A MIP-Based Approach for Multi-Robot Geometric Task-and-Motion Planning

Hejia Zhang, Shao-Hung Chan, Jie Zhong, Jiaoyang Li, Sven Koenig, Stefanos Nikolaidis

I. INTRODUCTION

Geometric task-and-motion planning (GTAMP) is an important subclass of task-and-motion planning (TAMP) where the robot has to move several objects to regions in the presence of other movable objects [1]. We focus on *multi-robot geometric task-and-motion planning* (MR-GTAMP), where the robots have to collaborate to move several objects to regions in the presence of movable obstacles.

We refer the reader to the full paper version of this work for more details [2].

II. PROBLEM FORMULATION

In a MR-GTAMP problem, we have a set of robots, a set of fixed rigid objects, a set of $n_{\mathbf{M}}$ movable rigid objects $\mathbf{M} = \{M_i\}_{i=1}^{n_{\mathbf{M}}}$ and a set of $n_{\mathbf{Re}}$ regions $\mathbf{Re} = \{Re_i\}_{i=1}^{n_{\mathbf{Re}}}$. We assume that all objects and regions have known and fixed shapes.

Each object has a configuration, which includes its position and orientation. Each robot has a configuration defined in its base pose space and joint space.

We want to find a sequence of joint pick-and-place actions of multiple robots to change the configuration of the objects to satisfy goal specification that is specified in form of a conjunction of statements of the form $\textsc{InRegion}(M, Re)$, where $M \in \mathbf{M}$ and $Re \in \mathbf{Re}$, which is true *iff* object M is contained entirely in region Re.

III. OUR APPROACH

We present our two-phase MR-GTAMP framework (Fig. 1) in this section. In the first phase, we compute the collaborative manipulation information, i.e., the occlusion and reachability information for individual robots and the potential collaborative relationships between the robots (Sec. III-A). In the second phase, we use a Monte-Carlo Tree Search exploration strategy to search for task-and-motion plans (Sec. III-B). The search process depends on a key component that generates promising task skeletons (Sec. III-C) and a key component that finds feasible object placements and motion trajectories for the task skeletons to construct executable task-and-motion plans (Sec. III-D).

Hejia Zhang, Shao-Hung Chan, Jie Zhong, Jiaoyang Li, Sven Koenig and Stefanos Nikolaids are with the Department of Computer Science, University of Southern California, Los Angeles, USA {hejiazha,shaohung,jzhong54,jiaoyanl,skoenig, nikolaid}@usc.edu.

A. Computing Collaborative Manipulation Information

Given a MR-GTAMP problem instance and the initial configurations of all objects and robots, our framework first computes the occlusion and reachability information for individual robots, e.g., whether an object blocks a robot from manipulating another object and whether a robot can reach a region to place an object there. We also compute whether two robots can perform a handover action for an object by computing whether they can both reach a predefined handover point to transfer the object.

B. Searching for Task-and-Motion Plans

Our search process for efficiently finding high-quality collaborative task-and-motion plans generates a search tree.

We propose a Monte-Carlo Tree Search (MCTS) exploration strategy to balance exploration and exploitation.

C. Key Component 1: Generating Promising Task Skeletons

One key component in the second phase (Sec. III-B) of our framework is to generate promising task skeletons $\{\bar{\mathbf{S}}\}$, i.e., sequences of actions without the placement and trajectory information, for moving a set of objects $\mathbf{M}^*$ given a sequence of already grounded joint actions $\mathbf{S}'$.

Building the collaborative manipulation task graph. To reason about the collaborative manipulation capabilities of the individual robots, we encode the computed information as a graph. We build a *collaborative manipulation task graph* (CMTG) (Fig. 2) to capture the precedence of the manipulations of different objects, i.e., we can only move an object after we move the obstacles that block the pick-and-place action we are going to execute, based on the computed information from the first phase (Sec. III-A).

Mixed-integer linear program formulation and solving. Given a CMTG $\mathbf{C}$, we find a set of task skeletons that specify which robot will move which object at each time step. Given a time step limit T, we cast the problem of finding a task skeleton that has a minimum number of objects to be moved as a mixed-integer linear program (MIP).

D. Key Component 2: Task-Skeleton Grounding

The second key component in the search phase (Sec. III-B) is to ground the task skeletons, i.e., to find the object placements and motion trajectories for the partially grounded pick-and-place actions. We use a reverse search algorithm inspired by [3].

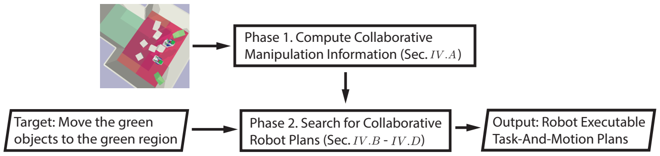

Fig. 1: Overview of the proposed framework.

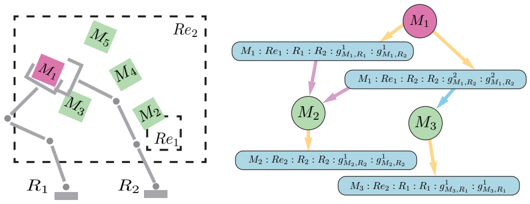

Fig. 2: (Left) An example scenario where we want to generate task skeletons to move object $\mathbf{M}_1$ given an empty sequence of grounded joint actions. (Right) The corresponding *collaborative manipulation task graph* for moving object M_1. The rounded rectangular nodes are *action nodes*. The circular nodes are *object nodes*. The red circular nodes represent objects that are specified to be moved. The yellow arrows represent *action edges*. The purple arrows represent *block-place edges*, and the blue arrow represents a *block-pick edge*.

REFERENCES

[1] B. Kim, L. P. Kaelbling, and T. Lozano-Pérez, "Adversarial actor-critic method for task and motion planning problems using planning experience," in *AAAI Conference on Artificial Intelligence*, vol. 33, no. 01, Jul. 2019, pp. 8017–8024.

[2] H. Zhang, S.-H. Chan, J. Zhong, J. Li, S. Koenig, and S. Nikolaidis, "A mip-based approach for multi-robot geometric task-and-motion planning."

[3] M. Stilman, J.-U. Schamburek, J. Kuffner, and T. Asfour, "Manipulation planning among movable obstacles," in *IEEE International Conference on Robotics and Automation*, 2007, pp. 3327–3332.