Proceedings of 2012 IEEE
International Conference on Mechatronics and Automation
August 5 - 8, Chengdu, China

Path Planning for Multisection Continuum Arms

Isuru S. Godage, David T. Branson, Emanuele Guglielmino, and Darwin G. Caldwell
Department of Advanced Robotics, Istituto Italiano di Tecnologia,
Via Morego 30, 16163 Genova, Italy.
{*isuru.godage; david.branson; emanuele.guglielmino; darwin.caldwell*}*@iit.it*

***Abstract*—Continuum arms have interesting features useful for navigation in unstructured environments such as minimally invasive surgeries and inspection tasks. However, planning motions to avoid obstacles for these arms is challenging due to their complex kinematics. In this paper a path planning and obstacle avoidance algorithm for multisection continuum arms in dynamic environments is presented. This work is potentially applicable to surgical procedures to navigate near vital organs and reach surgical targets without injuring surrounding tissues. On the macro scale it can snake a continuum arm through constrained spaces such as inside of tubes for search and rescue purposes. Simulation results are presented for obstacle avoidance in static and dynamic environments. The algorithm utilizes a mode shape function based kinematic model of continuum arms and yields accurate solutions efficiently. This approach can be easily extended for other configurations of continuum arms.**

***Index Terms*—continuum arms, path planning, obstacle avoidance.**

I. Introduction

Inspired by biological appendages such as octopus arms, squid tentacles, lizard tongues and elephant trunks, continuum robotic arms exhibit many interesting features that their rigid counter parts do not. They can bend in all directions to attain complex shapes and maneuvers, extend their length to achieve increased actuation space, and perform whole arm manipulation [1]. There are many robotic prototypes that have been proposed over the years such as [2], [3], [4], [5]. Among these variable length continuum arms, due to their scalability, have potential applications in many different areas from minimally invasive surgeries to search and rescue operations.

Simaan *et al.* proposed a miniature compliant implementation of a cable driven continuum arm as a Laryngeal surgical tool with omni-directional bending [6]. It was designed so that 2 to 3 such arms with high tip dexterity would enable suturing and soft-tissue handling with a single entry via a patients mouth. Other similar surgical robots proposed such as [7], [8], [9] require navigation through constrained spaces to reach surgical targets. Any undesirable contact between surgical tools and delicate organs during operations can have severe consequences. Therefore, accurate path planning is essential for these tools.

On the other hand, continuum arms reported in [3], [4], [10] can be used as a compliant manipulator for different applications such as object manipulation and inspection operations in unstructured environments. In order to achieve this the arm position needs to be carefully planned to avoid damage to the robotic arm when operating in harmful and congested environments such as sharp debris after disasters.

Work has been done to devise obstacle avoidance mechanism for redundant robots. Chirikjian *et al.* proposed an obstacle avoidance algorithm that uses primitive geometrical shapes to calculate planar robot poses [11]. However, these results are presented for a static environment only and due to the few geometrical shapes used to compute robot pose, very few robots fit the given criterion. Lyons *et al.* presented a path planning algorithm for a surgical robot [12]. It uses a nonlinear optimization technique to find suitable configurations out of many random starting positions. It too was evaluated for static environments where all obstacles are mathematically defined. However, real world applications do not always have the obstacle information to practically apply this method. Jing *et al.* proposed an adaptive algorithm for real-time obstacle avoidance for the OctArm robot [13]. This algorithm uses a planar curve parametric kinematic model to yield solutions [14]. However, the use of curve parameters have limitations as they do not accurately represent the actual kinematics of the robot, and they prevent the modeling of straight arm poses [5]. Thus there is the need for a general obstacle avoiding algorithm for continuum arms that is spatial and accurately represent the robot kinematics and works in dynamic environments.

In this paper an obstacle avoidance algorithm is presented for multisection continuum robotic arms in dynamic environments. The algorithm presented by Maciejewski *et al.* [15], originally developed for rigid link robots, was modified and customized to incorporate multisection continuum arms. The modifications consider a finite number of proximity sensors mounted on discreet locations on the arm that only provide distal information. It makes use of the redundancy of manipulators to achieve secondary objectives, in this case obstacle avoidance. This general approach can be extended to any continuum arm configurations. Numerically stable continuum arm kinematics originally proposed by Godage *et al.* [10] was used for simulation purposes and therefore yield fast results.

The organization of this paper is as follows. Section II presents related work in the area of continuum arms and obstacle avoidance. The proposed method is detailed in section III followed by simulation results in section IV. A discussion of simulation results and key points of this work is given in section V prior to the conclusion in section VI.

978-1-4673-1278-3/12/$31.00 ©2012 IEEE

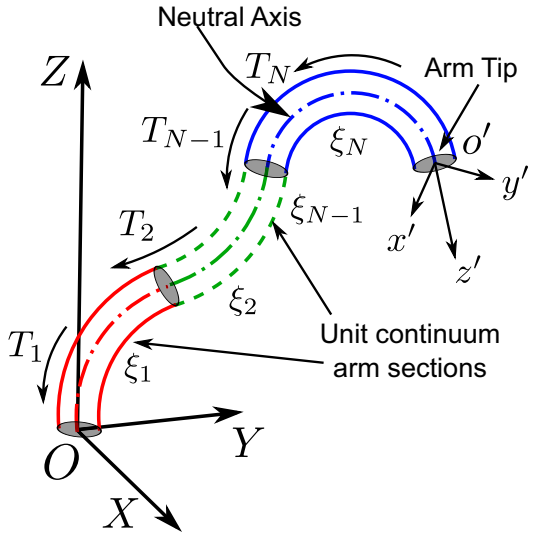

Fig. 1. Schematic of multisection continuum arm with arbitray N number of sections

II. Preliminaries

Related work to continuum arm kinematics and obstacle avoidance is presented in this section.

A. Continuum Arm Kinematic Model

This section provides a brief overview of the modal kinematic model presented for multisection continuum arms in [10]. As shown by the schematic in Fig. 1 a variable length continuum arm is constructed using many unit continuum sections. Each unit consist of three variable length actuators with original length L_0. Actuator length at any time is given by $L_0 + l_{ij}$ where l_{ij} is the j^{th} actuator length variable of the i^{th} section. Continuum arms have geometrically coupled complex kinematics contributed by its continuum sections. Any i^{th} continuum section's spatial shape can be described by three curve parameters that are functions of the section length variables; curvature radius, λ_i , angle subtended by the bending arc, ϕ_i, and the lateral angle of bending plane with respect to the $+X$ axis, θ_i. It can bend in a circular arc or purely extend with its actuation space. The modal transformation matrix (MTM) for this continuum section is given by

$$ {}^{i}_{i-1}\mathbf{T}_{\Phi}\left(\xi_i, \boldsymbol{q}_i\right) = \begin{bmatrix} \boldsymbol{\Phi}_R\left(\xi_i, \boldsymbol{q}_i\right) & \boldsymbol{\phi}_p\left(\xi_i, \boldsymbol{q}_i\right) \\ \mathbf{0}_{1\times 3} & 1 \end{bmatrix} \tag{1} $$

where Φ_R and ϕ_p stand for arm orientation and position of a point given by $\xi_i \in [0, 1]$ which is a scalar utilized to represent the points along the neutral axis from the section base ($\xi = 0$) to the top ($\xi = 1$). The joint space vector for the continuum section $\boldsymbol{q}_i = \begin{bmatrix} l_{i1} & l_{i1} & l_{i1} \end{bmatrix}^T$ stands for actuator length variables [10].

Continuum arms in a fixed base have limited potentials and are usually mounted on moving platforms [4]. Therefore we introduce an arbitrary floating base coordinate frame of joint variables, $\boldsymbol{q}_b \in \mathbb{R}^b$ where $b \leq 6$.

Using classical kinematic transformations, results for a single section can be extended to incorporate arbitrary N number of sections with resulting complete MTM given by

$$ \begin{aligned} {}^{N}_{0}\mathbf{T}_{\Phi}\left(\boldsymbol{\xi}, \boldsymbol{q}\right) &= \mathbf{T}_b \prod_{k=1}^{N} \left\{ {}^{k}_{k-1}\mathbf{T}_{\Phi}\left(\xi_k, \boldsymbol{q}_k\right) \mathbf{T}_k \right\} \\ &= \begin{bmatrix} \mathbf{R}\left(\boldsymbol{\xi}, \boldsymbol{q}\right) & \boldsymbol{p}\left(\boldsymbol{\xi}, \boldsymbol{q}\right) \\ \mathbf{0}_{1\times 3} & 1 \end{bmatrix} \end{aligned} \tag{2} $$

where $\mathbf{T}_b$ is the homogeneous floating base transformation matrix, $\mathbf{R}$ and $\boldsymbol{p}$ are modal orientation and position matrices respectively, and $\boldsymbol{q} = \{[\boldsymbol{q}_b^T \;\; \boldsymbol{q}_1^T \;\; \boldsymbol{q}_2^T \;\; \cdots \;\; \boldsymbol{q}_N^T \;]^T : \boldsymbol{q} \in \mathbb{R}^{3N+b}\}$ is the composite joint space vector and $\mathbf{T}_k$ accounts for section joint offsets. The scalars for individual sections constitute the scalar coefficient vector, $\boldsymbol{\xi} = \{[\xi_1 \;\; \xi_2 \;\; \cdots \;\; \xi_N \;]^T : \boldsymbol{\xi} \in \mathbb{R}^{N\times 1}\}$. This is then evaluated as $\boldsymbol{\xi} = \{\xi_r = 1 : \forall r < i \,,\, \xi_i \,,\, \xi_r = 0 : \forall r > i\}$ when moving from the arm base to the tip. The linear velocity of any points along the neutral axis are given by (3) where $\mathbf{J} = \frac{\partial \boldsymbol{p}(\boldsymbol{\xi}, \boldsymbol{q})}{\partial \boldsymbol{q}_j}$ is the modal Jacobian.

$$ \dot{\boldsymbol{p}}\left(\boldsymbol{\xi}, \boldsymbol{q}, \dot{\boldsymbol{q}}\right) = \mathbf{J}\left(\boldsymbol{\xi}, \boldsymbol{q}\right) \dot{\boldsymbol{q}} \tag{3} $$

B. Obstacle Avoidance Algorithm

There are many obstacle avoidance algorithms that have been proposed for rigid redundant manipulators with discreet joints. In this work, the method proposed by Maciejewski *et al.* [15] is modified and extended considering its practical implementation.

The general inverse kinematic solution for a manipulator end effector velocity is given by

$$ \dot{\boldsymbol{q}} = \mathbf{J}^{\dagger}\dot{\boldsymbol{p}} + \left(\mathbf{I} - \mathbf{J}^{\dagger}\mathbf{J}\right)\boldsymbol{\mu} \tag{4} $$

where $\mathbf{J}^{\dagger}$ is the pseudo inverse of $\mathbf{J}$, $\mathbf{I}$ is an identity matrix, and $\boldsymbol{\mu}$ is an arbitrary vector in $\dot{\boldsymbol{q}}$ space. The null space projection operator, $\left(\mathbf{I} - \mathbf{J}^{\dagger}\mathbf{J}\right)$, describes the redundancy of the system. This redundancy can be used for secondary goals or arm reconfiguration without affecting the primary goal. The algorithm is based on assigning an attractive velocity at the end effector towards the goal (primary goal) and repelling velocities at points along the manipulator that are too close to obstacles (secondary goals). Then the system is iteratively solved for the joint space velocity until the end effector reaches its target. This will be the essence for continuum arm obstacle avoidance path planning in the following section.

III. Methodology

The theoretical treatment of the obstacle avoidance algorithm for continuum arm is first presented in this section, and then practical constraints are imposed for application purposes.

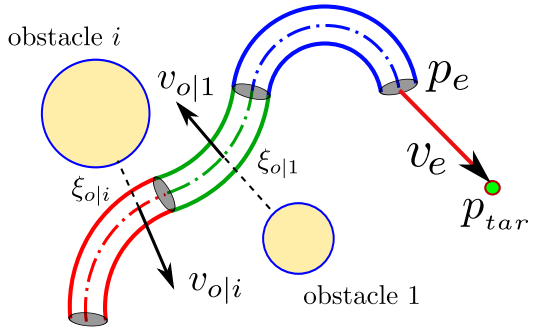

Fig. 2. Closest distances from the continuum arm to obstacles and repelling velocities at those points

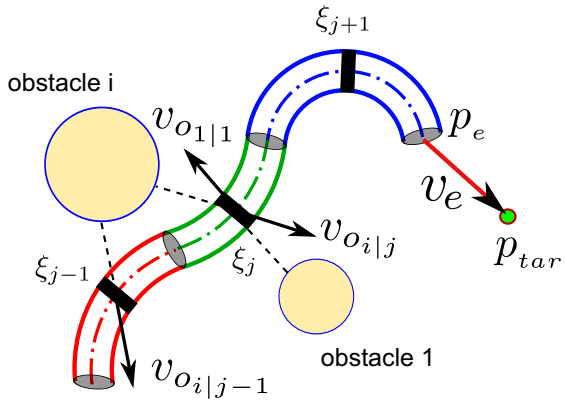

Fig. 3. Continuum arm with discreet sensor locations showing limited sensor region at each sensor band. The repel velocities are now added normal to the neutral axis away from obstacles.

A. Theory

This section formulates the obstacle avoidance problem for a general multisection continuum arm with N sections placed in an environment with M number of obstacles at defined locations $\boldsymbol{x}_k \in \mathbb{R}^3$ where $k \in [0, M]$. The theoretical approach presented in [15] is then modified to suit the deformable continuum arm under consideration.

In the development of the algorithm the following assumptions are made. An ideal sensor field around the arm is assumed capable of sensing spatial coordinates of the closest points to obstacles (Fig. 2). The continuum arm is kinematically described without dynamic properties such as compliance and therefore has uniform bending.

Let the target point to which the continuum arm tip should reach be $\boldsymbol{p}_{tar}$, and the continuum arm tip position be $\boldsymbol{p}_e$. The primary objective is to navigate $\boldsymbol{p}_e$ through a dynamic obstructive environment while avoiding contact with objects. Following a similar approach to that of [15], this is achieved by having tip a velocity towards $\boldsymbol{p}_{tar}$ and assigning repelling velocities at points on the continuum arm that are the closest within a predefined repelling region. These objectives, according to their priority, are mathematically defined as

$$\dot{\boldsymbol{p}}_e = \mathbf{J}_e \dot{\boldsymbol{q}} \tag{5a}$$

$$\dot{\boldsymbol{p}}_k = \mathbf{J}_k \dot{\boldsymbol{q}} \tag{5b}$$

where $\dot{\boldsymbol{p}}_e$ is the arm tip velocity, $\dot{\boldsymbol{p}}_k = \dot{\boldsymbol{p}}(\boldsymbol{\xi}_k, \boldsymbol{q})$ is the repelling velocity incident by the k^{th} object at ξ_k within the repelling region, and $\mathbf{J}_e$ and $\mathbf{J}_k$ are modal Jacobians at respective velocity points. The tip velocity can be calculated by utilizing the error between current and desired end effector location as $\dot{\boldsymbol{p}}_e = \eta(\boldsymbol{p}_{tar} - \boldsymbol{p}_e)$ where $\eta \in \mathbb{R}$ is the gain coefficient.

By incorporating the task priorities the simplified solution to (5a) and (5b) is given by

$$\dot{\boldsymbol{q}} = \mathbf{J}_e^{\dagger}\dot{\boldsymbol{p}}_e + \sum_{k=1}^{M} \alpha_{n|k} \left[\mathbf{J}_k \left(\mathbf{I} - \mathbf{J}_e^{\dagger}\mathbf{J}_e\right)\right]^{\dagger} \left(\alpha_{0|k}\dot{\boldsymbol{p}}_k - \mathbf{J}_k\mathbf{J}_e^{\dagger}\dot{\boldsymbol{p}}_e\right) \tag{6}$$

where the summation indicates the cumulative repelling velocities from all obstacles within the threshold region. To effectively smoothen the degree of influence that an obstacle has on the arm motion, scaler multipliers $\alpha_{n|k} = \alpha_n(d_k)$ and $\alpha_{o|k} = \alpha_0(d_k)$ that are functions of the closest distance to the k^{th} obstacle (d_k) are introduced [15]. The system given by (6) is then iteratively solved until the tip reaches the desired location.

B. Modifications for Practical Application

In the theory developed earlier a continuum sensory field around the arm is assumed. In an environment where all obstacle locations are known, the closest points to the arm can be calculated by evaluating the distance to obstacles with respect to ξ and choosing the closest. But in reality only a limited number of sensors can be mounted at discreet locations along the arm. Also, they have a limited directional range (i.e., ultrasound proximity sensors) and are only able to provide distal information without orientation. This presents problems regarding assigning direction of repelling velocities. Therefore, to account for a more practical application the original algorithm given by (6) is modified to incorporate directive sensors at discreet locations.

To implement this, let there be Q sensor locations (sensor bands hereafter) along the arm that are equally spaced when all of the actuators are at rest (Fig. 3). Each sensor band has a number of sensors around the arm where Fig. 4 illustrates the case for 8 sensors around each sensor band. When an obstacle is within the threshold range sensors indicate their presence with distal information. Since there is no orientation information repelling velocities are always assigned normal to the neutral axis away from the obstacle. When the object is within one sensor range (object A) the velocity is alloted opposite to the center line of the field regardless of the obstacle location $\left(v_{o|i}\right)$. In the same way, if an obstacle is detected by multiple sensors (object B) similar steps are taken as illustrated in Fig. 3, $\left(v_{o|k}\right)$.

Therefore, the modified system with discreet sensor bands is now given by (7) where the superscript s refer to the values computed for sensors bands at location s.

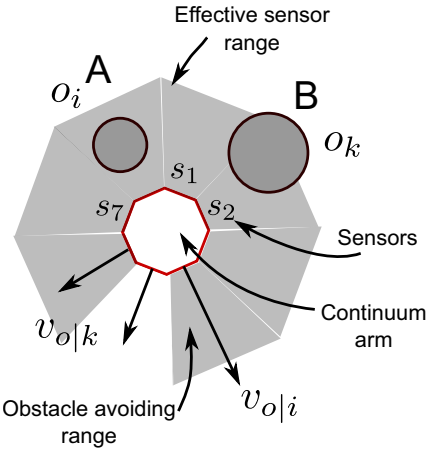

Fig. 4. Sensors mounted around the arm and their sensing fields. These provide distal information without orientation and therefore repel velocities are added opposite to the center of sensor field.

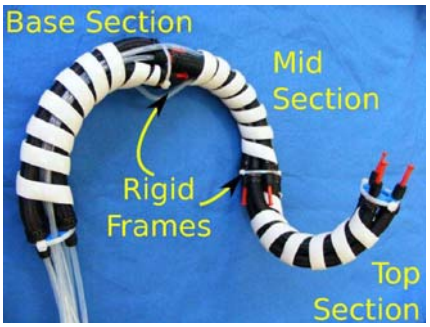

Fig. 5. A prototype three section continuum robotic arm

$$\dot{\boldsymbol{q}} = \mathbf{J}_e^{\dagger}\dot{\boldsymbol{p}}_e + \sum_{s=1}^{Q}\sum_{k=1}^{M}\alpha_{n|k}^{s}\left[\mathbf{J}_k^s\left(\mathbf{I}-\mathbf{J}_e^{\dagger}\mathbf{J}_e\right)\right]^{\dagger}\left(\alpha_{0|k}^{s}\dot{\boldsymbol{p}}_k^s - \mathbf{J}_k^s\mathbf{J}_e^{\dagger}\dot{\boldsymbol{p}}_e\right) \tag{7}$$

This system is iteratively solved until the arm tip reaches p_{tar}. In each iteration, the inverse solution for the main goal is first calculated by $\mathbf{J}_e^{\dagger}\dot{\boldsymbol{p}}_e$. Then the contribution from repelling velocities, as a result of obstacle presence, is calculated for different sensor locations, and the cumulative effect added to the iterative solution.

IV. Simulation Results

The obstacle avoidance algorithm developed for variable length continuum arms is simulated for static and dynamic environments. For the simulation the arm is restricted to 3 sections ($N = 3$, Fig. 5) as most prototypical robots have 3 sections [4], [5]. Floating base coordinates $\boldsymbol{q}_b = \begin{bmatrix} \alpha & \beta & z \end{bmatrix}^T$ are used where $\alpha \in \left[-\frac{\pi}{3}, \frac{\pi}{3}\right]$ and $\beta \in \left[-\frac{\pi}{3}, \frac{\pi}{3}\right]$ are rotation variables about the $+X$ and $+Y$ axes and z is the translating variable along the Z axis of the inertial frame (Fig. 1). Therefore the system has 12 degrees of freedom in total. Seven sensor band locations ($Q = 7$), equally spaced when the arm is at rest, are used along the continuum arm to sense the obstacle distances.

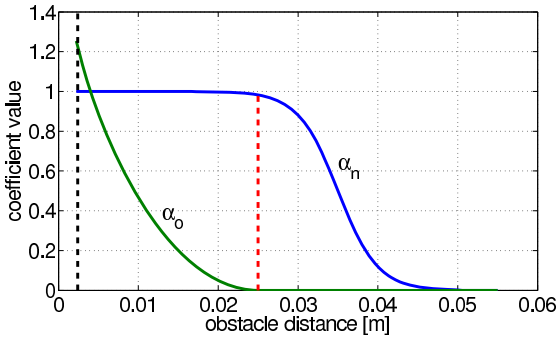

Fig. 6. Coefficient α_n and α_0 value change against obstacle distance

The coefficients used in these simulations, α_o and α_n, are given by

$$\alpha_n(d) = \begin{cases} 1 & : d \le r_c \\ \cos^2\left(\frac{\pi}{2}\frac{d-r_c}{r_0-r_c}\right) & r_c < d \le r_0 \\ 0 & : r_0 < d \end{cases} \tag{8}$$

$$\alpha_o(d) = \begin{cases} k_2\left\{1-\sin^{\frac{1}{4}}\left(\frac{\pi d}{2r_c}\right)\right\} & : r_s \le d \le r_c \\ 0 & : r_c < d \end{cases} \tag{9}$$

where d is the distance to the obstacle and r_c is the critical distance from the obstacle for which the repelling velocities are assigned. The range r_0 stands for the range of the sensor and is further than r_c ($r_c < r_0$). The range r_s is the distance to the obstacle where a collision is imminent. The gain k_2 controls the repelling velocity increase as the objects get closer. This gain should be carefully selected without high values so as to guarantee a smooth transition from free moving to obstacle avoidance motion. The gain variation of these coefficients as a function of obstacle distance (d) are shown in Fig. 6.

The variables values used in the simulations are given in Table I, where l_{ij} (i section number, j actuator number and $i, j \in [1, 2, 3]$) are the actuator length variations. For the ease of simulation, all actuators are assumed to have a similar actuation range. It can be seen by the actuator length variable and floating base variable values in the Table I, they have different numerical ranges. This can have undesired effects on numerical solutions especially because inverse Jacobians are being computed because some variable increments being too high in comparison to the others in parameter update steps and can result in lengthy simulation times. Therefore, to normalize the variable values, a weighting matrix, $\mathbf{W} = diag\{1 \cdots 1, 70, 70, 10\} \in \mathbb{R}^{12\times 12}$, has been used where no scaling is performed on actuator length variables (first 9 rows). Scaling factors for floating base variables α, β, and z were respectively 70, 70 and 10.

During simulations the continuum arm is given a trajectory within its reach to track among obstacles. The trajectory is defined by 20 equally placed point targets and they are presented stepwise to the iterative algorithm. When the arm reaches within the specified termination error, $d_T = 0.01m$,

TABLE I
VARIABLE VALUES

Variable	value	Unit	Variable	value	Unit
L_0	0.15	m	β	$-\frac{\pi}{3} \cdots \frac{\pi}{3}$	rad
l_{ij}	$0 \cdots 0.06$	m	r_0	0.05	m
z	$0 \cdots 0.6$	m	r_c	0.025	m
α	$-\frac{\pi}{3} \cdots \frac{\pi}{3}$	rad	r_s	0.005	m
k_2	2	-			

the target point is changed to the immediate next point and this process continues until the arm tip reaches the end of the tracking trajectory.

A. Obstacle Avoidance in Static Environments

This simulation demonstrates the arm tip tracking a dynamic target that goes through a static tube. The arm starts out in a neutral straight position and bends as it enters the constraining tube walls. The tube has an inner radius of 0.06 m and bending radius of 0.2 m. To simulate the tube walls equally spaced rings (shown in blue color) at $\frac{\pi}{16}$ rads apart are used and each ring has 8 points. Distances are calculated to these points ($d_k = \|\boldsymbol{p}_s - \boldsymbol{x}_k\|$ where $\boldsymbol{p}_s$ is the position vector of the s^{th} sensor band) from the sensor points of the arm to compute the iterative inverse solution. The trajectory was defined by 20 points along the center line of the tube.

It can be seen that the arm tip successfully navigates through the tunnel without collisions. Note how the inherent continuous bending of the continuum arm helps it to snake through the tube by assuming the tube's shape to avoid collisions. This is a very useful feature of continuum arms in comparison to rigid linked robotic arms for use in constrained environments such as minimally invasive surgeries and search-rescue operations. The simulation computation time on a Dell Precision 3500 personal computer system was $\sim 20s$ running on Matlab Simulink®.

B. Obstacle Avoiding in Dynamic Environment

For the results shown in Fig. 8 the arm is given a trajectory to follow defined by $p_{tar} = [-0.2 + 0.02u \quad 0 \quad 0.3 + 0.015u]^T$, where similar to the previous case $u = [1, 20]$ is the trajectory point number. The environment consist of dynamically varying obstacles where rigid spheres were chosen for mathematical convenience. In the figure the surrounding semi-transparent region shows the critical distance from the obstacle and the opaque region shows the sphere dimensions. The obstacle trajectories are calculated to pass through arm's tracking poses for the same trajectory without obstacles. This ensures demonstration of the algorithms adaptability in obstacle avoidance in a dynamic environment. The arm tip starts away from the target, shown with a green dot, and tracks the trajectory while successfully avoiding the moving obstacles. The solution trajectory was obtained within $\sim 10s$ of computation time.

V. DISCUSSION

The proposed algorithm uses the modal kinematic model presented in [10]. Since it is numerically stable without singular poses and yields simulation results efficiently. Otherwise, assuming straight arm poses, such as the starting pose in Fig. 7 is not possible. Its functional implementation also allows users to arbitrary many sections. The same obstacle avoidance algorithm used in this work can be applied to other configurations of continuum arm kinematic models such as [16].

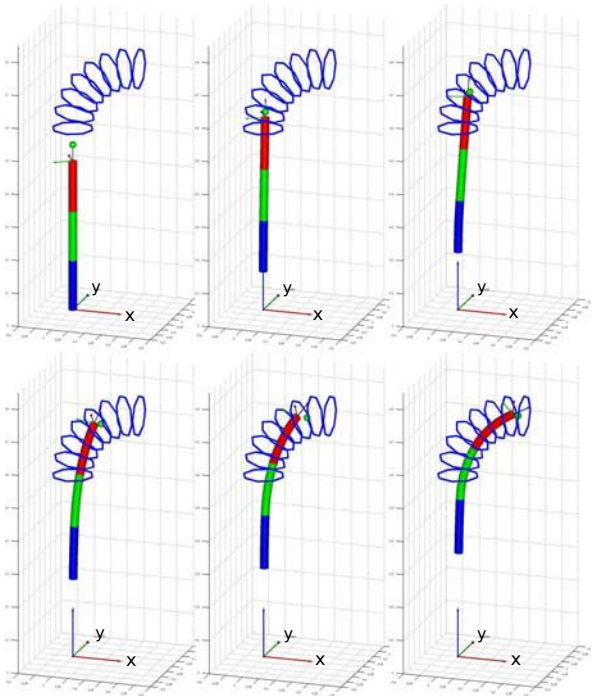

Fig. 7. Navigating the arm tip in a static environment. The arm tip follows a trajectory through a narrow curved tube avoiding contact with walls.

In case of a failure to reach the desired arm location, p_{tar} due to reasons such as beyond the actuation range or

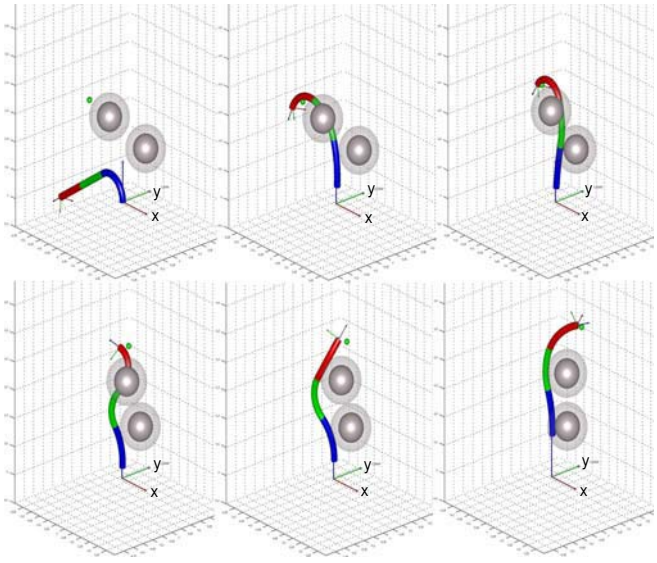

Fig. 8. Navigating the arm tip in a dynamic environment. The arm tip tracks a spatial trajectory avoiding contact with moving obstacles.

obstacle obstruction, the algorithm converges to the best pose where the arm tip is closest to the desired location. In such a case, additional mechanisms are incorporated into the algorithm to restart with a different pose and improve the previous solution. These starting poses are not random, but rather derived from the previous unsuccessful solutions. This ensures that the arm will not trap in a knot [13] and avoids possible collisions with objects due to arbitrary starting poses.

The results obtained above apply the obstacle avoidance algorithm to a large scale continuum arm that can support mounting proximity sensors, and this holds if the environment cannot be wholly sensed using alternative methods such as vision. Also, this makes the continuum arm self contained in non-engineered environments such as collapsed buildings. However, in the case of surgical tools the environment can be accurately sensed by employing state of the art technologies such as 3D ultrasound or MRI scanning. Then the distal information between the surgical tool and tissues can be calculated using image/video processing techniques and the same obstacle avoidance algorithm applied.

The algorithm can also be modified for other applications to obtain desired paths for given objectives. For instance in the case of surgical procedures, the continuum arm tool can plan the path closer to less critical tissues for higher safety. In this case, attracting velocities are used instead of repelling velocities to reach these objectives.

VI. Conclusions

An obstacle avoidance algorithm for multisection continuum arms was presented. The algorithm, originally theorized for rigid linked robots, was modified to apply to continuum arms and further constrained to simulate practical implementation. Simulations for static environment motions presented the arm navigating through a narrow tube, and dynamic environment simulation showed the arm avoiding obstacles while tracking a trajectory. Simulation results show a multisection continuum arm's potential for useful tasks in obstructive environments. This work utilizes the numerically stable mode shape function based kinematics for continuum arms previously presented by Godage *et al.* [10], giving efficient results without singularity concerns for multisection continuum arms. This approach can be extended for other continuum arm configurations and to include dynamic arm configurations.

Acknowledgments

This work was in part supported by the European Commission in the ICT-FET OCTOPUS Integrating Project, under contract n. 231608.

References

[1] W. McMahan and I. D. Walker, "Octopus-inspired grasp-synergies for continuum manipulators," in *IEEE Int. Conf. on Robotics and Biomimetics*, Bangkok, 2008, pp. 945–950.

[2] G. Immega, K. Antonelli, K. Inc, and B. Vancouver, "The KSI tentacle manipulator," in *IEEE Int. Conf. on Robotics and Automation*, 1995, pp. 3149–3154.

[3] W. McMahan, B. A. Jones, and I. D. Walker, "Design and implementation of a multi-section continuum robot: Air-octor," in *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, 2005, pp. 2578–2585.

[4] W. McMahan, V. Chitrakaran, M. A. Csencsits, D. M. Dawson, I. D. Walker, B. A. Jones, M. B. Pritts, D. Dienno, M. Grissom, and C. D. Rahn, "Field trials and testing of the OctArm continuum manipulator," in *IEEE Int. Conf. on Robotics and Automation*, 2006, pp. 2336–2341.

[5] I. S. Godage, D. T. Branson, E. Guglielmino, G. A. Medrano-Cerda, and D. G. Caldwell, "Shape function-based kinematics and dynamics for variable length continuum robotic arms," in *IEEE Int. Conf. on Robotics and Automation*, 2011, pp. 452–457.

[6] N. Simaan, R. Taylor, and P. Flint, "A dexterous system for laryngeal surgery," in *IEEE Int. Conf. on Robotics and Automation*, 2004, pp. 351 – 357.

[7] R. J. Webster, J. M. Romano, and N. J. Cowan, "Mechanics of precurved-tube continuum robots," *IEEE Tran. on Robotics*, vol. 25, no. 1, pp. 67–78, feb. 2009.

[8] P. Sears and P. Dupont, "A steerable needle technology using curved concentric tubes," in *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, 2006, pp. 2850–2856.

[9] J. Furusho, T. Ono, R. Murai, T. Fujimoto, Y. Chiba, and H. Horio, "Development of a curved multi-tube (cmt) catheter for percutaneous umbilical blood sampling and control methods of cmt catheters for solid organs," in *IEEE Int. Conf. Mechatronics and Automation*, vol. 1, 2005, pp. 410–415.

[10] I. S. Godage, E. Guglielmino, D. T. Branson, G. A. Medrano-Cerda, and D. G. Caldwell, "Novel modal approach for kinematics of multisection continuum arms," in *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, 2011, pp. 1093–1098.

[11] G. S. Chirikjian and J. W. Burdick, "An obstacle avoidance algorithm for hyper-redundant manipulators," in *IEEE Int. Conf. on Robotics and Automation*, 1990, pp. 625–631 vol.1.

[12] L. A. Lyons, R. J. Webster, and R. Alterovitz, "Motion planning for active cannulas," in *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, 2009, pp. 801–806.

[13] J. Xiao and R. Vatcha, "Real-time adaptive motion planning for a continuum manipulator," in *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, 2010, pp. 5919–5926.

[14] B. A. Jones and I. D. Walker, "Kinematics for multisection continuum robots," *IEEE Tran. on Robotics*, vol. 22, no. 1, pp. 43–55, 2006.

[15] A. A. Maciejewski and C. A. Klein, "Obstacle avoidance for kinematically redundant manipulators in dynamically varying environments," *The International Journal of Robotics Research*, vol. 4, no. 3, pp. 109–117, 1985.

[16] T. Zheng, D. T. Branson, E. Guglielmino, and D. G. Caldwell, "A 3D Dynamic Model for Continuum Robots Inspired by An Octopus Arm," in *IEEE Int. Conf. on Robotics and Automation*, 2011, pp. 3652–3657.