Brief Survey on Model-based Manipulation Planning of Rigid Objects

P. Jiménez

Institut de Robòtica i Informàtica Industrial (CSIC - UPC)
Llorens i Artigas 4-6, E-08028 Barcelona, Spain
e-mail: pjimenez@iri.upc.edu

1 Introduction

Rigid bodies aren't but a particular instance of the broader category of deformable objects, where for practical purposes and under the range of applied forces they can be considered as experimenting no change in shape or dimensions. The problem is viewed as an extension of the basic motion planning problem [12]. The manipulation task consists in changing their placement while avoiding collisions. This approach reflects the same methodological evolution than motion planning: from complete, exact cell-decomposition methods, in practice suited only for very simple instances of the problem, to sampling based algorithms, able to tackle with more degrees of freedom and more realistic settings. This framework should constitute a formal paradigm for further work involving more complex manipulation, like handling deformable objects. As such objects may also be just transferred from one point to the other, the knowledge of how to tackle this problem with the simpler rigid objects constitutes an unavoidable starting point.

2 First attempts: exact cell decompositions of polygonal workspaces

Research in Basic Motion Planning and in Manipulation Planning have had similar beginnings. The method for solving the "Piano Movers' Problem" arose in the Computational Geometry Community as one of the first exact algorithms for the Basic Robot Motion Planning or Path Planning Problem [19, 17, 18, 20, 12]. In the same series of papers, an upper bound on the time complexity in a semi-algebraic free space of given dimension was set for the first time. Similarly, the Manipulation Problem found one of its first formal expressions in terms of a genuine Computational Geometry formulation, in planar environments [25]. In order to prove NP-hardness of planning the motion of an object in the presence of movable obstacles, where their final positions are unspecified, a "mechanical" analogue of the 3-SAT instance is described there: in this environment, the logical operators have their counterparts in gates with movable parts that have to be displaced by the robot in order to complete a given route (or to arrive to a given goal position). In the same reference, a $O(n^2 \log n)$ query time algorithm is sketched (n being the total number of vertices in the environment), after $O(n^3 \log^2 n)$ preprocessing, for a polygonal robot displacing a movable polygon (with a finite set of grasps) to a given goal position amidst fixed obstacles.

In this pioneering work, the workspace is partitioned into a finite set of critical features (regions, edges and vertices) such that the connectivity properties of the robot hold within each of them. This means, for example, that if a given grasp position is possible (i.e., the robot and the movable object do not overlap any fixed obstacle) for a given robot location inside one of these critical regions, it will be also possible for any other location inside the same region (Figure1). Somehow the idea of combining the configuration space of the robot and the movable object is implicit in this formulation, and was made explicit in the research carried on during the following years [13, 14, 3, 2].

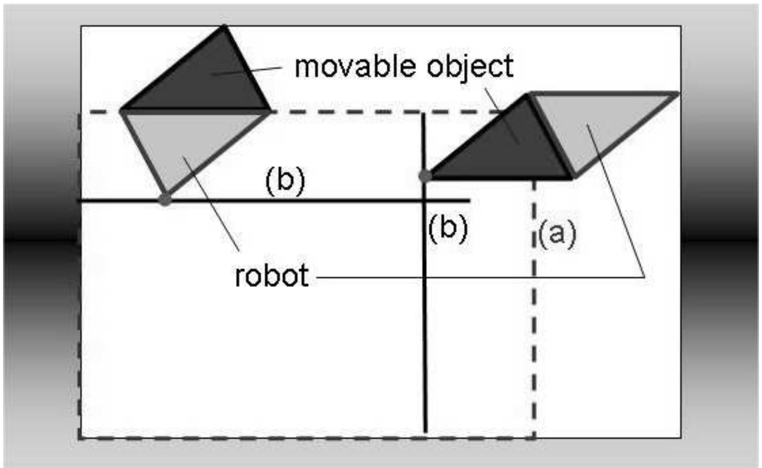

Figure 1: A very simple setting: a bounded rectangular region with no obstacles inside, and both the robot and the movable object (which can be grasped only where one of its sides matches one of the sides of the robot) are triangles which are only allowed to translate. Here just two types of critical curves delimiting critical regions (from the 4 described in [25])have been depicted: (a) corresponds to a maximal line segment of the configuration space of the movable object (dashed lines), whereas (b) are two lines swept out by the reference points of the robot and object for given graspings while touching the boundary of the workspace (this description has been simplified to make it understandable, see [25] for more details).

Together with the planning algorithms, the main contribution of these works lies in the formalization of the different aspects of the Manipulation Problem, beginning with the definition of the space: manipulation takes place in a composite configuration space ($\mathcal{C}$), which is the cartesian product of the robots' (one or several) and the movable objects' individual configuration spaces. In the closure of the free part of this composite space ($cl(\mathcal{C}_{free})$) (legal configurations), two important subsets have to be considered: $\mathcal{C}_{placement}$, where all the movable objects are in valid positions, and $\mathcal{C}_{grasp}$, which is the union of the different $\mathcal{C}_{grasp_j}$ corresponding to the configurations where each individual movable object M_j is grasped by one or more robots. These definitions are left intentionally loose in this survey, more precise specifications are given in some references (for example, $\mathcal{C}_{stable}$ is often defined instead of $\mathcal{C}_{placement}$ as the subset of *stable* placements for the movable objects under external forces like gravity). A *manipulation path* is defined as an alternating sequence of *transit* and *transfer* paths, where the first ones correspond to the motions of the robot without carrying any movable object, while the second ones are those where the movable objects are displaced. Transit paths are contained in $\mathcal{C}_{placement}$, whereas the transfer path of object M_j is included in $\mathcal{C}_{grasp_j}$ (Figures 2 and 3).

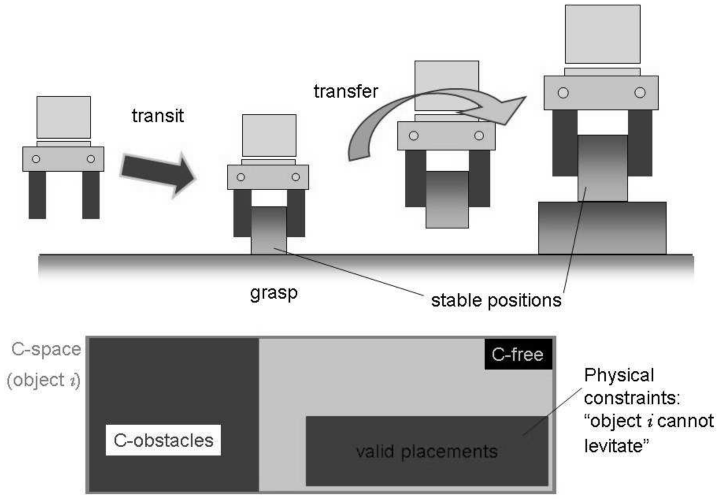

Figure 2: Transit and transfer paths. The set of valid placements of the movable object $\mathcal{C}_{placement}$ is a subset of the collision-free part of its configuration space.

As each transit (except possibly the first and final ones) and transfer paths necessarily begin and end in the intersection subspace $\mathcal{C}_{placement} \cap \mathcal{C}_{grasp}$, the first point addressed by the planning algorithms is to compute all the connected components of this intersection subset. Next, the connectivity of these connected components using transit and transfer paths has to be determined. Finally, means to planning paths inside these components have to be provided. The *Manipulation Graph* (*MG*) whose nodes are the aforementioned connected components (plus the initial and final configuration of the robot) and whose edges correspond to transfer and transit paths verifies the property that a manipulation path exists between two configurations of $\mathcal{C}_{placement}$ if and only if there exists a path between the corresponding nodes in the manipulation graph (as displayed in Figure 3).

A theoretical method for constructing the connected components of $\mathcal{C}_{placement} \cap \mathcal{C}_{grasp}$ and their connectivity through transfer and transit paths is provided in [13] for a single or several movable objects: it is based on the $\mathcal{P}$-invariant Collins decomposition of $\mathcal{C}$, where $\mathcal{P}$ is the collection of polynomials that describe $\mathcal{C}_{free}$ and its subsets as semialgebraic sets. Specific algorithms, on the other hand, were developed for very particular settings:

- **Several movable objects with a finite (discrete) set of grasps and placements** [3]. The computation of $\mathcal{C}_{placement} \cap \mathcal{C}_{grasp}$ is straightforward in this case, by

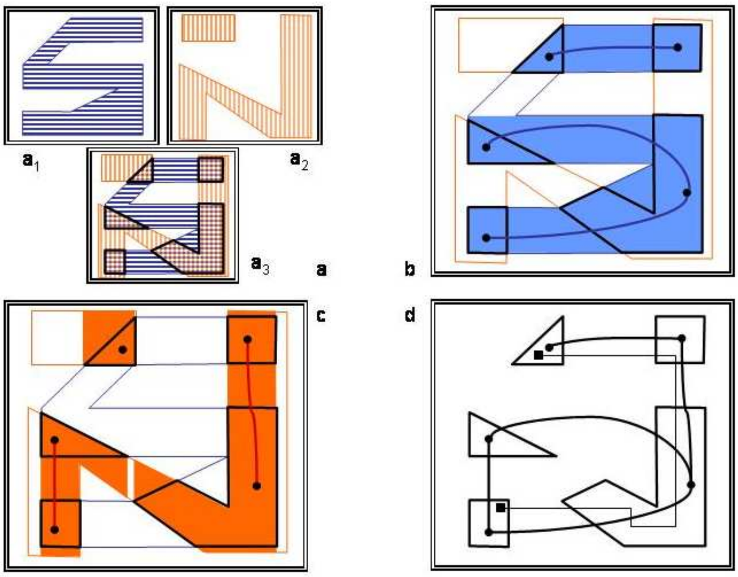

Figure 3: Transit and transfer paths (cont.). A polygonal version of Figures 2 and 3 in [23] is shown. The outer square bounds the composite configuration space of the robot and one or more movable objects. In the case of a single movable object, the S-shaped form represents its placement space $\mathcal{C}_{placement}$, whereas the small rectangle and the N-shaped form correspond to two subsets of its $\mathcal{C}_{grasp}$ space. Alternatively, in a multi-movable objects setting, the S-shape would collect the feasible placements of all objects, while the rectangle could be the grasp space of one object and the N-shape the grasp space of another one. (a) The intersection $\mathcal{C}_{placement} \cap \mathcal{C}_{grasp}$ has five connected components, that can be linked together either (b) via transit paths or (c) via transfer paths (the shadowed regions correspond to reachable configurations starting at configurations inside these connected components via -horizontal- transit and -vertical- transfer paths respectively). Finally, (d) displays the connectivity of the manipulation graph, and a possible manipulation path between two given configurations (small squares).

testing if grasps at the allowed placements are legal configurations. Standard path planning can be applied to compute transit and transfer (treating robot and movable object as a single entity) paths. A heuristic based on the distance of the objects to their final configurations can guide a simultaneous construction and search of the *MG*.

- **Discs (one robot and one movable object) in a polygonal workspace** [14]. The cells represented by the nodes of *MG*, and their connectivity, are computed from an ad hoc partition of $\mathcal{C}_{placement}$ into noncritical regions adapted from [19].

- **One movable (translating, polygonal) object with an infinite set of grasps and placements** [2]. Like in the previous case, $\mathcal{C}_{grasp} \in \mathcal{C}_{placement}$, and its connected components are computed by retraction on the boundary of $\mathcal{C}_{free}$ of a decomposition of this space into noncritical regions inspired by the same method.

All these methods have in common that they are only applicable to low-dimensional problems. Involved settings like multi-arm manipulation or handling of three-dimensional objects with multiple re-grasping operations require more powerful algorithms.

3 Multi-arm and re-grasp manipulation of rigid objects

Manipulation planning reflects the advent of new paradigms in robot motion planning. Randomized Path Planners (RPP), Probabilistic Roadmap Methods (PRM) or Ariadne's Clew Algorithm, all of them originally devised for solving involved instances of the basic motion planning problem, have their counterparts in the manipulation planning field. Although the formalism explained in the previous section is kept alive in the mind of the researchers, an explicit and complete characterization of $\mathcal{C}_{placement} \cap \mathcal{C}_{grasp}$ is now avoided. Instead, the algorithms try to identify landmarks inside the connected components of this intersection submanifold (if not directly provided by the user, as assumed in [15]), and connect them via transit and transfer paths. Alternatively, the sequence of transfer and transit paths may arise from the computation of a collision-free path of the manipulated object [10]. A third approach does not even distinguish between transit and transfer paths at all [5]. Next, we provide details on these approaches, beginning with the latter one.

Instead of decomposing the manipulation problem in a sequence of transit and transfer paths, a unified view is taken in [5] that considers manipulation as a specific instance of the basic path planning problem (with the addition of a formal *grasping constraint*), thus avoiding the decoupling between task and path planning. Holonomic grasps are characterized as those where the docking positions ($\mathcal{C}_{stable} \cap \mathcal{C}_{grasp}$) are singletons (this result is formally proven), and a penalty function method is provided for manipulation planning of an object given a start and a goal docking configuration. This method, called Progressive Variational Dynamic Programming, consists in solving a relaxed version of the grasping constraint, and successively tightening this constraint, using the path found at the previous iteration as input. At each iteration, the new path is computed with the

Variational Dynamic Programming method, devised in [4] for computing collision-free paths.

A two-stage planning approach is adopted in [10] for multi-arm manipulation planning, by differentiating between the (high-level) subtasks that are necessary to achieve a goal configuration and the associated transit and transfer subpaths. This is done by first focusing on determining the sequence of transfer subtasks guaranteed to be completed into transfer paths, and then link them together with the necessary transit tasks, assuming that there exists always a legal transit path for every transit subtask. Another assumption is that the possible grasps of the movable object is a finite and discrete grasp set, specifying both position and type (one or two-handed) of the grasp. A modified RRP is used to plan legal transfer paths: together with a collision-free path for the movable object, collisions for the grasping arms at the different grasp assignments are checked for at the goal configurations, and the illegal ones are pruned away from the list of grasp assignments at the previous configuration. Regrasps arise naturally from reconsidering all grasp assignments when the repeated pruning process makes them to vanish all at a given configuration of the object. An important limitation of the system is not allowing regrasps at positions of the object in contact with the obstacles: static stability has to be granted exclusively by one of the grasping robots while the other one is performing a regrasp. This algorithm is used in the context of manipulation planning applied to computer animation [9], for the generation of realistic human arm motions while manipulating objects.

Ariadne's Clew Algorithm (ACA) is used in [1] for manipulation task planning of redundant robots, which implies continuous intersection $\mathcal{C}_{placement} \cap \mathcal{C}_{grasp}$ subspaces. Like in the preceding and following approaches, planning is performed at two levels: at the top level, the *EXPLORE* part of ACA (called *E-MANIP*) tries to spread landmarks, which are reachable from the start configuration, in $\mathcal{C}_{placement} \cap \mathcal{C}_{grasp}$ by generating single manipulation paths (a transit followed by a transfer path), whereas the *SEARCH* routine tries to reach the goal from these landmarks via a single transfer path (if it does not succeed, *EXPLORE* places new landmarks starting at the previous ones, generating a tree of paths). The same *EXPLORE-SEARCH* strategy is followed at the lower level of the subproblems generated by *E-MANIP*, i.e., planning the particular transit and transfer paths in $\mathcal{C}_{placement}$ and $\mathcal{C}_{grasp}$ respectively, see Figure 4. Transit and transfer paths are implemented as *Manhattan paths*, which allows a straightforward formalization of *EXPLORE* and *SEARCH* as optimization problems to be solved with Genetic Algorithms. A set of discrete pre-defined grasps are assumed in the current version of the algorithm explained in [1].

Probabilistic Roadmap Methods lie in the core of many Manipulation Planning Algorithms [15, 23], due to their adequateness for coping with high-dimensional problems. In motion planning, collision-free configurations (landmarks) are generated randomly and interconnected via a local planner. Likewise, the strategy in manipulation planning is to generate random configurations in $\mathcal{C}_{placement} \cap \mathcal{C}_{grasp}$, preferably representing all connected components of this submanifold, and connect them with transit and transfer paths.

The first problem is circumvented in [15]: the user (or client software) has to provide a set of allowed stable placements and grasps for the manipulated object. A *Fuzzy PRM* technique is used both at task level (i.e., to compute a manipulation graph) and as a point-

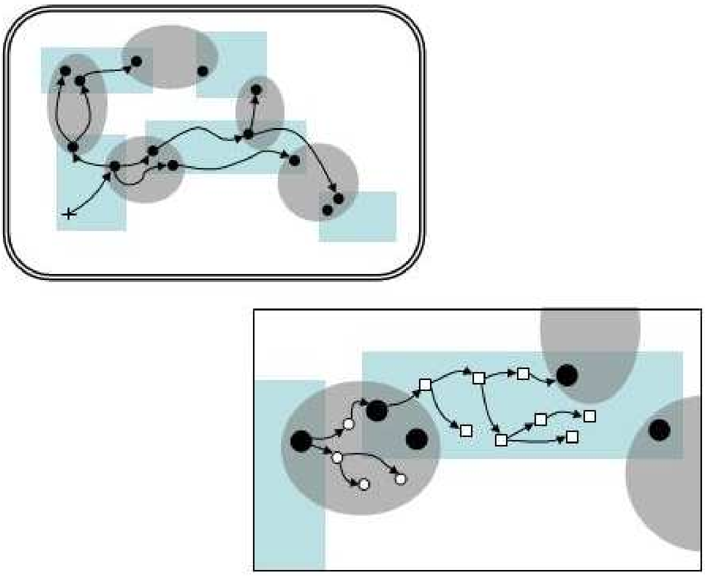

Figure 4: ACA applied to manipulation (from [1]). Up on the left, the top level part of the algorithm, where landmarks are placed in $\mathcal{C}_{placement} \cap \mathcal{C}_{grasp}$ to be reached from the start configuration, traversing via single manipulation paths. These are in turn obtained, as shown down on the right, by applying the same planning principle to $\mathcal{C}_{placement}$ and $\mathcal{C}_{grasp}$ subspaces, and obtaining the transit and transfer paths respectively.

to-point planner for computing the individual transit and transfer paths. This planning algorithm differs from standard PRM in that edges have an associated probability which is an estimation of the chance of this (sub-)path to be found by a local planner during the query phase. In manipulation planning, at high task level, which is in fact a single run of the learning phase of Fuzzy PRM, landmarks with the same grasp relation are connected via transfer edges, and landmarks with the same object location with transit edges. The second level corresponds to the query phase, the local planners are instances of Fuzzy PRM for the different possible grasps, i.e. for the transfer paths, as well as for the transit paths.

The PRM formulation is also present in the work of [7, 8]. Geometric PRM formulations corresponding to different aspects of the manipulation problem are linked together through a set of heuristic rules. This corresponds to an attempt to contemplate the task-level in the planning approach, and is reviewed in the aforementioned companion survey.

Computing the connected components of $\mathcal{C}_{placement} \cap \mathcal{C}_{grasp}$ is identified in [23][1] as determining the parameterization of the set of configurations that satisfy the closure constraints induced by considering the grasping robot and the placed object together as a closed kinematic chain, which can be solved with specific versions of PRM like the *Random Loop Generator* developed by the same authors [6]. A *Visibility-PRM* [24] is used to compute small roadmaps for the different connected components of $\mathcal{C}_{placement} \cap \mathcal{C}_{grasp}$.

[1]This paper gathers together the results presented in [21, 22, 16]

Transit and transfer paths are computed in a two-stage algorithm: first a roadmap for the robot is computed in absence of the movable objects, afterwards it is updated by checking for collisions of the edges with the current position of the movable object (see Figure 5). Use is made of the so called *reduction property* which refers to the possibility of transforming any path in the $\mathcal{C}_{placement} \cap \mathcal{C}_{grasp}$ roadmap into a finite sequence of valid transit and transfer paths. Bidirectional Rapidly-exploring Random Tree planners [11] are used to locally explore the space around the edges blocked by the movable objects, as shown in Figure 5(c).

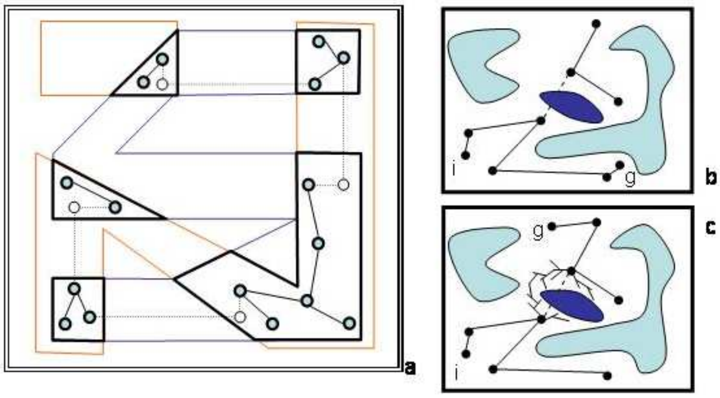

Figure 5: (a) Visibility-PRM is used for computing the small roadmaps inside the connected components of $\mathcal{C}_{placement} \cap \mathcal{C}_{grasp}$. The links of these roadmaps correspond to simultaneous changes of placement and grasp, which is not feasible, but can be reduced to a finite sequence of pure transit and transfer paths. These roadmaps are linked together with simple transit followed by transfer (or vice-versa) paths. (b) and (c) In the configuration space of the robot, a roadmap is computed in absence of the movable objects. Arcs of this pre-computed roadmap may be invalidated due to the presence of the movable objects (darker region). This may have no consequences for a particular query (b), but if it has, RRT-methods are applied locally (c).

Acknowledgments

This work is supported by the EU PACO-PLUS project FP6-2004-IST-4-27657 and by the Catalan Research Comission through the Consolidated Robotics Group.

References

1. Ahuactzin, J., Gupta, K., and Mazer, E. Manipulation planning for redundant robots: A practical approach. *International Journal of Robotics Research 17*, 7 (july 1998), 731–747.

2. Alami, R., Laumond, J. P., and Siméon, T. Two manipulation planning algorithms. In *WAFR: Proceedings of the workshop on Algorithmic foundations of robotics* (Natick, MA, USA, 1995), A. K. Peters, Ltd., pp. 109–125.

3. Alami, R., Siméon, T., and Laumond, J.-P. A geometrical approach to planning manipulation tasks. the case of discrete placements and grasps. In *The fifth international symposium on Robotics research* (Cambridge, MA, USA, 1990), MIT Press, pp. 453–463.

4. Barraquand, J., and Ferbach, P. Path planning through variational dynamic programming. In *Proc. 1994 IEEE Int. Conf. Robotics and Automation* (1994), pp. 1839–1846.

5. Barraquand, J., and Ferbach, P. A penalty function method for constrained motion planning. In *Proc. 1994 IEEE Int. Conf. Robotics and Automation* (1994), pp. 1235–1242.

6. Cortés, J., Siméon, T., and Laumond, J. A random loop generator for planning the motions of closed kinematic chains using prm methods. In *Proc. 2002 IEEE Int. Conf. Robotics and Automation* (2002), vol. 2, pp. 2141–2146.

7. Gravot, F., Alami, R., and Siméon, T. Playing with several roadmaps to solve manipulation problems. In *IEEE/RSJ International Conference on Intelligent Robots and System* (Natick, MA, USA, 2002), vol. 3, pp. 2311–2316.

8. Gravot, F., Alami, R., and Siméon, T. *Robotics Research*, vol. 15/2005 of *Springer Tracts in Advanced Robotics*. Springer Berlin / Heidelberg, 2005, ch. aSyMov:A Planner That Deals with Intricate Symbolic and Geometric Problems, pp. 100–110.

9. Koga, Y., Kondo, K., Kuffner, J., and Latombe, J.-C. Planning motions with intentions. *Computer Graphics 28*, Annual Conference Series (1994), 395–408.

10. Koga, Y., and Latombe, J.-C. On multi-arm manipulation planning. In *Proc. 1994 IEEE Int. Conf. Robotics and Automation* (1994), pp. 945–952.

11. Kuffner, J. J., and LaValle, S. M. Rrt-connect: An efficient approach to single-query path planning. In *Proceedings of the IEEE International Conference on Robotics and Automation, ICRA 00* (2000), pp. 995–1001.

12. Latombe, J.-C. *Robot Motion Planning*, vol. SECS 0124. Kluwer, Dordrecht, The Netherlands, 1991.

13. Laumond, J. P., and Alami, R. A geometrical approach to planning manipulation tasks: The case of a circular robot and a movable circular object amidst polygonal obstacles. Tech. Rep. 88314, LAAS, Toulouse, 1988.

14. Laumond, J. P., and Alami, R. A geometrical approach to planning manipulation tasks in robotics. Tech. Rep. 89261, LAAS, Toulouse, 1989.

15. Nielsen, C. L., and Kavraki, L. E. A two level fuzzy prm for manipulation planning. In *Proceedings of The IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)* (november 2000), pp. 1716–1722.

16. Sahbani, A., Cortés, J., and Siméon, T. A probabilistic algorithm for manipulation planning under continuous grasps and placements. In *Proceedings of the IEEE/IROS International Conference on Intelligent Robots and Systems* (2002), pp. 1560–1565.

17. Schwartz, J. T., and Sharir, M. On the Piano Movers' Problem: I. The case of a two-dimensional rigid polygonal body moving amidst polygonal barriers. *Communications on Pure and Applied Mathematics 36* (1983), 345–398.

18. Schwartz, J. T., and Sharir, M. On the Piano Movers' Problem: II. General techniques for computing topological properties of algebraic manifolds. *Communications on Pure and Applied Mathematics 36* (1983), 345–398.

19. Schwartz, J. T., and Sharir, M. On the Piano Movers' Problem: III. Coordinating the motion of several independent bodies: The special case of circular bodies moving amidst polygonal barriers. *International Journal of Robotics Research 2*, 3 (1983), 97–140.

20. Schwartz, J. T., Sharir, M., and Hopcroft, J. *Planning, Geometry, and Complexity of Robot Motion.* Ablex, Norwood, NJ, 1987.

21. Siméon, T., Cortés, J., Sahbani, A., and Laumond, J. A manipulation planner for pick and place operations under continuous grasps and placements. In *Proceedings IEEE International Conference on Robotics and Automation, ICRA 2002* (2002), vol. 2, pp. 2022–2027.

22. Siméon, T., Cortés, J., Sahbani, A., and Laumond, J. A general manipulation task planner. In *Algorithmic Foundations of Robotics V* (2003), J.-D. Boissonat, K. Goldberg, and S. Hutchinson, Eds., Springer - Verlag, pp. 311–328.

23. Siméon, T., Laumond, J.-P., Cortés, J., and Sahbani, A. Manipulation planning with probabilistic roadmaps. *International Journal of Robotics Research 23*, 7 (July–August 2004), 729–746.

24. Siméon, T., Laumond, J.-P., and Nissoux, C. Visibility-based probabilistic roadmaps for motion planning. *Advanced Robotics 14*, 6 (2000), 477–493.

25. Wilfong, G. Motion planning in the presence of movable obstacles. In *SCG '88: Proceedings of the fourth annual symposium on Computational geometry* (New York, NY, USA, 1988), ACM Press, pp. 279–288.