Model-Less Feedback Control of Continuum Manipulators in Constrained Environments

Michael C. Yip and David B. Camarillo, *Member, IEEE*

***Abstract*—Continuum manipulators offer a means for robot manipulation in a constrained environment, where the manipulator body can safely interact with, comply with, and navigate around obstacles. However, obstacle interactions impose constraints that conform the robot body into arbitrary shapes regardless of actuator positions. Generally, these effects cannot be wholly sensed on a continuum manipulator and, therefore, render model-based controllers incorrect, leading to artificial singularities and unstable behavior. We present a task-space closed-loop controller for continuum manipulators that does not rely on a model and can be used in constrained environments. Using an optimal control strategy on a tendon-driven robot, we demonstrate this method, which we term *model-less control*, which allows the manipulator to interact with several constrained environments in a stable manner. To the best of our knowledge, this is the first work in controlling continuum manipulators without using a model.**

***Index Terms*—Continuum manipulator, convex optimization, feedback control, optimal control, redundantly actuated system, robot control, tendon drive.**

I. Introduction

ONE of the major benefits of a continuum manipulator is that it can be placed into constrained environments, where its body can conform around and interact with obstacles in a safe manner. [1]. Because of this, continuum manipulators have a fast-growing number of applications, particularly in medicine, where compliance allows for safe minimally invasive interventions [2], [3]. However, because the compliance of the robot body creates infinite degrees of freedom, it is a challenging task to accurately model and control the position of the robot end-effector, especially in the presence of disturbances. Therefore, in constrained environments, task-space closed-loop control is used for overcoming these disturbances to accurately position the manipulator [4]–[10].

Manuscript received June 11, 2013; revised December 18, 2013; accepted February 26, 2014. Date of publication March 20, 2014; date of current version August 4, 2014. This paper was recommended for publication by Editor B. J. Nelson upon evaluation of the reviewers' comments. This work was supported by Natural Sciences and Engineering Research Council Canada and the Stanford Bio-X Fellowship program.

The authors are with the Department of Bioengineering, Stanford University, Stanford, CA 94305 USA (e-mail: mcyip@stanford.edu; dcamarillo@stanford.edu).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TRO.2014.2309194

A. Background

Task-space closed-loop controllers for discrete manipulators[1] [12] often use a kinematic model of the robot Jacobian that maps actuator velocities to the end-effector velocity [11]. For continuum manipulators, models have been developed for specific architectures that use tendons [10], [13], [14]; precurved concentric tubes [6], [8], [15]–[23]; pneumatics channels [24], [25]; shape-memory-alloy actuators [26]–[29]; and multibackbone actuators [30]–[33]. However, these models do not account for unknown disturbances from the environment. Therefore, problems with the models arise when the continuum manipulator is introduced into an environment with unknown constraints and obstacles.

For discrete manipulators, a model-based Jacobian is accurate in a constrained environment regardless of robot interactions with obstacles. This is because all the degrees of freedom are sensed and, therefore, obstacle effects on the robot configuration and Jacobian are known. However, for a continuum manipulator, its model-based Jacobian is inaccurate in a constrained environment because it has infinite degrees of freedom and only finite measurements; not all degrees of freedom can be sensed. As such, in an unknown environment, constraints on the robot's configuration and Jacobian cannot be effectively modeled.

Two effects of constraints on the robot Jacobian are *scaling* and *rotation*—that is, scaling in the singular values of the Jacobian matrix, and change in orientation of the Jacobian column vectors, respectively. Unsensed scaling and rotation can have adverse effects on closed-loop control. This is illustrated by examining two canonical examples. In Fig. 1(A), the robot's interaction with an obstacle results in constrained motion. Since the model cannot sense the obstacle, it presumes that the robot is at the end of its workspace, experiencing an artificial singularity. The true Jacobian has been scaled and rotated such that it is, in fact, nonsingular. In Fig. 1(B), constraints have caused a difference in orientation between the columns of the model-space and the columns of the true Jacobian. Therefore, the actuator's effect on position output is opposite to what is modeled, resulting in a positive-feedback loop, risking damaging the robot and the environment.

A few investigators have recently begun to address the challenges of controlling a continuum manipulator in the presence of constraints. Bajo and Simaan presented a kinematics-based method for identifying contact between a single obstacle and a continuum manipulator without *a priori* knowledge of the environment [34]. For each additional contact point, an

[1]Robot manipulators comprising rigid links with measurement of link actuation at each degree-of-freedom.

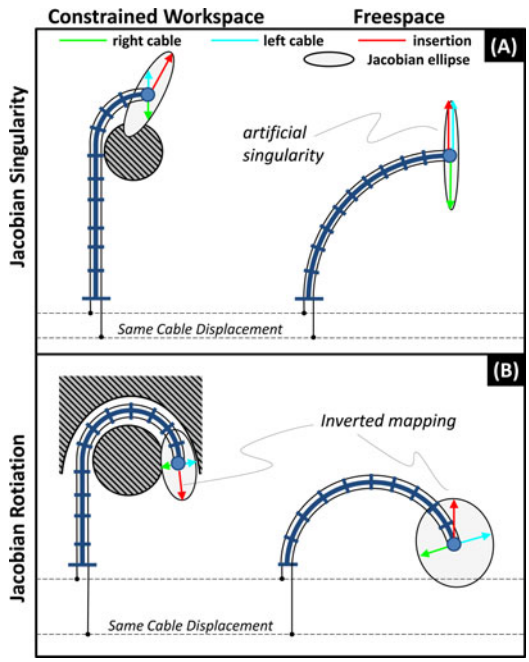

Fig. 1. Tendon-driven robot, in a constrained environment (left), and in freespace (right). In both cases, the actuations are kept the same. The red, blue, and green vectors represent the columns of the Jacobian, which is the effect of a positive actuation of the insertion, left tendon, and right tendon actuators on the tip position. The Jacobian is represented as the minimum volume ellipsoid for these columns [11]. A model that does not sense the environmental constraints will presume a freespace Jacobian even in a constrained situation, which can lead to (A) artificial singularities or (B) inverted mappings that can cause positive feedback loops.

additional position sensor was used to sense the extra contact point. Goldman *et al.* [35] developed a method for mapping measured environmental wrenches to generalized forces in the robot's configuration space; they then actuate the robot towards a minimum energy state, thereby allowing it to be compliant to its environment. Rucker *et al.* used an extended Kalman filter approach for estimating contact forces based on manipulator deflection and a statics model [20], and obtained a model for the robot Jacobian by combining their mechanics model with a differential model for predicting contact loading [36]. Xu *et al.* [37] developed a recursive linear-estimation method to accurately model the kinematic, bending, and friction models of their continuum manipulator.

These methods use complicated kinematic, mechanic, and dynamic models of the manipulators that require an accurate calibration and are sensitive to unmodeled effects (e.g., internal friction in the manipulator's drivetrain); furthermore, some methods require additional sensors along the robot body, which can be impractical to implement for continuum manipulators working in constrained spaces. A task-space closed-loop controller that does not rely on a model and does not require additional sensing on the robot body could offer a simple and robust solution to unknown environments.

B. Contributions

In this paper, we present a *model-less* approach for controlling a continuum manipulator based on empirical estimates of the robot Jacobian. We implement the method on a tendon-driven manipulator, which is redundantly actuated. Optimal control is used in task-space closed-loop feedback to derive the control inputs while minimizing tendon tensions. We show that with no model of our continuum manipulator, the model-less control approach can navigate in both freespace and constrained environments, in situations where a model-based controller would experience workspace singularities or become unstable. To the best of our knowledge, this is the first work in controlling a continuum manipulator without using a model.

II. Methods

A. Manipulator Jacobian

A robot manipulator can be described by a function $f : R^n \rightarrow R^m$ of the actuator inputs $\mathbf{y} = [y_1 \cdots y_n]^T$, to the end effector position output $\mathbf{x} = [x_1 \cdots x_m]^T$, such that

$$\mathbf{x} = f(\mathbf{y}). \tag{1}$$

The velocity of the robot manipulator is given by

$$\dot{\mathbf{x}} = \dot{f}(\mathbf{y}) \tag{2}$$

where $\dot{(\cdot)}$ is a time derivative. Any continuous function $\dot{f}$ can be approximated by a linear function in a local region of $\mathbf{x}$, such that

$$\dot{\mathbf{x}} \approx \mathbf{J}\dot{\mathbf{y}} \tag{3}$$

where $\mathbf{J} \in R^{m \times n}$ is the Jacobian matrix. The ith column of $\mathbf{J}$ represent the contribution of the ith actuator, y_i, on the end effector position. There is a specific $\mathbf{J}$ for each position $\mathbf{x}$ in the manipulator workspace. The Jacobian matrix can calculated by taking the partial derivatives of the f at the position $\mathbf{x}$, such that

$$\mathbf{J} = \left[\frac{\partial f(\mathbf{x})}{\partial y_1}^T \quad \cdots \quad \frac{\partial f(\mathbf{x})}{\partial y_n}^T \right]. \tag{4}$$

For discrete manipulators, f is only a function of the robot degrees of freedom, which are individually measured, and, therefore, f can be derived through forward kinematics. However, a flexible continuum manipulator has infinite degrees of freedom which are exercised as the manipulator interacts and conforms around obstacles. Therefore, the mapping function f between actuator inputs and end-effector motion is unknown and must be empirically estimated.

B. Model-Less Control Framework

The model-less controller first acquires an initial estimate of the Jacobian matrix offline, and then, when the robot is online, continuously estimates the robot Jacobian as it moves around the workspace. This method is presented in Fig. 2 and is described in detail below:

1) Initialize the system for $\hat{\mathbf{J}}$*:* A Jacobian matrix is constructed by moving each actuator independently an incremental amount, Δy_i while measuring the displacement $\Delta\mathbf{x}$. The ith column of $\mathbf{J}$ is constructed as $\mathbf{J_i} = \Delta\mathbf{x}/\Delta y_i$, such that

$$\mathbf{J} = [\mathbf{J_0} \cdots \mathbf{J_n}]. \tag{5}$$

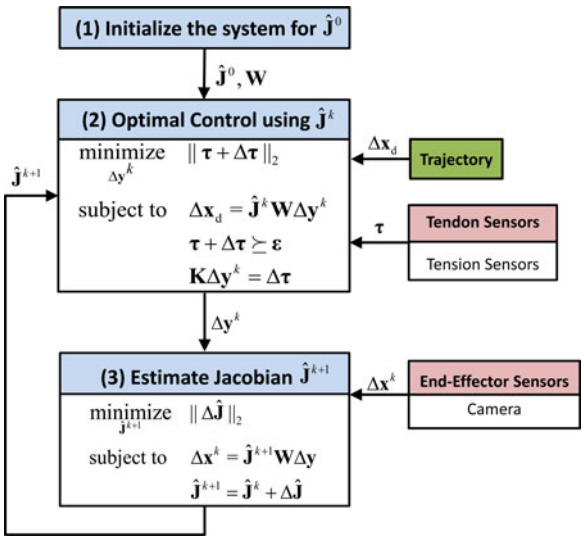

Fig. 2. Flowchart of constrained optimization method for model-less control.

A weighting matrix $\mathbf{W}$ is constructed to normalize the actuator signals, such that

$$\mathbf{W} = \text{diag}\left(\|\mathbf{J_0}\|_2, \ldots, \|\mathbf{J_n}\|_2\right) \tag{6}$$

where $\|\cdot\|_2$ is the vector two-norm operator. Then

$$\dot{\mathbf{x}} = \hat{\mathbf{J}}\mathbf{W}\dot{\mathbf{y}}$$

where $\hat{\mathbf{J}} = \mathbf{J}\mathbf{W}^{-1}$ is the Jacobian matrix with column vectors of unit length. $\mathbf{W}\dot{\mathbf{y}}$ normalizes and nondimensionalizes the actuator signals such that, when solving for actuator displacements using (7), desired displacements are not affected or biased by units of measurement or gearing.

We then can estimate the derivative terms by a first-order approximation, where

$$\Delta\mathbf{x} = \hat{\mathbf{J}}\mathbf{W}\Delta\mathbf{y}. \tag{7}$$

Tendon-driven continuum manipulators are often redundantly actuated in order to achieve bidirectional actuation [24]. In these cases, co-activation of tendons results in pretensions along the tendons and the body of the manipulator, which is a quantity of interest that can be minimized. Tendon tensions can be related to tendon displacements by a linear function [13], therefore we define a diagonal stiffness matrix $\mathbf{K} \in R^{n\times n}$, such that

$$\Delta\tau = \mathbf{K}\Delta\mathbf{y} \tag{8}$$

where τ is the tension in the tendons. Columns of $\mathbf{K}$ can be constructed by moving each actuator separately and measuring the change in tensions in all tendons. Tendon tensions τ are measured using a force sensor for each tendon.

2) Optimal Control using $\hat{J}^k$: Given a desired displacement for the end effector, $\Delta\mathbf{x_d}$, the actuator displacements $\Delta\mathbf{y}$ can be solved for through the following:

$$\begin{aligned} \underset{\Delta\mathbf{y}^k}{\text{minimize}} \quad & \|\tau+\Delta\tau\|_2 \\ \text{subject to} \quad & \Delta\mathbf{x}_\text{d} = \hat{\mathbf{J}}^k\mathbf{W}\Delta\mathbf{y}^k \\ & \tau+\Delta\tau \succeq \epsilon \end{aligned}$$

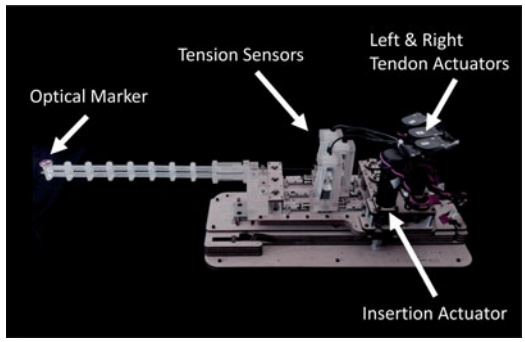

Fig. 3. Two-dimensional planar robot driven by two co-activated tendon servos for the curvature degree of freedom and a servo for insertion/retraction of the robot. The insertion and the curvature degrees of freedom are mechanically decoupled.

$$\Delta\tau = \mathbf{K}\Delta\mathbf{y}^k \tag{9}$$

where $\Delta\mathbf{y}^k$ is the optimization variable and $\hat{\mathbf{J}}^k$ is the Jacobian estimate from time k. The objective function to be minimized is the L2-norm of the tensions in the tendons. We chose $\|\tau+\Delta\tau\|_2$ in order to minimize the amount of unnecessary coactivation and axial compression on the manipulator and avoid buckling or damaging to the manipulator. The constraint equations satisfy (7); in addition, they ensure that tensions in each tendon must be greater than ϵ, which is a minimal tension that forces the tendons to remain taut in order to avoid tendon slack.

3) Estimate Jacobian $\hat{\mathbf{J}}^{k+1}$: Optimization in the previous step relies on continuous estimations of the Jacobian, and is estimated by solving the following optimization problem:

$$\begin{aligned} \underset{\hat{\mathbf{J}}^{\mathbf{k+1}}}{\text{minimize}} \quad & \|\Delta\hat{\mathbf{J}}\|_2 \\ \text{subject to} \quad & \Delta\mathbf{x}^k = \hat{\mathbf{J}}^{k+1}\mathbf{W}\Delta\mathbf{y}^k \\ & \hat{\mathbf{J}}^{k+1} = \hat{\mathbf{J}}^k + \Delta\hat{\mathbf{J}} \end{aligned} \tag{10}$$

where $\Delta\hat{\mathbf{J}}$ is the optimization variable, $\hat{\mathbf{J}}^k$ is the Jacobian matrix at time k, $\hat{\mathbf{J}}^{k+1}$ is the new Jacobian estimate, and $\{\Delta\mathbf{x}^k, \Delta\mathbf{y}^k\}$ are measured displacements from actuator and end-effector sensors between the last Jacobian estimate at time k and the present time. We chose to minimize the Frobenius norm (element-wise L2-norm) of $\Delta\hat{\mathbf{J}}$ such that the column vectors of the Jacobian transition smoothly.

C. Experimental Setup

A continuum manipulator was constructed to evaluate the model-less control algorithm (see Fig. 3). The robot is a 2-D planar manipulator, with a flexible backbone constructed of a polypropylene beam (length: 280 mm, cross-section: 0.8 mm × 12 mm, 2500–5400 psi, McMaster Carr, Santa Fe Springs, CA, USA). The backbone is segmented into eight equal subsections by 3-D printed plates acting as tendon guides. The curvature of the backbone is driven by two 0.6-mm diameter stranded steel cables (Sava Industries, Riverdale, NJ, USA) on either side and are equidistant from the central backbone. Each cable terminates on a tension sensor (LSP-5, range: 0–5N, Transducer Techniques, Temecula, CA, USA). The tension

sensor rests on a linear slide and is servo-actuated to produce tendon actuation. The backbone and tendon actuation assembly is mounted on a servo-actuated carriage that supports unaxial insertion/retraction. This decouples the insertion actuation from the tendon actuations. The three actuations for a planar manipulator form the redundantly-actuated system. A PID controller is implemented at each actuator for steady-state error rejection.

A 640×480 pixel resolution video feedback (Logitech C920, Newark, CA, USA) was used as an uncalibrated camera for position feedback at 20 Hz. An optical marker was secured to the tip of the robot manipulator. Using the OpenCV C++ library, color thresholding identifies the optical marker and a Harris corner detector was used to extract the location of the saddle point in the marker to subpixel resolution. The camera was placed 50 mm from the 2-D robot workspace pointed directly in line with the normal of the planar workspace, achieving a resolution of approximately 0.5 mm per pixel.

To accommodate for sensor noise and the camera resolution, the Jacobian reestimation step was performed only when the manipulator had moved more than eight pixels (4 mm or 1.4% of the manipulator length), and the new Jacobian estimate used a smoothing constant $\alpha = 0.5$, where $\hat{\mathbf{J}}^{k+1} = \hat{\mathbf{J}}^k + \alpha\Delta\hat{\mathbf{J}}$. These values were empirically acquired in order to minimize noise and digitization effects. We used α as a simple tuning parameter for the Jacobian estimate, where $0 \leq \alpha \leq 1.0$; $\alpha = 0$ does not update the Jacobian estimate and $\alpha = 1.0$ replaces of the prior Jacobian estimate with its instantaneous estimate. α should be adjusted to be lower when measurement noise is larger, acting as a smoothing filter. The optimal control method was asked to maintain a minimum tendon tension of $\tau = 0.3$N.

The system was controlled through a 64-bit Windows 7 PC with an Intel 3.4 GHz i5 processor with 8GB RAM. The robot control and the camera feedback were run on separate threads; the camera feedback thread was run at 20 Hz and the actuator control loop was run at 1 kHz. Although we used a camera for position feedback, the method is not reliant on any particular sensing modality and other sensors can be used for different circumstances (e.g., magnetic tracking or fiber Bragg sensing, which does not require line of sight). The robot was able to move at approximately 30 mm/sec in an unconstrained environment.

Because we defined the model-less controller as an optimization problem, objective functions that are convex (e.g., L1-norm, L2-norm, L-infinity norm, and the definitions used in this paper) converged to a globally optimal solution, and convergence occurred typically within 10–14 iterations in the optimization solver. We used the C++ version of CVX, a convex optimization solver [38], [39]. On our machine, only 10 ± 1.5 μs (mean and standard deviation) were required for each optimization problem and, therefore, it was not a significant contributor to computation time during a control loop timestep.

III. Results

A. Simulation

We first implemented the model-less controller in a MATLAB simulation to investigate its performance without effects of sensor noise, manipulator dynamics, and environmental constraints. The parameters and robot model used in the MATLAB simulation are described in the Appendix.

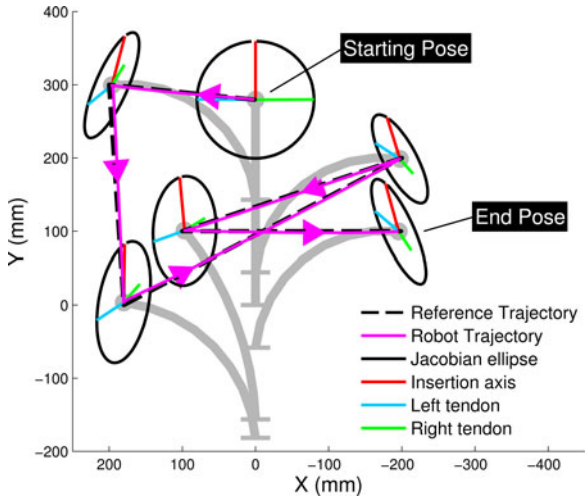

Fig. 4. MATLAB simulation of the model-less control method for a set of randomly selected locations. The model-less control is shown to track a reference trajectory closely (0.58 ± 0.33 mm) in simulated free-space motion and Jacobian ellipsoids are shown at each endpoint. This demonstrates the accuracy of the model-less control method in the absence of noise, disturbances, and manipulator dynamics.

Fig. 4 shows the robot moving in the simulated workspace to a set of randomly generated endpoints. The Jacobian estimate at each endpoint is plotted as an ellipse, and each column of the Jacobian estimate is shown as a red, green, and blue vector. This shows the effect of a positive actuation in the insertion, left tendon, and right tendon actuators on the end-effector position.

We observed that the Jacobian ellipses become flatter as the robot extends further away from the insertion axis due to 1) the columns of the Jacobian attenuating such that the singular values of the Jacobian are scaled down, and 2) the columns of the Jacobian rotating to become more in-line with one another. This behavior is expected as the robot reaches near the boundaries of the workspace. The ellipses also show that the Jacobian estimates throughout the simulation are approximately correct but not exact, for three reasons. First, because the optimization method is attempting to solve an under-constrained problem (10), there is more than one Jacobian solution that satisfies the constraints. The calculated Jacobian estimate, which minimizes the change in its Frobenius norm, may therefore exhibit slight misalignments (both in simulation or in physical experiments). Second, some inevitable cross-coupling of the actuator effects on the output position will result in cross-coupling in the elements of the Jacobian matrix. Third, the Jacobian estimation step uses backwards differencing to estimate the Jacobian, which therefore causes the estimate to lag the true Jacobian.

B. Trajectory Tracking in Free Space

The control algorithm was implemented on the physical robot and a square trajectory was given for the robot end effector to trace (see Fig. 5). Given an unconstrained environment, we demonstrated that the robot was able to move to the correct

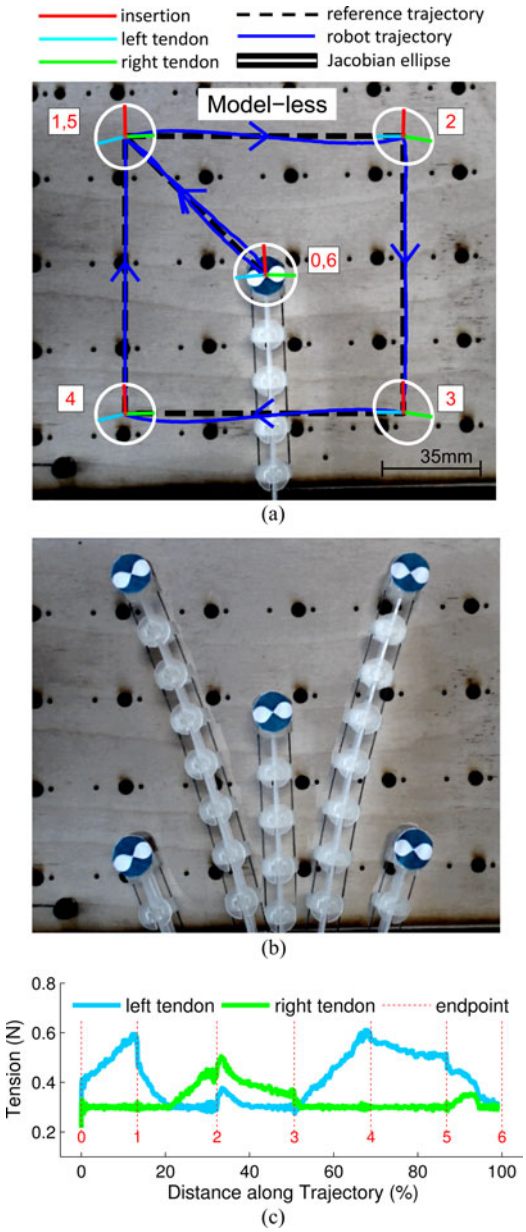

Fig. 5. Model-less control method tracks a square trajectory in freespace (1.22 ± 0.93 mm). (a) shows the reference and robot trajectories of the robot and where the numbers represents endpoints the robot is asked to move to. The model-less Jacobian ellipses at each endpoint are shown.(b) Shows the final configuration of the robot body at each endpoint, (c) shows the tendon tensions over time.

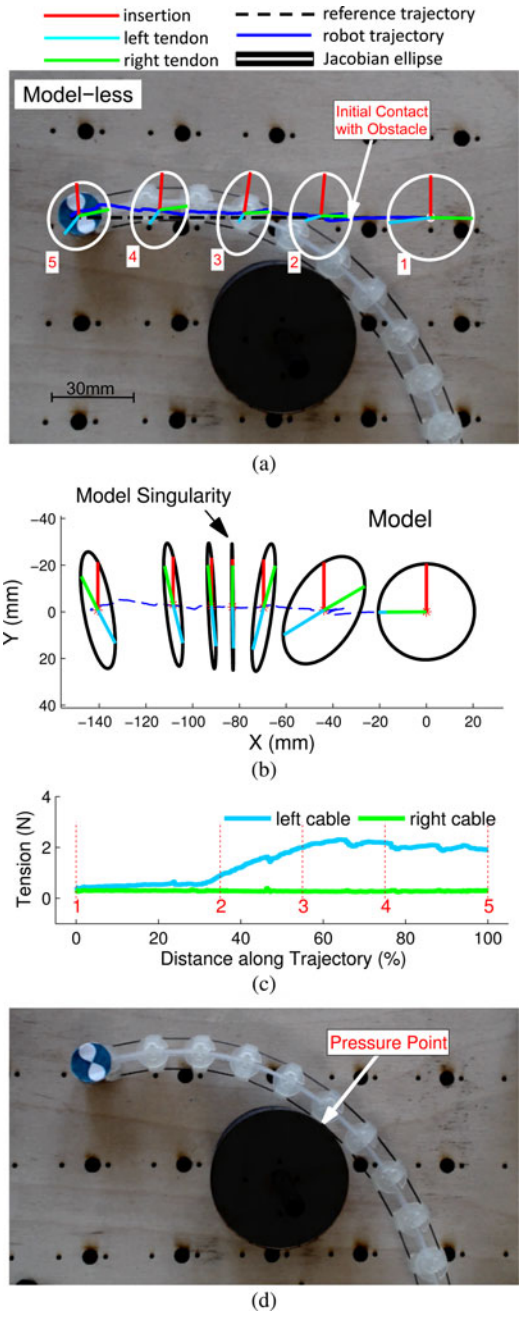

Fig. 6. Robot is able to conform around and move past an obstacle towards its desired endpoint (1.74 ± 1.3 mm). (a) Shows the reference trajectory and the actual trajectory of the robot. Snapshots of the model-less Jacobian ellipses are overlaid over the robot's trajectory. (b) Model-based Jacobian ellipses calculated based on actuator displacement are overlaid on the tip trajectory. This demonstrates how a model-based method would be become immobile prior to reaching the endpoint because of a singularity in the model Jacobian. (c) Shows the tension of the cables during the trajectory. (d) Final configuration of the robot body under model-less control.

locations, with zero steady-state (subpixel) error using the model-less controller. Fig. 5(c) shows the optimization framework is minimizing the tensions in the tendons, attempting to maintain a minimum tension to keep the system taut while tracking a trajectory. This minimizes both the axial compression of the manipulator and wasted coactivation energy in the tendons as well as results in smooth transitions in the tendon actuations.

C. Single Environmental Constraint

We introduced the robot into a constrained environment where it was given a straight trajectory that would involve collision with an obstacle (see Fig. 6). Under model-less control, the robot was able to conform around the obstacle and extend past the constraint to reach the target. The column vector for the left tendon (blue) attenuates as the robot moves into contact with the constraint, which accurately represents how the obstacle would constrain the robot's movement. The rotated direction of the

left tendon vector is why the model-less method was successful in extending its reach past the obstacle: throughout the entire trajectory, a positive actuation in the left tendon allowed the robot to continue to move towards the left, which is demonstrated by the rotation and scaling of the Jacobian estimate. In contrast, if a model-based approach had been used, the model-based Jacobian would not have accounted for the scaling and rotation because of the constraint, and would have encountered a singularity. This singularity would have occurred where the tendon actuation and insertion actuator columns align, which is well before the endpoint was reached [see Fig. 6(d)].[2]

In this simple example of a single obstacle interaction with the robot, a model-less controller is able to circumvent an artificial Jacobian singularity that would have arisen in a model-based controller.

D. Multiple Environmental Constraints

A more complex environment involves multiple obstacles and points of contact on the robot. In Fig. 7, the robot was given a user-defined trajectory that would result in contact with multiple obstacles in the environment; the trajectory involved snaking through the obstacles, resulting in an *S*-shaped curve on the body. This trajectory was generated by having the robot attempt to track the user's mouse position as it traced a path through the channel in real-time. Using model-less control, the robot was successful at following the reference trajectory even as it encountered new obstacles and multiple points of contact of its body. As the robot becomes more constrained by the obstacles, the Jacobian ellipse and column vectors diminish in size but remain well-conditioned. This accurately represents the scaling effect of an increasingly constrained robot body. The model-based Jacobians, shown in Fig. 7(b), become singular in several locations along the path and therefore would have been unable to trace very far along the reference trajectory.

The tendons can be seen to effectively trade-off actuation such that tension is minimized while still maintaining $> 0.3N$ tension on each tendon [see Fig. 7(d)]. At each contact with a new obstacle, the manipulator would deviate slightly from its reference trajectory while the Jacobian estimate would gradually account for the effects of the new constraints.

E. Maneuvering Through a Channel Constraint

We introduced the continuum manipulator into a channel environment that involves a significant change in direction and gave it a reference trajectory defined by the user. The narrow channel causes the insertion column of the Jacobian (red) to rotate over 90° (see Fig. 8). Using model-less control, the robot estimated this rotation and was successful at navigating the channel constraint obstacles. A model-based method with no information about the obstacles is shown in Fig. 8(b). The insertion column would have remained constant in the model throughout the entire trajectory, resulting in an inverted mapping of the actuator input to end-effector position output. Under closed-loop control, this would have resulted in a positive-feedback loop and, thus, unstable behavior.

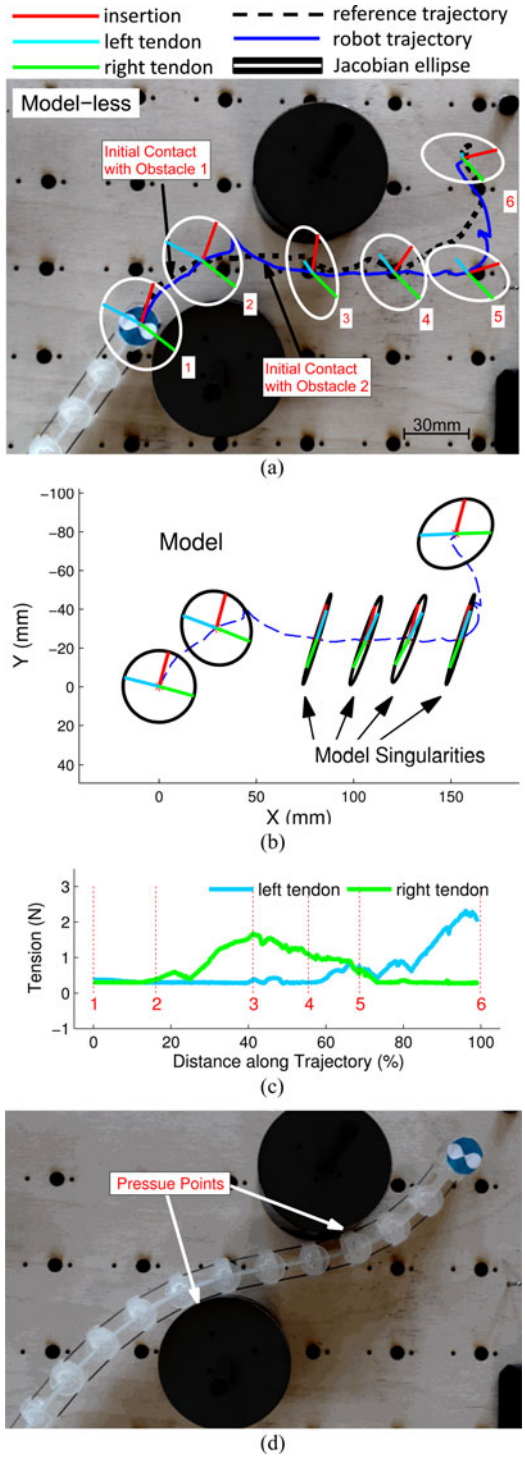

Fig. 7. Robot maneuvers through multiple obstacles that result in multiple unknown contact locations on the robot body (4.94 ± 4.02 mm). (a) Shows the reference trajectory and the actual trajectory of the robot. Jacobians are shown at incremental points along the trajectory, demonstrating successful navigation of the constrained workspace. (b) Demonstration of how a model-based Jacobian would encounter many model-based singularities throughout the trajectory that would halt robot motion. (c) Shows the tradeoff of tensions between the tendons. (d) Final configuration of the robot body under model-less control.

[2]The model-based Jacobians were derived by using the measured actuator positions as inputs into the MATLAB simulation. See the Appendix for details.

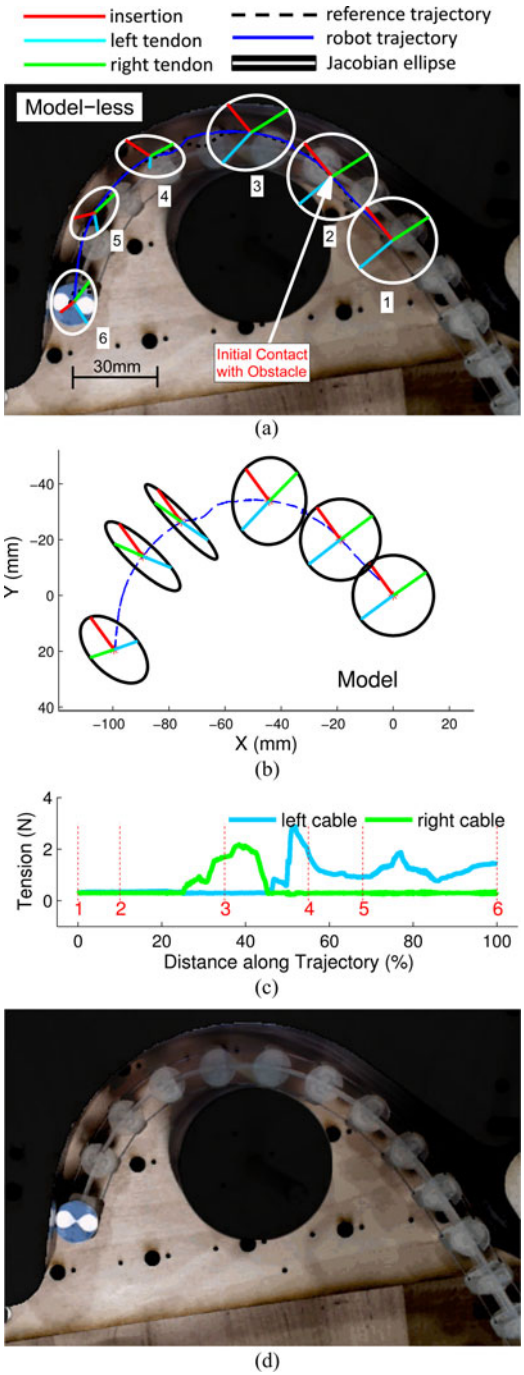

Fig. 8. Robot maneuvers through a channel constraint that results in multiple unknown contact locations on the robot body (1.58 ± 1.86 mm). (a) Reference trajectory and the actual trajectory of the robot. Model-less Jacobian ellipses during robot motion. The model-based Jacobian ellipses are shown to demonstrate that there are significant rotations in the column vectors that are not sensed. In these cases, the actuation directions are flipped in the model, which can lead to dangerous and unexpected actuation behavior. (c) Tensions in the cables. (d) Final configuration of the robot body after successfully navigating the channel using model-less control.

IV. Discussion

Because of their infinite degrees of freedom, continuum manipulators comply with obstacles in unknown environments in such a way that cannot be fully sensed. We have shown that a novel method called "model-less control" can be used for controlling a continuum manipulator within unknown and constrained environments. First, model-less control allows the robot to circumvent artificial singularities that would occur in model-based controllers (see Figs. 6 and 7) by estimating the scaling and rotation of the robot Jacobian caused by the manipulator's interaction with the constrained environment. Second, model-less control accurately estimates the robot Jacobian when the Jacobian matrix and its column vectors are rotated because of constraints (see Fig. 8). Conventional model-based methods would result in artificial Jacobian singularities, and inverted mappings between actuators and end-effector position, resulting in a positive-feedback loop and, thus, unstable behavior. Constraint scenarios will commonly occur in medical environments such as manipulating a catheter in the heart or through vasculature. When translating model-less control to an environment such as an operating table, robot control can be realized using magnetic sensors rather than camera positioning for feedback, where constraints and intracavitational manipulation blocks the line-of-sight. When line-of-sight is not an issue and a calibrated camera is available, techniques in position-based visual-servoing can be used [40], [41].

An advantage of a model-less control method is that because there is no knowledge of the robot's configuration, it can be easy to adapt for manipulators with any number, type, or configuration of actuators, and for an n-dimensional task-space. For example, implementing model-less control on a concentric tube robot [6] with three concentric tubes (with roll and insertion actuation) for 3-D Cartesian positioning would simply involve estimating a 3×6 Jacobian matrix. We also demonstrated that the optimal control method is able to take advantage of a redundantly-actuated system, while minimizing tendon tensions in the system. This prevents buckling of the robot body, and avoids over-stressing the actuators. Although we used a uncalibrated camera for position sensing, model-less control does not require any particular sensing modality.

The ability of model-less control to track the reference trajectory is affected by the workspace environment. In an unconstrained environment, the Jacobian changes smoothly and, therefore, allows a manipulator under model-less control to track the reference well. When the manipulator becomes constrained, the Jacobian needs to change much faster, resulting for more tracking error. In practice, the error can vary depending on the trajectory and the obstacle; for example, a channel constraint (see Fig. 8) will intuitively have less tracking error than when the manipulator is constrained through multiple obstacles (see Figs. 6 and 7), as the manipulator is physically constrained along a path.

There are several opportunities to improve the robustness and effectiveness of the model-less controller. To reduce tracking error, estimation parameters could be adaptive rather than static, adapting the parameters of the Jacobian estimation when there tracking error increases. A limitation in the current embodiment is that constraints on the continuum manipulator that impede the motion of the tip may cause a column of the Jacobian matrix to zero. A method for resolving issues with tip constraints is to

identify them using additional sensing or heuristics. Performing singular value decomposition on the Jacobian matrix will identify the nullspaces in the Jacobian; trajectories or actuations can be modified to avoid moving towards nullspaces and therefore keep the Jacobian of the manipulator well-conditioned. In addition, adding constraints to the optimization solver can be used to avoid zeroing the Jacobian estimate. If the controller does somehow move into and become stuck at a singularity, it can perform the empirical Jacobian initialization method described in Section II-B1 to acquire an exact Jacobian estimate for the location.

Model-less control relies heavily on actuator and end-effector position signals for estimating the robot Jacobian; therefore, it is important that the sensors need to provide accurate and clean signals. Increased noise in the sensor measurements would require greater signal processing and filtering. A simple method for accounting for noise is to adjust α, which acts as a moving average filter. However, both the α filter and the backward differencing of measurements in the Jacobian estimation step contribute to phase lag in the system, and is a current limitation of the method. Improving sensor resolution and reducing sensor noise would allow Jacobian estimation to be performed more frequently, resulting in less estimation lag.

There is currently no stability proof for the generalized optimization method; however, constraints on the optimization solver can be applied to keep the robot stable in practice, such as constraining the Jacobian estimate to be well-conditioned and that actuator and tension displacements and velocities are bounded. If the solution is infeasible, then the robot motion can be terminated or the controller can adapt in some other reasonable way.

In its current form, the model-less control method is suited for static environments only. In other applications such as dynamically moving workspaces, methods for differentiating the motion of the manipulator caused by the environment from those caused by the actuators become necessary. In addition, moving with significant accelerations may introduce nonnegligible manipulator dynamics, and there may be a practical limit to how quickly the manipulator can accelerate to a target speed.

V. Conclusion

In summary, continuum manipulators are difficult to control in a constrained environment where obstacle interactions change the robot Jacobian in unpredictable ways. Because a continuum manipulator has infinite degrees of freedom and a finite number of sensors, these constraints can affect the configuration of the robot in an unknown manner. Using a model-based closed-loop controller, this can lead to artificial singularities and unstable behavior in the robot.

We have presented a novel closed-loop feedback controller that overcomes these common limitations of continuum manipulators. Our robot controller does not rely on a model; it empirically estimates the robot Jacobian, thereby accounting for the changes in the Jacobian as a result of environmental constraints. We demonstrate that the model-less controller is able navigate a range of constrained environments, showing that it effectively expands the manipulator's reachable workspace and enables it to interact with its environment in a safe and stable manner.

To the best of our knowledge, this is the first work on controlling a continuum manipulator without the use of a model.

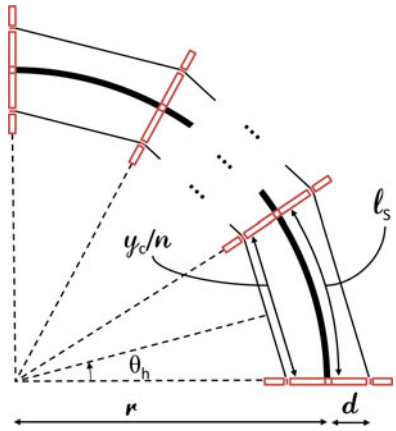

Fig. 9. This model is used only to acquire the model-based Jacobian matrix and for the MATLAB simulation of the robot kinematics. It is not used at all for model-less control.

Appendix

We showed that a model-based Jacobian is incorrect in certain canonical cases where the robot interacts with the environment. Here, we present the method used to acquire the model-based Jacobian, which is also used for simulating the robot in MATLAB. The model is based off the work of Gravagne and Walker's planar robot model [42]. We used this model as it is a simple and well-established model for continuum manipulators. Below, we provide a brief summary of the model for completeness.

We used the following assumptions for the model:

1) The backbone follows a constant curvature, shown in Fig. 9. Each backbone segment has the same radius r and origin of curvature.
2) The robot curvature is determined by the most tensed tendon. Coactivation effect on kinematics is ignored.
3) Distance between the tendon and the backbone is constant along the length of the manipulator.
4) Backbone length L, segment length l, distance between tendon, and backbone d are all constant. At rest, the curvature tendon length $y_c = L$, and the insertion actuator $y_i = 0$. $n+1$ tendon-guiding plates along the backbone divide the backbone into n segments.

The forward kinematics for the robot are derived below

$$\sin\theta_h = \frac{y_c/2n}{r-d} \tag{11}$$

$$r = \frac{l_s}{2\theta_h} \tag{12}$$

where $\theta_h = \frac{\theta}{2}$. Combining (11) and (12)

$$(l_s - 2\theta_h d)\sin\theta_h = \frac{y_c\theta_h}{n}. \tag{13}$$

We then minimize

$$f(\theta) = \frac{y_c\theta_h}{n} - (l_s - 2\theta_h d)\sin\theta_h \tag{14}$$

using the Levenberg–Marquardt method to find θ_h. Then

$$\Theta = 2n\theta_h \tag{15}$$

and the robot tip $[x, y]^T$ is

$$\mathbf{x} = \begin{bmatrix} r(1-\cos\Theta) \\ (r\sin\Theta - L) + y_i \end{bmatrix}. \tag{16}$$

REFERENCES

[1] S. Hirose, *Biologically Inspired Robots: Snake-Like Locomotors and Manipulators*. London, U.K.: Oxford Univ. Press, 1993.

[2] D. B. Camarillo, T. M. Krummel, and J. K. Salisbury, "Robotic technology in surgery: past, present, and future," *Amer. J. Surg.*, vol. 188, no. 4, pp. 2S–15S, Oct. 2004.

[3] L. S. Cowan and I. D. Walker, "The importance of continuous and discrete elements in continuum robots," *Int. J. Adv. Robot. Syst.*, vol. 10, 2013.

[4] R. S. Penning, J. Jung, J. A. Borgstadt, N. J. Ferrier, and M. R. Zinn, "Towards closed loop control of a continuum robotic manipulator for medical applications," in *Proc. IEEE Int. Conf. Robot. Autom.*, May 2011, pp. 4822–4827.

[5] R. S. Penning, J. Jung, N. J. Ferrier, and M. R. Zinn, "An evaluation of closed-loop control options for continuum manipulators," in *Proc. IEEE Int. Conf. Robot. Autom.*, May 2012, pp. 5392–5397.

[6] R. Webster, J. Swensen, J. Romano, and N. Cowan, "Closed-form differential kinematics for concentric-tube continuum robots with application to visual servoing," *Exp. Robot.*, vol. 54, pp. 485–494, 2009.

[7] A. Kapadia and I. D. Walker, "Task-space control of extensible continuum manipulators," *Proc. IEEE Robot. Syst.*, Sep. 2011, pp. 1087–1092.

[8] M. Mahvash and P. E. Dupont, "Stiffness control of surgical continuum manipulators," *IEEE Trans. Robot.*, vol. 27, no. 2, pp. 334–345, Apr. 2011.

[9] D. Camarillo, C. Carlson, and J. Salisbury, "Task-Space control of continuum manipulators with coupled tendon drive," *Exp. Robot.*, vol. 54, pp. 271–280, 2009.

[10] V. Chitrakaran, A. Behal, D. Dawson, and I. Walker, "Setpoint regulation of continuum robots using a fixed camera," *Robotica*, vol. 25, pp. 581–586, 2007.

[11] B. Siciliano and O. Khatib, *Springer Handbook of Robotics*. New York, NY, USA: Springer, 2008.

[12] G. Robinson and J. Davies, "Continuum robots—A state of the art," in *Proc. IEEE Int. Conf. Robot. Autom.*, May. 1999, vol. 4, pp. 2849–2854.

[13] D. Camarillo, C. Milne, C. Carlson, M. Zinn, and J. Salisbury, "Mechanics modeling of tendon-driven continuum manipulators," *IEEE Trans. Robot.*, vol. 24, no. 6, pp. 1262–1273, Dec. 2008.

[14] K. Ikuta, K. Yamamoto, and K. Sasaki, "Development of remote microsurgery robot and new surgical procedure for deep and narrow space," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2003, vol. 1, pp. 1103–1108.

[15] R. Webster, J. Romano, and N. Cowan, "Mechanics of precurved-tube continuum robots," *IEEE Trans. Robot.*, vol. 25, no. 1, pp. 67–78, Feb. 2009.

[16] R. J. Webster and B. A. Jones, "Design and kinematic modeling of constant curvature continuum robots: A review," *Int. J. Robot. Res.*, vol. 29, no. 13, pp. 1661–1683, 2010.

[17] D. C. Rucker and R. J. Webster III, "Parsimonious evaluation of concentric-tube continuum robot equilibrium conformation," *IEEE Trans. Bio-Med. Eng.*, vol. 56, no. 9, pp. 2308–11, Sep. 2009.

[18] D. C. Rucker, B. A. Jones, and R. J. Webster, "A geometrically exact model for externally loaded concentric-tube continuum robots," *IEEE Trans. Robot.*, vol. 26, no. 5, pp. 769–780, Oct. 2010.

[19] D. C. Rucker, R. J. Webster, G. S. Chirikjian, and N. J. Cowan, "Equilibrium conformations of concentric-tube continuum robots," *Int. J. Robot. Res.*, vol. 29, no. 10, pp. 1263–1280, 2010.

[20] D. Rucker and R. W. III, "Statics and dynamics of continuum robots with general tendon routing and external loading," *IEEE Trans. Robot.*, vol. 27, no. 6, pp. 1033–1044, Dec. 2011.

[21] P. E. Dupont, J. Lock, and B. Itkowitz, "Real-time position control of concentric tube robots," in *Proc. IEEE Int. Conf. Robot. Autom.*, May 2010, vol. 2010, pp. 562–568.

[22] P. E. Dupont, J. Lock, B. Itkowitz, and E. Butler, "Design and control of concentric-tube robots," *IEEE Trans. Robot.*, vol. 26, no. 2, pp. 209–225, Apr. 2010.

[23] H. Su, D. Cardona, and W. Shang, "A MRI-guided concentric tube continuum robot with piezoelectric actuation: a feasibility study," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2012, pp. 1939–1945.

[24] B. Jones and I. Walker, "Kinematics for multisection continuum robots," *IEEE Trans. Robot.*, vol. 22, no. 1, pp. 43–55, Feb. 2006.

[25] S. Neppalli, M. A. Csencsits, B. A. Jones, and I. D. Walker, "Closed-Form inverse kinematics for continuum manipulators," *Adv. Robot.*, vol. 23, no. 15, pp. 2077–2091, 2009.

[26] J. Jayender, M. Azizian, and R. V. Patel, "Autonomous robot-assisted active catheter insertion using image guidance," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, Oct. 2007, pp. 889–894.

[27] J. Jayender, M. Azizian, and R. R. V. Patel, "Autonomous image-guided robot-assisted active catheter insertion," *IEEE Trans. Robot.*, vol. 24, no. 4, pp. 858–871, Aug. 2008.

[28] J. Jayender, R. Patel, S. Nikumb, and M. Ostojic, "Modeling and control of shape memory alloy actuators," *IEEE Trans. Control Syst. Technol.*, vol. 16, no. 2, pp. 279–287, Mar. 2008.

[29] J. H. Crews and G. D. Buckner, "Design optimization of a shape memory alloy-actuated robotic catheter," *J. Intell. Mater. Syst. Struct.*, vol. 23, no. 5, pp. 545–562, 2012.

[30] N. Simaan, "Snake-like units using flexible backbones and actuation redundancy for enhanced miniaturization," in *Proc. IEEE Int. Conf. Robot. Autom.*, Apr. 2005, pp. 3023–3028.

[31] N. Simaan, R. Taylor, and P. Flint, "A dexterous system for laryngeal surgery," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2004, pp. 351–357.

[32] K. Xu and N. Simaan, "An investigation of the intrinsic force sensing capabilities of continuum robots," *IEEE Trans. Robot.*, vol. 24, no. 3, pp. 576–587, Jun. 2008.

[33] K. Xu and N. Simaan, "Analytic formulation for kinematics, statics, and shape restoration of multibackbone continuum robots via elliptic integrals," *J. Mechanisms Robot.*, vol. 2, no. 1, p. 011006, 2010.

[34] A. Bajo, S. Member, and N. Simaan, "Kinematics-Based detection and localization of contacts along multisegment continuum robots," *IEEE Trans. Robot.*, vol. 28, no. 2, pp. 291–302, Apr. 2012.

[35] R. Goldman, A. Bajo, and N. Simaan, "Compliant motion control for continuum robots with intrinsic actuation sensing," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2011, pp. 1126–1132.

[36] D. Rucker and R. Webster, "Deflection-based force sensing for continuum robots: A probabilistic approach," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2011, pp. 3764–3769.

[37] K. Xu and N. Simaan, "Actuation compensation for flexible surgical snake-like robots with redundant remote actuation," in *Proc. IEEE Int. Conf. Robot. Autom.*, May 2006, pp. 4148–4154.

[38] M. Grant and S. Boyd. (2013, Sep.). CVX: MATLAB software for disciplined convex programming, version 2.0 beta. [Online]. Available: http://cvxr.com/cvx.

[39] M. Grant and S. Boyd, "Graph implementations for nonsmooth convex programs," in *Recent Advances in Learning and Control (ser. Lecture Notes in Control and Information Sciences.)*. New York, NY, USA: Springer-Verlag, 2008, pp. 95–110.

[40] F. Chaumette and S. Hutchinson, "Visual servo control. I. Basic approaches," *IEEE Robot. Autom. Mag.*, vol. 13, no. 4, pp. 82–90, Dec. 2006.

[41] F. Chaumette and S. Hutchinson, "Visual servo control, Part II: Advanced approaches," *IEEE Robot. Autom. Mag.*, vol. 1, no. 1, pp. 109–118, Mar. 2007.

[42] I. Gravagne and I. Walker, "On the kinematics of remotely-actuated continuum robots," in *Proc. IEEE Int. Conf. Robot. Autom.*, Apr. 2000, pp. 2544–2550.

Michael C. Yip received the B.A.Sc. degree in mechatronics engineering from the University of Waterloo, Waterloo, ON, Canada, and the M.A.Sc. degree in electrical engineering from the University of British Columbia, Vancouver, BC, Canada. He is currently working toward the doctoral degree in bioengineering with Stanford University, Stanford, CA, USA.

During his studies, he worked in various industrial and research positions in the robotics and medical devices field and as a Research Assistant at Harvard University, Cambridge, MA, USA, and the Massachusetts Institute of Technology, Cambridge. His research interests involve medical robotics, robot design and control, image analysis and computer vision, augmented reality, and optimization.

David B. Camarillo (M'14) received the undergraduate degree in mechanical and aerospace engineering from Princeton University, Princeton, NJ, USA, and the Ph.D. degree in mechanical engineering from Stanford University, Stanford, CA, USA.

He is currently an Assistant Professor of Bioengineering with Stanford University. Both his graduate work and industry experience with Intuitive Surgical and Hansen Medical were in the area of surgical robotics. He performed his postdoctoral research in biophysics with the University of California, San Francisco, CA, USA, in 2011. He is an expert in instrumentation and biomechanics whose research interests include medical technology design as it applies to mild traumatic brain injury and flexible robotics for cardiovascular therapy.