Proceedings of the ASME 2019
Dynamic Systems and Control Conference
DSCC2019
October 8-11, 2019, Park City, Utah, USA

DSCC2019-8999

A NEW APPROACH TO MODEL CONSTANT CURVATURE CONTINUUM ROBOT DYNAMICS

Yujiong Liu, Pinhas Ben-Tzvi*
Robotics and Mechatronics Lab
Department of Mechanical Engineering
Virginia Tech
Blacksburg, VA 24060

ABSTRACT

Inspired by nature, continuum robots show their potential in human-centered environments due to the compliant-to-obstacle features and dexterous mobility. However, there are few such robots successfully implemented outside the laboratory so far. One reason is believed to be due to the real time control challenge for soft robots, which require a highly efficient, highly accurate dynamic model. This paper presents a new systematic methodology to formulate the dynamics of constant curvature continuum robots. The new approach builds on several new techniques: 1) using the virtual work principle to formulate the equation of motion, 2) using specifically selected kinematic representations to separate integral variables from the non-integral variables, and 3) using vector representations to put the integral in a compact form. By doing so, the hard-to-solve integrals are evaluated analytically in advance and the accurate inverse dynamics are established accordingly. Numerical simulations are conducted to evaluate the performances of the newly proposed model.

NOMENCLATURE

$\delta\mathbf{x}$	Virtual displacement of $\mathbf{x}$
$\mathbf{v}_i$, $\boldsymbol{\omega}_i$	Linear and angular velocity
$\mathbf{I}_i$	Moment of inertia
$\mathbf{J}_i$	Jacobian matrix
$\hat{\boldsymbol{\omega}}$	Skew symmetric matrix of vector $\boldsymbol{\omega}$
$\boldsymbol{\tau}_i$	Generalized force
R	Cross section radius
M_i	Mass of the ith segment
θ_i	Bending angle of the ith segment in curvature plane
φ_i	Rotation of the ith segment curvature plane
r_i	Curvature radius of the ith segment
κ_i	ith segment curvature
s_x, c_x	Convenient representations for $\sin\theta_i x$ and $\cos\theta_i x$ respectively
x	Integral variable ranging from 0 to 1
E	Elastic module
K	Bending stiffness
$L_{i,0}$	Unloaded length of the ith segment
$\mathbf{H}$	System inertia matrix

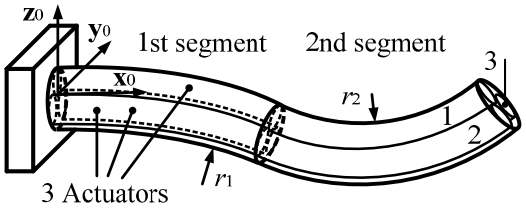

Figure 1: Illustration of a two-segment constant curvature continuum robot with three intrinsic actuators in each segment

1 INTRODUCTION

In last two decades, continuum robots drew increased interest from the robotics research community since these soft structures provide promising solutions for human-centered robotic applications due to their compliant-to-obstacle features and their dexterous mobility. A considerable amount of research was carried out since then. Practical implementations include the typical continuum elephant's trunk manipulator [1], the concentric tube continuum robot [2], the continuum robotic tails [3-5], etc. Theoretical research involves kinematic modeling [6], dynamic modeling [7], trajectory planning, and control [8]. More comprehensive reviews could be found in [9] and [10].

However, limited by the lack of highly efficient and accurate dynamic algorithm, real time control of most current continuum robots is still a challenge. Significant effort has been made to solve this problem which can be roughly categorized into two classes: the distributed parameter model and the lumped parameter model. Distributed parameter models [2, 7, 11, 12] are thought to have high accuracy due to their

* Corresponding author: bentzvi@vt.edu

considerations of the robots' unconstructed shapes. However, this approach usually generates large partial differential equations (PDE) and can be only solved offline. Lumped parameter models [13-16] improve the computational efficiency by assuming the material deformation rule in advance. This approach usually generates online solvable ordinary differential equations (ODE) but loses some accuracy. Therefore, to balance the accuracy and efficiency, choosing appropriate lumped parameter assumptions and formulating the corresponding dynamic model become a focus of this research.

For lumped parameter models, constant curvature is the most widely used deformation assumption (a two-segment example is illustrated in Fig. 1). Although effective, one significant drawback of this assumption is that it does not account for the curvature variance induced by gravity. To compensate for this drawback, Rone and Ben-Tzvi [13] divided the robot into multiple sub-segments. However, this approach did not account for the inertia loading of the continuum backbone, which turns out a hard-to-solve integral problem faced by most lumped parameter models. To solve this problem, Taylor serials approximation [14] and energy approximation [17] were used to evaluate the dynamics numerically.

By observing the integral challenge, the difficulty is believed to be due to the mixture of the integral variable and other non-integral kinematic terms. Therefore, if a mathematical representation can be found such that the integral variable can be isolated from the non-integral variables, then the integral might be evaluated analytically and the dynamics could be simplified significantly.

Following this idea, this paper aims to develop a new approach to solve the integral problem that appears in modeling constant curvature continuum robot dynamics. The core idea is to separate the integral variable from the non-integral variables in the kinematic modeling stage, which requires looking for good kinematic representations for the linear and angular terms. Fortunately, for linear kinematic terms, a simple vector representation is found and the vector representation turns out to be useful and simplifies the derivation significantly. By doing so, the integral variable is separated from the non-integral variables, which allows the integral to be evaluated analytically.

Note that the work in this paper has similar idea as in [16], which utilized Frenet frame representation to separate variables and evaluates the integral analytically. However, [16] focuses on a planar case and is based on Lagrange's method, which requires further differentiation of the terms $\partial L/\partial \dot{q}_i$. Instead, this paper utilizes the vectors directly associated with the integral variable to separate the integral variables and applies the virtual work principle to "avoid" this second stage differentiation (theoretically, the second differentiation cannot be avoided). But in virtual work based formulations [18, 19], the second differentiation appears explicitly as physically meaningful accelerations which make the method suitable for numerical computations.

In summary, this paper focuses on solving the integral challenge encountered in modeling constant curvature continuum robot dynamics. The major contributions are as follows: (1) the analytic integral of the continuum rod inertia forces was successfully evaluated by using separation of variables method (more specifically, writing kinematic terms as linear functions of the integral variable). Thus, there is no need for any functional approximation techniques (modal approach, Taylor expansion etc.); (2) an analytic dynamic model for constant curvature continuum robot is established by the virtual work principle, which is different from existing Lagrangian based or numerical approximation based approaches; (3) since the new method keeps the physical meaning of the kinematic terms, a recursive algorithm is proposed to compute the dynamic model.

The rest of this paper is organized as follows. Section 2 formulates the dynamic framework based on the virtual work principle. Section 3 evaluates the loading integrals using the separation of variables method and substantiates each term in the dynamic framework. Section 4 presents the corresponding numerical algorithm to compute the dynamics derived in sections 2 and 3. Section 5 conducts a preliminary numerical experiment to verify the dynamic model.

2 ESTABLISHING THE EQUATION OF MOTION USING VIRTUAL WORK PRINCIPLE

The virtual work principle provides another effective way to establish the equations of motion based on the d'Alembert principle, besides the classic Newton-Euler and Lagrangian based formulations. For a N rigid body system, the virtual work principle states that the total virtual work of the system should be always zero, for which the mathematical expression is given in Eq. (1). $\mathbf{F}_i$ and $\mathbf{M}_i$ are the active forces and moments on body i, respectively, and $\mathbf{I}_i$ is the inertia matrix of body i. $\mathbf{v}_i$, $\boldsymbol{\omega}_i$ are the linear and angular velocity, respectively.

$$\sum_{i=1}^{N} [\delta \mathbf{x}_i^T (\mathbf{F}_i - m\dot{\mathbf{v}}_i) + \delta \boldsymbol{\theta}_i^T (\mathbf{M}_i - \mathbf{I}_i \dot{\boldsymbol{\omega}}_i - \widehat{\boldsymbol{\omega}}_i \mathbf{I}_i \boldsymbol{\omega}_i)] = 0 \qquad (1)$$

By expressing the virtual displacements $\delta \mathbf{x}_i$ and $\delta \boldsymbol{\theta}_i$ by the generalized coordinates $\mathbf{q}$ ($\delta \mathbf{x}_i = \mathbf{J}_{i,v} \delta \mathbf{q}$, $\delta \boldsymbol{\theta}_i = \mathbf{J}_{i,\omega} \delta \mathbf{q}$), Eq. (1) can be written as follows

$$\sum_{i=1}^{N} (\mathbf{J}_{i,v}^T \mathbf{F}_i + \mathbf{J}_{i,\omega}^T \mathbf{M}_i) = \sum_{i=1}^{N} \left[\mathbf{J}_{i,v}^T m\dot{\mathbf{v}}_i + \mathbf{J}_{i,\omega}^T (\mathbf{I}_i \dot{\boldsymbol{\omega}}_i + \widehat{\boldsymbol{\omega}}_i \mathbf{I}_i \boldsymbol{\omega}_i) \right] \qquad (2)$$

where $\mathbf{J}_{i,v}$, $\mathbf{J}_{i,\omega}$ are the Jacobian matrices. Eq. (2) expresses the basic physical intuition of a mechanical system that the left side forces generate the right side motions. However, these forces and motions are equated in the generalized space where the physically meaningful quantities (velocities, accelerations, and forces) need be mapped by the corresponding Jacobians (a map between the workspace and the generalized space).

Based on Eq. (2), the equation of motion for the continuum robot with N segments can be formulated as follows

$$\sum_{i=1}^{\mathrm{N}} \mathbf{J}_{i,act}^{T} \boldsymbol{\tau}_{i,act} = \sum_{i=1}^{\mathrm{N}} \left(\boldsymbol{\tau}_{i,inr} + \boldsymbol{\tau}_{i,els} + \boldsymbol{\tau}_{i,gvt} + \boldsymbol{\tau}_{i,dmp} \right) \tag{3}$$

where $\boldsymbol{\tau}_{i,act}$ is the actuation force for the ith segment and $\mathbf{J}_{i,act}$ is the corresponding Jacobian matrix. $\boldsymbol{\tau}_{i,inr}$, $\boldsymbol{\tau}_{i,els}$, $\boldsymbol{\tau}_{i,gvt}$, and $\boldsymbol{\tau}_{i,dmp}$ are the inertia loading, elastic loading, gravitational loading, and the damping loading respectively.

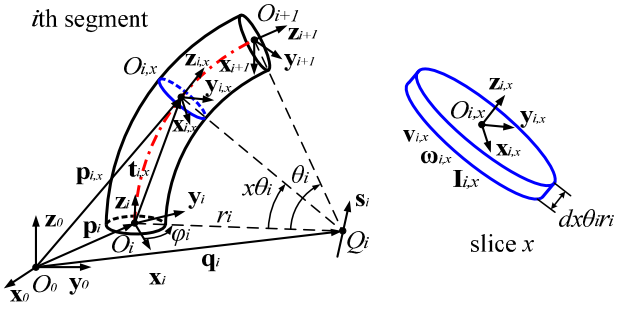

Figure 2: Kinematic configurations of the ith segment and one slice in the segment

For a flexible body consisting of continuous materials, the inertia loading, elastic loading, and the gravitational loading are usually integrals along the body, which demonstrate the challenge that this paper attempts to address. Figure 2 shows the kinematic configuration of the ith segment in a continuum robot. Note that since the angular energy was reported to contribute only 5% of the total energy [17], the angular loading in this paper is neglected in the dynamic formulation. The assumptions used in the formulation are listed as follows:

A1: Each segment bends in a circular arc shape, i.e. each segment conducts constant curvature bending.

A2: The circular shape bending does not involve twist motion, i.e. there is no torsional effect in elastic loading.

A3: Three local parameters (θ, r, φ) totally define the segment configuration. Each segment is driven by three actuators.

A4: The extensional and bending deformations are independent, thus the elastic loading follows the superposition principle of ideal elastic body.

A5: The angular loading of slice x is negligible.

3 EVALUATION OF THE LOADING INTEGRALS USING THE SEPARATION OF VARIABLES METHOD

The key to evaluate the integrals analytically is to separate the integral variable from the non-integral terms. This requires a good mathematical representation that contains the integral variable explicitly. The homogeneous transformation usually mixes all terms together which makes this separation difficult. By observing the problem setting in Fig. 2, it is found that the slice x can be regarded as a slice at O_i being rotated $x\theta_i$ with respect to the axis $\mathbf{s}_i$ which passes through the curvature center Q_i and is perpendicular to the bending plane. This observation allows representing each slice of the continuum segment by a linear form of the integral variable x. By doing so, the integral variable is explicitly isolated with other non-integral terms, which facilitates the following integration procedures significantly.

3.1 Inertia Loadings of the ith Segment

The inertia loading of the ith segment can be obtained by the integral

$$\boldsymbol{\tau}_{i,inr} = \int_0^1 \mathbf{J}_{i,x,v}^{T} dm_i \dot{\mathbf{v}}_{i,x} \tag{4}$$

where dm_i is the mass of slice x and $\mathbf{J}_{i,x,v}$ is the Jacobian of the slice center $O_{i,x}$ (also the center of mass). $\dot{\mathbf{v}}_{i,x}$ is the acceleration of $O_{i,x}$. The slice mass can be further obtained by the infinitesimal dx of the integral variable

$$dm_i = \frac{M_i}{\theta_i r_i} dx \tag{5}$$

Note that the angular part of Eq. (4) is neglected due to assumption A5. For the rest of the linear inertia loading, position vector $\mathbf{p}_{i,x}$ of the slice x center $O_{i,x}$ is required at first. By observing the geometry in Fig. 1, a good kinematic representation to isolate the integral variable is

$$\mathbf{p}_{i,x} = \mathbf{q}_i + \mathrm{s}_x \mathbf{a}_i - \mathrm{c}_x \mathbf{b}_i \tag{6}$$

where $\mathrm{s}_x = \sin\theta_i x$ and $\mathrm{c}_x = \cos\theta_i x$. $\mathbf{q}_i$, $\mathbf{a}_i$, and $\mathbf{b}_i$ are given in Eqs. (7) - (9).

$$\mathbf{q}_i = \mathbf{p}_i + \mathbf{b}_i \tag{7}$$

$$\mathbf{a}_i = r_i \mathbf{z}_i \tag{8}$$

$$\mathbf{b}_i = r_i(\cos\varphi_i \mathbf{x}_i + \sin\varphi_i \mathbf{y}_i) \tag{9}$$

Differentiating Eq. (6) yields the velocity of $O_{i,x}$

$$\mathbf{v}_{i,x} = \dot{\mathbf{q}}_i + \mathrm{s}_x \dot{\mathbf{a}}_i + x\mathrm{c}_x \mathbf{a}_i \dot{\theta}_i - \mathrm{c}_x \dot{\mathbf{b}}_i + x\mathrm{s}_x \mathbf{b}_i \dot{\theta}_i \tag{10}$$

The corresponding Jacobian transpose may be obtained as

$$\mathbf{J}_{i,x,v}^{T} = \mathbf{J}_{i,q}^{T} + \mathrm{s}_x \mathbf{J}_{i,a}^{T} + x\mathrm{c}_x \mathbf{J}_{i,\theta}^{T} \mathbf{a}_i^{T} - \mathrm{c}_x \mathbf{J}_{i,b}^{T} + x\mathrm{s}_x \mathbf{J}_{i,\theta}^{T} \mathbf{b}_i^{T} \tag{11}$$

where $\mathbf{J}_{i,q}$, $\mathbf{J}_{i,a}$, $\mathbf{J}_{i,b}$, and $\mathbf{J}_{i,\theta}$ are the Jacobians of $\dot{\mathbf{q}}_i$, $\dot{\mathbf{a}}_i$, $\dot{\mathbf{b}}_i$, and $\dot{\theta}_i$ respectively. Since the integral variable is separated from the non-integral terms in Eq. (11), vector representation may be applied to separate the non-integral terms (without x in subscripts), as shown in Eq. (12).

$$\mathbf{J}_{i,x,v}^T = \mathbf{J}_{i,v}^T \circ \boldsymbol{f}_{i,v}(x) \tag{12}$$

where "$\circ$" denotes the block wise matrix multiplication (each block is treated as an entry for evaluation).

$$\mathbf{J}_{i,v}^T = [\mathbf{J}_{i,q}^T \quad \mathbf{J}_{i,a}^T \quad \mathbf{J}_{i,\theta}^T\mathbf{a}_i^T \quad \mathbf{J}_{i,b}^T \quad \mathbf{J}_{i,\theta}^T\mathbf{b}_i^T] \tag{13}$$

$$\boldsymbol{f}_{i,v}(x) = [1 \quad \mathrm{s}_x \quad x\mathrm{c}_x \quad -\mathrm{c}_x \quad x\mathrm{s}_x]^T \tag{14}$$

Similarly, differentiating the velocities gives the acceleration relationship

$$\begin{aligned}\dot{\mathbf{v}}_{i,x} = {} & \ddot{\mathbf{q}}_i + \mathrm{s}_x\ddot{\mathbf{a}}_i + 2x\mathrm{c}_x\dot{\mathbf{a}}_i\dot{\theta}_i + x\mathrm{c}_x\mathbf{a}_i\ddot{\theta}_i - x^2\mathrm{s}_x\mathbf{a}_i\dot{\theta}_i^2 \\ & -\mathrm{c}_x\ddot{\mathbf{b}}_i + 2x\mathrm{s}_x\dot{\mathbf{b}}_i\dot{\theta}_i + x\mathrm{s}_x\mathbf{b}_i\ddot{\theta}_i + x^2\mathrm{c}_x\mathbf{b}_i\dot{\theta}_i^2 \\ & = \boldsymbol{f}_{i,a} \circ \mathbf{h}_{i,v}\end{aligned} \tag{15}$$

where

$$\mathbf{h}_{i,v}^T = [\ddot{\mathbf{q}}_i^T \ \ddot{\mathbf{a}}_i^T \ \dot{\mathbf{a}}_i^T\dot{\theta}_i \ \mathbf{a}_i^T\ddot{\theta}_i \ \mathbf{a}_i^T\dot{\theta}_i^2 \ \ddot{\mathbf{b}}_i^T \ \dot{\mathbf{b}}_i^T\dot{\theta}_i \ \mathbf{b}_i^T\ddot{\theta}_i \ \mathbf{b}_i^T\dot{\theta}_i^2] \tag{16}$$

$$\boldsymbol{f}_{i,a}(x) = [1 \ \mathrm{s}_x \ 2x\mathrm{c}_x \ x\mathrm{c}_x \ -x^2\mathrm{s}_x \ -\mathrm{c}_x \ 2x\mathrm{s}_x \ x\mathrm{s}_x \ x^2\mathrm{c}_x] \tag{17}$$

Therefore, the linear inertia loading $\boldsymbol{\tau}_{i,inr,v}$ is evaluated as

$$\begin{aligned}\boldsymbol{\tau}_{i,inr,v} = {} & \frac{M_i}{\theta_i r_i}\int_0^1 \mathbf{J}_{i,v}^T \circ \boldsymbol{f}_{i,v}(x)\boldsymbol{f}_{i,a}(x) \circ \mathbf{h}_{i,v}dx \\ = {} & \frac{M_i}{\theta_i r_i}\mathbf{J}_{i,v}^T \circ \mathbf{Q}_{i,v} \circ \mathbf{h}_{i,v}\end{aligned} \tag{18}$$

where $\mathbf{Q}_{i,v}(\theta_i) = \int_0^1 \boldsymbol{f}_{i,v}(x)\boldsymbol{f}_{i,a}(x)\,dx$ is a precomputed matrix only depending on θ_i. Note that all kinematic terms without involving the integral variable are obtained by the segment wise kinematic analysis in section 4.1.

3.2 Elastic Loadings

The elastic loading accounts for the torque conquering material deformation. Two types of deformation are considered in this paper for the constant curvature continuum robots: extension and bending. The torsional effects are neglected and the extensional and bending deformation are assumed to be independent. Therefore, the elastic loading satisfies the superposition principle, which is given as

$$\boldsymbol{\tau}_{i,els} = \boldsymbol{\tau}_{i,ext} + \boldsymbol{\tau}_{i,bnd} \tag{19}$$

where $\boldsymbol{\tau}_{i,els}$, $\boldsymbol{\tau}_{i,ext}$, and $\boldsymbol{\tau}_{i,bnd}$ represent the elastic loading, the extensional elastic loading, and the bending elastic loading respectively. $\boldsymbol{\tau}_{i,ext}$ is obtained by calculating the virtual work for slice x due to the extensional force, which is calculated as

$$\mathbf{F}_{i,x,ext} = \int_A \sigma_{i,x}\mathbf{z}_{i,x}dA = E\varepsilon_i A\mathbf{z}_{i,x} \tag{20}$$

where $A = \pi R^2$ is the slice cross section area, $\sigma_{i,x}$ is the stress, E is the elastic module, and $\varepsilon_{i,x} = (\theta_i r_i - L_{i,0})/L_{i,0}$ is the strain. Note that the integral is evaluated based on the independent deformation assumption. Therefore, the virtual work of the extensional force is given by

$$\delta W_{i,x,ext} = \delta u_{i,x}\mathbf{z}_{i,x}^T\mathbf{F}_{i,x,ext} = E\varepsilon_i A\delta u_{i,x} \tag{21}$$

where $\delta u_{i,x}$ is the virtual extensional quantity of slice x and is given in Eq. (22).

$$\begin{aligned}\delta u_{i,x} &= \delta(dx\theta_i r_i) = dx(r_i\delta\theta_i + \theta_i\delta r_i) \\ &= \delta\mathbf{q}^T dx\left(r_i\mathbf{j}_{i,\theta}^T + \theta_i\mathbf{j}_{i,r}^T\right)\end{aligned} \tag{22}$$

Therefore, the virtual work due to extensional elastic force is simplified into Eq. (23) and the extensional elastic loading $\boldsymbol{\tau}_{i,ext}$ is obtained by Eq. (24) accordingly.

$$\delta W_{i,x,ext} = \delta\mathbf{q}^T E\varepsilon_i A\left(r_i\mathbf{j}_{i,\theta}^T + \theta_i\mathbf{j}_{i,r}^T\right)dx = \delta\mathbf{q}^T\boldsymbol{\tau}_{i,x,ext} \tag{23}$$

$$\boldsymbol{\tau}_{i,ext} = \int_0^1 \boldsymbol{\tau}_{i,x,ext} = \frac{\pi R^2 E\left(\theta_i r_i - L_{i,0}\right)}{L_{i,0}}\left(r_i\mathbf{j}_{i,\theta}^T + \theta_i\mathbf{j}_{i,r}^T\right) \tag{24}$$

Similarly, the torque conquering bending deformation of segment i is calculated as

$$\mathbf{M}_{i,bnd} = K\kappa_i\mathbf{s}_i \tag{25}$$

where K is the bending stiffness, $\kappa_i = 1/r_i$ is the segment curvature, and $\mathbf{s}_i$ is the unit vector going through Q_i and perpendicular to the bending plane. Note that the common used bending stiffness formula $K = EI_{xx}$ (I_{xx} is the area moment of inertia) is not applicable here since the segments consist of actuators instead of solid elastic cores. The corresponding virtual curvature change is

$$\delta\kappa_i = \delta\left(\frac{1}{r_i}\right) = -\frac{1}{r_i^2}\delta r_i = -\frac{1}{r_i^2}\delta\mathbf{q}^T\mathbf{j}_{i,r}^T \tag{26}$$

Therefore, the virtual work due to bending elastic force is calculated as

$$\begin{aligned}\delta W_{i,bnd} &= \delta\kappa_i\mathbf{s}_i^T\mathbf{M}_{i,bnd} \\ &= -\tfrac{1}{r_i^2}\delta\mathbf{q}^T\mathbf{j}_{i,r}^T K\kappa_i \\ &= \delta\mathbf{q}^T\boldsymbol{\tau}_{i,bnd}\end{aligned} \tag{27}$$

which yields the segment wise bending elastic loading

$$\boldsymbol{\tau}_{i,bnd} = -\frac{K}{r_i^3}\mathbf{j}_{i,r}^T \tag{28}$$

3.3 Gravitational and Damping Loadings

The segment loading due to gravity is computed by the integral of the gravity loadings of slice x, as given in Eq. (29).

$$\boldsymbol{\tau}_{i,gvt} = \int_0^1 \mathbf{J}_{i,x,v}^T dm_i \mathrm{g}\mathbf{z}_0 = \frac{M_i \mathrm{g}}{\theta_i r_i}\int_0^1 \mathbf{J}_{i,x,v}^T \mathbf{z}_0 dx$$
$$= \frac{M_i \mathrm{g}}{\theta_i r_i}(\mathbf{J}_{i,v}^T \circ \mathbf{G})\mathbf{z}_0 \quad (29)$$

where $\mathbf{G}(\theta_i) = \int_0^1 \boldsymbol{f}_{i,v}(x)dx$ and g is the gravity constant. By observing the expressions of $\mathbf{G}$ and $\mathbf{Q}_{i,v}$, it is easy to find that $\mathbf{G}$ is actually the first column of $\mathbf{Q}_{i,v}$.

The damping loading is modeled as bending friction force that is proportional to the bending curvature changing rate. Therefore, mimicking the derivation of the bending elastic loading yields the damping loading as

$$\boldsymbol{\tau}_{i,dmp} = \frac{c_{i,dmp}\dot{r}_i}{r_i^4}\mathbf{j}_{i,r}^T \quad (30)$$

where $c_{i,dmp}$ is the damping coefficient.

3.4 Actuation Jacobians

Actuation Jacobian is related to the actual actuation type. The most commonly used actuation scheme for extensible continuum robot is to use three flexible linear actuators which could be realized by tendons [1, 13], hydraulisc [20], artificial muscles/pneumatics [14], Shape Memory Alloys [21], etc.

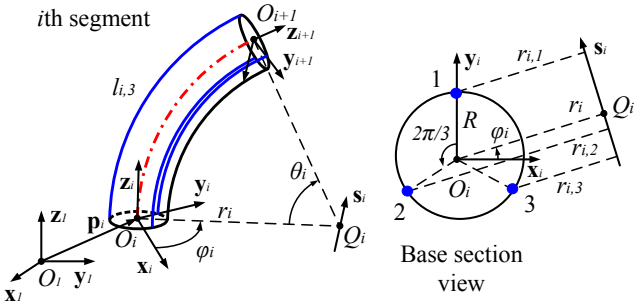

Figure 3: Actuator deployment of the ith segment

For this type of actuator deployment, the actuation Jacobian is derived by differentiating the actuator length $l_{i,j}$, which is given in Eq. (31) where i represents the ith segment and j denotes the jth actuator.

$$l_{i,j} = \theta_i r_{i,j} \qquad j = 1,2,3 \quad (31)$$

$$r_{i,j} = r_i - R\cos(\frac{2\pi}{3}j - \varphi_i - \frac{\pi}{6}) \quad (32)$$

Differentiating Eq. (31) yields

$$\dot{l}_{i,j} = r_i\dot{\theta}_i + \theta_i\dot{r}_i - R\cos\left(\frac{2\pi}{3}j - \varphi_i - \frac{\pi}{6}\right)\dot{\theta}_i$$
$$-R\theta_i\sin(\frac{2\pi}{3}j - \varphi_i - \frac{\pi}{6})\dot{\varphi}_i \quad (33)$$

Therefore, the Jacobian for the jth actuator in the ith segment is obtained by Eq. (34) and the overall Jacobian for the ith segment is given in Eq. (35). Note that the $\mathbf{j}_{i,\theta}$, $\mathbf{j}_{i,r}$, and the $\mathbf{j}_{i,\varphi}$ are the trivia Jacobians (for the generalized coordinates $\mathbf{q} = [q_1 \cdots q_k \cdots q_n]^T$, $\mathbf{j}_k = [0 \cdots 1 \cdots 0]$ is the trivia Jacobian of q_k) of θ_i, r_i and φ_i respectively.

$$\mathbf{j}_{i,j,act} = r_i\mathbf{j}_{i,\theta} + \theta_i\mathbf{j}_{i,r} - R\cos\left(\frac{2\pi}{3}j - \varphi_i - \frac{\pi}{6}\right)\mathbf{j}_{i,\theta}$$
$$-R\theta_i\sin(\frac{2\pi}{3}j - \varphi_i - \frac{\pi}{6})\mathbf{j}_{i,\varphi} \quad (34)$$

$$\mathbf{J}_{i,act}^T = [\mathbf{j}_{i,1,act}^T \quad \mathbf{j}_{i,2,act}^T \quad \mathbf{j}_{i,3,act}^T] \quad (35)$$

4 RECURSIVE CALCULATION OF THE PROPOSED DYNAMIC MODEL

This section presents the implementation algorithm of the proposed dynamic model. The inverse dynamics is computed first and the forward dynamics is then derived based on the inverse dynamics.

4.1 Segment Wise Kinematic Propagation

The forward propagation of the segment-wise kinematics is required in the recursive computation process. Since there is no integral variable involved in the segment-wise calculation, the propagation formulas are obtained by the same procedure as for the rigid body system (which can be found in most mechanics books, such as in reference [22]). That is, find the position relationship first, and then derive the velocity propagation and acceleration propagation by direct differentiations.

The segment wise position and orientation propagations are calculated as in Eqs. (36) - (37) where $\mathrm{s}_{\theta m} = \sin\theta_{\mathrm{i-1}}$ and $\mathrm{c}_{\theta m} = \cos\theta_{\mathrm{i-1}}$.

$$\mathbf{R}_i = e^{\hat{\mathbf{s}}_{i-1}\theta_{i-1}}\mathbf{R}_{i-1} \quad (36)$$

$$\mathbf{p}_i = \mathbf{p}_{i-1} + \mathrm{s}_{\theta m}\mathbf{a}_{i-1} + (1 - \mathrm{c}_{\theta m})\mathbf{b}_{i-1} \quad (37)$$

After obtaining the position relationships, the velocities and accelerations can be obtained by differentiating Eqs. (36) - (37) directly. Eq. (38) gives the angular velocity propagation formula.

$$\hat{\boldsymbol{\omega}}_i = e^{\hat{\mathbf{s}}_{i-1}\theta_{i-1}}\hat{\boldsymbol{\omega}}_{i-1}e^{-\hat{\mathbf{s}}_{i-1}\theta_{i-1}} + \frac{d\hat{\mathbf{s}}_{i-1}\theta_{i-1}}{dt} \quad (38)$$

4.2 Numerical Algorithm to Compute the Inverse Dynamics

The inverse dynamics are computed recursively from the first segment to the last segment where the algorithm computes all required kinematic information first by forward propagation and then obtain the corresponding loadings for each segment.

After summing up all the dynamic loadings, the actuation Jacobians are used to map the total loading back to the joint space, which are the required actuator inputs. Algorithm 1 shows the details of the inverse dynamics computation.

Algorithm 1 Inverse Dynamics Computation

for segment $i = 1{:}N$ **do**
  Evaluate the integral matrices $\mathbf{Q}_{i,v}$ and $\mathbf{G}_i$
  Compute Jacobians $\mathbf{J}_{i,act}$ and concatenate $\mathbf{J}_{act}$
  Compute the required position information
  Compute $\mathbf{J}_{i,v}$
  Compute the required velocity information
  Compute the required acceleration information
  Compute $\mathbf{h}_{i,v}$
  Compute $\boldsymbol{\tau}_{i,inr}$, $\boldsymbol{\tau}_{i,els}$, $\boldsymbol{\tau}_{i,gvt}$, and $\boldsymbol{\tau}_{i,dmp}$
  Compute $\boldsymbol{\tau}_{i,load} = \boldsymbol{\tau}_{i,inr} + \boldsymbol{\tau}_{i,els} + \boldsymbol{\tau}_{i,dmp} + \boldsymbol{\tau}_{i,gvt}$
  Compute $\boldsymbol{\tau}_{load} \mathrel{+}= \boldsymbol{\tau}_{i,load}$
end for
Compute the actuator force $\boldsymbol{\tau}_{act} = \mathbf{J}_{act}^{T\ -1}\boldsymbol{\tau}_{load}$

4.3 Obtain the Forward Dynamics Based on the Inverse Dynamics

Although the inverse dynamics is usually good enough for the practical application, forward dynamics is also desired for the purpose of simulations. Based on reference [22], forward dynamics could be obtained using the so called "differential inverse dynamics function" (pages 102-103 of reference [22]) which is given in Eq. (39) where $\mathbf{J}_\tau^T\boldsymbol{\tau} = \mathrm{ID}(\mathrm{md}, \mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}})$ is the inverse dynamics in the generalized space.

$$\mathrm{ID}_\delta(\mathrm{md}, \mathbf{q}, \ddot{\mathbf{q}}) = \mathrm{ID}(\mathrm{md}, \mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}}) - \mathrm{ID}(\mathrm{md}, \mathbf{q}, \dot{\mathbf{q}}, \mathbf{0}) \tag{39}$$

By using the differential inverse dynamics function, the system inertia matrix $\mathbf{H}$ can be obtain by Eq. (40) where $\mathbf{h}_\alpha$ is the αth column of $\mathbf{H}$ and $\boldsymbol{\delta}_\alpha$ is the unit vector with 1 on the αth entry.

$$\mathbf{h}_\alpha = \mathrm{ID}(\mathrm{model}, \mathbf{q}, \dot{\mathbf{q}}, \boldsymbol{\delta}_\alpha) - \mathrm{ID}(\mathrm{model}, \mathbf{q}, \dot{\mathbf{q}}, \mathbf{0}) \tag{40}$$

The forward dynamics is then formulated as

$$\mathbf{H}\ddot{\mathbf{q}} = \mathbf{J}_\tau^T\boldsymbol{\tau} - \mathrm{ID}(\mathrm{model}, \mathbf{q}, \dot{\mathbf{q}}, \mathbf{0}) \tag{41}$$

where $\boldsymbol{\tau}$ is the actuator input vector.

Table 1. Simulation Parameters

Var.	Value	Var.	Value
R	0.0325m	$c_{i,dmp}$	0.05 N m s/rad^2
$L_{i,0}$	0.24m	g	9.8067m/s^2
E	15MPa	K	0.25 N m^2
M	0.255Kg		

5 NUMERICAL SIMULATIONS

This section presents the numerical simulations of the proposed dynamic model for which all computations are conducted in Matlab. The continuum robot under consideration is a two-segment structure with three intrinsic actuators for each segment, as illustrated in Fig. 1. Table 1 summarizes the parameter values used in the simulations. It is worth noting that for models relying on the constant curvature assumption, a singularity occurs when the curvature equals zero (i.e. the radius r_i goes to infinity or radius angle θ_i goes to zero). For this case, special care must be taken to avoid the singularity. In numerical implementation, this may be done by setting a threshold (for instance, $r_i > 100$) and substituting the singular terms by their corresponding asymptotic values (for instance, the inertia loading in Eq. (18) is recomputed by assuming that O_iO_{i+1} is a line segment instead of an arc). However, this kind of handling may introduce unknown dynamic effects into the model and thus generate inaccurate results. Therefore, the trajectories in the simulations are selected to satisfy the test purposes and avoid the singularities at the same time.

Three cases are simulated to test the algorithms. The first case is the free falling test, for which the actuators are all set to zero and the robot is released from a stationary position near the horizontal plane ($\kappa_1 = \kappa_2 = 0.13$ and $\varphi_1 = \varphi_2 = -\pi/2$ to avoid the singularities). Figure 4 shows the time-lapse image of robot and Fig. 5 plots the corresponding curvature responses. Each frame in Fig. 4 has the same time interval of 0.05s and the transparency increases as time elapses. Due to the damping effects, the robot approaches stable position after two oscillations (around 1.2s).

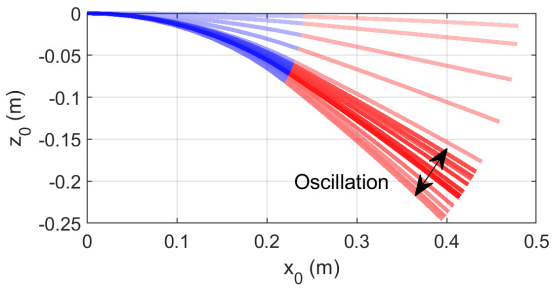

Figure 4: Time-lapse of the zero actuation response

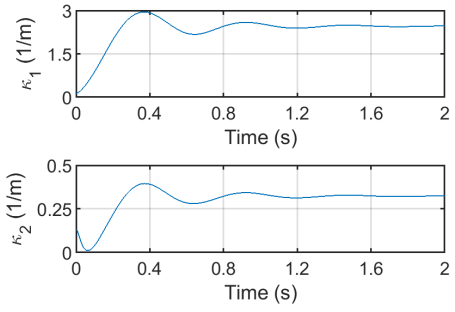

Figure 5: Curvature response of the zero actuation case

The second and the third cases are to evaluate the robot's performances under actuation, for which one in-plane bending and one out-of-plane bending are simulated. The in-plane bending applies $\boldsymbol{\tau}_{act,ip}$ on the actuators with the stationary initial position $\kappa_1 = \kappa_2 = 0.13$ and $\varphi_1 = \varphi_2 = \pi/2$. The out-of-plane bending utilizes $\boldsymbol{\tau}_{act,op}$ as the input and $\kappa_1 = \kappa_2 = 0.13$ and $\varphi_1 = \varphi_2 = -\pi/2$ as the initial condition (keeps stationary as other cases). Figure 6 illustrates the time-lapse responses (0.067s time interval) for both bending and Fig. 7 shows the curvature and curvature plane rotation responses of the out-of-plane bending case. From Fig. 7, low frequency oscillations in φ_1 and high frequency oscillations in φ_2 are observed, which reveals the complicated transient behaviors of soft body dynamics.

$$\boldsymbol{\tau}_{\text{act,ip}} = [-180 \quad 0 \quad 0 \quad -60 \quad 0 \quad 0]^{\text{T}}\text{N} \tag{42}$$

$$\boldsymbol{\tau}_{act,op} = [0 \quad -60 \quad 60 \quad 0 \quad 60 \quad -60]^{T}N \tag{43}$$

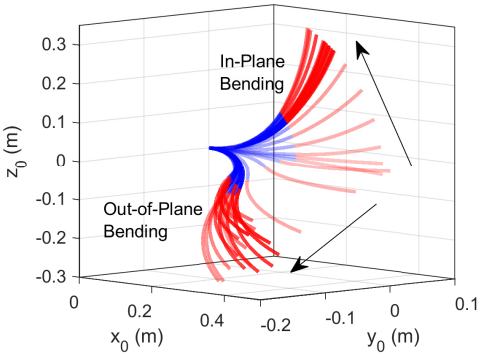

Figure 6: Time-lapse of the in-plane and out-of-plane bending responses

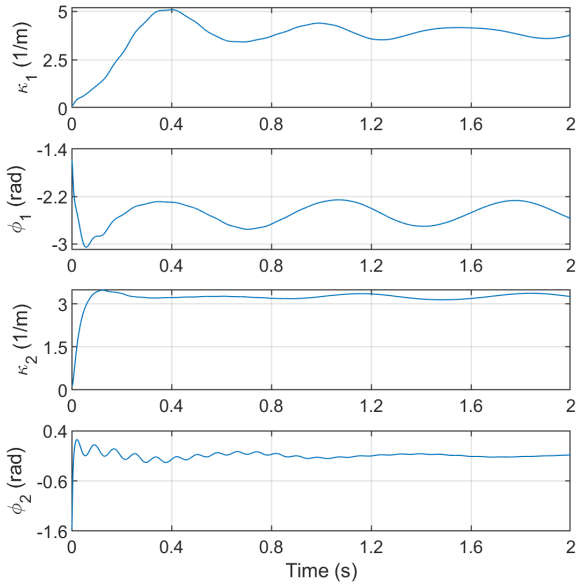

Figure 7: Curvature and curvature plane rotation responses of the out-of-plane bending

6 CONCLUSIONS AND FUTURE WORK

In this paper, an analytical dynamic model was developed as an extension to constant curvature continuum robots. The loading integrals were successfully evaluated by the separation of variables method such that the integral variable is separated from the non-integral terms. By doing so, the overall dynamic loadings can be computed as a traditional rigid body system and the integrals are evaluated in advance. The virtual work principle is used to assemble the dynamic loadings into the equation of motion while the differential inverse dynamics function is utilized to obtain the forward dynamics. A recursive algorithm based on the new inverse dynamic is proposed too. Numerical simulations for a two-segment continuum robot are conducted to validate the formulated model.

Since the model in this paper neglected the angular contributions of each slice, one main focus for future work will be on introducing these angular loadings into the model for better accuracy. In addition, more strict comparisons with other methods and most importantly, with hardware experiments, are required to verify the correctness of the proposed model. Finally, generalizing the methodology to other lumped parameter kinematic models of continuum robots will be another focus for future work.

ACKNOWLEDGMENTS

This material is partially based upon work supported by the National Science Foundation under Grant No. 1906727.

APPENDIX

In the following formulas, $\mathbf{c_\theta} = \mathbf{cos\theta_i}$, $\mathbf{s_\theta} = \mathbf{sin\theta_i}$, $\mathbf{c_{2\theta}} = \mathbf{cos2\theta_i}$, $\mathbf{s_{2\theta}} = \mathbf{sin2\theta_i}$. The components of matrix $\mathbf{Q_{i,v}}$ are listed as follows.

$\mathbf{Q}_{i,v}(1,1) = 1$
$\mathbf{Q}_{i,v}(2,1) = \mathbf{Q}_{i,v}(1,2) = (1 - \mathrm{c}_\theta)/\theta_i$
$\mathbf{Q}_{i,v}(3,1) = \mathbf{Q}_{i,v}(1,4) = (-1 + \mathrm{c}_\theta + \theta_i \mathrm{s}_\theta)/\theta_i^2$
$\mathbf{Q}_{i,v}(4,1) = \mathbf{Q}_{i,v}(1,6) = -\mathrm{s}_\theta/\theta_i$
$\mathbf{Q}_{i,v}(5,1) = \mathbf{Q}_{i,v}(1,8) = (-\theta_i \mathrm{c}_\theta + \mathrm{s}_\theta)/\theta_i^2$
$\mathbf{Q}_{i,v}(2,2) = 1 - \mathbf{Q}_{i,v}(4,6) = 1/2 - \mathrm{s}_{2\theta}/(4\theta_i)$
$\mathbf{Q}_{i,v}(3,2) = (-2\theta_i \mathrm{c}_{2\theta} + \mathrm{s}_{2\theta})/(8\theta_i^2)$
$\mathbf{Q}_{i,v}(3,2) = \mathbf{Q}_{i,v}(2,4) = -\mathbf{Q}_{i,v}(5,6) = -\mathbf{Q}_{i,v}(4,8)$
$\mathbf{Q}_{i,v}(4,2) = \mathbf{Q}_{i,v}(2,6) = (\mathrm{s}_{2\theta} - 1)/(4\theta_i)$
$\mathbf{Q}_{i,v}(5,2) = \mathbf{Q}_{i,v}(2,8) = (1 - \mathrm{c}_{2\theta} - 2\theta_i \mathrm{s}_{2\theta} + 2\theta_i^2)/(8\theta_i^2)$
$\mathbf{Q}_{i,v}(3,4) = (4\theta_i^3 + 6\theta_i \mathrm{c}_{2\theta} + (6\theta_i^2 - 3)\mathrm{s}_{2\theta})/(24\theta_i^3)$
$\mathbf{Q}_{i,v}(3,4) = -\mathbf{Q}_{i,v}(4,9)$
$\mathbf{Q}_{i,v}(4,4) = \mathbf{Q}_{i,v}(3,6) = \mathbf{Q}_{i,v}(5,2) - 1/2$
$\mathbf{Q}_{i,v}(5,4) = (-1 + (1 - 2\theta_i^2)\mathrm{c}_{2\theta} + 2\theta_i \mathrm{s}_{2\theta})/(8\theta_i^3)$
$\mathbf{Q}_{i,v}(5,4) = \mathbf{Q}_{i,v}(4,5) = \mathbf{Q}_{i,v}(3,8) = \mathbf{Q}_{i,v}(2,9)$
$\mathbf{Q}_{i,v}(:,3) = 2\mathbf{Q}_{i,v}(:,4)$
$\mathbf{Q}_{i,v}(1,5) = (2 + (\theta_i^2 - 2)\mathrm{c}_\theta - 2\theta_i \mathrm{s}_\theta)/\theta_i^3$
$\mathbf{Q}_{i,v}(2,5) = -\mathbf{Q}_{i,v}(5,8) = \mathbf{Q}_{i,v}(3,4) - 1/3$
$\mathbf{Q}_{i,v}(3,5) = (2\theta_i(2\theta_i^2 - 3)\mathrm{c}_{2\theta} - (6\theta_i^2 - 3)\mathrm{s}_{2\theta})/(16\theta_i^4)$

$\mathbf{Q}_{i,v}(3,5) = -\mathbf{Q}_{i,v}(5,9)$
$\mathbf{Q}_{i,v}(5,5) = (3 + 2\theta_i(2\theta_i^2 - 3)\mathrm{s}_{2\theta} + (6\theta_i^2 - 3)\mathrm{c}_{2\theta} - 2\theta_i^4)/(16\theta_i^4)$
$\mathbf{Q}_{i,v}(5,5) = \mathbf{Q}_{i,v}(3,9) - 1/4$
$\mathbf{Q}_{i,v}(:,7) = 2\mathbf{Q}_{i,v}(:,8)$
$\mathbf{Q}_{i,v}(1,9) = ((\theta_i^2 - 2)\mathrm{s}_\theta + 2\theta_i\mathrm{c}_\theta)/\theta_i^3$

REFERENCES

[1] Hannan, M. W. and Walker, I. D., 2003, "Kinematics and the implementation of an elephant's trunk manipulator and other continuum style robots," *Journal of robotic systems*, **20**(2), pp.45–63.

[2] Rucker, D.C., Jones, B.A. and Webster III, R.J., 2010, "A geometrically exact model for externally loaded concentric-tube continuum robots," *IEEE transactions on robotics*, **26**(5), p.769.

[3] Rone, W., Liu, Y. and Ben-Tzvi, P., 2019, "Maneuvering and Stabilization Control of a Bipedal Robot with a Universal-Spatial Robotic Tail", *Bioinspiration & Biomimetics*, **10**(1), p.016014.

[4] Liu, Y., Wang, J. and Ben-Tzvi, P., 2019, "A Cable Length Invariant Robotic Tail Using a Circular Shape Universal Joint Mechanism", *Journal of Mechanisms and Robotics, Transactions of the ASME*, In Press.

[5] Santiago, J.L.C., Godage, I.S., Gonthina, P. and Walker, I.D., 2016, "Soft robots and kangaroo tails: modulating compliance in continuum structures through mechanical layer jamming," *Soft Robotics*, **3**(2), pp.54-63.

[6] Jones, B.A. and Walker, I.D., 2006, "Practical kinematics for real-time implementation of continuum robots," *IEEE Transactions on Robotics*, **22**(6), pp.1087-1099.

[7] Rucker, D.C. and Webster III, R.J., 2011, "Statics and dynamics of continuum robots with general tendon routing and external loading," *IEEE Transactions on Robotics*, **27**(6), pp.1033-1044.

[8] Camarillo, D.B., Carlson, C.R. and Salisbury, J.K., 2009, "Task-space control of continuum manipulators with coupled tendon drive," *11th International Symposium on Experimental Robotics 2008 (Springer Tracts in Advanced Robotics, Vol. 54)*, Berlin, Springer, pp.271–280.

[9] Walker, I.D., 2013, "Continuous backbone 'continuum' robot manipulators," *Isrn robotics*, *2013*.

[10] Webster III, R.J. and Jones, B.A., 2010. Design and kinematic modeling of constant curvature continuum robots: A review. *The International Journal of Robotics Research*, *29*(13), pp.1661-1683.

[11] Spillmann, J. and Teschner, M., 2007, "CoRdE: Cosserat rod elements for the dynamic simulation of one-dimensional elastic objects," In *Proceedings of the 2007 ACM SIGGRAPH/Eurographics symposium on Computer animation*, pp. 63-72.

[12] Lang, H., Linn, J. and Arnold, M., 2011, "Multi-body dynamics simulation of geometrically exact Cosserat rods," *Multibody System Dynamics*, **25**(3), pp.285-312.

[13] Rone, W.S. and Ben-Tzvi, P., 2014, "Continuum robot dynamics utilizing the principle of virtual power," *IEEE Transactions on Robotics*, **30**(1), pp.275-287.

[14] Godage, I.S., Medrano-Cerda, G.A., Branson, D.T., Guglielmino, E. and Caldwell, D.G., 2016, "Dynamics for variable length multisection continuum arms," *The International Journal of Robotics Research*, **35**(6), pp.695-722.

[15] Godage, I.S., Wirz, R., Walker, I.D. and Webster III, R.J., 2015, "Accurate and efficient dynamics for variable-length continuum arms: a center of gravity approach," *Soft Robotics*, **2**(3), pp.96-106.

[16] Tatlicioglu, E., Walker, I.D. and Dawson, D.M., 2007, "Dynamic modelling for planar extensible continuum robot manipulators," In *ICRA 2007*, pp. 1357-1362.

[17] Godage, I.S., Webster III, R.J. and Walker, I.D., 2019. Center of Gravity-based Approach for Modeling Dynamics of Multisection Continuum Arms. *arXiv preprint arXiv:1901.01479*.

[18] Liu, Y., Ben-Tzvi, P., 2018, "Dynamic Modeling of a Quadruped with a Robotic Tail Using Virtual Work Principle", *Proceedings of the 2018 ASME IDETC/CIE, 42nd Mechanisms & Robotics Conference*, Quebec City, Canada, pp. V05BT07A021-V05BT07A021.

[19] Tsai, L.W., 2000. Solving the inverse dynamics of a Stewart-Gough manipulator by the principle of virtual work. *Journal of Mechanical design*, **122**(1), pp.3-9.

[20] Ikuta, K., Ichikawa, H., Suzuki, K. and Yajima, D., 2006, "Multi-degree of freedom hydraulic pressure driven safety active catheter," In *ICRA 2006*, pp.4161-4166.

[21] Jayender, J., Patel, R.V. and Nikumb, S., 2009, "Robot-assisted active catheter insertion: Algorithms and experiments," *The International Journal of Robotics Research*, **28**(9), pp.1101-1117.

[22] Featherstone, R., 2014. *Rigid body dynamics algorithms*. Springer.