A Manipulation Planner for Pick and Place Operations under Continuous Grasps and Placements *

T. Siméon, J. Cortés, A. Sahbani, J.P. Laumond

LAAS-CNRS
7, avenue du Colonel-Roche
31077 Toulouse Cedex - France
{nic, jcortes, asahbani,jpl}@laas.fr

Abstract

This paper addresses the manipulation planning problem which deals with motion planning for robots manipulating movable objects among static obstacles. We propose a manipulation planner capable of handling continuous domains for modeling both the possible grasps and the stable placements of a single movable object, rather than discrete sets generally assumed by the existing planners. The algorithm relies on a topological property that characterizes the existence of solutions in the subspace of configurations where the robot grasps the object placed at a stable position. This property leads to reduce the problem by structuring the search-space. It allows us to devise a manipulation planner that directly captures in a probabilistic roadmap the connectivity of sub-dimensional manifolds of the composite configuration space. First experiments demonstrate the feasibility and the efficiency of the approach.

1 Introduction

Several applications such as robot programming, design for manufacturing or animation of digital actors require some real or virtual artifact to move and manipulate objects within an environment composed of obstacles. Robot motion planning [11] classically consists of planning collision-free paths for robots moving amidst fixed obstacles. However, the presence of movable objects, i.e. objects that can only be moved by the robot, leads to a more complex version of the planning problem. For example to perform a Pick and Place operation the robot may be forced by the presence of joints limits or obstacle constraints to change grasp in order to place the object in its desired location. Changing the grasp requires to find some intermediate position at which the re-grasping of the object is possible. Also, the presence of movable objects makes possible for the robot to modify the structure of its environment by displacing an object that obstructs its motion. Planning a single robot motion may therefore require several subtasks: for instance, the robot should first move displace the obstructing object, and then reach its goal. In this example there are two motion subtasks.

Manipulation planning concerns the automatic generation of such sequences of collision-free motions. Its additional complexity compared to classical path planning is that such problems also consider the automatic decomposition of the manipulation task into elementary motion planning sub-tasks. The formal framework of manipulation planning [2] (see also Chapter 11 of Latombe's book [11]) shows that a solution is an alternate sequence of elementary motions, called transit and transfer paths, and separated by grasp and ungrasp operations. Several algorithms have been proposed to solve different instances of problems [2, 9, 3, 4, 1, 15]. While the existing planners generally assume discrete grasps and stable placements of the movable objects, dealing with continuous sets may allow more sophisticated planners to be devised. Hence, discrete sets suppose that the knowledge initially provided by the user contains some of the grasps and intermediate placements allowing to solve the problem; dealing with continuous sets allows the planner to automatically generate such grasps and placements.

The contribution of the paper is to propose a practical planner capable to deal with a continuous setting of the manipulation problem. The algorithm relies on a property established in [3]. This property recalled in Section 2 allows to reduce the problem by characterizing the existence of a solution in the lower dimensional subspace of configurations where the robot *grasps* the movable object *placed* at a stable position. Section 4 describes how the connected components of this subspace can be directly computed using an extension of

*This work is supported by the European Esprit Project 28226 MOLOG. See `http://www.laas.fr/molog` or [18].

the PRM-based visibility technique [17] capable to efficiently deal with closed-chain systems. This algorithm allows us to capture the connectivity of the search space into a small roadmap generally composed of a low number of connected components. Connections between these components using transit or transfer motions are then computed by solving a limited number of point-to-point path planning problems using the technique explained in Section 5. Finally, we discuss some experiments and the performance of the planner.

2 Manipulation Planning Problem

Let us consider a 3-dimensional workspace with a robot $\mathcal{A}$ and a movable object $\mathcal{M}$ moving among static obstacles[1] . The robot has n degrees of freedom and $\mathcal{M}$ is a rigid object with 6 degrees of freedom that can only move when it is grasped by the robot. Let C_{rob} and C_{obj} be the configuration spaces of the robot and the object, respectively. The composite configuration space of the system is $CS = C_{rob} \times C_{obj}$ and we call CS_{free} the subset in CS of all admissible configurations, i.e. configurations where the moving bodies do not intersect together or with the static obstacles.

Manipulation Constraints: A solution to a manipulation planning problem corresponds to a constrained path in CS_{free}. Such a solution path is an alternate sequence of two types of sub-paths verifying the specific constraints of the manipulation problem, and separated by grasp/ungrasp operations:

- *Transit Paths* where the robot moves alone while the object $\mathcal{M}$ stays stationary in a stable position. The configuration parameters of $\mathcal{M}$ remain constant along a transit path. Such motions allow to place the robot at a configuration where it can grasp the object. They are also involved when changing the grasp of the object.

- *Transfer Paths* where the robot moves while holding $\mathcal{M}$ with the same grasp. Along a transfer path, the configuration parameters q_{obj} of $\mathcal{M}$ change according to the grasp mapping induced by the forward kinematics of the robot: $q_{obj} = g(q_{rob})$.

Let P (resp. G) denote the set of stable placements (resp. grasps) of $\mathcal{M}$. Both types of paths lie in lower dimensional sub-spaces of CS_{free}. These sub-spaces are defined as follows:

- The *Placement* space CP is the sub-space of CS_{free} defined as the set of free configurations where the mobile object is placed at a stable position, i.e. a position $p \in P$ at which $\mathcal{M}$ can rest when it is not grasped by the robot. Each transit path lies in CP. However note that every path in CP is not necessarily a transit path. A transit path belongs to the sub-manifold corresponding to a given placement p.

- The *Grasp* space CG is the sub-space of CS_{free} defined as the set of free configurations under all possible grasps $g \in G$ of the object $\mathcal{M}$. A transfer path belongs to the sub-manifold defined by a particular grasp kept constant along the path.

Problem: Given the two sets (discrete or continuous) P and G defining the stable placements and feasible grasps, the manipulation planning problem is to find a manipulation path (i.e. an alternate sequence of transit and transfer paths) connecting two given configurations q_i and q_f of $CG \cup CP$. Manipulation planning then consists in searching for transit and transfer paths in a collection of sub-manifolds corresponding to particular grasps or stable placements of the movable object. Note that the intersection $CG \cap CP$ between the sub-manifold defines the places where transit paths and transfer paths can be connected. The manipulation planning problem appears as a constrained path planning problem inside (and between) the various connected components of $CG \cap CP$.

Discrete Case: In the case of a discrete number of grasps and stable placements, the intersection $CG \cap CP$ consists of a finite set [2] of configurations. In this case, it is possible to build a *Manipulation Graph* (MG) where each node corresponds to a configuration of $CG \cap CP$, and edges are constructed by searching for transfer (or transit) paths between nodes sharing the same grasp (or placement) of the mobile object(s). Most existing manipulation planners (e.g. [2, 12, 8, 9, 4, 1, 15]) consider that such discrete sets P and G are given as input to the planner.

Continuous Case: The structure of the search space is more complex when dealing with continuous sets P and G. For example, one may describe the set of stable placements by constraining the movable object to be placed on top of some horizontal faces of the static obstacles; such placement constraints would define P as a domain in a 3-dimensional manifold of CS_{obj} (two translations in the horizontal place and one rotation around the vertical axis). Also, one may consider G as a set of continuous domains such that the

[1] See [2, 11] for a more formal and general description of the manipulation problem

[2] Note however that the case of a redundant robot considered in [1] also leads to an infinite cardinality of the set $CG \cap CP$.

jaws of a parallel gripper have a sufficient contact with two given faces of $\mathcal{M}$. With such grasp constraints, G also corresponds to a 3-dimensional manifold (two translations parallel to the grasped faces and one rotation around the axis perpendicular to the faces). In such cases, $CG \cap CP$ is defined by continuous configuration domains.

Reduction Property: Extending the notion of manipulation graph to such continuous domains, first requires that the connectivity of $CG \cap CP$ adequally reflects the existence of a manipulation path. This is not *a priori* obvious since paths computed in $CG \cap CP$ may correspond to a continuous change of the grasp along the path, while transfer paths are constrained to stay in the sub-manifold defined by a constant grasp.

The *reduction property* [3] stated in [3] shows that two configurations which are in a same connected component of $CG \cap CP$ can be connected by a manipulation path. This property means that any path in $CG \cap CP$ can be transformed into a finite sequence of transit and transfer paths. It is then sufficient to study the connectivity of the various components of $CG \cap CP$ by transit and transfer paths.

Manipulation Graph: The reduction property allows to extend the Manipulation Graph to the continuous case: $MG's$ nodes correspond to the connected components of $CG \cap CP$ and an edge between two nodes indicates the existence of a transit/transfer path connecting two configurations of the associated components.

3 Planning Approach

While most of the existing planners assume a discrete set of feasible grasps and placements, the continuous case was only addressed in [3] for a translating polygonal robot in a 2D workspace. We now describe a more general approach for solving manipulation problems under continuous manipulation constraints. The main idea is to exploit the reduction property of Section 2 to decompose the construction of the manipulation graph. The two steps consist in:

- computing the connected components of $CG \cap CP$. The next section describes a method that directly captures the topology of $CG \cap CP$ inside a probabilistic roadmap [7, 16]. Each connected component of the roadmap defines a node of MG.

[3]Note that this property only holds for a single movable object under the hypothesis that the robot does not touch the static obstacles

- determining the connectivity of these connected components using transit-paths and transfer-paths. Section 5 explains the method used to determine such connections.

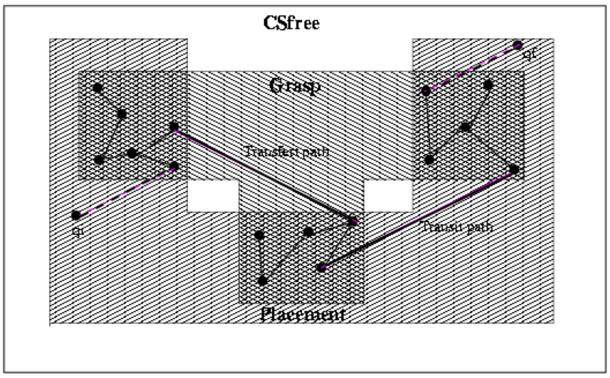

Figure 1: Manipulation Roadmap

Figure 1 illustrates the structure of a manipulation roadmap computed using this approach. The roadmap is structured into a small number of nodes (the connected components of $CG \cap CP$) connected with transit or transfer paths.

Solving a manipulation planning query is performed in three steps: the start and goal configurations are first connected to MG using the algorithm of Section 5. Then, the graph is searched for a path between both configurations. The solution path alternates transfer or transit paths (when traversing edges of MG) with other elementary paths computed in $CG \cap CP$ (inside nodes of MG). The last step is necessary to transform such paths into a finite sequence of transfer/transit paths. This can be done by a simple procedure that iteratively splits the path into pieces whose endpoints can be connected by a collision-free transit motion followed by a collision-free transfer motion.

4 Capturing $CG \cap CP$ Topology via Closed-Chain Systems

As mentioned above, the main critical issue is to capture the topology of $CG \cap CP$ which is a submanifold of the global configuration space with a lower dimension. The main difficulty in using probabilistic roadmap approaches is to face sub-dimensional manifolds. Indeed the probability to choose a configuration at random on a given sub-dimensional manifold is null.

The idea here is to explore $CG \cap CP$ as such. For this, we consider that $CG \cap CP$ is the configuration space of a single system consisting of the robot together with the movable object *placed* at a stable position. Maintaining the stable placement while the object is grasped by the robot induces a closed chain for the global system.

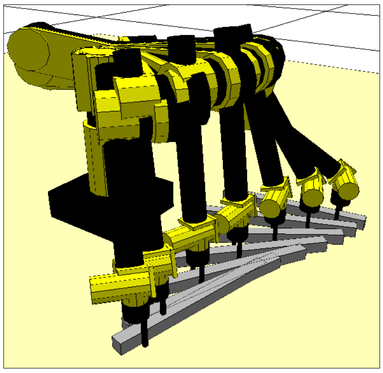

Figure 2: A path in $CG \cap CP$ computed for the virtual closed-chain system (the bar moves on the floor while sliding into the gripper's jaws)

Figure 2 shows the example of a closed chain system formed by a $6dof$ arm manipulating a long bar. The bar moves in contact with the floor while sliding within the gripper. The sliding motion of the gripper results from the additional degrees of freedom q_{grasp} introduced in the system to characterize the infinite set of grasps of the bar. In this example q_{grasp} is chosen to allow a translation of the parallel jaw gripper along the bar. Similarly, the set P of stable placements corresponds to the planar motions parameterized by a vector $q_{place} = (x, y, \theta)$ (two horizontal translations and a vertical rotation), that maintain the contact of the bar with the floor.

The motion shown in the Figure is a feasible motion in $CG \cap CP$. It is not admissible from the manipulation problem point of view. However, it can be transformed into a finite sequence of feasible transit and transfer paths.

We then propose to apply planning techniques for closed chain systems to capture the topology of $CG \cap CP$. Hence, several recent contributions extended the PRM framework [7, 16] to deal with such closed chain mechanisms [14, 6, 5]. We use the algorithm [5] that demonstrates good performance onto complex 3D closed chains involving more than ten degrees of freedom. The loop is broken (as initially proposed in [6]) into two chains. The random node generation combines a sampling technique called *Random Loop Generator* (*RLG*) with forward kinematics for the *active* chain and inverse kinematics for the remaining *passive* part of the loop in order to force the closure. The advantage of the *RLG* algorithm is that it produces random samples for the active chain that have a high probability to be reachable by the passive part. This significantly decreases the cost of computing and connecting closure configurations.

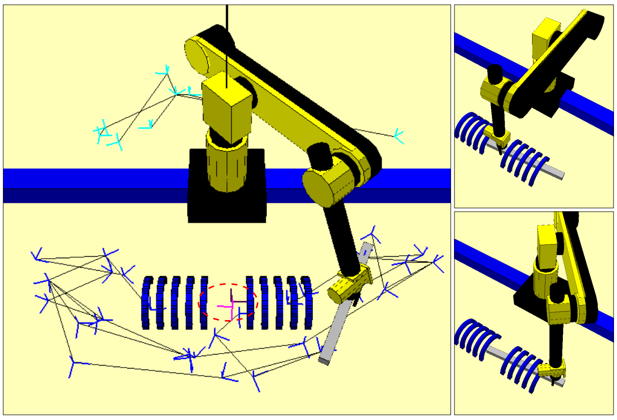

Figure 3: A Visibility Roadmap computed in $CG \cap CP$ (left) and two placements of the system inside two different connected components of $CG \cap CP$ (right)

Figure 3 shows the roadmap computed in $CG \cap CP$ by the algorithm using the *Visibility-PRM* technique [17, 13]. While the collision-free configuration space of the arm alone is connected, $CG \cap CP$ is not. The computed roadmap has four connected components: two main components separated by the long obstacle, and two other small components that correspond to placements of the movable object inside the cage obstacle, while it is grasped by the arm through the open passage in the middle of the cage. These two small components are inside the circle marked onto the left figure. They correspond to a similar position of the bar, rotated or not by 180 degrees. The corresponding placement of the system is illustrated onto the top right image. The bottom right image shows a node of the main component with the bar placed at the same position, but using a different grasp. Connecting this node to the small component is not possible because of the cage obstacle that limits the continuous change of grasp. The re-grasping of the bar requires the computation of a collision-free transit path using the method explained below.

As shown onto this example, the interest of the *Visibility-PRM* technique is to keep the number of nodes in the roadmap to a minimum and therefore to capture the topology of possibly complex spaces into small roadmaps. This, combined with the proposed structuring of $CG \cap CP$, significantly limits the number of (expensive) path-planning queries to be performed for searching connections with collision-free transfer or transit paths.

5 Connections with transit and transfer paths

Consider now two configurations in $CG \cap CP$ that can not be directly connected by a collision-free path in $CG \cap CP$ (i.e. configurations that do not belong to the same connected component of MG). These configurations both correspond to fixed grasps and placements of the movable object, noted $(g_i, p_i)_{i=1,2}$. The problem is now to determine whether both configurations can be connected by collision-free transit/transfer paths searched inside the corresponding leaves of CP and CG. Such a connection is possible if there exists:

- a transfer path from (g_1, p_1) to (g_1, p_2) followed by a transit path from (g_1, p_2) to (g_2, p_2).
- or a transit path from (g_1, p_1) to (g_2, p_1) followed by a transfer path from (g_2, p_1) to (g_2, p_2).

We next explain the method used by the planner to solve these path-planning queries. First, we compute a roadmap for the robot and the static obstacles, without considering the presence of the movable object. Then, before to solve a given transit-path query, the roadmap is updated, each edge being checked to be collision-free with respect to the current position of the object. When an edge does not correspond to a collision-free path, it is labeled as blocked. Then the search for a transit path is performed within the labeled roadmap. If there is no path, this means that there is no path even in the absence of movable objects. The problem has no solution. If there is a path not containing any blocked edge then a solution is found. Now let us consider the intermediate situation. There is a path that necessarily contains a blocked edge. In such cases, the algorithm tries to solve the problem locally. We use a Rapid Random Tree (RRT) algorithm [10] to connect the endpoints of the blocked edges. The main interest of RRT is to perform well locally. Its complexity depends on the length of the solution path. This means that the approach quickly finds easy solutions. It may be viewed as a dynamic updating of the roadmaps.

The same approach applies to find transfer paths within CG: the initial roadmap can be rapidly updated to consider the presence of the object attached to the robot.

6 Experimental Results

The manipulation planner has been implemented within the software platform *Move3D* Figure 4 shows an example of a manipulation problem within the environment already used in Figure 3. The task for the manipulator is to move the bar initially placed inside the cage to a goal configuration. Although the current implementation possibly deals with several classes of continuous grasps and placements, only one continuous set is used for this example to describe the possible grasps (along the bar) and placements (contact with the floor).

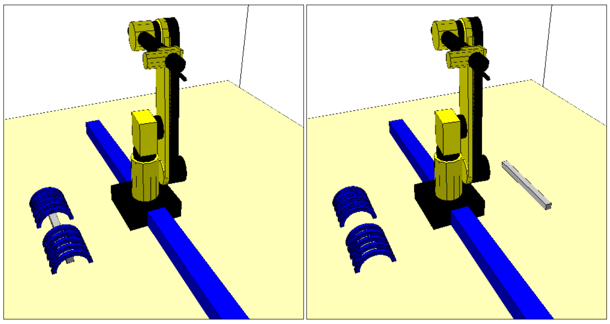

Figure 4: Manipulation planning problem

Figure 5 shows a solution to this manipulation problem. The manipulator first executes a transit path to grasp the movable object (using a grasp position automatically determined by the planner). Then follows a path computed in $CG \cap CP$ that translates the object so that its extremity goes out of the cage obstacle (this path is transformed in a post-processing step into a sequence alternating several transit/transfert paths). Then a transit motion is performed to regrasp the object by the extremity that was made accessible by the previous motion. The bar is then moved outside of the cage and a last transfer motion allows to reach the specified goal position of the bar. Finally, the robot moves back to its home position using a transit path. This difficult manipulation problem was solved in about 2 minutes on a 330MHz Sparc Ultra 10 Station.

7 Conclusion

We have presented a new approach to manipulation planning. The power of the approach lies in the fact that it can deal with continous grasps and placements. It relies on a structuring of the search space allowing to directly capture into a probabilistic roadmap the connectivity of the sub-manifolds that correspond to the places where transit paths and transfer path can be connected. This structuring allows us to design a manipulation planner that automatically generates among the continuous sets, the particular grasps and

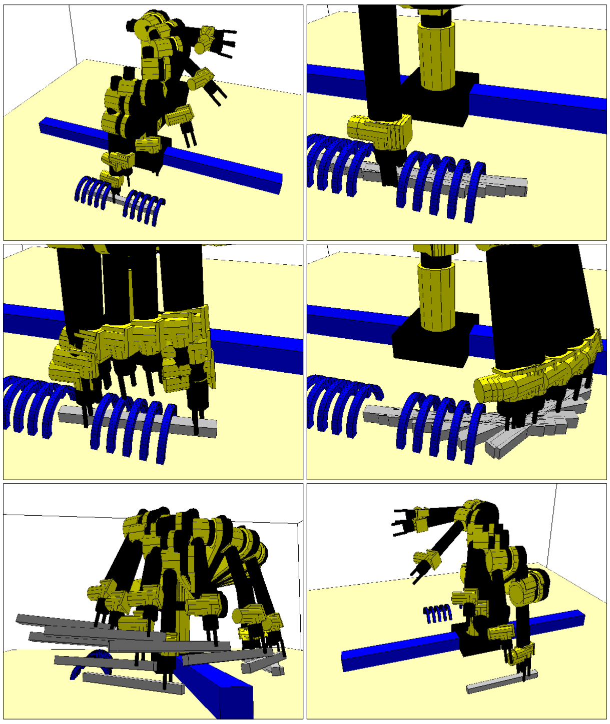

Figure 5: Manipulation solution

placements that make the problem solvable.

Preliminary results obtained with a first implementation of the approach demonstrate its efficacy to solve complex manipulation problems. There remain many possible improvements, in particular in the way to interleave the construction of $CG \cap CP$ with the connection of its connected components via transit or transfer paths. Also, the proposed planner is currently restricted to Pick and Place operations of a single movable object manipulated by a single robot. Considering the case of multiple movable objects and robots first requires studying the conditions under which the reduction property can be extended to such situations.

References

[1] J.M. Ahuactzin, K.K. Gupta and E. Mazer. Manipulation planning for redundant robots: a practical approach. In *Practical Motion Planning in Robotics*, K.K. Gupta and A.P. Del Pobil (Eds), J. Wiley, 1998.

[2] R. Alami, T. Siméon, J.P. Laumond, A geometrical Approach to planning Manipulation Tasks. The case of discrete placements and grasps." In *International Symposium on Robotics Research*, Tokyo, August 1989.

[3] R. Alami, J.P. Laumond and T. Siméon. Two manipulation planning algorithms. In *Algorithmic Foundations of Robotics (WAFR94)*, K. Goldberg et al (Eds), AK Peters, 1995.

[4] J. Barraquand and P. Ferbach. A penalty function method for constrained motion planning. In *IEEE Int. Conference on Robotics and Automation*, 1994.

[5] J. Cortés, T. Siméon and J.P. Laumond. A Random Loop Generator for planning the motions of closed kinematic chains with PRM methods. In *IEEE Int. Conference on Robotics and Automation*, 2002.

[6] L. Han and N. Amato. A kinematics-based probabilistic roadmap method for closed kinematic chains. In *Workshop on Algorithmic Foundations of Robotics*, 2000.

[7] L. Kavraki and J.C. Latombe. Randomized preprocessing of configuration space for fast path planning. In *IEEE Int. Conference on Robotics and Automation*, San Diego (USA), 1994.

[8] Y. Koga and J.C. Latombe. Experiments in dual-arm manipulation planning. In *IEEE Int. Conference on Robotics and Automation*, Nice, 1992.

[9] Y. Koga and J.C. Latombe. On multi-arm manipulation planning. In *IEEE Int. Conference on Robotics and Automation*, 1994.

[10] J. Kuffner and S. Lavalle. RRT-Connect: an efficient approach to single-query path planning. In *IEEE Int. Conference on Robotics and Automation.*, San-Francisco (USA), May 2000.

[11] J.C. Latombe. *Robot Motion Planning*, Kluwer, 1991.

[12] J.P. Laumond and R. Alami. A geometrical Approach to planning Manipulation Tasks in robotics. In *Technical Report n° 89261*, LAAS, Toulouse, 1989.

[13] JP. Laumond, T. Siméon. Notes on visibility roadmaps for motion planning. In *Workshop on Algorithmic Foundations of Robotics.(WAFR'00), Hannover (USA)*, March 2000.

[14] S. LaValle, J.H. Yakey, and L. Kavraki. A probabilistic roadmap approach for systems with closed kinematic chains. In *IEEE Int. Conference on Robotics and Automation*, 1999.

[15] Ch. Nielsen, L. Kavraki. A two-level fuzzy PRM for manipulation planning. In *IEEE Int. Conference on Intelligent Robots and systems*, Japan, 2000.

[16] M. Overmars and P. Švestka. A Probabilistic learning approach to motion planning. In *Algorithmic Foundations of Robotics (WAFR94)*, K. Goldberg et al (Eds), AK Peters, 1995.

[17] T. Siméon, J.P. Laumond and C. Nissoux. Visibility-based probabilistic roadmaps for motion planning. In *Advanced Robotics Journal* 14(6), 2000 (a short version appeared in IEEE IROS, 1999).

[18] T. Siméon, JP. Laumond, C. van Geem and J. Cortés. Computer Aided Motion: Move3D within MOLOG. IEEE Int. Conference on Robotics and Automation, Seoul (Korea), April 2001.