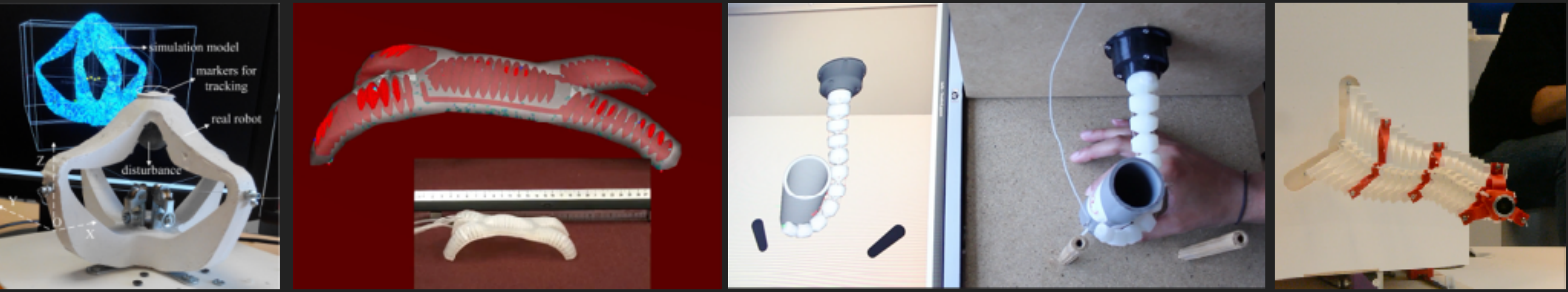
Christian Duriez, head of DEFROST team
Research director at INRIA, FRANCE



JNRH 2020

PHYSICS-BASED MODELING OF DEFORMABLE ROBOTS FOR REAL-TIME SIMULATION AND CONTROL

WHY SOFT ROBOTICS ?

- In nature, bodies are made with soft parts…
    - Is it an advantage in term of design ? it depends on the environment, the task etc…

Is it too complex or too simple ?

human skeleton: 11% of the body mass
skeletal muscles: 42% of the body mass

WHAT IS A « SOFT-ROBOT » ?

- Use of soft materials

Warein Robotics

WHAT IS A « SOFT-ROBOT » ?

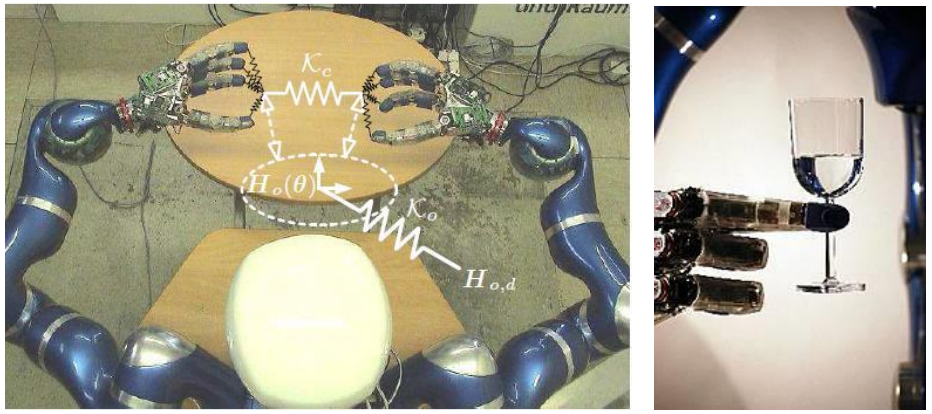
- Use of soft materials
- Deformable structure

DEFINING SOFT ROBOTICS

- Two « definitions » in the literature

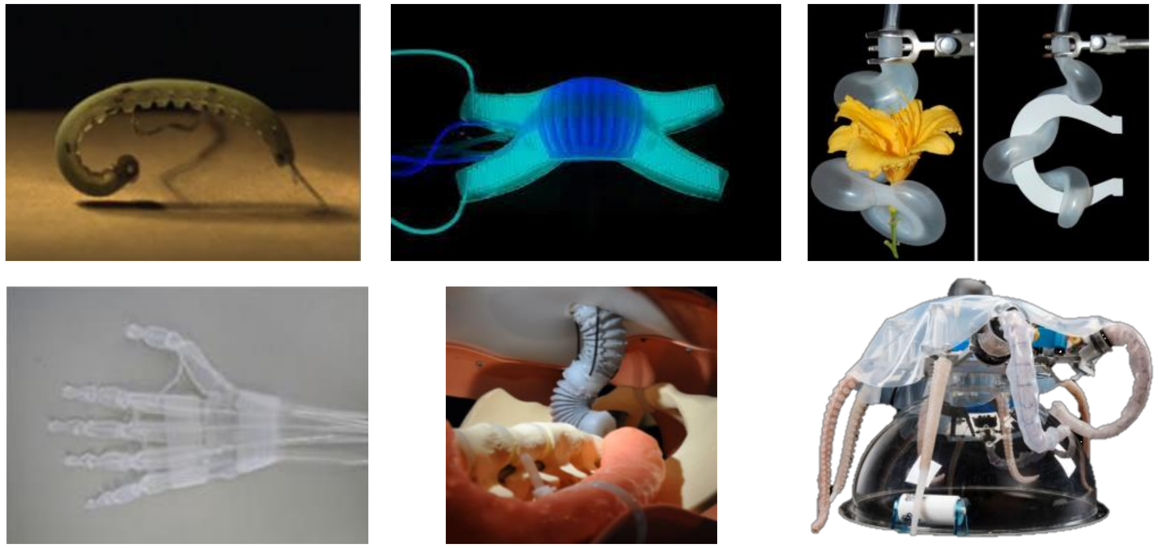
Compliant Joints
(but still articulated rigid structure)

- mechanically (or passively) compliant joints with variable stiffness
- compliance or impedance control



Deformable structure
(the motion of the robot is created by deformation)

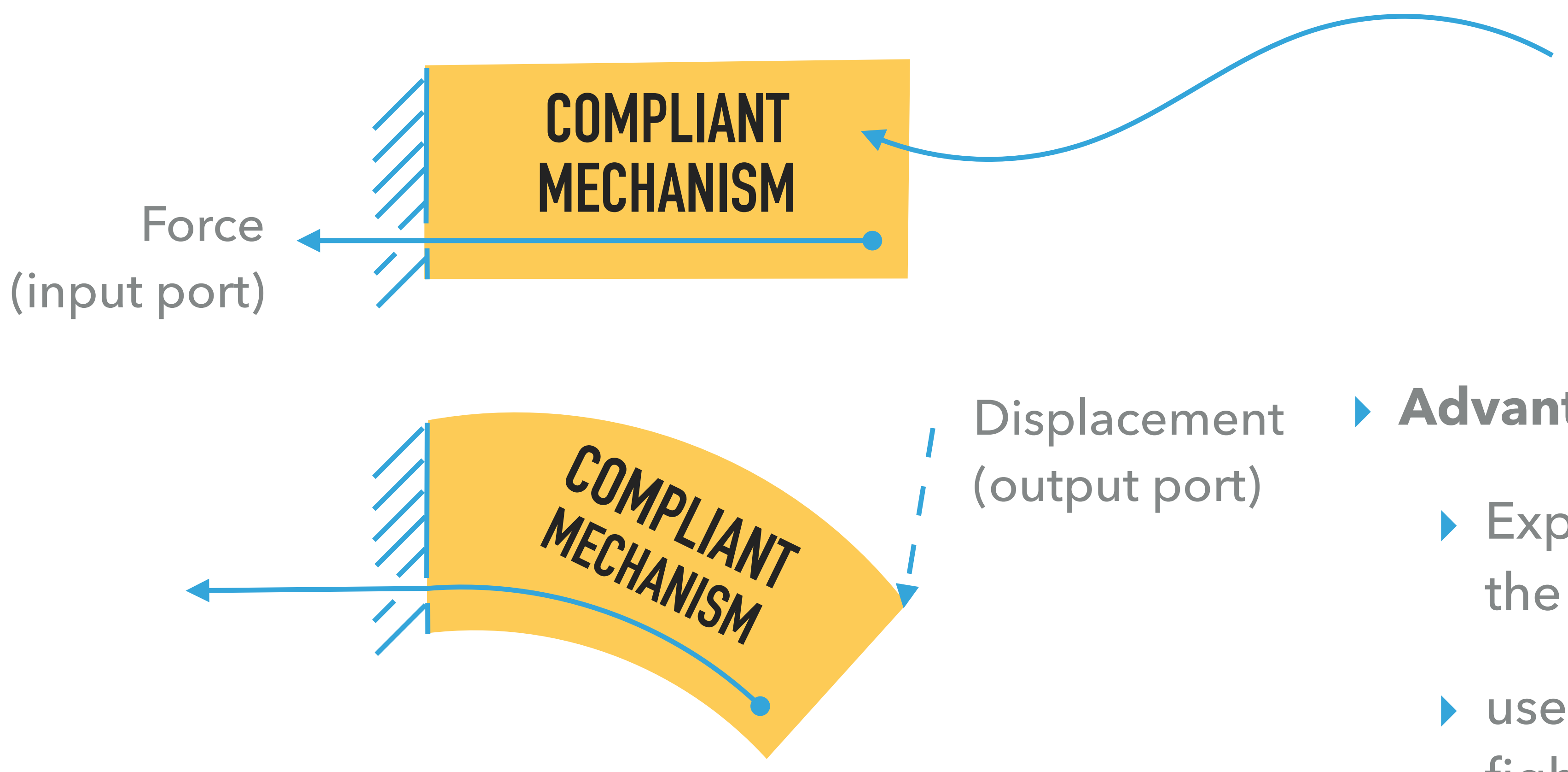
- Robots made of soft materials or structures that undergo high deformations in interaction
- Soft actuators and soft components



DEFINING SOFT ROBOTICS

- Compliant mechanisms

definition: **compliant mechanisms** are flexible mechanisms that transfer an input force and displacement at one port to an output force and displacement at another port through elastic body deformation.



Low elastic modulus
(Soft material)
or
High elastic modulus
(Geometrical structure)

- **Advantages for robotics**
  - Exploit the mechanical feedback from the interaction with the environment
  - use body compliance (instead of fighting it)

WHY NOW ?

PIONEERING WORK

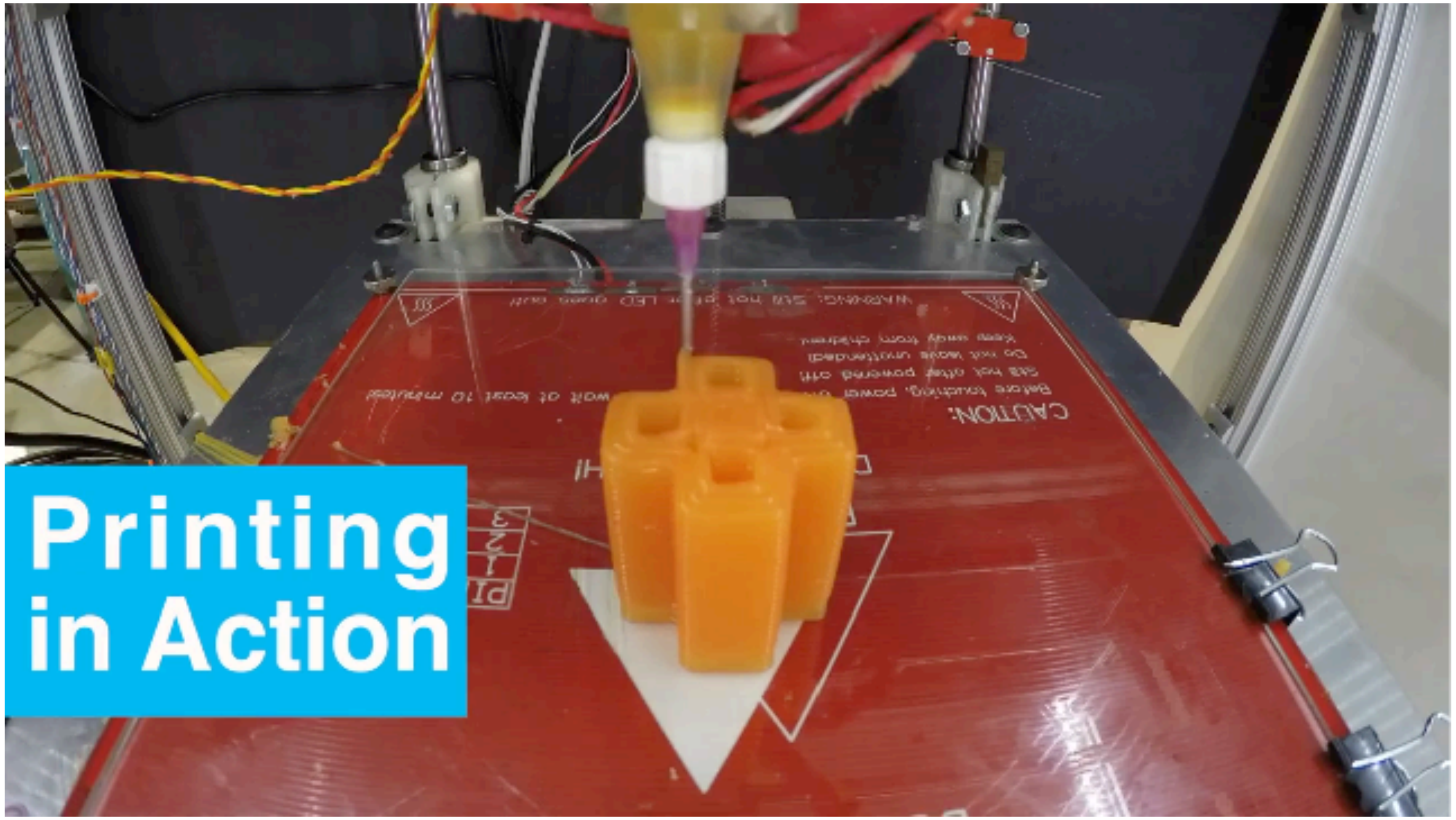
- [Mol78] Molaug, O. (1978). Flexible robot arm U.S. Patent No. 4,107,948. Washington, DC: U.S. Patent and Trademark Office.

- [Hir78] Hirose, S., & Umetani, Y. (1978). The development of soft gripper for the versatile robot hand. Mechanism and machine theory, 13(3), 351-359.

WHY NOW ?

Neri Oxman, MIT

- Link with 3D printing



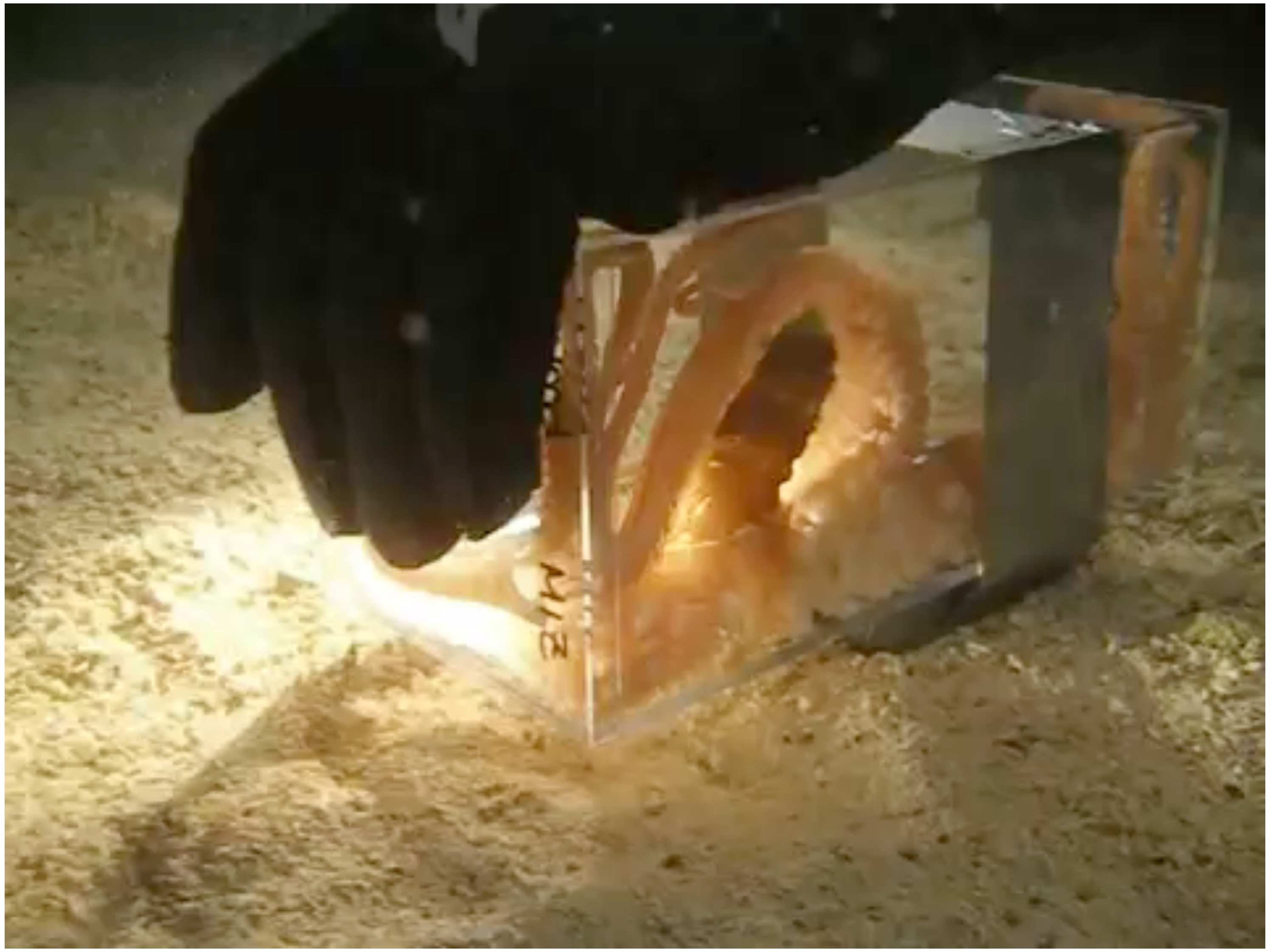
WHY NOW ?

- Mesostructured material

Vanneste et al. RAL (ICRA) 2020

WHY NOW ?

- Bio-inspiration



Octopus has inspired many groups in soft robotics

SUMMARY

- Introduction
  - Why soft robotics ?
  - What is soft robotics ?
  - Why now?
- **Design**
  - Bio-inspiration
  - Soft-robot technology
  - Morphological computation
- **Modeling and simulation**
  - FEM simulation in real-time
  - Constraint-based modeling
- **Control methods**
  - Inverse kinematics
  - Sensing & Closed-loop control
- Perspective / Conclusion

DESIGN

- Bio-inspiration
- Soft-robot technology
- Structure and optimisation

BIOINSPIRATION AND BIOMIMETICS

EXAMPLES OF COMPLIANT ROBOTS INSPIRED BY NATURE

- Elephant trunk
    - Ian Walker, Clemson University

Walker, Ian D., and Michael W. Hannan. "A novel'elephant's trunk'robot." *Advanced Intelligent Mechatronics, 1999. Proceedings. 1999 IEEE/ASME International Conference on*. IEEE, 1999.

EXAMPLES OF COMPLIANT ROBOTS INSPIRED BY NATURE

- Elephant trunk
  - Festo

Festo, INC. Bionic handling assistant –flexible and compliant movement. https://www.festo.com/group/en/cms/10241.htm

EXAMPLES OF COMPLIANT ROBOTS INSPIRED BY NATURE

- Fish
  - Compliant body motion to undulate and move

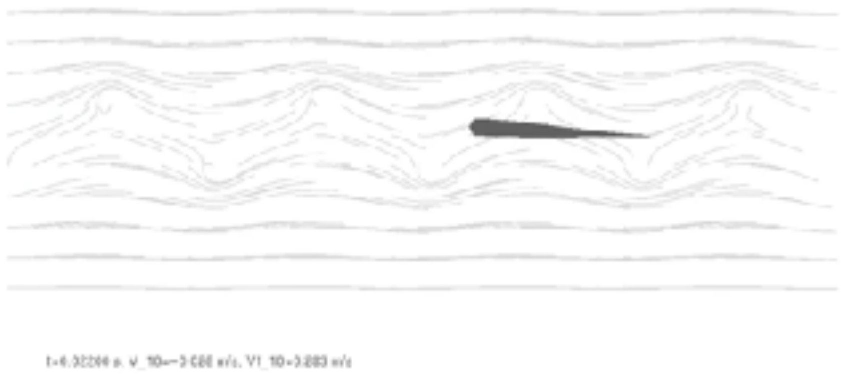
EXAMPLES OF COMPLIANT ROBOTS INSPIRED BY NATURE

- Fish
  - MIT

https://www.csail.mit.edu/research/sofi-soft-robotic-fish

EXAMPLES OF COMPLIANT ROBOTS INSPIRED BY NATURE

- Fish
  - Boyer et al, Mines Nantes, France



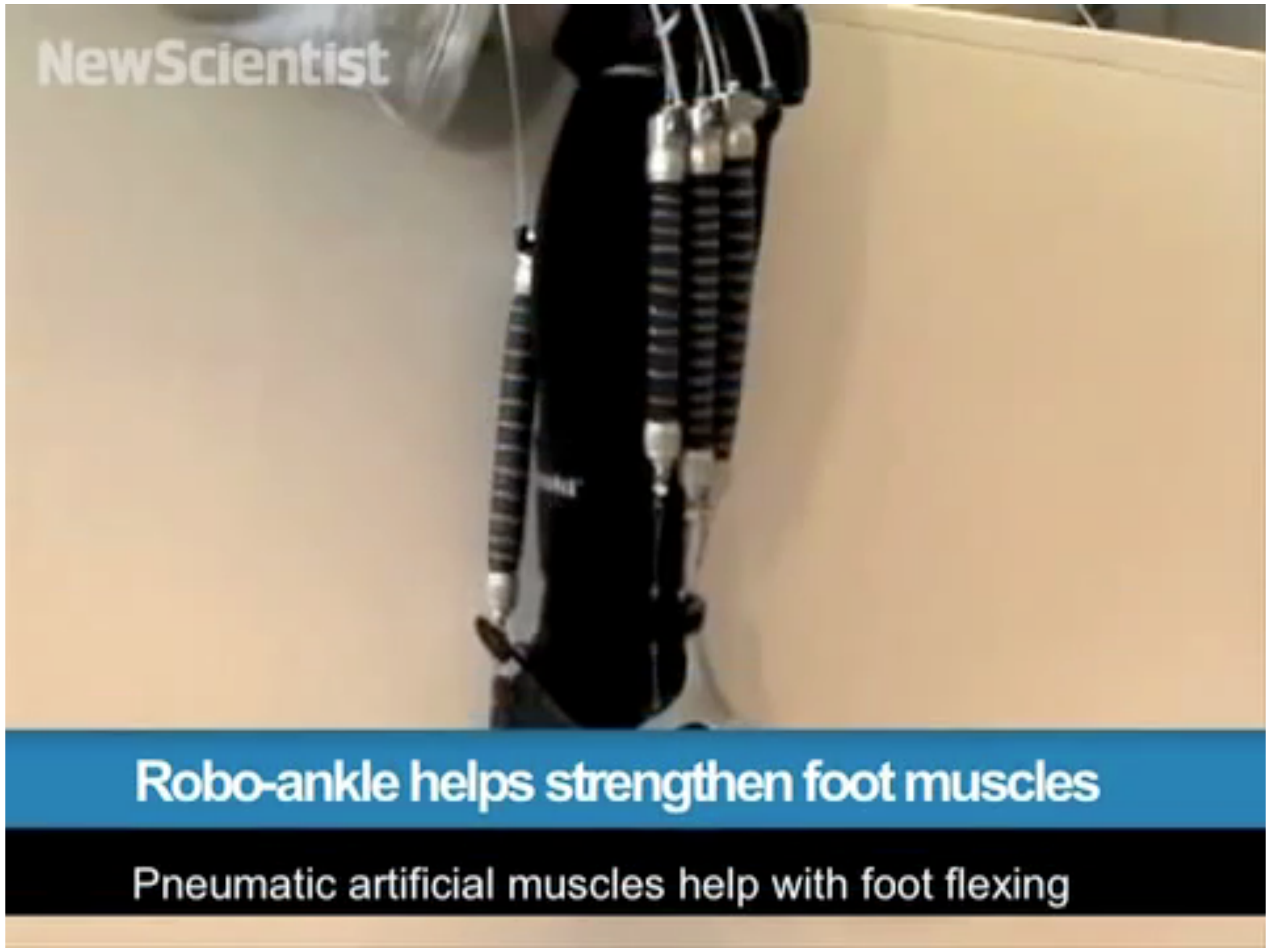
soft eel

SOFT ROBOT TECHNOLOGIES

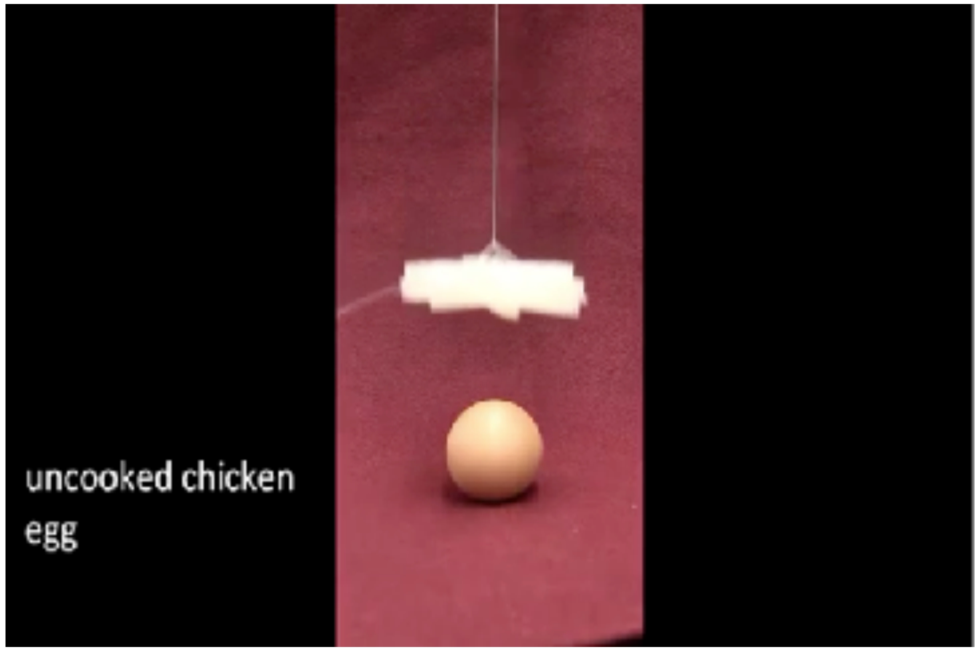
SOFT ACTUATORS

- Fluidic actuators

Hydraulic actuation



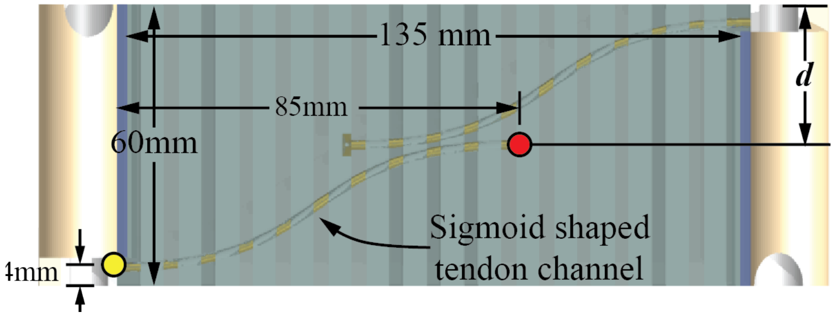
Mc Kibben muscles

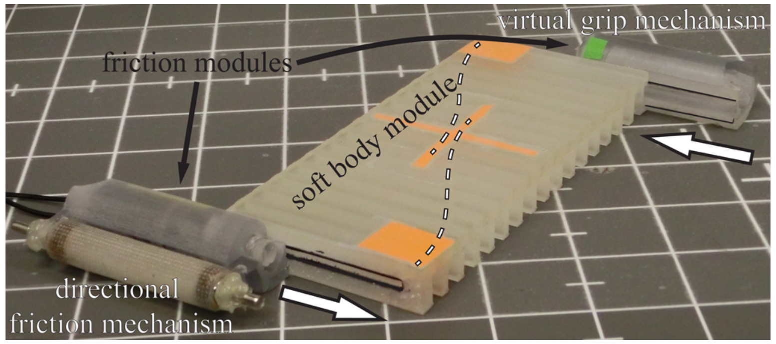


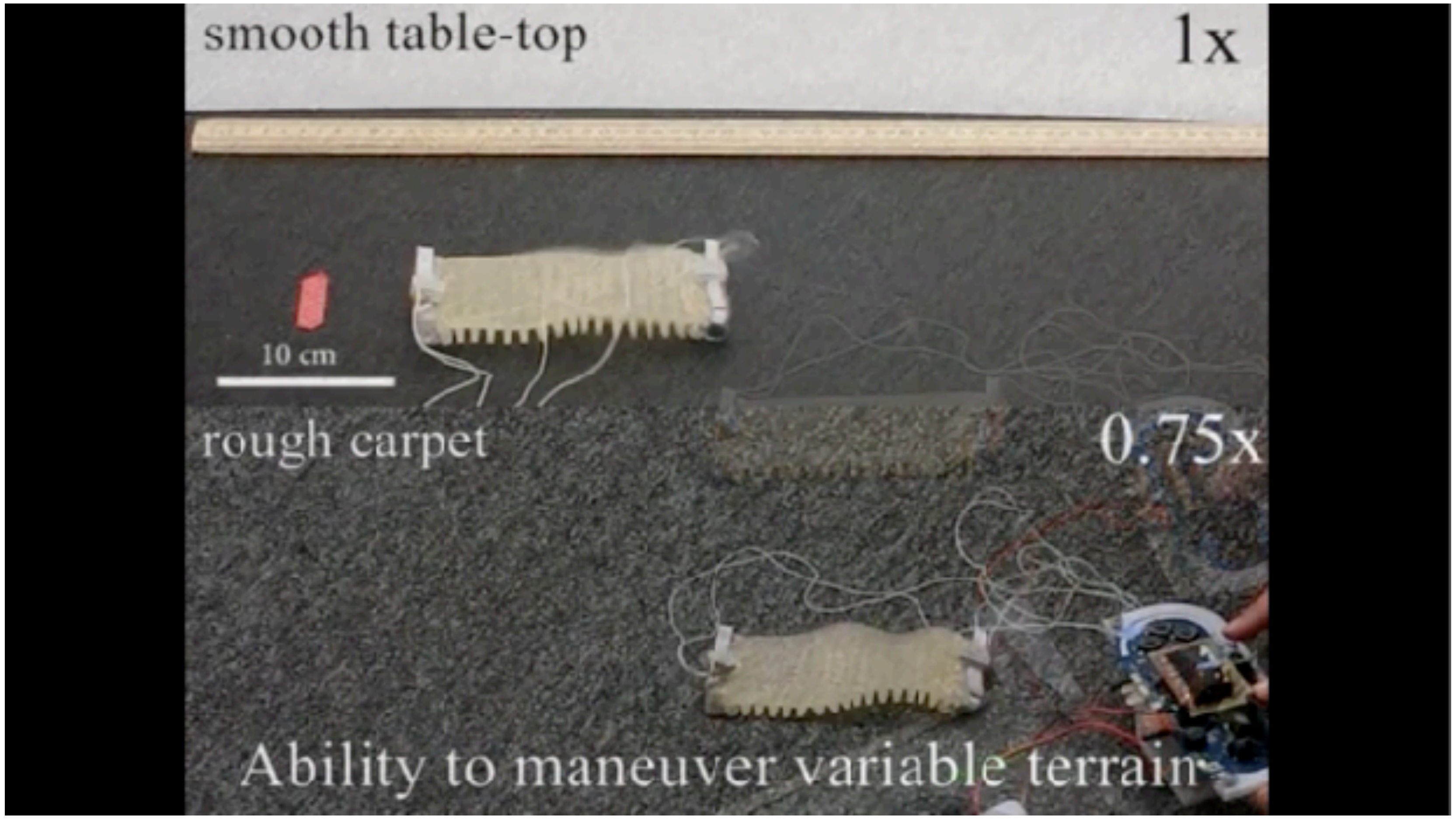
PneuNet

SOFT ACTUATORS

- Tendons



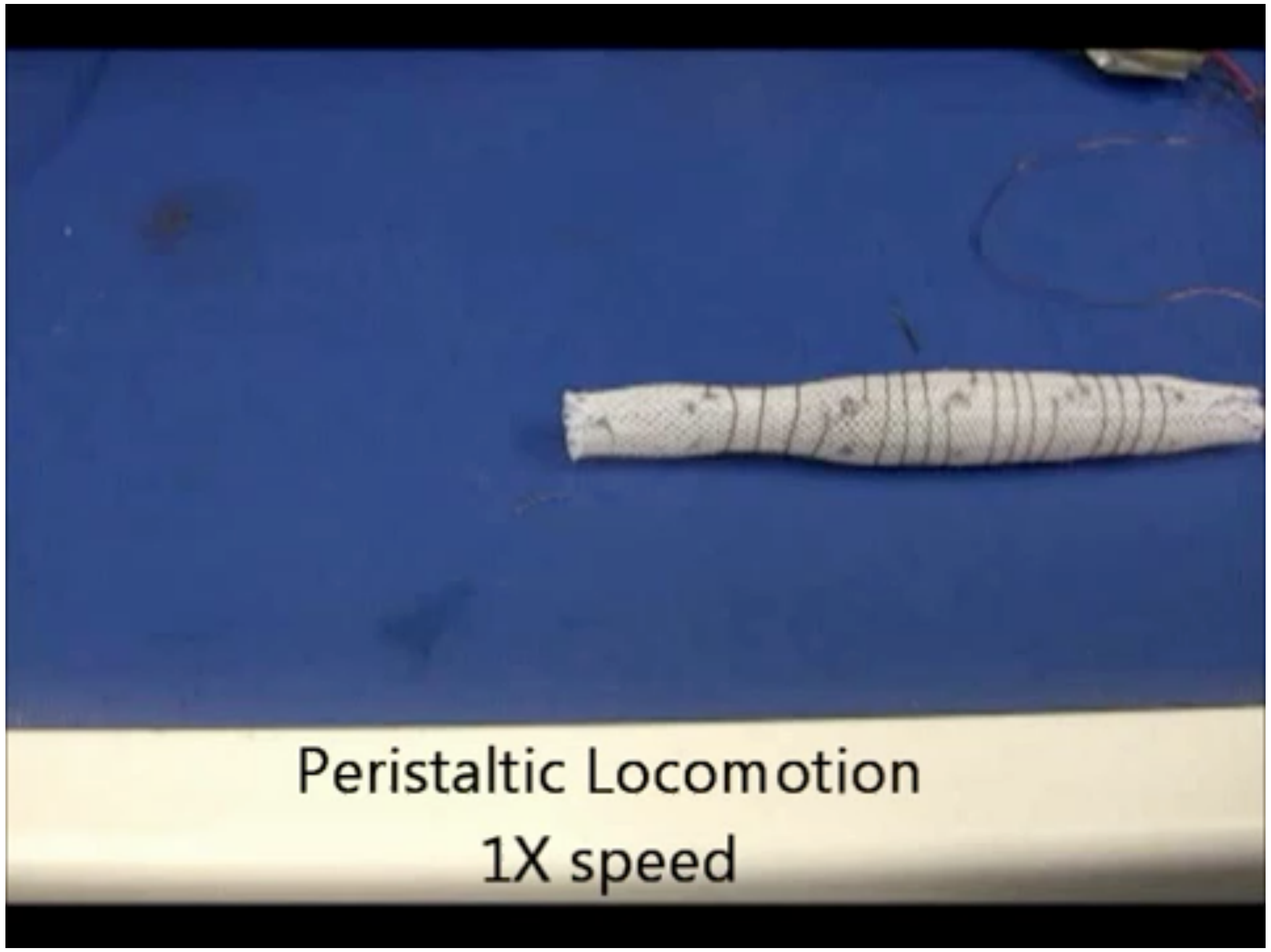


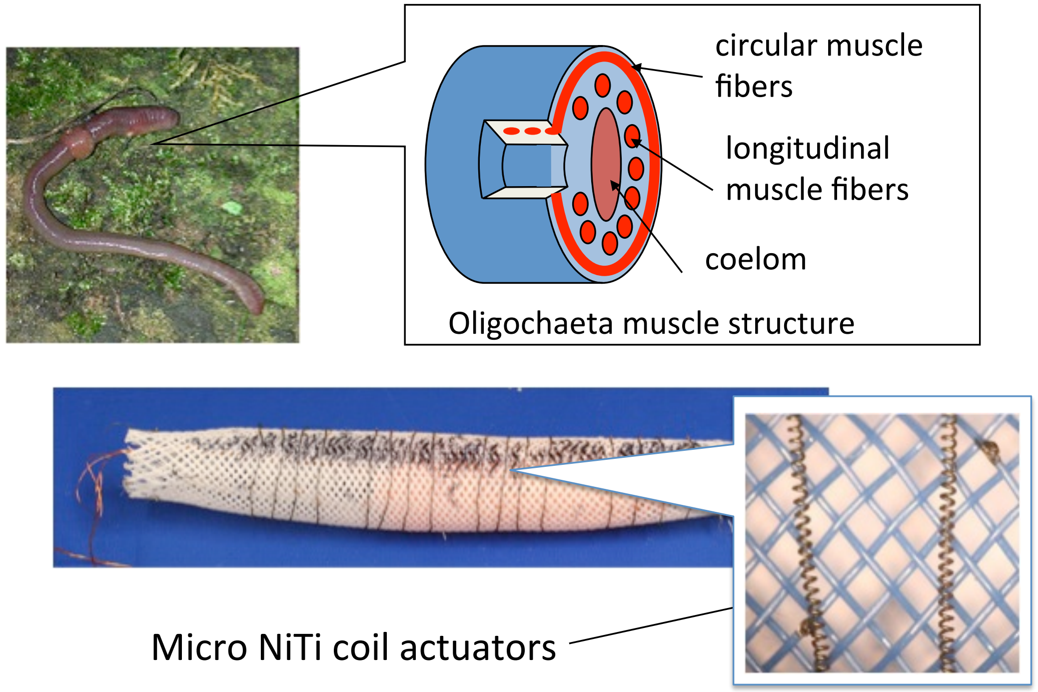


Design and locomotion control of soft robot using friction manipulation and motor-tendon actuation Vishesh Vikas, Eliad Cohen, Rob Grassi, Canberk So¨zer and Barry Trimmer

SOFT ACTUATORS

- Shape memory materials:



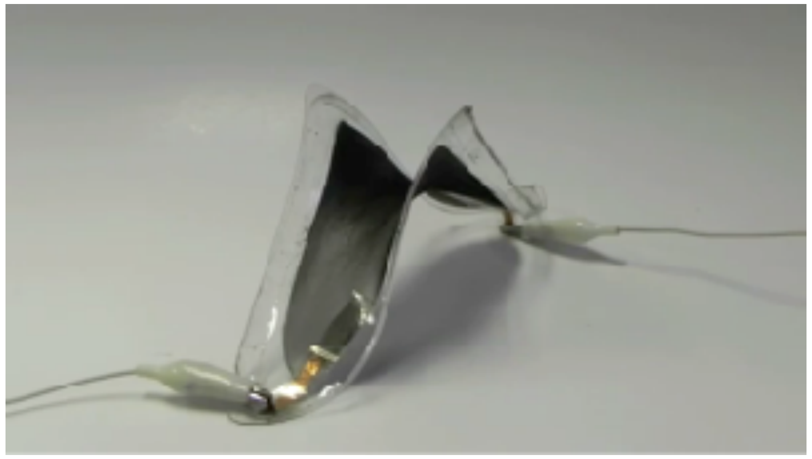


MIT & Harvard University

SOFT ACTUATORS

- Dielectric ElectroActive Polymers (EAP)
  - Polymers that exhibit a change in size or shape when stimulated by an electric field.
  - Can be used as actuator and sensor

SRI International.

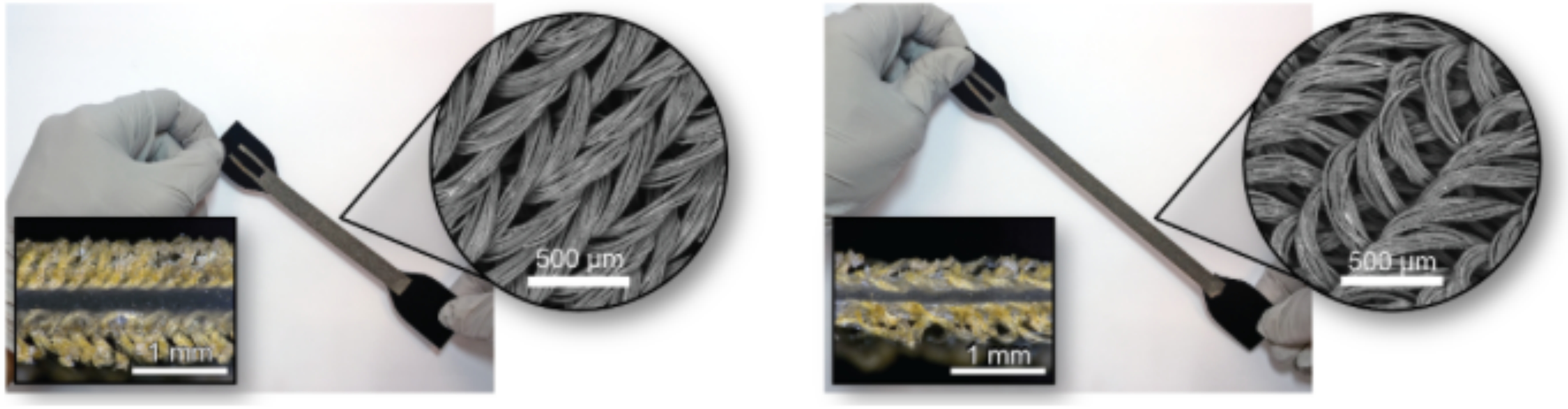


Univ Sydney

Eamex Corp.

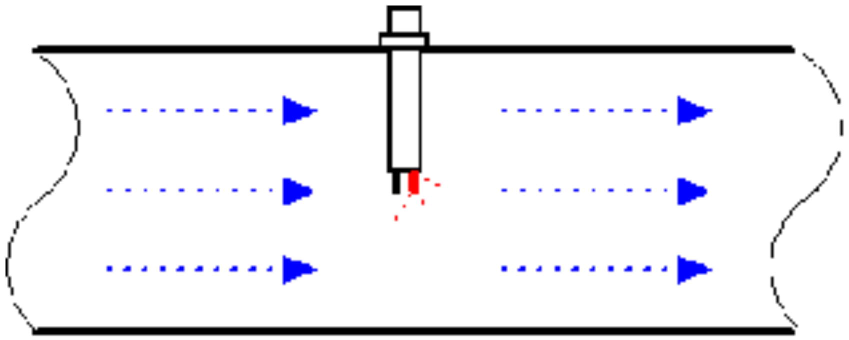
SENSORS

- Textile Silicone Hybrid Sensor
  - two outer electrode layers of highly stretchable silver plated knitted textile
  - a dielectric layer of silicone elastomer in between
  - Capacitance change when the sensor is stretched

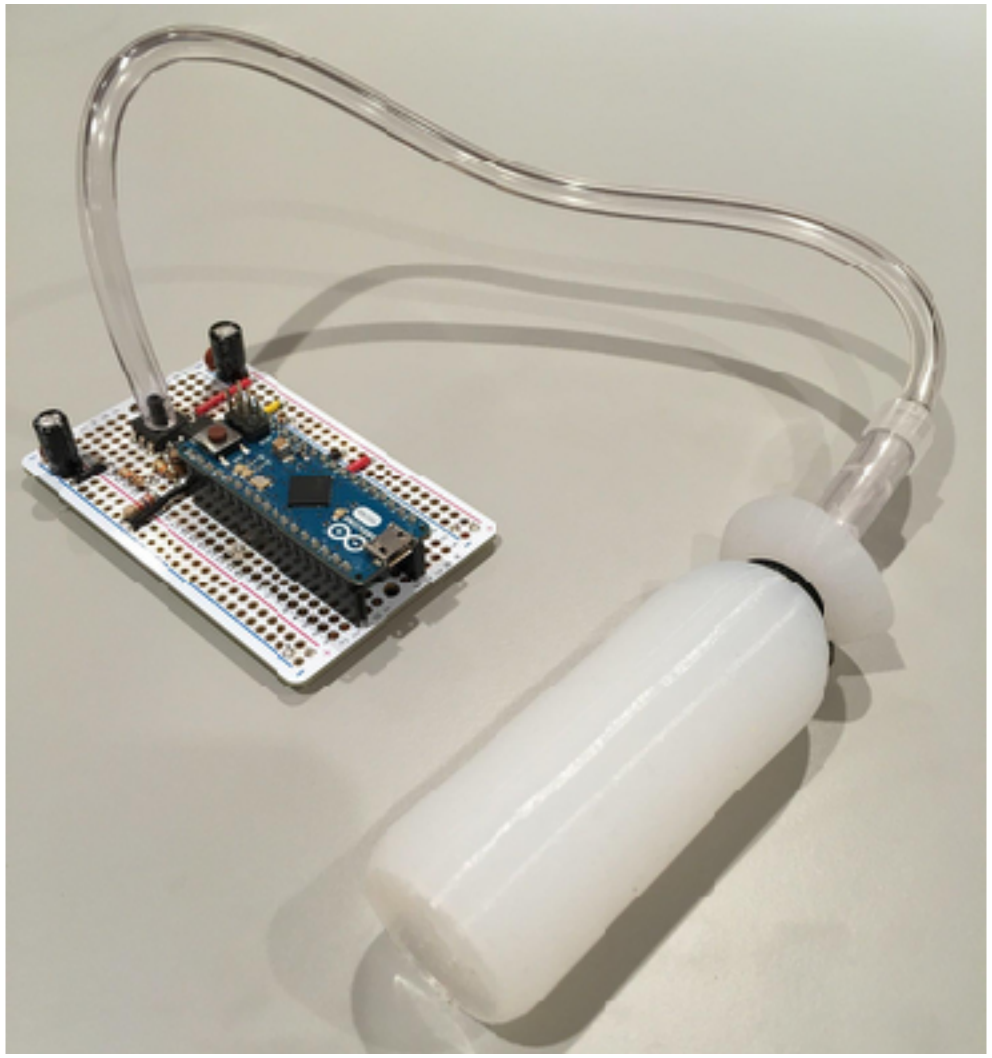


SENSORS

- Pneumatic sensors
  - Flow
  - Pressure



www.EngineeringToolBox.com



SENSORS

- Optical fibers & compliant structure

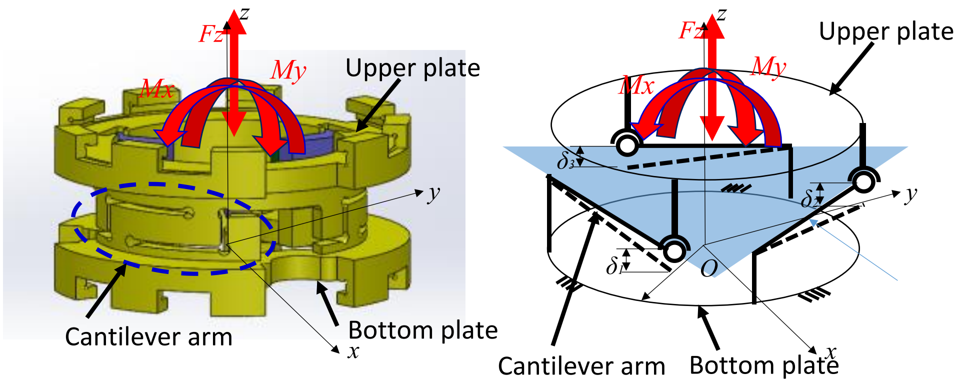
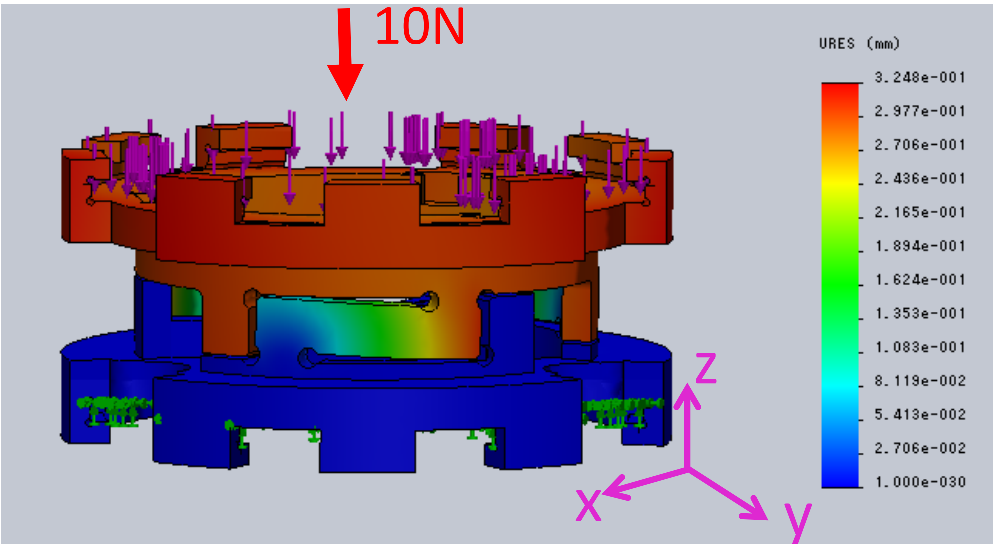
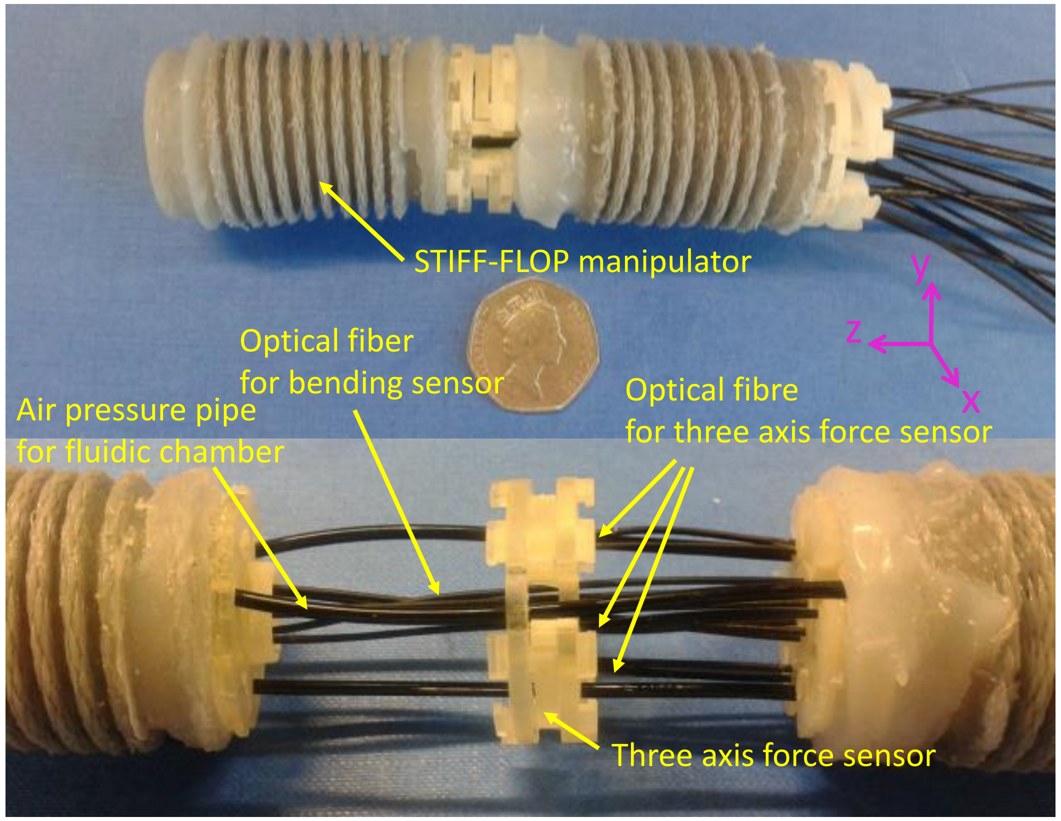
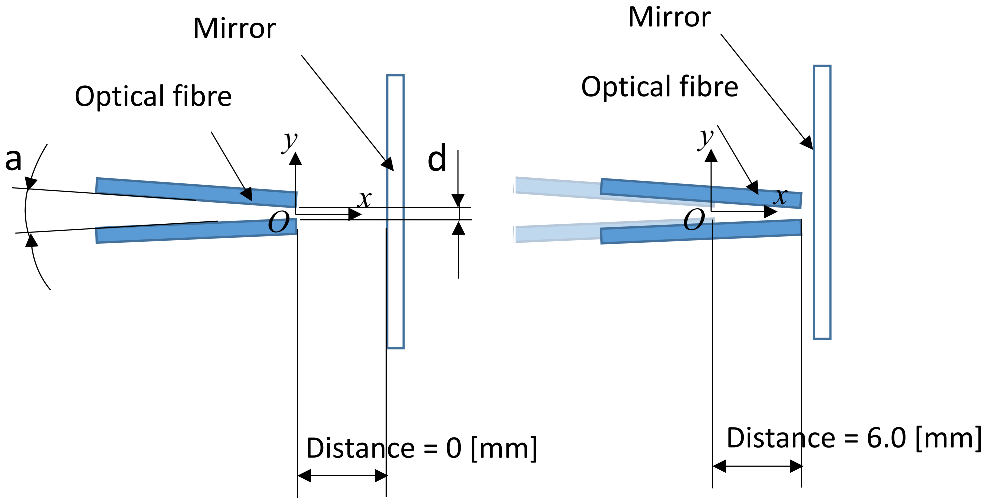


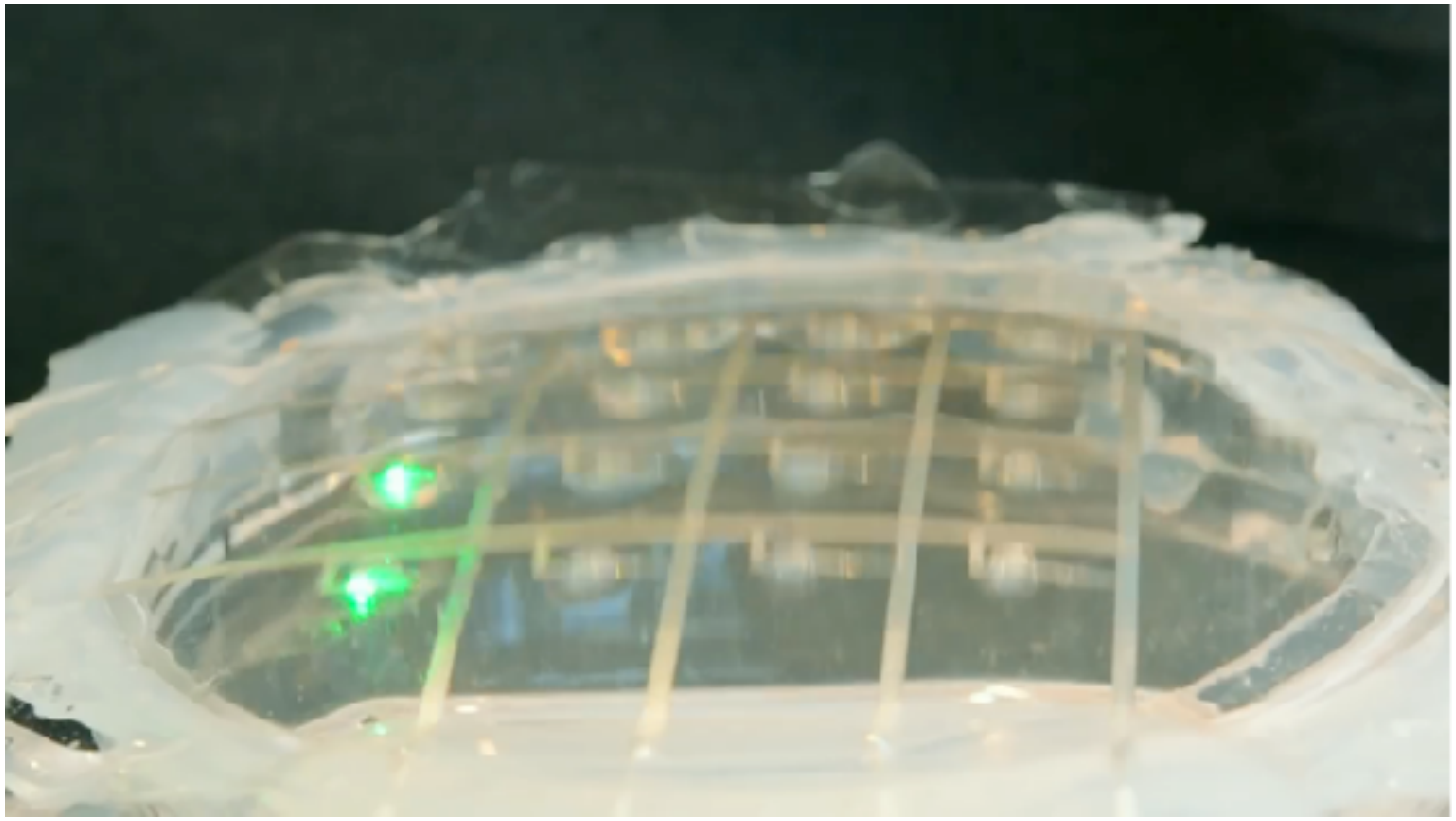
Figure 3. Equivalent spring model of the flexible tripod platform







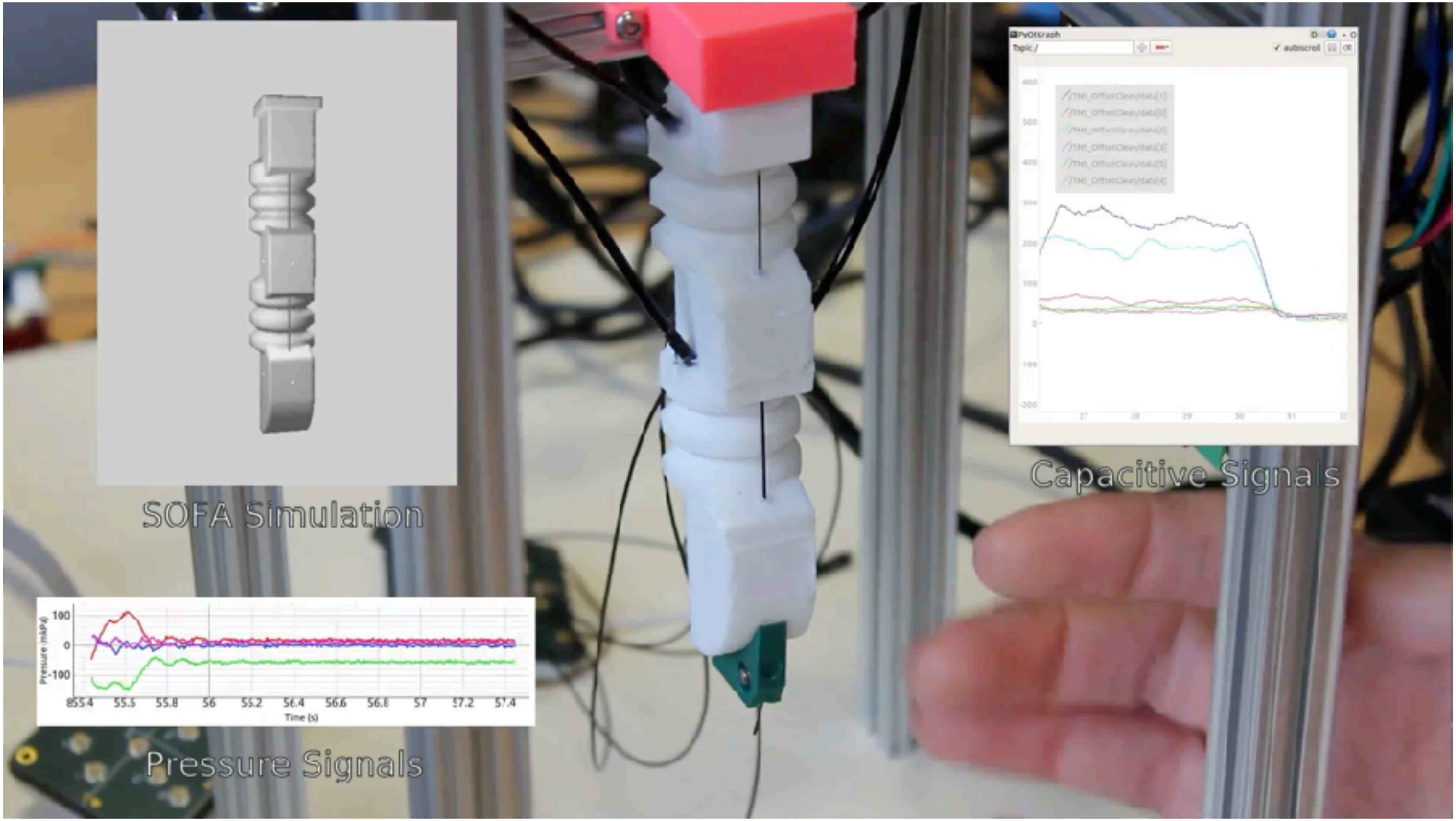
STRETCHABLE ELECTRONICS



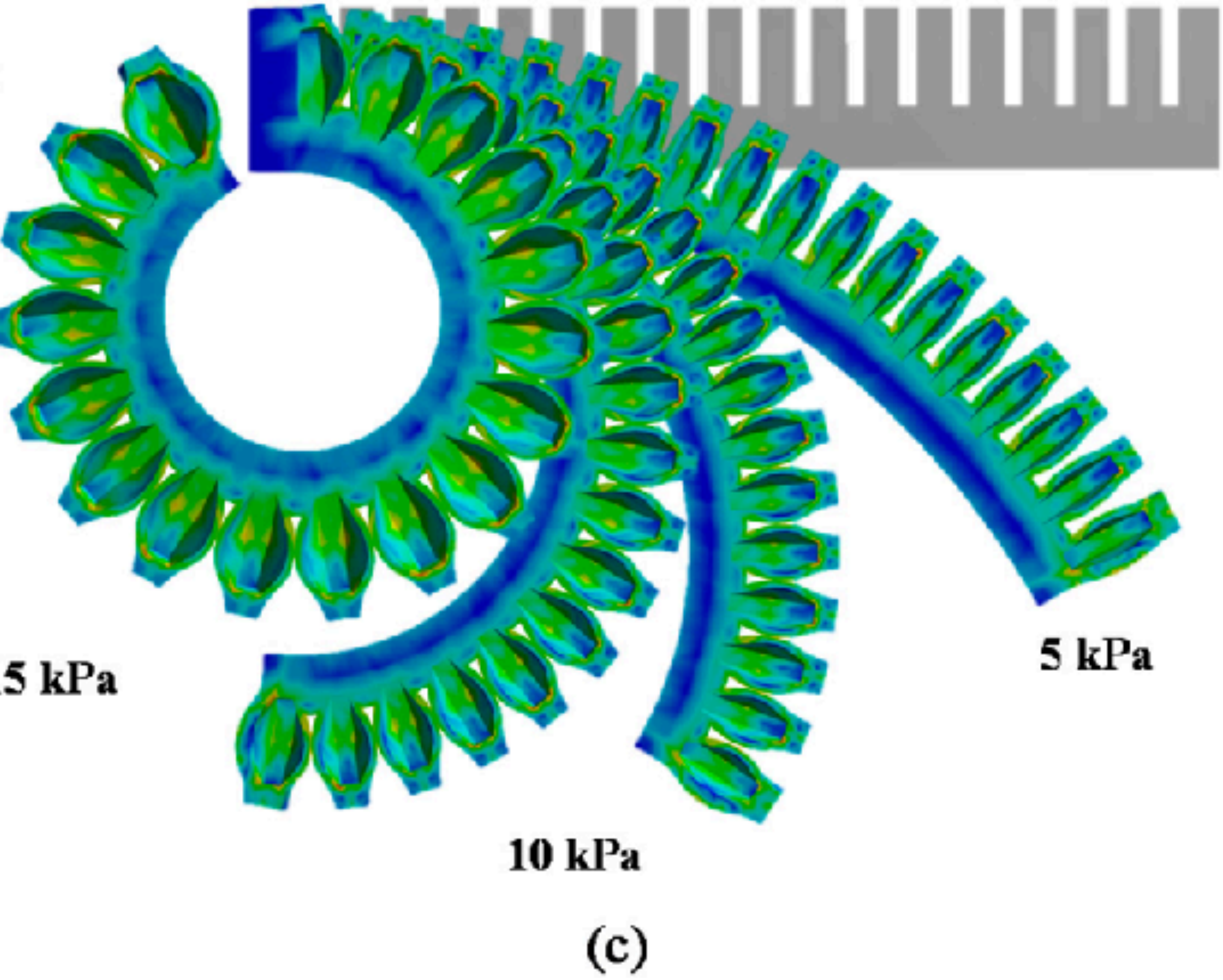
Hirsch, Arthur, Hadrien O. Michaud, Aaron P. Gerratt, Séverine De Mulatier, and Stéphanie P. Lacour. "Intrinsically stretchable biphasic (solid–liquid) thin metal films." *Advanced Materials* 28, no. 22 (2016): 4507-4512.

Paik, Jamie K., Rebecca K. Kramer, and Robert J. Wood. "Stretchable circuits and sensors for robotic origami." *Intelligent Robots and Systems (IROS), 2011 IEEE/RSJ International Conference on*. IEEE, 2011.

STRETCHABLE ELECTRONICS & SENSORS (CAPACITIVE AND PRESSURE)



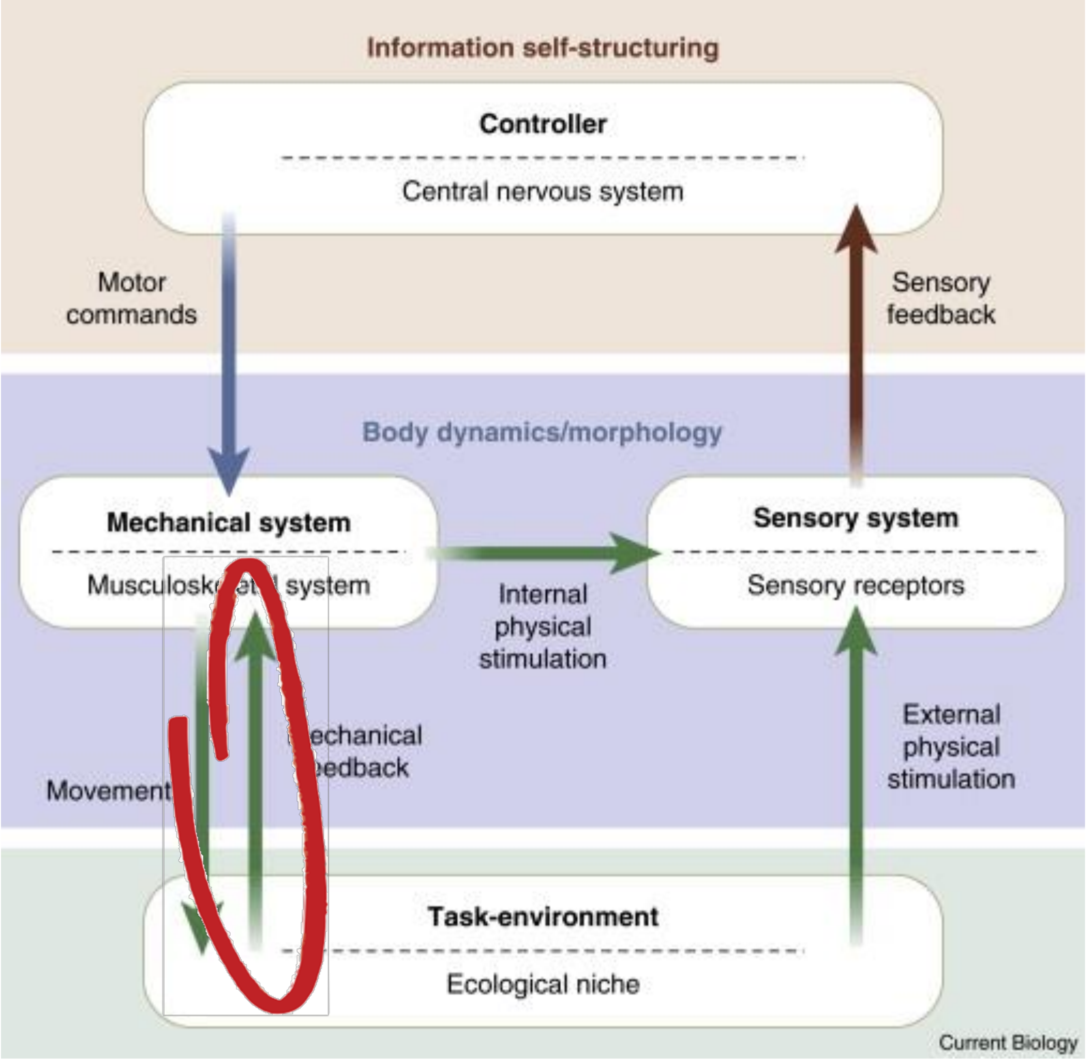
Escaida Navarro et al. accepted to RAL 2020



STRUCTURE AND OPTIMISATION

MORPHOLOGICAL COMPUTATION & EMBODIED INTELLIGENCE

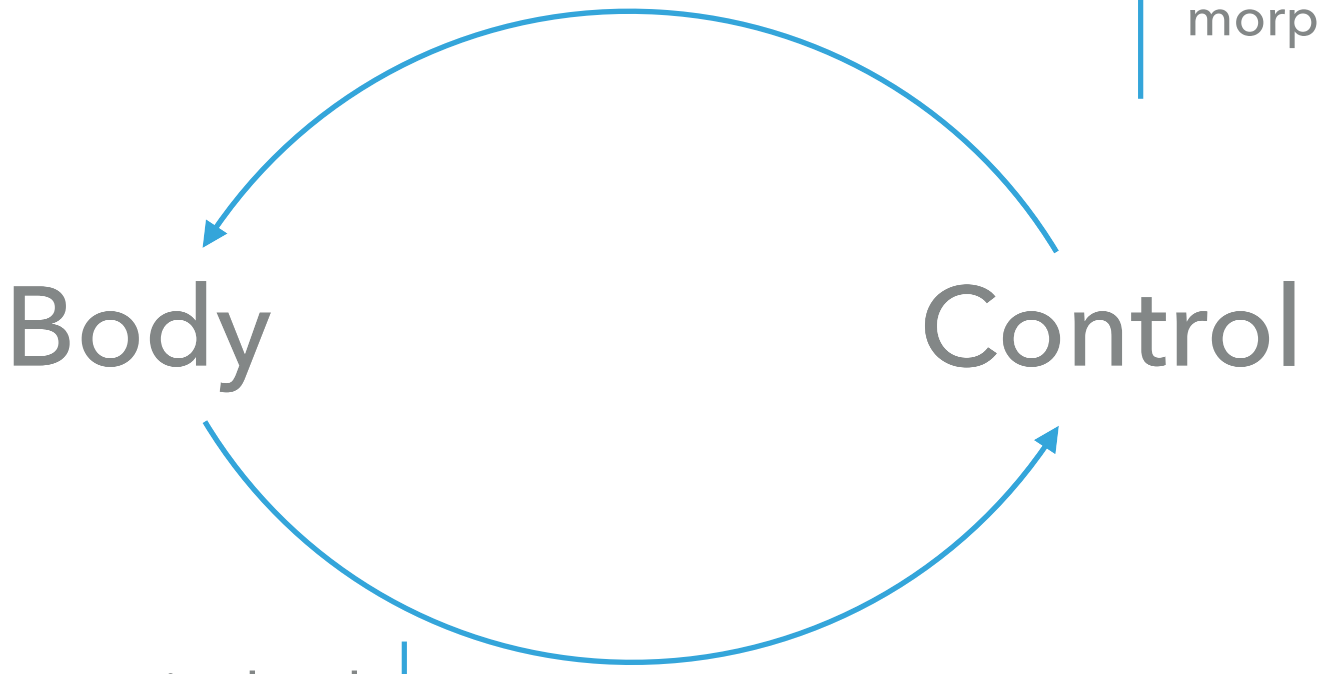
- A concept… not yet a theory…



Adaptive behaviour emerges from the interaction between the morphology of the **body**, the **environment** and the **task.**

MORPHOLOGICAL COMPUTATION & EMBODIED INTELLIGENCE

- A concept… not yet a theory…



Morphological computation

Part of the control is done (and delegated) naturally by the morphology of the body

Body

Control

The body contains local control elements or information processing elements

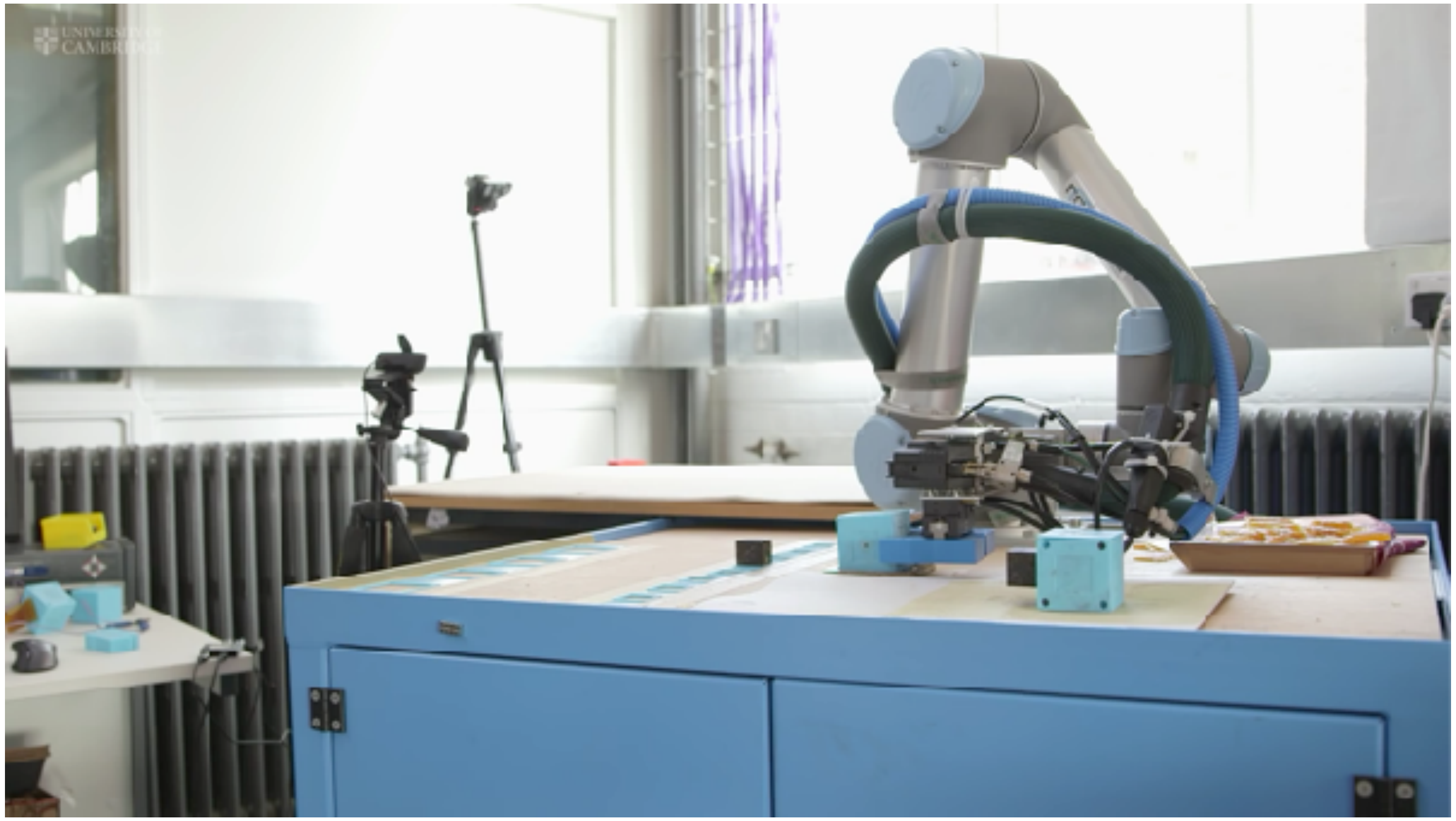
Embodied Intelligence

Pfeifer, Rolf, Fumiya Iida, and Gabriel Gómez. "Morphological computation for adaptive behavior and cognition." *International Congress Series*. Vol. 1291. Elsevier, 2006.

Pfeifer, Rolf, and Fumiya Iida. "Embodied artificial intelligence: Trends and challenges." *Embodied artificial intelligence*. Springer, Berlin, Heidelberg, 2004.

MORPHOLOGICAL COMPUTATION & EMBODIED INTELLIGENCE

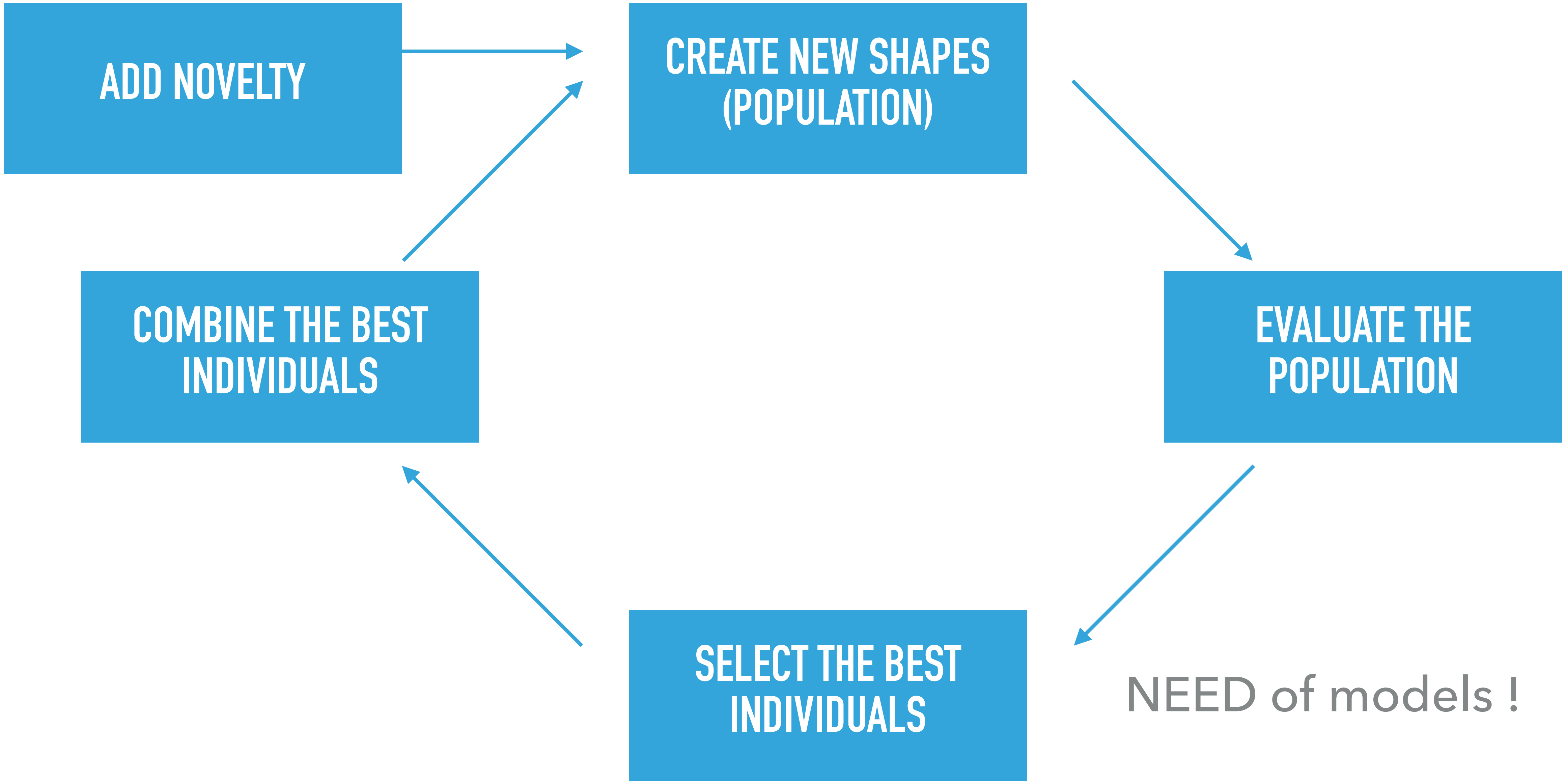
- Evolutionary algorithms



Cambridge University | Fumiya Iida's research group

MORPHOLOGICAL COMPUTATION & EMBODIED INTELLIGENCE

- Evolutionary algorithms



ADD NOVELTY

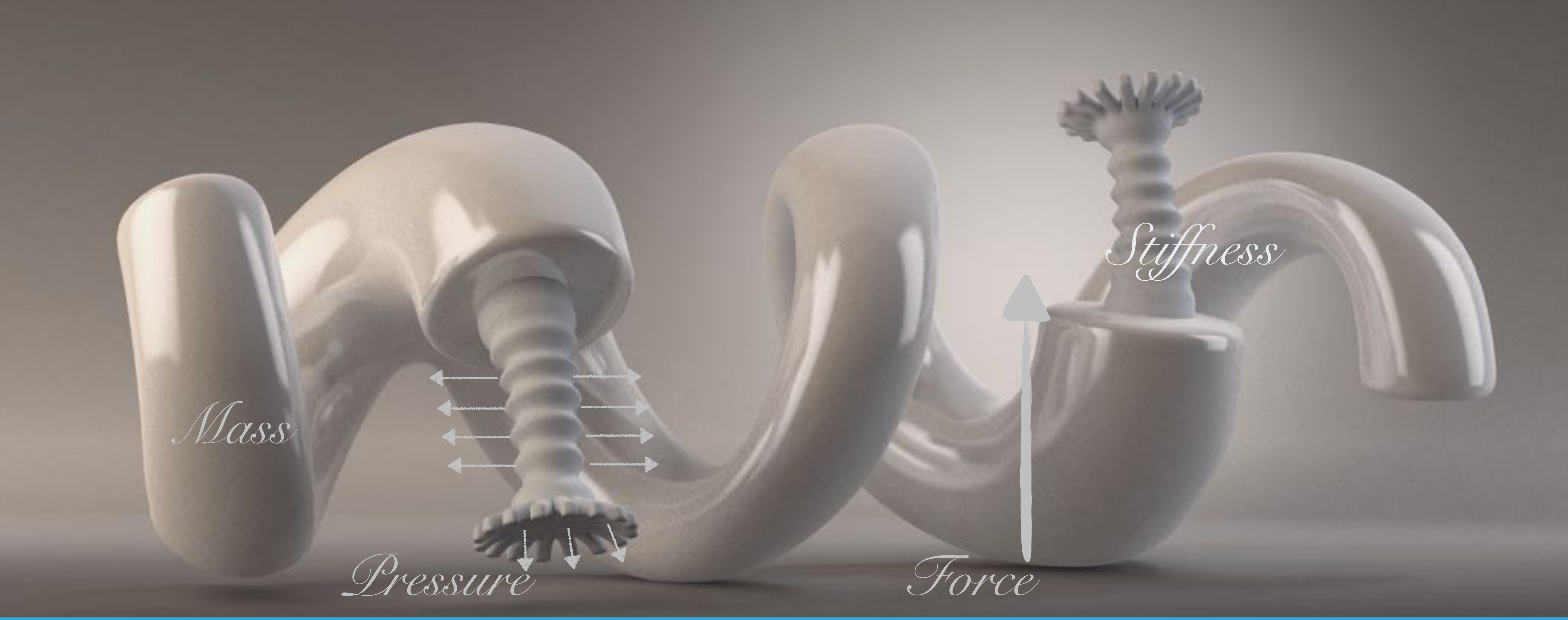
CREATE NEW SHAPES (POPULATION)

EVALUATE THE POPULATION

SELECT THE BEST INDIVIDUALS

COMBINE THE BEST INDIVIDUALS

NEED of models !



©Jonathan Pepe/DEFROST team

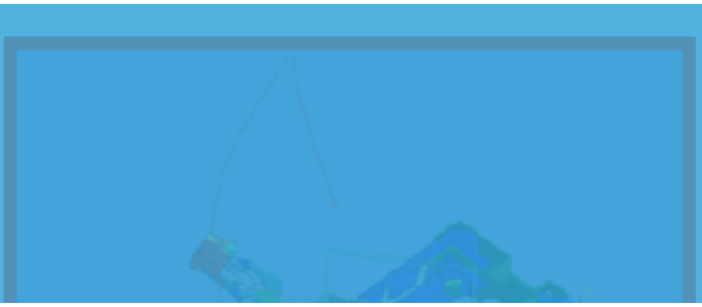
MODELING AND SIMULATION

- Modeling for real-time simulation
- Constraint-based modeling

PHYSICS-BASED SIMULATION FOR ROBOTICS SYSTEMS

- Many use-cases !

Design stage



safe verification environment

Learning s

planning

rol strategies

Making de

Simulator

tic-system

NVIDIA Isaac Platform

Da Vinci Training Simulator - Intuitive

PHYSICS-BASED SIMULATION…

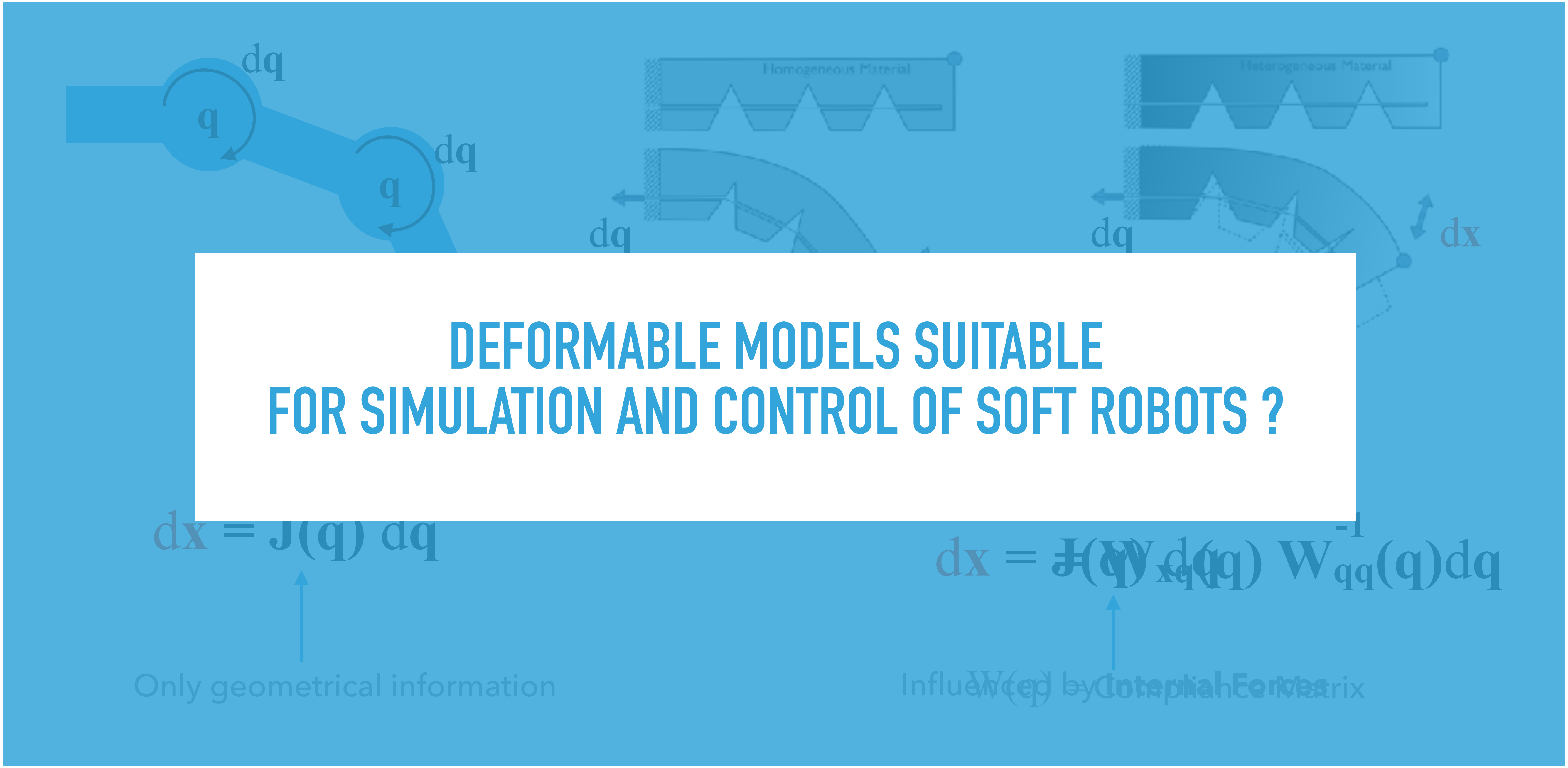
TOO COMPLICATED FOR SOFT-ROBOTICS ?

OR

ISN'T IT EVEN MORE IMPORTANT FOR SOFT-ROBOTICS ?

PHYSICS-BASED SIMULATION FOR SOFT ROBOTICS SYSTEMS

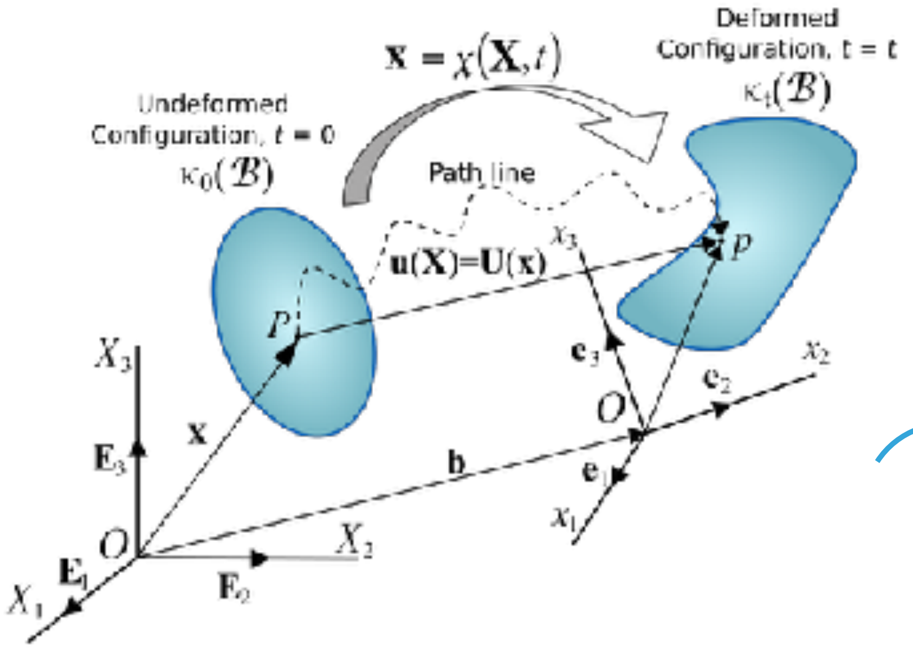
- Rigid Robot Kinematics derived from geometry
- Soft Robot Kinematics derived from mechanics !



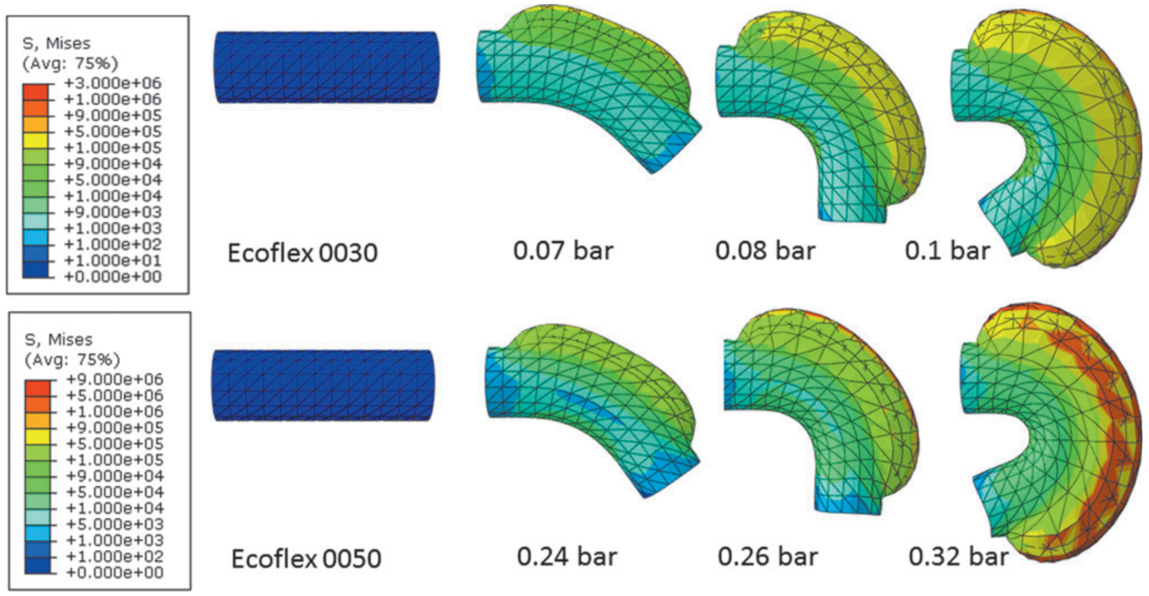
DEFORMABLE MODELS SUITABLE FOR SIMULATION AND CONTROL OF SOFT ROBOTS ?

DEFORMABLE MECHANICAL MODELS FOR ROBOTICS SYSTEMS

- soft robots are deformable solids: why not using continuum mechanics ?



Finite Strain Theory in continuum mechanics



Finite Element Analysis and Design Optimization of a Pneumatically Actuating Silicone Modulefor Robotic Surgery Applications

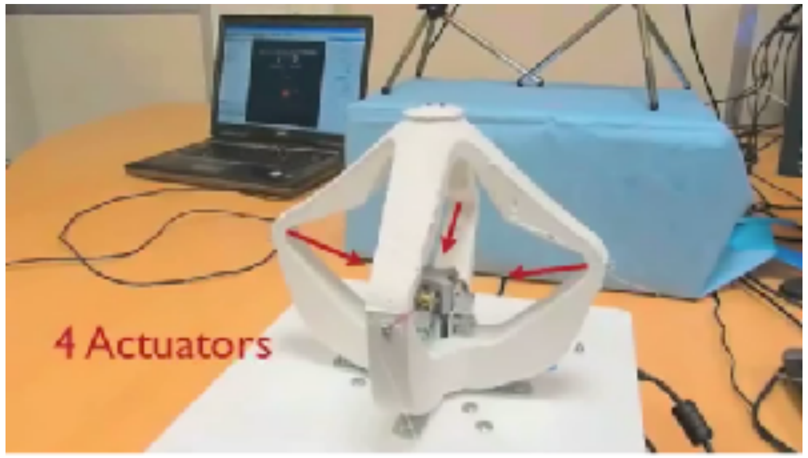
Elsayed et al. Soro 2014

- ++ « Classical » mechanics
- ++ material properties
- **– – no analytical solutions**

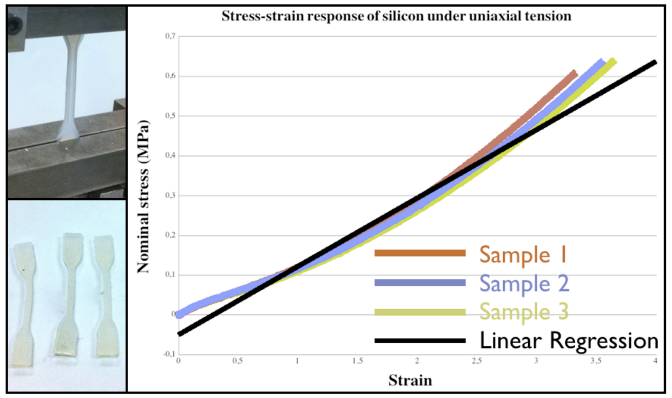
- ++ Well known in numerical mechanics
- ++ Existing software (don't need to know the continuum mechanics to use it)
- ++ take into account the geometry and material properties
- **– – computation time**

MECHANICAL DEFORMABLE MODELS

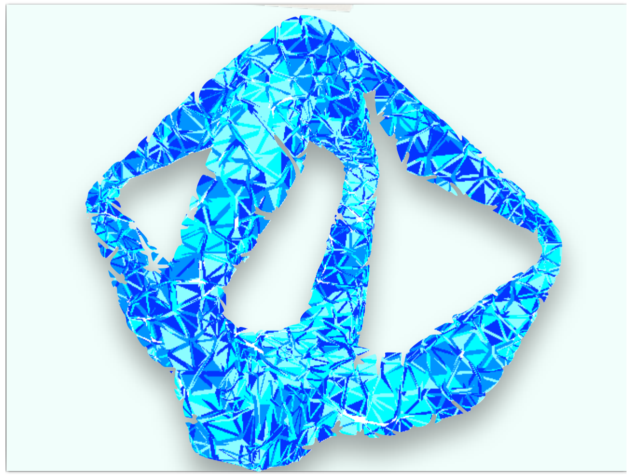
- FEM model



Deformable robot



Constitutive law



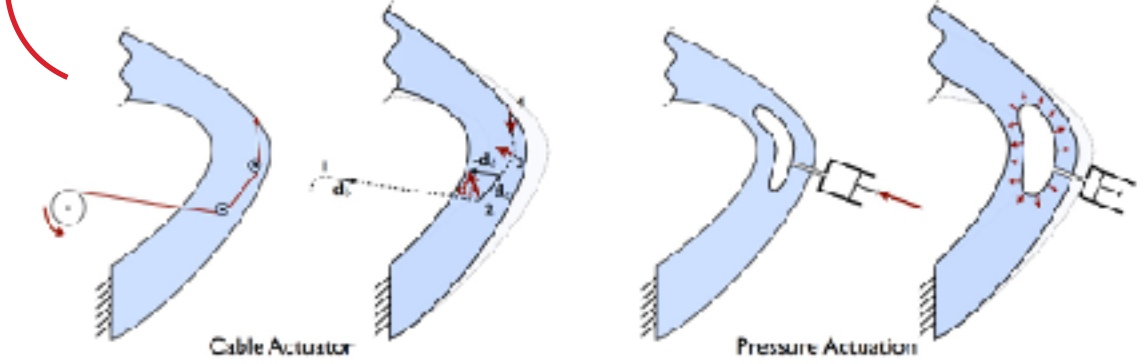
FEM mesh

- Newton's second law

$$\mathbb{M}(\mathbf{q})\dot{\mathbf{v}} = \mathbb{P}(t) - \mathbb{F}\,(\mathbf{q},\mathbf{v}) + \boxed{\mathbf{H}^T\lambda}$$

$\mathbf{q} \in \mathbb{R}^n$	Vector of generalized degrees of freedom (nodes of a deformable model)
$\mathbf{v} \in \mathbb{R}^n$	Vector of velocities
$\mathbb{M}(\mathbf{q}) : \mathbb{R}^n \mapsto \mathcal{M}^{n\times n}$	Inertia Matrix
$\mathbb{F}\,(\mathbf{q},\mathbf{v})$	Internal forces (non-linear model)
$\mathbb{P}(t)$	External forces
$\mathbf{H}^T\lambda \in \mathbb{R}^n$	Constraint force contribution

unknown forces: actuation & collision



TIME INTEGRATION SCHEMES WITH COLLISION EVENTS

acceleration is not defined !

V- <0 before impact and V+ >0 after impact.. between them, an infinite small time step

TIME-STEPPING METHOD

- Low order Integration scheme:

$$\begin{aligned}\mathbf{M}(\mathbf{v}_f - \mathbf{v}_i) &= h\left(\mathbb{P}(t_f) - \mathbb{F}(\mathbf{q}_f, \mathbf{v}_f)\right) + \ h\mathbf{H}^T\lambda_f \\ \mathbf{q}_f &= \mathbf{q}_i + h\mathbf{v}_f\end{aligned}$$

- 1 Linearization per step:

$$\mathbb{F}\left(\mathbf{q}_i + d\mathbf{q}, \mathbf{v}_i + d\mathbf{v}\right) = \mathbf{f}_i + \frac{\delta\mathbb{F}}{\delta\mathbf{q}}d\mathbf{q} + \frac{\delta\mathbb{F}}{\delta\mathbf{v}}d\mathbf{v}$$

No assumption on the type of deformable model (compatible with hyperelastic models)

- Matrix system to be solved:

$$\underbrace{\left(\mathbf{M} + h\frac{\delta\mathbb{F}}{\delta\mathbf{v}} + h^2\frac{\delta\mathbb{F}}{\delta\mathbf{q}}\right)}_{\boxed{\mathbf{A}}} \underbrace{d\mathbf{v}}_{\mathbf{x}} = \underbrace{-h^2\frac{\delta\mathbb{F}}{\delta\mathbf{q}}\mathbf{v}_i - h\left(\mathbf{f}_i + \mathbf{p}_f\right)}_{\boxed{\mathbf{b}}} + h\mathbf{H}^T\lambda$$

or (quasi static case)

$$\underbrace{\frac{\delta\mathbb{F}}{\delta\mathbf{q}}}_{\boxed{\mathbf{A}}} \underbrace{d\mathbf{q}}_{\mathbf{dx}} = \underbrace{\mathbb{P} - \mathbf{f}_i}_{\boxed{\mathbf{b}}} + \mathbf{H}^T\lambda$$

- **Compatible with rigid body dynamics:**

$$\mathbf{M}(\mathbf{v}_f - \mathbf{v}_i) = h\left(\mathbb{P}(t_f) - \mathbb{F}(\mathbf{q}_f, \mathbf{v}_f)\right) + \ h\mathbf{H}^T\lambda_f$$

Mass and Inertia ↗ Gravity forces ↑ Coriolis & centrifugal forces ↑

[Stewart & Trinkle (1996)]

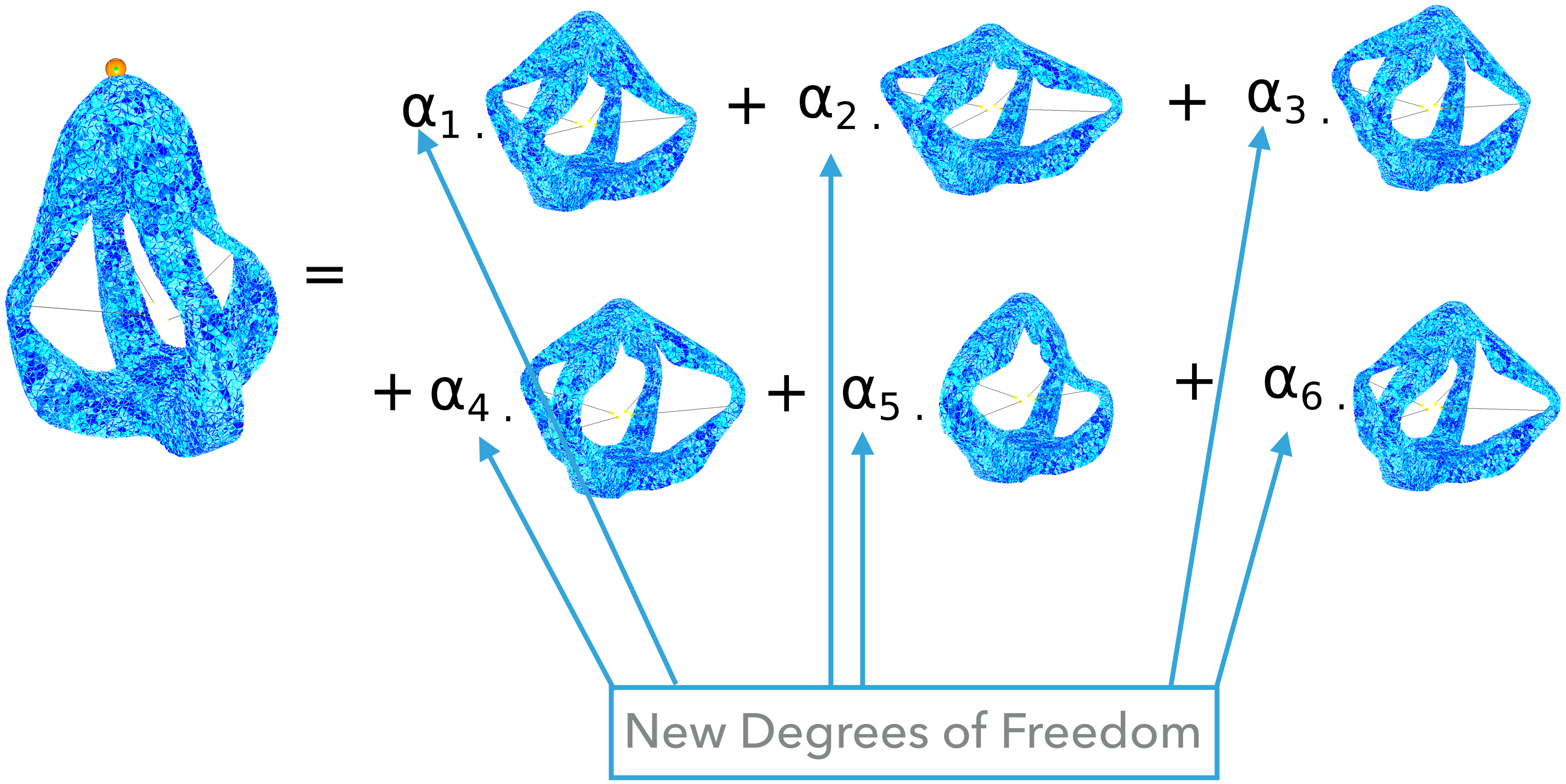
HOW TO MAKE IT REAL-TIME ?

- Use of numerical recipes
  - Optimisation of the structure of the matrix (see D.James *et al.*,…)
  - Preconditioners (see H.Courtecuisse *et al.*)
  - Domain decomposition (see Barbic *et al.*, Kry *et al.*, …)
  - …
- Code & formulation optimisation
  - GPU implementation (see J.Allard *et al., …)*
  - Multigrid or adaptive methods (Georgii *et al.*, Debune *et al.,…)*
  - Fast hyperelastic models (see S.Marchesseaux, H.Delingette *et al.*)
  - …
- Precomputation
  - Condensation methods (Cotin *et al.*,…)
  - Reduced-order methods (Barbic *et al.*, Goury *et al.*…)
  - …

Different communities: computer animation, biomechanics, numerical methods,…

MODEL ORDER REDUCTION

$$\mathbf{q}\,(t,\boldsymbol{\lambda}(t)) \approx \mathbf{q}(0) + \sum_{i=1}^{N} \phi_i \alpha_i(t,\boldsymbol{\lambda}(t)) = \mathbf{q}(0) + \boldsymbol{\Phi}\,\boldsymbol{\alpha}(t,\boldsymbol{\lambda}(t)) \quad (4)$$



New Degrees of Freedom

Goury & Duriez IEEE Trans on Robotics, 2018

MODEL ORDER REDUCTION

- This leads to the reduced equations

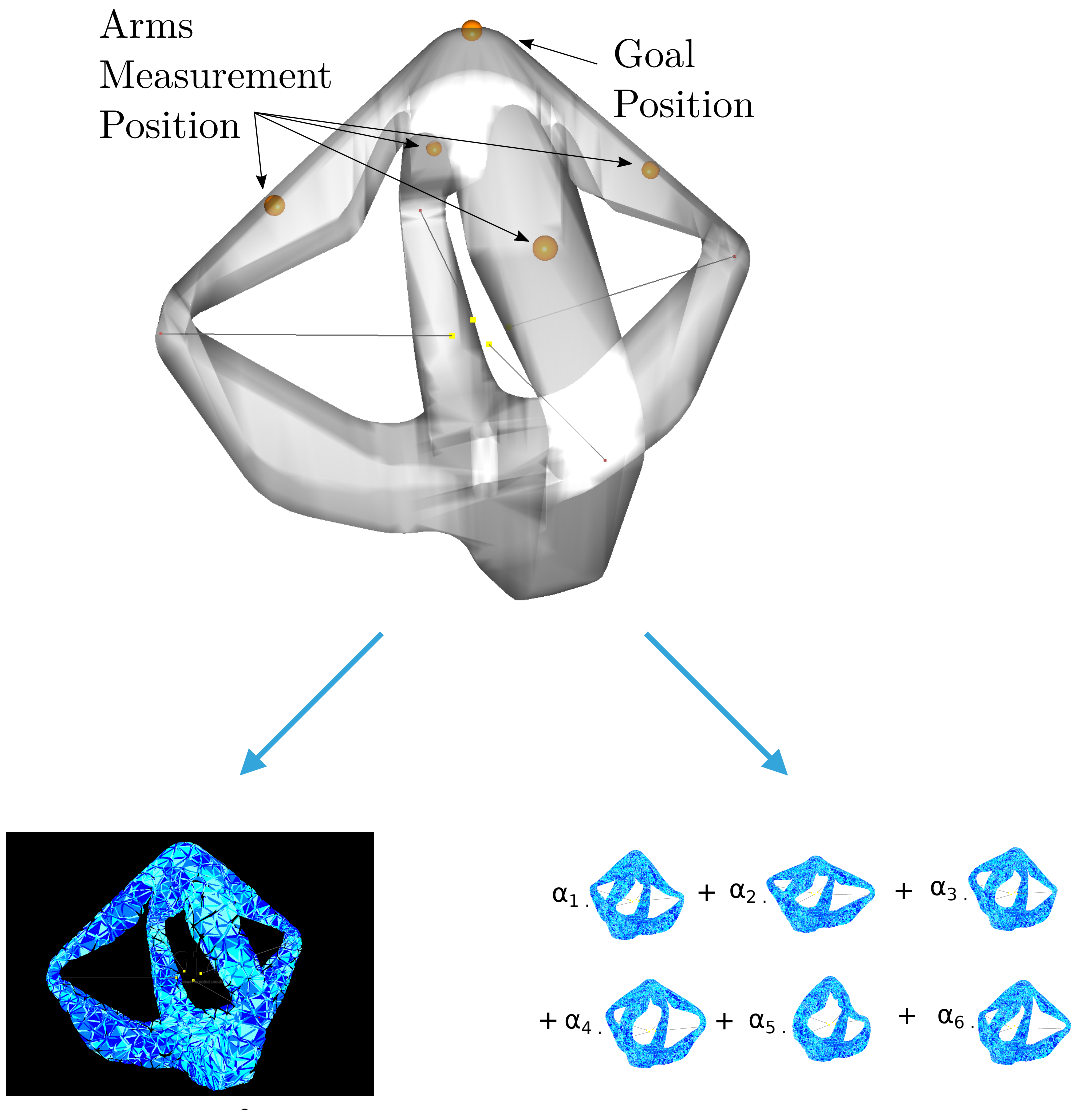
$$\underbrace{\boldsymbol{\Phi}^T \mathbf{A}(\mathbf{q}_t, \mathbf{v}_t)\boldsymbol{\Phi}}_{\mathbf{A}_r} \; d\dot{\alpha}(t+1) = \underbrace{\boldsymbol{\Phi}^T \mathbf{b}(\mathbf{q}_t, \mathbf{v}_t)}_{\mathbf{b}_r} + \underbrace{(\mathbf{H}\,\boldsymbol{\Phi})^T}_{\mathbf{H}_r} \boldsymbol{\lambda},$$

- Offline stage:
  - shake the robot within the range of its actuators => compute a snapshot space $\mathbf{S}$
  - Find basis that minimize the cost function $\widehat{J}(\boldsymbol{\Phi})^2 = \sum_{\boldsymbol{\lambda}^* \in \widehat{\boldsymbol{\Lambda}}} \sum_{t=t_0}^{t=t_{n_t}} \left\| \mathbf{q}(t, \boldsymbol{\lambda}^*(t)) - \sum \left(\phi_i{}^T \mathbf{q}(t, \boldsymbol{\lambda}^*(t)) \right) \phi_i \right\|_2^2 .$
  - After SVD $\mathbf{S} = \mathbf{U}\boldsymbol{\Sigma}\mathbf{V}^T$ (and truncation), we obtain the reduced basis:
- Hyper-reduction
  - additional stage for reducing computations with no loss of information

Goury & Duriez IEEE Trans on Robotics, 2018

MODEL ORDER REDUCTION

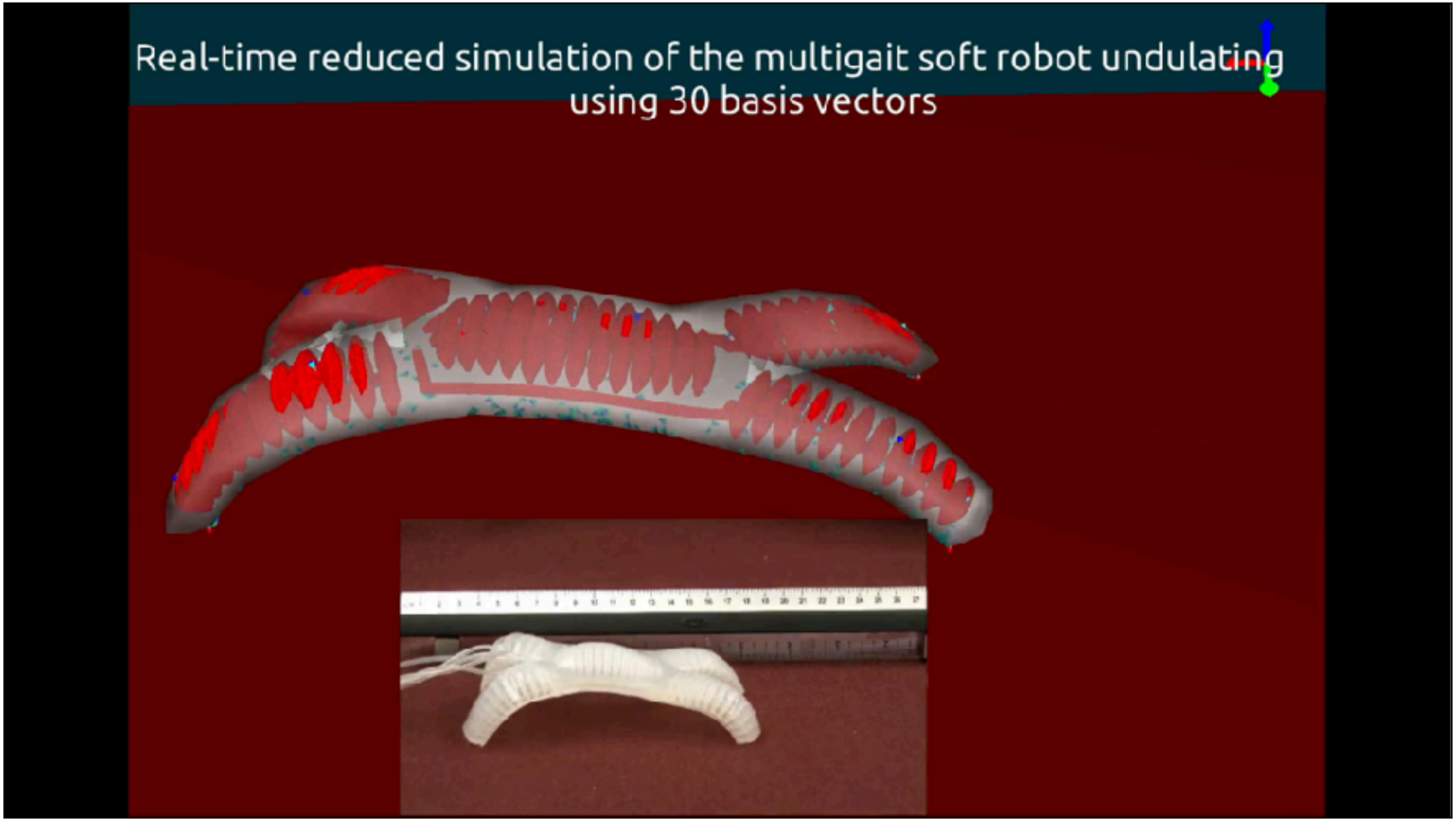
- Performance



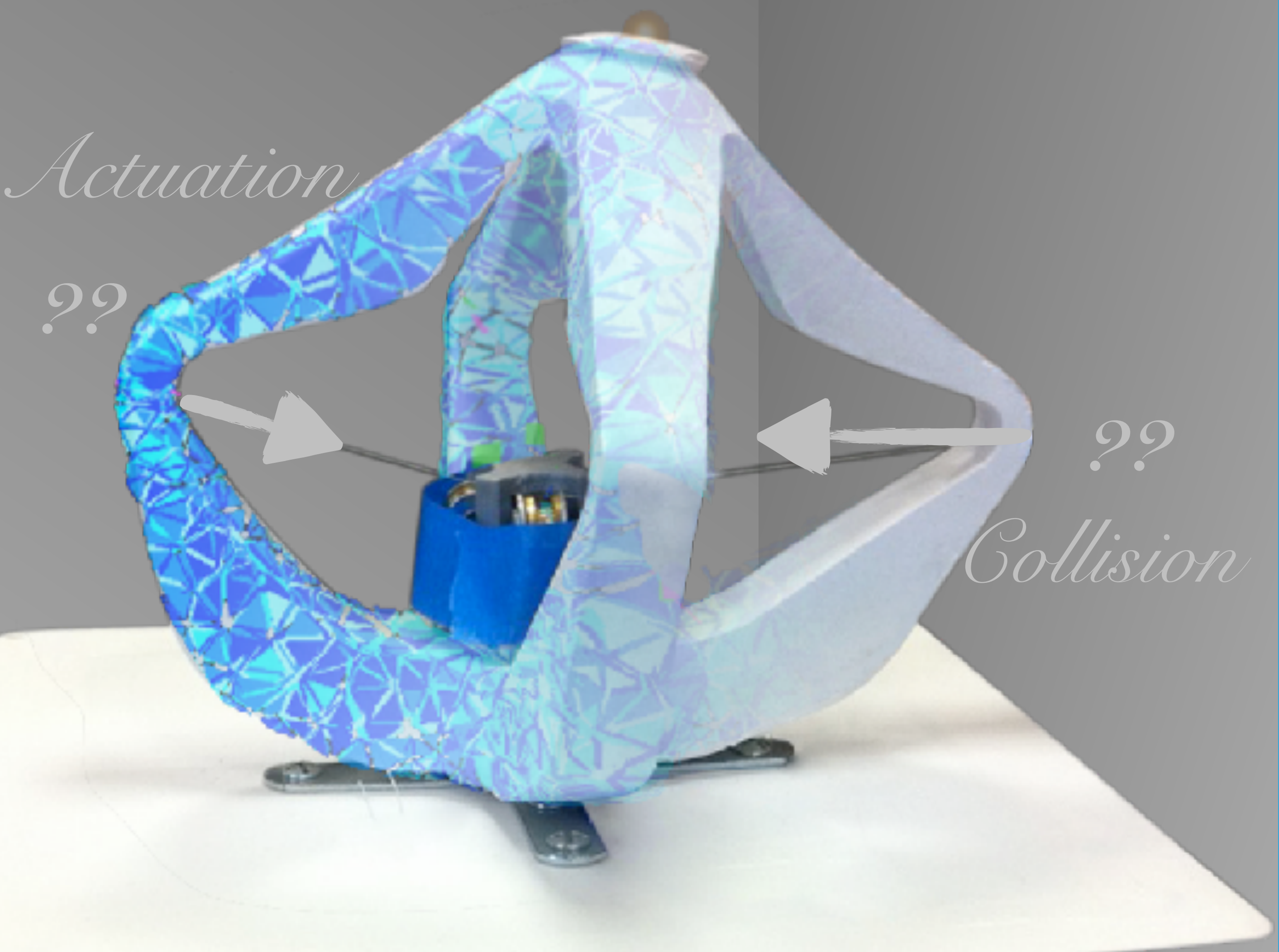
Performance	Coarse mesh	Model order reduction
Computation time	0.03s	0.02s
relative error for goal position	0.35	0.08
relative error for arm position	0.6	0.1

Goury & Duriez IEEE Trans on Robotics, 2018

MODEL ORDER REDUCTION

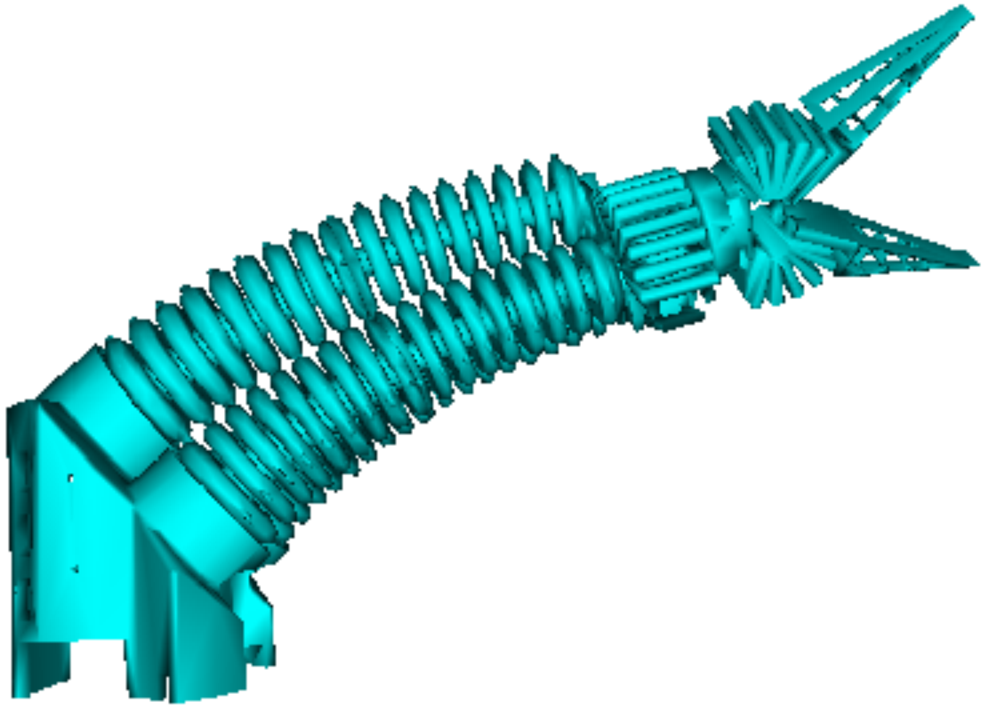


Goury & Duriez IEEE Trans on Robotics, 2018

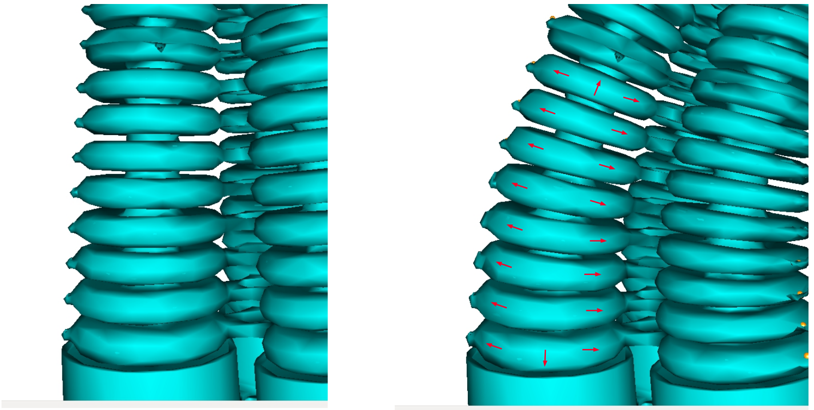


CONSTRAINT-BASED MODELING

CABLE AND PRESSURE

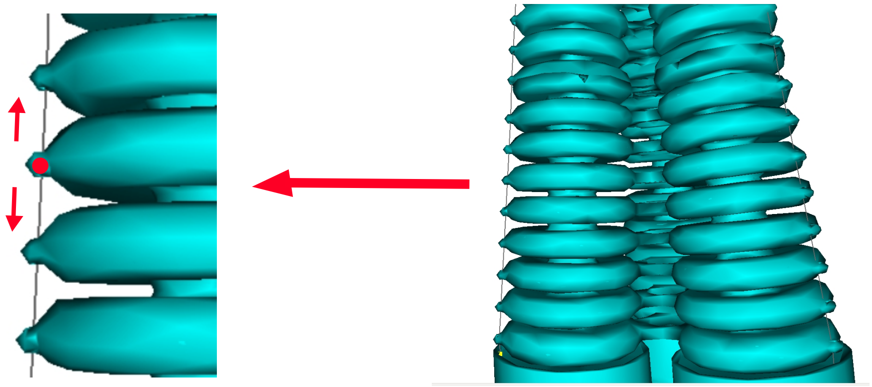


Festo deformable trunk



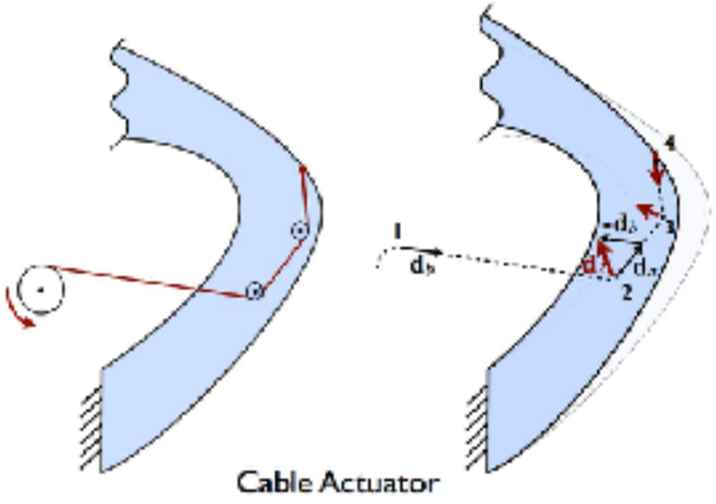
Pressure actuation

$\delta =$ volume change

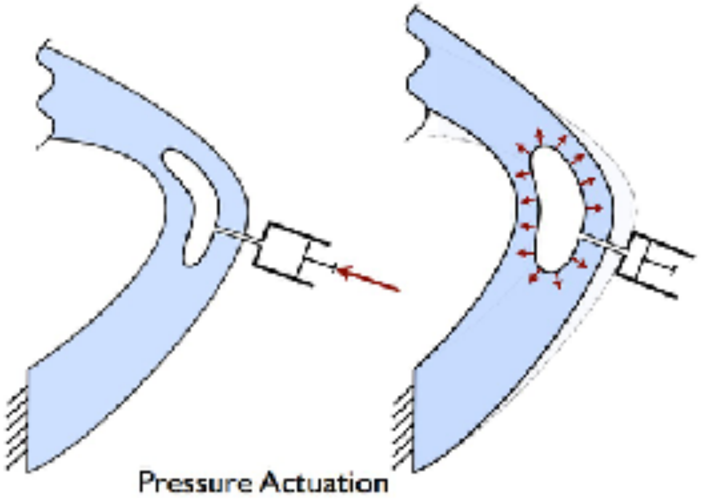


Cable actuation

$\delta =$ length change



Cable Actuator



Pressure Actuation

$\mathbf{H}^T\lambda \leftarrow$ cable force pressure

$\uparrow$

Force distribution

$$\underbrace{(M - hD - h^2K)}_{A}\, dv = \underbrace{hf_{ext} + hf(x_i, v_i) + h^2Kv_i}_{b}$$

$$\underbrace{-K}_{A}\, dx = \underbrace{f(x_{i-1}) + f_{ext}}_{b} \qquad \text{(quasi static)}$$

Modeling Approach

For effector, actuator and contact we use **Lagrange multipliers**:

$$\begin{pmatrix} A & H^T \\ H & 0 \end{pmatrix} \begin{pmatrix} dx \\ -\lambda \end{pmatrix} = \begin{pmatrix} b \\ \delta \end{pmatrix}$$

Modeling Approach

For effector, actuator and contact we use **Lagrange multipliers**:

$$\begin{pmatrix} A & H^T \\ \boxed{H} & 0 \end{pmatrix} \begin{pmatrix} dx \\ -\lambda \end{pmatrix} = \begin{pmatrix} b \\ \delta \end{pmatrix}$$

Constraint
Jacobian:
direction of the
constraint forces

Modeling Approach

For effector, actuator and contact we use **Lagrange multipliers**:

$$\begin{pmatrix} A & H^T \\ H & 0 \end{pmatrix} \begin{pmatrix} dx \\ \bigcirc\!\!\!\!\!\!-\lambda \end{pmatrix} = \begin{pmatrix} b \\ \delta \end{pmatrix}$$

Lagrange
multiplier:
constraint
effort

Modeling Approach

For effector, actuator and contact we use **Lagrange multipliers**:

$$\begin{pmatrix} A & H^T \\ H & 0 \end{pmatrix} \begin{pmatrix} dx \\ -\lambda \end{pmatrix} = \begin{pmatrix} b \\ \enclose{circle}{\delta} \end{pmatrix}$$

Shift, volume growth…

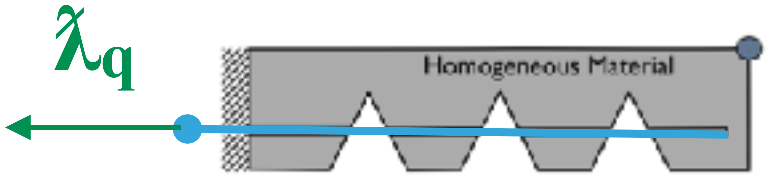
Schur Complement:

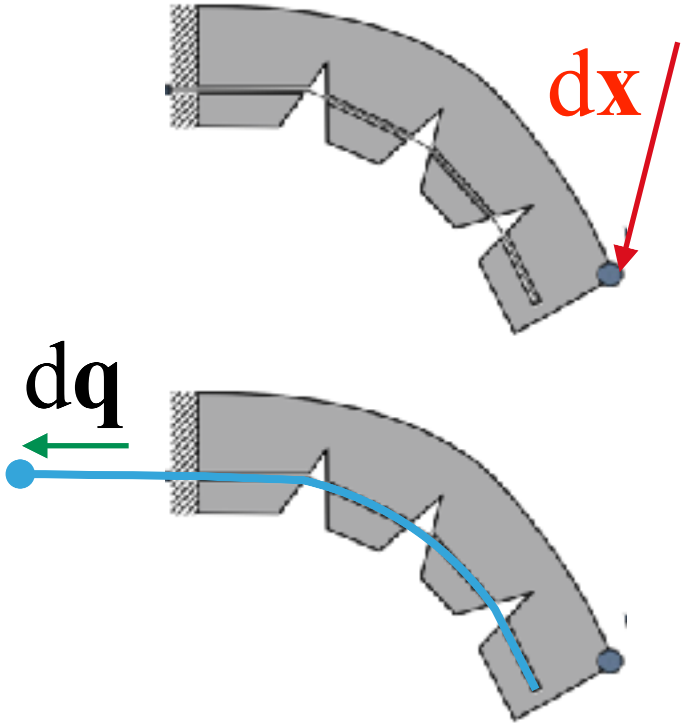
$$W = HA^{-1}H^T$$

WHAT IS W ? : THE COMPLIANCE

$$\mathbf{dx} = \mathbf{W_{xq}(q)}\,\boldsymbol{\lambda_q}$$

- Mechanical coupling
  - For a force on actuator space, what displacement on effector and actuator spaces ?
  - By combining compliances we obtain the kinematic model:

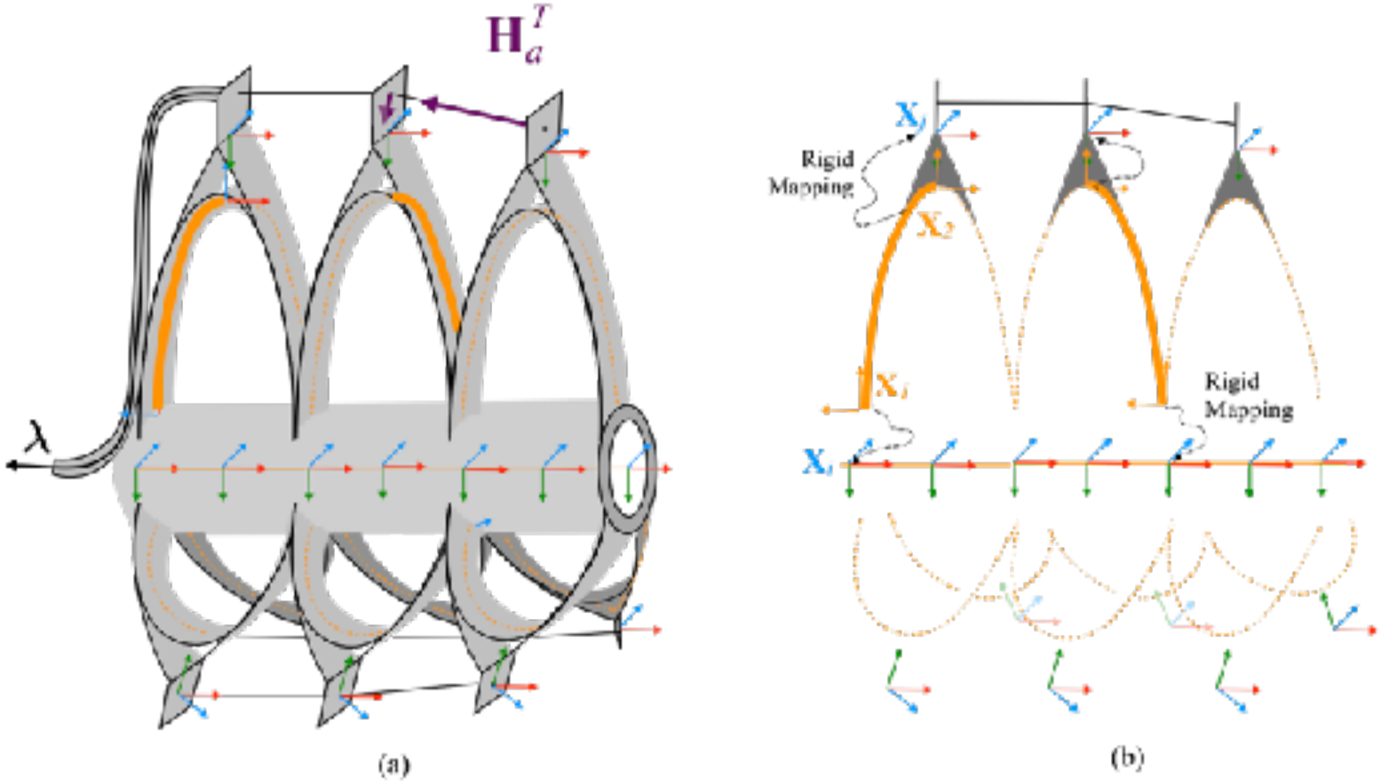


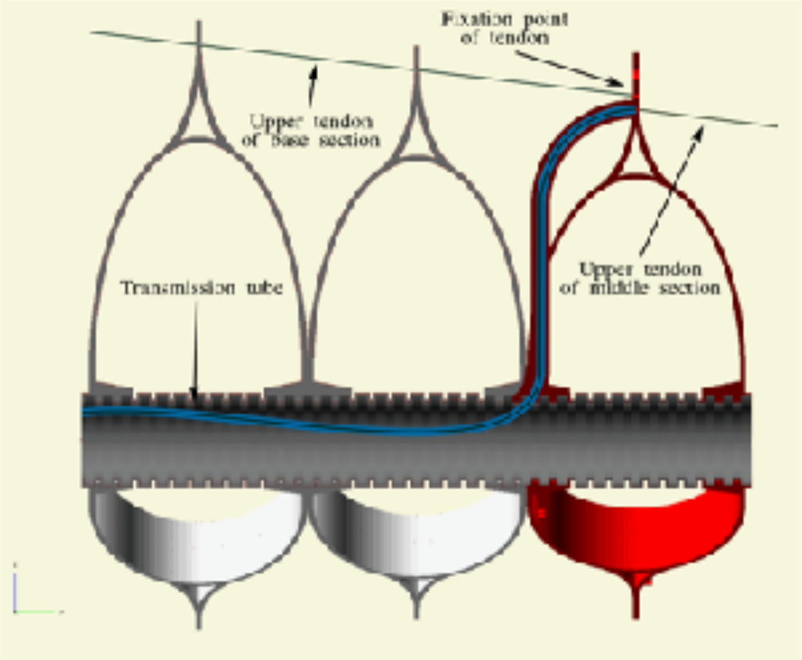


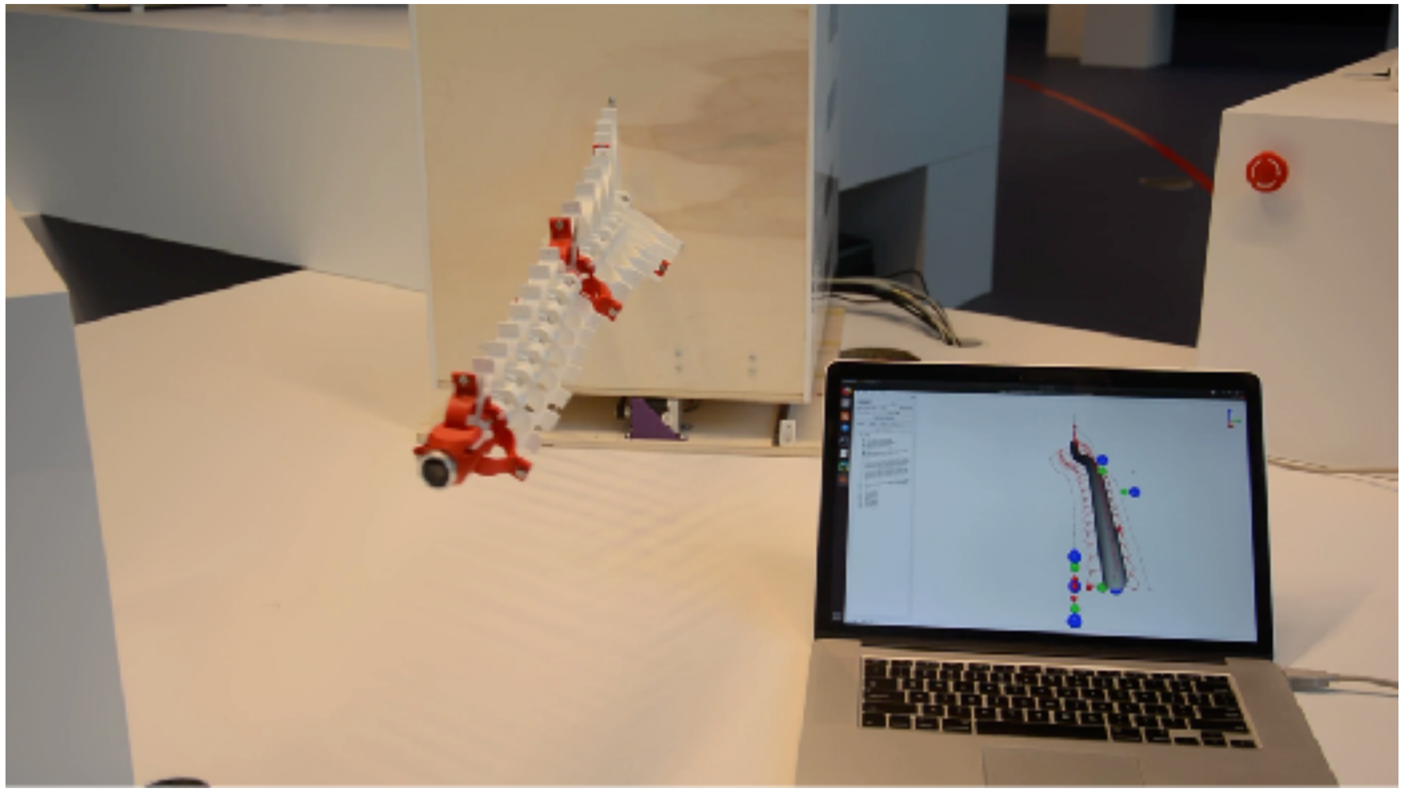
$$\mathbf{dq} = \mathbf{W_{qq}(q)}\,\boldsymbol{\lambda_q}$$

$$\mathbf{dx} = \mathbf{J(q)\,dq} = \mathbf{W_{xq}(q)\,W_{qq}^{-1}(q)dq}$$

SOFT-ROBOT DESIGN & CONTROL



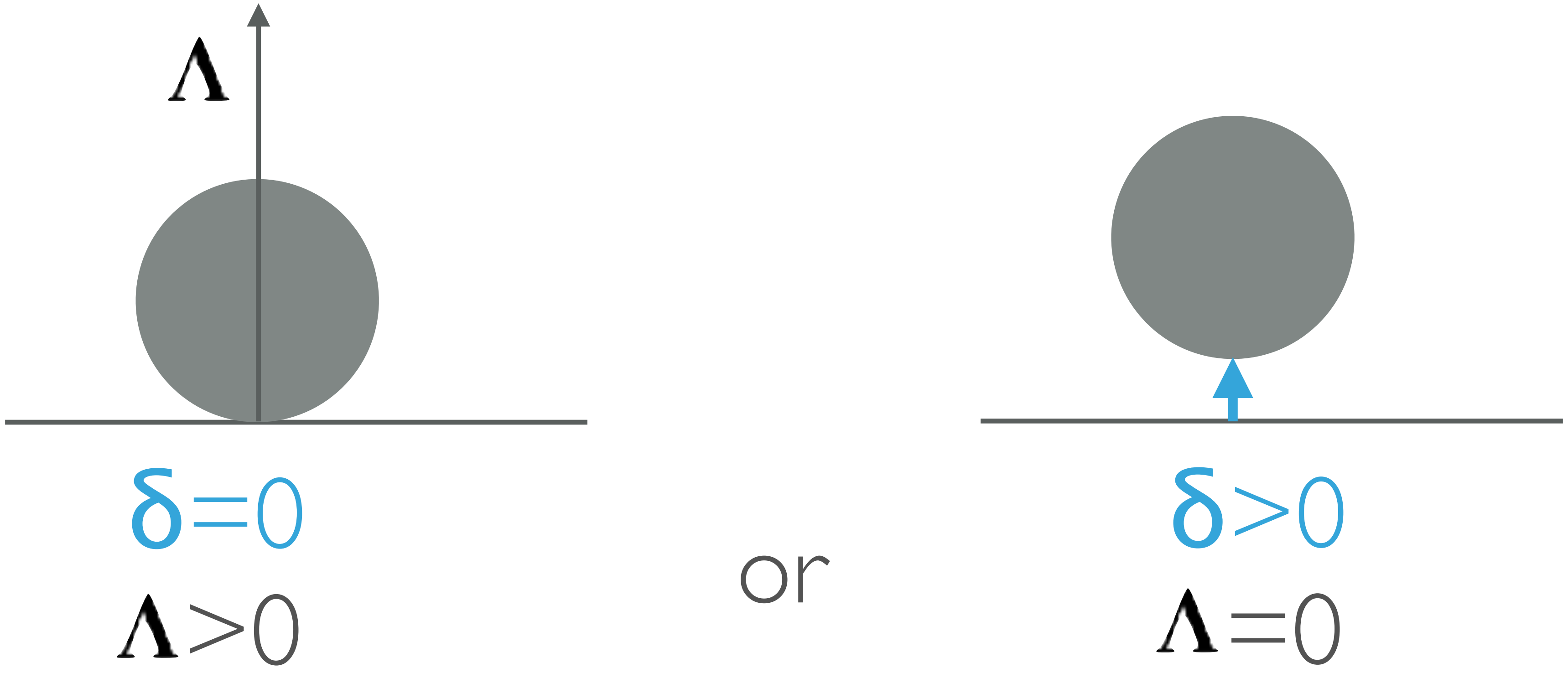




Bieze et al. IJRR 2020

COMPLEMENTARITY CONSTRAINTS

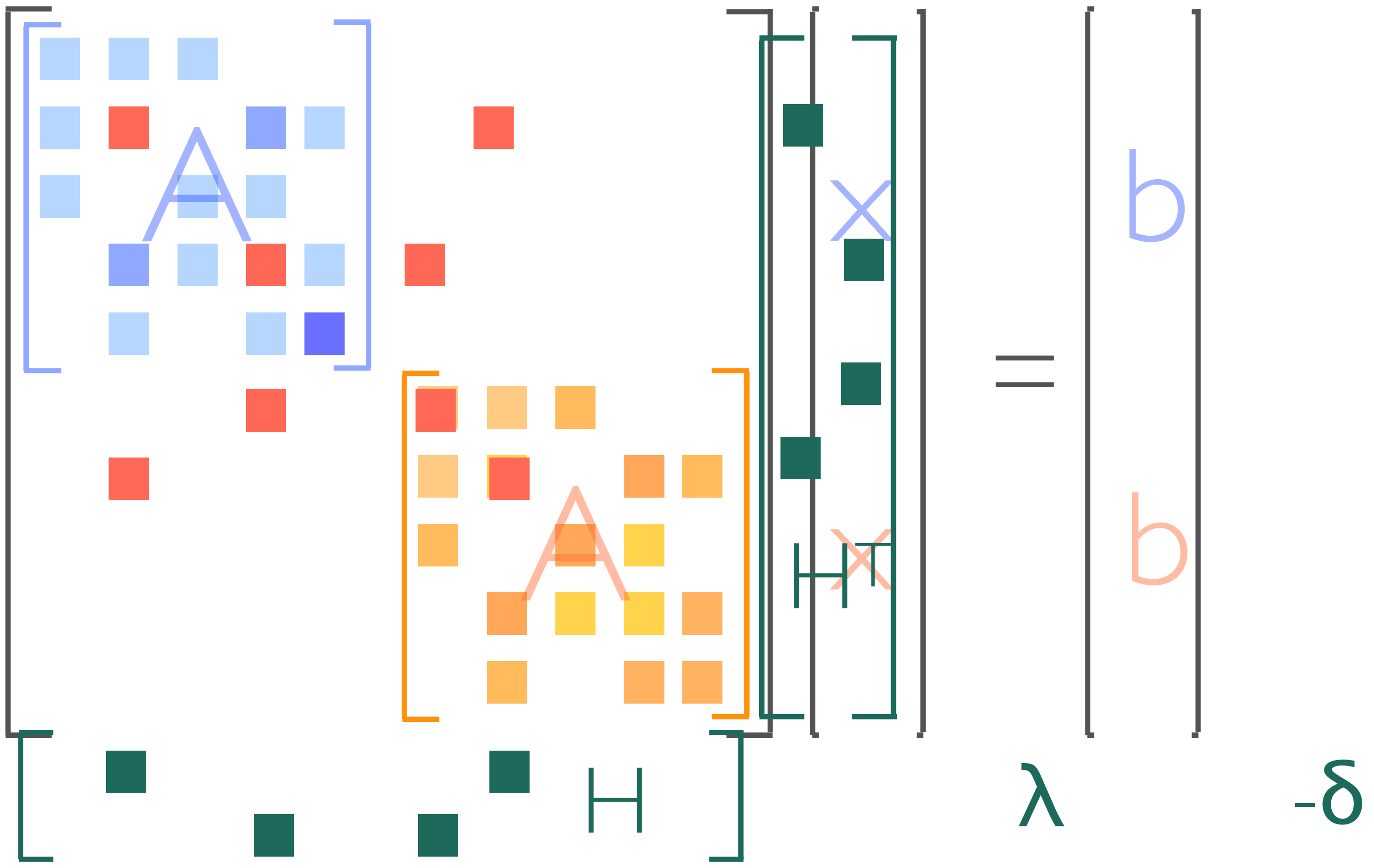
- **Contact**
  - Signorini's law = two constraint cases:



$$0 \leq \delta_n \perp \lambda_n \geq 0$$

CHALLENGES

- Large systems of coupled equations
  - Penalty
  - Lagrange multipliers

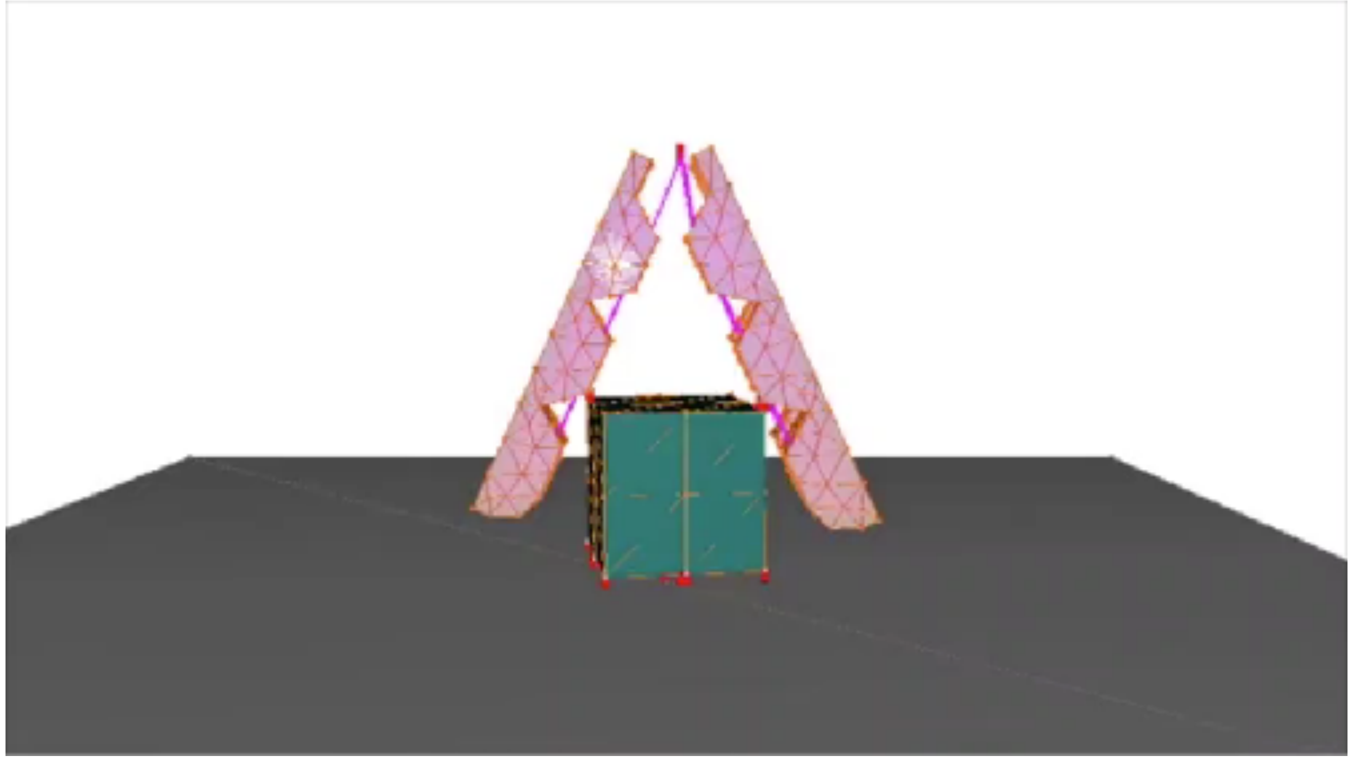


Schur Complement:

$$W = HA^{-1}H^T + HA^{-1}H^T$$

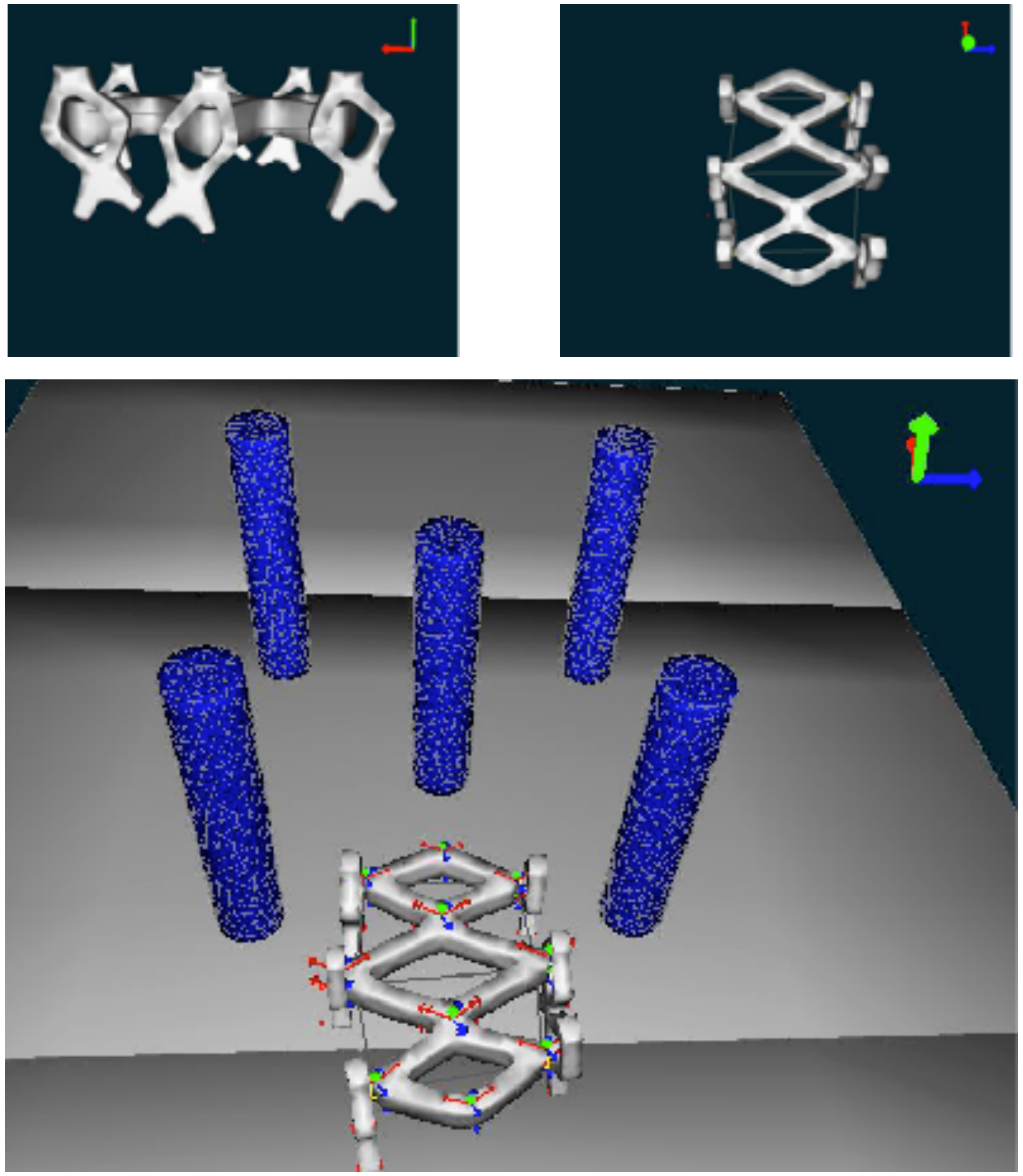
(Friction)Contact = (N)LCP solver

SOFT ROBOT INTERACTING WITH THEIR ENVIRONMENT



Speed x3

Grasping simulation



Locomotion

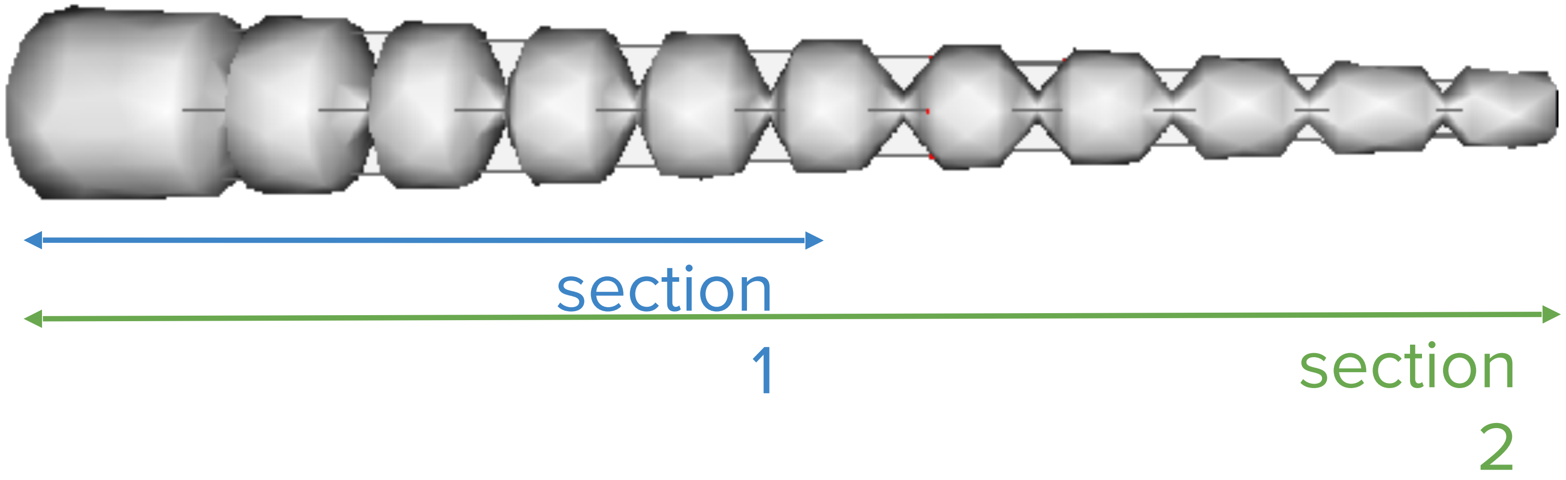
Coevoet et al. Advanced Robotics 2017

CONTROL METHODS

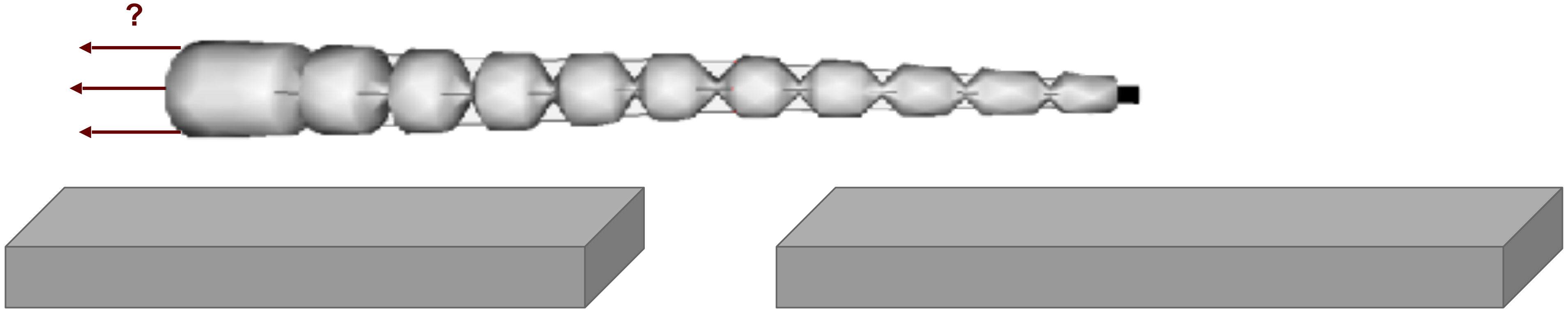
- Inverse kinematics
- Sensing & Closed-loop control

PROBLEM STATEMENT

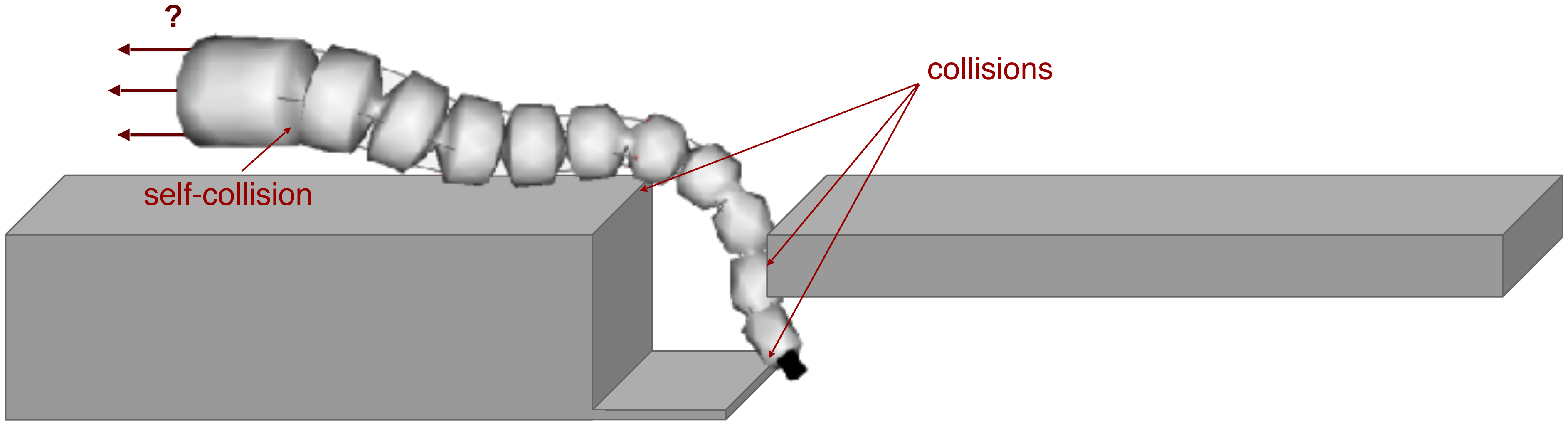
Soft trunk: 8 cables



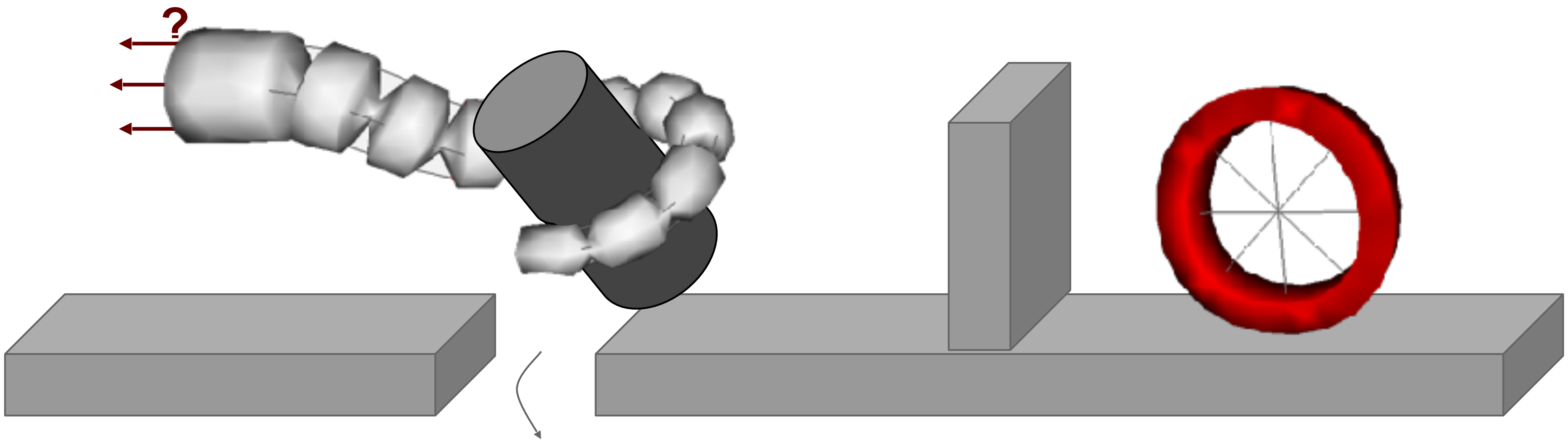
TASK: INSPECTION



TASK: INSPECTION WITH COLLISION



TASK: MANIPULATION & LOCOMOTION



Difficulties:

- Continuous deformation: infinite DoFs
- Hyper-redundant and under-actuated
- Highly sensitive to environmental factors

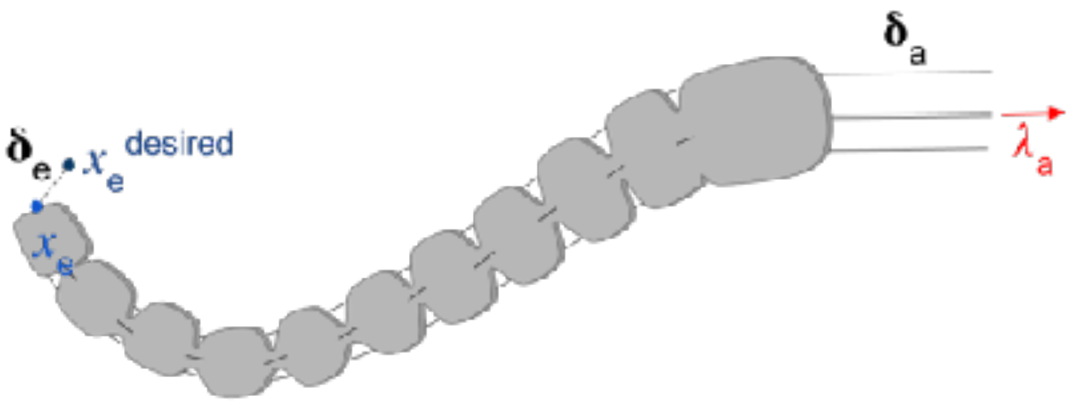
➔Modeling and control more complex

➔**Require new techniques**

Trivedi et al. (2008) - *Applied Bionics and Biomechanics*

Rus & Tolley (2015) - *Nature*

OPTIMIZATION : PROBLEM



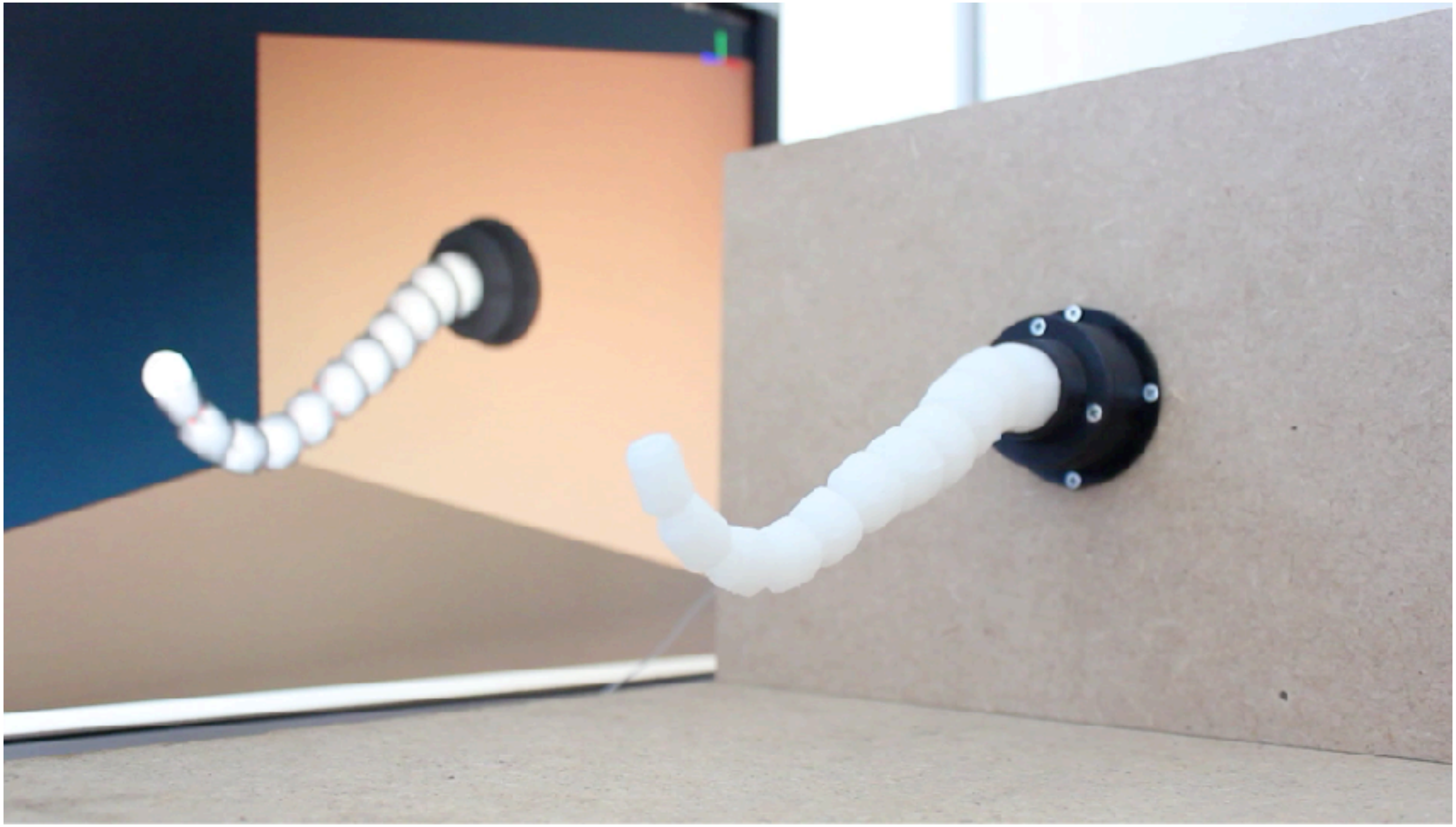
Formulation of Quadratic Program (QP) with linear constraints:

$$min(\ \|\ \boldsymbol{\delta}_e \ \|^2\ )= min \left(\frac{1}{2} \boldsymbol{\lambda_a}^T \mathbf{W}_{ea}^T \mathbf{W}_{ea} \boldsymbol{\lambda_a} + \boldsymbol{\lambda_a}^T \mathbf{W}_{ea}^T \boldsymbol{\delta}_e^{\text{free}} \right)$$

$$\begin{aligned}
&subject\ to\ \text{(course of actuators)}: \\
&\delta_{min} \leq \boldsymbol{\delta}_a = \mathbf{W}_{aa} \boldsymbol{\lambda_a} + \boldsymbol{\delta}_a^{\text{free}} \leq \delta_{max} \\
&and\ \text{(case of unilateral effort actuation)}: \\
&\boldsymbol{\lambda_a} \geq 0
\end{aligned}$$

OPTIMIZATION : EXPERIMENTS

	#DoFs	#Elem	W	QP	Sim.
trunk	2127	1972	2.8 *ms*	< 0.1 *ms*	24.1 *ms*



OPTIMIZATION : PROBLEM

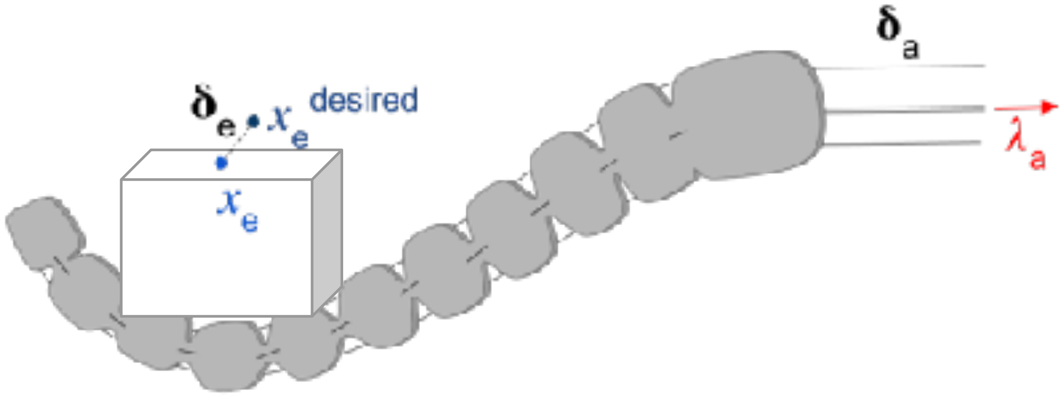
- Redundancy problem (not naturally well-posed)
  - If $\mathrm{W}_{ea}{}^{\mathrm{T}}\mathrm{W}_{ea}$ is not definite (num Effectors ≤ num Actuators),
    - Not unique solution
    - Add to objective: expression of actuators mechanical work $\boldsymbol{E = \Delta\delta_a \lambda_a = \lambda^T_a \mathbf{W}_{aa} \lambda_a}$

$$min\left(\frac{1}{2}\boldsymbol{\lambda_a}^T\mathbf{W}_{ea}^T\mathbf{W}_{ea}\boldsymbol{\lambda_a} + \boldsymbol{\lambda_a}^T\mathbf{W}_{ea}^T\boldsymbol{\delta}_e^{\text{free}} + \epsilon\, \boldsymbol{E}\right)$$

subject to (course of actuators) :
$\delta_{min} \leq \boldsymbol{\delta}_a = \mathbf{W}_{aa}\boldsymbol{\lambda_a} + \boldsymbol{\delta}_a^{\text{free}} \leq \delta_{max}$
and (case of unilateral effort actuation)
$\boldsymbol{\lambda_a} \geq 0$

$$\epsilon = \mathbf{1e^{-3}}\, ||\mathbf{W}_{ea}{}^{\mathbf{T}}\mathbf{W}_{ea}||_\infty / ||\mathbf{W}_{aa}||_\infty$$

OPTIMIZATION: CONTACTS



Formulation of Quadratic Program with linear Complementarity Constraints (QPCC):

$$\min_{(\lambda_a,\,\lambda_c)} \ ||\delta_e||^2$$

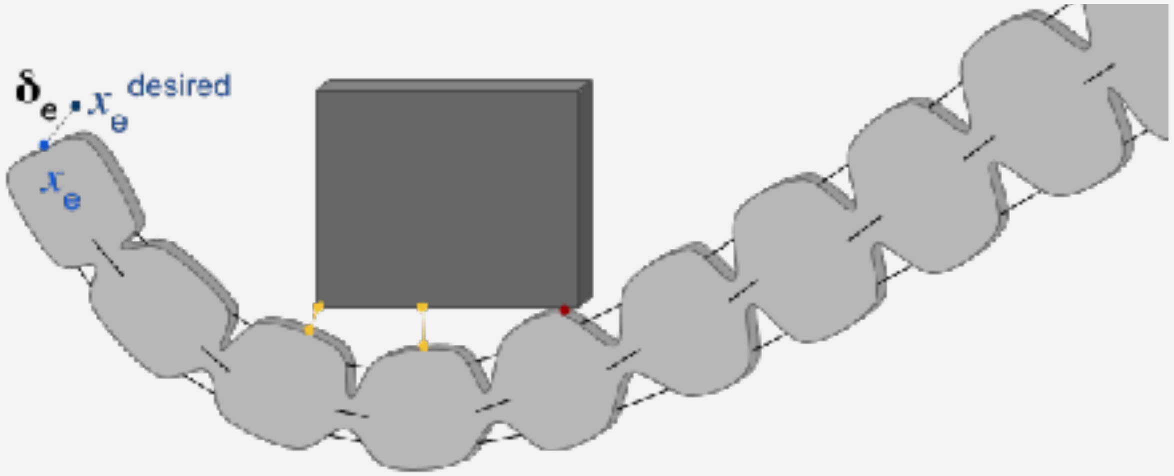
s.t : (1) $\delta_{max} \geq \delta_a \geq \delta_{min}$

(2) $0 \leq \lambda_c \perp \delta_c \geq 0$

(1) Constraints on actuators (such as limit on cable displacement)
(2) Complementarity constraint for contacts

QPCC Solver

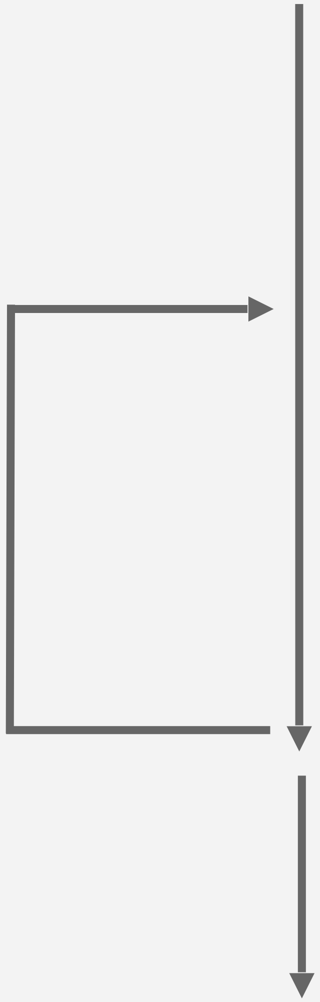
➜ **Specific solver** based on decomposition method:



How to find the feasible and optimal set $\mathbf{I}$ of active contacts?

QPCC Solver

Algorithm: Iterative method



Starts from feasible set I (solve the contact with fixed actuation)

Solve $(QP)_I$

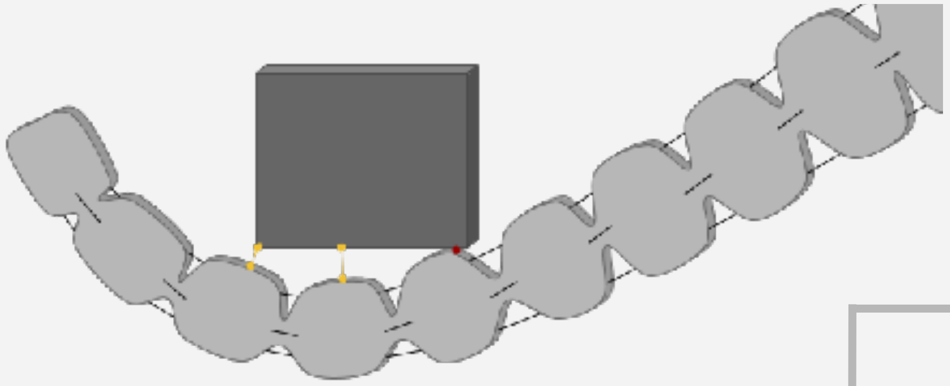
Look at constraints: reach boundary = candidate for pivot (Tan et al. (2012) - *TOG*)

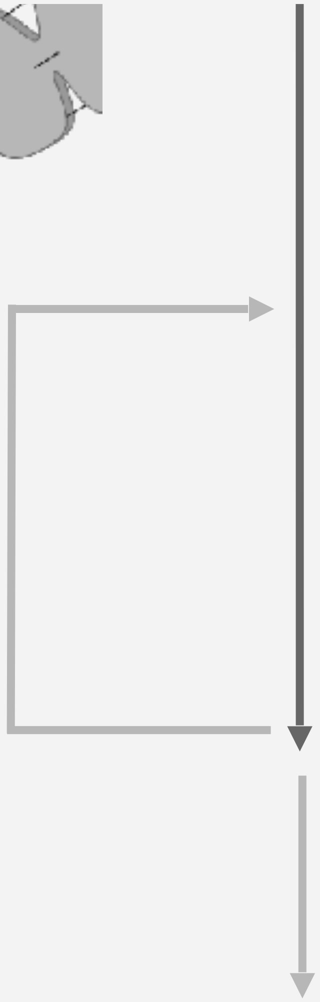
Pivot constraint with the greater dual variable

➜ New set I of active contacts

No more candidate for pivot
➜ solution

QPCC Solver





Starts from feasible set I (solve the contact with fixed actuation)

Solve $(QP)_I$

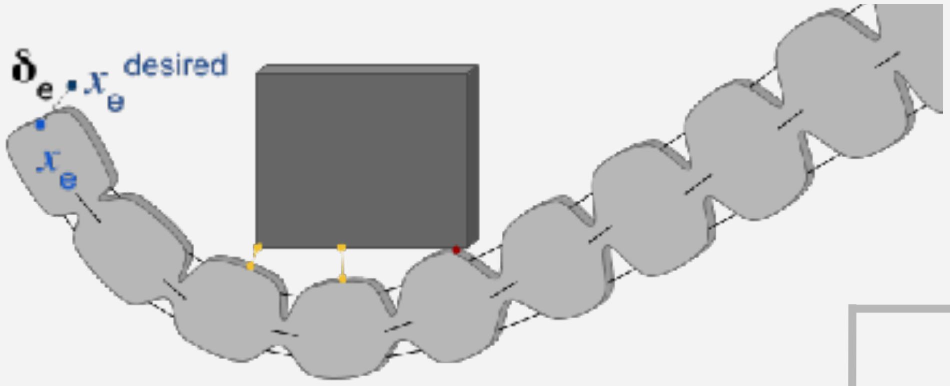
Look at constraints: reach boundary = candidate for pivot (Tan et al. (2012) - *TOG*)

➜ New set I of active contacts

Pivot constraint with the greater dual variable

No more candidate for pivot
➜ solution

QPCC Solver



Starts from feasible set I (solve the contact with fixed actuation)

Solve *(QP)_I*

Look at constraints: reach boundary = candidate for pivot (Tan et al. (2012) - *TOG*)

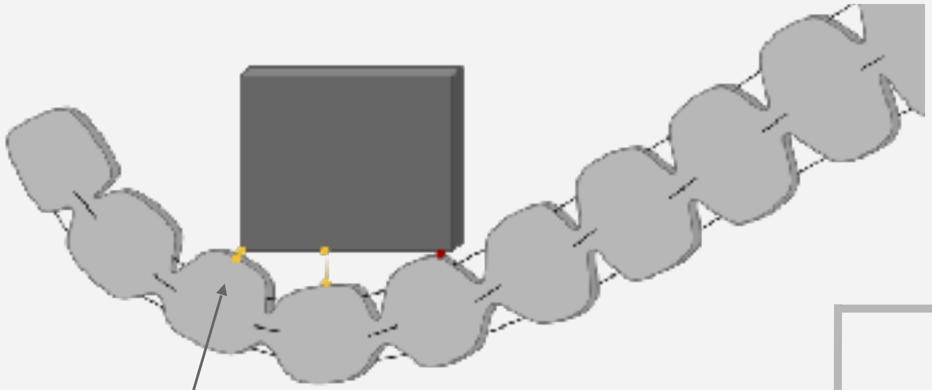
Pivot constraint with the greater dual variable

➜ New set I of active contacts

No more candidate for pivot

➜ solution

QPCC Solver



Candidate for pivot

Starts from feasible set I (solve the contact with fixed actuation)

Solve $(QP)_I$

Look at constraints: reach boundary = candidate for pivot (Tan et al. (2012) - *TOG*)

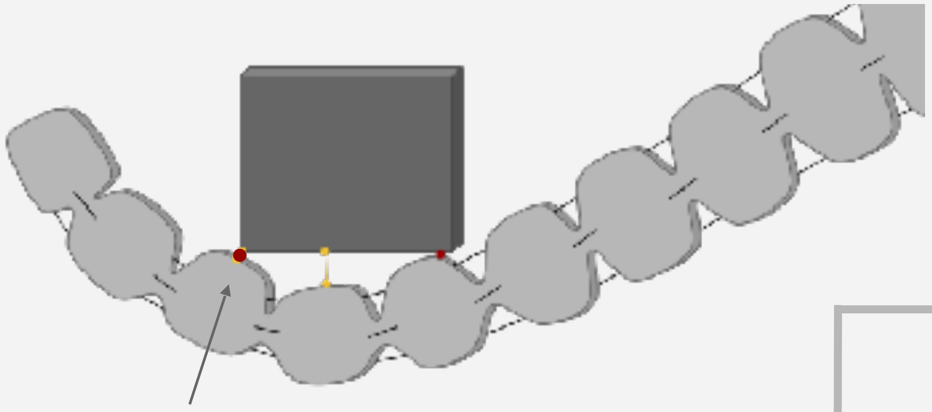
➜ New set I of active contacts

Pivot constraint with the greater dual variable

No more candidate for pivot

➜ solution

QPCC Solver



Pivot contact
to active

Starts from feasible set I (solve the contact with fixed actuation)

Solve *(QP)*$_I$

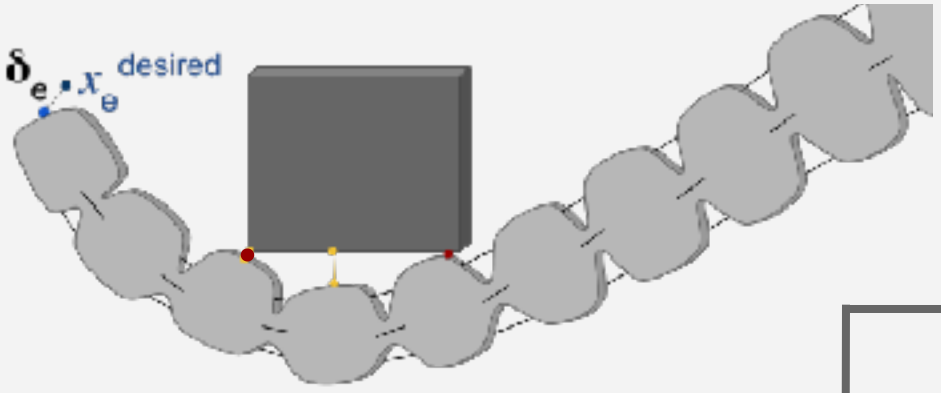
Look at constraints: reach boundary = candidate for pivot (Tan et al. (2012) - *TOG*)

Pivot constraint with the greater dual variable

➜ New set I of active contacts

No more candidate for pivot
➜ solution

QPCC Solver



Starts from feasible set I (solve the contact with fixed actuation)

Solve *$(QP)_I$*

Look at constraints: reach boundary = candidate for pivot (Tan et al. (2012) - *TOG*)

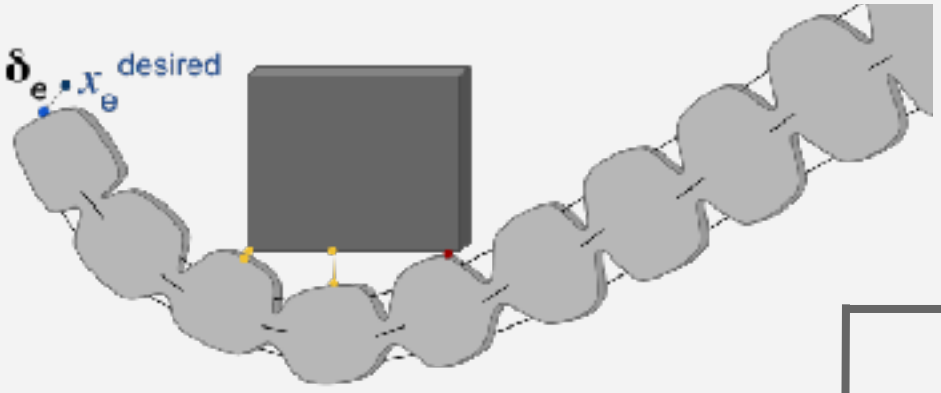
Pivot constraint with the greater dual variable

➜ New set I of active contacts

No more candidate for pivot

➜ solution

QPCC Solver



Starts from feasible set I (solve the contact with fixed actuation)

Solve *(QP)*$_I$

Look at constraints: reach boundary = candidate for pivot (Tan et al. (2012) - *TOG*)

➜ New set I of active contacts

Pivot constraint with the greater dual variable

No more candidate for pivot
➜ solution

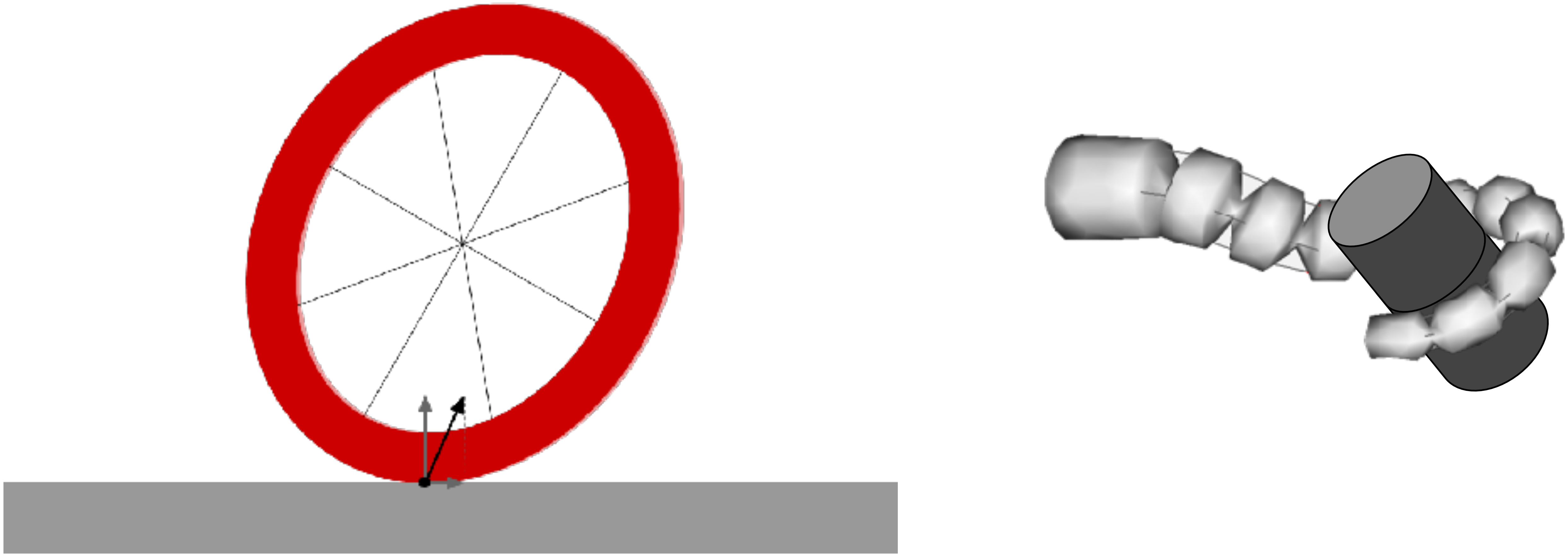
EXTENSION TO CONTACT

	#DoFs	#Elem	#Conct.	W	QP(s)	Sim.
trunk	2127	1972	51	10.09 *ms*	4.91 *ms*	34.52 *ms*

EXTENSION TO STICK CONTACT

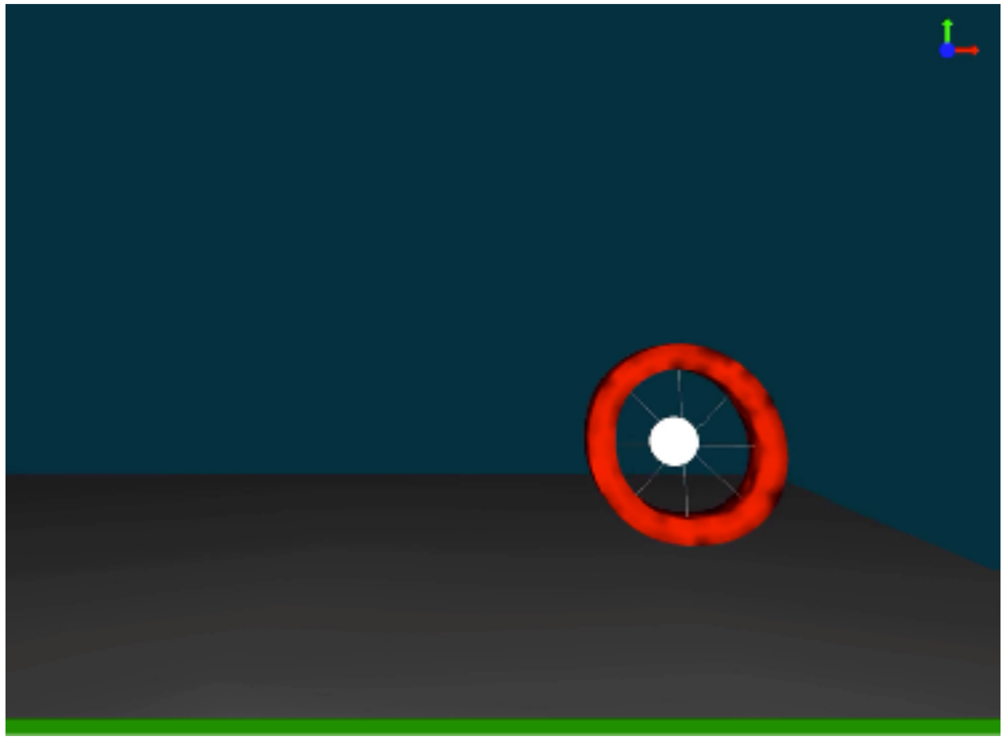
Contact force given by the composition of normal and tangential (friction) forces:

$$\mathbf{H}_c^T\lambda_c = \mathbf{H}_n^T\lambda_n + \mathbf{H}_t^T\lambda_t$$



EXTENSION TO STICK CONTACT

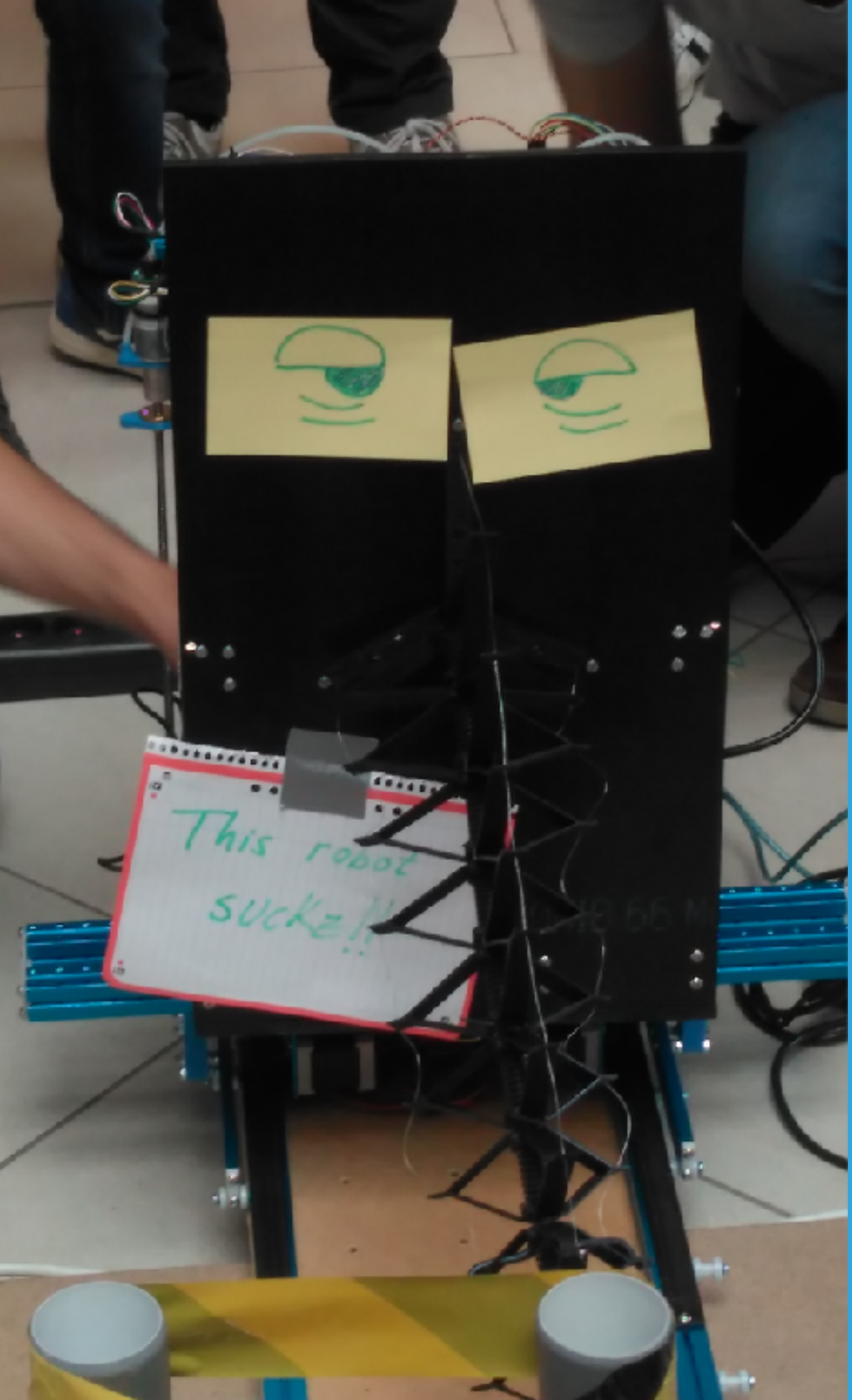
- Locomotion
- Manipulation



How to change the tendon lengths to make the wheel rolls to a desired position ?

How to pull on tendons to control the plastic cup motion ?

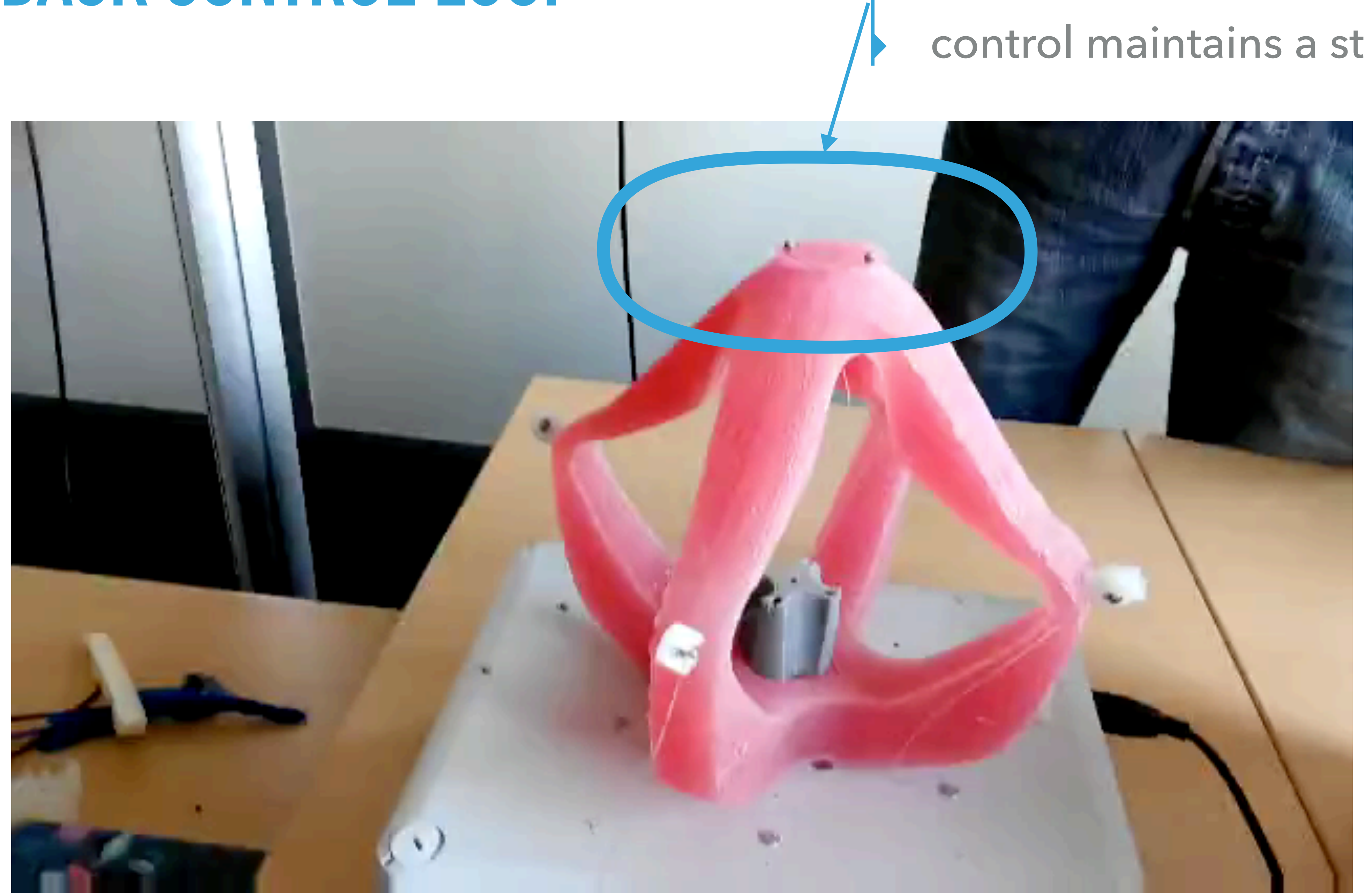
Coevoet et al., Robosoft Conference 2019



SENSING & CLOSED-LOOP CONTROL

FEEDBACK CONTROL LOOP

- position measured with camera
- control maintains a stable position

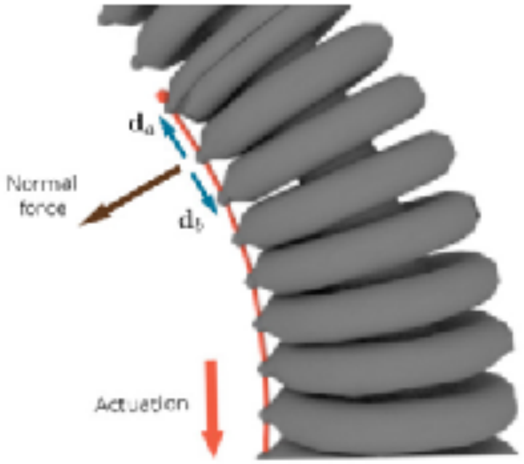


$$\color{red}{dx} = \mathbf{J(q)\ dq = W_{xq}(q)\ W_{qq}^{-1}(q)dq}$$

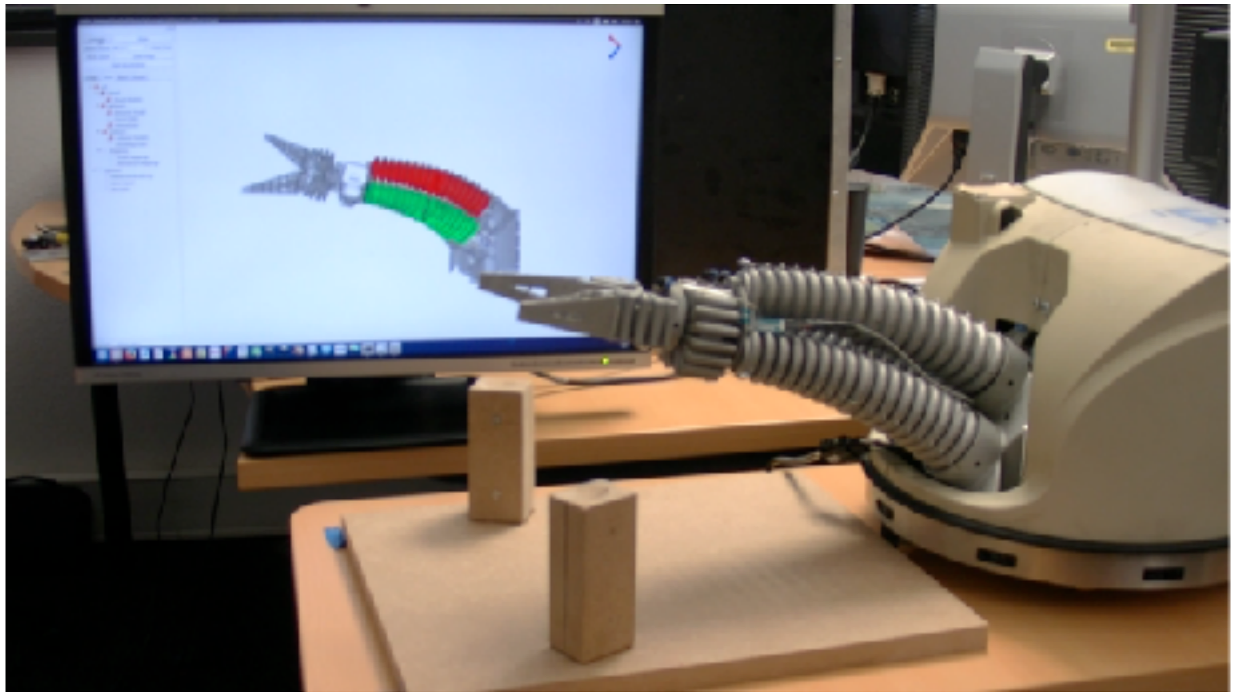
Zhang, Z., et al, IROS 2016

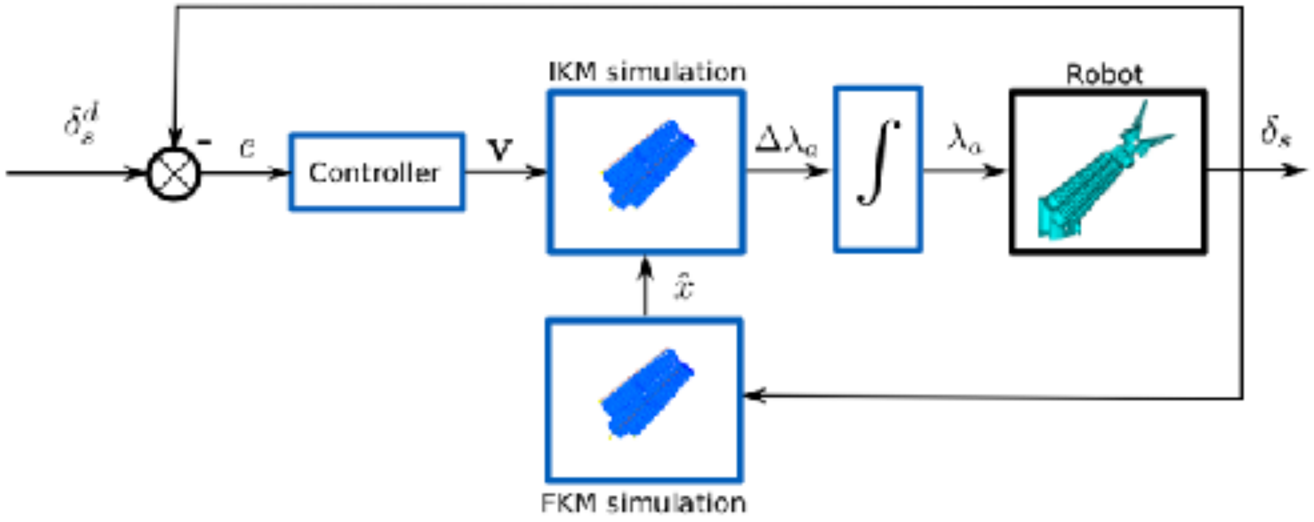
proof of convergence: Zhang, Z., et al, IROS 2017

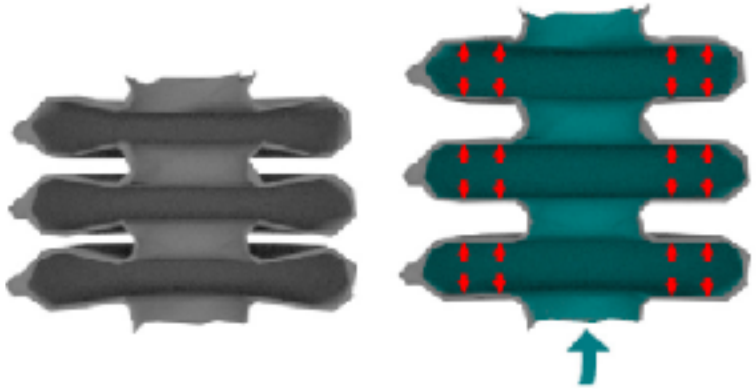
FEEDBACK CONTROL LOOP



Sensors : tendons (length)



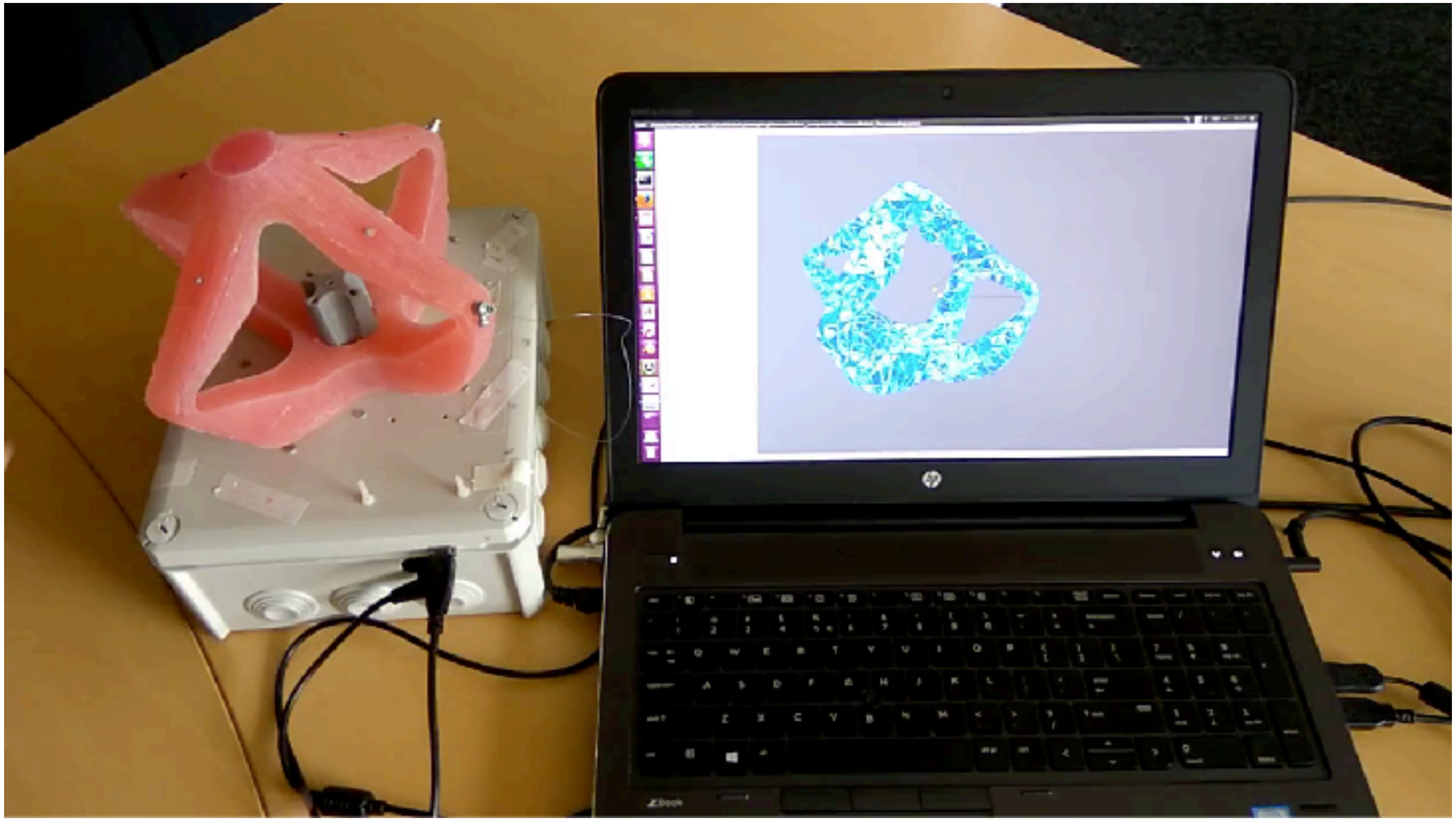




Actuators : pressure

Bieze et al, SoRo 2018

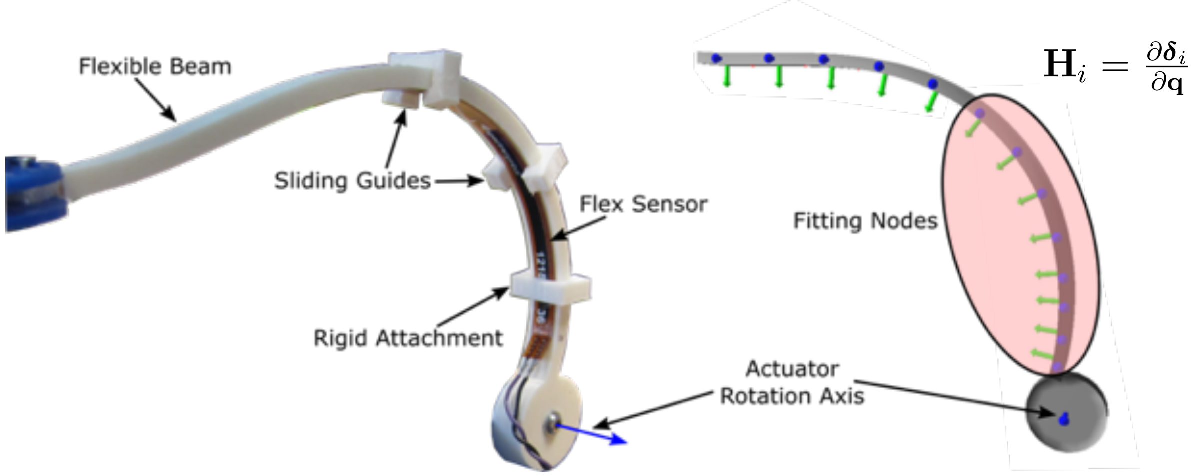
FUSE MODEL AND SENSOR INFORMATIONS

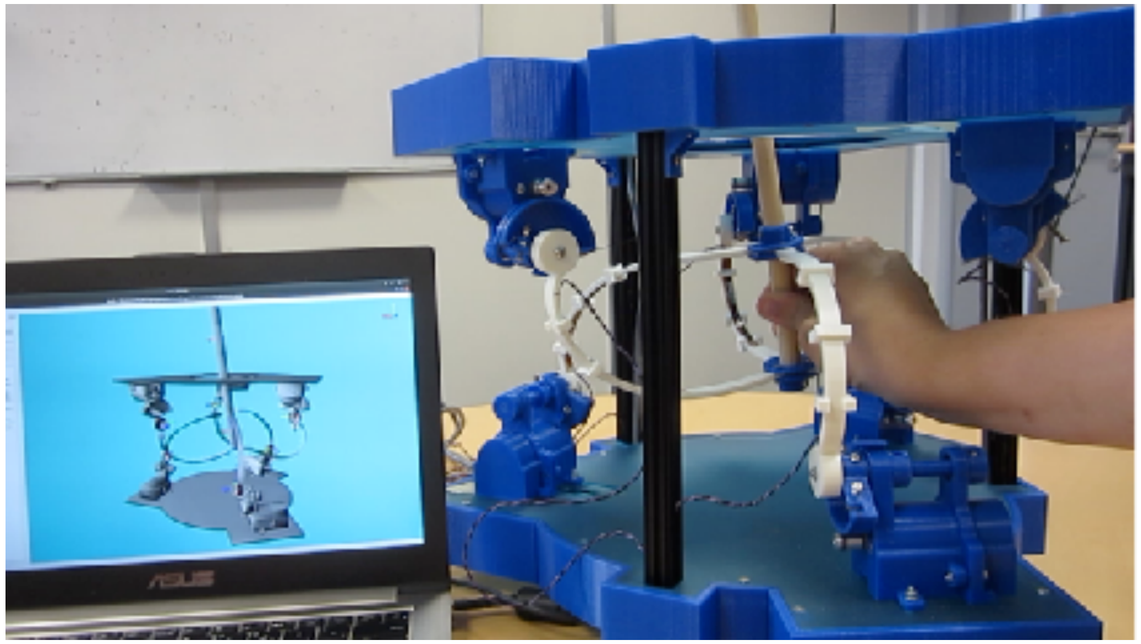


Zhang, Z. et al ICRA 2018 & Robotic and Automation Letters

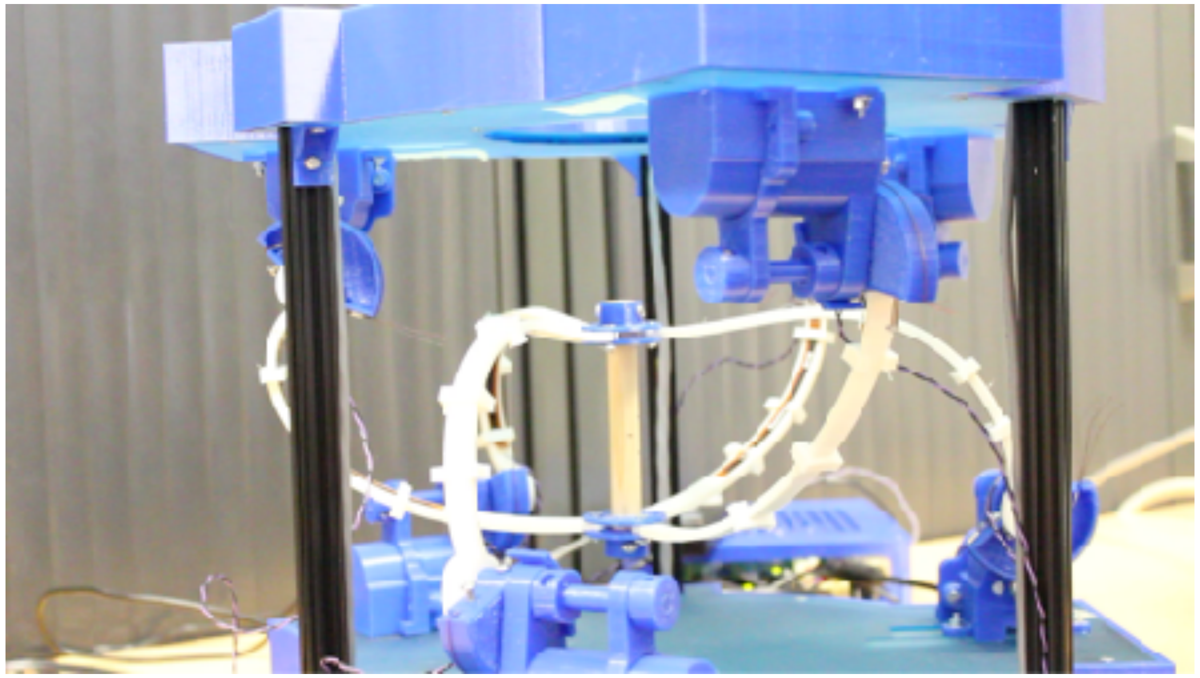
HAPTIC FEEDBACK CONTROL

- Fuse model and informations from sensors





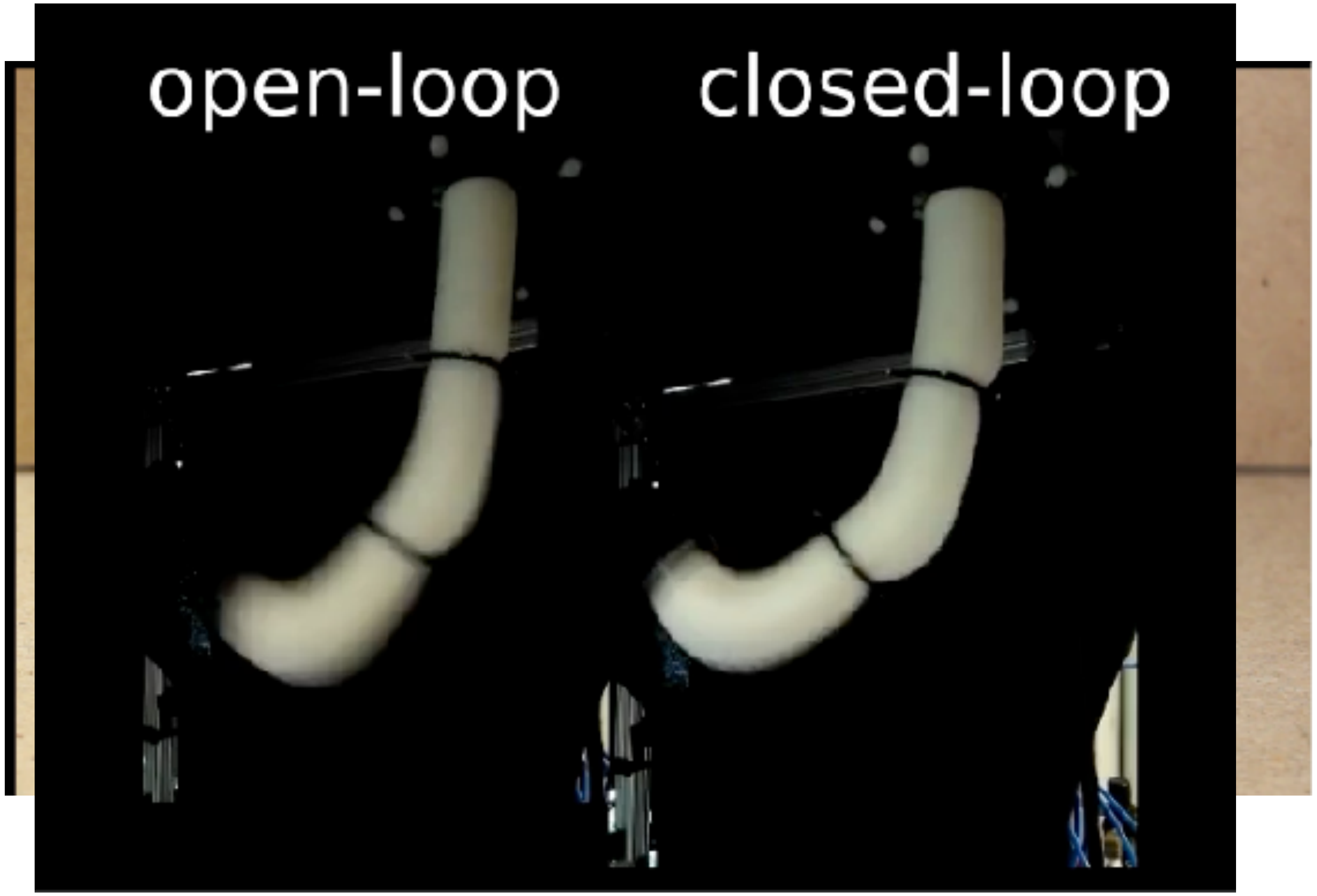
- Control at high rates (in parallel of the simulation of the robot)
- Linearization at low rate
- 2 inverse problem solved at high rates:
  - Positions found by sensors=> force
  - Actuation



DYNAMIC CONTROL

- Additional difficulty to cancel natural vibration: refresh rates
- Model order reduction + feedback control

- Observer in the reduced space
- Linearization at the desired position
- Remove low frequency vibrations



M. Thieffry et al., IEEE TCST 2019

Katzschmann, R. K., Thieffry M et al., IEEE Robosoft 2019

CONCLUSION & PERSPECTIVES

SOFT ROBOTICS

- Deformable/Soft robots = a new way of designing robots

- Research on hardware
  - How to design these robots ? bioinspiration ?
  - Choices of materials => links with « smart materials »
  - Integration (structure, electronics, actuators, sensors…) while keeping the desired softness !

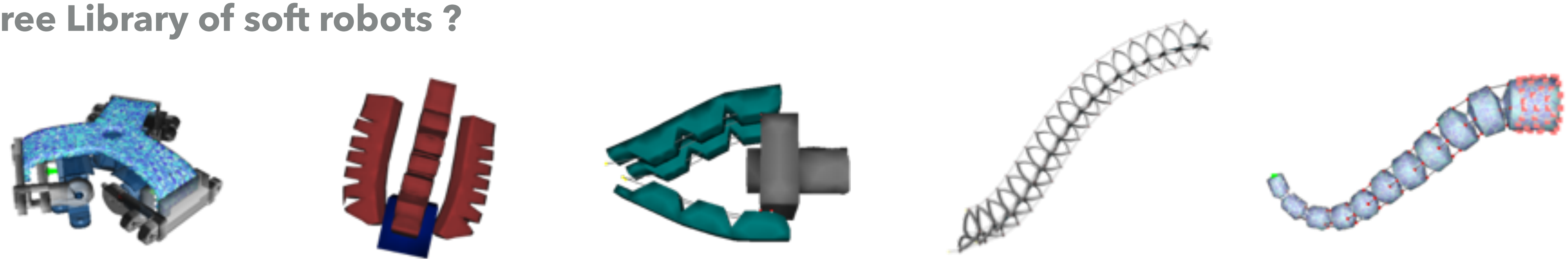
- Research on software
  - New models, based on finite element approaches or similar methods
  - No methodology (so far) for design
  - Importance of allowing contacts with environment
  - Control methods for systems with large number of degrees of freedom

SOFT-ROBOTICS : PLAY WITH IT WITH SOFA !

- Our code is based on SOFA (LGPL License)
    - Please visit the website sofa-framework.org (LGPL License)
        - You can use it & contribute to it !
        - We have launched a consortium: code maintenance, forum, training, code distribution…
        - If you want to support these activities, your institution can take a membership
            - For instance Kyungpook University (south Korea), University of Florida (US) are members

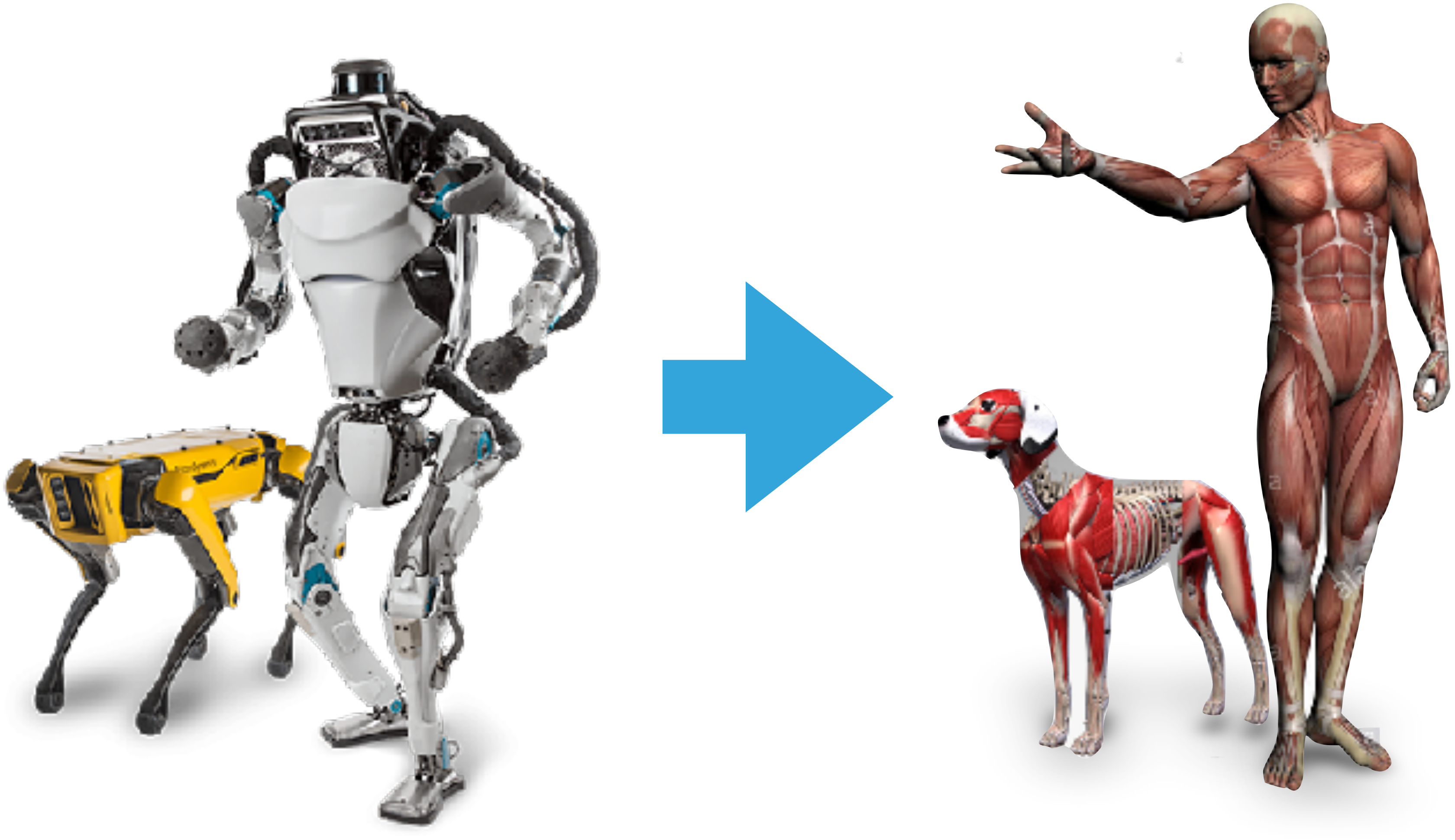
- Plugins developed by the team
    - soft-robot plugin: direct simulation of soft robots (LGPL License)
    - Plugin on reduced order model (GPL License)
    - Inverse simulation / control (patent pending…)
        - Free binaries distribution for research
        - To access the code, easy licence agreement with Inria for research

- We did our first Tutorial on Sunday, we hope to gather new users & contributors !
    - https://handsonsoftrobotics.lille.inria.fr (tutorial)
    - https://team.inria.fr/defrost/ (team website)
    - **Free Library of soft robots ?**



OPPORTUNITIES

- « Hybrid » robots made of soft and hard materials… ?



THANKS !

- Faculty
    - Jérémie Dequidt: Associate Prof **Université de Lille**
    - Gang Zheng: Senior researcher **INRIA** (Vice-Leader)
    - Olivier Goury: Junior researcher **INRIA**
    - Alexandre Kruszewsky: Associate Prof **Centrale Lille**
- Associate Engineers
    - Damien Marchal: Research engineer **CNRS** (half time)
    - Bruno Carrez: Research engineer **INRIA** (half time)
- Engineers
    - Thor Morales Bieze: Hardware Engineer
    - Meichun Lin: Software Engineer
    - Bruno Marquez: Software Engineer
    - Yinoussa Adagolodjo: Software Engineer
- PhD students
    - Antonin Bernardin: Simulation of suction cup (Collaboration with M. Marchal)
    - Walid Amehri: Controllability and observability of soft robots
    - Felix Vanneste: Mesostructured soft-robot
    - Pierre Schegg: Robot Catheter navigation
    - Ke Wu: Dynamic control
- Post-doc
    - Stefan Escaida: Sensors for soft robots
- Alumni (work presented in this presentation)
    - Zhongkai Zhang: Visual servoing for soft robots
    - Eulalie Coevet: Inverse models for soft-robots
    - Maxime Thieffry: Dynamic control
    - Margaret Koehler (Visiting from Stanford Univ)
    - Thomas Morzadec: Shape optimization

- https://team.inria.fr/defrost/ (team website)

This work has been supported by: