Extensions to control with DS

Coupling DS & Examples of Applications

Multi-attractor DS, Switching across DS, On-line update of DS

Coupled DS

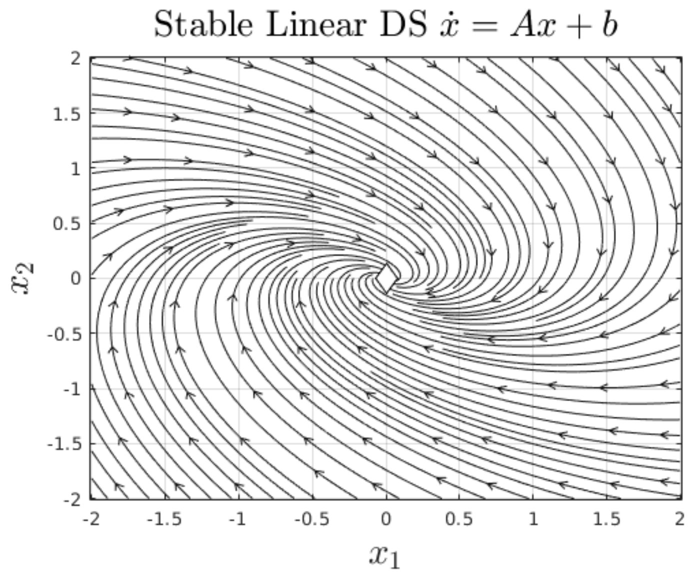
Isolated Dynamics

Until now, we have considered a DS control law in isolation.

$$\dot{x} = f(x)$$

The state of the system x at time t depends solely on the state of x at previous time step.

$$x(t) = x(t-1) + f(x)dt$$

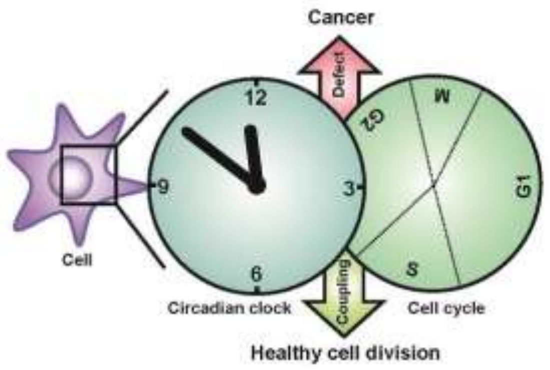


In general, systems do not function in isolation.
Their dynamics is influenced by the dynamics of their environment.

Coupled DS in Nature

In Biology, *the circadian clock and the cell cycle are modelled as two periodic processes that are coupled with one another. "In mammalian cells, circadian clocks consist of autonomous feedback loop oscillators ticking with an average period of about 24 h and controlling many downstream cellular processes."*

Droin, C., Paquet, E.R. & Naef, F. Low-dimensional dynamics of two coupled biological oscillators. Nat. Phys. 15, 1086–1094 (2019).

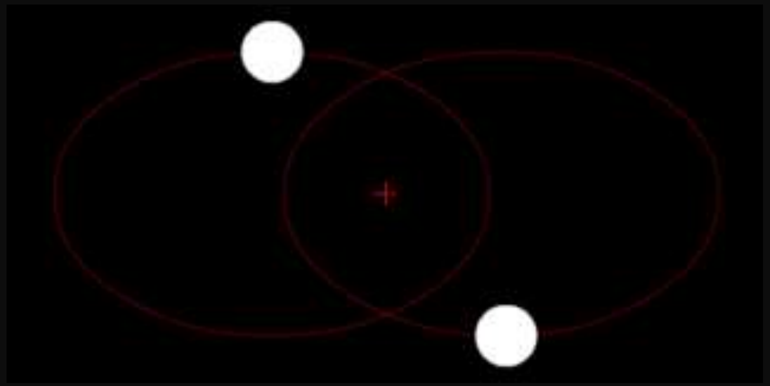


Shostak, A. Circadian Clock, Cell Division, and Cancer: From Molecules to Organism. Int. J. Mol. Sci. 2017, 18, 873.

In Astrophysics, a pair of close by stars act as a coupled system

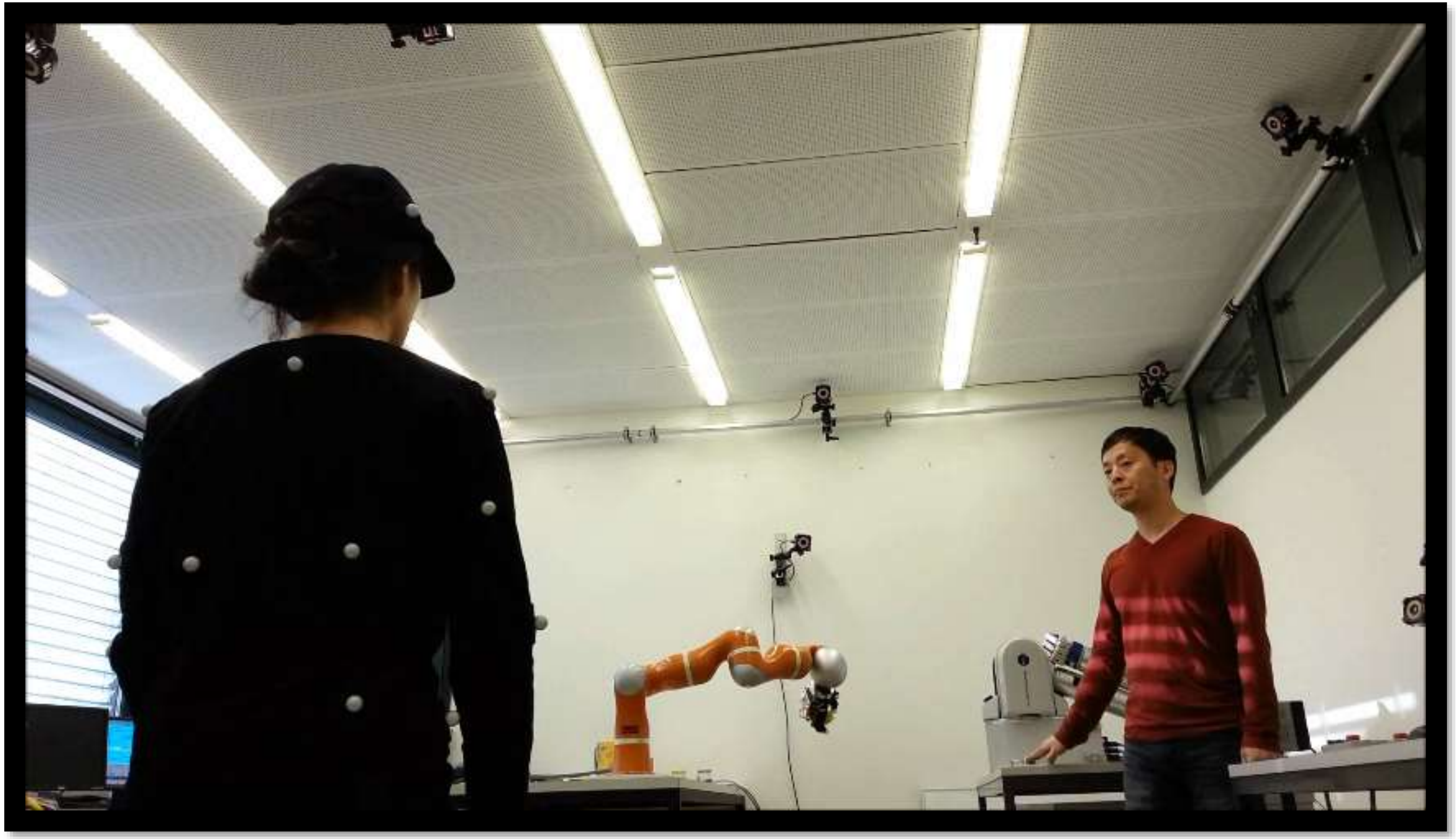
The two stars' motion are influenced by their relative masses.
In a perfectly balanced system, one would obtain a perfect oscillator, where each star rotates around an ellipse.

Heintz, Wulff Dieter. *Double stars*. Vol. 15. Springer Science & Business Media, 2012.



Coupled DS are often oscillatory in nature, but coupling can also be done to discrete movements, e.g. reach and grasp movements as well as bimanual reaching movements are coupled in amplitude and speed
(Jeannerod M (1984) The timing of natural prehension movements. J Mot Behav 16:235–254; Swinnen et al. " Behavioural Brain Research, 2001.

Arm-hand coordination



Coupled DS

In DS theory, the concept of "coupling" is used to express dependencies across dynamics.

Consider two variate x and y with dynamics:

$\dot{x} = f_x(x)$

$\dot{y} = f_y(y)$

x and y are coupled if any of the following happens:

$$\begin{cases} \dot{x} = f_x(x, y) \\ \dot{y} = f_y(y) \end{cases} \qquad \begin{cases} \dot{x} = f_x(x) \\ \dot{y} = f_y(x, y) \end{cases} \qquad \begin{cases} \dot{x} = f_x(x, y) \\ \dot{y} = f_y(x, y) \end{cases}$$

In the case of a linear DS on a multi-dimensional variable:

$x \in \mathbb{R}^N, \quad \dot{x} = Ax$

The dynamics on each dimension are uncoupled only if A is diagonal.

Coupled DS: Stability

Stability can be inherited through coupling

Consider the system:

$$\dot{x} = f_x(x),$$
$$\dot{y} = f_y(y,x) = A_y yx$$

If $\lim_{t\to\infty} f_x(x) = 0$ then, $\lim_{t\to\infty} f_y(y,x) = 0$

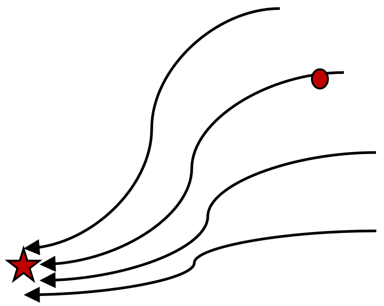
If the two DS are coupled, the stability of the coupled system must be studied.

$$\begin{cases} \dot{x} = f_x(x,y) \\ \dot{y} = f_y(x,y) \end{cases} \quad \begin{cases} \dot{x} = ax + by \\ \dot{y} = cx + dy \end{cases}$$

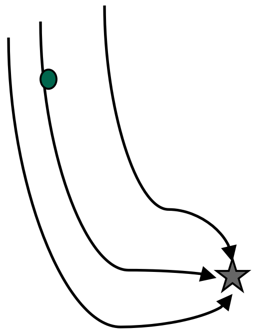
$$\begin{pmatrix} \dot{x} \\ \dot{y} \end{pmatrix} = \underbrace{\begin{bmatrix} a & b \\ c & d \end{bmatrix}}_{A} \begin{pmatrix} x \\ y \end{pmatrix}$$

Study the eigenvalues of A.

Uncoupled DS

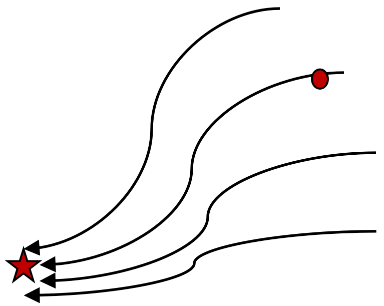


$$\dot{x} = f_x(x)$$

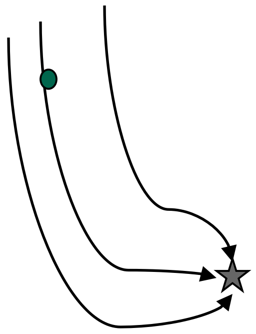


$$\dot{y} = f_y(y)$$

Coupled DS

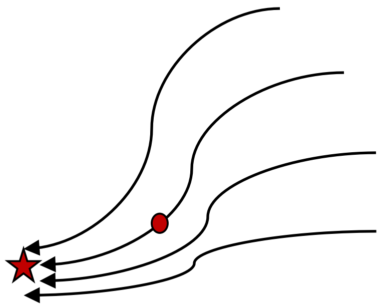


$$\dot{x} = f_x(x)$$

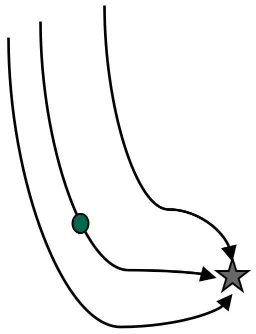


$$\dot{y} = f_y(y, x)$$

Coupled DS



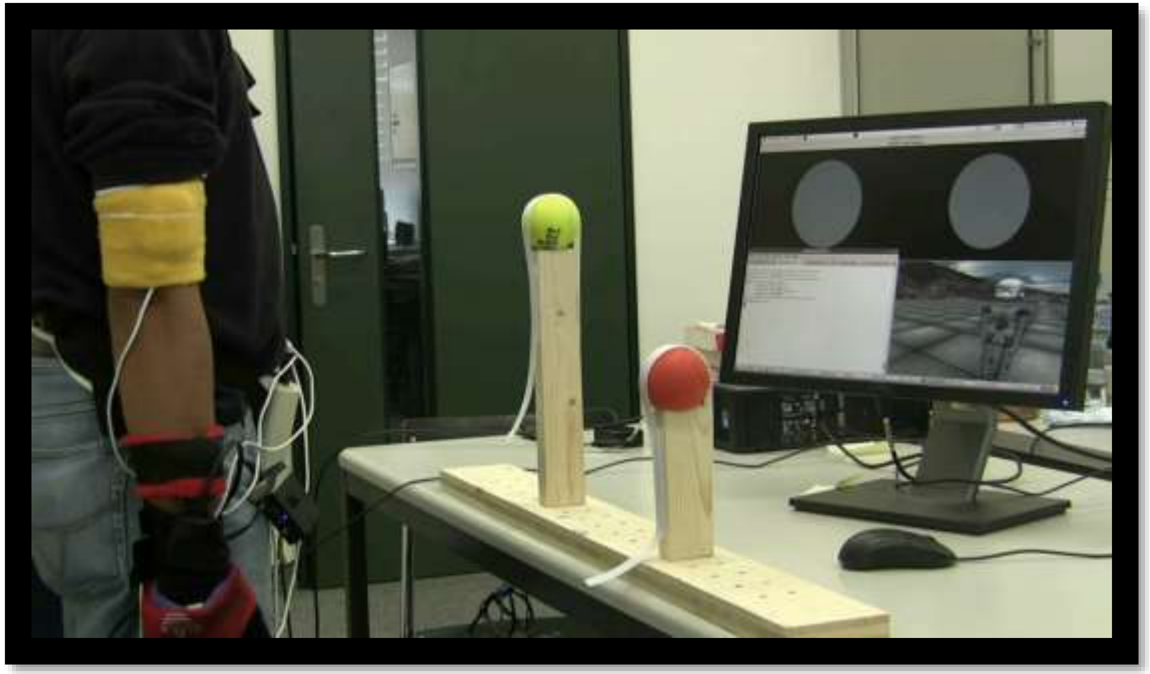
$$\dot{x} = f_x(x)$$



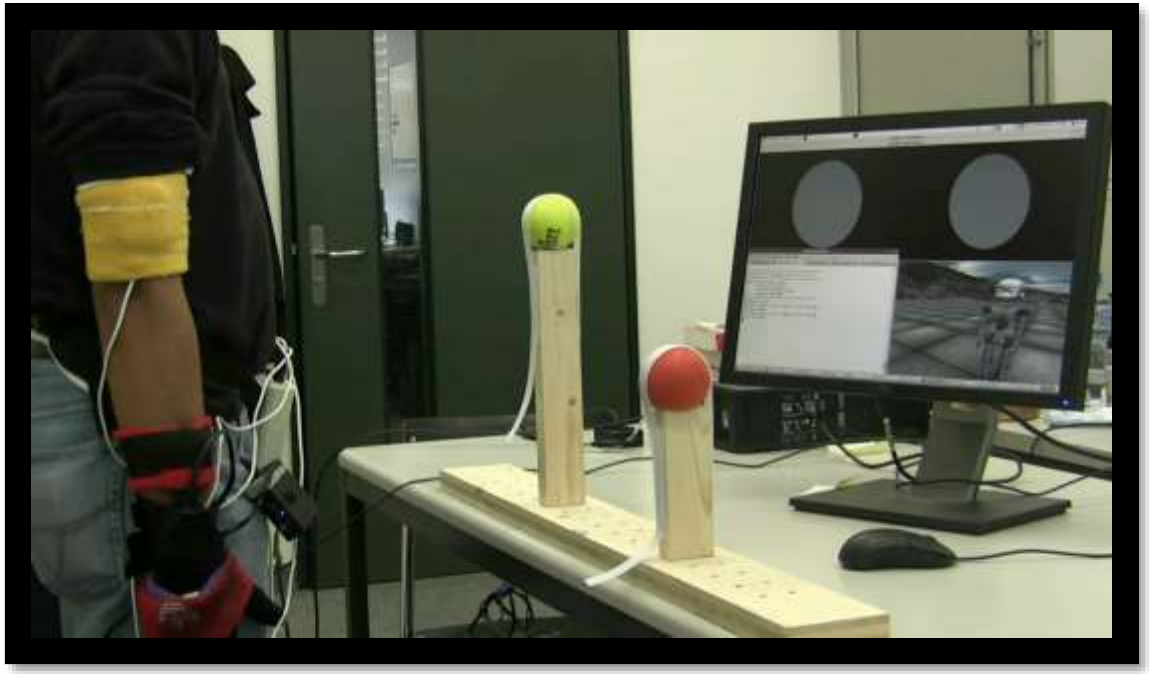
$$\dot{y} = f_y(y, x)$$

Example
Coupled DS for Hand-Arm Coordination

Hand-Arm Coupling under Disturbances



Unperturbed trial

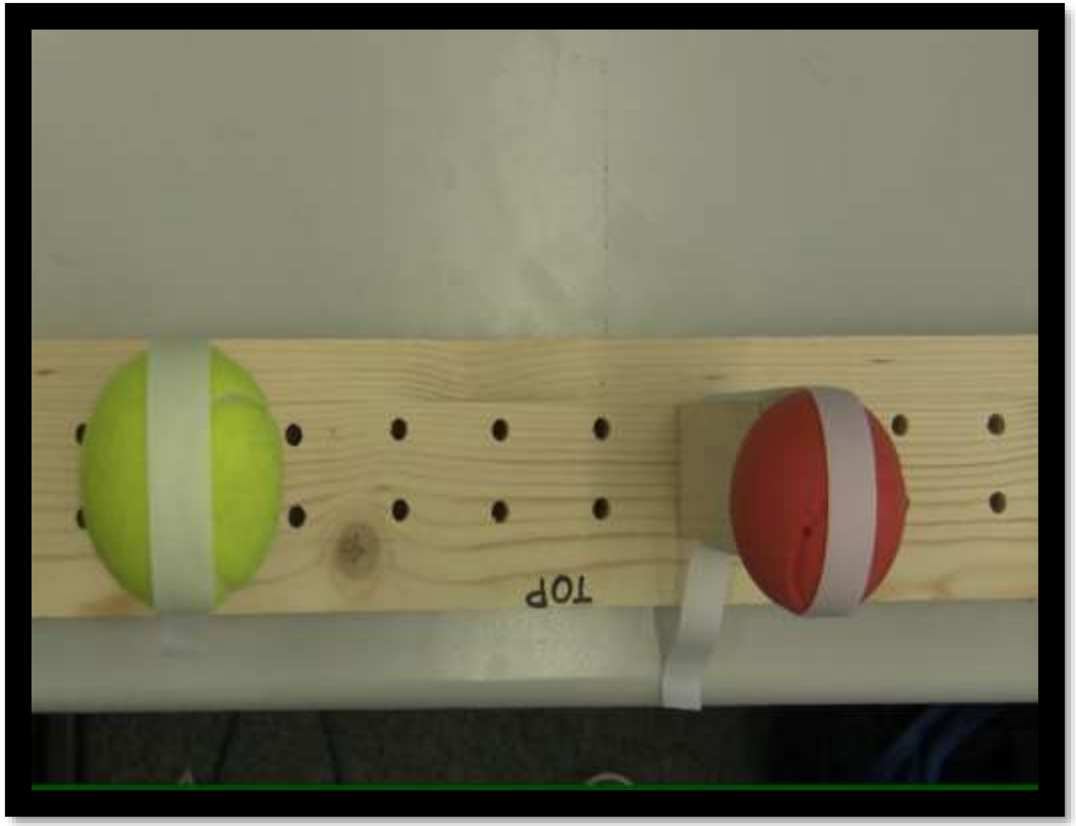


Perturbed trial

Hand-finger coordination: Fingers start opening (preshape) for the final posture at about half of the reaching cycle motion. Is this coupling preserved during perturbation?

A. Shukla and A. Billard, Robotics and Science Systems Conference, RSS 2011.

Hand-Arm Coupling under Disturbances



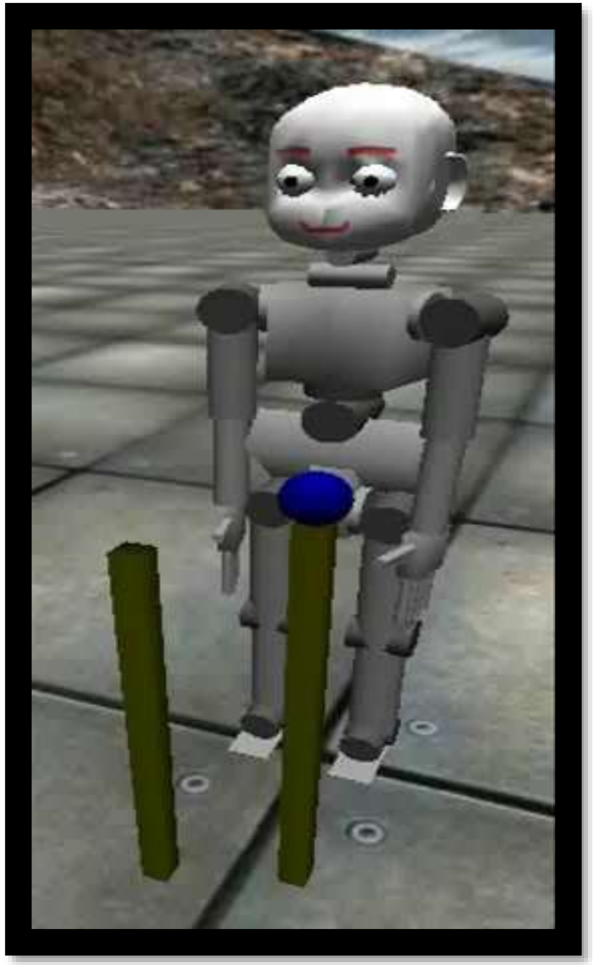
When the target is changed, subjects re-open completely the fingers while redirecting the hand to the new target's location.

→ This may be advantageous to adapt to a new configuration of the object that requires a larger hand aperture with a different fingers' positioning.

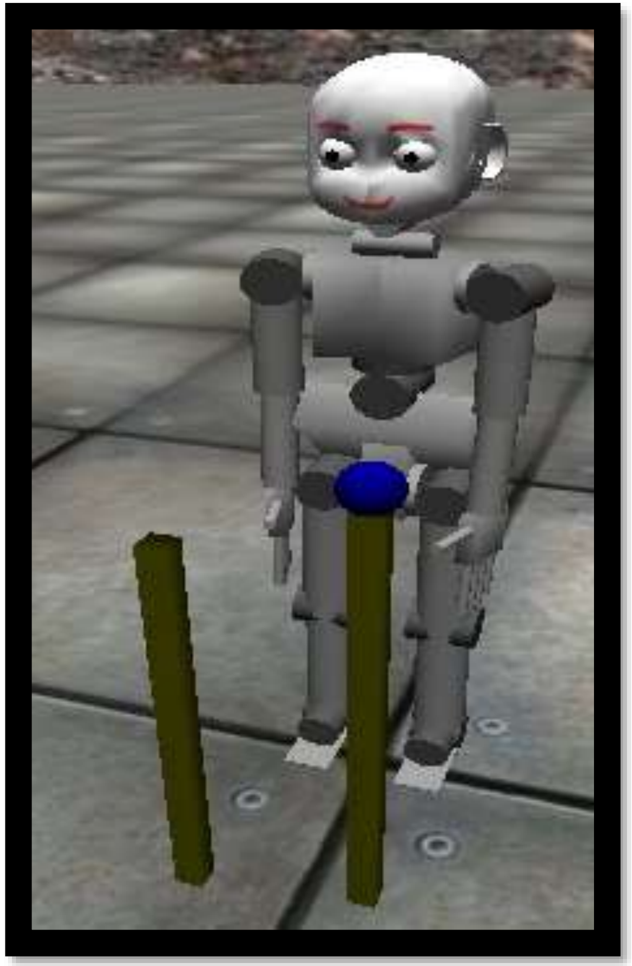
Subjects do not stop to re-plan the new motion. This retargeting strategy is done smoothly across all joints.

→ **Coupling** across fingers and hand dynamics of motion offers immediate satisfaction of constraints during on-the-fly re-computation of hand motion.

Usefulness for Grasping in Robots



Arm/hand and fingers are not coupled

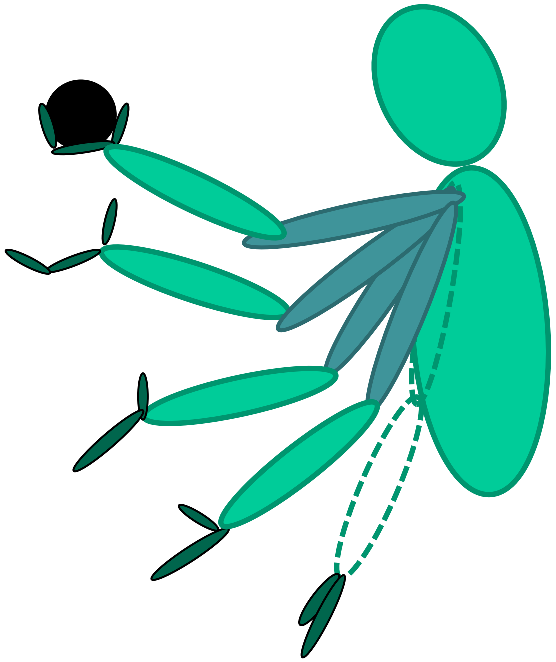


Arm/hand and fingers are explicitly coupled

Modeling Arm-Hand Coupling

$$\dot{x} = f_x(x)$$ Controller for hand transport – attractor on object

$$\dot{y} = f_y(y)$$ Controller for finger motion – attractor in joint space



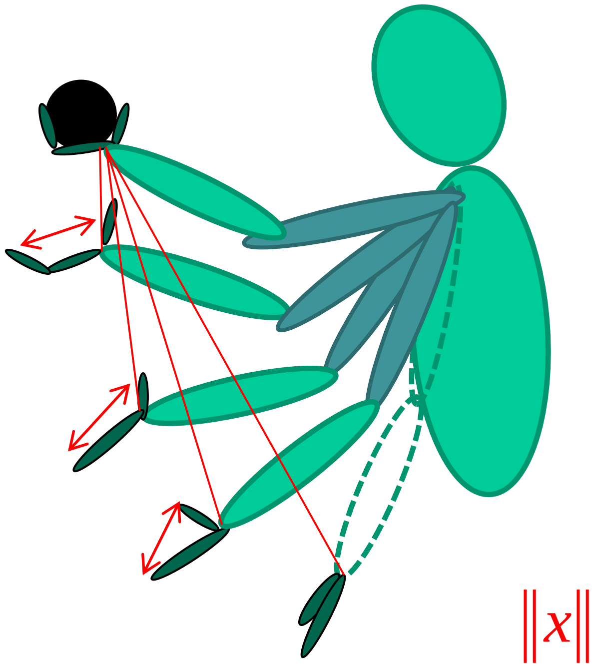
Modeling Arm-Hand Coupling

$$\dot{x} = f_x(x)$$

$$\dot{y} = f_y\left(y, \|x\|\right)$$

Dependency on distance to target

Couple finger-hand dynamics

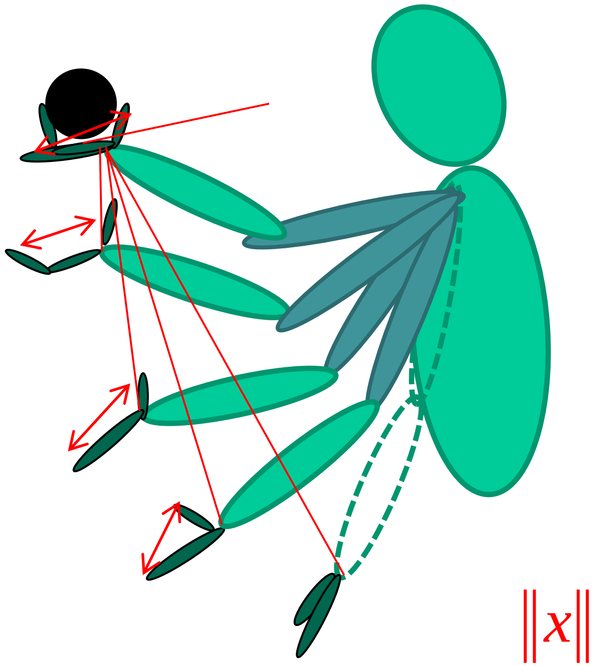


Learning the coupling

Learn $p\left(y,\|x\|\right)$ - dependency on distance to target - from human demonstrations

At run time, compute expected finger aperture:

$$\hat{y} = E\left\{p\left(y \mid \|x\|\right)\right\}$$



Learning the coupling

At run time, compute expected finger aperture:

$$\hat{y} = E\left\{ p\left(y \mid \|x\| \right) \right\}$$

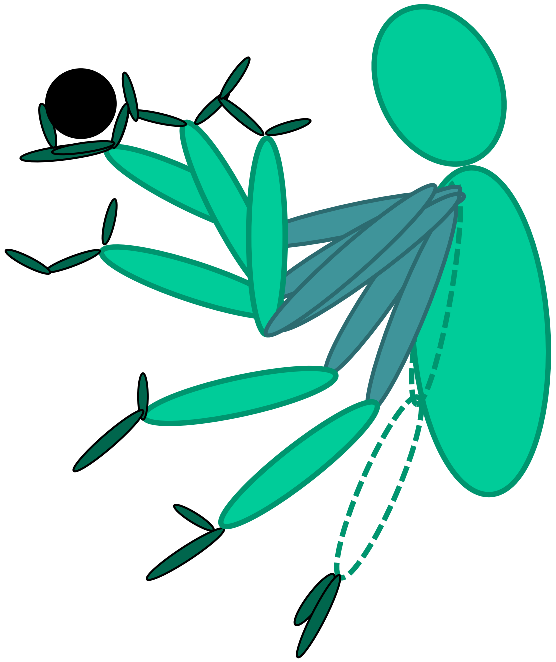
Drive finger motion:

$$\dot{y} = f_y\left(y - \beta \hat{y} \right), \quad y_{t+1} = y_t + \alpha \dot{y}_t$$

Amplitude and speed of finger reopening

Stability:

If $\lim_{t \to \infty} f_x(x) = 0$ and *If*



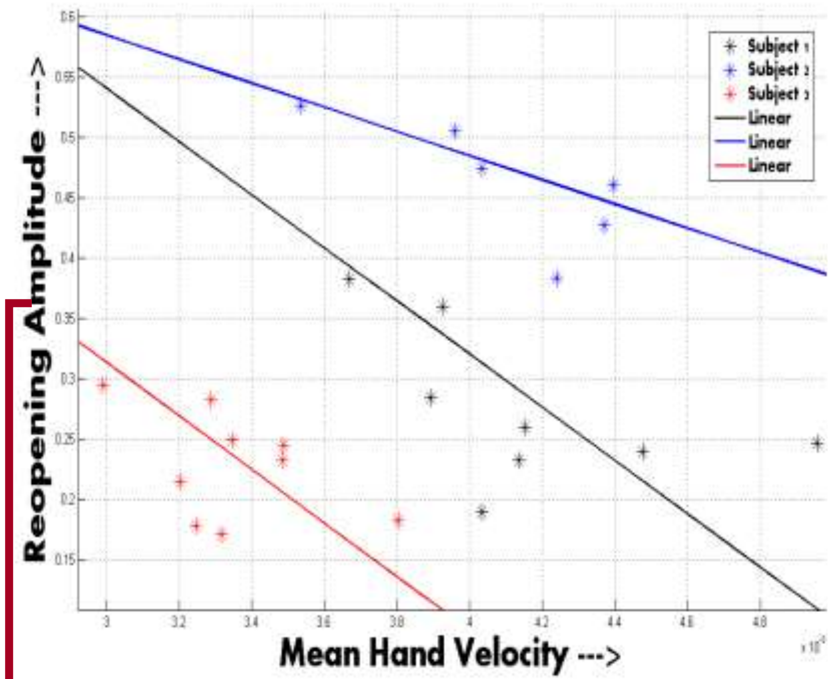
Learning the coupling

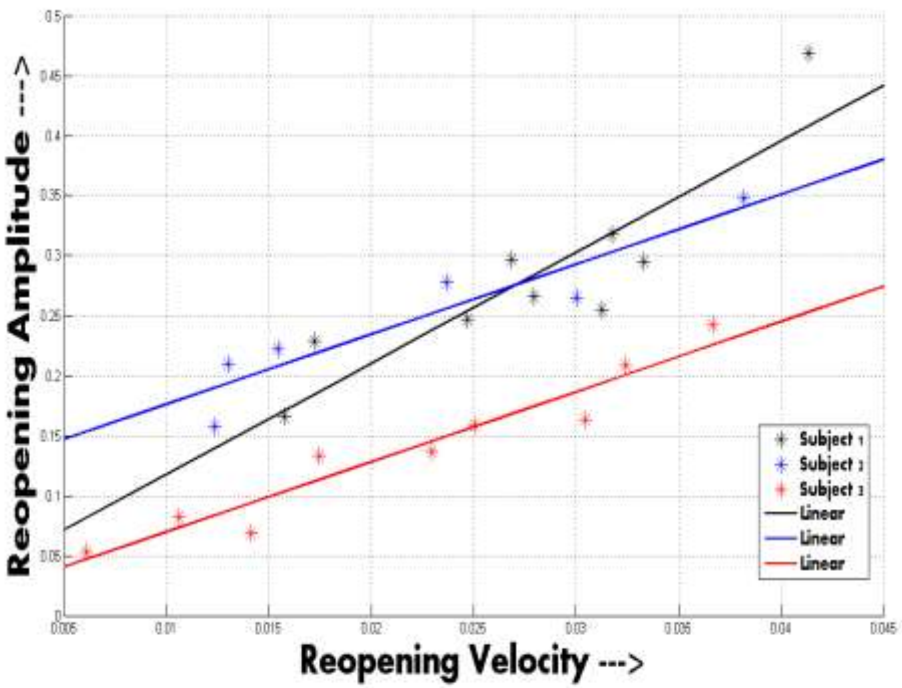
Drive finger motion:

$$\dot{y} = f_y\left(y - \beta \hat{y}\right), \quad y_{t+1} = y_t + \alpha \dot{y}_t$$

Learn α and β from human demonstrations

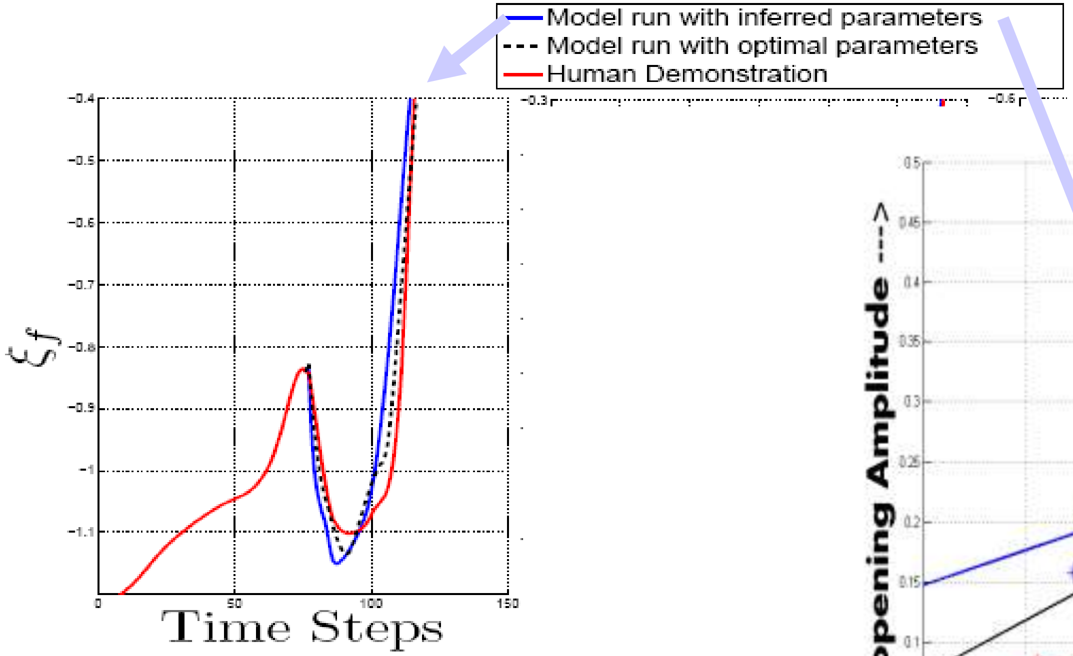
- α Re-opening velocity can be inferred from measuring mean hand velocity prior to perturbation
- β Re-opening amplitude can be inferred from reopening velocity

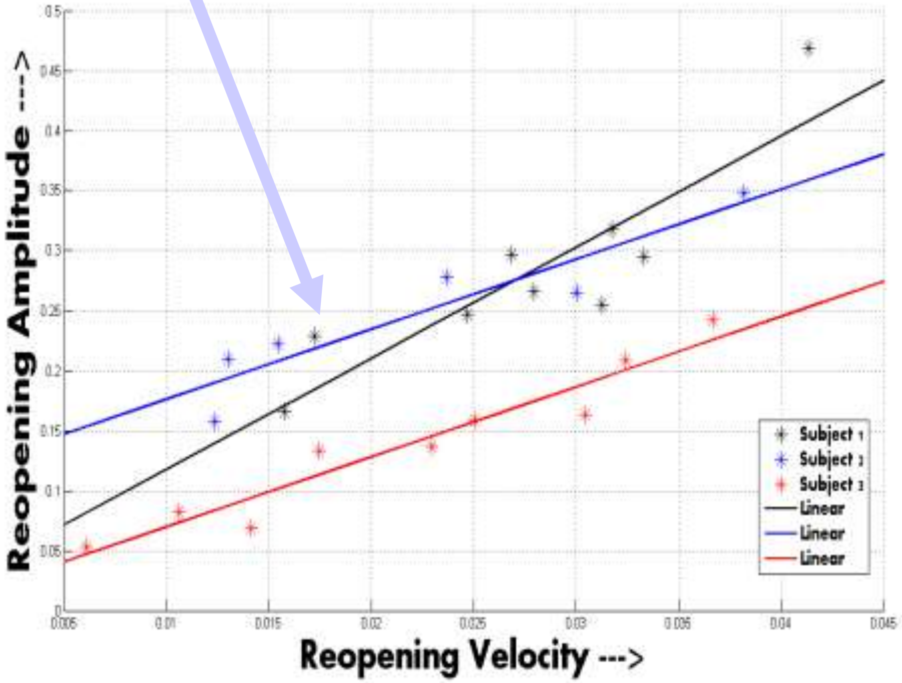




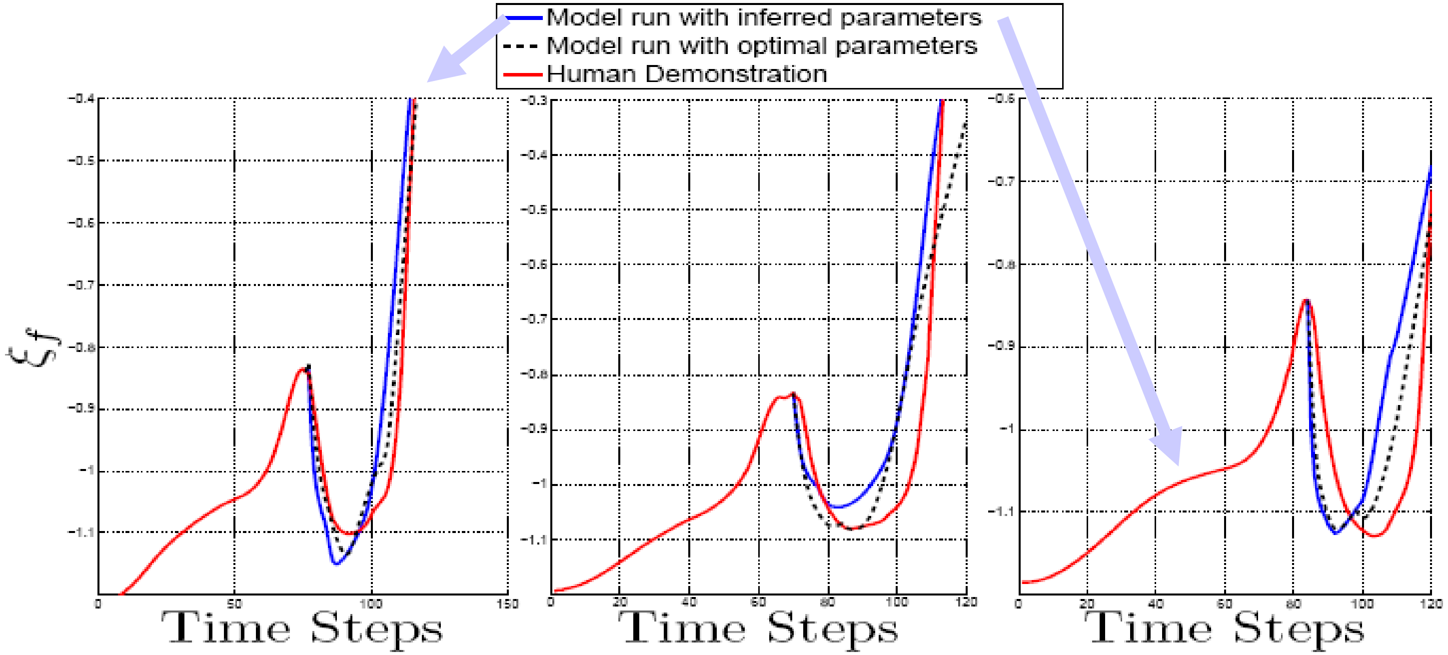
$$\dot{\xi}_t = E\left\{p\left(\dot{\xi}_t \mid \xi_t + \beta\hat{\xi}_t\right)\right\}, \qquad \xi_{t+1} = \xi_t + \alpha\dot{\xi}_t\Delta t$$

α parameter inferred from human data



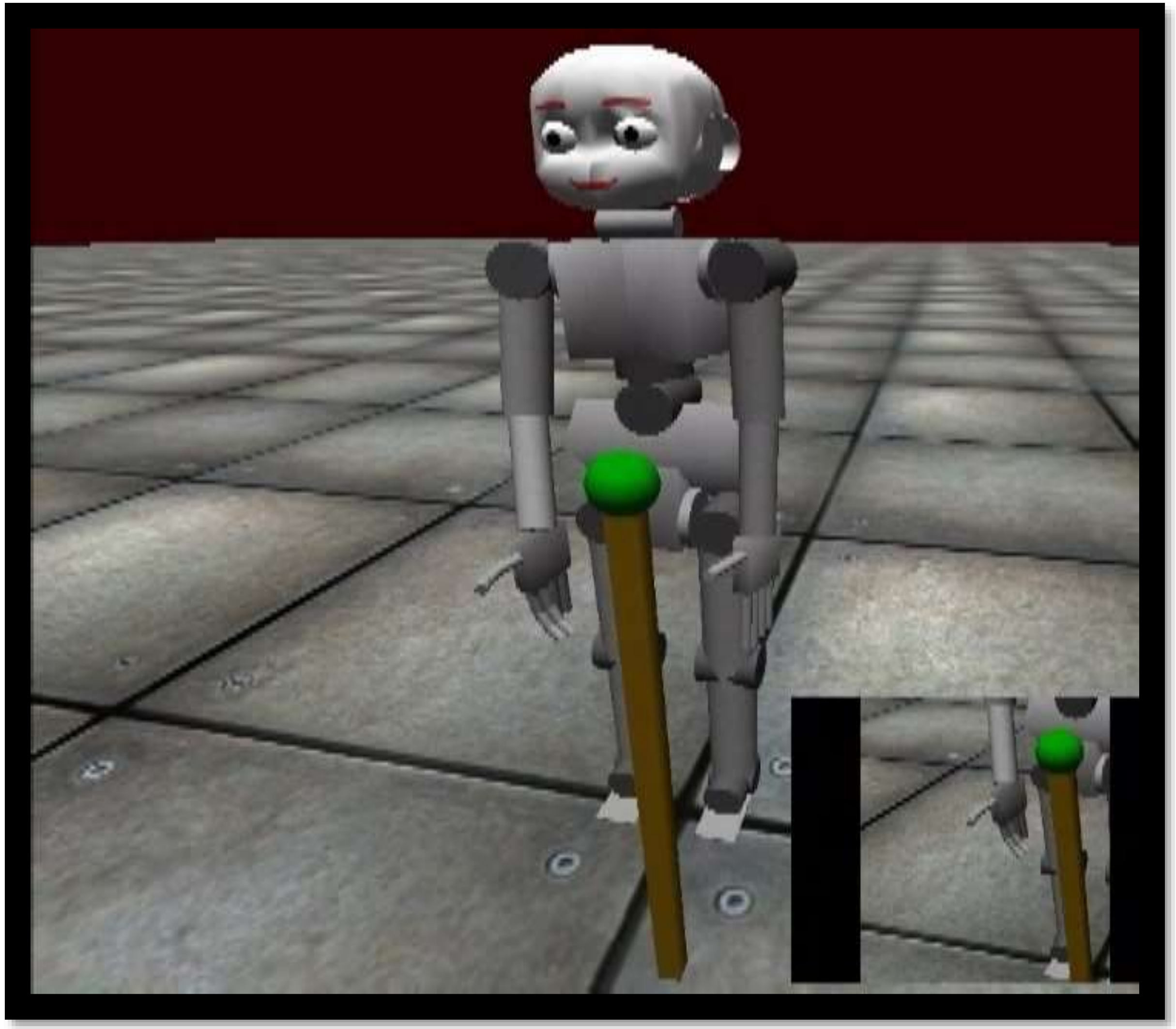


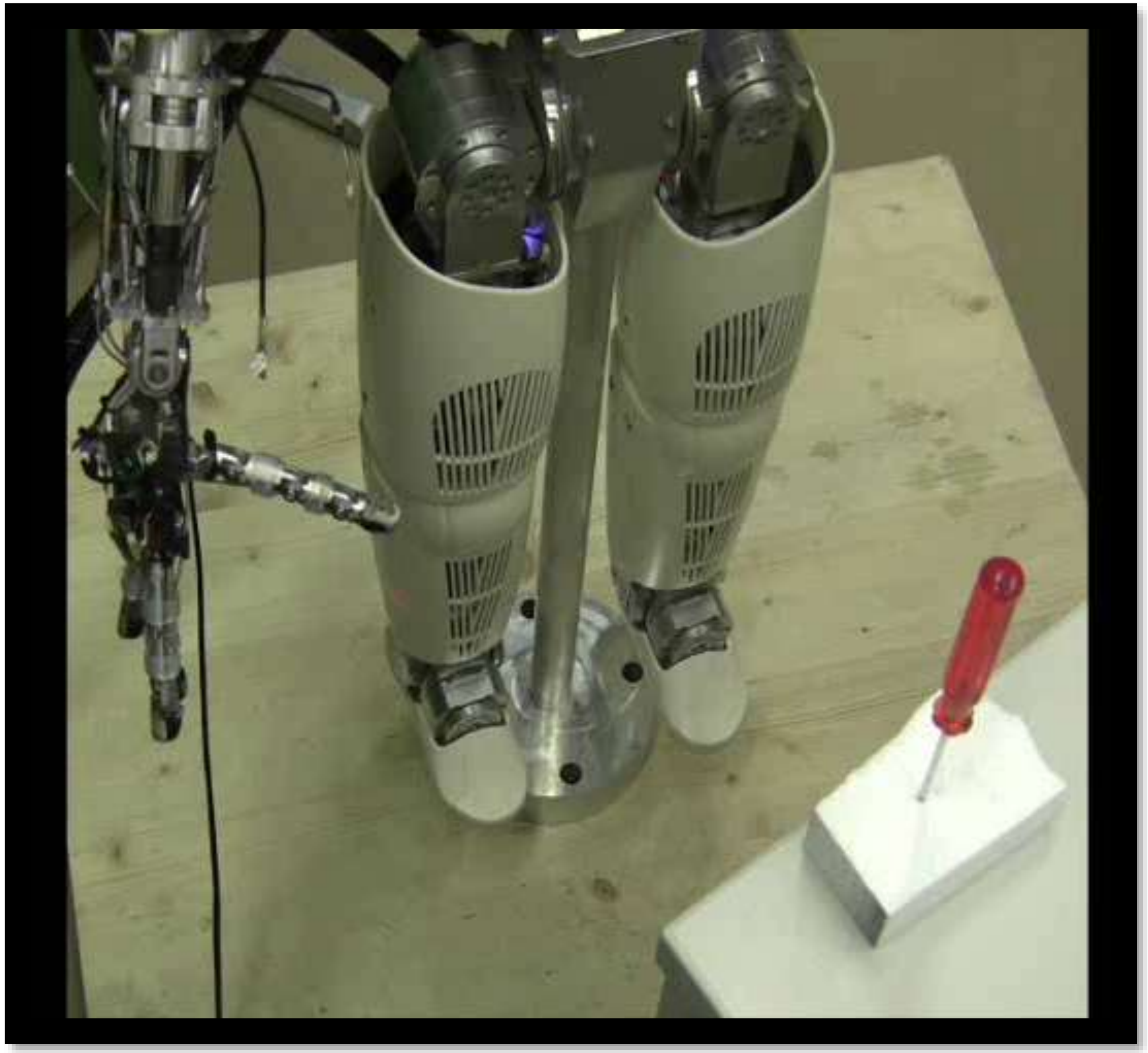
α parameter inferred from human data



3 trials for the same subject

The CDS model gives both a good qualitative and quantitative assessment of human motion.





Adaptation from pinch to power grasp
(train two separate Coupled DS for each grasp type)

Adaptation under visual or tactile disturbances

Example
Coupled DS for Bimanual Coordination

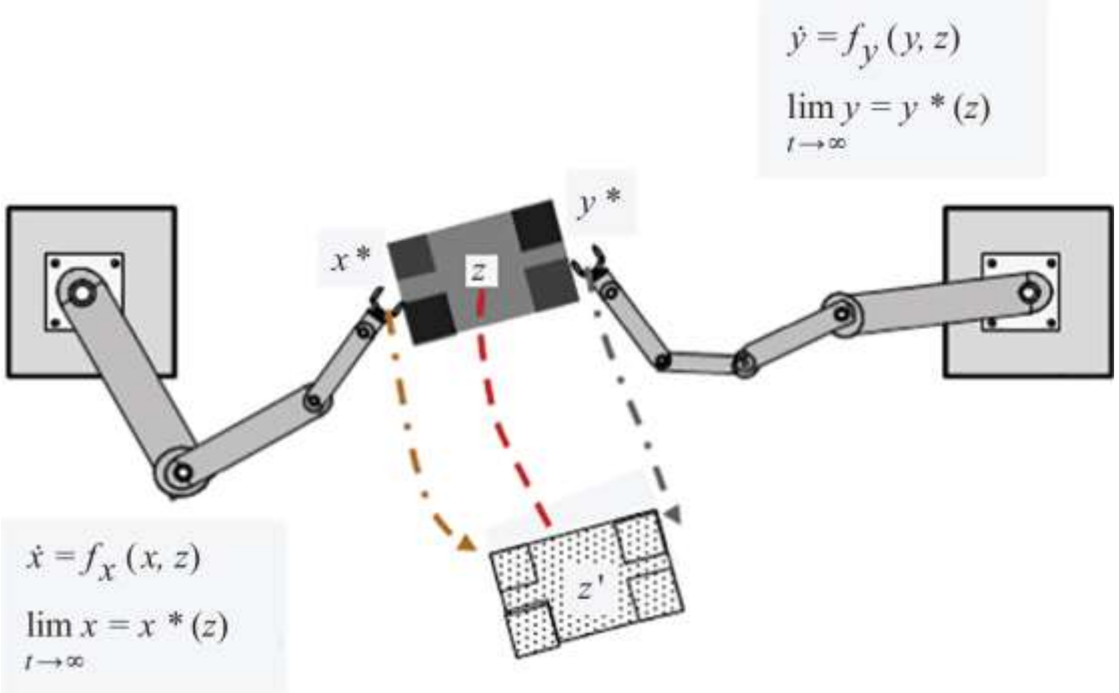
Coupling through External Variable

DS-s can be coupled through an external variable.

Let z be a virtual variable. x and y can be coupled through z:

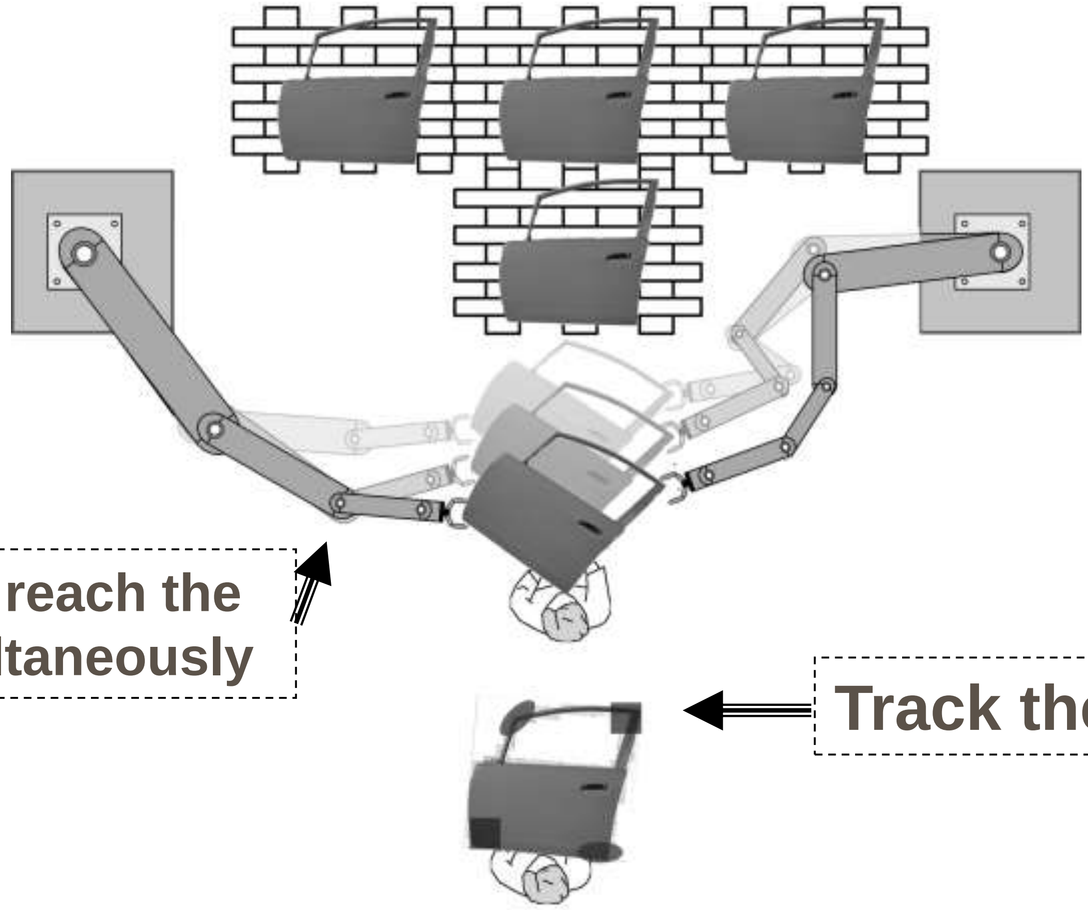
$$\dot{x} = f_x(x, z)$$

$$\dot{y} = f_y(y, z)$$



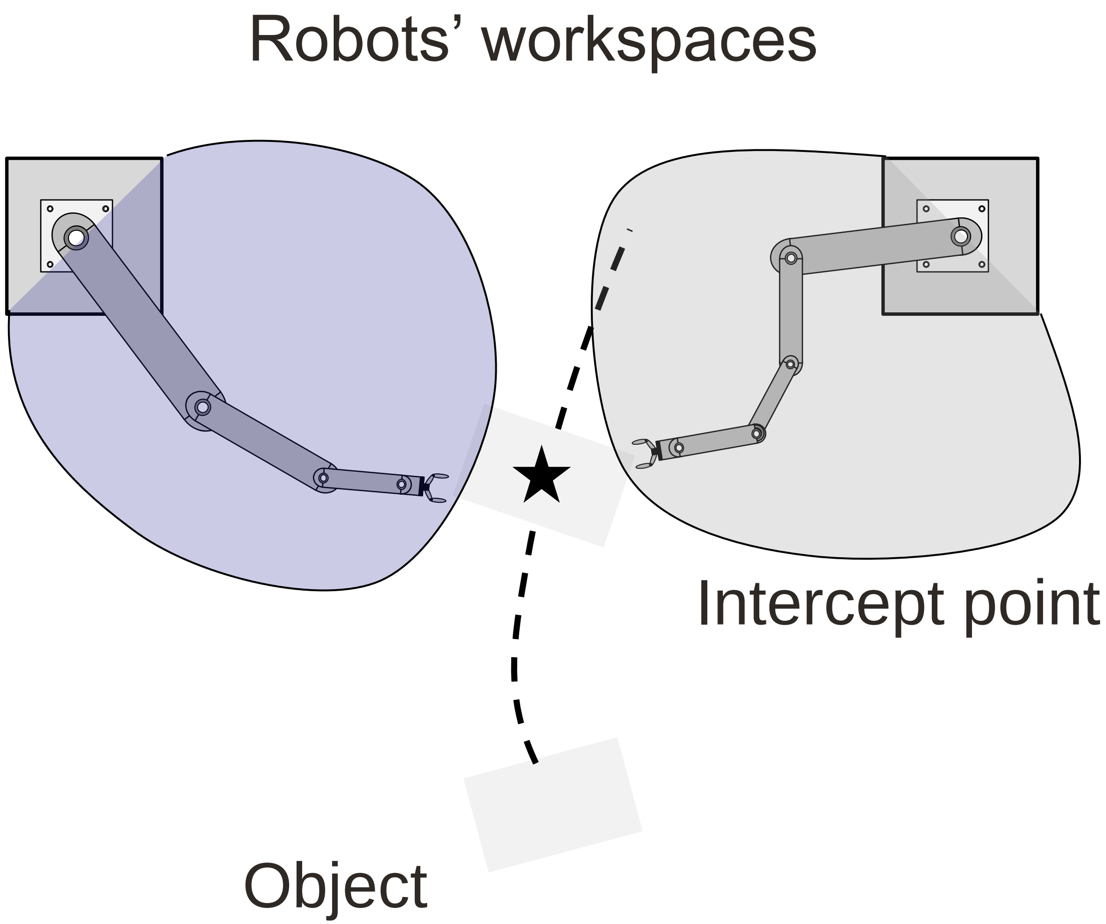

YOUCAR

EPFL



Robot arms reach the object simultaneously

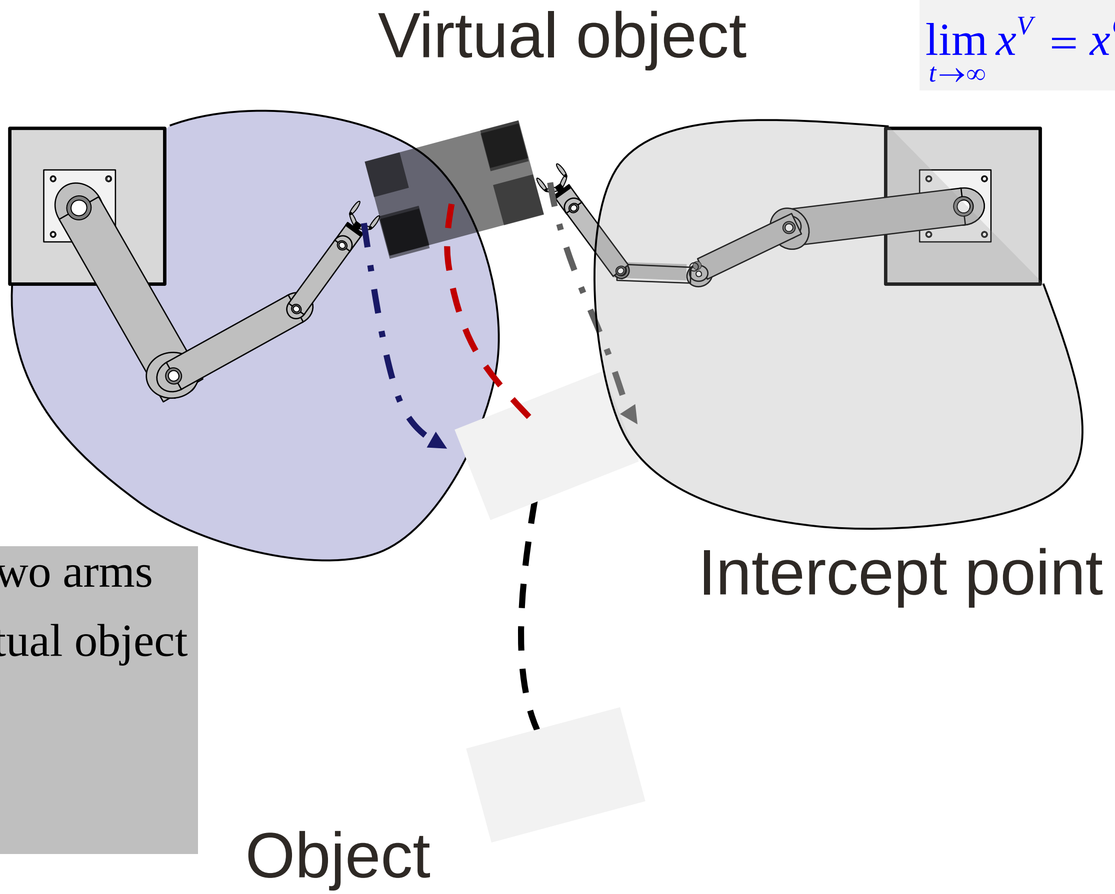
Track the object


Robots' workspaces
Intercept point
Object

Control law for virtual object coordinated with real object

$$\dot{x}^V = f_V\left(x^V, x^O\right)$$

$$\lim_{t\to\infty} x^V = x^O$$



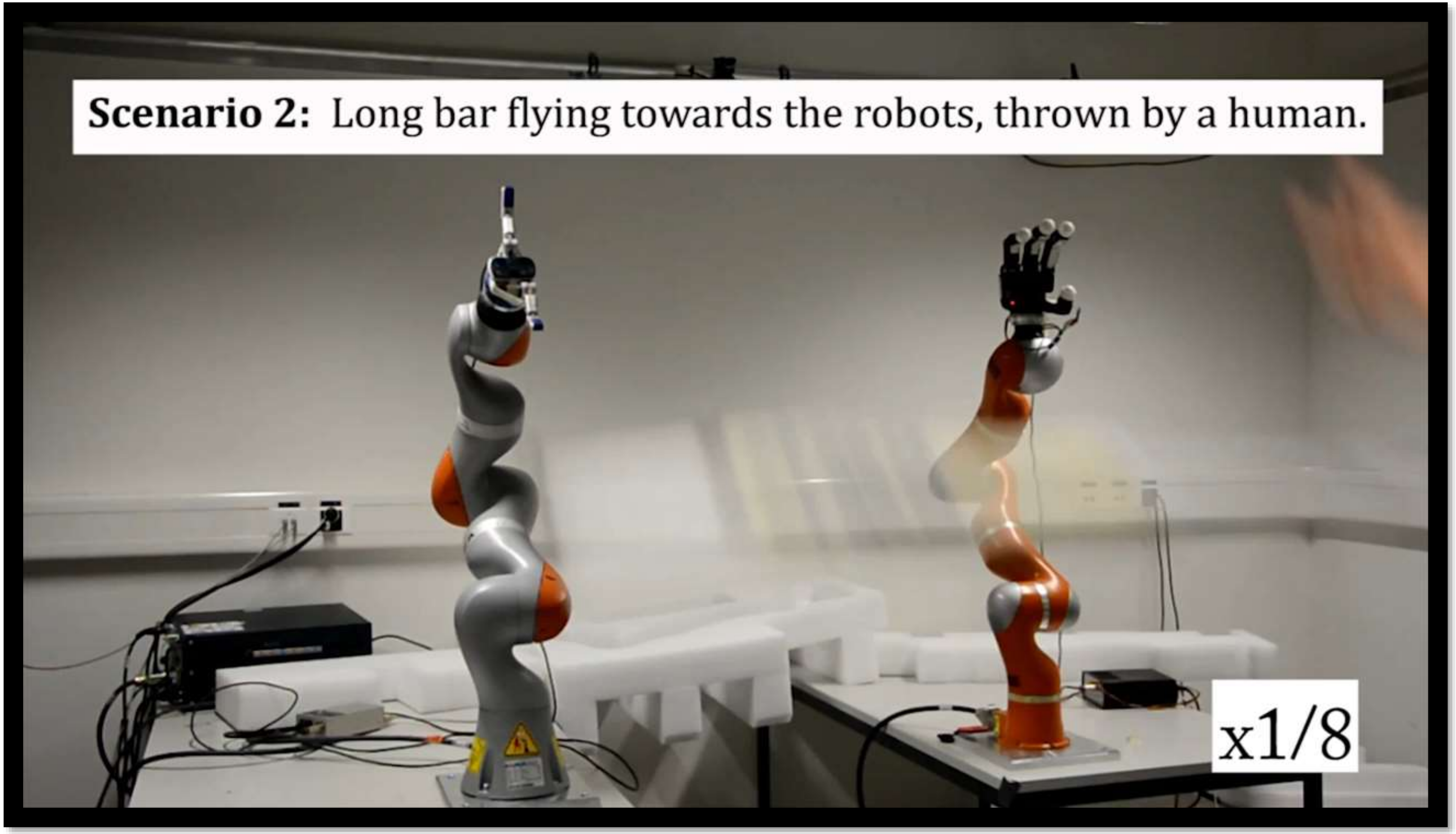
Control law for the two arms coordinated with virtual object

$$\dot{x}^R = f_R\left(x^R, x^V\right)$$

$$\lim_{t\to\infty} x^R = x^V$$

EPFL

KUKA AWARD FINALIST 2017



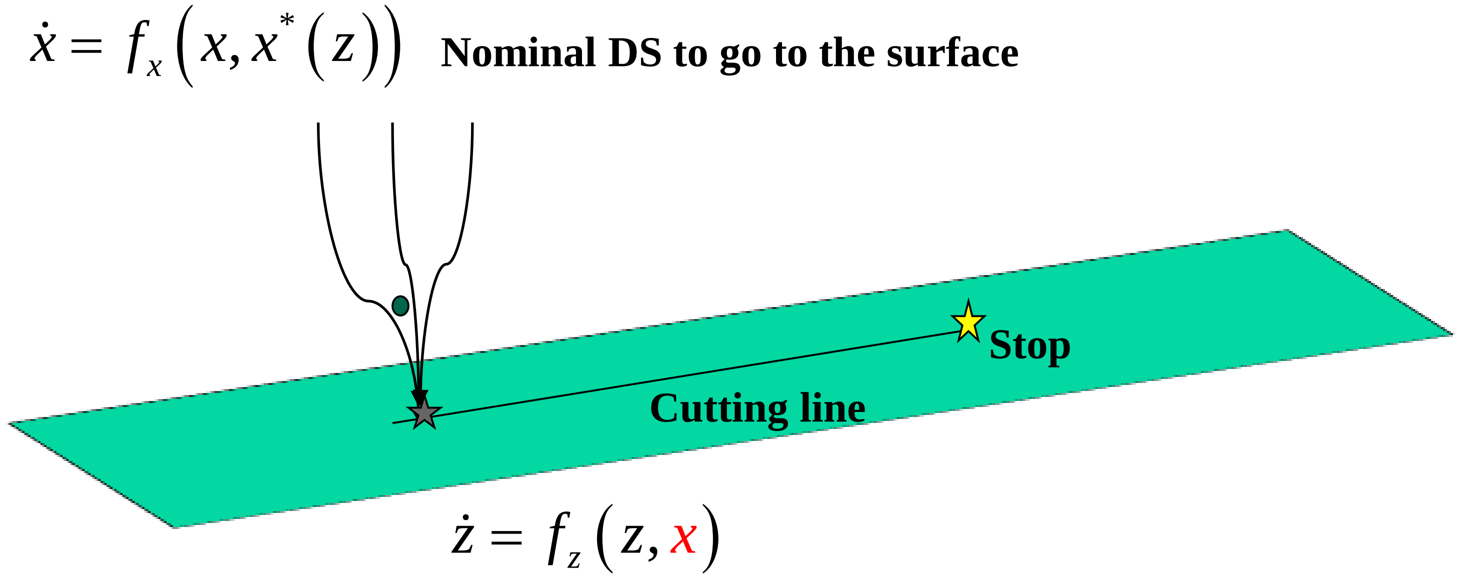
S.S. Mirrazavi Saliehian, N. Figueroa and A. Billard. RSS 2016,. Best Student Paper Award

Example
Coupled DS for Cutting Tissue

Coupling with Virtual Dynamics

Double Coupling across two DS-s.

$$\dot{x} = f_x(x, z)$$
$$\dot{z} = f_z(z, x)$$

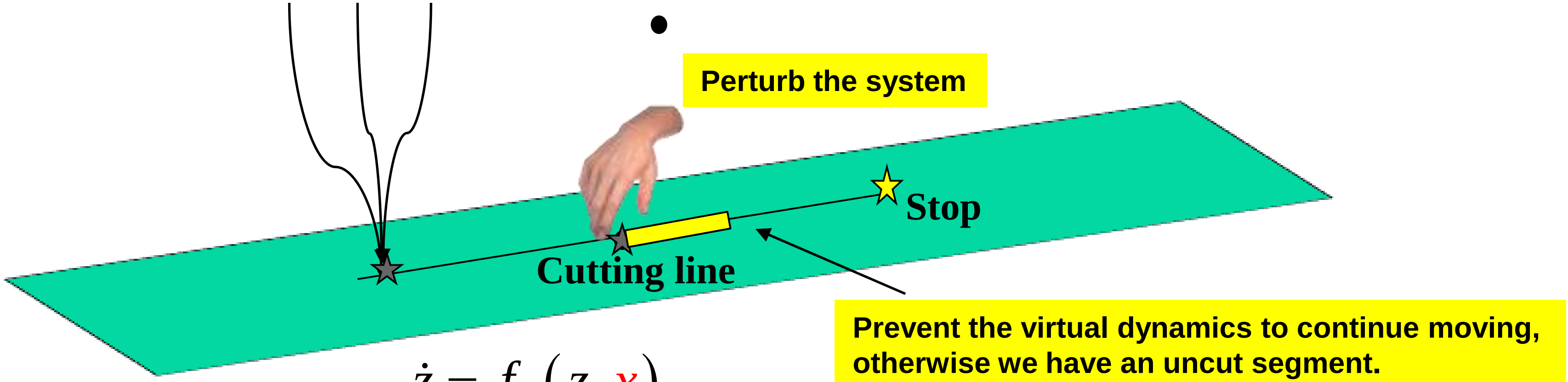


Coupling with Virtual Dynamics

Double Coupling across two DS-s.

$$\dot{x} = f_x(x, z)$$
$$\dot{z} = f_z(z, x)$$

$$\dot{x} = f_x\left(x, x^*(z)\right) = A(x - z):$$ Attractor is on the location of the virtual system



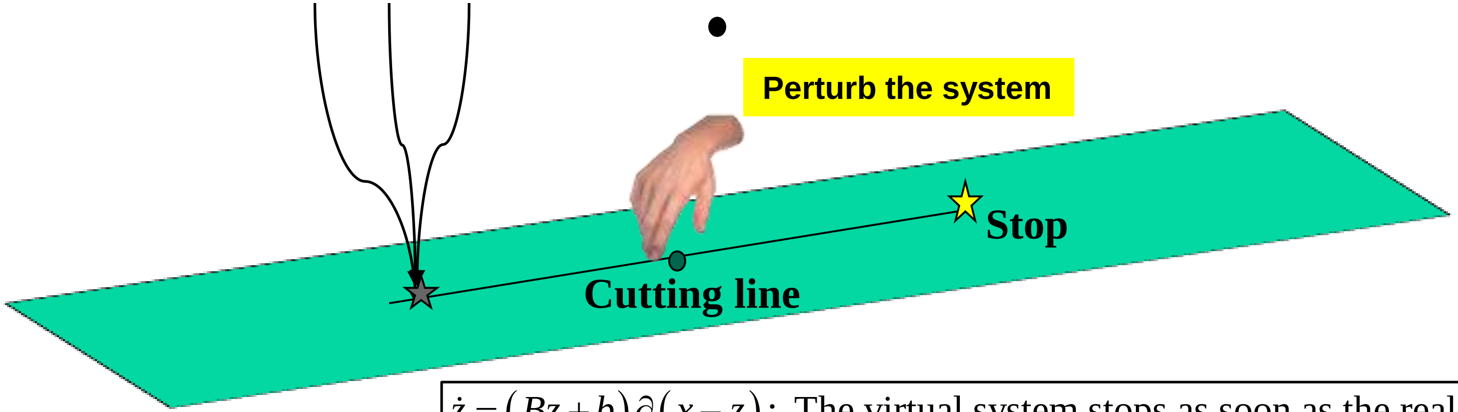
$$\dot{z} = f_z(z, x)$$

Coupling with Virtual Dynamics

Double Coupling across two DS-s.

$$\dot{x} = f_x(x, z)$$
$$\dot{z} = f_z(z, x)$$

$$\dot{x} = f_x\left(x, x^*(z)\right) = A(x - z):$$ Attractor is on the location of the virtual system

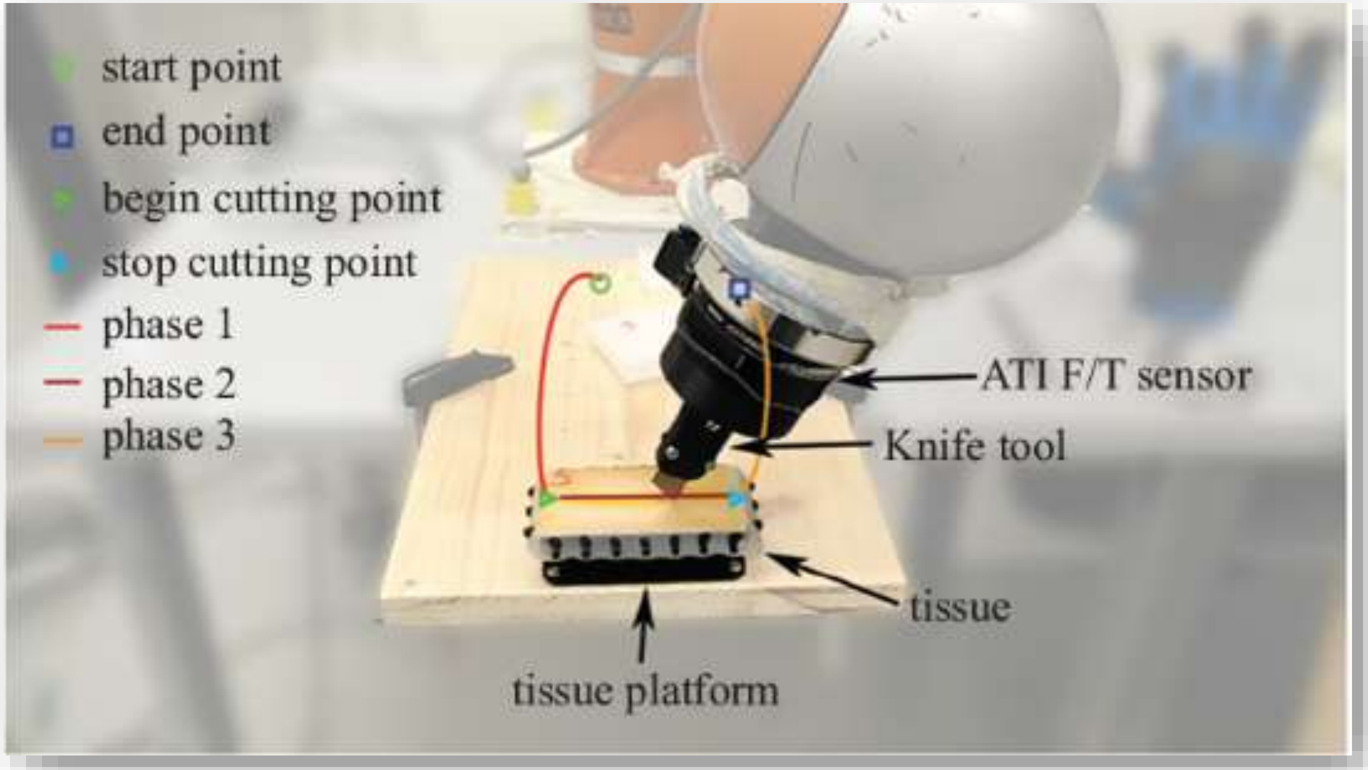


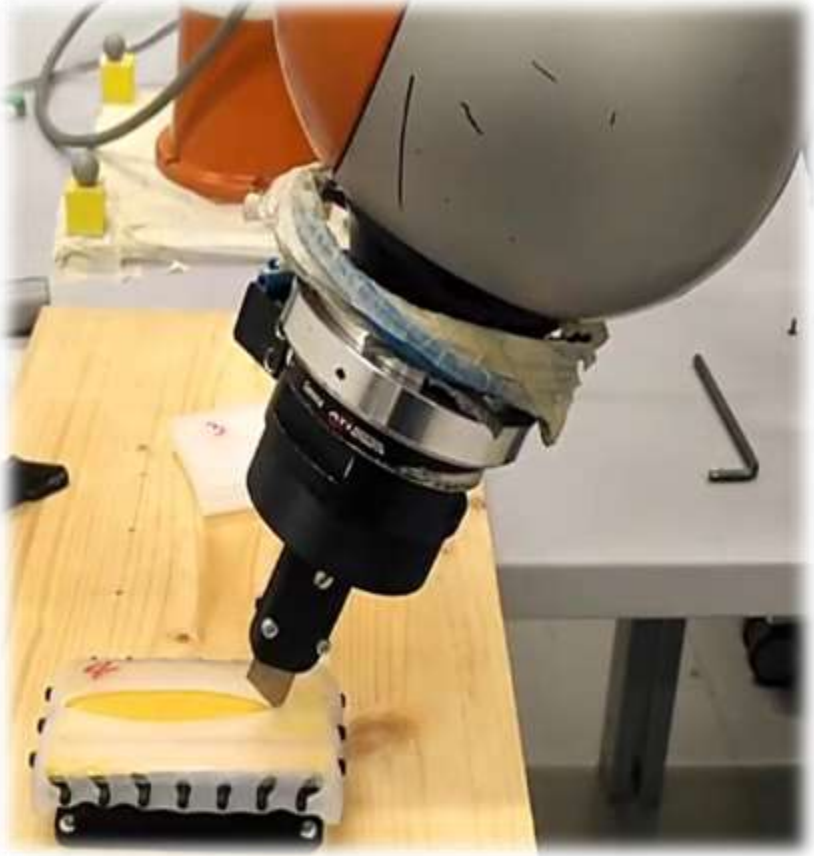
$\dot{z} = (Bz + b)\partial(x - z)$: The virtual system stops as soon as the real systems is not colocated

$\rightarrow$ It waits until the system comes back to cutting point.

Application for Cutting Soft Tissue

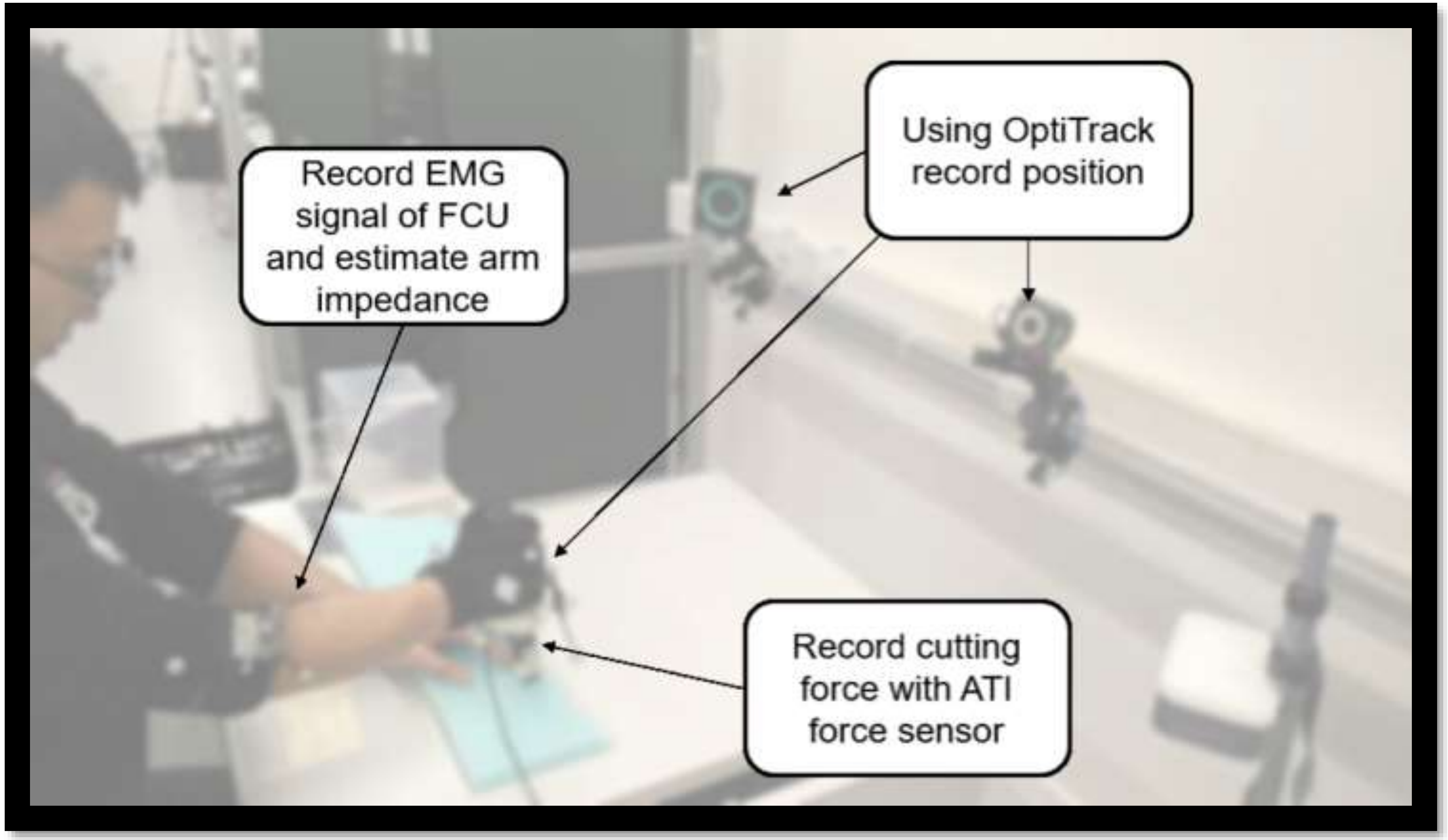
Goal: cut a piece of silicon in a straight line





Wu, R. and Billard, A., 2021. Learning From Demonstration and Interactive Control of Variable-Impedance to Cut Soft Tissues. *IEEE/ASME Transactions on Mechatronics*.

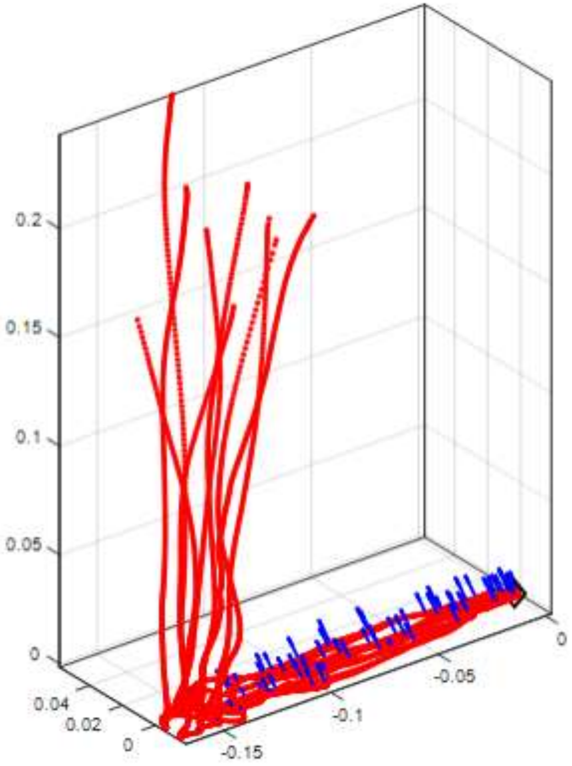
Human Demonstration



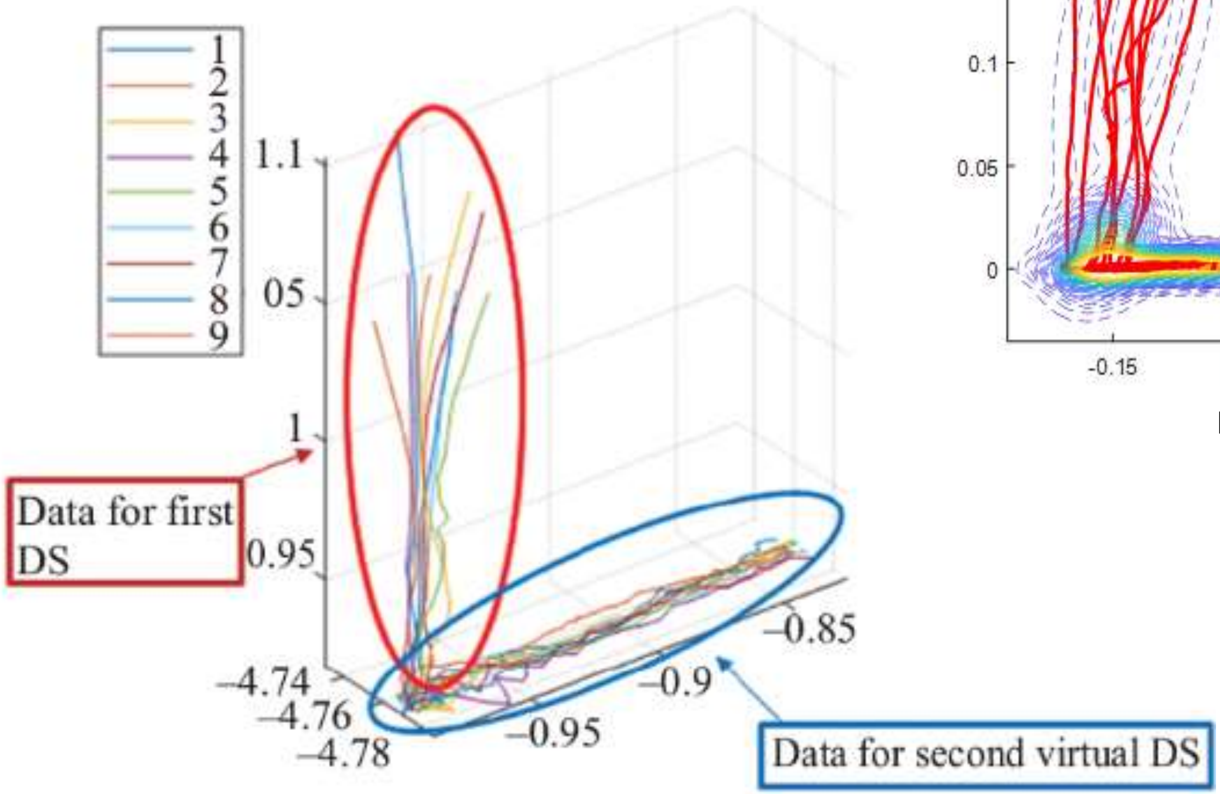
Recorded data:

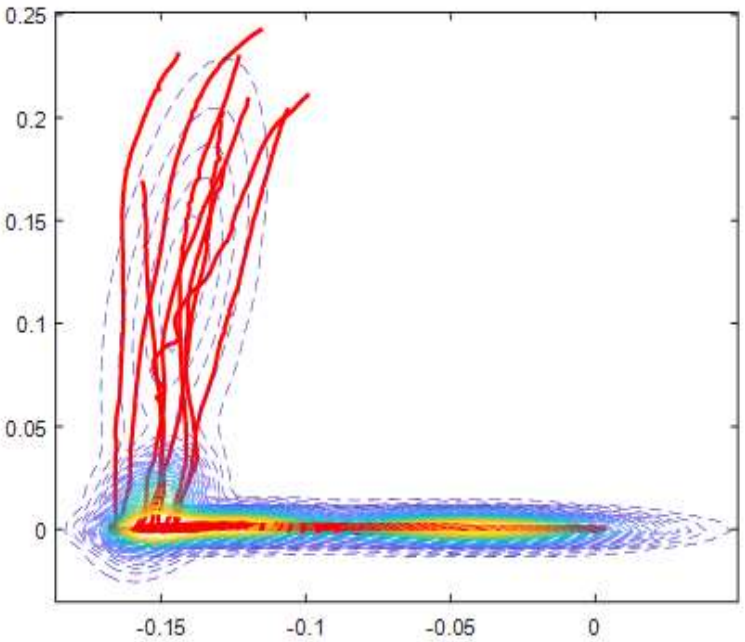
- Knife position
- Interaction force
- Human arm impedance

From Raw Data to DS



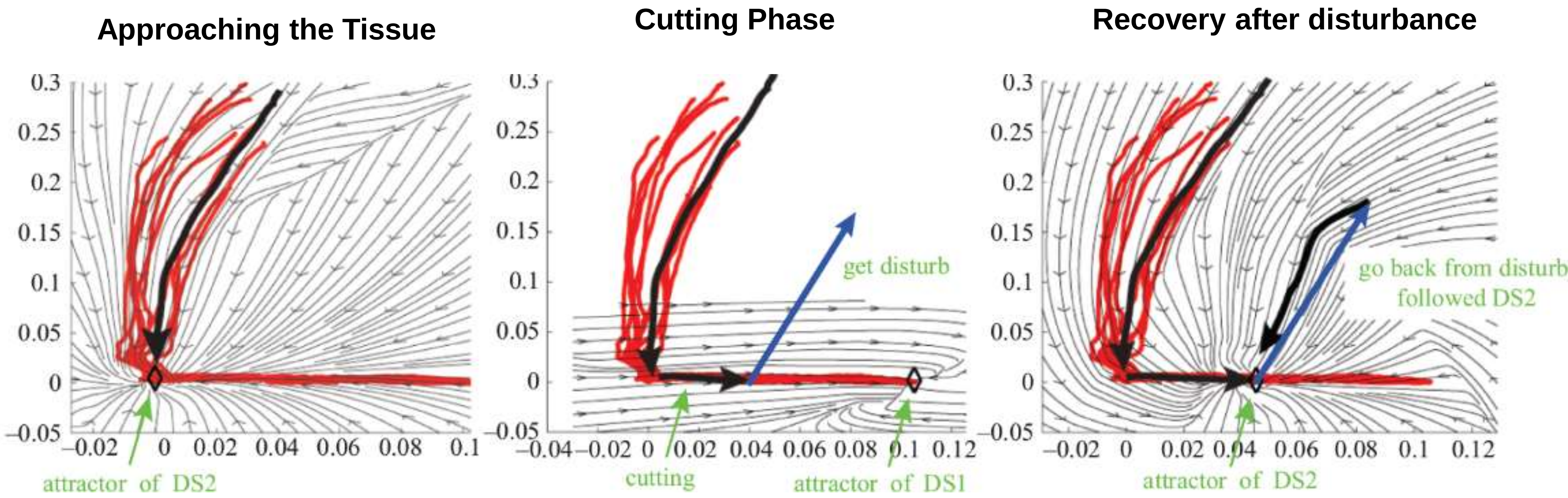
Raw Data



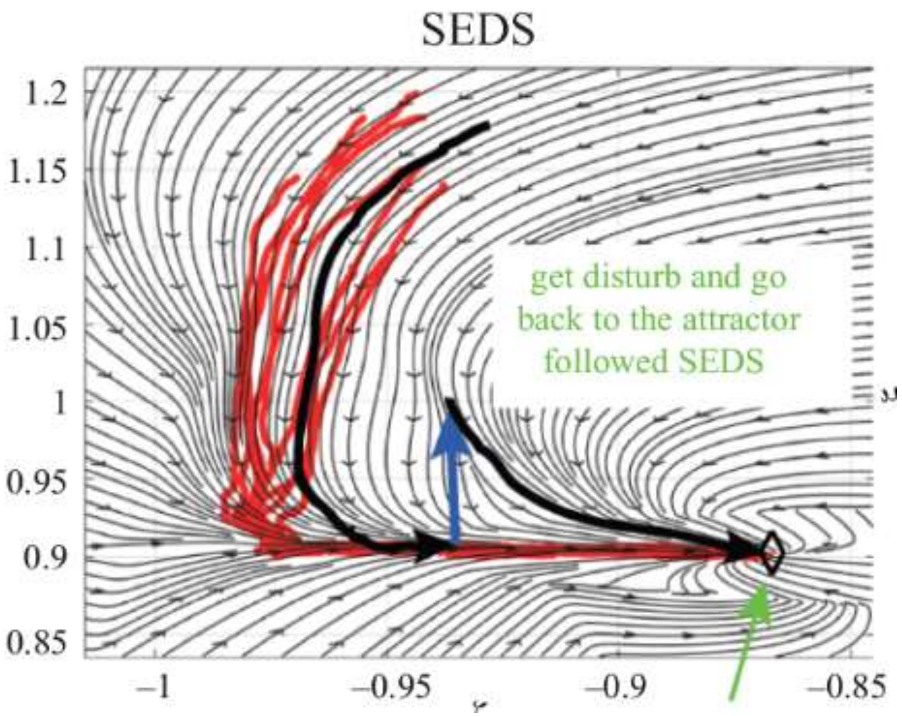


Learned Gauss Functions

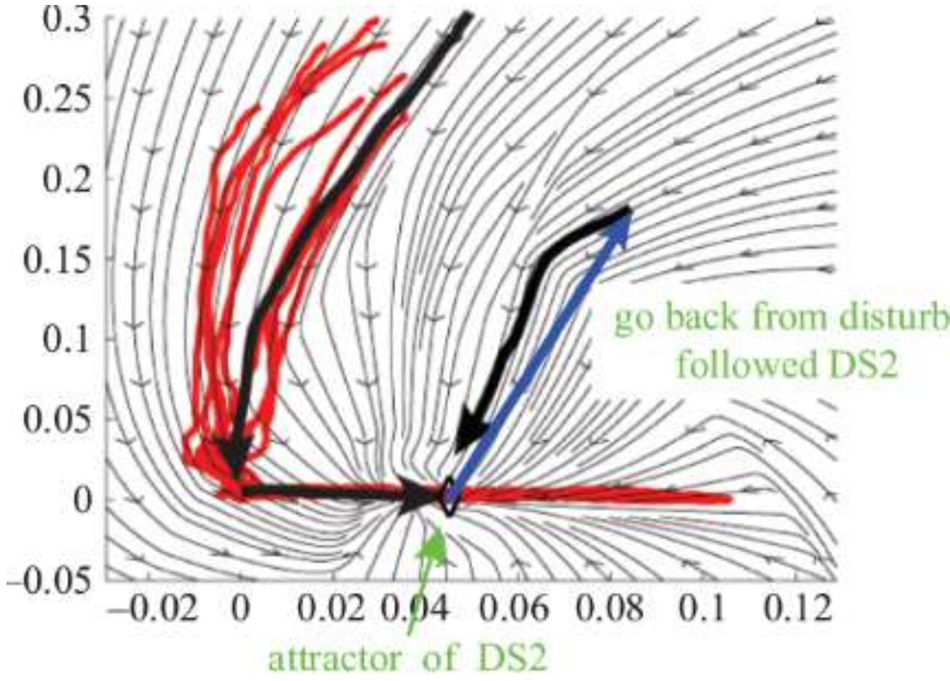
Modeling with Coupled DS



Comparison Coupled DS with SEDS

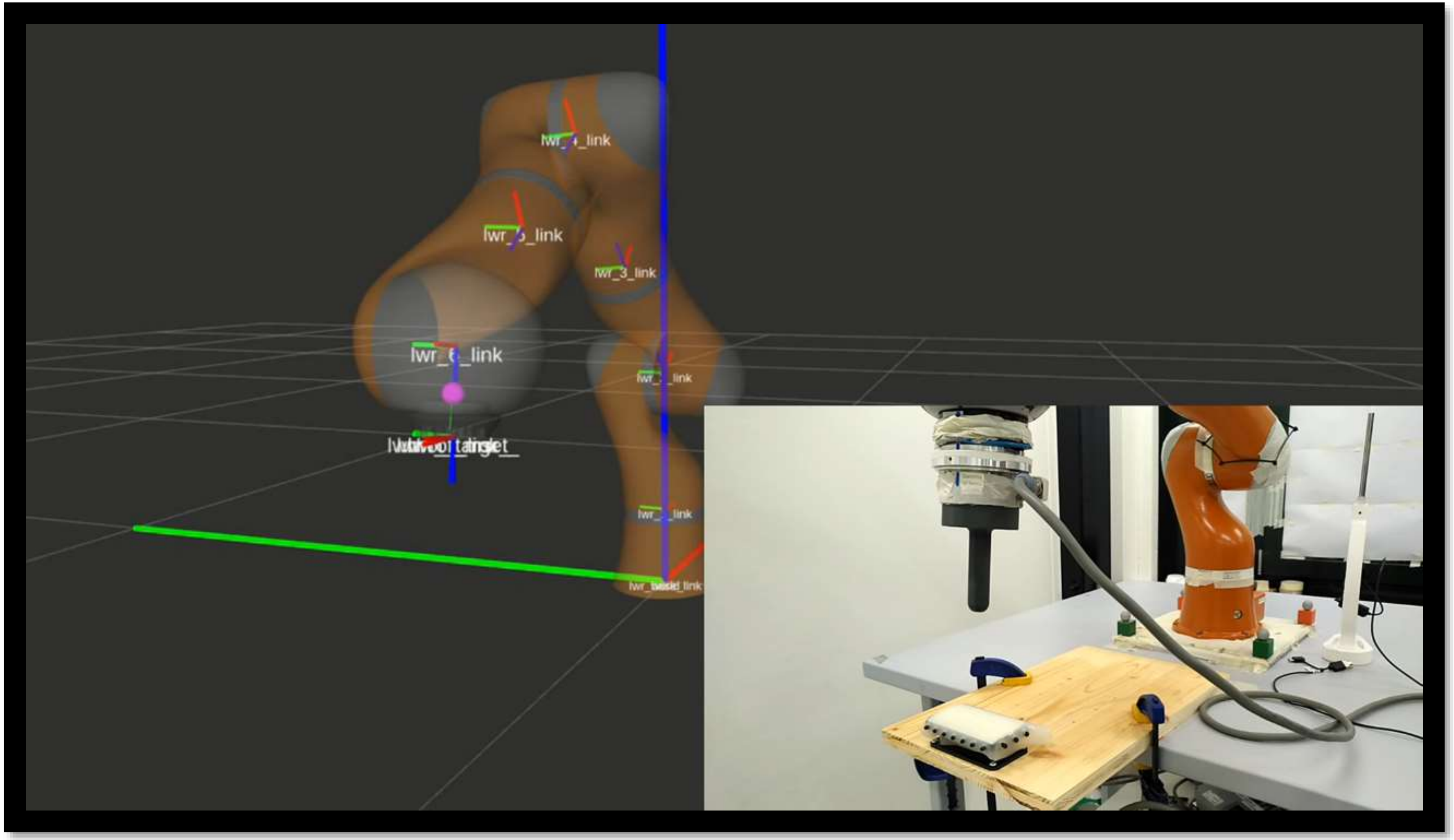


Recovery after disturbance

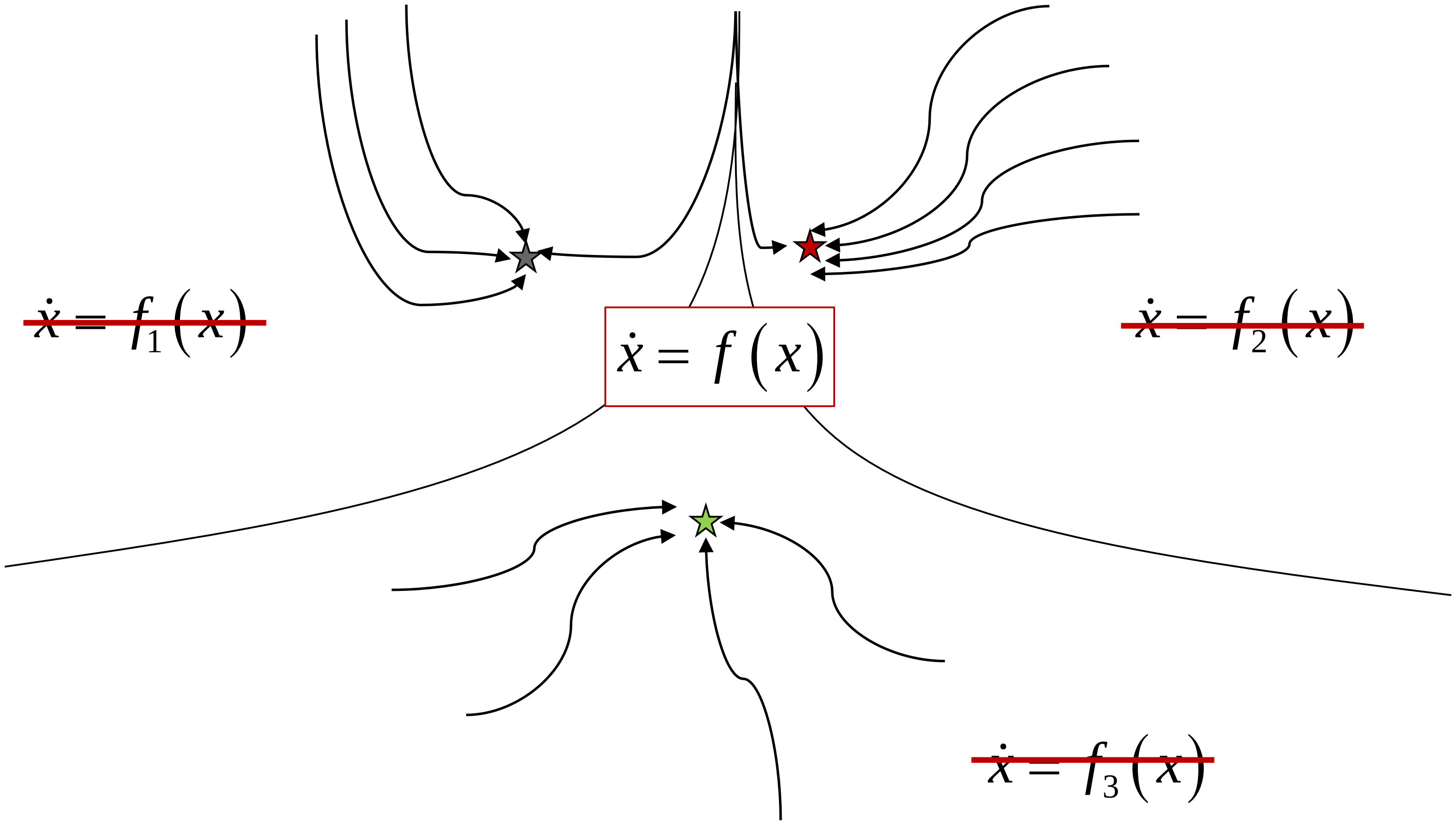


SEDS would simply send the knife to the end-effector but it would not come back to the cutting trajectory.

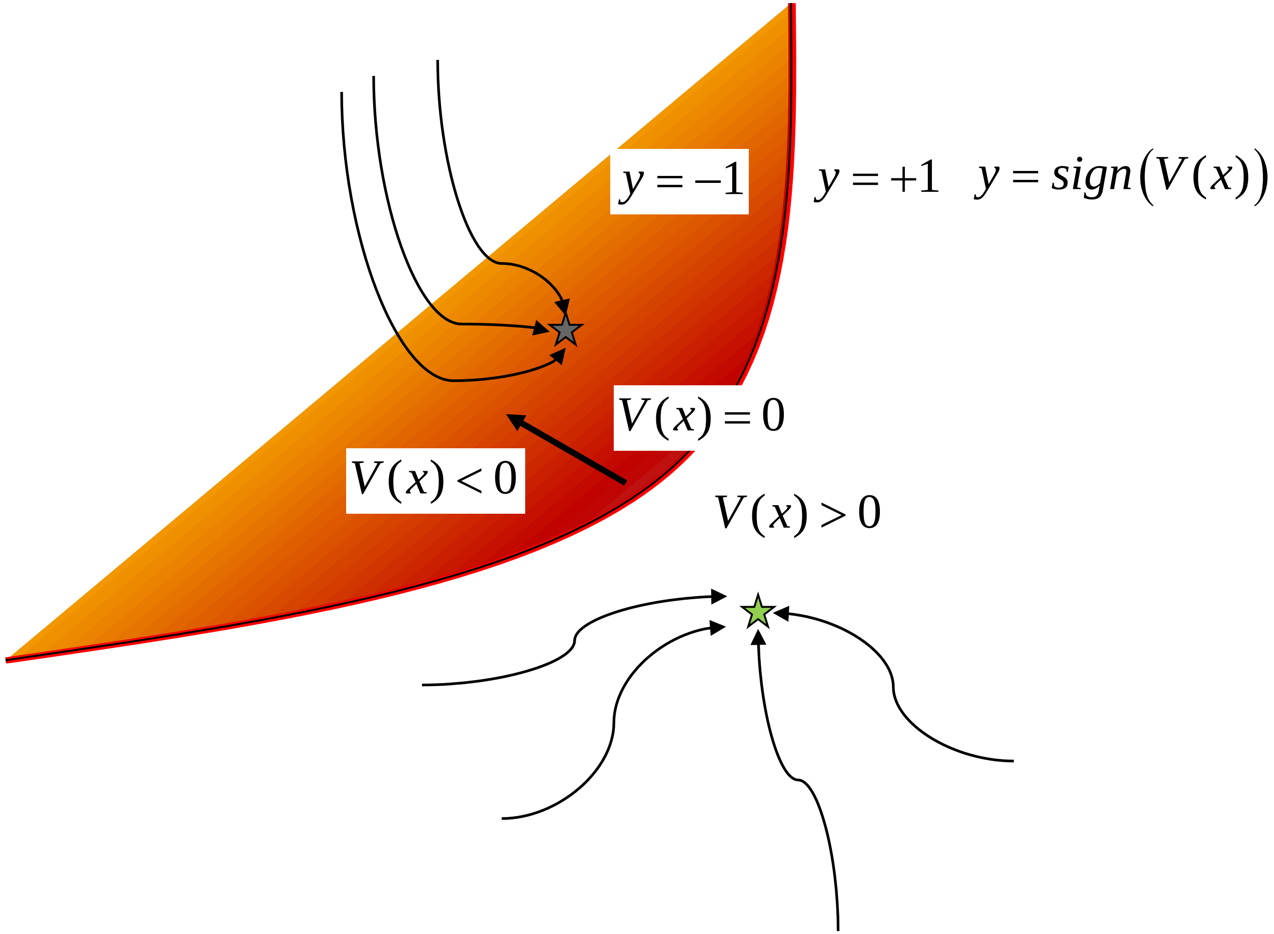
Robotic Implementation



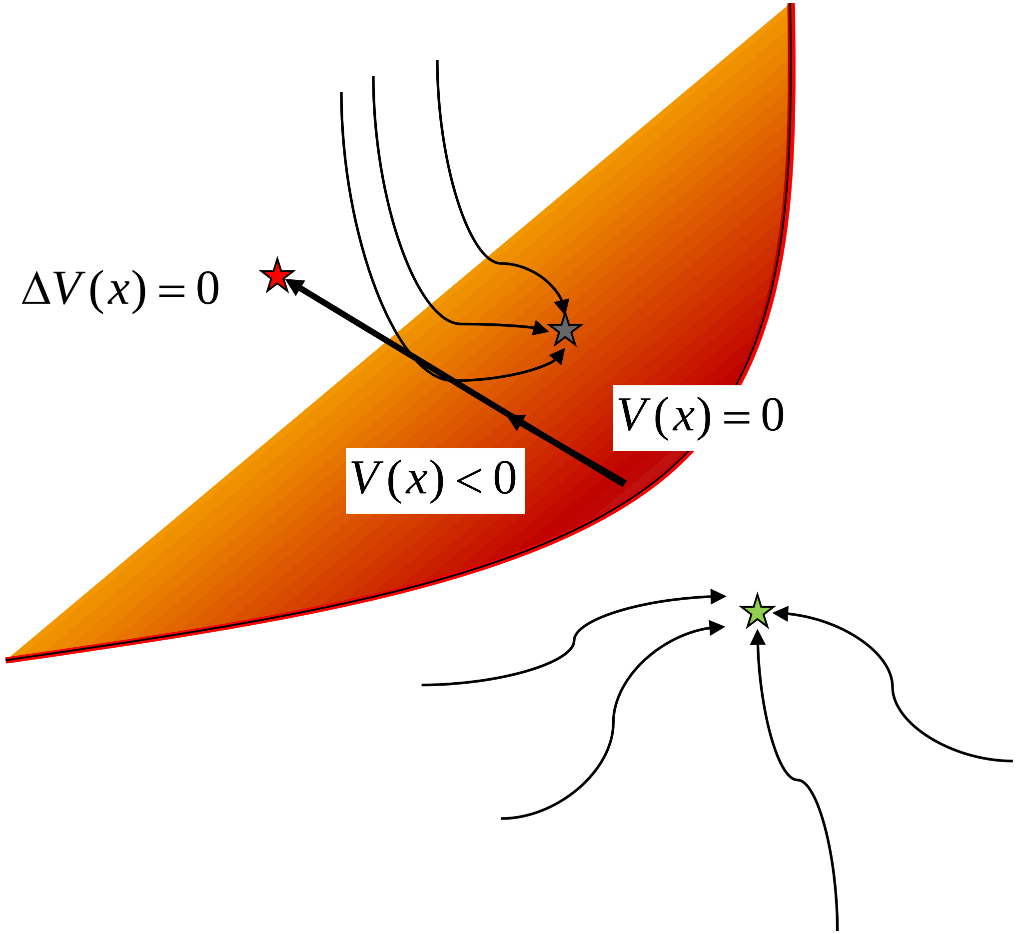
Multi-attractor DS


~~$\dot{x} = f_1(x)$~~
$\dot{x} = f(x)$
~~$\dot{x} = f_2(x)$~~
~~$\dot{x} = f_3(x)$~~

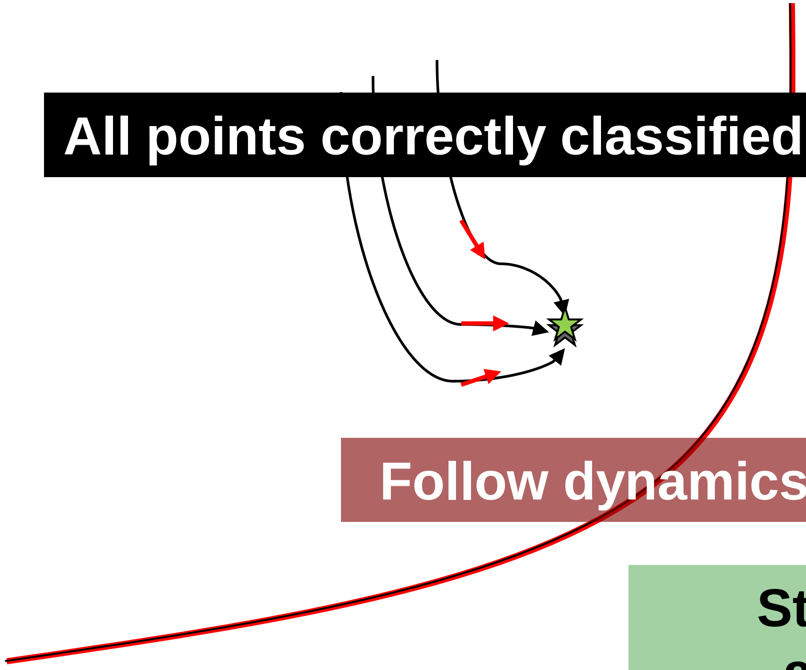
Boundary Created by Support Vector Machine



Boundary Created by Support Vector Machine



Augmented Support Vector Machine



All points correctly classified

Follow dynamics

Standard SVM

$$
\begin{aligned}
&\min_{\mathbf{w},\xi_i} \frac{1}{2}\|\mathbf{w}\|^2 + c\sum_{i=1}^{M}\xi_i \quad \textbf{subject to} \\
&y_i\left(\mathbf{w}^T\phi(\mathbf{x}_i)+b\right) \geq 1 \qquad \forall i = 1\cdots M \\
&y_i\mathbf{w}^T \mathrm{J}(\mathbf{x}_i)\dot{\hat{\mathbf{x}}}_i + \xi_i > 0 \qquad \forall i = 1\cdots M \\
&\xi_i > 0 \qquad \forall i = 1\cdots M \\
&\mathbf{w}^T \mathrm{J}(\mathbf{x}^*)\mathbf{e}_i = 0 \qquad \forall i = 1\cdots N
\end{aligned}
$$

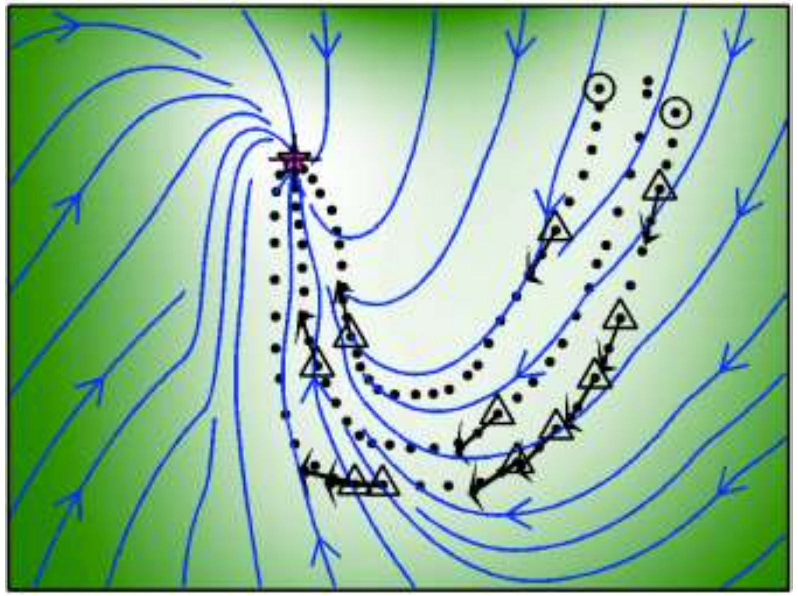
$\{\mathbf{e}_i\} \rightarrow$ Cannonical basis of $\mathbb{R}^N$

Stability at attractor

Convex Optimization

$\odot \equiv \alpha - \mathrm{SV}$

$\Delta \equiv \beta - \mathrm{SV}$



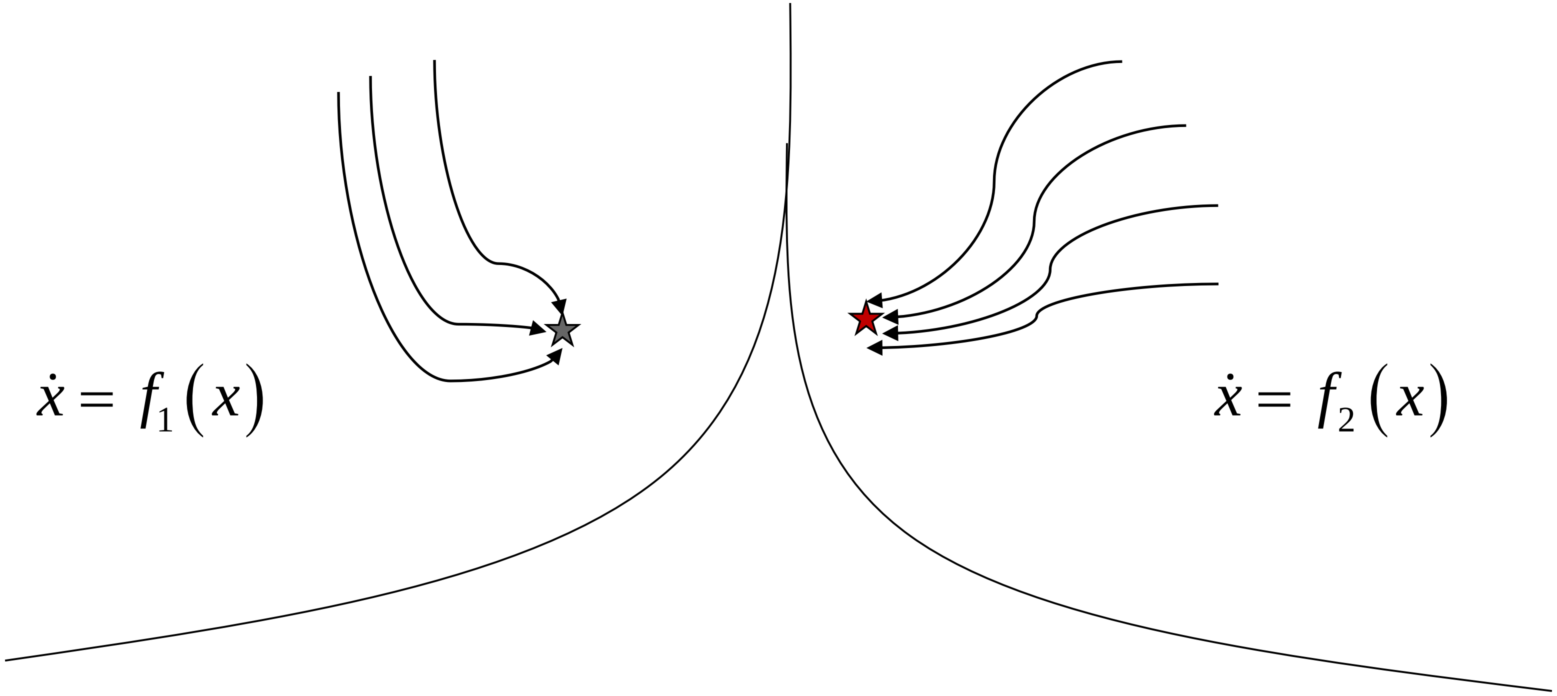
$$f(x) = \sum_{i=1}^{M} \alpha_i y_i k(\mathbf{x}, \mathbf{x}_i) + \sum_{i=1}^{M} \beta_i \hat{\dot{\mathbf{x}}}_i^{\mathrm{T}} \frac{\partial k(\mathbf{x}, \mathbf{x}_i)}{\partial \mathbf{x}_i} - \sum_{i=1}^{N} \gamma_i \mathbf{e}_i^{\mathrm{T}} \frac{\partial k(\mathbf{x}, \mathbf{x}^*)}{\partial \mathbf{x}^*} + b$$

Standard SVM α- SVs

New β- SVs

Non-linear bias

Const. bias

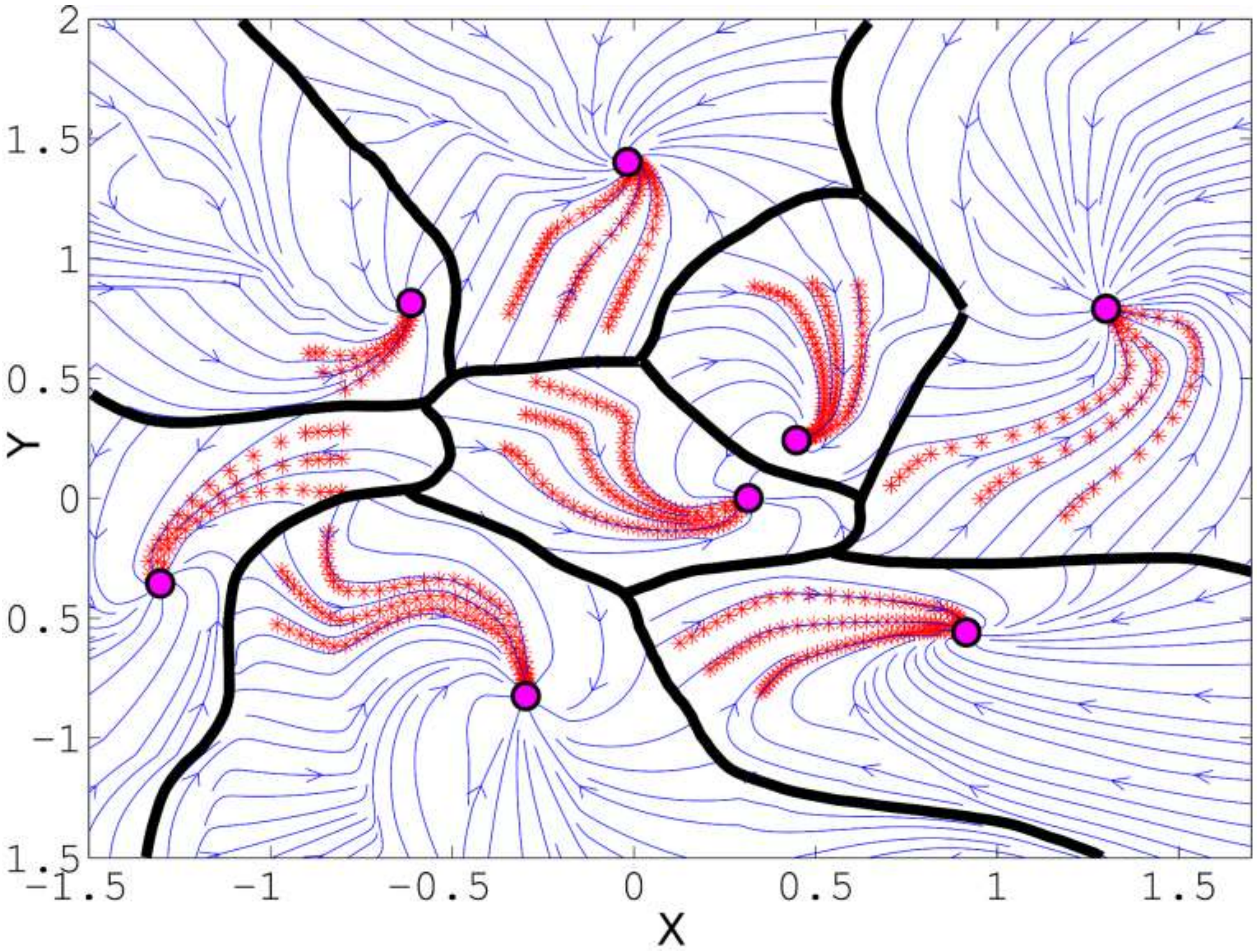


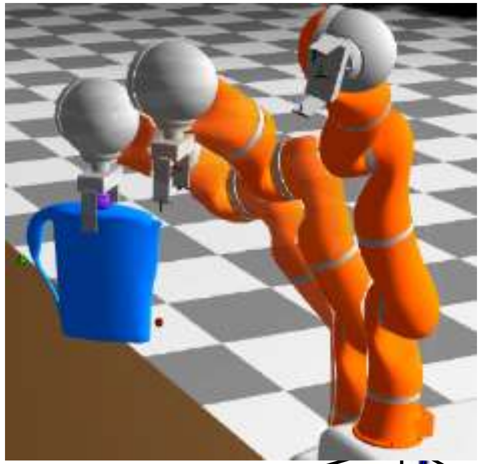
$$\dot{x} = f_1(x)$$

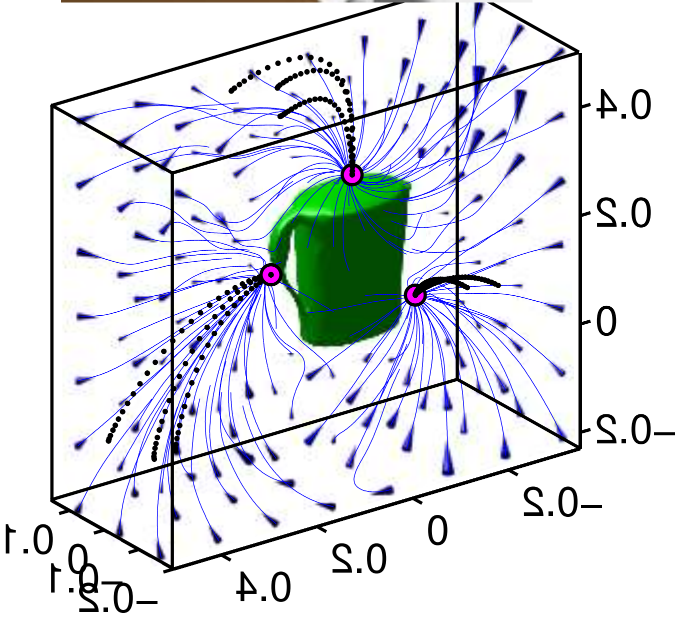
$$\dot{x} = f_2(x)$$

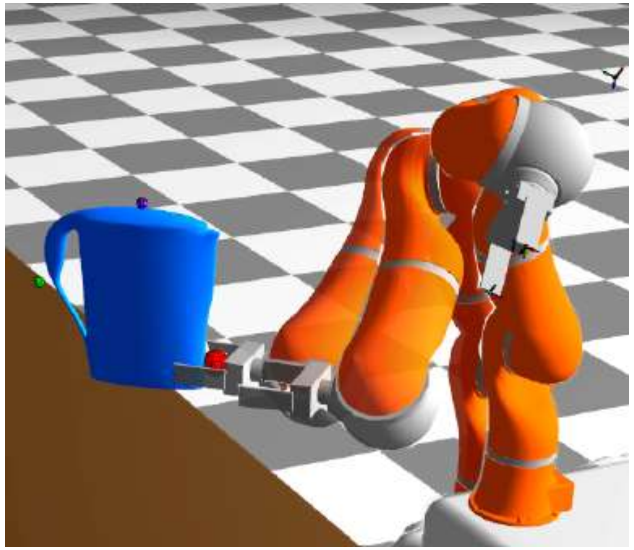
$$\dot{x} = f(x) = \arg\max_i f_i(x)$$

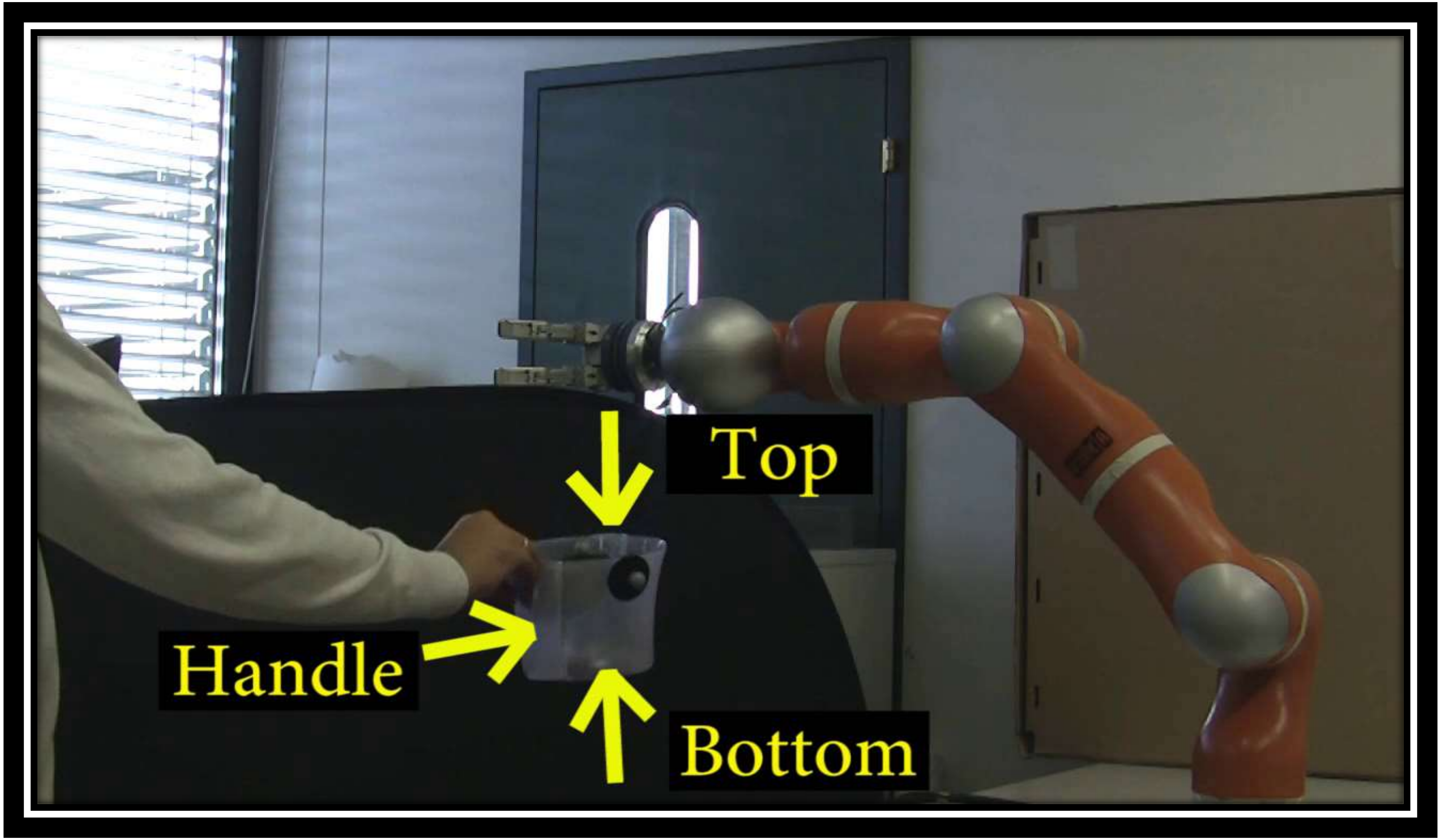
Exact partitioning of the space

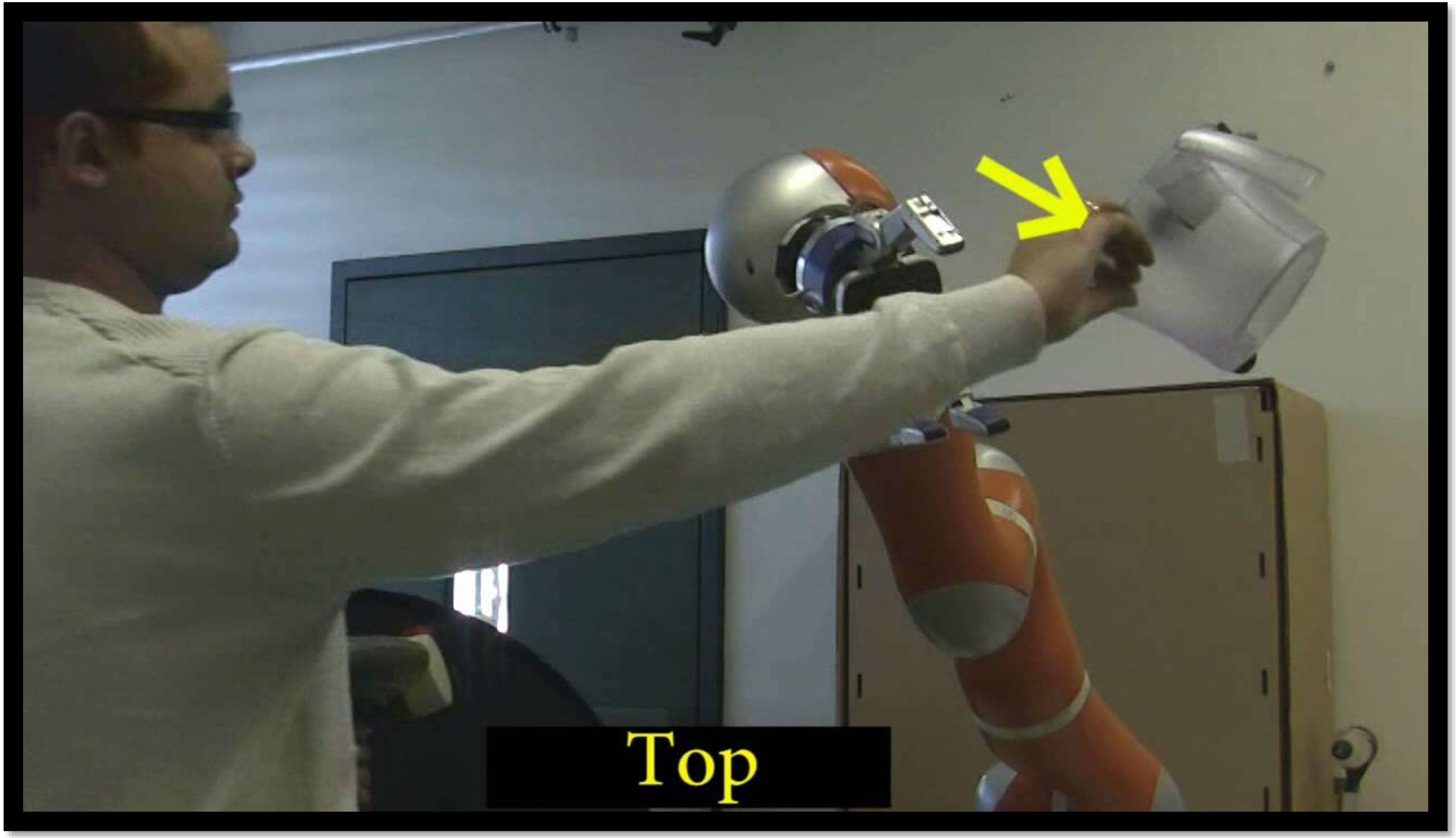


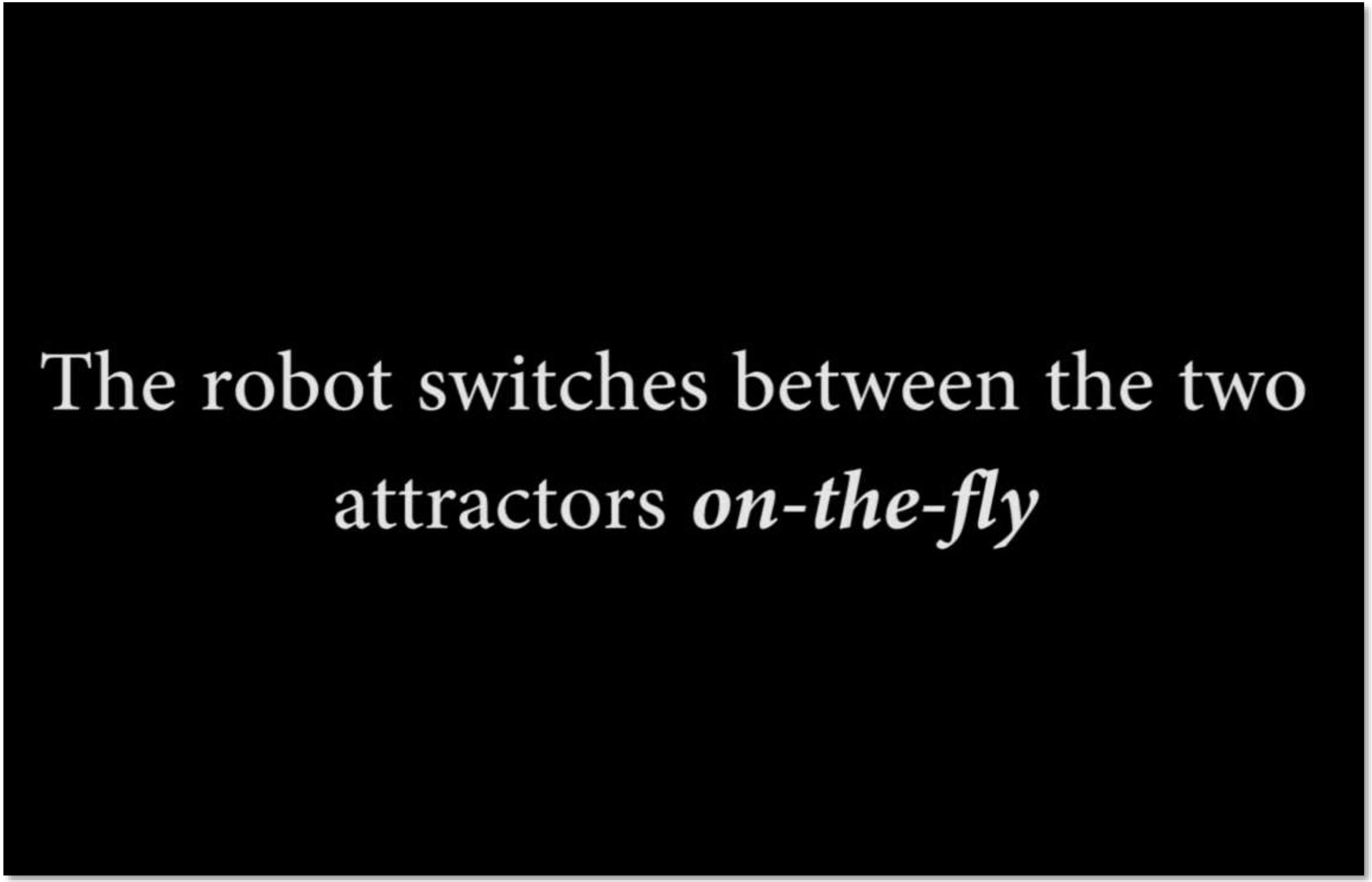







Top
Handle
Bottom

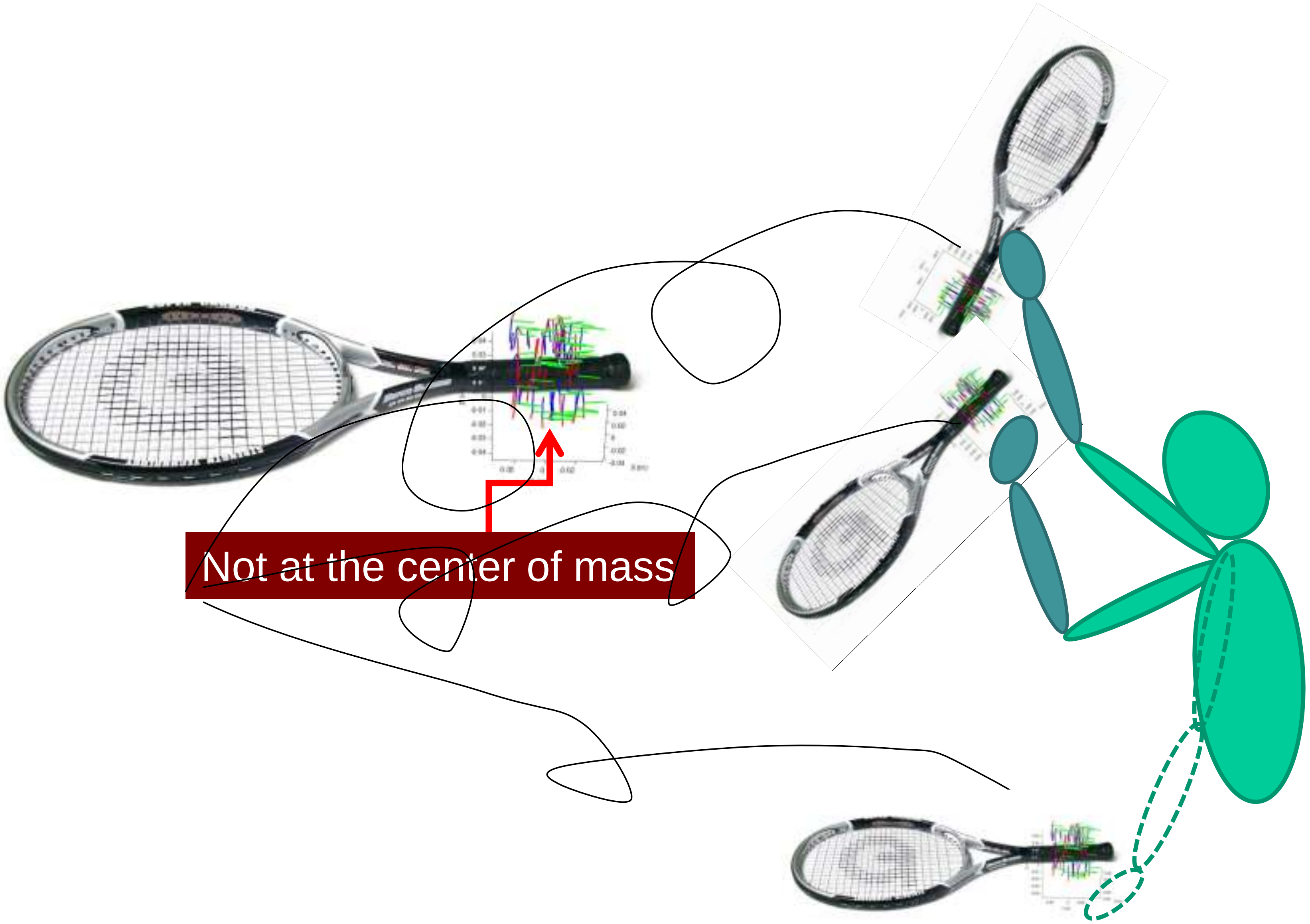

Top


The robot switches between the two
attractors *on-the-fly*

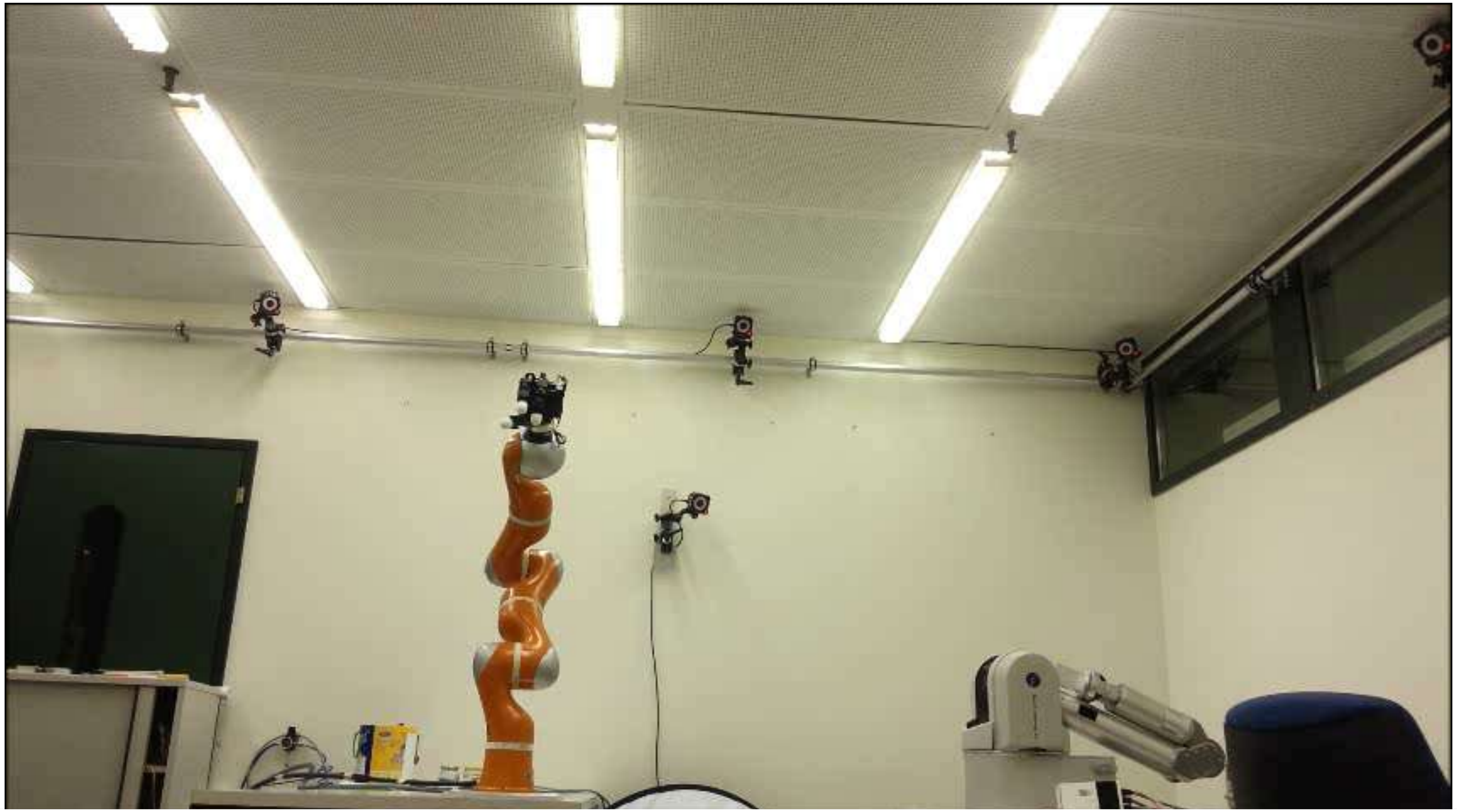

LASA
ÉCOLE POLYTECHNIQUE
FÉDÉRALE DE LAUSANNE

Catching Objects in Flight

Kim, Shukla and Billard, IEEE Transaction on Robotics, 2014, *Robotics King-Sun Fu Memorial Best Paper Award*


Not at the center of mass

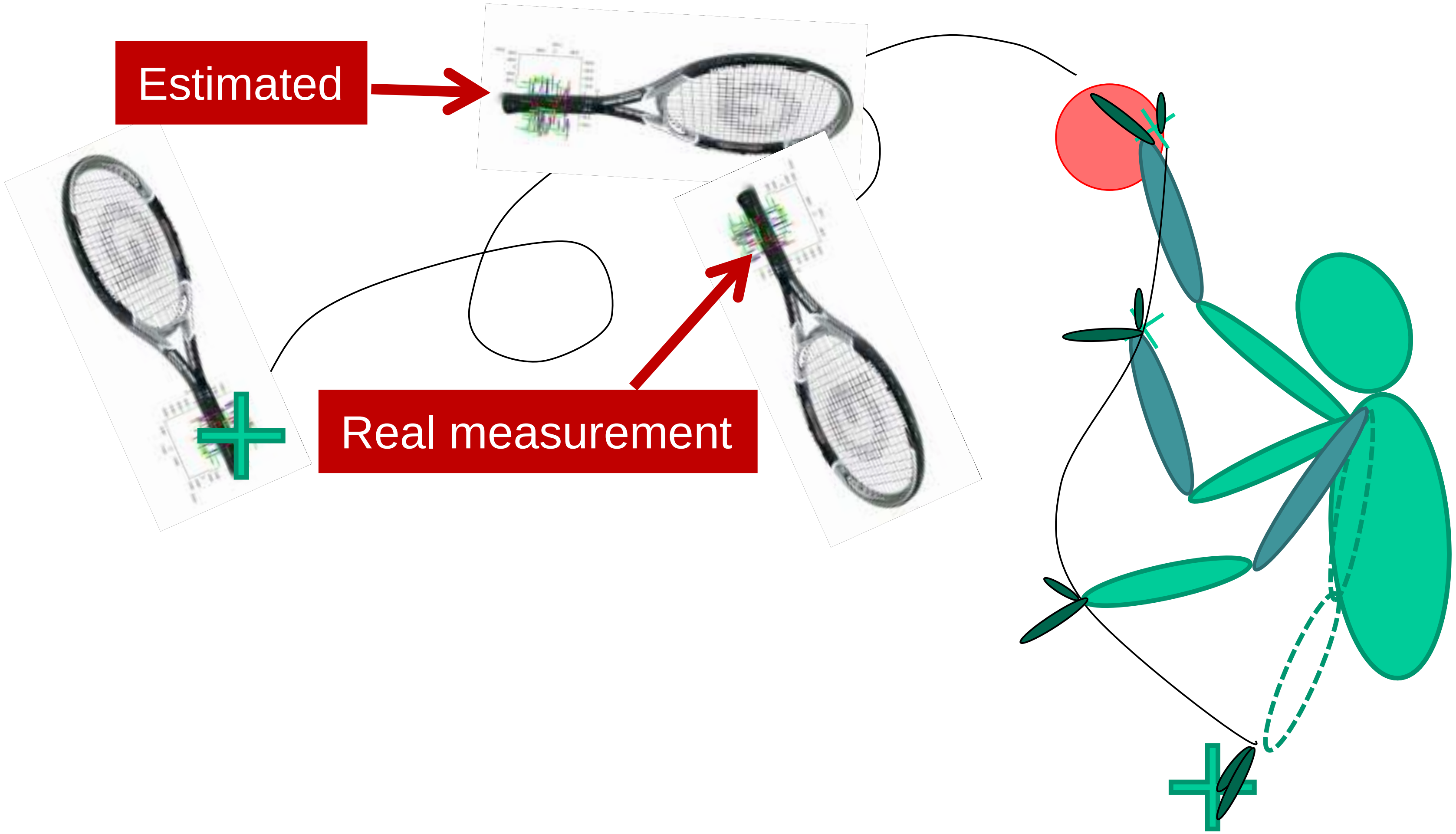
Modeling Object's Nonlinear Flying Dynamics

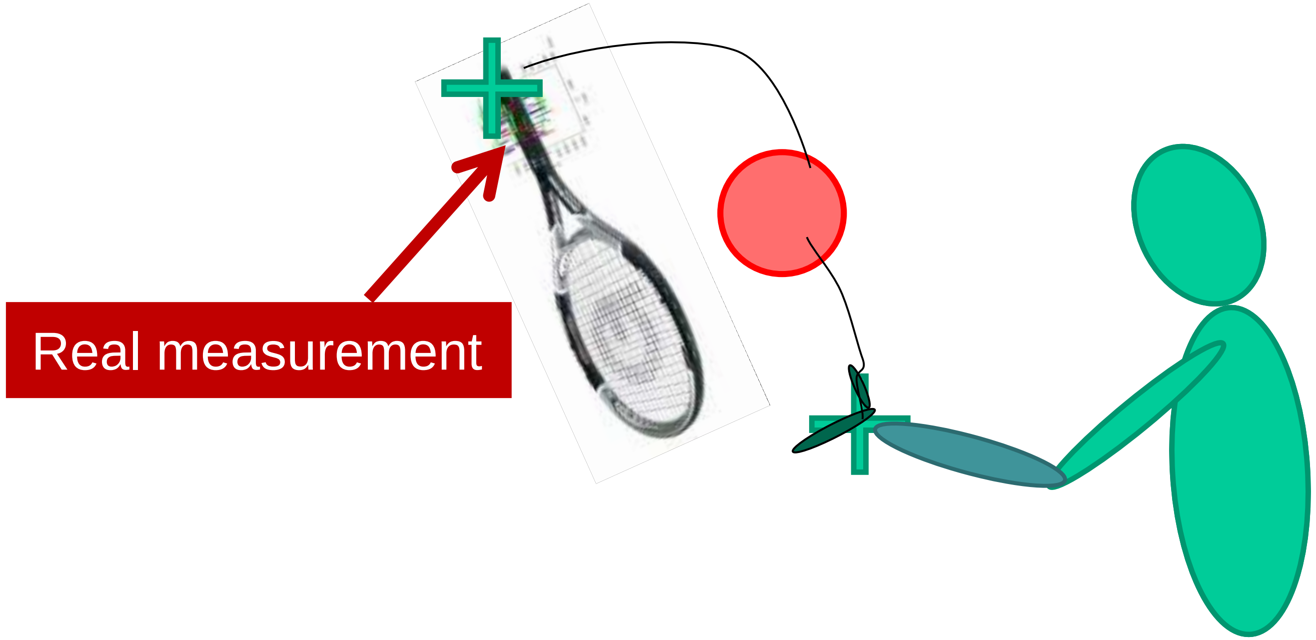


Build model of dynamics using
Support Vector Regression

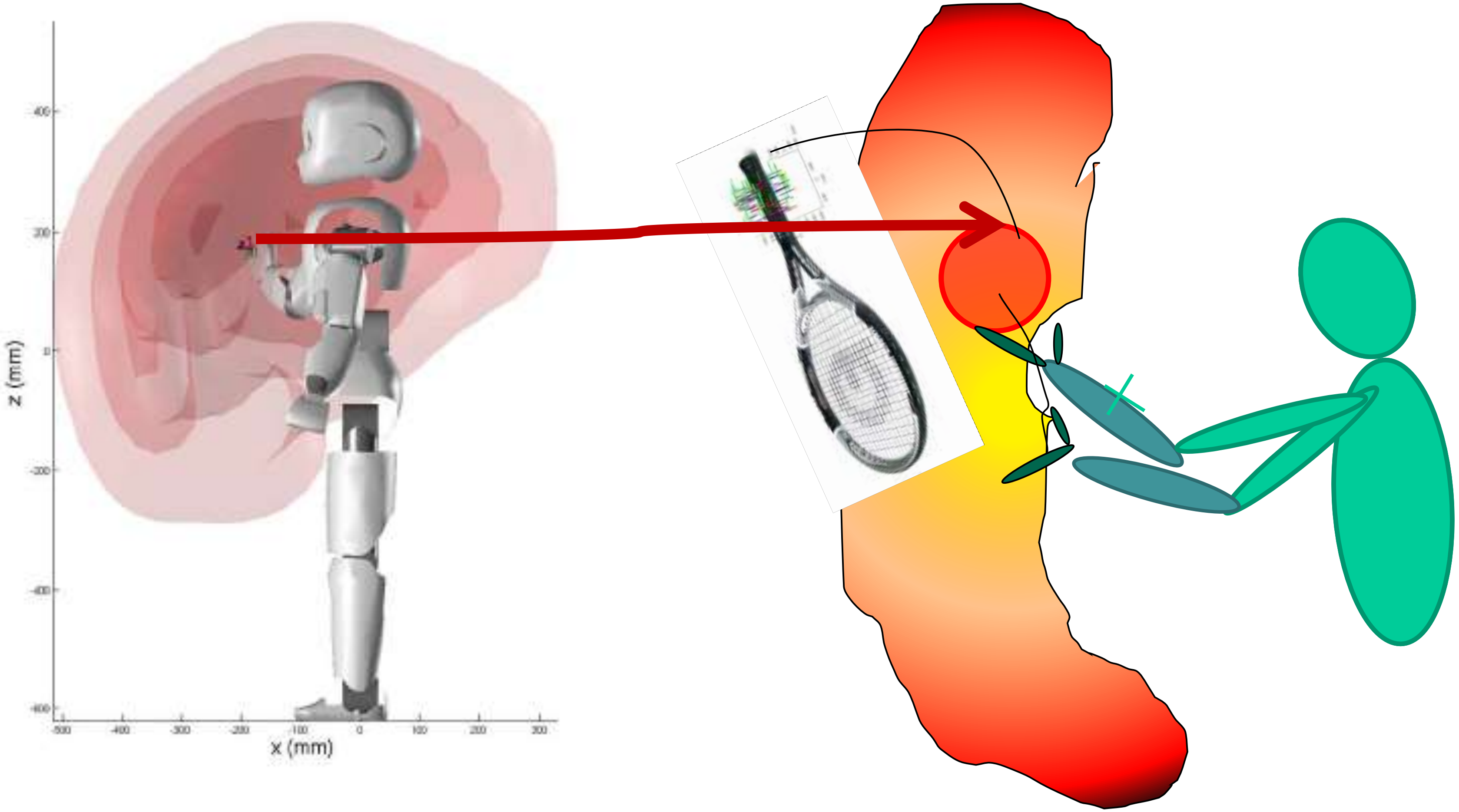
$$\ddot{x} = \sum_{i=1}^{M} \alpha_i k\left(\left[x^i \dot{x}^i\right]^T, \left[x\ \dot{x}\right]^T\right) + b$$

Compute derivative (closed-form) for Extended Kalman Filter

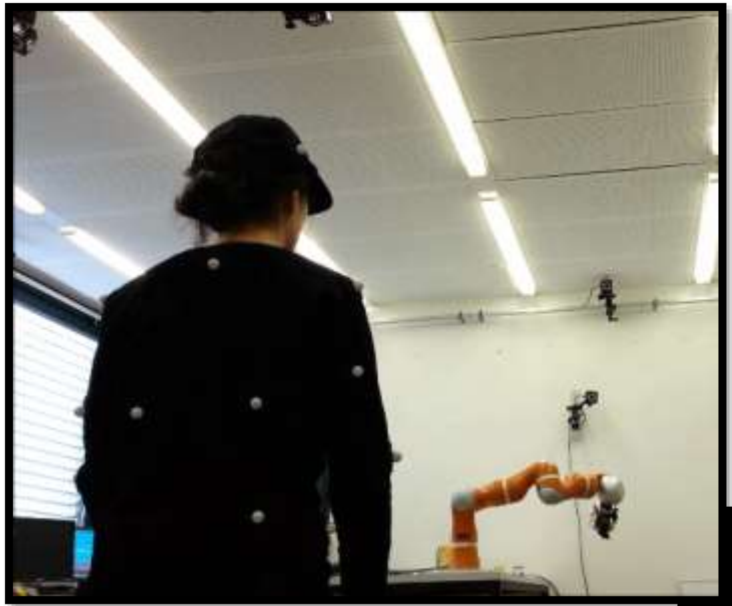

Estimated
Real measurement

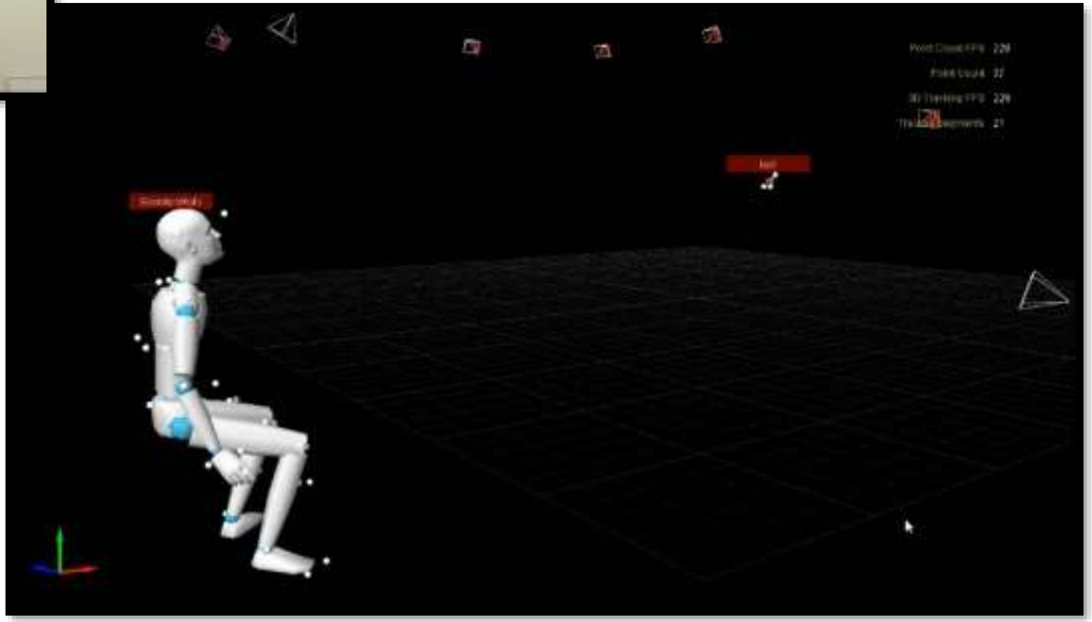

Real measurement

Learn most likely region to catch object

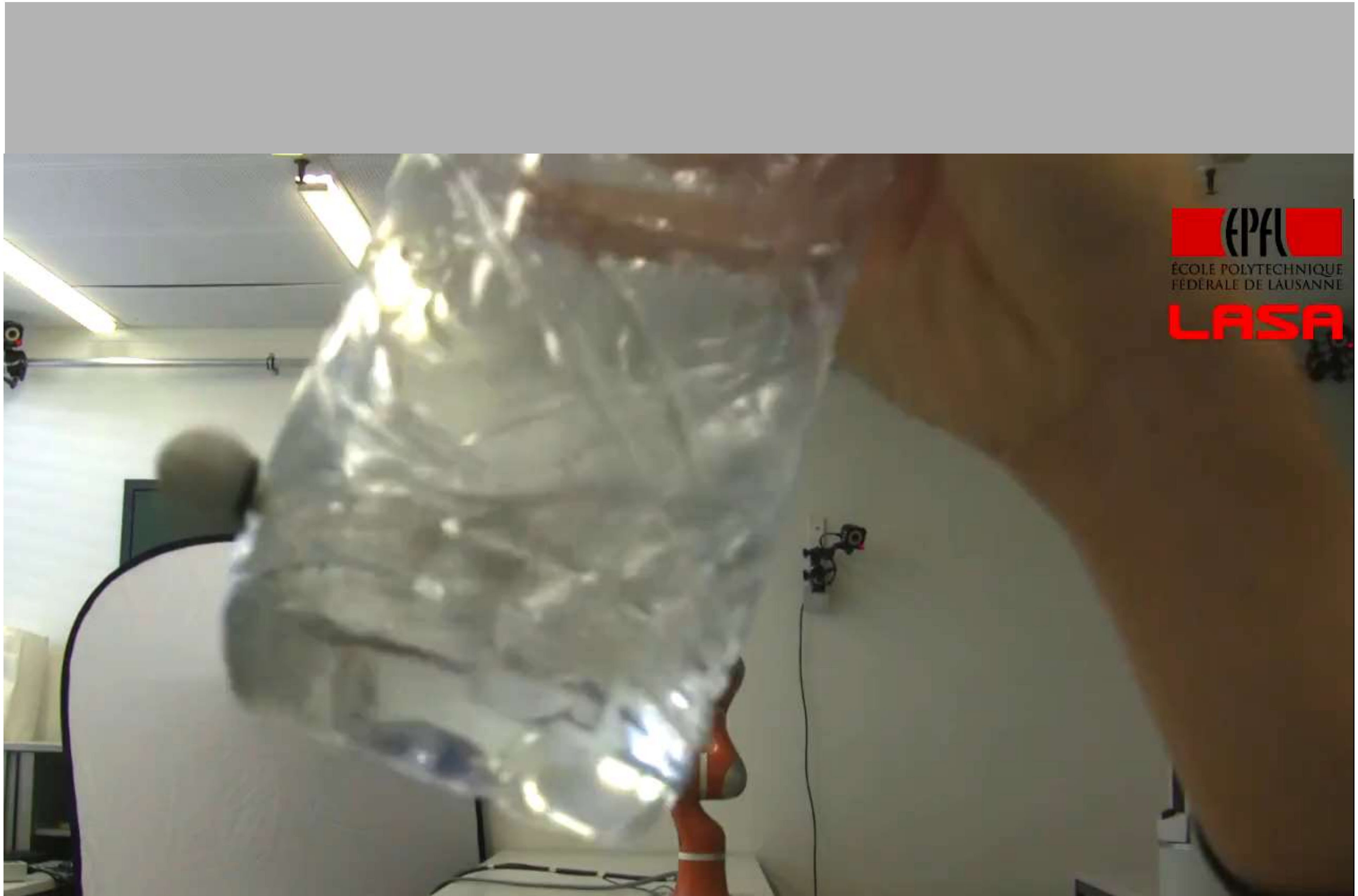


Learn arm-hand coupling



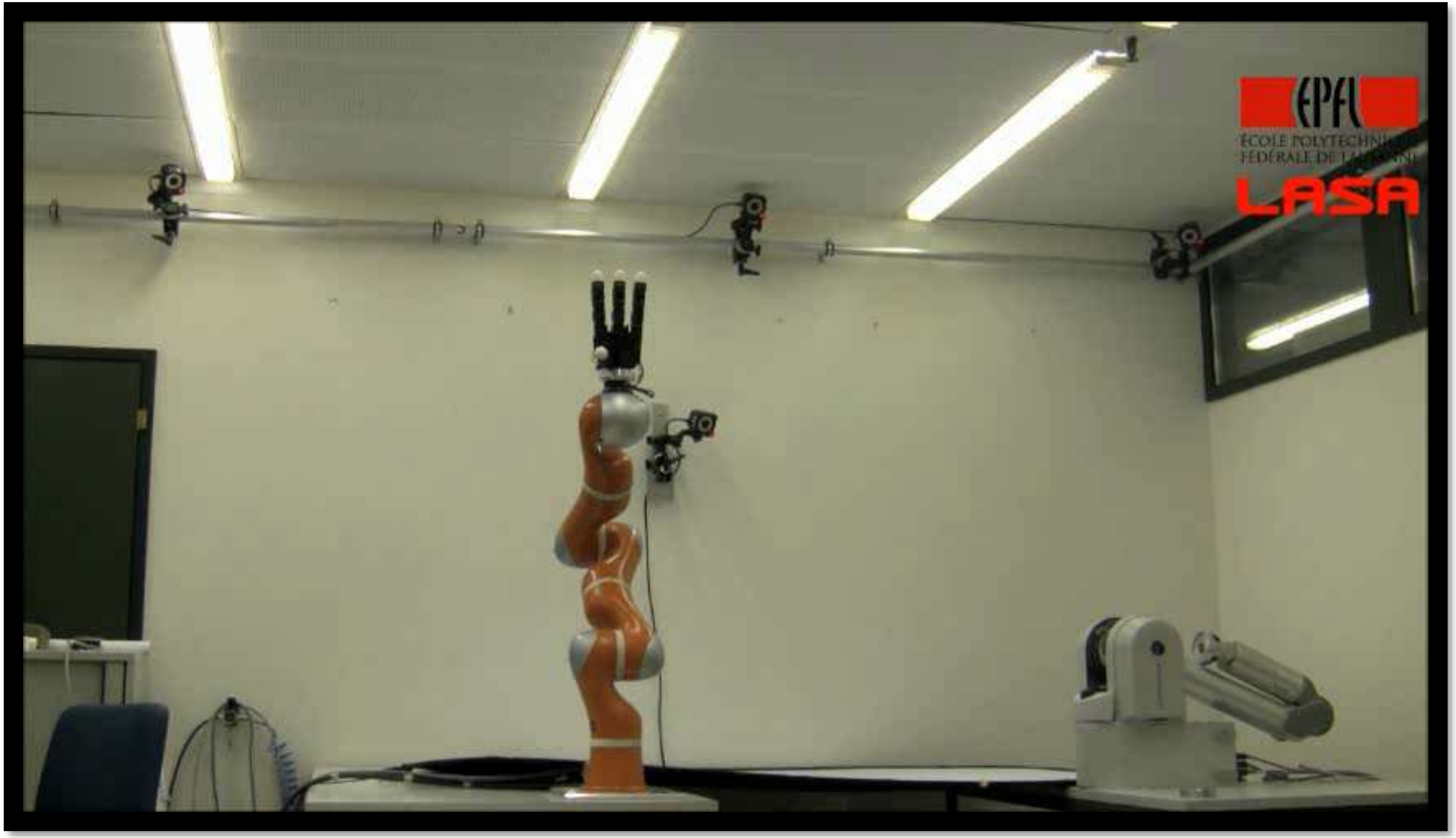



EPFL
ÉCOLE POLYTECHNIQUE
FÉDÉRALE DE LAUSANNE
LASA


EPFL
ÉCOLE POLYTECHNIQUE
FÉDÉRALE DE LAUSANNE
LASA

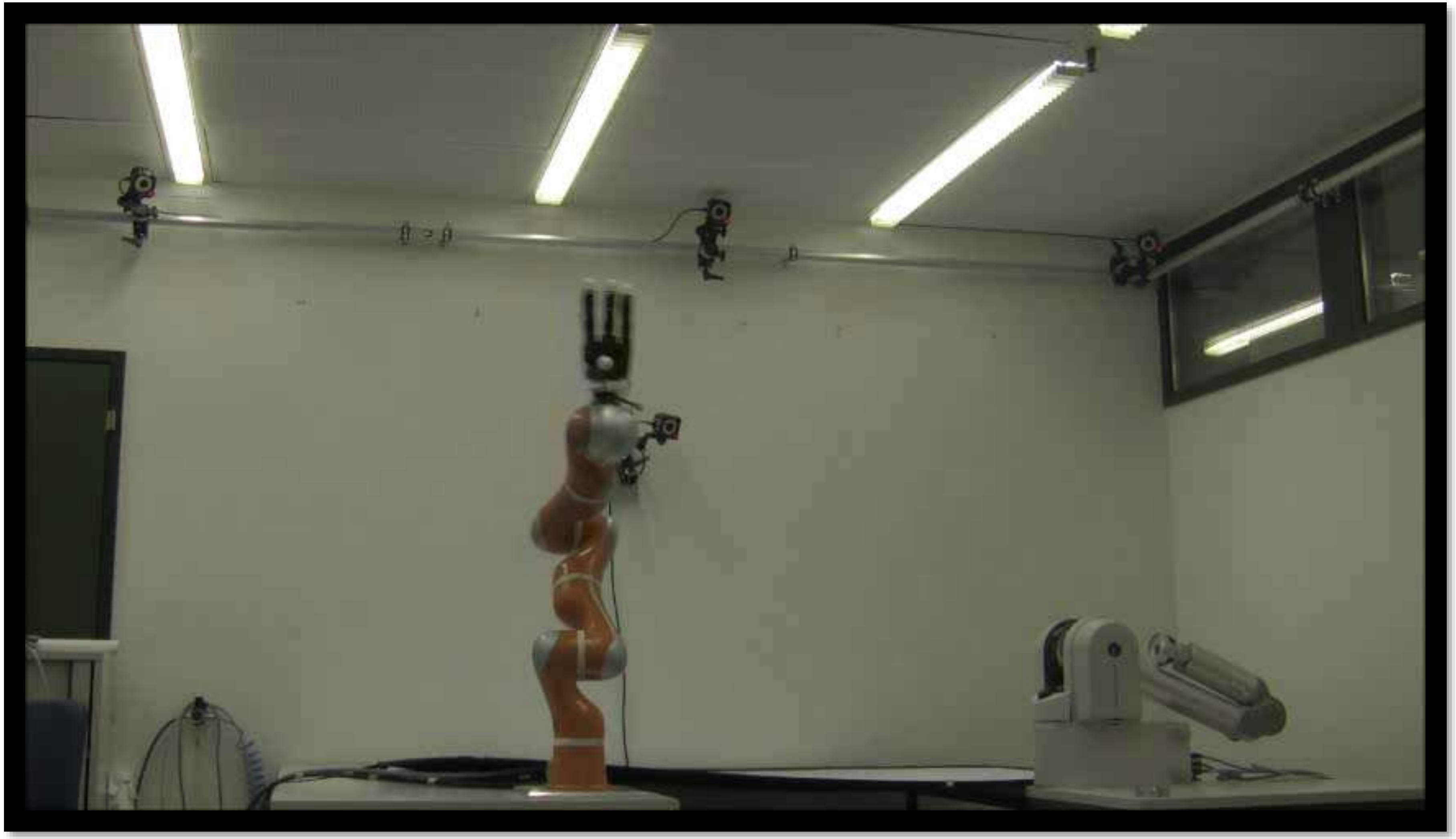
Learning and adaptive control for robots

EPFL

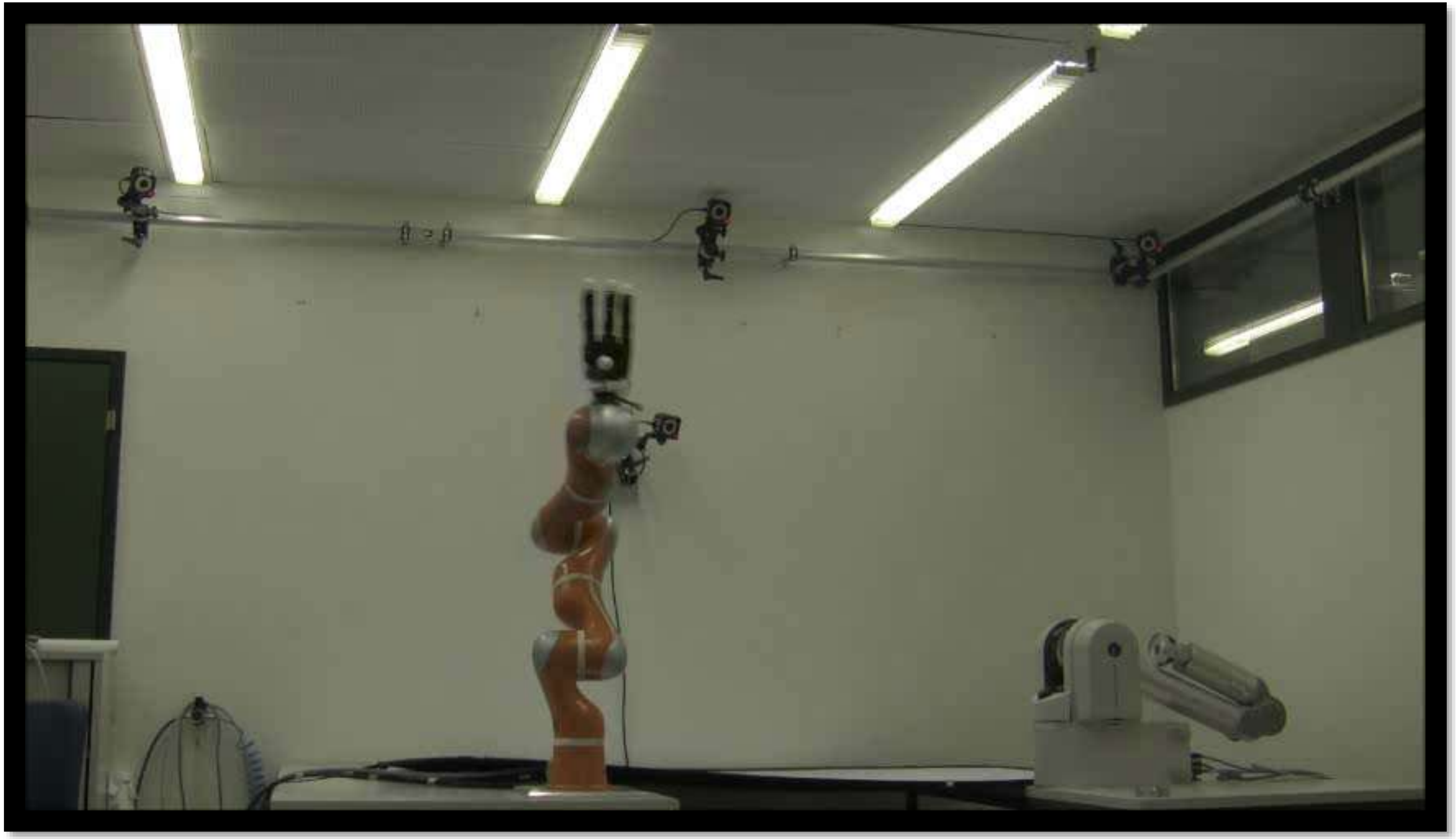


When it fails …

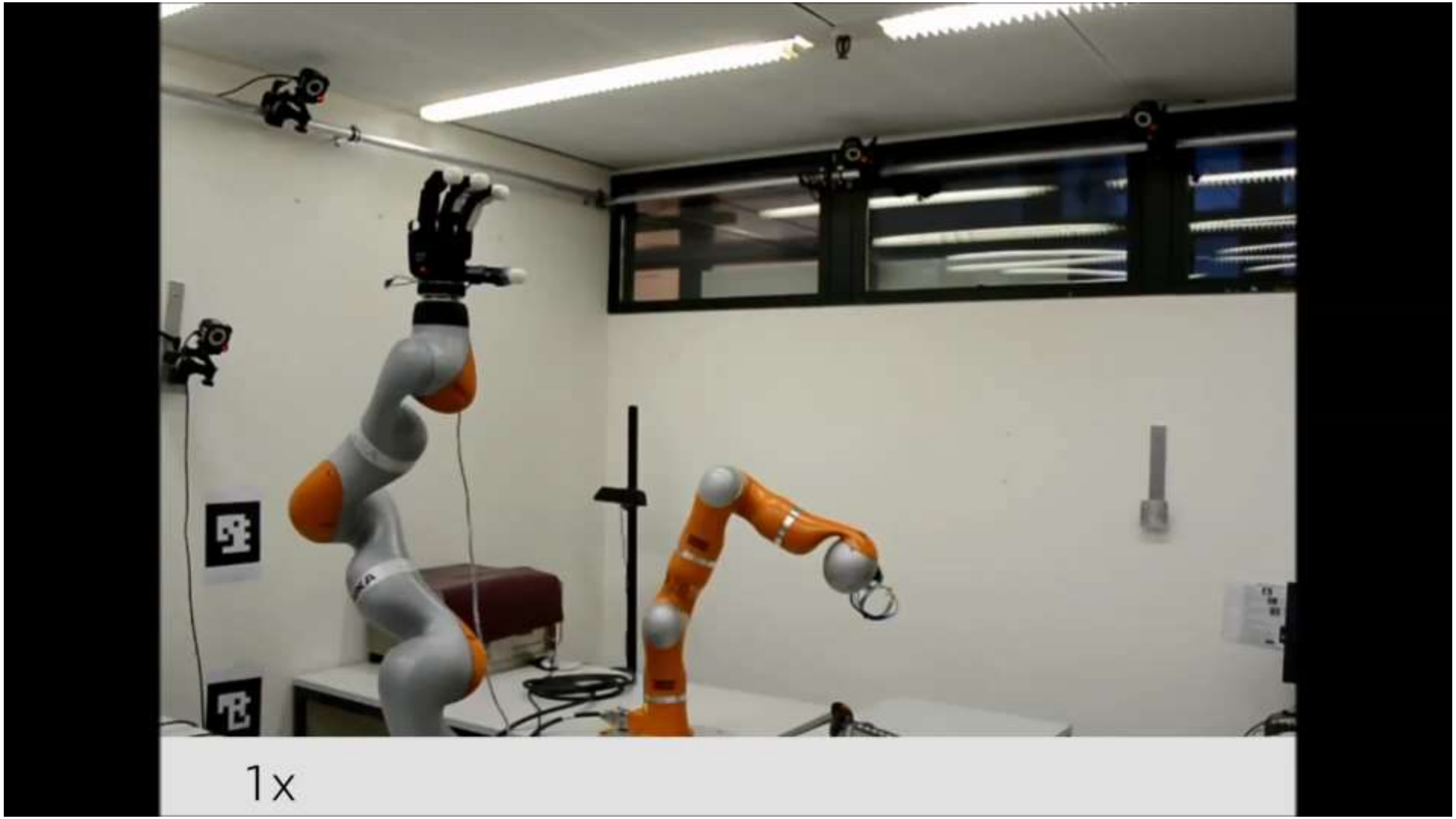
Just on time!



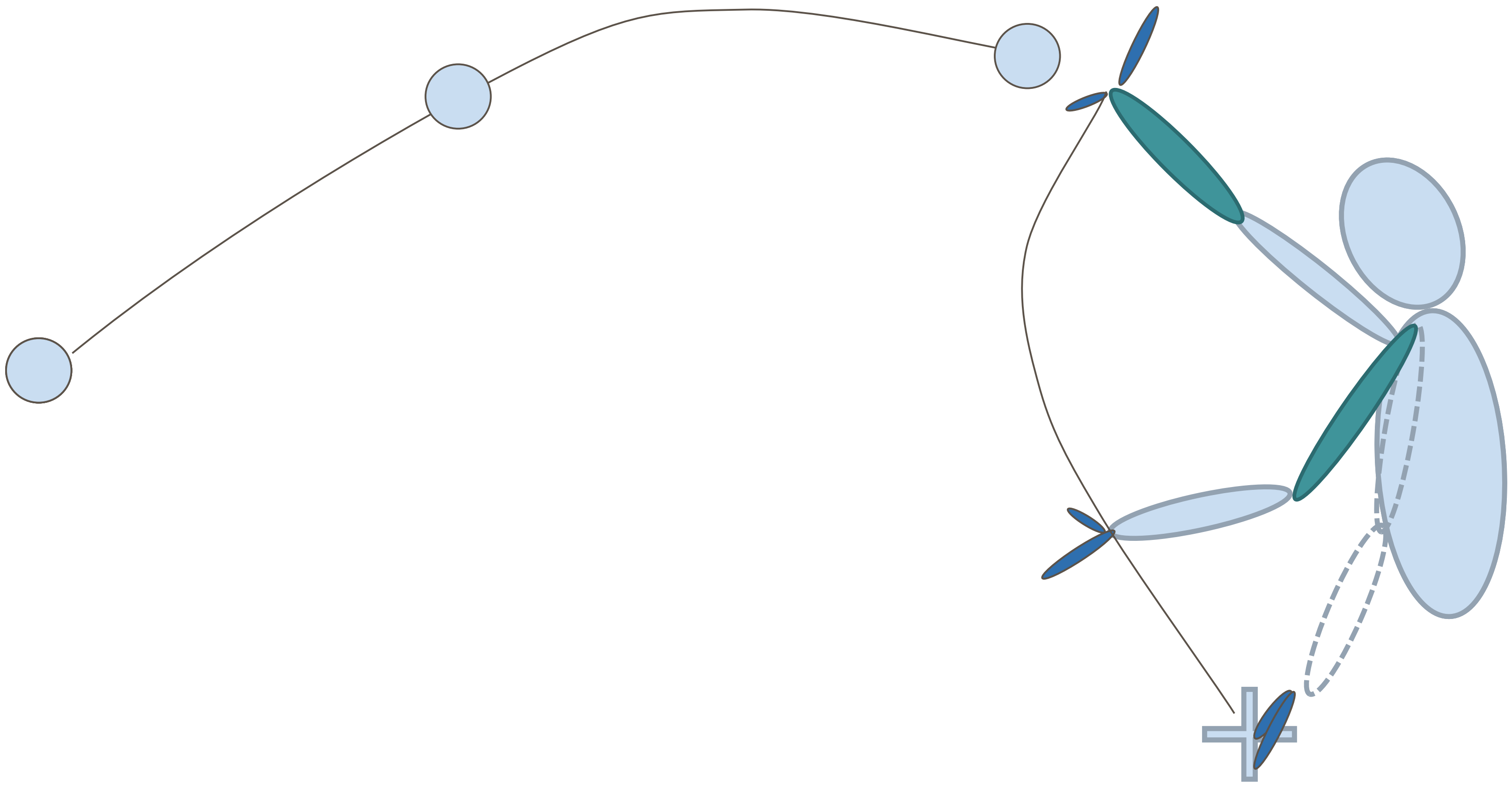
Failure due to imprecise closing of fingers



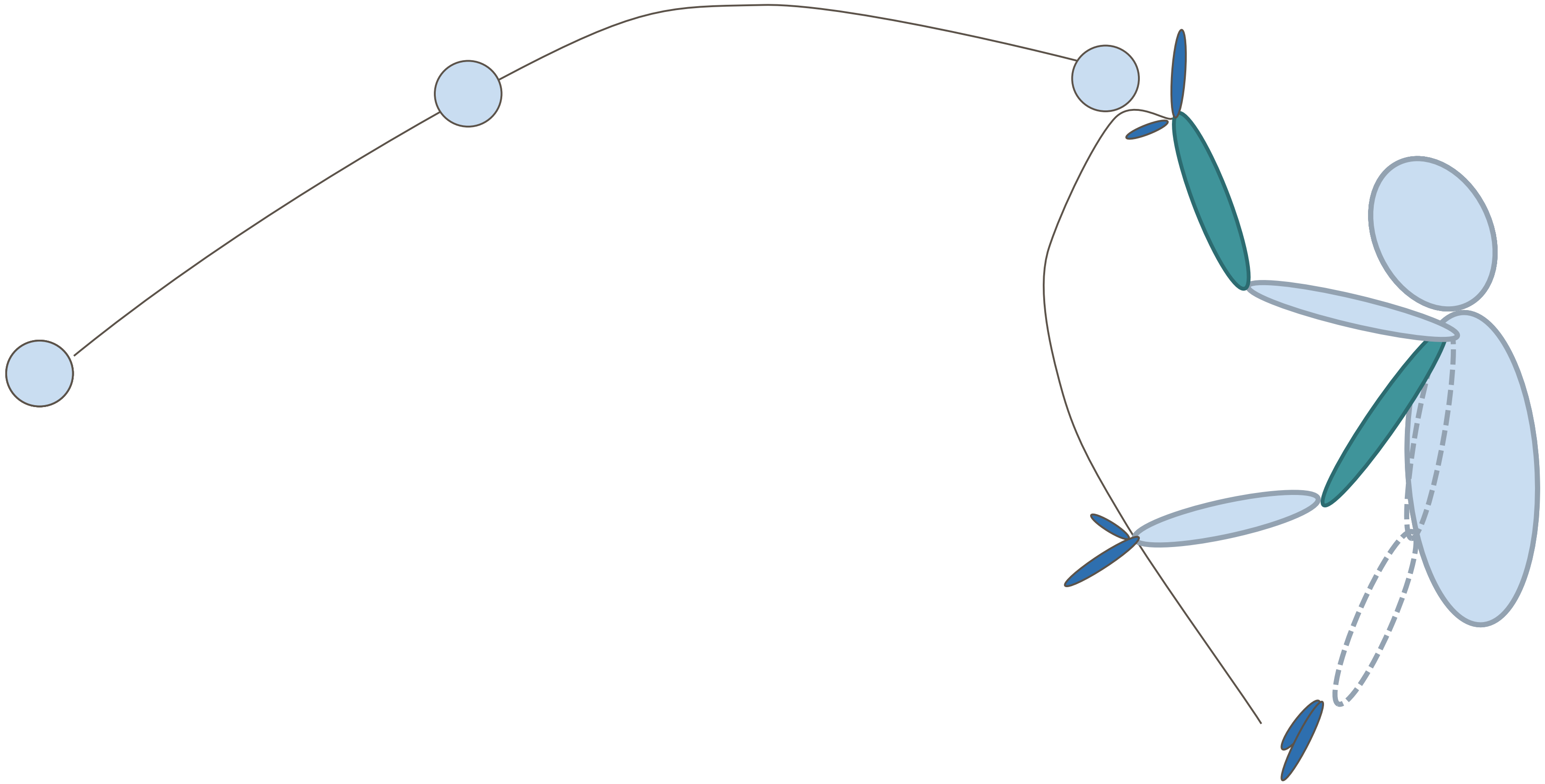
Failure for lack of time to close the fingers



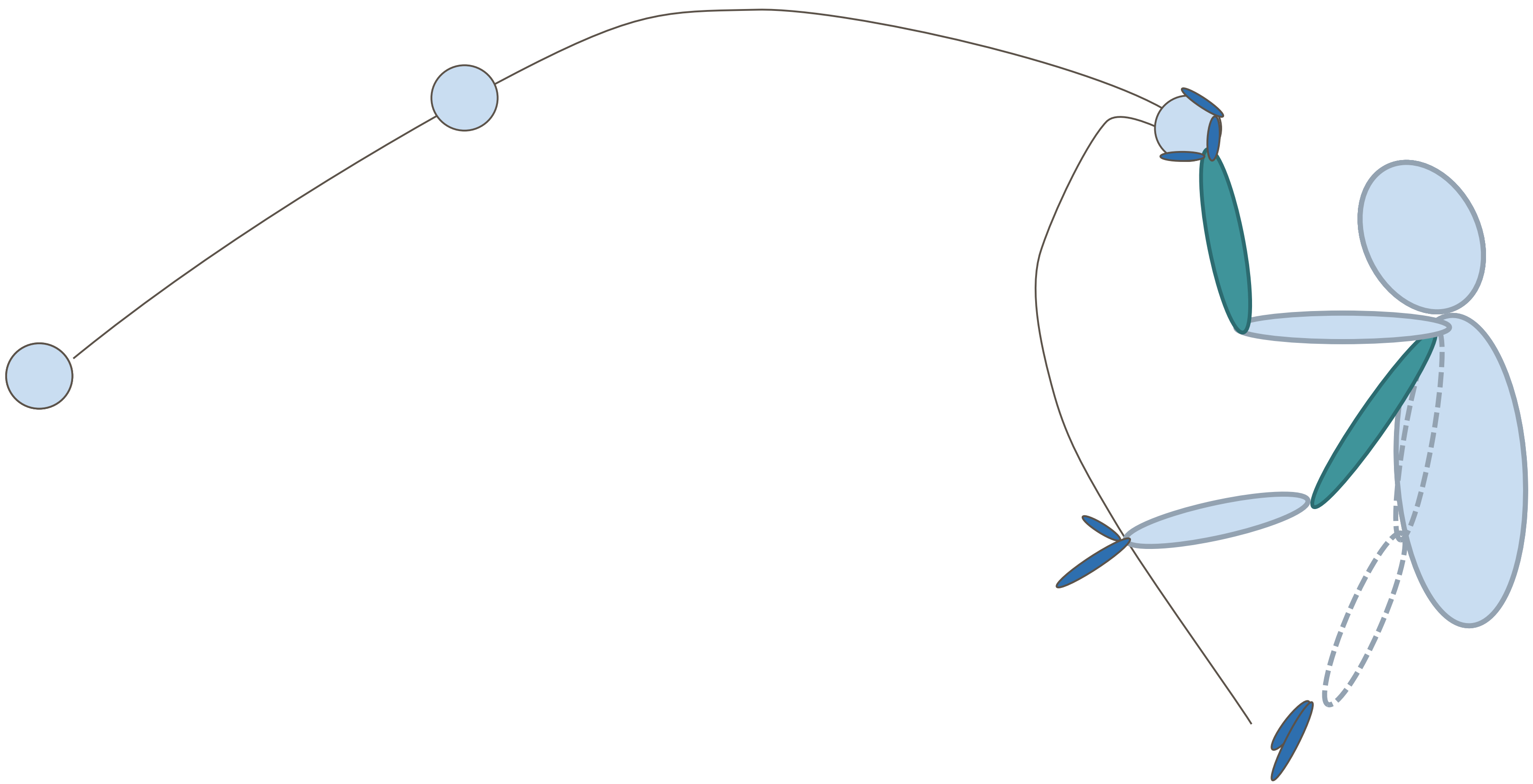
Soft Catching Strategy



Soft Catching Strategy



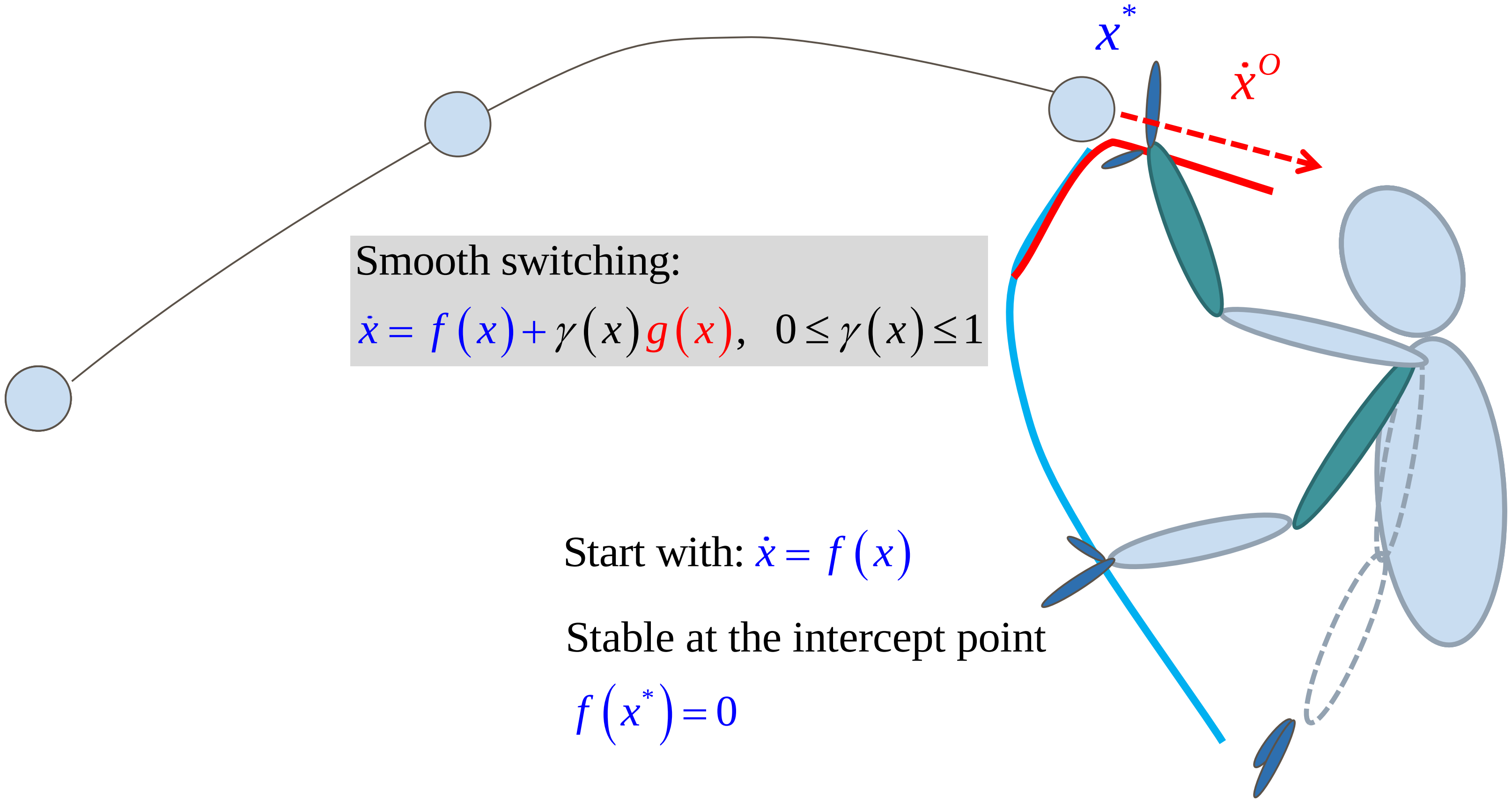
Soft Catching Strategy



Soft Catching Strategy

Tracks the object: $\dot{x} = g(x)$

$\lim_{t \to \infty} \| \dot{x} - \dot{x}^0 \| = 0$



Smooth switching:

$\dot{x} = f(x) + \gamma(x) g(x), \quad 0 \le \gamma(x) \le 1$

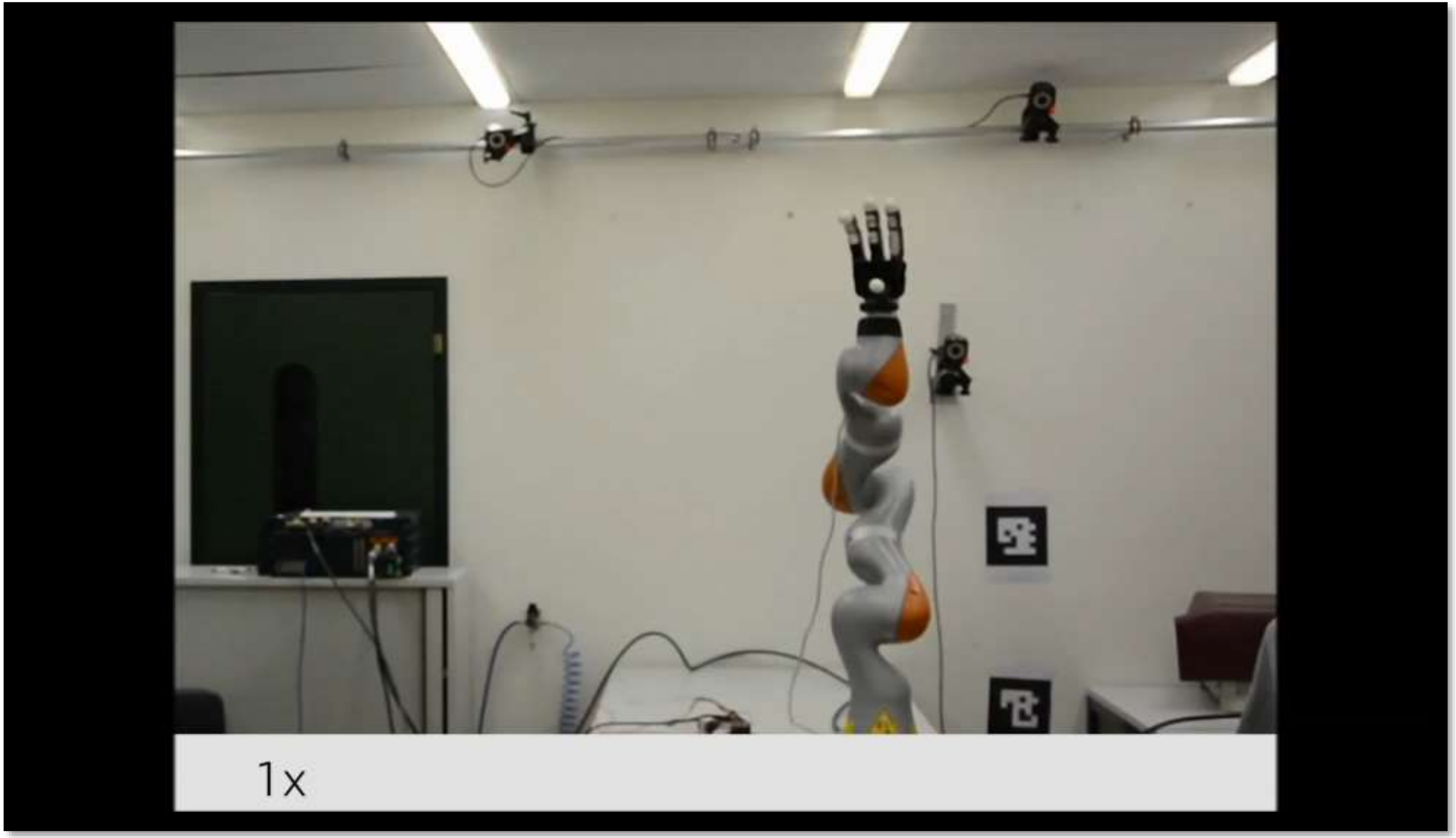
Start with: $\dot{x} = f(x)$

Stable at the intercept point

$f(x^*) = 0$

Mirrazavi, Khoramshavi, Billard, A dynamical systems approach for softly catching a flying object: Theory and experiments, IEEE Transaction on Robotics, 2016

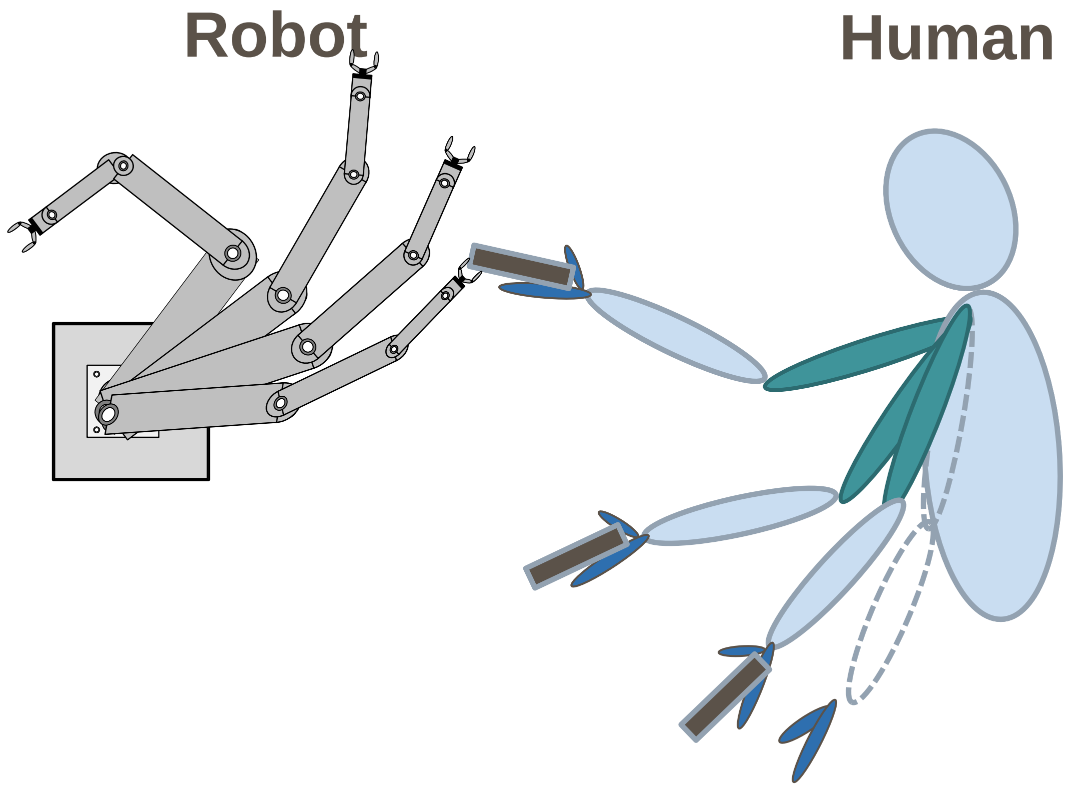
Increase performance to more than 80%



Finalist KUKA Award 2017

Force-modulated DS

Object Handover

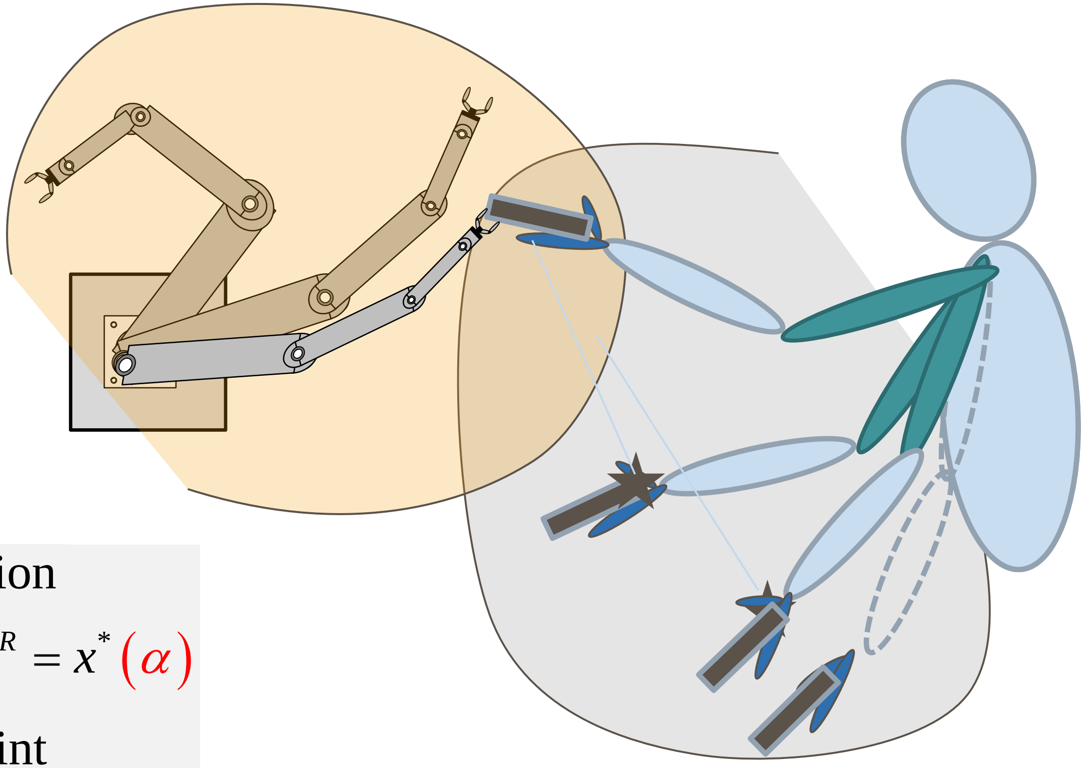


Industrial Application

ocado

Control law for estimating human motion

$\dot{x}^H = A_H x^H$



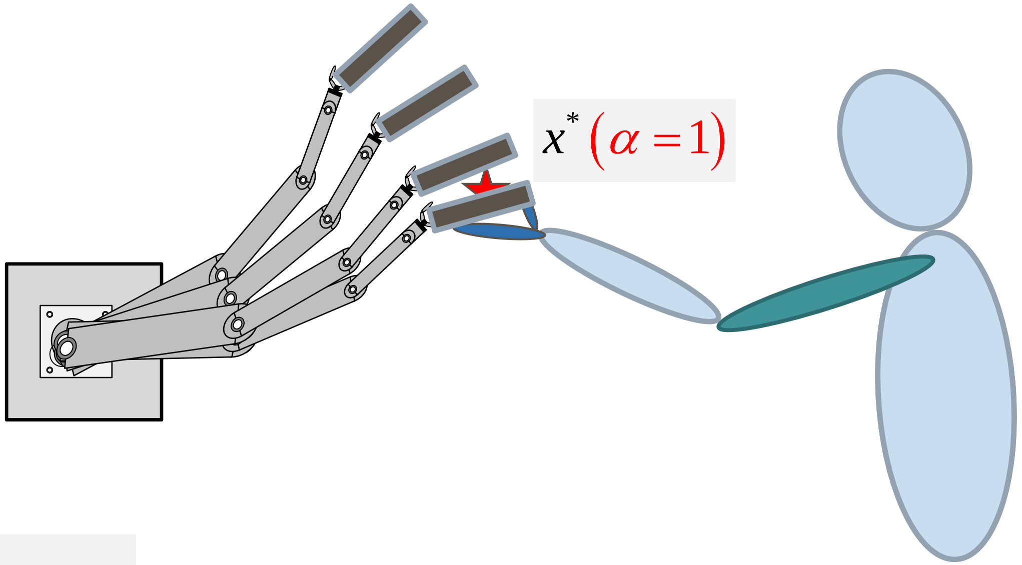
α : Load Share

Control law for robot motion

$\dot{x}^R = f\left(x^R, g(\alpha)\right), \quad \lim_{t\to\infty} x^R = x^*(\alpha)$

Origin on the intercept point

$\alpha = 1:$ Full load on robot



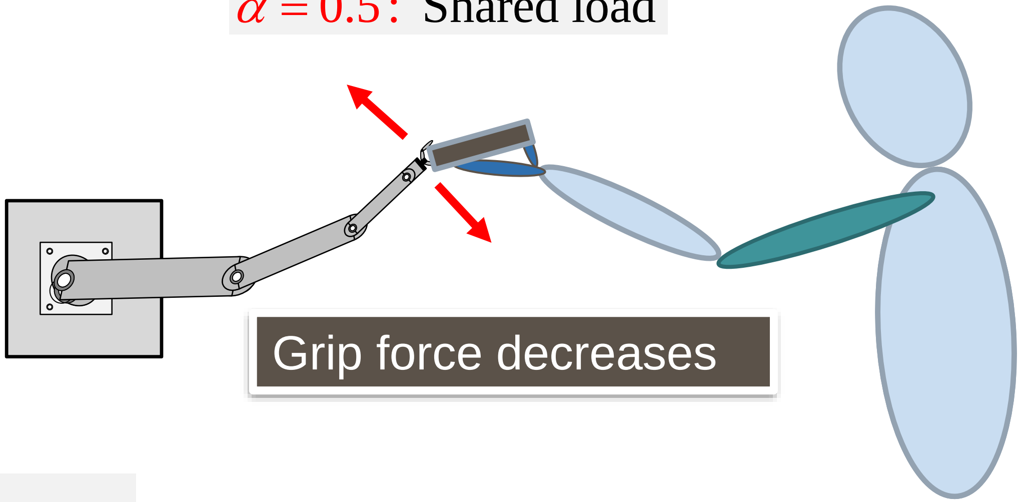
$\alpha:$ Load Share

Control law for robot motion

$$\dot{x}^R = f\left(x^R, g(\alpha)\right), \quad \lim_{t \to \infty} x^R = x^*(\alpha)$$

Origin on the intercept point

EPFL

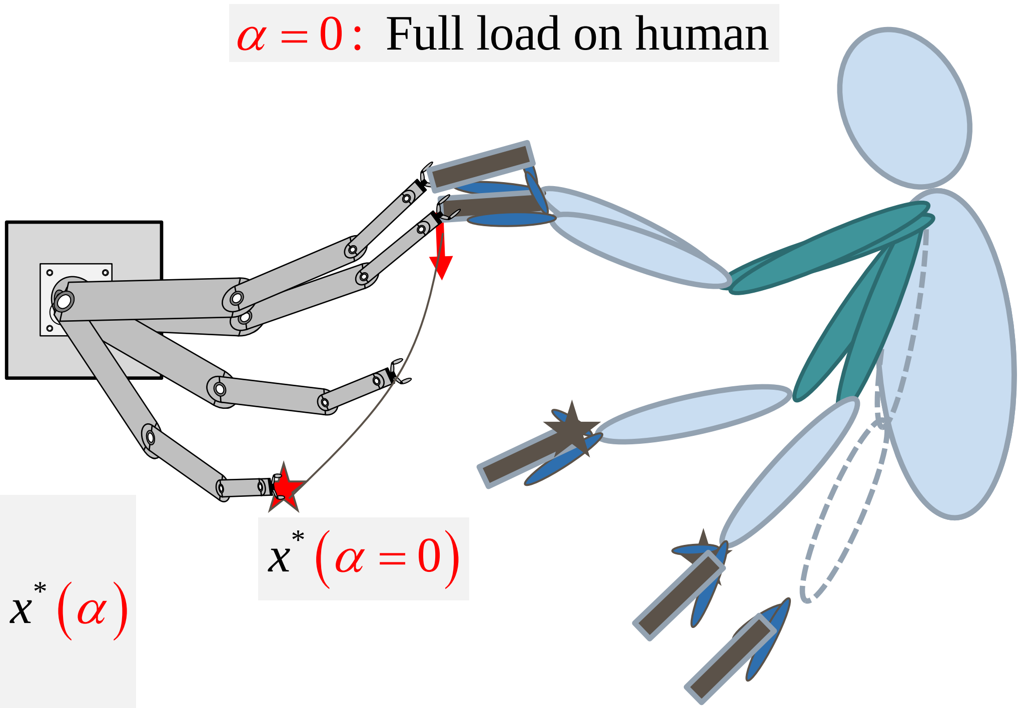


$\alpha:$ Load Share

Control law for robot motion

$$\dot{x}^R = f\left(x^R, g(\alpha)\right), \quad \lim_{t\to\infty} x^R = x^*(\alpha)$$

Origin on the intercept point



α : Load Share

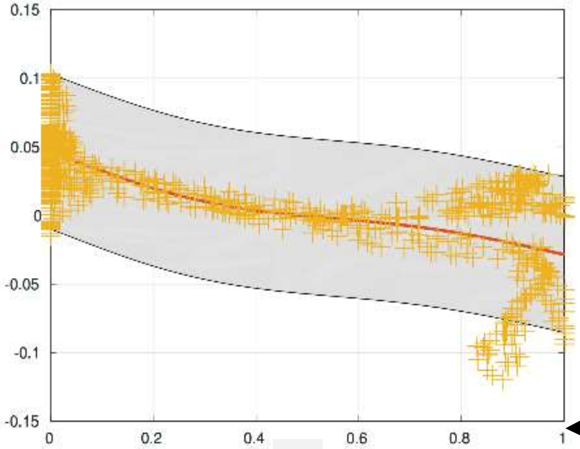
Control law for robot motion

$$\dot{x}^R = f\left(x^R, g(\alpha)\right), \quad \lim_{t\to\infty} x^R = x^*(\alpha)$$

Origin on the intercept point

Learning and adaptive control for robots

$g(\alpha)$



$\alpha \downarrow$

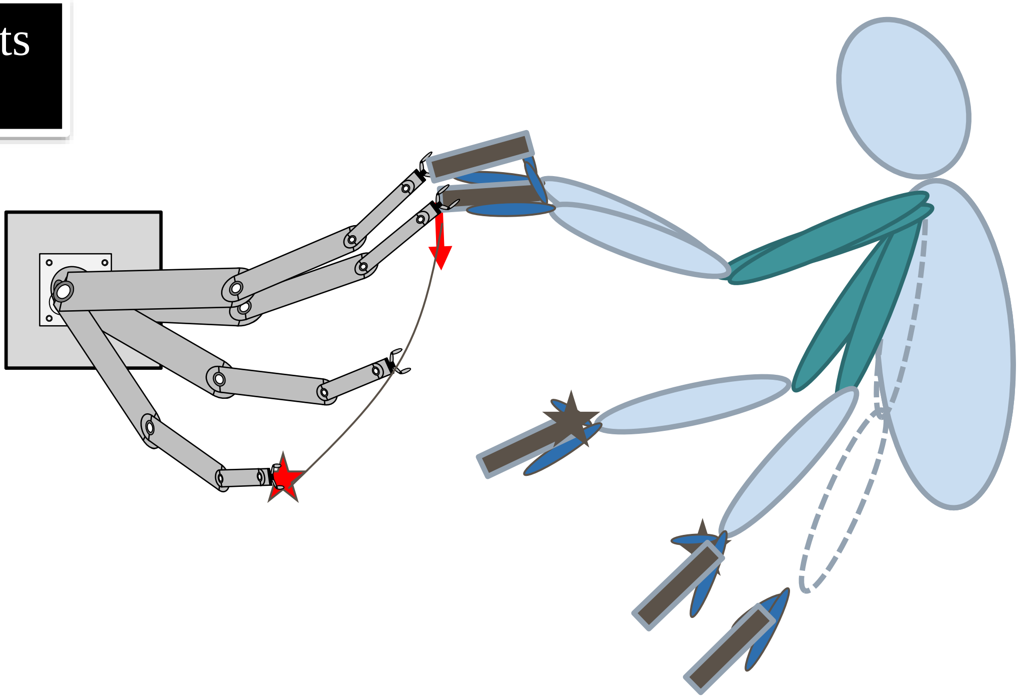
α

Receiver supports object

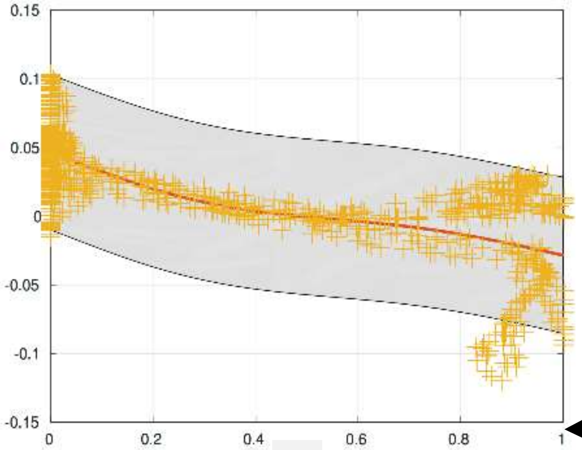
Giver supports object

Modeling $g(\alpha)$ from human data

$\alpha:$ Load Share



Learning and adaptive control for robots



$g(\alpha)$

$\alpha \downarrow$

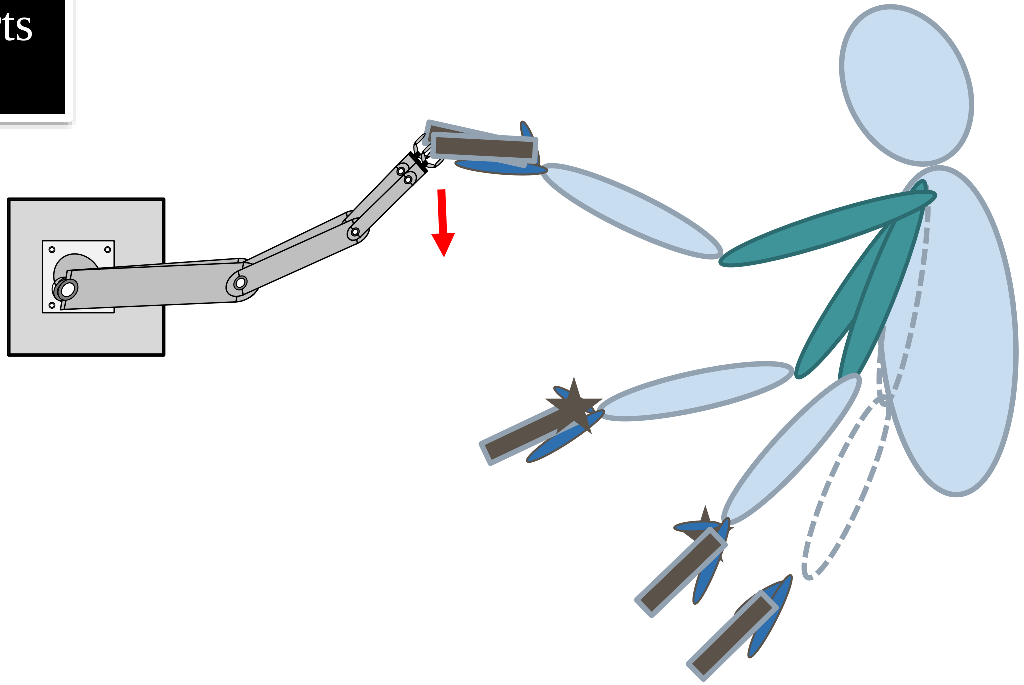
α

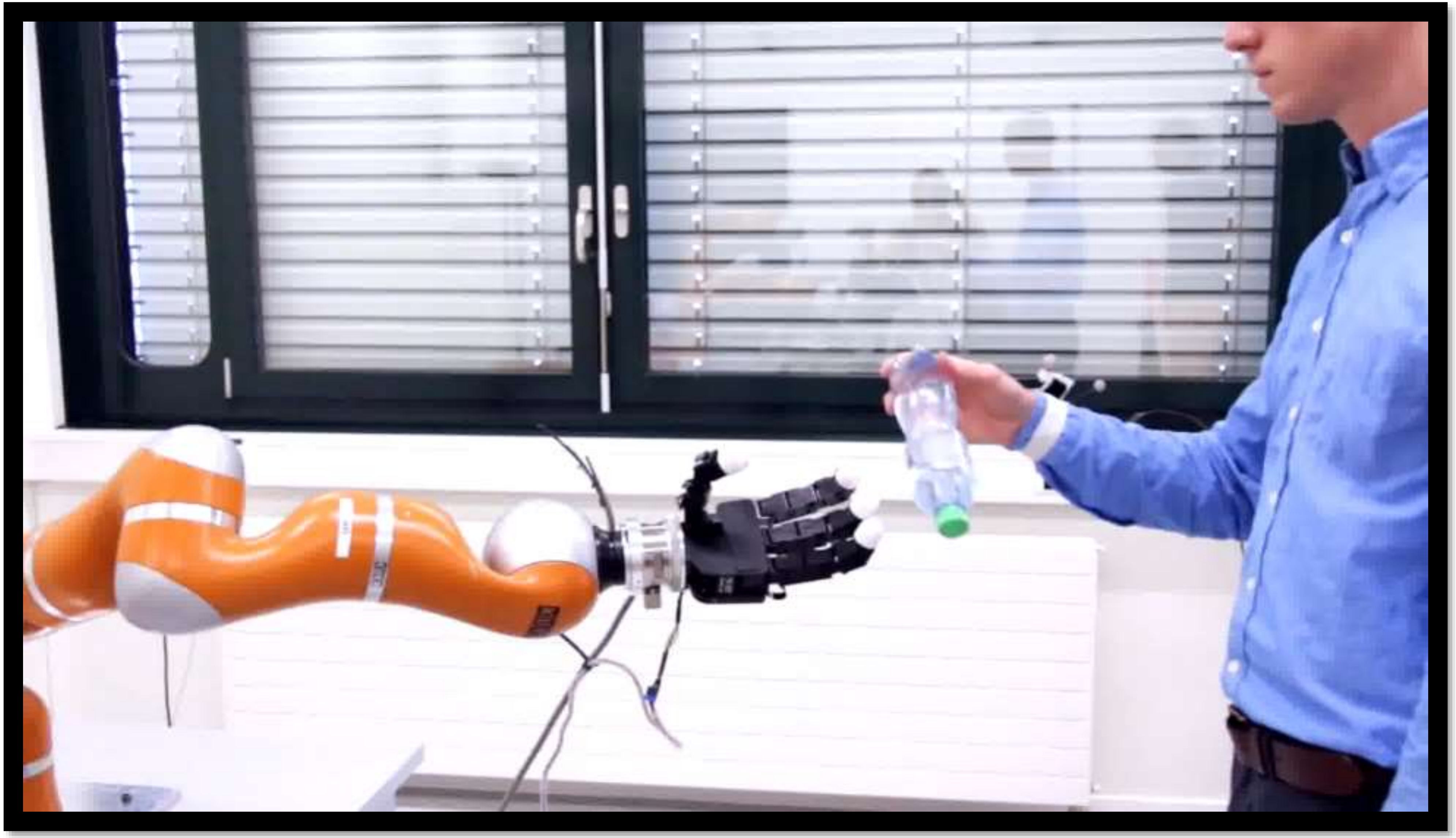
Receiver supports object

Giver supports object

Modeling $g(\alpha)$ from human data

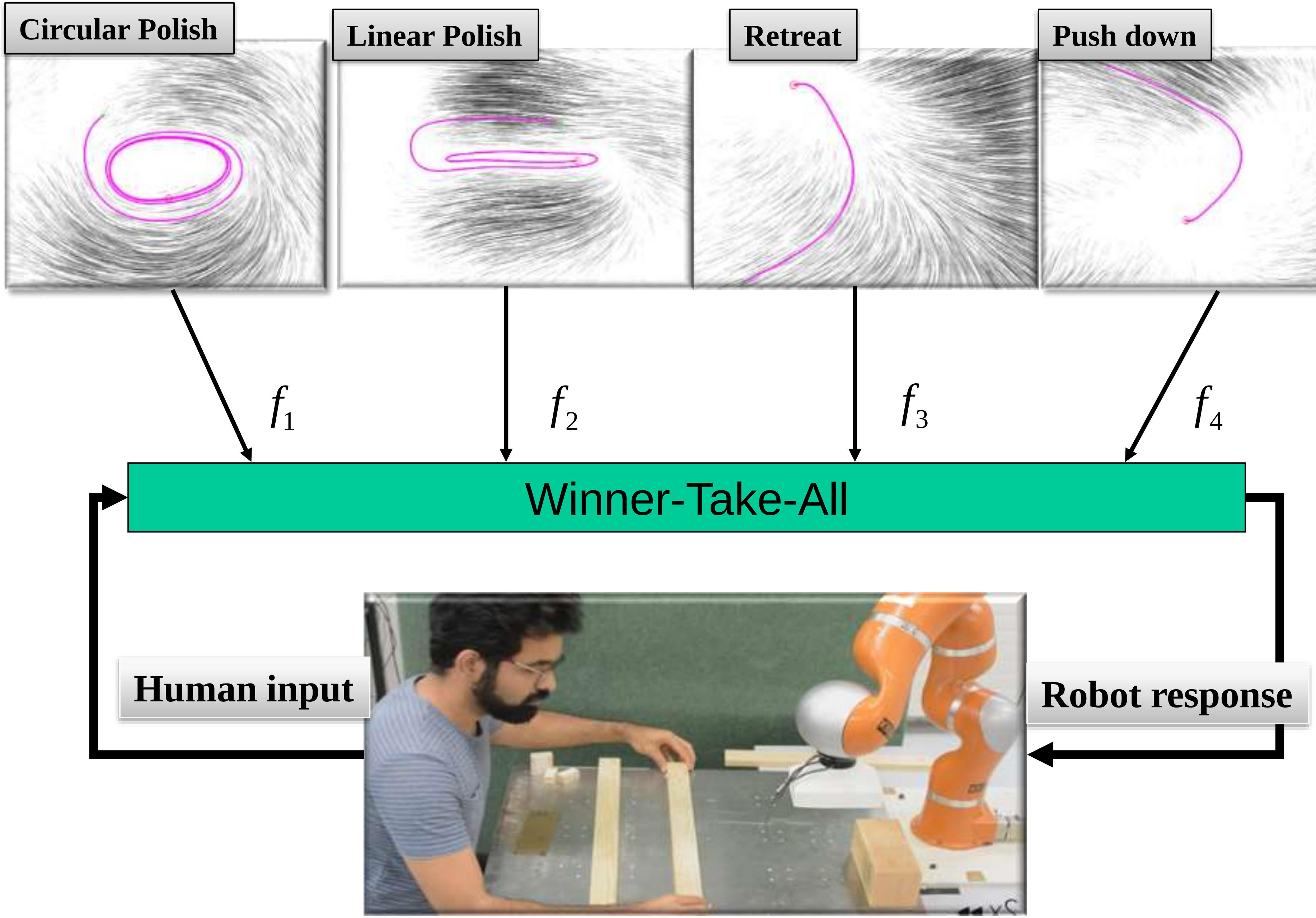
α : Load Share

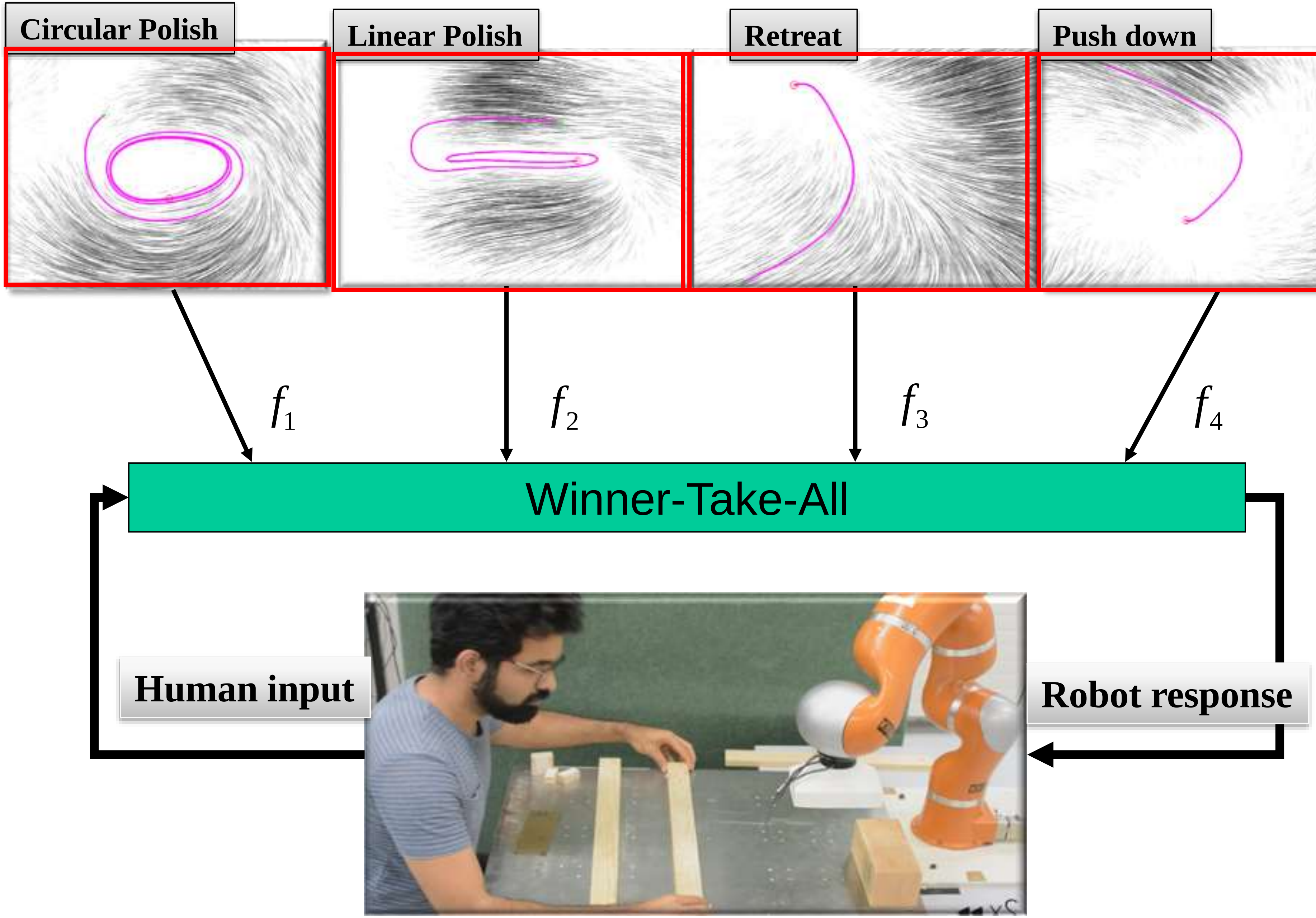




Medina, Duvallet, Karnam, Billard, A human-inspired controller for fluid human-robot handover, IEEE Conf. on Humanoid Robots, Humanoids 2016

Switching across DS


Circular Polish
Linear Polish
Retreat
Push down
f_1
f_2
f_3
f_4
Winner-Take-All
Human input
Robot response


Circular Polish
Linear Polish
Retreat
Push down
f_1
f_2
f_3
f_4
Winner-Take-All
Human input
Robot response

The human intends to place the box on either the right or left side.
The exact location for the placement is unknown to the robot.

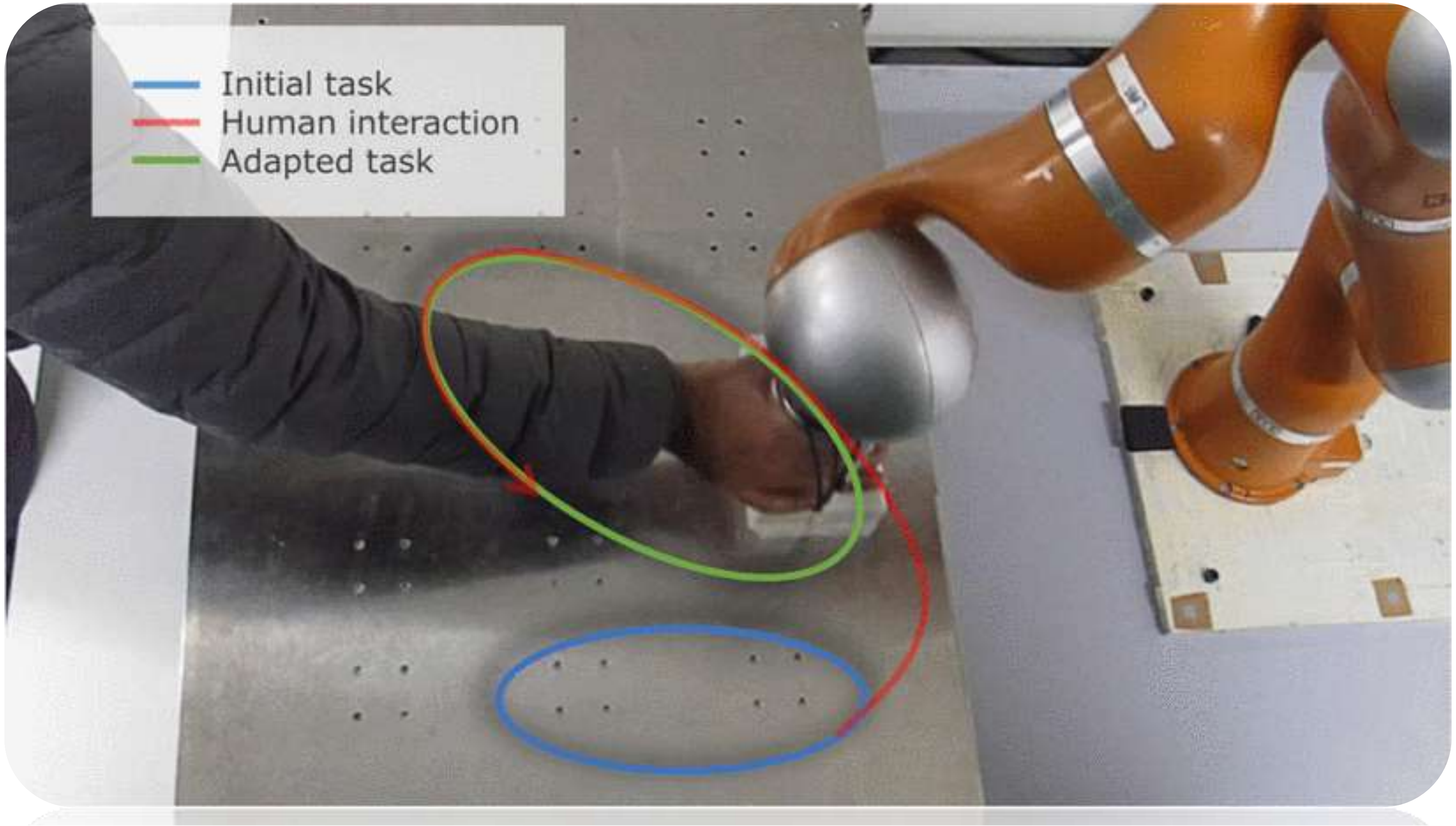
Robot's view
with four possible tasks:

Adapted Task:

- Move forward
- Move backward
- Place right
- Place left

Khoramshahi, M. and Billard, A., 2019. A dynamical system approach to task-adaptation in physical human–robot interaction. *Autonomous Robots*, *43*(4), pp.927-946.

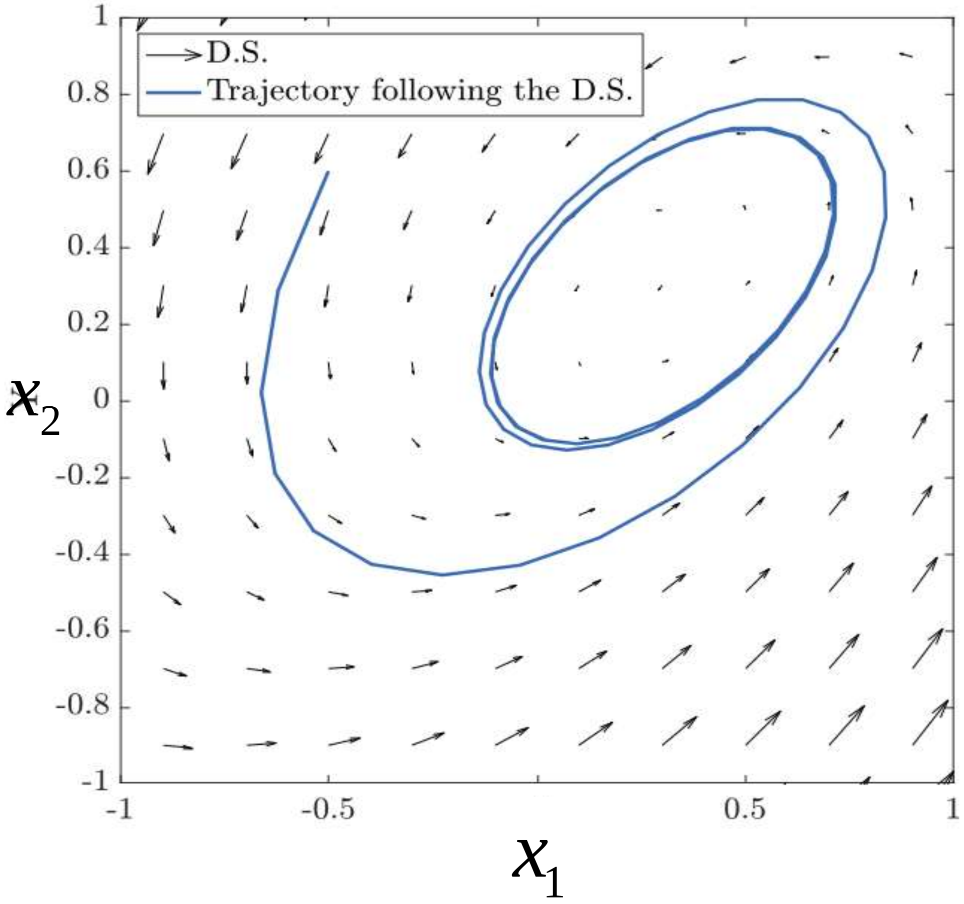
Adapting DS Parameters on-line



Automatically adapt center and size of the limit cycle, based on human demonstration

Control law for robot motion

$\dot{x} = f(x; \theta), \quad \theta:$ parameters to be adapted



Limit cycle in polar coordinates:

$\dot{r} = -\alpha(r - r^*), \quad \alpha$: radial velocity

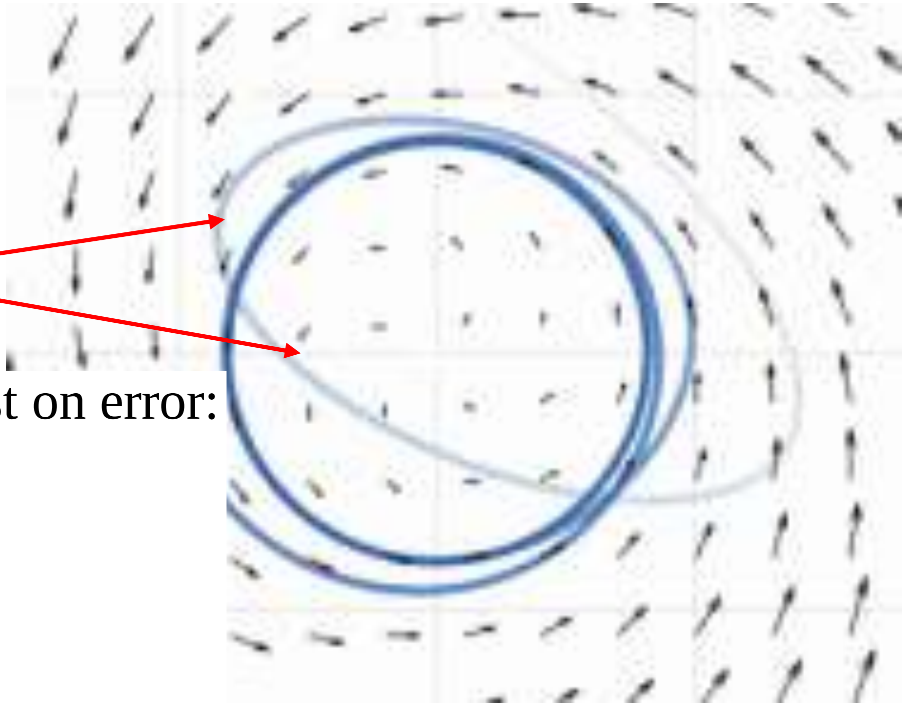
$\dot{\phi} = \omega, \quad \omega:$ angular velocity

$\alpha, \omega:$ parameters to be adapted

Estimate parameters on-line as soon as a discrepancy is recorded

Error measure over K points measured every Δt:

$$e = \frac{1}{K}\sum_{k=0}^{K}\Big(f\big(x(t-k\Delta t);\theta\big) - \dot{x}(t-k\Delta t)\Big)$$



Departs from original vector field

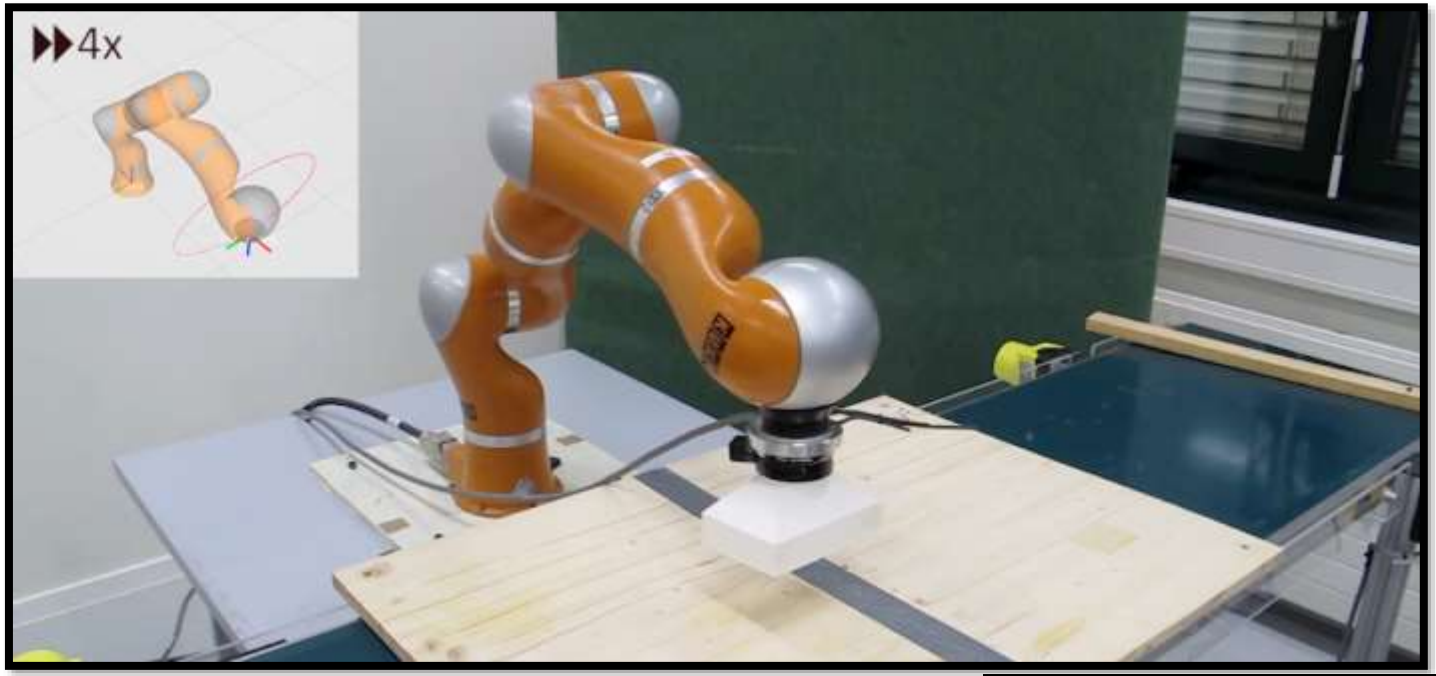
Update the parameters following gradient of the quadratic cost on error:

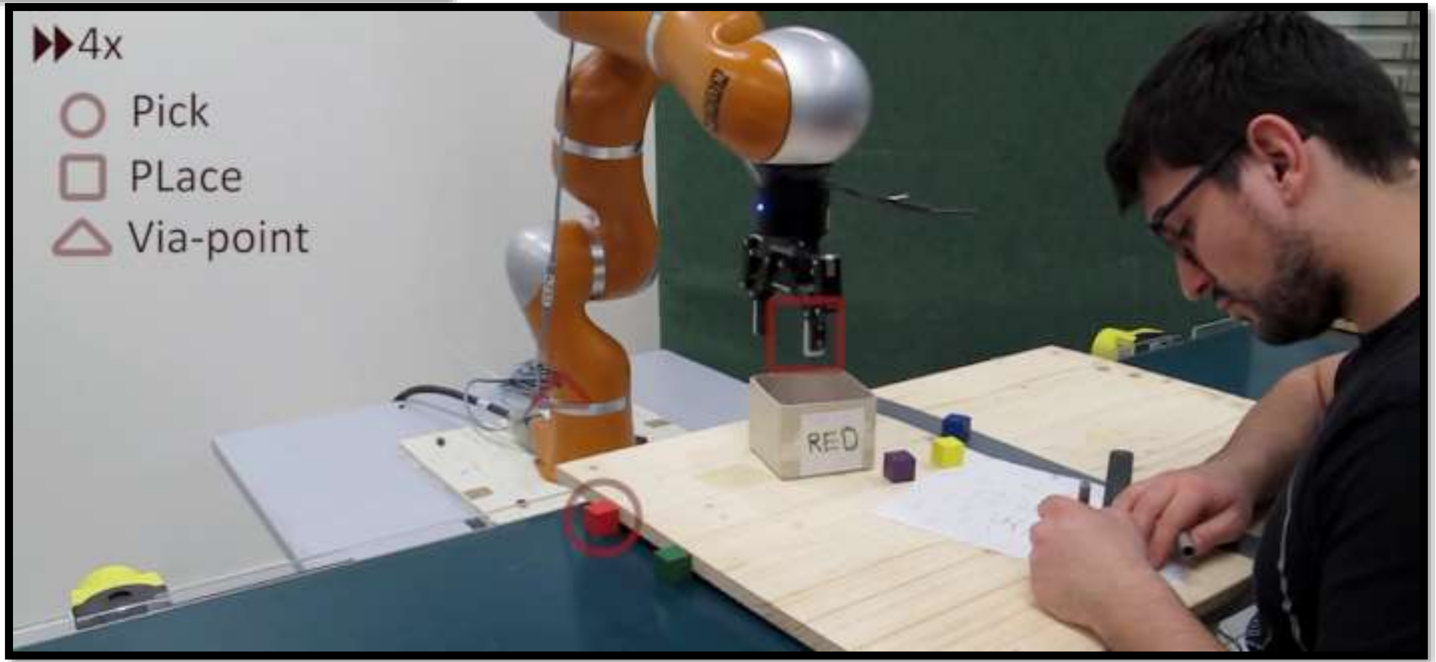
Quadratic cost: $J(\theta) = e(\theta)e^T(\theta)$

$$\frac{\partial J}{\partial \theta_i} = \frac{1}{K} e^T(\theta) \sum_{k=0}^{K-1} \frac{\partial f\big(x(t-k\Delta t);\theta\big)}{\partial \theta_i};$$ for each parameter θ_i

One can follow the same procedure to update the parameters of a linear DS: $\dot{x} = A(x - x^*)$.

Parameters to be learned are all elements of A and of x^*.





Khoramshahi, M., Laurens, A., Triquet, T. and Billard, A., 2018, October. From human physical interaction to online motion adaptation using parameterized dynamical systems. In *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)* (pp. 1361-1366).

Summary

- Trajectory generation with DS is a powerful and versatile technique, allowing for the generation of a variety of behavior to control one or multiple robots.
    - They can be parameterized to introduce external dependencies such as dependency on force or load, or to another dynamics.
- Coupling DS allows to generate temporal and spatial dependencies between different dynamics.
    - This can be applied to control different robots (or robot's parts) in synchrony.
    - When combined with the concept of virtual dynamics, it can force the system to track a desired path in space.
- We have seen other learning methods to estimate the DS
    - One can learn multi attractor systems through a partitioning of the space
    - One can estimate the parameters of a known DS (limit cycle, linear DS) online through simple gradient descent.