A Geometric Approach to Hyper-Redundant Manipulator Obstacle Avoidance

G. S. Chirikjian

J. W. Burdick

School of Engineering and Applied Science,
California Institute of Technology,
Pasadena, CA 91125

The term hyper-redundant refers to redundant manipulators with a very large or infinite number of degrees of freedom. These manipulators are analogous in morphology to snakes, elephant trunks, and tentacles. While a variety of obstacle avoidance algorithms for nonredundant and mildly redundant manipulators exist, little analysis has been performed for hyper-redundant robots. This paper presents a strictly geometric algorithm for hyper-redundant manipulator obstacle avoidance which relies on the use of "tunnels" in the obstacle-filled workspace. Methods of differential geometry are used to formulate equations which guarantee that sections of the manipulator are confined to the tunnels, and therefore avoid obstacles. A general formulation is presented with an example to illustrate this approach.

1 Introduction

A hyper-redundant manipulator is a redundant manipulator in which the number of redundant degrees of freedom is large or infinite. These manipulators are analogous in morphology to snakes or tentacles. A number of recent works have been devoted to the design and system aspects of snake-like manipulators [8, 13, 14], although less work has been done in the kinematic analysis of these manipulators. Because of their highly articulated structures, hyper-redundant manipulators are naturally suited to operation in highly constrained environments.

Several methods for dealing with the problem of robot obstacle avoidance in a time-independent workspace environment have been developed by other investigators [1, 3, 10]. One popular method is the artificial potential field [10]. In this method, an artificial repulsive potential field is assumed between the manipulator and obstacles in the workspace. Similarly, in other methods a measure of distance from obstacles to the manipulator can be defined, and optimized to yield configurations in which the manipulator does not touch obstacles [1]. These methods have potential drawbacks. In the potential field approach, the robot can "get stuck" in a potential well, although methods to circumvent this problem have been investigated [11]. The computational burden of the optimization methods is dependent upon the degree of redundancy, and therefore prohibitive for hyper-redundant manipulators.

This paper presents a novel approach, termed "tunneling,"to the obstacle avoidance problem which is applicable to hyper-redundant manipulators. The tunneling method relies upon a novel kinematic formulation, which can be found in [4], and is briefly reviewed in Section 2. Section 3 presents the obstacle avoidance methodology. Section 4 presents an example to illustrate the technique.

2 Kinematics of Hyper-Redundant Manipulators

Hyper-redundant manipulators with constant base to end effector length are referred to throughout this paper as nonextensible manipulators. A hyper-redundant nonextensible manipulator may be comprised of many rigid links, as in Fig. 1(*a*), or the physical construction of the device may be truly continuous such as a pneumatic or tendon based structure, as in Fig. 1(*b*).

A "backbone curve" of constant length can be defined which exactly captures the continuous manipulator shape. In the case of many discrete links, the essential macroscopic features of the manipulator can be captured by a continuous backbone curve of the same length as the sum of the link lengths. As reviewed in Section 4, the obstacle avoidance computations can be performed using the continuous backbone curve, which is then used, via a "fitting" procedure, to define the joint angles of a discrete-linked manipulator. In [5, 6] the fitting procedure is described in detail, whereas the focus of this paper is the determination of appropriate backbone curves for obstacle avoidance.

2.1 Planar Hyper-Redundant Manipulator Kinematics. Figure 1(*b*) is the "backbone" curve of a nonextensible manipulator. Attach a frame defining an $x_1 - x_2$ coordinate system to the base of the backbone curve. The backbone curve is the locus of all points in the base frame which have position defined by $\bar{x}(s, t) = [x_1(s, t), x_2(s, t)]^T$. s is the backbone curve arc length, measured from the origin, and it is assumed that all lengths in the plane are scaled to units of the manipulator length so that s is in the range $0 \leq s \leq 1$.

A time varying curve with arbitrary and possibly time varying position and orientation in the plane can be described by the equation:

Contributed by the Mechanisms Committee and presented at the Design Technical Conference, Chicago, IL, Sept. 16-19, 1990 of THE AMERICAN SOCIETY OF MECHANICAL ENGINEERS. Manscript received March 1990. Associate Technical Editor: J. M. McCarthy.

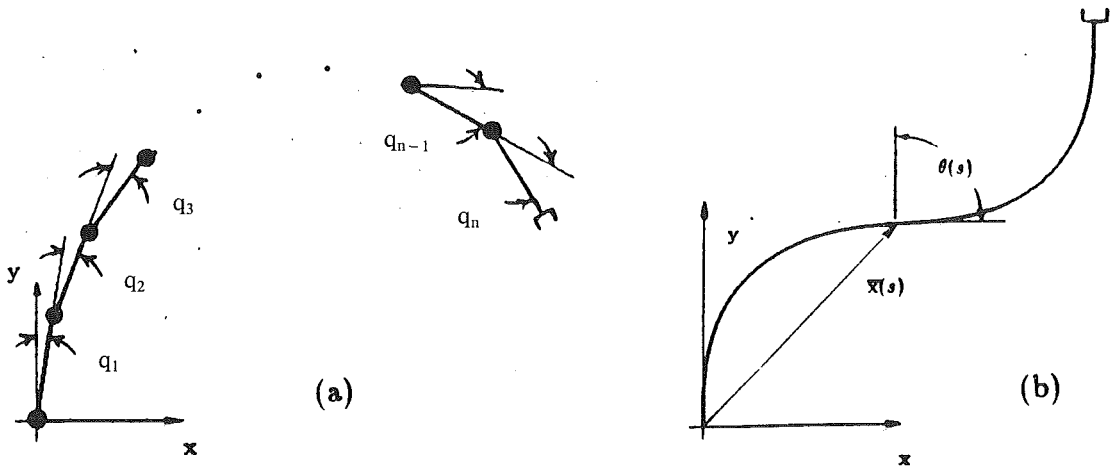

Fig. 1 Discrete and continuous hyper-redundant manipulators

$$\bar{r}(s,t) = \mathbf{Q}(t)\bar{x}(s,t) + \bar{c}(t) \tag{1}$$

$\bar{r}(s, t)$ is a vector from a fixed reference frame to a point on the backbone curve. $\bar{c}$ is a 2×1 vector, and $\mathbf{Q}$ is a 2×2 rotation matrix which respectively define the position and orientation of the backbone curve base frame in the global reference frame. They may be time dependent if the base of the manipulator is translating and rotating.

If we assume that $\bar{x}(0, t) = \bar{0}$ and that the tangent to the backbone curve at its base points in the x_2-direction, the following intrinsic equations define the shape of a planar continuous hyper-redundant manipulator in its base frame [6]:

$$x_1(s,t) = \int_0^s \sin\theta(\sigma,t)\,d\sigma$$
$$x_2(s,t) = \int_0^s \cos\theta(\sigma,t)\,d\sigma \tag{2}$$

where:

$$\theta(s,t) = \int_0^s \kappa(\sigma,t)\,d\sigma. \tag{3}$$

$\kappa(s, t)$is the *curvature function*, which is defined as the magnitude of the rate of change of the unit tangent vector at a point s along the curve: $\kappa(s, t) = |\partial^2\bar{x}/\partial s^2| = \partial\theta/\partial s$. $\theta(s_0,t)$ is the angle which the tangent to the curve at the point $s = s_0$ makes with the tangent at the point $s = 0$, i.e., the angle it makes with the x_2 axis measured in a clockwise sense.

For compactness in notation, Eq. (2) can be written in a complex notational form which is used in the remainder of this paper:

$$z(s,t) = x_1(s,t) + ix_2(s,t) = i\int_0^s e^{-i\theta(\sigma,t)}\,d\sigma. \tag{4}$$

While the forward kinematics of planar nonextensible manipulators is given by Eqs. (2) and (3), the inverse kinematics and trajectory planning can be reduced to determining the spatial and temporal behavior of the curvature function, $\kappa(s, t)$. That is, the function $\kappa(s, t)$ will determine the bending of the mechanism as a function of time, and is selected to achieve desired objectives, which depend upon the nature of the hyper-redundant manipulator application. A methodology, based on a modal expansion of $\kappa(s, t)$, to efficiently solve inverse kinematics problems is presented in [4, 6].

2.2 Spatial Hyper-Redundant Manipulator Kinematics. Three functions are required to uniquely determine the kinematics of spatial hyper-redundant manipulators. Two of these functions determine the shape of the manipulator backbone curve. These functions are denoted $\mathcal{K}(s, t)$ and $\mathcal{T}(s, t)$ and are referred to here as *quasi-curvature* and *quasi-torsion*. For comparison of these quantities to the curvature, κ, and torsion, τ defined in classical differential geometry of curves see [4]. Further defining

$$K(s,t) = \int_0^s \mathcal{K}(\sigma,t)\,d\sigma \quad T(s,t) = \int_0^s \mathcal{T}(\sigma,t)\,d\sigma \tag{5}$$

every point on the backbone curve can be described by the equations:

$$\bar{x}(s,t) = \begin{bmatrix} \int_0^s \sin K(\sigma,t)\cos T(\sigma,t)\,d\sigma \\ \int_0^s \cos K(\sigma,t)\cos T(\sigma,t)\,d\sigma \\ \int_0^s \sin T(\sigma,t)\,d\sigma \end{bmatrix}. \tag{6}$$

The same initial conditions on the position and tangent of the curve at the base of planar manipulators hold for spatial manipulators as well.

The third function $\mathcal{R}(s, t)$ is the *rate of roll distribution*, which determines how coordinate frames in planes normal to the backbone curve rotate with respect to each other [4]. The rate of roll distribution plays no role in the current obstacle avoidance formulation.

3 Overview of the Obstacle Avoidance Scheme

The method presented in this paper is applicable to time-independent workspace environments. It is assumed that the layout of obstacles in the workspace is well known, such as in an industrial setting, or assuming a sufficiently accurate vision sensing system. Executing a task in a field of obstacles is equated to defining a path around obstacles to which the manipulator must adhere. Such a path, as illustrated in Fig. (2), provides a trajectory or "tunnel" in which the nonextensible-manipulator can "slither" to circumvent the obstacles.

In practice, an automatic means for selecting one or more feasible tunnels which successfully negotiate the obstacle field could be generated using previously published methods. Free-space methods [2] based on generalized cones could be used to identify free path segments which could be assembled together to form the "tunnels" required by this method. Once the end-effector has passed through the obstacle field, the

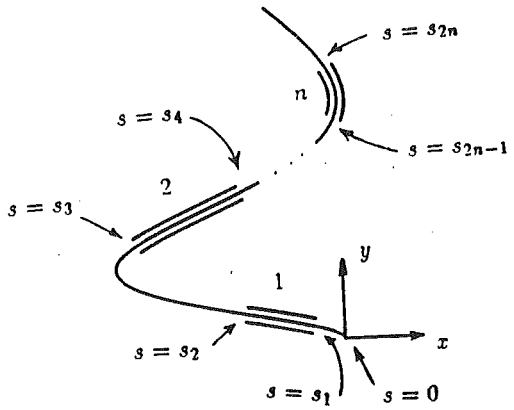

Fig. 2 Nonextensible manipulator constrained to a system of tunnels

portion of the manipulator in the obstacle field can remain stationary while the unconstrainted portion of the manipulator can perform useful work. The methods presented in [4, 6] can be used to control the geometry of the unconstrained manipulator segment outside of the obstacle field.

Two problems must be solved to implement this obstacle avoidance scheme. First, the proper geometrical constraints on the manipulator must be determined so that it statically conforms to the tunnel constraints. This is accomplished by determining the appropriate curvature functions for those sections confined to tunnels, and defining compatibility equations for the free sections. Second, the proper time rate of change of the manipulator configuration must be computed so that the moving manipulator obeys the geometric constraints of the tunnels as it moves through them. In this way the manipulator can "slither" through the tunnels from its starting configuration to its final configuration while obeying all of the geometric constraints. In this paper we do not assume or preclude any particular actuation scheme. Both position and rate information are computed, which in turn can be used to drive most conceivable actuator servo control algorithms. The actuators in turn control the manipulator shape.

3.1 Planar Manipulator with Tunnel Constraints. Figure (2) shows a hyper-redundant manipulator in which certain segments of its length are constrained to pass through tunnel segments in order to avoid obstacles. Let the segments which are constrained to fit inside a tunnel be termed *interior* segments, while the unconstrained segments are termed *exterior* segments. Number the segments sequentially starting from the base of the manipulator, assuming that the first segment is always an exterior segment. The interior and exterior segments will respectively have even and odd indices.

A curvature function which will satisfy the section by section constraints has the form:

$$\kappa(s,t) = \sum_{i=1}^{2n} \kappa_i(s,t)\, W(s,s_{i-1},s_i) \tag{7}$$

where $s_0 = 0$ and $s_i = s_i(t)$ for $i > 0$, and i indexes the manipulator segments. $\kappa_i(s, t)$ is a *local curvature function*, and W is termed a *window function*:

$$W(s,s_{i-1},s_i) = H(s-s_i) - H(s-s_{i-1}).$$

H is the Heaviside, or unit, step function. In other words, the curvature function is defined as a piecewise continuous function, where each segment of the manipulator is assigned a different curvature function to satisfy its local constraints. For interior segments the curvature function, termed an *interior* local curvature function, must take the form of a traveling wave:

$$\kappa_{2i} = \kappa_{2i}(s - s_{2i-1}(t))$$

whereas the curvature function describing exterior segments (an *exterior* local curvature function) can have the more general form:

$$\kappa_{2i+1} = \kappa_{2i+1}(s,t).$$

However, the exterior segments have kinematic restrictions on the position and orientation of the manipulator at the entrance and exit of interior segments:

$$\int_{s_{2i}}^{s_{2i+1}} \kappa_{2i+1} ds = \theta_{2i}^{2i+1} \tag{8}$$

and

$$i \int_{s_{2i}}^{s_{2i+1}} \exp\left[-i \int_{s_{2i}}^{s} \kappa_{2i+1} d\sigma \right] ds = z_{2i}^{2i+1}. \tag{9}$$

θ_{2i}^{2i+1} and z_{2i}^{2i+1} are the constant orientation and position of the frame at the entrance of the $i+1$st tunnel with respect to the frame at the exit of the ith tunnel. In this way, the manipulator backbone curve is at least a once continuously differentiable curve along its whole length.

The velocity constraints on the free sections of the manipulator are found by simply taking the time derivative of the position constraints represented by (8) and (9) to yield:

$$\dot{\theta}_{2i}^{2i+1} = \frac{d}{dt} \int_{s_{2i}}^{s_{2i+1}} \kappa_{2i+1} ds = 0 \tag{10}$$

and

$$\dot{z}_{2i}^{2i+1} = i \frac{d}{dt} \int_{s_{2i}}^{s_{2i+1}} \exp\left[-i \int_{s_{2i}}^{s} \kappa_{2i+1} d\sigma \right] ds = 0. \tag{11}$$

Since $s_k = s_k(t)$, Liebnitz's rule can be used to write the velocity constraints (10) and (11) explicitly as:

$$\int_{s_{2i}}^{s_{2i+1}} \frac{\partial \kappa_{2i+1}}{\partial t} ds + \kappa_{2i+1}(s_{2i+1},t)\, \dot{s}_{2i+1} - \kappa_{2i+1}(s_{2i},t)\, \dot{s}_{2i} = 0 \tag{12}$$

and

$$-i \int_{s_{2i}}^{s_{2i+1}} \frac{\partial \theta_{2i}}{\partial t} e^{-i\theta_{2i}} ds + e^{-i\theta_{2i}(s_{2i+1},t)}\, \dot{s}_{2i+1} - \dot{s}_{2i} = 0. \tag{13}$$

where

$$\theta_{2i}(s,t) = \int_{s_{2i}}^{s} \kappa(\sigma,t) d\sigma.$$

Note that for the parts of the manipulator which fully occupy a tunnel, $s_{2i+1} - s_{2i} = \text{const}$, which can also be written

$$\dot{s}_{2i+1} = \dot{s}_{2i}.$$

The velocity of every point along the manipulator is of the form:

$$v(s,t) = i \frac{\partial}{\partial t} \int_0^s e^{-i\theta(\sigma,t)} d\sigma = \int_0^s \frac{\partial \theta}{\partial t} e^{-i\theta} d\sigma. \tag{14}$$

3.2 Spatial Manipulator with Tunnel Constraints. Equations very similar in nature to those defined for the planar obstacle avoidance problem are now defined for the spatial case. Figure (2) is still applicable, and the same indexing system is used.

The functions $\mathcal{K}$ and $\mathcal{T}$ are taken to be of the form:

$$\mathcal{K}(s,t) = \sum_{i=1}^{2n} \mathcal{K}_i(s,t)\, W(s,s_{i-1},s_i)$$

$$\mathcal{T}(s,t) = \sum_{i=1}^{2n} \mathcal{T}_i(s,t)\, W(s,s_{i-1},s_i) \tag{15}$$

where

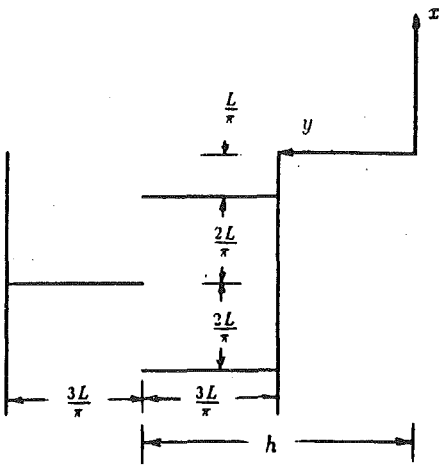

Fig. 3 Dimensions of the obstacle field

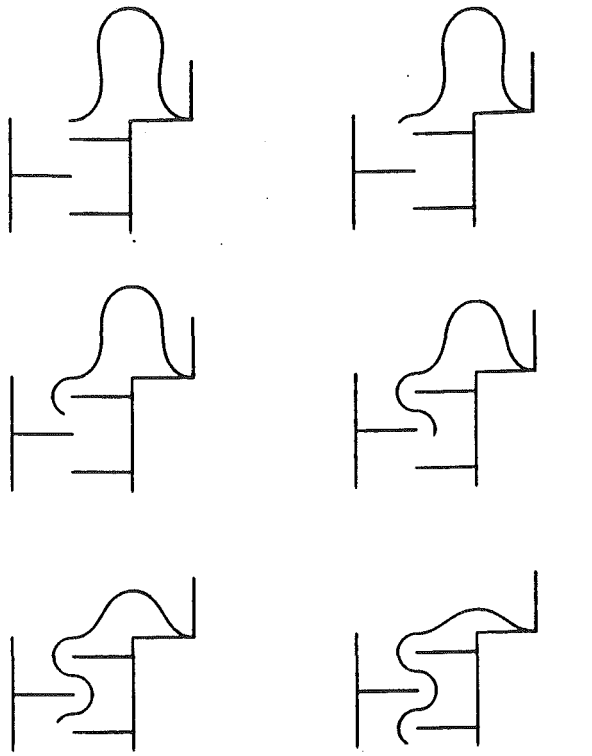
Fig. 4 Continuous hyper-redundant manipulator in an obstacle field

$$\mathcal{K}_{2i}=\mathcal{K}_{2i}(s-s_{2i-1}(t)), \quad \mathfrak{I}_{2i}=\mathfrak{I}_{2i}(s-s_{2i-1}(t))$$

are of the form of traveling waves, and

$$\mathcal{K}_{2i+1}=\mathcal{K}_{2i+1}(s,t), \quad \mathfrak{I}_{2i+1}=\mathfrak{I}_{2i+1}(s,t).$$

are of a more general form, but must obey the constraint equations

$$\int_{s_{2i}}^{s_{2i+1}} \mathcal{K}_{2i+1}ds=\text{const}_1, \quad \int_{s_{2i}}^{s_{2i+1}} \mathfrak{I}_{2i+1}ds=\text{const}_2 \qquad (16)$$

and

$$\bar{x}(s_{2i+1},t)-\bar{x}(s_{2i},t)=\bar{\mathbf{d}}_i \qquad (17)$$

where $\bar{\mathbf{d}}_i$ is the constant vector measuring the difference in position of the entrance of the i + 1st tunnel with respect to the exit of the ith tunnel as measured in the *base frame*.

Rate equations corresponding to (16) are of the same form as (12). The explicit rate form of (17) is a complicated function of the quasi-curvature and quasi-torsion, and can be written symbolically as

$$\dot{s}_{2i+1}\frac{\partial \bar{x}}{\partial s}(s_{2i+1},t)+\frac{\partial \bar{x}}{\partial t}(s_{2i+1},t)=\dot{s}_{2i}\frac{\partial \bar{x}}{\partial s}(s_{2i},t)+\frac{\partial \bar{x}}{\partial t}(s_{2i},t).$$

Taking the time derivative of Eq. (6) the velocity of every point on the manipulator can be written as

$$\bar{v}(s,t)=\begin{bmatrix} -\int_0^s \frac{\partial T}{\partial t}\sin K \sin T d\sigma+\int_0^s \frac{\partial K}{\partial t}\cos K \cos T d\sigma \\ -\int_0^s \frac{\partial T}{\partial t}\cos K \sin T d\sigma-\int_0^s \frac{\partial K}{\partial t}\sin K \cos T d\sigma \\ \int_0^s \frac{\partial T}{\partial t}\cos T d\sigma \end{bmatrix} \qquad (18)$$

4 A Specific Example

Figures 3 and 4 illustrate a specific example of the planar formulation presented in Section 3.1. The hyper-redundant manipulator must pass through a single maze-like tunnel and reach a goal on the other side. In the first unconstrained exterior section, i.e., $0 < s < s_1$, the curvature function will be assumed to have the form[1]:

$$\kappa_1(s,t)=a(t)\cos\frac{2\pi s}{s_1(t)}$$

where $a(t)$ will be determined such that the point $\bar{x}(s_1, t)$ on the moving, or slithering, manipulator is always coincident with the tunnel entrance.

Integrating the forward kinematics Eqs. (2) and (3), we get

$$x_1(s_1,t)=\int_0^{s_1}\sin\left(\frac{as_1}{2\pi}\sin\frac{2\pi s}{s_1}\right)ds=0 \qquad (19)$$

and

$$x_2(s_1,t)=\int_0^{s_1}\cos\left(\frac{as_1}{2\pi}\sin\frac{2\pi s}{s_1}\right)ds=J_0\left(\frac{as_1}{2\pi}\right)s_1. \qquad (20)$$

where J_0 is the zeroth order Bessel function.

Let $h > 0$ be the distance from the base of the manipulator to the entrance of the tunnel. h is a constant and the following must hold:

$$h=J_0\left(\frac{as_1(t)}{2\pi}\right)s_1(t).$$

For h to be positive, we restrict

$$0<\frac{as_1}{2\pi}<\mu_1$$

where $\mu_1 \approx 2.405$ is the first zero of J_0.

The condition for stationarity of $\bar{x}(s_1, t)$ at the entrance of the tunnel, while s_1 changes, is

$$a(t)=\frac{2\pi}{s_1(t)}J_0^{-1}\left(\frac{h}{s_1(t)}\right) \qquad (21)$$

and

$$\kappa_1(s,t)=\frac{2\pi}{s_1(t)}J_0^{-1}\left(\frac{h}{s_1(t)}\right)\cos\frac{2\pi s}{s_1(t)} \qquad (22)$$

where J_0^{-1} is a restricted inverse Bessel function, defined in [4].

[1]This form for the curvature functions was chosen to yield a closed form inverse kinematic solutions for the local manipulator segment. Other closed form inverse kinematic solutions can be found in [4, 6].

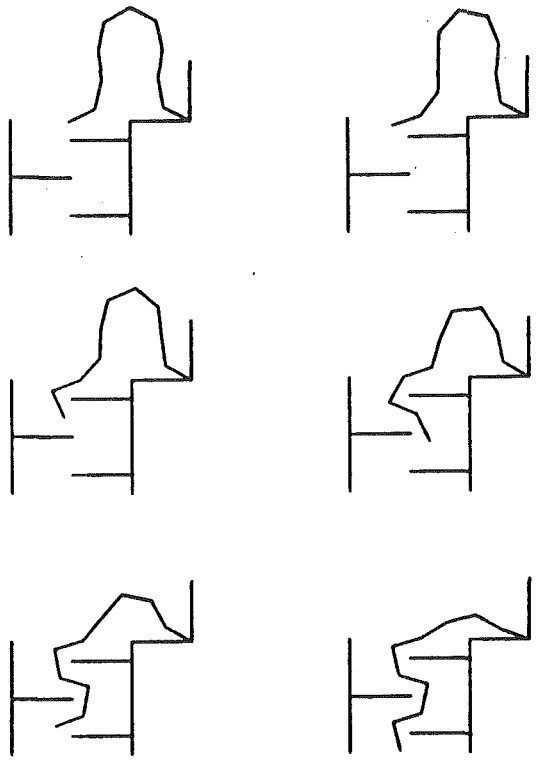

Fig. 5 Discrete hyper-redundant manipulator obstacle avoidance

The obstacle environment is illustrated in Fig. (3). With $\bar{x}(s_1, t) = [0, h]^T$ fixed, we can now determine an appropriate $\kappa_2(s, t)$ which is the curvature of the section of manipulator confined to the tunnel. For this particular obstacle field κ_2 can be defined as follows:

$$\kappa_2(s,t) = -\frac{\pi}{L}\left[W(s,s_1,s_1+L) - W(s,s_1+L,s_1+2L) + W(s,s_1+2L,s_1+3L)\right]. \quad (23)$$

This choice of κ_2 corresponds to three consecutive semi-circular arcs. The magnitude of the curvature over each of the three sections, each with arc length L, is π/L, corresponding to semicircles of radius $r = L/\pi$. The window functions take the value of unity over each of the semicircles, and the sign indicates the sense in which the arc turns. A positive sign indicates clockwise bending, and a negative sign indicates counterclockwise bending of the manipulator.

The composite curvature function for this example is then

$$\kappa(s,t) = \kappa_1(s,t)\,W(s,0,s_1) + \kappa_2(s,t)\,W(s,s_1,1) \quad (24)$$

Integrating Eq. (24) in the variable s, as in Eq. (3), we find that

$$\theta(s,t) = J_0^{-1}\left(\frac{h}{s_1}\right)\sin\left(\frac{2\pi s}{s_1}\right) - \frac{\pi}{L}(s-s_1)\,W(s,s_1,s_1+L) + \left[\frac{\pi}{L}(s-s_1-L)-\pi\right]W(s,s_1+L,s_1+2L) - \frac{\pi}{L}(s-s_1-2L)\,W(s,s_1+2L,s_1+3L) - \pi W(s,s_1+3L,1). \quad (25)$$

A time history of this tunneling obstacle avoidance maneuver is shown in Fig. 4, corresponding to the obstacle with dimensions shown in Fig. 3. The configurations shown correspond to $h = 0.4$, $L = 0.2$, and $s_1 = 1.00, 0.94, 0.83, 0.68, 0.54, 0.44$.

We now demonstrate the adaptation of the continuous manipulator analysis, via a fitting procedure, to a discrete n-link planar manipulator with n revolute joints (such as in Fig. 1(*a*)). All links are assumed to be of the same length, and the total length of the manipulator is normalized to 1. The forward kinematics for this manipulator is:

$$x_{ee} = \frac{1}{n}\sum_{i=1}^{n}\sin\left(\sum_{j=1}^{i} q_j\right) \quad (26)$$

$$y_{ee} = \frac{1}{n}\sum_{i=1}^{n}\cos\left(\sum_{j=1}^{i} q_j\right) \quad (27)$$

$$\theta_{ee} = \sum_{i=1}^{n} q_i \quad (28)$$

where x_{ee}, y_{ee}, and θ_{ee} denote the position and orientation of the end effector, and q_j denotes the angle of the jth joint.

The discrete manipulator shape and end-effector location are fitted to the continuous curve model by minimizing the sum of squared distance between points on both manipulators located at $s = \frac{i}{n}$ for $i = 1, \ldots, n$. To do this, we first assume the continuous curve solution can be used to approximately compute the discrete manipulator joint angles:

$$q_{j+1} = \theta\left(\frac{2j+1}{2n}\right) - \theta\left(\frac{2j-1}{2n}\right) + \epsilon_{j+1}. \quad (29)$$

In other words, we take the angles of the discrete case to be approximately the change in angle over a corresponding section of length in the continuous case. This approximation will lead to errors, and to account for these errors, we introduce n free "fitting" parameters, ϵ_j, which can be adjusted to minimize the error between the discrete and continuous manipulator shapes. The sum of the squared distances from the joints on the discrete manipulator to the corresponding points on the continuous manipulator is:

$$G = \frac{1}{2}\sum_{k=1}^{n}\left[\left(\int_0^{\frac{k}{n}}\sin\theta ds - \frac{1}{n}\sum_{i=1}^{k}\sin\sum_{j=1}^{i} q_j\right)^2 + \left(\int_0^{\frac{k}{n}}\cos\theta ds - \frac{1}{n}\sum_{i=1}^{k}\cos\sum_{j=1}^{i} q_j\right)^2\right]. \quad (30)$$

This function is to be minimized with respect to the $\{\epsilon_j\}$. Assuming the $\{\epsilon_j\}$ are small, (30) can be linearized to provide n linear equations in $\{\epsilon_j\}$ which can be efficiently solved (see [6] for details). If the calculated $\{\epsilon_j\}$ are small, the linearizing assumption is justified. If not, the linear approximation fitting procedure can be iterated, or a much less efficient nonlinear solution procedure must be implemented. This will generally only be the case when there are points on the backbone curve with large curvature and/or the discrete manipulator has a small number of links. Figure 5 shows an example of a 10 link manipulator which has been "fitted" to the configurations in Fig. 4.

Conclusions

This paper presented a novel obstacle avoidance concept, based on "tunneling," for hyper-redundant manipulators of constant length. A general formulation was given which allows a manipulator to maneuver through a complicated sequence of interior and exterior segments. Computer simulations were presented for a particular example. In principle, connecting circular arcs (or more complicated blending curves) and line segments, as in the example, can be used to construct a system of tunnels for maneuvering in arbitrarily complex obstacle fields. The benefit of this method over existing potential field and optimal methods for application to hyper-redundant ma-

nipulators is that a comparatively efficient set of kinematic equations based on differential geometry is computed[2], allowing much faster solutions. The method was demonstrated for a continuous curve model which can be used to represent continuous hyper-redundant manipulators or the approximate backbone curve of discrete manipulators. While the applicability of the present model diminishes with diminishing degree of redundancy, it works quite well in situations such as the continuous manipulator case, where conventional methods of analysis do not apply.

References

1 Ballieul, J., "Avoiding Obstacles and Resolving Kinematic Redundancy," *Proceedings of the 1986 IEEE International Conference on Robotics and Automation*, San Francisco, CA, 1986, pp. 1698–1704.

2 Brooks, R. A., "Solving the Find-Path Problem by Good Representation of Free Space," *IEEE Transactions on Systems, Man, and Cybernetics*, Vol. SMC-13, No. 3, March 1983, pp. 190–197.

3 Canny, J., *The Complexity of Robot Motion Planning*, The MIT Press, Cambridge, MA, 1988.

4 Chirikjian, G. S., and Burdick, J. W., "Kinematics of Hyper-Redundant Manipulators," *Proceedings of the ASME Mechanisms Conference*, Chicago, Il, Sept. 16–19, 1990.

5 Chirikjian, G. S., and Burdick, J. W., "An Obstacle Avoidance Algorithm for Hyper-Redundant Manipulators," *Proceedings of 1990 IEEE Conference on Robotics and Automation*, Cincinnati, Oh, May 14–17, 1990.

6 Chirikjian, G. S., and Burdick, J. W., "A Modal Approach to the Kinematics of Hyper-Redundant Manipulators," *Robotics and Mechanical Systems Report*. No. RMS-89-3, Dept. of Mechanical Engineering, California Institute of Technology, Pasadena, CA, 91125, September 1989.

7 Chirikjian, G. S., and Burdick, J. W., "Hyper-Redundant Robotic Locomotion: Locomotion Without Wheels Tracks, or Legs," *Robotics and Mechanical Systems Technical Report* No. RMS-89-6, Dept. of Mechanical Engineering, California Institute of Technology, March 1990.

8 Hirose, S., and Umetani, Y., "Kinematic Control of Active Cord Mechanism With Tactile Sensors," *Proceedings of Second International CISM-IFT Symposium on Theory and Practice of Robots and Manipulators*, pp. 241–252, 1976.

9 Ivanescu, M., and Badea, I., "Dynamic Control for a Tentacle Manipulator," Int. Conf. on Robotics and Factories of the Future, pp. 371–328, Dec 4–7, 1984, Charlotte, NC, USA.

10 Khatib, O., "Real-Time Obstacle Avoidance for Manipulators and Mobile Robots," *Int. J. Robotics Research*, Vol. 5, No. 1, 1986.

11 Khosla, P., and Volpe, R., "Superquadric Artificial Potentials for Obstacle Avoidance and Approach," *Proceedings of the 1988 IEEE International Conference on Robotics and Automation*, Philadelphia, PA, 1988, pp. 1778–1784.

12 Millman, R. S., and Parker, G. D., *Elements of Differential Geometry*, Prentice-Hall Inc., Englewood Cliffs, NJ, 1977.

13 Shahinpoor, M., Kalhor, H., and Jamshidi, M., "On Magnetically Activated Robotic Tensor Arms," *Proceedings of the International Symposium on Robot Manipulators: Modeling, Control, and Education*, Nov. 12–14, 1986, Albuquerque, NM.

14 Tesar, D., and Butler, M. S., "A Generalized Modular Architecture for Robot Structures," ASME *Manufacturing Review*, Vol. 2, No. 2, June 1989, pp. 91–118.

[2]The combined calculations for the execution and simulation of all the figures in Section 4 required under 3 seconds of computation on a Sun Microsystems 4/260.