

Computing Jacobians and Compliance Matrices for Externally Loaded Continuum Robots

D. Caleb Rucker, *Student Member, IEEE*, and Robert J. Webster III, *Member, IEEE*

Abstract—Kinematic models that account for deformation due to applied loads have recently been developed for a variety of continuum robots. In these cases, a set of nonlinear differential equations with boundary conditions must often be solved to obtain the robot shape. Thus, computing manipulator Jacobians and compliance matrices efficiently is not straightforward. In this paper, we propose a method for obtaining an arc length parametrized Jacobian and compliance matrix. Our approach involves obtaining an augmented Jacobian by propagating the necessary partial derivatives through the model equations, resulting in a new set of differential equations. These equations can be solved as an initial value problem, via a single numerical integration. Our method can be generally applied to various continuum robot architectures, regardless of the specific actuation system used. We provide a specific case study using this method to obtain the Jacobian for a concentric-tube robot.

I. INTRODUCTION

When no external loads are applied, many continuum robot architectures will assume an approximately piecewise constant curvature shape [1], as indicated by modeling results in [2], [3]. In these cases, the robot’s forward kinematic model can be written in closed form and the Jacobian can be derived using geometric arguments (see for example [4], [5], [6], [7], [8]).

However, when the robot’s actuators do not cause it to approximate piecewise constant curvature and/or external loading causes the robot to deform, robot shape is defined by a set of differential equations with boundary conditions. In this paper we start with the existence of a general kinematic model of this form. Models for a variety of continuum robots have been expressed in this way, including cable-driven robots [9], [10], [11], soft pneumatic robots [12], and concentric-tube robots [13], [14]. In all these cases, efficiently obtaining the robot’s Jacobian is challenging, due to the computational complexity of the kinematic model.

Gravagne and Walker have provided a planar formulation for the Jacobian and compliance matrix of an externally loaded continuum robot where the actuators provide a set of discrete or continuous torques along the length [9]. Jones et al. [10] suggested a method for obtaining the manipulator Jacobian for a 3D tendon-actuated robot under external loading, by using a Cosserat-rod-based model and applying finite differences to approximate the Jacobian. Our proposed

method generalizes these approaches to include continuum robots of various architectures and actuation strategies, and provides exact equations to calculate the Jacobian which do not require finite differences.

We provide a specific case study in this paper applying our framework to concentric tube robots [15], [14], [13], which are also known as “active cannulas” due to their potential medical applications. Even without external loads, the kinematics of an active cannula is defined by a nonlinear boundary value problem. Our case study illustrates the ability of the framework described in this paper to obtain the arc length parameterized manipulator Jacobian when an arbitrary cannula design is subject to external loads. It also provides an arc length parameterized compliance matrix mapping displacement of any point along the robot to a point wrench applied at the end effector.

II. PROBLEM STATEMENT

Models for the deformation of continuum robots with actuator values \mathbf{q} and under a six degree-of-freedom point wrench \mathbf{w} can be written in the following form [13], [10], [12]:

$$\begin{aligned} g' &= g\hat{\xi}(\mathbf{y}), \\ \mathbf{y}' &= \mathbf{f}(s, \mathbf{y}, g, \mathbf{q}, \mathbf{w}) \end{aligned} \quad (1)$$

where $g(s) \in \text{SE}(3)$ is a homogeneous transformation defining the backbone location and orientation at arc length s , $\xi \in \mathfrak{se}(3)$ (a body frame twist describing how g evolves in s , see [16]) is a function of a set of variables \mathbf{y} , and the $'$ denotes a derivative with respect to s .

At the base of the robot ($s = 0$), some subset of the elements of \mathbf{y} are unknown, which we denote \mathbf{y}_u . These unknown variables typically include torsional and bending strains. The initial position and orientation g , and the remaining elements of \mathbf{y} , denoted by \mathbf{y}_k , may be specified in terms of \mathbf{y}_u and the actuator positions \mathbf{q} . Thus, the boundary conditions at the proximal end of the robot are:

$$\begin{aligned} g(0) &= H(\mathbf{y}_u(0), \mathbf{q}, \mathbf{w}), \\ \mathbf{y}_k(0) &= \mathbf{h}(\mathbf{y}_u(0), \mathbf{q}, \mathbf{w}). \end{aligned} \quad (2)$$

At the distal end of the robot ($s = \ell$), there are general boundary conditions to be satisfied:

$$\mathbf{b}(\mathbf{y}(\ell), g(\ell), \mathbf{q}, \mathbf{w}) = \mathbf{0}. \quad (3)$$

Shooting methods can be used to solve such boundary value problems (BVP’s). Since \mathbf{b} is a function of the unknown initial conditions \mathbf{y}_u , a shooting method consists of using a nonlinear root-finding algorithm to iteratively converge on

This material is based upon work supported in part by National Science Foundation grant 0651803 and in part by National Institutes of Health grant R21 EB011628 and in part by a Technology Research Grant from Intuitive Surgical.

D. C. Rucker and R. J. Webster III are with Vanderbilt University, Nashville, TN 37235, USA (e-mail: {daniel.c.rucker, robert.webster}@vanderbilt.edu).

values for \mathbf{y}_u which satisfy $\mathbf{b} = \mathbf{0}$. For a particular value of $\mathbf{y}_u(0)$, evaluation of \mathbf{b} simply requires numerical integration of the initial value problem, which can be computed using standard methods such as the Runge-Kutta or Adams-Bashforth families of algorithms.

We are interested in obtaining the spatial manipulator Jacobian, which is defined as follows [16]:

$$J^s = \left[\left(\frac{\partial g}{\partial q_1} g^{-1} \right)^\vee \dots \left(\frac{\partial g}{\partial q_n} g^{-1} \right)^\vee \right].$$

Similarly, we define the spatial manipulator compliance matrix with respect to a tip wrench as,

$$C^s = \left[\left(\frac{\partial g}{\partial w_1} g^{-1} \right)^\vee \dots \left(\frac{\partial g}{\partial w_6} g^{-1} \right)^\vee \right].$$

These matrices can be generalized as continuous functions of arc length, and thus describe the motion of the entire robot with respect to changes in the actuator positions or the components of an applied wrench as,

$$(\dot{g}(s)g^{-1}(s))^\vee = J^s(s, \mathbf{q}, \mathbf{w})\dot{\mathbf{q}} + C^s(s, \mathbf{q}, \mathbf{w})\dot{\mathbf{w}},$$

where the dot denotes a derivative with respect to time.

A straightforward approach to approximating J^s and C^s is to use a finite difference approximation on the BVP to compute each partial derivative. Thus, if $g(s)$ is the solution of the BVP given by (1), (2), and (3), the columns of the Jacobian and compliance matrix are given by

$$\begin{aligned} J_i^s &\approx \left(\frac{g_i(s) - g(s)}{\Delta q_i} g(s)^{-1} \right)^\vee \quad i = 1 \dots n, \\ C_i^s &\approx \left(\frac{g_i(s) - g(s)}{\Delta w_i} g(s)^{-1} \right)^\vee \quad i = 1 \dots 6, \end{aligned} \quad (4)$$

where $g_i(s)$ is the solution of the BVP with $q_i = q_i + \Delta q_i$ or $w_i = w_i + \Delta w_i$ respectively, and n is the number of actuators. Using this method (which we henceforth call BVP finite differences), the manipulator Jacobian and compliance matrix are obtained after $n + 6$ solutions of the boundary value problem. To increase computational efficiency, it is desirable to have a method which uses information from the related initial value problem (IVP) instead. In the next section, we describe such a method. It is based on building an augmented Jacobian matrix for the IVP problem which treats the actuator positions, the elements of the tip wrench, and the unknown initial conditions simultaneously as independent variables.

III. IVP JACOBIANS AND COMPLIANCE MATRICES

In this section, we consider the initial value problem below which is identical to our original boundary value problem given by (1), (2), and (3), except that the initial conditions are now specified for all variables, and the distal boundary

conditions have been removed:

$$\begin{aligned} \mathbf{y}_u(0) &= \mathbf{y}_{u,0} \\ g(0) &= H(\mathbf{y}_u(0), \mathbf{q}, \mathbf{w}) \\ \mathbf{y}_k(0) &= \mathbf{h}(\mathbf{y}_u(0), \mathbf{q}, \mathbf{w}) \\ g' &= \widehat{g\xi}(\mathbf{y}) \\ \mathbf{y}' &= \mathbf{f}(s, \mathbf{y}, g, \mathbf{q}, \mathbf{w}) \end{aligned} \quad (5)$$

We define a Jacobian matrix E with respect to the solution of this IVP, consisting of sub-matrices E_q , E_w and E_u which describe changes in g in with respect to changes in \mathbf{q} , \mathbf{w} , and $\mathbf{y}_u(0)$ respectively as,

$$\begin{aligned} E &= [E_q \quad E_w \quad E_u], \\ E_q &= \left[\left(\frac{\partial g}{\partial q_1} g^{-1} \right)^\vee \dots \left(\frac{\partial g}{\partial q_n} g^{-1} \right)^\vee \right], \\ E_w &= \left[\left(\frac{\partial g}{\partial w_1} g^{-1} \right)^\vee \dots \left(\frac{\partial g}{\partial w_6} g^{-1} \right)^\vee \right], \\ E_u &= \left[\left(\frac{\partial g}{\partial y_{u,1}(0)} g^{-1} \right)^\vee \dots \left(\frac{\partial g}{\partial y_{u,m(0)} g^{-1}} \right)^\vee \right]. \end{aligned}$$

Similarly, we construct matrices B_q , B_w , and B_u which describe the change in the function \mathbf{b} defined in (3) as,

$$\begin{aligned} B &= [B_q \quad B_w \quad B_u], \\ B_q &= \frac{\partial \mathbf{b}}{\partial \mathbf{q}}, \quad B_w = \frac{\partial \mathbf{b}}{\partial \mathbf{w}}, \quad B_u = \frac{\partial \mathbf{b}}{\partial \mathbf{y}_u(0)}. \end{aligned}$$

Using the above definitions, one can write down the time derivative of the robot pose, as well as the rate of change of the boundary condition function in terms of the time derivatives of \mathbf{q} , \mathbf{w} , and $\mathbf{y}_u(0)$ as,

$$(\dot{g}g^{-1})^\vee = E_q\dot{\mathbf{q}} + E_w\dot{\mathbf{w}} + E_u\dot{\mathbf{y}}_u(0) \quad (6)$$

$$\dot{\mathbf{b}} = B_q\dot{\mathbf{q}} + B_w\dot{\mathbf{w}} + B_u\dot{\mathbf{y}}_u(0). \quad (7)$$

Now, if the original boundary value problem has been solved, we want \mathbf{b} to remain at zero for any $\dot{\mathbf{q}}$ and $\dot{\mathbf{w}}$. Thus, if we set $\dot{\mathbf{b}} = \mathbf{0}$ in (7), we can solve for the $\dot{\mathbf{y}}_u(0)$ which will continue to satisfy the boundary conditions, namely,

$$\dot{\mathbf{y}}_u(0) = -B_u^\dagger (B_q\dot{\mathbf{q}} + B_w\dot{\mathbf{w}})$$

Substituting the result into (6) yields

$$(\dot{g}g^{-1})^\vee = (E_q - E_u B_u^\dagger B_q) \dot{\mathbf{q}} + (E_w - E_u B_u^\dagger B_w) \dot{\mathbf{w}},$$

from which we can see that the actual manipulator Jacobian and compliance matrix (as continuous functions of s) can be expressed in terms of the initial value Jacobians as follows,

$$J^s(s) = E_q(s) - E_u(s) B_u^{-1} B_q \quad (8)$$

$$C^s(s) = E_w(s) - E_u(s) B_u^{-1} B_w. \quad (9)$$

In the following sections we discuss two ways of computing E and B : approximation by finite differences, and derivation of differential equations that define E and B exactly.

IV. IVP MATRICES VIA FINITE DIFFERENCES

If $g(s)$ is the solution of the BVP, It is straightforward to approximate the columns of E and B matrices from a finite difference method on the IVP as follows:

$$\begin{aligned} E_{q,i} &\approx \left(\frac{g_i(s) - g(s)}{\Delta q_i} g^{-1}(s) \right)^\vee, & B_{q,i} &\approx \frac{\mathbf{b}_i - \mathbf{b}}{\Delta q_i} \\ E_{w,i} &\approx \left(\frac{g_i(s) - g(s)}{\Delta w_i} g^{-1}(s) \right)^\vee, & B_{w,i} &\approx \frac{\mathbf{b}_i - \mathbf{b}}{\Delta w_i} \\ E_{u,i} &\approx \left(\frac{g_i(s) - g(s)}{\Delta y_{u,i}(0)} g^{-1}(s) \right)^\vee, & B_{u,i} &\approx \frac{\mathbf{b}_i - \mathbf{b}}{\Delta y_{u,i}(0)}, \end{aligned}$$

where $g_i(s)$ is the solution of the BVP with $q_i = q_i + \Delta q_i$, $w_i = w_i + \Delta w_i$, or $y_{u,i} = y_{u,i} + \Delta y_{u,i}$ respectively. With this method (which we call IVP finite differences), the Jacobian and compliance matrices are obtained after $n+m+6$ numerical integrations of the IVP problem defined in (5), where m is the number of elements in $\mathbf{y}_u(0)$. This requires much less computation than the finite difference BVP algorithm in (4) because when using a shooting method, each BVP solution usually requires m IVP integrations to obtain the gradients and move one step. Thus, even if the shooting method converges in a single step (which is not guaranteed), the number of IVP integrations required by implementation of (4) is $(m)(n+6)$.

Next, we propose a method which is not based on finite difference approximations, but on deriving a new set of differential equations which define E and B exactly. Using this method, E and B are obtained after one integration of an IVP (albeit a much larger one than the one required for forward kinematics). Our simulation results in Section VI show that this method provides an improvement in efficiency over both BVP and IVP finite differencing, although it is not as easily parallelizable as the IVP finite differencing approach.

V. DERIVATIVE PROPAGATION APPROACH

As a brief aside to illustrate our basic approach in derivative propagation, consider a simple example IVP,

$$\mathbf{y}(0) = \mathbf{h}(\mathbf{q}), \quad \mathbf{y}' = \mathbf{f}(s, \mathbf{y}), \quad (10)$$

and suppose that we are interested in obtaining a matrix V describing how \mathbf{y} changes with respect to \mathbf{q} , namely,

$$V = \frac{\partial \mathbf{y}}{\partial \mathbf{q}}.$$

We can differentiate V with respect to s , and use (10) to obtain

$$\frac{dV}{ds} = \frac{d}{ds} \left(\frac{\partial \mathbf{y}}{\partial \mathbf{q}} \right) = \frac{\partial}{\partial \mathbf{q}} \left(\frac{d\mathbf{y}}{ds} \right) = \frac{\partial \mathbf{f}}{\partial \mathbf{q}} = \frac{\partial \mathbf{f}}{\partial \mathbf{y}} \frac{\partial \mathbf{y}}{\partial \mathbf{q}}.$$

Then, taking the partial derivative of the initial conditions, we obtain an IVP that defines V

$$V(0) = \frac{\partial \mathbf{h}}{\partial \mathbf{q}}, \quad \frac{dV}{ds} = \frac{\partial \mathbf{f}}{\partial \mathbf{y}} V \quad (11)$$

The above technique is often used in the implementation of shooting methods to supply derivative information for the

update step, and we will adapt this technique in the subsection below to obtain E and B for the more complicated IVP defined in (5).

A. IVP Matrices Via Derivative Propagation

We wish to obtain a differential equation which defines the matrix E . To derive this, we first consider the i^{th} column of the matrix E_q ,

$$E_{q,i} = \left(\frac{\partial g}{\partial q_i} g^{-1} \right)^\vee. \quad (12)$$

We can derive an expression for its arc length derivative as follows:

$$\begin{aligned} \frac{d}{ds} E_{q,i} &= \left(\frac{\partial}{\partial q_i} \left(\frac{dg}{ds} \right) g^{-1} + \frac{\partial g}{\partial q_i} \frac{d}{ds} (g^{-1}) \right)^\vee \\ &= \left(\frac{\partial g}{\partial q_i} \widehat{\boldsymbol{\xi}} g^{-1} + g \frac{\partial \widehat{\boldsymbol{\xi}}}{\partial q_i} g^{-1} + \frac{\partial g}{\partial q_i} \frac{d}{ds} (g^{-1}) \right)^\vee \\ &= \left(g \frac{\partial \widehat{\boldsymbol{\xi}}}{\partial q_i} g^{-1} \right)^\vee = Ad_g \frac{\partial \boldsymbol{\xi}}{\partial q_i}, \end{aligned}$$

where we have used $g' = g\widehat{\boldsymbol{\xi}}$, and the fact that

$$\frac{d}{ds} (g^{-1}) = -\widehat{\boldsymbol{\xi}} g^{-1}.$$

Ad_g is the adjoint transformation of g , defined as

$$Ad_g = \begin{bmatrix} R & \widehat{\mathbf{p}}R \\ \mathbf{0} & R \end{bmatrix}.$$

One can similarly obtain expressions for the columns of E_w and E_u . Thus, the derivative of E is given by,

$$E' = Ad_g \begin{bmatrix} \frac{\partial \boldsymbol{\xi}}{\partial \mathbf{q}} & \frac{\partial \boldsymbol{\xi}}{\partial \mathbf{w}} & \frac{\partial \boldsymbol{\xi}}{\partial \mathbf{y}_u(0)} \end{bmatrix} = Ad_g \frac{\partial \boldsymbol{\xi}}{\partial \mathbf{y}} V, \quad (13)$$

where,

$$\begin{aligned} V &= [V_q \quad V_w \quad V_u], \\ V_q &= \frac{\partial \mathbf{y}}{\partial \mathbf{q}}, \quad V_w = \frac{\partial \mathbf{y}}{\partial \mathbf{w}}, \quad V_u = \frac{\partial \mathbf{y}}{\partial \mathbf{y}_u(0)}. \end{aligned}$$

The columns of B can be expressed as

$$\begin{aligned} B'_{q,i} &= \frac{\partial \mathbf{b}(\cdot)}{\partial \mathbf{y}} V_{q,i} + \frac{\partial \mathbf{b}(\cdot)}{\partial \text{vec}(g)} \text{vec} \left(\widehat{E}_{q,i} g \right) + \frac{\partial \mathbf{b}(\cdot)}{\partial q_i}, \\ B'_{w,i} &= \frac{\partial \mathbf{b}(\cdot)}{\partial \mathbf{y}} V_{w,i} + \frac{\partial \mathbf{b}(\cdot)}{\partial \text{vec}(g)} \text{vec} \left(\widehat{E}_{w,i} g \right) + \frac{\partial \mathbf{b}(\cdot)}{\partial w_i}, \\ B'_{u,i} &= \frac{\partial \mathbf{b}(\cdot)}{\partial \mathbf{y}} V_{u,i} + \frac{\partial \mathbf{b}(\cdot)}{\partial \text{vec}(g)} \text{vec} \left(\widehat{E}_{u,i} g \right), \end{aligned} \quad (14)$$

where $\text{vec}(\cdot)$ simply reshapes a matrix into a column vector, and we use $\partial \mathbf{b}(\cdot)$ to denote the partial derivative of the function $\mathbf{b}(\mathbf{y}(\ell), g(\ell), \mathbf{q}, \mathbf{w})$ with respect to the argument appearing in the given term's denominator. This is necessary because \mathbf{q} and \mathbf{w} affect the value of \mathbf{b} both directly and also indirectly through changes in $\mathbf{y}(\ell)$ and $g(\ell)$.

We now only need an equation for V to have a complete set of equations defining E and B . The necessary procedure

is analogous to (11). For the i^{th} columns of V_q , V_w , and V_u we have

$$\begin{aligned} V'_{q,i} &= \frac{\partial \mathbf{f}(\cdot)}{\partial \mathbf{y}} V_{q,i} + \frac{\partial \mathbf{f}(\cdot)}{\partial \text{vec}(g)} \text{vec}(\widehat{E}_{q,i}g) + \frac{\partial \mathbf{f}(\cdot)}{\partial q_i}, \\ V'_{w,i} &= \frac{\partial \mathbf{f}(\cdot)}{\partial \mathbf{y}} V_{w,i} + \frac{\partial \mathbf{f}(\cdot)}{\partial \text{vec}(g)} \text{vec}(\widehat{E}_{w,i}g) + \frac{\partial \mathbf{f}(\cdot)}{\partial w_i}, \\ V'_{u,i} &= \frac{\partial \mathbf{f}(\cdot)}{\partial \mathbf{y}} V_{u,i} + \frac{\partial \mathbf{f}(\cdot)}{\partial \text{vec}(g)} \text{vec}(\widehat{E}_{u,i}g). \end{aligned} \quad (15)$$

where we have used (12) to substitute in for $\frac{\partial g}{\partial q_i}$. At $s = 0$ it is straightforward to calculate the $V(0)$ and $E(0)$ from the initial conditions.

$$\begin{aligned} V(0) &= \begin{bmatrix} \frac{\partial \mathbf{y}(0)}{\partial \mathbf{q}} & \frac{\partial \mathbf{y}(0)}{\partial \mathbf{w}} & \frac{\partial \mathbf{y}(0)}{\partial \mathbf{y}_u(0)} \end{bmatrix} \\ E_{q,i}(0) &= \left(\frac{\partial H}{\partial q_i} g(0)^{-1} \right)^\vee, \\ E_{w,i}(0) &= \left(\frac{\partial H}{\partial w_i} g(0)^{-1} \right)^\vee, \\ E_{u,i}(0) &= \left(\frac{\partial H}{\partial y_{u,i}(0)} g(0)^{-1} \right)^\vee \end{aligned} \quad (16)$$

We now have an IVP, consisting of (13), (14), (15), and (16), which when integrated simultaneously with the equations in (5), yields the initial value matrices E and B from which we can obtain J^s and C^s using (8) and (9). Furthermore, we obtain the matrix B_u , which we can use in our shooting method to solve the boundary value problem efficiently.

VI. CASE STUDY: A CONCENTRIC-TUBE ROBOT

In this section we apply the methods derived above to obtain a continuous set of Jacobians and compliance matrices along the length of a concentric-tube robot, or active cannula. A model for the statics of an active cannula under external loads has been derived [13], and we give a brief, self-contained summary of that model here before applying the methods of Section V.

We first assign homogeneous reference frames along each precurved tube, consisting of a position vector $\mathbf{r}_i^*(s)$ and rotation matrix $R_i^*(s)$. By convention, the frame is chosen so that the z -axis of $R_i^*(s)$ remains tangent to the curve such that,

$$g_i^*(s) = \begin{bmatrix} R_i^*(s) & \mathbf{r}_i^*(s) \\ \mathbf{0}^\top & 1 \end{bmatrix}.$$

The precurvature vector of each tube is then given by

$$\mathbf{u}_i^*(s) = \left(R_i^{*T}(s) \dot{R}_i^*(s) \right)^\vee.$$

The cross sectional geometry and material properties of each tube give rise to a stiffness matrix,

$$K_i = \begin{bmatrix} E_i I_i & 0 & 0 \\ 0 & E_i I_i & 0 \\ 0 & 0 & G_i J_i \end{bmatrix},$$

where E_i is Young's modulus, I_i is the second moment of area of the tube cross section, G_i is the shear modulus, and J_i is the polar moment of inertia of the tube cross section.

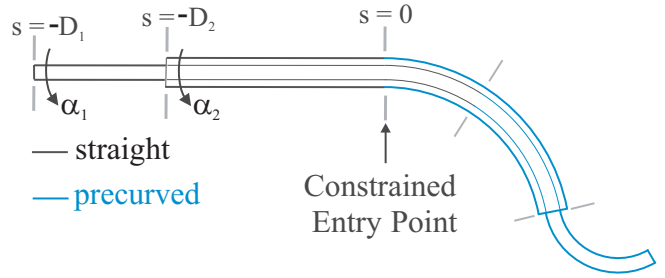


Fig. 1: Diagram of a two tube cannula showing transition points where continuity of shape and internal moment must be enforced. The constrained point of entry into the workspace is designated as the arc length zero position.

In the general model of [13], these are allowed to vary with the arc length of the tube, but for simplicity of exposition, here we will assume that they are constant.

A Cosserat-rod approach is then used to derive a set of differential equations governing the rod shape:

$$\begin{aligned} g' &= \widehat{g\xi} \\ \theta'_i &= u_{i,z} - u_{1,z} \\ \begin{bmatrix} u'_{1,x} \\ u'_{1,y} \end{bmatrix} &= -K^{-1} \left(\sum_{i=1}^n R_{\theta_i} \left(K_i (\theta'_i \frac{dR_{\theta_i}^T}{d\theta_i} \mathbf{u}_1 - \dot{\mathbf{u}}_i^*) \right. \right. \\ &\quad \left. \left. + \widehat{\mathbf{u}}_i K_i (\mathbf{u}_i - \mathbf{u}_i^*) \right) + \widehat{\mathbf{e}}_3 R_1^T \mathbf{w}_f \right) \Big|_{x,y} \\ u'_{i,z} &= u_{i,z}^* + \frac{E_i I_i}{G_i J_i} (u_{i,x} u_{i,y}^* - u_{i,y} u_{i,x}^*). \end{aligned} \quad (17)$$

where θ_i defines the axial angle of the i^{th} tube relative to the tube taken as a reference, \mathbf{w}_f is the force in the applied tip wrench, $K = \sum_{i=1}^n K_i$, and R_{θ_i} , \mathbf{u}_i , and ξ are expressed using the state variables $\mathbf{y} = [\theta_2, \dots, \theta_n, u_{1,x}, u_{1,y}, u_{1,z}, u_{2,y}, \dots, u_{n,z}]$ as follows:

$$\begin{aligned} R_{\theta_i} &= \begin{bmatrix} \cos \theta_i & -\sin \theta_i & 0 \\ \sin \theta_i & \cos \theta_i & 0 \\ 0 & 0 & 1 \end{bmatrix}, \\ \mathbf{u}_i &= R_{\theta_i}^\top \mathbf{u}_1 + \theta'_i \mathbf{e}_3, \quad \xi = [0 \ 0 \ 1 \ \mathbf{u}_1^\top]^\top. \end{aligned}$$

We can see that (17) is of the form given in (1). At $s = 0$, the unknown variables are $\mathbf{y}_u = [u_{1,x}, u_{1,y}, u_{1,z}, u_{2,y}, \dots, u_{n,z}]$. As in (2), the rest of the initial variables and the initial pose can be specified in terms of these unknown quantities and the actuator values as follows:

$$\begin{aligned} \mathbf{r}(0) &= [0 \ 0 \ 0]^\top, \\ R(0) &= \begin{bmatrix} \cos(\alpha_1 + D_1 u_{1,z}) & -\sin(\alpha_1 + D_1 u_{1,z}) & 0 \\ \sin(\alpha_1 + D_1 u_{1,z}) & \cos(\alpha_1 + D_1 u_{1,z}) & 0 \\ 0 & 0 & 1 \end{bmatrix}, \\ \theta_i &= (\alpha_i + D_i u_{i,z}) - (\alpha_1 + D_1 u_{1,z}), \end{aligned}$$

where the actuator value vector \mathbf{q} consists of the rotations and translations of each tube, $[\alpha_1, \dots, \alpha_n, D_1, \dots, D_n]$, shown in Fig. 1.

In the common case that the innermost tube (tube 1) extends further than the other tubes, the internal moment at its tip must be equal to the moment of the externally

applied wrench. At the end of each of the other tubes, the axial component of the internal moment must be zero. This leads to the following boundary condition functions:

$$\begin{aligned} \mathbf{u}_1(\ell_1) - \mathbf{u}_1^*(\ell_1) - K_1^{-1} R_1^T \mathbf{w}_m &= \mathbf{0} \\ u_{i,z}(\ell_i) - u_{i,z}^*(\ell_i) &= 0, \end{aligned}$$

where ℓ_i is the arc length where tube i ends.

At arc length points where a tube undergoes a step change in curvature or ends, the total internal moment must be continuous. This can lead to a discontinuous solution for \mathbf{u}_1 , and $u_{i,z}$, and there are transition conditions which algebraically relate the variables on either side of the transition point. They are

$$\begin{aligned} \mathbf{u}_1^+ &= K^{-1+} \left(K^- \mathbf{u}_1^- + \sum_{i=1}^n R_{\theta_i} \left(K_i^+ (\mathbf{u}_i^{*+} - \theta_i'^+ \mathbf{e}_3) \right. \right. \\ &\quad \left. \left. - K_i^- (\mathbf{u}_i^{*-} - \theta_i'^- \mathbf{e}_3) \right) \right) \\ u_{i,z}^+ &= u_{i,z}^- + u_{i,z}^{*+} - u_{i,z}^{*-}, \end{aligned}$$

where $+$ and $-$ denote the value on the distal and proximal side of the transition point. For general transition conditions of the form $\mathbf{y}^+ = \mathbf{f}_t(\mathbf{y}^-)$, the transition conditions for V can be written using the chain rule. For instance:

$$V_q^+ = \frac{\partial \mathbf{f}_t}{\partial \mathbf{y}} V_q^-.$$

In the case of the concentric tube model, the arc length location where the transition occurs may also be dependent on D_i . At these locations, the columns V_{D_i} and J_{D_i} transition across the boundary as follows:

$$\begin{aligned} V_{D_i}^+ &= \frac{\partial \mathbf{f}_t}{\partial \mathbf{y}} (V_{D_i}^- - \mathbf{f}(s, \mathbf{y}^-, g, \mathbf{w})) + \mathbf{f}(s, \mathbf{y}^+, g, \mathbf{w}) \\ J_{D_i}^+ &= J_{D_i}^- - Ad_g \xi^- + Ad_g \xi^+. \end{aligned}$$

A. Simulations Evaluating Computational Efficiency

We now provide simulation results using the model above for a concentric-tube robot with three tubes. Each tube has a straight section at its proximal end followed by a distal section with constant curvature. The diameters, lengths, curvatures, and mechanical properties of the tubes used in our simulation are provided in Table I.

We compare the computational efficiency of the three approaches in this paper in Table II. The Jacobian was calculated at 100 evenly distributed configuration-space points

TABLE I: Physical Parameters for Tubes used in Simulations

	Tube 1 (Inner)	Tube 2 (Middle)	Tube 3 (Outer)
Inner Diameter (mm)	0.50	1.25	2.00
Outer Diameter (mm)	1.00	1.75	2.50
Straight Length (mm)	450	250	100
Curved Length (mm)	150	150	100
Curvature (m^{-1})	20	10	5
Young's Mod. (E) (GPa)	60	60	60
Shear Mod. (J) (GPa)	23.1	23.1	23.1

TABLE II: Computation Times for Proposed Methods

Integration Scheme	BVP Finite Differences	IVP Finite Differences	Derivative Propagation
RK 4/5	2.594 s	0.457 s	0.137 s
RK 2/3	1.494 s	0.251 s	0.076 s
AB 4	0.940 s	0.161 s	0.040 s

using three numerical integration schemes with a fixed arc length step size of 0.01 m, a fourth/fifth and a second/third order Runge-Kutta scheme and a fourth order Adams-Bashforth multi-step method. All were implemented in MATLAB, running in the Windows operating system, on a laptop computer with a 2.8 GHz dual core processor. The derivative propagation approach shows a clear advantage over the other methods.

B. Resolved Rates Servoing Simulation

We used the derivative propagation approach to solve the inverse kinematics problem via resolved rate servoing, commanding tip velocities in the desired direction. We commanded the tip of the cannula to trace the outline of the letter "V", as shown in Fig. 2. To illustrate the correctness of the Jacobian when the robot is experiencing an external load, we performed the same simulation with a 0.4 N downward force applied to the tip of the cannula. Results of this simulation are shown in Fig. 3. Intuition suggests that the body of the robot will have to extend further upward in order to compensate for the load at its tip, and this behavior is visible in the figure.

VII. CONCLUSIONS

In this paper we have described a method for computing arc length parameterized Jacobians and compliance matrices

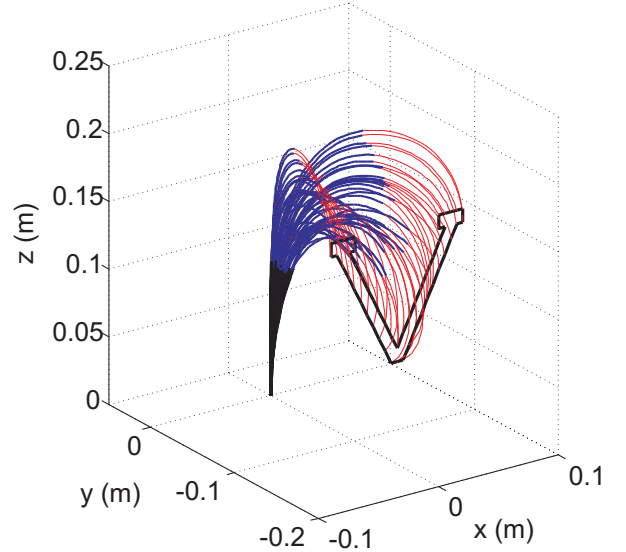


Fig. 2: The three tube active cannula described in Table I traces the outline of the letter "V". The inverse kinematic solution is achieved by computing the Jacobian and compliance matrix using the proposed approach. The inner, middle, and outer tubes are shown in red, blue, and black respectively.

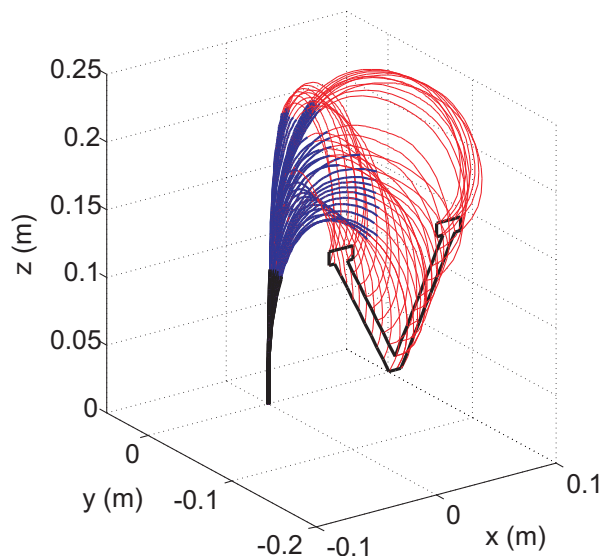


Fig. 3: The three tube active cannula described in Table I traces the outline of the letter “V” while under a constant tip force of 0.4 N in the negative z-direction. The inverse kinematic solution is achieved by computing the Jacobian and compliance matrix using the proposed approach. The inner, middle, and outer tubes are shown in red, blue, and black respectively.

for continuum robots described by differential equations with boundary conditions. These differential equations and boundary conditions arise in models of continuum robots that do not assume piecewise constant curvature in response to actuation and/or are subject to externally applied loads that cause appreciable deformation.

Our original boundary value problem defined in (1), (2), and (3) allows both the actuator actions and the external wrench to influence every aspect of the model, including the initial conditions, the governing equations, and the boundary conditions. For this reason, the approach developed in Section V is applicable to a wide variety of continuum robot architectures and actuation strategies including tendon, pneumatic, and concentric tube designs.

We foresee many future uses for these Jacobians and compliance matrices. The compliance matrix will be useful for force sensing, force control, and analysis during the design process and use of a given robot. Similarly, the Jacobian will enable evaluation of manipulability. It will also enable rapid inverse kinematics, control of any desired point along the robot, control of multiple points simultaneously, or in the limit as the number of points increases, shape control.

This is useful for teleoperation, visual servoing, and many other applications. We anticipate that the ability to account for external loads and variable curvature along the length of the robot when calculating Jacobians and compliance matrices will facilitate new practical applications for continuum robots, particularly those where interaction with objects in the environment is required.

REFERENCES

- [1] R. J. Webster III and B. A. Jones, “Design and kinematic modeling of constant curvature continuum robots: A review,” *International Journal of Robotics Research*, vol. 29, no. 13, pp. 1661–1683, 2010.
- [2] K. Xu and N. Simaan, “Analytic formulation for kinematics, statics and shape restoration of multi-backbone continuum robots via elliptic integrals,” *ASME Journal of Mechanisms and Robotics*, vol. 2, no. 1, pp. 011 006–1 to 011 006–13, 2010.
- [3] C. Li and C. D. Rahn, “Design of continuous backbone, cable-driven robots,” *ASME Journal of Mechanical Design*, vol. 124, no. 2, pp. 265–271, 2002.
- [4] B. A. Jones and I. D. Walker, “Kinematics for multisection continuum robots,” *IEEE Transactions on Robotics*, vol. 22, no. 1, pp. 43–55, 2006.
- [5] M. W. Hannan and I. D. Walker, “Kinematics and the implementation of an elephant’s trunk manipulator and other continuum style robots,” *Journal of Robotic Systems*, vol. 20, no. 2, pp. 45–63, 2003.
- [6] R. J. Webster III, J. P. Swenson, J. M. Romano, and N. J. Cowan, “Closed-form differential kinematics for concentric-tube continuum robots with application to visual servoing,” *Springer Tracts in Advanced Robotics*, vol. 54, pp. 485–494, 2009.
- [7] Y. Bailly and Y. Amirat, “Modeling and control of a hybrid continuum active catheter for aortic aneurysm treatment,” *IEEE International Conference on Robotics and Automation*, pp. 924–929, 2005.
- [8] K. Xu and N. Simaan, “Intrinsic wrench estimation and its performance index for multisegment continuum robots,” *IEEE Transactions on Robotics*, vol. 26, no. 3, pp. 555–561, 2010.
- [9] I. A. Gravagne and I. D. Walker, “Manipulability, force, and compliance analysis for planar continuum robots,” *IEEE Transactions on Robotics and Automation*, vol. 18, no. 3, pp. 263–273, June 2002.
- [10] B. Jones, R. Gray, and K. Turlapati, “Three dimensional statics for continuum robotics,” *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 2659–2664, 2009.
- [11] D. C. Rucker and R. J. Webster III, “Exact mechanics of continuum robots with general tendon routing,” *International Symposium on Experimental Robotics*, 2010, (In Press).
- [12] D. Trivedi, A. Lotfi, and C. Rahn, “Geometrically exact models for soft robotic manipulators,” *IEEE Transactions on Robotics*, vol. 24, pp. 773 – 780, 2008.
- [13] D. C. Rucker, B. A. Jones, and R. J. Webster III, “A geometrically exact model for externally loaded concentric-tube continuum robots,” *IEEE Transactions on Robotics*, vol. 26, no. 5, pp. 769–780, 2010.
- [14] P. E. Dupont, J. Lock, B. Itkowitz, and E. Butler., “Design and control of concentric-tube robots,” *IEEE Transactions on Robotics*, vol. 26, pp. 209–225, 2010.
- [15] D. C. Rucker, R. J. Webster III, G. S. Chirikjian, and N. J. Cowan, “Equilibrium conformations of concentric-tube continuum robots,” *International Journal of Robotics Research*, vol. 29, no. 10, pp. 1263–1280, 2010.
- [16] R. M. Murray, Z. Li, and S. S. Sastry, *A Mathematical Introduction to Robotic Manipulation*. Boca Raton, FL: CRC Press, 1994.