Complexity of the Mover's Problem and Generalizations

Extended Abstract

John H. Reif*
Computer Science Department
University of Rochester
Rochester, New York 14627

0. Summary

This paper concerns the problem of moving a polyhedron through Euclidean space while avoiding polyhedral obstacles.

1. Introduction

The classical mover's problem in d-space is:

Input: (R,S,p_I,p_F) where R is a set of polyhedral obstacles fixed in Euclidean d-space, and S (say, a sofa) is a rigid polyhedron with distinguished positions p_I and p_F.

Property: Can S be moved (by a sequence of translations and rotations in d-space) from position p_I to p_F without contacting any element of R?

We consider the discretized version of this problem; the input R and S are given as systems of linear inequalities within a fixed accuracy τ, $0 < \tau < 1$.

A correct algorithm for the discretized mover's problem is allowed to yield an inconclusive answer (i.e., "maybe") only in the case that:

1) S, contracted by a factor of $(1-\tau)$, can be moved from p_I to p_F, but

2) the move is impossible if S is expanded by a factor of $(1+\tau)$.

We present (Section 4) a polynomial-time algorithm for the classical mover's problem in 2 and 3 dimensions. In spite of considerable previous work on this problem[1,2,3,4] by workers in the robotics field, no algorithm guaranteed to run in polynomial time and with fixed accuracy has previously appeared.

The mover's problem may be generalized to allow S (the object to be moved) to consist of multiple polyhedra freely linked together at various distinguished vertices. (A typical example is a robot arm with multiple joints.) Again, the input is specified by systems of linear inequalities with accuracy τ.

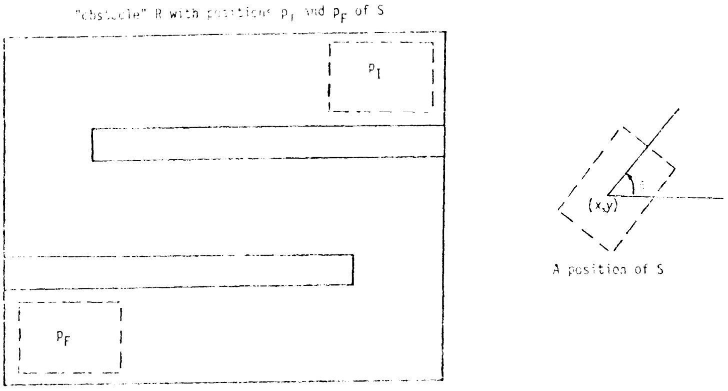

Figure 1: A 2-D Mover's Problem: Can rectangle S be moved from p_I to p_F without contacting an obstacle in R?

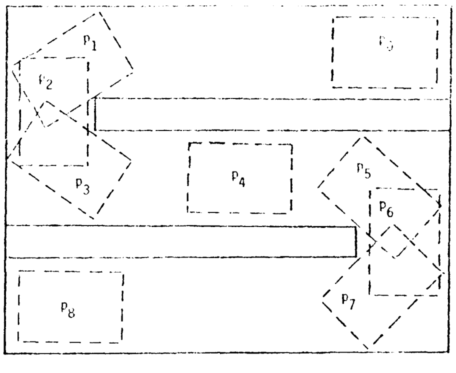

Figure 2: A solution to the 2-D Mover's Problem of Figure 1. S may be moved through positions $p_I = p_0, p_1, \ldots, p_8 = p_F$.

*The author has recently moved to Aiken Computation Laboratory, Harvard University, Cambridge, Massachusetts.

CH1471-2/79/0000-0421$00.75 © 1979 IEEE

Generalized mover's problems have been considered[5,6] in addition to the previously sited references.

We present (Section 3) a polynomial-space algorithm for the generalized mover's problem. Furthermore, we show (Section 2) the generalized mover's problem in 3-space is P-space hard, by a direct log-space reduction from the acceptance problem for polynomial space bounded Turing machines.

The generalized mover's problem is thus the first known p-space complete computational geometry problem (and is one of very few known p-space complete combinatorial problems which is not a game with a polynomial time limit.)

2. The Generalized Mover's Problem is P-Space Hard

Let M be a deterministic Turing Machine with polynomial space bound $s(n) \geq n$. We assume without loss of generality, M has

1) one tape with tape alphabet $\{0,1\}$, and

2) a planar finite state diagram.

Let Q be the set of states of M, with distinguished initial state $q_I \in Q$. Let $\omega \in \{0,1\}^n$ be an input string. We construct a generalized mover's problem (R,S,p_I,p_F) in 3-space which has a solution if and only if M accepts ω. Assuming a fixed binary encoding of the generalized mover's problem into $\{0,1\}^*$, our construction requires $O(\log n)$ work space on a deterministic Turing machine.

In our constructed generalized mover's problem, the obstacles R define a flat slab (of thickness $2s(n)+1$). Within the slab is a network of open channels. These channels are connected as in the finite-state control of M:

1) there are $|Q|$ junctions between the channels and each junction is named for a distinct state $q \in Q$, and

2) the channels are in one-to-one correspondence with the state transitions of M.

Figure 3 illustrates the object S to be moved; it consists of a vertical bar of length s(n) units, with s(n) "arms" freely linked to the bar at unit intervals. The cross-section of a typical channel is illustrated in Figure 4, and consists of $2s(n)-1$ "arm slots" at levels $\{1,2,\ldots,2s(n)-1\}$. S is positioned within a channel as in Figure 5, with each arm in either the left or right portion of an arm slot. (The "elbow" at the end of each arm constrains the arm to remain either to the left or to the right.) The key idea is that the position of S encodes the tape contents of M:

1) the position (left,right) of the ith arm describes the symbol (0 or 1, respectively) on the ith tape cell, and

2) the vertical position of S describes the position of the read/write head of M (in particular, the arm of S currently in the arm slot of level s(n) corresponds to the currently scanned tape cell).

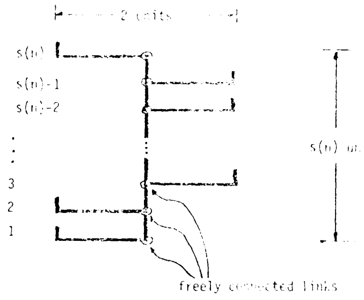

Figure 3: The object S to be moved, encoding tape contents 0,0,1,...,1,1,0.

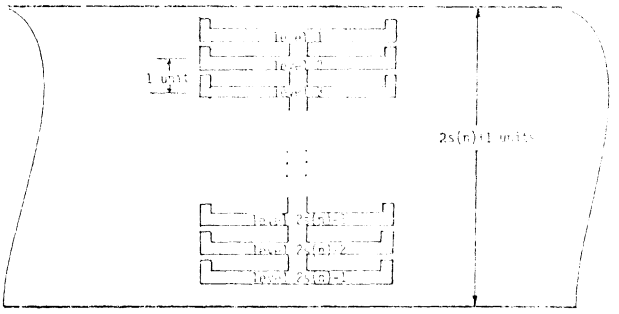

Figure 4: A typical cross-section of a channel.

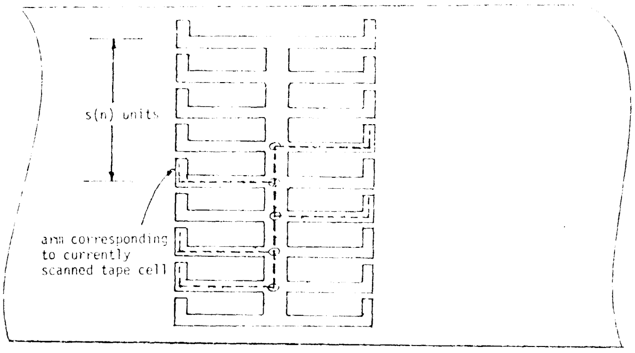

Figure 5: S positioned within a cross-section of a channel, with s(n) = 5. S encodes tape contents 0,0,1,0,1 and tape cell 4 is currently under the tape head.

We now provide the further details of the construction of the channels which simulate the state transitions of M.

If the tape is moved to the right in a state transition, then in the corresponding channel the arm slots slant upward one unit so as to force S to be moved vertically up one unit. A symmetric construction is used in the case a tape is moved to the left (see Figure 6).

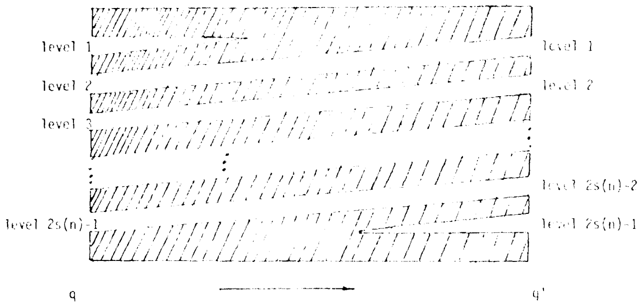

Figure 6: A lengthwise cross-section of a channel which forces S up one unit in a state transition from q to q'.

To test if the currently scanned tape cell is 0 or 1, we construct a junction which forces S to branch to left or right channels, depending on the position of the arm within the arm slot of level s(n). (If this arm is originally positioned to the left, then S branches left, and vice versa.) See Figure 7.

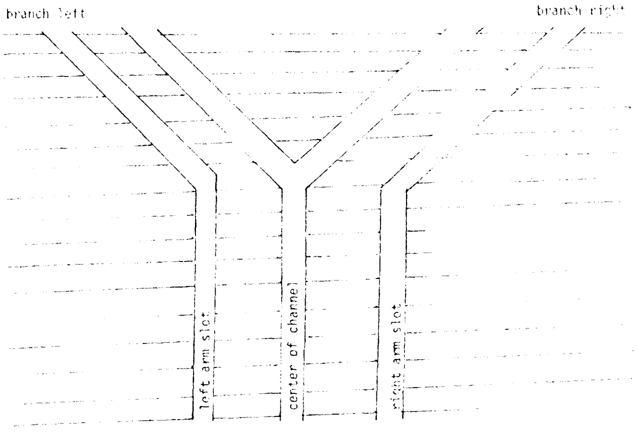

Figure 7: Cross-section at level s(n) of a junction which forces S to branch left or right depending on the position of that arm at level s(n).

We assume that the read/write head of M is not moved on any transition during which a tape cell is written.

In a state transition in which the currently scanned tape cell is set to 1, we have a corresponding channel with a special cross-section described in Figure 8, which forces the arm at level s(n) to move from the left to the right. (Note: the arms of S are so configured that it is not possible for an arm to be positioned in the center of a channel cross-section.) The case in which a tape cell originally 1 is set to 0 is symmetric.

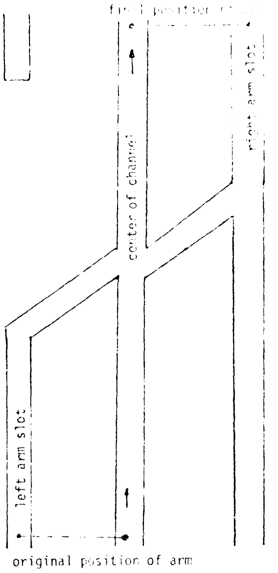

Figure 8: A channel cross-section at level s(n) which forces the arm of S at depth s(n) from the left to the right position of the arm slot.

In the initial position p_I, let S be within the junction corresponding to the initial state q_I and with the arms of S encoding the input string w. Let the final position p_F have S within the junction corresponding to the final state q_F and with the arms of S in some arbitrary position. (We assume that this final junction does not constrain the arms of S.)

In the resulting generalized mover's problem, S can be moved from position p_I to position p_F if and only if M accepts input string ω.

3. A Polynomial-Space Algorithm for the Generalized Mover's Problem

We now sketch a simple polynomial-space algorithm for the generalized mover's problem in d-space, with a set of polyhedral obstacles R fixed in d-space and a connected set S of polyhedra freely linked at various distinguished vertices, and distinguished positions p_I and p_F of S. The input is assumed to be specified within accuracy τ, $0 < \tau < 1$.

A *position* p of S will be specified by a sequence $p = (x_1, \ldots, x_{d'})$ of length $d' = d+(d-1)h+1$ with each element of x_i given within accuracy τ. We wish to determine if S can be moved from position p_I to position p_F without contacting any of the obstacles of R.

Let $\Phi_{R,S}(p)$ be the predicate which is *true* iff position p is both a feasible position of S and S contacts no element of R. $\Phi_{R,S}$ can be specified by a system of multinomial inequalities (of size polynomial in the input) over given values of the variables $x_1, x_2, \ldots, x_{d'}$. Thus $\Phi_{R,S}(p)$ can be computed, within accuracy τ, in polynomial time for any given position p.

The following algorithm runs in nondeterministic polynomial space (and can be implemented in deterministic polynomial space[7]).

Algorithm A

```
     Input $(R,S,p_I,p_F)$
[1]  $p \leftarrow p_I$
[2]  if $\Phi_{R,S}(p) \neq$ true then fail
[3]  if $p = p_F$ then accept
[4]  nondeterministically choose a position p'
     of S so that no vertex of S is moved more
     than $\tau$
[5]  $p \leftarrow p'$
[6]  goto [2].
```

To solve a generalized mover's problem (R,S,p_I,p_F) within accuracy τ, we let $S_{+\tau}$ and $S_{-\tau}$ be derived from S by *expanding* by a factor $(1+\tau)$ and *contracting* by $(1-\tau)$, respectively.

First apply algorithm A to $(R,S_{+\tau},p_I,p_F)$; if the algorithm accepts then output "yes" and halt. Else apply algorithm A to $(R,S_{-\tau},p_I,p_F)$; if the algorithm does not accept then output "no." Otherwise, output "maybe." (Recall that a correct algorithm for the mover's problem is allowed to output an inconclusive answer if the problem is not determinable within accuracy τ.)

4. A Polynomial-Time Algorithm for the Classical Mover's Problem in 2 and 3-D

We consider here the classical mover's problem (R,S,p_I,p_F) for dimensions $d = 2$ and 3. Recall from Section 1 that this problem is restricted to moving a *single* polyhedron S (with *no freely connected links*) through *d-space* avoiding polyhedral obstacles R. Our basic approach is to transform the classical mover's problem to the problem of moving *a single point in d'-space* (*where $d' = 3$ if $d = 2$ and $d' = 6$ if $d = 3$*), avoiding certain obstacles (forbidden subspaces). Each position of S corresponds to a point in d'-space, and each of the obstacles in the transformed mover's problem corresponds to positions in which S contacts an obstacle of R (see Figure 9).

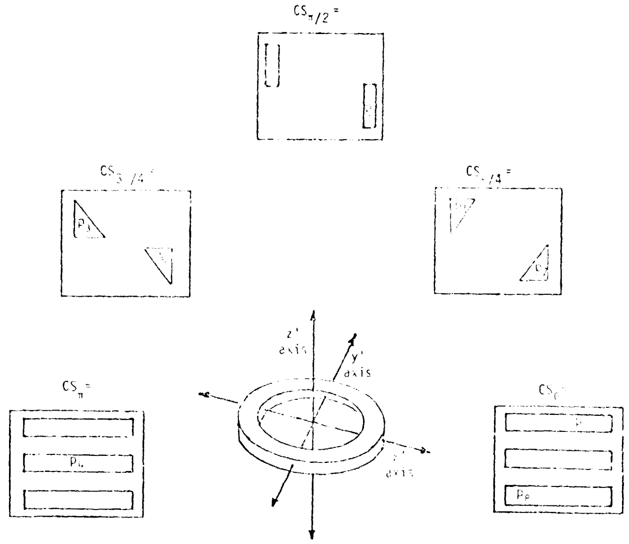

Figure 9: Transformed mover's problem from Figure 1. The 1-contact sets (the obstacles of the transformed problem) define a torus with cross-sections CS_θ illustrated for $\theta = 0, \pi/4, \pi/2, 3\pi/4, \pi$. S may be moved through positions $p_I = p_0, p_1, \ldots, p_8 = p_F$ as in Figure 2.

Certain previous research has also taken this approach[3,4]. The fundamental difficulty is that the obstacles in the transformed problem are nonlinear. Lozano-Perez and Wesley[4] approximate the obstacles in the transformed problem by linear constraints; however, to solve the mover's problem within the accuracy τ of the input, in the worst case an exponential number (in the input size) of linear constraints are required to approximate the obstacles in the transformed problem. To solve the 2 and 3 dimensional mover's problems within both polynomial time and accuracy τ, we describe the obstacles in the transformed problem by systems of multinomials of low degree with coefficients specified within accuracy τ.

Some definitions are now required. For each position p of S, let the contact vertices of p be those vertices of R contacting S, and those vertices of S contacting R. For $\mu \geq 0$, let a μ-contact set be a maximal connected set of positions with at least μ common contact vertices. If $|R|,|S|$ are the number of inequalities defining R,S, then there are $O(|R|^{\mu}|S|^{\mu})$ μ-contact sets.

We can show that for $\mu \geq d'$, any μ-contact set is also a d'-contact set. For $\mu = 2,3,\ldots,d'$ the μ-contact sets will be constructed by intersecting all (μ-1)-contact sets with 1-contact sets.

Consider, for example, a classical mover's problem (R,S,p_I,p_F) restricted to 2-space, with the obstacles R consisting of a set of line segments and S a single polygon. A position of S can be specified by a triple (x,y,θ) where (x,y) are the cartesian coordinates of some fixed vertex of S and θ is the angle of rotation around this vertex. We define a mapping t from the position of S to 3-space. Let $t(x,y,\theta) = (x',y',z')$ where $y = z'$, $\tan(\theta) = x'/y'$, and $x = (x')^2 + (y')^2 - \delta$, for some sufficiently large constant $\delta \geq 0$. (δ may be taken as the diameter of a circle enclosing S.) See Figure 10.

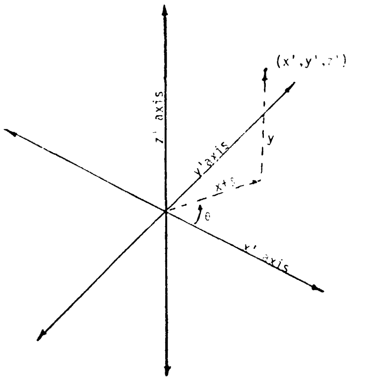

Figure 10: The mapping $t(x,y,\theta) = (x',y',z')$.

In this case with d = 2, the 1-contact sets (the obstacles in this transformed problem in 3-space) are quadratic surface patches of the form: $\{r(x,y,\theta) | \alpha_0 x+\beta_0 y = \cos(\theta-\theta_0)$ and $\alpha_i x+\beta_i y \geq \lambda_i \cos(\theta-\theta_i)$ for $i = 1,2\}$ where the α_i, β_i, λ_i, θ_i are constants given within the accuracy τ of the input.

The quadratic inequalities describing each of these $O(|R|\,|S|)$ 1-contact sets can be determined in constant time from pairs of line segments and vertices taken from R and S.

The $O(|R|^2|S|^2)$ 2-contact sets are formed from intersections of all pairs of 1-contact sets. Comba[8] presents an efficient procedure for determining the intersection of general quadratic surface patches. Since the 1-contact sets are linear x and y in the case of d = 2, each of the 2-contact sets are quadratic and can be simply determined without the application of Comba's procedure. The 3-contact sets may be computed by Comba's procedure from intersections of 1 and 2-contact sets, both of which are quadratic.

In the 3-dimensional version of the classical mover's problem (R,S,p_I,p_F), a position of S can be specified by a tuple $(x,y,z,\theta_1,\theta_2,\theta_3)$ where (x,y,z) are the cartesian coordinates of a distinguished vertex of S and $\theta_1,\theta_2,\theta_3$ specify the angular displacement of S. An appropriate mapping t can be defined from positions of S to 6-space. In this case 1-contact sets (the obstacles in this transformed problem) can be defined by constraints of the form:

$$\{t(x,y,z,\theta,\psi) | \alpha_0 x+\beta_0 y+\gamma_0 z = \cos(\theta_1-\psi_{01})\cdot\cos(\theta_2-\psi_{02})\cdot\cos(\theta_3-\psi_{03})$$

and

$$\alpha_i x+\beta_i y+\gamma_i z \geq \lambda_i \cos(\theta_1-\psi_{i1})\cos(\theta_2-\psi_{i2})\cos(\theta_3-\psi_{i3})$$

for $i = 1,2,\ldots,k\}$ where the α_i, β_i, γ_i, λ_i, ψ_{ij} are constants.

Again, each of these systems of inequalities describing a 1-contact set can be easily constructed by considering pairs of faces and vertices of R and S.

For $\mu = 2,3,\ldots,6 = d'$ the μ-contact sets are constructed by intersecting all 1-contact and (μ-1)-contact sets. (These intersections require the repeated solution with accuracy τ of multinomials of low degree with many subcases; the details will be presented in a later version of this paper.)

Now we sketch the further computations required to solve the mover's problem in 2 and 3 dimensions, assuming the μ-contact sets have been determined.

Let a μ-unique contact set be a maximal connected set of positions, each with precisely the same set of contact vertices of size μ. Our goal is to determine all 0-unique contact sets; these are the maximal connected sets of positions between which S may be moved without contacting an obstacle in R.

For each $\delta = 0,1,\dots,d'$, let C_δ be the class of μ-contact sets of dimension δ, with $\mu \geq 1$. Also, let $\bar{C}_\delta$ be the class of all μ-unique contact sets of dimension δ, with $\mu \geq 0$. Observe that $\bar{C}^{d'}$ is the set of all o-unique contact sets, as required.

We can easily show that $\bar{C}_0 = C_0$. For $\delta = 1,\dots,d'$ we have an efficient algorithm for constructing $\bar{C}_\delta$ from the previously constructed C_δ and $\bar{C}_0,\dots,\bar{C}_{\delta-1}$. (In the case that $\delta = 1$, the C_1 are "1-dimensional curves" and the $\bar{C}_1$ correspond to the "curve segments" partitioned by the "points" of $\bar{C}_0 = C_0$. In the case that $\delta = 2$, the C_2 are "surface patches" partitioned into planar maps by $\bar{C}_0$ (the "vertices") and $\bar{C}_1$ (the "edges"). The $\bar{C}_2$ are the faces of these maps. The cases $\delta > 2$ are similar, with C_δ partitioned by $\bar{C}_0,\dots,\bar{C}_{\delta-1}$ into a δ-cell complex.)

Finally, to determine if S may be moved from position p_I to p_F without contacting an obstacle of R, we test if:

1) $\Phi_{R,S}(p_I) = \Phi_{R,S}(p_F) = \text{true}$ (where $\Phi_{R,S}$ is the polynomial-time predicate of Section 3 which holds for all positions for which there is no contact between S and R) and

2) p_I, p_F are in the same o-unique contact set $P \in \bar{C}^{d'}$.

If this test is successful, we construct a move for S from p_I to p_F, as follows. Let p_I' (p_F') be any position of S on the boundary of the o-unique contact set P, reached by a linear translation of S from p_I (p_F). Then simply move S on the boundary of P from p_I' to p_F'.

The worst case time cost of our algorithm for the mover's problem is $O(|R|^{d'}|S|^{d'})$, (where $|R|$ and $|S|$ are the number of inequalities defining R and S).

The time complexity of this algorithm can be significantly improved in certain special cases by applying the divide and conquer techniques of Preparata and Muller[9] and Zolnowsky[10]. For example, the 2-dimensional mover's problem requires $O(k^4|R|\,|S|)$ time if S is convex and R consists of k convex obstacles. Also, in the case where the mover's problem contains numerous obstacles with very fine detail (i.e., many vertices) we have developed efficient approximate techniques which enclose nonconvex sets of obstacles with convex polyhedra, using again the convex hull algorithms of Preparata and Muller and of Zolnowsky.

5. Conclusion: Applications - Theoretical & Applied

This work was originally motivated by applications to robotics: the author felt it was important to examine computational complexity issues in robots given the recent development of mechanical devices autonomously controlled by micro and minicomputers, and the swiftly increasing computational power of these controllers (see Paul[11]).

However, our computational complexity results have more general applications, both theoretical and applied (those detailed below are discussed more thoroughly in the full paper).

The technique (used in our polynomial time algorithm for the mover's problem) of mapping an optimization problem in low dimension to a simpler problem in higher dimensions can also be applied to a variety of other optimization problems in computational geometry. For example, we present a polynomial time algorithm for a packing problem discussed by Shamos[12] involving the optimal orientation of identically shaped polygons so as to minimally pack the polygons within a thin rectangular strip (each polygon must have the same orientation). The transformed problem in this case is to find the minimal width cross-section (corresponding to an optimal orientation) of a torus defined by quadratic surface patches.

The reduction technique used in Section 2 to show that the generalized mover's problem is P-space complete can also be used to show a variety of other combinatorial problems P-space complete. (This is significant, since almost all known P-space complete combinatorial problems are games.) For example, we show the following coloring problem to be P-space complete:

Input Graph $G = (V,E)$ with initial and final k-colorings C_I, C_F respectively, and integer $m > 0$.

Problem Is there a sequence of k-colorings of G, $C_I = C_0, C_1, \dots, C_{j-1}, C_j = C_F$ such that C_i differs from C_{i-1} on at most m vertices?

(Similar P-space complete problems can be derived from other known NP-complete problems such as 3-SAT, KNAPSACK, and HAMILTONIAN-CIRCUIT.)

6. References

[1] Nilsson, N.J., "A mobile automaton: An application of artificial intelligence techniques," Proceedings, IJCAI-69, 509-520.

[2] Ignat'yev, M.B., F.M. Kulakov, and A.M. Pokrovski, "Robot-Manipulator Control Algorithms," NTIS Report JPRS 59717, August 1973.

[3] Udupa, S., "Collision detection and avoidance in computer controlled manipulators," Cal. Inst. Tech., Ph.D. thesis, 1977.

[4] Lozano-Perez, T. and M.A. Wesley, "An algorithm for planning collision-free paths amongst polyhedral obstacles," IBM T.J. Watson Research Report RC 7171, June 1978.

[5] Paul, R., "Modelling trajectory calculation and servoing of a computer controlled arm," Ph.D. thesis, Stanford University, Nov. 1972.

[6] Widdoes, C., "A heuristic collision avoider for the Stanford robot arm," Stanford CS Memo 227, June 1974.

[7] Savitch, W.J., "Relationships between nondeterministic and deterministic tape complexities," J. Computer Sci. 4, 177-192, 1970.

[8] Comba, P.G., "A procedure for detecting intersections of three-dimensional objects," J. ACM 15, 3, 354-366, July 1968.

[9] Preparata, F.P. and D.E. Muller, "Finding the intersection of a set of n half-spaces in time O(nlogn)," Univ. Illinois Coordinated Science Lab. TR ##-803(ACT-7); UILU-ENG77-2250, December 1977.

[10] Zolnowsky, J.E., "Topics in computational geometry," SLAC Report 206, STAN-CS-78-659, May 1978.

[11] Paul, R., "Robots, Models, and Automation," Computer, 19-27, July 1979.

[12] Shamos, M.I., "Notes on combinatorial geometry," Computer Science Department, Carnegie-Mellon University, 1977.