Composite Fast-Slow Backstepping Design for Nonlinear Singularly Perturbed Newton-Euler Dynamics: Application to Soft Robots.

Olalekan Ogunmolu

***Abstract*— Controlling infinite degrees-of-freedom soft robots and deformable structures is well-known to be computationally prohibitive. This work proposes a composite two-time-scale singularly perturbed nonlinear controller for overcoming the long computational times of their recursive forward-inverse dynamics. Our contribution is four-pronged: (i) we decompose the global dynamics to a decentralized time-scale system of separate subdynamics; (ii) we then prescribe a set of stabilizing controllers for regulating each subsystem's dynamics; and (iii) we study the interconnected singularly perturbed system and analyze its stability; and (iv) lastly, we rigorously analyze the computational complexity of the new Newton-Euler recursive algorithm for the inverse dynamics controller. Our analyses are backed up by numerical experiments on a single arm of the Octopus-inspired robot as a benchmark. TO-DO: Results show performance gains in computation time and a better handling of the inherent nonlinearity of deformable structures and materials. Without loss of generality, we reckon that our scheme may find wide adoption in the (learning-based) control of deformable structures that are currently pervasive in soft robotics, as well as mixed and augmented reality today.**

Supplementary material—All codes for reproducing the experiments reported in this paper are available online: https://github.com/robotsorcerer/dcm.

I. Introduction

Soft manipulators, inspired by the functional role of living organisms' soft tissues, provide better active environmental interaction and compliance compared to ubiquitous rigid robot manipulators in society. This is due to their biomimicry of nervous and musculoskeletal systems that enable their configurability. This has fostered their applications in 6D dexterous bending and whole-arm manipulation [5], minimally invasive surgery in tight spaces [13, 14], inspection [9], and assistive rehabilitation [15, 12], among others.

The last few years have witnessed a flurry of static and dynamic models' development towards competing with the real-time computational advantages of rigid robot static and dynamic models. A non-exhaustive list includes approaches from morphoelastic filament theory [11, 8, 4], generalized Cosserat rod theory [20, 3], the constant curvature model [5], the piecewise constant curvature model [23, 16], a continuum Cosserat model [17] and a generalization of the dynamics of soft robots to rigid body-like dynamics in the discrete Cosserat model [18, 19].

Finite element modeling [2] provides precision in modeling accuracy; however, the long computational time for solutions defeats real-time control for reactive dexterous manipulation/movement. The kinematics and dynamics of

The author is with Microsoft Research NYC lekanmolu@microsoft.com.

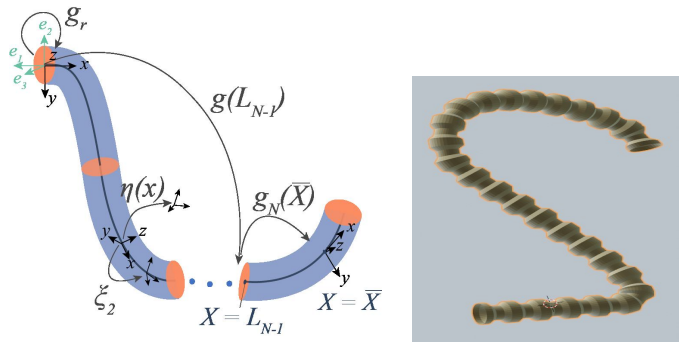

Fig. 1. [Left]: Configuration of an Octopus robot arm, reprinted from Molu et al. [10]. [Right]: 3D visualization of the arm.

dexterous soft multibody systems that is closest in spirit with Euler-Lagrange and Newton-Euler dynamic models of rigid body systems is the discrete version of the continuous Cosserat approach by Renda et al. [19]. Employing an accurate geometrically exact model for the finite displacements of soft bodies, it provides model reduction in the degrees-of-freedom (DOF) needed for infinite DOF nonlinear deformable structures. It's particularly more attractive because reconciles modeling soft multibody systems as it is done for rigid manipulators.

However, controlling compliant robot manipulators (or actuators) still remains a challenging task for real-time applications. For real-time control, the piecewise constant strain model [19] possesses an $O(N)$ computational complexity [10] for N-discrete Cosserat multibodied systems. Computation and control efficacy can only improve with more discretized Cosserat sections in the robot. However, with more sections come elongated computation times even on modern single-instruction multiple thread (SIMT) computers [10]. To mitigate this computational burden of soft structures' control, we take the view of reduced order modeling and control with singular perturbation techniques. As opposed to recently explored optimal control singular perturbation for rigid robots [7, 6], we emphasize a nonlinear control of two-time-scale soft multibody systems and through high-gain feedback, we provide an interpretation of the old concept of zero dynamics.

Taking inspiration from the peripheral nervous system of the Octopus, Shih et al. [22] articulated a mixed mode hierarchical control system comprised of a model-free reinforcement learning decision-maker and a model-based energy shaping controller for low-level motor tasks. Our work considers singular perturbations for legitimizing

ad-hoc simplifications of dynamic models without resorting to empirical hierarchical deformation based on the Octopus' neurophisiology. In emphasis and principle, our mathematical formulation takes this bioinspiration; however, we exert ourselves rigorously in the hierarchical deformation and control by taking a dynamic system's view that allows the negligence of (i) parasitic parameters which complicate system model; (ii) minute time constants, and mass densities etc; (iii) the overparameterization caused by sensitive neural network (and hence non-interpretability of) models used for the high-level controller in [22]. This allows us to focus on the dominant slow dynamics that affect system motion and deviations from the true system model result in fast subdynamics which are controlled on a separate time-resolution scale.

The two subsystems are on separate time-scales and a multirate sampling of state measurements asynchronously controls each subsystem: a fast sampling of the fast state variable is employed in the fast backstepping controller and slow-sampling of the slow state variable is employed in the slow controller. There is not a stringent requirement for communication between both systems so that the overall controller takes the form of a decentralized one.

The rest of this article is organized as follows: background and foundational theoretical machinery are described in §II; §III introduces the singularly perturbed dynamics, backstepping controller for the two-time-scale decomposed problem and stability analyses; numerical simulations are presented in §V, and we conclude the paper in §VI.

II. Notations and Preliminaries

Time variables e.g. t, T, will always be real numbers. Matrices and vectors are respectively upper- and lower-case Roman letters. The strain field and strain twist vectors are Greek letters, that is $\xi \in \mathbb{R}^6$ and $\eta \in \mathbb{R}^3$, respectively. Sets, screw stiffness, wrench tensors, and the gravitational vector are upper-case Calligraphic characters. Distributed wrench tensors are signified with an overbar, e.g. $\bar{\mathcal{F}}$. For a curve $X : [0, L]$ (L is the curve's length) at time t, the robot's configuration is $\mathcal{X}_t(X)$. The matrix A's Frobenius norm is denoted $\|A\|$ while its Euclidean norm is $\|A\|_2$. The Lie algebra of the Lie group $\mathbb{SE}(3)$ is $\mathfrak{se}(3)$. The special orthogonal group consisting of corkscrew rotations is $SO(3)$. For a configuration $\boldsymbol{g}(X) \in \mathbb{SE}(3)$, its adjoint and coadjoint are respectively $\mathrm{Ad}_{\boldsymbol{g}}$, $\mathrm{Ad}^{\star}_{\boldsymbol{g}}$; these are parameterized by the curve, X. In generalized coordinate, the joint vector of a soft robot is denoted $q = [\xi_1^{\top}, \ldots, \xi_{n_\xi}^{\top}]^{\top} \in \mathbb{R}^{6n_\xi}$.

A. SoRo Configuration

Depicted in Fig. 1, the inertial frame is the basis triad $(\boldsymbol{e}_1, \boldsymbol{e}_2, \boldsymbol{e}_3)$ and $\boldsymbol{g}_r$ is the inertial to base frame transformation. For a cable-driven arm, the point at which actuation occurs is labeled $\bar{X}$. The configuration matrix that parameterizes curve L_n in X is denoted $\boldsymbol{g}_{L_n}$. The cable runs through the z-axis (x-axis in the spatial frame) in the (micro) body frame. The robot's z-axis is offset in orientation from the inertial frame by -90° so that a transformation from the base to the inertial frames is

$$\boldsymbol{g}_r = \begin{pmatrix} 0 & -1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}. \tag{1}$$

B. Continuous Strain Vector and Twist Velocity Fields

Suppose that $p(X) \in \mathbb{R}^6$ describes a microsolid's position on the soft body at t and let $R(X)$ be the corresponding orientation matrix. Let the pose be $[p(X), R(X)]$. Then, the robot's C-space, parameterized by a curve $g(\cdot) : X \to \mathbb{SE}(3)$, is $g(X) = \begin{pmatrix} R(X) & p(X) \\ \mathbf{0}^{\top} & 1 \end{pmatrix}$. Suppose that $\varepsilon(X) \in \mathbb{R}^3$ and $\gamma(X) \in \mathbb{R}^3$ respectively denote the linear and angular strain components of the soft arm. Then, the arm's strain field is a state vector, $\breve{\xi}(X) \in \mathfrak{se}(3)$, along the curve $\boldsymbol{g}(X)$ i.e. $\breve{\xi}(X) = \boldsymbol{g}^{-1}\partial g/\partial X \triangleq \boldsymbol{g}^{-1}\partial_x \boldsymbol{g}$. In the microsolid frame, the matrix and vector representation of the strain state are respectively $\breve{\xi}(X) = \begin{pmatrix} \hat{\gamma} & \varepsilon \\ \mathbf{0} & 0 \end{pmatrix} \in \mathfrak{se}(3)$, $\xi(X) = \begin{pmatrix} \gamma^{\top} & \varepsilon^{\top} \end{pmatrix}^{\top} \in \mathbb{R}^6$. Read $\hat{\gamma}$: the anti-symmetric matrix representation of γ. Read $\breve{\xi}$: the isomorphism mapping the twist vector, $\xi \in \mathbb{R}^6$, to its matrix representation in $\mathfrak{se}(3)$. Furthermore, let $\nu(X), \omega(X)$ respectively denote the linear and angular velocities of the curve $g(X)$. Then, the velocity of $\boldsymbol{g}(X)$ is the twist vector field $\breve{\eta}(X) = \boldsymbol{g}^{-1}\partial \boldsymbol{g}/\partial t \triangleq \boldsymbol{g}^{-1}\partial_t \boldsymbol{g}$. In the microsolid frame, $\breve{\eta}(X) = \begin{pmatrix} \hat{\omega} & \nu \\ \mathbf{0} & 0 \end{pmatrix} \in \mathfrak{se}(3)$, $\eta(X) = \begin{pmatrix} \omega^{\top} & \nu^{\top} \end{pmatrix}^{\top} \in \mathbb{R}^6$.

C. Discrete Cosserat-Constitutive PDEs

The PCS model assumes that (ξ_i, η_i) $i = 1, \ldots, N$ robot sections are constant. Spatially spliced along sectional boundaries, the overall strain position and velocity of the entire soft robot is a piecewise sum of the sectional strain field parameters.

Using d'Alembert's principle, the generalized dynamics equation for PCS model of Fig. 1 under external and actuation loads admits the form [19]

$$\begin{aligned} &\underbrace{\left[\int_0^{L_N} J^{\top}\mathcal{M}_a J dX\right]}_{M(q)}\ddot{q} + \underbrace{\left[\int_0^{L_N} J^{\top}\mathrm{ad}^{\star}_{J\dot{q}}\mathcal{M}_a J dX\right]}_{C_1(q,\dot{q})}\dot{q} + \\ &\underbrace{\left[\int_0^{L_N} J^{\top}\mathcal{M}_a \dot{J} dX\right]}_{C_2(q,\dot{q})}\dot{q} + \underbrace{\left[\int_0^{L_N} J^{\top}\mathcal{D}J\|J\dot{q}\|_p dX\right]}_{D(q,\dot{q})}\dot{q} \\ &\underbrace{-(1-\rho_f/\rho)\left[\int_0^{L_N} J^{\top}\mathcal{M}\mathrm{Ad}_{\boldsymbol{g}}^{-1} dX\right]}_{N(q)}\mathrm{Ad}_{\boldsymbol{g}_r}^{-1}\mathcal{G} - \underbrace{J^{\top}(\bar{X})\mathcal{F}_p}_{F(q)} \\ &\underbrace{-\int_0^{L_N} J^{\top}\left[\nabla_x\mathcal{F}_i - \nabla_x\mathcal{F}_a + \mathrm{ad}^{\star}_{\eta_n}(\mathcal{F}_i - \mathcal{F}_a)\right] dX}_{\tau(q)} = 0 \end{aligned} \tag{2}$$

for a Jacobian $J(X)$, (see definition in [19]), wrench of internal forces $\mathcal{F}_i(X)$, distributed wrench of actuation loads $\bar{\mathcal{F}}_a(X)$, and *distributed* wrench of the applied external forces $\bar{\mathcal{F}}_e(X)$. The torque and (internal) force are respectively M_k, F_k for sections k; and $\mathcal{M}(X)$ is the screw mass inertia matrix, given as $\mathcal{M}(X) = \text{diag}\,(I_x, I_y, I_z, A, A, A)\,\rho$ for a body density ρ, sectional area A, bending, torsion, and second inertia operator I_x, I_y, I_z respectively.

Equation (2) can be appropriately written in standard Newton-Euler (N-E) form as

$$M(q)\ddot{q} + [C_1(q,\dot{q}) + C_2(q,\dot{q} + D(q,\dot{q}))]\,\dot{q} = \tau(q) + F(q) + N(q)\text{Ad}_{g_r}^{-1}\mathcal{G}. \tag{3}$$

In (2), $\mathcal{M}_a = \mathcal{M} + \mathcal{M}_f$ is a lumped sum of the microsolid mass inertia operator, $\mathcal{M}$, and that of the added mass fluid, $\mathcal{M}_f$; dX is the length of each section of the multi-robot arm; $\mathcal{D}(X)$ is the drag fluid mass matrix; $J(X)$ is the Jacobian operator; $\|\cdot\|_p$ is the translation norm of the expression contained therein; ρ_f is the density of the fluid in which the material moves; ρ is the body density; $\mathcal{G}$ is the gravitational vector defined as $\mathcal{G} = [0,0,0,-9.81,0,0]^T$; and $\mathcal{F}_p$ is the applied wrench at the point of actuation $\bar{X}$. These terms form the overall mass $M(q)$, Coriolis forces $C_i(q,\dot{q}), i = 1,2$, buoyancy-gravitational forces $N(q)$, drag matrix $D(q,\dot{q})$ and external force $F(q)$ in (3).

Suppose that

$$z = \begin{pmatrix} z_1 & z_2 \end{pmatrix}^\top \equiv \begin{pmatrix} q & \dot{q} \end{pmatrix}^\top \tag{4}$$

then we may transform (3) into the following set of first-order differential equations

$$\dot{z}_1 = z_2, \tag{5a}$$
$$\dot{z}_2 = M^{-1}\left\{\tau - (C_1 + C_2 + D)z_2 + F + N\text{Ad}_{g_r}^{-1}\mathcal{G}\right\}, \tag{5b}$$

where we have omitted the templated arguments for simplicity. We refer readers to Renda et al. [19], Boyer and Renda [1], Renda et al. [18] for further details.

III. Hierarchical Control Scheme

Our goal is to design a *multi-resolution feedback control scheme* which steer an arbitrary point in the joint space, $q(t)$ at time t, to a target point $q^d = (q_1^d, \ldots, q_N^d)^\top$[1] based on backstepping control design. Owing to the long computational times required to realize effective control [10], we transform the Cosserat system into a singularly perturbed system shown in Fig. 2. Under standard singular perturbation theory (SPT) assumptions, we take a composite control system viewpoint – systematically separating the fast and slow dynamics of (3) into a nonlinear two time-scale system comprising separate fast and slow controllers.

A. Singularly Perturbed Composite Controller

Seeking a robust response to parametric variations, noise sensitivity, and parasitic "small" time constants components of the dynamics that increase model order, we separate the

[1]Here, q_i is the joint space for a section of the multisection manipulator.

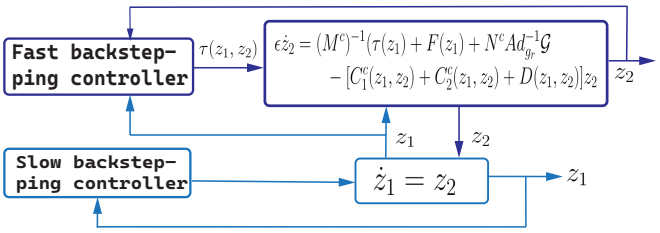

Fig. 2. Schematic of the proposed two-time-scale singularly perturbed system.

fast- (i.e. z_2) from the slow-changing (i.e. z_1) dynamics of (3). Thus, we write

$$\dot{z}_1 = f(z_1, z_2, \epsilon, u_s, t),\ z_1(t_0) = z_1(0),\ z_1 \in \mathbb{R}^{6N} \tag{6a}$$
$$\epsilon\dot{z}_2 = h(z_1, z_2, \epsilon, u_f, t),\ z_2(t_0) = z_2(0),\ z_2 \in \mathbb{R}^{6N} \tag{6b}$$

where f and h are $\mathcal{C}^n(n \gg 0)$ differentiable functions of their arguments, $\epsilon > 0$ denotes all small parameters to be ignored[2], u_s is the slow sub-dynamics' control law, and u_f is the fast sub-dynamics' controller.

Let us set $\epsilon = 0$; this reduces the system's dimension so that (6b) becomes the algebraic equation

$$0 = h(\bar{z}_1, \bar{z}_2, 0, u_s, t) \tag{7}$$

where $\bar{(\cdot)}$ signifies a variable in the system with $\epsilon = 0$. We proceed with the following assumptions.

Assumption 1 (Real and distinct root): Equation (7) has a unique and distinct root, given as $\bar{z}_2 = \phi(\bar{z}_1, t)$. Substituting $\bar{z}_2$ into (7), we have the form

$$0 = h(\bar{z}_1, \phi(\bar{z}_1, t), 0, u_s, t) \triangleq \bar{h}(\bar{z}_1, u_s, t),\ \bar{z}_1(t_0) = z_1(0), \tag{8}$$

where we have chosen the initial condition of the original system. Thus, the *quasi steady-state* (or slow) subsystem is

$$\dot{\bar{z}}_1 = f(\bar{z}_1, \bar{h}(\bar{z}_1, u_s, t), 0, u_s, t) \triangleq f_s(\bar{z}_1, u_s, t). \tag{9}$$

The variation of the slow subsystem (9) from the original system's response constitutes the fast transient $\tilde{z}_2 = z_2 - \bar{h}(\bar{z}_1, u_s, t)$ on a time scale $T = t/\epsilon$ so that

$$\frac{dz_1}{dT} = \epsilon f(z_1, z_2, \epsilon, u_s, t), \tag{10a}$$
$$\frac{d\tilde{z}_2}{dT} = \epsilon\frac{dz_2}{dt} - \epsilon\frac{\partial\bar{h}_1}{\partial\bar{z}_1}\dot{\bar{z}}_1, \tag{10b}$$
$$= h(z_1, \tilde{z}_2 + \bar{h}(\bar{z}_1, u_s, t), \epsilon, u_f, t) - \epsilon\frac{\partial\bar{h}_1}{\partial\bar{z}_1}\dot{\bar{z}}_1.$$

When $\epsilon = 0$, for the fast subsystem we must have

$$\frac{d\tilde{z}_2}{dT} = h(z_1, \tilde{z}_2 + \bar{h}(\bar{z}_1, u_s, t), 0, u_f, t). \tag{11}$$

[2]Restriction to a two-time-scale is not binding and one can choose to expand the system into multiple sub-dynamics across multiple time scales.

B. Two-Time-Scale Dynamics Extraction

First, consider a cable-actuated robot. At the point $\bar{X}$ (see Fig. 1) for the configuration $\mathcal{X}_t(\bar{X})$, the robot's motion is principally a consequence of the deformation of microslids around $\mathcal{X}_t(\bar{X})$. Denote the composite mass of these microsolids as $\mathcal{M}^{\text{core}}$. The motion of every other microsolid (with mass $\mathcal{M}^{\text{pert}}$) can be considered a perturbation from $\mathcal{M}^{\text{core}}$ so that $\mathcal{M}^{\text{pert}} = \mathcal{M} \setminus \mathcal{M}^{\text{core}}$. For fluid-driven robots' deformation such as fiber reinforced elastomeric enclosures (FREEs) with deformable shells [21, 14], principal motion is a consequence of deformation near the actuation area; let the core mass of these microsolids be $\mathcal{M}^{\text{core}}$). And let $\mathcal{M}^{\text{pert}}$ constitute the mass of the remnant micro-solids.

On a first glance at (2), it may seem that separating the slow and fast portions of the matrices $M(q)$, $C_1(q,\dot{q})$, $C_2(q,\dot{q})$ and $N(q)$ with different perturbation parameters is the most straightforward separation scheme. However, on a closer observation, the integro-matrices have in common the distributed mass density $\mathcal{M}$. Thus, we can choose a uniform perturbation parameter, $\epsilon = \|\mathcal{M}^{\text{core}}\|/\|\mathcal{M}^{\text{pert}}\|$ for separating the system dynamics.

The matrix densities of interest are separable as follows

$$M(q) = M^c(q) + M^p(q), \tag{12a}$$
$$C_1(q,\dot{q}) = C_1^c(q,\dot{q}) + C_1^p(q,\dot{q}), \tag{12b}$$
$$C_2(q,\dot{q}) = C_2^c(q,\dot{q}) + C_2^p(q,\dot{q}), \tag{12c}$$
$$N(q) = N^c(q) + N^p(q), \tag{12d}$$

where $(\cdot)^c, (\cdot)^p$ respectively denote the core and perturbed matrices over abscissa indices $[L^c_{\min}, L^c_{\max}]$ and $[L^p_{\min}, L^p_{\max}]$, respectively. Given the robot configuration in Fig. 1, we choose $0 \le L^p_{\min} < L^c_{\min}$ and $L^c_{\max} < L^p_{\max} \le L$. We write the singularly perturbed form of (5) as

$$\dot{z}_1 = z_2, \tag{13a}$$
$$\epsilon M \dot{z}_2 = \tau(z_1) + F(z_1) + N(z_1)\mathrm{Ad}_{\boldsymbol{g}_r}^{-1}\mathcal{G} - [C_1(z_1,z_2) + C_2(z_1,z_2) + D(z_1,z_2)]\, z_2. \tag{13b}$$

The justification for this model is described in what follows.

1) Quasi steady-state sub-dynamics extraction: Following the arguments in the foregoing, on the perturbed microsolids deformation is minute so that the strain twists and acceleration dynamics i.e. $(\dot{z}_1, \dot{z}_2)$ are equally small. Thus, for the slow subdynamics (5) transforms into

$$\dot{z}_1 = z_2, \tag{14a}$$
$$\epsilon M^p(z_1)\dot{z}_2 = \tau(z_1) + F(z_1) + N^p(z_1)\mathrm{Ad}_{\boldsymbol{g}_r}^{-1}\mathcal{G} - [C_1^p(z_1,z_2) + C_2^p(z_1,z_2) + D(z_1,z_2)]\, z_2. \tag{14b}$$

For $\epsilon = 0$, the resulting algebraic equation is

$$\tau(\bar{z}_1) + F(\bar{z}_1) + N^p(\bar{z}_1)\mathrm{Ad}_{\boldsymbol{g}_r}^{-1}\mathcal{G} - [C_1^p(\bar{z}_1,\bar{z}_2) + C_2^p(\bar{z}_1,\bar{z}_2) + D(\bar{z}_1,\bar{z}_2)]\,\bar{z}_2 = 0 \tag{15}$$

so that

$$\bar{z}_2 = [C_1^p(\bar{z}_1,\bar{z}_2) + C_2^p(\bar{z}_1,\bar{z}_2) + D(\bar{z}_1,\bar{z}_2)]^{-1}\{\tau(\bar{z}_1) + F(\bar{z}_1) + N(\bar{z}_1)\mathrm{Ad}_{\boldsymbol{g}_r}^{-1}\mathcal{G}\} \tag{16}$$

where the bar signifies variables within the reduced system. Thus, the slow subsystem can be written as

$$\dot{\bar{z}}_1 = \bar{z}_2,$$
$$\triangleq [C_1^p(\bar{z}_1,\bar{z}_2) + C_2^p(\bar{z}_1,\bar{z}_2) + D(\bar{z}_1,\bar{z}_2)]^{-1}\{\tau(\bar{z}_1) + F(\bar{z}_1) + N(\bar{z}_1)\mathrm{Ad}_{\boldsymbol{g}_r}^{-1}\mathcal{G}\}. \tag{17}$$

2) Fast subsystem dynamics extraction: The residual of the original system from the slow subsystem i.e. $\tilde{z}_2 = z_2 - \bar{z}_2$ is the fast transient on a time scale $T = t/\epsilon$. The fastest subsystem's dynamics evolves as

$$\frac{dz_1}{dT} = \epsilon z_2, \quad T = t/\epsilon, \tag{18a}$$
$$(M^c(z_1) + \epsilon\Lambda^m)\frac{d\tilde{z}_2}{dT} = \tau(z_1) + F(z_1) + (N^c(z_1) + \epsilon\Lambda^n)\mathrm{Ad}_{\boldsymbol{g}_r}^{-1}\mathcal{G} - (C_1^c + \epsilon\Lambda_1^c + C_2^c + \epsilon\Lambda_2^c + D)\, z_2 \tag{18b}$$

where we have occasionally omitted the templated arguments for ease of notation, and $\Lambda^m = M^p/\epsilon$, $\Lambda_1^c = C_1^p/\epsilon$, $\Lambda_2^c = C_2^p/\epsilon$, $\Lambda^n = N^p/\epsilon$, for $\epsilon = \|\mathcal{M}^{\text{core}}\|/\|\mathcal{M}^{\text{pert}}\|$ and the matrix $(M^c + \epsilon\Lambda^m)$ is assumed to be invertible. Carrying out the standard $\epsilon = 0$ on the fast time scale, we find that

$$\frac{dz_1}{dT} = 0, \tag{19a}$$
$$M^c(z_1)\frac{d\tilde{z}_2}{dT} = \tau(z_1) + F(z_1) + N^c(z_1)\mathrm{Ad}_{\boldsymbol{g}_r}^{-1}\mathcal{G} - [C_1^c(z_1,\tilde{z}_2) + C_2^c(z_1,\tilde{z}_2) + D(z_1,\tilde{z}_2)]\,\tilde{z}_2. \tag{19b}$$

C. Fast-Slow Controller Design and Stability Analysis

We proceed to design nonlinear backstepping controllers for the two separate time-scale problems developed in §III-B.

1) Stability Analysis of the Slow Subsystem: Suppose that we define an input u_s for the slow subsystem (17) so that $\dot{\bar{z}}_1 = u_s$. The tracking errors and corresponding error dynamics are

$$e_1 = z_1 - q^d, \quad \bar{e}_1 = \bar{z}_1 - q^d, \tag{20a}$$
$$\dot{e}_1 = \dot{z}_1 - \dot{q}^d, \quad \dot{\bar{e}}_1 = \dot{\bar{z}}_1 - \dot{q}^d; \tag{20b}$$

where we have used the same desired joint trajectory, q^d, for the full first subsystem and its variant when $\epsilon = 0$.

Theorem 1: The control law,

$$u_s = \dot{q}^d - e_1 - 2\tilde{z}_2 \tag{21}$$

guarantees an asymptotic stability of the origin such that for all $t \ge 0$, $z_1(t) \in S$ for a compact subset S of $\mathbb{R}^{6N}$. That is, $z_1(t)$ remains bounded as $t \to \infty$.

Proof: Let us treat $\dot{z}_1 - \dot{\bar{z}}_1$ as a perturbation of the slow subsystem $\dot{z}_1 = \bar{z}_2$. We take the Lyapunov function candidate,

$$V(z_1) = \frac{1}{2}e_1^\top(z_1)K_p e_1(z_1) \tag{22}$$

where K_p is a diagonal matrix of positive damping (gains). We see that for a constant q^d

$$\frac{\partial V}{\partial z_1}\dot{\bar{z}}_1 = -e_1^\top K_p \bar{z}_2.$$

This implies that $\bar{z}_2 = 0$ guarantees an asymptotically stable equilibrium of (17), hence satisfying Assumption 1. It follows that

$$\dot{V}(z_1) = \frac{\partial V}{\partial e_1}\dot{\bar{e}}_1 + \frac{\partial V}{\partial e_1}(\dot{e}_1 - \dot{\bar{e}}_1), \tag{23a}$$
$$= e_1^\top K_p \dot{\bar{e}}_1 + e_1^\top K_p(\dot{e}_1 - \dot{\bar{e}}_1), \tag{23b}$$
$$= e_1^\top K_p(u_s - \dot{q}^d) + e_1^\top K_p(\dot{z}_1 - \dot{\bar{z}}_1). \tag{23c}$$

Under (21), we must have

$$\dot{V}(z_1) \leq -e_1^\top K_p e_1 - e_1^\top K_p(2\tilde{z}_2 + z_2 - \bar{z}_2), \tag{24a}$$
$$\leq -e_1^\top K_p e_1 - e_1^\top K_p \tilde{z}_2. \tag{24b}$$

Thus, for all $t \geq 0$ and for a fixed $\tilde{z}_2$, for a compact subset S of $\mathbb{R}^{6N}$, $z_1(t) \in S$. Hence, $z_1(t)$ remains bounded as $t \to \infty$. ∎

2) Stability of the Boundary Layer Subsystem: We prescribe the controller for the fast subdynamics in this part, treating z_1 as a fixed parameter.

Theorem 2: Let

$$u_f(z_1, \tilde{z}_2) = M^c(z_1)(\ddot{q}_d + e_1) + \Big[\breve{C}^c(z_1, \tilde{z}_2) + \tag{25}$$
$$D(z_1, \tilde{z}_2)\Big]\tilde{z}_2 - \breve{C}^c(z_1, \tilde{z}_2)e_2 - F(z_1) - N^c(z_1)\mathrm{Ad}_{g_r}^{-1}\mathcal{G}.$$

Then, for $e_2 = \tilde{z}_2 - u_s$ where u_s is the virtual input, (25) exponentially stabilizes the boundary layer system (19).

Proof: We can write

$$\dot{e}_2 = \dot{\tilde{z}}_2 - \dot{u}_s = \dot{\tilde{z}}_2 - \ddot{q}^d + \dot{e}_1. \tag{26}$$

From (20), $\dot{e}_1 = z_2 - u_s + u_s - \dot{q}^d \triangleq e_2 - e_1$. Hence,

$$\dot{e}_2 = \dot{\tilde{z}}_2 - \ddot{q}^d + e_2 - e_1. \tag{27}$$

We make the following assumption about M^c.

Assumption 2: The core mass matrix, $M^c(z_1)$, is positive definite and bounded from below; $(M^c(z_1))^{-1}$ is also positive.

On the time-scale $T = t/\epsilon$, consider the following Lyapunov function candidate for the fast subsystem

$$W(z_1, \tilde{z}_2) = \frac{1}{2}e_2^\top M^c(z_1)e_2, \tag{28}$$

where $W(z_1, \tilde{z}_2) > 0\ \forall (\tilde{z}_2 \neq \bar{z}_2)$, z_1 is fixed, $M^c(z_1)$ corresponds to the mass matrix in (19) and $W(z_1, \bar{z}_2) = 0$. Dropping the arguments of W for ease of notation, we find that

$$\frac{dW}{dT} = \frac{1}{\epsilon}\frac{\partial W}{\partial e_2}\dot{e}_2, = \frac{1}{\epsilon}e_2^\top M^c \frac{de_2}{dT} + \frac{1}{2\epsilon}e_2^\top \frac{dM^c}{dT}e_2. \tag{29}$$

Let us parameterize the fast subsystem's controller as (for a fixed z_1)

$$u_f(z_1, \tilde{z}_2) = M^c \frac{du_s}{dT} + [C_1^c(z_1, \tilde{z}_2) + C_2^c(z_1, \tilde{z}_2)]u_s \tag{30}$$
$$+ D(z_1, \tilde{z}_2)\tilde{z}_2 - (F(z_1) + N^c(z_1)\mathrm{Ad}_{g_r}^{-1}\mathcal{G}) + \tilde{u}(z_1, \tilde{z}_2)$$

where $\tilde{u}(z_1, \tilde{z}_2)$ is a residual control to be designed. Setting $\breve{C}^c(z_1, \tilde{z}_2) = C_1^c(z_1, \tilde{z}_2) + C_2^c(z_1, \tilde{z}_2)$, and plugging (31) into (5), we have

$$M^c \frac{d}{dT}(\tilde{z}_2 - u_s) + \breve{C}^c(z_1, \tilde{z}_2)(\tilde{z}_2 - u_s) = \tilde{u}(z_1, \tilde{z}_2). \tag{31}$$

Hence, (29) becomes

$$\frac{dW}{dT}(z_1, \tilde{z}_2) = \frac{e_2^\top}{\epsilon}\Big(u(z_1, \tilde{z}_2) - \breve{C}e_2\Big) + \frac{1}{2\epsilon}e_2^\top \frac{dM^c}{dT}e_2,$$
$$= \frac{e_2^\top}{\epsilon}u(z_1, \tilde{z}_2) \tag{32}$$

where we have employed the skew-symmetric property, $dM^c/dT - 2\breve{C}(z_1, \tilde{z}_2) = 0$ (an extension of Property 3 in [10] to the time scale $T = t/\epsilon$) to arrive at the above. Suppose we set $\tilde{u}(z_1, \tilde{z}_2) = -Me_2$, we must have

$$\frac{dW}{dt}(z_1, \tilde{z}_2) = -e_2^\top M^c(z_1)e_2 \triangleq -2W(z_1, \tilde{z}_2). \tag{33}$$

Hence, exponential stability of the boundary layer system follows as a result of the controller (31),

$$u_f(z_1, \tilde{z}_2) = M^c(z_1)(\dot{u}_s - e_2) + \breve{C}^c(z_1, \tilde{z}_2)u_s + D(z_1, \tilde{z}_2)\tilde{z}_2$$
$$- (F(z_1) + N^c(z_1)\mathrm{Ad}_{g_r}^{-1}\mathcal{G}). \tag{34}$$

Since $\dot{u}_s = \dot{\tilde{z}}_2 - \dot{e}_2 \equiv \ddot{q}_d + e_1 - e_2$, then $\dot{u}_s - e_2 = \ddot{q}_d + e_1$. *A fortiori*, the torque input (34) admits the form (25). This concludes the proof. ∎

3) Interconnected Conditions: Stability of the Singularly Perturbed System: Let $\phi = (0, 1)$ and consider $\Lambda(z_1, z_2)$ as a weighted combination of $V(z_1)$ and $W(z_1, \tilde{z}_2)$ i.e. ,

$$\Sigma(z_1, z_2) = (1 - \phi)V(z_1) + \phi W(z_1, \tilde{z}_2),\ 0 < \phi < 1. \tag{35}$$

It follows that,

$$\dot{\Sigma}(z_1, z_2) = (1 - \phi)\left[\frac{\partial V}{\partial e_1}\dot{e}_1 + \frac{\partial V}{\partial e_1}(\dot{e}_1 - \dot{\bar{e}}_1)\right] + \frac{\phi}{\epsilon}\frac{\partial W}{\partial e_2}\dot{e}_2$$
$$\leq -(1 - \phi)e_1^\top K_p(e_1 + \tilde{z}_2) - \frac{\phi}{\epsilon}e_2^\top M^c e_2 \tag{36}$$
$$\leq -(1 - \phi)e_1^\top K_p e_1 - (1 - \phi)e_1^\top K_p \tilde{z}_2 - \frac{\phi}{\epsilon}e_2^\top M^c e_2 \tag{37}$$

which is clearly negative definite for any $\phi \in (0, 1)$. Therefore, Σ is a Lyapunov function for the singularly perturbed system (13) for $\epsilon = \|\mathcal{M}^c\|/\|\mathcal{M}^p\|$ and we conclude that the origin of the singularly perturbed system is asymptotically stable.

IV. Computational Complexity Analysis

We adopt the 'big O' notation in defining the cost of computation measured against size of a kinematic tree. The inverse dynamics of the discrete Cosserat model when computed with the recursive Newton-Euler algorithm costs $O(N)$ computational flops for an n-number of Cosserat rods used in the soft body system. In this section, we prescribe the new singularly perturbed controller's algorithm complexity based on the recursive Newton-Euler algorithm.

V. Numerical Results

VI. Discussions and Conclusion

References

[1] Frederic Boyer and Federico Renda. Poincaré's equations for cosserat media: application to shells. *Journal of Nonlinear Science*, 2016. 3

[2] Eulalie Coevoet, Adrien Escande, and Christian Duriez. Optimization-based Inverse Model of Soft Robots with Contact Handling. *IEEE Robotics and Automation Letters*, 2(3):1413–1419, 2017. 1

[3] Eugène Maurice Pierre Cosserat and François Cosserat. *Théorie des corps déformables*. A. Hermann et fils, 1909. 1

[4] Mattia Gazzola, LH Dudte, AG McCormick, and Lakshminarayanan Mahadevan. Forward and inverse problems in the mechanics of soft filaments. *Royal Society open science*, 5(6):171628, 2018. 1

[5] Isuru S Godage, David T Branson, Emanuele Guglielmino, Gustavo A Medrano-Cerda, and Darwin G Caldwell. Shape function-based kinematics and dynamics for variable length continuum robotic arms. In *2011 IEEE International Conference on Robotics and Automation*, pages 452–457. IEEE, 2011. 1

[6] Koji Ishihara and Jun Morimoto. Real-time Model Predictive Control with two-step optimization based on singularly perturbed system. *IEEE-RAS International Conference on Humanoid Robots*, 2015-Decem:173–180, 2015. ISSN 21640580. 1

[7] Koji Ishihara, Takeshi D. Itoh, and Jun Morimoto. Full-Body Optimal Control Toward Versatile and Agile Behaviors in a Humanoid Robot. *IEEE Robotics and Automation Letters*, 5(1):119–126, 2020. ISSN 23773766. 1

[8] Bartosz Kaczmarski, Alain Goriely, Ellen Kuhl, and Derek E Moulton. A Simulation Tool for Physics-informed Control of Biomimetic Soft Robotic Arms. *IEEE Robotics and Automation Letters*, 2023. 1

[9] Joshua S Mehling, Myron A Diftler, Mars Chu, and Michael Valvo. A minimally invasive tendril robot for in-space inspection. In *The First IEEE/RAS-EMBS International Conference on Biomedical Robotics and Biomechatronics, 2006. BioRob 2006.*, pages 690–695. IEEE, 2006. 1

[10] Lekan Molu, Shaoru Chen, and Audrey Sedal. Lagrangian Properties and Control of Soft Robots Modeled with Discrete Cosserat Rods. *(submitted to) IEEE International Conference on Robotics and Automation*, 2023. URL https://scriptedonachip.com/downloads/Papers/SoRoPD.pdf. 1, 3, 5

[11] Derek E Moulton, Thomas Lessinnes, and Alain Goriely. Morphoelastic Rods III: Differential Growth and Curvature Generation in Elastic Filaments. *Journal of the Mechanics and Physics of Solids*, 142:104022, 2020. 1

[12] R. W. Nuckols, S. Lee, K. Swaminathan, D. Orzel, R. D. Howe, and C. J. Walsh. Individualization of exosuit assistance based on measured muscle dynamics during versatile walking. *Science Robotics*, 6(60): eabj1362, 2021. 1

[13] Olalekan Ogunmolu, Adwait Kulkarni, Yonas Tadesse, Xuejun Gu, Steve Jiang, and Nicholas Gans. Soft-neuroadapt: A 3-dof neuro-adaptive patient pose correction system for frameless and maskless cancer radiotherapy. In *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 3661–3668. IEEE, 2017. 1

[14] Olalekan Ogunmolu, Xinmin Liu, Nicholas Gans, and Rodney D Wiersma. Mechanism and model of a soft robot for head stabilization in cancer radiation therapy. In *2020 IEEE International Conference on Robotics and Automation (ICRA)*, pages 4609–4615. IEEE, 2020. 1, 4

[15] Panagiotis Polygerinos, Zheng Wang, Kevin C Galloway, Robert J Wood, and Conor J Walsh. Soft robotic glove for combined assistance and at-home rehabilitation. *Robotics and Autonomous Systems*, 73: 135–143, 2015. 1

[16] Ke Qiu, Jingyu Zhang, Danying Sun, Rong Xiong, Haojian Lu, and Yue Wang. An efficient multi-solution solver for the inverse kinematics of 3-section constant-curvature robots. *arXiv preprint arXiv:2305.01458*, 2023. 1

[17] Federico Renda, Michele Giorelli, Marcello Calisti, Matteo Cianchetti, and Cecilia Laschi. Dynamic model of a multibending soft robot arm driven by cables. *IEEE Transactions on Robotics*, 30(5):1109–1122, 2014. 1

[18] Federico Renda, Vito Cacucciolo, Jorge Dias, and Lakmal Seneviratne. Discrete cosserat approach for soft robot dynamics: A new piece-wise constant strain model with torsion and shears. *IEEE International Conference on Intelligent Robots and Systems*, 2016-Novem:5495–5502, 2016. ISSN 21530866. 1, 3

[19] Federico Renda, Frédéric Boyer, Jorge Dias, and Lakmal Seneviratne. Discrete cosserat approach for multi-section soft manipulator dynamics. *IEEE Transactions on Robotics*, 34(6):1518–1533, 2018. 1, 2, 3

[20] M. B. Rubin. *Cosserat Theories: Shells, Rods, and Points*. Springer-Science+Business Medis, B.V., 2000. 1

[21] Audrey Sedal, Daniel Bruder, Joshua Bishop-Moser, Ram Vasudevan, and Sridhar Kota. A continuum model for fiber-reinforced soft robot actuators. *Journal of Mechanisms and Robotics*, 10(2):024501, 2018. 4

[22] Chia-Hsien Shih, Noel Naughton, Udit Halder, Heng-Sheng Chang, Seung Hyun Kim, Rhanor Gillette, Prashant G Mehta, and Mattia Gazzola. Hierarchical control and learning of a foraging cyberoctopus. *Advanced Intelligent Systems*, page 2300088, 2023. 1, 2

[23] Robert J. III Webster and Bryan A. Jones. Design and kinematic modeling of constant curvature continuum robots: A review. *The International Journal of Robotics Research*, 29(13):1661–1683, 2010. 1