A Hybrid Reasoning Approach to Geometric Rearrangement of Multiple Movable Objects on Cluttered Surfaces

Esra Erdem[1] Giray Havur[1] Guchan Ozbilgin[2] Volkan Patoglu[1]

***Abstract*— We introduce a novel computational method for geometric rearrangement of multiple movable objects on a cluttered surface, where objects can change locations more than once by pick and/or push actions. This method consists of four stages: (i) finding tentative collision-free final configurations for all objects (all the new objects together with all other objects in the clutter) while also trying to minimize the number of object relocations, (ii) gridization of the continuous plane for a discrete placement of the initial configurations and the tentative final configurations of objects on the cluttered surface, (iii) finding a sequence of feasible pick and push actions to achieve the final discrete placement for the objects in the clutter from their initial discrete place, while simultaneously minimizing the number of object relocations, and (iv) finding feasible final configurations for all objects according to the optimal task plan calculated in stage (iii). For (i) and (iv), we introduce algorithms that utilize local search with random restarts; for (ii), we introduce a mathematical modeling of the discretization problem and use the state-of-the-art ASP reasoners to solve it; for (iii) we introduce a formal hybrid reasoning framework that allows embedding of geometric reasoning in task planning, and use the expressive formalisms and reasoners of ASP. We illustrate the usefulness of our integrated AI approach with several scenarios that cannot be solved by the existing approaches.**

I. Introduction

Successful deployment of robotic assistants in our society requires these systems to deal with high complexity and wide variability of their surroundings to perform typical everyday tasks robustly and without sacrificing safety. For instance, natural human environments (such as refrigerator shelves, desks and tables) are often cluttered. While performing everyday chores at home, rearrangement of such clutter needs to be routinely performed either to unclutter the environment or to make space for new objects.

Geometric rearrangement planning with multiple movable objects is a challenging problem, since this task not only requires manipulation of objects lying around on the cluttered surface, but also necessitates manipulation of the new objects to be placed. Furthermore, since the order of manipulation actions matters (it may not be possible to place an object before making enough space for it), and a feasible plan may require a single object be moved *multiple times* (for instance, to swap places of two or more objects in the clutter), geometric reasoning alone is not sufficient to solve these problems; planning of manipulation actions (i.e. pick-and-place, push operations), need to be integrated with the continuous geometric problem.

An extended version of this paper has appeared in the proceedings of IEEE ICRA 2014 [1]. This work is supported by TUBITAK Grants 111E116 and 113M422.

[1] E. Erdem, G. Havur, V. Patoglu are with Faculty of Engineering and Natural Sciences, Sabancı University, İstanbul, Turkey. {esraerdem,havurgiray,vpatoglu}@sabanciuniv.edu

[2] G. Ozbilgin is with Department of Electrical and Computer Engineering, Ohio State University, Columbus, OH, USA. Ozbilgin's work is carried out while studying at Sabancı University. ozbilgin.1@osu.edu

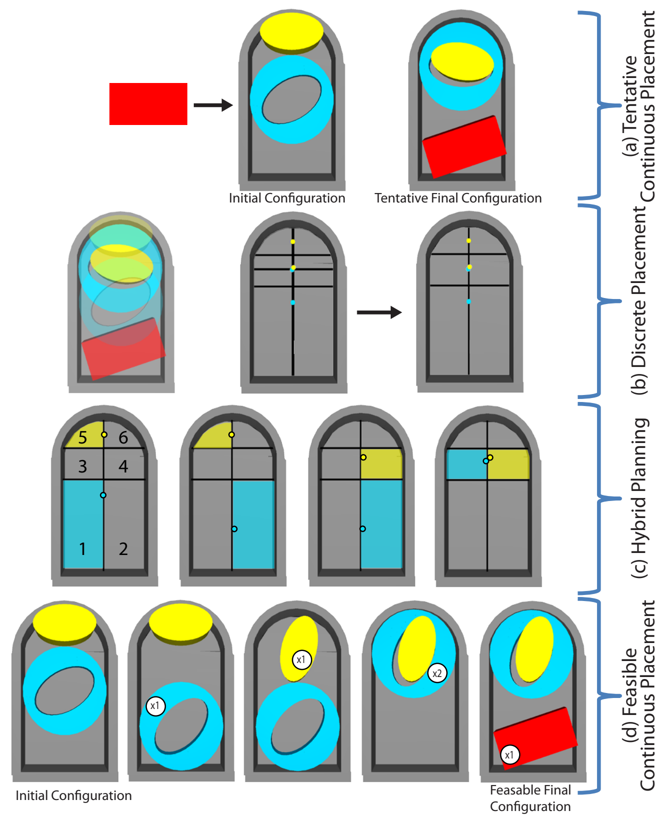

Fig. 1. Tight Placement Scenario

Motivated by these challenges, we introduce a novel multi-stage computational method that integrates various approaches of AI. Our method consists of four stages: tentative continuous placement, discrete placement, hybrid planning, and feasible continuous placement. Figure 1 illustrates our approach on a sample problem as detailed in Section III.

- **Tentative continuous placement** stage finds tentative collision-free final configurations for all objects (all the new objects together with all other objects in the clutter) while also trying to minimize the number of object relocations. Note that neither the order of move actions, nor the feasibility of achieving this tentative goal configuration through these actions are considered at this stage. For a tentative continuous placement, we introduce a local search algorithm that tries to maximize the total area of the surface covered by objects as the objects are placed. This algorithm utilizes a random sampling based collision detection to decide where to place objects on the surface, and in which orientation. Heuristics and random restarts are considered to avoid local optima.

- **Discrete placement** stage takes as input, the initial configurations and the tentative final configurations of all objects on the cluttered surface, and divides the surface into a *minimum* number of non-uniform grid cells. During gridization of the continuous plane, an object is allowed to (partially) span multiple grid cells as long as each grid cell contains the centroid (center of mass) of *a single object*. For discrete placement, we formalize the optimal gridization problem in Answer Set Programming (ASP) [2], [3]—an expressive logic-based formalism, and compute solutions using the award-winning, state-of-the-art ASP solver CLASP [4], [5].

- **Hybrid planning** stage aims to find a sequence of feasible move actions (i.e., pick and push actions) to achieve the final *discrete* placement of the objects in the clutter from their initial discrete placement, while simultaneously minimizing the number of object relocations. New objects are not considered at this stage as they can be placed to their (tentative) final locations on the surface after the clutter has been rearranged. We formulate the discrete rearrangement problem for the relocated objects, as a hybrid planning problem in the spirit of [6], and use the expressive formalisms and efficient solvers of ASP to solve it. As illustrated in [7], ASP provides a formal hybrid planning framework that combines high-level representation and logic-based reasoning with low-level geometric reasoning and feasibility checks to find optimal feasible plans. So, to increase feasibility of (discrete) plans, we embed some geometric constraints on continuous placement of objects (e.g., availability of a collision-free space on the table for each manipulation action, taking into account the other objects in the clutter as well as the robotic manipulator) in the logical formalism of ASP.

- **Feasible continuous placement** stage finds feasible final configurations for all objects according to the optimal task plan. Even though hybrid planning stage performs some continuous geometric checks to increase feasibility of calculated task plans, due to computationally intractable nature of the problem, embedding all continuous reasoning tasks in the domain description is not feasible. For feasible continuous placement, we introduce a local search algorithm that tries to minimize the number of collisions in an execution of the hybrid plan. This algorithm utilizes a random sampling based collision detection for objects that are manipulated, and random restarts to avoid local optima.

In both continuous placement stages, our approach can utilize domain-specific constraints, such as placing an object (e.g., a monitor) in a specific area on the surface (e.g., in the corner of the table). In hybrid planning stage, it can take into account domain-specific constraints as well: for instance, some objects may be heavy and cannot be picked by the robot but pushed.

Our framework also features several re-planning loops: In particular, in case the local search algorithm can not find a feasible placement after a predetermined number of random re-initializations, then it identifies the problematic manipulation action and asks the hybrid planner to return a new discrete plan that does not involve that action. This replanning loop continues until a feasible continuous placement is found, or the hybrid planner can not return any task plan. If no task plan can be found, then the whole framework is randomly re-initialized with a new tentative continuous placement.

It is important to emphasize here the tight coupling between task and geometric planning in our hybrid planning framework. Firstly, there exists a bilateral interaction between task planning and geometric reasoning: the logic based reasoner guides the probabilistic geometric reasoner by finding an optimal task-plan; if there is no feasible geometric solution for that task-plan then the geometric reasoner guides the task planner by modifying the planning problem with new temporal constraints. Secondly, we embed geometric reasoning in logic-based reasoner as described above while computing a task-plan; in that sense the geometric reasoner guides task planner to find geometrically feasible solutions. A systematic evaluation of various methods of integrating task planning with geometric reasoning, including the methods described above, can be found in an accompanying paper [8].

II. RELATED WORK

The computational problem of geometric rearrangement with multiple movable objects and its variations (like navigation among movable obstacles [9], [10], or nonprehensile manipulation under clutter [11], [12]) have been studied in literature subject to several restrictions due to their high computational complexity. Indeed, even a simplified variant with only one movable obstacle is proved to be NP-hard [13], [14]. The most common assumption that has been applied in most of the related literature is the restriction of manipulation plans to monotone plans–plans in which an object can be moved at most once. However, such a limitation causes failure when an object needs to be manipulated more than once, as seen in the scenario presented in Figure 1.

Some related works relax this monotonicity assumption by searching a manipulation solution in the robot C-space [15], [16], [17], or in the combined space of the robot and the objects altogether [18]. However, these methods are not computationally feasible for the problems with high-dimensional configuration spaces or with large number of movable objects.

Manipulation of the movable objects depends also on types of manipulation action. For instance, Cosgun et al. [19] tries to place an object on a cluttered surface, by first grasping the object, and then allowing this object to push other objects in the clutter to create a space for itself. Dogar and Srinivasa [11], [12] can accommodate both pick and place actions and non-prehensile actions such as pushes. However, these approaches are also restricted to monotone plans.

Our focus is to introduce a computational framework for general manipulation plans. That is, unlike many of the related studies in literature [9], [10], [18], [20], [21], we do not restrict ourselves to monotone plans. We consider pick and place actions as well as push actions. Furthermore, our approach computes feasible and optimal task plans, thanks to hybrid reasoning aspect of our method.

III. Tight Placement Scenario

The tight placement scenario and its solution are depicted in Figure 1. In this scenario, the rearrangement surface is a drawer with a semi-circular side. Initially, there are two movable objects in the drawer: a blue circle with a elliptical hole at the center and a yellow ellipse. The goal is to place a large red rectangular object in the drawer. The ellipse tightly fits into the circle with no contacts, and both the circle and the ellipse need to be placed at the semi-circular end of the table for the rectangular object to fit. The problem is challenging, since a solution requires the circular object to be moved at least twice, once to make space for the ellipse and the second time to clear enough space for the rectangle. This problem cannot be solved by approaches with monotonicity assumption, since a feasible plan always requires multiple relocations of an object.

First, a tentative continuous placement is calculated as in Figure 1(a) according to the local search strategy. Second, discrete placement is applied to this tentative continuous placement. The optimal grid shown in Figure 1(b) has 6 cells, while 16 cells would be considered with the naive approach of uniform gridization. According to this grid, initially the yellow ellipse is located in Cell 5, while the blue circle is in Cell 1. The final grid location for objects are as follows: the yellow ellipse is in Cell 4, the blue circle is in Cell 3, and the red rectangle is in Cell 1. Third, using the optimal grid, initial and final grid locations of the objects, a shortest hybrid plan is computed for the relocation of the objects. This plan, depicted in Figure 1(c), has 3 steps: place blue circle to Cell 2, place yellow ellipse to Cell 4, and place blue circle to Cell 3. Note that, once rearrangement on the surface is completed, the plan is appended with another place action to put red rectangle in Cell 1. As a last step, we check the feasibility of the task plan by using a local search technique. The results of the feasible continuous placement are presented in Figure 1(d).

IV. Discussion

We have proposed a novel computational method for geometric rearrangement of multiple movable objects on a cluttered surface, where the objects can be relocated more than once within a plan by pick and place or push actions. In particular, we have utilized local search and automated reasoning techniques at different stages of the problem to compute feasible solutions to this problem. We have taken advantage of the expressive representation languages of automated reasoners, for describing the optimal discretization problem and for embedding results of external computations (e.g., for geometric reasoning) during task planning. We have applied the proposed computational approach to several sample scenarios that cannot be solved by the existing approaches.

More detailed explanations of our method can be found in an extended version of this note [1]. Videos illustrating feasibility of our solutions through dynamic simulations with several mobile manipulators are available at `http://cogrobo.sabanciuniv.edu/?p=762`.

References

[1] G. Havur, G. Ozbilgin, E. Erdem, and V. Patoglu, "Geometric rearrangement of multiple movable objects on cluttered surfaces: A hybrid reasoning approach," in *Proc. of IEEE ICRA*, 2014.

[2] V. Lifschitz, "What is answer set programming?" in *Proc. of AAAI*. MIT Press, 2008, pp. 1594–1597.

[3] G. Brewka, T. Eiter, and M. Truszczynski, "Answer set programming at a glance," *Commun. ACM*, vol. 54, no. 12, pp. 92–103, 2011.

[4] M. Gebser, B. Kaufmann, A. Neumann, and T. Schaub, "clasp: A conflict-driven answer set solver," in *Proc. of LPNMR*. Springer, 2007, pp. 260–265.

[5] M. Järvisalo, D. Le Berre, O. Roussel, and L. Simon, "The international sat solver competitionss," *AI Magazine*, vol. 33, no. 1, p. 89, 2012.

[6] E. Erdem, K. Haspalamutgil, C. Palaz, V. Patoglu, and T. Uras, "Combining high-level causal reasoning with low-level geometric reasoning and motion planning for robotic manipulation," in *Proc. 2011 IEEE Int. Conf. Robotics and Automation (ICRA)*, 2011.

[7] E. Erdem, E. Aker, and V. Patoglu, "Answer set programming for collaborative housekeeping robotics: Representation, reasoning, and execution," *Intelligent Service Robotics*, vol. 5, no. 4, pp. 275–291, 2012.

[8] E. Erdem, V. Patoglu, and P. Schüller, "A systematic analysis of levels of integration between high-level task planning and low-level feasibility checks," in *Proc. RCRA 2014*, 2014.

[9] M. Stilman, J.-U. Schamburek, J. Kuffner, and T. Asfour, "Manipulation planning among movable obstacles," in *Proc. of IEEE ICRA*, 2007, pp. 3327–3332.

[10] M. Stilman and J. Kuffner, "Planning among movable obstacles with artificial constraints," *The International Journal of Robotics Research*, vol. 27, no. 11-12, pp. 1295–1307, 2008.

[11] M. Dogar and S. Srinivasa, "A framework for push-grasping in clutter," *Robotics: Science and Systems VII*, 2011.

[12] M. R. Dogar and S. S. Srinivasa, "A planning framework for non-prehensile manipulation under clutter and uncertainty," *Autonomous Robots*, vol. 33, no. 3, pp. 217–236, 2012.

[13] G. Wilfong, "Motion planning in the presence of movable obstacles," in *Proc. of the Fourth Annual Symposium on Computational Geometry*, ser. SCG '88, 1988, pp. 279–288.

[14] E. D. Demaine, M. L. Demaine, M. Hoffmann, and J. O'Rourke, "Pushing blocks is hard," *Comput. Geom.*, vol. 26, no. 1, pp. 21–36, 2003.

[15] J. Van Den Berg, M. Stilman, J. Kuffner, M. Lin, and D. Manocha, "Path planning among movable obstacles: a probabilistically complete approach," in *Algorithmic Foundation of Robotics VIII*. Springer, 2009, pp. 599–614.

[16] O. Ben-Shahar and E. Rivlin, "To push or not to push: on the rearrangement of movable objects by a mobile robot," *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics*, vol. 28, no. 5, pp. 667–679, 1998.

[17] P. C. Chen and Y. K. Hwang, "Practical path planning among movable obstacles," in *Proc. of IEEE ICRA*, 1991, pp. 444–449.

[18] J. Barry, K. Hsiao, L. P. Kaelbling, and T. Lozano-Pérez, "Manipulation with multiple action types," in *Experimental Robotics*. Springer, 2013, pp. 531–545.

[19] A. Cosgun, T. Hermans, V. Emeli, and M. Stilman, "Push planning for object placement on cluttered table surfaces," in *Proc. of IROS*, 2011, pp. 4627–4632.

[20] K. Okada, A. Haneda, H. Nakai, M. Inaba, and H. Inoue, "Environment manipulation planner for humanoid robots using task graph that generates action sequence," in *Proc. of IROS*, 2004, pp. 1174–1179.

[21] M. Stilman and J. J. Kuffner, "Navigation among movable obstacles: Real-time reasoning in complex environments," *International Journal of Humanoid Robotics*, vol. 2, no. 04, pp. 479–503, 2005.