Teaching motor skills from humans to humanoids

Aude G. Billard
Dan H. Grollman

LASA
EPFL, Swiss Federal Institute of Technology
Lausanne, Switzerland

{aude.billard,daniel.grollman}@epfl.ch
http://lasa.epfl.ch

"Humanoids: What's next?"
Dec 7th, 2010, Nashville TN.

Programming by Demonstration

Imitation Learning / Learning from Demonstration

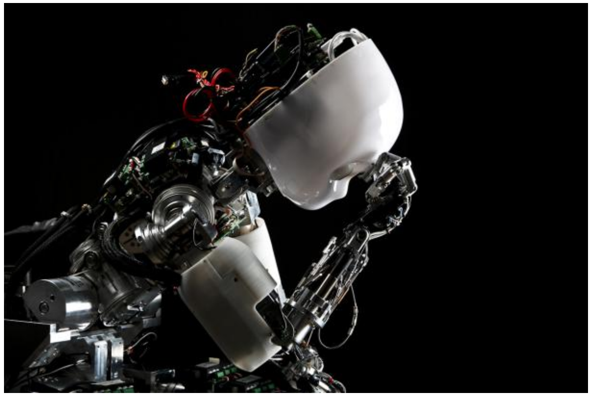
Claimed to be a "natural" means of teaching robots.

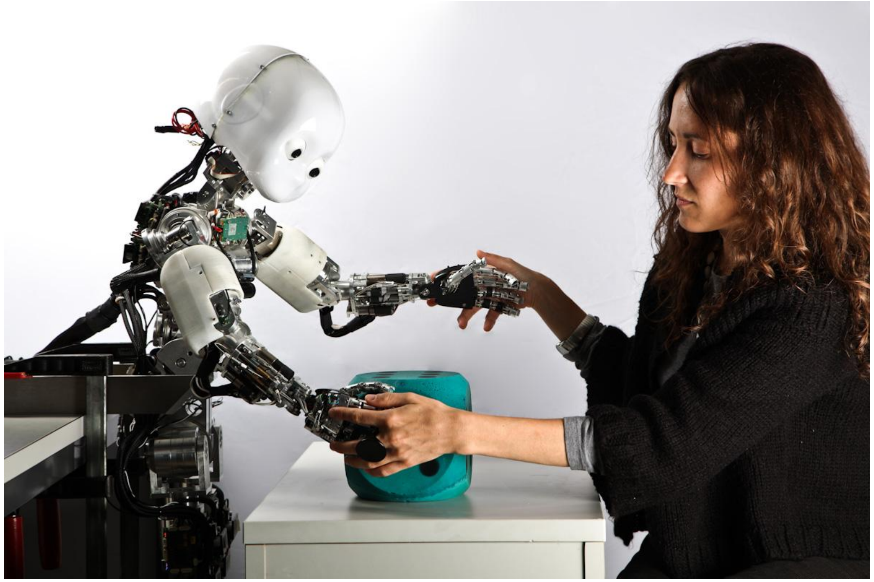
Natural: Inspired by how humans educate each other

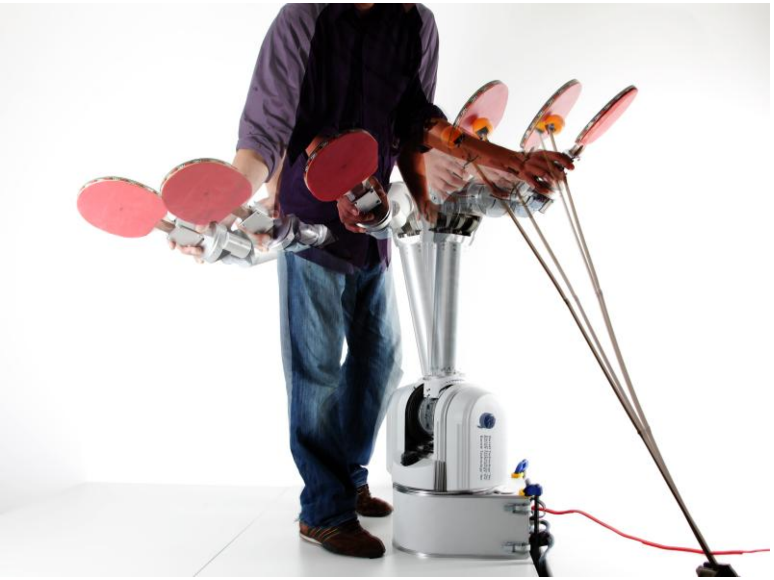
Humanoid robot: Interact (learn) as a human does

Teaching Robots to Do Tasks that Humans Do
Machine Learning, Control

- Teaching skills as humans do
- Teaching by showing the task







Kinesthetic Teaching:
Guiding the robot through the motion
Applicable to any type of robotic systems

Topics covered in this presentation

On the relative importance of time:
Time-independent vs. time-dependent encoding
(Contributors: M. Khansari, E. Gribovskaya, S. Kim)

Learning from multiple modalities:
Vision, touch, proprioceptive information
(Contributors: B. Argall, E. Sauser)

Learning from bad examples
(Contributors: D. Grollman)

Humanoids: What's next?

Simulation

Vision / Speech / Perception

Mechanisms

Walking - Perturbation

Movement representation: Time dependent or not?

Time dependency

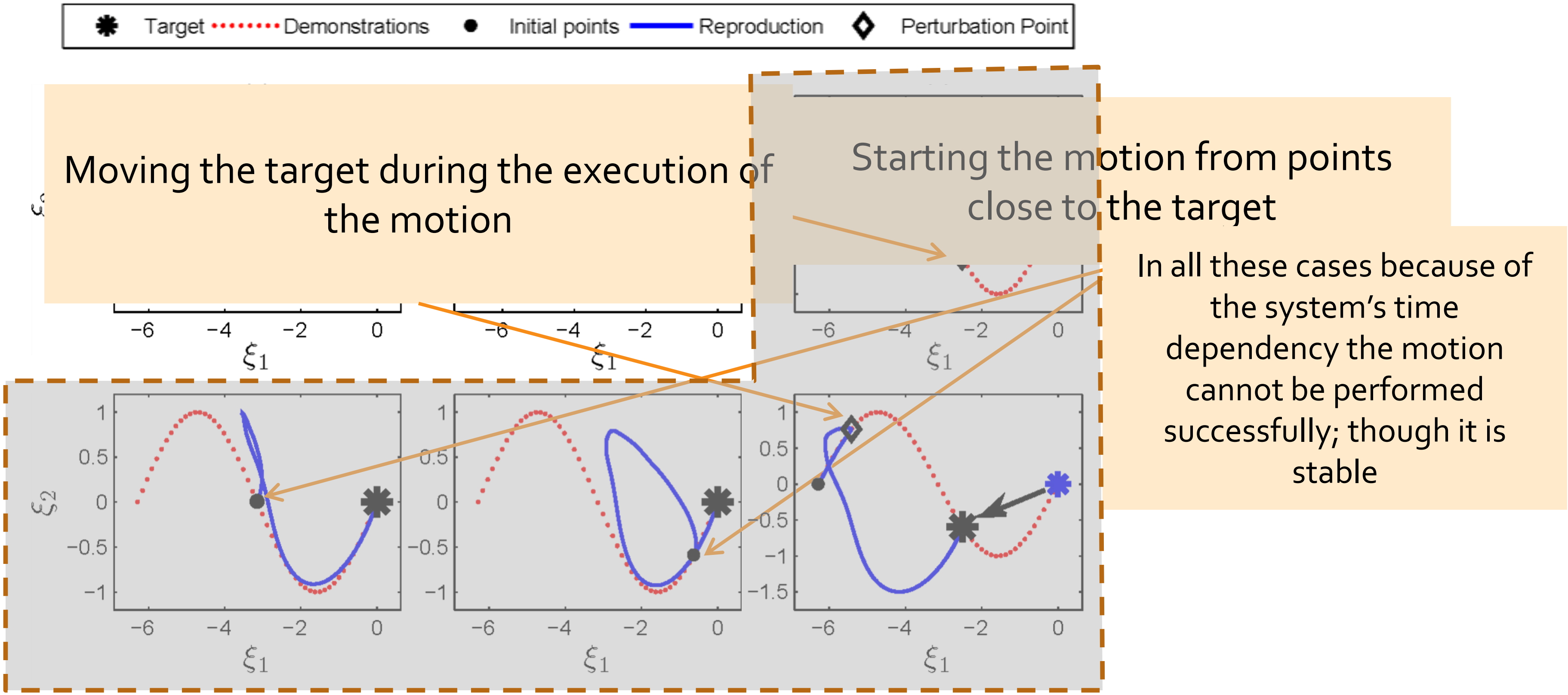
- Time-dependent trajectory encodings
  - splines, planners, HMM, GMM, etc
  - Open-loop
    - track deviations and heuristically realign after perturbation

Time-Dependent

Sensitivity of *time-dependent* systems to *external perturbations*:

A sine motion is learnt using *Dynamic Movement Primitives (DMP)*

$$\tau \ddot{x} = -D\dot{x} + K(g-x) - K(g-x_0)s + K\hat{f}(s)$$



Time dependency

- Time-dependent trajectory encodings
  - splines, planners, HMM, GMM, etc
  - Open-loop
    - track deviations and heuristically realign after pertbation

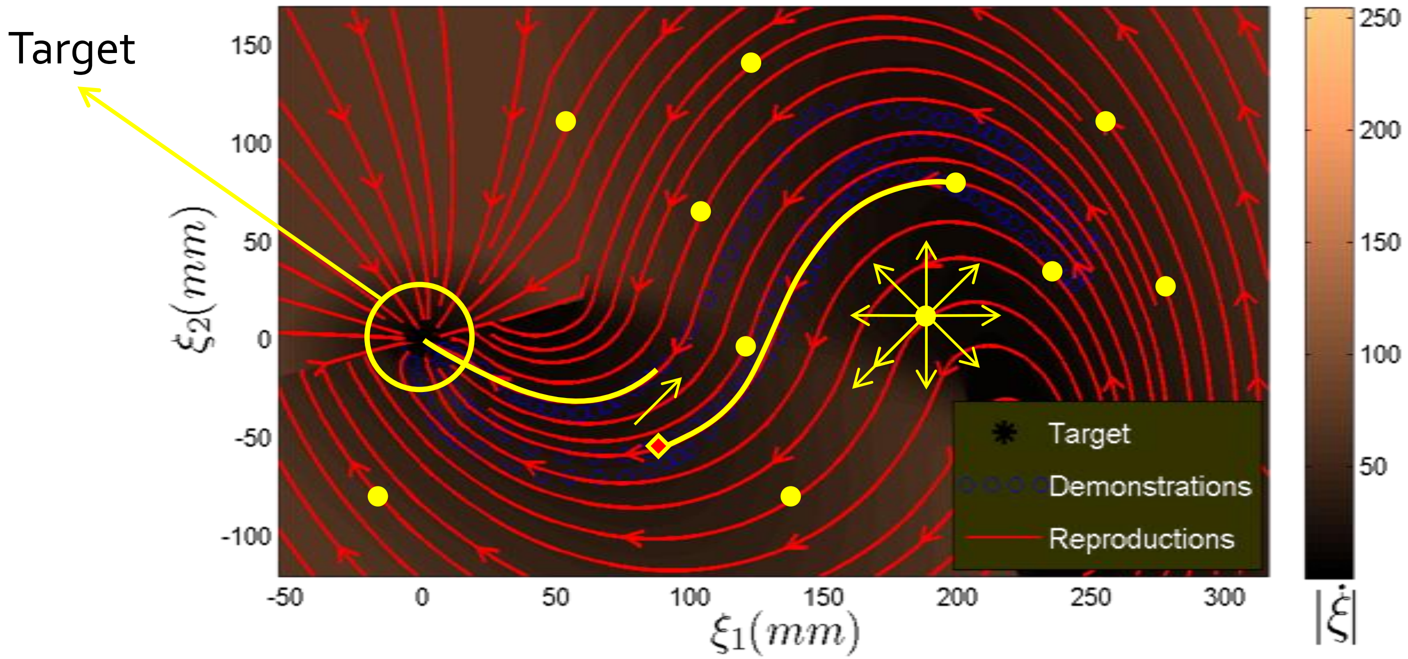
- Time-independent description
  - autonomous dynamical system
  - Closed-loop
    - Trajectories defined throughout state space
    - How to stabilize?

Learning to be robust to perturbations

Learning a single law of motion → Dynamical Systems are core to the way the human brain computes motion

Time-independent system

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



Learning to be robust to perturbations

Learning a single law of motion → Dynamical Systems are core to the way the human brain computes motion

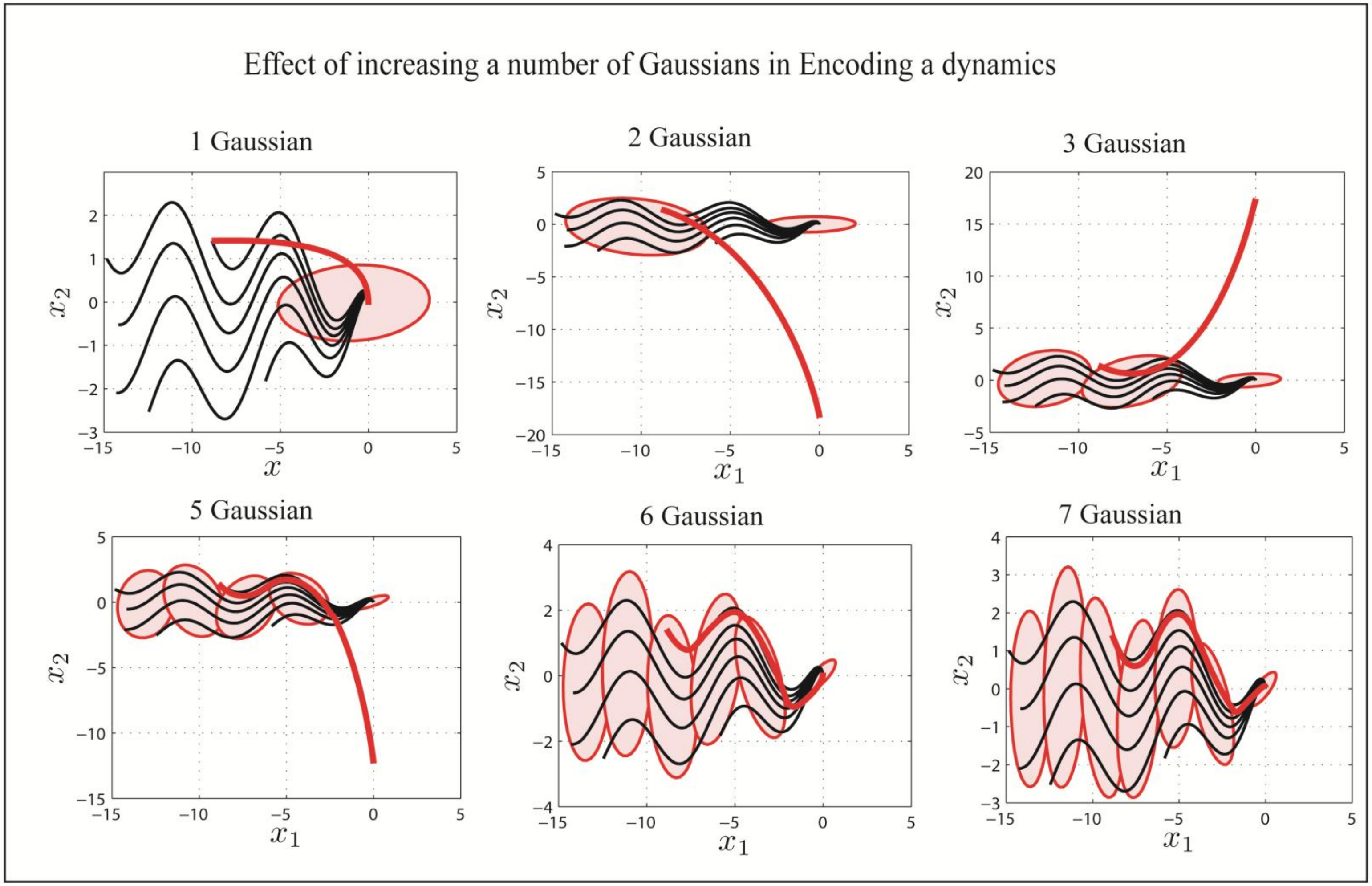
Time-independent system

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

Build an estimate through non-linear mixture of linear systems through mixture of Gaussians

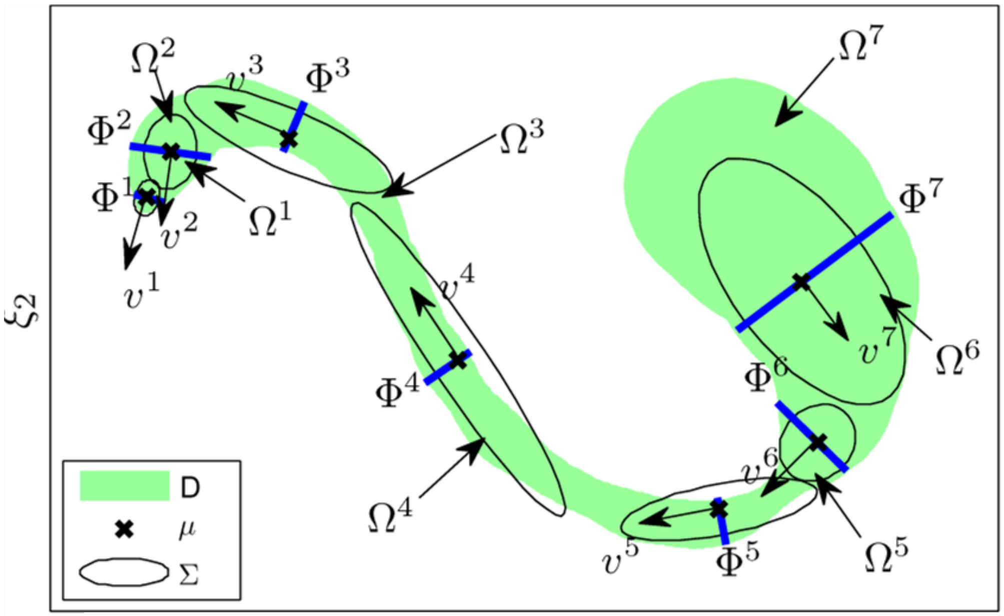
$$\hat{\dot{\xi}} = \sum_{k=1}^{K} \underbrace{\frac{\mathcal{N}(\xi;\theta^k)}{\sum_{i=1}^{K}\mathcal{N}(\xi;\theta^i)}}_{h^k(\xi)} \left(\underbrace{\Sigma^k_{\dot{\xi}\xi}(\Sigma^k_{\xi})^{-1}}_{A^k}\xi + \underbrace{(\dot{\mu}^k_{\dot{\xi}} - \Sigma^k_{\dot{\xi}\xi}(\Sigma^k_{\xi})^{-1}\mu^k_{\xi})}_{b^k} \right)$$

$$\hat{\dot{\xi}} = \sum_{k=1}^{K} h_k(\xi)\left(A_k\xi + b_k\right)$$



Increase # of Gaussians from 1 – check stability
GMM fit with EM – Optimizes likelihood, not stability

Local Stability



Determine conditions for ensuring asympt. Stability – set open parameters of GMM

Incremental algorithm, optimization under constraint

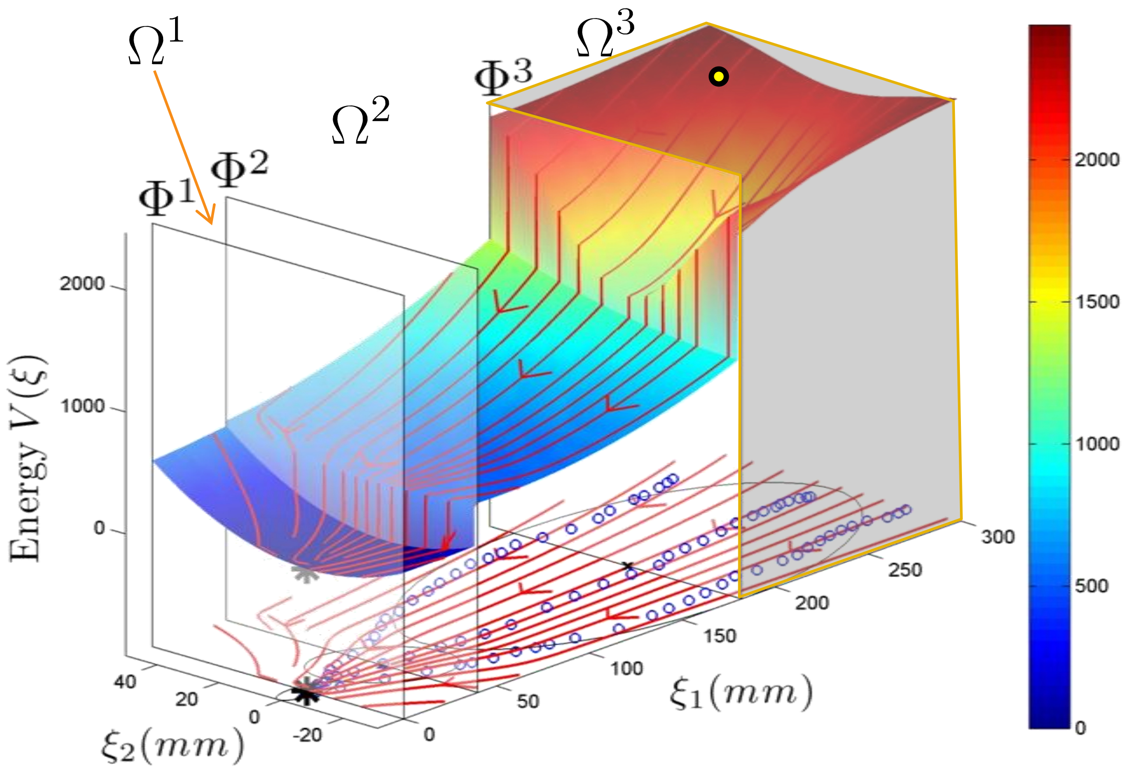
Stability in a given region

$$\begin{cases} \textbf{(a)} \begin{cases} (\xi - \mu_\xi^{k+1})^T(\Sigma_\xi^{k+1})^{-1}\hat{\dot{\xi}} > (\xi - \mu_\xi^{k})^T(\Sigma_\xi^{k})^{-1}\hat{\dot{\xi}} & \forall \xi \in \Omega^k \;\; \& \;\; \xi \neq 0 \;\; \& \;\; \forall k \in 1..K-1 \\ (\xi - \mu_\xi^{K})^T(\Sigma_\xi^{K})^{-1}\hat{\dot{\xi}} < 0 & \forall \xi \in \Omega^K \end{cases} \\ \textbf{(b)} \; (v^k)^T \hat{\dot{\xi}} > 0 \qquad \forall \xi \in \Phi^k \;\; \& \;\; \xi \neq 0 \;\; \& \;\; \forall k \in 1..K \\ \textbf{(c)} \; b^* = h^1(0)b^1 + h^2(0)b^2 = 0 \qquad \xi = 0 \in \Omega^1 \end{cases}$$

Local Stability Analysis of DSs

First stability condition ensures one region funnels into the next

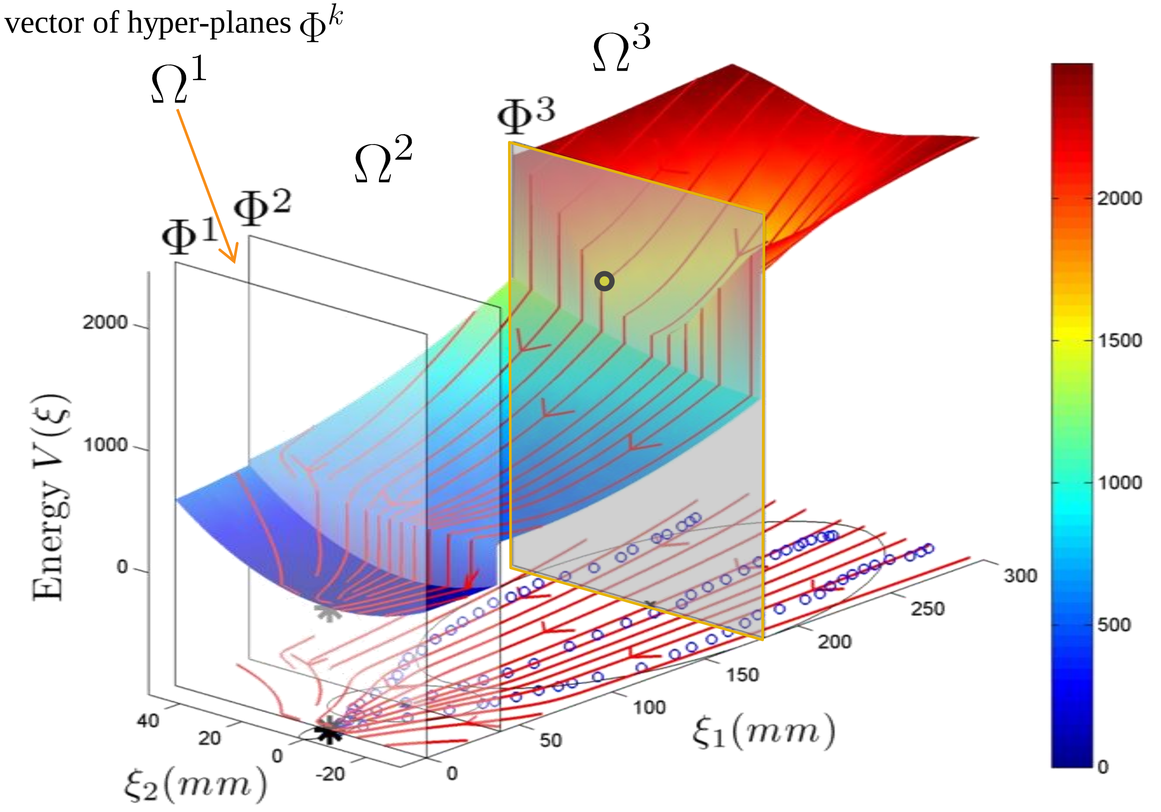
$$(1) \quad (\xi - \mu_\xi^{k+1})^T (\Sigma_\xi^{k+1})^{-1} \hat{\dot{\xi}} > (\xi - \mu_\xi^k)^T (\Sigma_\xi^k)^{-1} \hat{\dot{\xi}} \qquad \forall \xi \in \Omega^k$$



Local Stability Analysis of DSs

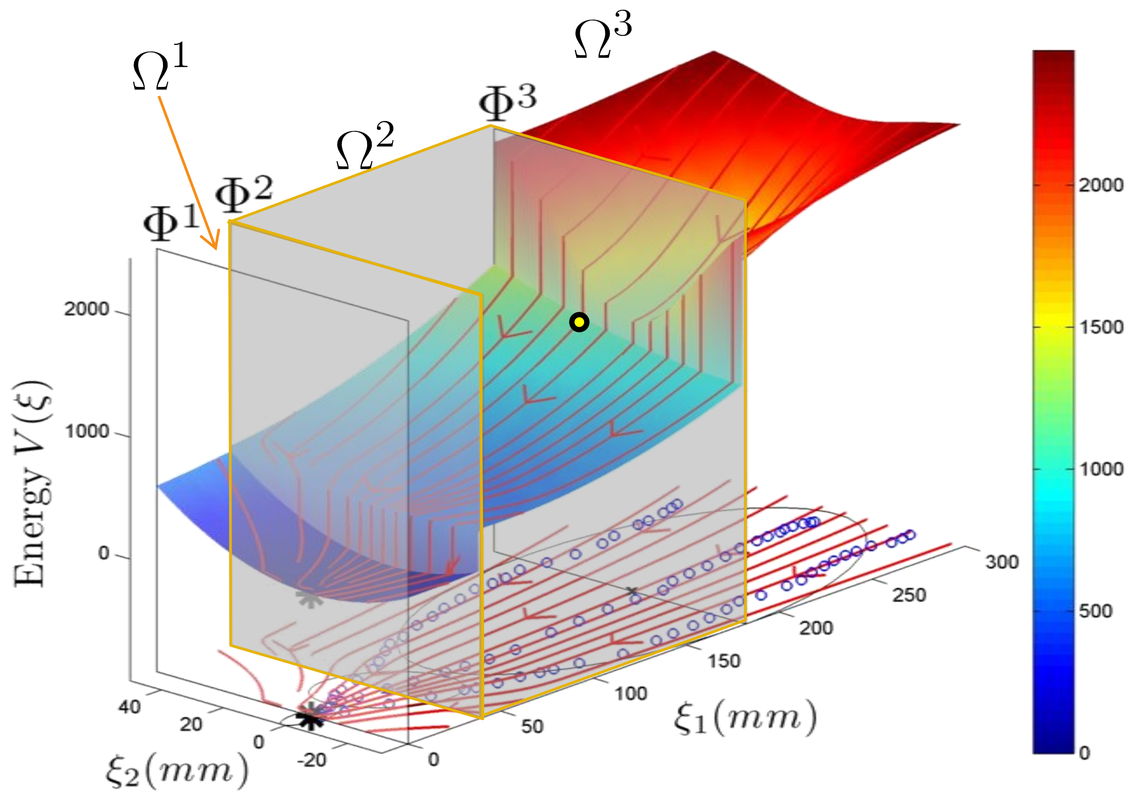
Second stability condition ensures the correct direction of the transition of the flow at hyper-planes **(2)** $(\upsilon^k)^T\hat{\dot{\xi}} > 0 \quad \forall \xi \in \Phi^k \;\&\; \xi \neq 0 \;\&\; \forall k \in 1..K$

* υ^k are the normal vector of hyper-planes Φ^k



Local Stability Analysis of DSs

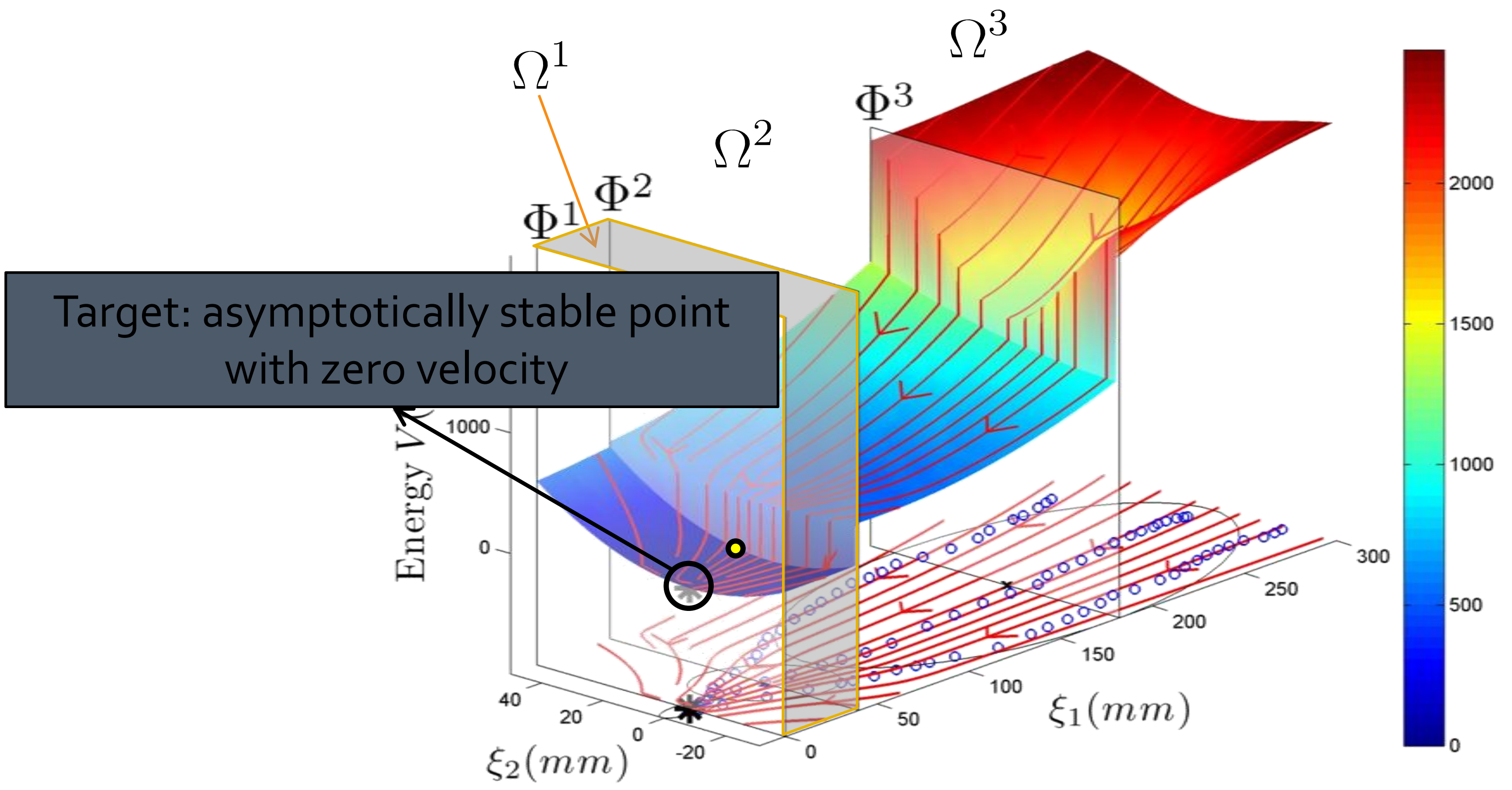
➤ Again first and second stability conditions should be checked $\forall \xi \in \Omega^k \;\&\; \xi \in \Phi^k$ until the motion reaches the last domain.



Local Stability Analysis of DSs

Third stability condition ensures that origin $\xi = 0$ is the equilibrium point of the system (has the minimum energy)

$$\textbf{(3)} \quad \sum_{k=1}^{2} \frac{\mathcal{N}(0;\pi^k,\mu^k,\Sigma^k)}{\sum_{i=1}^{2}\mathcal{N}(0;\pi^i,\mu^i,\Sigma^i)} \left(\mathcal{N}(0;\pi^k,\mu^k,\Sigma^k)(\dot{\mu}^k_{\dot{\xi}} - \Sigma^k_{\dot{\xi}\xi}(\Sigma^k_{\xi})^{-1}\mu^k_{\xi})\right) = 0$$

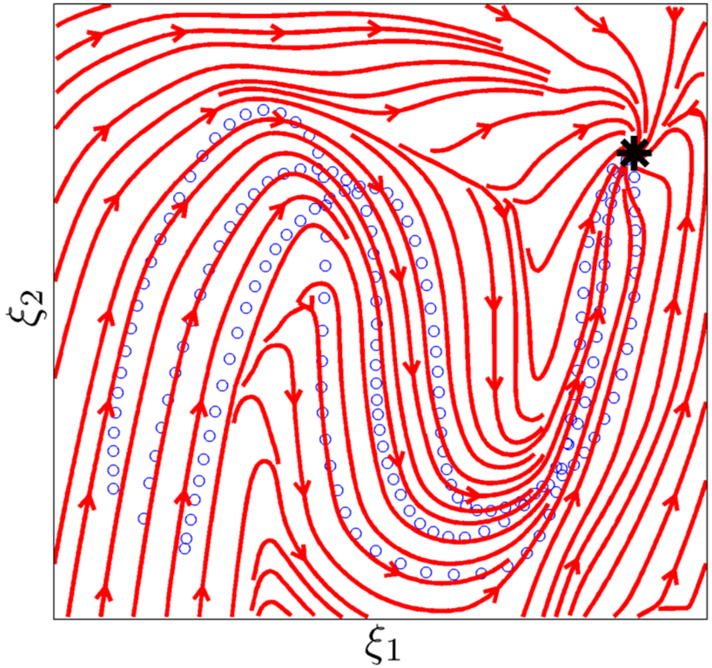


Globally stable estimate of the dynamics of motion

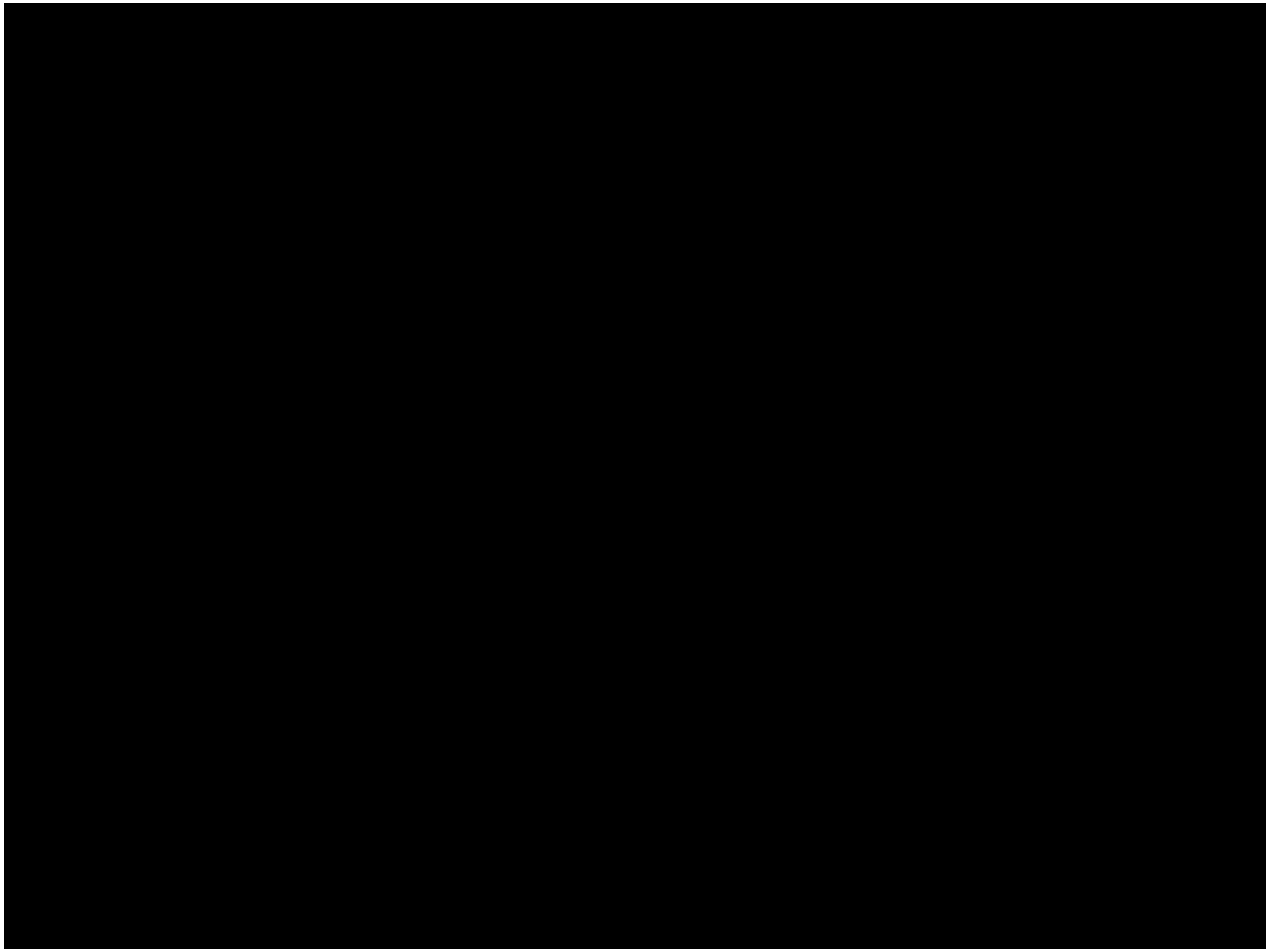
$$\min_{\boldsymbol{\theta}} J(\boldsymbol{\theta}) = \frac{1}{N}\sum_{n=1}^{N}\sum_{t=0}^{T^n}\Big((\hat{\xi}^{t,n}(\boldsymbol{\theta}) - \xi^{t,n})^2 + (\hat{\dot{\xi}}^{t,n}(\boldsymbol{\theta}) - \dot{\xi}^{t,n})^2\Big)$$

subject to

$$\begin{cases} \textbf{(a)}\ b^k = -A^k\xi^T \\ \textbf{(b)}\ \frac{1}{2}(A^k + (A^k)^T) \text{ are negative definite} \\ \textbf{(c)}\ \Sigma_{\dot{\xi}}^k \text{ are positive definite} \qquad \forall k \in 1..K \\ \textbf{(d)}\ 0 < \pi^k \leq 1 \\ \textbf{(e)}\ \sum_{k=1}^{K}\pi^k = 1 \end{cases}$$



M. Khansari and A. Billard, IROS 2010.



Time Independent -> No Timing Control

One cannot control explicitly the timing of a time-independent system

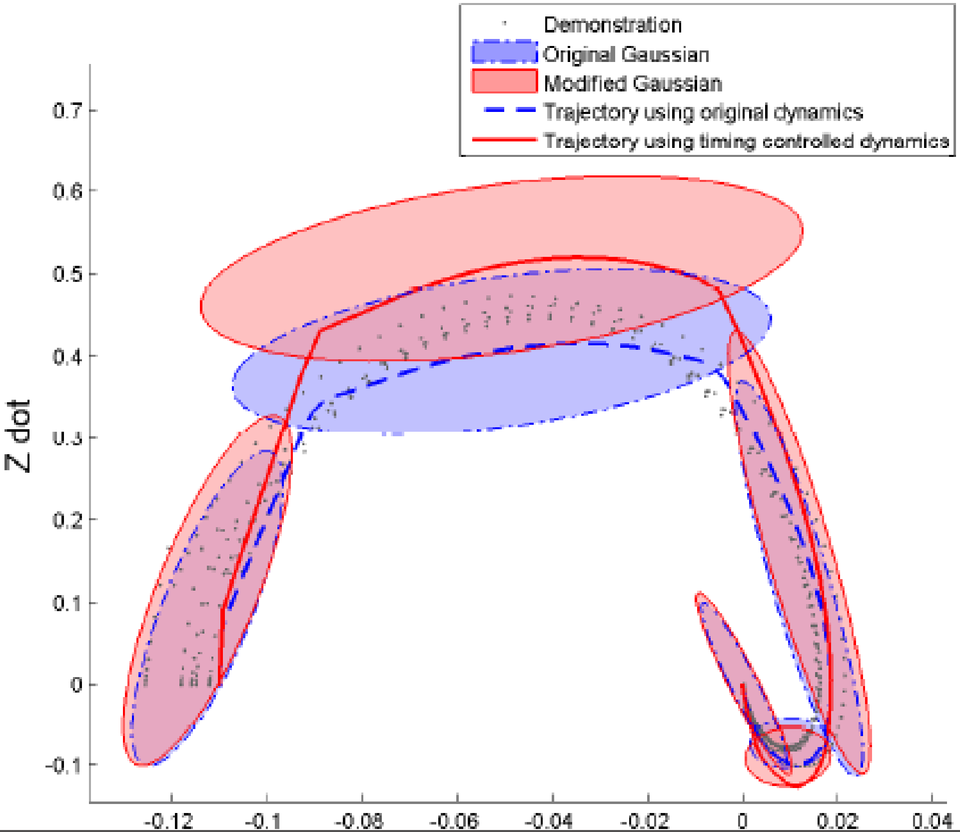
Keep the time-independency, but adapt the speed profile by moving with a constant factor the means and covariance of the model

$$\tilde{\mu}_k^{\dot{\xi}} = \lambda \mu_k^{\dot{\xi}}$$

$$\tilde{\Sigma}_k^{\dot{\xi}\xi} = \lambda \Sigma_k^{\dot{\xi}\xi}$$

New control law

$$\dot{\hat{\xi}} = \tilde{\hat{f}}(\xi) = \lambda \hat{f}(\xi)$$



Catching a flying object

Position trajectory generation by velocity integration :

$$\xi^{t_{j+1}} = \xi^{t_j} + \lambda^{t_i} \sum_{l=1}^{L} \dot{\hat{\xi}}^{\left\{t_j + \frac{\Delta t}{L} l\right\}} \frac{\Delta t}{L}$$

Timing Controller :

$$\lambda^{t_{i+1}} = \lambda^{t_i} + k_p\left(\hat{T}^{t_i} - T\right) - k_d\left(\hat{T}^{t_i} - \hat{T}^{t_{i-1}}\right)$$

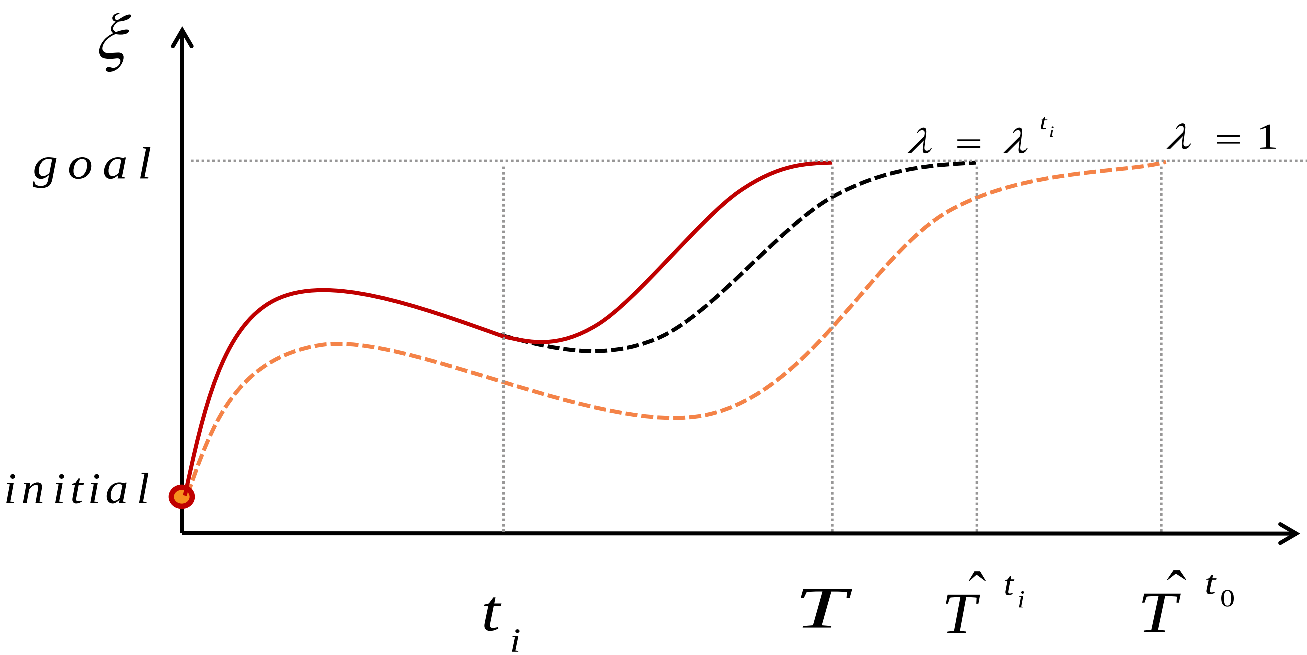
where t_i is a time at i^{th} controlling step,

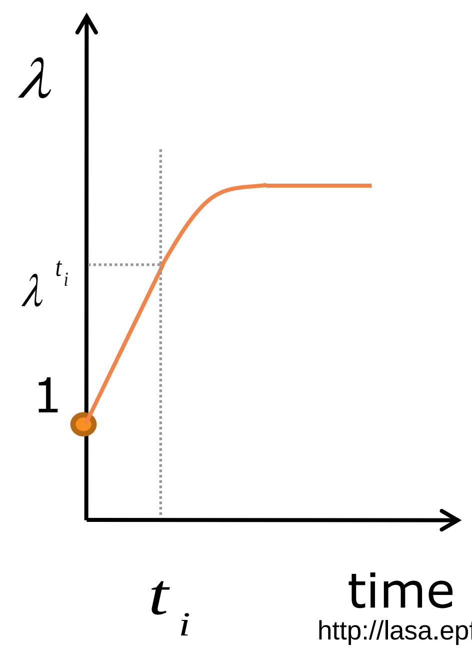
$t_{i+1} = t_i + \Delta t,\ t_0 = 0;$

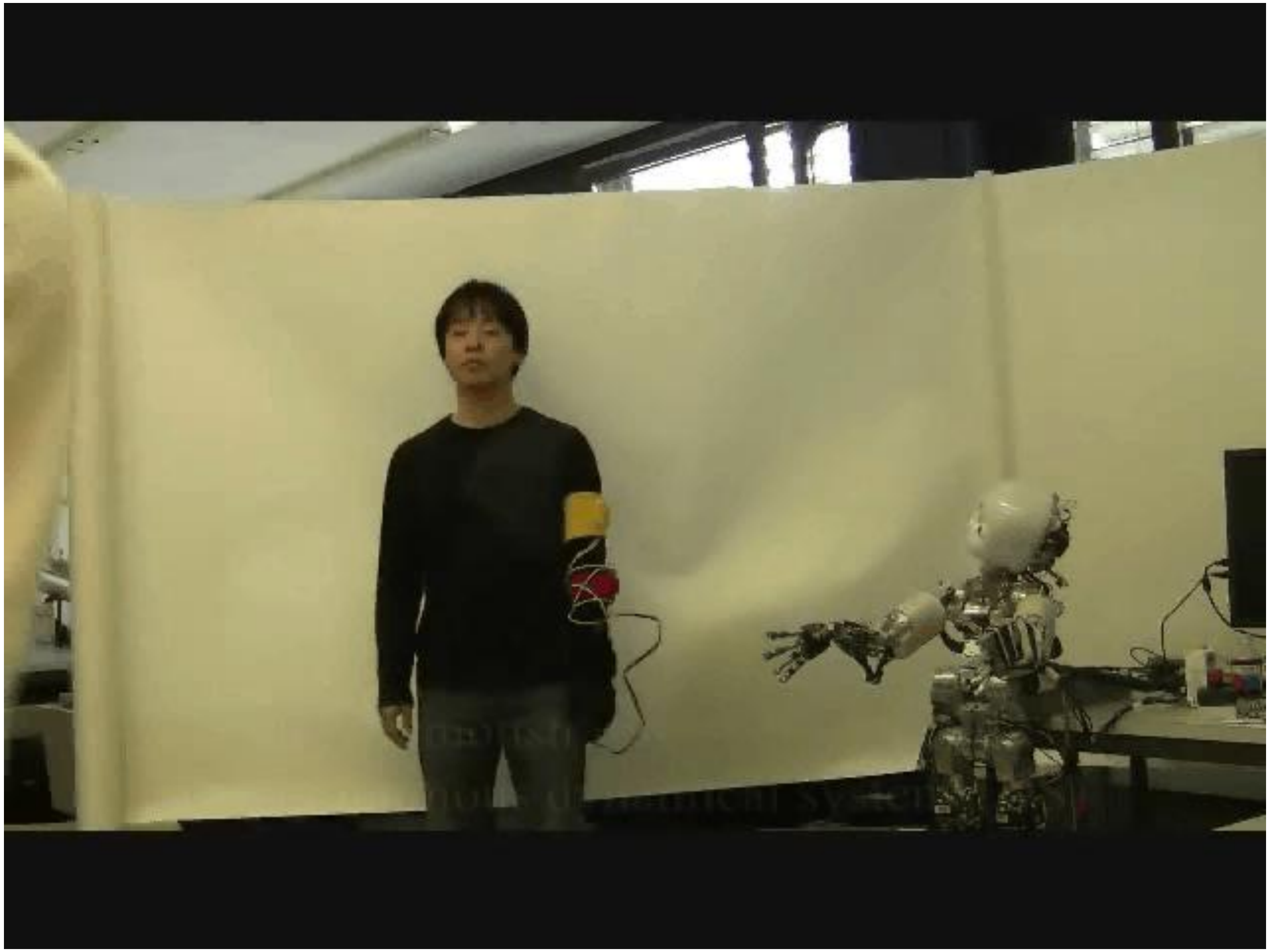
λ^{t_i} is a velocity multiplier, $\lambda^{t_0} = 1;$

k_p and k_d are the proportional and derivative gains respectively;

$\hat{T}^{t_i}$ is an estimated motion duration starting from the beginning of motion at time t_0 as calculated at time t_i







Topics covered in this presentation

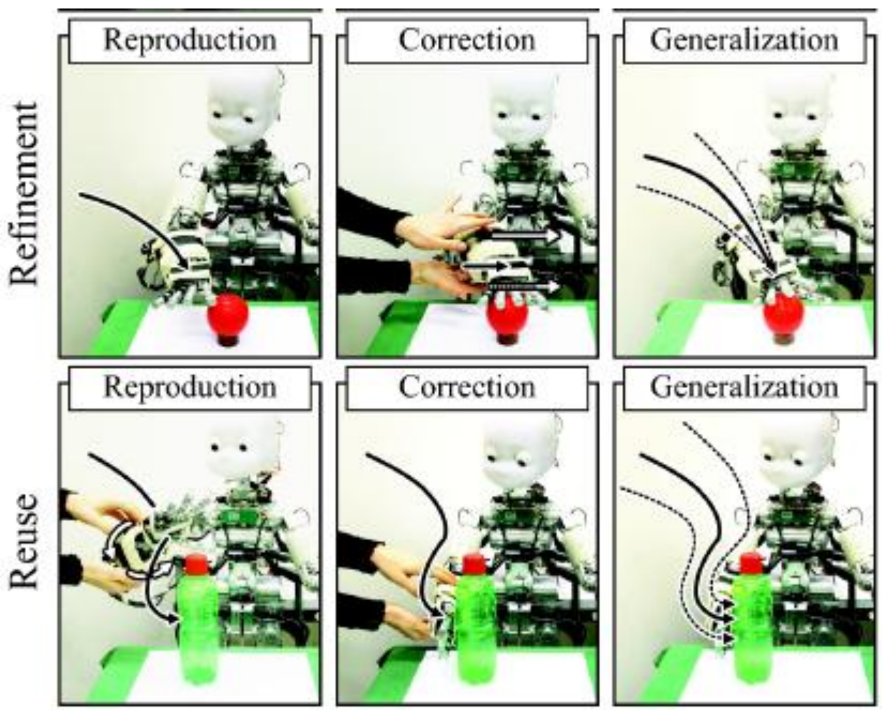
On the relative importance of time:
Time-independent vs. time-dependent encoding
(Contributors: M. Khansari, E. Gribovskaya, S. Kim)

Learning from multiple modalities:
Vision, touch, proprioceptive information
(Contributors: B. Argall, E. Sauser)

Learning from bad examples
(Contributors: D. Grollman)

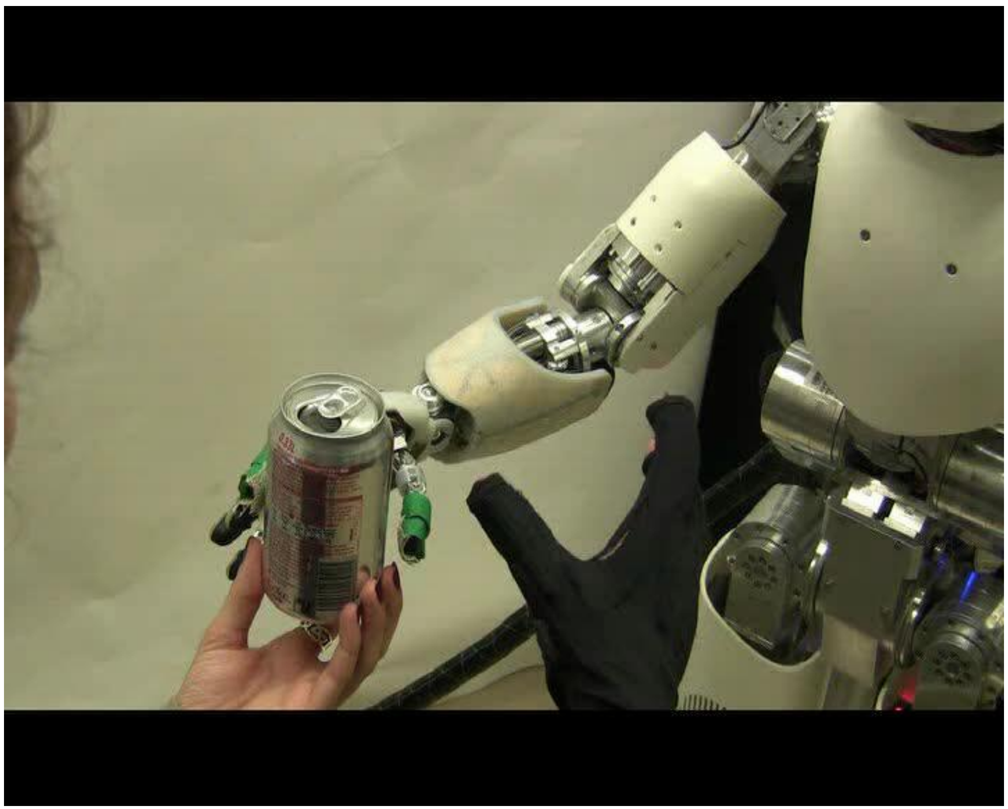
Interactive learning to reuse and refine tasks

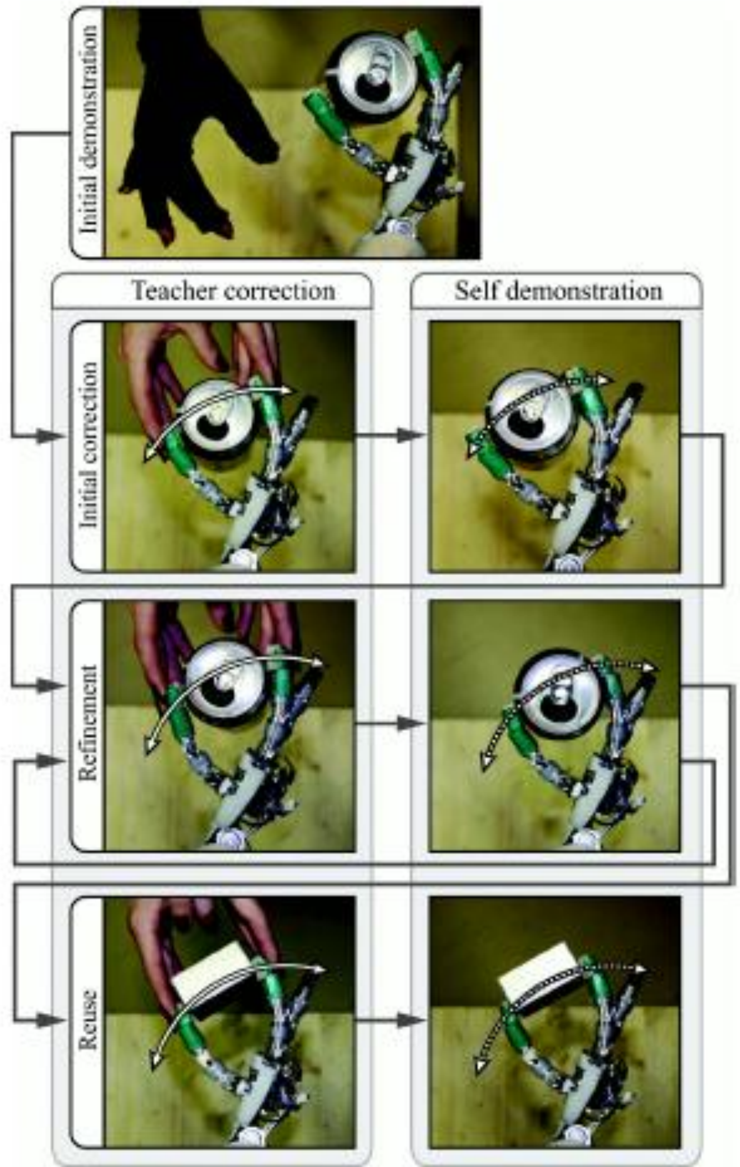
- ❑ Learning is incremental by nature
- ❑ Knowledge acquired in one task can be transmitted to another task



Teaching through tactile sensing

Learning fine manipulation tasks through tactile sensing at the finger tips.





Topics covered in this presentation

On the relative importance of time:

Time-independent vs. time-dependent encoding

(Contributors: M. Khansari, E. Gribovskaya, S. Kim)

Learning from multiple modalities:

Vision, touch, proprioceptive information

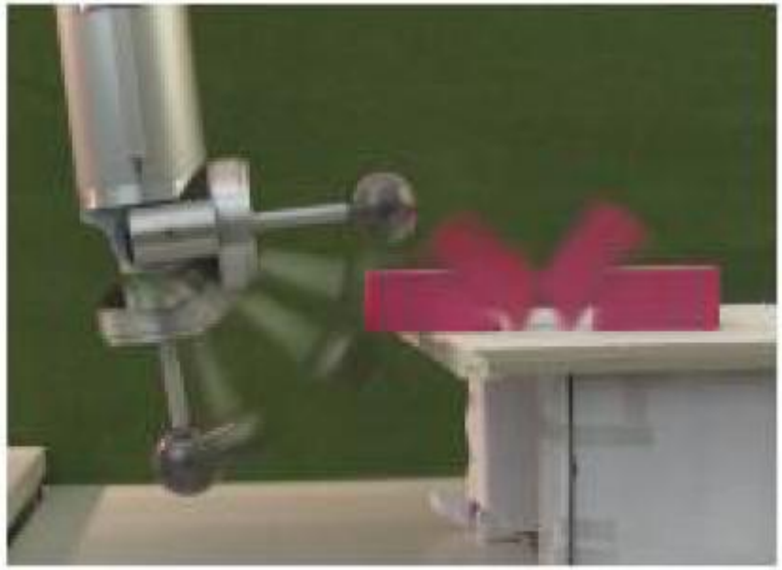
(Contributors: B. Argall, E. Sauser)

Learning from bad examples

(Contributors: D. Grollman)

Learning what Not to Do

Collect Failed Demonstrations



Consider only failed human demonstrations

(classical PbD approaches assume success)

Demonstrations = an example of what not to do.

Avoid repeating same mistakes

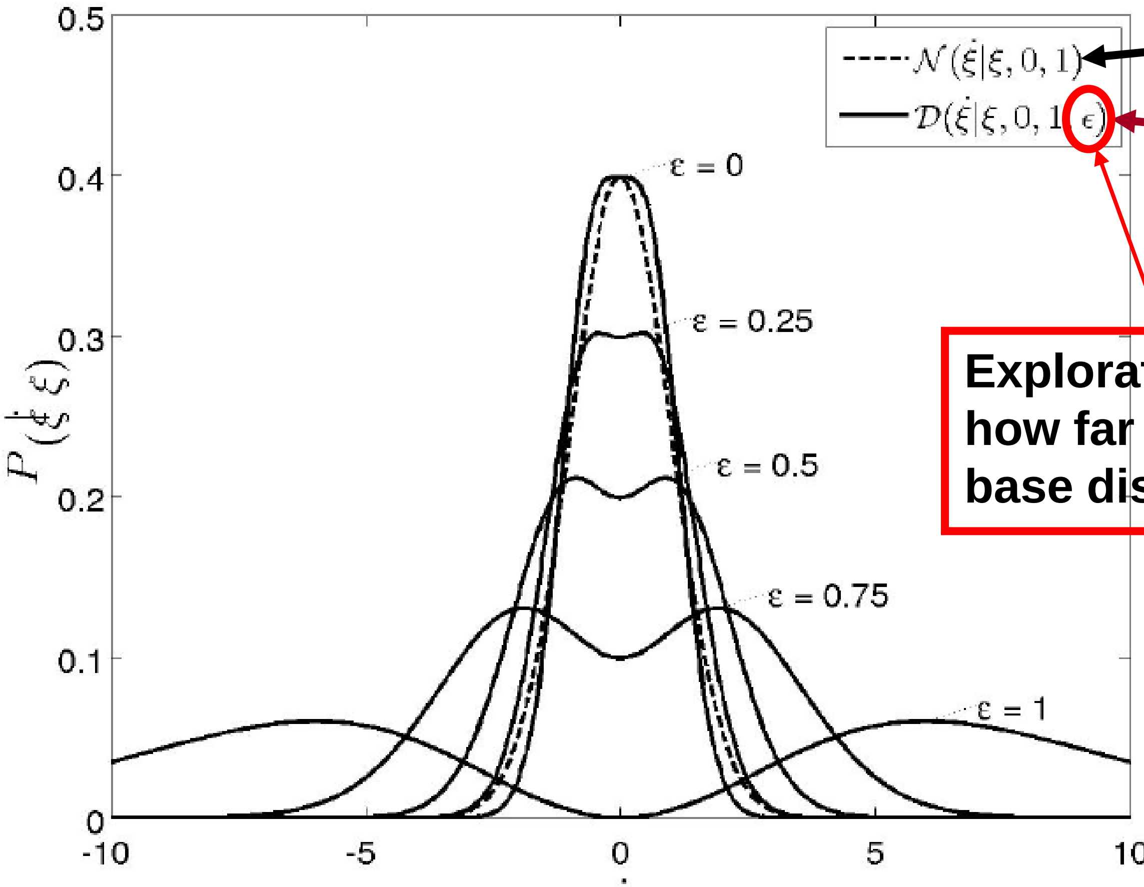
Instead of maximizing the similarity to demonstrator

Learning what Not to Do

Build a distribution (DONUT) that moves away from the bad demonstrations

→ Explore around the demonstrations and use the covariance to guide the exploration.

→ *Move away from things that have been visited a lot during unsuccessful demonstrations but remain within the vicinity of the demonstrations.*



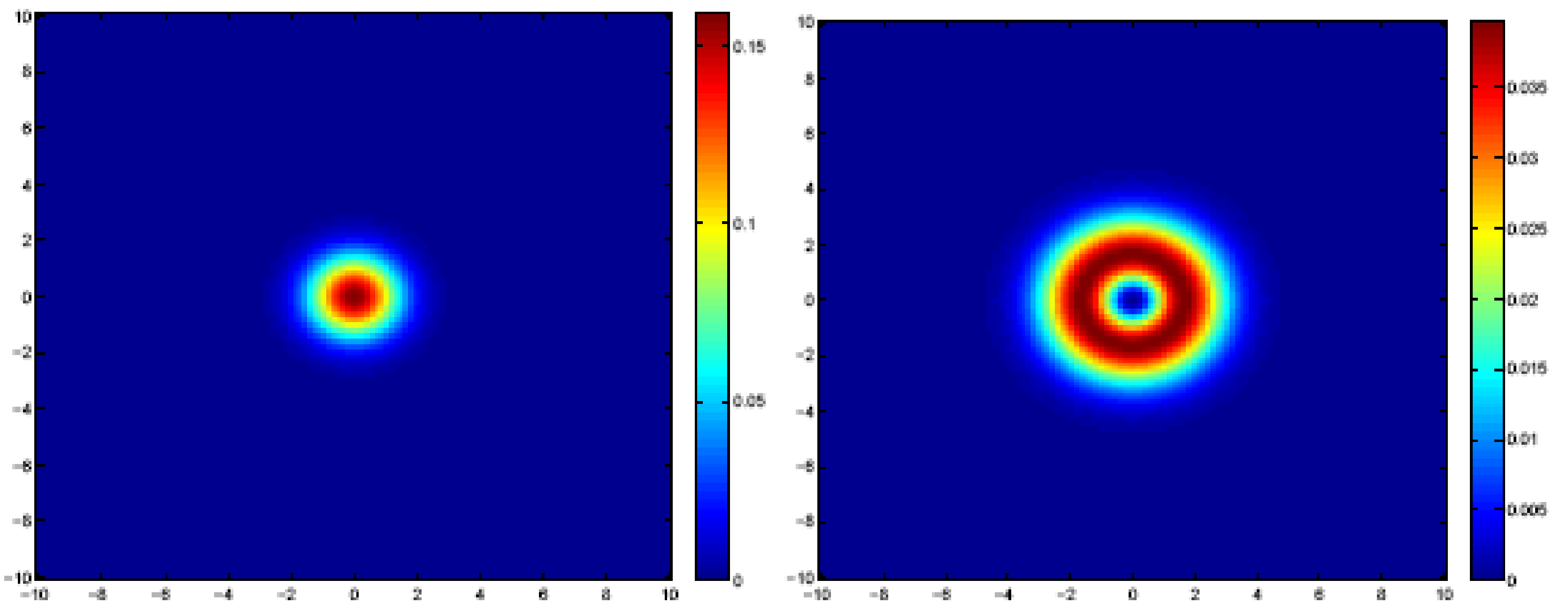
Base distribution: GMM

Donut distribution

Exploration parameter that determines how far away the peaks are from the base distribution.

Learning what Not to Do

Build a distribution (DONUT) that moves away from the bad demonstrations
→ Explore around the demonstrations and use the covariance to guide the exploration.
→ *Move away from things that have been visited a lot during unsuccessful demonstrations but remain within the vicinity of the demonstrations.*




ÉCOLE POLYTECHNIQUE
FÉDÉRALE DE LAUSANNE
LEARNING ALGORITHMS AND
SYSTEMS LABORATORY
LASA
Learning from failed demonstrations
Daniel Grollman
Aude Billard

The Future of Humanoids

This talk:

PbD enables robots to learn as humans do
Still fundamental issues to work on (Stability, time)
Interactive learning as humans do
Learning from failures, as humans do

Humanoids: Robots that think / learn / act / look as humans do?

Provocative question: Humanoids: What's next? / Why bother?

What is a non-humanoid robot?
Are all robots humanoid in some respect?
Can we build an alien robot?
Is HUMANOIDS necessary?