LisTex United
2026 Team Description Paper

Raghav Arora[1], Afonso Certo[2], Rodrigo Coimbra[2], Justin Hart[1], Yoonwoo Kim[1], Pedro U. Lima[2], Duarte Santos[2], Peter Stone[1], and Lingyun Xiao[1]

[1] The University of Texas at Austin, USA
[2] Institute for Systems and Robotics, IST - University of Lisbon, Portugal

Abstract. LisTex United is a new RoboCup@Home team resulting of a joint project (FOMO-HODOR) between the Institute for Systems and Robotics at IST, U. Lisbon and UT Austin, with contributions from SocRob@Home and UT Austin Villa team members. The team focuses on using a humanoid robot (BOOSTER T1 with fingered hands) to perform the home tasks, so as to investigate and address the challenges posed by autonomous humanoid robots at home. The FOMO-HODOR project will leverage multi-modal Large Language Models (LLMs) in various roles to improve robotics, aiming to harness their full potential within practical general-purpose deployment scenarios and leveraging their advanced reasoning capabilities for the effective planning of complex tasks. LisTex United will build on the vast experience of its team members in RoboCup@Home to integrate general speech interaction and task planning methods with humanoid-specific topics, such as full-body motion planning, vision-based grasping and locomotion.

1 Introduction

UT Austin Villa has participated in seven RoboCup@Home competitions. This has leveraged a strong program of research that has been in part inspired by the team efforts in this competition to build a comprehensive service robot system which is used in UT Austin laboratories, in real-world deployments, and to compete in RoboCup@Home.

The SocRob@Home team has represented the Institute for Systems and Robotics (ISR-Lisboa) in RoboCup since 1998. Originating from the ISR-Lisboa SocRob (Soccer Robots or Society of Robots) research initiative, the team has participated in multiple RoboCup leagues, including RoboCupSoccer Simulation, 4-Legged and Middle Size, RoboCupRescue Real Robot and, since 2016, RoboCup@Home Open Platform, where it got to the podium twice in 2023 and 2024. SocRob@Home has involved over 100 students, from Bachelor's to Ph.D. levels, post-doc researchers and professors, fostering the integration of task planning, navigation, perception and manipulation towards advanced domestic service robotics.

The two research groups joined effort in a research project, with a case study on RoboCup@Home, carried out by the LisTex United team. There are significant research challenges that must be met to achieve reliable and contextually

aware task planning for robots. Prior attempts at LLM integrations on robots have been relatively simplistic. The project seeks to better integrate the technology into a more cohesive robotics architecture, aiming at a FOundation MOdel for HumanOid DOmestic RObots (FOMO-HODOR).

The impact of the envisioned framework can be substantial. LLMs have not yet been well-integrated into robotics AI architectures, with existing work mostly using them as a substitute for the current state-of-the-art methods. Unfortunately, LLMs are also missing many of the strengths of the state of the art in robotics. Developing this framework will allow it to be transitioned to a variety of robots, performing a variety of tasks, and is likely to have far-reaching impacts for broader and more rapid deployment of robot technologies in real-world scenarios.

2 Software and Scientific Contributions

This section describes the scientific contributions made to enable humanoids to perform Robocup@Home tasks including knowledge representation, semantic perception, whole body manipulation, and locomotion.

2.1 Robot Architecture

Our robot architecture is designed to support dynamic human–robot interactions in complex environments. The three–layer structure, illustrated in Figure 1, integrates the robot's skill components—such as perception, navigation, and manipulation—with high-level reactive and deliberative control modules. The top layer sequences and executes skills, remaining reactive during execution to handle environmental changes. A central knowledge base facilitates information sharing among all components, providing a unified world model accessible to both reactive and deliberative processes. The deliberative control layer leverages this shared knowledge to reason about the environment and generate plans for tasks that cannot be statically decomposed, following the design described by Jiang *et al.* [1]. This layered architecture provides a flexible foundation that accommodates both cloud-based and onboard planning pipelines, enabling different approaches to be seamlessly integrated within the same overall framework.

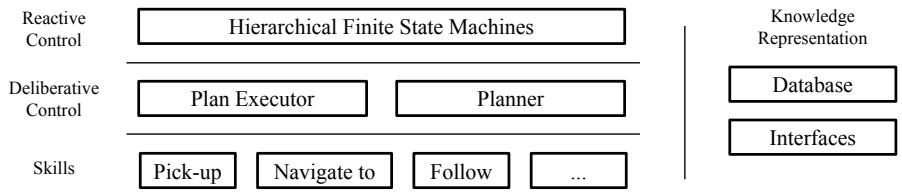


Fig. 1: Implementation of our robot architecture on Booster T1.

2.2 Knowledge Representation and Planning

Our knowledge representation subsystem stores grounded robot knowledge in a SQL database in order to allow for fast access and easy querying. Queries can be formed using custom C++ and Python libraries. For instance, in the *Storing Groceries* and *GPSR* tasks, the knowledge base is used to query object properties such as categories and default locations. The knowledge base can be dynamically updated by our perception system described below. Fig. 2 shows the knowledge base after the robot has detected a ketchup bottle on the dining table.

At UT Austin, ongoing work extends this subsystem with an interface based on predicate logic, allowing the knowledge base to be seamlessly integrated into task planning. Core to this approach is the ability to reason about hypothetical objects that are requested by users but not yet perceived by the robot. This capability is crucial to our solution of incomplete commands in earlier versions of the *EGPSR* test. Details on our knowledge representation and planning system can be found in [2].

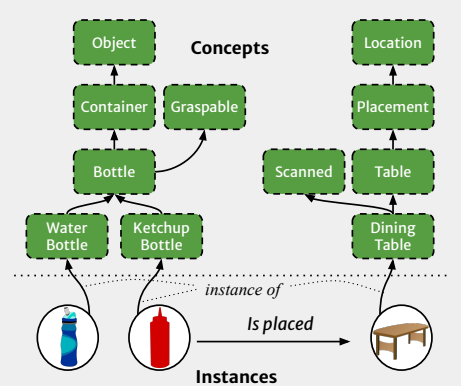


Fig. 2: Visualization of a knowledge base grounded in the robot's perception.

In parallel, the Lisbon team has developed two open-source toolboxes [3,4] that consolidate modeling, planning, and execution algorithms into a single package. These toolboxes implement state-of-the-art approaches for representing multi-robot systems using generalized stochastic Petri nets with rewards (GSPNRs). In addition, an innovative algorithm was introduced to streamline the model design process by automatically generating a GSPNR from a topological map and coordinating the system according to a specified policy.

2.3 Human-Robot Communication and Task Planning Using Foundation Models

Reliable human–robot communication is crucial for tasks like *GPSR* and *E-GPSR*, where spoken instructions must be converted from speech-to-text (STT) on the fly and translated into structured commands the controller can execute.

On the UT Austin side, our *current* STT pipeline runs on the HSR (Human Service Robot) platform in ROS 1. A custom node implements a two-stage ASR pipeline in which Vosk performs streaming recognition and decides when to stop listening, and a local Whisper model refines the transcription using task-specific hint phrases. The final transcript is exposed via a ROS service and topic, providing an on-board, low-latency STT server. We *plan* to port this design to the Booster T1 humanoid in ROS 2, using `whisper_ros` or a similar wrapper for streaming Whisper with voice activity detection and reusing the same STT interface.

For command understanding on the Austin side, we follow our 2024 approach: a large language model (e.g., GPT-4o) converts the STT transcript into a structured JSON object aligned with the RoboCup command grammar. The LLM is prompted to correct common ASR errors and ensure the resulting schema is valid. For template-based tasks (*GPSR*/*EGPSR*), the JSON is transformed into a state machine; for free-form tasks, the system can also produce other structured formats such as PDDL problem descriptions [5]. In the Booster T1 deployment, both Austin and Lisbon pipelines are intended to plug into this shared structured-command interface.

In a separate line of work on the Lisbon side, we have developed a lightweight pipeline that leverages LLMs to translate natural language instructions directly into executable robot actions [6]. Unlike traditional architectures that separate natural language understanding and planning, this approach integrates reasoning and execution within a single model. To ensure onboard autonomy, we fine-tuned a compact LLM that runs directly on the robot, eliminating the need for external APIs or cloud-based processing. The system is designed to be resilient to STT errors by incorporating multiple transcription hypotheses directly into the LLM prompt, allowing the model to reason over alternative interpretations of the user's utterance. All components of this work, including the codebase, trained models, and datasets, have been released as open source to promote transparency and reproducibility.

For speech synthesis, our stack supports text-to-speech (TTS) via `tts_ros` (Coqui TTS) or lightweight local engines such as Kokoro. Audio is played through `audio_common` to the robot's internal or external speaker, allowing the robot to confirm user commands, request clarifications, and provide status updates.

2.4 Manipulation

Grasp Sampling Our semantic perception system provides instance-level point clouds and 3D bounding boxes for objects of interest. We have integrated two grasp pose generators. The first is a state-of-the-art model GraspGen [7]. The

model is trained for diffusion-based 6-DOF robotic grasp generation from table-top manipulator data. As we are using a booster t1 humanoid this year, and we have found that GraspGen works the best for the humanoid after transforming the target object's point cloud to look like it came from a top-down camera pointed at the surface. We use Grounded-SAM [8] to segment the objects, and GraspGen to detect dense grasp poses on most objects including those with complex geometries. We post-process the poses and rank them according to their scores provided by the model.

Motion Planning Once the gripper's target poses are determined, collision-free joint trajectories need to be planned in order for the robot to achieve a desired pose. Our solution uses CuRobo [9], a CUDA accelerated motion planning library with custom configurations for various pick and place scenarios. The motion planner first plans as if the humanoid has an omnidirectional base with a lifting column instead of legs. This simplifies the motion planning problem and removes the need for the motion planner to consider the dynamics of the robot and the world. When following the plan, the upper body of the humanoid can directly follow the trajectories from the motion planner. For the lower body, we plan to train a flow matching policy that outputs the trajectory of the lower body conditioned on the head pose and the height of the waist. The role of the trained policy is to balance the robot while following the trajectory from the motion planner.

The bounding boxes of collision objects and surfaces from the perception module are populated into the collision world. Since motion planning takes a significant amount of time, reducing this bottleneck greatly improves the efficiency of the robot. For tasks such as *Storing Groceries*, the robot has to repeatedly visit the same location to manipulate objects. We have employed several strategies to speed up the manipulation pipeline. First, we pick up the objects in the ascending order of their distances to the edge, so the number of potential collisions are reduced. Second, we wrap the motion planning module in a concurrence container of the state machine, so that motion planners can be computed in parallel with execution.

2.5 Locomotion

Stable and robust locomotion are critical for the robot to navigate the arena, especially given the added complexity of dynamic bipedal walking on humanoid robots as opposed to statically stable or wheeled systems.

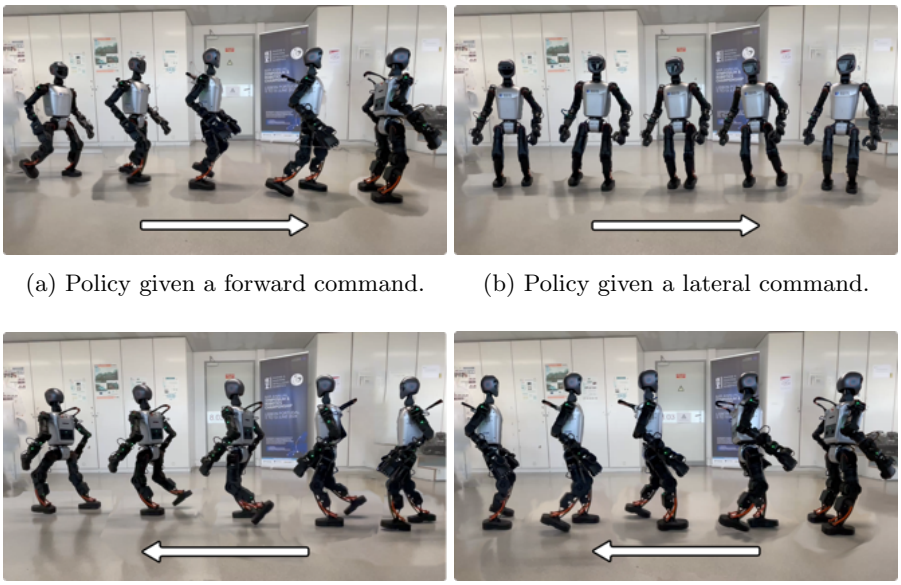
Our robot is equipped with long, highly articulated arms that provide many degrees of freedom for complex grasps and manipulation behaviors, but also introduces substantial additional mass on the upper body, making balance and whole-body stabilization more difficult to maintain, especially during execution.

Furthermore, because the robot's locomotion and manipulation will operate jointly, the system must remain stable across a wide range of arm configurations both when standing and when walking. This requirement exposed limitations of

the T1 motion model provided by BOOSTER, which did not achieve sufficient stability for large configuration changes of the upper body during manipulation.

To address this, we chose to develop our own locomotion policy, with omnidirectional movement (as showcased in Figure 3), trained using Twin-Delayed Deep Deterministic Policy Gradient [10], more specifically FastTD3 [11], which introduces parallel simulation, large-batch updates, and a distributional critic to the algorithm. This reinforcement learning algorithm was chosen for its reliable and fast convergence towards robust walking behaviors. This choice enabled us to accelerate the training–validation–deployment cycle, allowing rapid development on locomotion behaviors while preserving stability characteristics.

Policy deployment was performed through the official code structure for locomotion of the T1 robot [12], whilst the policy was verified via sim-to-sim validation across MuJoCo [13], Isaac Sim [14], and Webots [15]. However, both the training environment and the validation-deployment pipeline had to be further adapted to support our T1 model (i.e., with the high degree of freedom manipulation arms), as there did not exist any previous work on training and deploying a locomotion policy for this specific T1 model.



(a) Policy given a forward command. (b) Policy given a lateral command.

(c) Policy given a forward command. (d) Policy given a backwards command.

Fig. 3: Developed locomotion policy deployment on the real robot.

The qualification video further showcases the omnidirectional capabilities of the policy deployed on the real robot, including forward, backward, lateral, and rotational motions, as well as a direct comparison with the original BOOSTER

T1's motion model, where a noticeable improvement in robustness and gait fluidity is observed.

The future plan is to use the locomotion policy with the upper body motion plan and train a flow-matching policy that outputs a whole body motion for combined manipulation and locomotion for efficient task execution.

3 Conclusions

LisTex United is the RoboCup@Home team resulting from a joint research project between UT Austin and ISR-Lisboa, funded by the UTAustin-Portugal program (project ref https://doi.org/10.54499/2024.14126.UTA), "FOMO-HODOR: FOundation MOdels for HumanOid DOmestic RObots", web page `https://irsgroup.isr.tecnico.ulisboa.pt/fomo-hodor/`.

The project considers three research tasks:

- improving the use of LLMs in robot task planning, by decomposing high-level tasking into steps to be executed;
- addressing the engineering challenges of integrating LLMs with components of a robot's AI architecture, with the aim of enabling effective interactions with human users, and targeting participation in RoboCup@Home, i.e., a competition where robots perform domestic tasks;
- enabling whole body manipulation for humanoid embodiment.

Within the RoboCup@Home competition, there is a task called General-Purpose Service Robot (GPSR), in which human operators speak commands to the robot, that has to plan and execute the sequence of actions composing the task assigned by the command. The teams collaborating on this project have extensive experience with RoboCup@Home and GPSR, and are developing a framework that integrates prior work by both teams, in robot architectures, knowledge representation, task planning and human-robot communication, with modules specific to a humanoid robot platform, the Booster T1 with fingered hands, such as locomotion, navigation and vision-based grasping.

We plan to show some of our research progresses in RoboCup@Home 2026, using a humanoid in a domestic environment populated with humans.

References

1. Yuqian Jiang, Nick Walker, Minkyu Kim, Nicolas Brissonneau, Daniel S Brown, Justin W Hart, Scott Niekum, Luis Sentis, and Peter Stone. Laair: A layered architecture for autonomous interactive robots. In *Proceedings of the AAAI Fall Symposium on Reasoning and Learning in Real-World Systems for Long-Term Autonomy (LTA)*, October 2018.
2. Yuqian Jiang, Nick Walker, Justin Hart, and Peter Stone. Open-world reasoning for service robots. In *Proceedings of the 29th International Conference on Automated Planning and Scheduling (ICAPS 2019)*, July 2019.

3. Carlos Azevedo and Pedro U Lima. A gspn software framework to model and analyze robot tasks. In *2019 IEEE International Conference on Autonomous Robot Systems and Competitions (ICARSC)*, pages 1–6. IEEE, 2019.
4. Carlos Azevedo, António Matos, Pedro U Lima, and Jose Avendaño. Petri net toolbox for multi-robot planning under uncertainty. *Applied Sciences*, 11(24):12087, 2021.
5. Bo Liu, Yuqian Jiang, Xiaohan Zhang, Qiang Liu, Shiqi Zhang, Joydeep Biswas, and Peter Stone. Llm+p: Empowering large language models with optimal planning proficiency, 2023.
6. Afonso Certo, Bruno Martins, Carlos Azevedo, and Pedro U. Lima. Large language model-based robot task planning from voice command transcriptions. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2025.
7. Adithyavairavan Murali, Balakumar Sundaralingam, Yu-Wei Chao, Jun Yamada, Wentao Yuan, Mark Carlson, Fabio Ramos, Stan Birchfield, Dieter Fox, and Clemens Eppner. Graspgen: A diffusion-based framework for 6-dof grasping with on-generator training. *arXiv preprint arXiv:2507.13097*, 2025.
8. Tianhe Ren, Shilong Liu, Ailing Zeng, Jing Lin, Kunchang Li, He Cao, Jiayu Chen, Xinyu Huang, Yukang Chen, Feng Yan, Zhaoyang Zeng, Hao Zhang, Feng Li, Jie Yang, Hongyang Li, Qing Jiang, and Lei Zhang. Grounded sam: Assembling open-world models for diverse visual tasks, 2024.
9. Balakumar Sundaralingam, Siva Kumar Sastry Hari, Adam Fishman, Caelan Garrett, Karl Van Wyk, Valts Blukis, Alexander Millane, Helen Oleynikova, Ankur Handa, Fabio Ramos, Nathan Ratliff, and Dieter Fox. Curobo: Parallelized collision-free robot motion generation. In *2023 IEEE International Conference on Robotics and Automation (ICRA)*, pages 8112–8119, 2023.
10. Scott Fujimoto, Herke Hoof, and David Meger. Addressing function approximation error in actor-critic methods. In *International conference on machine learning*, pages 1587–1596. PMLR, 2018.
11. Younggyo Seo, Carmelo Sferrazza, Haoran Geng, Michal Nauman, Zhao-Heng Yin, and Pieter Abbeel. Fasttd3: Simple, fast, and capable reinforcement learning for humanoid control. *arXiv preprint arXiv:2505.22642*, 2025.
12. Yushi Wang, Penghui Chen, Xinyu Han, Feng Wu, and Mingguo Zhao. Booster gym: An end-to-end reinforcement learning framework for humanoid robot locomotion. *arXiv preprint arXiv:2506.15132*, 2025.
13. Emanuel Todorov, Tom Erez, and Yuval Tassa. Mujoco: A physics engine for model-based control. In *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 5026–5033. IEEE, 2012.
14. NVIDIA. Isaac Sim.
15. Webots. http://www.cyberbotics.com. Open-source Mobile Robot Simulation Software.

Booster Software and External Devices [OPL]

We use a T1 humanoid from *Booster Robotics*. No modifications have been applied to the base hardware, but the arms were endowed with 2 InspireRobots RH56DFX 6 DoF hands, necessary to manipulate objects in @Home.

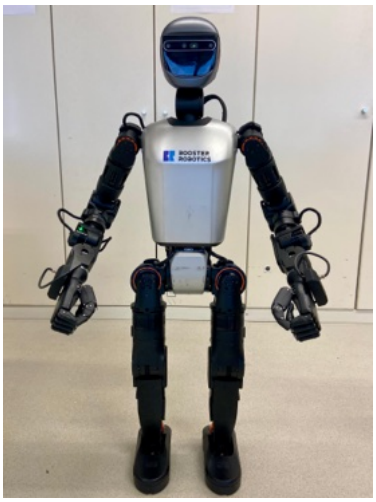


Fig. 4: Booster T1

Robot's Software Description

We are using the following 3rd party software:

- Object recognition: GroundedSAM
- People and activity recognition: YOLOv8
- Manipulation: GraspGen
- Knowledge Base: PostgreSQL
- Speech Recognition: Whisper, Vosk
- Planning and reasoning: Clingo, PDDLStream
- State Machine: SMACH (ROS)

External Devices

No external device planned on being used.

Cloud Services

We are using the following cloud services:

- Speech recognition: Google Cloud Speech API
- Large language model: GPT-5