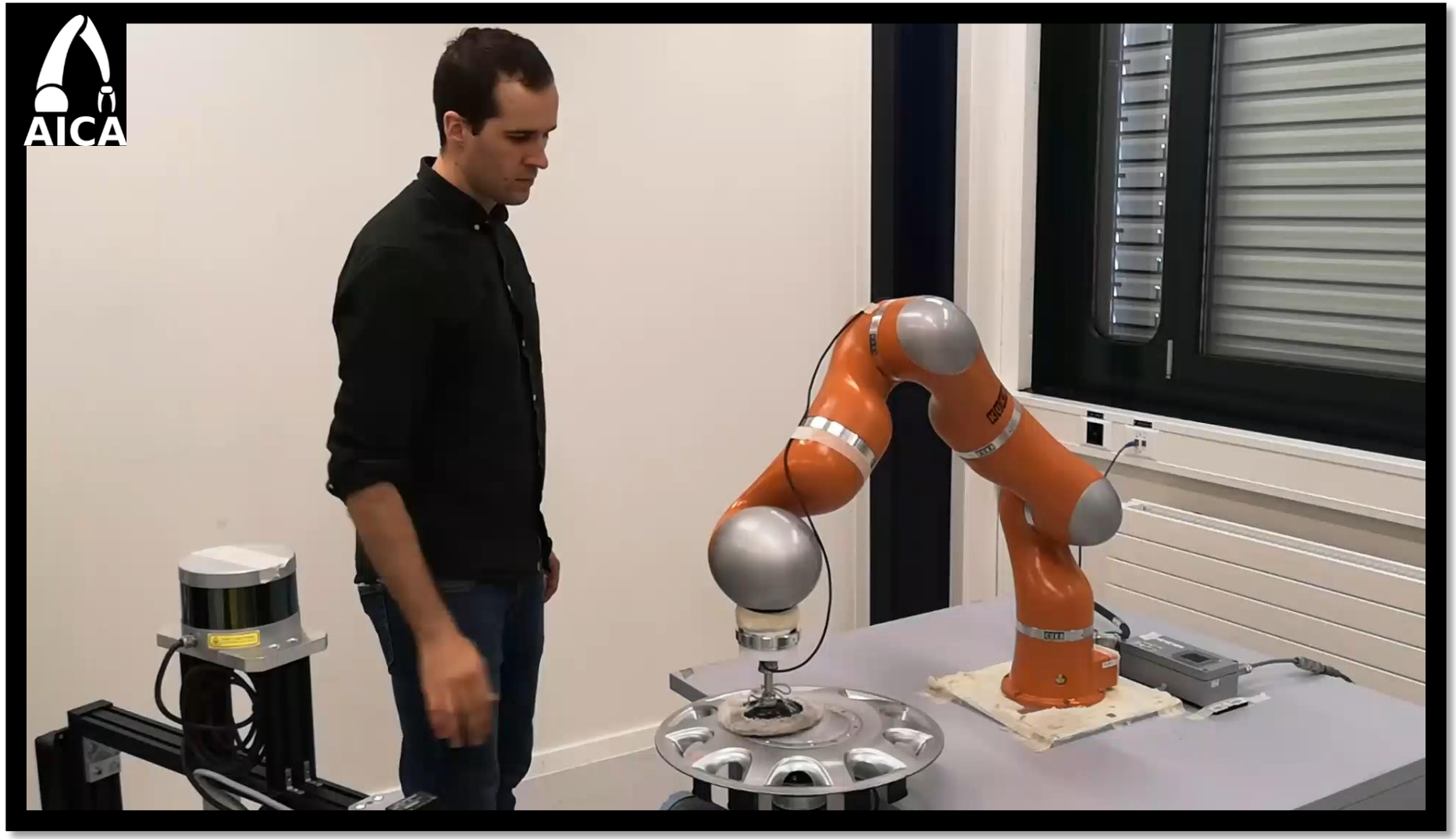
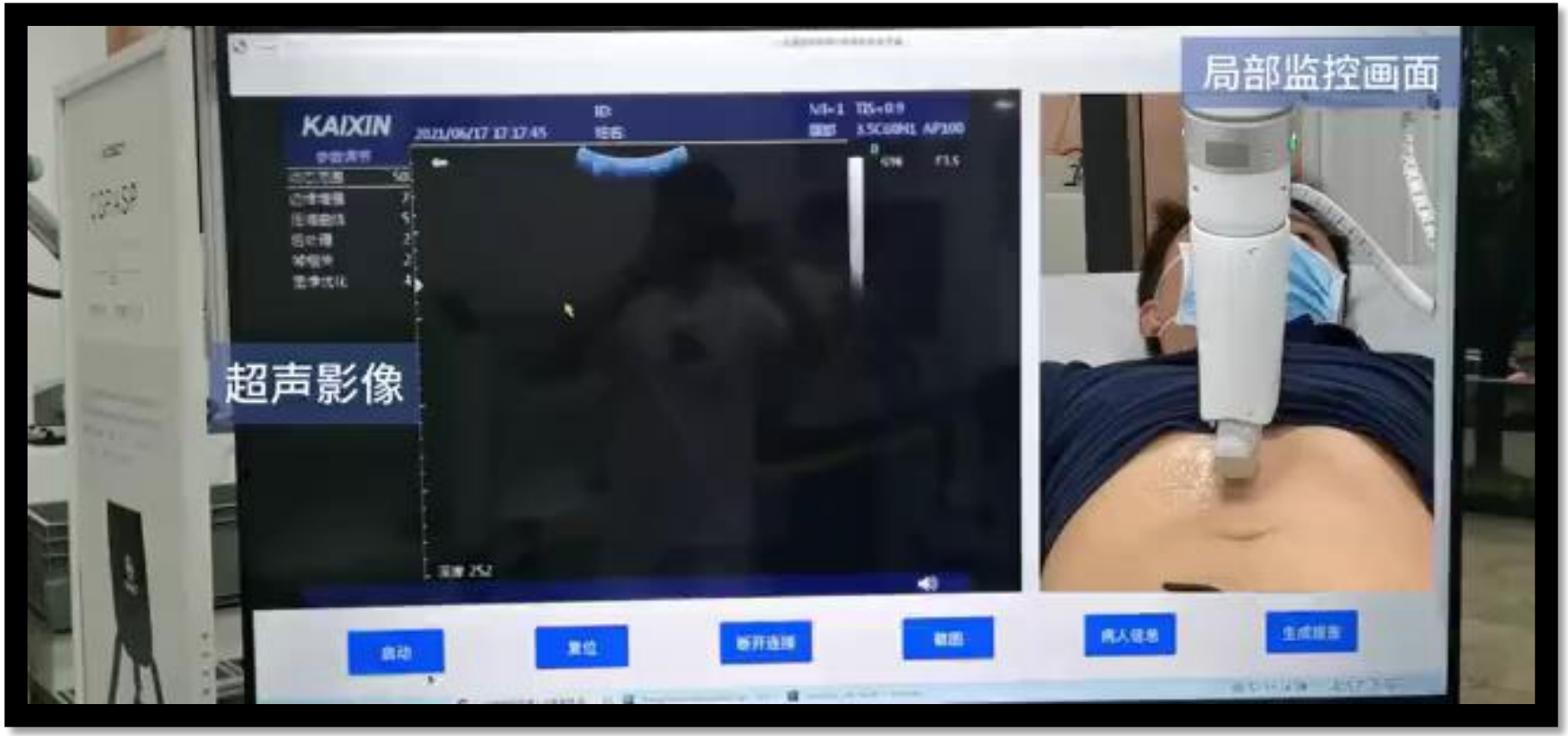
Force Control with Dynamical Systems

Amanhoud, W., Khoramshahi, M. and Billard, A. (2019) "A Dynamical System Approach to Motion and Force Generation in Contact Tasks." In Proceedings of Robotics, Science and Systems (RSS).

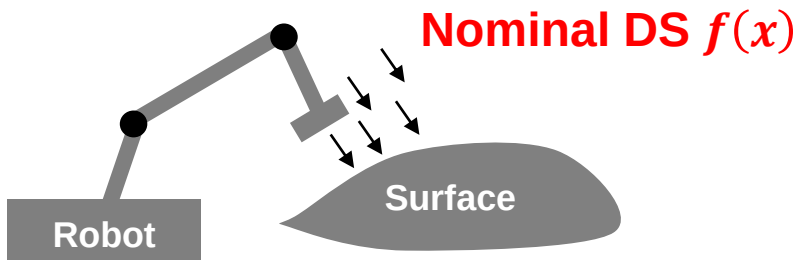
Controlling Robot's Forces when in Contact



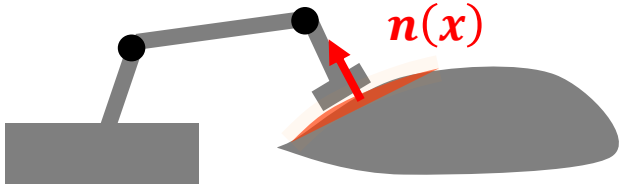
Controlling Robot's Forces when in Contact



Controlling for force at contact



Controlling for force at contact: Principle

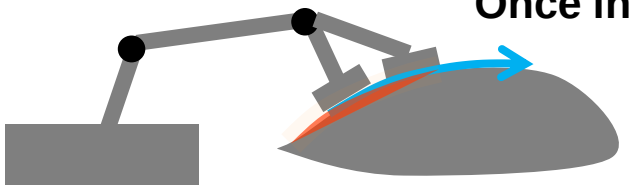


n(x) normal vector to the surface

Make contact

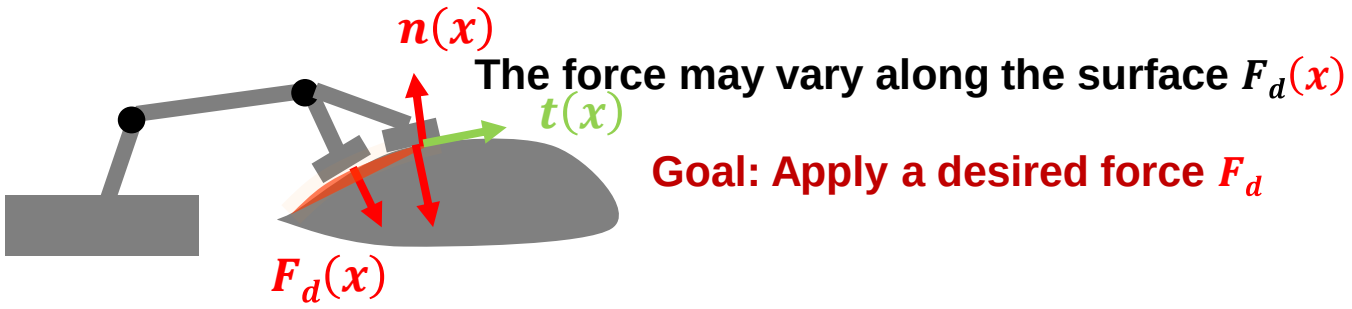
Assumption: Surface is impenetrable

Controlling for force at contact: Principle



Once in contact, move tangential to the surface

Controlling for force at contact: Principle

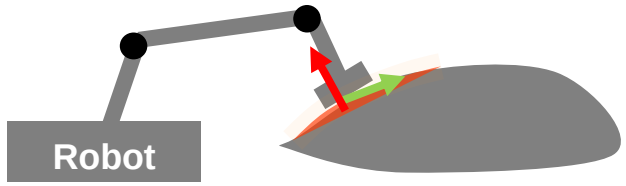


Idea: If we can project the control torques onto the surface, we simplify the computation.

→ We assume that we can compute the normal to the surface when in contact and we project the control onto a frame of reference moving along the surface in direction of motion.

Passive-DS for Controlling Forces on the Surface

To generate forces, we need to control the robot's torques. We use the passive DS approach.



Position controlled direction

Force controlled direction

$$\tau_c = -D(x)(\dot{x} - \dot{x}_d)$$

Robot's control torques

$$\dot{x}_d = \underbrace{f(x)}_{\text{Reach/Move on the surface}} + \underbrace{f_n(x)}_{\text{Apply the contact force}}$$

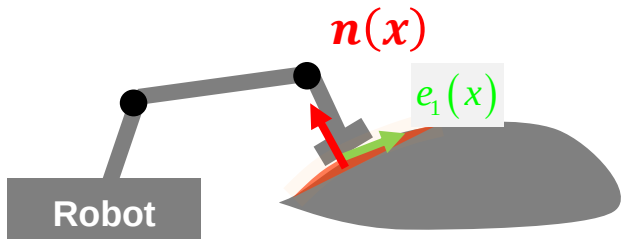
To separate control of force and control of motion , we decompose the nominal DS into two components:

$$\dot{x}_d = f(x) + f_n(x)$$

Force is applied only once in contact, hence $f_n(x) = 0$ in free space

Passive-DS for Controlling Forces on the Surface

To generate forces, we need to control the robot's torques. We use the passive DS approach.



Position controlled direction

Force controlled direction

$$\tau_c = \lambda_1 f(x) + \lambda_1 f_n(x) - D(x)\dot{x}$$

Robot's control torques

Eigencomposition of $D(x)$

$$D(x) = Q(x)\Lambda(x)Q(x)^T$$

$$Q(x) = \left[e_1(x)\ e_2(x)\right]$$

$$e_1(x) = \frac{f(x)}{\|f(x)\|}$$

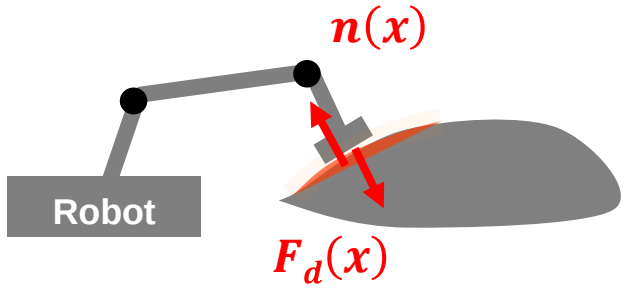
$$e_2(x) = n(x)$$

Fixed impedance

$$\Lambda = \begin{bmatrix} \lambda_1 & \\ & \lambda_2 \end{bmatrix}$$

Passive-DS for Controlling Forces on the Surface

To generate forces, we need to control the robot's torques. We use the passive DS approach.



Force controlled direction

$$\tau_c = \lambda_1 f(x) + \underbrace{\lambda_1 f_n(x)}_{F_d(x)n(x)} - D(x)\dot{x}$$

Robot's control torques

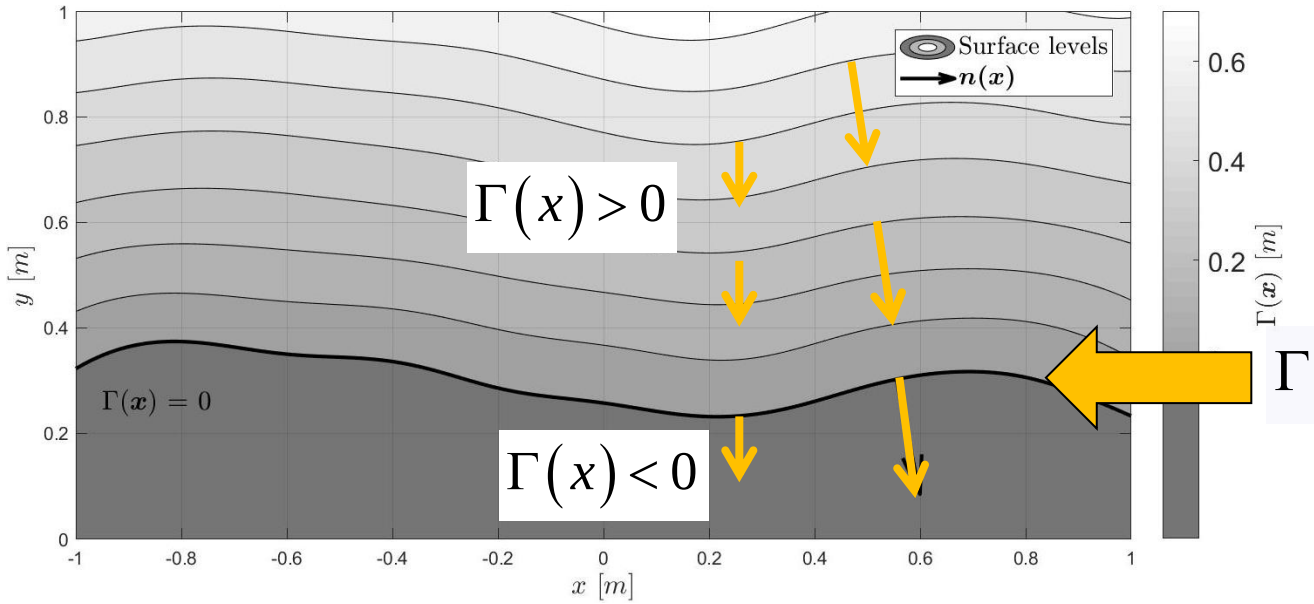
Design to apply the desired contact force

$$\text{set } f_n(x) = \frac{F_d(x)}{\lambda_1} n(x)$$

Modeling the Surface

We model the surface through a continuous function $\Gamma(x)$, that measures the distance to the surface.

$\Gamma(x)$ is continuously differentiable such that we can compute the normal vector $n(x)$ at any position x



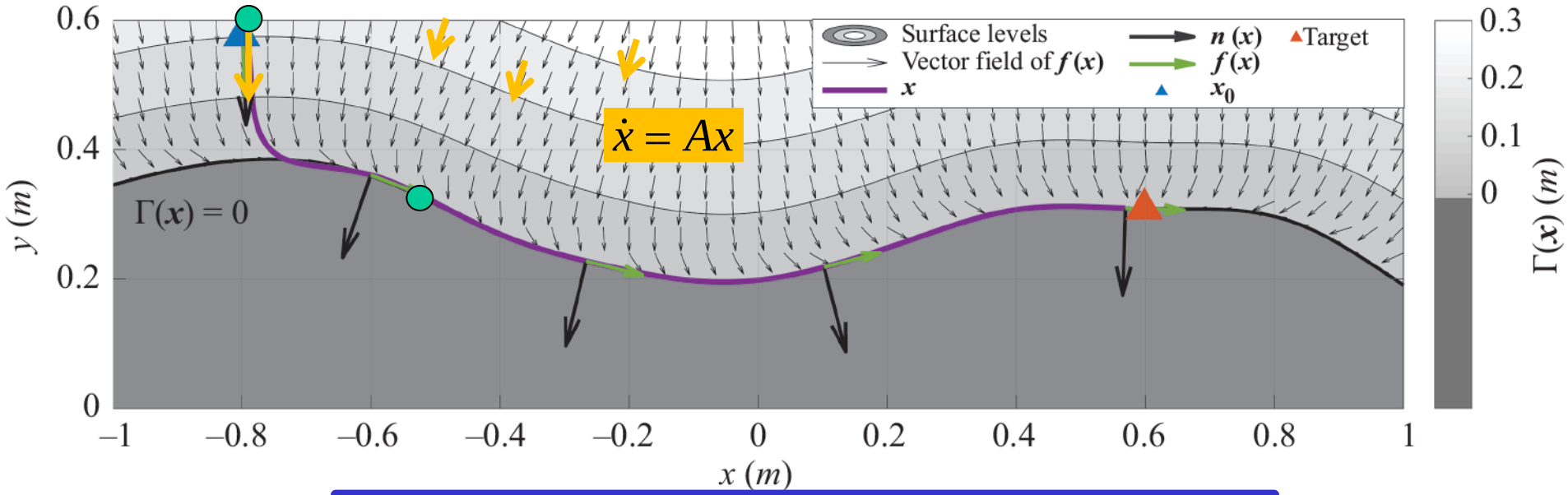
$\Gamma(x) = 0$ (on the surface)

Nominal DS

We define a linear DS moving downwards towards the surface.

We modulate this DS to force it to move along the surface.

To stop the motion at a target on the surface, we can set this as the attractor of the DS.



$f(x) = R(x)\,Ax$

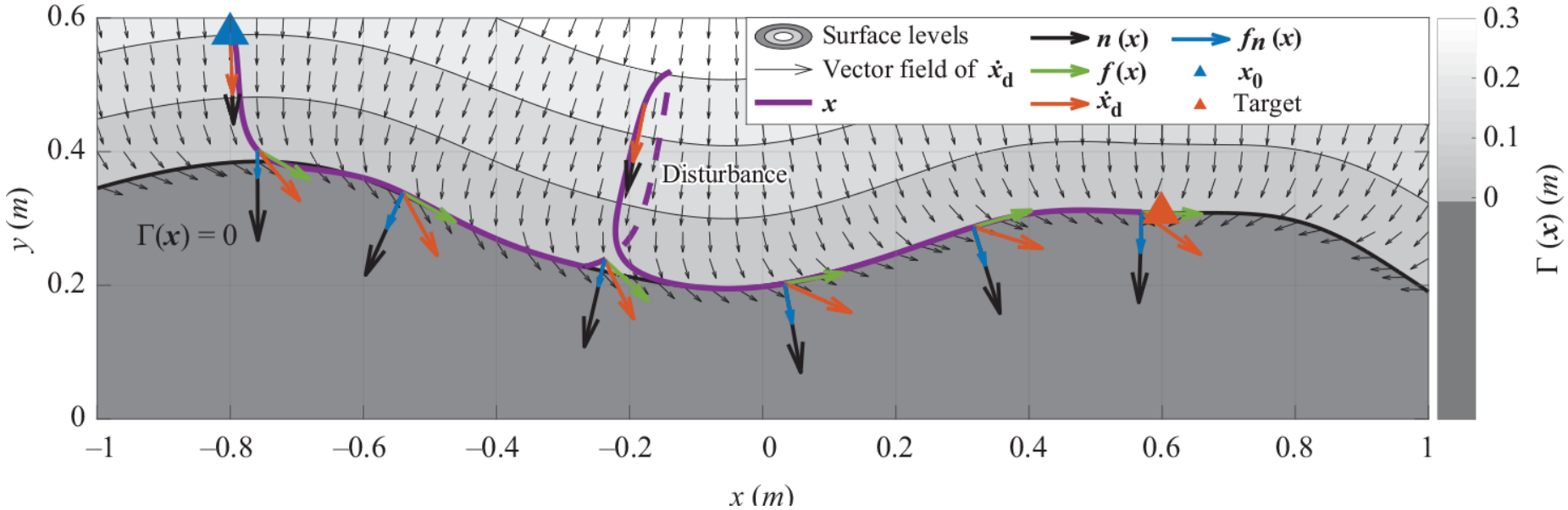
$R(x)$: Rotation to align to the surface once in contact

(same as in obstacle avoidance with constant velocity)

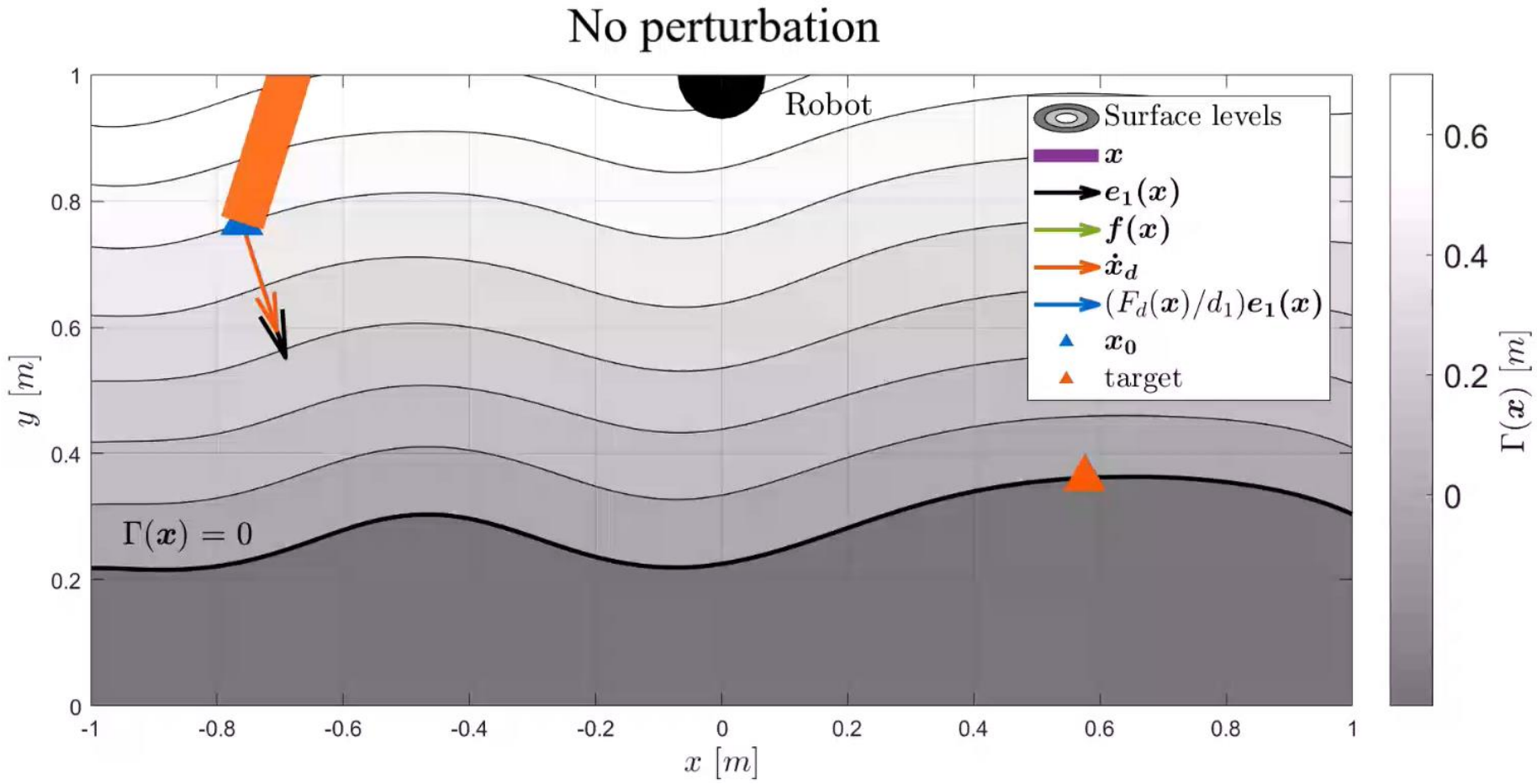
Visualization of the Decomposition

To separate control of force and control of motion , we decompose the nominal DS into two components:

$$\dot{x}_d = f(x) + f_n(x) \qquad f_n(x) = 0 \quad (\text{in free space})$$

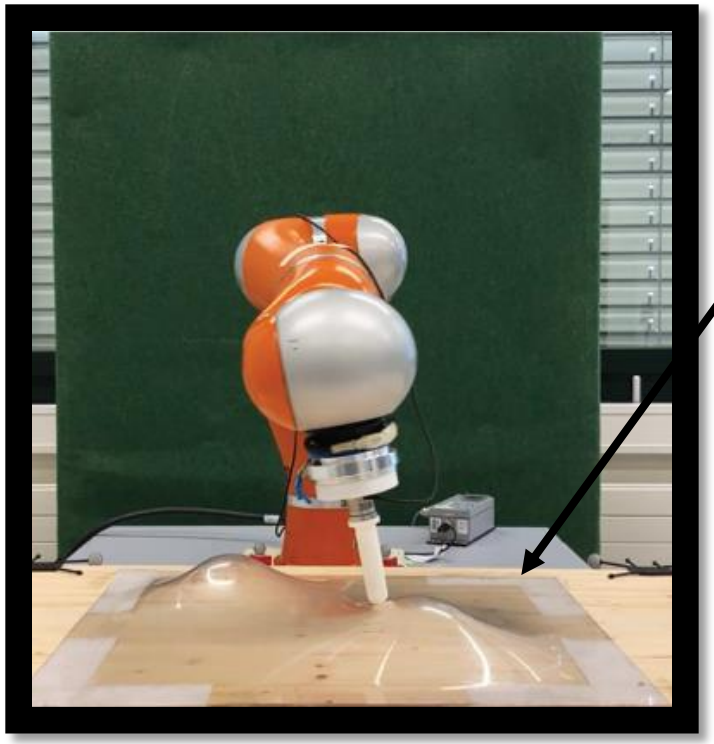


Animation of Principle



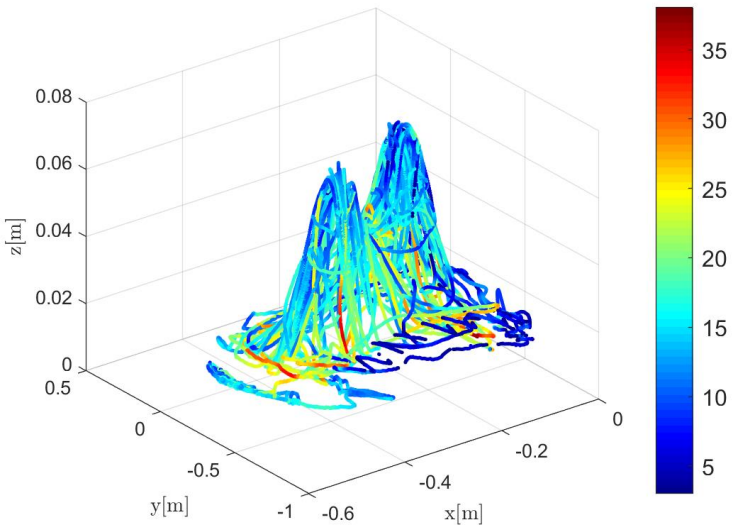
Robotic Demonstration

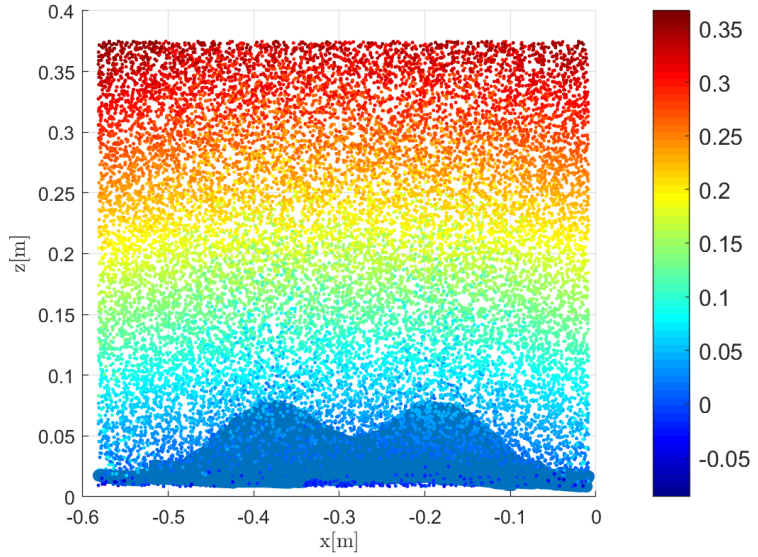
Task: The robot must polish in circular motion the surface applying a constant force of 20N.



Learn a model of the surface
using Support Vector Regression

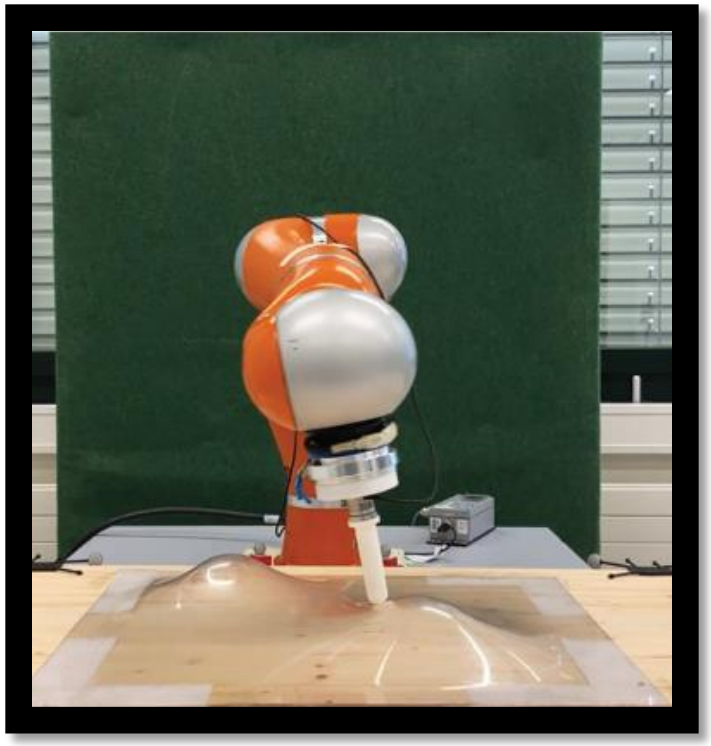
Input dataset to learn the surface model:



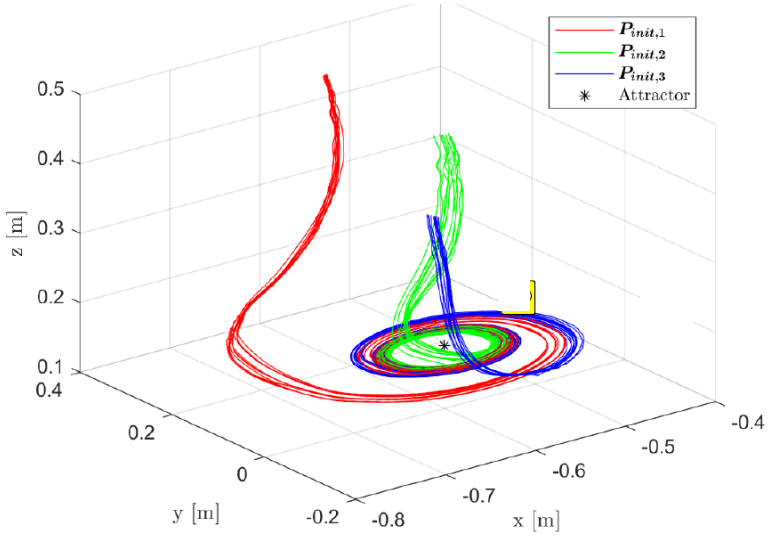


Robotic Demonstration

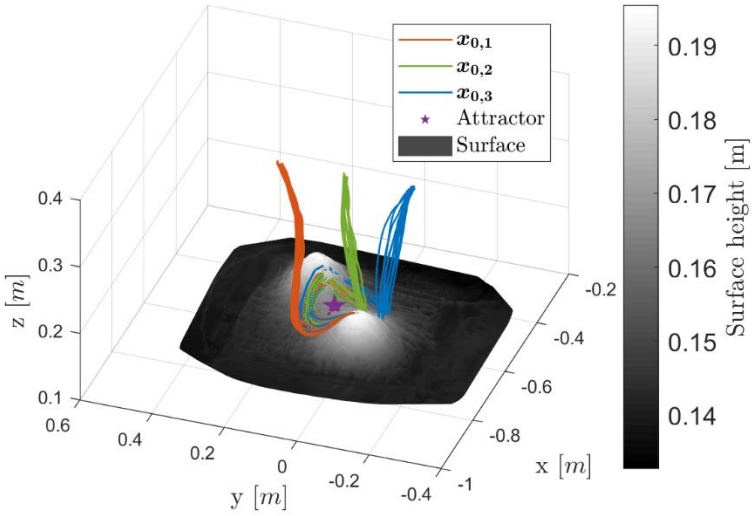
Task: The robot must polish in circular motion the surface applying a constant force of 20N.



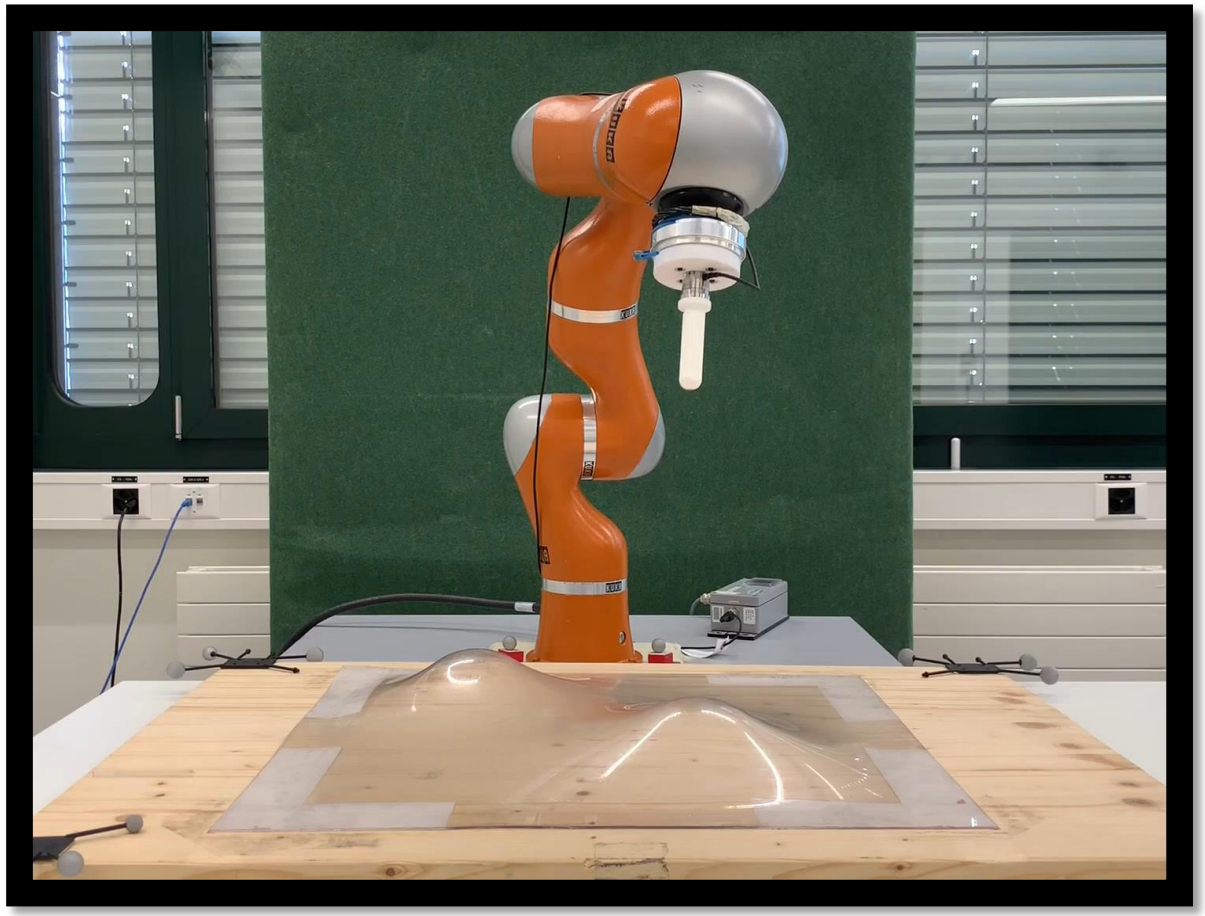
Define a nominal DS that creates a limit cycle on the surface.

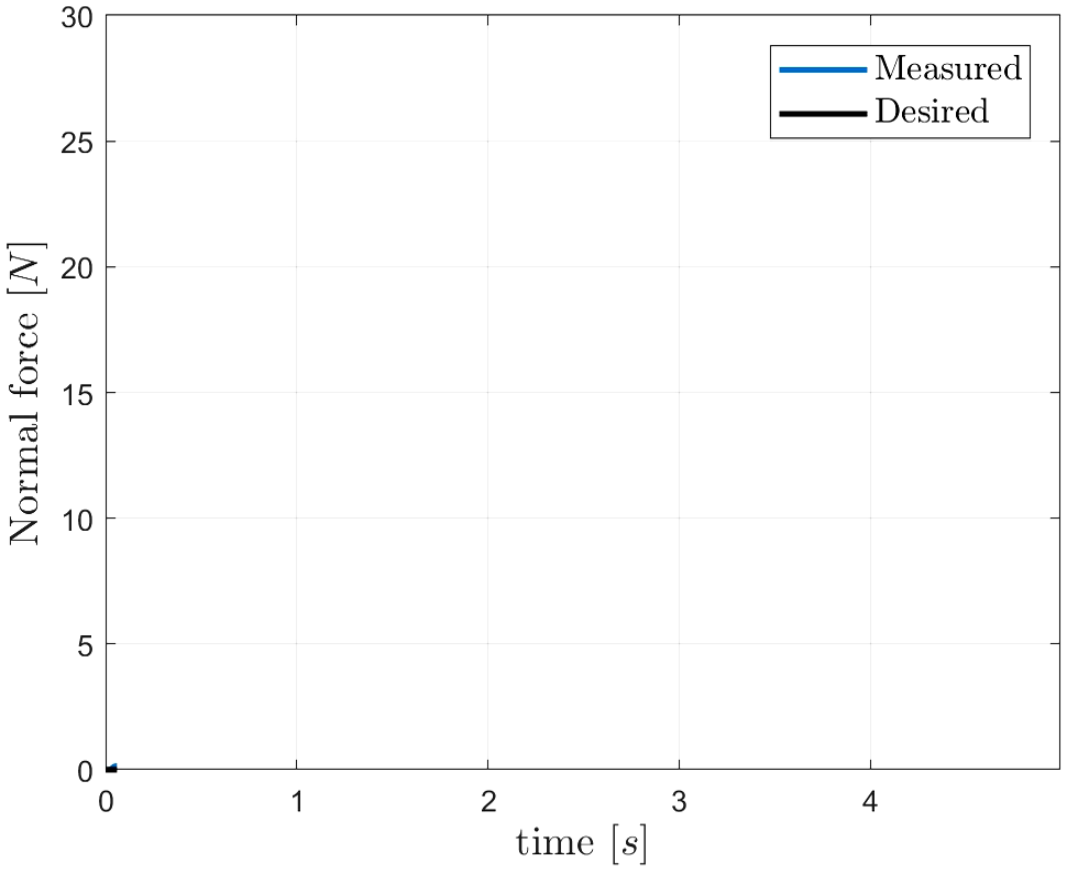


Project on the surface.



Robotic Demonstration – Robustness to Various Disturbances

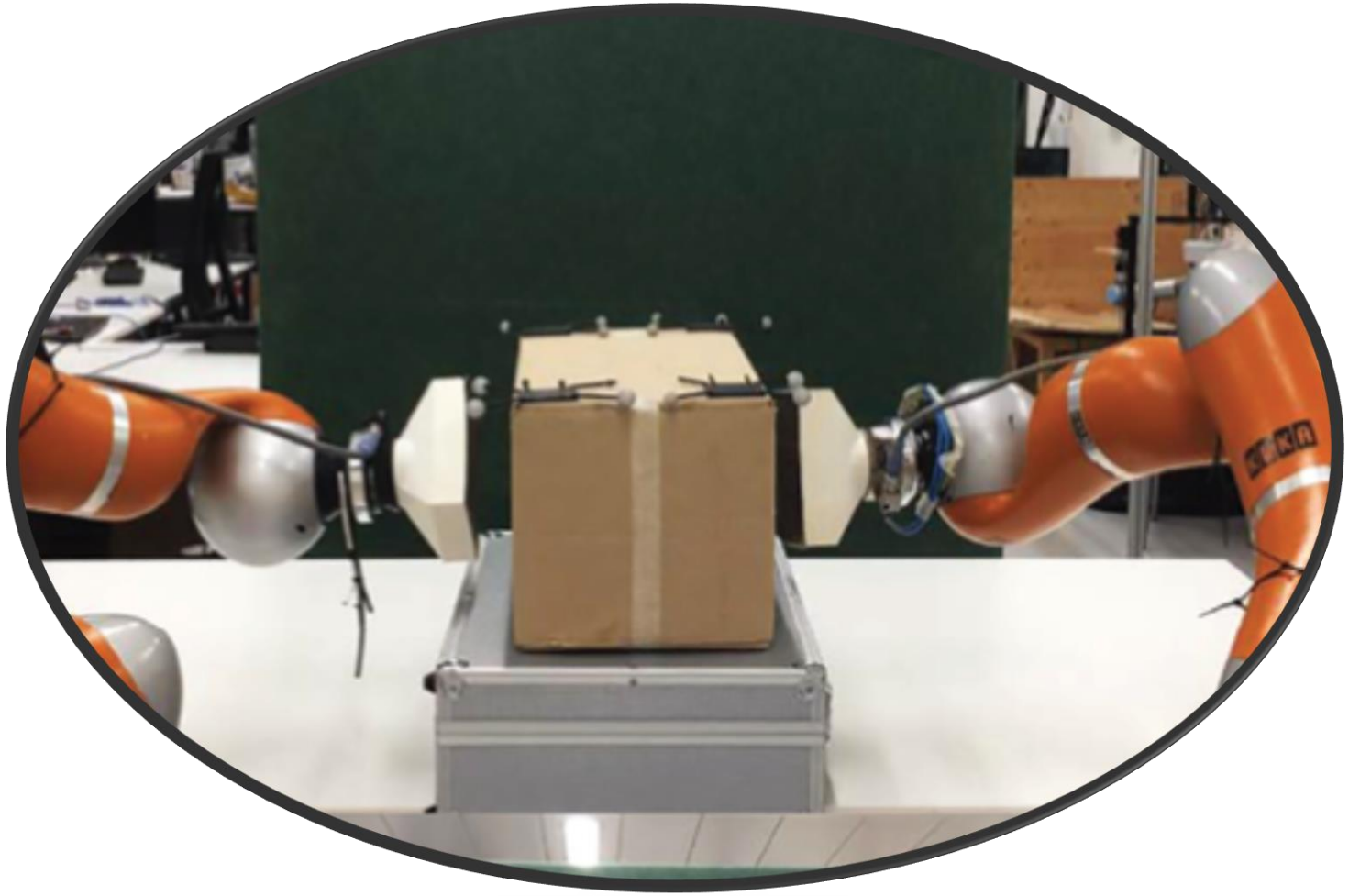




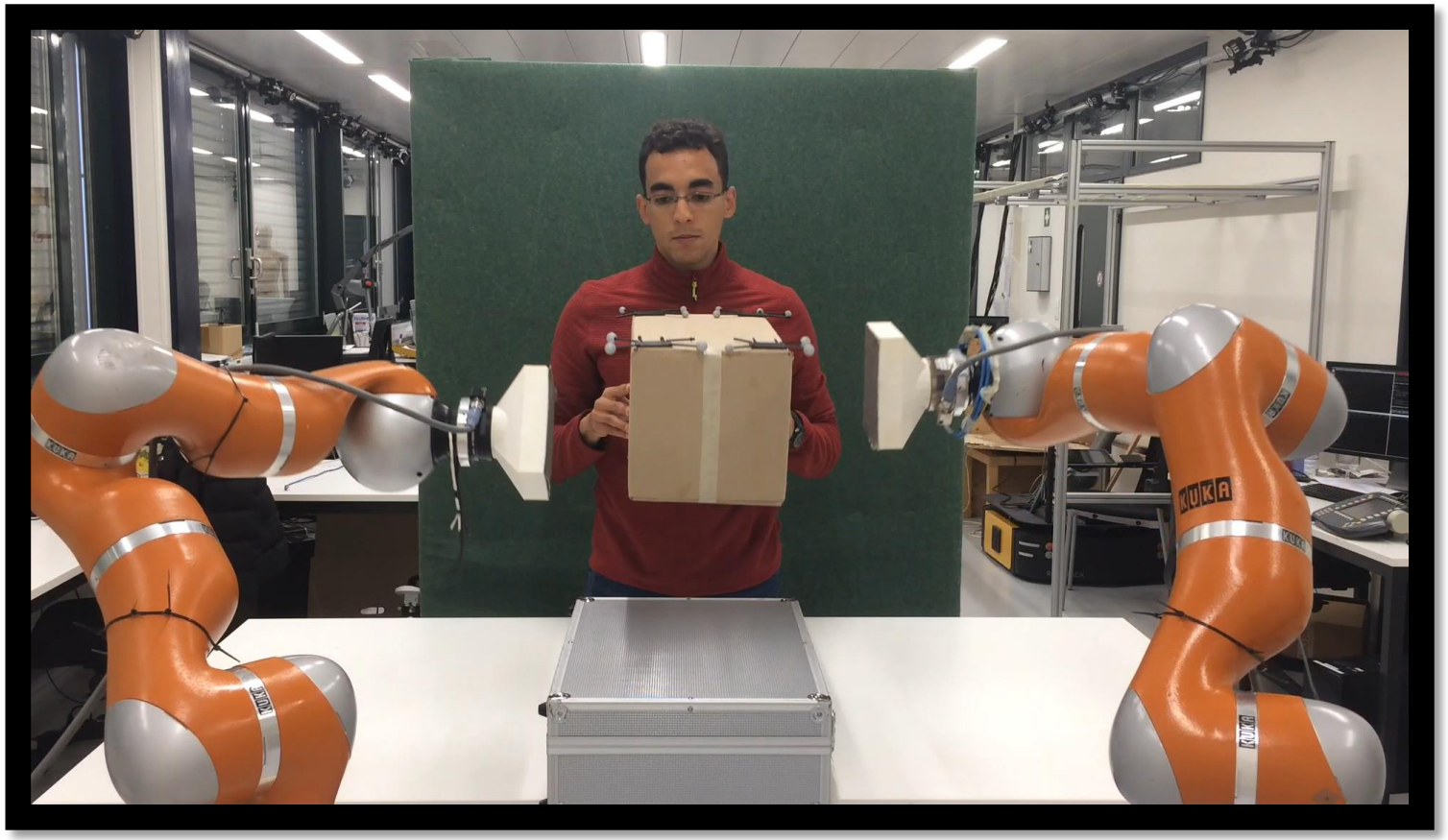
Passivity see exercise session

Extension to Control Bimanual Platform

Task: The robots must reach either side of the box and apply enough force to support the box's weight.



Extension to Control Bimanual Platform



Variables to Control Bimanual Platform

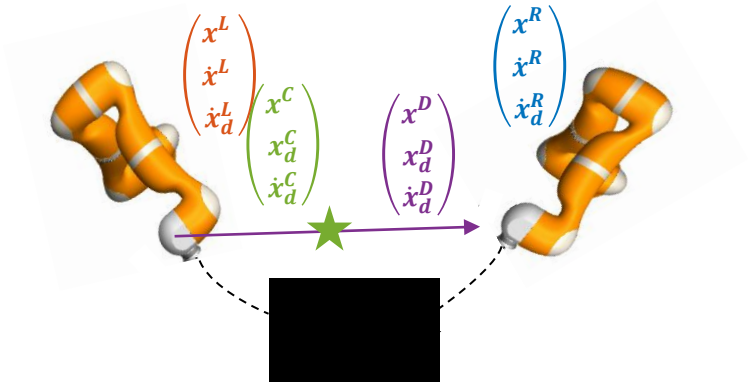
- **Robots' center position and distance vector**

$$\begin{cases} x^C = \dfrac{x^L + x^R}{2} \\ x^D = x^R - x^L \end{cases}$$

- **Robots' nominal dynamics**

$$\begin{cases} f^R(x^C, x^D) = \dot{x}_d^C + \dfrac{\dot{x}_d^D}{2} \\ f^L(x^C, x^D) = \dot{x}_d^C + \left(-\dfrac{\dot{x}_d^D}{2}\right) \end{cases}$$

To simplify control and ensure coordination, compute control in relative position



Positioning + grasping dynamics

Nominal Dynamics for Bimanual Platform

- **Desired** robots' center position and distance vector **dynamics:**

$$\begin{cases} \dot{x}_d^C = A_C\left(x_d^C - x^C\right) \\ \dot{x}_d^D = A_D\left(x_d^D - x^D\right) \end{cases}$$

Center positioning dynamics

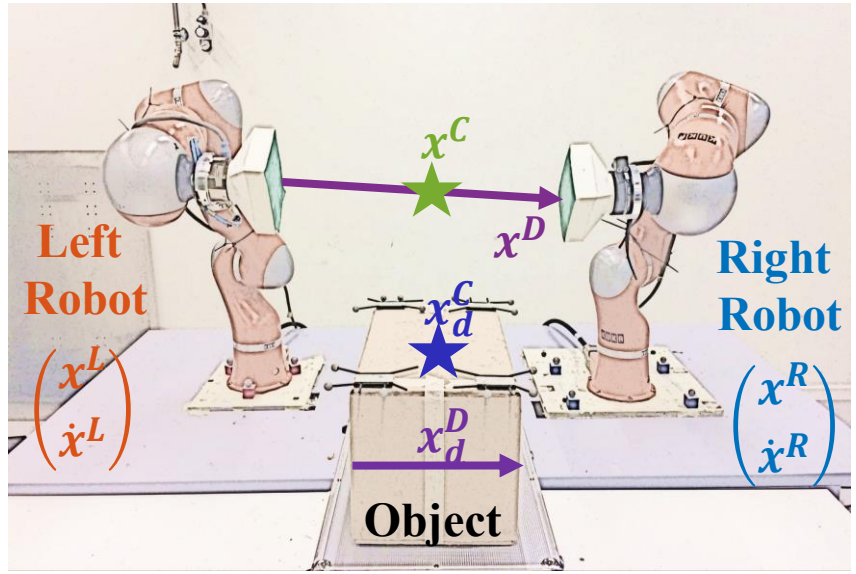
Closing dynamics

with: $A_C, A_D \geq 0$

- Robots' **nominal dynamics:**

$$\begin{cases} f^R\left(x^L, x^R\right) = \dot{x}_d^C + \dfrac{\dot{x}_d^D}{2} \\ f^L\left(x^L, x^R\right) = \dot{x}_d^C + \left(-\dfrac{\dot{x}_d^D}{2}\right) \end{cases}$$

Center positioning + Closing dynamics



Force Desired for Bimanual Platform

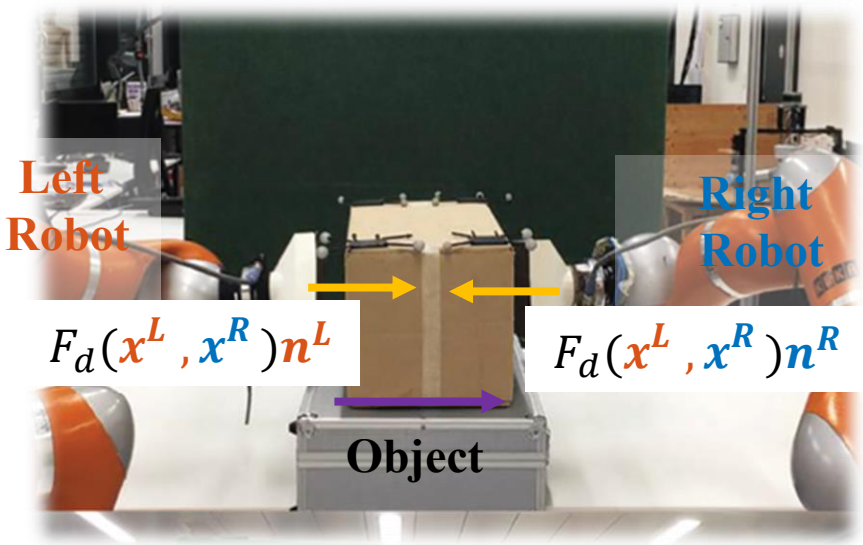
- Robots' **desired modulated dynamics:**

$$\begin{cases} \dot{x}_d^R = f^R(x^L, x^R) + \dfrac{F_d(x^L, x^R)}{\lambda_1^R} n^R \\ \dot{x}_d^L = f^L(x^L, x^R) + \dfrac{F_d(x^L, x^R)}{\lambda_1^L} n^L \end{cases}$$

with: $n^R = -n^L = \dfrac{x_d^D}{\|x_d^D\|}$ **Grasping direction**

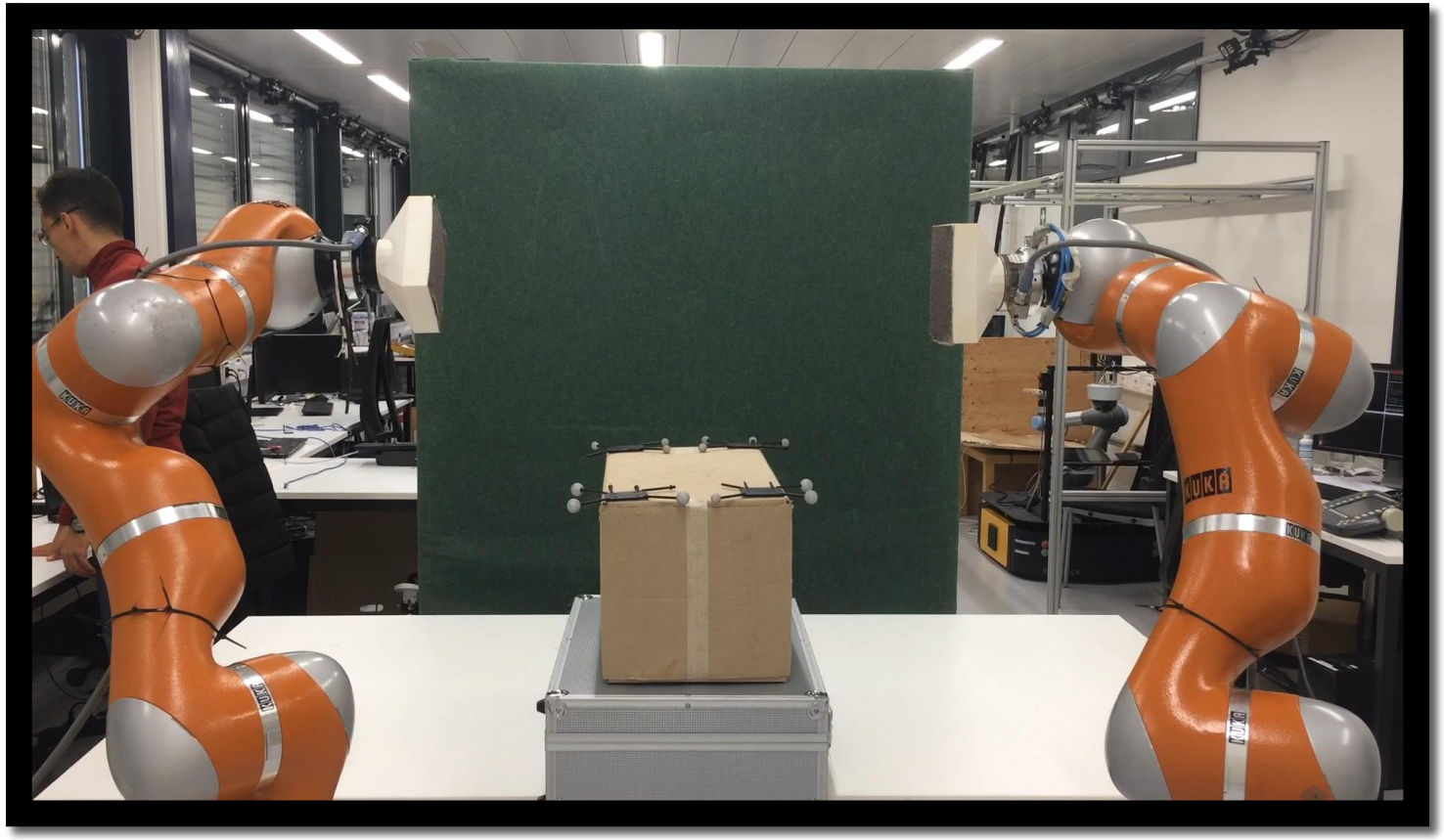
$\bar{F}_d(x^L, x^R)$: **Desired contact force**

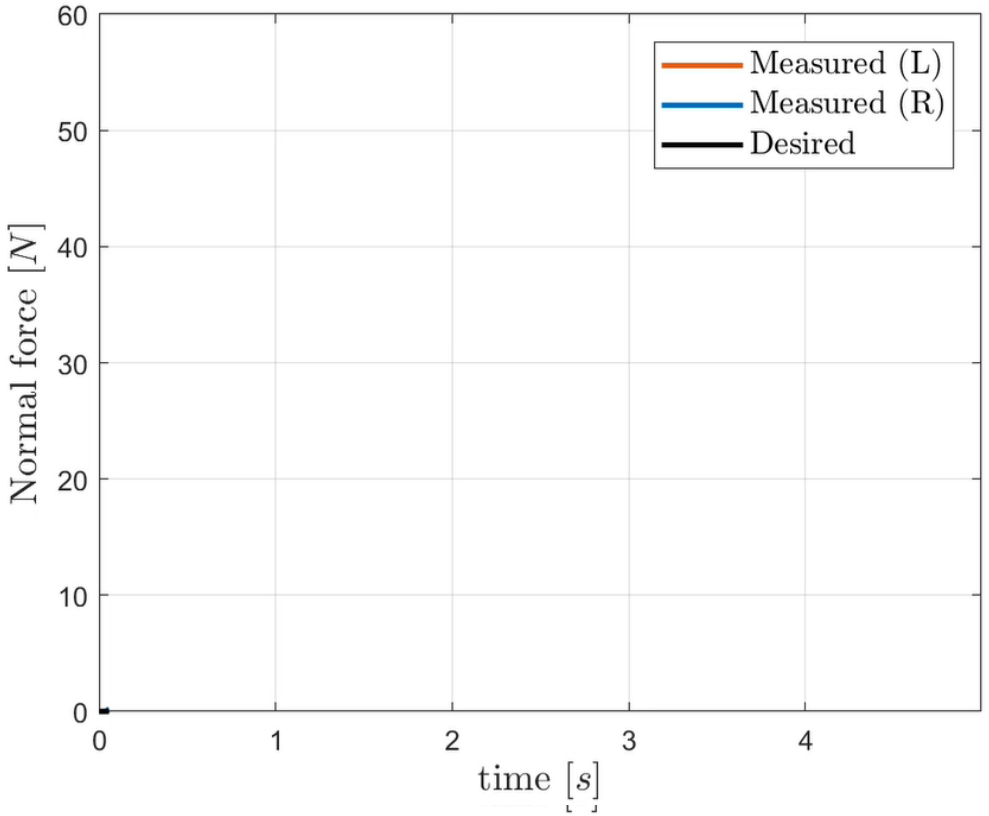
λ_1^L, λ_1^R: **Impedance gains**



- Ensures the passivity through a tank for energy of both arms

Robot Demonstration

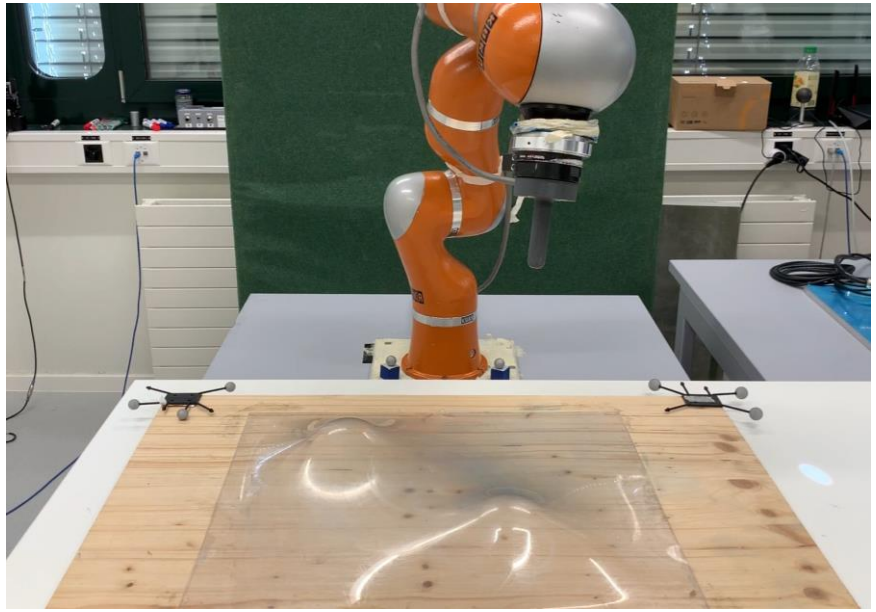




Learning Force Adaptation

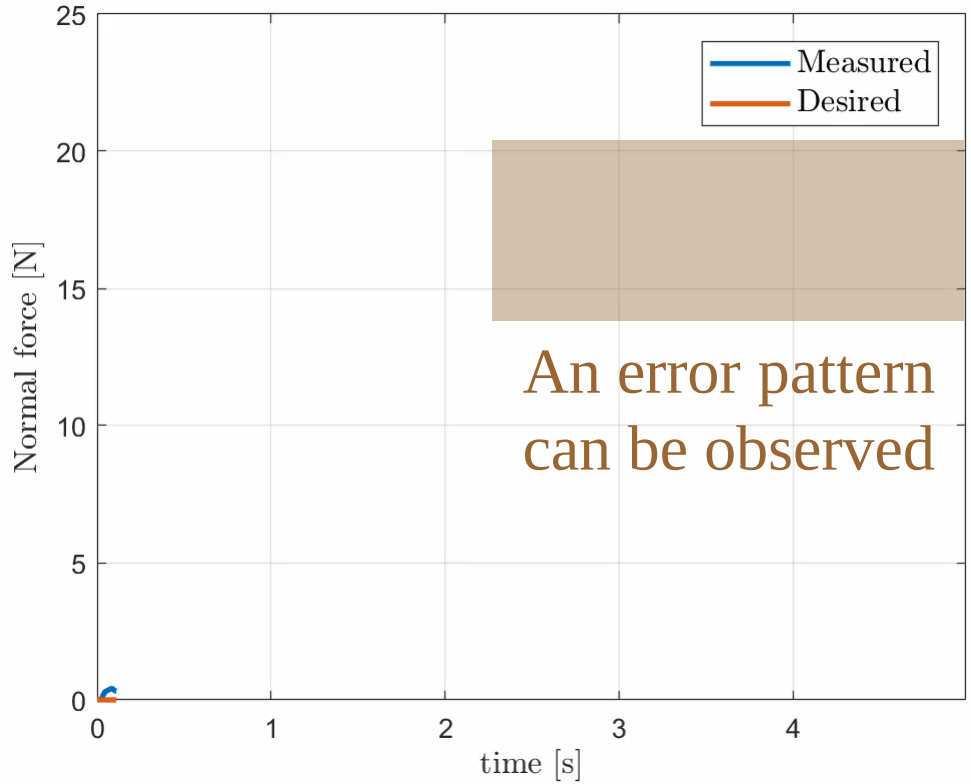
Amanhoud, W., Khoramshahi, M., Bonnesoeur, M. and Billard, A., 2020, May. Force adaptation in contact tasks with dynamical systems. In *2020 IEEE International Conference on Robotics and Automation (ICRA)* (pp. 6841-6847).

Imprecise Force Generation



Force tracking errors result from:

- Uncertainties in the surface
- Uncertainties in the robot model
- Measurement noises



Goal: Exploit the **adaptability** and **robustness** of the time-invariant **DS framework to learn force compensation models**

State-Dependent Force Correction

❖ Force generation with DS:

$$\dot{\boldsymbol{x}}_{\boldsymbol{d}} = \boldsymbol{f}(\boldsymbol{x}) + \boldsymbol{f}_{\boldsymbol{n}}(\boldsymbol{x})$$

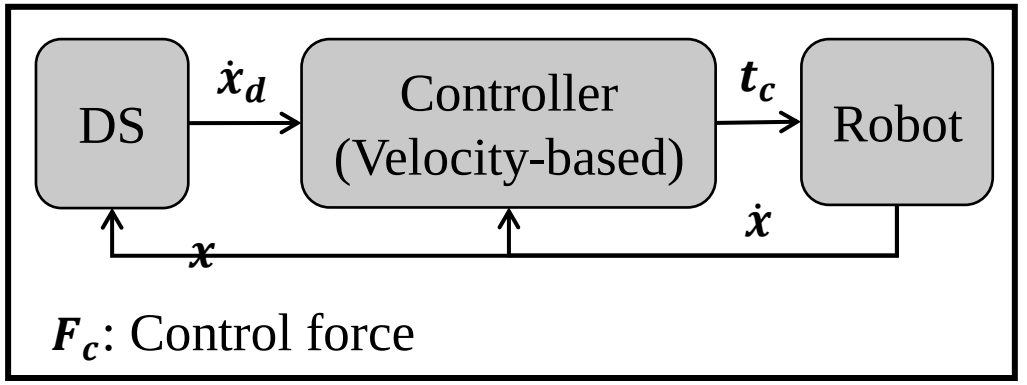
Nominal DS (responsible for motion)

Modulation term (responsible for contact force)

❖ Introduction of a state-dependent force correction model $\tilde{F}_d(\boldsymbol{x}, \boldsymbol{\theta})$:

$$\boldsymbol{f}_{\boldsymbol{n}}(\boldsymbol{x}) = \frac{F_d(\boldsymbol{x}) + \tilde{F}_d(\boldsymbol{x}, \boldsymbol{\theta})}{\lambda_1}\boldsymbol{n}(\boldsymbol{x})$$

DS in closed-loop configuration

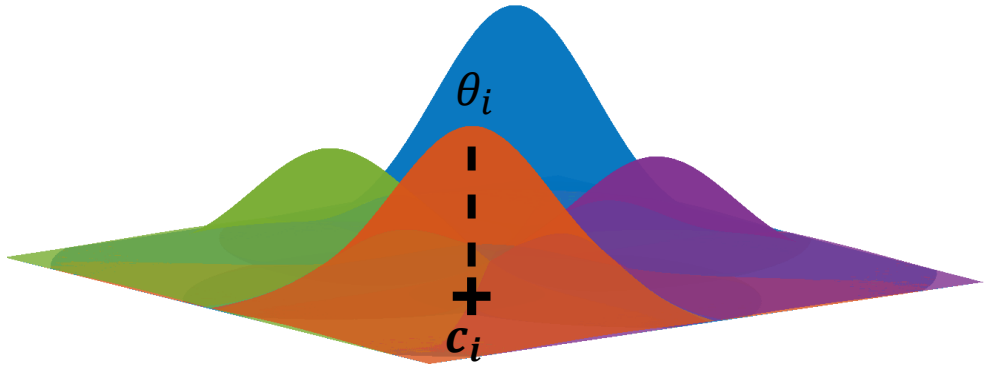


$\dot{\boldsymbol{x}}_{\boldsymbol{d}} \in \mathbb{R}^3$	Desired dynamics
$\boldsymbol{f}(\boldsymbol{x}) \in \mathbb{R}^3$	Nominal DS
$\boldsymbol{f}_{\boldsymbol{n}}(\boldsymbol{x}) \in \mathbb{R}^3$	Normal modulation term
$F_d(x) \geq 0$	Desired force profile
$\boldsymbol{n}(\boldsymbol{x}) \in \mathbb{R}^3$	Normal direction to the surface
$\lambda_1 \geq 0$	Impedance gain

State-Dependent Force Correction

❖ Design $\tilde{F}_d(\boldsymbol{x}, \boldsymbol{\theta})$ using Gaussian Radial Basis kernel functions (RBFs):

$$\tilde{F}_d(\boldsymbol{x}, \boldsymbol{\theta}) = \frac{\sum_{i=1}^{K} \theta_i \varphi(\boldsymbol{x} - \boldsymbol{c_i})}{\sum_{j=1}^{K} \varphi(\boldsymbol{x} - \boldsymbol{c_j})} \quad \text{with:} \quad \underbrace{\varphi(\boldsymbol{x}) = \exp^{\frac{-\|\boldsymbol{x}\|^2}{2\sigma^2}}}_{\textbf{Gaussian kernel}}$$



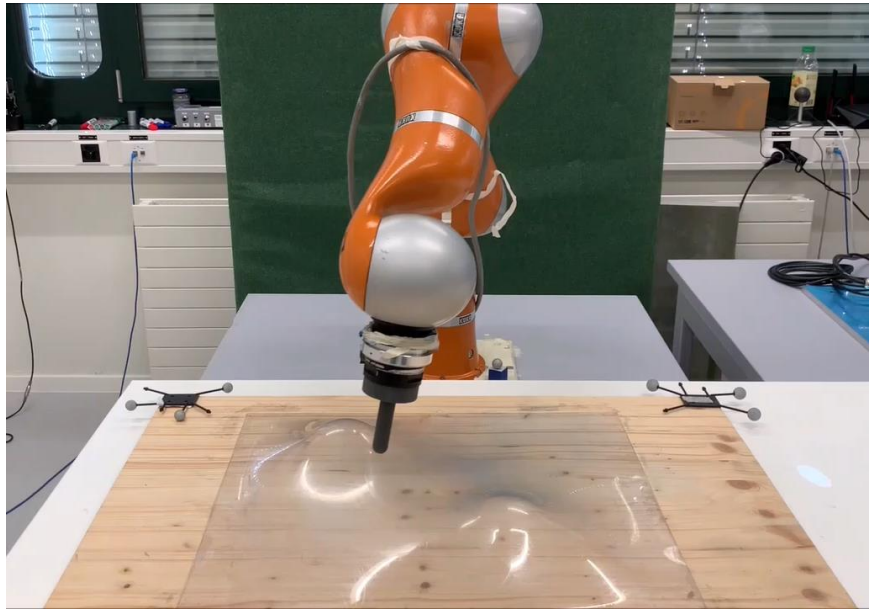
Hyper-parameters

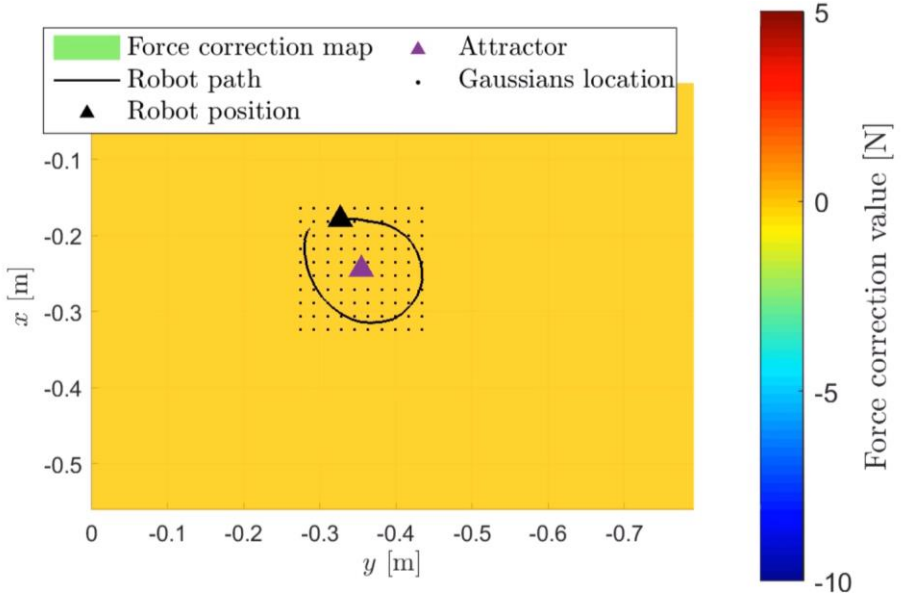
- K: Number of Gaussian
- $\boldsymbol{c_i}$: Center position of Gaussian i
- σ: Kernel width

❖ Update the weights $\boldsymbol{\theta}$ to minimize the normal force error F_e:

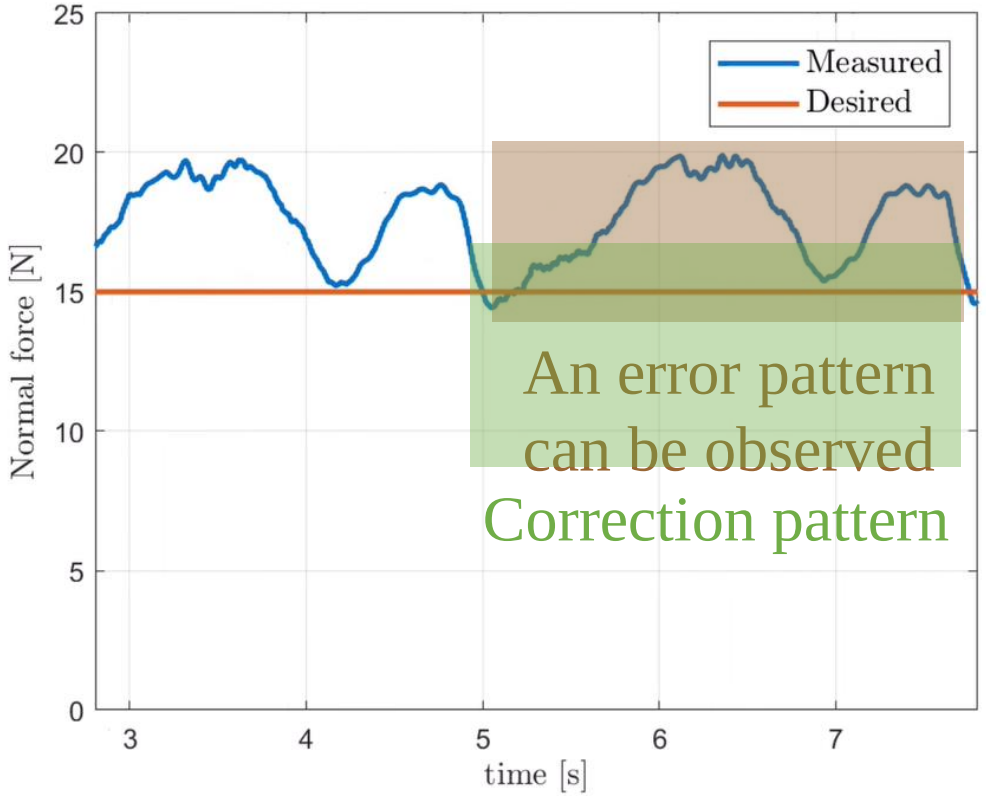
$$F_e = F_d(\boldsymbol{x}) - \boldsymbol{n}(\boldsymbol{x})^T \boldsymbol{F_m}$$

- $F_d(\boldsymbol{x})$: Desired contact force
- $\boldsymbol{n}(\boldsymbol{x})$: Normal vector to the surface
- $\boldsymbol{F_m}$: Measured contact forces

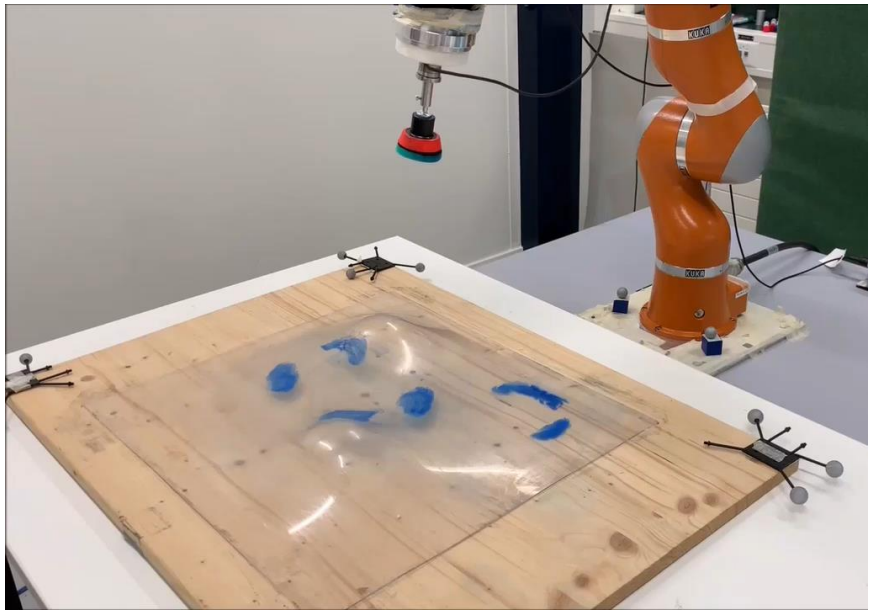


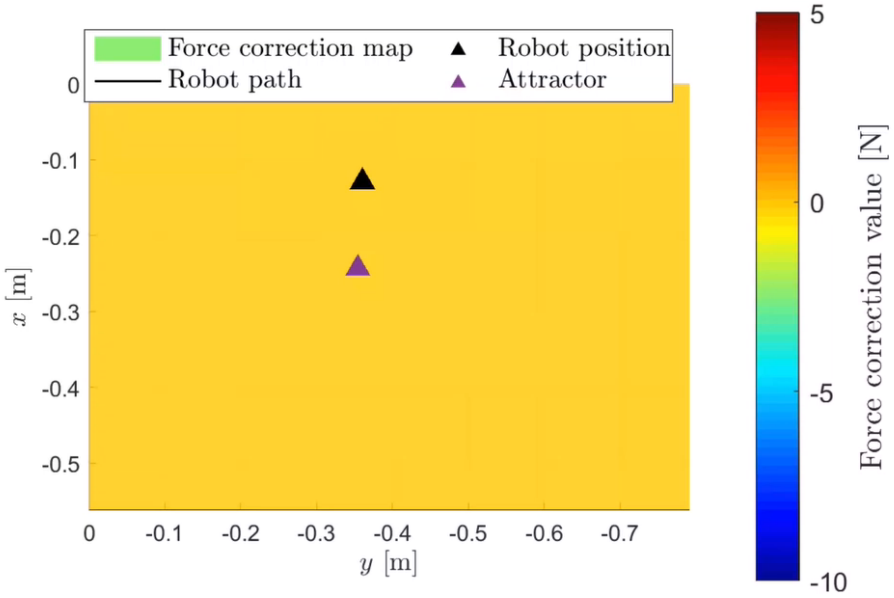


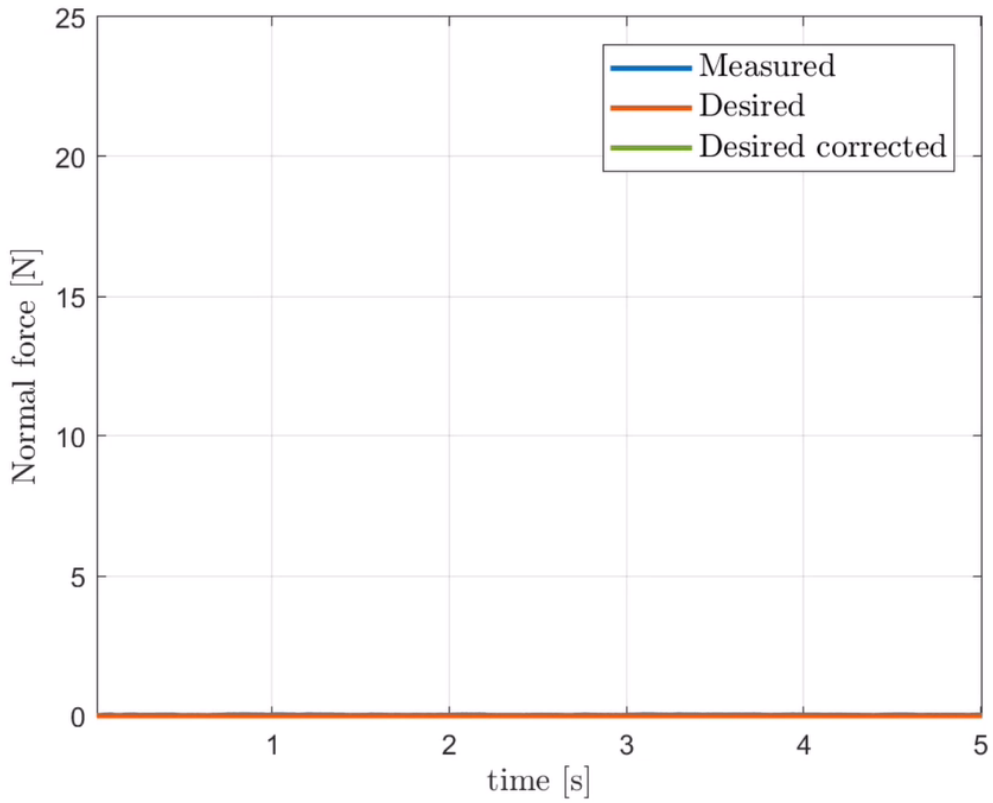
Initial error range



4x




Force correction map
Robot path
Robot position
Attractor
0
-0.1
-0.2
-0.3
-0.4
-0.5
x [m]
0
-0.1
-0.2
-0.3
-0.4
-0.5
-0.6
-0.7
y [m]
5
0
-5
-10
Force correction value [N]


Measured
Desired
Desired corrected
Normal force [N]
25
20
15
10
5
0
1
2
3
4
5
time [s]

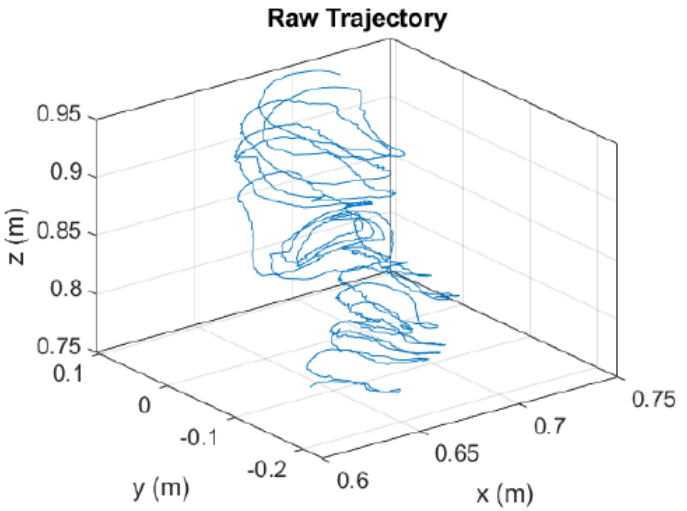
4x

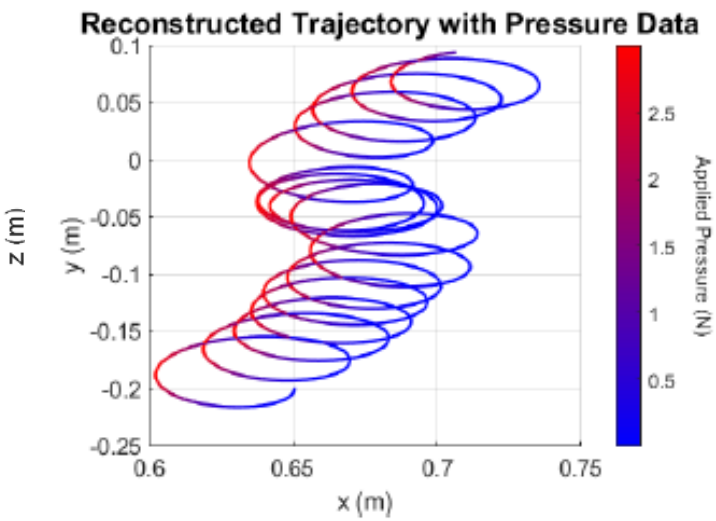
Other Examples of Learned-Force Based Control with DS

Khoramshahi, M., Henriks, G., Naef, A., Salehian, S.S.M., Kim, J. and Billard, A., 2020, May. Arm-hand motion-force coordination for physical interactions with non-flat surfaces using dynamical systems: Toward compliant robotic massage. In *2020 IEEE International Conference on Robotics and Automation (ICRA)* (pp. 4724-4730).

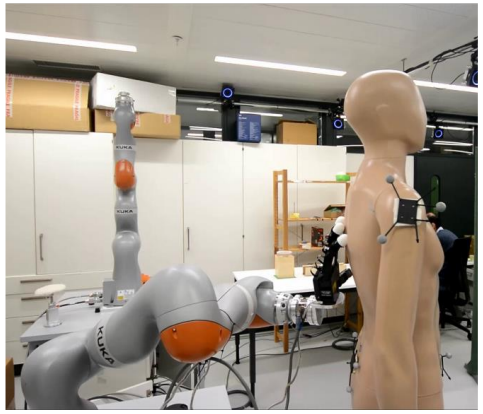
Example of application: Learning Motion and Force Pattern to Massage

Human recordings





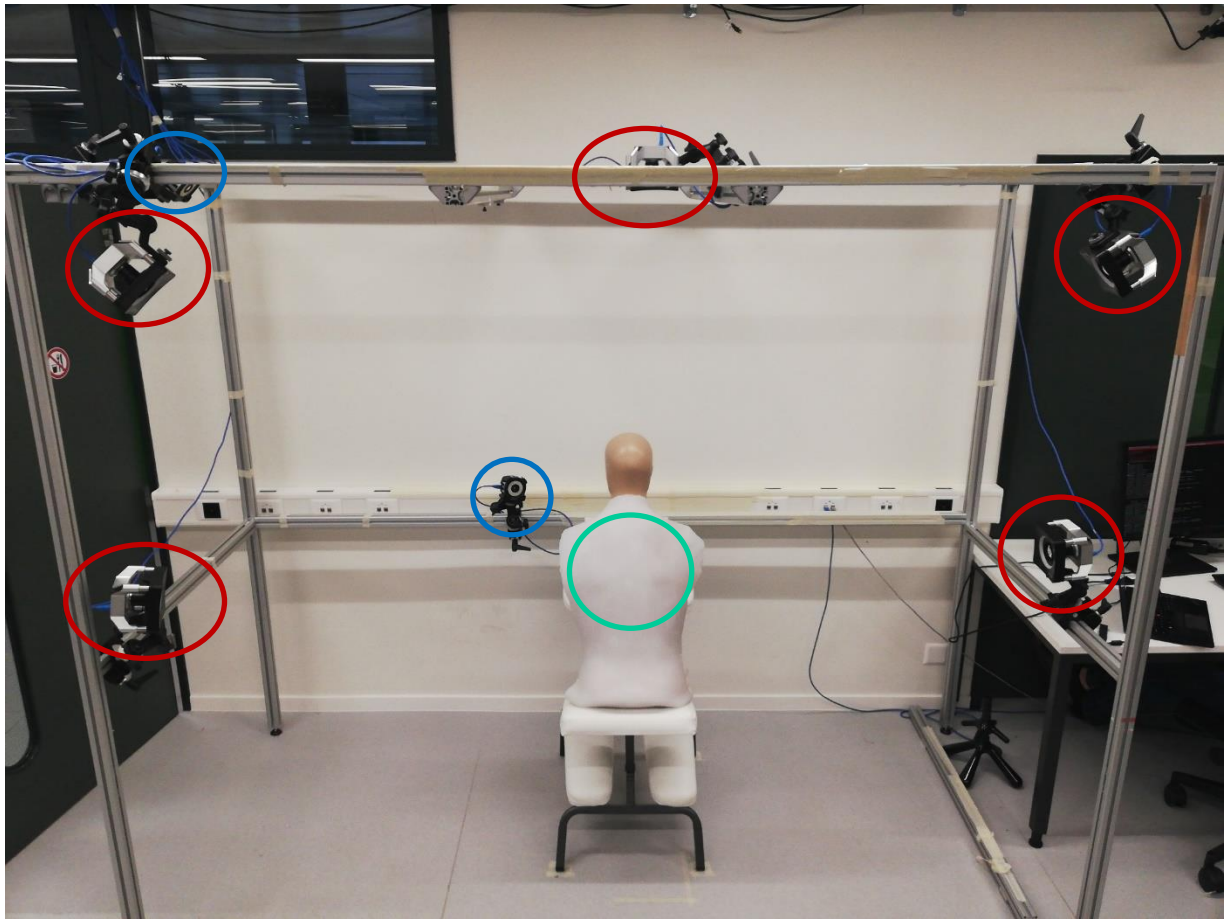
Modeling



Robotic massage

Supported by Samsung

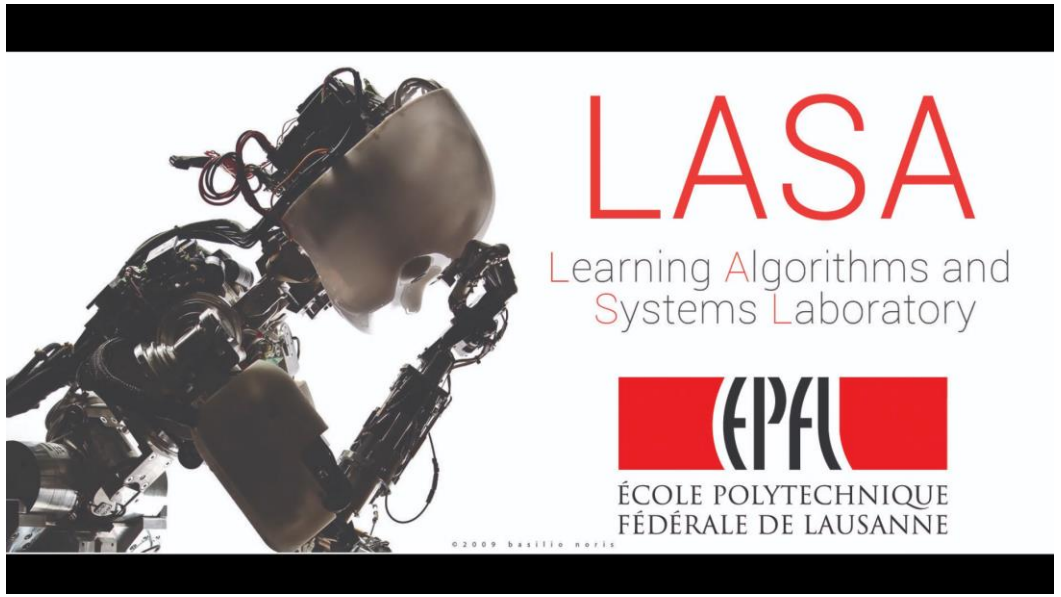
GATHERING DATA FOR TRAINING – EXPERIMENTAL SET-UP

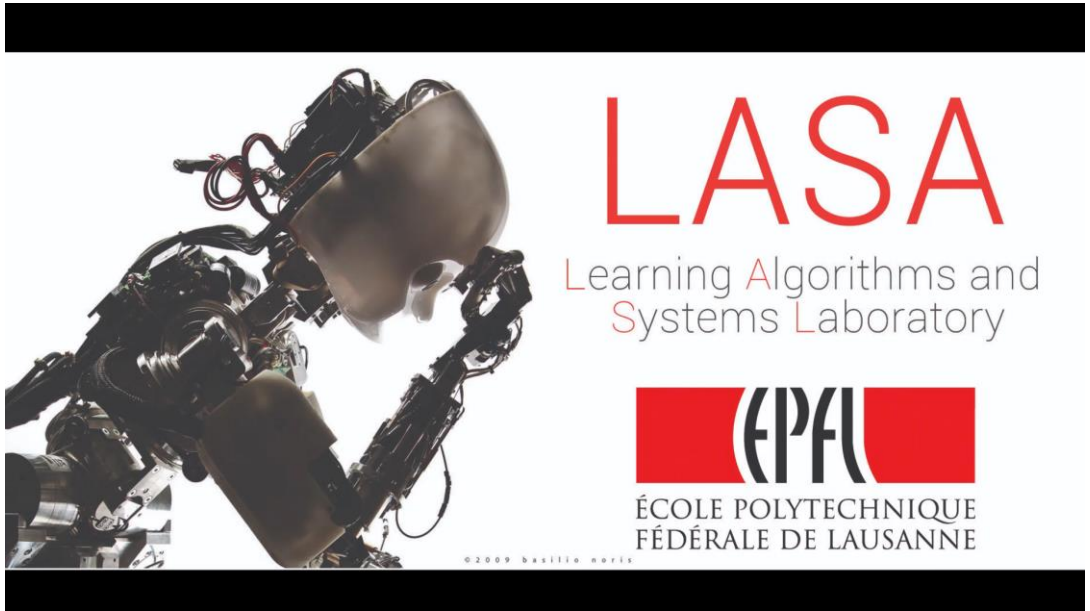


- Mannequin on massage chair
- Silicone sheet placed over the mannequin
  - Represent human tissue for the robot

- 7 Optitrack cameras
  - 5 x Prime 17w
  - 2 x S250:E
    - Limited to 125 Hz
- Infrared marker positions (X, Y, Z) captured

Demonstration by Massage Therapist



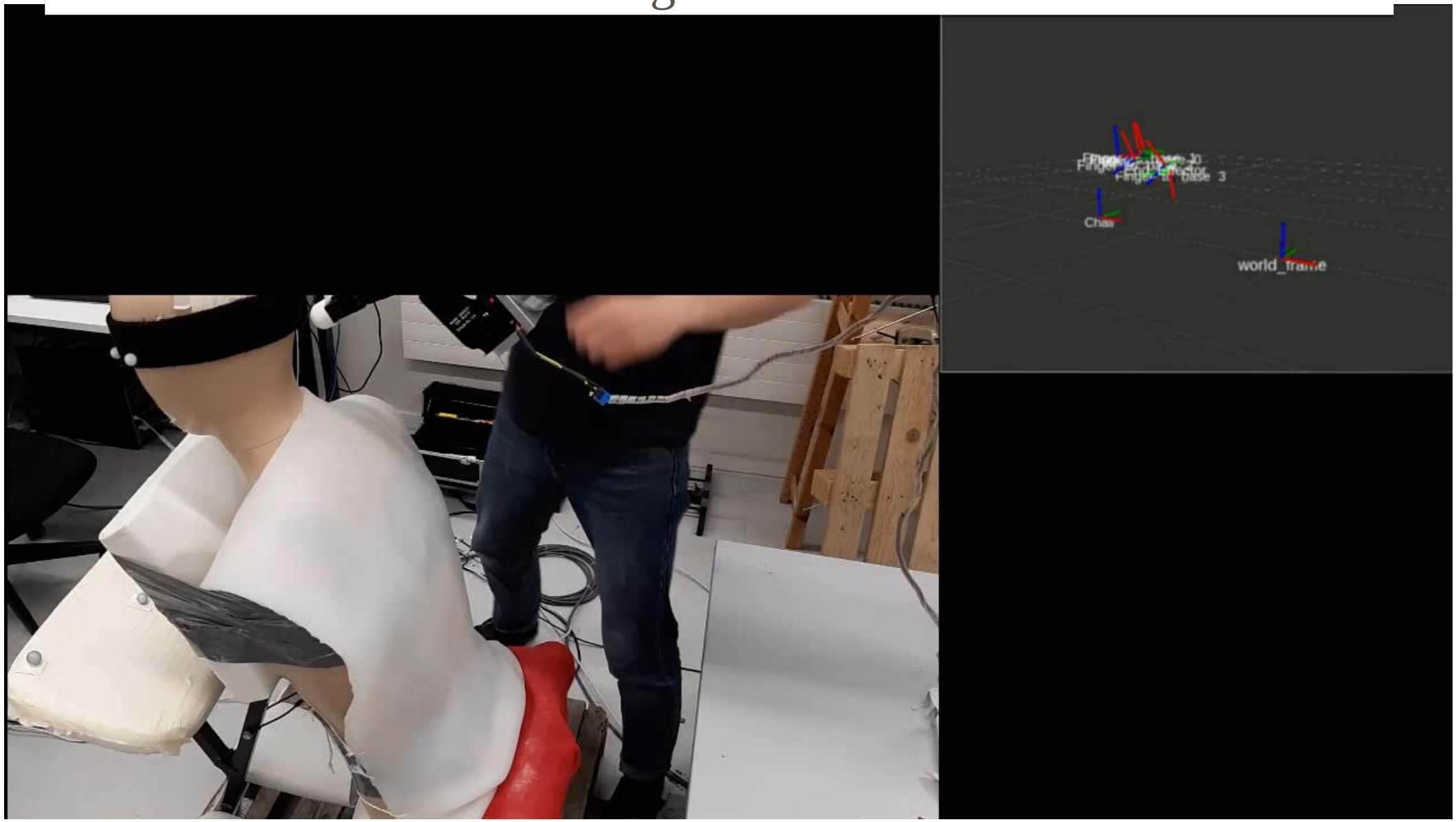


FORCE MEASUREMENT

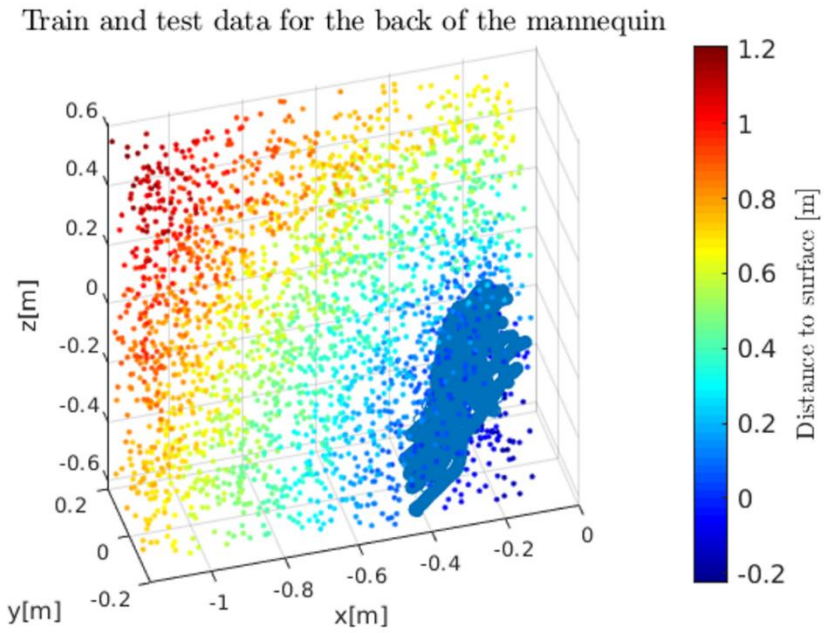
- Pressure Profiles Systems, FingerTPS

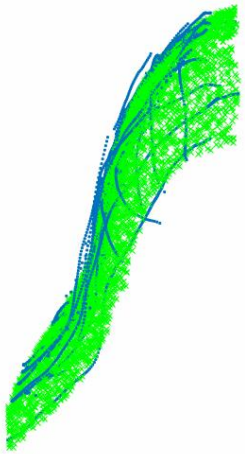
+ Sensors have multiple sensing areas
+ Multiple sizes
+ Fabric covered sensor
- Force mapped across entire surface
- 40 Hz sampling rate

Modeling Back Surface



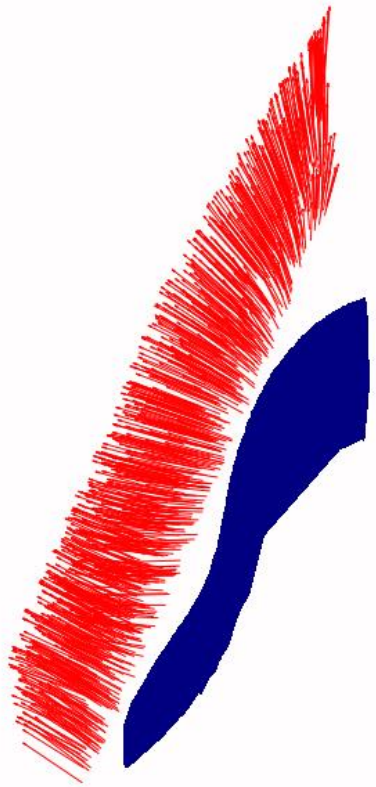
Modeling Back Surface

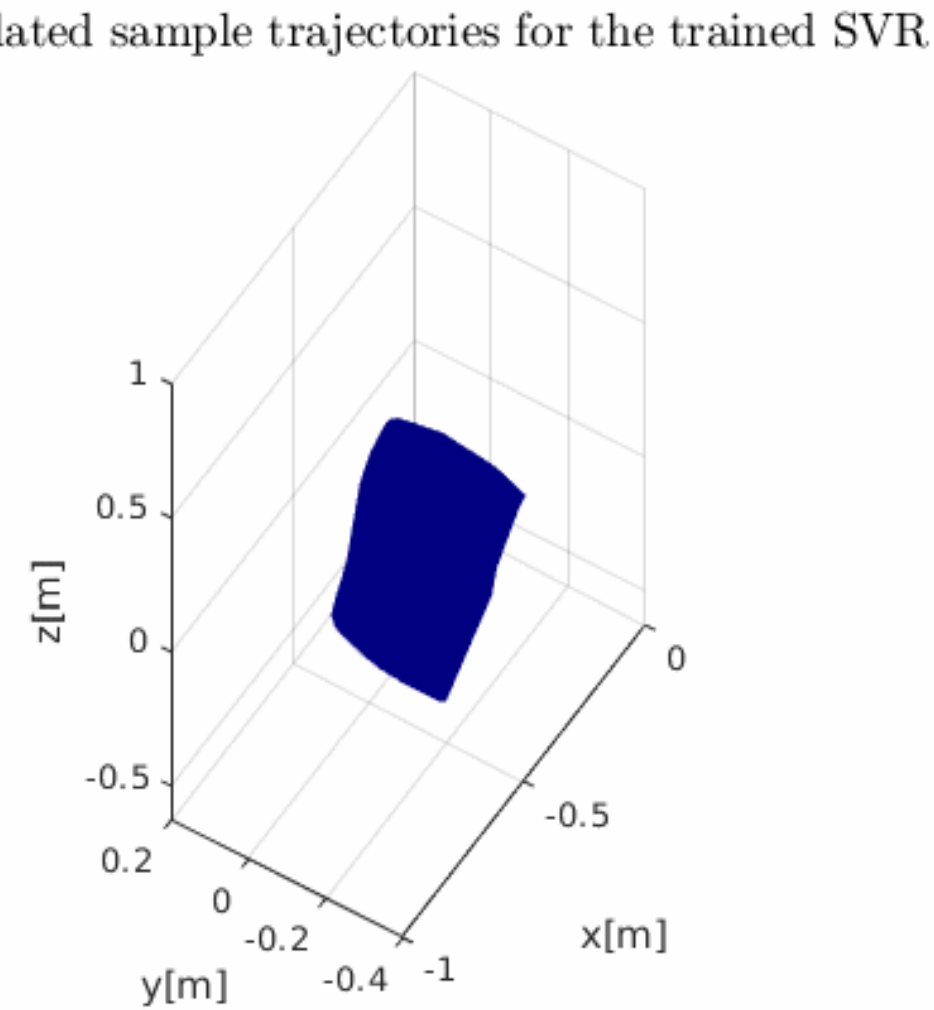




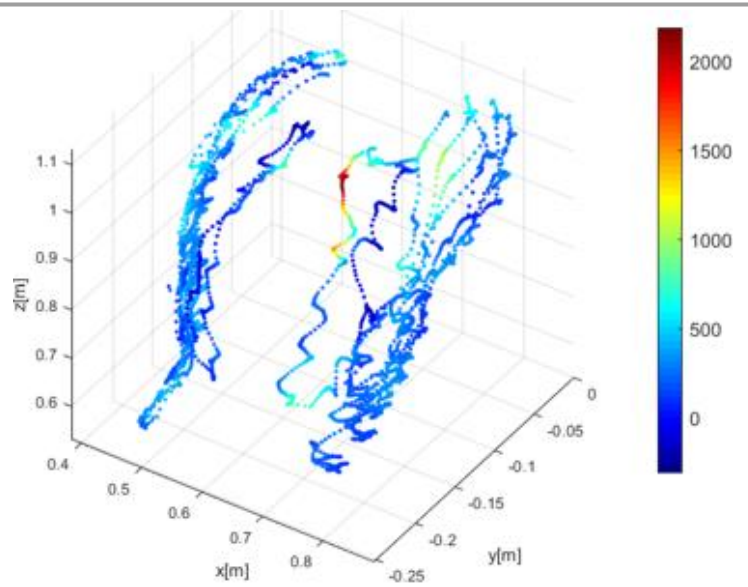
- **Cross-validation used for model validation with 5 folds.**
- **Grid search to find optimal C and σ for the SVR.**
- **Best testing accuracy achieved with a RMSE of 0.52 [cm] for points on the back**

Reconstructed surface & gradients for the trained SVR

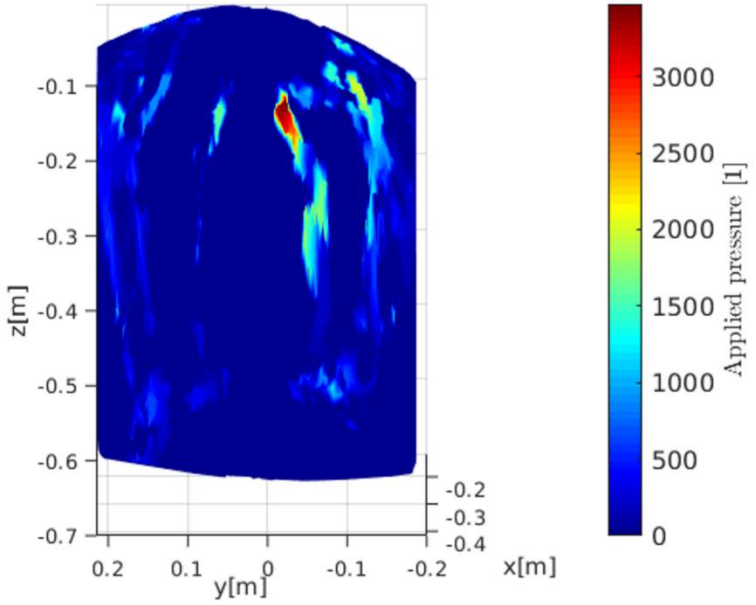




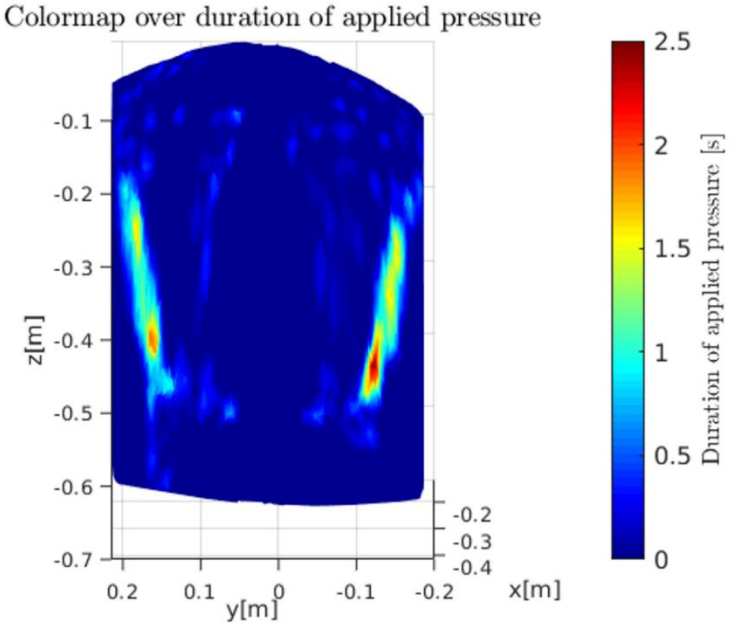
Test data from movements on the back of the mannequin

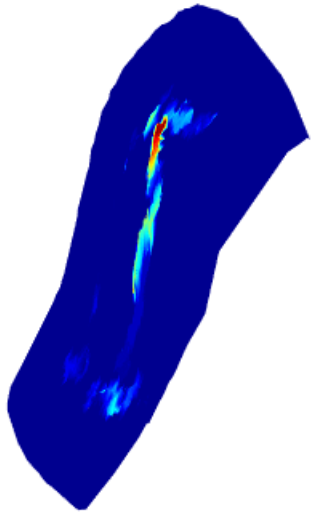


Force applied on the back of the mannequin

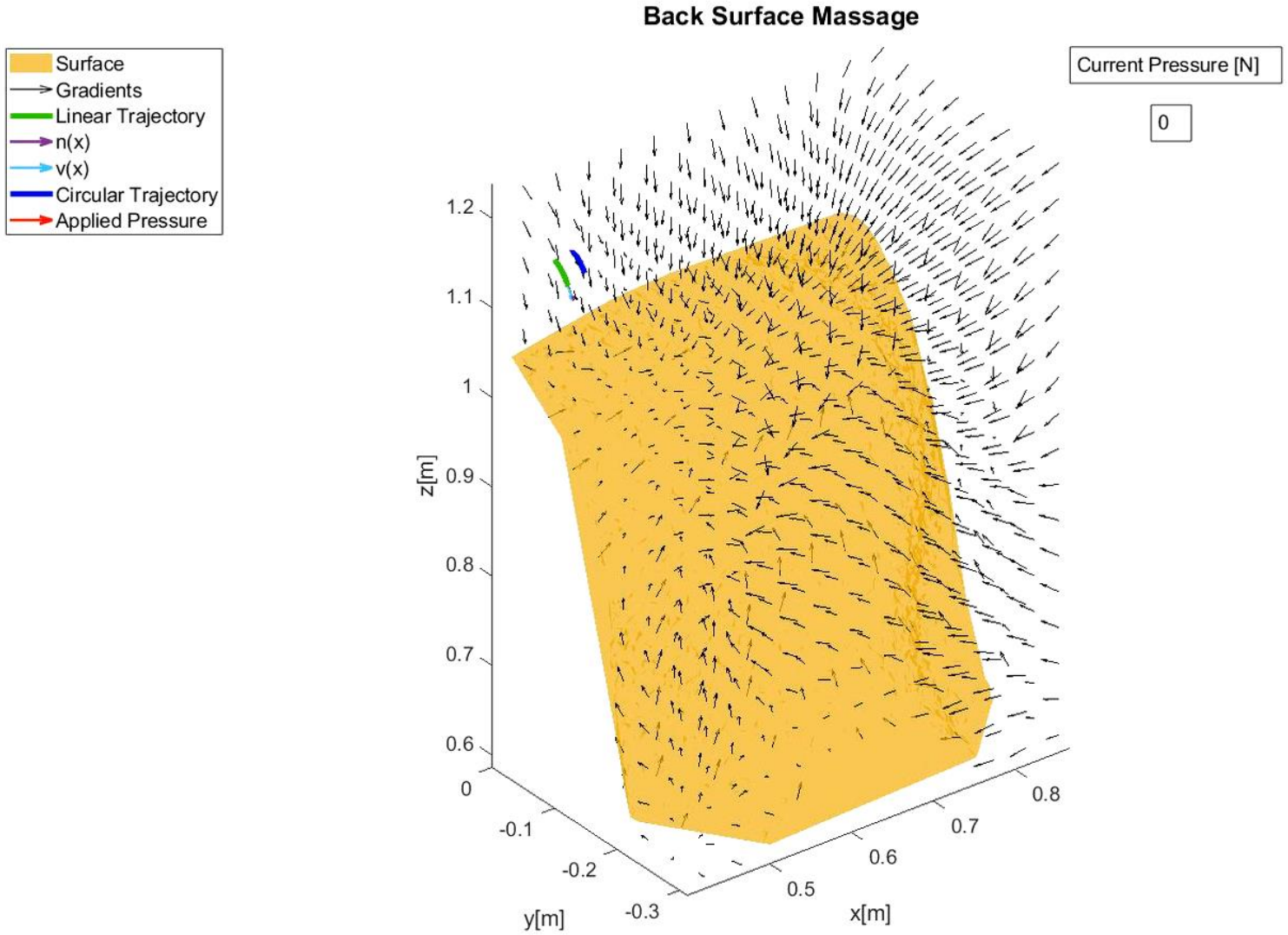


Colormap over duration of applied pressure





Reconstruction of Motion on Back Surface

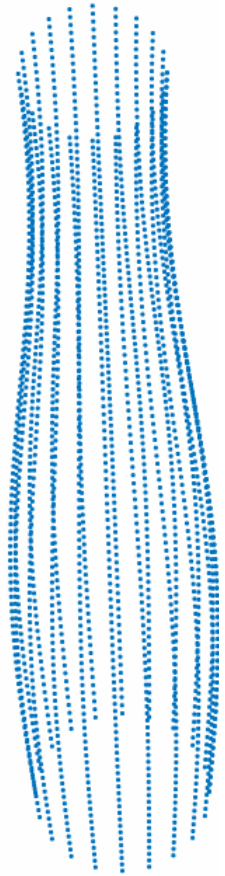


Learning Arm Massage

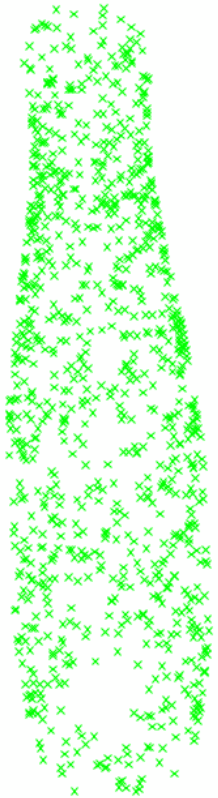
Khoramshahi et al, ICRA 2020

Supported by Samsung

Create artificial data for the arm

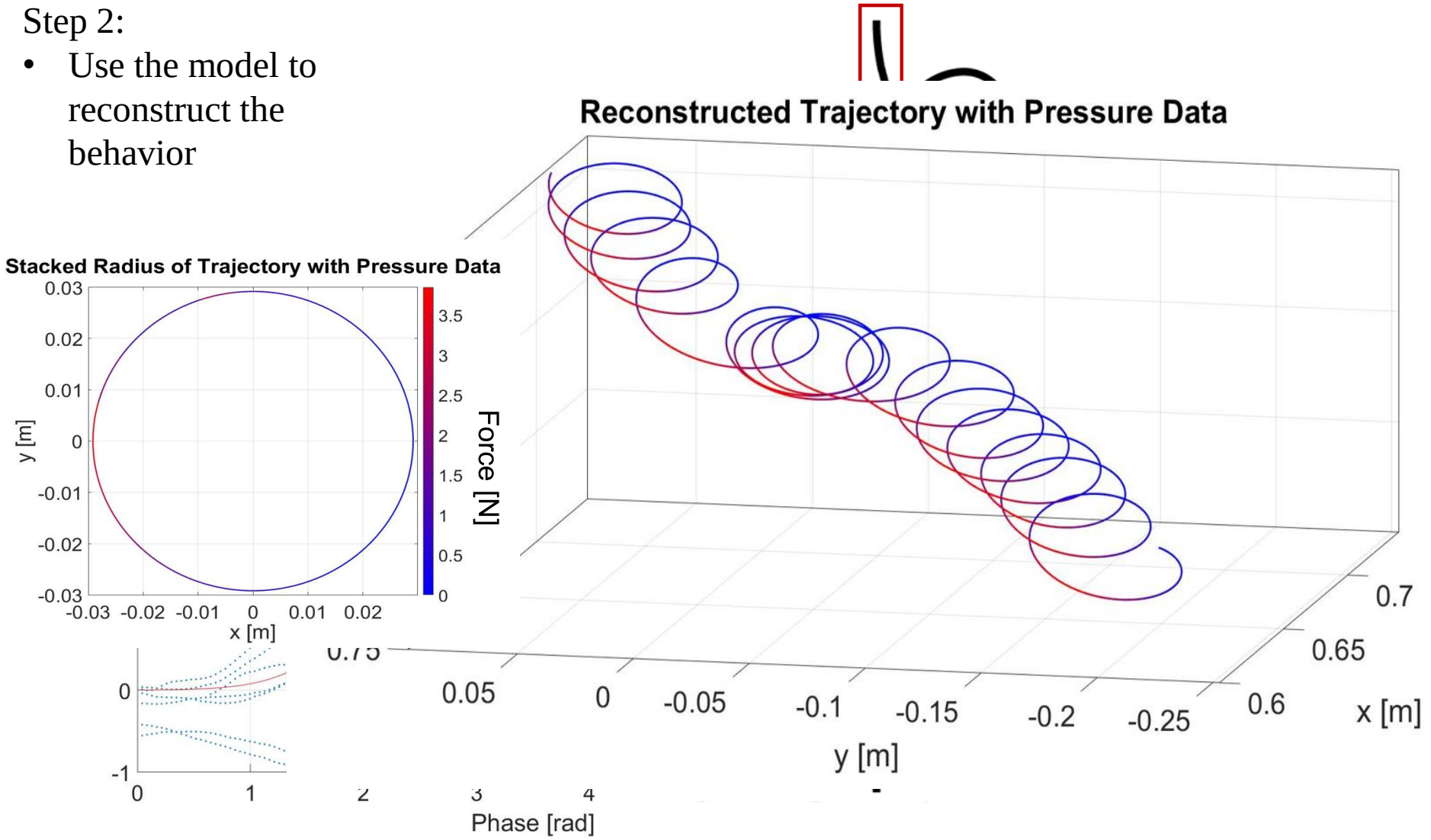


Train SVR and predict the surface



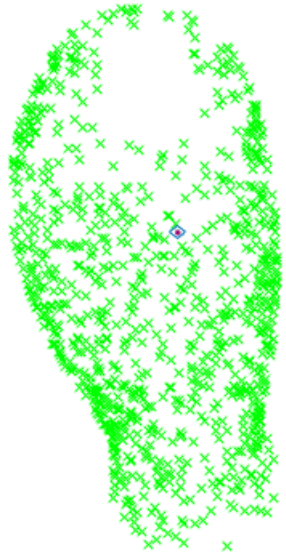
Step 2:

- Use the model to reconstruct the behavior

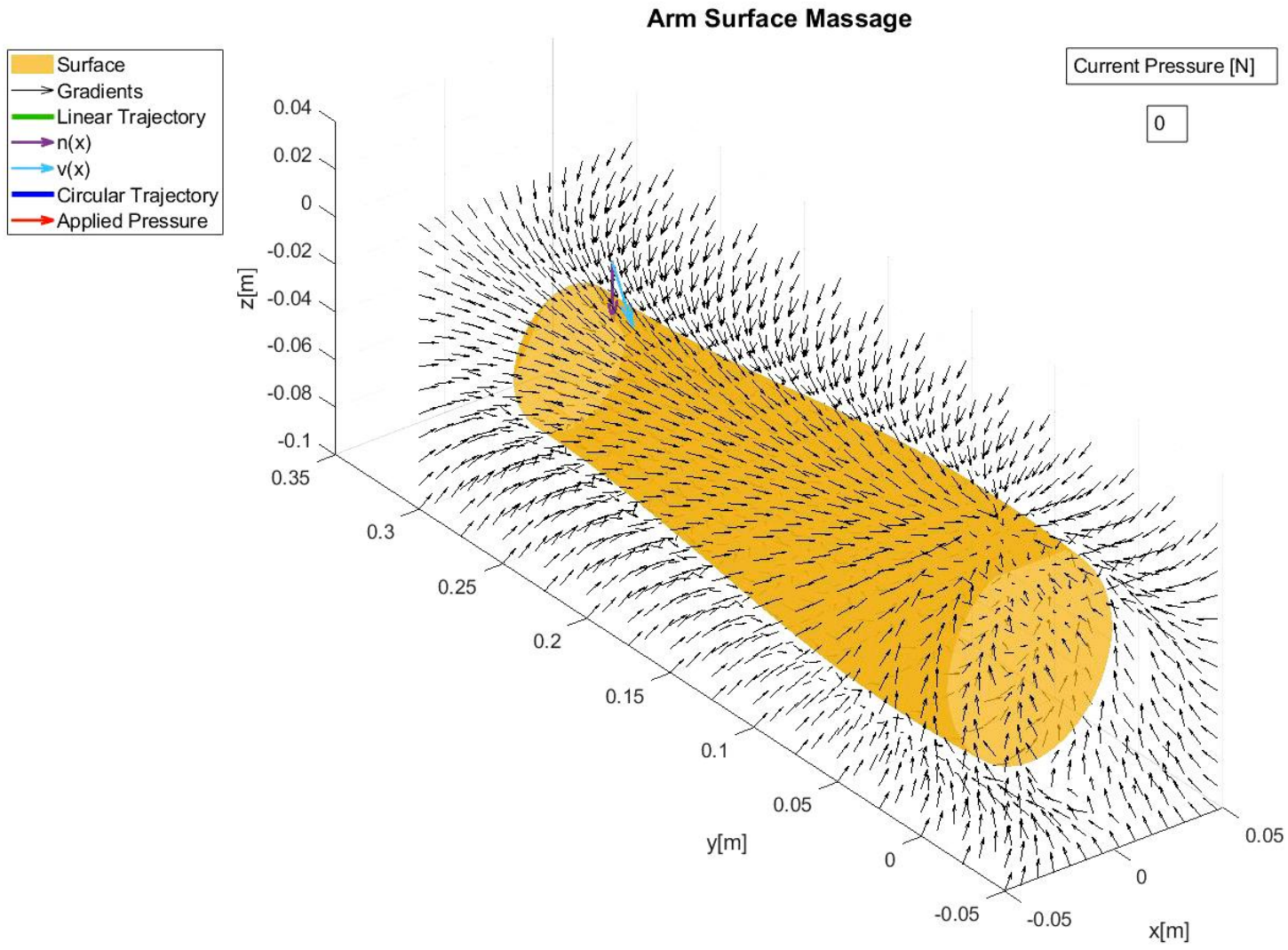


Khoramshahi et al, ICRA 2020

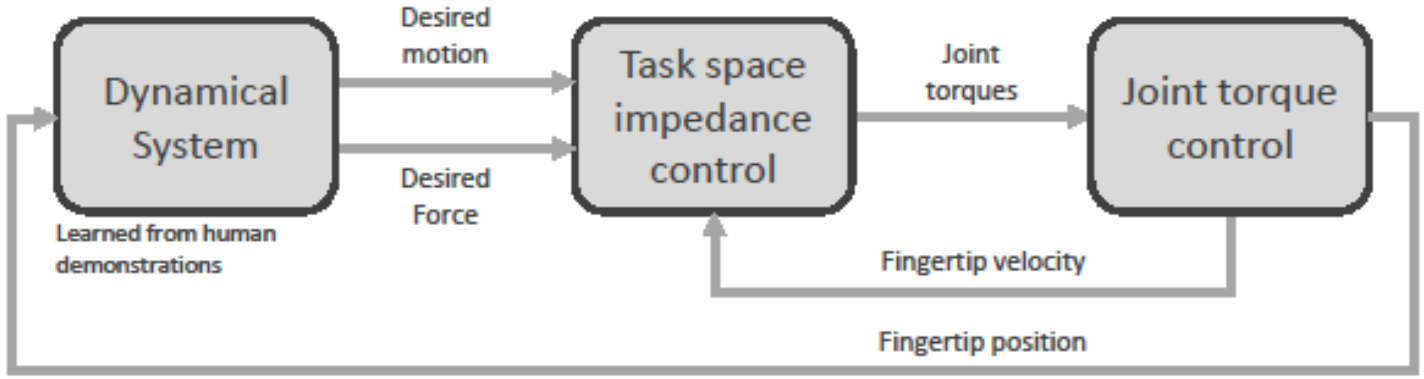
Reconstruction of Motion on Arm Surface

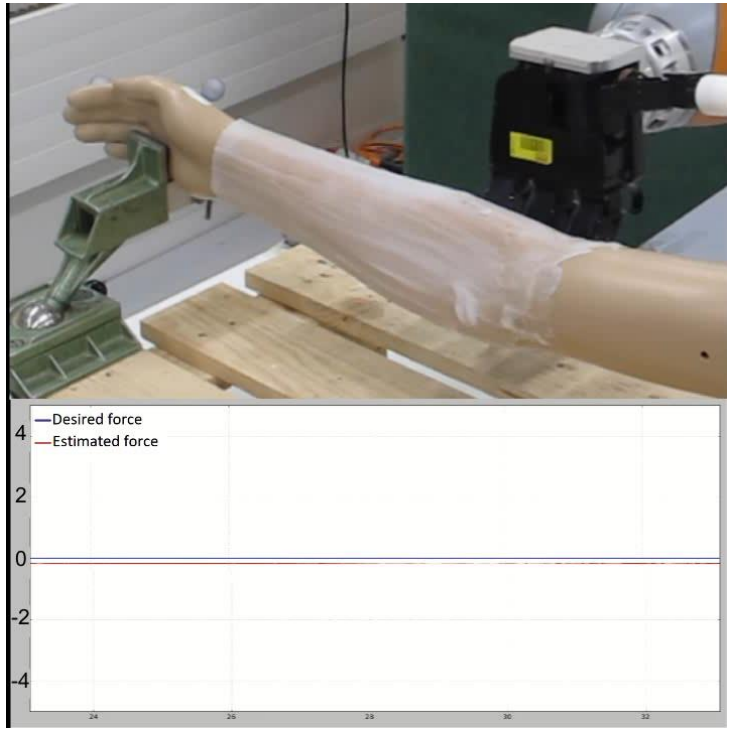


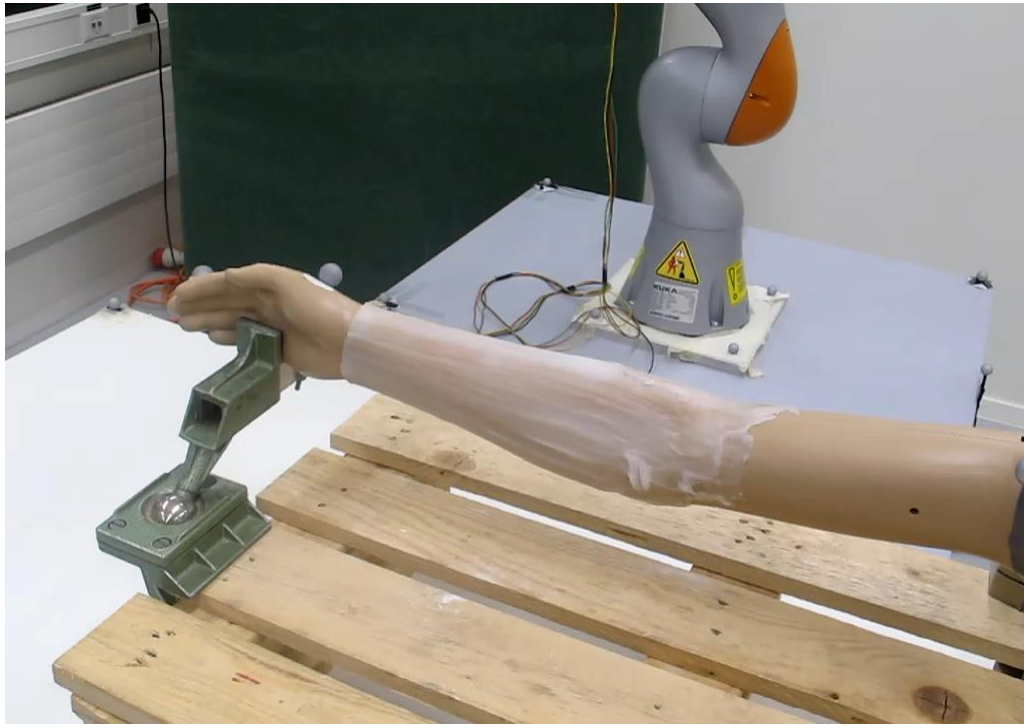
Reconstruction of Motion on Arm Surface



Learning and adaptive control for robots







Summary

- Introduced a means to use Passive-DS to perform force control.
  - The DS is decomposed into two parts, one controlling for motion along the surface, the other controlling for force.
  - The control is simplified by introducing a function Γ that determines the distance and normal to the surface everywhere (akin to principle used in obstacle avoidance) → This allows decoupling control of force from motion along two orthogonal axes.
  - Machine learning (e.g nonlinear regression through SVR) can be used to model the function Γ.
  - A force-based model associated to position is used for modulate the force along the DS
  - To show that the system remains passive, as the DS is not necessarily conservative, the tank is required.
- Pattern of force can be learned
  - To compensate for unmodeled interaction forces (nonlinear friction, other dynamics of robot poorly modelled)
  - To learn a position-dependent pattern of force. Showed an example of application to model massage