Appendix – AC States Algorithm and Results

This page is left blank intentionally.

Exogenous Markov Decision Process (Exo-MDP) Machinery

- Consider the tuple $\mathcal{M} := (\mathcal{X}, \mathcal{Z}, \mathcal{A}, T, R, H)$
  - Starting distribution $\mu \in \Delta(\mathcal{Z})$;
  - Agent receives observations $\{x_h\}_{h=1}^{H} \in \mathcal{X}$ from the emission function $q : \mathcal{Z} \rightarrow \Delta(\mathcal{X})$;
  - Agent transitions between latent states via $T : \mathcal{Z} \times \mathcal{A} \rightarrow \Delta(\mathcal{S})$;
    - And rewards by $R : \mathcal{X} \times \mathcal{A} \rightarrow \Delta([0, 1])$
- Trajectories: $(z_1, x_1, a_1, r_1, \cdots, z_H, a_H, r_H)$ from repeated interactions;
  - $z_1 \sim \mu_1(\cdot)$, $z_{h+1} \sim T(\cdot|z_h, a_h)$, $x_h \sim q(\cdot|z_h)$ and $r_h \sim R(x_h, a_h, x_{h+1})$ for all $h \in [H]$.
- Define $supp(q(\cdot|z)) = \{x \in \mathcal{X} | q(x|z) > 0\}$ for any z.

Exo-MDP Machinery

Block MDP assumption $supp(q(\cdot|z_1)) \cap sup(q(\cdot|z_2)) = \emptyset$ for all $z_1 \neq z_2$.

- Agent chooses $a \sim \pi(z_h|x_h)$
- There exists non-stationary episodic policies $\Pi_{NS} := \Pi^H \supseteq (\pi_1, \cdots, \pi_H)$;
- Optimal policy $\pi^\star = \text{argmax} V_{\pi \in \Pi_{NS}}(\pi)$;
  - For $V_{\pi \in \Pi_{NS}} = \sum_h = 1^H r_h$.
- EXO-BMDP: Essentially a Block MDP [1] such that the latent states admits the form $z = (s, e)$, where $s \in \mathcal{S}$, $e \in E$.
- $\mu(z) = \mu(s)\mu\xi$ and $T(z'|z, a) = T(s'|s, a)T_e(e'|e)$

AC State Algorithm

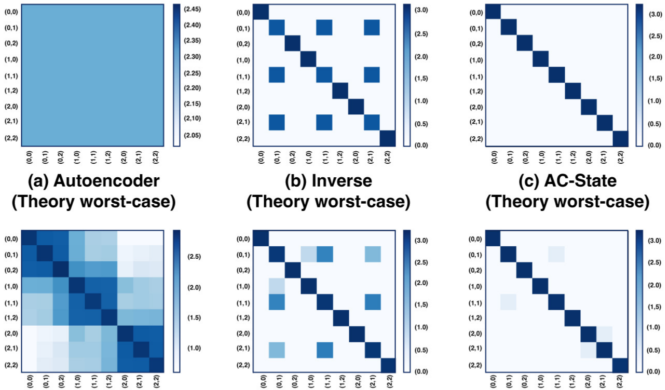
Algorithm 1 AC-State Algorithm for Latent State Discovery Using a Uniform Random Policy

1: Initialize observation trajectory x and action trajectory a. Initialize encoder f_θ. Assume any pair of states are reachable within exactly K steps and a number of samples to collect T, and a set of actions $\mathcal{A}$, and a number of training iterations N.
2: $x_1 \sim U(\mu(x))$
3: **for** t = 1, 2, ..., T **do**
4: $\quad a_t \sim U(\mathcal{A})$
5: $\quad x_{t+1} \sim \mathbb{P}(x'|x_t, a_t)$
6: **for** n = 1, 2, ..., N **do**
7: $\quad t \sim U(1, T)$ and $k \sim U(1, K)$
8: $\quad \mathcal{L} = \mathcal{L}_{\texttt{AC-State}}(f_\theta, t, x, a, k) + \mathcal{L}_{\text{Bottleneck}}(f_\theta, x_t) + \mathcal{L}_{\text{Bottleneck}}(f_\theta, x_{t+k})$
9: $\quad$ Update θ to minimize $\mathcal{L}$ by gradient descent.

AC State in Action

Exogenous distractors riddance.

Agent Controllable States Representation



(a) Autoencoder (Theory worst-case)

(b) Inverse (Theory worst-case)

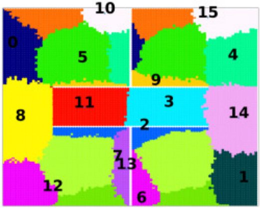
(c) AC-State (Theory worst-case)

(d) Autoencoder (Empirical)

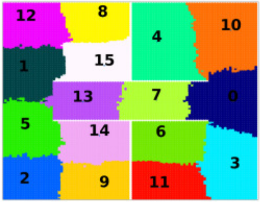
(e) Inverse (Empirical)

(f) AC-State (Empirical)

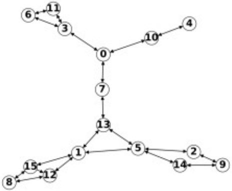
PCLAST Segmentation Results



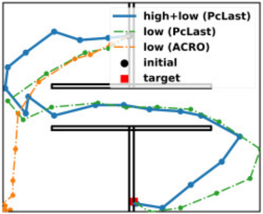
(a) Clusters ACRO



(b) Clusters PCLAST



(c) State-transitions PCLAST



(d) Planning Trajectories

PCLAST Segmentation Results

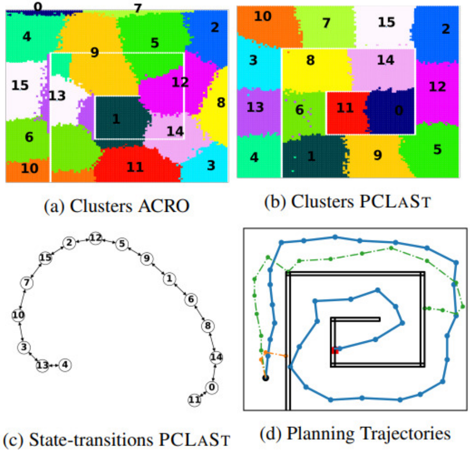
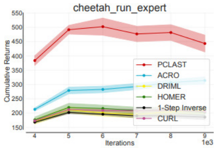
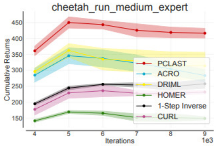
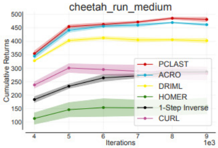


Figure 6. Clustering, Abstract-MDP, and Planning are shown for *Maze-Spiral* environment. Details same as Figure 5.

PCLAST – Cheetah Environment







Morphological Computation

This page is left blank intentionally.

Appendix – SoRos

This page is left blank intentionally.

Morphological Computation – Overview

- The principle of morphological computation in nature
  - Morphology: shape, geometry, and mechanical properties.
  - Computation: sensorimotor information transmission among geometrical components.
- Morphology and computation in artificial robots
  - Cosserat Continua and reduced soft robot models.
  - **Reductions**: Structural Lagrangian properties and control.
- Towards real-time strain regulation and control
  - **Simplexity**: Hierarchical and fast versatile control with reduced variables.

Credits

Shaoru Chen

Postdoc, MSR

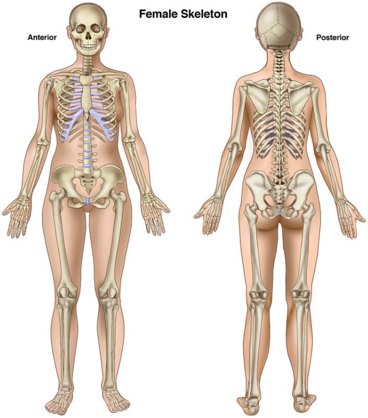
Lekan Molu

Senior Researcher, MSR

Morphology and computation

- Morphology: Emergent behaviors of natural organisms from complex sensorimotor nonlinear mechanical feedback from the environment.
  - Shape affecting behavioral response.
  - Geometrical Arrangement of motors such that processing and perception affect computational characteristics.
  - Mechanical properties that allow the engineering of emergent behaviors via adaptive environmental interaction.
- Computation: The information transformation among the system geometrical units, upon environmental perception, that effect morphological changes in shape and material properties.

MC in vertebrates – a case for soft designs



An adult human skeleton $\approxeq$ 11% of the body mass. ©Brittanica

- The arrangement and compliance of body parts, perception, and computation creates emergence of complex interactive behavior.
- Soft bodies seem critical to the emergence of adaptive natural behaviors.
- Morphological computation is crucial in the design of robots that execute adaptive natural behavior.

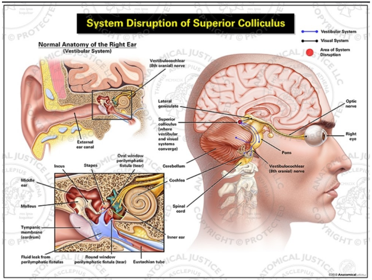
Simplexity in Morphological Computation

- Simplexity: Exploiting structure for effective control.
  - The geometrical tuning of the morphology and neural circuitry in the brain of mammals that simplify the perception and control of complex natural phenomena.
  - Not exactly simplified models or reduced complexity.
  - But rather, sparse connections and finite variables to execute adaptive sensorimotor strategies!
- Example: Saccades (focused eye movements) are controlled by (small) Superior Colliculus in the human brain.
  - Plug: Complex neural circuitry; simple control systems!

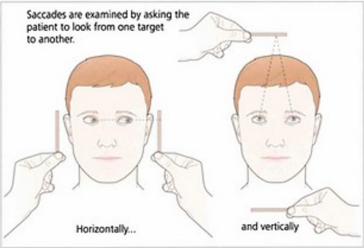
Simplexity: The Central Pattern Generator

- A neural mechanism (in vertebrates) that generates motor control with minimal parameters.
- CPG: Neurons and synapses couple to generate effective motor activation for rhythmic environmental motion.
  - In Lampreys, only two signals trigger swimming motion, for example!
  - This CPG enables indirect use of brain computational power via nonlinear feedback from stretch receptor neurons on Lamprey's skin.

Saccades and the Superior Colliculus



©Anatomical Justice.



Credit: Vision and Learning Center.

Morphing in Invertebrates: Cephalopods

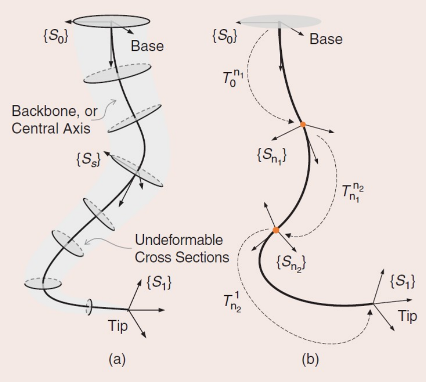
Cuttlefish. ©Monterey Bay Museum

Octopus. ©Smithsonian Magazine

The Octopus and Cuttlefish

- No exoskeleton, or spinal cord.
- A muscular hydrostat: transversal, longitudinal, and oblique muscles along richly innervated arms and mechanoreceptors:
  - Allows for bending, stretching, stiffening, and retraction.
  - Diverse compliance across eight arms imply sophisticated motion strategies in the wild!
- Simplexity enhanced by a peripheral nervous system and a central nervous system.

Soft Robot Mechanism in Focus

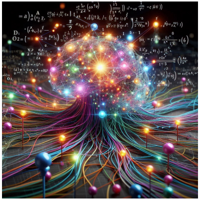


A continuum soft robot whose mechanics can be well-described with Cosserat rod theory. Reprinted from (**(author?)** [2])

- One dimension is quintessentially longer than the other two.
- Characterized by a central axis with undeformable discs that characterize deformable cross-sectional segments.
- Strain and deformation, via e.g. Cosserat rod theory, enables precise finite-dimensional mathematical models.

A Finite and Reliable Model

- A soft robot's usefulness is informed by control system that melds its body deformation with internal actuators.
- By design, this calls for a high-fidelity model or a delicate balancing of complex morphology and data-driven methods.



- Non-interpretable; non-reliable.
- ×Continuous coupled interaction between the material, actuators, and external affordances.

The case for model-based control

- Soft robots are infinite degrees-of-freedom continua i.e., PDEs are the main tools for analysis.
- Nonlinear PDE theory is tedious and computationally intensive.
- Notable strides in reduced-order, finite-dimensional mathematical models that induce tractability in continuum models.

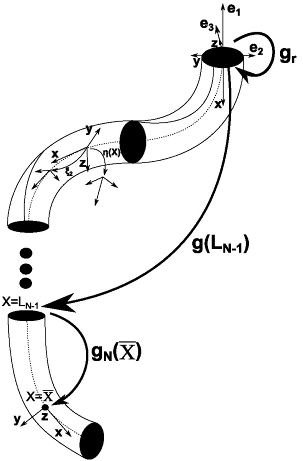
Tractable reduced-order models

- Morphoelastic filament theory: [8; 5; 3];
- Generalized Cosserat rod theory: [14; 1];
- The constant curvature model: [4];
- The piecewise constant curvature model: [15; 9]; and
- Ordinary differential equations-based discrete Cosserat model: [11; 10].

Cosserat-based piecewise constant strain model

- A discrete Cosserat model: **(author?)** [10].
  - Shapes defined by a finite-dimensional functional space, parameterized by a curve, $X : [0, L]$..
  - Assumes constant strains between finite nodal points on robot's body.
  - Strain-parameterized dynamics on a reduced special Euclidean-3 group (SE(3)).

The piecewise constant strain model



Credit: [10].

- C-space: $g(X) : X \to \mathbb{SE}(3) = \begin{pmatrix} \boldsymbol{R}(X) & \boldsymbol{p}(X) \\ 0^\top & 1 \end{pmatrix}$.
- Strain and twist vectors: $\{\boldsymbol{\eta}, \boldsymbol{\xi}\} \in \mathbb{R}^6$.
  - $\{\boldsymbol{\eta}, \boldsymbol{\xi}\} := \{\boldsymbol{q}, \dot{\boldsymbol{q}}\}$
- Strain field: $\breve{\eta}(X) = g^{-1}\partial g/\partial X$.
- Twist field: $\breve{\xi}(X) = g^{-1}\partial g/\partial t$.

The piecewise constant strain model

- $X \in [0, L]$ is divided into N intervals: $[0, L_1], \cdots, [L_{N-1}, L_N]$.
- In [10]'s proposition, the robot's mass divides into N discrete sections $\{\mathcal{M}_n\}_{n=1}^{N}$;
- Each with constant strain $\boldsymbol{\eta}_n$
- Strain field: $\breve{\boldsymbol{\eta}}(X) = g^{-1}\partial g/\partial X$.
- Twist field: $\breve{\boldsymbol{\xi}}(X) = g^{-1}\partial g/\partial t$.

Dynamic equations

From the continuum equations for a cable-driven soft arm [[12]], we can derive the following dynamic equation [[10]]:

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

Structural properties – mass inertia operator

$$\boldsymbol{M}(\boldsymbol{q})\ddot{\boldsymbol{q}} + \left[\boldsymbol{C}_1(q, \dot{\boldsymbol{q}}) + \boldsymbol{C}_2(\boldsymbol{q}, \dot{\boldsymbol{q}})\right]\dot{\boldsymbol{q}} = \boldsymbol{F}(\boldsymbol{q}) + \boldsymbol{N}(\boldsymbol{q})\mathrm{Ad}_{\boldsymbol{g}_r}^{-1}\mathcal{G} + \tau(\boldsymbol{q}) - \boldsymbol{D}(\boldsymbol{q}, \dot{\boldsymbol{q}})\dot{\boldsymbol{q}}. \tag{2}$$

Property 1 (Boundedness of the Mass Matrix)

The mass inertial matrix $\boldsymbol{M}(\boldsymbol{q})$ is uniformly bounded from below by $m\boldsymbol{I}$ where m is a positive constant and $\boldsymbol{I}$ is the identity matrix.

Proof of Property 1.

This is a restatement of the lower boundedness of $\boldsymbol{M}(\boldsymbol{q})$ for fully actuated n-degrees of freedom manipulators [[13]]. □

Structural properties – parameters Identification

Property 2 (Linearity-in-the-parameters)

There exists a constant vector $\boldsymbol{\Theta} \in \mathbb{R}^{l}$ *and a regressor function* $Y(\boldsymbol{q}, \dot{\boldsymbol{q}}, \ddot{\boldsymbol{q}}) \in \mathbb{R}^{N \times l}$ *such that*

$$\boldsymbol{M}(\boldsymbol{q})\ddot{(\boldsymbol{q})} + [\boldsymbol{C}_1(\boldsymbol{q}, \dot{\boldsymbol{q}}) + \boldsymbol{C}_2(\boldsymbol{q}, \dot{\boldsymbol{q}}) + \boldsymbol{D}(\boldsymbol{q}, \dot{\boldsymbol{q}})]\, \dot{\boldsymbol{q}} - \boldsymbol{F}(\boldsymbol{q})\boldsymbol{N}(\boldsymbol{q})Ad_{\boldsymbol{g}_r}^{-1}\mathcal{G} = \boldsymbol{Y}(\boldsymbol{q}, \dot{\boldsymbol{q}}, \ddot{\boldsymbol{q}})\Theta. \tag{3}$$

Structural properties – skew symmetry of system inertial forces

Property 3 (Skew symmetric property)

The matrix $\dot{M}(\boldsymbol{q}) - 2\left[C_1(\boldsymbol{q}, \dot{\boldsymbol{q}}) + C_2(\boldsymbol{q}, \dot{\boldsymbol{q}})\right]$ *is skew-symmetric.*

Skew-symmetric of robot's mass and Coriolis forces

By Leibniz's rule, we have

$$\dot{\boldsymbol{M}}(\boldsymbol{q}) = \frac{d}{dt}\left(\int_0^{L_N} \boldsymbol{J}^T \mathcal{M}_a \boldsymbol{J} dX\right) = \int_0^{L_N} \frac{\partial}{\partial t}\left(\boldsymbol{J}^T \mathcal{M}_a \boldsymbol{J}\right) dX$$
$$\triangleq \int_0^{L_N} \left(\dot{\boldsymbol{J}}^T \mathcal{M}_a \boldsymbol{J} + \boldsymbol{J}^T \dot{\mathcal{M}}_a \boldsymbol{J} + \boldsymbol{J}^T \mathcal{M}_a \dot{\boldsymbol{J}}\right) dX. \tag{4}$$

Therefore, $\dot{\boldsymbol{M}}(\boldsymbol{q}) - 2\left[C_1(\boldsymbol{q}, \dot{\boldsymbol{q}}) + C_2(\boldsymbol{q}, \dot{\boldsymbol{q}})\right]$ becomes

$$\int_0^{L_N} \left(\dot{\boldsymbol{J}}^\top \mathcal{M}_a \boldsymbol{J} + \boldsymbol{J}^\top \dot{\mathcal{M}}_a \boldsymbol{J} + \boldsymbol{J}^\top \mathcal{M}_a \dot{\boldsymbol{J}}\right) dX - 2\int_0^{L_N} \left(\boldsymbol{J}^\top \text{ad}^\star_{\boldsymbol{J}\dot{\boldsymbol{q}}} \mathcal{M}_a \boldsymbol{J} + \boldsymbol{J}^\top \mathcal{M}_a \dot{\boldsymbol{J}}\right) dX \tag{5}$$

$$\triangleq \int_0^{L_N} \left(\dot{\boldsymbol{J}}^\top \mathcal{M}_a \boldsymbol{J} + \boldsymbol{J}^\top \dot{\mathcal{M}}_a \boldsymbol{J} - \boldsymbol{J}^\top \mathcal{M}_a \dot{\boldsymbol{J}}\right) dX - 2\int_0^{L_N} \boldsymbol{J}^\top \text{ad}^\star_{\boldsymbol{J}\dot{\boldsymbol{q}}} \mathcal{M}_a \boldsymbol{J} dX. \tag{6}$$

Skew-Symmetric Property Proof

Similarly, $-\left[\dot{\boldsymbol{M}}(\boldsymbol{q}) - 2\left[\boldsymbol{C}_1(\boldsymbol{q},\dot{\boldsymbol{q}}) + \boldsymbol{C}_2(\boldsymbol{q},\dot{\boldsymbol{q}})\right]\right]^\top$ expands as

$$-\dot{\boldsymbol{M}}^\top(\boldsymbol{q}) + 2\left[\boldsymbol{C}_1^\top(\boldsymbol{q},\dot{\boldsymbol{q}}) + \boldsymbol{C}_2^\top(\boldsymbol{q},\dot{\boldsymbol{q}})\right] =$$

$$\int_0^{L_N} dX^\top \left(-\boldsymbol{J}^\top \mathcal{M}_a \dot{\boldsymbol{J}} - \boldsymbol{J}^\top \dot{\mathcal{M}}_a \boldsymbol{J} - \dot{\boldsymbol{J}}^\top \mathcal{M}_a \boldsymbol{J}\right) + 2\int_0^{L_N} dX^\top \left(\boldsymbol{J}^\top \mathcal{M}_a \mathrm{ad}_{\boldsymbol{J}\dot{\boldsymbol{q}}} \boldsymbol{J} + \dot{\boldsymbol{J}}^\top \mathcal{M}_a \boldsymbol{J}\right)$$

$$\triangleq \int_0^{L_N} \left(\boldsymbol{J}^\top \mathcal{M}_a \dot{\boldsymbol{J}} - \dot{\boldsymbol{J}}^\top \mathcal{M}_a \boldsymbol{J} - \boldsymbol{J}^\top \dot{\mathcal{M}}_a \boldsymbol{J}\right) dX - 2\int_0^{L_N} J^\top \mathrm{ad}^\star_{\boldsymbol{J}\dot{\boldsymbol{q}}} \mathcal{M}_a \boldsymbol{J} dX \qquad (7)$$

which satisfies the identity:

$$\dot{\boldsymbol{M}}(\boldsymbol{q}) - 2\left[\boldsymbol{C}_1(\boldsymbol{q},\dot{\boldsymbol{q}}) + \boldsymbol{C}_2(\boldsymbol{q},\dot{\boldsymbol{q}})\right] = -\left[\dot{\boldsymbol{M}}(\boldsymbol{q}) - 2\left[\boldsymbol{C}_1(\boldsymbol{q},\dot{\boldsymbol{q}}) + \boldsymbol{C}_2(\boldsymbol{q},\dot{\boldsymbol{q}})\right]\right]^\top. \qquad (8)$$

A fortiori, the skew symmetric property follows.

MC Takeaways: Simplexity

- Simplexity: Reliance on a few parameters to model an infinite-DoF system:

$$\boldsymbol{M}(\boldsymbol{q})\ddot{\boldsymbol{q}} + [\boldsymbol{C}_1(\boldsymbol{q}, \dot{\boldsymbol{q}}) + \boldsymbol{C}_2(\boldsymbol{q}, \dot{\boldsymbol{q}})]\,\dot{\boldsymbol{q}} = \boldsymbol{F}(\boldsymbol{q}) + \boldsymbol{N}(\boldsymbol{q})\mathrm{Ad}_{\boldsymbol{g}_r}^{-1}\mathcal{G} + \tau(\boldsymbol{q}) - \boldsymbol{D}(\boldsymbol{q}, \dot{\boldsymbol{q}})\dot{\boldsymbol{q}}.$$

- Simplexity: From PDE to ODE, i.e. inifinite-dimensional analysis (Continuum PDE) to finite-dimensional ODE!

Control exploiting structural properties

Regarding the generalized torque $\boldsymbol{\tau}(\boldsymbol{q})$ as a control input, $\boldsymbol{u}(\boldsymbol{q}, \dot{\boldsymbol{q}})$, feedback laws are sufficient for attaining a desired soft body configuration.

Theorem 1 (Cable-driven Actuation)

For positive definite diagonal matrix gains $\boldsymbol{K}_D$ and $\boldsymbol{K}_p$, without gravity/buoyancy compensation, the control law

$$\boldsymbol{u}(\boldsymbol{q}, \dot{\boldsymbol{q}}) = -\boldsymbol{K}_p \tilde{\boldsymbol{q}} - \boldsymbol{K}_D \dot{\boldsymbol{q}} - \boldsymbol{F}(\boldsymbol{q}) \tag{9}$$

under a cable-driven actuation globally asymptotically stabilizes system (2), where $\tilde{\boldsymbol{q}}(t) = \boldsymbol{q}(t) - \boldsymbol{q}^d$ is the joint error vector for a desired equilibrium point $\boldsymbol{q}^d$.

Computational Control exploiting structural properties

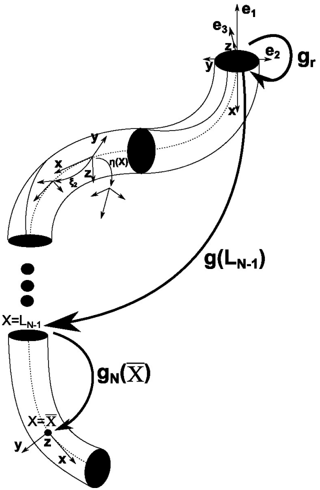
Corollary 2 (Fluid-driven actuation)

If the robot is operated without cables, and is driven with a dense medium such as pressurized air or water, then the term $F(\boldsymbol{q}) = 0$ *so that the control law* $\boldsymbol{u}(\boldsymbol{q}, \dot{\boldsymbol{q}}) = -\boldsymbol{K}_p \tilde{\boldsymbol{q}} - \boldsymbol{K}_D \dot{\boldsymbol{q}}$ *globally asymptotically stabilizes the system.*

Proof.

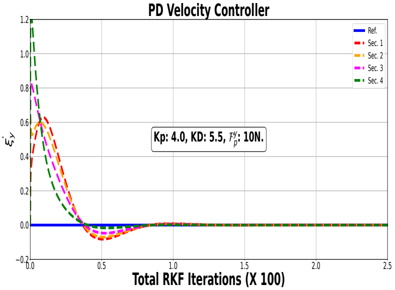
Proofs in Section V of **(author?)** [7]. □

Robot parameters

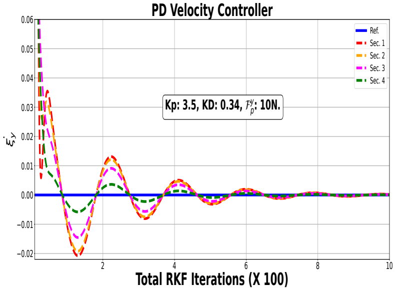


- Tip load in the $+y$ direction in the robot's base frame.
- Poisson ratio: 0.45; $\mathcal{M} = \rho[I_x, I_y, I_z, A, A, A]$ with $\rho = 2,000 kgm^{-3}$;
- $\boldsymbol{D} = -\rho_w \nu^T \nu \check{\boldsymbol{D}} \nu / |\nu|$.
- $X \in [0, L]$ discretized into 41 segments.

Computational Control exploiting structural properties

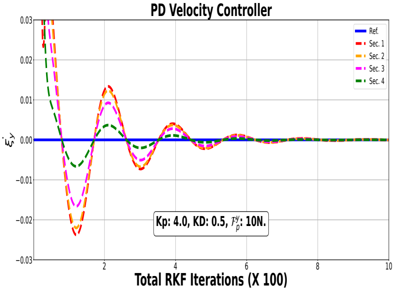


Cable-driven, strain twist setpoint terrestrial control.

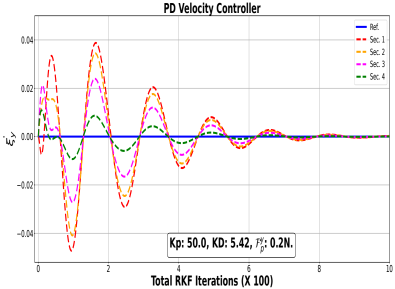


Fluid-actuated, strain twist setpoint terrestrial control.

Computational Control exploiting structural properties

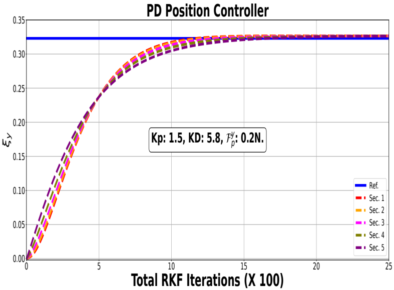


Fluid-actuated, strain twist setpoint underwater control.

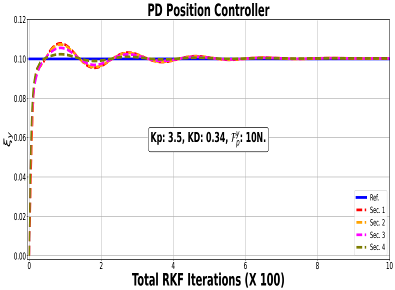


Cable-driven, strain twist setpoint regulation.

Computational Control exploiting structural properties



Cable-based position control with a small tip load, 0.2N.



Terrestrial position control.

Exploiting Mechanical Nonlinearity for Feedback!

This page is left blank intentionally.

Hierarchical Dynamics and Control

- Reaching steps towards the real-time strain control of multiphysics, multiscale continuum soft robots.
- Separate subdynamics — aided by a perturbing time-scale separation parameter.
- Respective stabilizing nonlinear backstepping controllers.
- Stability of the interconnected singularly perturbed. system.
- Fast numerical results on a single arm of the Octopus robot arm.

Decomposition of SoRo Rod Dynamics

- $\mathcal{M}_i^{\text{core}}$: composite mass distribution as a result of microsolid i's barycenter motion;
- $\mathcal{M}^{\text{pert}}$: motions relative to $\mathcal{M}_i^{\text{core}}$, considered as a perturbation;
- $\mathcal{M} = \mathcal{M}^{\text{pert}} \cup \mathcal{M}^{\text{core}}$.
- Introduce the transformation: $[\boldsymbol{q}, \dot{\boldsymbol{q}}] = [\boldsymbol{q}, \boldsymbol{z}]$, rewrite (2):

$$\boldsymbol{M}(\boldsymbol{q})\dot{\boldsymbol{z}} + [\boldsymbol{C}_1(\boldsymbol{q},\boldsymbol{z}) + \boldsymbol{C}_2(\boldsymbol{q},\boldsymbol{z}) + \boldsymbol{D}(\boldsymbol{q},\boldsymbol{z})]\,\boldsymbol{z} - \boldsymbol{F}(\boldsymbol{q}) - \boldsymbol{N}(\boldsymbol{q})\text{Ad}_{\boldsymbol{g}_r}^{-1}\mathcal{G} = \tau(\boldsymbol{q})$$

Dynamics separation

Suppose that $\boldsymbol{M}^p = \int_{L_{\text{min}}^p}^{L_{\text{max}}^p} \boldsymbol{J}^\top \boldsymbol{\mathcal{M}}^{pert} \boldsymbol{J} dX$, and $\boldsymbol{M}^c = \int_{L_{\text{min}}^c}^{L_{\text{max}}^c} \boldsymbol{J}^\top \mathcal{M}^{core} \boldsymbol{J} dX$, then,

$$\boldsymbol{M}(\boldsymbol{q}) = (\boldsymbol{M}^c + \boldsymbol{M}^p)(\boldsymbol{q}), \; \boldsymbol{N} = (\boldsymbol{N}^c + \boldsymbol{N}^p)(\boldsymbol{q}), \tag{10a}$$

$$\boldsymbol{F}(\boldsymbol{q}) = (\boldsymbol{F}^c + \boldsymbol{F}^p)(\boldsymbol{q}), \quad \boldsymbol{D}(\boldsymbol{q}) = (\boldsymbol{D}^c + \boldsymbol{D}^p)(\boldsymbol{q}) \tag{10b}$$

$$\boldsymbol{C}_1(\boldsymbol{q}, \dot{\boldsymbol{q}}) = (\boldsymbol{C}_1^c + \boldsymbol{C}_1^p)(\boldsymbol{q}, \dot{\boldsymbol{q}}), \tag{10c}$$

$$\boldsymbol{C}_2(\boldsymbol{q}, \dot{\boldsymbol{q}}) = (\boldsymbol{C}_2^c + \boldsymbol{C}_2^p)(\boldsymbol{q}, \dot{\boldsymbol{q}}). \tag{10d}$$

Dynamics Separation

Furthermore, let

$$\boldsymbol{M} = \underbrace{\begin{bmatrix} \boldsymbol{\mathcal{H}} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{0} \end{bmatrix}}_{\boldsymbol{M}^c(\boldsymbol{q})} + \underbrace{\begin{bmatrix} \boldsymbol{0} & \boldsymbol{\mathcal{H}}_{\text{slow}}^{\text{fast}} \\ \boldsymbol{\mathcal{H}}_{\text{slow}}^{\text{fast}\top} & \boldsymbol{\mathcal{H}}_{\text{slow}} \end{bmatrix}}_{\boldsymbol{M}^p(\boldsymbol{q})}, \tag{11}$$

where $\boldsymbol{\mathcal{H}}_{\text{slow}}^{\text{fast}}$ denotes the decomposed mass of the perturbed sections of the robot relative to the core sections.

- Let robot's state, $\boldsymbol{x} = [\boldsymbol{q}^\top, \boldsymbol{z}^\top]^\top$ decompose as $\boldsymbol{q} = [\boldsymbol{q}_{\text{fast}}^\top, \boldsymbol{q}_{\text{slow}}^\top]^\top$ and $\boldsymbol{z} = [\boldsymbol{z}_{\text{fast}}^\top, \boldsymbol{z}_{\text{slow}}^\top]^\top$,
- Define $\bar{\boldsymbol{M}}^p = \boldsymbol{M}^p/\epsilon$, and let $\boldsymbol{u} = [\boldsymbol{u}_{\text{fast}}^\top, \boldsymbol{u}_{\text{slow}}^\top]^\top$ be the applied torque.

SoRo Dynamics Separation

$$(\boldsymbol{M}^c + \epsilon \bar{\boldsymbol{M}}^p)\dot{\boldsymbol{z}} = \boldsymbol{s} + \boldsymbol{u}, \tag{12}$$

where

$$\boldsymbol{s} = \begin{bmatrix} \boldsymbol{s}_{\text{fast}} \\ \boldsymbol{s}_{\text{slow}} \end{bmatrix} = \begin{bmatrix} \boldsymbol{F}^c + \boldsymbol{N}^c \text{Ad}_{\boldsymbol{g}_r}^{-1} \boldsymbol{\mathcal{G}} - [\boldsymbol{C}_1^c + \boldsymbol{C}_2^c + \boldsymbol{D}^c]\boldsymbol{z}_{\text{fast}} \\ \boldsymbol{F}^p + \boldsymbol{N}^p \text{Ad}_{\boldsymbol{g}_r}^{-1} \boldsymbol{\mathcal{G}} - [\boldsymbol{C}_1^p + \boldsymbol{C}_2^p + \boldsymbol{D}^p]\boldsymbol{z}_{\text{slow}} \end{bmatrix}. \tag{13}$$

- Since $\boldsymbol{\mathcal{H}}_{\text{fast}}$ is invertible, let

$$\bar{\boldsymbol{M}}^p = \begin{bmatrix} \bar{\boldsymbol{M}}_{11}^p & \bar{\boldsymbol{M}}_{12}^p \\ \bar{\boldsymbol{M}}_{21}^p & \bar{\boldsymbol{M}}_{22}^p \end{bmatrix} \text{ and } \boldsymbol{\Delta} = \begin{bmatrix} \boldsymbol{0} & \boldsymbol{0} \\ \bar{\boldsymbol{M}}_{21}^p \boldsymbol{\mathcal{H}}_{\text{fast}}^{-1} & \boldsymbol{0} \end{bmatrix}. \tag{14}$$

SoRo Dynamics Separation

Premultiplying both sides by $\boldsymbol{I} - \epsilon\boldsymbol{\Delta}$, it can be verified that

$$\begin{bmatrix} \boldsymbol{\mathcal{H}}_{\text{fast}} & \bar{\boldsymbol{M}}_{12}^{p} \\ \boldsymbol{0} & \bar{\boldsymbol{M}}_{22}^{p} \end{bmatrix} \begin{bmatrix} \dot{\boldsymbol{z}}_{\text{fast}} \\ \epsilon\dot{\boldsymbol{z}}_{\text{slow}} \end{bmatrix} = \begin{bmatrix} \boldsymbol{s}_{\text{fast}} \\ \boldsymbol{s}_{\text{slow}} - \epsilon\bar{\boldsymbol{M}}_{21}^{p}\boldsymbol{\mathcal{H}}_{\text{fast}}^{-1}\boldsymbol{s}_{\text{fast}} \end{bmatrix} + \begin{bmatrix} \boldsymbol{u}_{\text{fast}} \\ \boldsymbol{u}_{\text{slow}} - \epsilon\bar{\boldsymbol{M}}_{21}^{p}\boldsymbol{\mathcal{H}}_{\text{fast}}^{-1}\boldsymbol{u}_{\text{fast}} \end{bmatrix} \tag{15}$$

which is in the standard singularly perturbed form (**??**):

$$\dot{\boldsymbol{z}}_1 = \boldsymbol{f}(\boldsymbol{z}_1, \boldsymbol{z}_2, \epsilon, \boldsymbol{u}_s, t), \;\; \boldsymbol{z}_1(t_0) = \boldsymbol{z}_1(0), \;\; \boldsymbol{z}_1 \in \mathbb{R}^{6N}, \tag{16a}$$

$$\epsilon\dot{\boldsymbol{z}}_2 = \boldsymbol{g}(\boldsymbol{z}_1, \boldsymbol{z}_2, \epsilon, \boldsymbol{u}_f, t), \;\; \boldsymbol{z}_2(t_0) = \boldsymbol{z}_2(0), \;\; \boldsymbol{z}_2 \in \mathbb{R}^{6N} \tag{16b}$$

SoRo Fast Subsystem Extraction

On the fast time scale $T = t/\epsilon$, with $dT/dt = 1/\epsilon$ so that,

$$\dot{z}_{\text{fast}} = \frac{dz_{\text{fast}}}{dt} \equiv \frac{1}{\epsilon}\frac{dz_{\text{fast}}}{dT} \triangleq \frac{1}{\epsilon} z'_{\text{fast}}$$

; and

$$\epsilon \dot{z}_{\text{slow}} = z'_{\text{slow}}.$$

Fast subdynamics:

$$z'_{\text{fast}} = \epsilon \mathcal{H}_{\text{fast}}^{-1}(s_{\text{fast}} + u_{\text{fast}}) - \mathcal{H}_{\text{fast}}^{-1} \mathcal{H}_{\text{slow}}^{\text{fast}} z'_{\text{slow}}, \tag{17a}$$

$$z'_{\text{slow}} = \mathcal{H}_{\text{slow}}^{-1}(s_{\text{slow}} - u_{\text{slow}}) - \mathcal{H}_{\text{fast}}^{-1}(s_{\text{fast}} - u_{\text{fast}}) \tag{17b}$$

where the slow variables are frozen on this fast time scale.

SoRo Slow Subsystem Extraction

- We let $\epsilon \to 0$ in (15), so that what is left, i.e.,

$$\dot{\boldsymbol{z}}_{\text{slow}} = \mathcal{H}_{\text{slow}}^{-1}(\boldsymbol{s}_{\text{slow}} + \boldsymbol{u}_{\text{slow}}) \tag{18}$$

  constitutes the system's slow dynamics; where the fast components are frozen on this slow time scale.

This page is left blank intentionally

Control of the Fast Strain Subdynamics

- Consider the transformation: $\begin{bmatrix} \boldsymbol{\theta} \\ \boldsymbol{\phi} \end{bmatrix} = \begin{bmatrix} \boldsymbol{q}_{\text{fast}} \\ \boldsymbol{z}_{\text{fast}} \end{bmatrix}$ so that $\boldsymbol{\theta}' = \epsilon \boldsymbol{z}_{\text{fast}} \triangleq \boldsymbol{\nu} :=$ A virtual input.
- Let $\{\boldsymbol{q}^d_{\text{fast}}, \dot{\boldsymbol{q}}^d_{\text{fast}}\} = \{\boldsymbol{\xi}^d_1, \ldots, \boldsymbol{\xi}^d_{n_\xi}, \boldsymbol{\eta}^d_1, \ldots, \boldsymbol{\eta}^d_{n_\xi}\}_{\text{fast}}$ be the desired joint space configuration for the fast subsystem.

Theorem 3 ([6])

The control law

$$\boldsymbol{u}_{fpos} = \boldsymbol{q}^d_{fast}(t_f) - \boldsymbol{q}_{fast}(t_f) + \boldsymbol{q}'^d_{fast}(t_f)$$

is sufficient to guarantee an exponential stability of the origin of $\boldsymbol{\theta}' = \boldsymbol{\nu}$ such that for all $t_f \geq 0$, $\boldsymbol{q}_{fast}(t_f) \in S$ for a compact set $S \subset \mathbb{R}^{6N}$. That is, $\boldsymbol{q}_{fast}(t_f)$ remains bounded as $t_f \to \infty$.

Control of the Fast Strain Subdynamics

Proof Sketch 1 (Proof of Theorem 3)

$$\boldsymbol{e}_1 = \boldsymbol{\theta} - \boldsymbol{q}_{fast}^{d}, \implies \boldsymbol{e}_1' = \boldsymbol{\theta}' - \boldsymbol{q}_{fast}'^{d} \triangleq \boldsymbol{\nu} - \boldsymbol{q}_{fast}'^{d}. \tag{19}$$

$$Choose\ \boldsymbol{V}_1(\boldsymbol{e}_1) = \frac{1}{2}\boldsymbol{e}_1^\top \boldsymbol{K}_p \boldsymbol{e}_1 \tag{20}$$

$$Then,\ \boldsymbol{V}_1' = \boldsymbol{e}_1^\top \boldsymbol{K}_p \boldsymbol{e}_1' = \boldsymbol{e}_1^\top \boldsymbol{K}_p(\boldsymbol{\nu} - \boldsymbol{q}_{fast}'^{d}). \tag{21}$$

For $\boldsymbol{\nu} = \boldsymbol{q}_{fast}'^{d} - \boldsymbol{e}_1$, $\boldsymbol{V}_1' = -\boldsymbol{e}_1 \boldsymbol{K}_p \boldsymbol{e}_1 \leq 2\boldsymbol{V}_1$.

Stability Analysis of the Fast Velocity Subdynamics

Theorem 4 ([6])

Under the tracking error $\boldsymbol{e}_2 = \phi - \boldsymbol{\nu}$ *and matrices* $(\boldsymbol{K}_p, \boldsymbol{K}_q) = (\boldsymbol{K}_p^\top, \boldsymbol{K}_q^\top) > 0$*, the control input*

$$\boldsymbol{u}_{fvel} = \frac{1}{\epsilon}\mathcal{H}_{fast}[\boldsymbol{q}''^{d}_{fast} + \boldsymbol{e}_1 - 2\boldsymbol{e}_2 - \boldsymbol{K}_q^\top(\boldsymbol{K}_q\boldsymbol{K}_q^\top)^{-1}\boldsymbol{K}_p\boldsymbol{e}_1] + \frac{1}{\epsilon}\mathcal{H}_{slow}^{fast}\boldsymbol{z}'_{slow} - \boldsymbol{s}_{fast} \tag{22}$$

exponentially stabilizes the fast subdynamics (17).

Stability Analysis of Fast Velocity Subdynamics

Proof Sketch 2 (Sketch Proof of Theorem 4)

Recall from the position dynamics controller:

$$\begin{aligned} \boldsymbol{e}_1' &= \boldsymbol{\theta}' - \boldsymbol{q}_{fast}'^{d} \triangleq \boldsymbol{z}_{fast} - \boldsymbol{q}_{fast}'^{d} + (\boldsymbol{\nu} - \boldsymbol{\nu}) && \text{(23a)} \\ &= (\boldsymbol{\phi} - \boldsymbol{\nu}) + (\boldsymbol{\nu} - \boldsymbol{q}_{fast}'^{d}) \triangleq \boldsymbol{e}_2 - \boldsymbol{e}_1. && \text{(23b)} \end{aligned}$$

It follows that

$$\begin{aligned} \boldsymbol{e}_2' &= \boldsymbol{\phi}' - \boldsymbol{\nu}' = \boldsymbol{z}_{fast}' + \boldsymbol{e}_1' - \boldsymbol{q}_{fast}''^{d} && \text{(24)} \\ &= \mathcal{H}_{fast}^{-1}\left[\epsilon \boldsymbol{u}_{fast} + \epsilon \boldsymbol{s}_{fast} - \mathcal{H}_{slow}^{fast} \boldsymbol{z}_{slow}'\right] + (\boldsymbol{e}_2 - \boldsymbol{e}_1) - \boldsymbol{q}_{fast}''^{d}. \end{aligned}$$

Stability Analysis of the Fast Velocity Subdynamics

Proof Sketch 3 (Sketch Proof of Theorem 4)

For diagonal matrices $\boldsymbol{K}_p, \boldsymbol{K}_q$ with positive damping, let us choose the Lyapunov candidate function

$$\boldsymbol{V}_2(\boldsymbol{e}_1, \boldsymbol{e}_2) = \boldsymbol{V}_1 + \frac{1}{2}\boldsymbol{e}_2^\top \boldsymbol{K}_q \boldsymbol{e}_2 = \frac{1}{2}[\boldsymbol{e}_1 \;\; \boldsymbol{e}_2] \begin{bmatrix} \boldsymbol{K}_p & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{K}_q \end{bmatrix} \begin{bmatrix} \boldsymbol{e}_1 \\ \boldsymbol{e}_2 \end{bmatrix}.$$

If $\tilde{\boldsymbol{q}}_{fast} = \boldsymbol{q}_{fast} - \boldsymbol{q}^d_{fast}$ and $\tilde{\boldsymbol{q}}'_{fast} = \boldsymbol{q}'_{fast} - \boldsymbol{q}'^d_{fast}$, then the controller

$$\boldsymbol{u}_{fvel} = \frac{1}{\epsilon}\boldsymbol{\mathcal{H}}_{fast}[\boldsymbol{q}''^d_{fast} - \tilde{\boldsymbol{q}}_{fast} - 2\tilde{\boldsymbol{q}}'_{fast} - \boldsymbol{K}_q^\top(\boldsymbol{K}_q\boldsymbol{K}_q^\top)^{-1}\boldsymbol{K}_p\tilde{\boldsymbol{q}}_{fast}] + \frac{1}{\epsilon}\boldsymbol{\mathcal{H}}^{fast}_{slow}\boldsymbol{z}'_{slow} - \boldsymbol{s}_{fast},$$

exponentially stabilizes the system;

Stability Analysis of the Fast Velocity Subdynamics

Proof Sketch 4 (Sketch Proof of Theorem 4)

since it can be verified that

$$\begin{aligned} \boldsymbol{V}_2' &= \boldsymbol{e}_1^\top \boldsymbol{K}_p(\boldsymbol{e}_2 - \boldsymbol{e}_1) \\ &\quad - \boldsymbol{e}_2^\top \boldsymbol{K}_q \left(\boldsymbol{e}_2 - \boldsymbol{K}_q^\top (\boldsymbol{K}_q \boldsymbol{K}_q^\top)^{-1} \boldsymbol{K}_p \boldsymbol{e}_1\right) && \text{(25a)} \\ &= -\boldsymbol{e}_1^\top \boldsymbol{K}_p \boldsymbol{e}_1 - \boldsymbol{e}_2^\top \boldsymbol{K}_q \boldsymbol{e}_2 && \text{(25b)} \\ &\triangleq -2\boldsymbol{V}_2 \leq 0. && \text{(25c)} \end{aligned}$$

Stability analysis of the slow subdynamics

Set $\boldsymbol{e}_3 = \boldsymbol{z}_{\text{slow}} - \boldsymbol{\nu}$ so that $\dot{\boldsymbol{e}}_3 = \dot{\boldsymbol{z}}_{\text{slow}} - \dot{\boldsymbol{\nu}}$. Then,

$$\dot{\boldsymbol{e}}_3 = \dot{\boldsymbol{z}}_{\text{slow}} - \ddot{\boldsymbol{q}}^d_{\text{fast}} + (\boldsymbol{e}_2 - \boldsymbol{e}_1), \tag{26a}$$

$$= \mathcal{H}^{-1}_{\text{slow}}(\boldsymbol{s}_{\text{slow}} + \boldsymbol{u}_{\text{slow}}) - \ddot{\boldsymbol{q}}^d_{\text{fast}} + (\boldsymbol{e}_2 - \boldsymbol{e}_1). \tag{26b}$$

Theorem 5

The control law

$$\boldsymbol{u}_{slow} = \mathcal{H}_{slow}(\boldsymbol{e}_1 - \boldsymbol{e}_2 - \boldsymbol{e}_3 + \ddot{\boldsymbol{q}}^d_{fast}) - \boldsymbol{s}_{slow} \tag{27}$$

exponentially stabilizes the slow subdynamics.

Stability analysis of the slow subdynamics

Proof.

Consider the Lyapunov function candidate

$$V_3(e_3) = \frac{1}{2} e_3^\top K_r e_3 \text{ where } K_r = K_r^\top > 0. \quad (28)$$

It follows that

$$\dot{V}_3(e_3) = e_3^\top K_r \dot{e}_3 \quad (29a)$$

$$= e_3^\top K_r \left[\mathcal{H}_{slow}^{-1}(s_{slow} + u_{slow}) - \ddot{q}_{fast}^d + e_2 - e_1 \right]. \quad (29b)$$

Substituting u_{slow} in (27), it can be verified that

$$\dot{V}_3(e_3) = e_3^\top K_r e_3 \triangleq -2V_3(e_3) \leq 0. \quad (30)$$

Hence, the controller (27) stabilizes the slow subsystem. □

Stability of the singularly perturbed interconnected system

Let $\varepsilon = (0, 1)$ and consider the composite Lyapunov function candidate $\Sigma(\boldsymbol{z}_{\text{fast}}, \boldsymbol{z}_{\text{slow}})$ as a weighted combination of $\boldsymbol{V}_2$ and $\boldsymbol{V}_3$ i.e. ,

$$\Sigma(\boldsymbol{z}_{\text{fast}}, \boldsymbol{z}_{\text{slow}}) = (1-\varepsilon)\boldsymbol{V}_2(\boldsymbol{z}_{\text{fast}}) + \varepsilon \boldsymbol{V}_3(\boldsymbol{z}_{\text{slow}}),\ 0 < \varepsilon < 1. \tag{31}$$

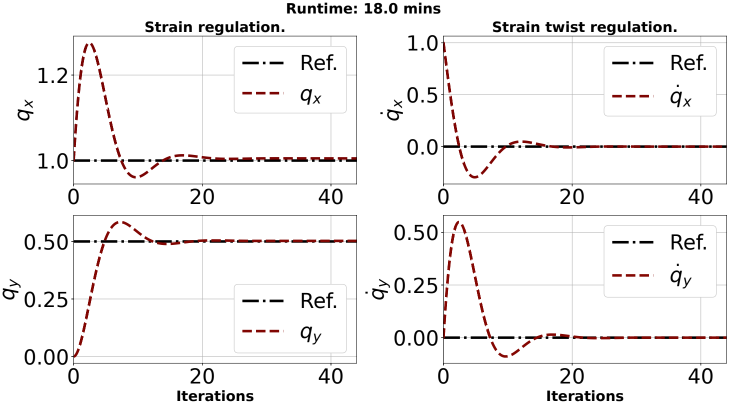
It follows that,

$$\begin{aligned}\dot{\Sigma}(\boldsymbol{z}_{\text{fast}}, \boldsymbol{z}_{\text{slow}}) &= (1-\varepsilon)[\boldsymbol{e}_1^\top \boldsymbol{K}_p \dot{\boldsymbol{e}}_1 + \boldsymbol{e}_2^\top \boldsymbol{K}_q \dot{\boldsymbol{e}}_2] + \varepsilon \boldsymbol{e}_3^\top \boldsymbol{K}_r \dot{\boldsymbol{e}}_3, \\ &= -2(\boldsymbol{V}_2 + \boldsymbol{V}_3) + 2\varepsilon \boldsymbol{V}_2 \le 0\end{aligned} \tag{32}$$

which is clearly negative definite for any $\varepsilon \in (0, 1)$. Therefore, we conclude that the origin of the singularly perturbed system is asymptotically stable under the control laws.

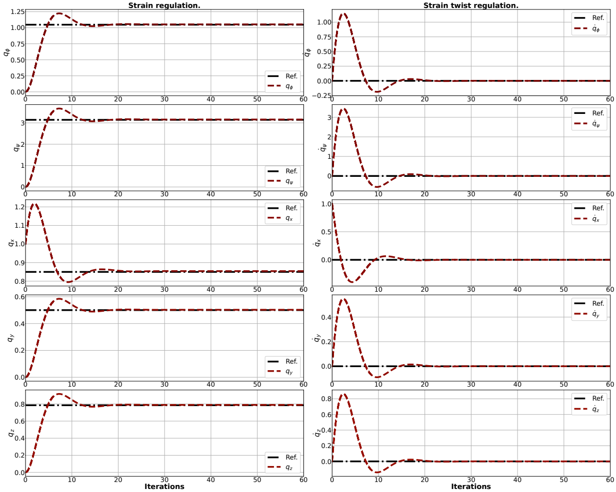
$$\boldsymbol{u}(\boldsymbol{z}_{\text{fast}}, \boldsymbol{z}_{\text{slow}}) = (1-\varepsilon)\boldsymbol{u}_{\text{fast}} + \varepsilon \boldsymbol{u}_{\text{slow}}. \tag{33}$$

Asynchronous, time-separated control



Ten discretized PCS sections: 6 fast, 4 slow subsections. $\mathcal{F}_p^y = 10\,N$, with $\boldsymbol{K}_p = 10$, $\boldsymbol{K}_d = 2.0$ for $\boldsymbol{\eta}^d = [0, 0, 0, 1, 0.5, 0]^\top$ and $\boldsymbol{\xi}^d = \boldsymbol{0}_{6\times 1}$.

Five-axes control



Time Response Comparison with Non-hierarchical Controller

Pieces			Runtime (mins)	
Total	Fast	Slow	Hierarchical SPT (mins)	Single-layer PD control (hours)
6	4	2	18.01	51.46
8	5	3	30.87	68.29
10	7	3	32.39	107.43

Table: Time to Reach Steady State.

Contributions

- Layered singularly perturbed techniques for decomposing system dynamics to multiple timescales.
- Stabilizing nonlinear backstepping controllers were introduced to the respective subdynamics for fast strain regulation.

Discussions

- Leverage the *multiphysics of (often) heterogeneous soft material components*;
- Neat manipulation strategies for motion is a *multiscale problem* that requires imbuing geometric mathematical reasoning into the control strategies for desired movements.
- Challenge: Merging the long-term planning horizon of spatial perception tasks with the *fast time-constant* (typically milliseconds or microseconds) requirements of the precise control of soft, compliant pneumatic/mechanical systems across multiple time-scales;

Discussions

- Process spatial information (Lagrangian) often within a long-time horizon context (Eulerian) for the real-time control or planning across multiple time-scales.

References I

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

[2] Cosimo Della Santina, Christian Duriez, and Daniela Rus. Model-based control of soft robots: A survey of the state of the art and open challenges. *IEEE Control Systems Magazine*, 43(3):30–65, 2023.

[3] 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.

[4] 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.

[5] 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.

[6] Lekan Molu. Fast Whole-Body Strain Regulation in Continuum Robots. *(submitted to) American Control Conference*, 2024.

[7] Lekan Molu and Shaoru Chen. Lagrangian Properties and Control of Soft Robots Modeled with Discrete Cosserat Rods. In *IEEE International Conference on Decision and Control, Milan, Italy*. IEEE, 2024.

[8] 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.

[9] 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.

[10] Federico Renda, Frédéric Boyer, Jorge Dias, and Lakmal Seneviratne. Discrete cosserat approach for multisection soft manipulator dynamics. *IEEE Transactions on Robotics*, 34(6):1518–1533, 2018.

References II

[11] 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.

[12] 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.

[13] José Guadalupe Romero, Romeo Ortega, and Ioannis Sarras. A globally exponentially stable tracking controller for mechanical systems using position feedback. *IEEE Transactions on Automatic Control*, 60(3):818–823, 2014.

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

[15] 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.

References I

Simon Du, Akshay Krishnamurthy, Nan Jiang, Alekh Agarwal, Miroslav Dudik, and John Langford.
Provably efficient rl with rich observations via latent state decoding.
In *International Conference on Machine Learning*, pages 1665–1674. PMLR, 2019.

Yonathan Efroni, Dylan J Foster, Dipendra Misra, Akshay Krishnamurthy, and John Langford.
Sample-efficient reinforcement learning in the presence of exogenous information.
(Accepted for publication at) Conference on Learning Theory, 2022.

Yonathan Efroni, Dipendra Misra, Akshay Krishnamurthy, Alekh Agarwal, and John Langford.
Provably filtering exogenous distractors using multistep inverse dynamics.
In *International Conference on Learning Representations*, 2022.

Deepak Pathak, Pulkit Agrawal, Alexei A Efros, and Trevor Darrell.
Curiosity-driven exploration by self-supervised prediction.
In *International conference on machine learning*, pages 2778–2787. PMLR, 2017.

Tongzhou Wang, Simon S Du, Antonio Torralba, Phillip Isola, Amy Zhang, and Yuandong Tian.
Denoised mdps: Learning world models better than the world itself.
arXiv preprint arXiv:2206.15477, 2022.

Amy Zhang, Rowan McAllister, Roberto Calandra, Yarin Gal, and Sergey Levine.
Learning invariant representations for reinforcement learning without reconstruction.
arXiv preprint arXiv:2006.10742, 2020.