12th IFToMM World Congress, Besançon, June 18-21, 2007

Redundancy resolution using a tractrix and its application to real-time simulations of hyper-redundant manipulators, snakes and tying of knots

S. Sreenivasan*
Dept. of Mechanical Engineering
Indian Institute of Science
Bangalore, INDIA

Piyush Goel†
Dept. of Mechanical Engineering
Indian Institute of Science
Bangalore, INDIA

Ashitava Ghosal‡
Dept. of Mechanical Engineering
Indian Institute of Science
Bangalore, INDIA

Abstract— *To realistically simulate the motion of flexible objects such as ropes, strings, snakes, or human hair, one strategy is to discretise the object into a large number of small rigid links connected by rotary or spherical joints. The discretised system is highly redundant and the rotations at the joints (or the motion of the other links) for a desired Cartesian motion of the end of a link cannot be solved uniquely. In this paper, we propose a novel strategy to resolve the redundancy in such hyper-redundant systems. We make use of the classical tractrix curve and its attractive features. For a desired Cartesian motion of the 'head' of a link, the 'tail' of the link is moved according to a tractrix, and recursively all links of the discretised objects are moved along different tractrix curves. We show that the use of a tractrix curve leads to a more 'natural' motion of the entire object since the motion is distributed uniformly along the entire object with the displacements tending to diminish from the 'head' to the 'tail'. We also show that the computation of the motion of the links can be done in real time since it involves evaluation of simple algebraic, trigonometric and hyperbolic functions. The strategy is illustrated by simulations of a snake, tying of knots with a rope and a solution of the inverse kinematics of a planar hyper-redundant manipulator.*

I. Introduction

In realistic computer animation of deformable objects such as snakes, ropes, trees, grass or human hair, the *length* of the object is preserved. To achieve this, one approach is to discretise the deformable object into a large number of rigid links connected by joints and then apply motion at the joints to give realistic animation. A deformable object discretised in such a way can be visualised as a hyper-redundant manipulator [1]. The main difficulty in analysis of hyper-redundant manipulators is that given a desired motion of the end-effector or a point on the manipulator, there exists an *infinite* number of solutions for the motion of the joints (or the motion of links). The problem of obtaining a unique solution is called as the *resolution* of redundancy and there exists vast amount of literature on resolution of redundancy. One of the main approaches is at the level of velocity (or infinitesimal motions) where the non-square manipulator Jacobian matrix relating the end-effector velocities to the joint velocities is inverted using a pseudo-inverse (see the review paper by Klein and Huang [2] and the textbook by Nakamura [3] and the references contained therein). The pseudo-inverse based approach has the attractive property of minimising the sum of the squares of the joint rates. The pseudo-inverse approach, with a null space term, has also been extended for obstacle and singularity avoidance, minimising joint torques and optimising additional quantities such as a manipulability index. The pseudo-inverse approach is, however, a purely numerical and local approach. Other approaches use elimination of variables [4], modeling backbone or spine curve by splines [5], solution of a partial differential equation from a continuum approach [1], equally sharing the angle [6] and workspace density for discretely actuated hyper-redundant manipulator [7]. In the recent past there has been a renewed interest in this area due to the need for more realistic computer animations and for developing real-time tools for microsurgery simulations [8]. In reference [8], the authors have developed a 'follow the leader' approach to simulate tying of knots in microsurgery where the suture or string is discretised into large number of rigid links. One end of the string is held fixed and the other end is moved in a manner to tie a desired knot. The intermediate links in the string follow the link ahead of it from which the name of the strategy is derived.

In this paper, we present a novel strategy for resolution of redundancy at the level of position. The resolution scheme is based on a classical curve called the *tractrix*. The deformable object is discretised into a large number of rigid links connected by joints. For a desired Cartesian motion of one end of link in the chain, the other end is moved according to the tractrix equation, and recursively all links are also moved along different tractrix curves. The attractive properties of a tractrix curve leads to a more realistic motion of the entire deformable object since, the motion is distributed uniformly along the entire object with the displacements tending to diminish from the 'head' to the 'tail'. We also

*The author is presently with the Geometric Software Solution Ltd., Pune, India. e-mail: sreeenivasanss@yahoo.com
†email: gpiyush@mecheng.iisc.ernet.in
‡Corresponding author. e-mail: asitava@mecheng.iisc.ernet.in

show that the computation of the motion of the links can be done in real time since it involves evaluation of simple algebraic, trigonometric and hyperbolic functions. The strategy is illustrated by simulations of the motion of a snake, tying of knots with a rope and a solution of the inverse kinematics of a planar hyper-redundant manipulator.

The paper is organised as follows: in section II we present a brief overview of the tractrix curve and extend the notion of the tractrix when the head moves along an arbitrary direction in a plane. In section III we extend the tractrix to spatial 3D motion and derive an algorithm to obtain the points on the tractrix when the end of a link moves in 3D space. In section IV, we present an algorithm based on the tractrix, to resolve redundancy in hyper-redundant systems and obtain motion of deformable objects such as snakes and ropes. In section V, we present numerical simulation results and present the conclusions in section VI.

II. An overview of the tractrix curve

The tractrix arises in the following problem posed to Leibniz: What is the path of an object starting of with a vertical offset when a string of constant length drags it along a straight horizontal line? By associating the object with a dog, the string with a leash, and the pull along a horizontal line with the dog's master, the curve has the descriptive name hund curve (hound curve) in German. Leibniz found the curve using the fact that the axis is an asymptote to the tractrix [9].

The above concept of the curve traced by the dog is also valid for a a single link moving in the plane as shown in figure 1. If the head P denoted by j_1 is made to move along a straight line ST parallel to the X-axis, the motion of the tail denoted by j_0, such that the velocity of j_0 is along the link, is a tractrix shown by the dotted curve in figure 1. Using the fact that the velocity vector at j_0 is always aligned

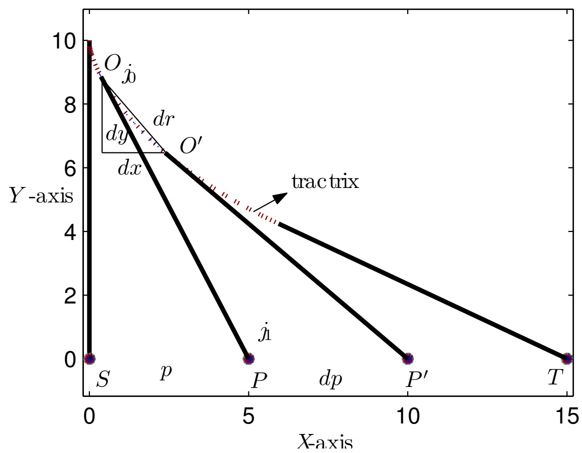

Fig. 1. Motion of a link when one end is pulled along the line ST parallel to X-axis

with the link, i.e., with the tangent to the tractrix, the tractrix equation can be derived from the differential equation of the tangent

$$\frac{dy}{dx} = \frac{-y}{\sqrt{L^2 - y^2}} \tag{1}$$

where L is length of the link. The above differential equation can be solved in closed from, and we can write

$$x = L \log \frac{y}{L - \sqrt{L^2 - y^2}} - \sqrt{L^2 - y^2} \tag{2}$$

The solution of the differential equation can also be written in a parametric form, with p as the parameter, as

$$x(p) = p - L \tanh(\frac{p}{L}), \quad y(p) = L \operatorname{sech}(\frac{p}{L}) \tag{3}$$

Since the instantaneous motion of the tail j_0 is directed along the link, the following optimality property holds: given an infinitesimal displacement dp of j_1 along ST, the length of the path traversed by the tail j_0, denoted by a vector dr, presents a local minimum of all possible paths for j_0. Furthermore, the ratio between dr and dp obeys an inequality $dr \leq dp$. The inequality follows from the following reasoning: Let x and y be the coordinates of point j_0, and p is the x-coordinate of j_1. From figure 1 we get

$$p = x + \sqrt{L^2 - y^2} \tag{4}$$

The displacement dr can be written as $dr = \sqrt{dx^2 + dy^2}$, and using elementary calculus and the tractrix equation, we get

$$\frac{dr}{dx} = \frac{L}{\sqrt{L^2 - y^2}} \tag{5}$$

and dr/dp can be obtained as

$$\frac{dr}{dp} = \frac{\sqrt{L^2 - y^2}}{L} \leq 1 \tag{6}$$

where we get an equality if the link is along the line ST.

We next consider the case of the head moving along an arbitrary straight line (not necessarily the X axis) given by $y_e = mx_e$ where $m = y_p/x_p$ is the slope of the line connecting the initial position of the head and the destination point of the head (x_p, y_p). The differential equation for the tangent can now be written as

$$\frac{dy}{dx} = \frac{y - y_e}{x - x_e} \tag{7}$$

From the length constraint, $L^2 = (x - x_e)^2 + (y - y_e)^2$, we can solve for x_e and we get,

$$x_e = \frac{-B \pm \sqrt{B^2 - 4AC}}{2A} \tag{8}$$

where $A = 1 + m^2, B = 2my + 2x, C = x^2 + y^2 - L^2$. From the above expression for x_e, we can see that there are two values possible of x_e for every x and y. The positive sign is used when the slope of the link (m_1), with respect to a new coordinate system with the path of the head as the X-axis is negative and vice versa. Substituting the expressions obtained for x_e from equation (8) and $y_e = mx_e$ in equation (7), and integrating it we get the tractrix shown in figure 2.

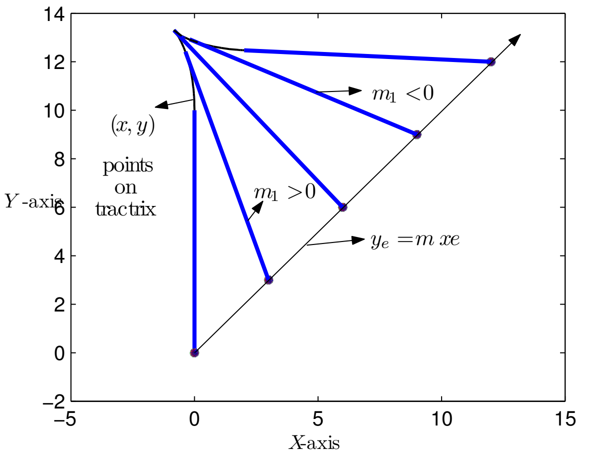

Fig. 2. Motion of a link when one end is pulled along the line $y_e = mx_e$

III. Extension of the tractrix to spatial motion

The equations describing a tractrix can be extended to 3D space. In 3D space we will have two differential equations of the form

$$\frac{dy}{dx} = \frac{y - y_e}{x - x_e}, \quad \frac{dz}{dx} = \frac{z - z_e}{x - x_e} \tag{9}$$

The equations of the path followed by head are

$$y_e = m_1 x_e, \quad z_e = m_2 x_e \tag{10}$$

where, $m_1 = y_p/x_p$, $m_2 = z_p/x_p$, and (x_p, y_p, z_p) is the destination point of the head. It may be noted that the above equations assumes that the link is initially lying along Y-axis; however, similar equations can be obtained if the link is along the Z or the X axis. We also have the constraint of length preservation

$$L^2 = (x - x_e)^2 + (y - y_e)^2 + (z - z_e)^2 \tag{11}$$

Similar to the planar case, one can numerically integrate the above differential equations and obtain the path taken by the tail in 3D space.

Instead of the numerical solution of the differential equations, we can also find the point on the tractrix in 3D space in closed form. For this purpose a reference plane is constructed using the three points, namely the initial positions of head, tail and the destination point of the head denoted by $\mathbf{X}_p = (x_p, y_p, z_p)^T$. The X-axis of the reference plane is aligned with the path of head. The parametric equation of tractrix is then solved to get the position of the tail (x_r, y_r) in the reference plane. Finally, to obtain the position of the tail in global co-ordinates, the points (x_r, y_r) are transformed from the reference (local) to the global co-ordinate system.

A. Algorithm to obtain the tractrix curve in 3D space

1 Define the vector $\mathbf{S} = \mathbf{X}_p - \mathbf{X}_h$ where $\mathbf{X}_h$ is the current location of the head.
2 Define the vector $\mathbf{T} = \mathbf{X} - \mathbf{X}_h$ where $\mathbf{X} = (x, y, z)^T$ is the tail of the link lying on the tractrix.
3 Define the new reference coordinate system $\{r\}$ with the X-axis along $\mathbf{S}$. Hence $\hat{\mathbf{X}}_r = \frac{\mathbf{S}}{|\mathbf{S}|}$.
4 Define the Z-axis as $\hat{\mathbf{Z}}_r = \frac{\mathbf{S}\times\mathbf{T}}{|\mathbf{S}\times\mathbf{T}|}$.
5 Define rotation matrix ${}^0_r[\,R\,] = [\hat{\mathbf{X}}_r \quad \hat{\mathbf{Z}}_r \times \hat{\mathbf{X}}_r \quad \hat{\mathbf{Z}}_r]$.
6 The Y-coordinate of the tail (lying on the tractrix) is given by $y = \hat{\mathbf{Y}}_r \cdot \mathbf{T}$ and the parameter p can be obtained as $p = L\,\mathrm{sech}^{-1}(\frac{y}{L}) \pm |\mathbf{S}|$.
7 From p, we can obtain the X and Y coordinate of the point on the tractrix in the reference coordinate system as

$$\begin{aligned} x_r &= \pm|\mathbf{S}| - L\tanh(\frac{p}{L}) \\ y_r &= L\,\mathrm{sech}(\frac{p}{L}) \end{aligned} \tag{12}$$

8 Once x_r and y_r are known, the point on the tractrix $(x, y, z)^T$ in the global fixed coordinate system $\{0\}$ is given by

$$(x, y, z)^T = \mathbf{X}_h +^0_r [\,R\,](x_r, y_r, 0)^T \tag{13}$$

The above steps are illustrated in figure 3. It may be noted that the point on the tractrix is obtained in terms of cross and dot products of vectors and hyperbolic functions, and do not require solution of differential equations. This makes the algorithm amenable for real time computations.

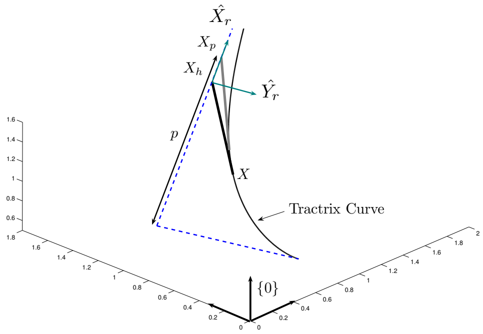

Fig. 3. Tractrix in global and local coordinates

IV. Algorithm for resolution of redundancy

The basic equations of a tractrix in 3D space can be used for resolution of redundancy for any serial discretised deformabale object. Consider a deformable object such as a snake or a rope discretised into n rigid links $l_1, l_2, ..., l_n$ with joints $j_1, j_2, ..., j_{n-1}$. For spatial motion, we assume that the joints are spherical joints. Consider the last two links l_n and l_{n-1}. The head of the link l_n denoted by the point j_n is moved to a new position $j_{n_{\mathrm{new}}}$ given by $(x_p, y_p, z_p)^T$. From the steps given in the previous section, we can obtain the displaced location of the tail $(x, y, z)^T$ as it follows a tractrix (see equation (13). The link l_{n-1} is attached to the link l_n and hence the head of link l_{n-1} should now be moved from the existing location to $(x, y, z)^T$. The location of the tail of link l_{n-1} (following a tractrix) can again be obtained from the steps given in previous section. It maybe noted that the reference plane and the rotation matrix obtained in the steps described in section III are *not* the same for the two links. Following similar steps, we recursively obtain the motion of the head and tail of all links down to the first link l_1.

We can make the following remarks about the above algorithm.

- Instead of the flexible object being moved from the end, if it is moved from any point on the body, then we can divide the object into two parts and apply the steps listed in section III to the two parts individually.
- In case of a hyper-redundant robot, the joint angles can be obtained by simple vector algebra since the initial and final position of all the links are known.
- Under a tractrix motion, when the head of the link l_n moves by dr_n the displacements of all the links obey the inequality $dr_0 \leq dr_1 \leq ... \leq dr_{n-1} \leq dr_n$, with the equality $dr_i = dr_{i-1}$ reached only when the line of motion of joint j_i coincides with link l_i. This observation follows from equation (6). A consequence of this observation is that the motion of the links away from the head gets progressively smaller and appears to 'die' out. This feature gives a more realistic visualization of the motion of the flexible object and was observed in animations of a snake and knot tying.
- The above property also imply that for a tractrix motion, the total displacement of all links except the head, $\sum_{i=0}^{i=n-1} |dr_i|$, is minimised. Note that this is different from a pseudo-inverse based resolution of redundancy where the infinitesimal motion of the joints are minimised in the *least square* sense. The results obtained from the tractrix based resolution of redundancy are thus different from pseudo-inverse based methods.

A. Obstacle avoidance

While moving the links of the deformable object such as in tying a knot, we must ensure that the discretised links do not intersect or collide with each other. To effectively simulate the motion of the discretised links, we need to detect collisions between the links and then develop a strategy to manage the collisions. There exists a wide variety of algorithms and literature for collision detection (see, for example, the review paper [10] and the references contained there in). In our implementation, we have used a simple conservative strategy of bounding each link by a sphere and then checking for the distance between the centre of the spheres. If the distance is such that there is collision, we move the centre of one of the links along the common normal by a distance slightly greater that $(l_i + l_{i+1})/2$ where l_i is the length of the i^{th} link. Unfortunately this motion can result in a collision at some other link. Hence, we must recursively follow a strategy of detection and collision management for all links. If the number of iterations exceeds a given value the object is considered to be locked at that configuration.

V. Simulation results

The tractrix based redundancy resolution has been applied to visualize the motion of a snake, tying a knot in a rope, and for solving the inverse kinematics of hyper-redundant planar manipulator. We present snap shots of animation results for each of these objects.

Motion of a snake in 3D: We model a snake with 40 links. The head of the snake is moved along an arbitrary curve and the motion of each of the links are obtained according to the tractrix strategy. From the motion of the links, an animation is created using Matlab. Figure 4 shows the configuration of the snake at few instances.

A single-handed knot: We model a rope with 40 rigid links each one unit long. One end of the rope is moved in a fashion so that a knot is tied near the centre of the rope. Several configurations of the rope while tying the knot are shown in figure 5. An animation of the knot tying process was created in Matlab from the computed configurations and since the computations are fairly simple, this could be done in real time.

Two-handed knot: In case the knot is to be tied by moving both the ends of the rope (by moving both hands) then the two ends are moved alternatively. Again, the motion of all the links are obtained in real-time by using the tractrix based approach. Various configurations of the rope while two-handed knot tying are shown in figure 6. It was observed that the collision avoidance takes most of the time in the simulation of single-handed and two-handed knots.

Planar hyper-redundant manipulator: In the case of inverse kinematics of a hyper-redundant manipulator, the desired (x_p, y_p) of the end-effector (same as the head) is specified. We obtain the motion of all the links by the tractrix approach. In a tractrix the tail of the first link also moves although by a small amount. However, for a manipulator the position j_0 should be fixed. To overcome this problem, we move the point j_0 to $(0, 0)$ and all other links are translated rigidly (no joints are rotated). This results in the end-effector moving away from the desired (x_p, y_p). We repeat the tractrix based approach till the head reaches (x_p, y_p) within a prescribed error bound. It may be noted that convergence is guaranteed since the tractrix motion dies down along the links (see observation listed in the previous section). Figure 7 gives intermediate configurations of a 10 link hyper-redundant planar manipulator where each link length is 0.2 units. The initial configuration and the final (desired) end-effector (x, y) coordinates are as marked in figure 7. The intermediate configuration with blue markers are obtained from the tractrix based approach and the configuration with black markers are from a pseudo-inverse formulation. The plot of joint rotations from the tractrix and pseudo-inverse based approaches are shown in figure 8.

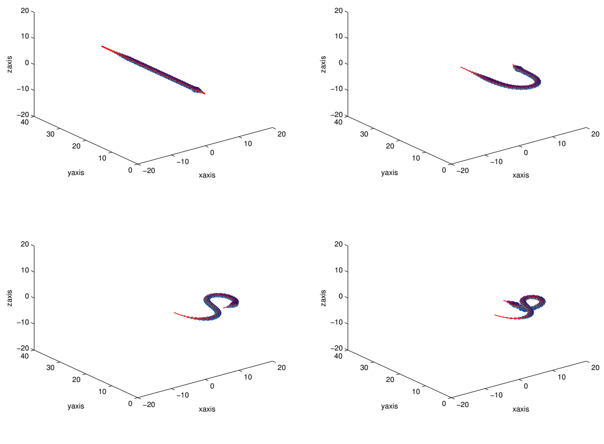

Fig. 4. Simulation of the motion of snake

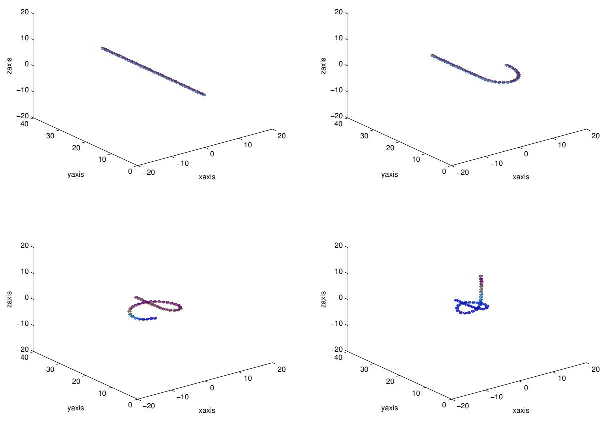

Fig. 5. Simulation of the single-handed knot tying

VI. Conclusion

In this paper, we have presented a novel tractrix based scheme for resolution of redundancy for hyper-redundant manipulators and simulation and visualisation of motion of flexible objects such as snakes and ropes. The flexible object is discretised into a large number of rigid links and for an arbitrary chosen motion of the head of one link, the motion of all other links are computed by using the equations

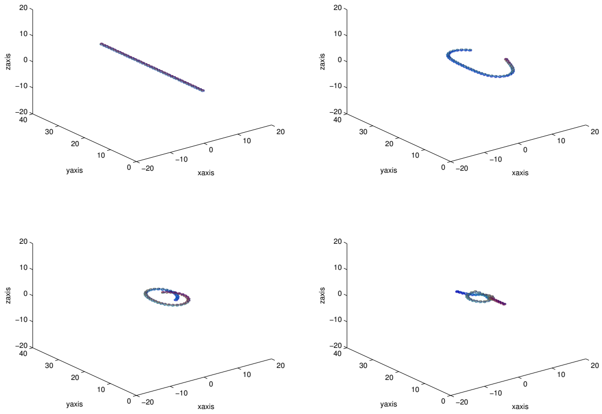

Fig. 6. Simulation of a two-handed knot tying

of a tractrix. One of the key property of a tractrix is that the motions of links decreases as one goes away from the head and this makes the visualization of tying knots and motion of a snake more realistic. In addition, since the computations involve simple vector algebra and evaluation of hyperbolic functions, the simulation and visualization can be done in real-time.

References

[1] Chirikjian, G. S., Burdick, J. W. (1994), 'A modal approach to hyper-redundant manipulator kinematics', *IEEE Trans. on Robotics and Automation*,**10**(3), pp. 343-354.

[2] Klein, C. A. and Huang, C. H. (1983), 'Review of the pseudo-inverse for control of kinematically redundant manipulators', *IEEE Trans. on Systems, Man, and Cybernetics*, **SMC-13**(3), pp. 245-250 .

[3] Nakamura, Y. (1991), *Advanced Robotics: Redundancy and Optimization*, Addison-Wesley.

[4] Ghosal, A. and Roth, B. (1988), 'A new approach for kinematic resolution of redundancy', *The Int. Journal of Robotics Research*, **7**, pp. 22-35.

[5] Zanganeh, K. E. and Angeles, J. (1995), 'The inverse kinematics of hyper-redundant manipulators using splines', *Proc. IEEE Int. Conf. on Robotics and Automation*, **3**, pp. 2797-2802.

[6] Kimura, S., Takahashi, M., Okuyama, T., Tsuchiya, S. and Suzuki, Y. (1998), 'A fault-tolerant control algorithm having a decentralized autonomous architecture for space hyper-redundant manipulators', *IEEE Trans. on Systems, Man, and Cybernetics*, **28**(4), pp. 521-528.

[7] Wang, Y. and Chirikjian, G. S. (2004), 'Workspace generation of hyper-redundant manipulators as a diffusion process on $SE(N)$', *IEEE Trans. on Robotics and Automation*,**20**(3), pp.399-408.

[8] Brown, J., Sorkin, S., Latombe, J. C., Montgomery, K. and Stephanides, M. (2002) 'Algorithmic tools for real-time microsurgery simulation', *Medical Image Analysis*, **6**(3), pp. 289-300.

[9] http://mathworld.wolfram.com/Tractrix.html

[10] Jimenez, P., Thomas, F. and Torras, C., (2001) '3D collision detection: A survey', *Computers and Graphics*, **25**(2), pp. 269-285

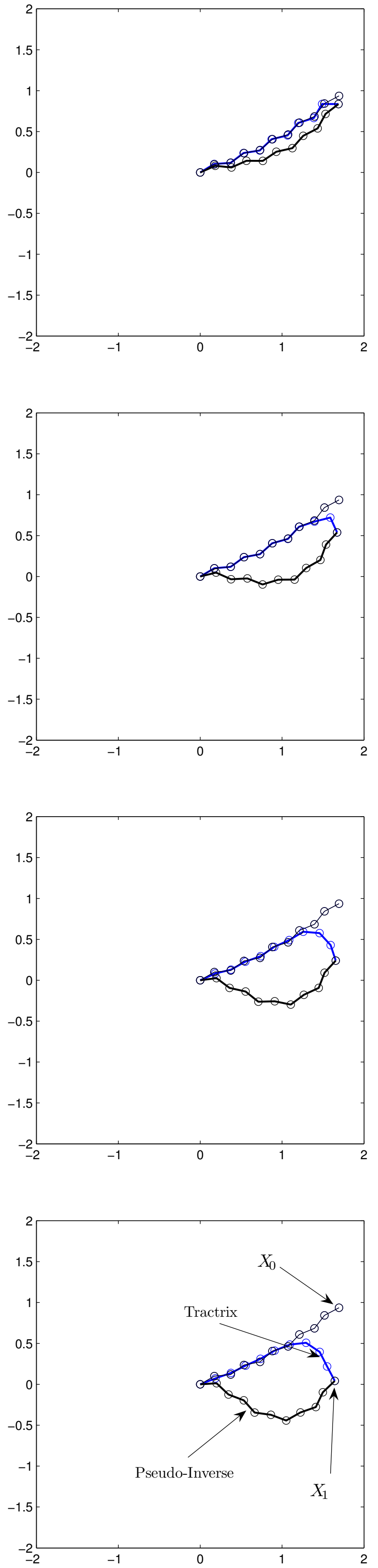

Fig. 7. Intermediate configurations of the planar hyper-redundant manipulator

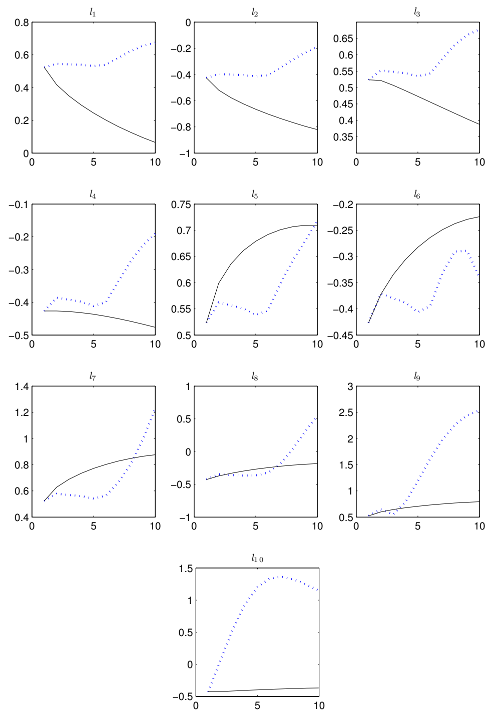

Fig. 8. Joint rotations in the planar hyper-redundant manipulator: dotted blue tractrix, continuous black pseudo-inverse