Utilizing the Topology of Configuration Space in Real-Time Multiple Manipulator Path Planning

John J. Fox Anthony A. Maciejewski
School of Electrical Engineering
Purdue University
West Lafayette, Indiana 47907

Abstract— **This paper provides a set of algorithms which allow qualitative information regarding the connectivity of configuration space to be quickly established. A mechanism is presented which utilizes these results to determine the effects of the motions of one manipulator on the configuration space of the other. These algorithms are then used as a basis for a simple planner which is capable of rapidly computing collision-free paths for multiple SCARA manipulators operating within overlapping workspaces.**

I. Introduction

Recently, there has been a growing recognition of the advantages achievable by placing more than one manipulator in a common workspace. Besides being able to perform tasks in parallel, the manipulators may be used cooperatively, thereby increasing the dexterity and load carrying capabilities which may be brought to bear on a particular task. Unfortunately, these advantages come at a cost, including the problem of determining paths for each of the manipulators which will avoid striking obstacles in the environment while, at the same time, avoiding collisions with the other robot.

In the past several years there have been numerous approaches to this problem including treating the manipulators as a redundant system [1] and the use of cellular decomposition techniques [2]. Among the numerous related algorithms which consider the motion of robots moving amidst obstacles are the spatial indexing of configuration space-time [3], and the use of the relative velocities of the objects and the robots to transform the problem into one of several static problems.

One particularly popular approach [4] imposes priorities upon the manipulators and then plans the paths of one robot at a time, using the higher priority robots as obstacles in the configuration space-time representation of the lower priority robots. Another common approach to planning paths for robots which must avoid moving obstacles is to decompose the problem into a two phase approach, commonly referred to as Path-Velocity Decomposition [5]. In this approach, the problem is simplified by solving for the motion amongst the static obstacles and subsequently planning the velocity along these paths so as to avoid the moving obstacles. Although this approach is both conceptually as well as computationally appealing, it suffers from being unable to find paths in situations in which a solution may be intuitively obvious. In particular, as illustrated in Fig. 1, there are cases in which the solution to the first phase of planning results in paths along which no appropriate velocity profile exists. Often, if the path planning phase had chosen some other path for either of the two manipulators, then a solution to the overall problem would have been found. This work focuses on this issue and attempts to find a solution in such a situation.

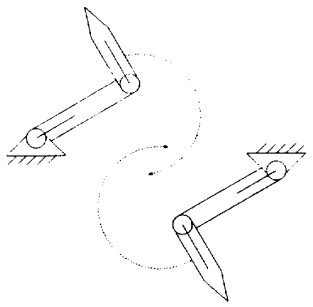

Fig. 1. A typical situation in which the initial choice of paths keeps a path-velocity decomposition planner from finding a solution during velocity planning. The arrows indicate the volume which will be swept through by the links.

A. Notation/Terminology

Throughout the paper, the following notation will be employed.

$\mathcal{W}$	= the workspace
$\mathcal{C}$	= the configuration space
$\mathcal{B}_i$	= an obstacle in the workspace
$\mathcal{B}$	= all of the obstacles in $\mathcal{W}$
$\mathcal{CB}_i$	= the configuration space representation of the B_i
$\mathcal{CB}$	= all of the obstacles in $\mathcal{C}$
$\mathcal{C}_{free}$	= the manipulator's free space
θ_{12}	$= \theta_1 + \theta_2$
c_i	$= \cos\theta_i$
s_i	$= \sin\theta_i$

This work was supported by the National Science Foundation under grant CDR 8803017 to the Engineering Research Center for Intelligent Manufacturing Systems and by the NEC Corporation.

Furthermore, the following terms will be used. A "region" will be considered a path-connected subspace of $\mathcal{C}$ which has the same pair of obstacles to its left and right in configuration space. A "channel" will be some sequence of regions and a "path" will be a sequence of configurations in $\mathcal{C}_{free}$.

B. Assumptions

A number of simplifying assumptions were made in this work. The most obvious was to model obstacles in the workspace as points and the SCARA manipulators as line segments. The purpose of these initial simplifications was to focus the presentation on fundamental aspects of the algorithm. Section III develops a more general characterization of obstacles and [6] discusses robot links which have been modeled as polygons.

C. Overview of Paper

The remainder of this paper is organized as follows: Section II reviews basic results from forward and inverse kinematics. Algorithms for establishing the presence of intersections between obstacles in $\mathcal{C}$ are developed in Section III. Section IV begins by presenting a mechanism for mapping connected regions in one manipulator's configuration space into the other manipulator's configuration space. It concludes by using the various algorithms which have been developed as the basis for a simple planner which computes collision-free motions for multiple manipulators. Section V illustrates the operation of this planner. Finally, Section VI provides a discussion of the results of this work and indicates some of its limitations.

II. Kinematics

The benefits of path planning in a robot's configuration space [7] have been well established in the literature [8][9]. The underlying concept of this approach is the recognition that a robot may be represented as a point in configuration space traveling through a set of obstacles which are obtained as the result of a transformation on the real obstacles in the manipulator's workspace. The process of path planning is then heavily dependent on the relationship between the manipulator's configuration space, $\mathcal{C}$, and its workspace, $\mathcal{W}$. In this section, the nature of the relationship between these two spaces is presented at both the position level and the velocity level.

For the manipulator depicted in Fig. 2, the transformation which describes the relationship between a manipulator's configuration, (θ_1, θ_2), and the Cartesian position of the end effector, (x_{eff}, y_{eff}), is easily calculated using forward kinematics [10] as

$$\begin{bmatrix} x_{eff} \\ y_{eff} \end{bmatrix} = \begin{bmatrix} L_1 c_1 + L_2 c_{12} \\ L_1 s_1 + L_2 s_{12} \end{bmatrix} \tag{1}$$

where L_1 and L_2 are the lengths of links 1 and 2.

Solving (1) for θ_1 and θ_2 yields the equally well-known inverse transformation describing the inverse kinematics

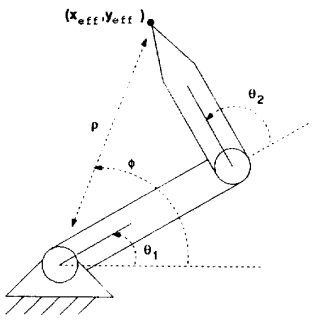

Fig. 2. The geometry of a SCARA-type manipulator.

for this manipulator:

$$\theta_1 = \tan^{-1}\frac{y_{eff}}{x_{eff}} \pm \cos^{-1}\frac{x_{eff}^2 + y_{eff}^2 + L_1^2 - L_2^2}{2L_1\sqrt{x_{eff}^2 + y_{eff}^2}} \tag{2}$$

and

$$\theta_2 = \pm\cos^{-1}\frac{x_{eff}^2 + y_{eff}^2 - L_1^2 - L_2^2}{2L_1 L_2}. \tag{3}$$

The relationship between the end effector velocities and the joint velocities is readily obtained by differentiating (1) to obtain

$$\begin{bmatrix} \dot{x}_{eff} \\ \dot{y}_{eff} \end{bmatrix} = \begin{bmatrix} -L_1 s_1 - L_2 s_{12} & -L_2 s_{12} \\ L_1 c_1 + L_2 c_{12} & L_2 c_{12} \end{bmatrix} \begin{bmatrix} \dot{\theta}_1 \\ \dot{\theta}_2 \end{bmatrix} \tag{4}$$

Recall that the ultimate goal of this development is to provide a mechanism for rapidly determining whether obstacles in $\mathcal{C}$ intersect. Computing this intersection using (2) and (3) would require the simultaneous solution of nonlinear equations, a task which is, in general, nontrivial. However, for many of the purposes of this work, it will suffice to establish the presence of an intersection without knowing precisely where it occurs. As will be shown in Section III, a characterization of configuration space obstacles has been developed which is sufficient for establishing the presence of intersections between configuration space obstacles. The basis of this characterization is a representation of the obstacle by its tangents. As shown in [6], the tangent of an obstacle in $\mathcal{C}$ corresponding to a point in $\mathcal{W}$ is readily obtained as

$$\frac{d\theta_2}{d\theta_1} = \frac{\ell_2 + L_1 c_2}{-\ell_2} \tag{5}$$

where ℓ_2 is the length along the second link at which the contact with the obstacle occurs.

III. The Topology of $\mathcal{C}_{free}$

The notion of utilizing topological properties in generating obstacle representations for planning has been

studied extensively. Several researchers [11] [12] have generated representations of $\mathcal{C}$ by convolving a representation of a free flying robot with a representation for an obstacle. Because the mechanism which was used for performing these convolutions tended to produce a small number of vertices, edges, and faces which were either redundant or otherwise non-realizable, a second stage was employed which utilized topological information to cull out these extraneous pieces of information. Additional researchers [13] have chosen to characterize the topology of $\mathcal{W}$ rather than $\mathcal{C}$ and use an octree representation to evaluate the connectivity of the free workspace. Here, the topological properties of the obstacles are used as a filter for removing regions of $\mathcal{C}$ in which a collision-free path cannot be found.

Before discussing the details of calculating topological features of $\mathcal{C}$, it is instructive to consider what type of global features can be used to improve the efficiency of most path planners. One important feature of free space for two-dimensional revolute manipulators which has been previously identified, is the existence of "highways" [14]. Physically, a highway is a distinguished subspace of configuration space for which a collision free path can be planned simply by using a line segment parallel to the θ_1 axis for some relatively large range of θ_2 values. Other global features of free space which are not guaranteed to exist include a path from one highway to the other, referred to as an "isthmus." If this feature does not exist then this implies that the free space is further partitioned. Regions of free space that are connected to only one of the two highways will be referred to as "peninsulas". Additional details on these properties and their computation may be found in [6], however it should be apparent from their definitions that the ability to determine whether obstacles in $\mathcal{C}$ intersect will be critical to any algorithm which will be used. The purpose of the remainder of this section is to address the question of how to best determine whether such an intersection exists.

A. Intersections Between Point Obstacles in $\mathcal{W}$

Consider the ways in which a SCARA manipulator may come into contact with a pair of point obstacles:

1. both contacts may take place along the first link,
2. one contact may be along the first link and the other along the second link,
3. both contacts may take place along the second link.

Testing for the first condition is trivial. If the two obstacles are represented in polar coordinates as (ρ_f, ϕ_f) and (ρ_g, ϕ_g), then the obstacles intersect if both are at a radius less than L_1 and $\phi_f = \phi_g$. The second case is only slightly more complicated since one must only check to see whether the two values of θ_1 for the end effector to be in contact with the one obstacle bracket the value of ϕ for the obstacle at $\rho \leq L_1$. The remainder of this section will consider the final case.

Let $\mathcal{B}_f$ and $\mathcal{B}_g$ be point obstacles in $\mathcal{W}$ which have the Cartesian coordinates (x_f, y_f) and (x_g, y_g) with respect to the base of the manipulator. If the points are both assumed to be at a radius greater than L_1, then the following lemma provides a necessary condition for testing for an intersection between obstacles in configuration space.

Lemma 1: If $\mathcal{CB}_f \bigcap \mathcal{CB}_g \neq \emptyset$ then $\mathcal{CB}_f \bigcap \mathcal{CFL} \neq \emptyset$ and $\mathcal{CB}_g \bigcap \mathcal{CFL} \neq \emptyset$ where

$$\mathcal{CFL} = \left\{ (\theta_1, \theta_2) \mid \theta_1 + \theta_2 = \tan^{-1}\left(\frac{y_f - y_g}{x_f - x_g}\right) \right\}. \quad (6)$$

The proof to this can be readily seen by recognizing that Lemma 1 merely states that the second link of the manipulator must be parallel to the line supporting the two obstacles if it is to be in contact with both of them simultaneously. Unfortunately, because the inverse tangent function does not return a unique value, application of this lemma would require checking for the intersection of the configuration space obstacles with each of the lines which have slope -1 and an appropriate intercept. However, by simply choosing a particular member of this set of functions, not only does this lemma become easier to apply, but it is also strengthened in such a way as to become sufficient. In particular, if the direction in polar coordinates of the vector from $\mathcal{B}_g$ to $\mathcal{B}_f$ is denoted by ϕ_Δ, then the intersection between $\mathcal{CB}_g$ and $\mathcal{CB}_f$ can be established by testing for intersection with the particular line

$$\theta_1 + \theta_2 = \phi_\Delta. \quad (7)$$

If the set of configurations which lie along this line are denoted $\mathcal{CL}$ where

$$\mathcal{CL} = \{(\theta_1, \theta_2) \mid \theta_1 + \theta_2 = \phi_\Delta\} \quad (8)$$

then one can obtain the following lemma.

Lemma 2: Given the assumptions and definitions of the previous paragraphs, $\mathcal{CB}_f \bigcap \mathcal{CB}_g \neq \emptyset$ if and only if $\mathcal{CB}_f \bigcap \mathcal{CL} \neq \emptyset$ and $\mathcal{CB}_g \bigcap \mathcal{CL} \neq \emptyset$.

Physically, this condition states that if $\rho_f \geq \rho_g > L_1$ then the second link of the manipulator must point in the direction defined by the vector from the obstacle at a smaller radius to the obstacle at a larger radius. The necessity of this condition may be established fairly easily from Lemma 1. The sufficiency condition may also be established by using a simple geometric construction to examine the elements which are members of the intersections of $\mathcal{CL}$ with $\mathcal{CB}_f$ and $\mathcal{CB}_g$.

At first glance, it may appear that the benefits of having established this property are negligible since the net effect appears to have been to eliminate the need to calculate the intersection between two nonlinear functions at the expense of now having to twice calculate the intersection between nonlinear functions with a straight line. Fortunately, however, the obstacles in configuration space have additional properties which prove particularly useful.

Recall from calculus that the local extrema in distance from a curve to a line is at the points along the curve at which the tangent matches the slope of the line.

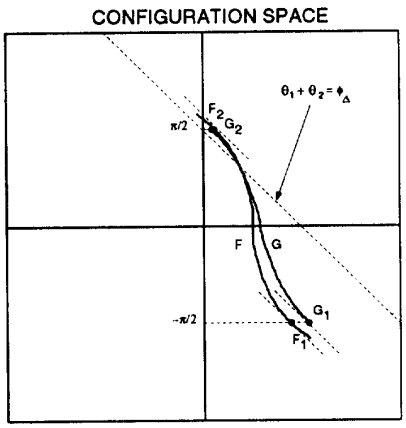

Fig. 3. An example in which the intersection test succeeds. F_i and G_i represent the points which must be computed in order to apply the test.

Setting the slope of the obstacle as described by (5) equal to the slope of the constraint equation results in

$$\frac{\ell_2 + L_1 c_2}{-\ell_2} = -1, \tag{9}$$

which has the solution $\theta_2 = \pm\pi/2$. If the obstacle does not extend past $\theta_2 = \pm\pi/2$, then the local extrema with respect to these lines will be at those points which correspond to the end effector resting upon the obstacle. Hence, the local extrema of an obstacle with respect to the line along which an intersection must lie may be calculated via two applications of the inverse kinematics function. Thus, an algorithm to determine if the line defined by (8) intersects a configuration space obstacle, would evaluate the curve describing the obstacle at only two points, i.e. those at $\theta_2 = \pm\pi/2$, and determine if they lie in the opposing half spaces defined by the line along which the intersection must take place. Similar results exist for obstacles which are not constrained to lie at a radius greater than L_1. An example of this test succeeding is illustrated in Fig. 3.

B. Intersections Between Points and Line Obstacles

In the previous section, the manner in which a manipulator may come into contact with multiple point obstacles was analyzed. This yielded an easily computed test for establishing the presence of intersections between the representation of point obstacles in $\mathcal{C}$. This section follows an analogous development for somewhat more general obstacles, namely line segments and circles. The interest in these particular classes of obstacles will become apparent in Section IV where the effects of motions through regions in one manipulator's configuration space are studied with regards to their effects on the connectivity of the other manipulator's configuration space.

As in the test developed above, the first step in developing this algorithm is to characterize the line segment with regards to its local extrema in $\mathcal{C}$ with respect to lines of slope -1. Clearly, these extrema must take place along the boundary of the configuration space obstacle. If one considers the manner in which the manipulator may be in contact with the obstacle then it becomes clear that this boundary can be decomposed into simpler curves corresponding to cases in which either the second link of the manipulator slides along the end points of the line segment, or by the end effector of the manipulator traversing the interior of the line segment in a manner which is described by (4). The extrema of the obstacle may be computed by determining the extrema along each of the four portions of the boundary and applying appropriate logic.

Let $\mathcal{B}_S$ be a line segment in $\mathcal{W}$ with endpoints $\mathcal{B}_f = (x_f, y_f)$ and $\mathcal{B}_g = (x_g, y_g)$. A necessary first step in the characterization of $\mathcal{CB}_S$ is then to characterize $\mathcal{CB}_f$ and $\mathcal{CB}_g$ as in the previous section. The extrema of the remaining portions of $\mathcal{CB}_S$ may be computed by considering the obstacle at the velocity level. If it is assumed that $[\delta x, \delta y]$ is a vector along the line segment, then the tangent of those portions of the boundary due to the traversal of the end effector along the interior of the line segment is obtained by solving (4) as

$$\frac{d\theta_2}{d\theta_1} = \frac{-L_1 c_1 \delta x - L_2 c_{12} \delta x - L_1 s_1 \delta y - L_2 s_{12} \delta y}{L_2 c_{12} \delta x + L_2 s_{12} \delta y}. \tag{10}$$

Setting this slope to -1 and solving for θ_1 yields the condition that

$$\theta_1 = \tan^{-1}\left(-\frac{\delta x}{\delta y}\right) \tag{11}$$

which must be satisfied in order to be at a local extrema along the interior portions of $\mathcal{CB}_S$. Before continuing, it should be stressed that this result is not limited to line segments. In fact, the extrema of any obstacle with respect to lines in $\mathcal{C}$ of slope -1 which place the end effector on the obstacle will always satisfy this condition, so long as the obstacle may be represented by a differentiable curve.

From (11), the specific configurations at which the extrema occur are given by

$$\theta_1 = \tan^{-1}\left(-\frac{x_f - x_g}{y_f - y_g}\right). \tag{12}$$

Determining the corresponding value of θ_2 may be performed via the inverse kinematic equations.

Now, consider a point obstacle at (x_i, y_i). If $K_1 = \tan^{-1}\left(\frac{y_f - y_i}{x_f - x_i}\right)$ and $K_2 = \tan^{-1}\left(\frac{y_g - y_i}{x_g - x_i}\right)$, then the potential orientations of the second link which bring it simultaneously in contact with both the point obstacle and the line segment may be described by the family of lines

$$\mathcal{CL}_{seg} = \{(\theta_1, \theta_2) \mid K_1 \leq \theta_1 + \theta_2 \leq K_2\} \tag{13}$$

where it has been assumed, without loss of generality, that $K_1 \leq K_2$. Given this information, and the characterization of $\mathcal{CB}_S$ which is illustrated in Fig. 4, an algorithm to test for intersections between the representations of points and line segments is readily obtained.

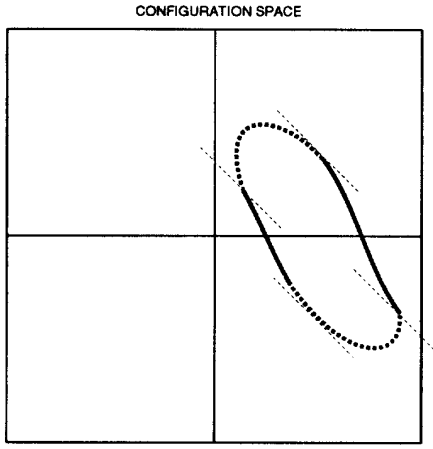

Fig. 4. The characterization of an obstacle due to a line segment. The solid lines are those portions of the obstacle due to the manipulator sliding along the end points of the line segment. The bold dashed lines represent those portions of the obstacle due to the traversal of the end effector along the interior of the line segment. The normal weight dashed lines are the extrema of the obstacle with respect to lines of slope -1.

C. Intersections Between Point and Circular Obstacles

Characterizing configuration space obstacles which represent circles or arcs may be done in a manner directly analogous to the method presented for characterizing line segments. First, consider the situation in which the end effector is lying along a circle $x = x_0 + r\cos(\psi)$ and $y = y_0 + r\sin(\psi)$. Differentiating the equations describing such a circle and applying (11) yields a description of those configurations in which the end effector is in contact with the circle while the corresponding obstacle in $\mathcal{C}$ is at a local extrema. This condition is described by,

$$\theta_1 = \psi + n\pi. \tag{14}$$

In other words, the first link must be parallel to the line segment between the center (x_0, y_0) and the point at which the end effector touches the obstacle. Substituting this condition into the forward kinematic equations and solving for those configurations which place the end effector on the circle yields those configurations as

$$\theta_1 = \tan^{-1}\left(\frac{y_0}{x_0}\right) \pm \cos^{-1}\left(\frac{d^2 + (\pm L_1 - r)^2 - L_2^2}{2L_1 d}\right) \tag{15}$$

and

$$\theta_2 = \mp\cos^{-1}\left(\frac{d^2 - (\pm L_1 - r)^2 - L_2^2}{2(L_1 - r)L_2}\right), \tag{16}$$

where $d = \sqrt{x_0^2 + y_0^2}$.

Determining the configuration which places the contact somewhere within the interior of the second link is most readily accomplished by recognizing two critical facts. First, the necessary condition on the relationship between θ_1 and the polar coordinates of the contact given by (14) continues to hold, so $\theta_1 = \psi + n\pi$. Second, for the configuration to lie along the boundary of $\mathcal{C}_{circle}$, the second link of the manipulator must lie along a tangent of the circle, hence $\theta_1 + \theta_2 = \psi \pm \frac{\pi}{2}$. When these facts are combined, it is clear that $\theta_1 = \psi + n\pi$ and $\theta_2 = \pm\frac{\pi}{2}$.

It is easy to show that the location along the link at which the contact takes place is given by

$$\ell_2 = \sqrt{-(L_1 - r)^2 + d^2}. \tag{17}$$

If the evaluation of (17) results in $0 \leq \ell_2 \leq L_2$, then the configurations satisfying this condition are given by

$$\begin{aligned} \theta_1 &= \cos^{-1}\left(\frac{x_0(L_1 - d) \pm \ell_2 y_0}{\ell_2^2 + (L_1 - d)^2}\right) \\ \theta_2 &= \mp\frac{\pi}{2} \end{aligned} \tag{18}$$

An approach similar to the one described for line segments would then result in an intersection test.

IV. A Simple Planner

The key to our approach to planning collision-free paths is a mechanism for meshing channels in different configuration spaces. The method for accomplishing this is based upon mapping regions in one configuration space into the configuration space of the other manipulator as an obstacle. Clearly, this approach is heavily motivated by the work of [4] in that the motions of one manipulator are modeled as an obstacle in the configuration space-time of the other manipulator. The difference in this work is that the choice of a specific path within the region is not considered until after the global planning stage has been completed. Instead the set of all possible paths through a region are, in effect, considered when generating the configuration space-time obstacle. The remainder of this section deals first with the mechanism chosen for transforming the set of possible motions of one manipulator into obstacles in the other manipulator's configuration space and then proceeds to describe the planner which has been implemented.

As mentioned above, the principal result of this section is an ability to study the sets of possible motions of one manipulator with regard to their effects on the topology of another manipulator's configuration space. More specifically, a representation is built that approximates the set of all possible postures of the robot when it is in any configuration within a region. The approach which was used to perform this operation relies heavily on the computation of those portions in the workspace called "shadows" [15] which describe the regions through which the link of the manipulator will sweep while it stays in contact with the obstacle. When an approximation is built which encloses the shadows of all of those points obtained by interpolating between the polar coordinates of the two obstacles along the boundaries of the region, the resulting area is a conservative approximation of the set of all postures in which the manipulator may find itself when it is at any configuration within the region. This is illustrated in Fig. 5. The resulting region may then be mapped into the configuration space of the other robot and treated

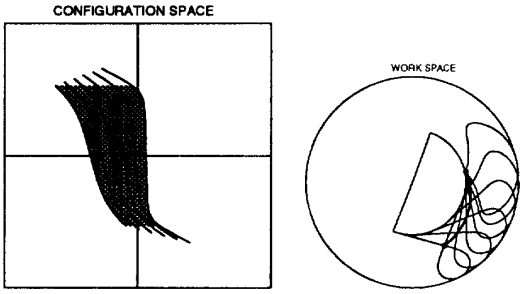

Fig. 5. The process used for determining the potential postures of a manipulator when it is in any configuration in a region. The region in $\mathcal{C}$ filled with grey is being mapped into its corresponding manipulator postures. Also depicted is the c-space representation of the artificial obstacle obtained by interpolating in polar coordinates between the actual obstacles. The corresponding workspace is also depicted. The bold, filled circles represent the actual obstacles. The bold line between them depicts the artificial obstacle. The normal weight curves represent the shadows of some of the points along the artificial obstacle.

as though it were a static obstacle in this manipulator's workspace. Finally, the effects of this artificial obstacle on the topology of the second manipulator's freespace are determined using the tests of the previous section.

First, consider the portion of $\mathcal{W}$ in which the manipulator may lie when it is in any configuration which brings it into contact with a point obstacle. Let $\mathcal{B}_i$ denote such a point obstacle which is at the polar coordinates (ρ_i, ϕ_i) with respect to the base of the manipulator. The curve drawn with a solid line in Fig. 6 illustrates the path followed by the end effector as the manipulator moves under such constraints. The equation describing this curve may be obtained in polar coordinates as

$$\Theta(r) = \cos^{-1}\left[\frac{\rho^2 + (L_2 - r)^2 - L_1^2}{2\rho(L_2 - r)}\right] \qquad (19)$$

where Θ is measured with respect to the line passing between the origin of the manipulator's base coordinate system and the point obstacle and r is measured with respect to the obstacle. Note that the relationship between this curve and the robot's configuration as it tracks this curve is given by $\Theta = \theta_1 + \theta_2 - \phi$.

The set of points swept out by the entire link as it slides along $\mathcal{B}_i$ may be determined by expanding the area enclosed by (19) to include all those points which may be reached by projecting a line segment of length L_2 from each of the points along (19) through $\mathcal{B}_i$. The line segments which form the boundaries of this region are quickly obtained by recognizing that they must represent the positions of the second link when it is tangent to the curve of (19) and its end point is at an extrema in Θ. Furthermore, the robot configurations which place the second link in such a position are given by noting that these extrema in Θ occur when $\theta_2 = \pm\frac{\pi}{2}$.

Finally, the construction is completed by considering those points which come into contact with the first link. This is accomplished by considering the sector of radius

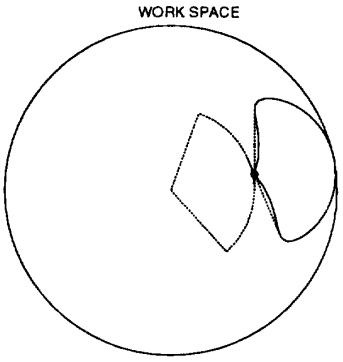

Fig. 6. Generating the shadow of one of the region's bounding obstacles. The solid line in the depiction of the workspace represents the path followed by the end effector as it slides along the point obstacle. The dashed line illustrates the portions of $\mathcal{W}$ which must be added to this area to obtain the shadow of the obstacle.

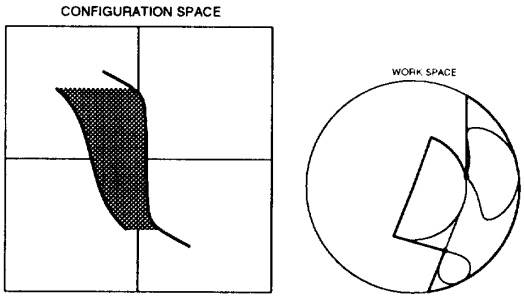

Fig. 7. The bold lines approximate the set of all possible positions for the manipulator when it is in any configuration in the region illustrated in grey. The figure also depicts the shadows of the two bounding obstacles.

L_1 which subtends the angle formed by the end points of the two line segments computed in the previous step and the base of the manipulator. The area which results from this construction is illustrated with dashed lines in Fig. 6. For the sake of notation, the portion of this region which lies at a radius greater than ρ_i will be referred to as the "outer shadow" of $\mathcal{B}_i$. The remainder of this region will be denoted the "inner shadow" of obstacle $\mathcal{B}_i$.

An approximation for the area in $\mathcal{W}$ in which the manipulator may lay when it is in any configuration for an entire region is illustrated in Fig. 7. This area is constructed by first establishing the shadows of each of the two obstacles forming the boundaries of the region. Then, it is expanded by extending the line segments forming the boundaries of the outer shadows so that an arc centered at the base of the manipulator and passing through the end points of these segments will be of sufficient radius to enclose both of the outer shadows. This approximation is then completed by including the inner shadows of the two obstacles along with the area which lies between them.

At this point, a conservative approximation has been developed which represents the entire area in $\mathcal{W}$ in which a manipulator may be when it is at any configuration within the region. The utility of this information comes when determining the effects of choosing any path through a region in one configuration space on the topology of the other manipulator's configuration space. Since the approximation which has been developed is composed entirely of line segments and arcs, considering this approximation to be an obstacle in the other robot's workspace permits the tests of Section III to be utilized in establishing the effects of one robot's motions on the topology of the other robot's freespace without knowing a priori which particular path will be chosen.

Given these results, a simple planning algorithm can be readily described. The basis of the work being presented, as well as that of our earlier work [6] is that the planning process may be broken into a two phase approach during which the free space is first searched for a channel using qualitative information on its topology and then fitted with a specific path using local geometric information. The process of searching for the channels is simplified by limiting the search to those which use the highways as intermediate goals, not unlike the approach employed in [14]. Furthermore, the search is guided by using the heuristic that the channel be the one most likely to yield the shortest path. These channels are then tested for potential conflicts by using the results of the previous sections. If a conflict is found via the tests of the previous section then a velocity planner is invoked to modify the rate at which the channel will be traversed. If the velocity planning does not yield a pair of conflict-free channels then the planner continues by iterating through pairs of channels of increasing length until it finds a solution.

More specifically, the planner is initialized by evaluating the topology of each manipulator's free space, a process which involves computing the sets of isthmuses and to which highways, if any, the initial and goal configurations are connected. Having done this, the planner proceeds by choosing the pair of channels which have not yet been examined and which are most likely to yield the shortest paths. By using its extent in θ_2 as an approximation for the time required to pass through a region, the two channels are then temporally synchronized. Once this is accomplished, the results of Sections III and IV are used to determine whether the traversal of a particular region will affect the connectivity of the other manipulator's configuration space in such a way as to indicate a potential collision. If so, the planner generates a new pair of channels and estimates the amount of time which would be required to traverse these new channels. If the time required to traverse the new pair exceeds the time to traverse the pair which has just failed, then the planner attempts to modify the manipulator velocities along the older pair of channels in an attempt to avoid collisions, as in Path-Velocity Decomposition. If the result of this attempt at velocity planning is a set of trajectories which require no more time than the estimated time to traverse the new set of channels, then the planner returns the trajectories as the result. If at some point a pair of conflict-free channels is found, then a local path planner is invoked to choose a particular path within each of the channels. Finally, if, at some point, the planner hits some predetermined limit on the estimated length of the path, then it returns with a failure.

V. An Example

In this section, the operation of the planner on a typical problem is illustrated. Consider the problem of planning a path between the configurations labeled (I) and (G) in Fig. 8(a). In the example which is illustrated, the planner first attempts to plan a path which leads the robot on the left along the path illustrated, while having the robot on the right traverse the isthmus closest to the initial configuration. Both solutions would be reasonable if considered individually, as they would be close to being the shortest paths in $\mathcal{C}$ for each manipulator. If, however, the robots were in fact to traverse these channels, then they would sweep through the regions in $\mathcal{W}$ illustrated in Fig. 8(b). and it is clear that a collision would occur regardless of the velocity profiles chosen along the paths. Hence, if a Path-Velocity Decomposition were employed in this situation, it is not unreasonable to believe that it would not be able to find a solution. However, by utilizing the algorithms described above, the planner is capable of quickly recognizing that the two proposed paths are unacceptable. It then chooses an alternate channel for one of the robots and tests this pair of channels for collisions. The resulting motions of the manipulators are shown in Fig. 8(c). In this case, the new pair of channels do not bring the manipulators into potentially dangerous situations and, as a result, it is not necessary to invoke the velocity planning phase. Computing this solution required approximately 130 ms of CPU time on a SPARC II workstation.

VI. Results/Conclusions

This paper has described an approach to the problem of planning collision-free motions for multiple SCARA manipulators operating within overlapping workspaces. The primary results have been the development of two fundamental concepts, namely:

1. the ability to quickly establish the presence of certain topological feature in $\mathcal{C}_{free}$, and

2. the ability to quickly compute an approximation of the effects of one robot's motion on the topology of the other robot without a priori knowledge of the particular path which will be chosen through the region.

These concepts have been illustrated by the development of a simple planning system for multiple SCARA manipulators which finds solutions which form a superset of those found through a straightforward implementation of Path-Velocity Decomposition. Furthermore, the low computational costs of generating candidate paths and testing them for interactions tends to offset the combinatoric nature of the search process to yield a relatively quick algorithm. The major drawbacks of the planner which has been presented is that it is not complete and some of the paths which result may be considerably less than optimal.

References

[1] C. Shih, J. Sadler, and W. Gruver, "Collision avoidance for two SCARA robots," *in Proc. 1991 IEEE Int. Conf. Robotics Automat.*, pp. 674–679, (Sacramento, CA), April 9-11 1991.

[2] S. Fortune, G. Wilfong, and C. Yap, "Coordinated motion of two robot arms," *in Proc. 1986 IEEE Int. Conf. Robotics Automat.*, pp. 1216–1223, (San Francisco, CA), April 7-10 1986.

[3] K. Fujimura and H. Samet, "Path planning among moving obstacles using spatial indexing," *in Proc. 1988 IEEE Int. Conf. Robotics Automat.*, pp. 1662–1667, (Philadelphia, PA), April 24-29 1988.

[4] M. Erdmann and T. Lozano-Pérez, "On multiple moving objects," *Algorithmica*, vol. , no. 2, pp. 477–521, 1987.

[5] K. Kant and S. W. Zucker, "Toward efficient trajectory planning: The path-velocity decomposition," *Int. J. Robotics Res.*, vol. 5, no. 3, Fall 1986.

[6] A. A. Maciejewski and J. J. Fox, "Path planning and the topology of configuration space," *IEEE Trans. Robotics Automat.*, vol. 9, no. 4, pp. 444–456, Aug. 1993.

[7] T. Lozano-Pérez, "Automatic planning of manipulator transfer movements," *IEEE Trans. on Syst., Man, and Cybern.*, vol. SMC-11, no. 10, pp. 681–698, October 1981.

[8] J. C. Latombe, *Robot Motion Planning*, Kluwer Academic Publishers, Boston, MA, 1991.

[9] Y. Hwang and N. Ahuja, "Gross motion planning-A survey," *ACM Comp. Surv.*, vol. 24, no. 3, pp. 219–292, Sept. 1992.

[10] K. S. Fu, R. C. Gonzalez, and C. S. G. Lee, *Robotics: Control, Sensing, Vision, and Intelligence*, McGraw Hill, New York, 1987.

[11] C. Bajaj and M. S. Kim, "Generation of configuration space obstacles: The case of a moving sphere," *IEEE J. Robotics Automat.*, vol. 4, no. 1, pp. 94–99, Feb. 1988.

[12] R. C. Brost, "Computing metric and topological properties of configuration-space obstacles," *in Proc. 1989 IEEE Int. Conf. Robotics Automat.*, pp. 170–176, (Scottsdale, AZ), May 14-18 1989.

[13] P. Wenger and P. Chedmail, "On the connectivity of manipulator free workspace," *J. Robotic Systems*, vol. 8, no. 6, pp. 767–799, 1991.

[14] W. Meyer and P. Benedict, "Path planning and the geometry of joint space obstacles," *in Proc. 1988 IEEE Int. Conf. Robotics Automat.*, pp. 215–219, (Philadelphia, PA), April 24-29 1988.

[15] V. Lumelsky, "Effect of kinematics on motion planning for planar robot arms moving amidst unknown obstacles," *IEEE J. Robotics Automat.*, vol. RA-3, no. 3, pp. 207–223, June 1987.

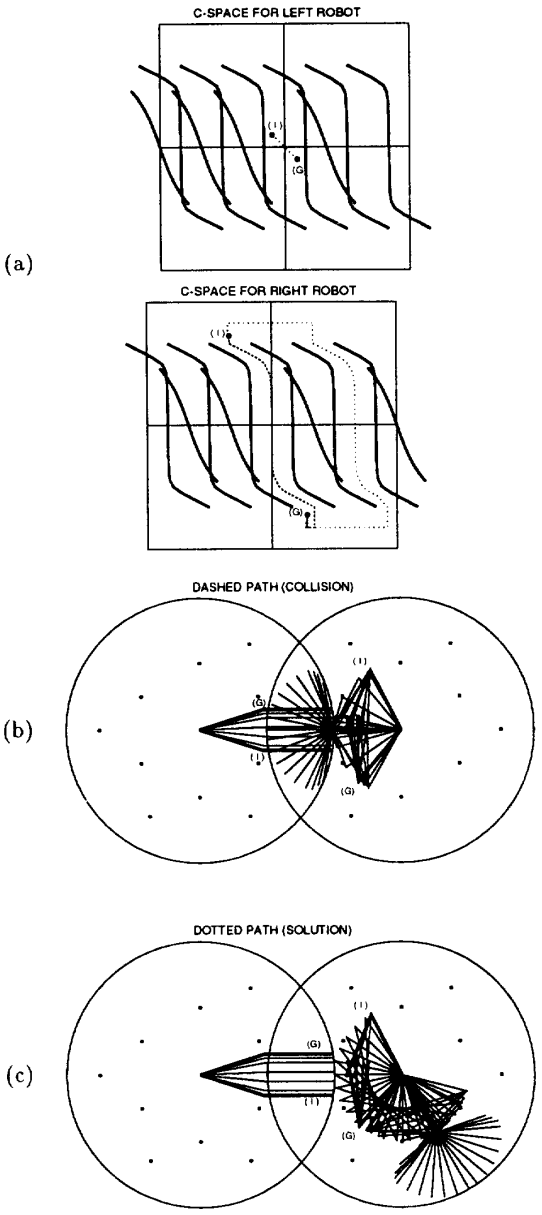

Fig. 8. An example of a situation in which a Path-Velocity Decomposition will not yield a solution although one exists. The configuration spaces in (a) depict the paths which are tried for each of the manipulators as it attempts to find a solution. The path for the manipulator to the right depicted with a dashed line yields a collision which cannot be avoided via velocity planning (see (b)). After the planner determines that a path is not possible, it backtracks and computes the path shown in (c), which is collision-free. This solution required approximately 130 ms. on a Sparc-II workstation.