On Benchmarks for Combined Task and Motion Planning

Fabien Lagriffoul

***Abstract*—Planning systems developed for Combined Task and Motion Planning (CTAMP) problems are most of the time evaluated using their own benchmarks. As a direct consequence, no comparison of these systems is currently possible, and as a side effect, there is a risk for planners to overspecialize. We compare several CTAMP benchmarks from three different perspectives (logical, geometric, dependency). In this rough projection of the space of problems, we point out a bias towards certain regions, while another region is rarely explored in recently proposed benchmarks.**

I. Introduction

CTAMP has become an active field of research in the AI and Robotics communities, as reflected by the emergence of dedicated tracks and workshops in the main conferences within these fields. In 2009, early attempts to combine logical and geometric search spaces originated both from AI, e.g., Semantic Attachments [2], and robotics communities, e.g., aSyMov [1]. Since then, a multitude of approaches have flourished, together with their own experimental evaluation methods, but no common benchmark has been established yet.

Over the years, task planning has established a set of benchmarks. These benchmarks represent a small subset of all possible planning problems, whose diversity ensures planners not to overspecialize on specific domains. A fair comparison of algorithms is supported by theoretical results, i.e., planners compete on the same classes of problems (planning/scheduling, satisficing/optimal, deterministic/probabilistic). There is no reason why the same cannot be done for CTAMP problems. A recurrent argument is that comparing CTAMP planners is difficult because of differences between robotic platforms. However, the platform is not as crucial as for robotic benchmarks, if full observability and determinism of actions are assumed (which is often the case). Then, a fair comparison is possible by imposing a minimal set of requirements that preserve the essence of the problem (see for instance benchmarks (4) and (5) in Fig. 3). We hope through this abstract and our poster to initiate discussions about CTAMP benchmarking during the workshop.

Establishing benchmarks is important for developing *general purpose* planning techniques for robots, and for better understanding CTAMP problems. These benchmarks should be a representative sample of the space of problems. The benchmarks presented along this paper are *not* representative of the space of possible problems, but they illustrate three features of CTAMP problems on which we focus in this paper: *logical* difficulty, *geometric* difficulty and *dependency*. Our hypothesis is that recent work is biased towards geometrically difficult problems and problems with weak logic-geometric dependencies. On the other hand, logically difficult problems with strong logic-geometric dependencies are under-represented.

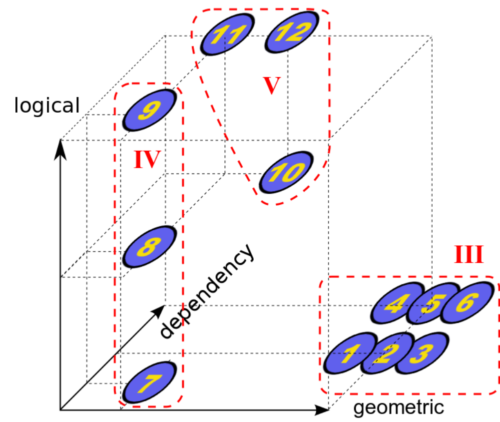

Fig. 1. Three-axis model. In red, the three types of problems discussed in the corresponding sections.

II. Three-axis model

The benchmarks are compared using a 3-axis chart (see fig. 1), where each axis represents the “difficulty” of problems seen from three different perspectives:

- logical;
- geometric;
- dependency.

Precisely defining the difficulty of a problem is not easy. A better definition than what follows is left open for discussions.

For the *logic* and *geometry* axes, we estimate the difficulty of the problem in terms of *subgoal interaction*. Subgoal interaction is well known in task planning. It prevents from solving subproblems sequentially, which usually requires more backtracking. Subgoal interaction applies to path planning as well [13]. We consider the problem which remains to be solved assuming that a solution at the discrete level is known, i.e., assuming that we know which modes [5] are to be used, and in which order. Assuming that, there still remains to choose correct instances for grasps and placements that do not *interfere* with each other, which we define as the geometric difficulty.

The *dependency* axis represents the dependencies between logical and geometric levels. Dependencies originate from the fact that the connectivity of the configuration space (CS) is

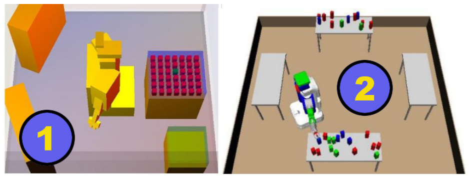

Fig. 2. Three variations of the clutter problem. The goal is to bring one or several objects surrounded by movable obstacles to a target location(s). In (3), the goal is to grasp the red object.

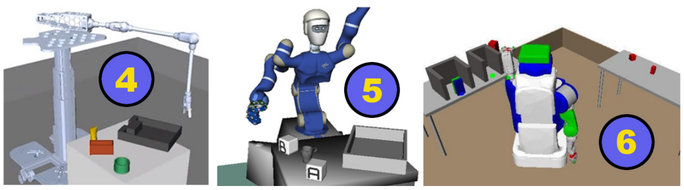

Fig. 3. Havur et al. experiment on rearrangement planning of multiple objects (4) uses a mobile base with a 4-DoF manipulator. Our setup (5) uses the 7-DoF manipulator of the humanoid robot Justin (courtesy DLR) in fixed position. The goal is to exchange the positions of the tray and the red object / cup. Both setups are equivalent if (i) the humanoid robot uses only one arm, (ii) the arm can place objects at any pose on the table (iii) the dimensions of the objects relative to the dimension of the table is the same. In (6), the task is to put the green block at the green point and the blue block at the blue point.

changed by logical actions, and that some logical actions are not feasible from certain subspaces of the CS. Hence, dependencies make the order of logical actions strongly constrained by the geometric level.

Next, we will consider three main families of benchmarks: benchmarks with emphasis on geometry (section III), benchmarks with weak dependencies (section IV), and benchmarks with logic reasoning and strong dependencies (section V).

III. Benchmarks with Emphasis on Geometry

Though labeled with "CTAMP", these problems are essentially high dimensional manipulation planning problems [10], also related to Multimodal Path Planning [5], Navigation Among Movable Obstacles (NAMO) [12] or Rearrangement Planning [8].

Clutter-like benchmarks have been used to evaluate several planners [3, 4, 11], and come under different variants (1, 2, 3 in Fig. 2). At the geometric level, all of them are difficult due to goal interaction caused by the lack of free space. (1) is somewhat simpler because there is free space on the table on the left, (2) presents more subgoals with possible interactions, and (3) less subgoals but greater chance of interaction because of the limited space on the table. At the logical level, these problems are large, but easy, because the only logical constraint is that only one object can be manipulated at a time. Subgoals can be achieved sequentially, and the ordering of actions has limited impact on the resolution.

Rearrangement benchmarks are also essentially difficult at the geometric level, owing to strong goal interaction. Even knowing the sequence of logical actions requires to carefully select objects poses that do not interact. They have been used in recent work [6, 9, 4] (4, 5, 6 in Fig. 3). Benchmark (6)

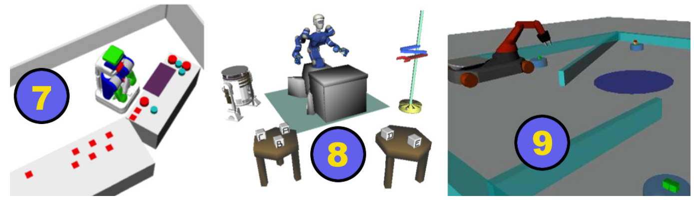

Fig. 4. In Srivasta et. al.'s Dinner domain (7), the task is to bring the objects to target locations on the empty table. Lagriffoul and Andres' cleaning scenario (8), the task is to have the dirty blocks cleaned (by the humanoid robot) and back at their initial location. The towers of Hanoi challenge scenario (9), by Havur et. al.

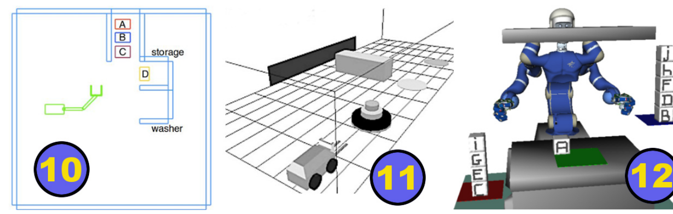

Fig. 5. The washing problem by Lozano-Perez et. al. (10), the goal is to wash block A and place it in the storage. In (11), a variation of the tower of Hanoi problem by Cambon et. al. In (12), the blocks-world-3D problem by Lagriffoul and Andres, the goal is to build an ordered pile at any location using top-grasps, the obstacle above the table prevents from stacking more than 2 blocks on the table (handover is not allowed).

is similar to (4,5) in terms of geometric goal interaction, but brings an additional dimension to the problem in terms of grasp choice.

Both Clutter-like and Rearrangement benchmarks have dependencies because occlusions (1,2,3), available space w.r.t object size (4,5) and grasping poses (6) change the connectivity of the CS. However, the dependencies are stronger in Rearrangement problems, because the ordering of actions is more subject to the geometric constraints than in Clutter-like problems.

IV. Benchmarks with Weak Dependencies

Some examples of this type of benchmark are represented in Fig. 4. At the logical level, (7) is simple for the same reasons that clutter-like problems are. In comparison, (8) is more difficult because the subgoal of being clean interacts with the subgoal of being at the target location. (9) is known to be logically difficult, achieving the goal requires to repeatedly achieve and destroy subgoals through the intermediate steps. At the geometric level, these problems are not too difficult.

The three problems have very weak dependencies between logical and geometric levels. The connectivity of the CS is not affected by logical actions, therefore all actions are geometrically feasible regardless of the ordering of actions at the logical level. In principle, one could solve the task planning problem first, and subsequently call a motion planner for each action.

V. Benchmarks with logic reasoning and strong dependencies

In the three problems represented in Fig. 5, there is subgoal interaction at the logical level, which requires substantial amount of logical reasoning as the size of the problem increases. (10) and (11) are equivalent to (8) and (9) in this respect. At the geometric level (10) and (12) present weak subgoal interaction (the choice of intermediate poses may matter), which is not the case in (11).

All these problems have strong dependencies. In (10), the dependencies stem from the initial configuration only, i.e., once the blocks B, C, D have been removed, the connectivity of the CS does not change much. In (11,12) the dependencies remain because they are caused by a fixed obstacle. In (11), each time a disc is placed at the central peg, the large disc cannot be moved from one side of the room to the other. In (12), the initial piles need to be unstacked on the table, and all the states with piles containing more than two blocks are unfeasible.

Because of the strong dependencies, these problems are not amenable to precomputation of the connectivity of the CS, or caching of the geometric paths. On large instances, the logical state space becomes huge, therefore tightly interleaving logical and geometric reasoning is not a feasible approach.

VI. Discussion

We wanted through this comparison of different benchmarks to raise the matter of benchmarking for CTAMP. In order to be useful, benchmarks should cover a wide range of problem classes, which is currently not the case. There is a bias towards manipulation planning problems with little logical reasoning (and only *one* robot). Problems emphasizing logical reasoning have been used as benchmarks, but often these problems had weak dependencies, which is less challenging since task and motion planning can be done separately.

Problems with logical reasoning *and* strong dependencies have been used as benchmarks, but mostly in older work [1, 7], and with small problem instances. In our recent work [9], we have investigated how to *decouple* logic and geometric reasoning for addressing problems with logical reasoning and strong dependencies, problems which are seldom tackled in recent work.

Having said that, what are the next steps? Further theoretical work is needed to characterize CTAMP problems. Is it possible to establish a common language for describing CTAMP problems? Can we prove lower bounds on interesting sub-classes of problems? Can we draw on methods from other fields, e.g., hybrid reasoning? Can we think of a set of benchmarks to begin with? Which metrics should be used?

Acknowledgments

This work has been supported by the Swedish Knowledge Foundation (KKS) project Semantic Robots.

References

[1] S. Cambon, R. Alami, and F. Gravot. A hybrid approach to intricate motion, manipulation and task planning. *The Int. Journal of Robotics Research*, 28(1):104–126, 2009.

[2] Christian Dornhege, Patrick Eyerich, Thomas Keller, Sebastian Trüg, Michael Brenner, and Bernhard Nebel. Semantic attachments for domain-independent planning systems. In *Proc. of Int. Conf. on Automated Planning and Scheduling (ICAPS)*, pages 114–121, 2009.

[3] C. R. Garrett, Tomás Lozano-Pérez, and Leslie Pack Kaelbling. Heuristic search for task and motion planning. In *Proc. of the Workshop on Planning and Robotics (PlanRob)*, pages 148–156, 2014.

[4] Caelan Reed Garrett, Tomas Lozano-Perez, and Leslie Pack Kaelbling. Backward-forward search for manipulation planning. In *Int. Conf. on Intelligent Robots and Systems (IROS)*, 2015.

[5] Kris Hauser, Victor Ng-Thow-Hing, and Hector Gonzalez-Baños. Multi-modal motion planning for a humanoid robot manipulation task. In *Proc. of the Int. Symp. on Robotic Research*, pages 307–317, 2007.

[6] Giray Havur, Guchan Ozbilgin, Esra Erdem, and Volkan Patoglu. Geometric rearrangement of multiple movable objects on cluttered surfaces: A hybrid reasoning approach. In *2014 IEEE International Conference on Robotics and Automation, ICRA 2014, Hong Kong, China, May 31 - June 7, 2014*, pages 445–452, 2014.

[7] Leslie Pack Kaelbling and Tomás Lozano-Pérez. Hierarchical task and motion planning in the now. In *Proc. of Int. Conf. on Robotics and Automation (ICRA)*, pages 1470–1477, 2011.

[8] Athanasios Krontiris and Kostas Bekris. Dealing with difficult instances of object rearrangement. In *Proceedings of Robotics: Science and Systems*, Rome, Italy, July 2015.

[9] Fabien Lagriffoul and Benjamin Andres. Combining task and motion planning: A culprit detection problem. *The International Journal of Robotics Research*, 2016.

[10] Thierry Simeon, Jean-Paul Laumond, Juan Cortes, and Anis Sahbani. Manipulation Planning with Probabilistic Roadmaps. *The International Journal of Robotics Research*, 23(7-8):729–746, August 2004.

[11] S. Srivastava, E. Fang, L. Riano, R. Chitnis, S. Russell, and P. Abbeel. Combined task and motion planning through an extensible planner-independent interface layer. In *IEEE Int. Conf. on Robotics and Automation (ICRA)*, 2014.

[12] M. Stilman, J.-U. Schamburek, J. Kuffner, and T. Asfour. Manipulation planning among movable obstacles. In *Robotics and Automation, 2007 IEEE International Conference on*, pages 3327 –3332, april 2007.

[13] D.J. Zhu and Stanford University. Computer Science Dept. *Exploring the Interaction of Geometry and Search in Path Planning*. Computer Science Department: Report STAN CS. Stanford University, 1992.