# *Simulating Artificial Muscles for Controlling a Robotic Arm with Fluctuation*

***Max Braun***
*Osaka University, Yuragi Project, 2007/2008*

Yuragi

ゆらぎ *“Fluctuation”*

ゆらぎ *“Fluctuation”*

*Biology*

*Information science*

*Materials science*

*Robotics*

ゆらぎ *“Fluctuation”*

*Biology*

*Information science*

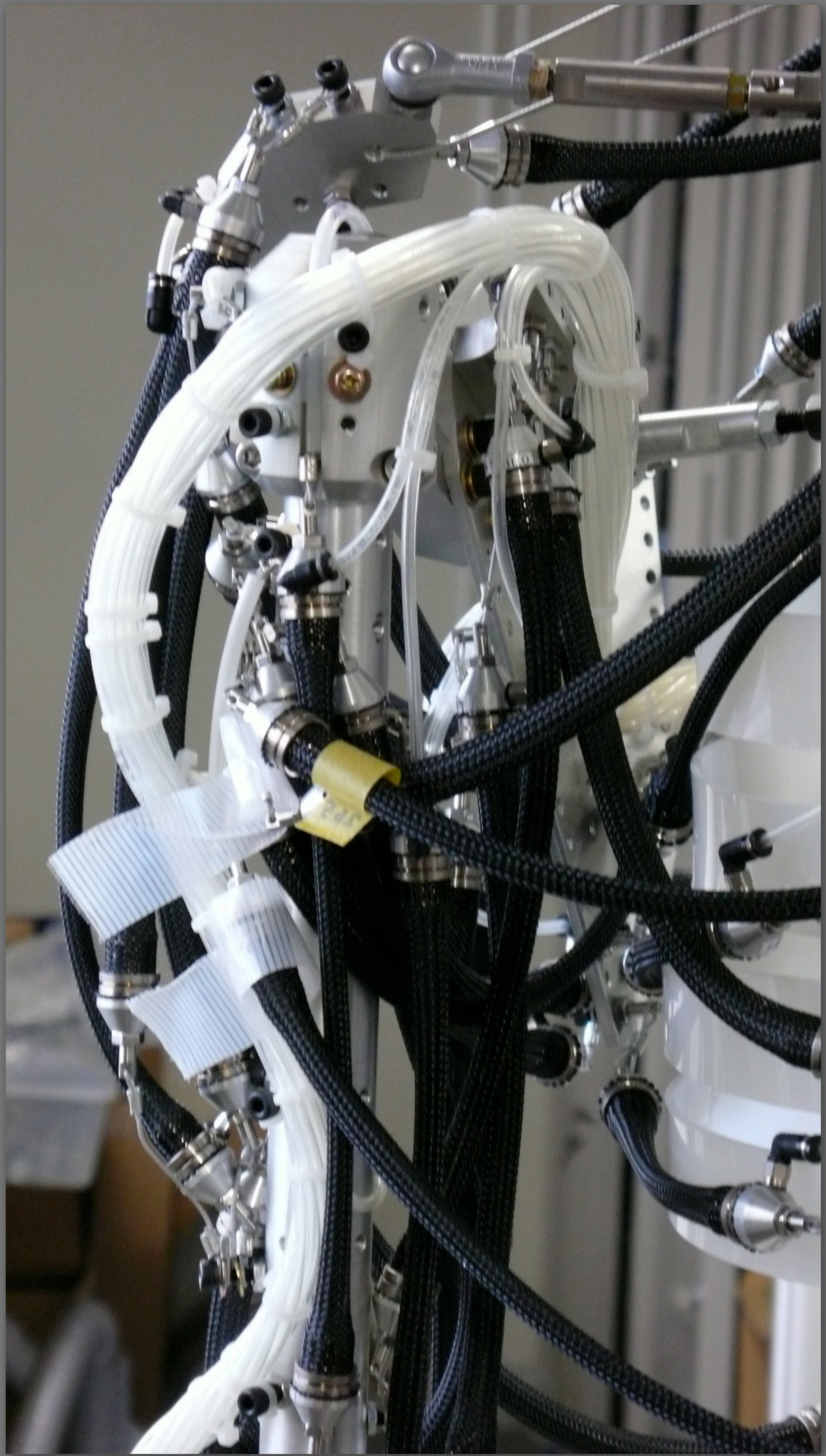
*Materials science*

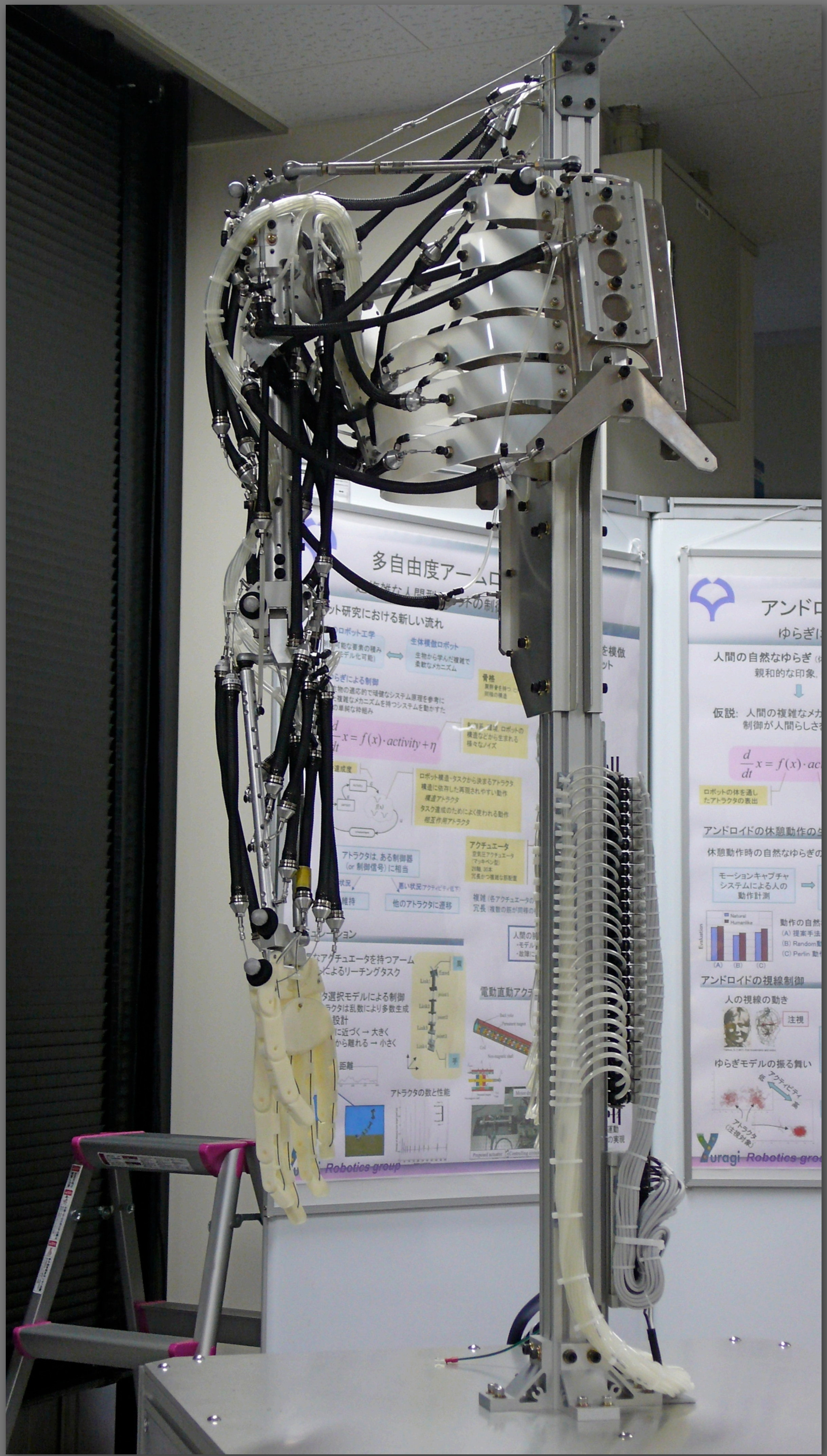
*Robotics*

1. *The robotic arm and its artificial muscles*
2. *Simulating artificial muscles*
3. *The simulator software*
4. *Control algorithm and results*

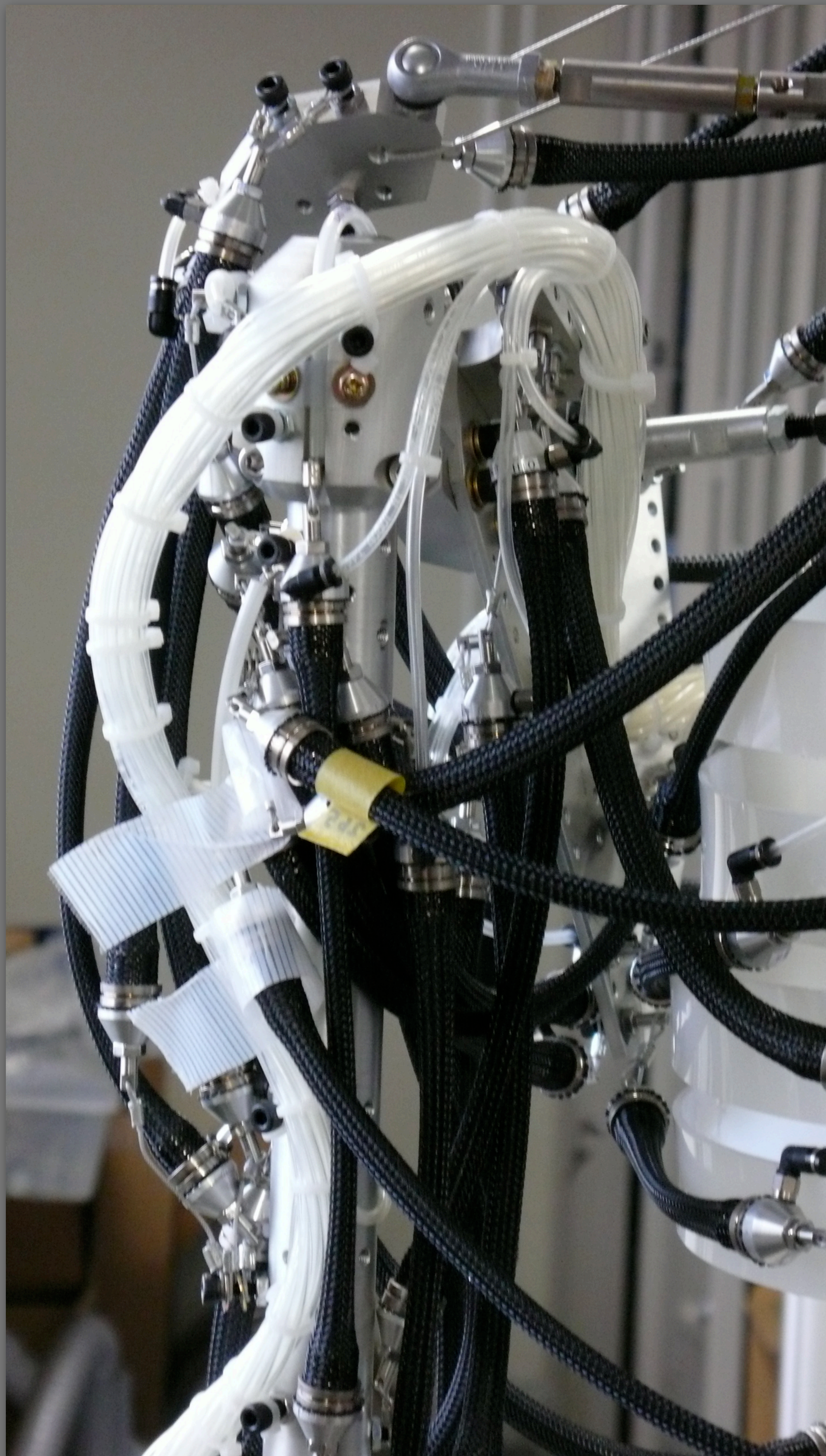
1. *The robotic arm and its artificial muscles*
2. *Simulating artificial muscles*
3. *The simulator software*
4. *Control algorithm and results*


人間の自然なゆらぎ
アンドロイドの視線制御
人の視線の動き
ゆらぎモデルの振る舞い

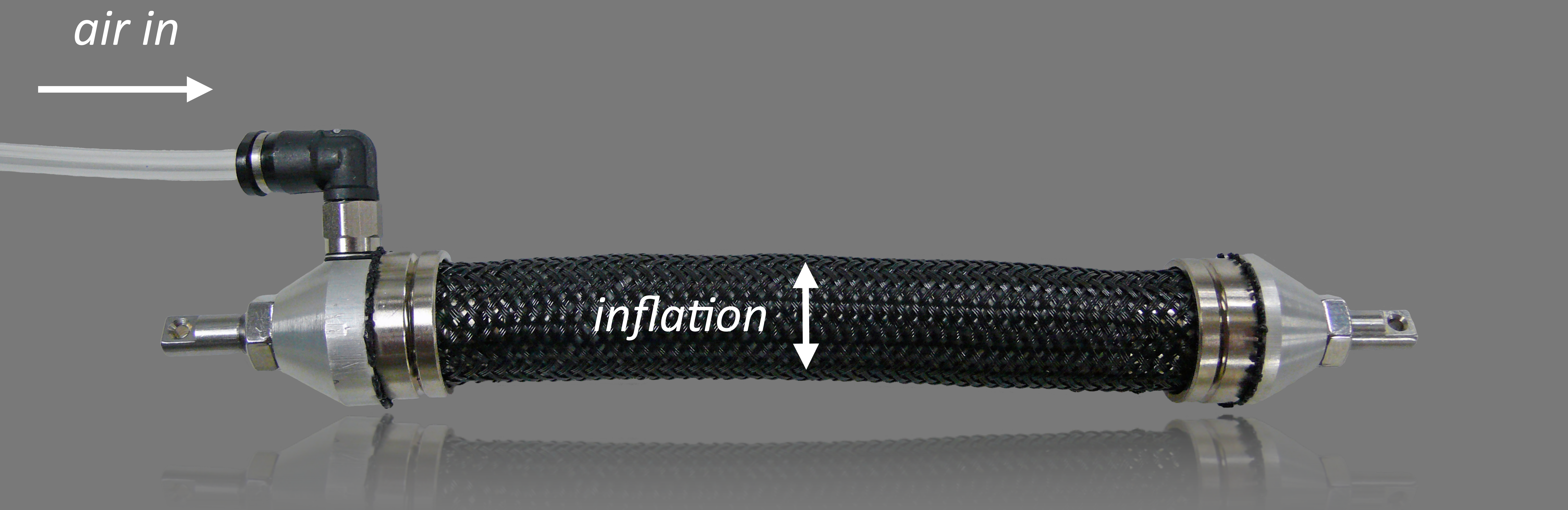


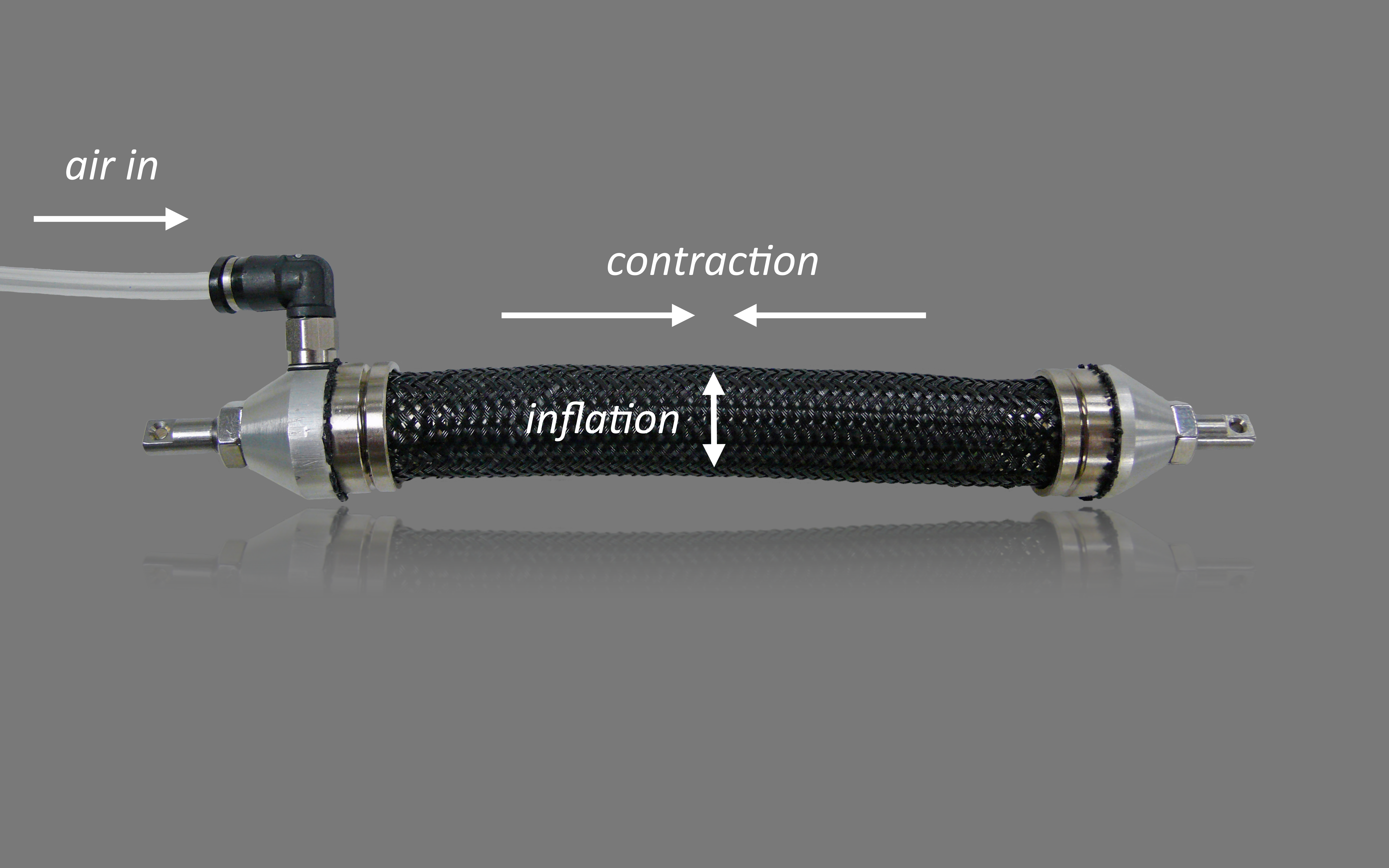


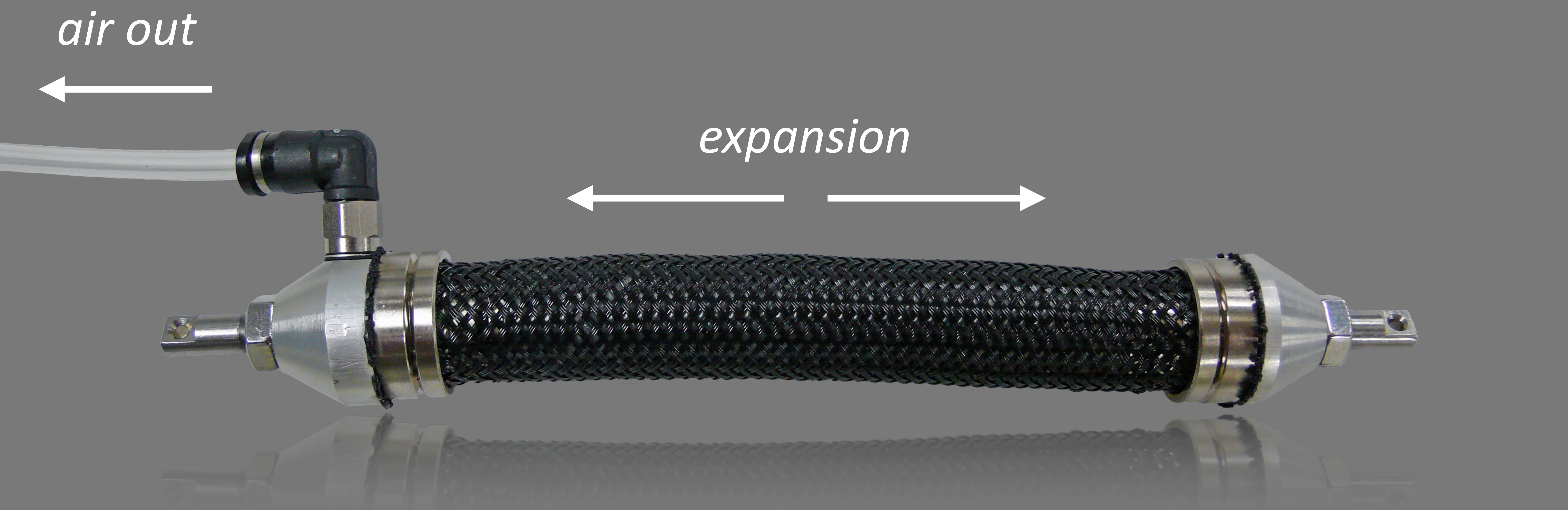
ロボット研究における新しい流れ
人間の自然なゆらぎ
アンドロイドの視線制御
人の視線の動き
ゆらぎモデルの振る舞い
Robotics group

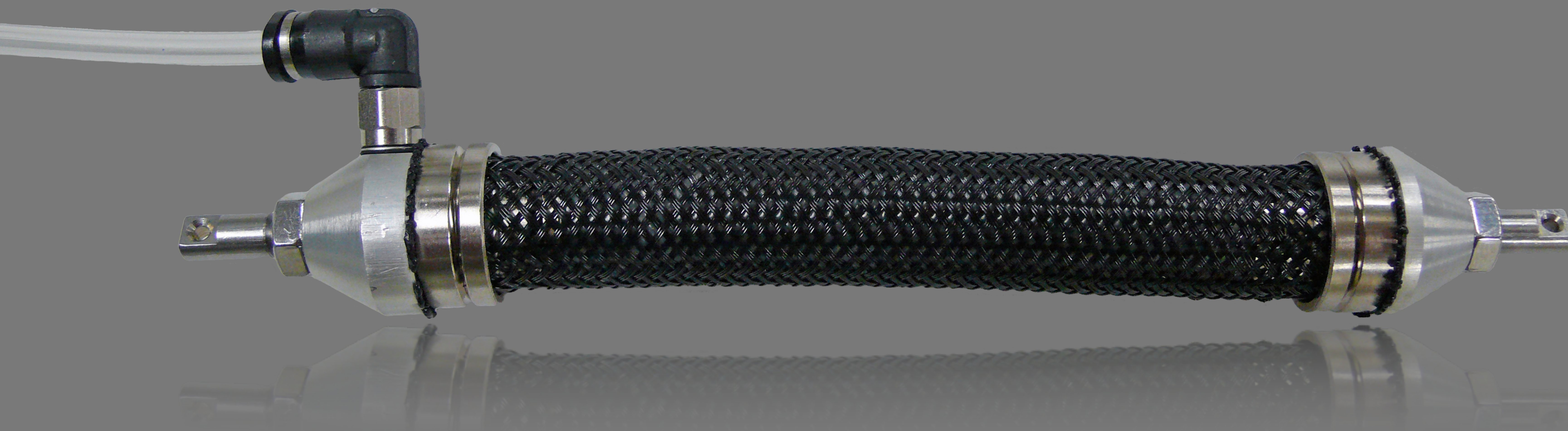

air in

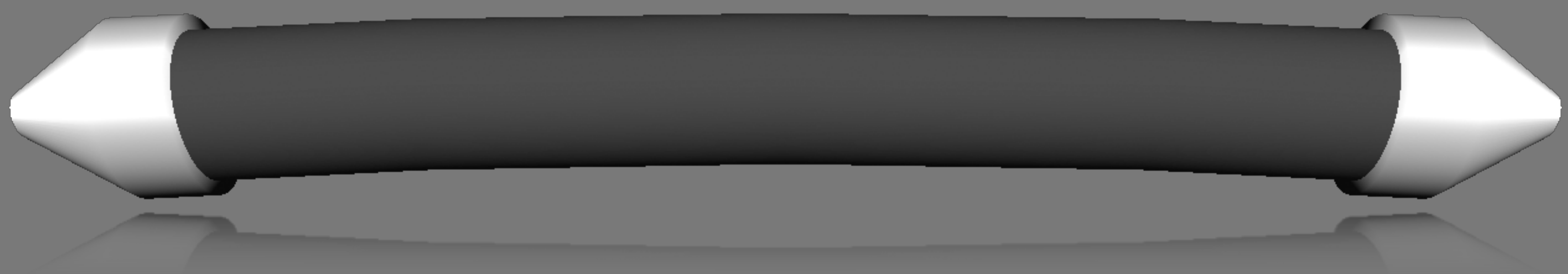

air in
inflation

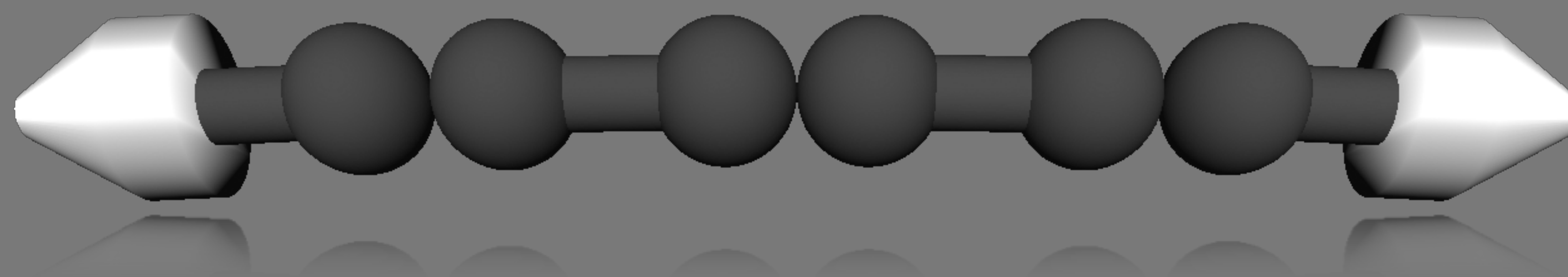

air in
contraction
inflation

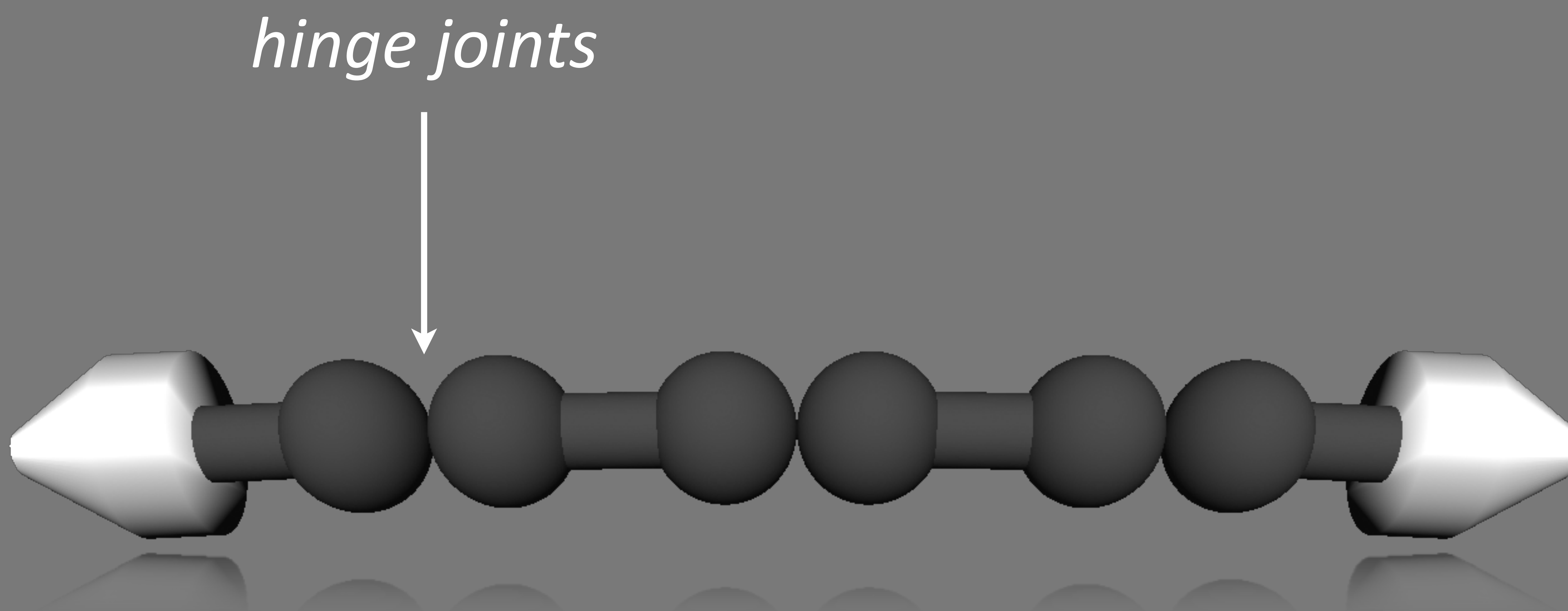

air out
expansion

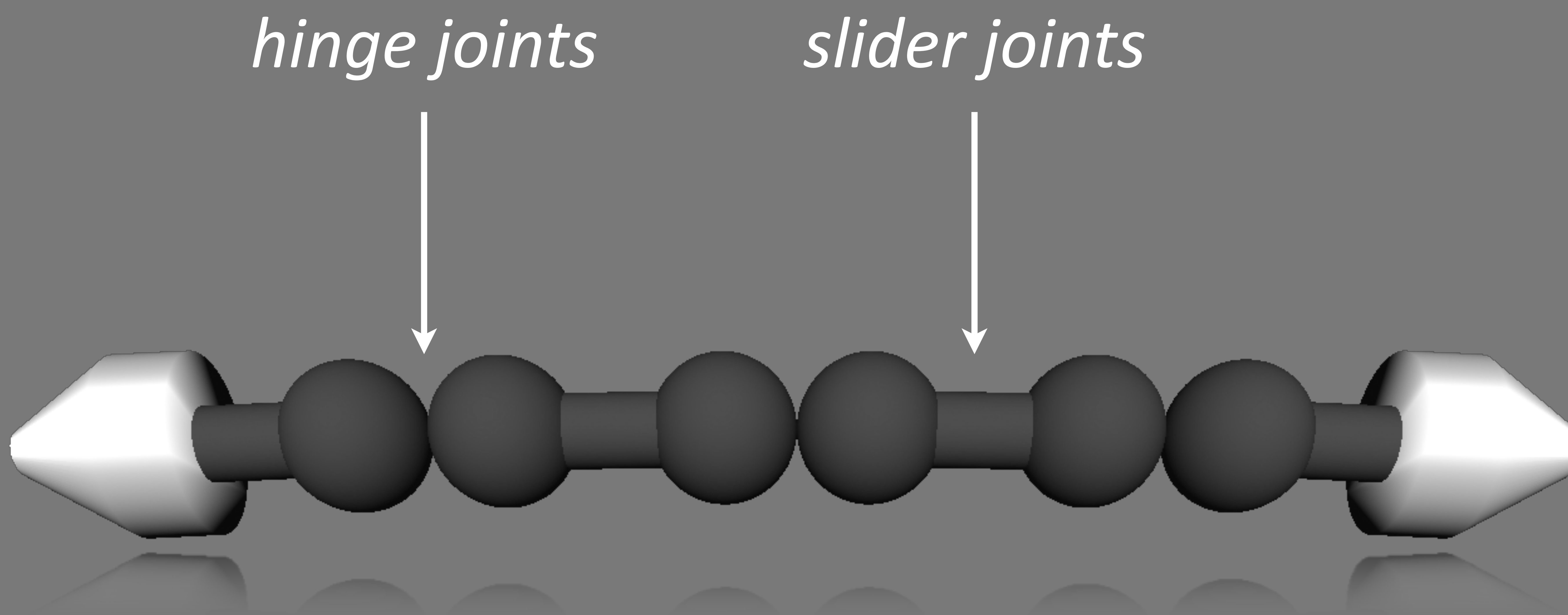
1. *The robotic arm and its artificial muscles*
2. *Simulating artificial muscles*
3. *The simulator software*
4. *Control algorithm and results*

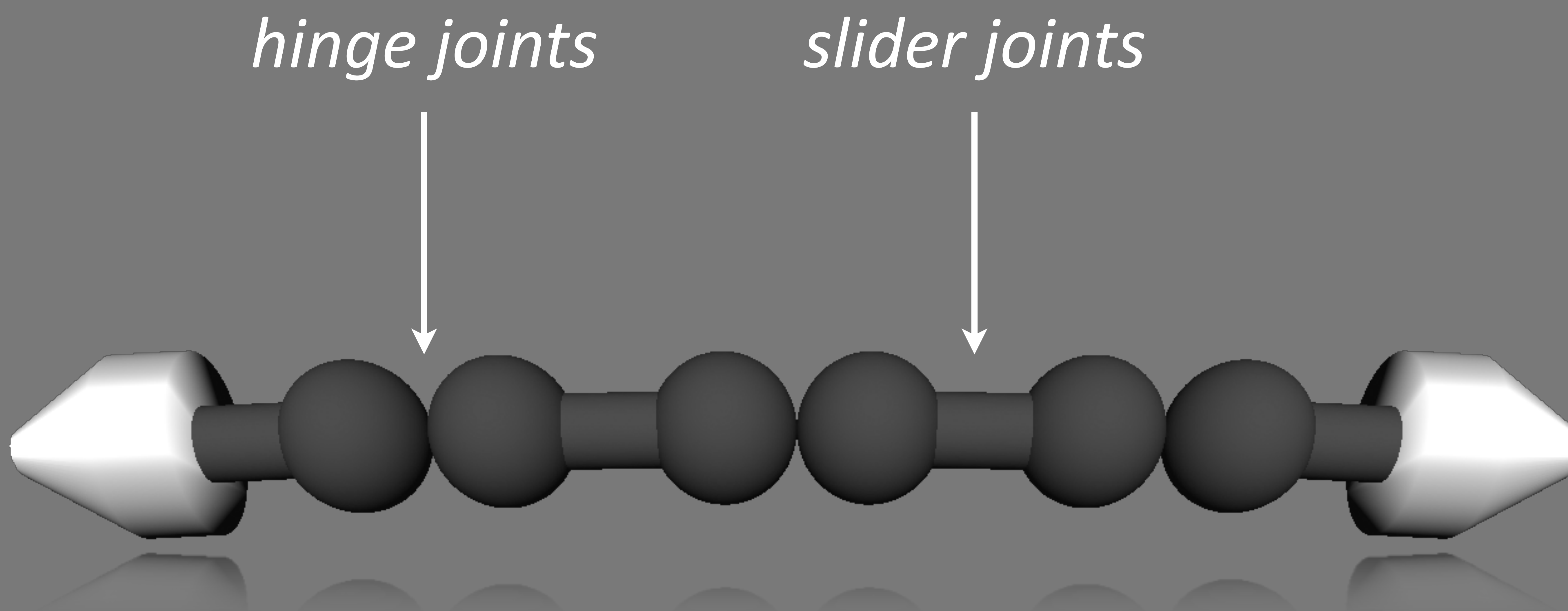




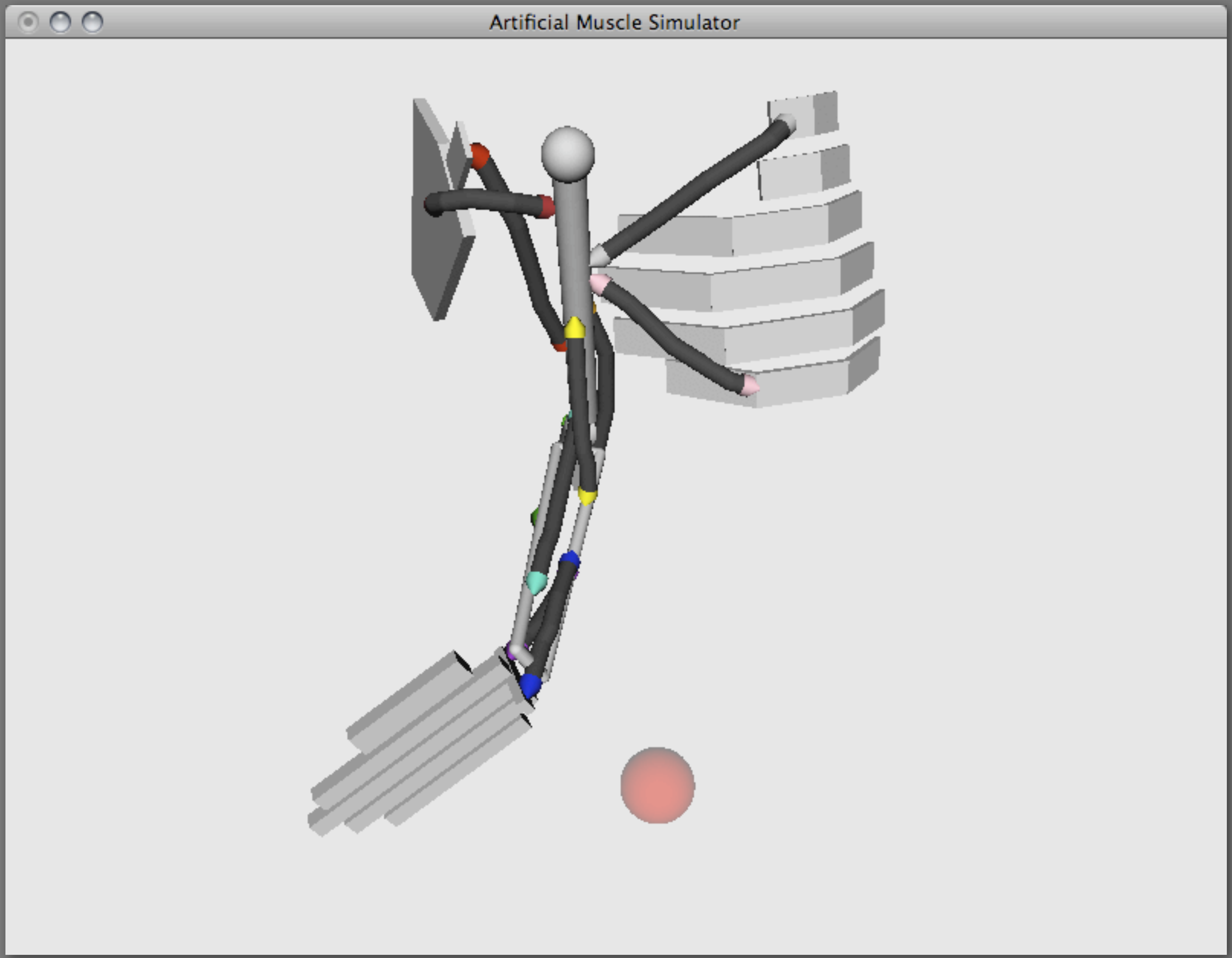



hinge joints


hinge joints
slider joints




***Open Dynamics Engine** (ODE) + **OpenGL***




# Demo

# *Refining the model*

*Experiment:*

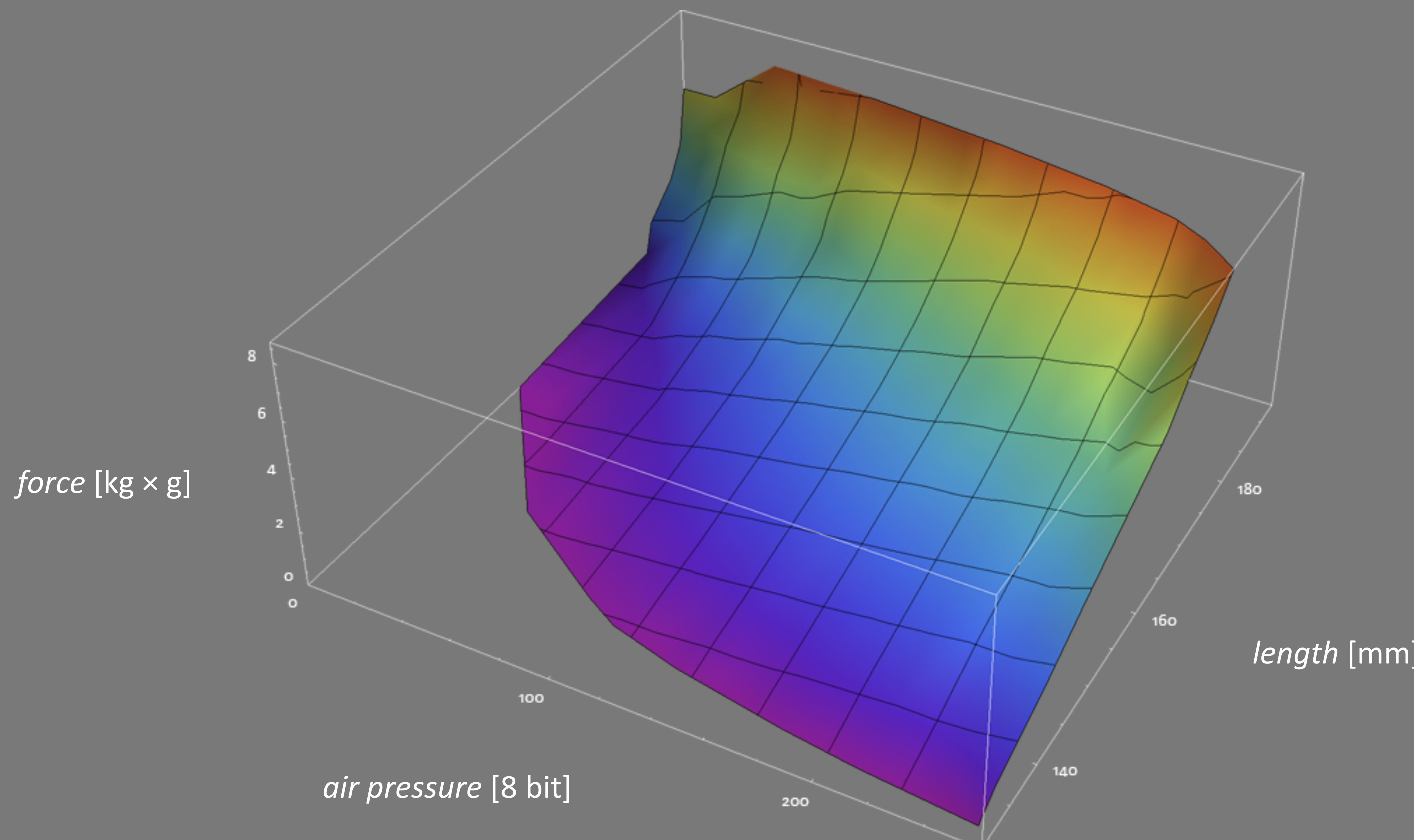
- Set ***air pressure*** (8 bit)
- Set ***weight*** (i.e. force, 0–8.7 kg × g)
- Measure ***length*** (e.g. 132–193 mm)

*Simulator:*

- Take ***air pressure***
- Measure ***length***
- Exert ***force***

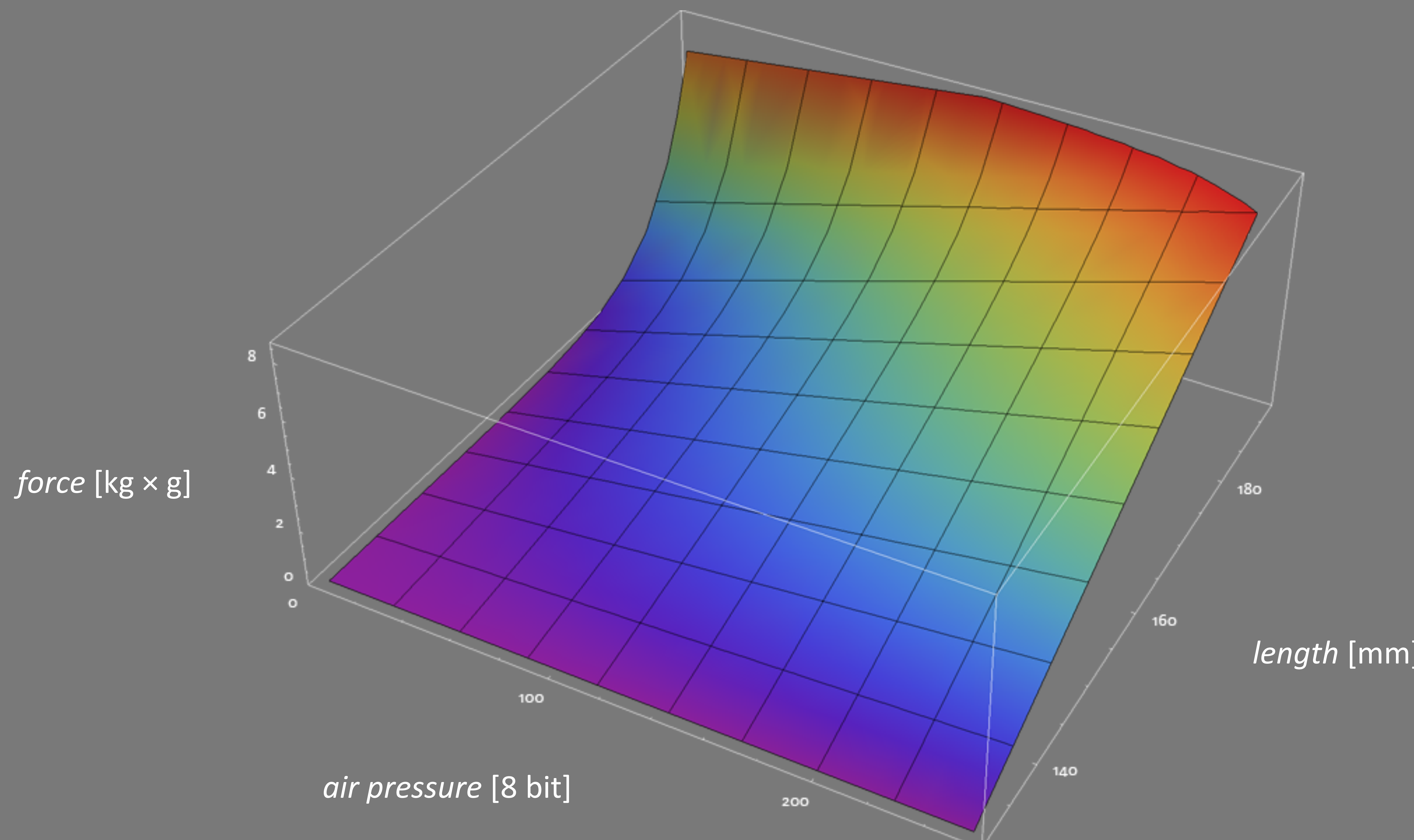
# *Measurement*

From a *set of data points* to a *functional representation*



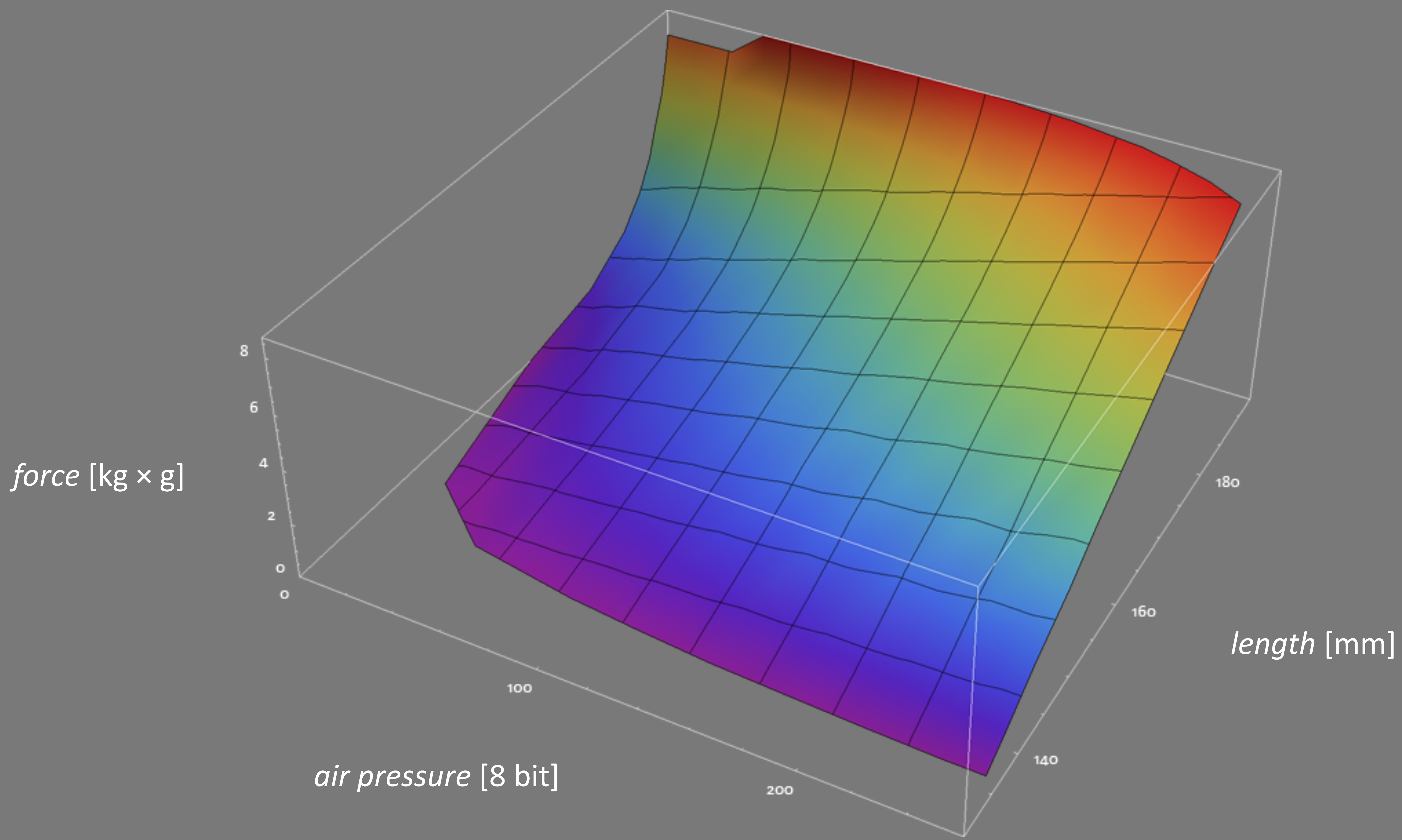
# *Function fit*

$$F(p,l) = c \cdot \left( \frac{p}{p_{\max}} \cdot \frac{l - l_{\min}}{l_{\max} - l_{\min}} + a \cdot \left(1 - \frac{p}{p_{\max}}\right) \cdot \exp\left(b \cdot \frac{l - l_{\min}}{l_{\max} - l_{\min}}\right)\right)$$



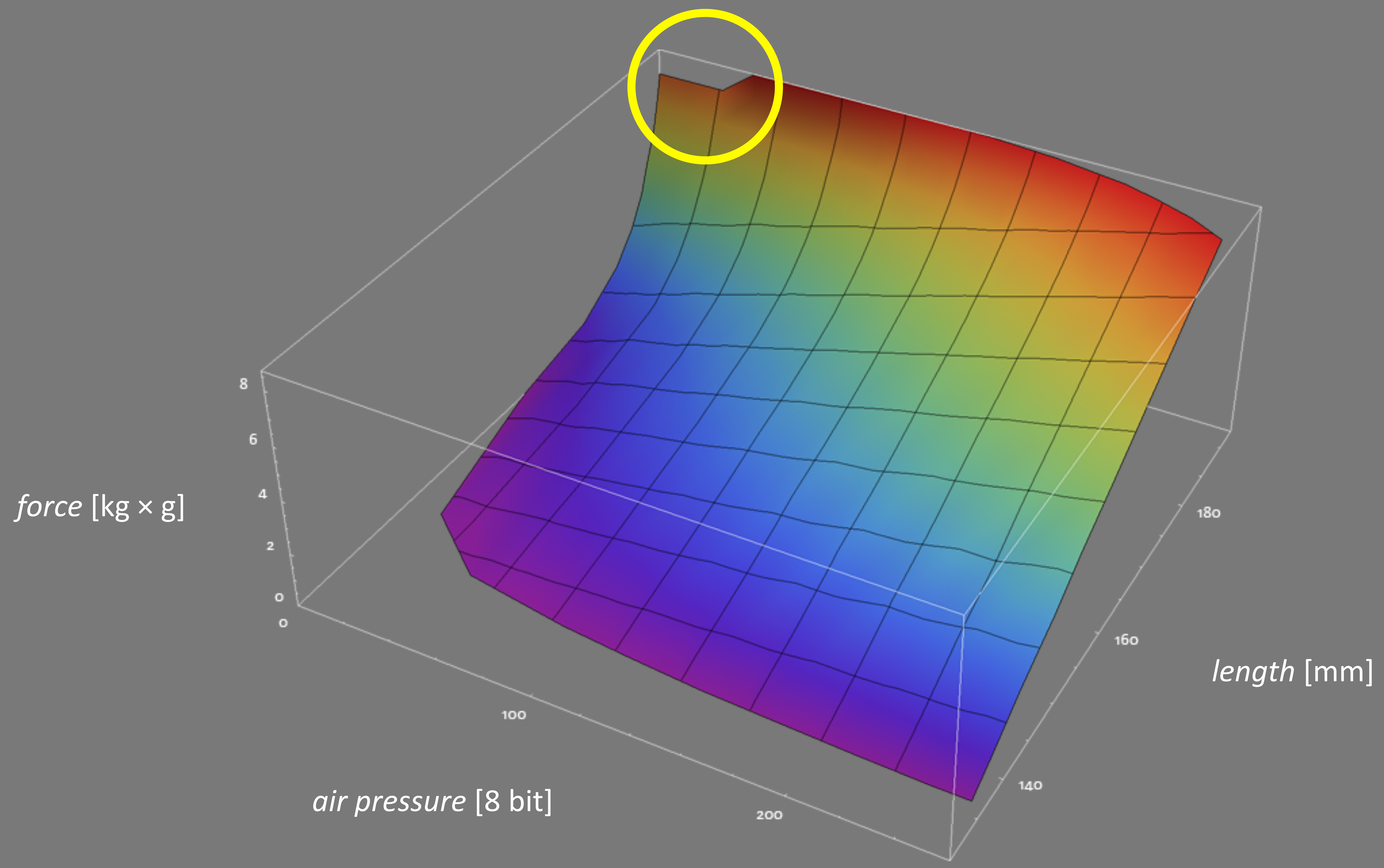
# *Measurement in simulator*

$$\frac{\sum_i (\mathrm{measured}_i - \mathrm{simulated}_i)^2}{\sum_i \mathrm{measured}_i^2} \approx 4.2 \cdot 10^{-4}$$

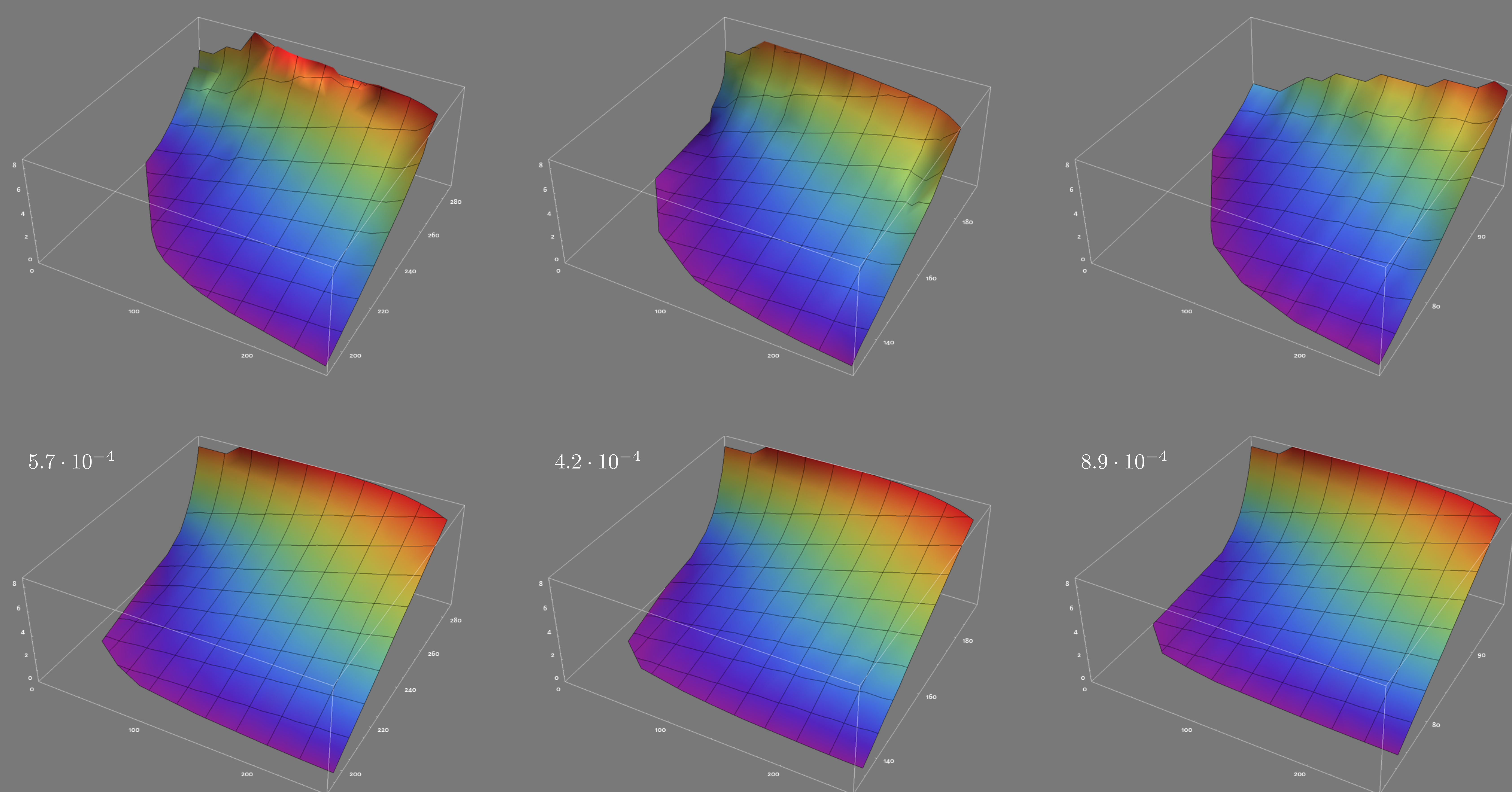


# *Measurement in simulator*

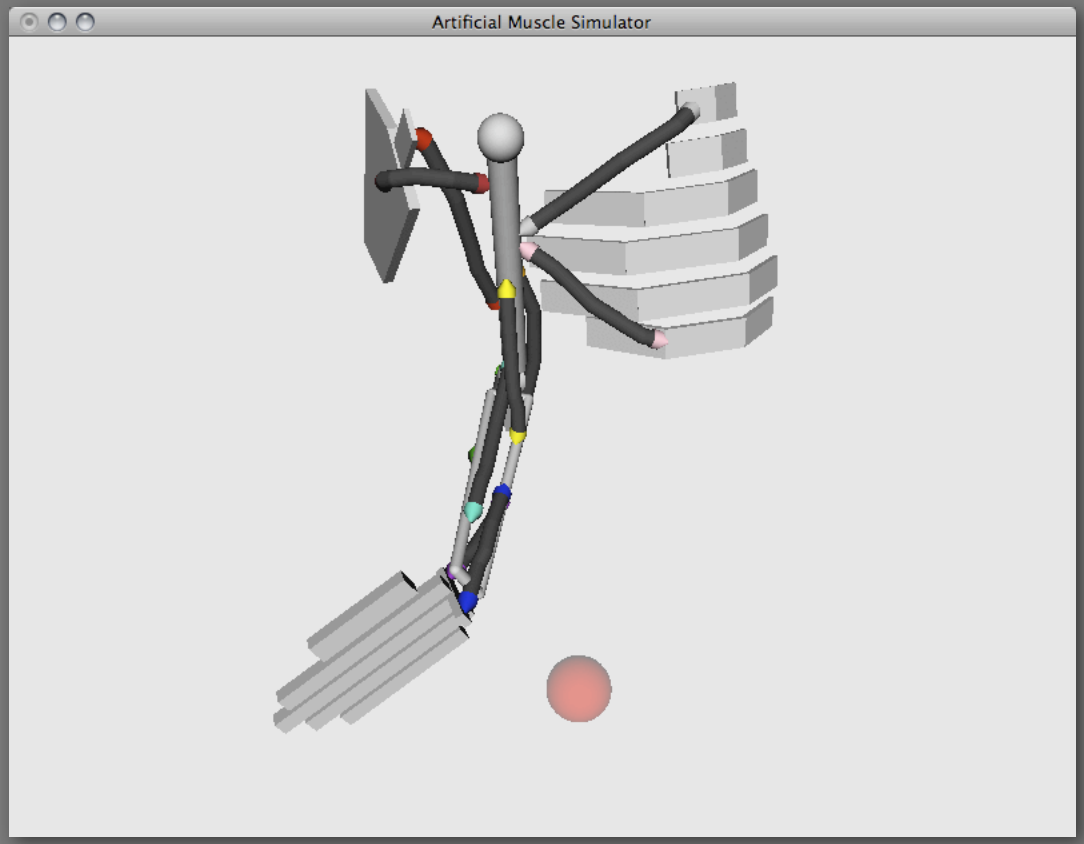
$$\frac{\sum_i (\text{measured}_i - \text{simulated}_i)^2}{\sum_i \text{measured}_i^2} \approx 4.2 \cdot 10^{-4}$$

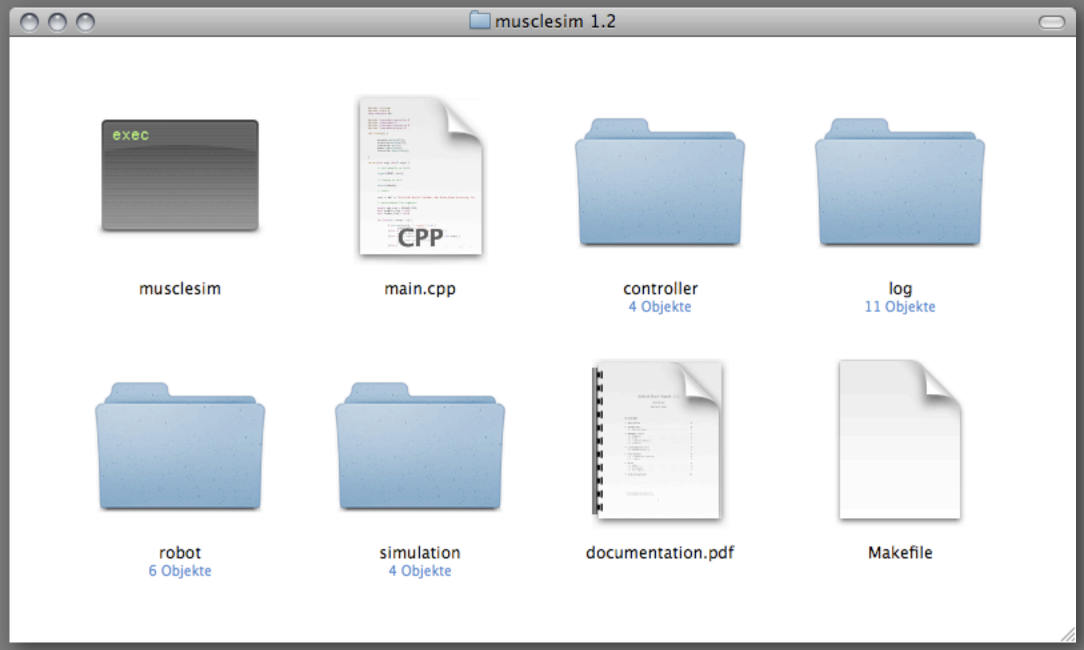


***long** muscle* | ***medium** muscle* | ***short** muscle*



1. *The robotic arm and its artificial muscles*
2. *Simulating artificial muscles*
3. ***The simulator software***
4. *Control algorithm and results*



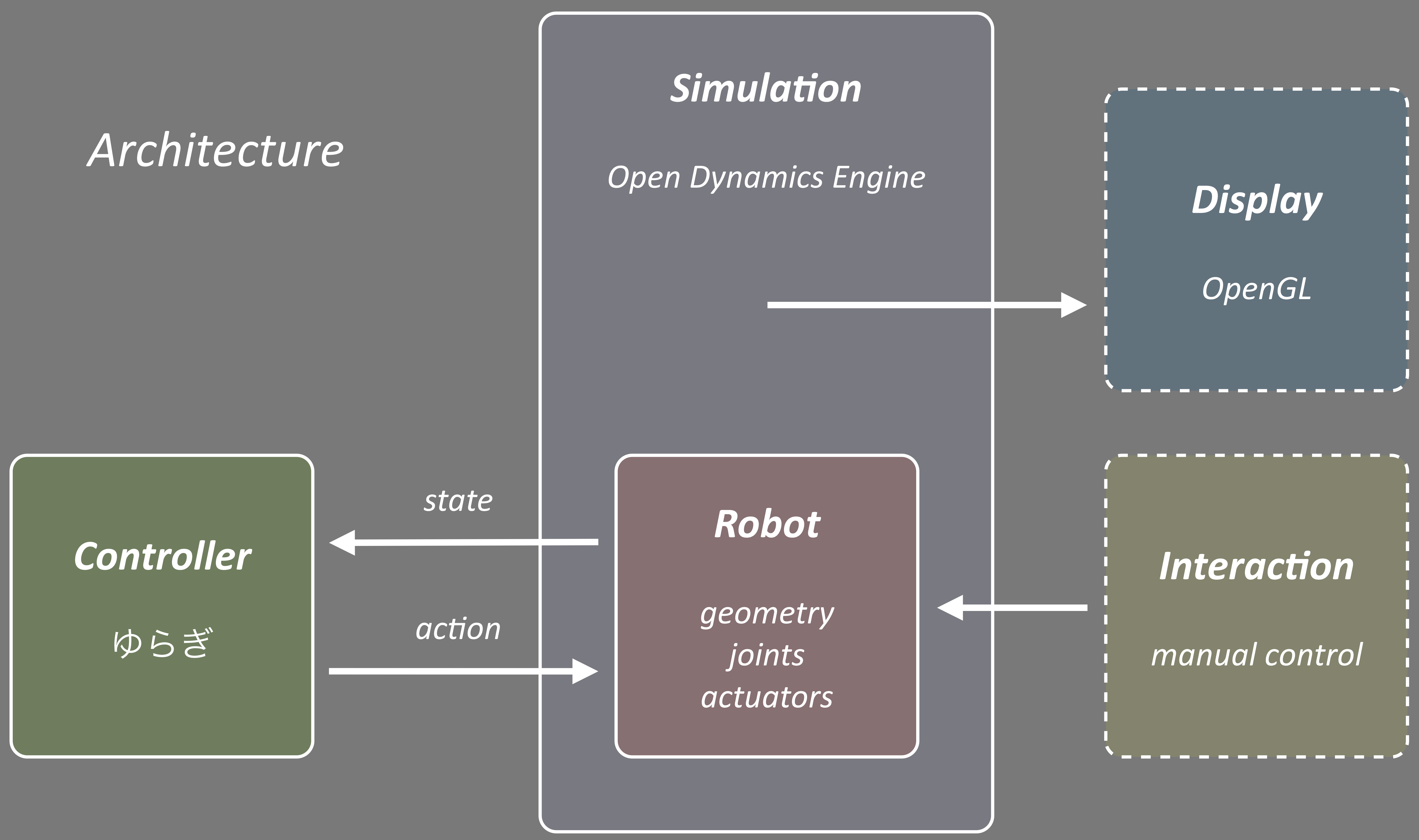

# *Software engineering*

*User tasks:*

1. ***Build*** a robot
   - Create geometry
   - Connect with joints
   - Place actuators
2. ***Control*** the robot dynamically

*Realization:*

- Abstraction of ***ODE*** and ***OpenGL***
- Written in ***C++*** (also interface)


Architecture
Simulation
Open Dynamics Engine
Display
OpenGL
Controller
ゆらぎ
state
action
Robot
geometry
joints
actuators
Interaction
manual control

# Documentation

## Artificial Muscle Simulator 1.2

Max Braun*

March 21, 2008

### Contents



*max.braun@ams.eng.osaka-u.ac.jp

1. *The robotic arm and its artificial muscles*
2. *Simulating artificial muscles*
3. *The simulator software*
4. *Control algorithm and results*

## *Attractor selection model*

$$\dot{\mathbf{x}} = a \cdot \nabla P(\mathbf{x}) + \eta$$

# *Attractor selection model*

*state change* $\longrightarrow$

$$\dot{\mathbf{x}} = a \cdot \nabla P(\mathbf{x}) + \eta$$

# *Attractor selection model*

*potential field* $$P(\mathbf{x}) = -\sum_{i=1}^{n} \frac{1}{\sqrt{2\pi\sigma^2}} \exp(-\frac{1}{2\sigma^2}|\mathbf{x}-\mathbf{x}_i|^2)$$

*state change* $$\boxed{\dot{\mathbf{x}} = a \cdot \nabla P(\mathbf{x}) + \eta}$$

# *Attractor selection model*

*potential field* $$P(\mathbf{x}) = -\sum_{i=1}^{n} \frac{1}{\sqrt{2\pi\sigma^2}} \exp(-\frac{1}{2\sigma^2}|\mathbf{x}-\mathbf{x}_i|^2)$$

*state change* $$\dot{\mathbf{x}} = a \cdot \nabla P(\mathbf{x}) + \eta$$

*attractors*

# *Attractor selection model*

*potential field* $$P(\mathbf{x}) = -\sum_{i=1}^{n} \frac{1}{\sqrt{2\pi\sigma^2}} \exp(-\frac{1}{2\sigma^2}|\mathbf{x} - \mathbf{x}_i|^2)$$

*state change* $\longrightarrow$ $$\boxed{\dot{\mathbf{x}} = a \cdot \nabla P(\mathbf{x}) + \eta}$$ $\longleftarrow$ *noise*

*attractors*

# *Attractor selection model*

*potential field* $P(\mathbf{x}) = -\sum_{i=1}^{n} \frac{1}{\sqrt{2\pi\sigma^2}} \exp(-\frac{1}{2\sigma^2}|\mathbf{x} - \mathbf{x}_i|^2)$

*state change*

$$\dot{\mathbf{x}} = a \cdot \nabla P(\mathbf{x}) + \eta$$

*noise*

*attractors*

*activity* $a = |\hat{\mathbf{p}} - \mathbf{p}|^{-1}$

# *Attractor selection model*

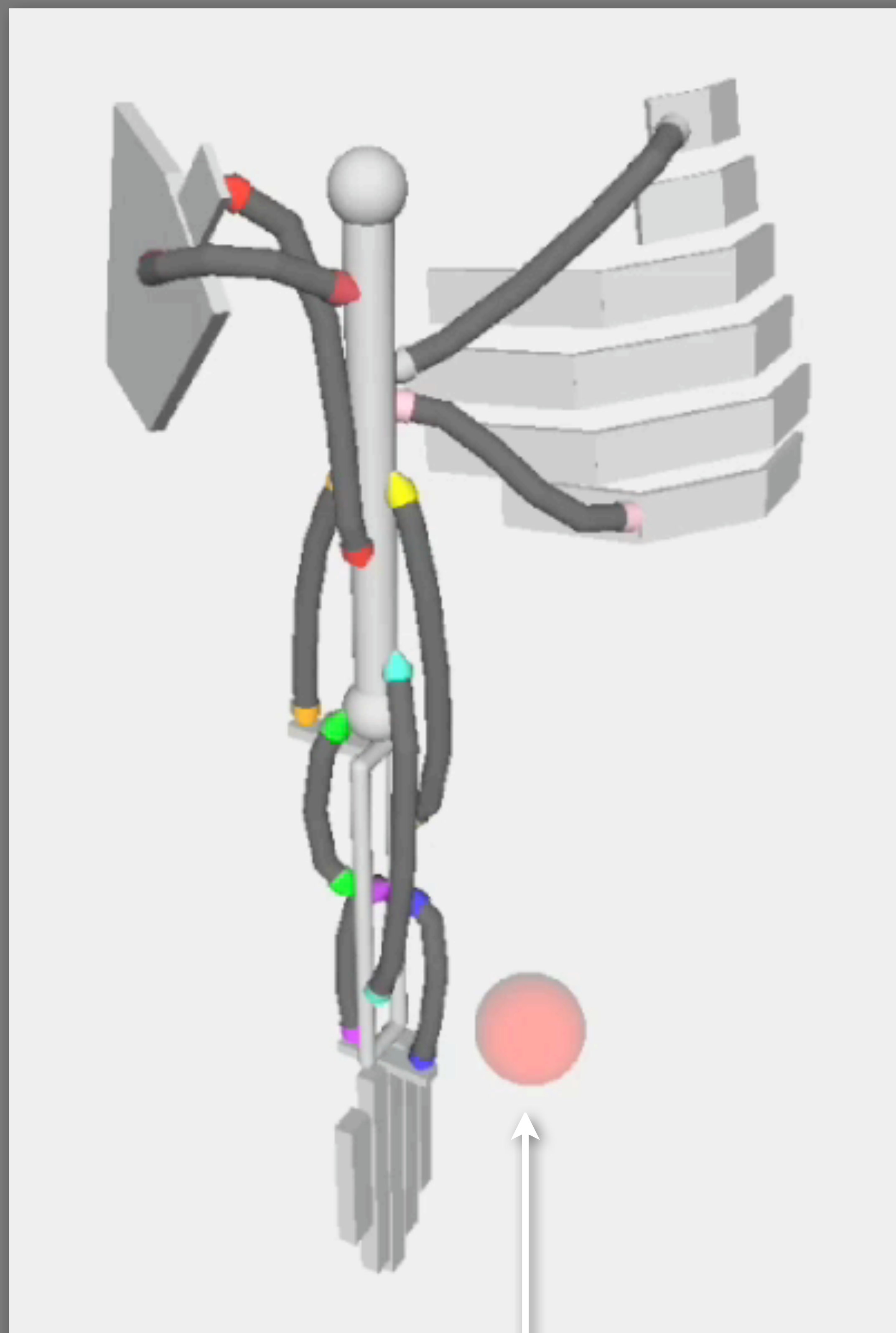
*potential field* $P(\mathbf{x}) = -\sum_{i=1}^{n} \frac{1}{\sqrt{2\pi\sigma^2}} \exp(-\frac{1}{2\sigma^2}|\mathbf{x}-\mathbf{x}_i|^2)$

*state change* $\dot{\mathbf{x}} = a \cdot \nabla P(\mathbf{x}) + \eta$ *noise*

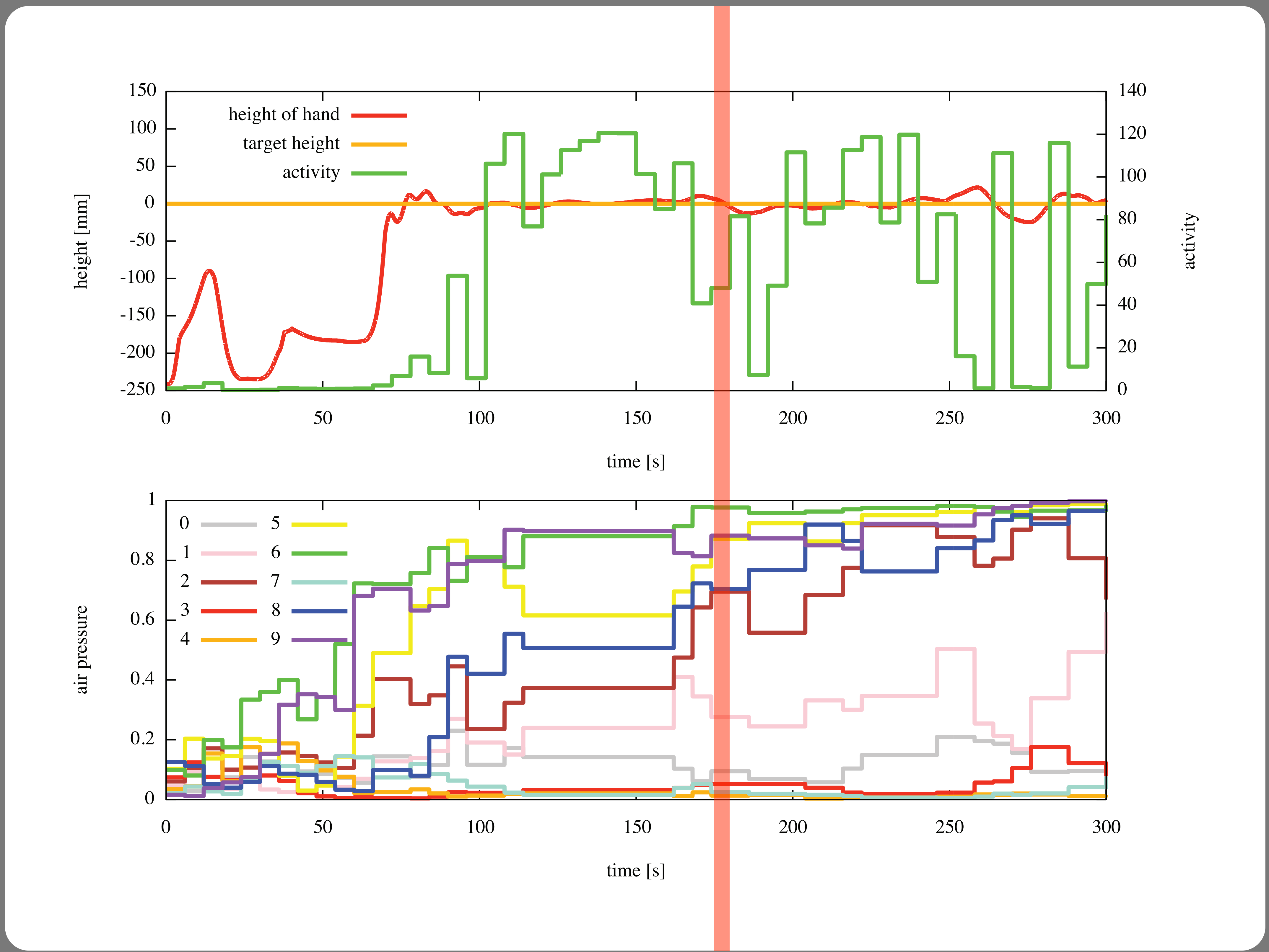
*attractors*

*activity* $a = |\hat{\mathbf{p}} - \mathbf{p}|^{-1}$

*target* *position*



target



height of hand
target height
activity
height [mm]
activity
time [s]
0
1
2
3
4
5
6
7
8
9
air pressure
time [s]

***IROS 2008*** submission

*Y. Nakamura, I. Fukuyori, M. Braun,*
*Y. Matsumoto, H. Ishiguro:*
***Control of human-like robotic arm based on biological fluctuation.***

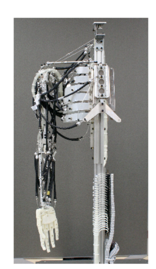


Abbildung 1: Roboterarm nach menschlichem Vorbild

***Zeitschrift Künstliche Intelligenz*** submission

*M. Braun, Y. Matsumoto, H. Ishiguro:*
***Yuragi: Ausnutzung von Fluktuation am Beispiel eines komplexen Roboterarms.***

大阪大学　“ゆらぎ”プロジェクト