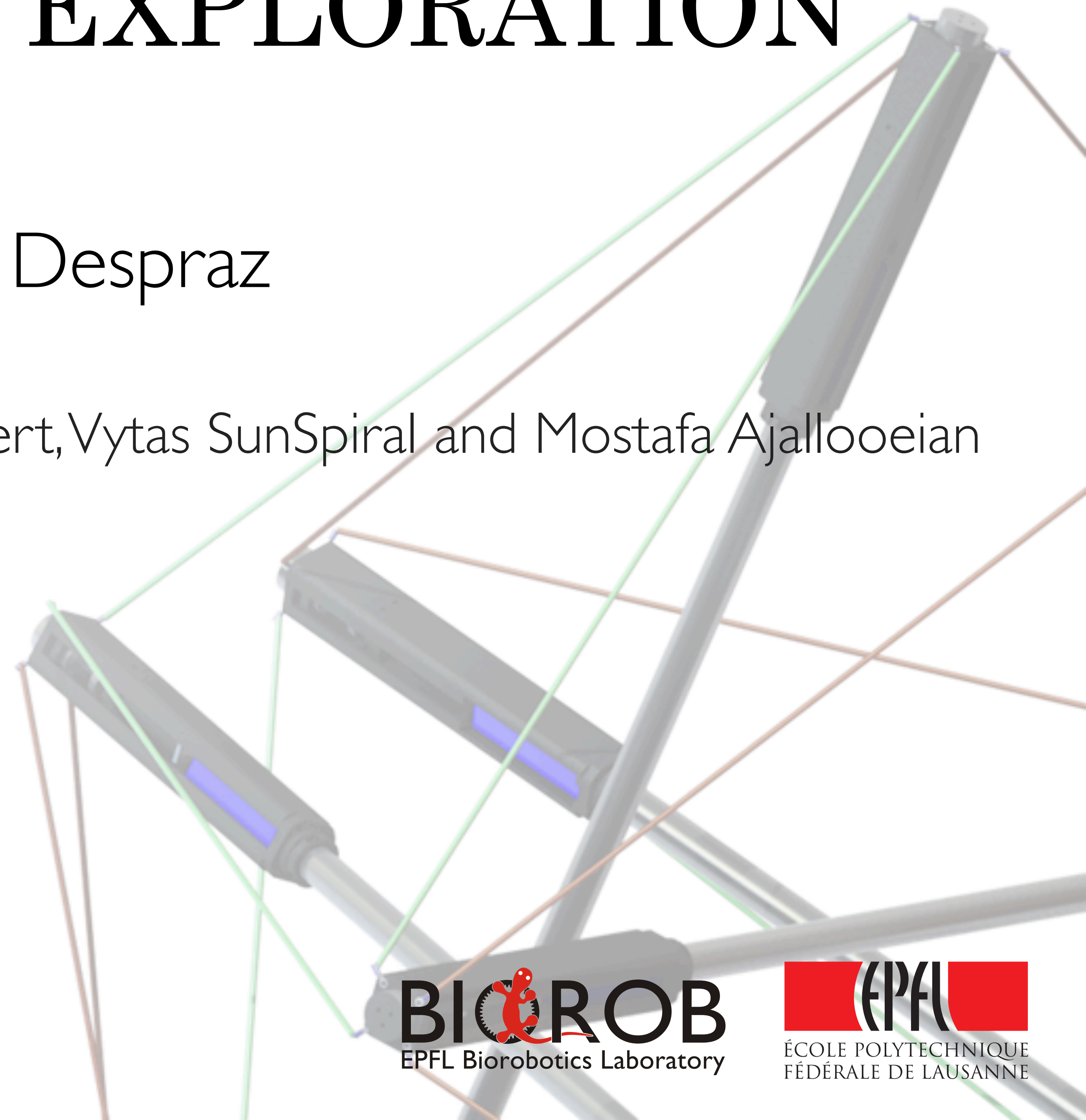
SUPERBALLBOT - STRUCTURES FOR PLANETARY LANDING AND EXPLORATION

Jérémie Despraz

under the supervision of Auke Jan Ijspeert, Vytas SunSpiral and Mostafa Ajallooeian



BIOROB
EPFL Biorobotics Laboratory

ÉCOLE POLYTECHNIQUE FÉDÉRALE DE LAUSANNE

OUTLINE

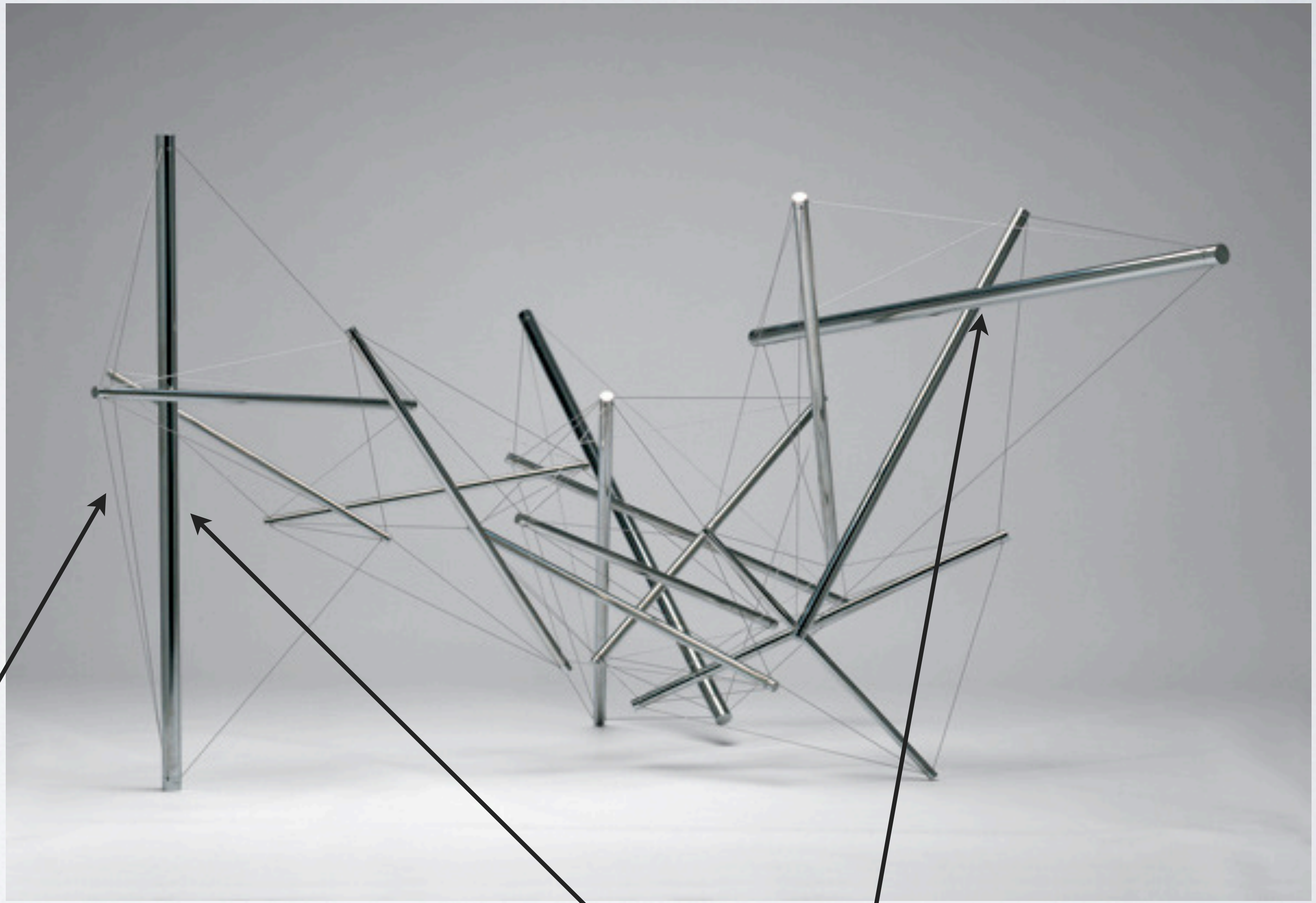
- Introduction to tensegrity structures and their advantages
- Presentation of the tensegrity robot and simulation tools
- Presentation of the developed controllers (reactive - CPG - hybrid)
- Simulator validation with real hardware
- Questions / discussion

TENSEGRITY STRUCTURES

Only pure tension or pure compression!

TENSEGRITY STRUCTURES

Only pure tension or pure compression!

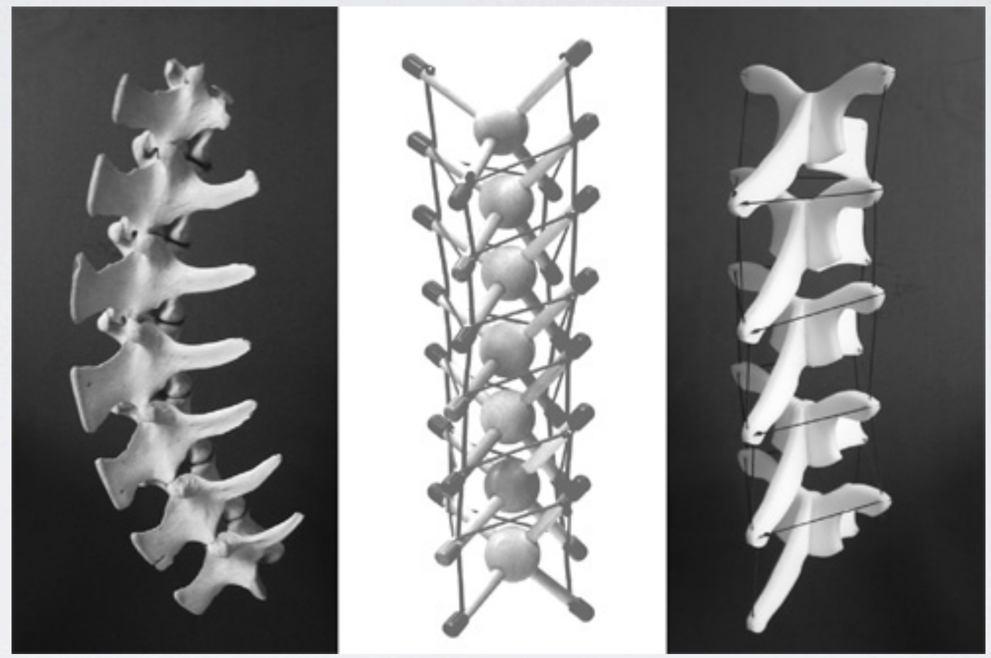


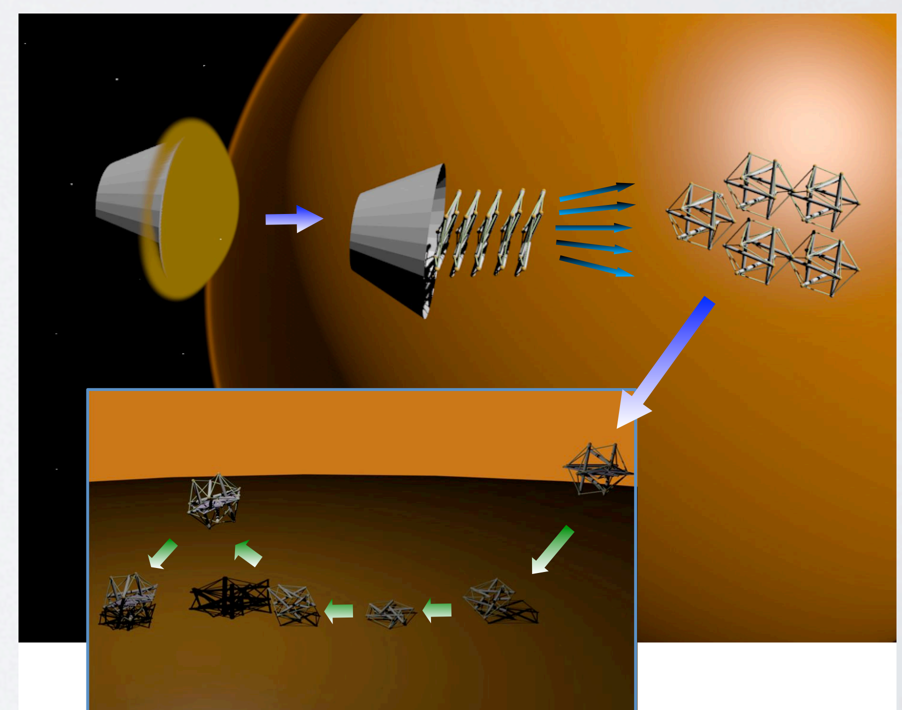
tensile elements

compression elements

TENSEGRITY STRUCTURES

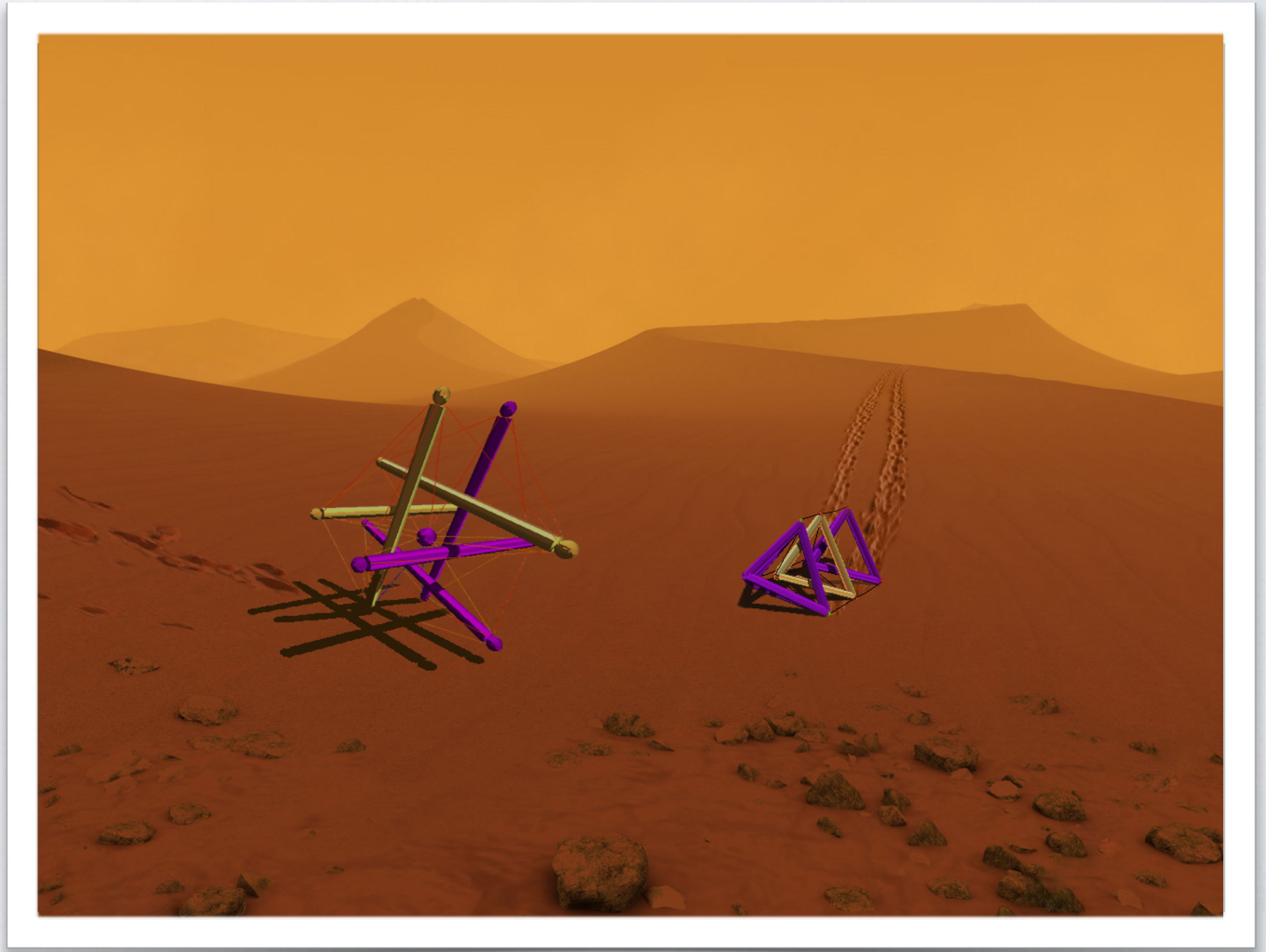
- Lightweight
- Compliant
- Robust to shocks
- Robust to failures
- Deployable
- Bio-inspired





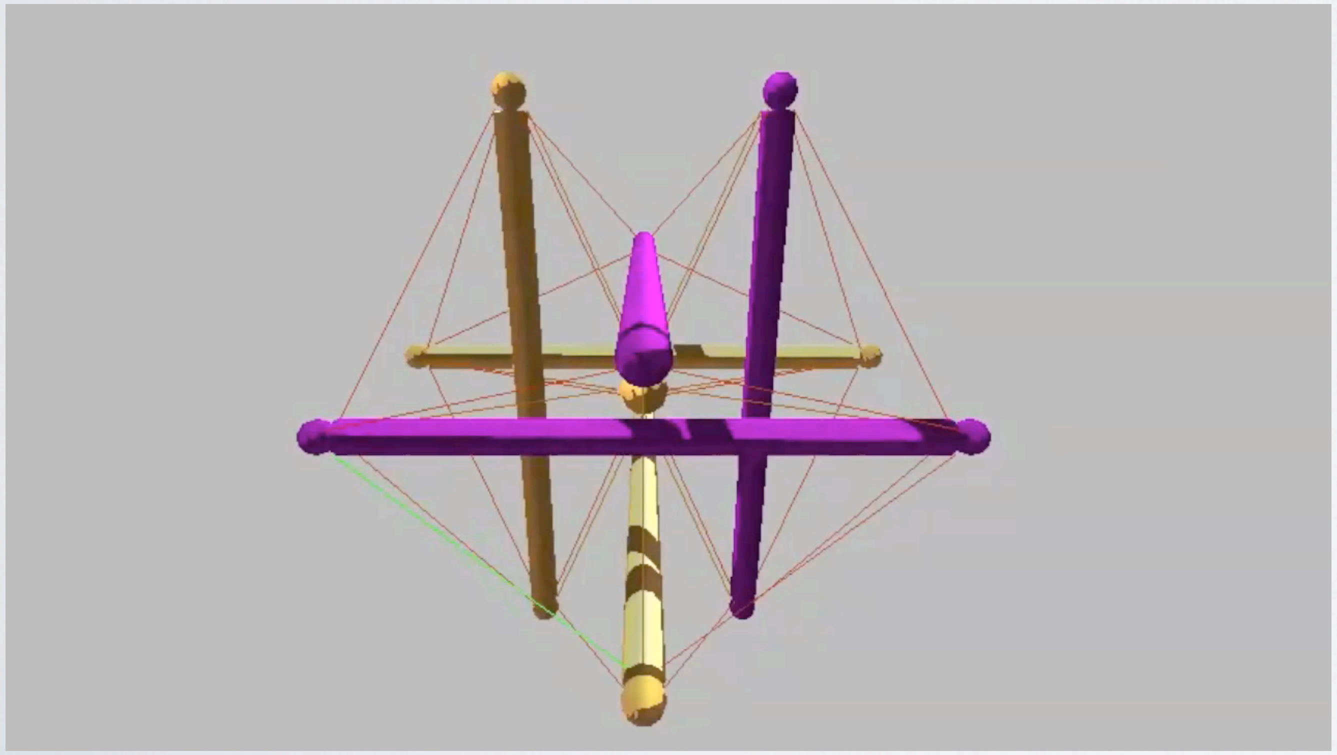
HOW TO MAKE TENSEGRITY ROBOTS MOVE ?

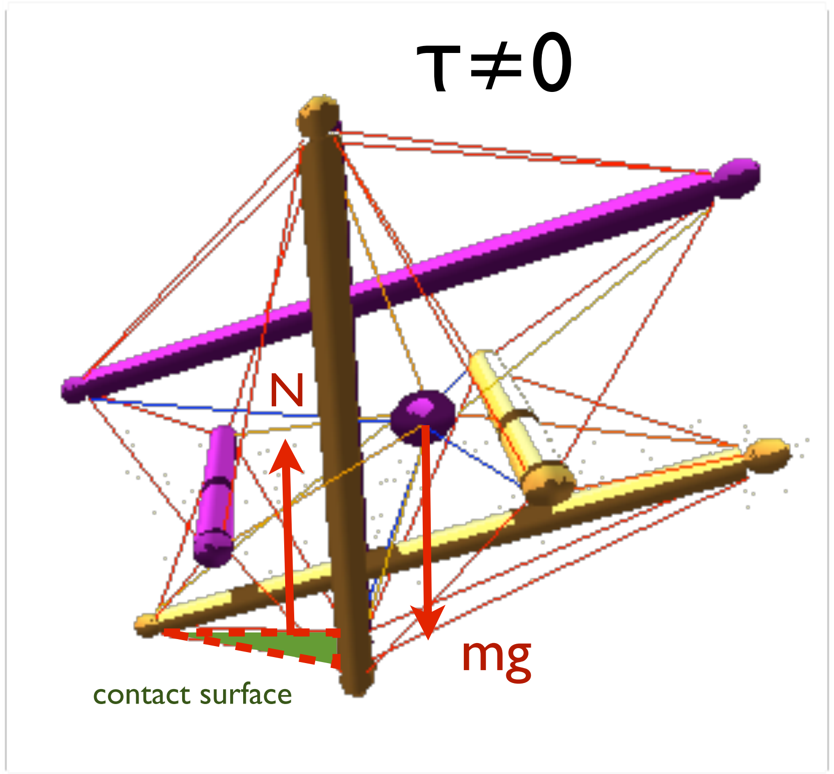
- Not suitable for classical control designs
- High compliance
- Oscillating structure
- Few scientific studies



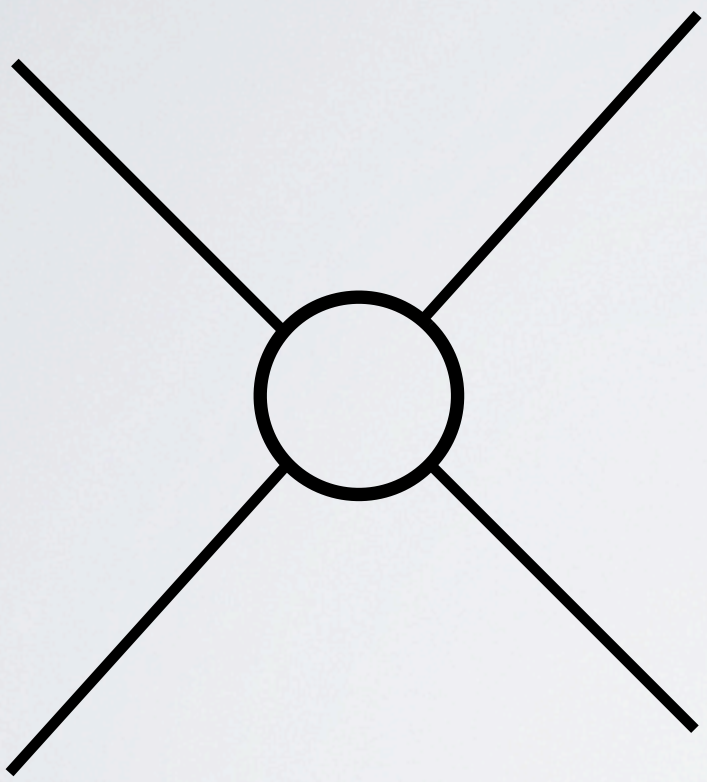
DRIVING PRINCIPLE

- Creation of a torque by
  - moving the center of mass
  - reducing the contact surface

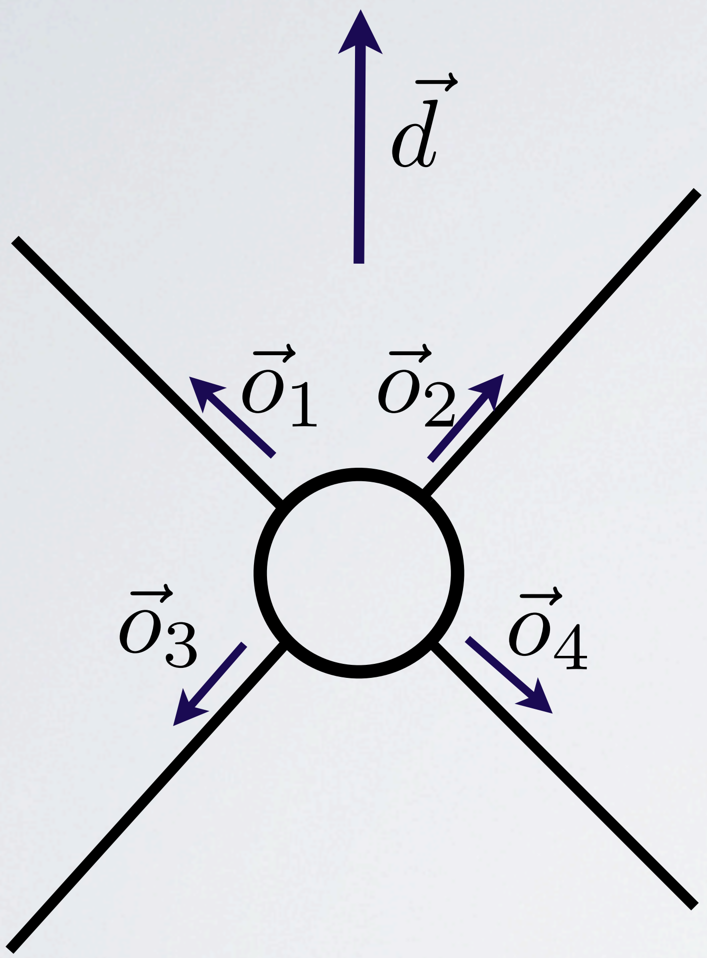




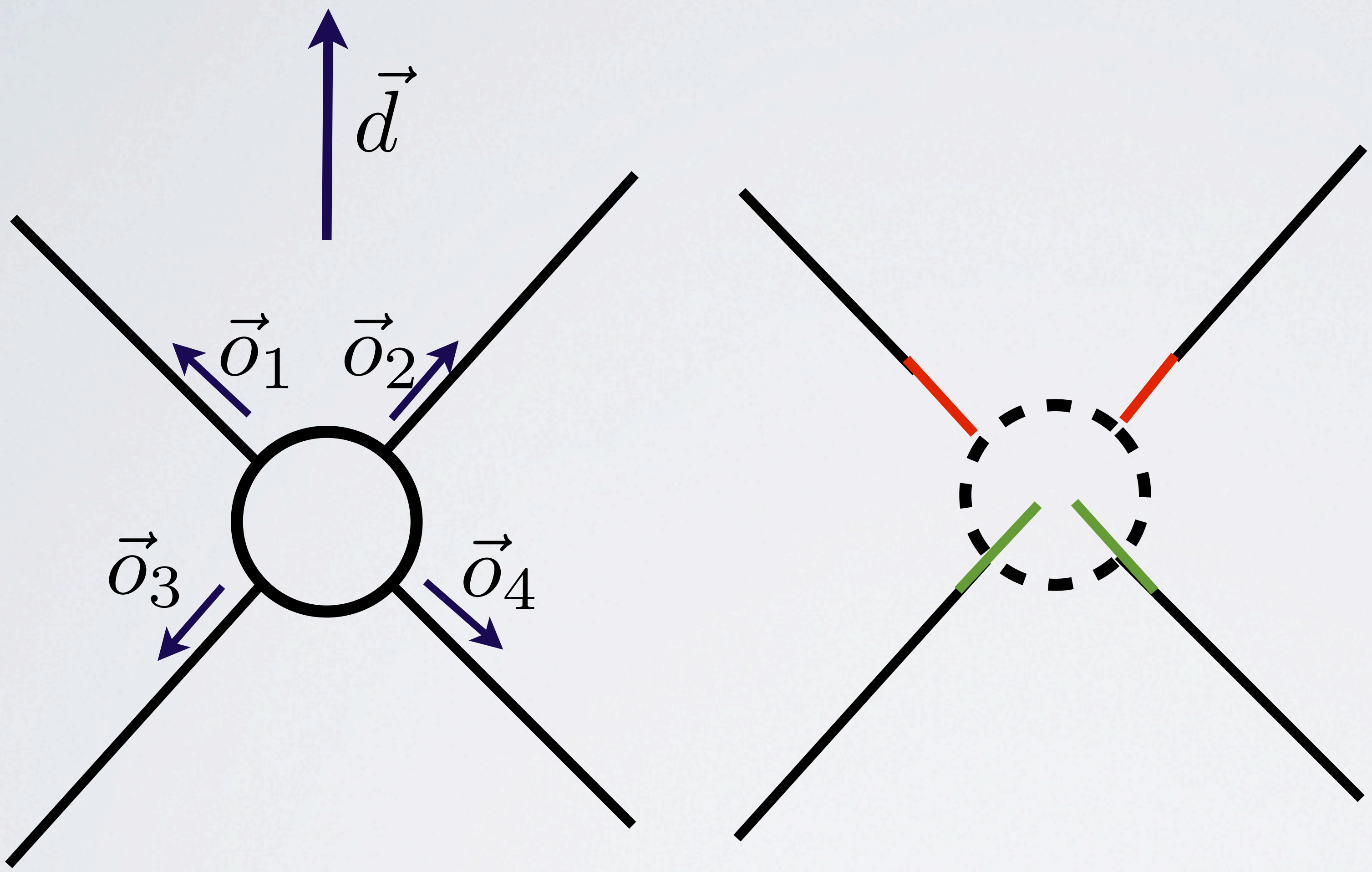
REACTIVE - PRINCIPLE



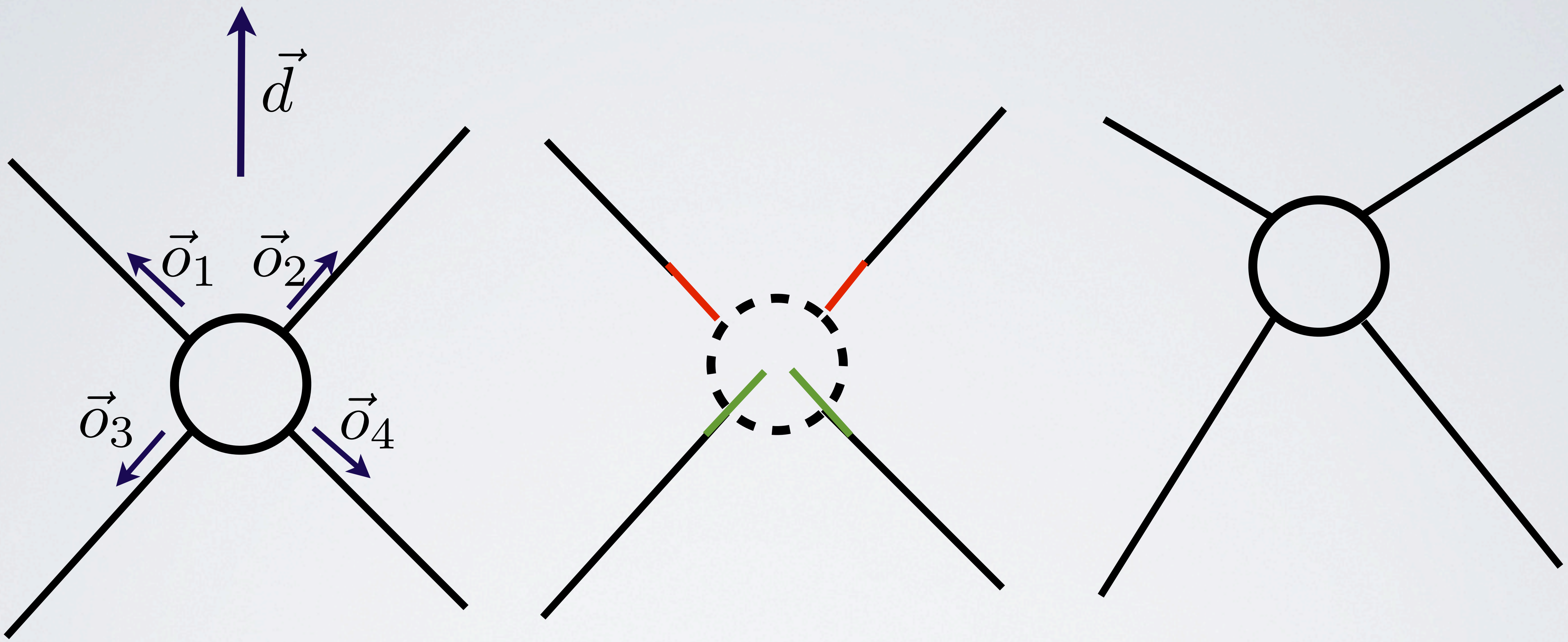
REACTIVE - PRINCIPLE



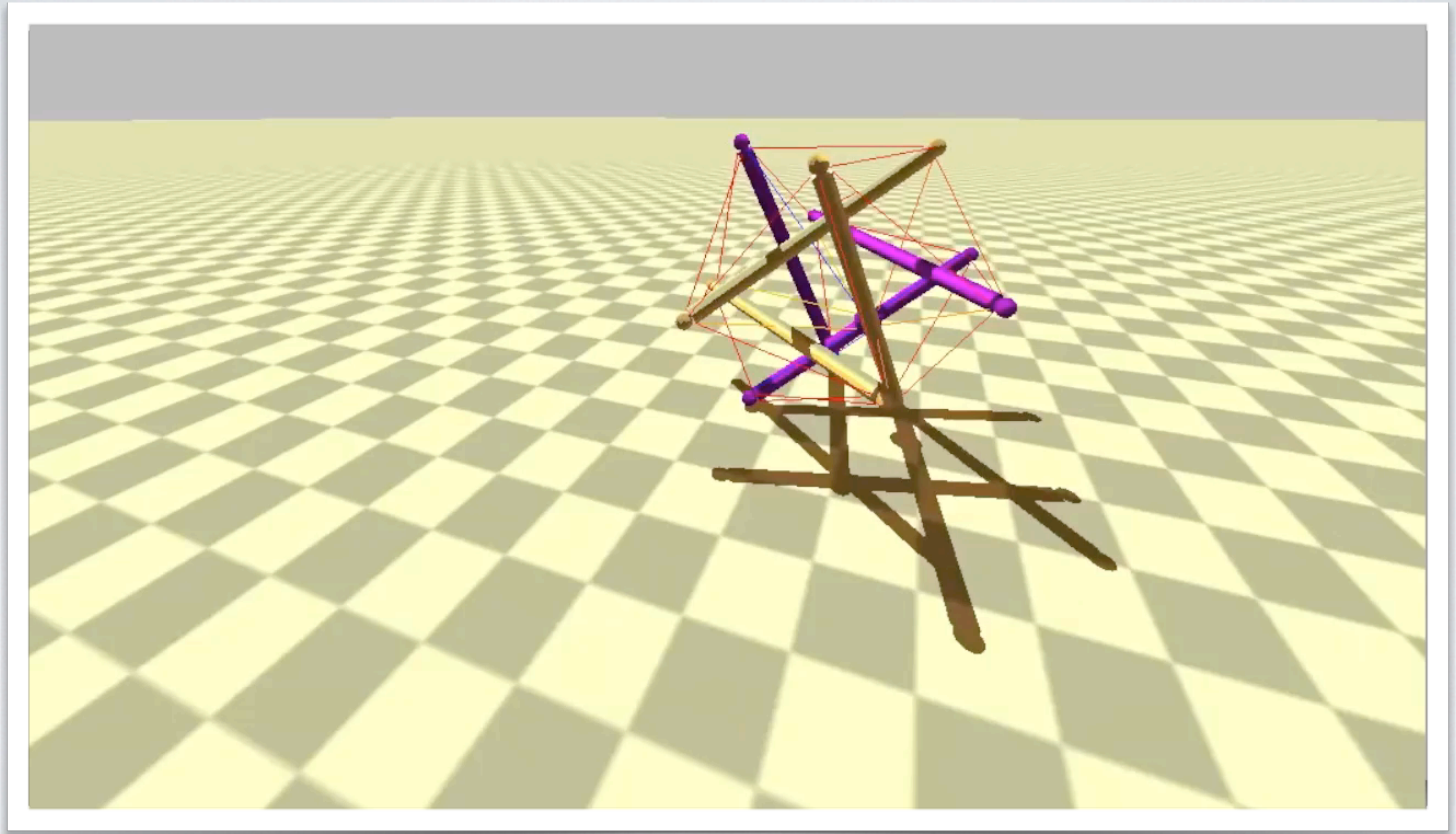
REACTIVE - PRINCIPLE



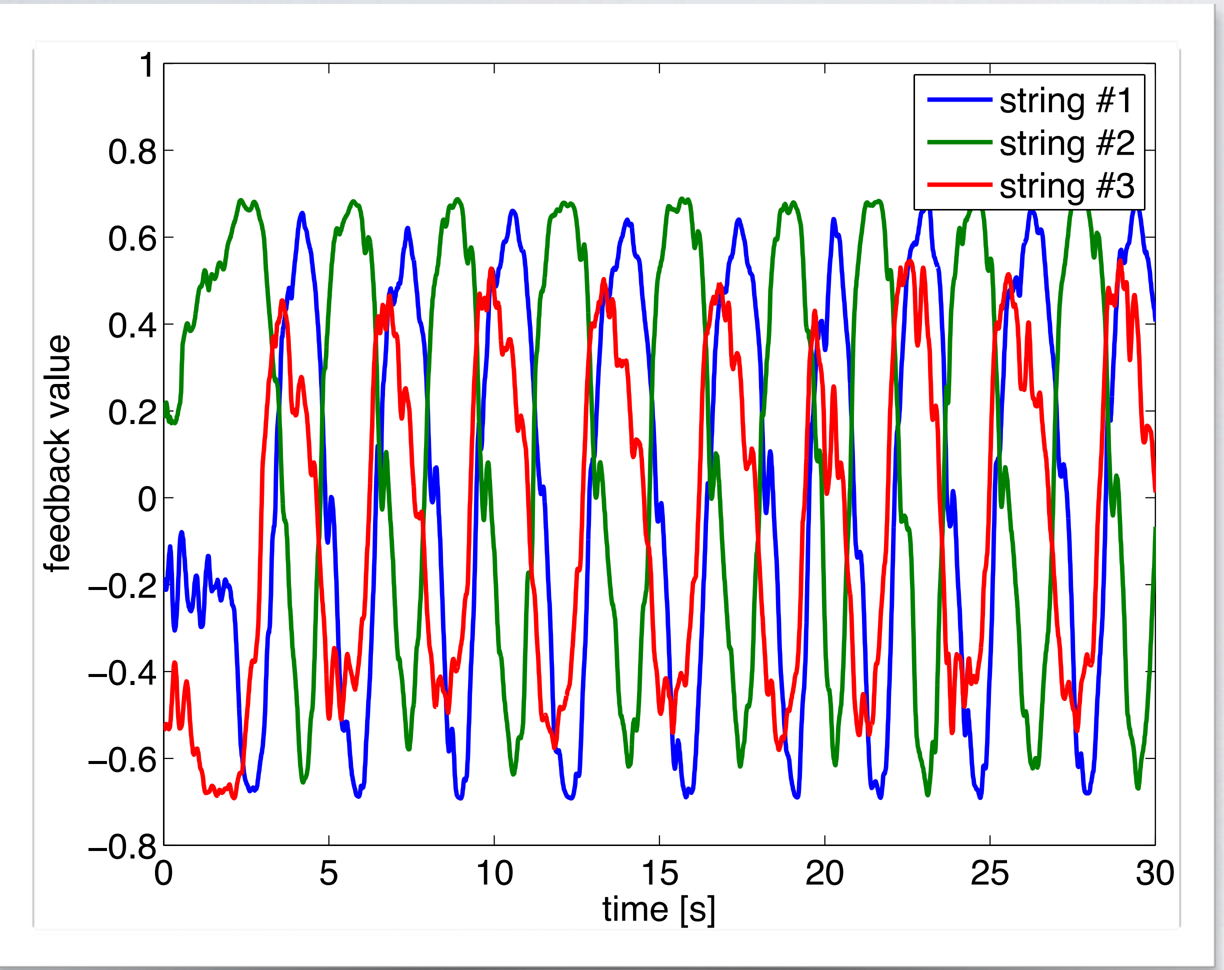
REACTIVE - PRINCIPLE



REACTIVE



TOWARDS CPGS



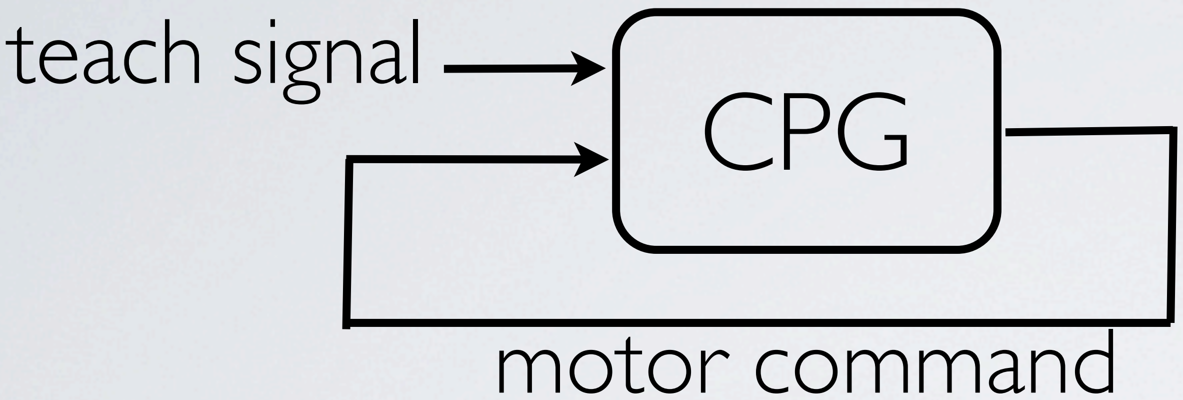
The feedback signal (value of the scalar product) is very regular over time

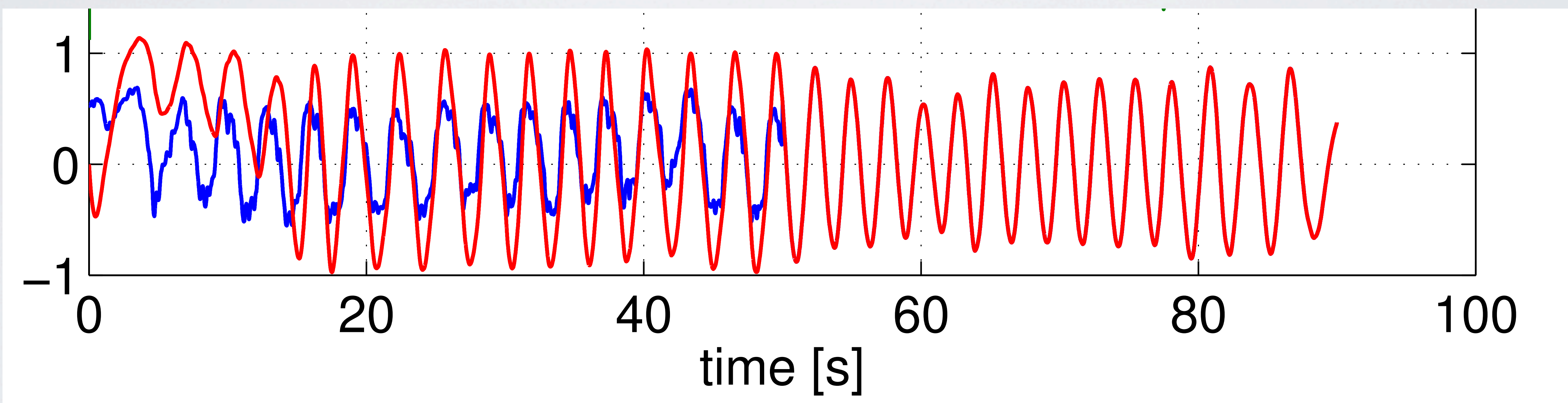
It can be stored in a dynamical system

$$\dot{x} = \gamma(\mu - (x^2 + y^2))x - \omega y + \epsilon f(t)$$
$$\dot{y} = \gamma(\mu - (x^2 + y^2))y + \omega x$$
$$\dot{\omega} = -\epsilon f(t) \frac{y}{x^2 + y^2}$$

CENTRAL PATTERN GENERATORS

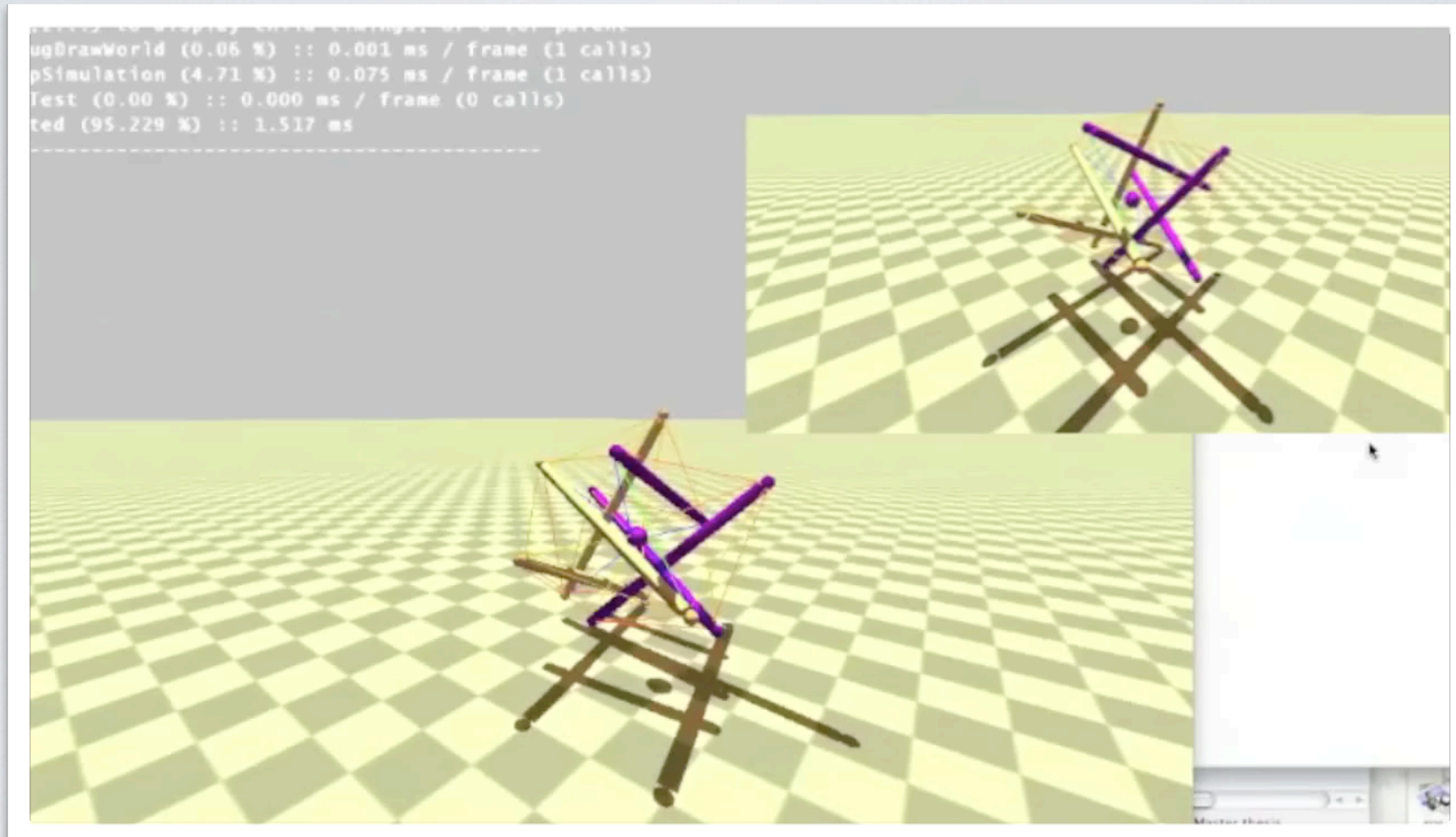
- Learning phase:





- Controlling phase:

CENTRAL PATTERN GENERATORS



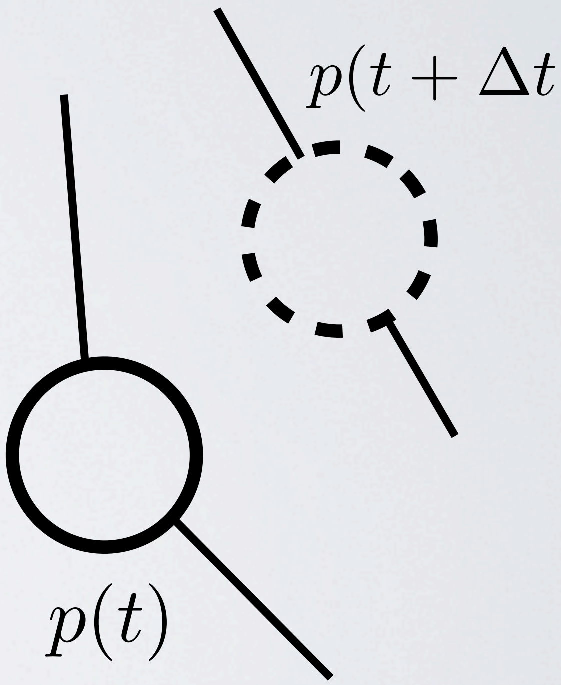
INVERSE KINEMATICS

complicated closed form approximated by numerical methods

$$\vec{p} = \vec{p}_0 + \vec{p}_1 \Delta\ell + \vec{p}_2 \Delta\ell^2 + \ldots$$

Numerical method*

$$\Delta\ell = f(\vec{p})$$



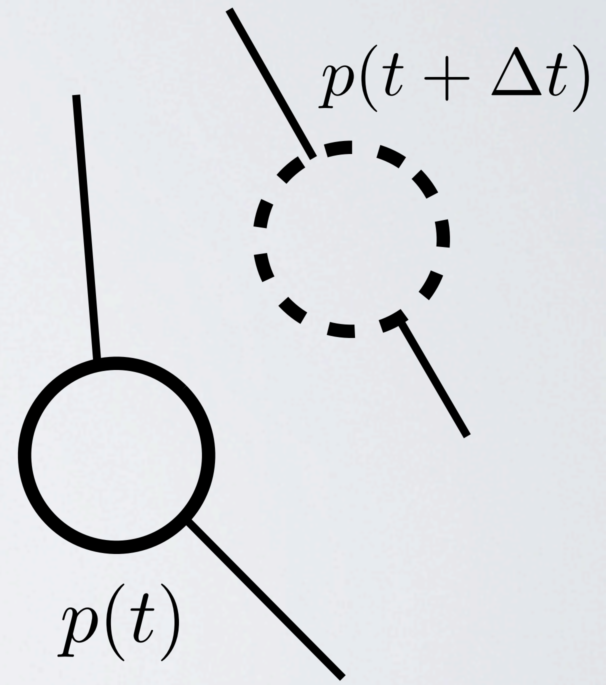
INVERSE KINEMATICS

complicated closed form approximated by numerical methods

$$\vec{p} = \vec{p}_0 + \vec{p}_1 \Delta\ell + \vec{p}_2 \Delta\ell^2 + \ldots$$

Numerical method*

$$\Delta\ell = f(\vec{p})$$

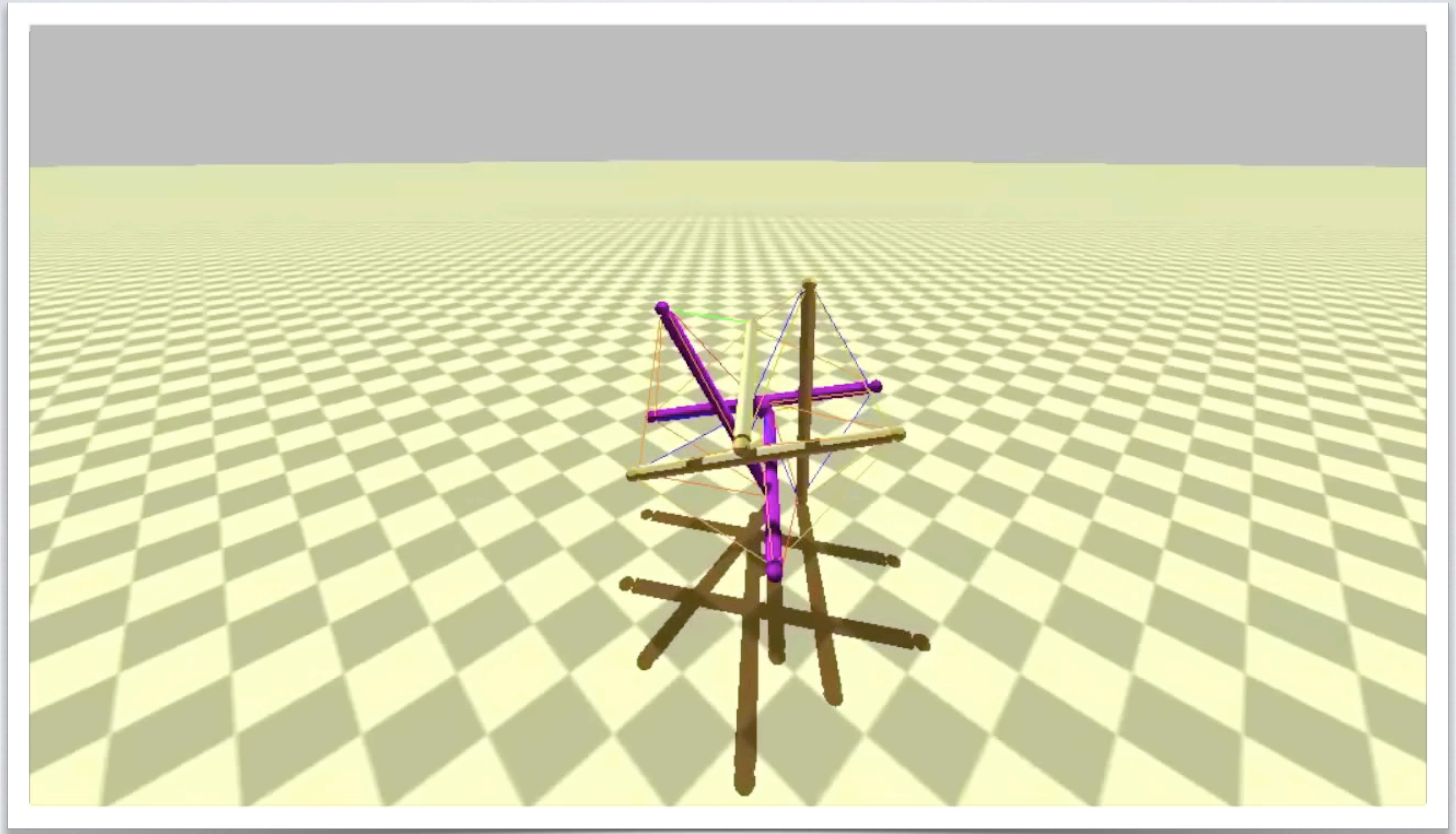


*

- First order: Transpose Jacobian Method

- Second order: Newton Method

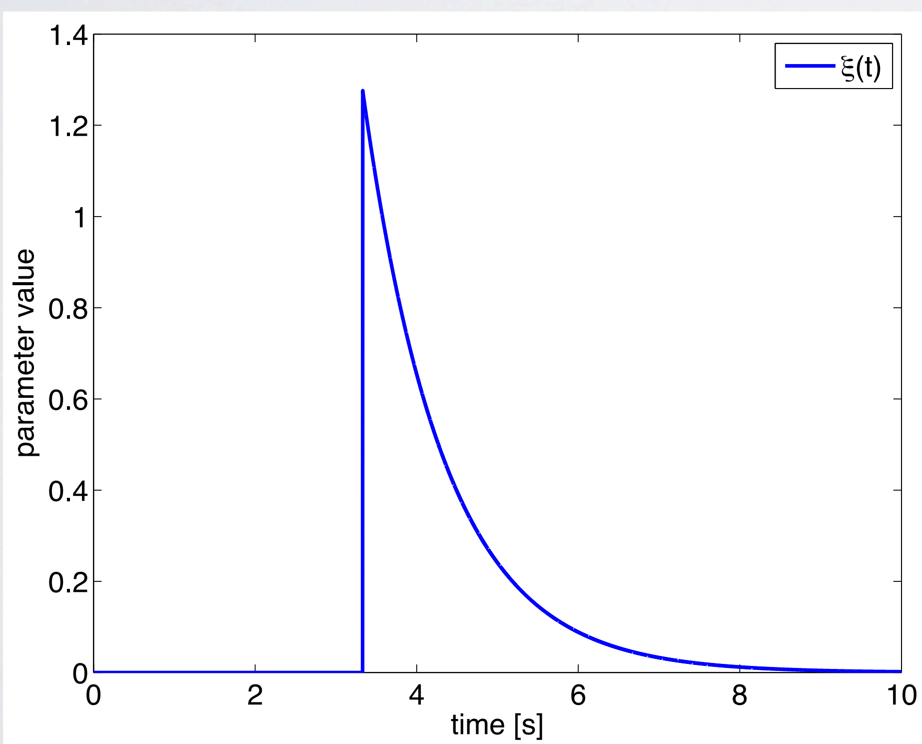
INVERSE KINEMATICS

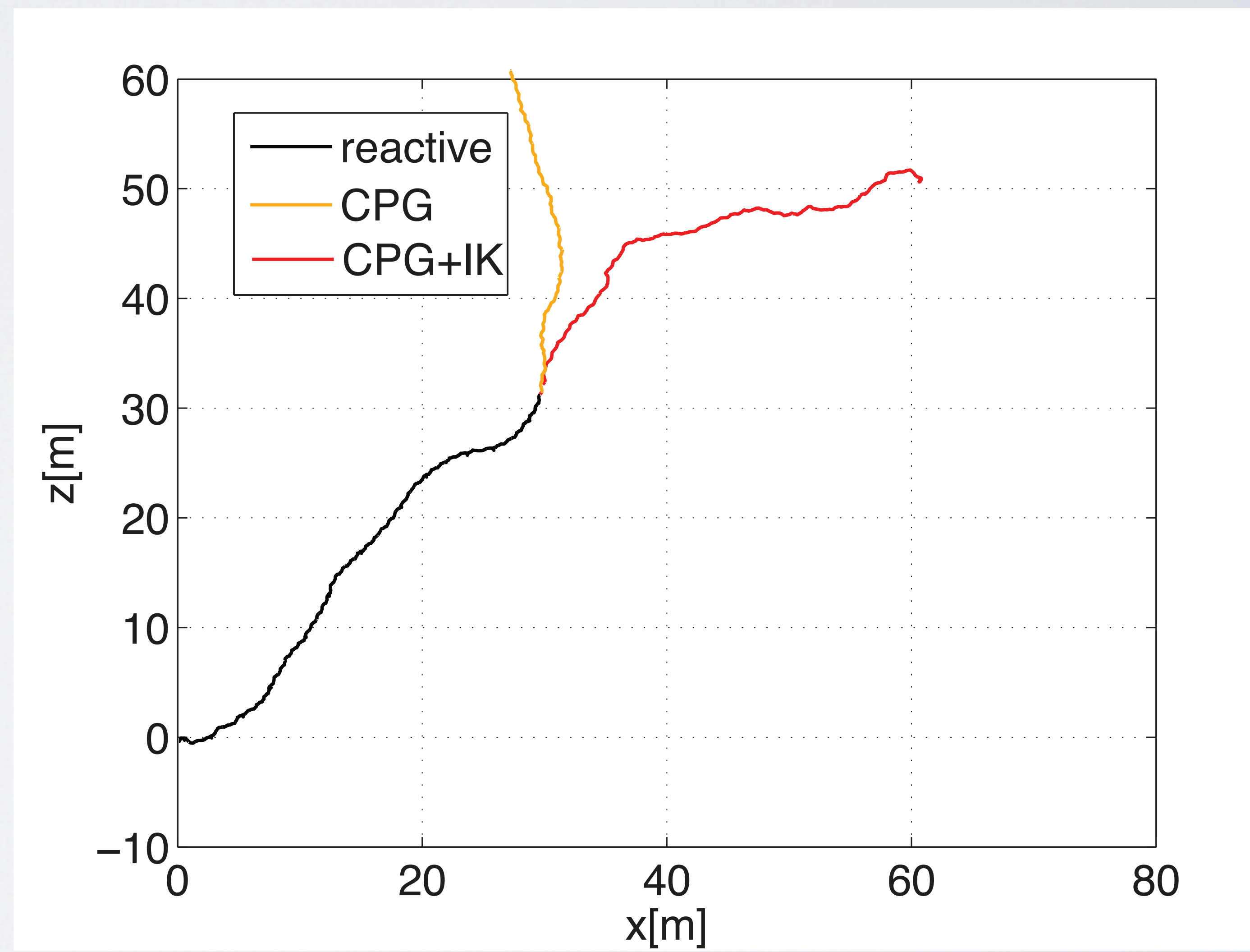


HYBRID

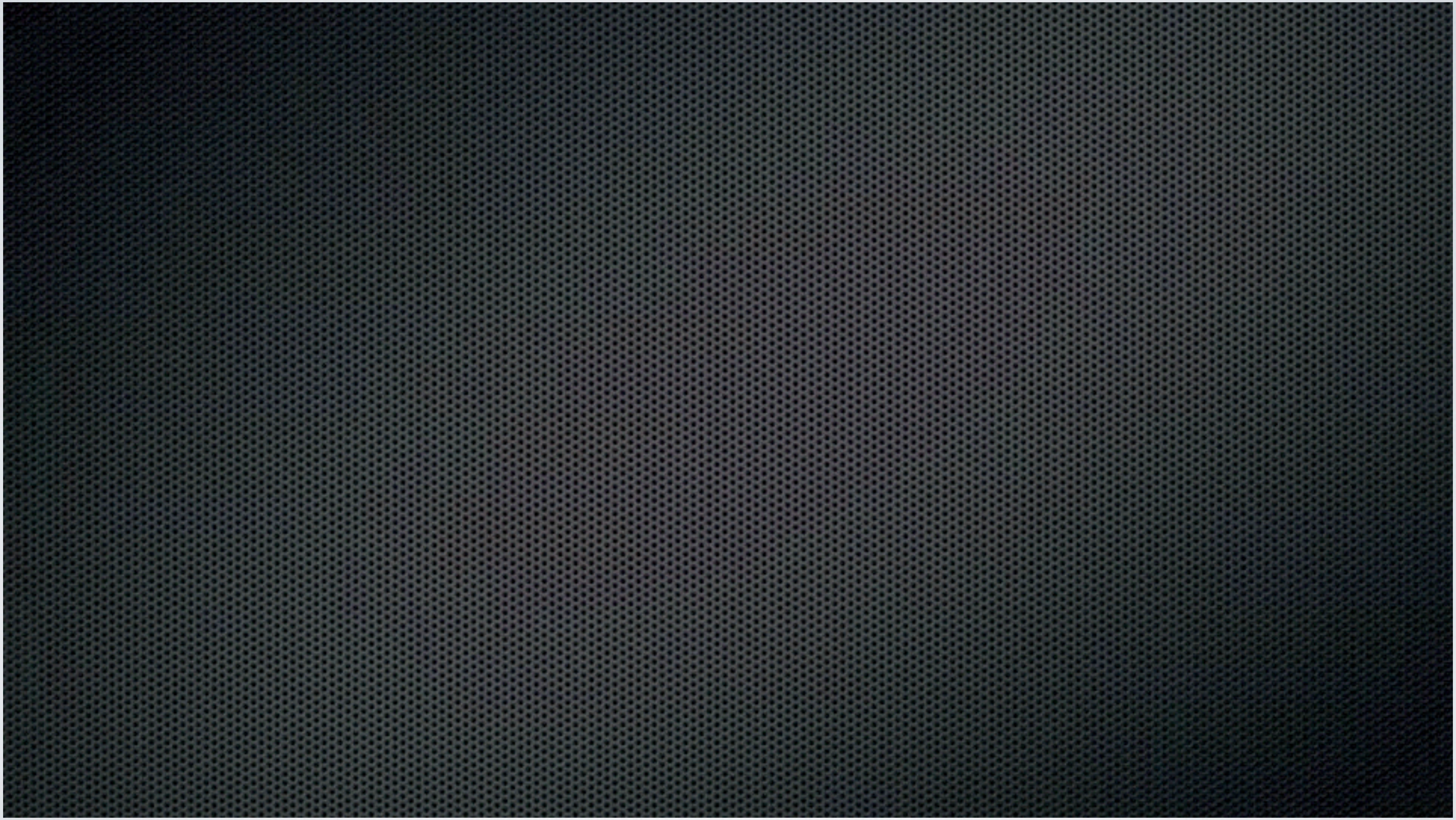
- Controlling phase:

fading memory property:

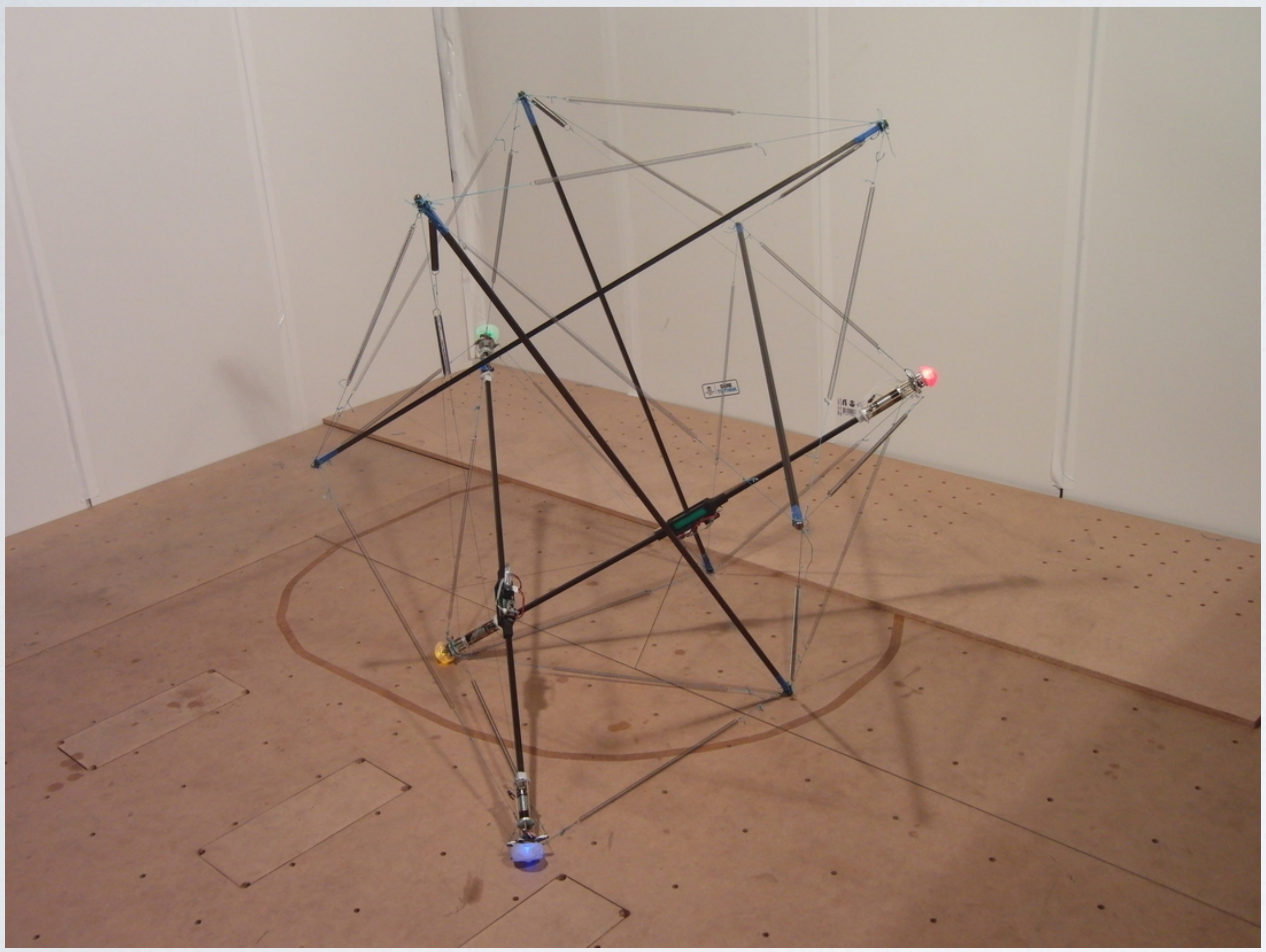




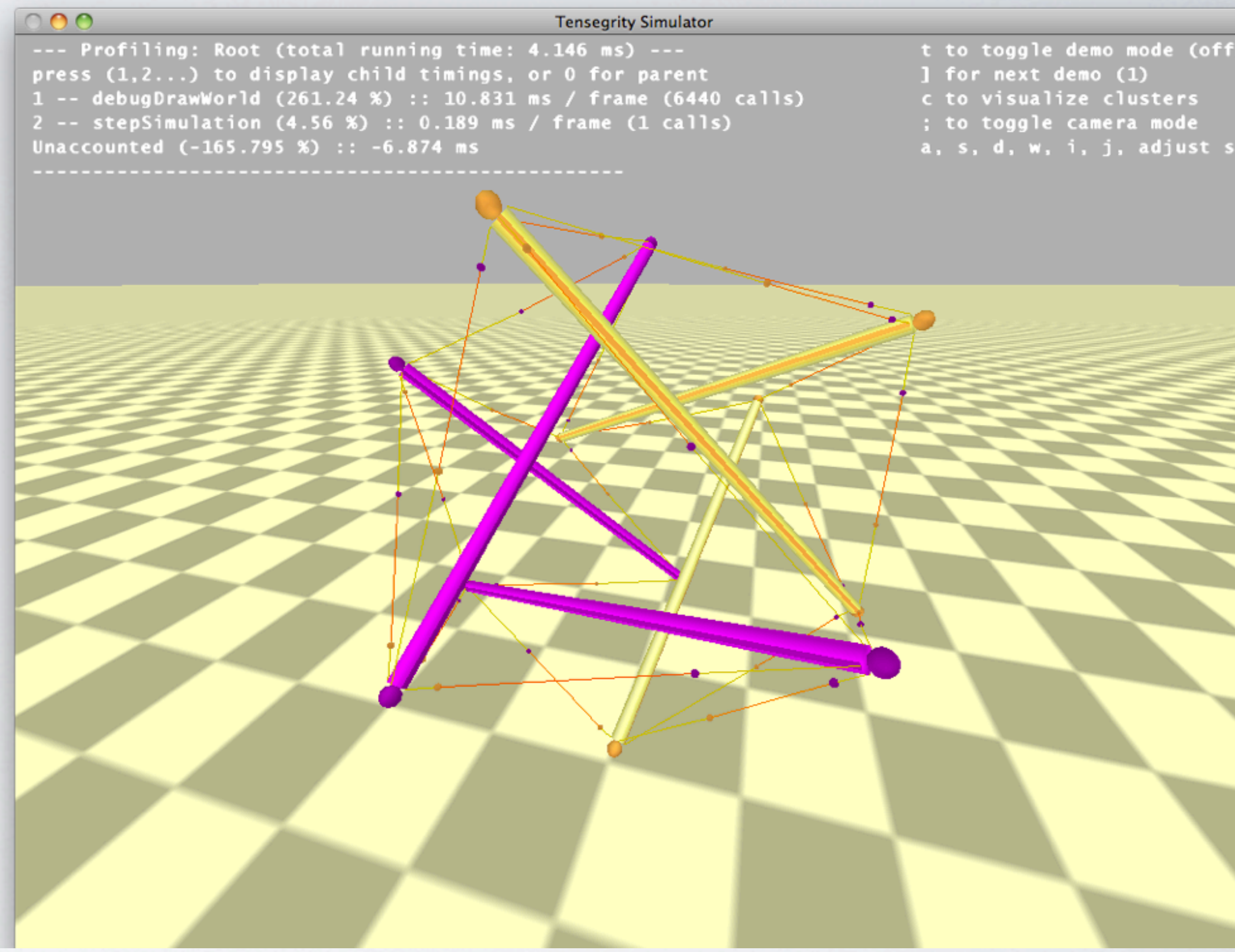
RESULTS SUMMARY



SIMULATOR VALIDATION



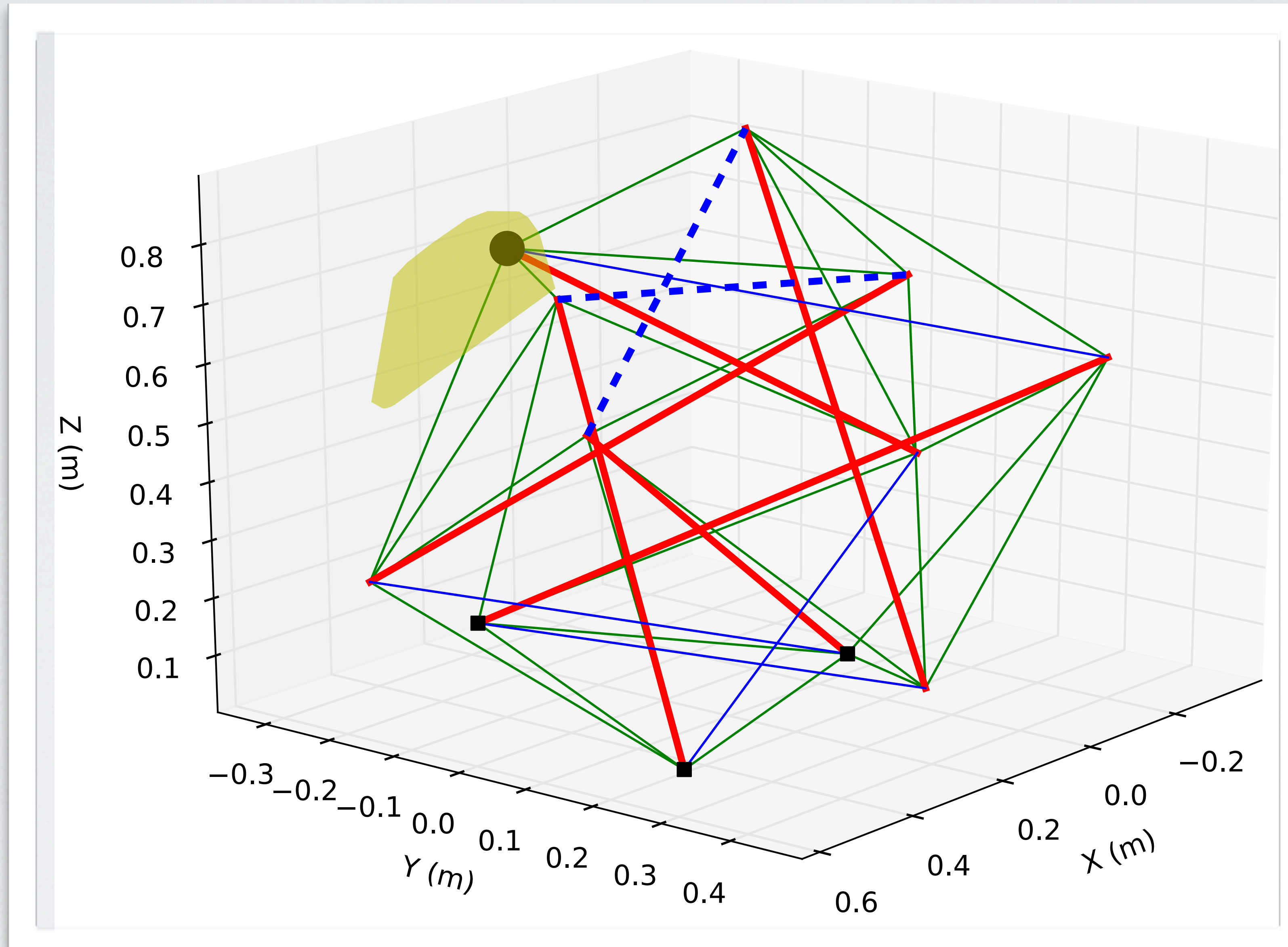
Reservoir compliant tensegrity robot (ReCTeR)



Nasa tensegrity robotics toolkit (NTRT)

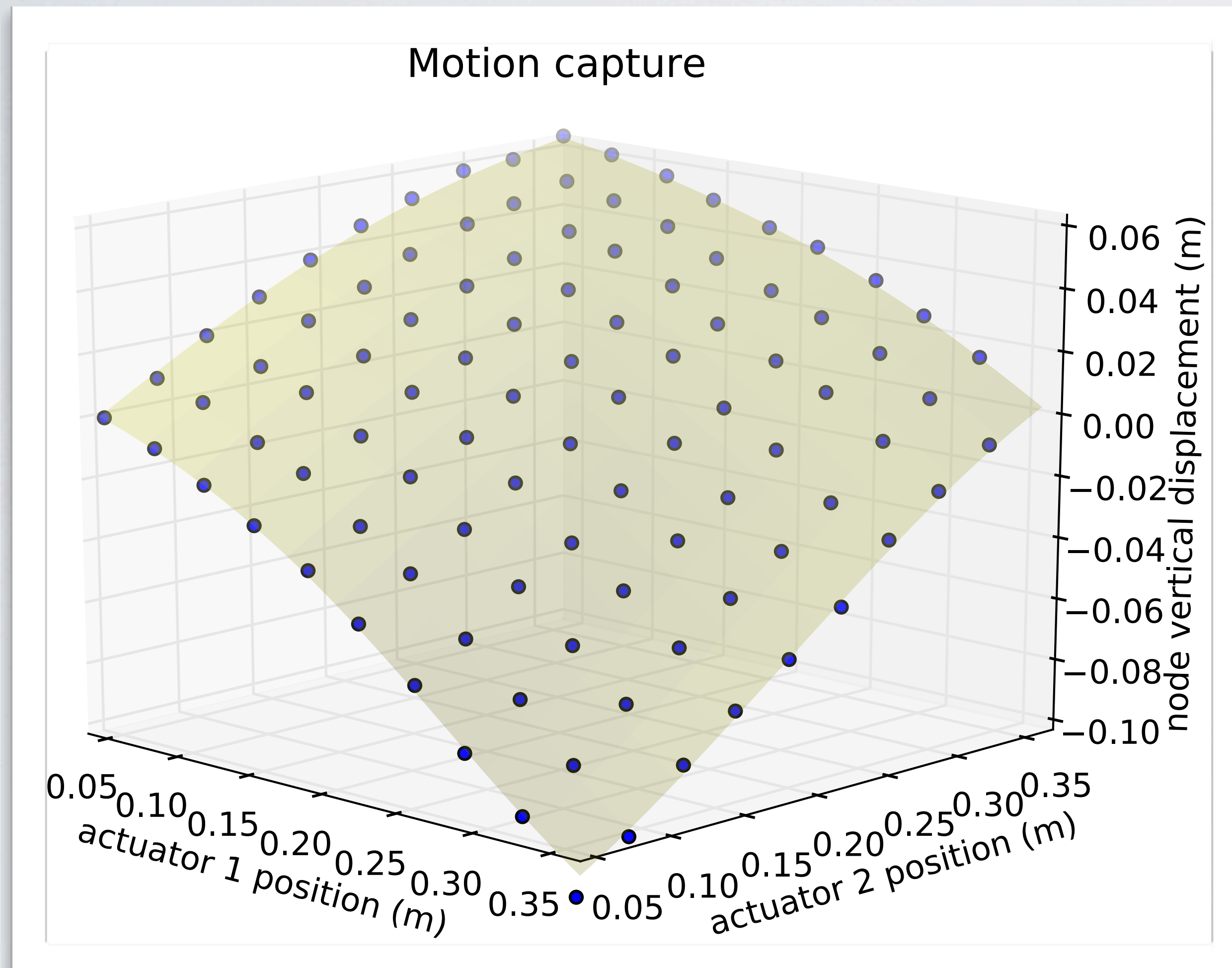
SIMULATOR VALIDATION

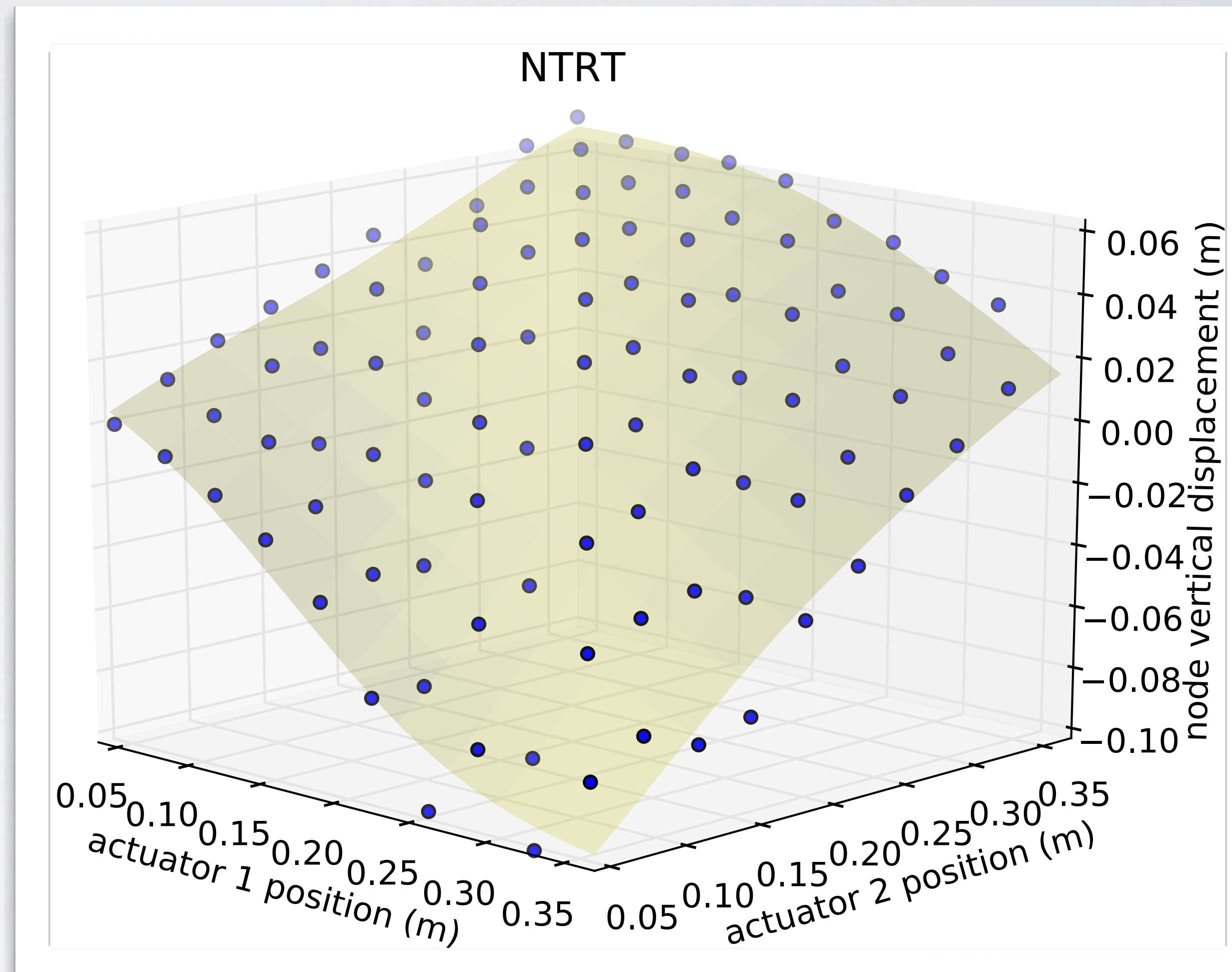
Static



SIMULATOR VALIDATION

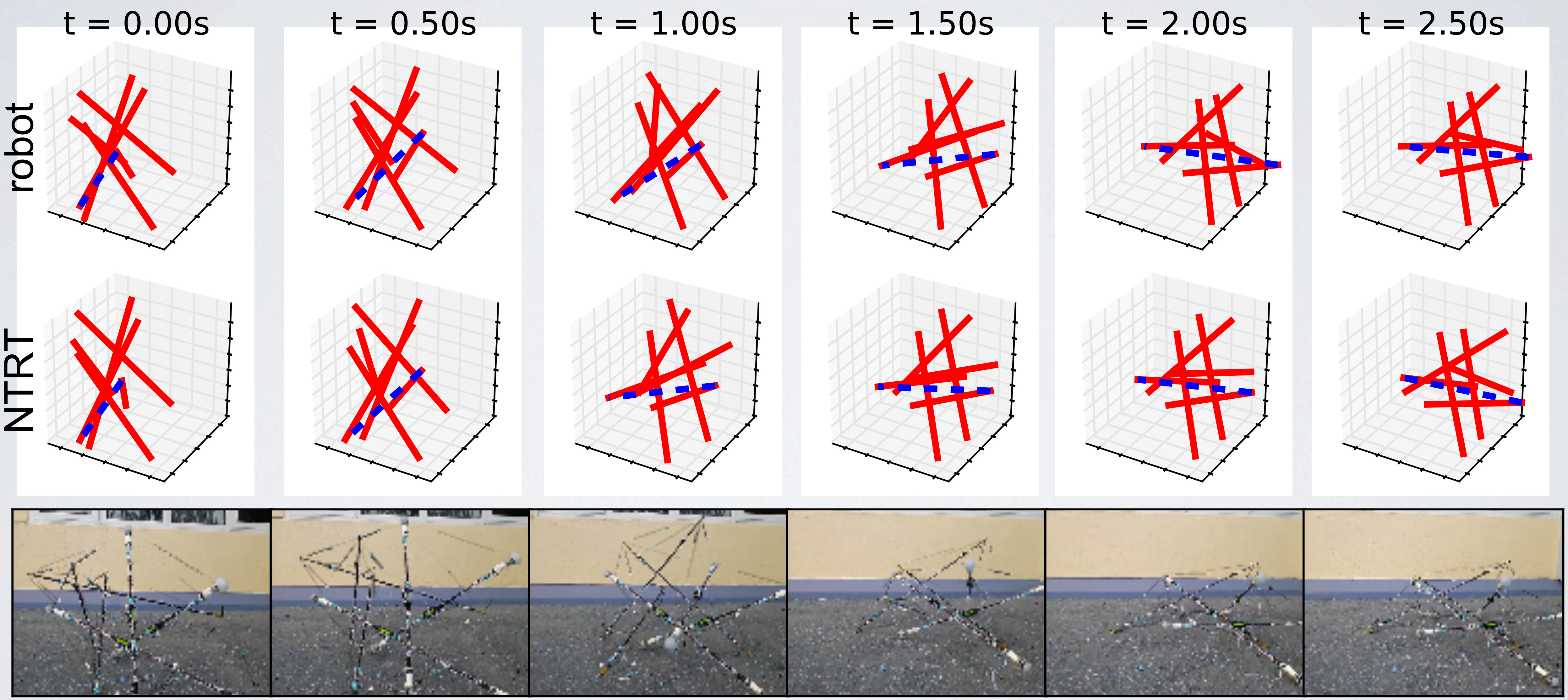
Static





SIMULATOR VALIDATION

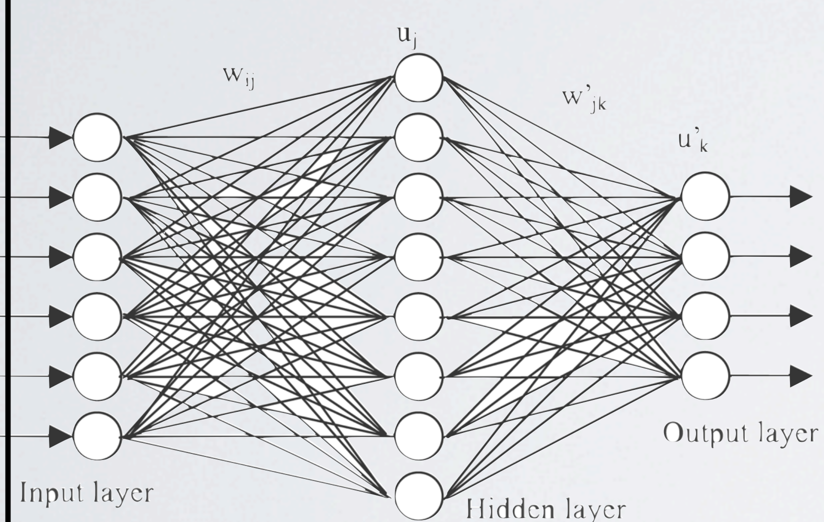
Dynamics



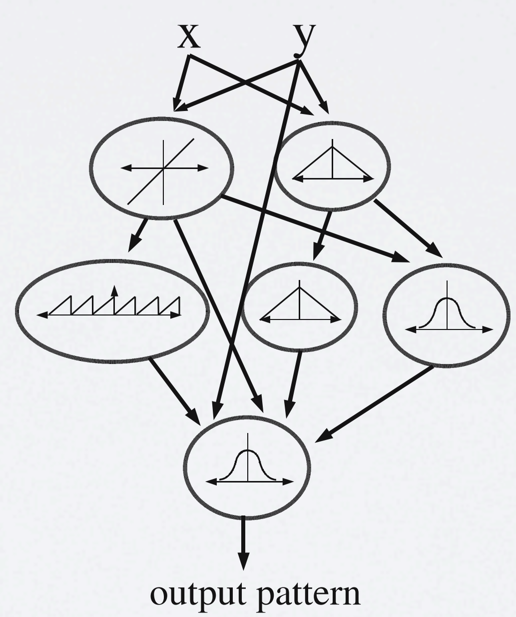
FUTURE WORK

Several promising options

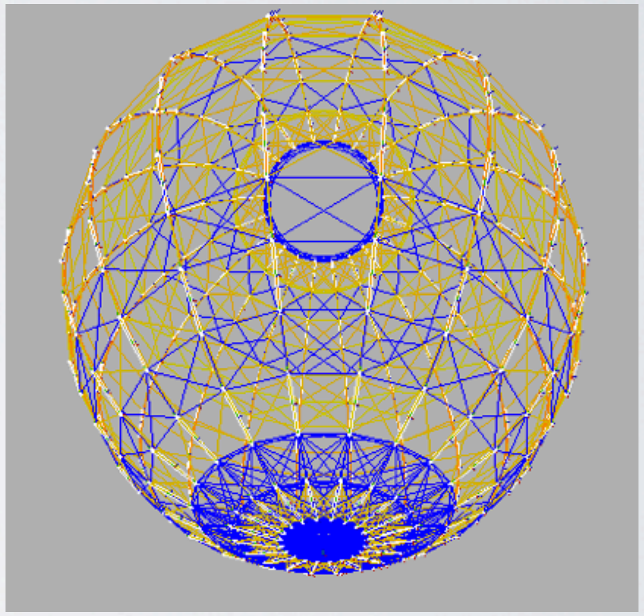
Neural Networks



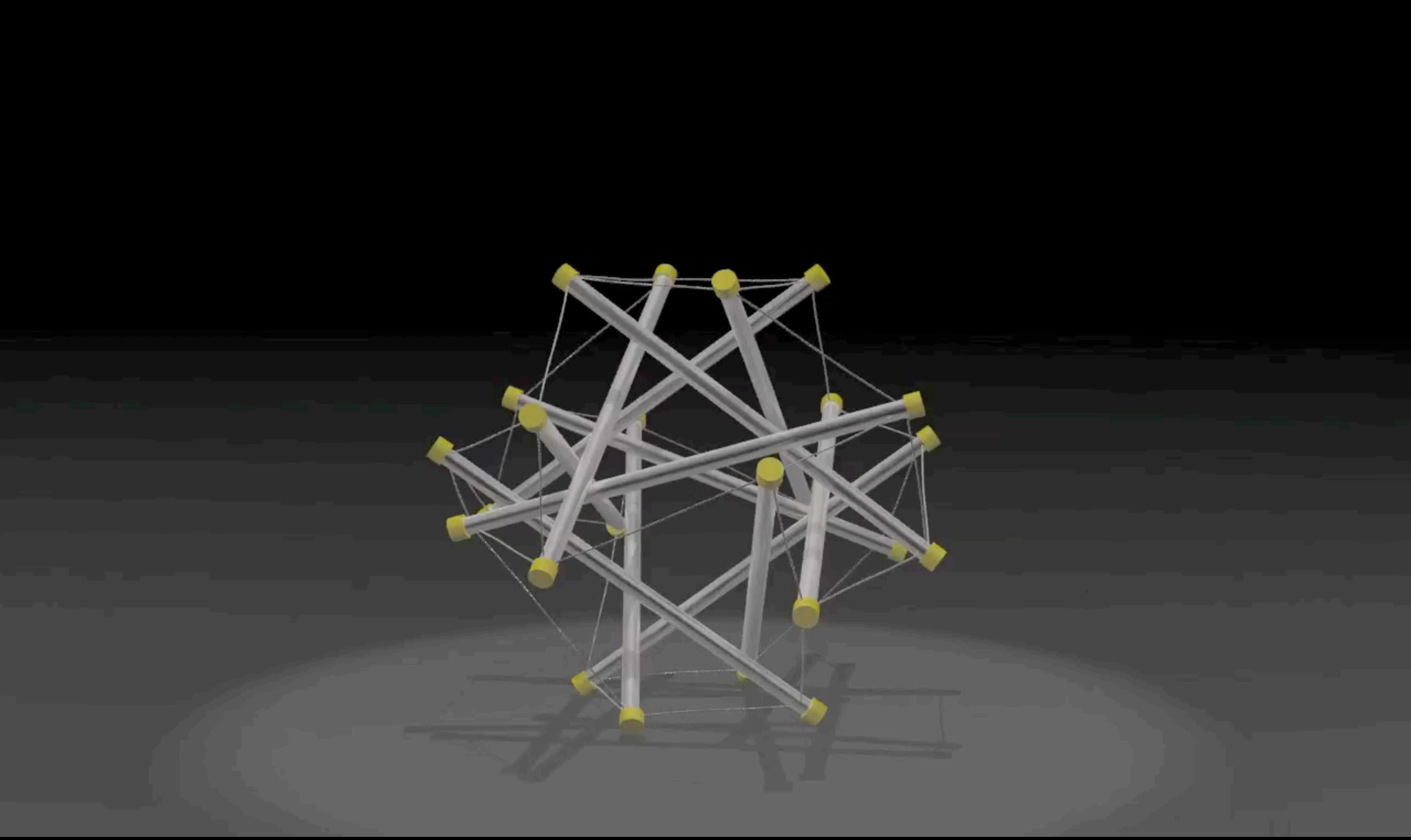
Evolutionary Algorithms



Soft Robotics



(video from University of Idaho Tensegrity Group)

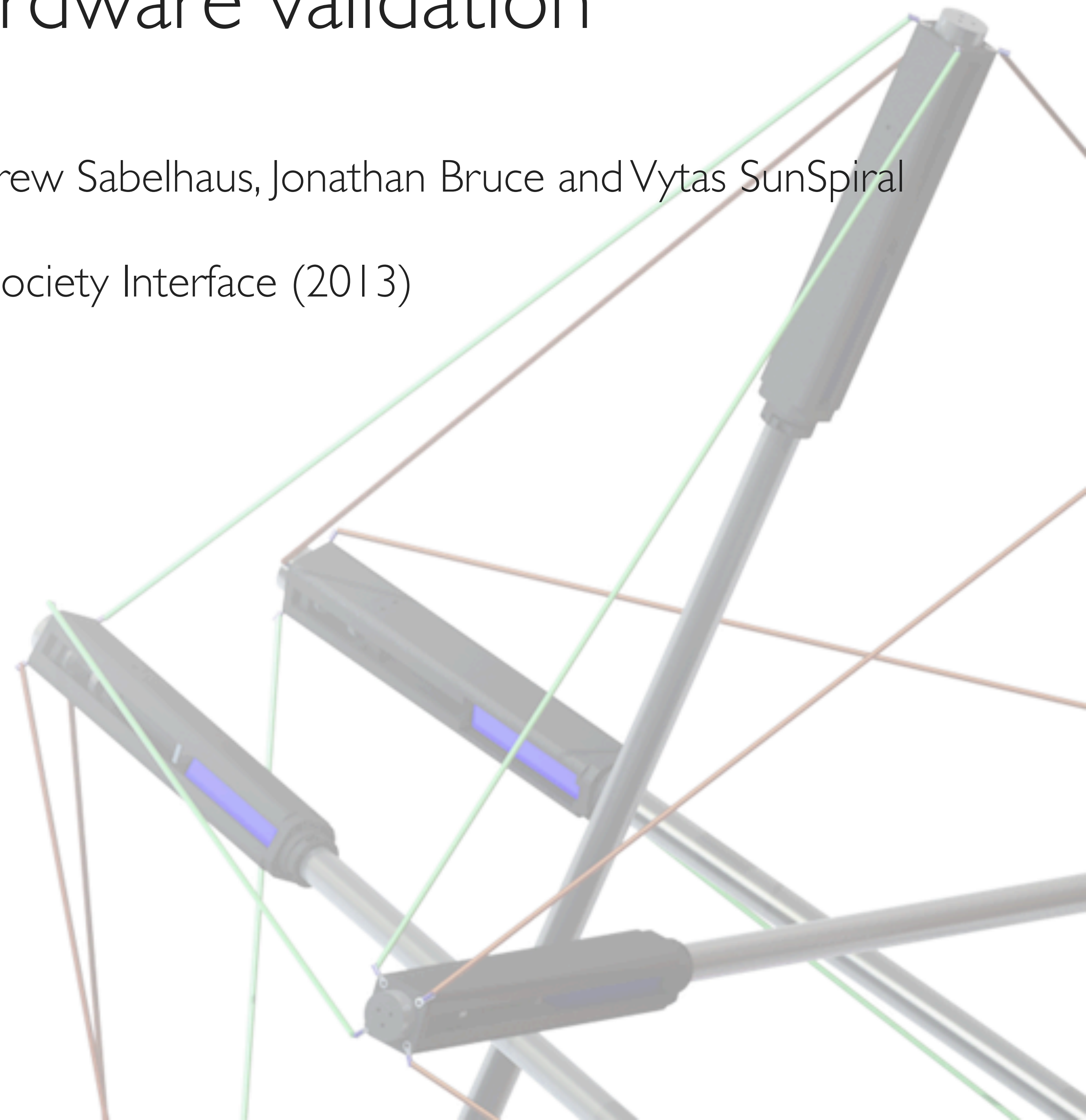


RELATED PUBLICATION

"Design and control of compliant tensegrity robots through simulation and hardware validation"

Ken Caluwaerts, Jérémie Despraz, Atil Iscen, Andrew Sabelhaus, Jonathan Bruce and Vytas SunSpiral

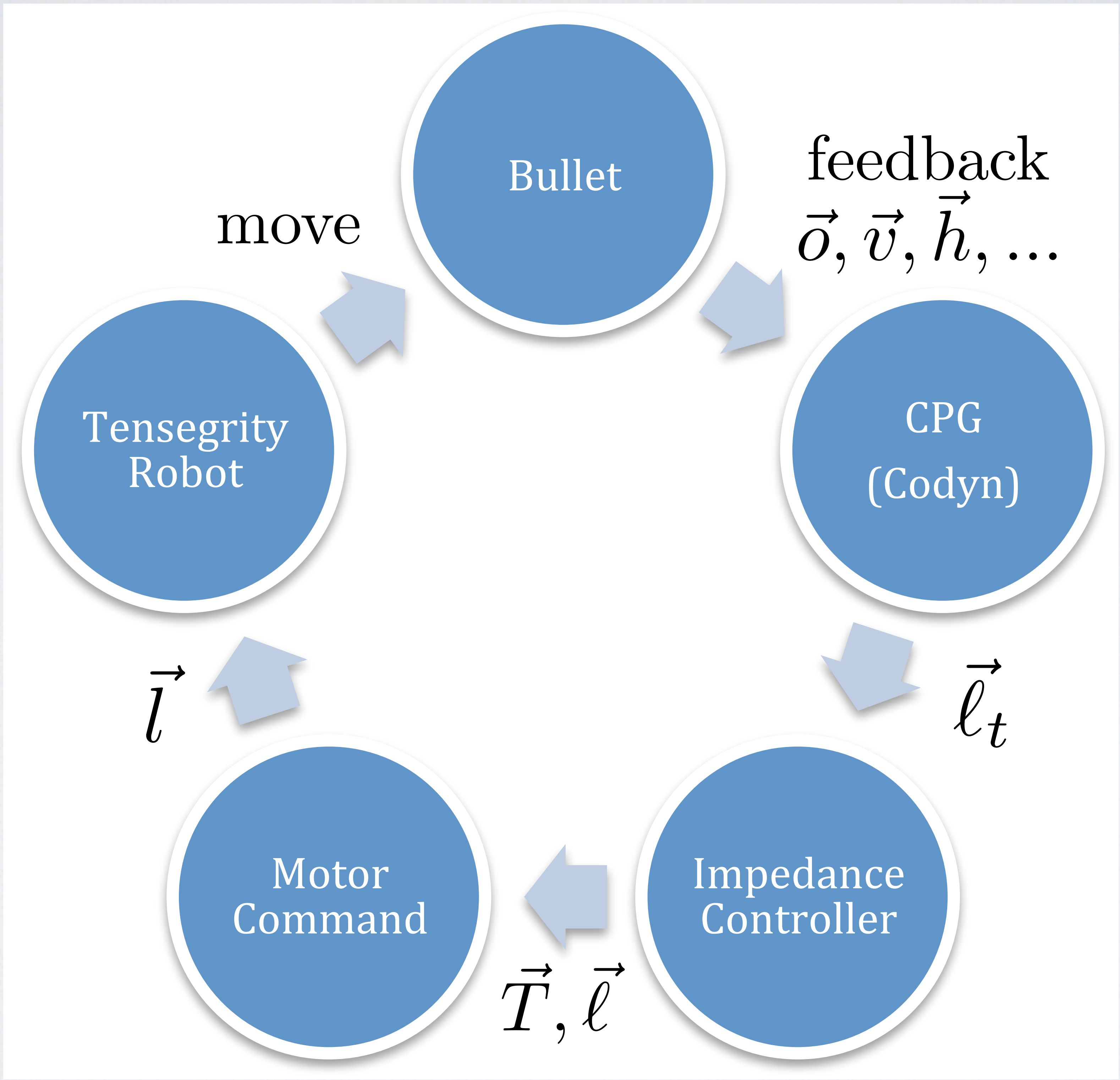
Journal of the Royal Society Interface (2013)



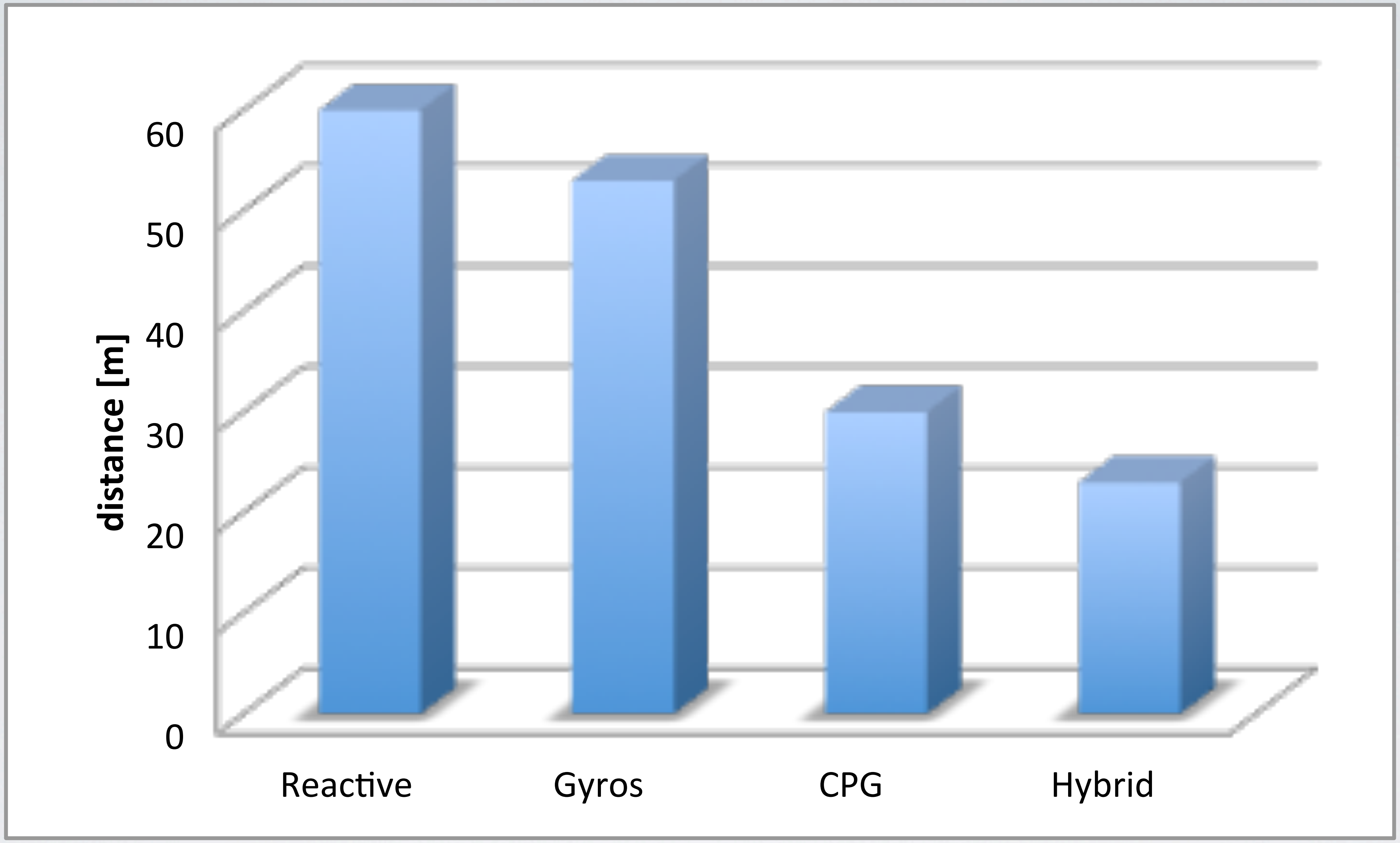
Thank you!

Vytas SunSpiral
Auke Jan Ijspeert
Mostafa Ajallooeian
Jesse van den Kieboom
Ken Caluwaerts
Atil Iscen

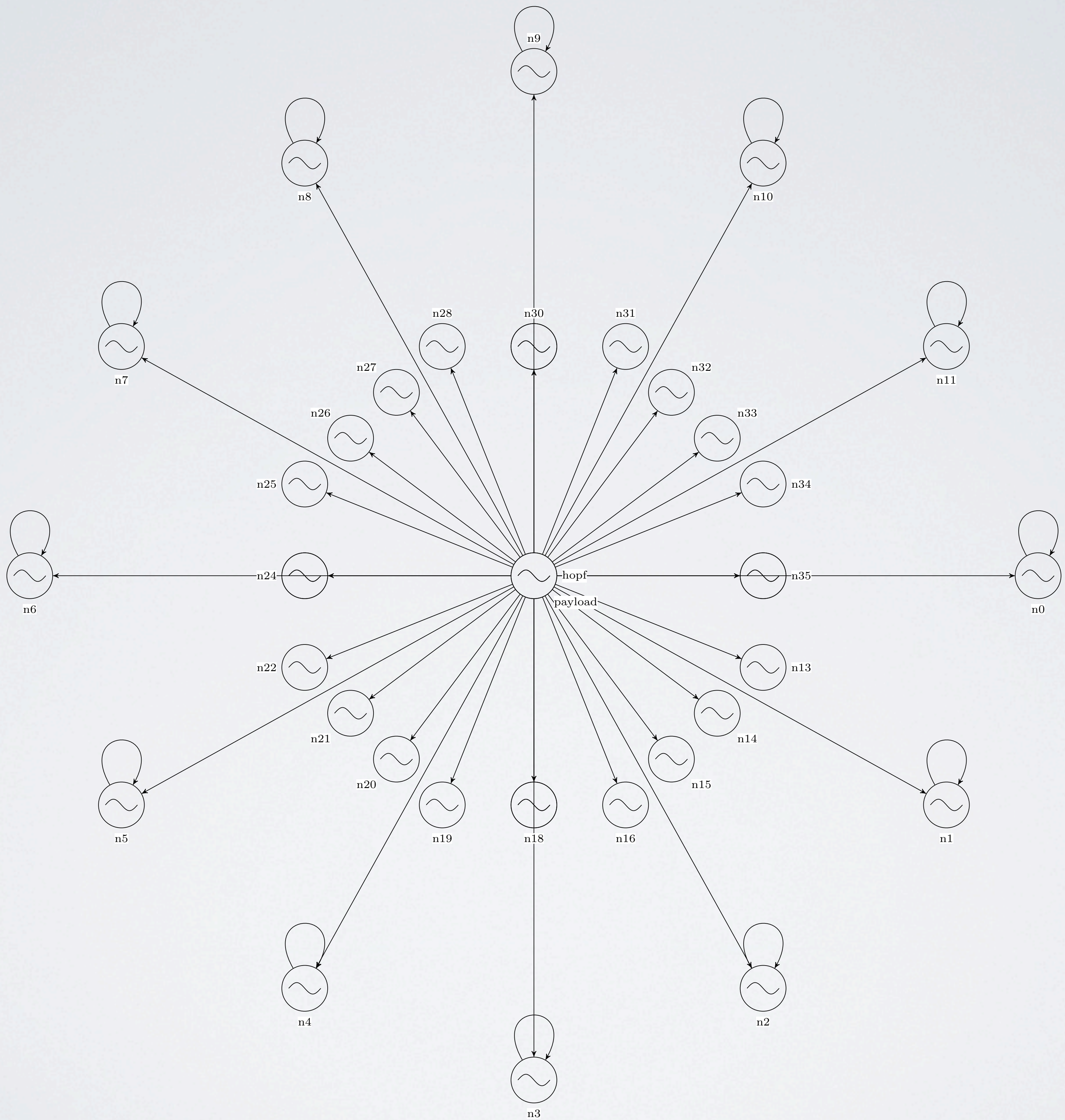
SIMULATION FLOWCHART



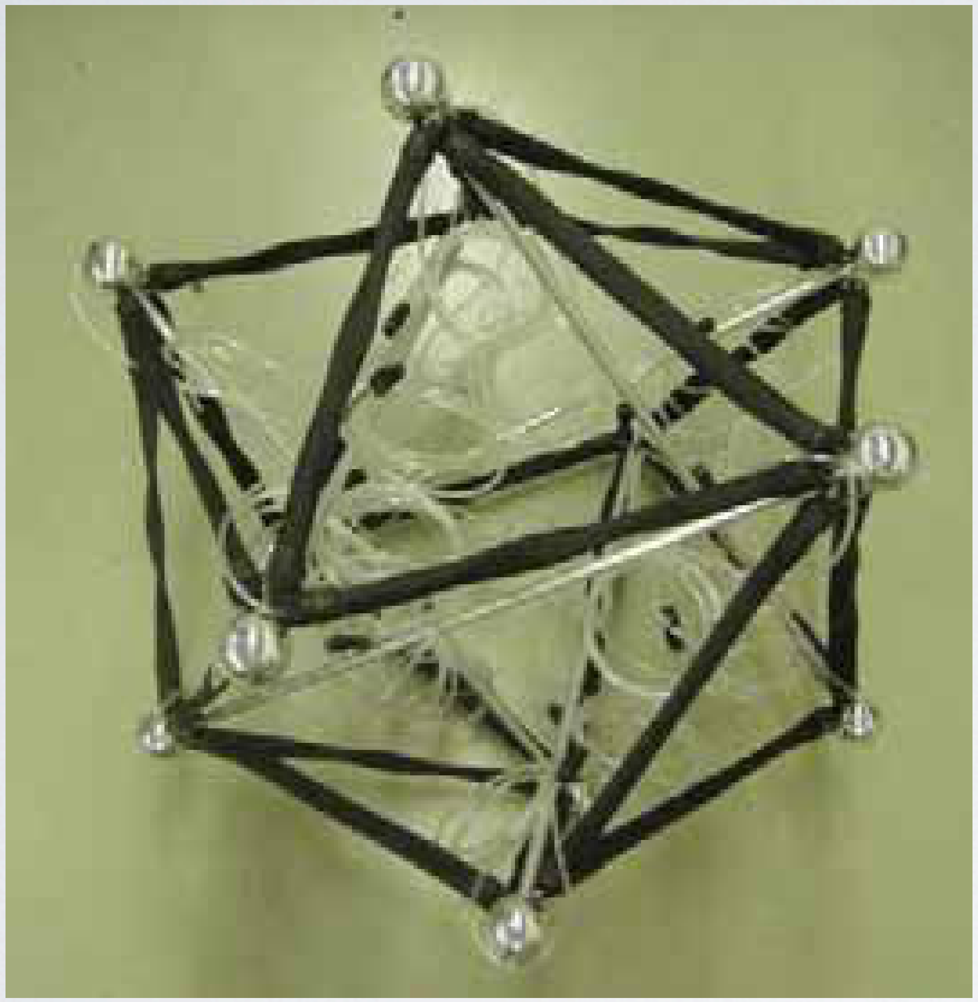
FITNESS RESULTS

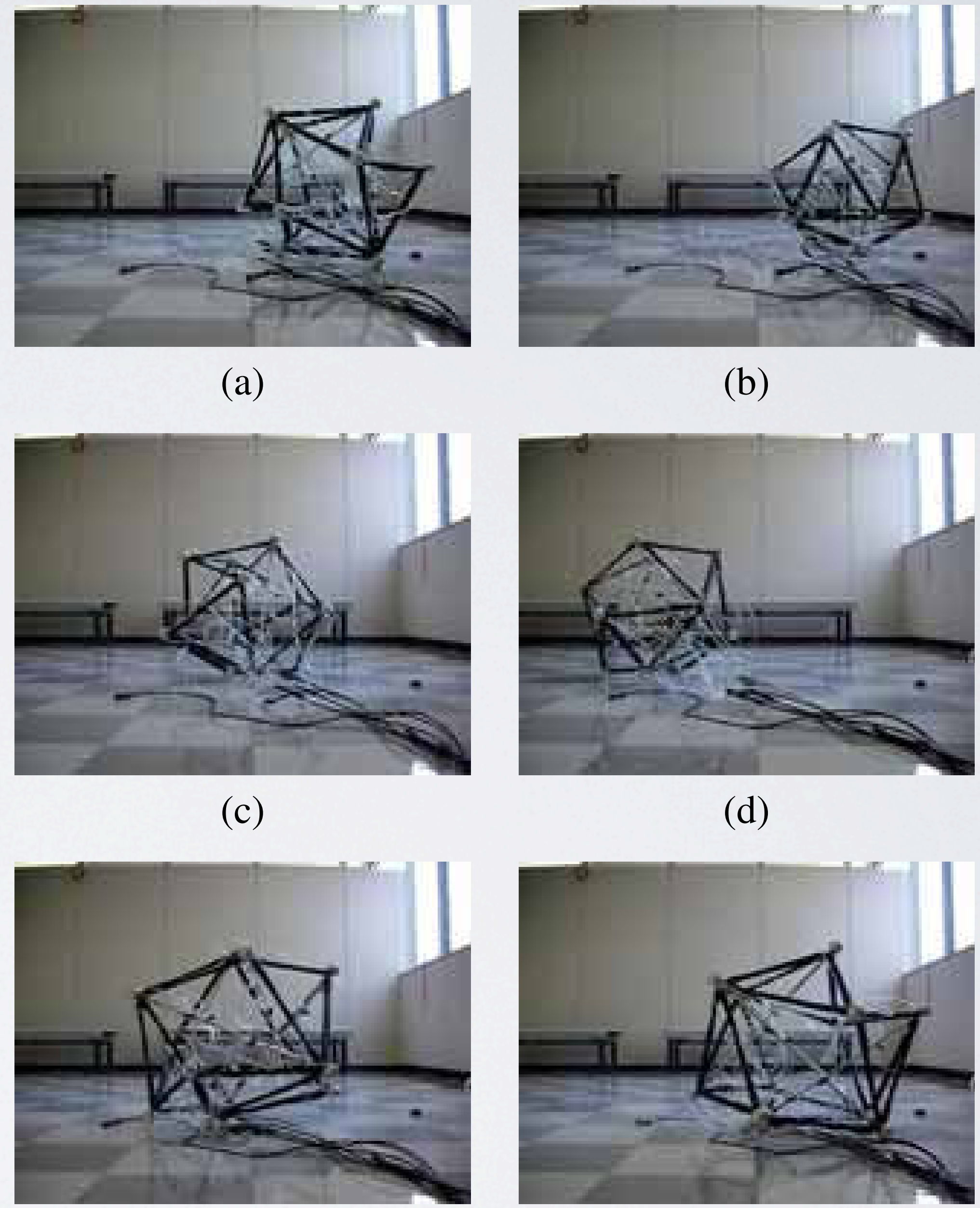


CPG NETWORK

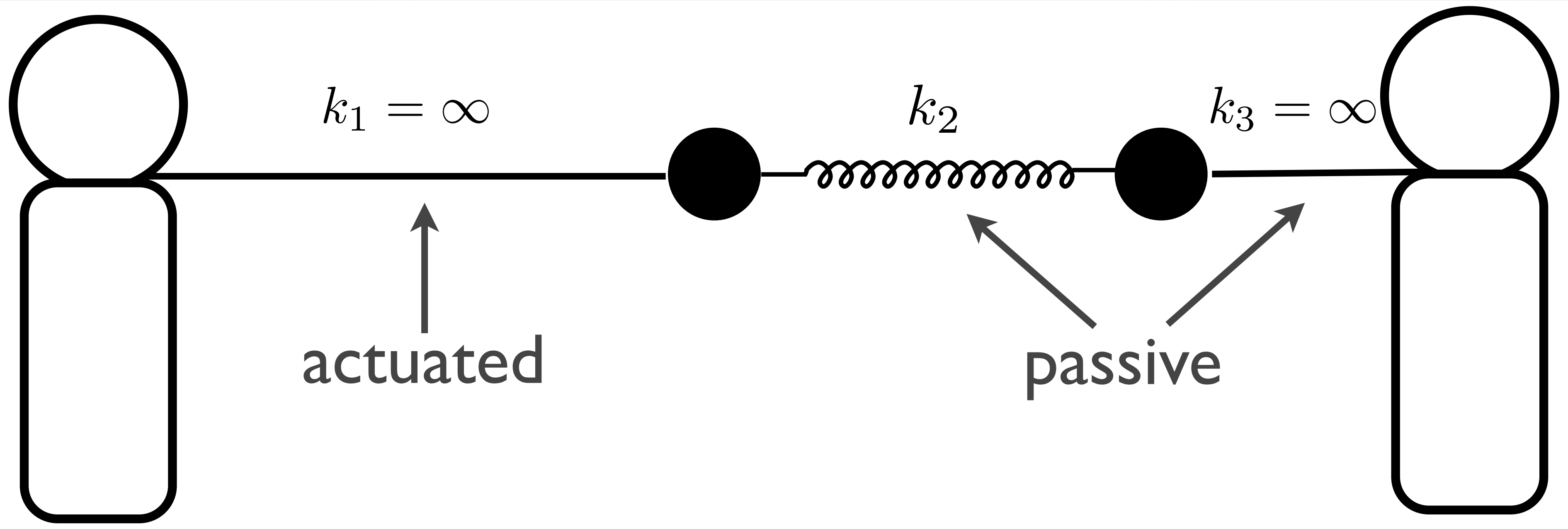

n9
n8
n10
n28
n30
n31
n7
n27
n32
n11
n26
n33
n25
n34
n6
n24
hopf
n35
n0
payload
n22
n13
n21
n14
n20
n15
n5
n19
n18
n16
n1
n4
n2
n3

KOIZUMI ET AL.





3 SEGMENTS MUSCLES



$$\begin{cases} F_i = k_i(\ell_i - l_i) - \eta \dfrac{(l_i^{(t)} - l_i^{(t-1)})}{dt} & , l_i > \ell_i \\ 0 & , \text{otherwise} \end{cases}$$