Motion Planning for Mobile Robot Navigation Using a Toyota Human Support Robot

Robert Shi & Dr. Wolfgang Merkt & Dr. Ioannis Havoutis

UNIVERSITY OF OXFORD

DEPARTMENT OF ENGINEERING SCIENCE

Abstract

Service robots for home care applications require the ability to perform pick-and-place actions for object manipulation. It is common to split the pick-and-place task into navigation and manipulation-while-the-robot-is-stationary. This project builds on prior work on loco-manipulation planning and demonstrates a continuous pick-and-place motion. The following introduces a drive-by pick-and-place motion planning method in which a robot follows a predefined base trajectory and performs a grasping action without stopping the base motion. In particular, the drive-by pick-and-place method is implemented on a Toyota Human Support Robot (HSR) to achieve time-efficient, collision-free, whole-body trajectories in a static environment.

Motivation

- The HSR has an omni-directional base with 3 degrees-of-freedom and an arm with 5 degrees-of-freedom. The end effector is a gripper with two actuated fingers.
- Built-in functions allow the HSR to reach either a base or a gripper goal pose. This is sufficient to complete a pick-and-place action, but it is discontinuous and slow compared to a drive-by approach. The 2 methods are shown in Figure 1.

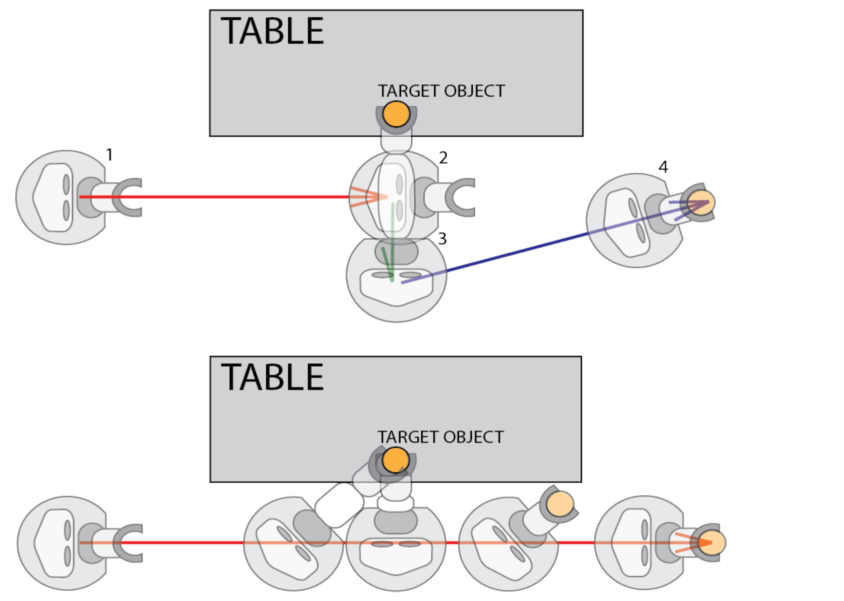


Figure 1: (Top) Pick-up part of a pick-and-place action using built-in HSR functions.
(Bottom) Drive-by pick-and-place action.

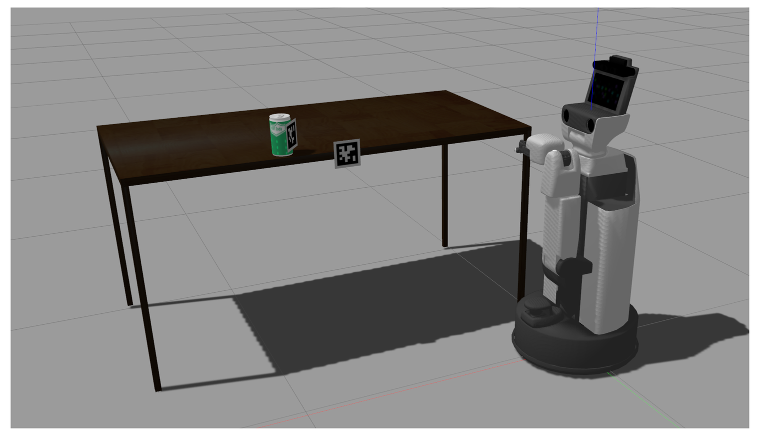


Figure 2: Screen shot of the problem setup in Gazebo

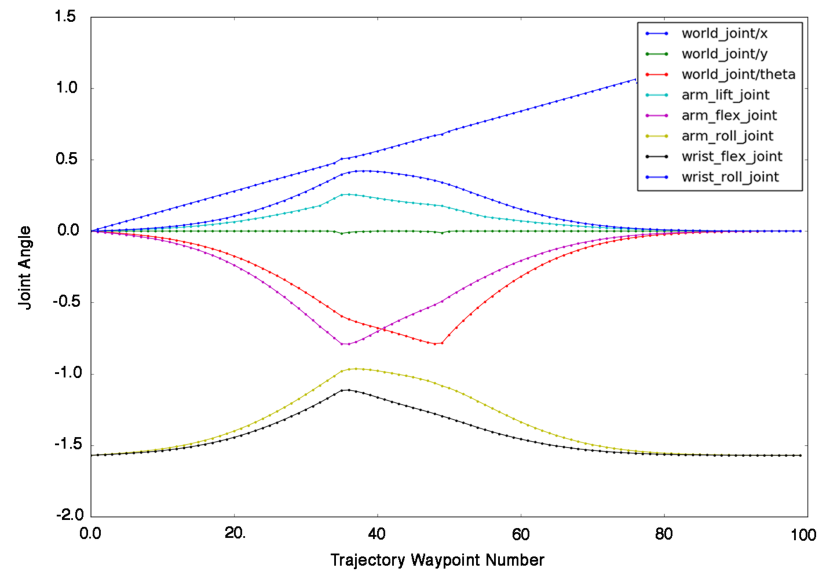


Figure 5 (left): Plot of the joint position values output from the AICO solver. The trajectory has 100 waypoints over a 10 second grasping period. The base trajectory given was a 1.4m linear, linearly spaced trajectory that runs parallel to the Ikea Nyboda table, see bottom of Figure 1 and Figure 2.

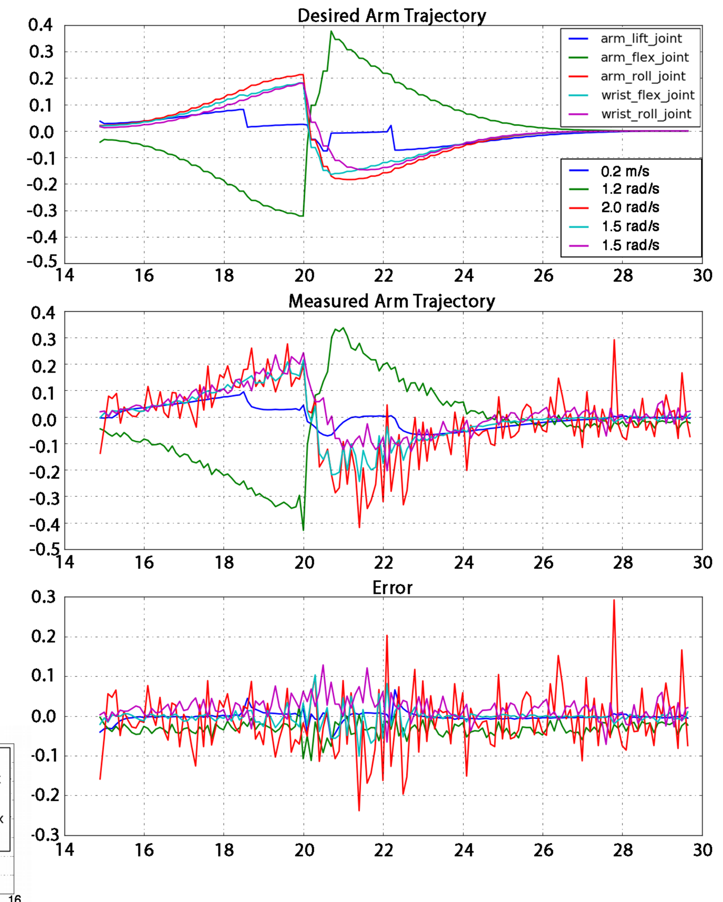


Figure 6 (right): Plots of HSR arm joint velocities at each waypoint. For each plot, the horizontal axis has units of time(sec) and the vertical axis has units of velocity. (Top): expected joint velocity values for the simulated HSR in Gazebo. (Middle): Velocity measurements sampled from the simulated HSR at 10 Hz. (Bottom): Plot of difference between velocity measurements and desired values. This velocity has a duration of 15 seconds, since the time step between the trajectory waypoints shown in Figure 5 was set to 0.15 seconds. The legends denote the HSR arm joints and the velocity limits.

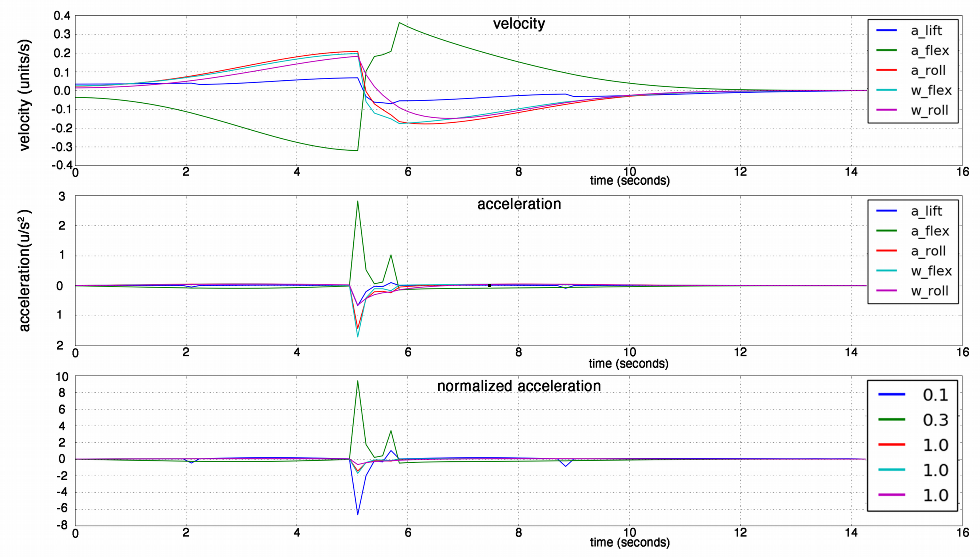


Figure 7 (left): Desired HSR arm joint Acceleration plot. (Top) Desired velocity plot. (Middle): Desired HSR acceleration values calculated by differentiating desired joint velocity values. (Bottom): Acceleration values normalized by the respective HSR arm joint acceleration limits. Acceleration limits are denoted in the legend.

RESULTS

Position Plots

- The HSR follows the position waypoints with negligible error.
- Figure 8 depicts the HSR accomplishing a drive-by pick-up action in simulation.

Velocity Plots

- Velocity waypoint values fall withing the HSR hardware limits.
- Significant noise in arm_roll_joint velocity measurements may be due to a low damping coefficient defined for the simulated HSR.

Acceleration Plots

- The HSR exceeds acceleration limits by a significant margin at the transition between sub-reaching and sub-grasping segments of the Grasping step.

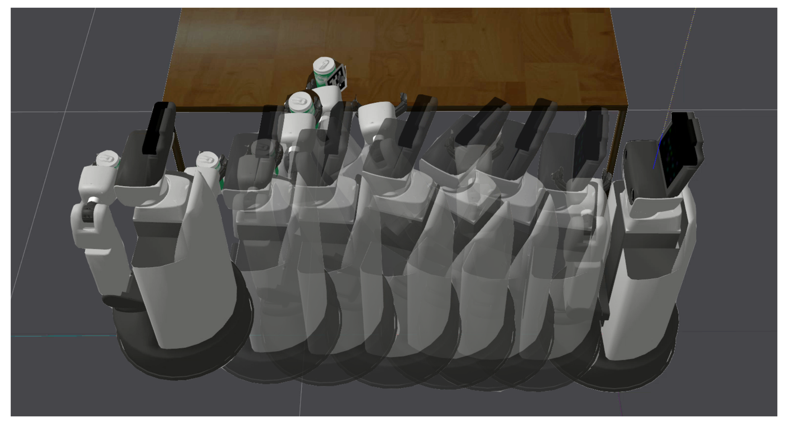
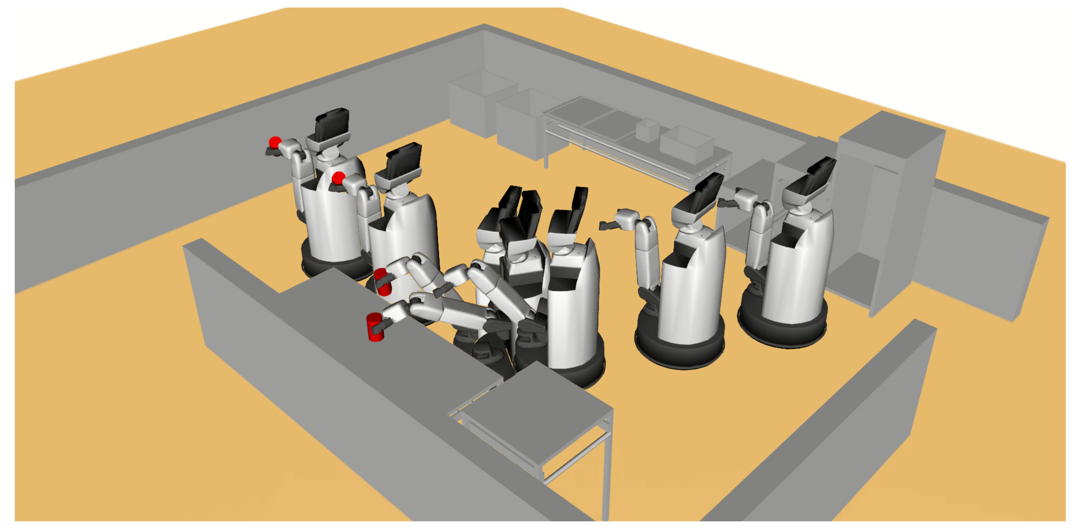


Figure 8: Frames of the grasping phase of the drive-by pice-and-place action simulated in Gazebo. This trajectory is plotted in Figures 5, 6, 7.

Figure 9: Frames of a planned pick-up action in Rviz. The base trajectory was planned using RRT-Connect with an intermediate goal configuration.



Approach

Problem Description

The HSR must follow a predefined base trajectory to drive by a table and pick up a target object without stopping the base.

Method

- This project follows the method proposed in [1].
- The trajectory is split into 3 steps: Reaching, Grasping, and Placing. See Figure 3.
  - Reaching and Placing steps require collision-free base motion. Only binary constraints are considered, so these steps are solved by a sampling-based planner.
  - The Grasping phase requires collision-free motion for the arm and base. The gripper must also reach the target object and stay around it until the object is fully grasped. Binary and differential constraints are considered so a trajectory optimization method is used. Figure 4 describes the taskmaps incorporated in the cost function.

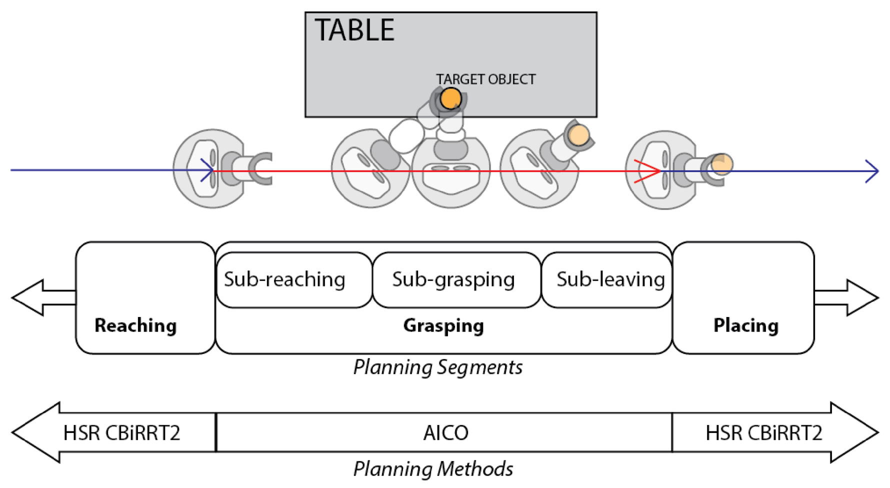


Figure 3: Overview of planning segments. Reaching and placing paths for the HSR base are found with sampling-based planners such as Constrained BiDirectional Rapidly Exploring Random Trees, implemented in the HSR software package.[2] The whole-body grasping segment is solved for with Approximate Inference Control (AICO)[3], an non-linear trajectory optimization solver, wrapped in EXOTica[4]. Figure 4 depicts the taskmaps used for the grasping segment.

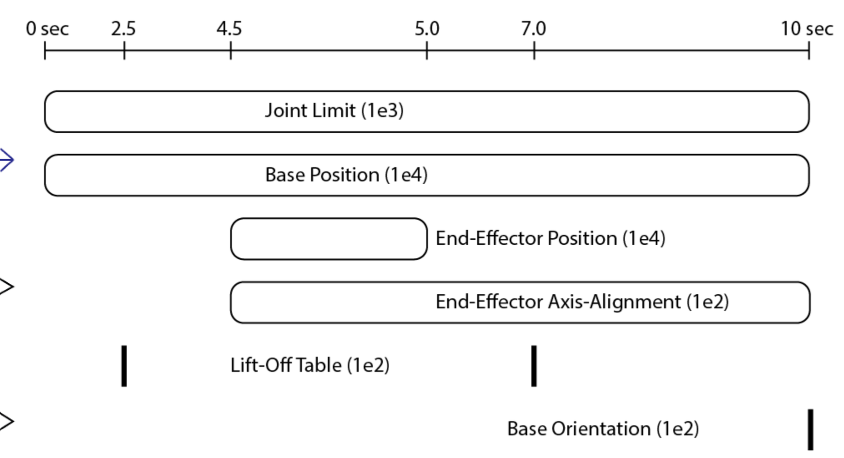


Figure 4: A graphical depiction of the timing of taskmap activation and task weights for the AICO solver [3] in EXOTica [4]. The graphic depicts the grasping segment, with 4.5-5.0 second interval is the sub-grasping phase shown in Figure 3. Elongated rounded rectangles denote taskmaps that are active for a duration of time. Bars represent task maps that push the HSR to achieve desired values at a point in time. The whole-body grasping trajectory is solved for a 10-second trajectory with waypoints every 0.1 seconds. The trajectory is scaled to accommodate for velocity constraints imposed by HSR hardware.

EVALUATION

- The focus of this project was on a drive-by pick-up action. However, a drive-by placing action follows a similar process, since an object is released instead of grasped.
- The drive-by pick-up action succeeds in simulation.
- Further testing of trajectories on the actual HSR is necessary to confirm and reduce acceleration values.
- Acceleration values may be reduced by further tuning of:
  - The timing of taskmaps/sub-segments of the Grasping segment.
  - The weights of HSR arm joints. Penalizing motion of the arm_flex and arm_lift joints would increase the motion in arm joints with higher acceleration limits.
  - The time step between trajectory waypoints. Increasing the duration of the trajectory would reduce velocity and acceleration.
- Further work:
  - Plan the base trajectory for the Grasping segment instead of assuming it is given. RRT-Connect is used in Figure 9.
  - Extend the drive-by pick-up planner to allow for drive-by placing of objects

DYNAMIC ROBOT SYSTEMS GROUP
OXFORD ROBOTICS INSTITUTE

Robert Shi | robert.shi@oriel.ox.ac.uk

REFERENCES

[1] Yang, Yiming, et al. “Planning in Time-Configuration Space for Efficient Pick-and-Place in Non-Static Environments with Temporal Constraints.” 2018 IEEE-RAS 18th International Conference on Humanoid Robots (Humanoids), 2018, pp. 1–9, doi:10.1109/HUMANOIDS.2018.8624989.
[2] Yamamoto, Takashi, et al. “Development of Human Support Robot as the Research Platform of a Domestic Mobile Manipulator.” ROBOMECH Journal, vol. 6, no. 1, 2019, p. 4, doi:10.1186/s40648-019-0132-3.
[3] Toussaint, Marc. “Robot Trajectory Optimization Using Approximate Inference.” Proceedings of the 26th Annual International Conference on Machine Learning - ICML ’09, ACM Press, 2009, pp. 1–8, doi:10.1145/1553374.1553508.
[4] Ivan, Vladimir, et al. “EXOTica: An Extensible Optimization Toolset for Prototyping and Benchmarking Motion Planning and Control.” Robot Operating System (ROS), edited by Anis Koubaa, vol. 778, Springer International Publishing, 2019, pp. 211–40, doi:10.1007/978-3-319-91590-6_7.