An introduction to modelling and control of flexible and soft robots

Enrico Franco

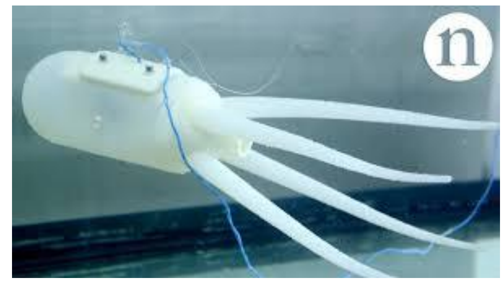
Research Associate
Mechanical Engineering Department, MiM Lab
Imperial College London

Summary

- Motivation: flexible and soft robots
- Modelling and control challenges
- Rigid-link model and port-Hamiltonian formulation
- Overview of port-Hamiltonian and passivity-based control
- Application of passivity-based control to soft robots
- Overview of Mupad and Maple files
- Overview of Matlab code and simulation results
- Limitations and future work

What are soft robots?

How do we define a robot?

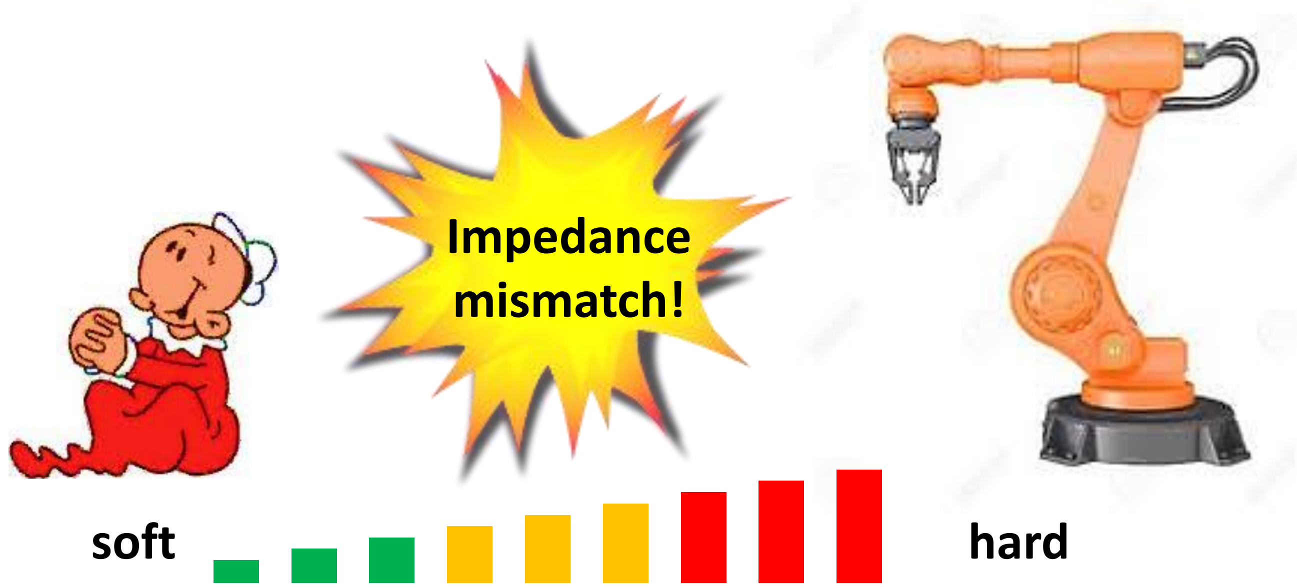


- Mechanism
- Sensors and actuators
- Control and autonomy

.. and a soft robot?

- Deformable bodies
- Stiffness similar to the environment

Why soft robots?



Potential applications

Safe interactions

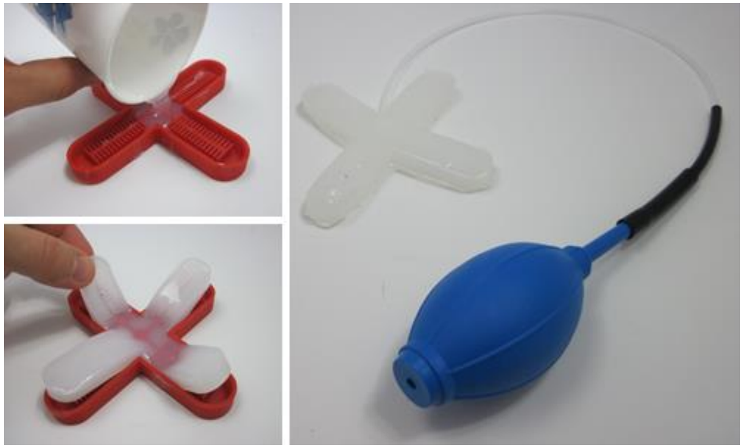
- with people
- with fragile objects
- with other robots

Unstructured environment

- surgery and rehabilitation
- agriculture and retail

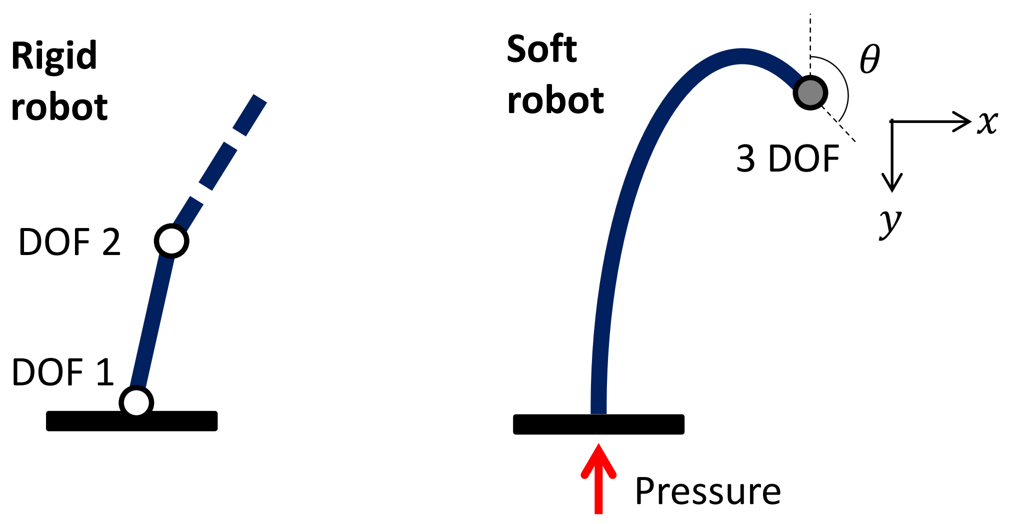
Design & manufacturing

- Material → soft and durable
- Manufacturing → complex shapes
- Sensing → accurate and compact
- Actuation → powerful and lightweight



Modelling and Control

- Deformability → **additional DOF**
- Limited sensing → more DOF than sensors
- Underactuation → more DOF than actuators
- Nonlinearities → material properties, actuator saturation



Typical control aims

Regulation

- Position and orientation
- Tip of the manipulator (optionally intermediate points)

→ drive θ to θ^*

Path following / tracking

- Time-varying setpoint (e.g. pick and place)
- Continuous path (e.g. surgical resection)

→ drive $\theta(t)$ to $\theta^*(t)$

Overview of control approaches

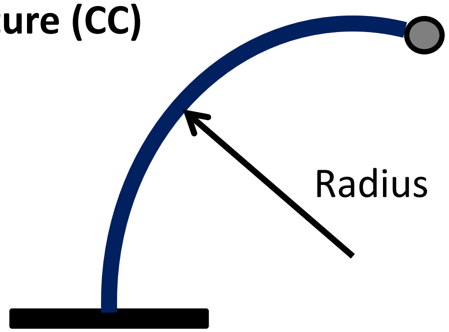
Model free (quasi-static)

- Numerical models, based on training data

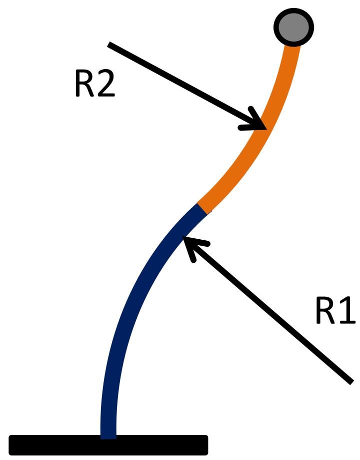
Model based (quasi-static)

- Kinematic models, based on simplifying assumptions (CC or PCC)

Constant curvature (CC)



Piecewise constant curvature (PCC)



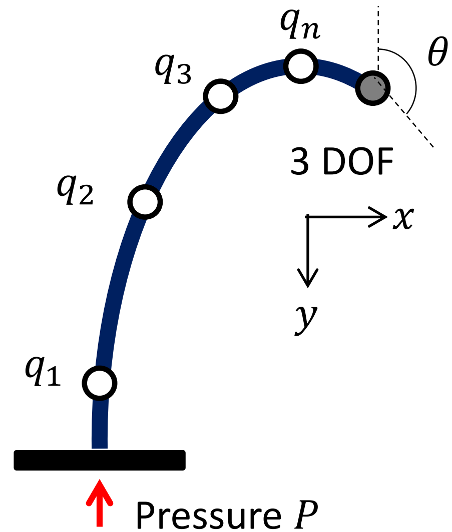
Dynamic control → an open research question... our goal!

Soft robotics: the beginning..

Rigid-link dynamic model

Virtual elastic joints [1]

- But how many? → Enough for kinematic equivalence
- Joint stiffness → $k = P/\theta$ at equilibrium
- It is an approximation!



Actuation and sensing

Controlled pressure $u = P$

Measured states $\theta = q_1 + q_2 + q_3 + q_n$ and $\dot{\theta}$

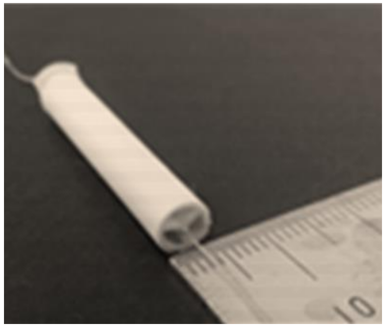
→ **n DOF and only one actuator!**

[1] Yu Y-Q, Howell L L, Lusk C, Yue Y, and He M-G, J. Mech. Des., 2005, (doi).

System energy

Simplifying assumptions

- Linear elasticity and damping
- Planar system (2D), no gravity
- Same control input in all joints



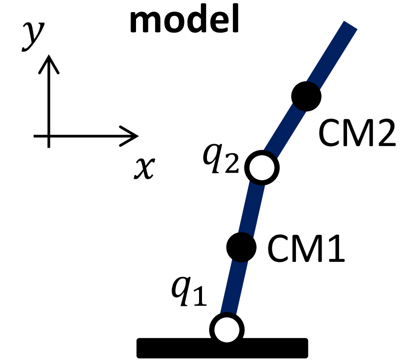
Energy

- Potential energy $V = \frac{1}{2} q^T K q \quad K = \mathrm{diag}\{k\} \quad q = [q_1, \ldots, q_n]$
- Kinetic energy [2] $T = \frac{1}{2} \dot{q}^T \mathcal{M} \dot{q} \qquad M(q) = \mathcal{M}^T > 0$
- Work of external torque $\delta \qquad W = \delta\theta$

[2] Godage I S, Wirz R, Walker I D, and Webster R J, Soft Robotics, 2015, (doi).

Kinetic energy

Example 1 $n = 2$, equal links



position of centre of mass (CM)

$$\begin{cases} x_1 = \frac{l}{2}\sin(q_1) \\ y_1 = \frac{l}{2}\cos(q_1) \end{cases} \text{and} \begin{cases} x_2 = l\sin(q_1) + \frac{l}{2}\sin(q_1 + q_2) \\ y_2 = l\cos(q_1) + \frac{l}{2}\cos(q_1 + q_2) \end{cases}$$

velocity of CM

$$\begin{cases} \dot{x}_1 = \frac{l}{2}\cos(q_1)\,\dot{q}_1 \\ \dot{y}_1 = -\frac{l}{2}\sin(q_1)\dot{q}_1 \end{cases} \text{and} \begin{cases} \dot{x}_2 = l\cos(q_1)\dot{q}_1 + \frac{l}{2}\cos(q_1 + q_2)(\dot{q}_1 + \dot{q}_2) \\ \dot{y}_2 = -l\sin(q_1)\,\dot{q}_1 - \frac{l}{2}\sin(q_1 + q_2)(\dot{q}_1 + \dot{q}_2) \end{cases}$$

$$\longrightarrow \quad T = \frac{1}{2}m\left[\frac{5}{4}(\dot{q}_1)^2 + \frac{1}{4}(\dot{q}_1 + \dot{q}_2)^2 + \cos(q_2)\cos(q_2)\,(\dot{q}_1 + \dot{q}_2)\dot{q}_1\right]$$

Port-Hamiltonian modelling

System states: position $q \in \mathbb{R}^n$ and momenta $p = \mathcal{M}\dot{q}$

$$\begin{bmatrix} \dot{q} \\ \dot{p} \end{bmatrix} = \begin{bmatrix} 0 & I^n \\ -I^n & -R \end{bmatrix} \begin{bmatrix} \nabla_q H \\ \nabla_p H \end{bmatrix} + \begin{bmatrix} 0 \\ G \end{bmatrix} u \qquad \textbf{(1)}$$

$H = \frac{1}{2} p^T \mathcal{M}^{-1} p + V$ (Hamiltonian) u (control input)

$\nabla_q H = \nabla_q V + \frac{1}{2} \nabla_q (p^T \mathcal{M}^{-1} p)$ and $\nabla_p H = \mathcal{M}^{-1} p$

Thus $\dot{q} = \mathcal{M}^{-1} p = \mathcal{M}^{-1} \mathcal{M} \dot{q} = \dot{q}$ identity!

and $\dot{p} = -\nabla_q H - R \nabla_p H$

Example 2 constant $\mathcal{M}$ and $V = k q^T q / 2$

$\dot{p} = \mathcal{M}\ddot{q} = -kq - R\dot{q} + Gu$ or $\mathcal{M}\ddot{q} + R\dot{q} + kq = Gu$

Passivity-based control

- Open Loop $$\begin{bmatrix}\dot{q}\\ \dot{p}\end{bmatrix} = \begin{bmatrix}0 & I^n\\ -I^n & -R\end{bmatrix}\begin{bmatrix}\nabla_q H\\ \nabla_p H\end{bmatrix} + \begin{bmatrix}0\\ G\end{bmatrix} u =$$

- **Closed Loop** [3] $$= \begin{bmatrix}0 & \mathcal{M}^{-1}\mathcal{M}_d\\ -\mathcal{M}_d\mathcal{M}^{-1} & J-D\end{bmatrix}\begin{bmatrix}\nabla_q H_d\\ \nabla_p H_d\end{bmatrix}$$ **(2)**

Open-loop $H = \frac{1}{2}p^T\mathcal{M}^{-1}p + V$

Closed-loop $H_d = \frac{1}{2}p^T\mathcal{M}_d^{-1}p + V_d$ with parameters $\mathcal{M}_d, V_d$ and J, D

[3] Ortega R, Spong M W, Gomez-Estern F, and Blankenstein G, “IEEE Trans. Automatic Control, 2002, (doi).

Full actuation

Control aim

Position regulation: $q^* = \operatorname{argmin}(V_d)$

Fully actuated system $\longrightarrow$ G is invertible!

$$u = G^{-1}\big(\nabla_q H + R\nabla_p H - \mathcal{M}_d \mathcal{M}^{-1}\nabla_q H_d + (J - D)\nabla_p H_d\big) \qquad \textbf{(3)}$$

Example 3 constant $\mathcal{M}$ and $V = \frac{1}{2}kq^T q$ Set $\mathcal{M}_d = \mathcal{M}, J = 0$

Set $V_q = \frac{1}{2}k_p(q - q^*)^T(q - q^*)$ so that $q^* = \operatorname{argmin}(V_d)$

$u = G^{-1}(kq + k_p(q^* - q) - (D - R)\dot{q}) \rightarrow$ **linear controller**

Closed loop $\mathcal{M}\ddot{q} + D\dot{q} + k_p(q - q^*) = 0$ **negative real poles!**

Partial actuation

Partially actuated system → G is not invertible!

Additional conditions on the parameters!

define $G^{\perp}$ as null-space of G

$$G^{\perp}\left(\nabla_q V - \mathcal{M}_d \mathcal{M}^{-1} \nabla_q V_d\right) = 0 \qquad \rightarrow \boldsymbol{V_d}$$

$$G^{\perp}\left(\frac{1}{2}\nabla_q(p^T\mathcal{M}^{-1}p) - \frac{1}{2}\mathcal{M}_d\mathcal{M}^{-1}\nabla_q\left(p^T\mathcal{M}_d^{-1}p\right) + J\nabla_p H_d\right) = 0 \rightarrow \boldsymbol{\mathcal{M}_d}$$

Systems of PDE → **matching on unactuated states**

$$u = G^{\dagger}\left(\nabla_q H - \mathcal{M}_d\mathcal{M}^{-1}\nabla_q H_d + (J - D_1)\nabla_p H_d\right) \qquad \textbf{(4)}$$

where $G^{\dagger} = \left(G^T G\right)^{-1} G^T$

The role of damping

Compare equation (3) and equation (4):

$$u = G^{-1}\left(\nabla_q H + \boldsymbol{R\nabla_p H} - \mathcal{M}_d\mathcal{M}^{-1}\nabla_q H_d + (J - D)\nabla_p H_d\right) \qquad \textbf{(3)}$$

$$u = G^{\dagger}\left(\nabla_q H - \mathcal{M}_d\mathcal{M}^{-1}\nabla_q H_d + (J - D_1)\nabla_p H_d\right) \qquad \textbf{(4)}$$

Physical damping typically affects all positions q

The controller cannot cancel damping on unactuated joints!

Solution $\longrightarrow$ damping is only added by the controller

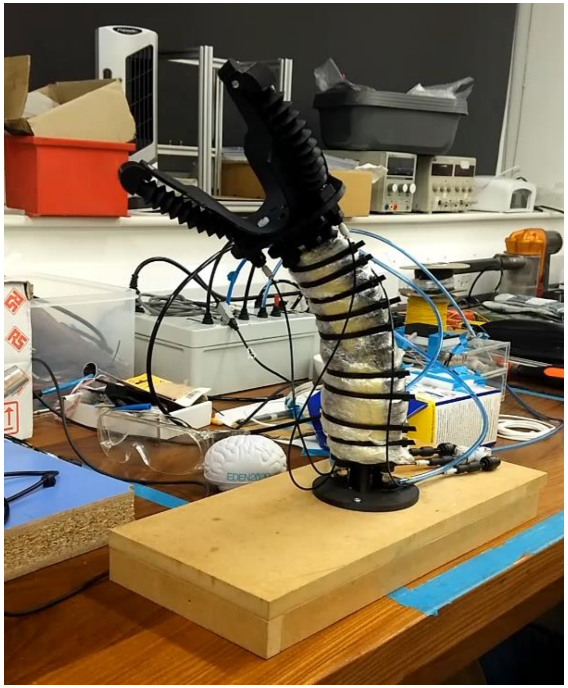
Open loop $\boldsymbol{R\nabla_p H}$ $\longrightarrow$ closed loop $\boldsymbol{D\nabla_p H_d}$

with $D = D_0 + D_1$

$D_0 = R\mathcal{M}_d\mathcal{M}^{-1}$ and damping injection $D_1 = Gk_vG^T$

Half-time review

- Rigid-link model to approximate soft continuum robots
- Overview of port-Hamiltonian formulation
- Passivity-based control for mechanical systems



Solving the PDEs for a rigid-link model

Kinetic PDE setting $\mathcal{M}_d = k_m \mathcal{M}$ solves the PDE with $J = 0$

$$G^{\perp}\left(\frac{1}{2}\underbrace{\nabla_q(p^T \mathcal{M}^{-1} p)}_{} - \frac{1}{2} k_m \underbrace{\nabla_q \left(\frac{1}{k_m} p^T \mathcal{M}^{1} p\right)}_{} + J \nabla_p H_d\right) = 0$$

Potential PDE depends on the specific system

First step $\longrightarrow$ define $G^{\perp}$ such that $G^{\perp} G = 0$ and $\operatorname{rank}\left(\begin{bmatrix} G^T \\ G^{\perp} \end{bmatrix}\right) = n$

Case 1: $n = 2$ and $G^T = [1 \quad 1]$ $G^{\perp} = [1 \quad -1]$ and $\begin{bmatrix} G^T \\ G^{\perp} \end{bmatrix} = \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}$

Case 2: $n = 3$ and $G^T = [1 \quad 1 \quad 1]$ $G^{\perp} = \begin{bmatrix} 1 & 0 & -1 \\ 0 & 1 & -1 \end{bmatrix}$

Solving the potential PDE

Second step $\longrightarrow$ solve the PDE and impose minimizer in q^*

Example 4 $n = 3$ and $V = \frac{1}{2} k q^T q$ with $G^\perp = \begin{bmatrix} 1 & 0 & -1 \\ 0 & 1 & -1 \end{bmatrix}$

- System of 2 PDEs $G^\perp \left(\nabla_q V - \mathcal{M}_d \mathcal{M}^{-1} \nabla_q V_d \right) = 0$

$$\begin{cases} k(q_1 - q_3) = k_m(\nabla_{q_1} V_d - \nabla_{q_3} V_d) \\ k(q_2 - q_3) = k_m(\nabla_{q_2} V_d - \nabla_{q_3} V_d) \end{cases}$$

$\longrightarrow$ **symbolic computing (Maple)**

- Strict minimizer conditions: $\nabla_q V_d(q^*) = 0$ and $\nabla_q^2 V_d(q^*) > 0$

$$V_d = \underbrace{\frac{k}{2k_m}\left(\sum_{i=1}^{3} q_i^2 - \frac{1}{3}\left(\sum_{i=1}^{3} q_i \right)^2 \right)}_{\textbf{Maple solution}} + \underbrace{\frac{k_p}{2}\left(\sum_{i=1}^{3} q_i - \sum_{i=1}^{3} q_i^* \right)^2}_{\textbf{from minimizer condition}}$$

Control law

From the theory $\longrightarrow$ $u = G^{\dagger}\left(\nabla_q H - k_m \nabla_q H_d + (J - D_1)\nabla_p H_d\right)$

with $\mathcal{M}_d = k_m \mathcal{M}$ and $J = 0$.

Set also $D_1 = G k_v G^T$ so that $G^{\perp} D_1 \nabla_p H_d = 0$

$$u = G^{\dagger}\left(\nabla_q V + \cancel{\frac{1}{2}\nabla_q\left(p^T \mathcal{M}^{-1} p\right)} - k_m \nabla_q V_d \cancel{- \frac{1}{2} k_m \nabla_q\left(\frac{1}{k_m} p^T \mathcal{M}^{-1} p\right)}\right) + u_d$$

$$u_d = -k_v G^T \nabla_p H_d$$

Example 5 $\quad n = 3$ and $G^T q = q_1 + q_2 + q_3 = \theta$

$$u = \frac{k}{3}\theta - k_p k_m (\theta - \theta^*) - \frac{k_v}{k_m}\dot{\theta}$$ **linear controller!**

Effect of payload

- Open Loop $$\begin{bmatrix}\dot{q}\\ \dot{p}\end{bmatrix} = \begin{bmatrix}0 & I^n\\ -I^n & -R\end{bmatrix}\begin{bmatrix}\nabla_q H\\ \nabla_p H\end{bmatrix} + \begin{bmatrix}0\\ G\end{bmatrix}u - \begin{bmatrix}0\\ \boldsymbol{\delta}\end{bmatrix} =$$

(5)

- **Closed Loop** $$= \begin{bmatrix}0 & \mathcal{M}^{-1}\mathcal{M}_d\\ -\mathcal{M}_d\mathcal{M}^{-1} & J - D\end{bmatrix}\begin{bmatrix}\nabla_q H_d + \boldsymbol{\Lambda}\\ \nabla_p H_d\end{bmatrix}$$

Matching equations [4] $G^{\perp}\left(\delta - \mathcal{M}_d\mathcal{M}^{-1}\Lambda\right) = 0$

δ is constant, it is measured or estimated!

with Λ vector of closed-loop non-conservative forces

and minimizer condition $\nabla_q V_d(q^*) + \boldsymbol{\Lambda} = 0$

$$u' = u + G^{\dagger}\left(\delta - \mathcal{M}_d\mathcal{M}^{-1}\Lambda\right) \qquad (6)$$

[4] Franco E, Int. J. Adapt. Control Signal Process., 2019, (doi).

Payload compensation

Example 6 $n = 3$ and torque δ (**constant and known**)

- System of 2 equations $G^{\perp}\left(\delta - \mathcal{M}_d \mathcal{M}^{-1}\Lambda\right) = 0$

$$\begin{cases} k(\delta_1 - \delta_3) = k_m(\Lambda_1 - \Lambda_3) \\ k(\delta_2 - \delta_3) = k_m(\Lambda_2 - \Lambda_3) \end{cases}$$

Λ is also a constant vector!

- find Λ that satisfies equation (5) at equilibrium $(q, p) = (q^*, 0)$

$$\longrightarrow \quad \nabla_q V_d(q^*) + \Lambda = 0$$

$$\Lambda^T = \frac{1}{3k_m}\left[2\delta_1 - (\delta_2 + \delta_3), \quad 2\delta_2 - (\delta_1 + \delta_3), \quad 2\delta_3 - (\delta_2 + \delta_1)\right]$$

Control law $u' = \frac{k}{3}\theta - k_p k_m(\theta - \theta^*) - \frac{k_v}{k_m}\dot{\theta} + \frac{1}{3}(\delta_1 + \delta_2 + \delta_3)$

Note on stability

Without disturbances [3] Storage function $H_d = \frac{1}{2} p^T \mathcal{M}_d^{-1} p + V_d$

Lyapunov stability: if $H_d > 0$ and $\dot{H}_d < 0$ $\longrightarrow$ stable equilibrium

For our system $\dot{H}_d = \nabla_q H_d^T \dot{q} + \nabla_p H_d^T \dot{p}$

Substitute equation (2) $\begin{bmatrix} \dot{q} \\ \dot{p} \end{bmatrix} = \begin{bmatrix} 0 & k_m \\ -k_m & -G k_v G^T - R k_m \end{bmatrix} \begin{bmatrix} \nabla_q H_d \\ \nabla_p H_d \end{bmatrix}$

$$\dot{H}_d = \cancel{\nabla_q H_d^T \left(k_m \nabla_p H_d\right)} - \nabla_p H_d^T (\cancel{k_m \nabla_q H_d} + G k_v G^T \nabla_p H_d + R k_m \nabla_p H_d)$$

Finally $\dot{H}_d = -\nabla_p H_d^T \left(G k_v G^T + R k_m\right) \nabla_p H_d \leq 0$

Stability with payload or disturbances (optional) in the references [4,5]

[5] Franco E, Int. J. Control, 2019, (doi).

Review of the theory

- Kinetic PDE $\longrightarrow$ trivial solution $\mathcal{M}_d = k_m \mathcal{M}$
- Potential PDE $\longrightarrow$ define $G^{\perp}, V_d =$ maple solution + free term
- Minimizer in q^* $\longrightarrow$ $\nabla_q V_d(q^*) = 0$ and $\nabla_q^2 V_d(q^*) > 0$
- Effect of payload $\longrightarrow$ additional term in the control input

Overview of symbolic computing software

Matlab Mupad and Symbolic Toolbox

https://uk.mathworks.com/discovery/mupad.html

Maple

https://www.maplesoft.com/support/help/Maple/view.aspx?path=pdsolve

Supporting files

• Maple_n3.mw	code to solve PDEs
• Topics_in_control_n3.mn	symbolic computation
• Topics_in_control_simulations.m	script to run the ODE
• Topics_in_control_ode_n3.m	function containing ODE

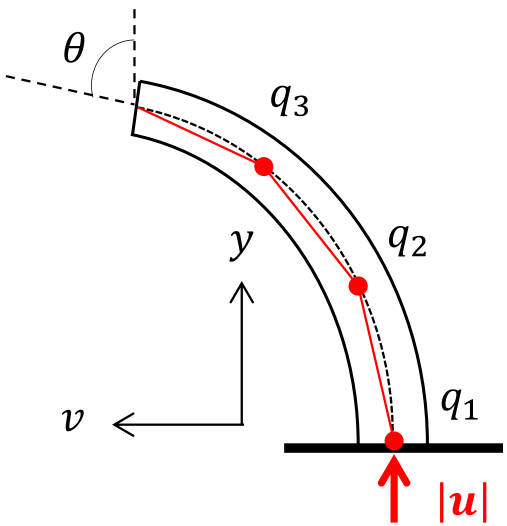
[6] Franco E, Garriga-Casanovas, Int. J. Robotics Research, 2020, (doi).

Simulation results

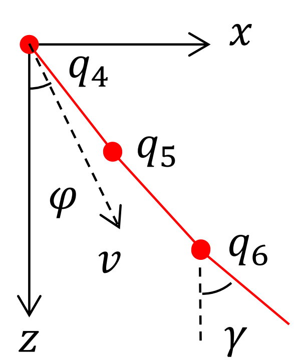
Extension to 3D [6] → two angles (θ, γ)

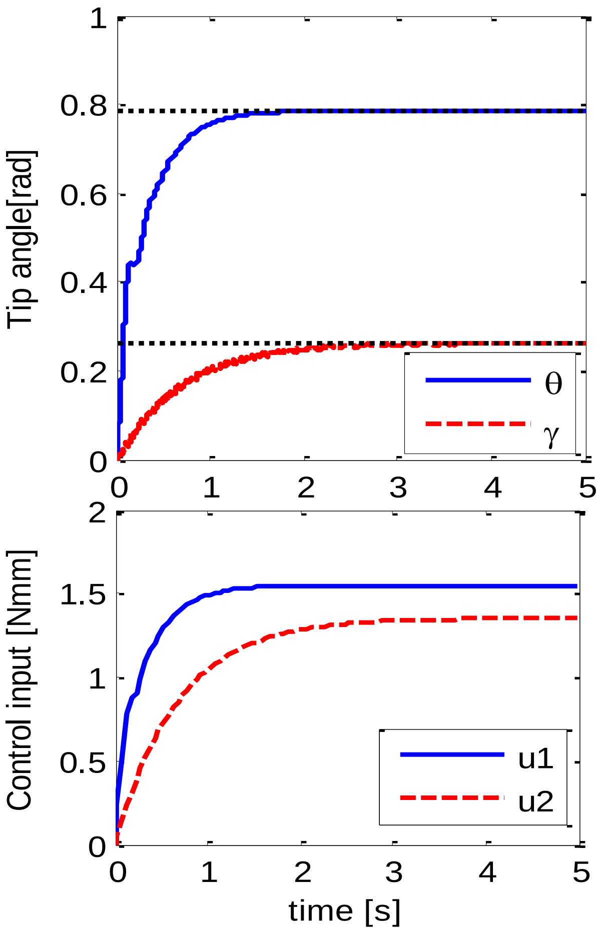
- 3 DOF in plane + 3 DOF out of plane
- regulation $(\theta, \gamma) = (\theta_d, \gamma_d)$

Bending plane



Out of plane





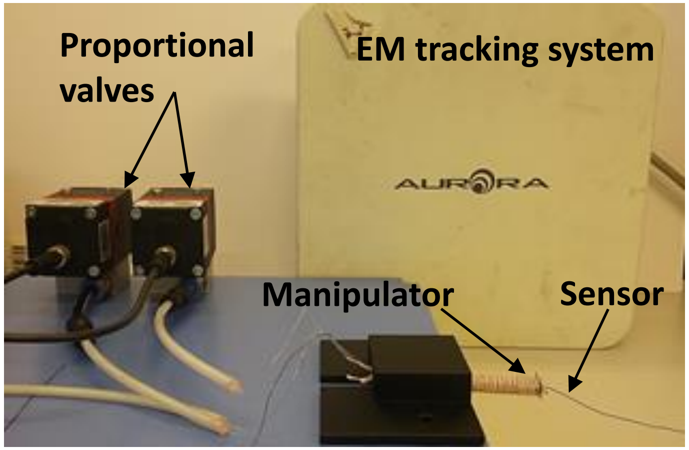
[7] Franco E, Garriga-Casanovas A, Rodriguez y Baena F, Astolfi A, *IEEE Conference on Decision and Control*, 2019, (link).

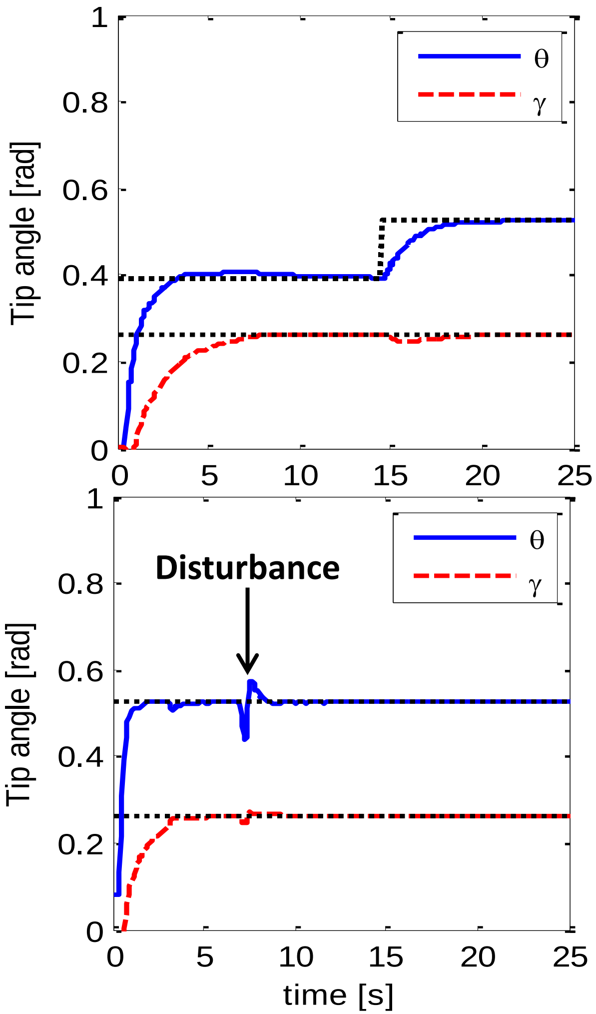
Experimental results

Silicone rubber prototype

- Model uncertainties
- Agreement with simulations
- Disturbance: tip force

Experimental setup

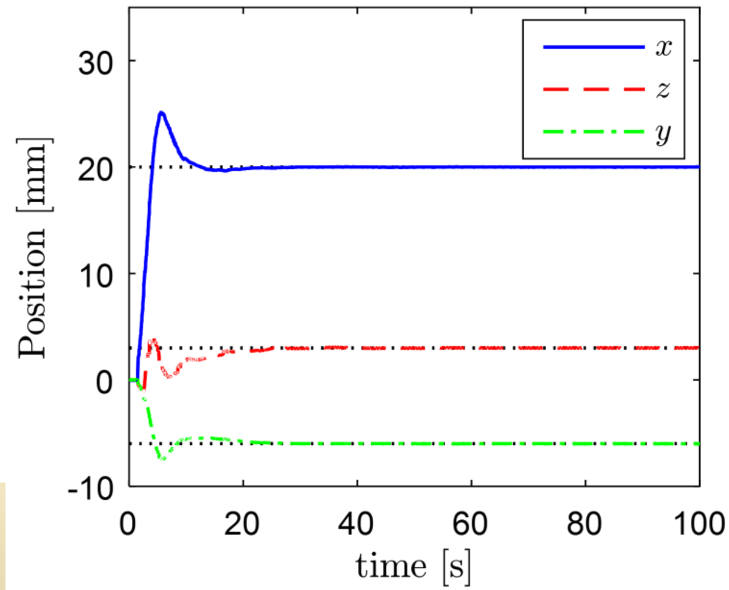




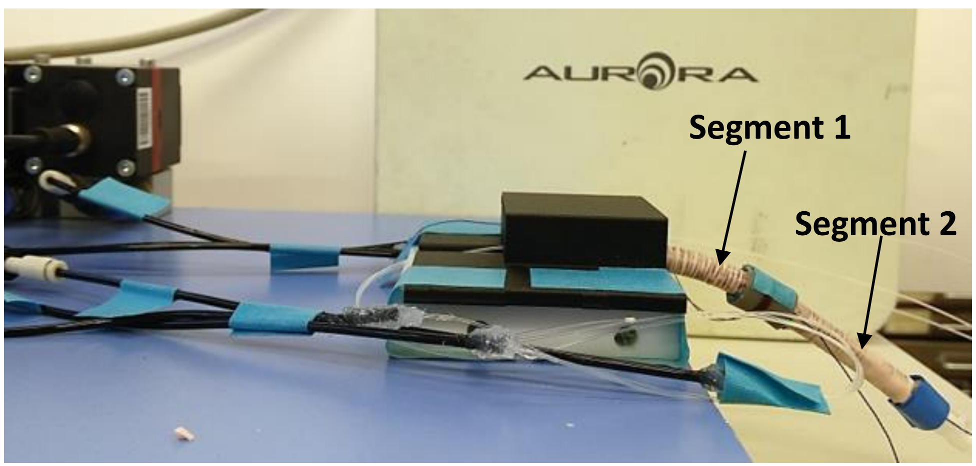
Position control in Cartesian space

Two actuators in series

- Additional control input
- Disturbances (e.g. gravity)
- Kinematic uncertainties



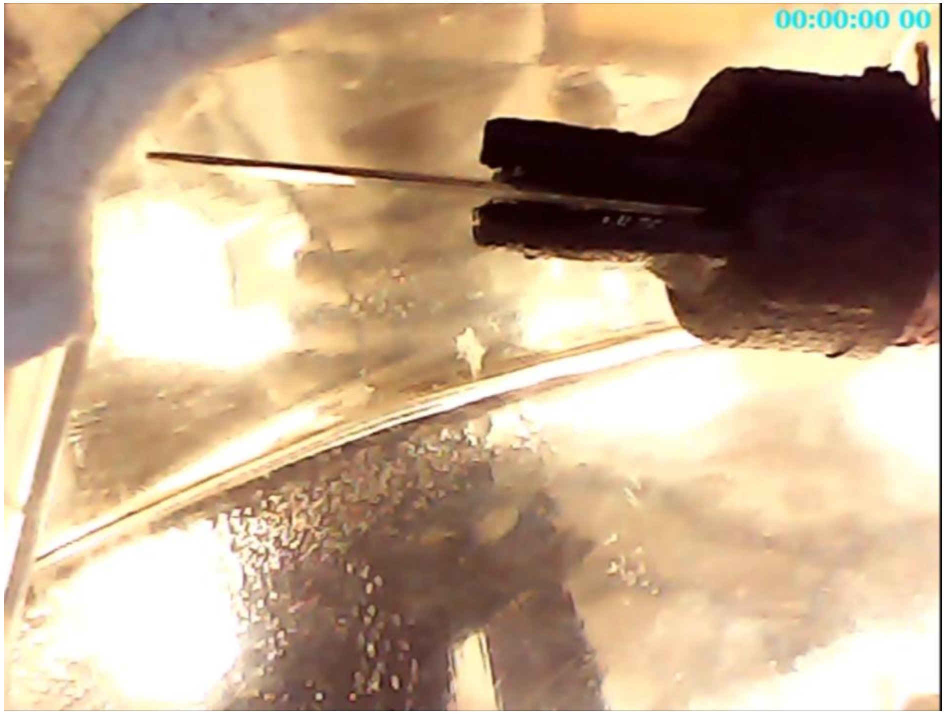
Experimental setup



within the reachable workspace!

Medical application

Internal camera



External camera

Review of learning outcomes

- Soft robotics state-of-the-art
- Modelling of soft robots: rigid-link systems with elastic joint
- Control of soft robots: port-Hamiltonian passivity based control
- Overview of symbolic computation: Maple and Mupad

Ongoing research in control

- Reachable workspace
- Tracking in the presence of disturbances
- Robot-robot and human-robot interactions

References summary

Modelling (background reading)

[1] Yu Y-Q, Howell L L, Lusk C, Yue Y, and He M-G, J. Mech. Des., 2005, (doi).

[2] Godage I S, Wirz R, Walker I D, and Webster R J, Soft Robotics, 2015, (doi).

Passivity based control (background reading)

[3] Ortega R, Spong M W, Gomez-Estern F, and Blankenstein G, IEEE Trans. Automatic Control, 2002, (doi).

Disturbance rejection (not covered in the lecture and not part of the workshop)

[4] Franco E, Int. J. Adapt. Control Signal Process., 2019, (doi).

[5] Franco E, Int. J. Control, 2019, (doi).

Application to soft robotics (all the derivations seen in the slides and much more)

[6] Franco E, Garriga-Casanovas A, In. J. Robotics Research, 2020, (PDF provided).

[7] Franco E, Garriga-Casanovas A, Rodriguez y Baena F, Astolfi A, *IEEE Conference on Decision and Control*, 2019, (link).

Coursework instructions

FILES TO BE SUBMITTED AS PART OF YOUR COURSEWORK

- Word document
- Mupad file “Topics_in_control_n4_solutions”
- Matlab script “Topics_in_control_ode_n4_solutions”
- Maple file “Maple_n4_solutions”

The Word document serves as summary of your submission. Please copy the relevant expressions from Mupad or Maple into Word (paste as image), clearly labelling them and clearly defining all variables and parameters.

Please comment your code following the structure of the files provided. Clearly label variables and parameters in case you chose a different naming convention. Please ensure to include all files in a zip folder with your full name.

The slides and the Matlab files provided should be sufficient to complete the workshop. The references contain supplementary information beyond the scope of the lecture thus they are intended as optional reading material.