A Survey of Motion Planning and Related Geometric Algorithms

J.T. Schwartz and M. Sharir
Courant Institute of Mathematical Sciences, New York University, New York, NY 10003, U.S.A.; and School of Mathematical Sciences, Tel-Aviv University, Ramat-Aviv, Tel-Aviv 69978, Israel

1. Introduction

This paper surveys recent developments in motion planning and related geometric algorithms, a theoretical research area that has grown rapidly in response to increasing industrial demand for automatic manufacturing systems which use robotic manipulators and sensory feedback devices, and, more significantly, in anticipation of a future generation of substantially more autonomous and intelligent robots. These future robots are expected to possess advanced capabilities of sensing, planning, and control, enabling them to gather knowledge about their environment, construct a symbolic world model of the environment, and use this model in planning and carrying out tasks set to them in high-level style by an application programmer.

Current research in theoretical robotics therefore aims to identify these basic capabilities that an autonomous intelligent robot system will need to advance understanding of the mathematical and algorithmic principles fundamental to these capabilities.

Among these capabilities, planning involves the use of an environment model to carry out significant parts of a robot's activities automatically. The aim is to allow the robot's user to specify a desired activity in very high-level general terms, and then have the system fill in the missing low-level details. For example, the user might specify the end product of some assembly process, and ask the system to construct a sequence of assembly substeps; or, at a less demanding level, to plan collision-free motions which pick up individual subparts of an object to be assembled, transport them to their assembly position, and insert them into their proper places.

Techniques for the automatic planning of robot motions have advanced

Artificial Intelligence 37 (1988) 157–169
0004-3702/88/$3.50

substantially during the last several years. This area has shown itself to have significant mathematical content; tools drawn from classical geometry, topology, algebraic geometry, algebra, and combinatorics have all been used in it. This work relates closely to work in computational geometry, an area which has also progressed very rapidly during the last few years.

This survey concentrates on exact algorithmic solutions to the motion planning problem, and does not address other heuristic or approximating approaches which have also been recently developed, some of which may have significant practical advantages. These include works by Lozano-Pérez, Brooks, Mason and Taylor (cf. [4–6, 30, 31]) of the MIT school, to which researchers at many other places are beginning to contribute.

2. Statement of the Problem

In its simplest form, the motion planning problem can be defined as follows. Let B be a robot system consisting of a collection of rigid subparts (some of which may be attached to each other at certain joints while others may move independently) having a total of k degrees of freedom, and suppose that B is free to move in a two- or three-dimensional space V amidst a collection of obstacles whose geometry is known to the robot system. The *motion planning problem* for B is: Given an initial position Z_1 and a desired final position Z_2 of B, determine whether there exists a continuous obstacle-avoiding motion of B from Z_1 to Z_2, and if so plan such a motion.

This problem has been studied in many recent papers (cf. [4–6, 14, 15, 17, 19, 21–24, 26–28, 30–32, 37, 39–41, 45, 49–52, 60, 63, 68, 70]). It is equivalent to the problem of calculating the path-connected components of the (k-dimensional) space FP of all *free positions* of B (i.e. the set of positions of B in which B does not contact any obstacle), and is therefore a problem in "computational topology". In general FP is a high-dimensional space with irregular boundaries, and is thus hard to calculate efficiently.

This standard motion planning problem can be extended and generalized in many possible ways. For example, if the geometry of the environment is not fully known to the robot system, one must employ "exploratory" approach in which plan generation is tightly updated to gathering of data on the environment and to dynamic updating of a world model. Another interesting extension of the motion planning problem is to the case in which the environment contains objects moving in some known and predictable manner. Although this problem has been little studied, the few results obtained so far seem to indicate that it is inherently harder than the static problem.

All the problem variants mentioned so far aim to determine whether a collision-free path exists between two specified system positions, and, if so, to produce such a path. A further issue is to produce a path which satisfies some criterion of optimality. For example, if a mobile robot is approximated as a single moving point, one might want to find the shortest Euclidean path between the initial and final system positions. In more complex situations the notion of optimal motion is less clearly defined, and has as yet been little studied.

Studies of the motion planning problem tend to make heavy use of many algorithmic techniques in computational geometry. Various motion-planning-related problems in computational geometry will also be reviewed.

3. Motion Planning in Static and Known Environments

As above, let B be a moving robot system, k be its number of degrees of freedom, V denote the two- or three-dimensional space in which B is free to move, and FP denote the space of free positions of B, as defined above. The space FP is determined by the collection of algebraic inequalities which express the fact that at position Z the system B avoids collision with any of the obstacles present in its workspace. We will denote by n the number of inequalities needed to define FP, and call it the "geometric (or combinatorial) complexity" of the given instance of the motion planning problem. As noted, we make the reasonable assumption that the parameters describing the degrees of freedom of B can be chosen in such a way that each of these inequalities is algebraic. Indeed, the group of motions (involving various combinations of translations and rotations) available to a given robot can ordinarily be given algebraic representation, and the system B and its environment V can typically be modeled as objects bounded by a collection of algebraic surfaces (e.g., polyhedral, quadratic, or spline-based).

3.1. The general motion planning problem

Assuming then that FP is an algebraic or semi-algebraic set in E^k, Schwartz and Sharir [50] show that the motion planning problem can be solved in time polynomial in the number n of algebraic constraints defining FP and in their maximal degree, but double exponential in k. The general procedure described uses a decomposition technique due to Collins [9] and originally applied to Tarski's theory of real closed fields. Though hopelessly inefficient in practical terms, this result nevertheless serves to calibrate the computational complexity of the motion planning problem.

3.2. Lower bounds

The result just cited suggests that motion planning becomes harder rapidly as the number k of degrees of freedom increases; this conjecture has in fact been proved for various model "robot" systems. Specifically, Reif [45] proved that motion planning is PSPACE-hard for a certain 3-D system involving arbitrarily many links and moving through a complex system of narrow tunnels. Since then PSPACE-hardness has been established for simpler moving systems, including 2-D systems of mechanical linkages (Hopcroft, Joseph, and Whitesides [15]), a system of 2-D independent rectangular blocks sliding inside a rectangular box (Hopcroft, Schwartz and Sharir [17]), and a single 2-D arm with many links moving through a 2-D polygonal space (Joseph and Plantinga [21]). Several weaker results establishing NP-hardness for still simpler systems have also been obtained.

3.3. The "projection method"

In spite of these negative worst-case results, algorithms of varying levels of efficiency for planning the motions of various single robot systems have been developed. These involve several general approaches to the design of motion planning algorithms. The first such approach, known as the *projection method*, uses ideas similar to those appearing in the Collins decomposition procedure described above. One fixes some of the problem's degrees of freedom (for the sake of exposition, suppose just one parameter y is fixed, and let $\bar{x}$ be the remaining parameters); then one solves the resulting restricted $(k-1)$-dimensional motion planning problem. This subproblem solution must be such as to yield a discrete combinatorial representation of the restricted free configuration space (essentially, a cross-section of the entire space FP) that changes only at a finite collection of "critical" values of the final parameter y. These critical values of y are then calculated; they partition the entire space FP into connected cells, and by calculating relationships of adjacency between these cells one can describe the connectivity of FP by a discrete *connectivity graph* CG. This graph has the aforesaid cells as vertices, and has edges which represent relationships of cell adjacency in FP. The connected components of FP correspond in a one-to-one manner to the connected components of CG, reducing the problem to a discrete path searching problem in CG.

This relatively straightforward technique was applied in a series of papers by Schwartz and Sharir on the "piano movers" problem, to yield polynomial-time motion planning algorithms for various specific systems, including a rigid polygonal object moving in 2-D polygonal space [49], two or three independent discs moving in coordinated fashion in 2-D polygonal space [51], certain types of multi-arm linkages moving in 2-D polygonal space [60], and a rod moving in 3-D polyhedral space [52]. These initial solutions were coarse and not very efficient; subsequent refinements have improved substantially their efficiency. For example, Leven and Sharir [26] obtained an $O(n^2 \log n)$ algorithm for the case of a line segment (a "rod") moving in 2-D polygonal space (improving the $O(n^5)$ algorithm of [49]); the Leven–Sharir result was later shown to be nearly optimal.

3.4. The "retraction method" and other approaches to the motion planning problem

Several other important algorithmic motion planning techniques were developed subsequent to the series of papers just reported. The so-called *retraction method* proceeds by retracting the configuration space FP onto a lower-dimensional (usually a one-dimensional) subspace N, so that two system positions in FP lie in the same connected component of FP if and only if their retractions to N lie in the same connected component of N. This reduces the dimension of the problem, and if N is one-dimensional the problem becomes one of searching a graph.

O'Dunlaing and Yap [39] introduced this retraction technique in the simple case of a disc moving in 2-D polygonal space. Here the subspace N can be taken to be the Voronoi diagram associated with the set of given polygonal obstacles. Their technique yields an $O(n \log n)$ motion planning algorithm. After this first paper, O'Dunlaing, Sharir and Yap [40, 41] generalized the retraction approach to the case of a rod moving in 2-D polygonal space by defining a variant Voronoi diagram in the 3-D configuration space FP of the rod, and by retracting onto this diagram. This achieves $O(n^2 \log n \log^* n)$ performance in this case (a substantial improvement on the naive projection technique first applied to this case, but nevertheless a result shortly afterward superceded by Leven and Sharir [26]).

A similar retraction approach was used by Leven and Sharir [27] to obtain an $O(n \log n)$ algorithm for planning the purely translational motion of a simple convex object amidst polygonal barriers. This last result uses another generalized variant of Voronoi diagram. (A somewhat simpler $O(n \log^2 n)$ algorithm, based on a general technique introduced by Lozano-Pérez and Wesley [32], was previously obtained by Kedem and Sharir [23] (cf. also [22]); this last result exploits an interesting topological property of intersecting planar Jordan curves.)

Recently Sifrony and Sharir [63] devised another retraction-based algorithm for the motion of a rod in 2-D polygonal space. The retraction used maps the rod's free configuration space FP onto a network containing all edges of the boundary of FP, plus some additional arcs which connect particular vertices of FP. They obtain an $O(n^2 \log n)$ algorithm which has the advantage that it runs much faster than that of Leven and Sharir [26] if the obstacles do not lie close to one another.

Hybrid techniques are also appropriate for certain cases of motion planning. For example, in an analysis of the motion planning problem for a convex polygonal object moving in 2-D polygonal space, Kedem and Sharir [24] obtained an $O(n^2\beta(n)\log n)$ motion planning algorithm (where $\beta(n)$ is a very slowly growing function of n), using a hybrid approach which involves projection of FP onto a 2-D space in which the orientation θ of the object is fixed, followed by retraction of the 2-D space roughly onto its boundary. This result makes use of a combinatorial result of Leven and Sharir [28].

4. Variants of the Motion Planning Problem

4.1. Optimal motion planning

The only optimal motion planning which has been studied extensively thus far is that in which the moving system is represented as a single point, in which case one aims to calculate the shortest Euclidean path connecting initial and final system positions, given that specified obstacles must be avoided. Most existing work on this problem assumes that the obstacles are either polygonal (in 2-space) or polyhedral (in 3-space).

The 2-D case is considerably simpler than the 3-D case. When the free space V in 2-D is bounded by n straight edges, it is easy to calculate the desired shortest path in time $O(n^2 \log n)$. This is done by constructing a *visibility graph* VG whose edges connect all pairs of boundary corners of V which are visible from each other through V, and then by searching for a shortest path through VG (see [60] for a sketch of this idea). This procedure was improved to $O(n^2)$ by Asano et al. [1], by Welzl [69], and by Reif and Storer [47], using a cleverer method for constructing VG. Their quadratic-time bound has been improved in certain special cases. However, it is not known whether shortest paths for a general polygonal space V can be calculated in subquadratic time. Among the special cases allowing more efficient treatment the most important is that of calculating shortest paths inside a simple polygon P. Lee and Preparata [25] gave a linear-time algorithm for this case, assuming that a triangulation of P is given in advance. (As a matter of fact, a recent algorithm of Tarjan and van Wyk shows that triangulation in $O(n \log\log n)$ time is possible.) The Preparata–Lee result was recently extended by Guibas et al. [12], who gave a linear-time algorithm which calculates all shortest paths from a fixed source point to all vertices of P.

Other results on 2-D shortest paths include an $O(n \log n)$ algorithm for *rectilinear* shortest paths avoiding n rectilinear disjoint rectangles [48]; an $O(n^2 \log n)$ algorithm for Euclidean shortest motion of a circular disc in 2-D polygonal space [7]; algorithms for cases in which the obstacles consist of a small number of disjoint convex regions [47]; algorithms for the "weighted region" case (in which the plane is partitioned into polygonal regions and the path has a different multiplicative cost weight when it passes through each of these regions) [35]; and some other special cases.

The 3-D polyhedral case is substantially more difficult. To date, only exponential-time algorithms for the general polyhedral case have been developed [47, 62], and it is not yet known whether the problem is really intractable. However, more efficient algorithms exist in certain special cases. The simplest case is that in which we must calculate the shortest path between two points lying on the surface of a convex polyhedron; it was shown that this can be done in $O(n^2 \log n)$ time (see [30, 62]). Generalizations of this result include algorithms for shortest paths along a (not necessarily convex) polyhedral surface [36], algorithms for shortest paths in 3-space which must avoid a fixed number of convex polyhedra [3, 57], and an approximating pseudo-polynomial scheme for the general case [43].

4.2. Adaptive and exploratory motion planning

If the environment is not known to the robot system a priori, but the system is equipped with sensory devices, motion planning assumes a more "exploratory" character. If only tactile (or proximity) sensing is available, then a plausible strategy might be to move along a straight line (in physical or configuration space) directly to the target position, and when an obstacle is reached, to follow its boundary until the original straight line of motion is reached again [33]. If vision is also available, then other possibilities need to be considered, e.g. the system could obtain partial information about its environment by viewing it from the present position, and then "explore" it to gain progressively more information until the desired motion can be fully planned. However, problems of this sort have hardly begun to be investigated.

Even when the environment is fully known to the system, other interesting issues arise if the environment is changing. For example, when some of the objects in the robot's environment are picked up by the robot and moved to a different position, one wants fast techniques for incremental updating of the environment model and the data structures used for motion planning. Moreover, whenever the robot grasps an object to move it, robot plus grasped object become a new moving system and may require a different motion planning algorithm, but one whose relationship to motions of the robot alone needs to be investigated. Adaptive motion planning problems of this kind have hardly been studied as yet.

4.3. Motion planning in the presence of moving obstacles

Interesting generalizations of the motion planning problem arise when some of the obstacles in the robot's environment are assumed to be moving along known trajectories. In this case the robot's goal will be to "dodge" the moving obstacles while moving to its target position. In this "dynamic" motion planning problem, it is reasonable to assume some limit on the robot's velocity and/or acceleration. Two initial studies of this problem by Reif and Sharir [46] and by Sutner and Maass [65] indicate that the problem of avoiding moving obstacles is substantially harder than the corresponding static problem. By using time-related configuration changes to encode Turing machine states, they show that the problem is PSPACE-hard even for systems with a small and fixed number of degrees of freedom. However, polynomial-time algorithms are available in a few particularly simply special cases.

5. Results in Computational Geometry Relevant to Motion Planning

The various studies of motion planning described above make extensive use of efficient algorithms for the geometric subproblems which they involve, for which reason motion planning has encouraged research in computational geometry. Problems in computational geometry whose solutions apply to robotic motion planning are described in the following subsections.

5.1. Intersection detection

The problem here is to detect intersections and to compute shortest distances, e.g. between moving subparts of a robot system and stationary or moving obstacles. Simplifications which have been studied include that in which all objects involved are circular discs (in the 2-D case) or spheres (in the 3-D case). In a study of the 2-D case of this problem, Sharir [55] developed a generalization of Voronoi diagrams for a set of (possibly intersecting) circles, and used this diagram to detect intersections and computing shortest distances between discs in time $O(n \log^2 n)$ (an alternative approach to this appears in [20]). Hopcroft, Schwartz and Sharir [16] present an algorithm for detecting intersections among n 3-D spheres which also runs in time $O(n \log^2 n)$. However, this algorithm does not adapt in any obvious way to allow proximity calculation or other significant problem variants.

Other intersection detection algorithms appearing in the computational geometry literature involve rectilinear objects and use multi-dimensional searching techniques for achieving high efficiency (see [34] for a survey of these techniques).

5.2. Generalized Voronoi diagrams

The notion of Voronoi diagram has proven to be a useful tool in the solution of many motion planning problems. We have also mentioned the use of various variants of Voronoi diagram in the retraction-based algorithms for planning the motion of a disc [39], or of a rod [40, 41], or the translational motion of a convex object [27], and in the intersection detection algorithm for discs mentioned above [55]. The papers just cited, and some related works [29, 71] describe the analysis of these diagrams and the design of efficient algorithms for their calculations.

5.3. Davenport–Schinzel sequences

Davenport–Schinzel sequences are combinatorial sequences of n symbols which do not contain certain forbidden subsequences of alternating symbols. Sequences of this sort appear in studies of efficient techniques for calculating the lower envelope of a set of n continuous functions, if it is assumed that the graphs of any two functions in the set can intersect in some fixed number of points at most. These sequences, whose study was initiated in [10, 11], have proved to be powerful tools for analysis (and design) of a variety of geometric algorithms, many of which are useful for motion planning.

More specifically, an (n, s) Davenport–Schinzel sequence is defined to be a sequence U composed of n symbols, such that (i) nor two adjacent elements of U are equal, and (ii) there do not exist $s+2$ indices $i_1 < i_2 < \cdots < i_{s+2}$ such that $u_{i_1} = u_{i_3} = u_{i_5} = \cdots = a$, $u_{i_2} = u_{i_4} = u_{i_6} = \cdots = b$, with $a \neq b$. Let $\lambda_s(n)$ denote the maximal length of an (n, s) Davenport–Schinzel sequence. Early study by Szemeredi [66] of the maximum possible length of such sequences shows that $\lambda_s(n) \leq C_s n \log^* n$, where C_s is a constant depending on s. Improving on this result, Hart and Sharir [13] proved that $\lambda_3(n) = \Theta(n\alpha(n))$ where $\alpha(n)$ is the very slowly growing inverse of the Ackermann function. In [56, 59] Sharir established the bounds

$$\lambda_s(n) = O(n\alpha(n)^{O(\alpha(n)^{s-3})})$$

and

$$\lambda_s(n) = \Omega(n\alpha^{\lfloor (s-1)/2 \rfloor}(n))$$

for $s > 3$. These results show that, in practical terms, $\lambda_s(n)$ is an almost linear function of n (for any fixed s).

Recently, numerous applications of these sequences to motion planning have been found. These include:

(i) an upper bound of $O(kn\lambda_6(kn))$ on the number of simultaneous triple contacts of a convex k-gon translating and rotating in 2-D polygonal space bounded by n edges [28]; an extension of this result was used to produce an $O(kn\lambda_6(kn)\log kn)$ motion planning algorithm for a moving convex k-gon in such a 2-D space [24];

(ii) an $O(mn\alpha(mn)\log m \log n)$ algorithm for separating two interlocking simple polygons by a sequence of translations [44], where it assumed that the polygons have m and n sides respectively;

(iii) an $O(n^2\lambda_{10}(n)\log n)$ algorithm for finding the shortest Euclidean path between two points in 3-space avoiding the interior of two disjoint convex polyhedra having n faces altogether [3].

Other applications are found in [2, 8, 13, 41, 61].

5.4. Topological results related to motion planning

Motion planning is equivalent to the topological problem of calculating the connected components of semi-algebraic varieties in E^k (namely free configuration spaces of robot systems). It is therefore of interest to study topological properties of such varieties which have close relationships to motion planning. A result of this kind by Hopcroft and Wilfong [19], which applies techniques drawn from homology theory, shows that when an object A moves in the presence of just a single (connected) planar obstacle B, then if collision-free motion of A is possible between two positions in which it makes contact with B, then A can move between these two positions so that it always stays in contact with B.

REFERENCES

1. Asano, T., Asano, T., Guibas, L., Hershberger, J. and Imai, H., Visibility polygon search and Euclidean shortest paths, in: *Proceedings 26th Symposium on Foundations of Computer Science* (1985) 155–164.
2. Atallah, M., Dynamic computational geometry, in: *Proceedings 24th Symposium on Foundations of Computer Science* (1983) 92–99.
3. Baltsan, A. and Sharir, M., On shortest paths between two convex polyhedra, *J. ACM* **35** (1988) 267–287.
4. Brooks, R.A., Solving the find-path problem by good representation of free space, *IEEE Trans. Syst. Man Cybern.* **13** (1983) 190–197.
5. Brooks, R.A., Planning collision-free motions for pick-and-place operations, *Int. J. Rob. Res.* **2** (4) (1983) 19–40.
6. Brooks, R.A. and Lozano-Pérez, T., A subdivision algorithm in configuration space for findpath with rotation, AI Memo 684, MIT, Cambridge, MA (1982).
7. Chew, L.P., Planning the shortest path for a disc in $O(n^2 \log n)$ time, in: *Proceedings ACM Symposium on Computational Geometry* (1985) 214–223.
8. Cole, R. and Sharir, M., Visibility problems for polyhedral terrains, *J. Symbolic Comput.* (to appear).
9. Collins, G.E., Quantifier elimination for real closed fields by cylindrical algebraic decomposition, in: *Proceedings Second GI Conference on Automata Theory and Formal Languages*, Lecture Notes in Computer Science **33** (Springer, New York, 1975) 134–183.
10. Davenport, H., A combinatorial problem connected with differential equations, II, *Acta Arithmetica* **17** (1971) 363–372.
11. Davenport, H. and Schinzel, A., A combinatorial problem connected with differential equations, *Am. J. Math.* **87** (1965) 684–694.
12. Guibas, L., Hershberger, J., Leven, D., Sharir, M. and Tarjan, R.E., Linear time algorithms for shortest path and visibility problems inside triangulated simple polygons, *Algorithmica* **2** (1987) 209–233.
13. Hart, S. and Sharir, M., Nonlinearity of Davenport–Schinzel sequences and of generalized path compression schemes, *Combinatorica* **6** (1986) 151–177.
14. Hopcroft, J.E., Joseph, D.A. and Whitesides, S.H., On the movement of robot arms in 2-dimensional bounded regions, *SIAM J. Comput.* **14** (1985) 315–333.
15. Hopcroft, J.E., Joseph, D.A. and Whitesides, S.H., Movement problems for 2-dimensional linkages, *SIAM J. Comput.* **13** (1984) 610–629.
16. Hopcroft, J.E., Schwartz, J.T. and Sharir, M., On the complexity of motion planning for multiple independent objects; PSPACE hardness of the 'warehouseman's problem', *Int. J. Rob. Res.* **3** (4) (1984) 76–88.
17. Hopcroft, J.E., Schwartz, J.T. and Sharir, M., Efficient detection of intersections among spheres, *Rob. Res.* **2** (4) (1983) 77–80.
18. Hopcroft, J.E., Schwartz, J.T. and Sharir, M. (Eds.), *Planning, Geometry and Complexity of Robot Motion* (Ablex, Norwood, NJ, 1987).
19. Hopcroft, J.E. and Wilfong, G., On the motion of objects in contact, Tech. Rept. 84-602, Computer Science Department, Cornell University, Ithaca, NY (1984).

20. Imai, H., Iri, M. and Murota, K., Voronoi diagram in the Laguerre geometry and its applications, Tech. Rept. RMI 83-02, Department of Mathematical Engineering and Instrumentation Physics, University of Tokyo (1983).
21. Joseph, D.A. and Plantinga, W.H., On the complexity of reachability and motion planning questions, in: *Proceedings ACM Symposium on Computational Geometry* (1985) 62–66.
22. Kedem, K., Livne, R., Pach, J. and Sharir, M., On the union of Jordan regions and collision-free translational motion amidst polygonal obstacles, *Discrete Comput. Geometry* **1** (1986) 59–71.
23. Kedem, K. and Sharir, M., An efficient algorithm for planning collision-free translational motion of a convex polygonal object in 2-dimensional space amidst polygonal obstacles, in: *Proceedings ACM Symposium on Computational Geometry* (1985) 75–80.
24. Kedem, K. and Sharir, M., An efficient motion-planning algorithm for a convex polygonal object in two-dimensional polygonal space, *Discrete Comput. Geometry* (to appear).
25. Lee, D.T. and Preparata, F.P., Euclidean shortest paths in the presence of rectilinear barriers, *Networks* **14** (1984) 393–410.
26. Leven, D. and Sharir, M., An efficient and simple motion planning algorithm for a ladder moving in two-dimensional space amidst polygonal barriers, *J. Algorithms* **8** (1987) 192–215.
27. Leven, D. and Sharir, M., Planning a purely translational motion for a convex object in two-dimensional space using generalized Voronoi diagrams, *Discrete Comput. Geometry* **2** (1987) 9–31.
28. Leven, D. and Sharir, M., On the number of critical free contacts of a convex polygonal object moving in 2-D polygonal space, *Discrete Comput. Geometry* **2** (1987) 255–270.
29. Leven, D. and Sharir, M., Intersection and proximity problems and Voronoi diagrams, in: Schwartz, J. and Yap, C. (Eds.), *Advances in Robotics* **I** (Erlbaum, Hillsdale, NJ, 1987) 187–228.
30. Lozano-Pérez, T., Spatial planning: A configuration space approach, *IEEE Trans. Comput.* **32** (2) (1983) 108–119.
31. Lozano-Pérez, T., A simple motion planning algorithm for general robot manipulators, Tech. Rept., AI Lab, MIT, Cambridge, MA (1986).
32. Lozano-Pérez, T. and Wesley, M., An algorithm for planning collision-free paths among polyhedral obstacles, *Comm. ACM* **22** (1979) 560–570.
33. Lumelsky, V.J. and Stepanov, A., Path planning strategies for a traveling automaton in an environment with uncertainty, Tech. Rept. 8504, Center for Systems Science, Yale University, New Haven, CT (1985).
34. Mehlhorn, K., *Data Structures and Algorithms, III: Multidimensional Searching and Computational Geometry* (Springer, New York, 1984).
35. Mitchell, J., Mount, D. and Papadimitriou, C., The discrete geodesic problem, Tech. Rept., Department of Operations Research, Stanford University, Stanford, CA (1985).
36. Mitchell, J. and Papadimitriou, C., The weighted region problem, Tech. Rept., Department of Operations Research, Stanford University, Stanford, CA (1985).
37. Moravec, H.P., Robot rover visual navigation, Ph.D. Dissertation, Stanford University, Stanford, CA (1981).
38. Mount, D.M., On finding shortest paths on convex polyhedra, Tech. Rept., Computer Science Department, University of Maryland, College Park, MD (1984).
39. O'Dunlaing, C. and Yap, C., A 'retraction' method for planning the motion of a disc, *J. Algorithms* **6** (1985) 104–111.
40. O'Dunlaing, C., Sharir, M. and Yap, C., Generalized Voronoi diagrams for a ladder, I: Topological analysis, *Comm. Pure Appl. Math.* **39** (1986) 423–483.
41. O'Dunlaing, C., Sharir, M. and Yap, C., Generalized Voronoi diagrams for a ladder, II: Efficient construction of the diagram, *Algorithmica* **2** (1987) 27–59.
42. O'Rourke, J., Lower bounds on moving a ladder, Tech. Rept. 85/20, Department of EECS, Johns Hopkins University, Baltimore, MD (1985).

43. Papadimitriou, C., An algorithm for shortest path motion in three dimensions, *Inf. Proc. Lett.* **20** (1985) 259–263.
44. Pollack, R., Sharir, M. and Sifrony, S., Separating two simple polygons by a sequence of translations, *Discrete Comput. Geometry* **3** (1988) 123–136.
45. Reif, J.H., Complexity of the mover's problem and generalizations, in: *Proceedings 20th IEEE Symposium on Foundations of Computer Science* (1979) 421–427.
46. Reif, J.H. and Sharir, M., Motion planning in the presence of moving obstacles, Tech. Rept. 39/85, The Eskenasy Institute of Computer Science, Tel-Aviv University (1985).
47. Reif, J.H. and Storer, J.A., Shortest paths in Eucidean space with polyhedral obstacles, Tech. Rept. CS-85-121, Computer Science Department, Brandeis University, Waltham, MA (1985).
48. de Rezende, P.J., Lee, D.T. and Wu, Y.F., Rectilinear shortest paths with retangular barriers, in: *Proceedings ACM Symposium on Computational Geomedtry* (1985) 204–213.
49. 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, *Comm. Pure Appl. Math.* **36** (1983) 345–398.
50. Schwartz, J.T. and Sharir, M., On the piano movers' problem, II: General techniques for computing topological properties of real algebraic manifolds, *Adv. Appl. Math.* **4** (1983) 298–351.
51. 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, *Rob. Res.* **2** (3) (1983) 46–75.
52. Schwartz, J.T. and Sharir, M., On the piano movers' problem, V: The case of a rod moving in three-dimensional space amidst polyhedral obstacles, *Comm. Pure Appl. Math.* **37** (1984) 815–848.
53. Schwartz, J.T. and Sharir, M., Efficient motion planning algorithms in environments of bounded local complexity, Tech. Rept. 164, Computer Science Department, Courant Institute, New York (1985).
54. Schwartz, J.T. and Sharir, M., Mathematical problems and training in robotics, *Notices Am. Math. Soc.* **30** (1983) 478–481.
55. Sharir, M., Intersection and closest-pair problems for a set of planar discs, *SIAM J. Comput.* **14** (1985) 448–468.
56. Sharir, M., Almost linear upper bounds on the length of general Davenport–Schinzel sequences, *Combinatorica* **7** (1987) 131–143.
57. Sharir, M., On shortest paths amidst convex polyhedra, *SIAM J. Comput.* **16** (1987) 561–572.
58. Sharir, M., On the two-dimensional Davenport Schinzel problem, Tech. Rept. 193, Computer Science Department, Courant Institute, New York (1985).
59. Sharir, M., Improved lower bounds on the length of Davenport Schinzel sequences, *Combinatorica* **8** (1988) 117–124.
60. Sharir, M. and Ariel-Sheffi, E., On the piano movers' problem, IV: Various decomposable two-dimensional motion planning problems, *Comm. Pure Appl. Math.* **37** (1984) 479–493.
61. Sharir, M. and Livne, R., On minima of functions, intersection patterns of curves, and Davenport–Schinzel sequences, in: *Proceedings 26th Symposium Foundations of Computer Science* (1985) 312–320.
62. Sharir, M. and Schorr, A., On shortest paths in polyhedral spaces, *SIAM J. Comput.* **15** (1986) 193–215.
63. Sifrony, S. and Sharir, M., An efficient motion planning algorithm for a rod moving in two-dimensional polygonal space, *Algorithmica* **2** (1987) 367–402.
64. Spirakis, P. and Yap, C.K., Strong NP-hardness of moving many discs, *Inf. Proc. Lett.* **19** (1984) 55–59.
65. Sutner, K. and Maass, W., Motion planning among time dependent obstacles, Manuscript (1985).

66. Szemeredi, E., On a problem by Davenport and Schinzel, *Acta Arithmetica* **25** (1974) 213–224.
67. Tarjan, R.E. and van Wyk, C., An $O(n \log \log n)$ time algorithm for triangulating simple polygons, *SIAM J. Comput.* **17** (1988).
68. Udupa, S., Collision detection and avoidance in computer controlled manipulators, Ph.D. Dissertation, California Institute of Technology, Pasadena, CA (1977).
69. Welzl, E., Constructing the visibility graph for n line segments in $O(n^2)$ time, *Inf. Proc. Lett.* **20** (1985) 167–172.
70. Yap, C.K., Coordinating the motion of several discs, Tech. Rept. 105, Computer Science Department, Courant Institute, New York (1984).
71. Yap, C.K., An $O(n \log n)$ algorithm for the Voronoi diagram of a set of simple curve segments, Tech. Rept. 161, Computer Science Department, Courant Institute, New York (1985).

Received September 1986