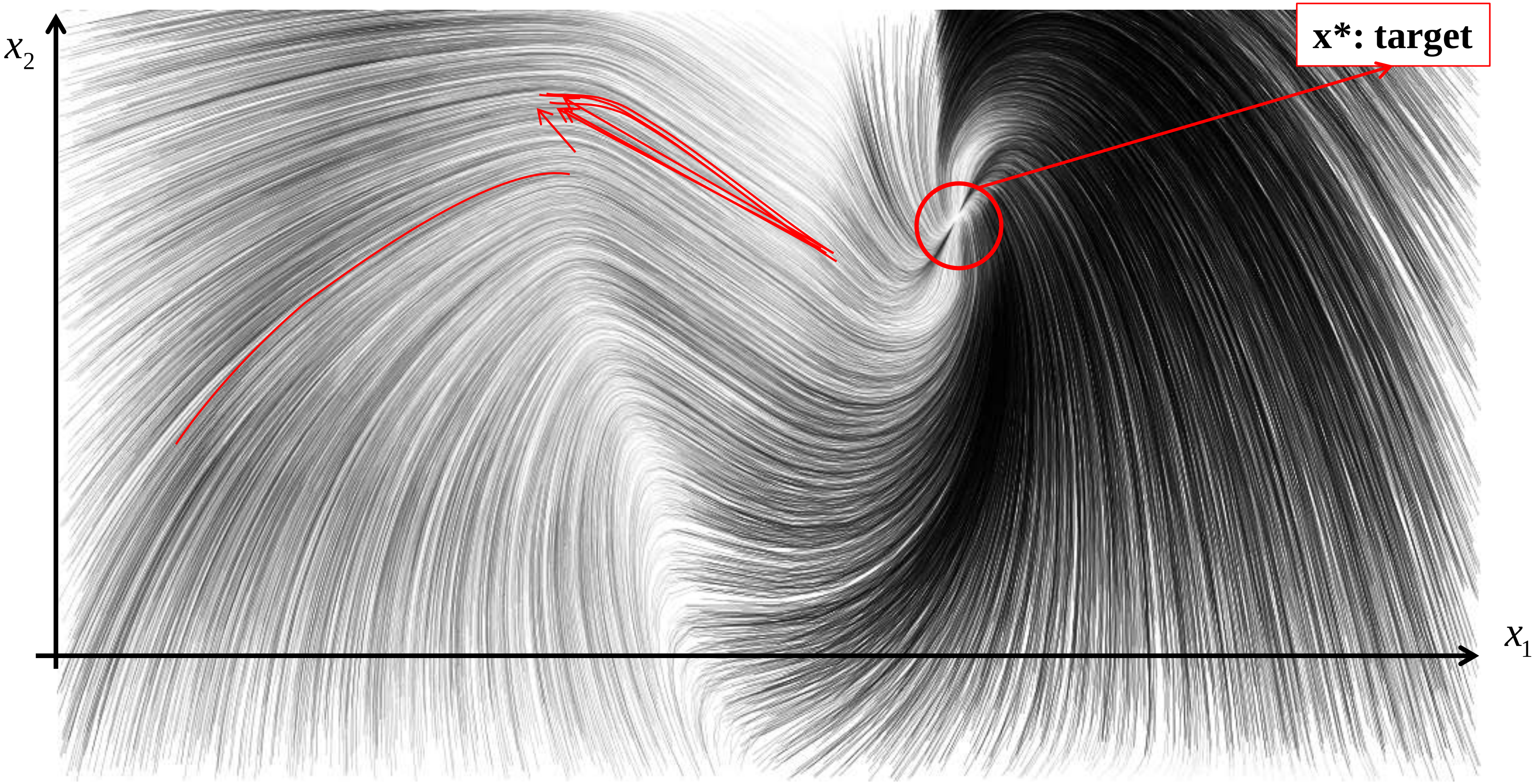
Lecture 11

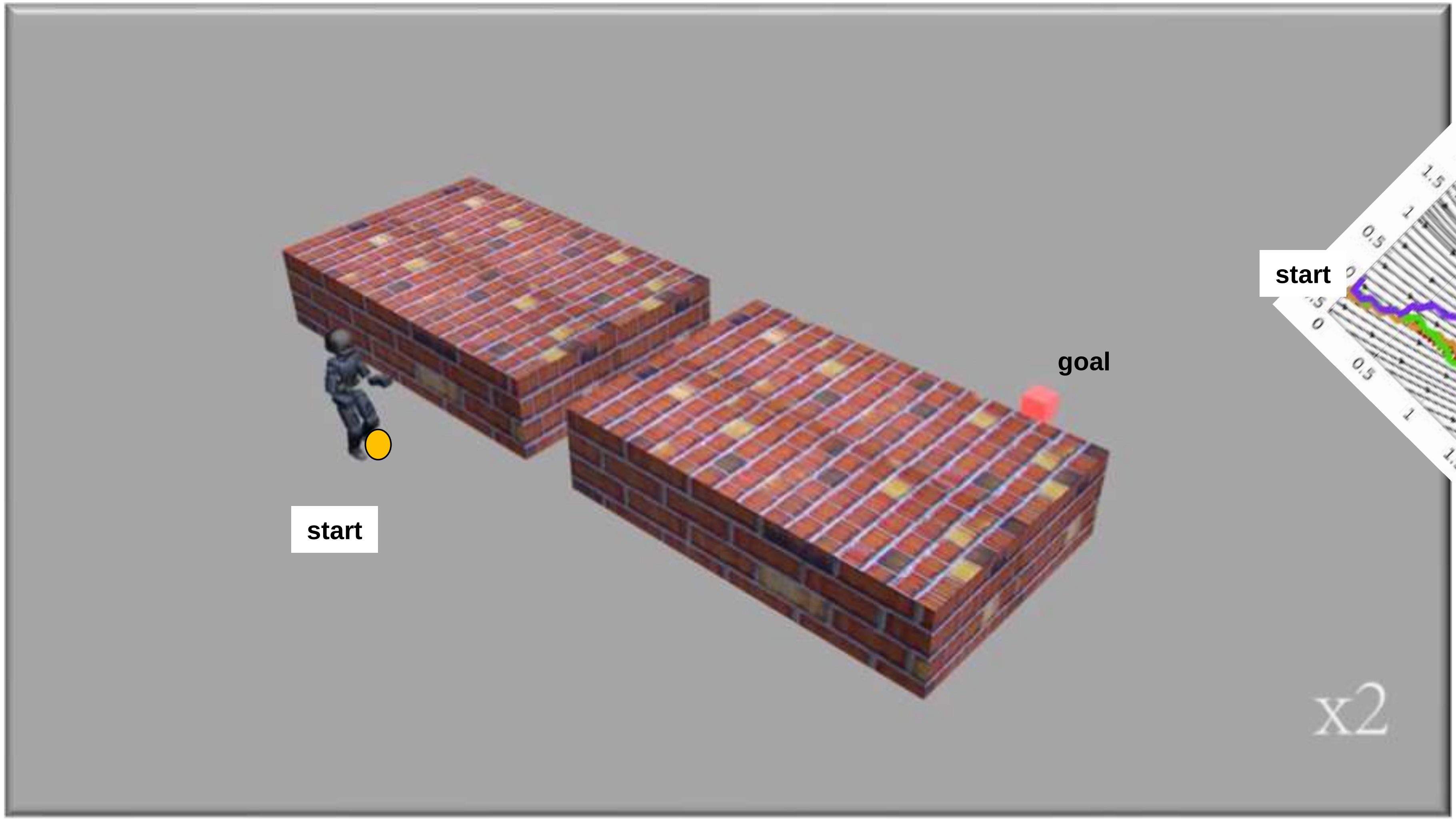
Impedance Control with Dynamical Systems – Passive DS

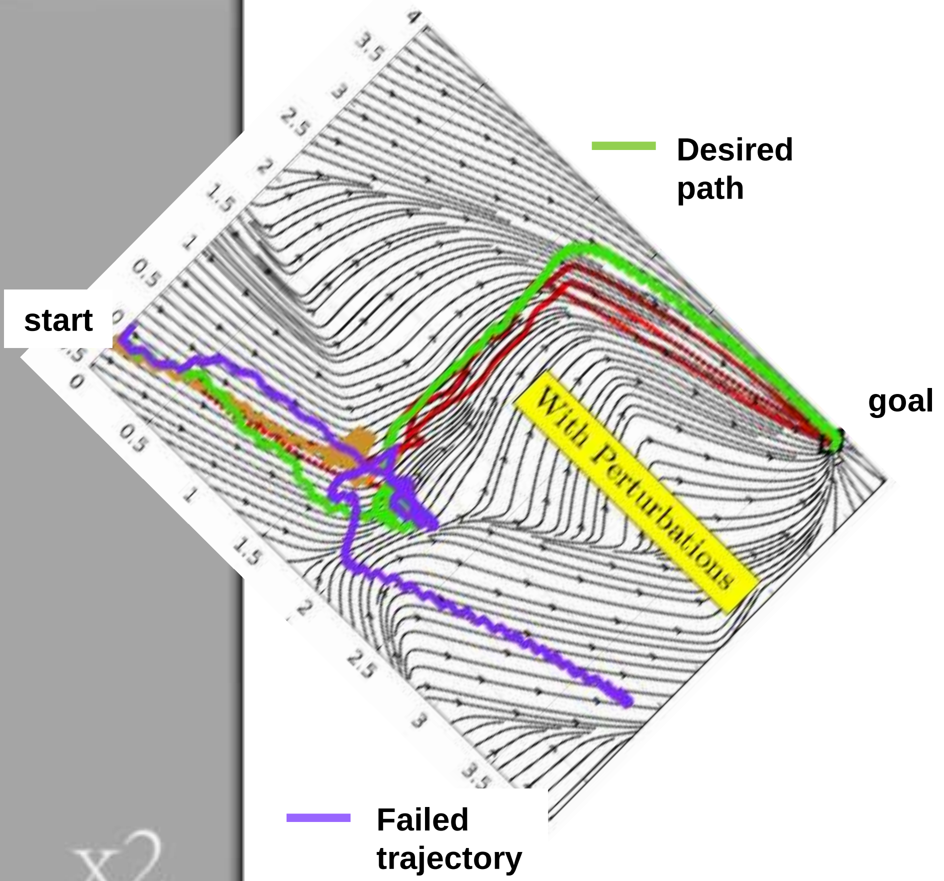
Force Control with Dynamical Systems

DS control makes the system infinitely compliant!



DS control makes the system infinitely compliant!





EPFL

Robot's Dynamics, Assumptions and Requirements

Dynamics equation of the robot

$$\underbrace{M(x)\ddot{x}}_{\text{Inertia}}+\underbrace{C(x,\dot{x})\dot{x}}_{\text{Coriolis}}+\underbrace{g(x)}_{\text{Gravity}} = \underbrace{\tau_c}_{\text{Control Input}} + \underbrace{\tau_e}_{\text{External Forces}}$$

Design a control law for generating control torques τ_c

For the system to remain stable under external disturbances, we need to show that it remains ***passive.*** *(see Annexes A.6)*

Control torques τ_c must be modulated to ensure that the system remains passive.

Stability of the System through Passivity Analysis

$\dot{x} = f(x,u), \quad u \in \mathbb{R}^p$: input

We must verify that the energy injected by the input **u** does not destabilize the system.

→ Verify that the system is *closed-loop passive*

Recall: To study stability of *f(x)*, we had used *Lyapunov stability*. We defined a way to measure the energy of the system and we verified that it decreased over time before eventually vanishing at the attractor.

Passivity extends the Lyapunov stability concept to systems that are **subjected to an external input *u*.**

To determine the evolution of the energy of the system, we define a variable:

$y = h(x), \; y \in \mathbb{R}^m$

Passivity: Definition

Definition A.8 *(Passivity): A system with the form*

$$\dot{x} = f(x, u)$$
$$y = h(x) \qquad \text{(A.9)}$$

is passive if there is a lower-bounded storage function $V: \mathbb{R}^N \rightarrow \mathbb{R}_{0\leq}$ *such that*

$$\underbrace{V(x(t)) - V(x(0))}_{\text{Stored energy}} \leq \underbrace{\int_0^t u(s)^T y(s)ds}_{\text{Supplied energy}} \qquad \text{(A.10)}$$

For strict <, the system is dissipative

is satisfied for all $0 \leq t$, *all input functions* u, *and all initial conditions* $x(0) \in \mathbb{R}^N$.

When is this equivalent to Lyapunov stability?

EPFL

Passivity: Definition

Definition A.10 *(Passivity—definition 3): A system with the form*

$$\dot{x} = f(x, u)$$
$$y = h(x) \tag{A.12}$$

is passive if there is a continuously differentiable, lower-bounded storage function $V: \mathbb{R}^N \to \mathbb{R}_{0\leq}$ *such that along the trajectories generated by* (A.12)

$$\dot{V}(t) \leq u(t)^T y(t) \tag{A.13}$$

is satisfied for all $0 \leq t$*, all input functions* u *and all initial conditions* $x(0) \in \mathbb{R}^N$.

To show that a system is passive, we must define a storage function and modulate the control input in such a way that passivity remains possible.

EPFL

Goals for the design of the control torques

Dynamics equation of the robot

$$\underbrace{M(x)\ddot{x}}_{\text{Inertia}} + \underbrace{C(x,\dot{x})\dot{x}}_{\text{Coriolis}} + \underbrace{g(x)}_{\text{Gravity}} = \underbrace{\tau_c}_{\text{Control Input}} + \underbrace{\tau_e}_{\text{External Forces}}$$

Design a control law for generating control torques τ_c

Goals for the control system:

- The robot should move according to a desired dynamics, set by $\dot{x} = f(x)$.
- The system should remain *passive*.
- If *f(x) is Lyapunov stable*, the control should dissipate energy solely in directions perpendicular to *f(x)*.

Format of Control Torques

Dynamics equation of the robot

$$\underbrace{M(x)\ddot{x}}_{\text{Inertia}} + \underbrace{C(x,\dot{x})\dot{x}}_{\text{Coriolis}} + \underbrace{g(x)}_{\text{Gravity}} = \underbrace{\tau_c}_{\text{Control Input}} + \underbrace{\tau_e}_{\text{External Forces}}$$

Feedback term:

$$\tau_c = \underbrace{g(x)}_{\text{Gravity compensation}} - \underbrace{D(x)\dot{x}}_{\text{Damping}}$$

Control torques τ_c must be modulated to ensure that the system remains passive.
→ Modulate *D(x)*.

Internal control system to compensate gravity (*gravity compensation mode on standard robots*).

Constraints on Control Torques for Passivity

Feedback term:

$$\tau_c = \underbrace{g(x)}_{\text{Gravity compensation}} - \underbrace{D(x)\dot{x}}_{\text{Damping}}$$

What is the first constraint we must set for *D(x)*?

$$D(x) \succ 0, \quad \forall x$$

Robot's dynamics

We verify that the system remains passive under external disturbances τ_e.

We set: $\begin{cases} u = \tau_e \\ y = \dot{x} \end{cases}$

We verify that : $\dot{V} \le \tau_e^{T} \dot{x}$ $\quad V$: kinetic energy $\quad V = \frac{1}{2}\dot{x}^T M(x)\dot{x}$

Passivity Verification I

We verify that : $\dot{V} \le \tau_e{}^T \dot{x}$

$$\dot{V} = \dot{x}^T M(x)\ddot{x} + \frac{1}{2}\dot{x}^T \dot{M}(x)\dot{x}$$

Replacing with $M(x)\ddot{x} + C(x,\dot{x})\dot{x} + g(x) = \tau_c + \tau_e$ and $\tau_c = g(x) - D(x)\dot{x}$

$$\dot{V} = \frac{1}{2}\dot{x}^T\left(\dot{M}(x) - 2C(x,\dot{x})\right) - \dot{x}^T D(x)\dot{x} + \dot{x}\tau_e \;\; \le \tau_e{}^T \dot{x}$$

$\dot{M}(x) - 2C(x,\dot{x}) = 0$

Skew-symmetric

< 0

since $D(x) \succ 0$

Tracking Feedback Control Loop

Feedback term:

$$\tau_c = \underbrace{g(x)}_{\text{Gravity compensation}} \underbrace{-D(x)(\dot{x} - f(x))}_{\text{Tracking}}$$

The system must follow a desired dynamics $\dot{x} = f(x)$

Traditional Tracking Feedback Control Loop

$$g(x) - D(\dot{x} - \dot{x}^d) - \cancel{K(x - x^d)} = \tau_c$$

Tracking in position given by the DS

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

State-dependent Impedance modulation

Shaping the Impedance

$$g(x) - D(x)\left(\dot{x} - f(x)\right) = \tau_c$$

Eigencomposition of $D(x)$

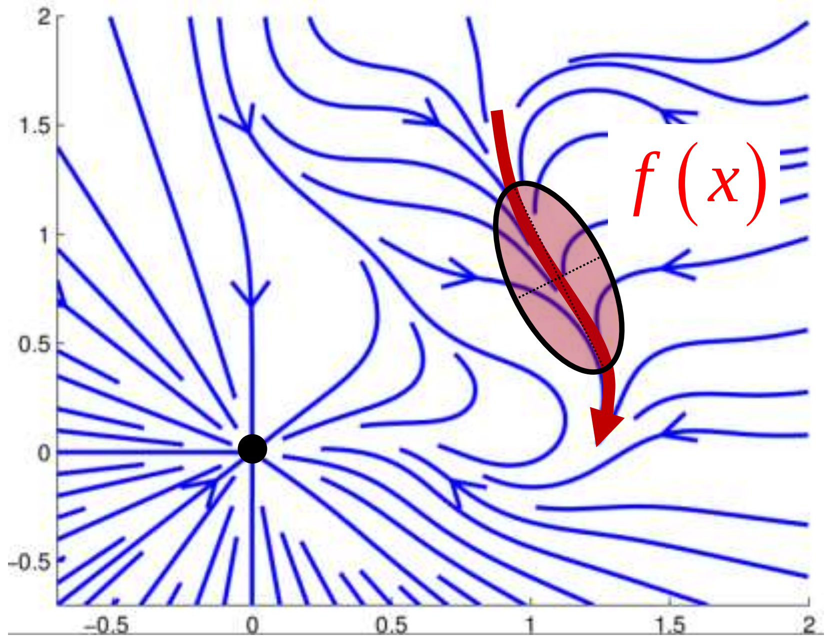
$$D(x) = Q(x)\Lambda(x)Q(x)^T$$

We set *f(x)* to be aligned with an eigenvector of *D(x)*

$$Q(x) = \left[e_1(x)\; e_2(x)\right], \quad e_1(x) = \frac{f(x)}{\|f(x)\|}, \quad e_1(x)^T e_2(x) = 0.$$

The eigenvalues will set the impedance

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



Shaping the Impedance

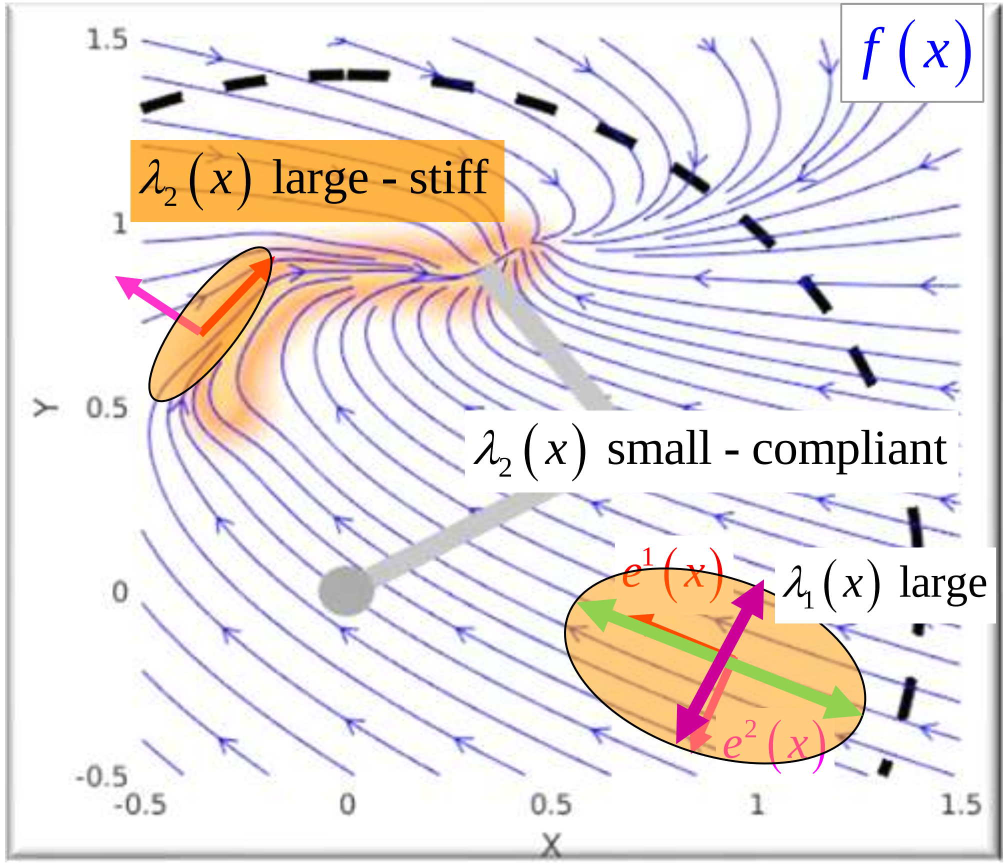
$$g(x) - D(x)\left(\dot{x} - f(x)\right) = \tau_c$$

The eigenvalues will set the impedance

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

Set $\lambda_1(x)$ to be very stiff for accurate tracking.

Modulate $\lambda_2(x)$ to comply with orthogonal disturbances.



$\lambda_1(x)$ ~ axes length of ellipse (see impedance class)

Passivity

$$g(x) - D(x)(\dot{x} - f(x)) = \tau_c \qquad \Rightarrow \qquad g(x) + \lambda_1 f(x) - D(x)\dot{x} = \tau_c$$

$\dot{x} = f(x)$ is an eigenvector of D: $e_1(x) = \frac{f(x)}{\| f(x) \|}$.

$\dot{x} = f(x)$ is a Lyapunov stable function

with an associated Lyapunov function $\mathrm{V}_f(x)$.

$\mathrm{V}_f(x)$ is a potential function and we can write:

$f(x) = -\nabla V_f(x)$

Passivity Verification II

We set $V = \underbrace{\frac{1}{2}\dot{x}^T \dot{M}(x)\dot{x}}_{\text{Kinetic Energy}} + \underbrace{\lambda_1 V_f(x)}_{\text{Potential Energy of } f(x)}$

We verify that : $\dot{V} \le \tau_e^{\,T} \dot{x}$

$$\dot{V} = \dot{x}^T M(x)\ddot{x} + \frac{1}{2}\dot{x}^T \dot{M}(x)\dot{x} + \lambda_1 \dot{V}_f(x)$$

Ch

Replacing with $M(x)\ddot{x} + C(x,\dot{x})\dot{x} + g(x) = \tau_c + \tau_e$, $\dot{V}_f(x) = \nabla V_f(x)^T \dot{x}$

and $g(x) - D(x)\left(\dot{x} - f(x)\right) = \tau_c$

?

$$\dot{V} = \frac{1}{2}\dot{x}^T\left(\dot{M}(x) - 2C(x,\dot{x})\right) - \dot{x}^T D(x)\dot{x} + \lambda_1 \dot{x}^T f(x) + \lambda_1 \dot{V}_f(x) + \dot{x}\tau_e \le \tau_e^{\,T}\dot{x}$$

$\dot{M}(x) - 2C(x,\dot{x}) = 0$
Skew-symmetric

< 0
since $D(x) \succ 0$

$= 0$
$f(x) = -\nabla V_f(x)$

Non-conservative DS – Energy Tank

$\dot{x} = f(x)$ is not conservative.

Decompose f into a conservative and non-conservative terms:

$$f(x) = f_c(x) + f_r(x)$$

Conservative part follows:

$$f_c(x) = -\nabla V_c(x)$$

The energy injection must now be actively controlled as we are left with a uncontrolled term:

$$\dot{V} = \ldots + f_r(x)^T \dot{x}$$

Introduce a new variable s to account for energy stored:

$$V(x,\dot{x},s) = \underbrace{\frac{1}{2}\dot{x}^T \dot{M}(x)\dot{x}}_{\text{Kinetic Energy}} + \underbrace{\lambda_1 V_c(x)}_{\text{Potential Energy of conservative flow } f_c(x)} + \underbrace{s}_{\text{Energy Tank}}$$

Set a tank limit $\bar{s}$ beyond which we do not allow the system to absorb energy anymore.

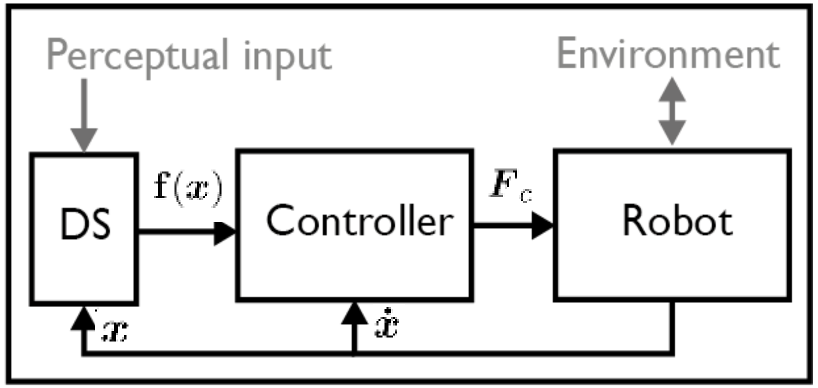
Modify the control torque to never exceed the tank limit:

$$\tau_c = g(x) - D(x)\dot{x} - \lambda_1 f_c(x) - \lambda_1 \beta(s,\bar{s}) f_r(x)$$

See exercise session to determine appropriate function: $\beta(s,\bar{s})$

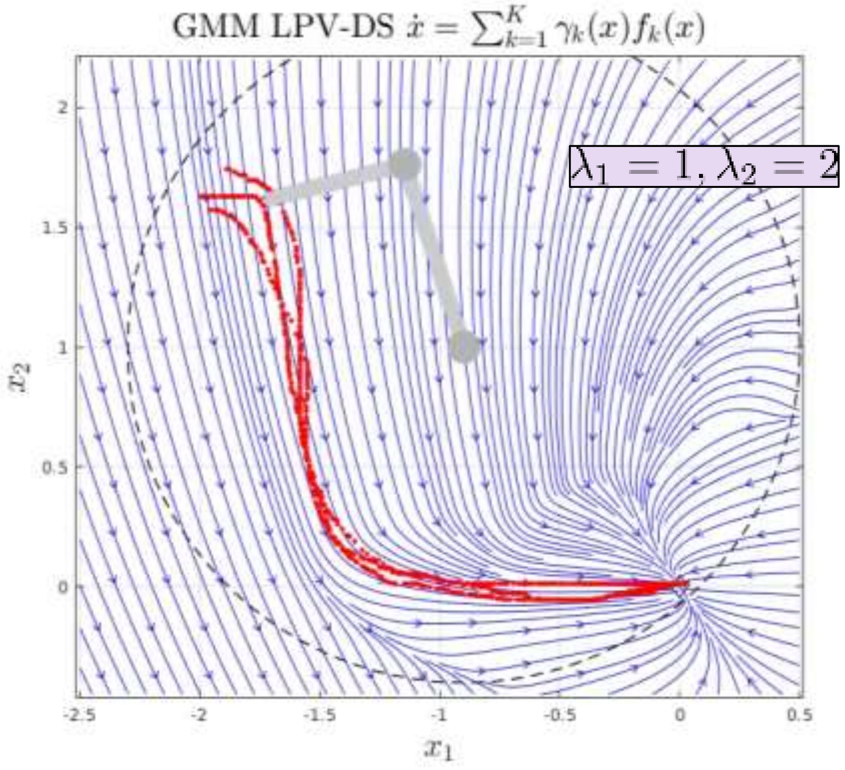
Open-Loop vs Closed-Loop Control with DS

DS in feedback configuration



Legend

x, v	*actual position, actual velocity*
u	*control command*



Given an autonomous DS:

$$\dot{\boldsymbol{x}} = \mathrm{f}(\boldsymbol{x}) \rightarrow \quad \{\lim_{t\to\infty} \|\boldsymbol{x} - \boldsymbol{x}^*\| = 0$$

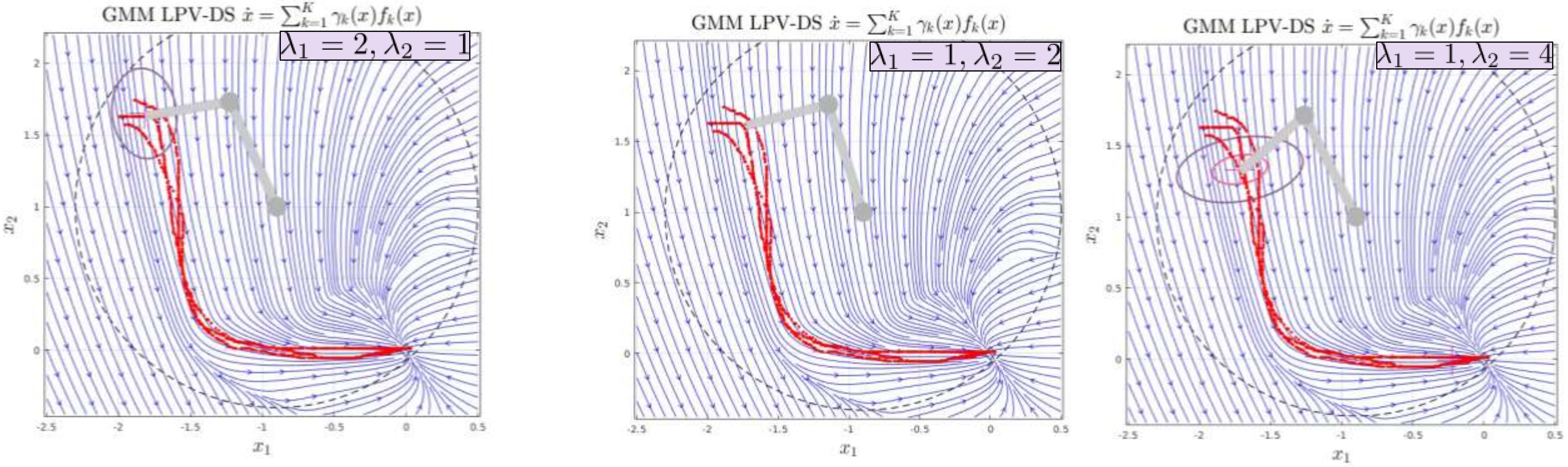
$\boldsymbol{x}$ $\dot{\boldsymbol{x}}$

$\boldsymbol{F}_c$

DS can be:

- Linear DS
- Nonlinear DS via LPV-DS or SEDS (Lecture 3)
- Modulated DS (Lecture 6)
- A non conservative DS

Impedance Modulation on LPV-DS



Desired Behavior is determined by choice of damping eigenvalues.

Apparent Stiffness

Closed-loop Dynamics with Passive-DS Control-law

$$M(x)\dot{x} + D(x)(\dot{x} - f(x)) = \tau_e$$

Equivalent to Damping term in classical impedance control law

Classical Impedance-like Control

$$M(x)\dot{x} + D(\dot{x} - \dot{x}^d) - K(x - x^d) = \tau_e$$

What about the stiffness term?

Observed/Apparent Stiffness can be derived [2]

$$\tilde{K}(x) = \frac{\partial \tau_e}{\partial x} = -\lambda_1 \frac{\partial f(x)}{\partial x}$$

$$e_i^T \tilde{K}(x) e_i = -\lambda_1 e_i^T \frac{\partial f(x)}{\partial x} e_i$$

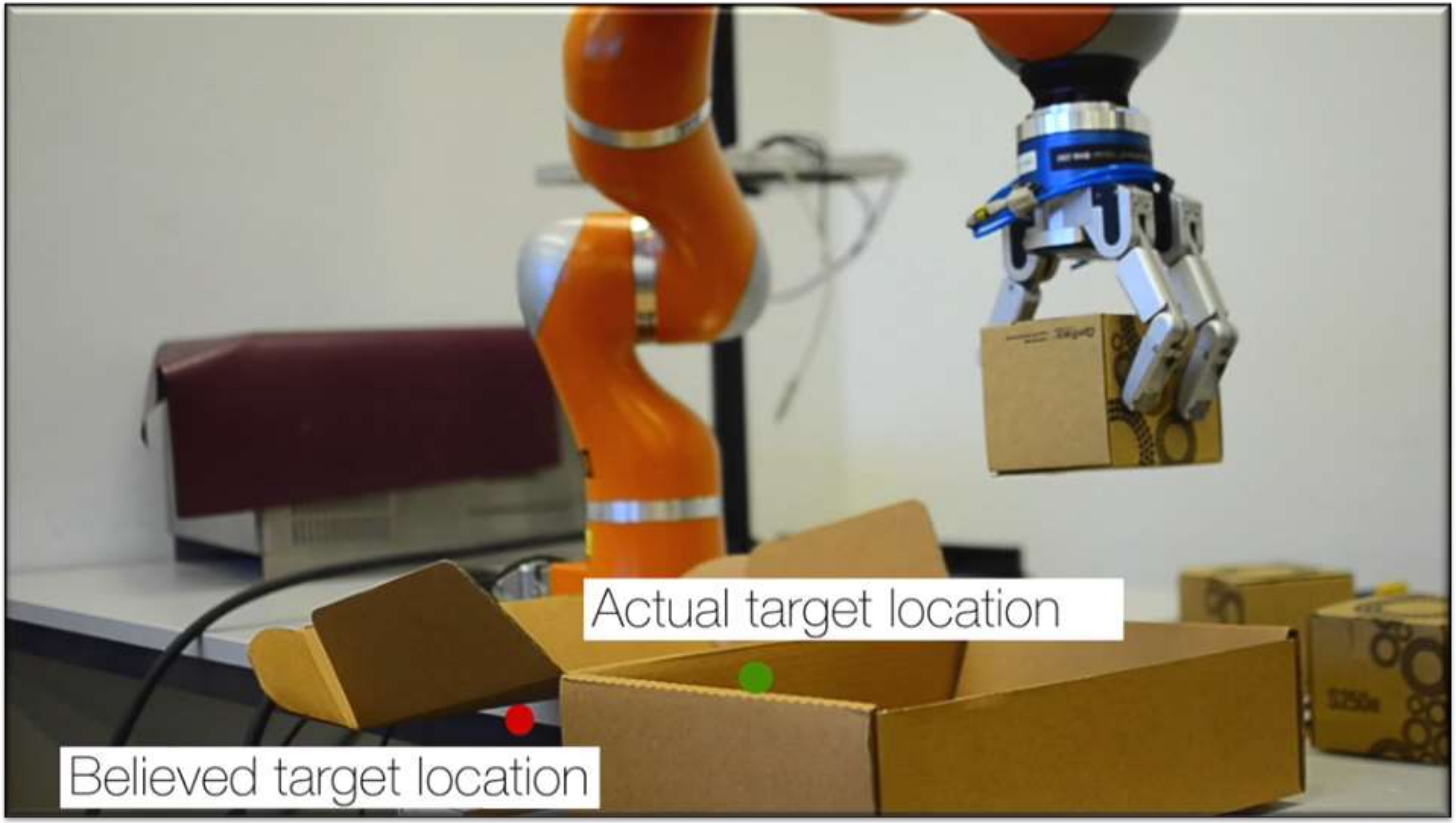
Unit norm vector

Rayleigh Quotient

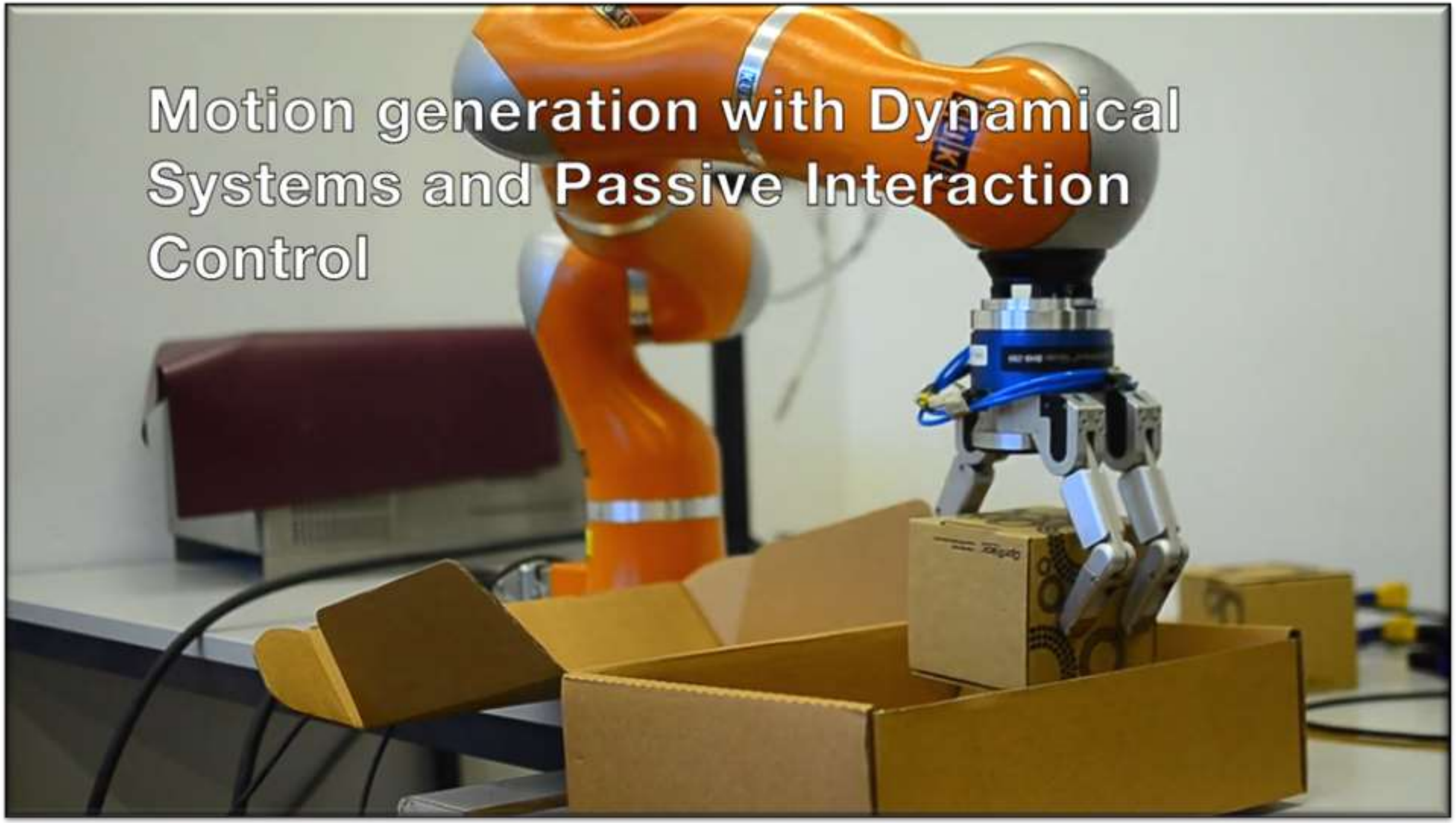
Stiffness is dependent on chosen damping values and convergence rate of the DS via the Jacobian!

[2] Khoramshahi, Billard, Autonomous Robots 2018

Passive-DS for Robot Control



Passive-DS for Robot Control



Passive-DS for Robot Control

EPFL

Summary

- Introduced a means to combine impedance control with DS control.
  - The DS acts as a trajectory generator.
  - Impedance control generates torques to track the output of the DS.

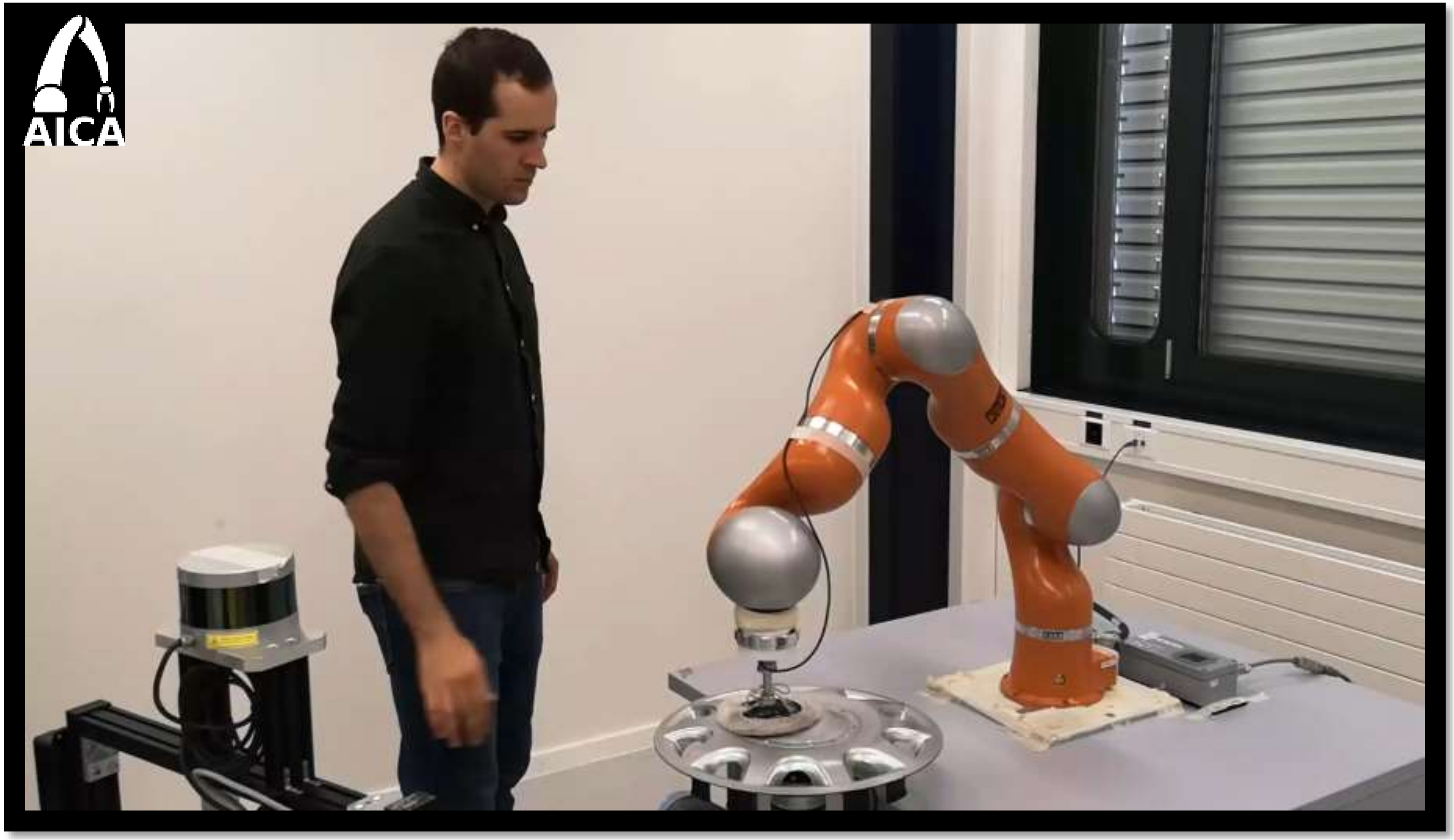
- Extended the notion of Lyapunov stability and introduced passivity to characterize a system subjected to disturbance in the form of external forces
  - Showed that when the nominal DS is conservative (Lyapunov stable), the system is passive.
  - When the DS is not conservative, one must introduce the notion of tank to track energy injected into the system.

- The impedance gains (damping matrix eigenvalues) modulate the response of the system when subjected to external disturbances (external forces).
  - Impedance is directional – aligned with the flow of the DS
  - High impedance in the direction of the DS will force the system to track accurately the DS
  - Low impedance in orthogonal directions allows to dissipate energy.

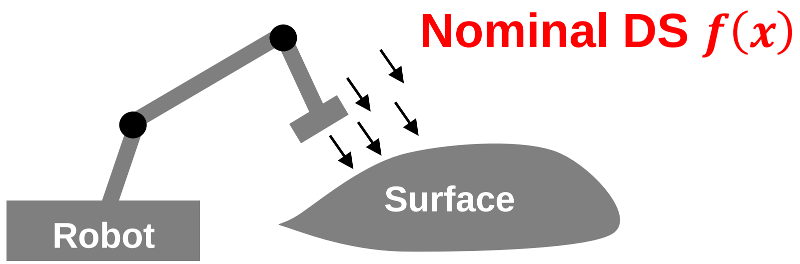
Force Control with Dynamical Systems

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).

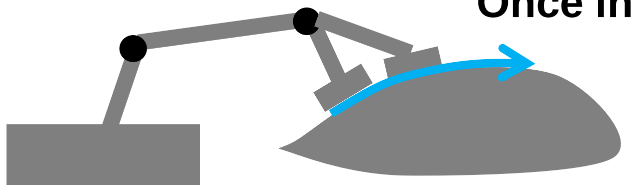
Controlling for force at contact



Controlling for force at contact

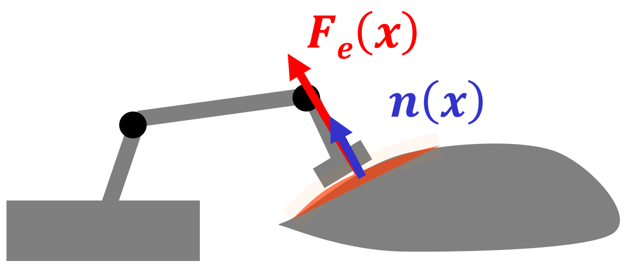


Controlling for force at contact: Principle



Once in contact, moves tangential to the surface

Controlling for force at contact: Principle



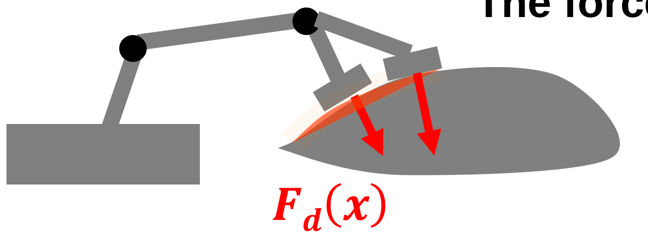
Surface impenetrable

Make contact

$$f(x)^T n(x) = 0 \quad (\text{in contact})$$

$$f(x)^T n(x) > 0 \quad (\text{in free space})$$

Controlling for force at contact: Principle



The force may vary along the surface $F_d(x)$

Goal: Apply a desired force F_d

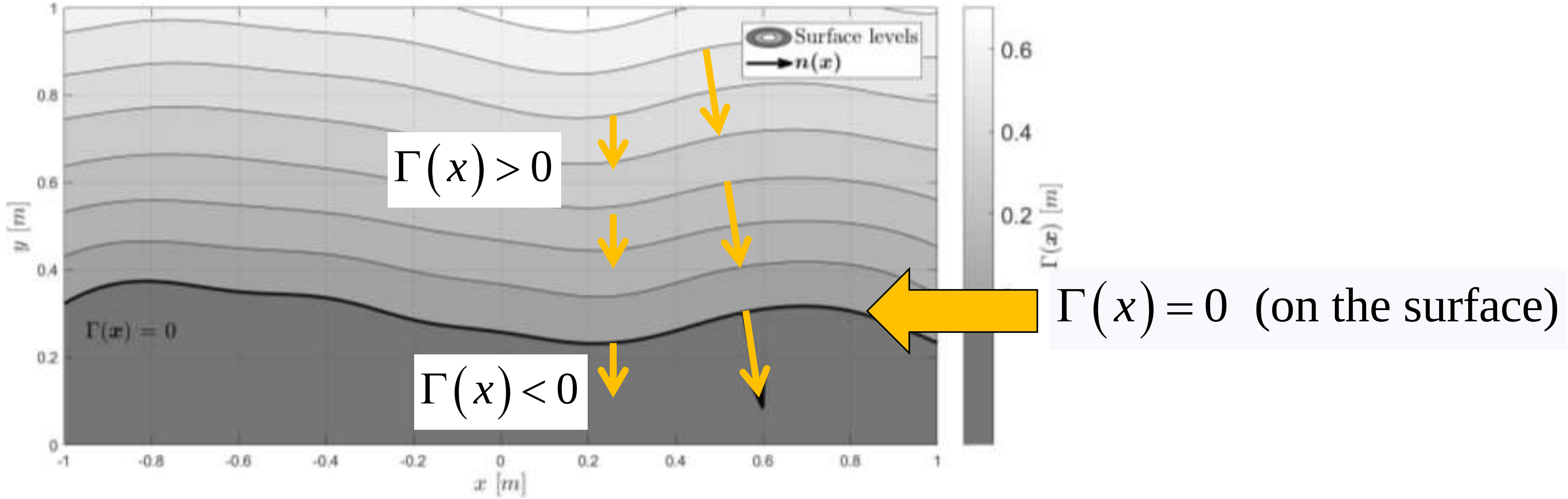
Idea: If we can project the control torques onto the surface, we simplify the computation.

→ Need a model of the surface.

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

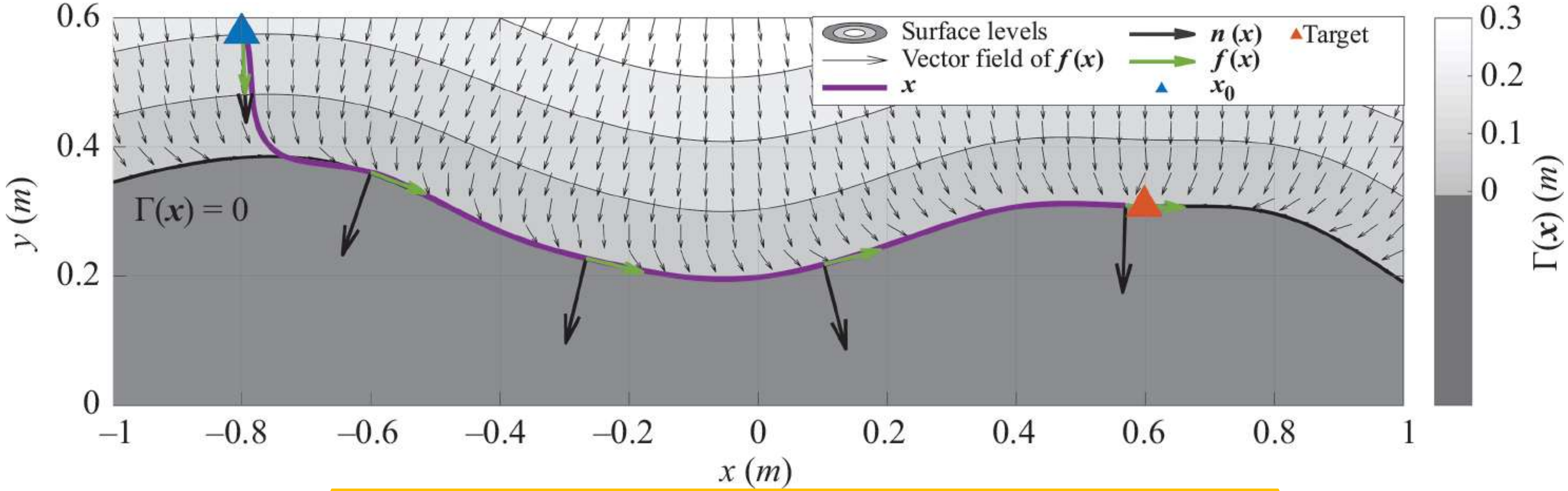


Nominal DS

We define a linear DS moving downwards towards the surface. $\dot{x} = Ax$

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.



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

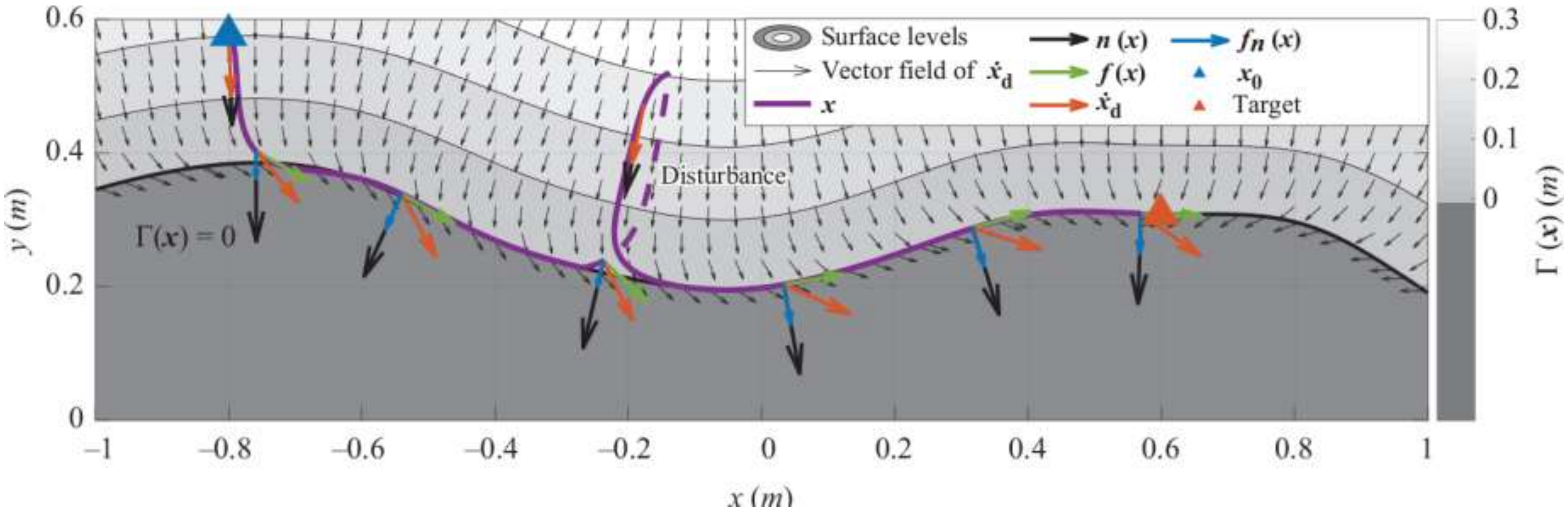
$R(x)$: Rotation to align to the surface once in contact

(same as in obstacle avoidance with constant velocity)

Nominal DS

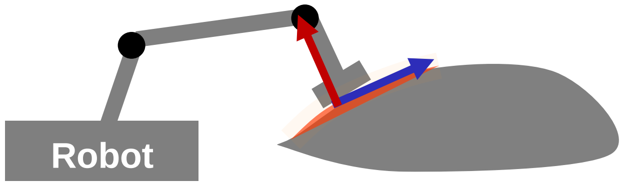
To separate control of force, decompose the nominal DS into two components:

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



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

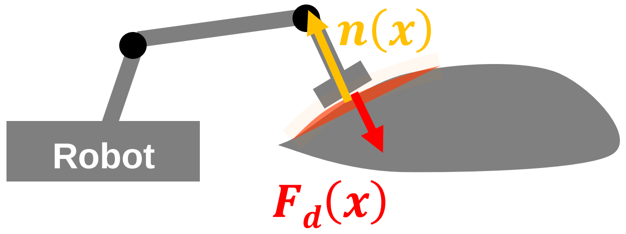
$$\dot{x}_d = f(x) + f_n(x)$$

$f(x)$: Reach/Move on the surface

$f_n(x)$: Apply the contact force

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

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

Robot's control torques

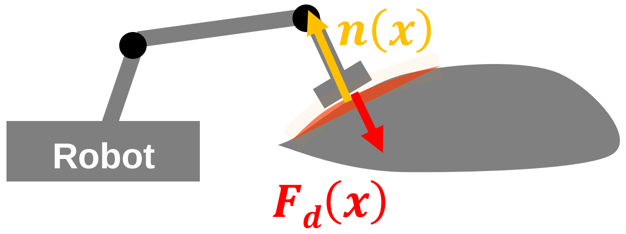
$$\dot{x}_d = f(x) + F_d(x)n(x)$$

$f(x)$: Reach/Move on the surface

$F_d(x)n(x)$: Apply the contact force

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

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

Robot's control torques

$F_d(x)n(x)$ — Apply the contact force

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

Eigencomposition of $D(x)$

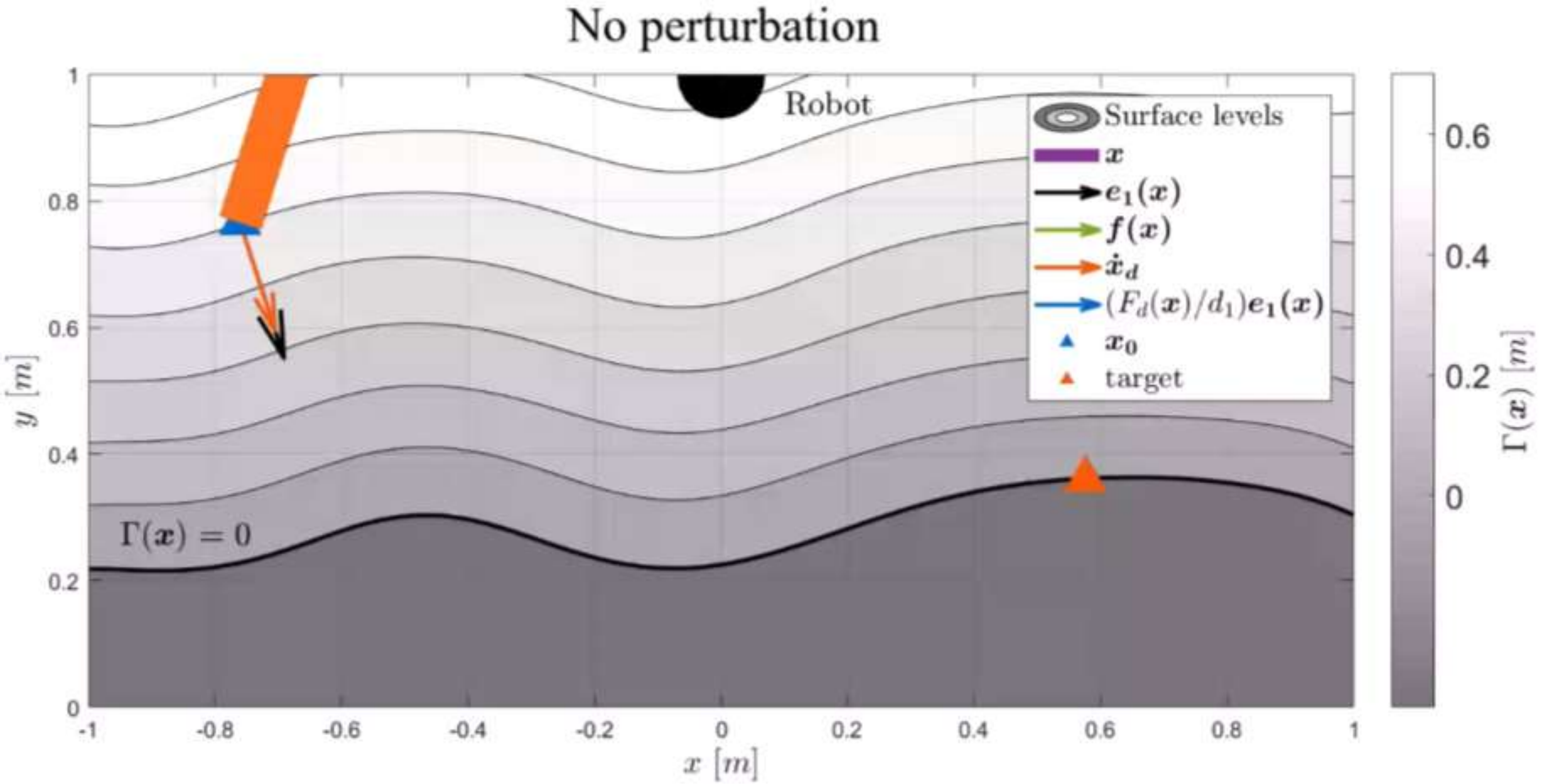
$$D(x) = E(x)\Lambda(x)E(x)^T$$

$$E(x) = [e_1(x) \; e_2(x)], \quad e_1(x) = \frac{f(x)}{\|f(x)\|}$$

Fixed impedance

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

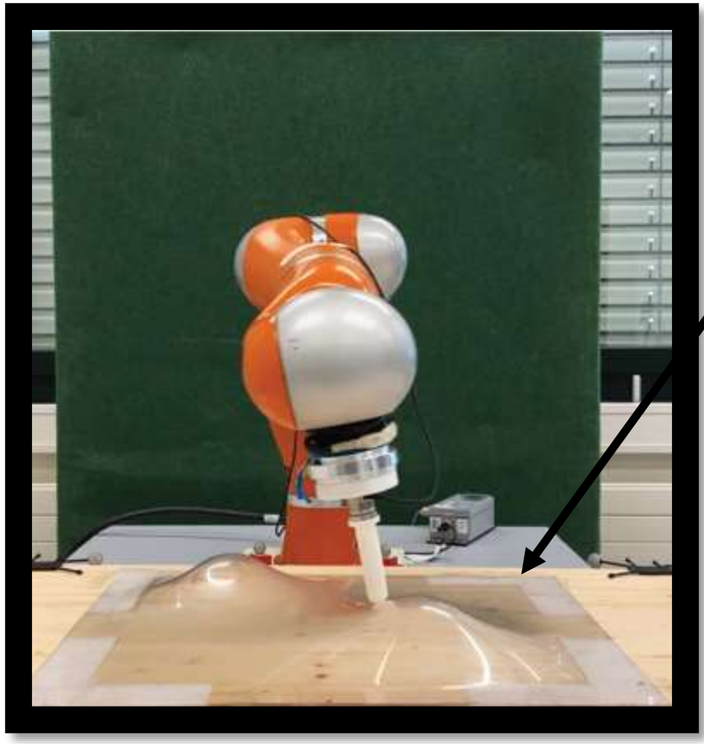
Animation of Principle



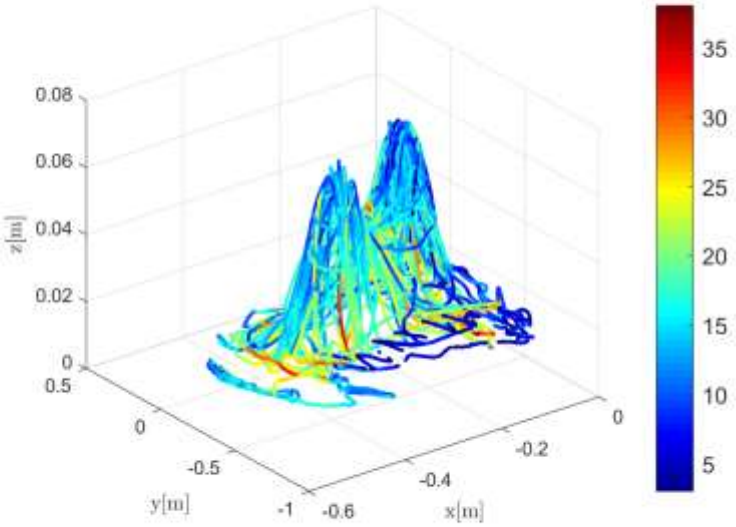
Robotic Demonstration

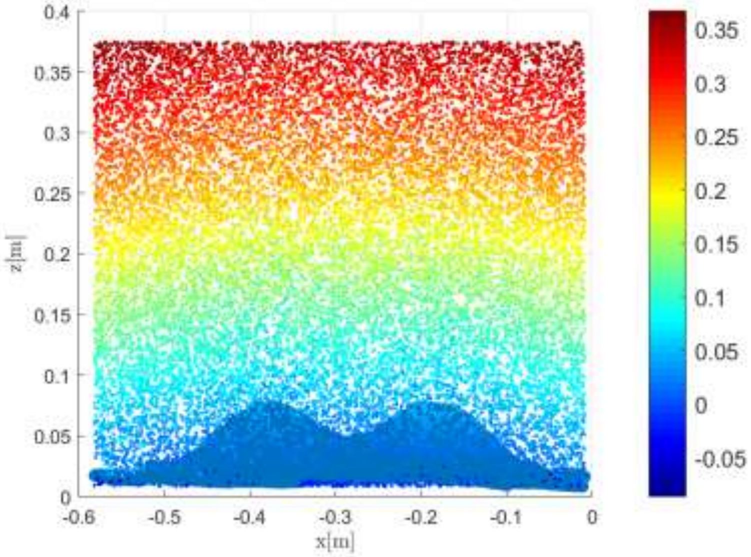
Task: The robot must polish in circular motion the surface applying a constant force of 20N.

Learn a model of the surface using Support Vector Regression



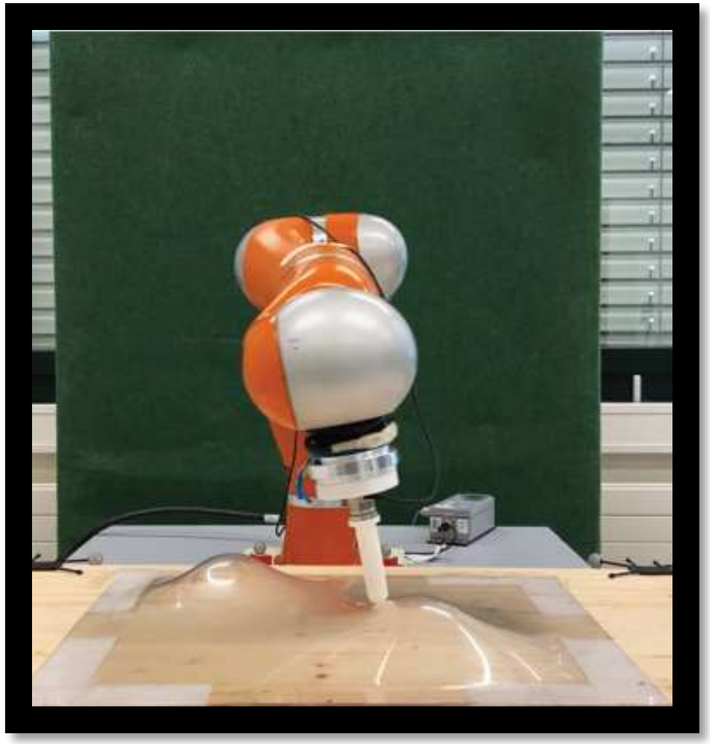
Input dataset to learn the surface model:



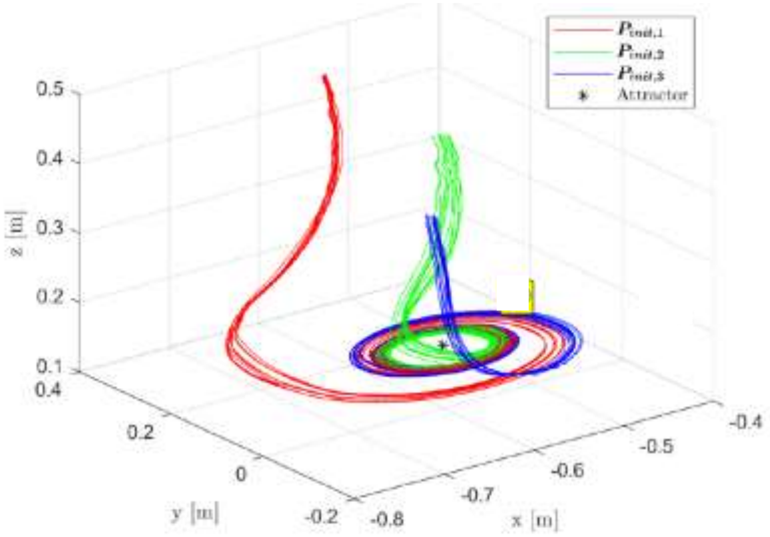


Robotic Demonstration

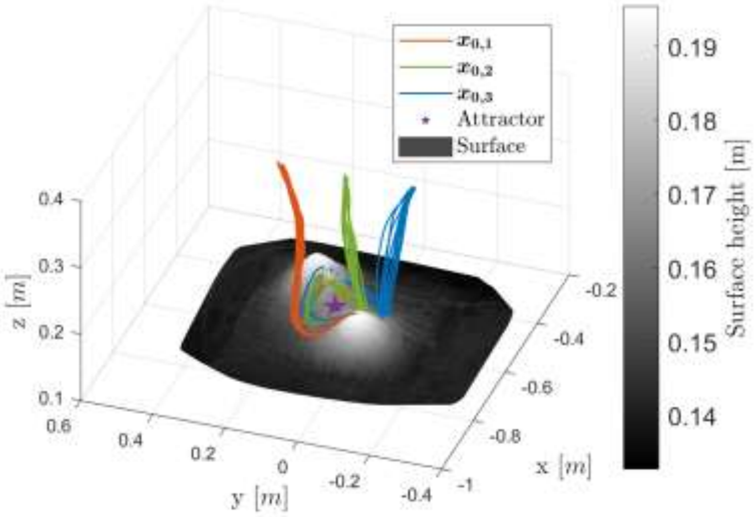
Task: The robot must polish in circular motion the surface applying a constant force of 20N.



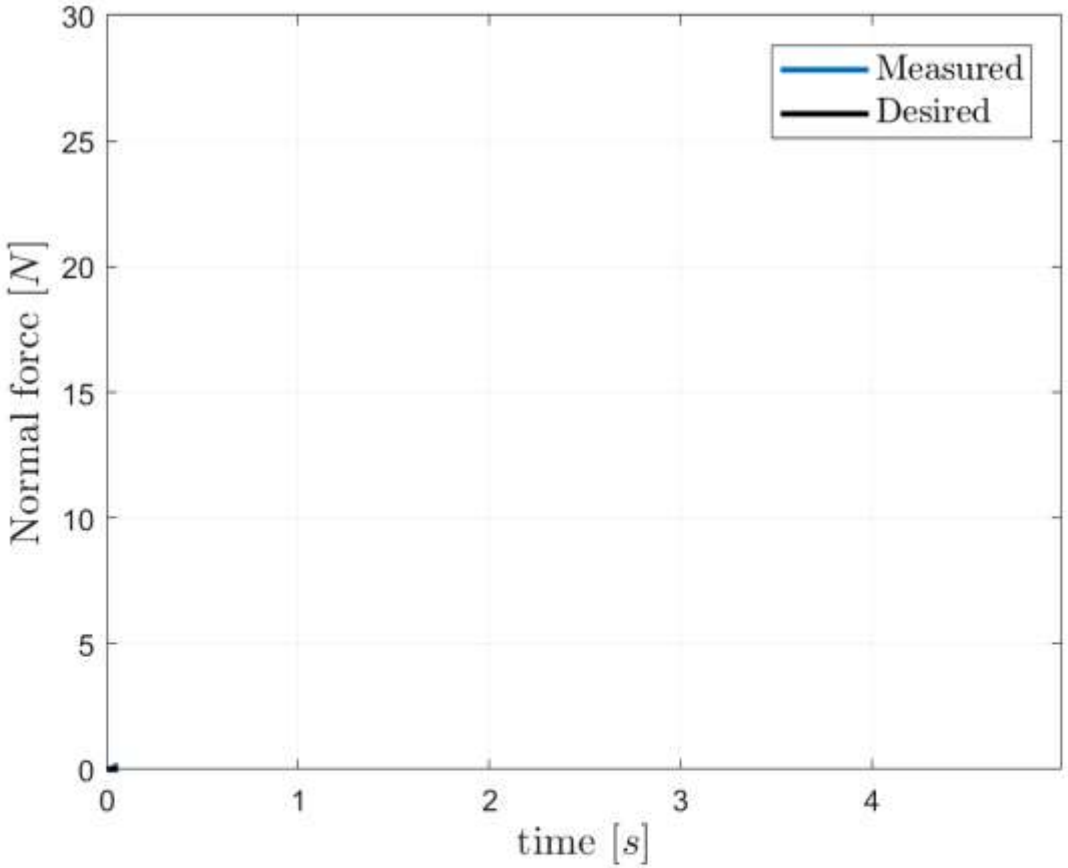
Define a nominal DS that creates a limit cycle on the surface.



Project on the surface.



Robotic Demonstration – Robustness to Various Disturbances



EPFL

Passivity Analysis

$W(\boldsymbol{x}, \dot{\boldsymbol{x}})$	Energy storage function
$V(\boldsymbol{x})$	Potential function
$0 \le s \le s_{max}$	Energy tank level
$0 \le \alpha(s) \le 1$ $0 \le \beta(s,u) \le 1$ $0 \le \gamma(s,v) \le 1$	Scalar variables controlling energy flow between robot and tank

❖ Passivity analysis

$$W(\boldsymbol{x}, \dot{\boldsymbol{x}}) = \underbrace{\frac{1}{2}\dot{\boldsymbol{x}}^T \boldsymbol{B}(\boldsymbol{x})\dot{\boldsymbol{x}}}_{\text{Kinetic energy}} + \underbrace{d_1 V(\boldsymbol{x})}_{\text{Potential function of } \boldsymbol{f}(\boldsymbol{x})} \qquad \text{with } \boldsymbol{f}(\boldsymbol{x}) = -\nabla V(\boldsymbol{x})$$

❖ Introduce energy tank "s" to restore passivity

$$\dot{W}(\boldsymbol{x}, \dot{\boldsymbol{x}}) = \underbrace{-d_1(\lambda_d(\boldsymbol{x}) - 1)\dot{\boldsymbol{x}}^T \boldsymbol{f}(\boldsymbol{x})}_{?} + \underbrace{F_d(\boldsymbol{x})\dot{\boldsymbol{x}}^T \boldsymbol{e}_1(\boldsymbol{x})}_{?} \underbrace{- \dot{\boldsymbol{x}}^T \boldsymbol{D}(\boldsymbol{x})\dot{\boldsymbol{x}}}_{\le 0} + \dot{\boldsymbol{x}}^T \boldsymbol{F}_{ext}$$

$$\dot{s} = \underbrace{\alpha(s)\dot{\boldsymbol{x}}^T \boldsymbol{D}(\boldsymbol{x})\dot{\boldsymbol{x}}}_{\text{Fill the tank}} \underbrace{- \beta(s,u) d_1(\lambda_d(\boldsymbol{x}) - 1) u - \gamma(s,v) F_d(\boldsymbol{x}) v}_{\text{Mainly contribute to empty the tank and implement non-passive actions}}$$

≤ 0 ? ≥ 0 ? ≤ 0

with $\begin{cases} u = \dot{\boldsymbol{x}}^T \boldsymbol{f}(\boldsymbol{x}) \\ v = \dot{\boldsymbol{x}}^T \boldsymbol{e}_1(\boldsymbol{x}) \end{cases}$

passivity not guaranteed

Passivity Analysis

$\boldsymbol{W}(\boldsymbol{x}, \dot{\boldsymbol{x}})$	Energy storage function
$V(\boldsymbol{x})$	Potential function
$0 \leq s \leq s_{max}$	Energy tank level
$0 \leq \alpha(s) \leq 1$ $0 \leq \beta(s,u) \leq 1$ $0 \leq \gamma(s,v) \leq 1$	Scalar variables controlling energy flow between robot and tank

- Control law adaptation if tank is depleted

$$\boldsymbol{F_u} = d_1 \lambda'_a(\boldsymbol{x}) \boldsymbol{f}(\boldsymbol{x}) + \gamma'(s,v) F_d(\boldsymbol{x}) \boldsymbol{e_1}(\boldsymbol{x}) - \boldsymbol{D}(\boldsymbol{x})\dot{\boldsymbol{x}} + \boldsymbol{g}(\boldsymbol{x})$$

Modulation gain adaptation: $\lambda'_a(\boldsymbol{x}) = \begin{cases} 1 & \text{if } s < 0 \text{ and } u < 0 \\ \lambda_a(\boldsymbol{x}) & \text{otherwise} \end{cases}$

Desired force profile scaling: $\gamma'(s,v) = \begin{cases} 1 & \text{if } v < 0 \\ \gamma(s,v) & \text{otherwise} \end{cases}$

- Passivity analysis with tank dynamics

with $\begin{cases} u = \dot{\boldsymbol{x}}^{\boldsymbol{T}} \boldsymbol{f}(\boldsymbol{x}) \\ v = \dot{\boldsymbol{x}}^{\boldsymbol{T}} \boldsymbol{e_1}(\boldsymbol{x}) \end{cases}$

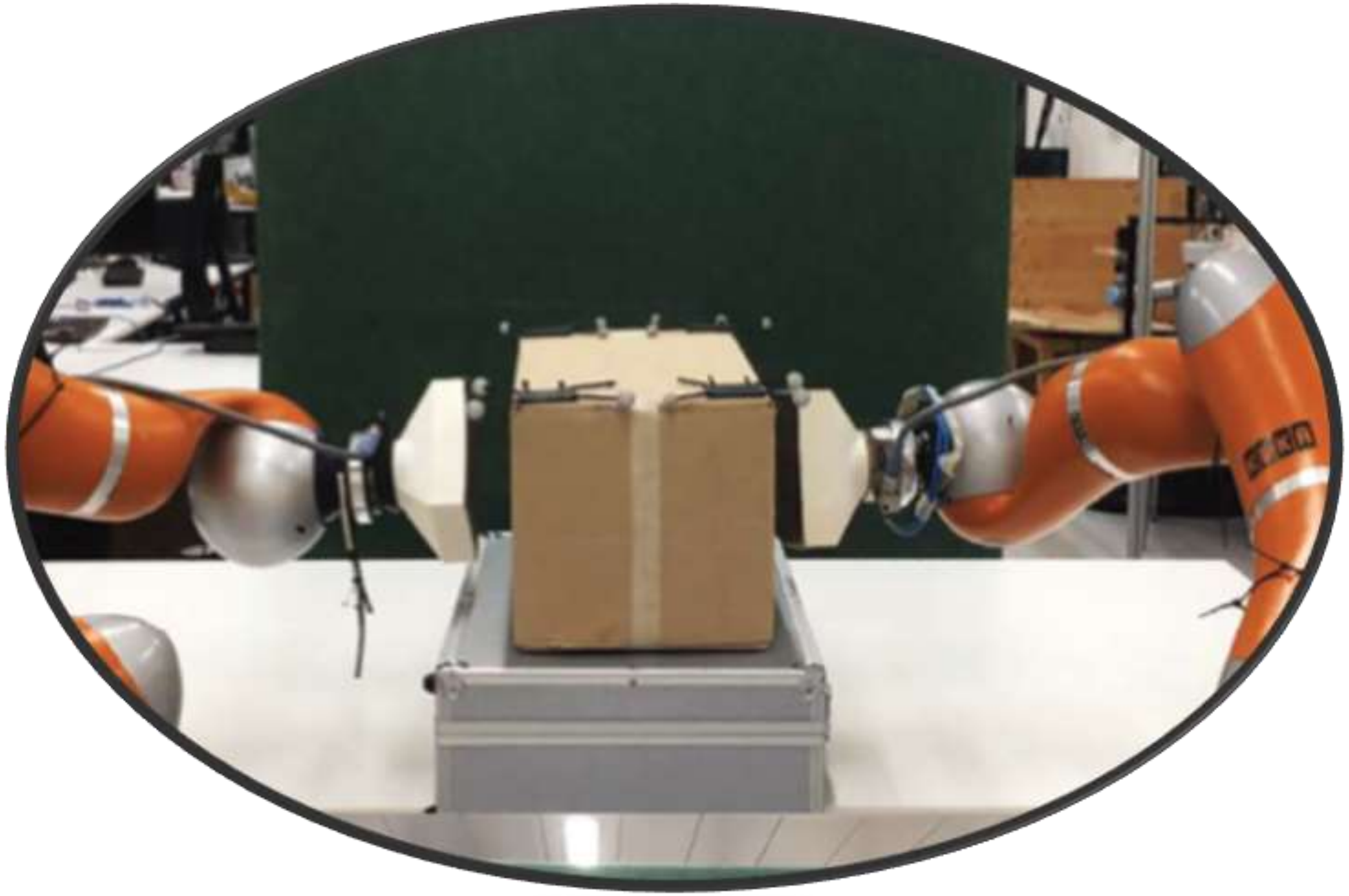
$$W(\boldsymbol{x}, \dot{\boldsymbol{x}}, s) = \frac{1}{2} \dot{\boldsymbol{x}}^{\boldsymbol{T}} \boldsymbol{B}(\boldsymbol{x}) \dot{\boldsymbol{x}} + d_1 V(\boldsymbol{x}) + s$$

$$\dot{W}(\boldsymbol{x}, \dot{\boldsymbol{x}}, s) = \underbrace{d_1(\lambda'_a(\boldsymbol{x}) - 1)\big(1 - \beta(s,u)\big)u + F_d(\boldsymbol{x})\big(\gamma'(s,v) - \gamma(s,v)\big)v - \big(1 - \alpha(s)\big)\, \dot{\boldsymbol{x}}^{\boldsymbol{T}} \boldsymbol{D}(\boldsymbol{x}) \dot{\boldsymbol{x}}}_{\leq 0} + \dot{\boldsymbol{x}}^{\boldsymbol{T}} \boldsymbol{F_{ext}}$$

Passivity is restored

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

EPFL

Variables to Control Bimanual Platform

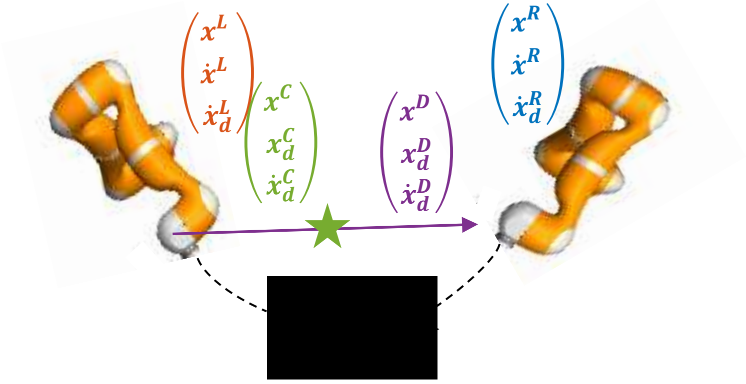
- **Robots' center position and distance vector**

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

- **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**

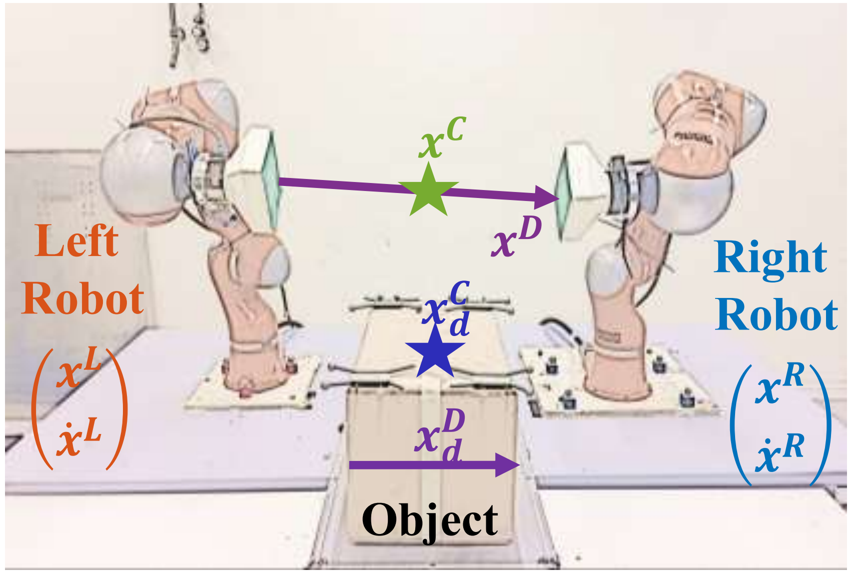
Closing **dynamics**

with: $A_C, A_D \geq 0$

- Robots' **nominal dynamics:**

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

Center positioning
+
Closing **dynamics**



Force Desired for Bimanual Platform

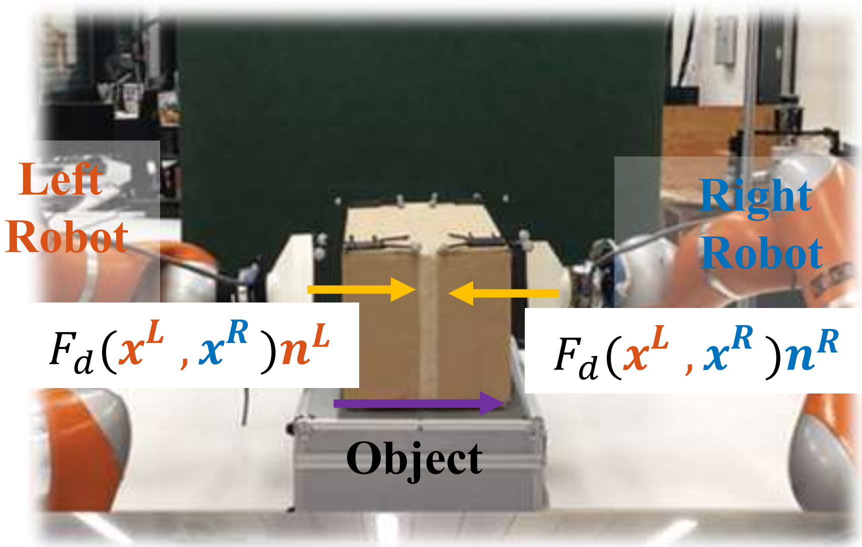
- Robots' **desired modulated dynamics:**

$$\begin{cases} \dot{x}_d^R = f^R(x^L, x^R) + \dfrac{F_d(x^L, x^R)}{d_1^R} n^R \\ \dot{x}_d^L = f^L(x^L, x^R) + \dfrac{F_d(x^L, x^R)}{d_1^L} n^L \end{cases}$$

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

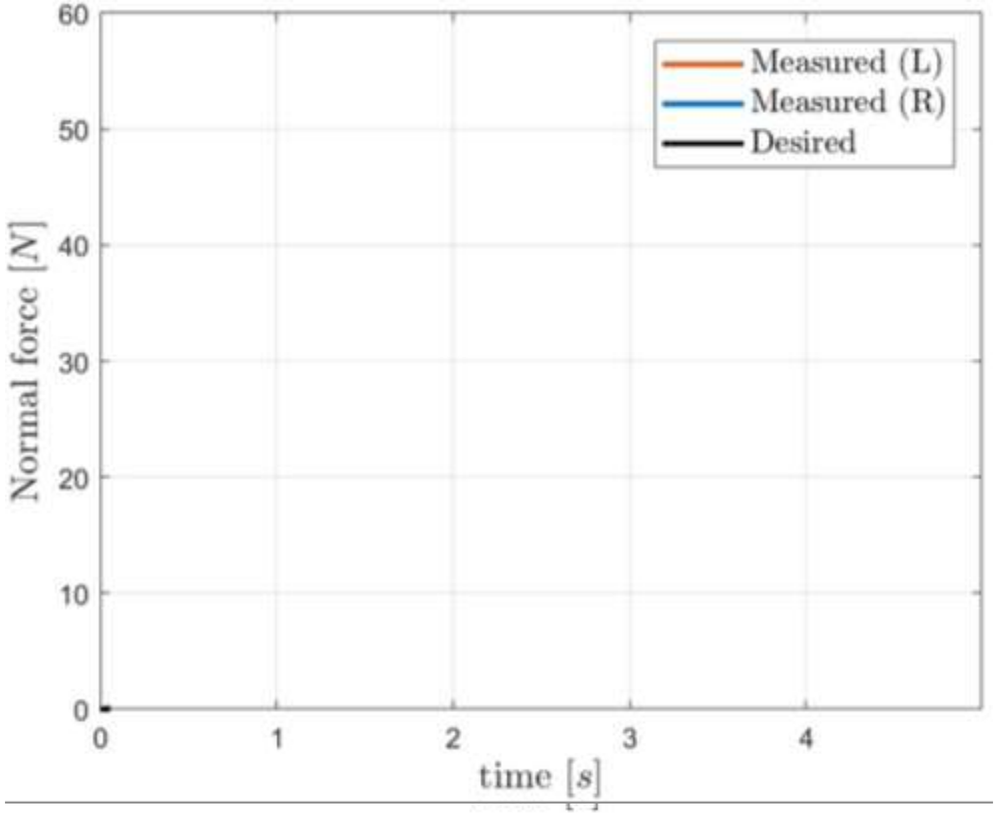
$F_d(x^L, x^R)$: **Desired contact force**

d_1^L, d_1^R: **Impedance gains**



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

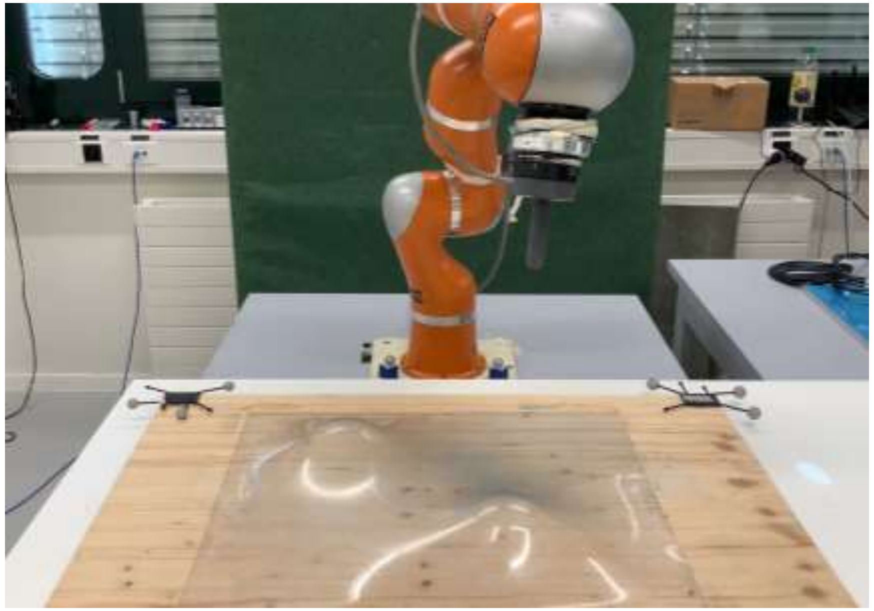
Robot Demonstration



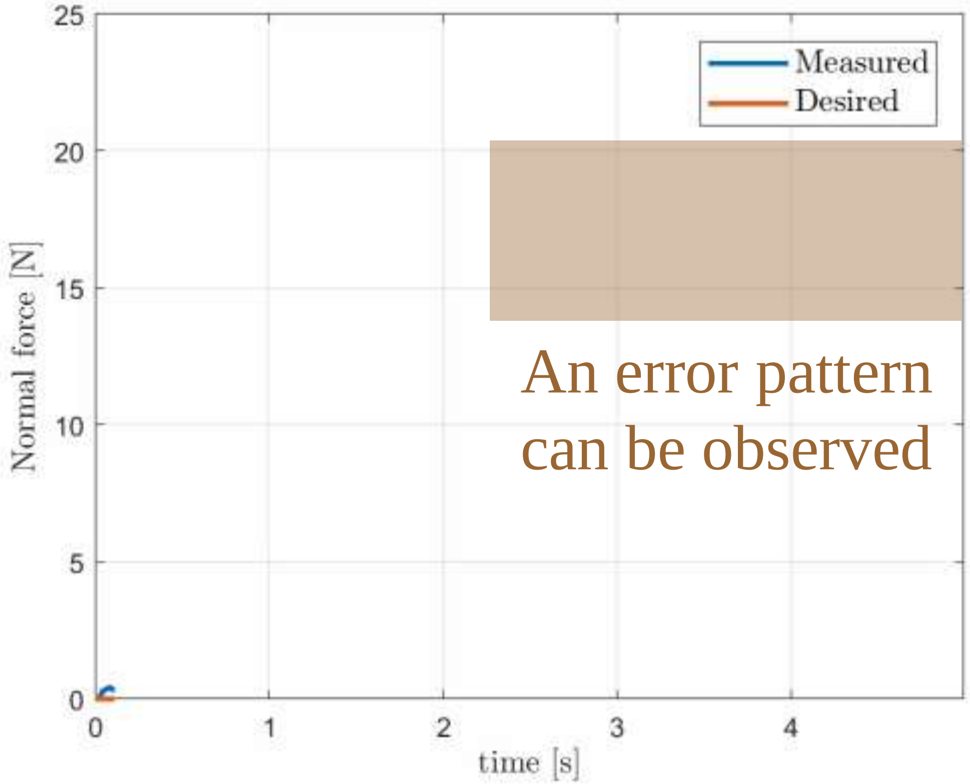
Learning Force Adaptation

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:

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

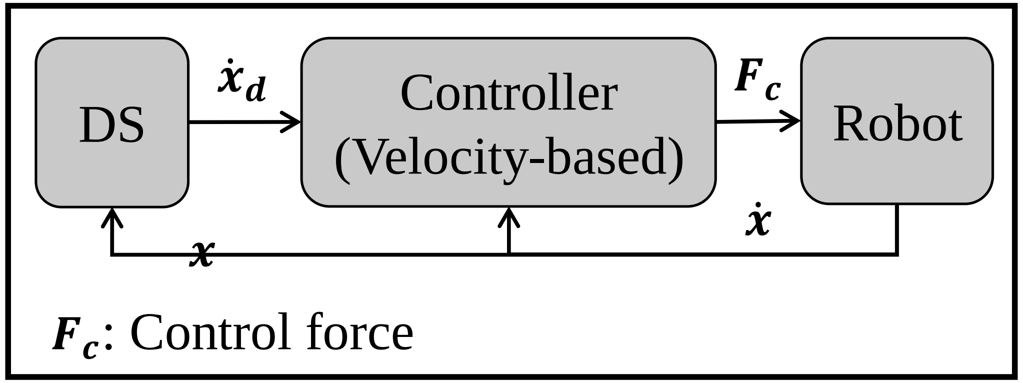
Nominal DS (responsible of motion) → $\boldsymbol{f}(\boldsymbol{x})$

Modulation term (responsible of contact force) → $\boldsymbol{f}_n(\boldsymbol{x})$

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

$$\boldsymbol{f}_n(\boldsymbol{x}) = \frac{F_d(\boldsymbol{x}) + \tilde{F}_d(\boldsymbol{x}, \boldsymbol{\theta})}{d_1}\boldsymbol{n}(\boldsymbol{x})$$

DS in closed-loop configuration

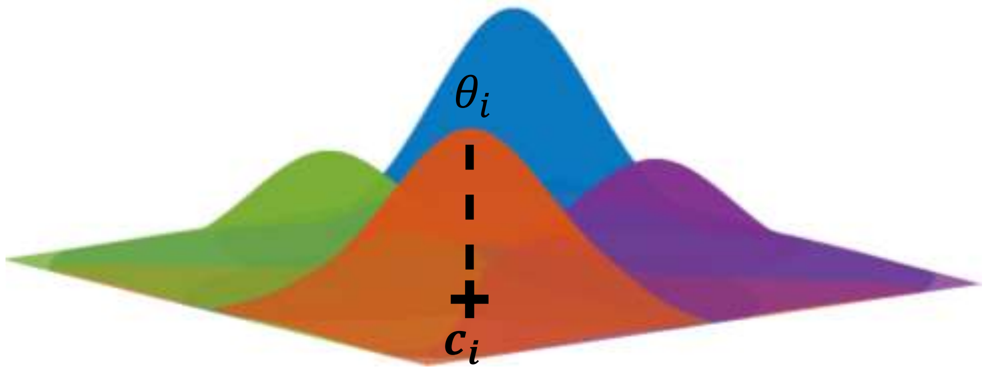


$\dot{\boldsymbol{x}}_d \in \mathbb{R}^3$	Desired dynamics
$\boldsymbol{f}(\boldsymbol{x}) \in \mathbb{R}^3$	Nominal DS
$\boldsymbol{f}_n(\boldsymbol{x}) \in \mathbb{R}^3$	Normal modulation term
$F_d(x) \geq 0$	Desired force profile
$\boldsymbol{n}(\boldsymbol{x}) \in \mathbb{R}^3$	Normal direction to the surface
$d_1 \geq 0$	Impedance gain

State-Dependent Force Correction

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

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

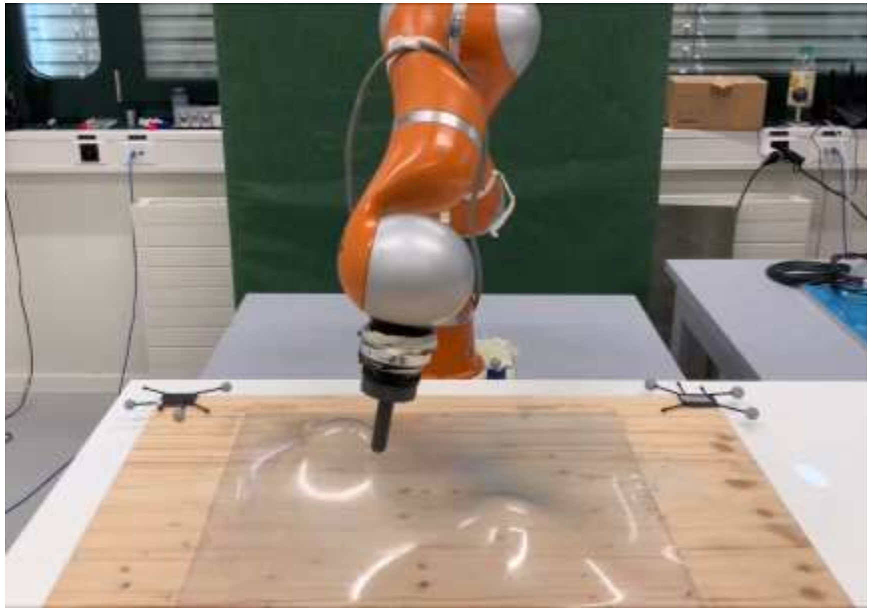


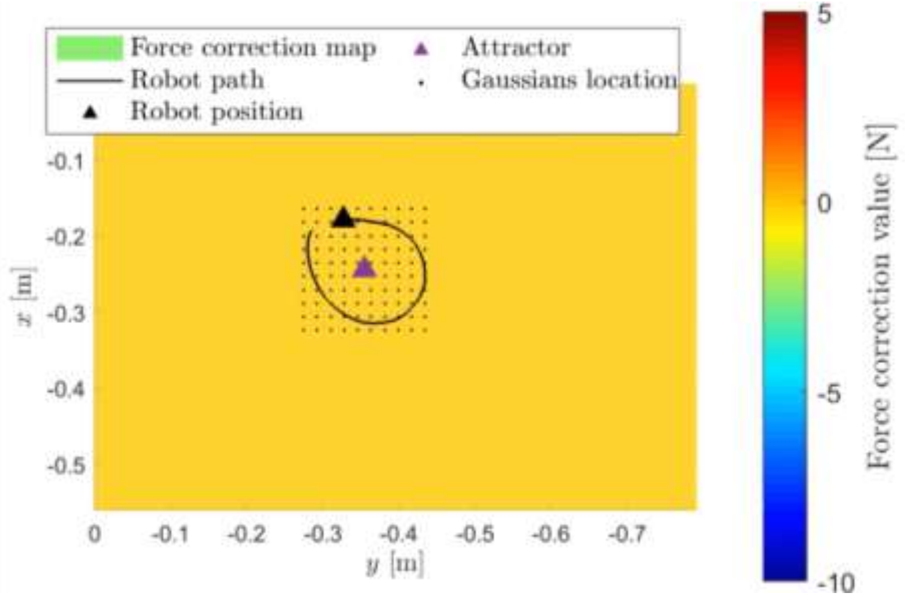
Hyper-parameters
- K: Number of Gaussian
- $\boldsymbol{c_i}$: Center position of Gaussian i
- σ: Kernel width

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

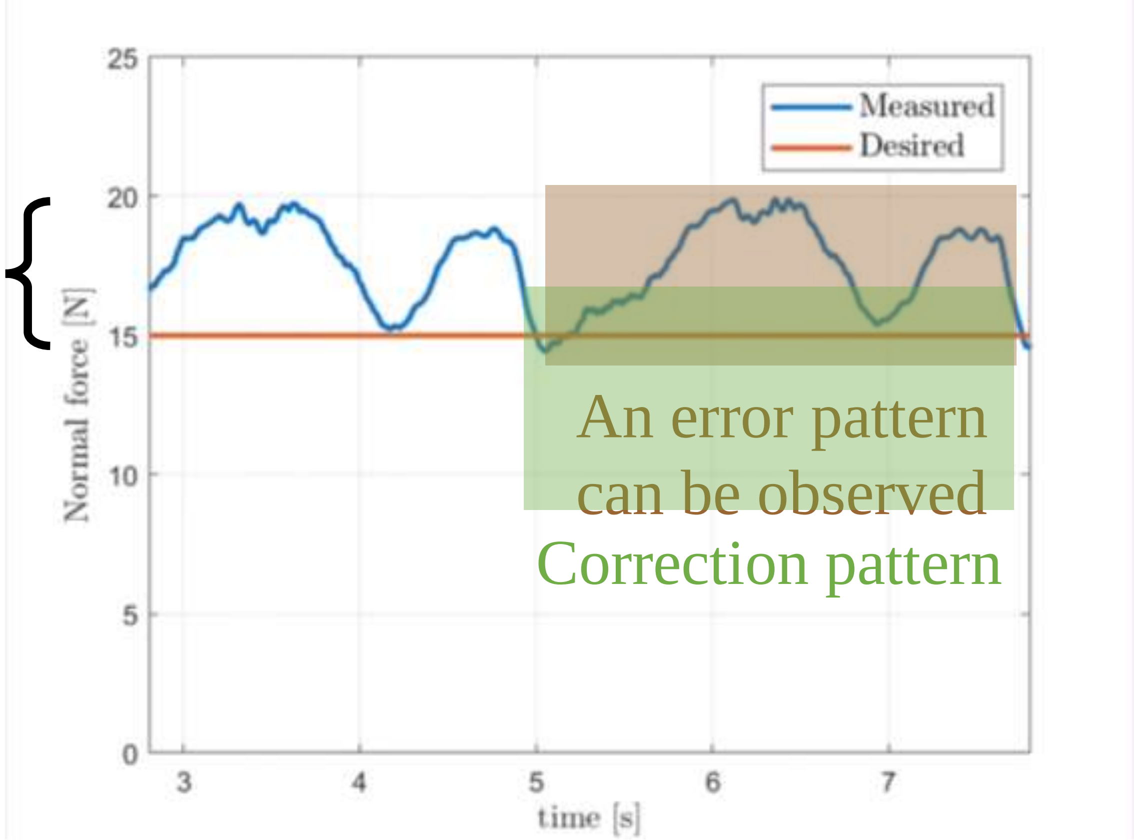
$$F_e = F_d(\boldsymbol{x}) - \boldsymbol{n}(\boldsymbol{x})^{\boldsymbol{T}} \boldsymbol{F_m}$$

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

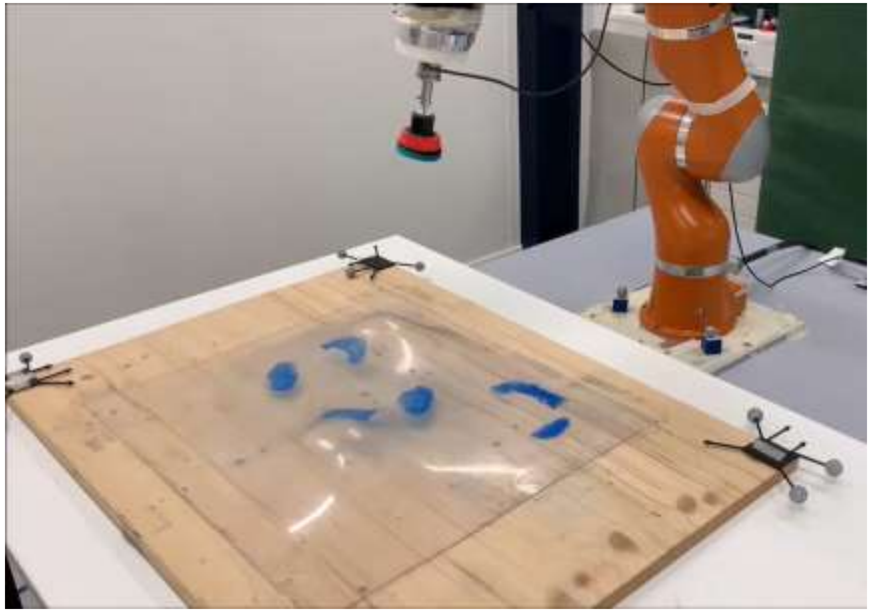


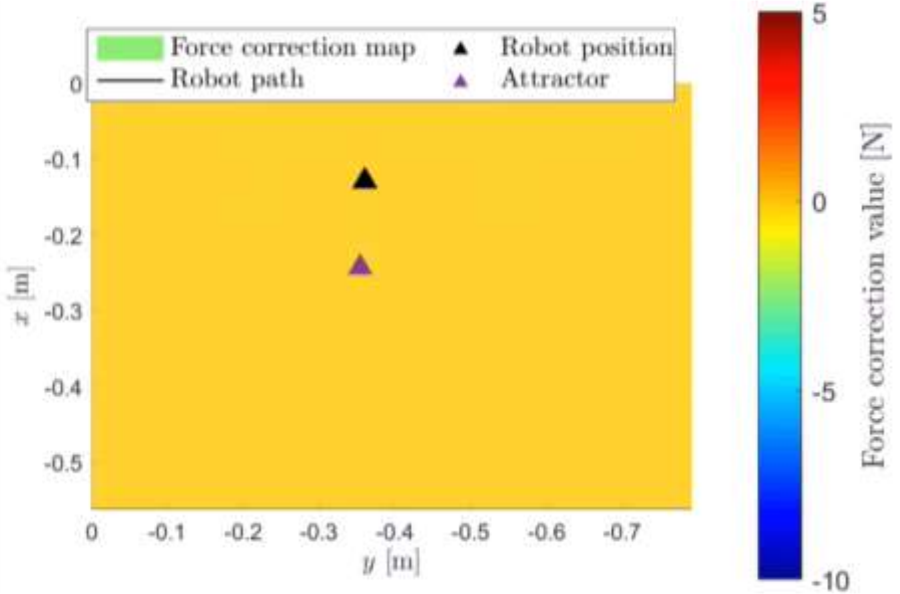


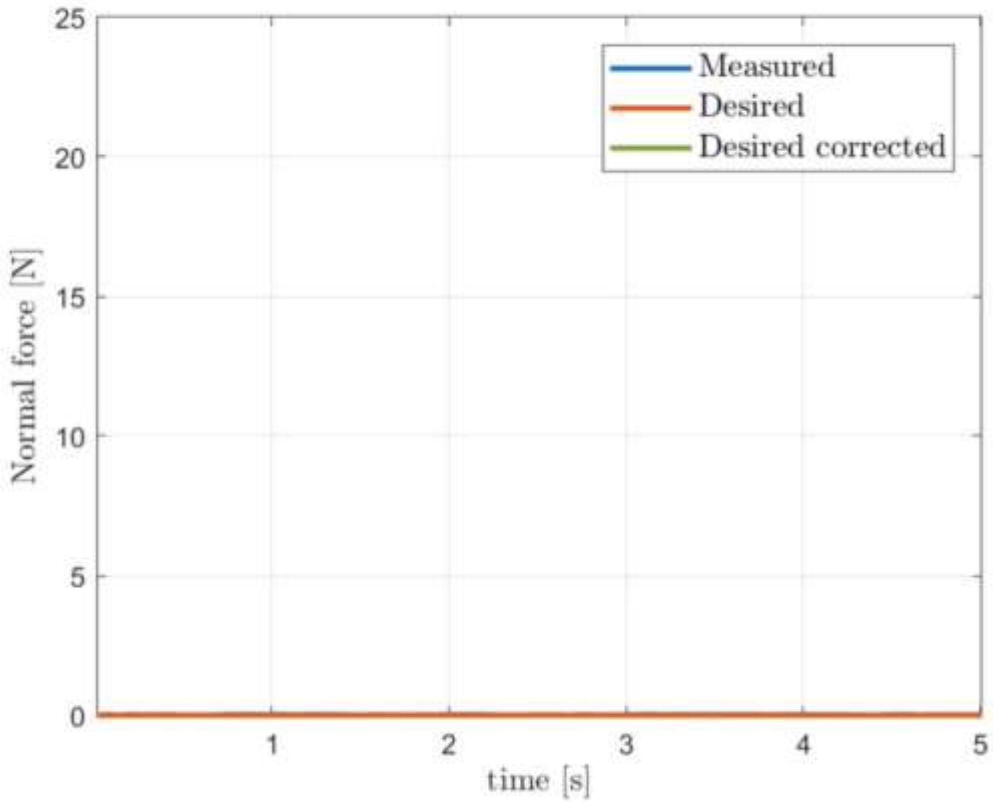
Initial error range



4x




Force correction map
Robot position
Robot path
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
25
20
15
10
5
0
Normal force [N]
1
2
3
4
5
time [s]

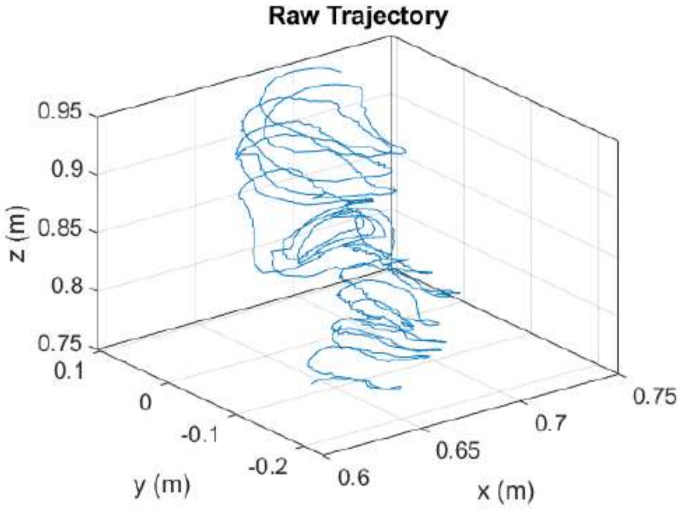
4x

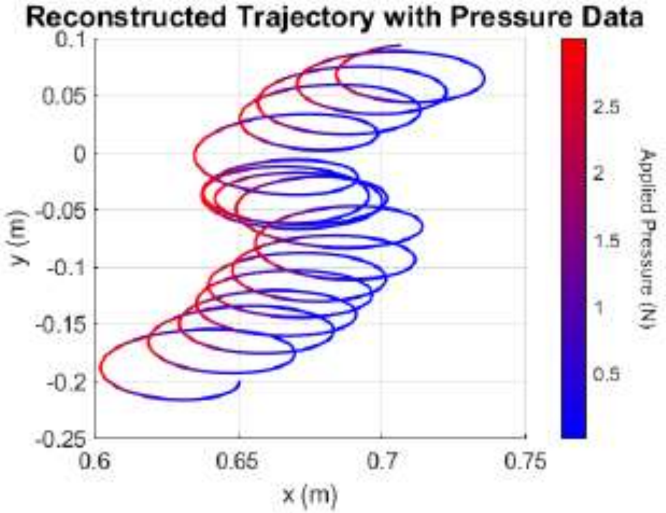
Other Examples of Learned-Force Based Control with DS

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).

Learning Arm Massage

Human recordings





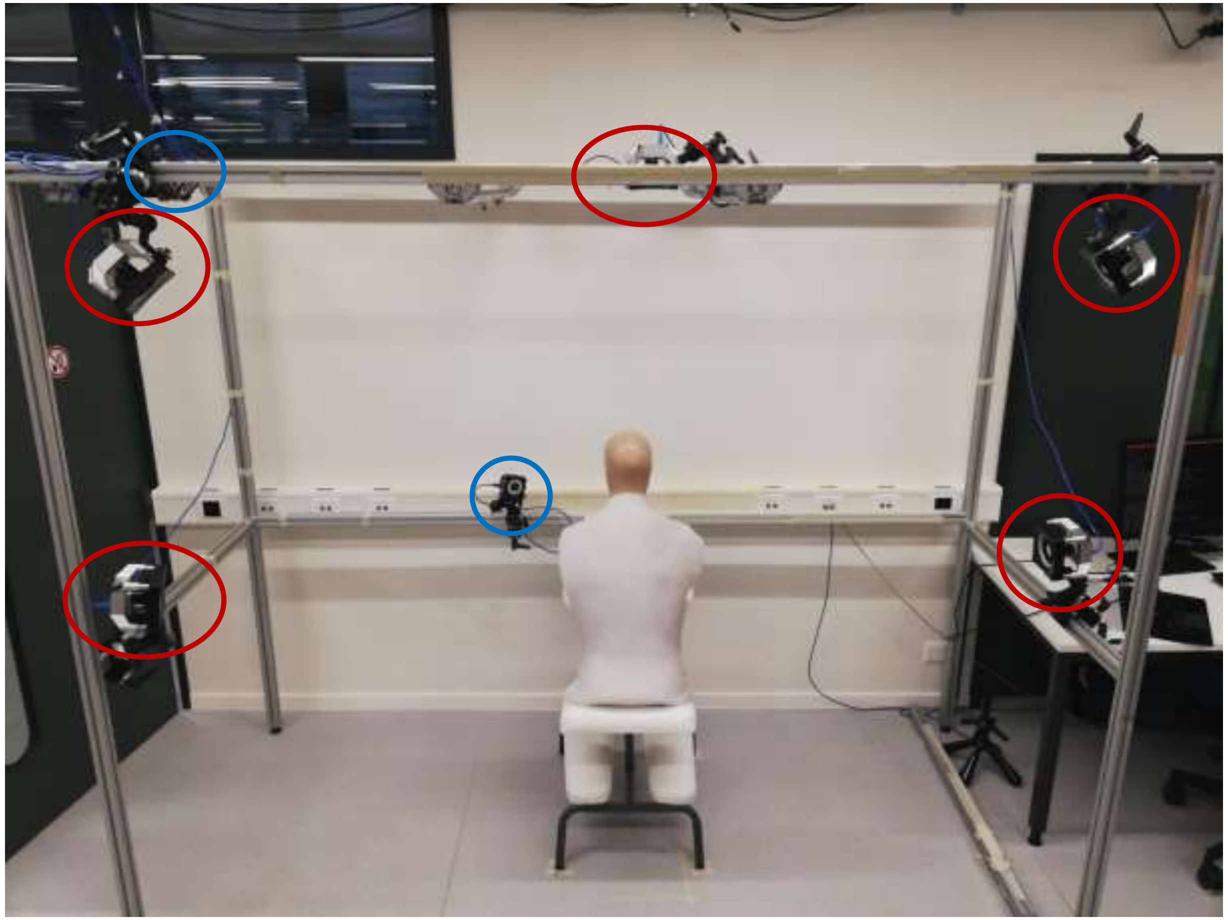
Modeling

Robotic massage

Supported by Samsung

EPFL

Motion Tracking



- **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**

Force Tracking

- Pressure Profiles Systems, FingerTPS

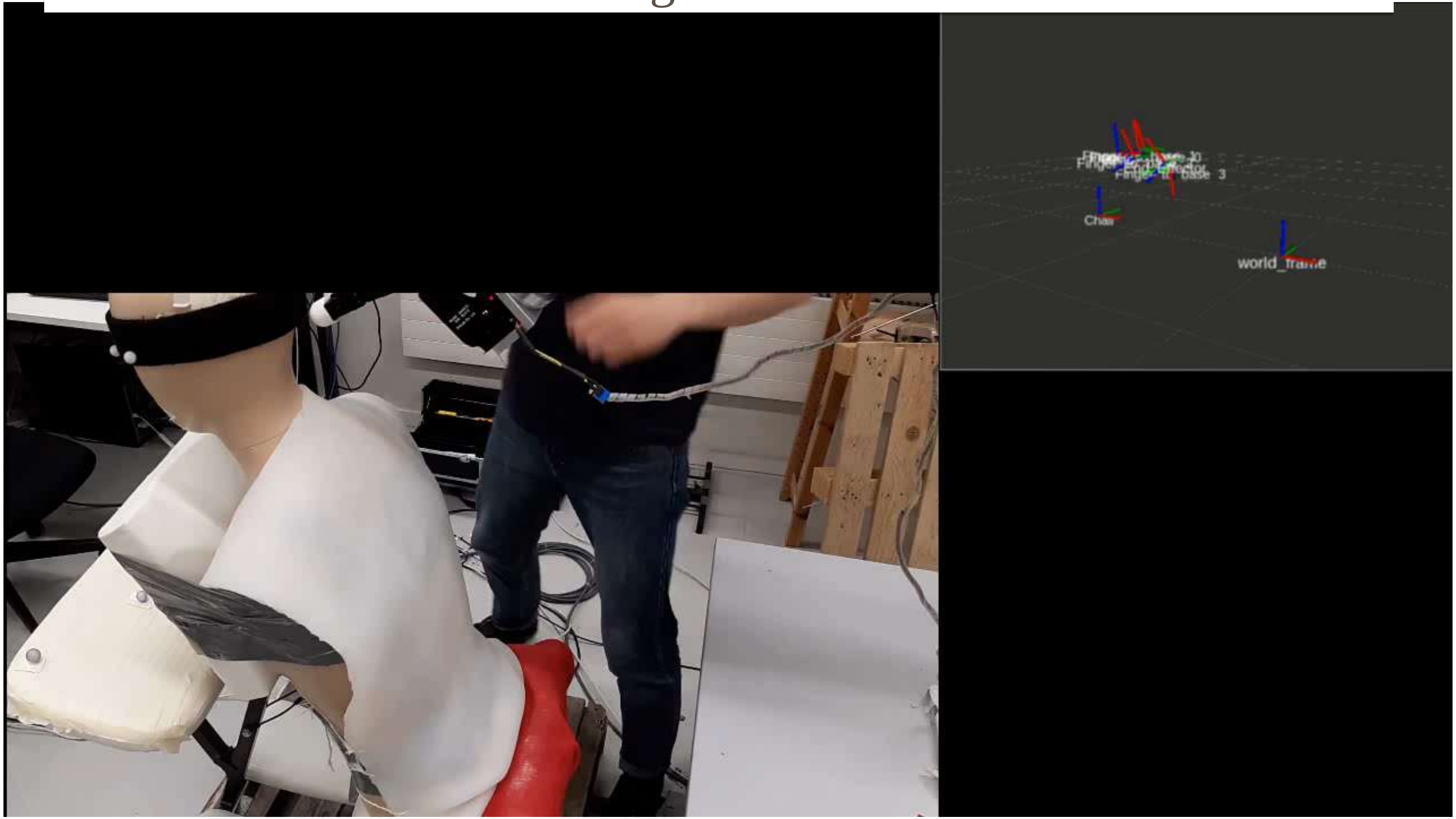
+ Sensors have multiple sensing areas
+ Multiple sizes
+ Fabric covered sensor
- Force mapped across entire surface
- 40 Hz sampling rate

Demonstration by Massage Therapist

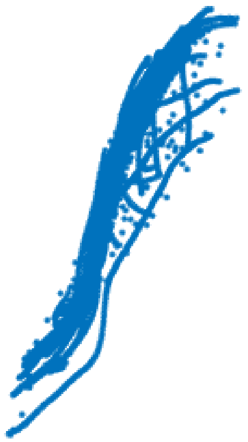
- No Thumb, Hands Together

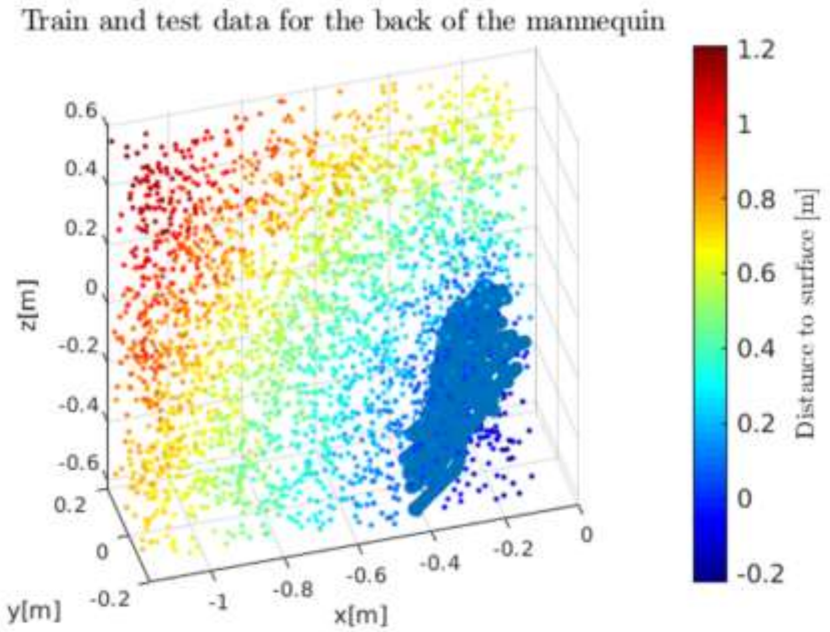
- Thumb only

Modeling Back Surface



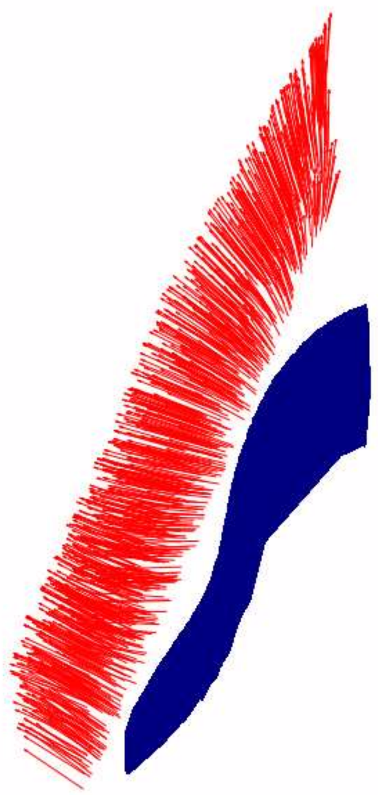
Modeling Back Surface

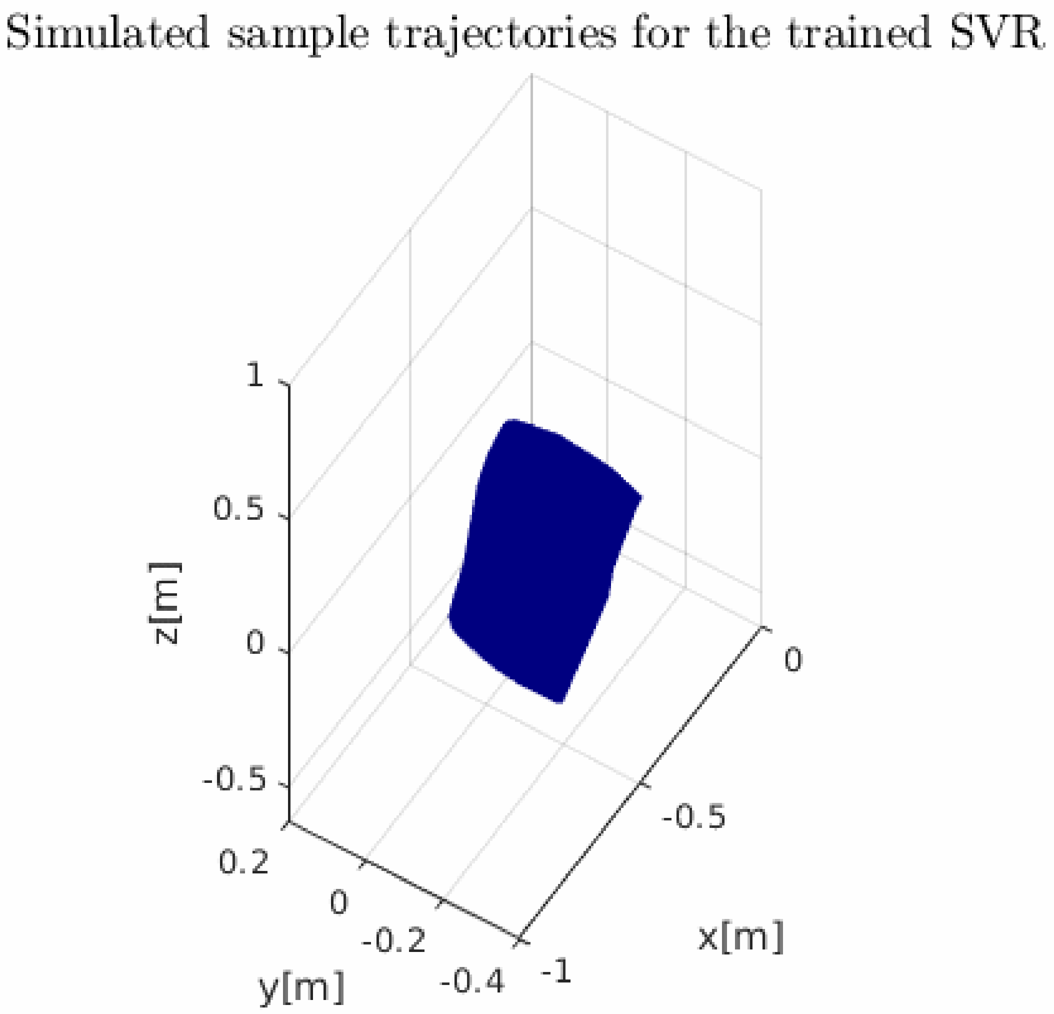




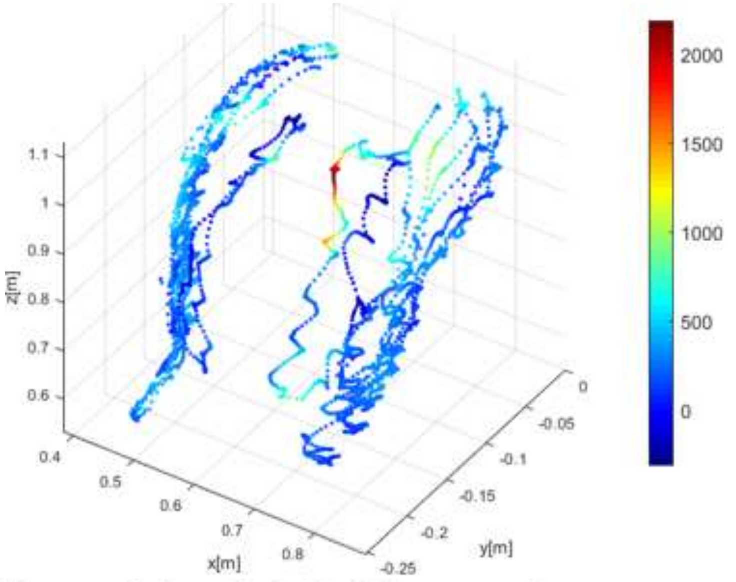
- Cross-validation used for model validation with 5 folds.
- Grid search to find optimal C and σ for the SVR.
- Best testing accuracy achieved with a RMSE of 0.52 [cm] for points on the back

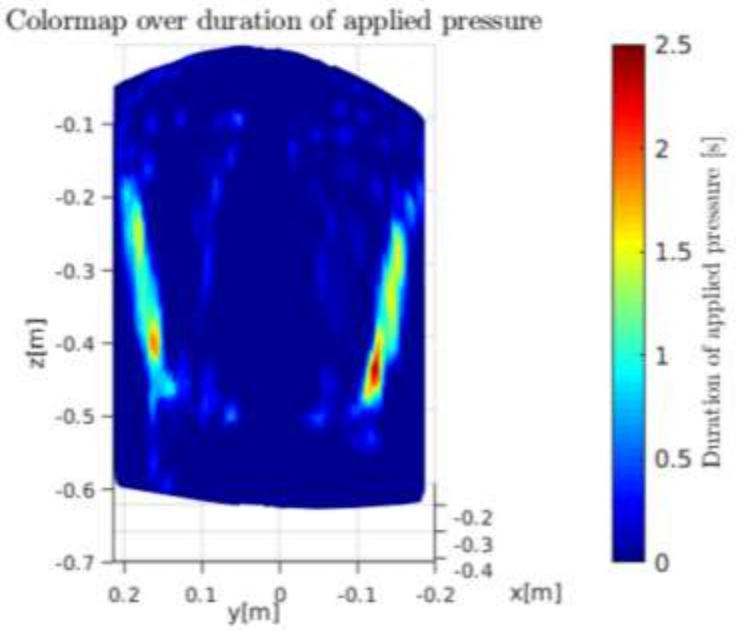
Reconstructed surface & gradients for the trained SVR

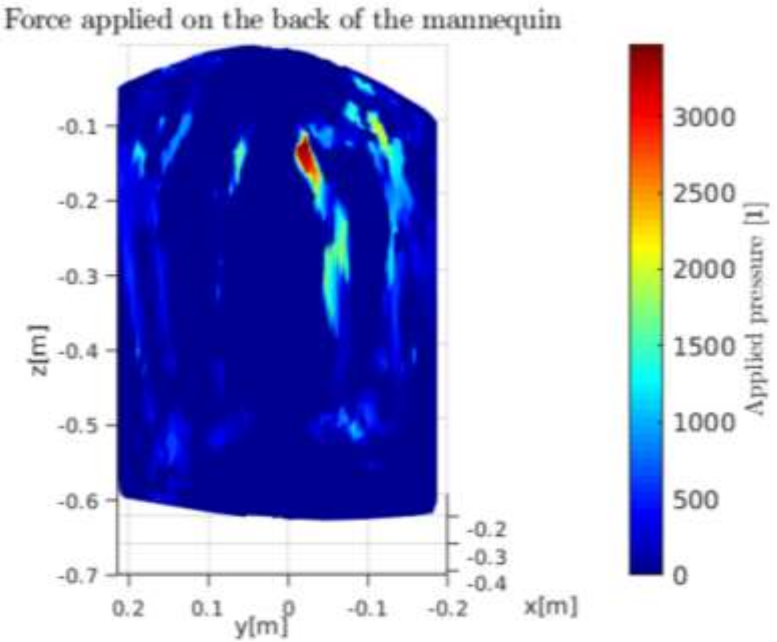


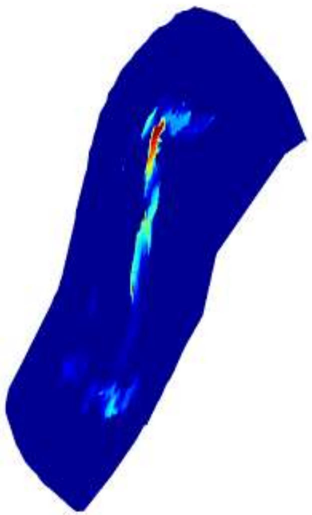


Test data from movements on the back of the mannequin

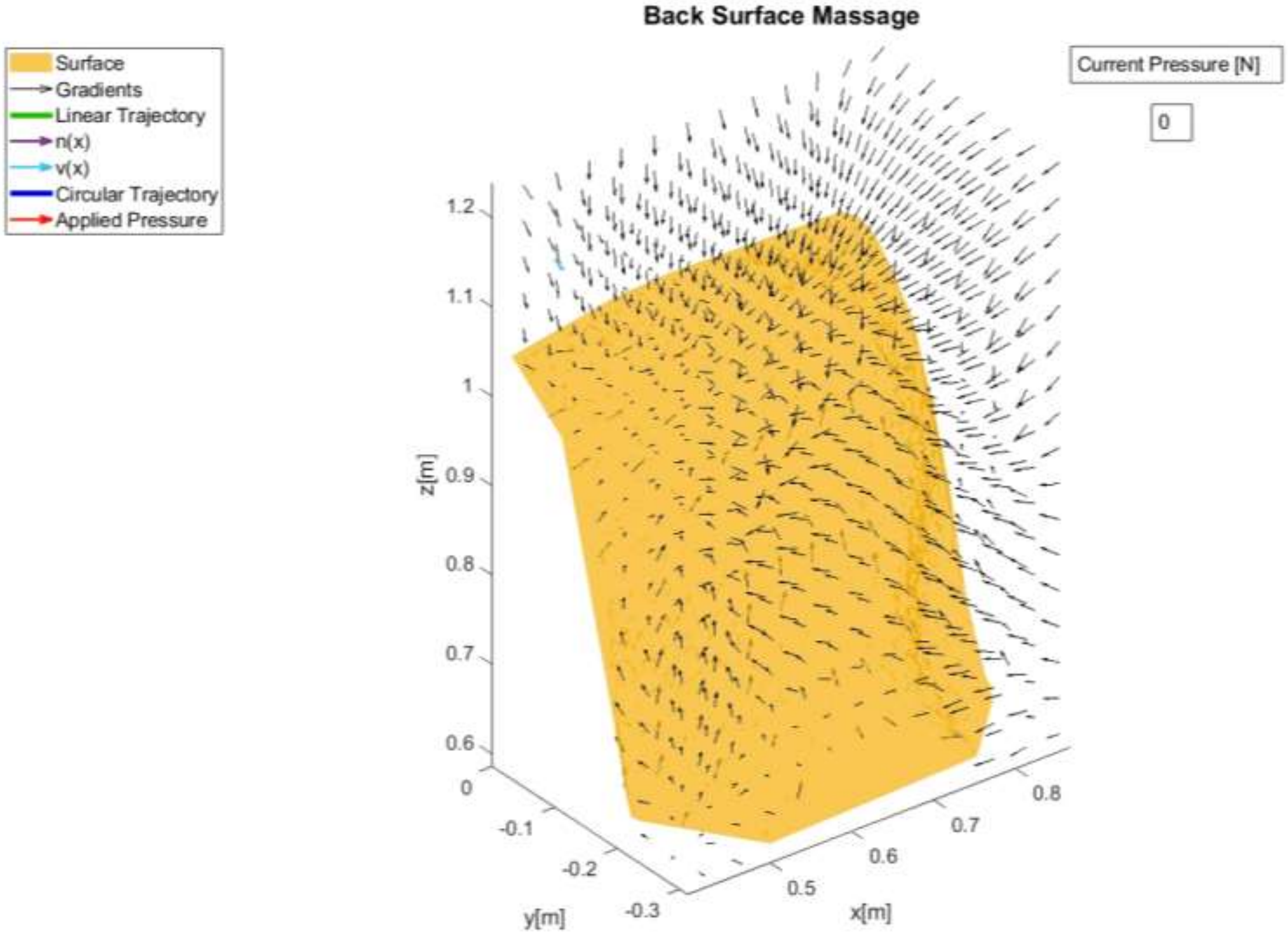








Reconstruction of Motion on Back Surface

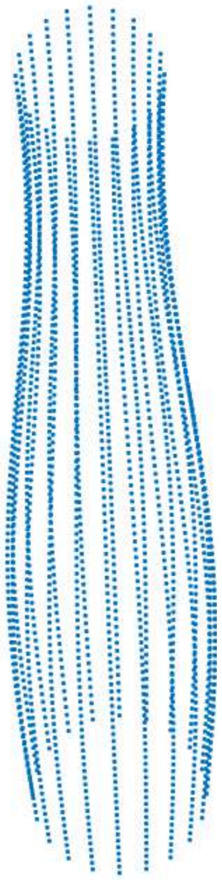


Learning Arm Massage

Khoramshahi et al, ICRA 2020

Supported by Samsung

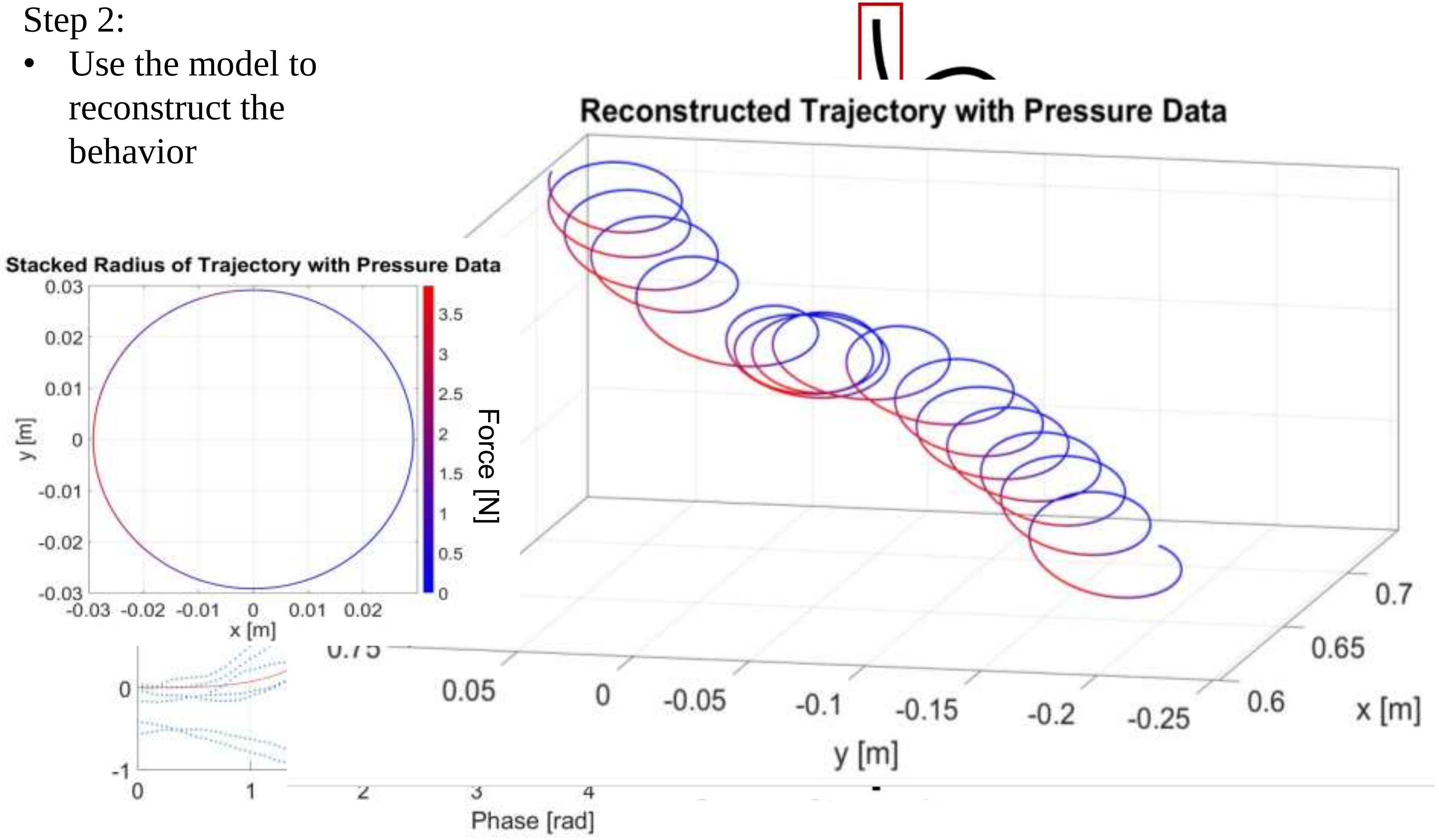
Create artificial data for the arm



Train SVR and predict the surface

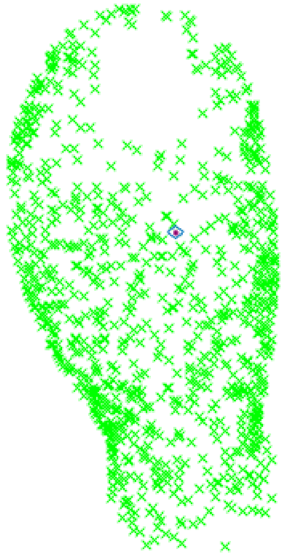
Step 2:

- Use the model to reconstruct the behavior

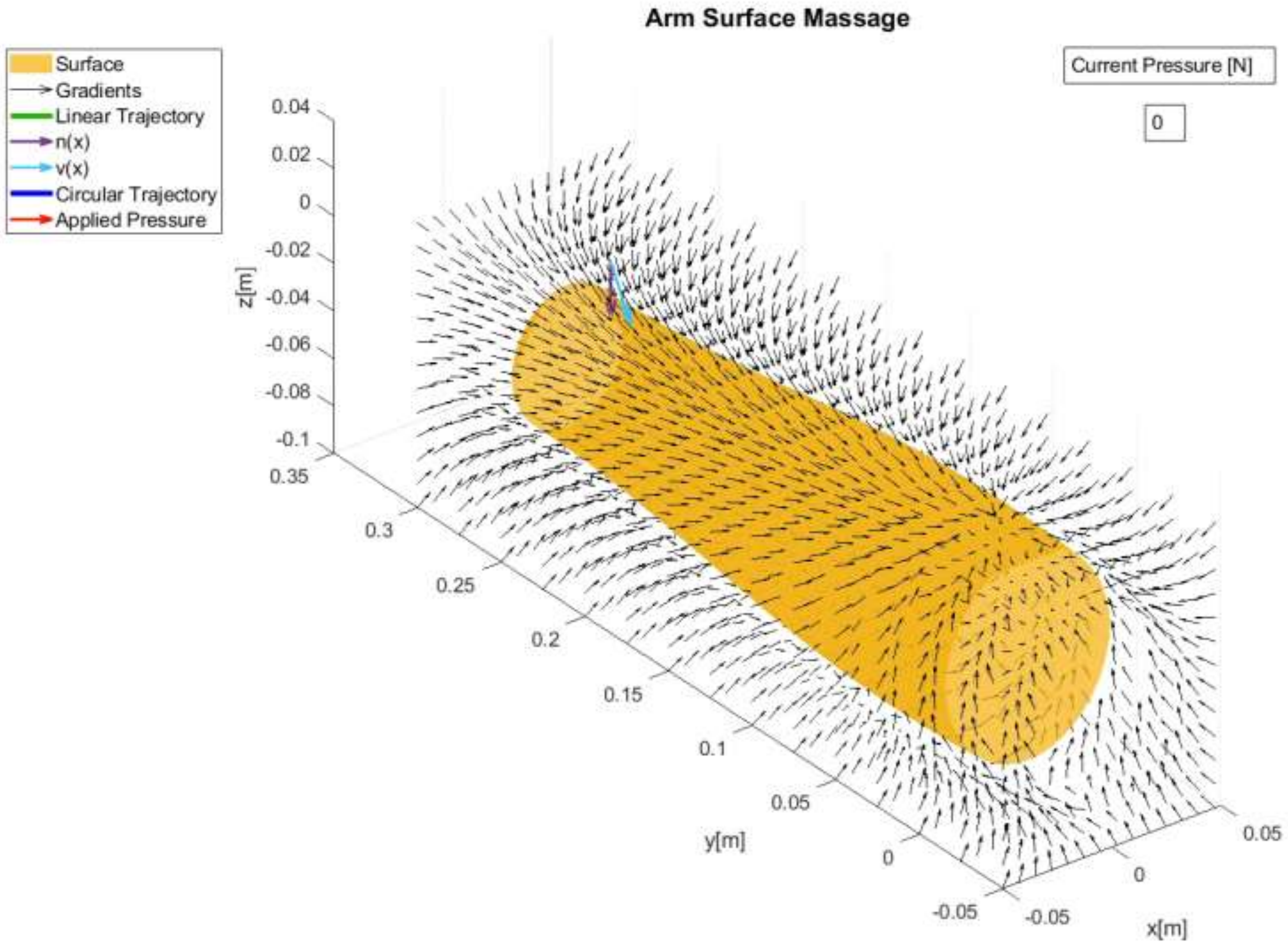


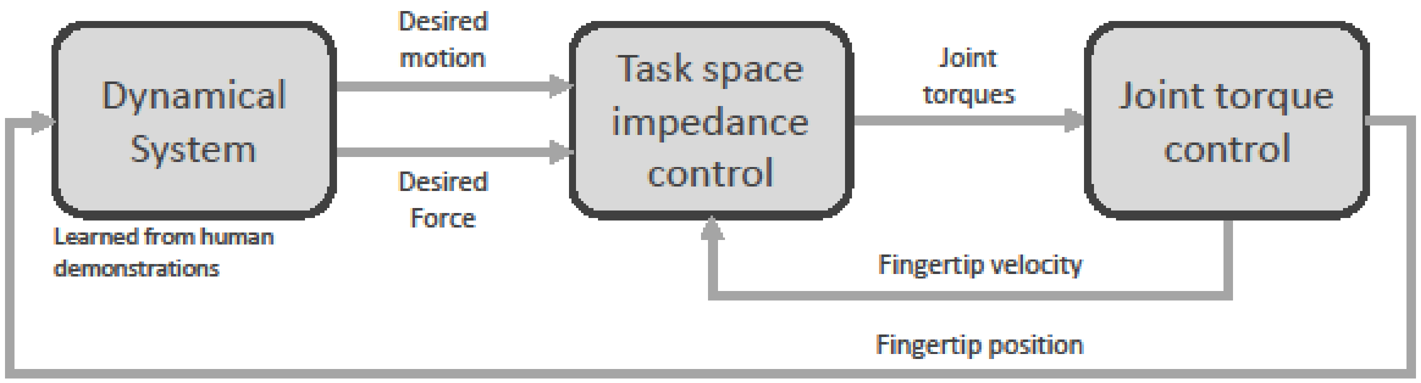
Khoramshahi et al, ICRA 2020

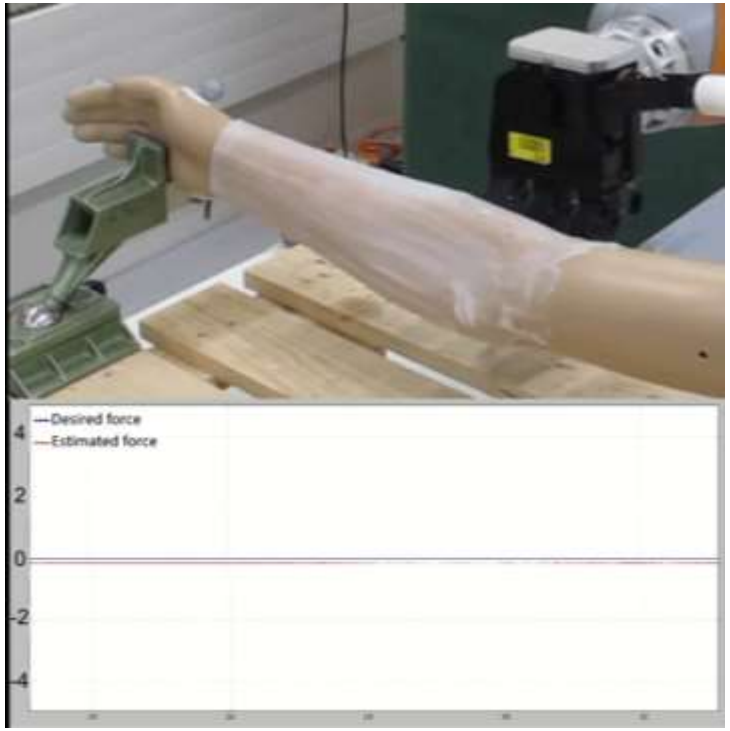
Reconstruction of Motion on Arm Surface



Reconstruction of Motion on Arm Surface




Dynamical System
Desired motion
Desired Force
Task space impedance control
Joint torques
Joint torque control
Learned from human demonstrations
Fingertip velocity
Fingertip position


Desired force
Estimated force
4
2
0
-2
-4

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