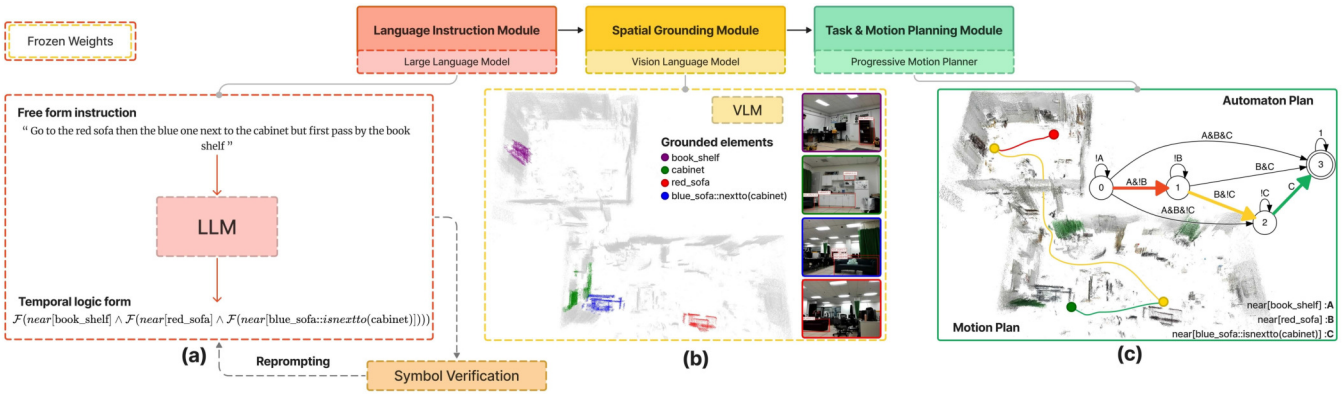

Fig. 2: **[A]** LIMP translates natural language instructions into temporal logic expressions, where open-vocabulary referents are applied to predicates that correspond to robot skills—note the context-aware resolution of the phrase "blue one" to the referent "blue_sofa". **[B]** Vision-language models detect referents, while spatial reasoning disambiguates referent instances to generate a 3D semantic map that localizes instruction-specific referents. **[C]** Finally, the temporal logic expression is compiled into a finite-state automaton, which a task and motion planner uses with dynamically-generated task progression semantic maps to progressively identify goals and constraints in the environment, and generate a plan that satisfies the high-level task specification.

models and temporal logics to dynamically generate instruction-conditioned semantic maps that enable robots to construct verifiable controllers for following navigation and mobile manipulation instructions with open vocabulary referents and complex spatiotemporal constraints.

II. Language Instruction Grounding for Motion Planning

LIMP interprets expressive natural language instructions to generate instruction-conditioned semantic maps, enabling robots to solve long-horizon tasks with complex spatiotemporal constraints (Figure 2). Our approach has a modular structure consisting of a Language Instruction Module, a Spatial Grounding Module and a Task and Motion Planning Module.

A. Language Instruction Module

In this module, we leverage a large language model (LLM) ψ to translate a natural language instruction l into a linear temporal logic specification φ_l with a novel composable syntax for referent disambiguation. We achieve this through a two-stage in-context learning strategy. The first stage prompts ψ to translate l into a conventional LTL formula ϕ_l where propositions are referent objects. The second stage takes l and ϕ_l as input and prompts ψ to generate a new formula φ_l with predicate functions corresponding to parameterized robot skills, as shown in Figure 2.a.

B. Spatial Grounding Module

In the spatial grounding module, we detect and localize specific instances of objects referenced in a given instruction. This module leverages a vision-language model (VLM) to detect all referent occurrences from prior observations of the environment. An initial semantic map with all detected referent instances is generated by backprojecting pixels in segmented referent masks unto a 3D map. The spatial comparators of each referent object is resolved with respect to the origin coordinate frame of reference and used to filter referent instances to localize the exact referent instances described in the instruction.

C. Task and Motion Planning Module

Finally, our TAMP module synthesizes and sequences navigation and manipulation behaviors to produce a plan that satisfies the temporal and spatial constraints expressed in the given instruction. Our TAMP algorithm compiles the LTL formula with parameterized robot skills into an equivalent finite-state automaton to generate a verifiably correct task and motion plan. A path from the initial to the accepting state in this automaton is a high-level task plan that interleaves navigation and manipulation objectives required to satisfy the instruction. Automaton states are connected by transition edges representing the logical expressions required for transitions. For each transition, our algorithm executes the necessary low-level behaviors: for manipulation subgoals, it executes the appropriate parameterized skill; for navigation subgoals, it dynamically localizes goal and constraint regions and performs continuous path planning using the Fast Marching Tree algorithm (FMT*) [16].

III. Discussion and Conclusion

We perform a large scale evaluation and demonstrate our approach on 150 instructions in five real-world environments. In our experiments, LIMP performs comparably to state-of-the-art baselines on standard open-vocabulary tasks and additionally achieves a 79% success rate on complex spatiotemporal instructions, significantly outperforming baselines that only reach 38%. Beyond the verification benefits of symbolic planning, our approach ensures each robot step adheres to constraints while achieving subgoals, contrasting existing approaches [10], [12] which struggle to adhere to strict temporal constraints—for example, avoiding a specific referent while approaching another.

Foundation models hold significant promise for advancing the next generation of autonomous robots. Our results suggest that combining these models—LLMs for language and VLMs for vision—with established methods for safety, explainability, and verifiable behavior synthesis can lead to more reliable and capable robotic systems. Visit our project website to see our full paper and robot demonstration videos.

REFERENCES

[1] Y. Hu, Q. Xie, V. Jain, J. Francis, J. Patrikar, N. Keetha, S. Kim, Y. Xie, T. Zhang, H.-S. Fang, S. Zhao, S. Omidshafiei, D.-K. Kim, A.-a. Agha-mohammadi, K. Sycara, M. Johnson-Roberson, D. Batra, X. Wang, S. Scherer, C. Wang, Z. Kira, F. Xia, and Y. Bisk, "Toward General-Purpose Robots via Foundation Models: A Survey and Meta-Analysis," Oct. 2024, arXiv:2312.08782 [cs]. [Online]. Available: http://arxiv.org/abs/2312.08782

[2] R. Firoozi, J. Tucker, S. Tian, A. Majumdar, J. Sun, W. Liu, Y. Zhu, S. Song, A. Kapoor, K. Hausman, B. Ichter, D. Driess, J. Wu, C. Lu, and M. Schwager, "Foundation models in robotics: Applications, challenges, and the future," *The International Journal of Robotics Research*, p. 02783649241281508, Sept. 2024, publisher: SAGE Publications Ltd STM. [Online]. Available: https://doi.org/10.1177/02783649241281508

[3] S. Y. Gadre, M. Wortsman, G. Ilharco, L. Schmidt, and S. Song, "CoWs on Pasture: Baselines and Benchmarks for Language-Driven Zero-Shot Object Navigation," in *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, June 2023, pp. 23 171–23 181. [Online]. Available: https://ieeexplore.ieee.org/document/10203853/

[4] D. Shah, B. Osiński, B. Ichter, and S. Levine, "LM-Nav: Robotic Navigation with Large Pre-Trained Models of Language, Vision, and Action," in *Proceedings of The 6th Conference on Robot Learning*. PMLR, Mar. 2023, pp. 492–504, iSSN: 2640-3498. [Online]. Available: https://proceedings.mlr.press/v205/shah23b.html

[5] C. Huang, O. Mees, A. Zeng, and W. Burgard, "Visual Language Maps for Robot Navigation," in *2023 IEEE International Conference on Robotics and Automation (ICRA)*, May 2023, pp. 10 608–10 615. [Online]. Available: https://ieeexplore.ieee.org/document/10160969

[6] ——, "Audio Visual Language Maps for Robot Navigation," in *Experimental Robotics*, M. H. Ang Jr and O. Khatib, Eds. Cham: Springer Nature Switzerland, 2024, pp. 105–117.

[7] J. X. Liu, Z. Yang, I. Idrees, S. Liang, B. Schornstein, S. Tellex, and A. Shah, "Grounding Complex Natural Language Commands for Temporal Tasks in Unseen Environments," in *Proceedings of The 7th Conference on Robot Learning*. PMLR, Dec. 2023, pp. 1084–1110, iSSN: 2640-3498. [Online]. Available: https://proceedings.mlr.press/v229/liu23d.html

[8] P. Anderson, A. Chang, D. S. Chaplot, A. Dosovitskiy, S. Gupta, V. Koltun, J. Kosecka, J. Malik, R. Mottaghi, M. Savva, and A. R. Zamir, "On Evaluation of Embodied Navigation Agents," July 2018, arXiv:1807.06757 [cs]. [Online]. Available: http://arxiv.org/abs/1807.06757

[9] S. Yenamandra, A. Ramachandran, K. Yadav, A. S. Wang, M. Khanna, T. Gervet, T.-Y. Yang, V. Jain, A. Clegg, J. M. Turner, Z. Kira, M. Savva, A. X. Chang, D. S. Chaplot, D. Batra, R. Mottaghi, Y. Bisk, and C. Paxton, "HomeRobot: Open-Vocabulary Mobile Manipulation," in *Proceedings of The 7th Conference on Robot Learning*. PMLR, Dec. 2023, pp. 1975–2011, iSSN: 2640-3498. [Online]. Available: https://proceedings.mlr.press/v229/yenamandra23a.html

[10] 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)*, May 2023, pp. 11 509–11 522. [Online]. Available: https://ieeexplore.ieee.org/document/10161534

[11] P. Liu, Y. Orru, J. Vakil, C. Paxton, N. Shafiullah, and L. Pinto, "Demonstrating OK-Robot: What Really Matters in Integrating Open-Knowledge Models for Robotics," in *Robotics: Science and Systems*. Robotics: Science and Systems Foundation, July 2024. [Online]. Available: http://www.roboticsproceedings.org/rss20/p091.pdf

[12] 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)*, May 2023, pp. 9493–9500. [Online]. Available: https://ieeexplore.ieee.org/document/10160591

[13] E. A. Emerson, "Temporal and Modal Logic," in *Handbook of Theoretical Computer Science, Volume B: Formal Models and Semantics*, J. v. Leeuwen, Ed. Elsevier and MIT Press, 1990, pp. 995–1072. [Online]. Available: https://doi.org/10.1016/b978-0-444-88074-1.50021-4

[14] J. Pan, G. Chou, and D. Berenson, "Data-Efficient Learning of Natural Language to Linear Temporal Logic Translators for Robot Task Specification," in *2023 IEEE International Conference on Robotics and Automation (ICRA)*, May 2023, pp. 11 554–11 561. [Online]. Available: https://ieeexplore.ieee.org/document/10161125

[15] B. Quartey, A. Shah, and G. Konidaris, "Exploiting Contextual Structure to Generate Useful Auxiliary Tasks," in *NeurIPS 2023 Workshop on Generalization in Planning*, vol. abs/2303.05038, 2023, arXiv: 2303.05038. [Online]. Available: https://doi.org/10.48550/arXiv.2303.05038

[16] L. Janson, E. Schmerling, A. Clark, and M. Pavone, "Fast marching tree: A fast marching sampling-based method for optimal motion planning in many dimensions," *The International Journal of Robotics Research*, vol. 34, no. 7, pp. 883–921, June 2015, publisher: SAGE Publications Ltd STM. [Online]. Available: https://doi.org/10.1177/0278364915577958