LAAS-CNRS

Task Detection for Motion Imitation on a Humanoid Robot

Sovannara Hak

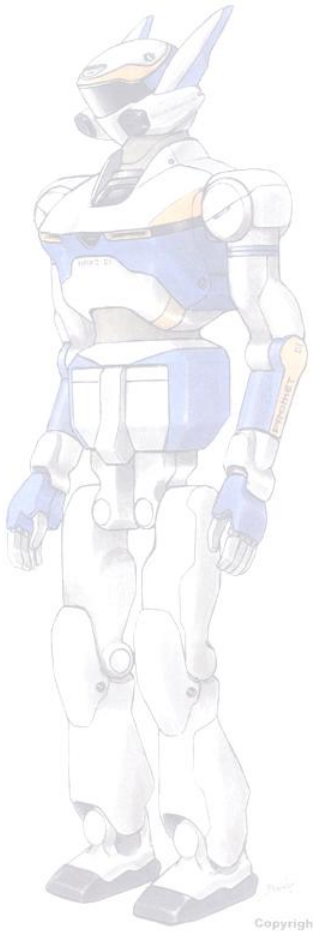
Advisors : Jean-Paul Laumond, Nicolas Mansard and Olivier Stasse

JNRH June 3rd, 4th 2010

LAAS-CNRS

Outline

1. High Speed Decision Making
2. Motion Imitation
   1. Problem Statement
   2. Task Function Formalism
   3. Hypothesis
   4. Simultaneous recognition and control
   5. Example
3. Conclusion



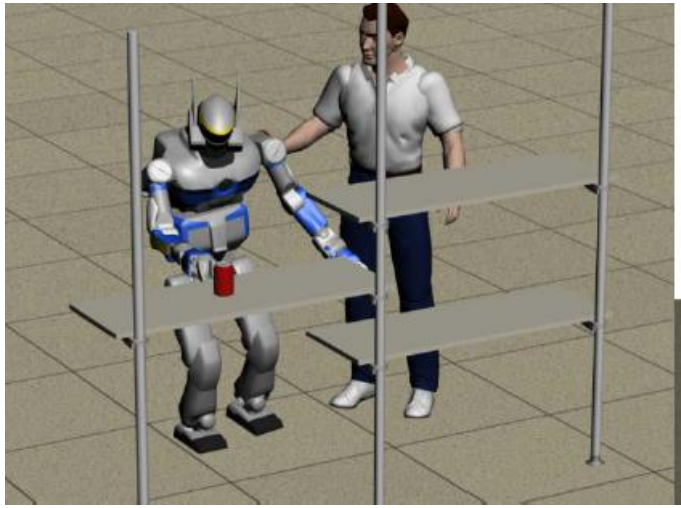
High Speed Decision Making

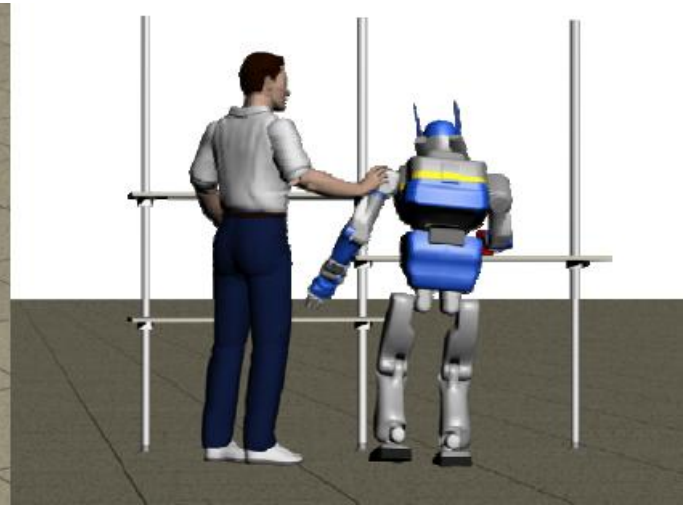
- Take a pertinent decision in the blink of an eye

R-Blink Project

http://staff.aist.go.jp/olivier.stasse/rblink/index.html

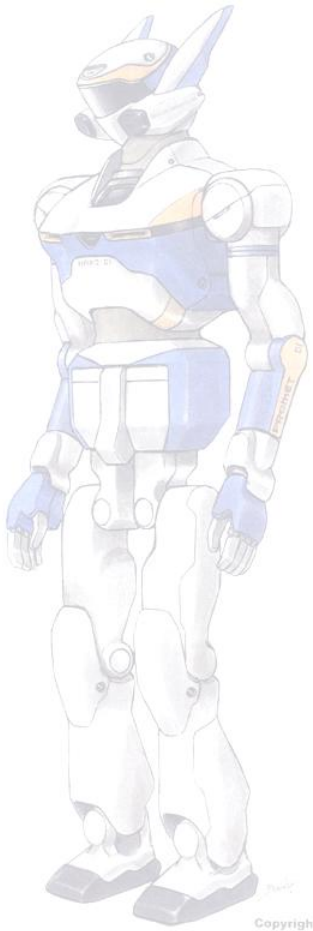
- French National Research Agency - ANR
- Exploring High Speed Decision Making For Motion Generation with a Humanoid Robot
- 4 Researchers, 1 post-doc, 4 PhD students





Outline

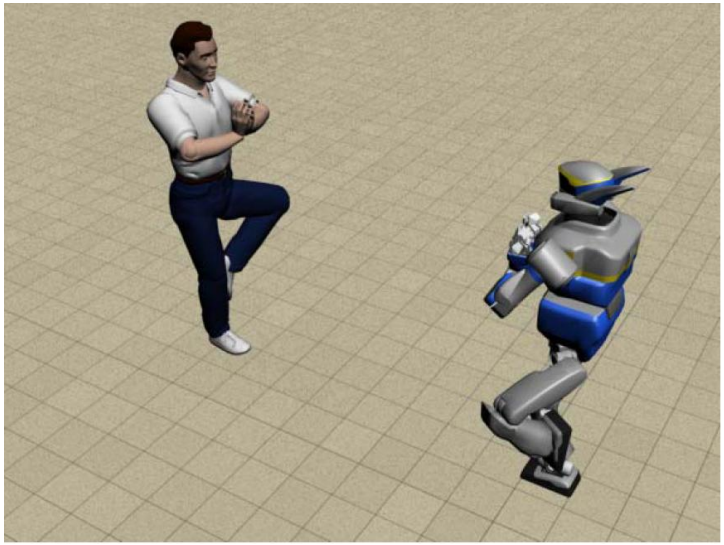
1. High Speed Decision Making
2. Motion Imitation
   1. Problem Statement
   2. Task Function Formalism
   3. Hypothesis
   4. Simultaneous recognition and control
   5. Example
3. Conclusion



Motion Imitation (1)

Problem Statement

- Humanoid robot control
  - Highly redundant
  - Dynamic constraints
- Reactive imitation



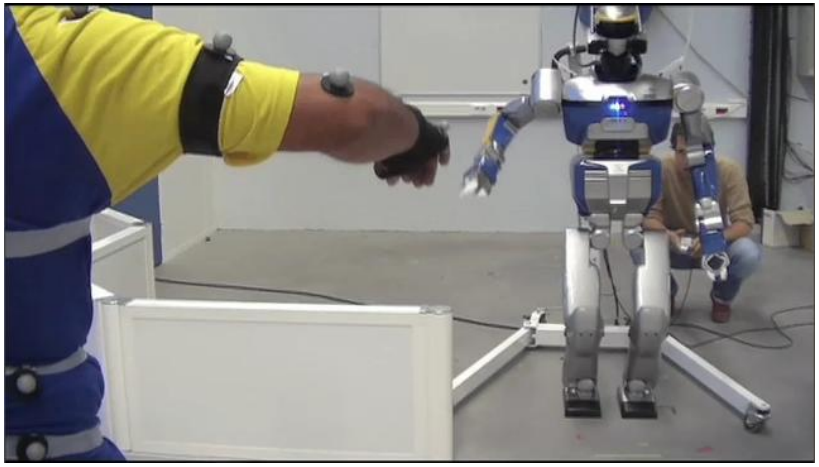
Provided a motion, perform a reactive imitation

Motion Imitation (2)

- S. Nakaoka, A. Nakazawa, F. Kanehiro, K. Kaneko, M. Morisawa and K. Ikeuchi (2005)

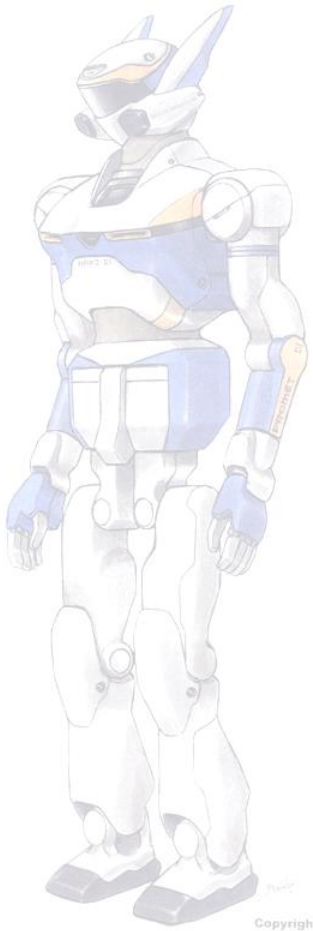
Motion Imitation (3)

- F. Montecillo, M. Sreenivasa and J.P. Laumond (icinco 2010)



Outline

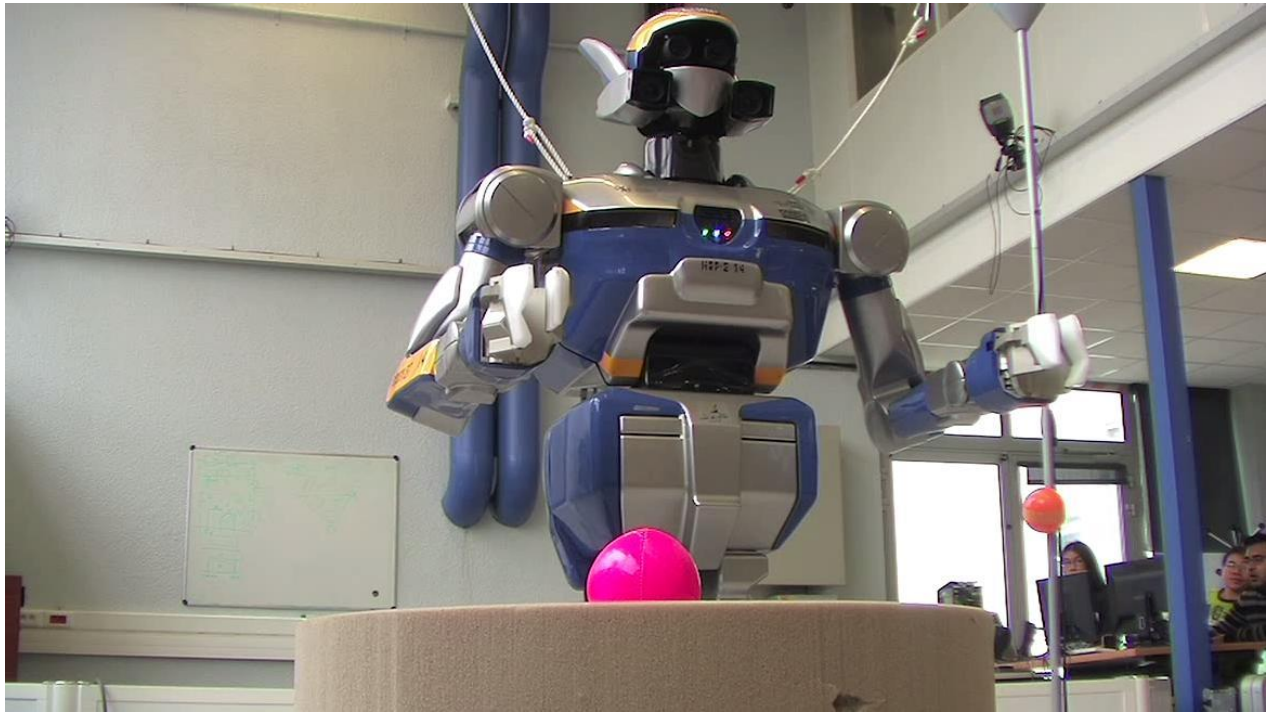
1. High Speed Decision Making
2. Motion Imitation
    1. Problem Statement
    2. Task Function Formalism
    3. Hypothesis
    4. Simultaneous recognition and control
    5. Example
3. Conclusion



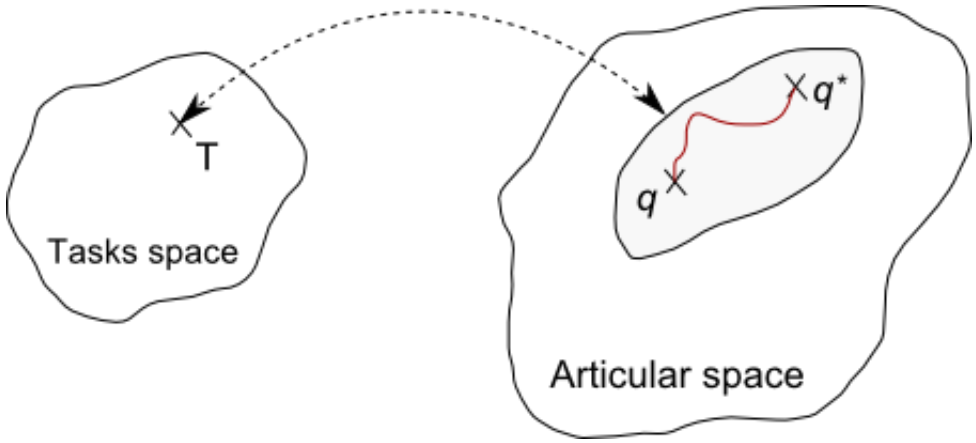
Task Function Formalism (1)

- Task example : position of a the hand of the robot

$$e = s^* - s$$

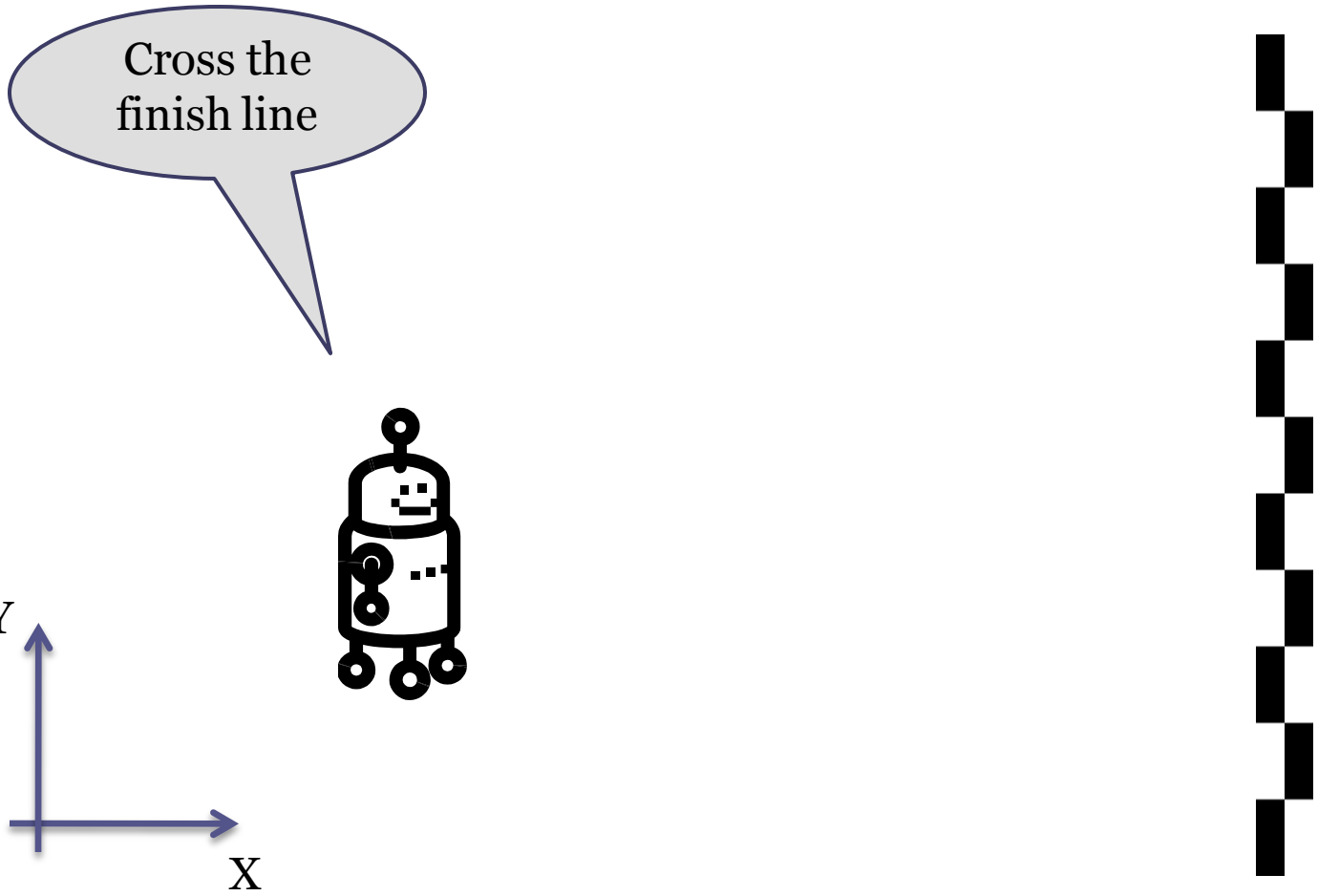


Task Function Formalism (2)

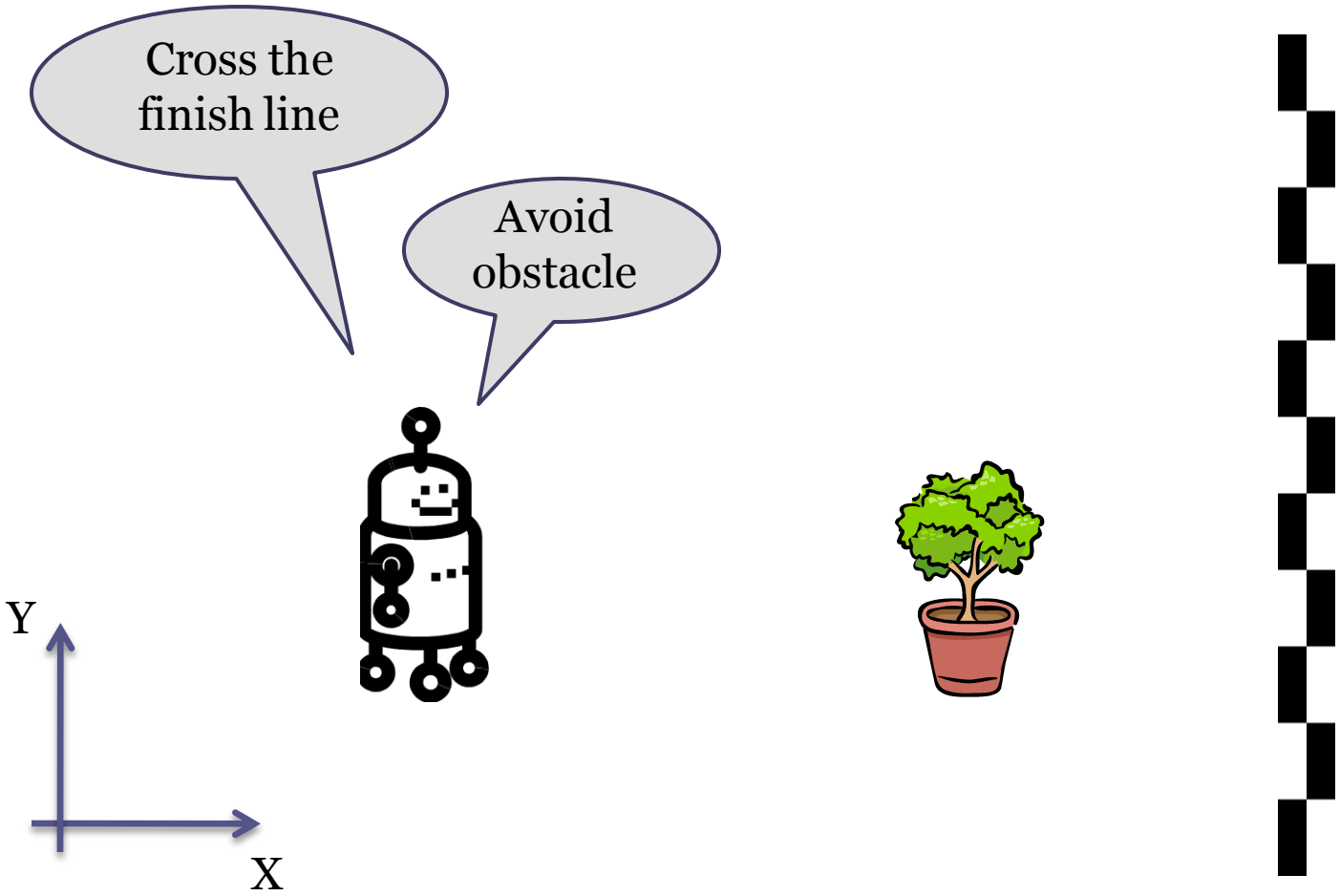


Relationship between the tasks space and articular space

Task Redundancy (1)

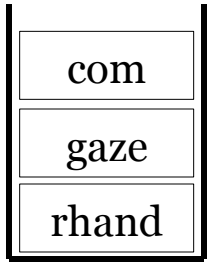


Task Redundancy (2)



Stack of Tasks

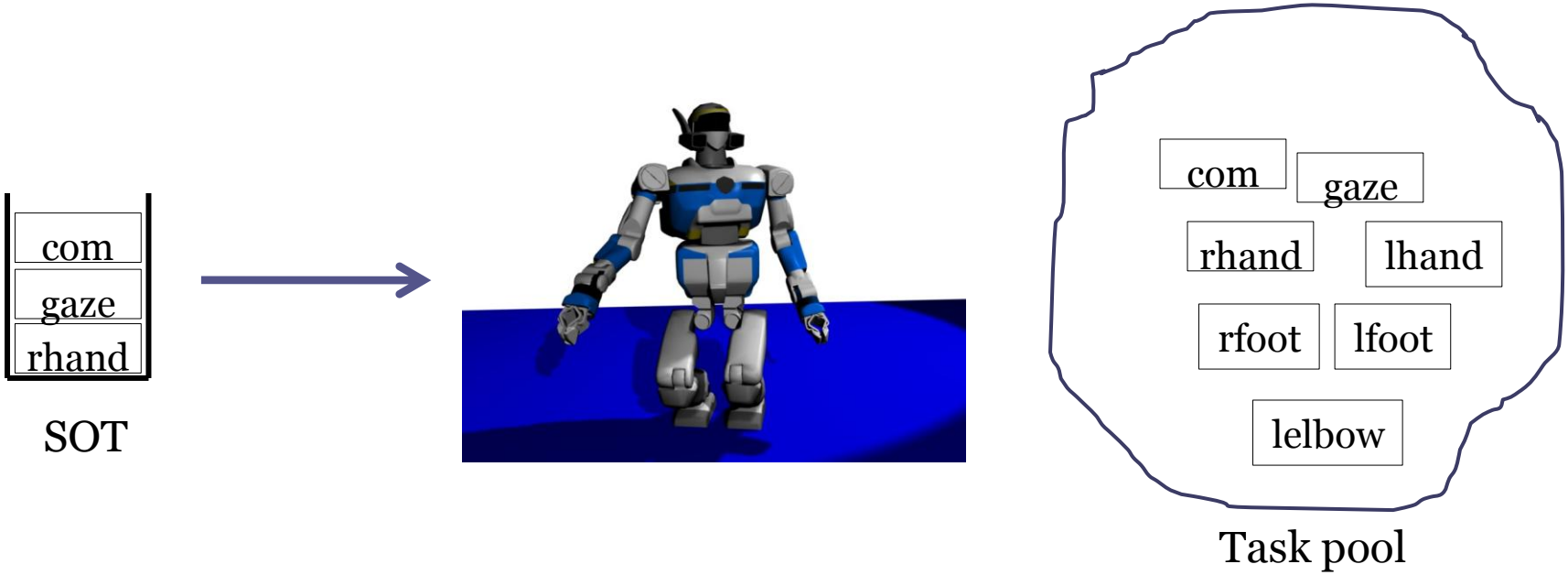
- N. Mansard and F. Chaumette. Task Sequencing for sensor-based control. *Transaction on robotics*. (2007)
  - Complete implementation of the task function formalism
  - Structure preserves continuity of the control law
    - Adding, removing, swapping priority order of a task
  - Constraint locally taken into account



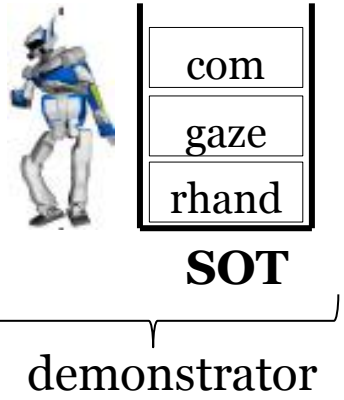
$$\dot{\mathbf{q}}_i = \dot{\mathbf{q}}_{i-1} + (\mathbf{J}_i \mathbf{P}^A_{i-1})^+ (\dot{\mathbf{e}}_i - \mathbf{J}_i \dot{\mathbf{q}}_{i-1})$$

Motion Imitation (1)

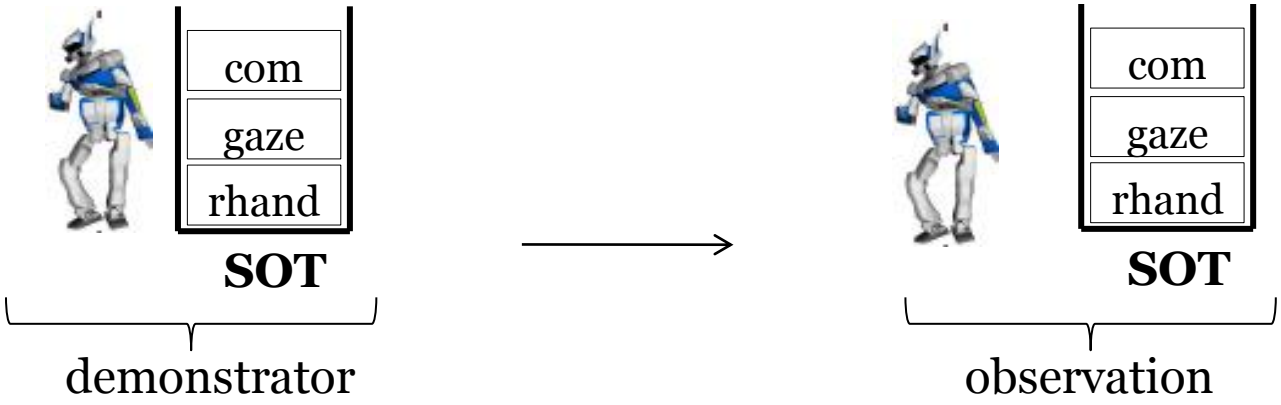
- Imitation of a motion is equivalent to execute the same stack of tasks
- Motion is supposed to be generated from a stack of tasks
- All tasks involved in a motion belong to a known set of possible tasks
- Model of both teacher and replicant are known



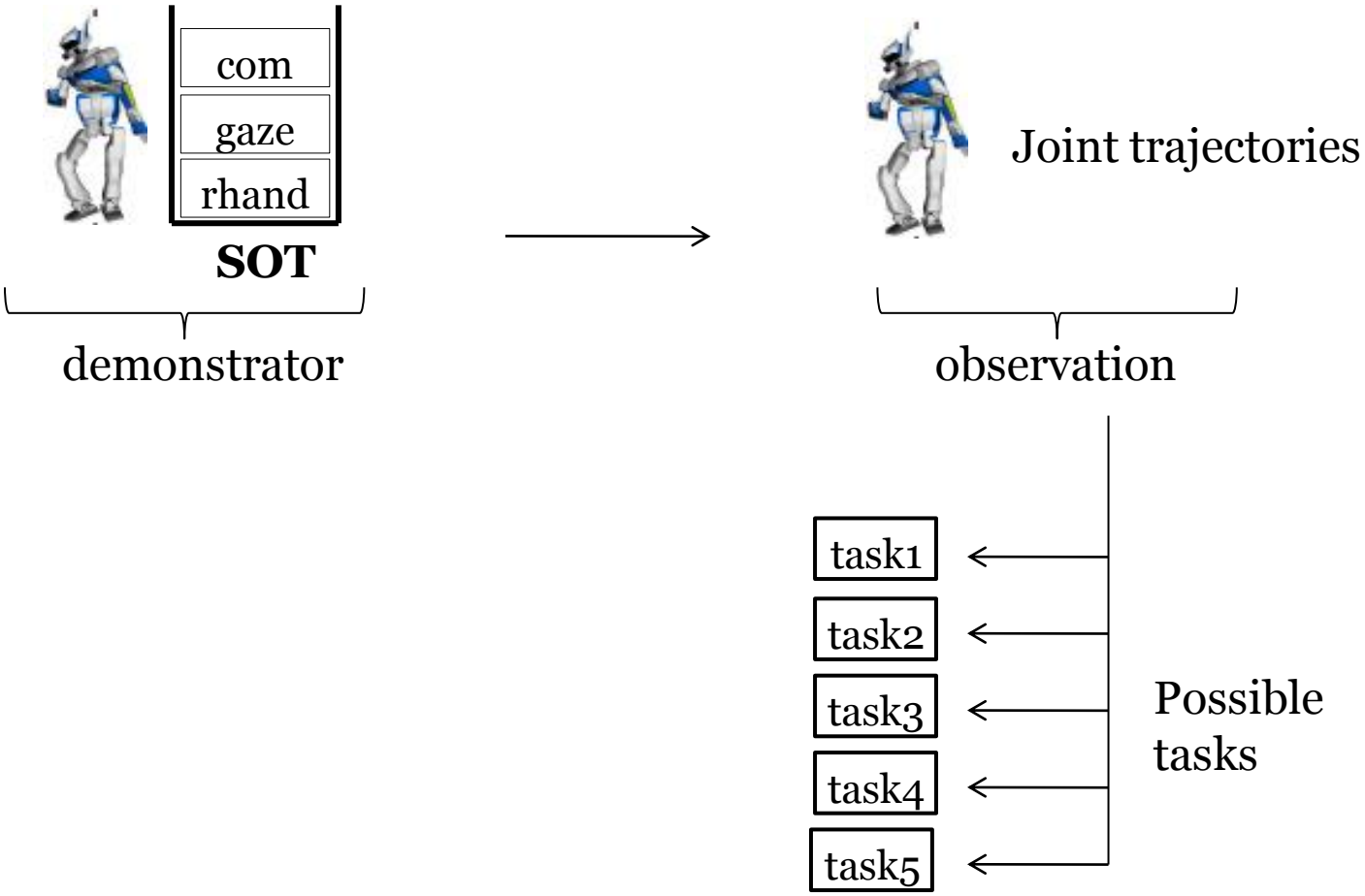
Motion Imitation (2)



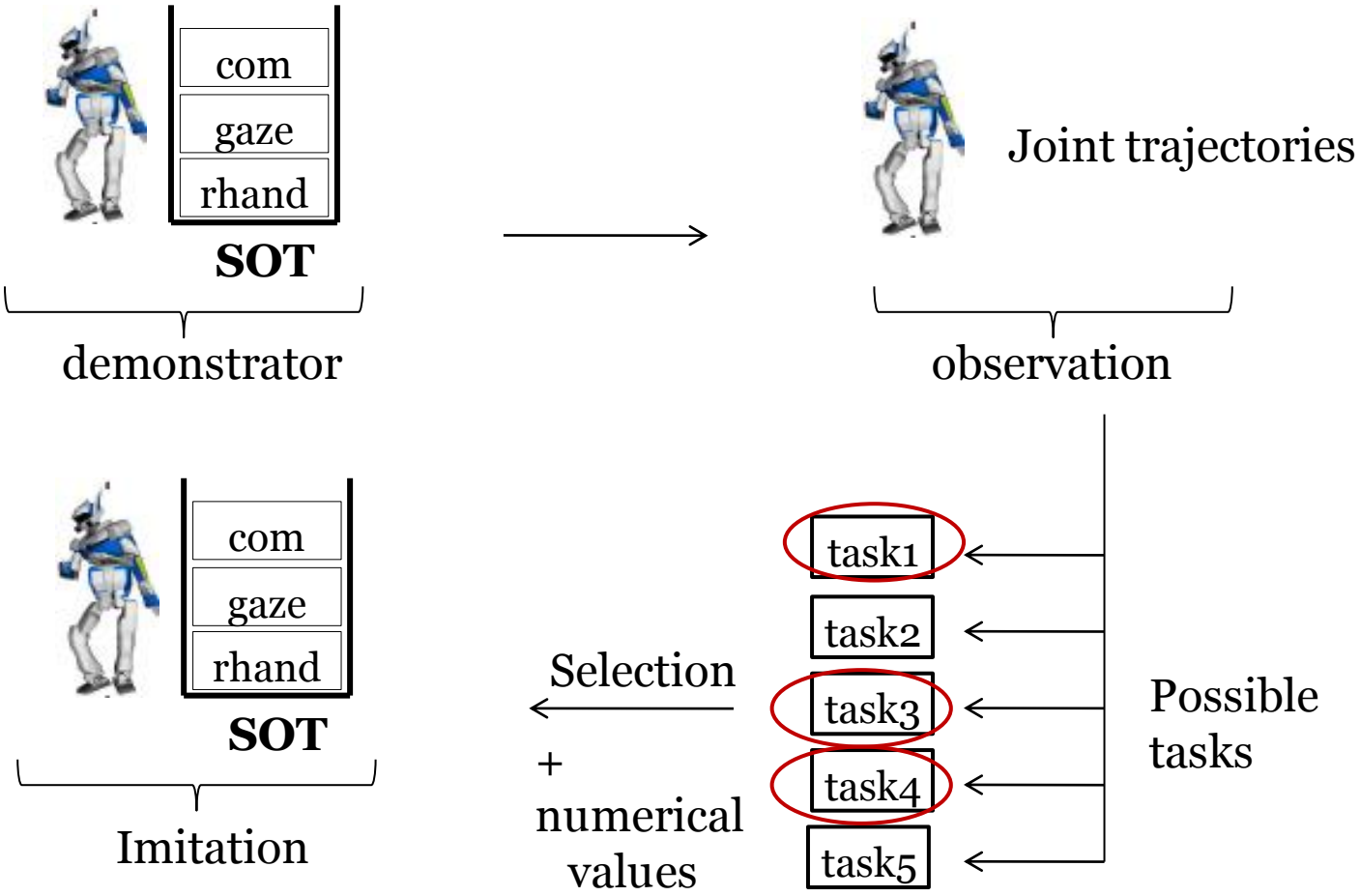
Motion Imitation (2)



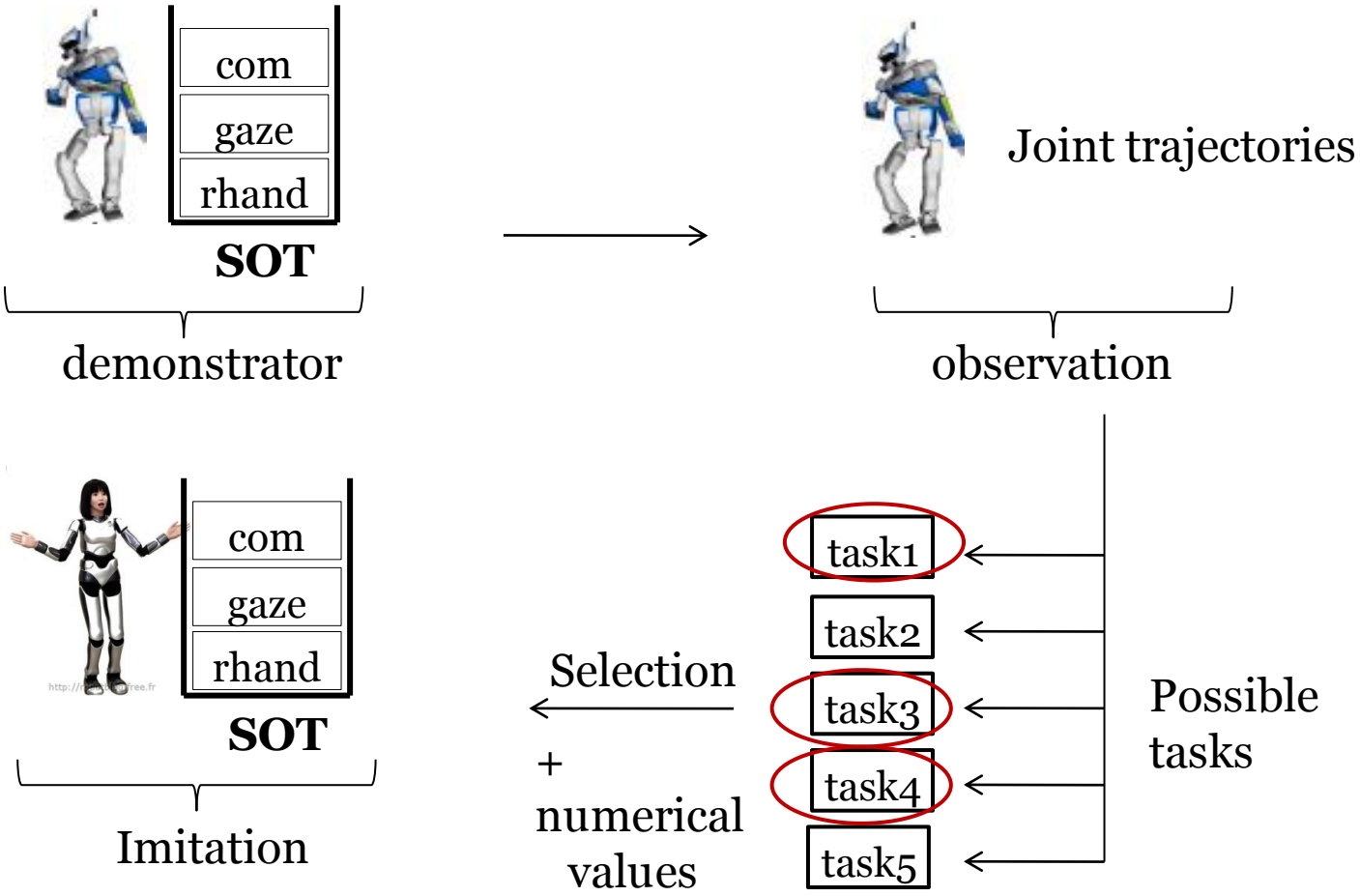
Motion Imitation (2)



Motion Imitation (2)



Motion Imitation (2)



Motion Imitation (2)

Joint trajectories

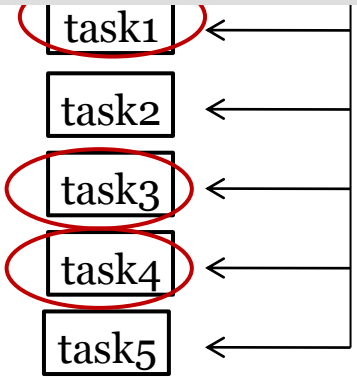
Simultaneous recognition and control

Imitation

Selection

\+

numerical values



Possible tasks

Task Fitting and numerical values

- Detection of a task
- Fit an observation data with a least square optimization

$$\mathbf{x}^* = \arg\min_{\mathbf{x}} \|\mathbf{p}^*(t) - \mathbf{p}_{\mathbf{x}}(t)\|$$

Optimal parameters of a task

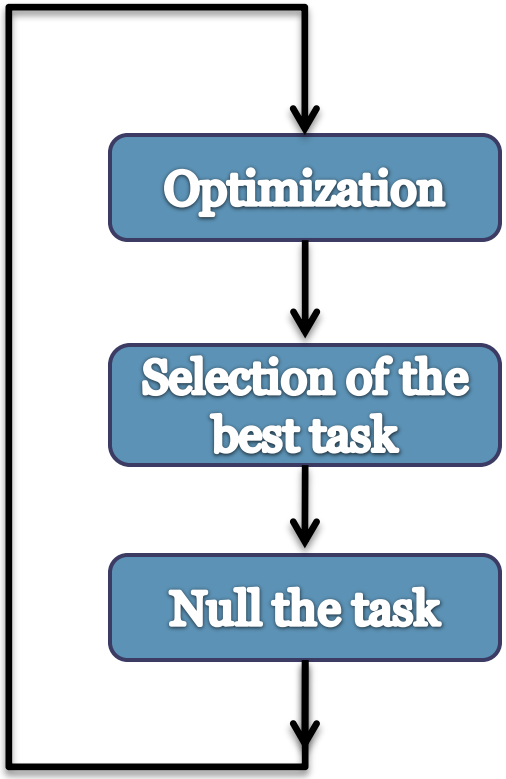
Observation

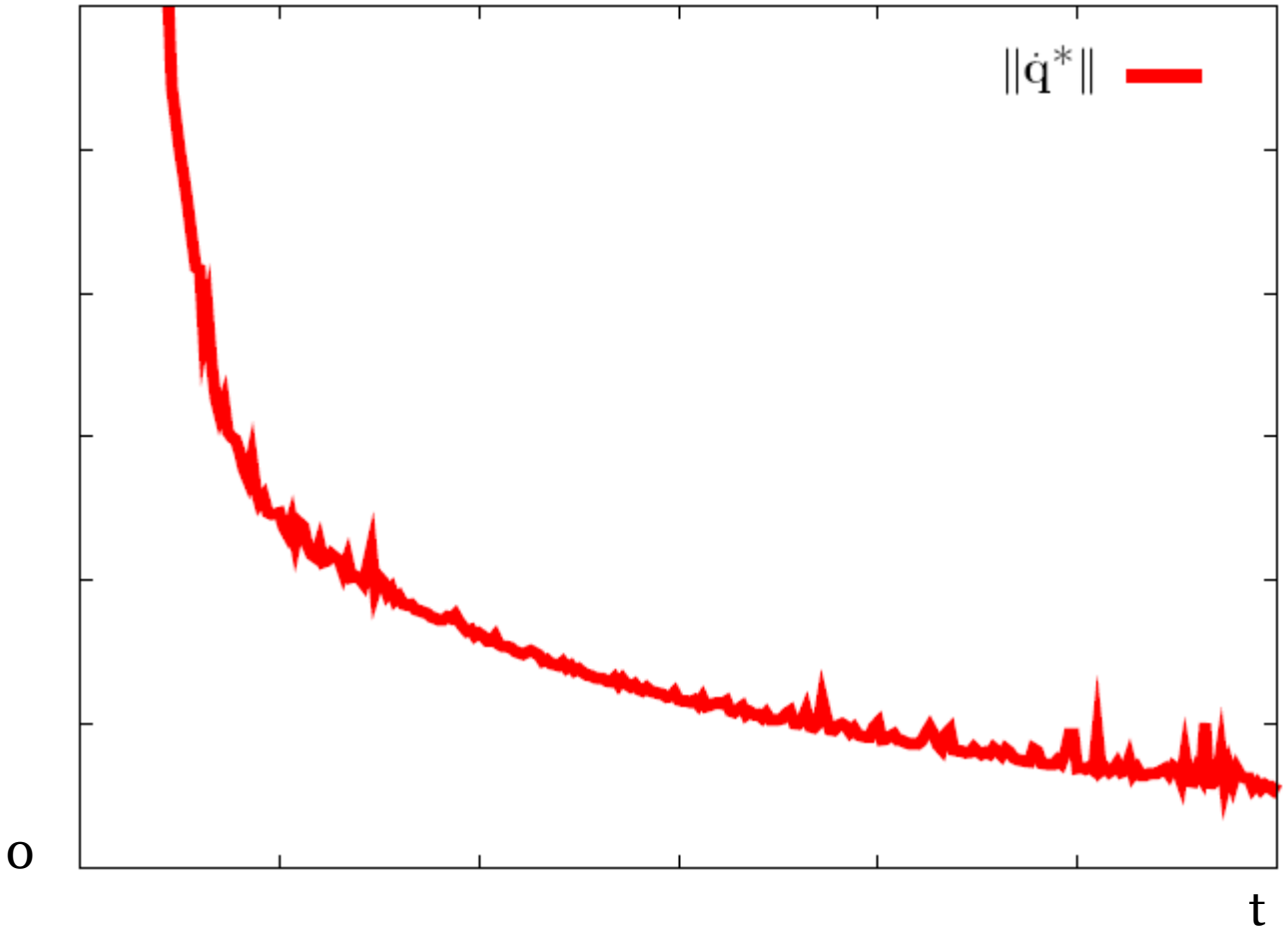
Model of the task parametered by x

- Value of the cost function is used to discriminate tasks

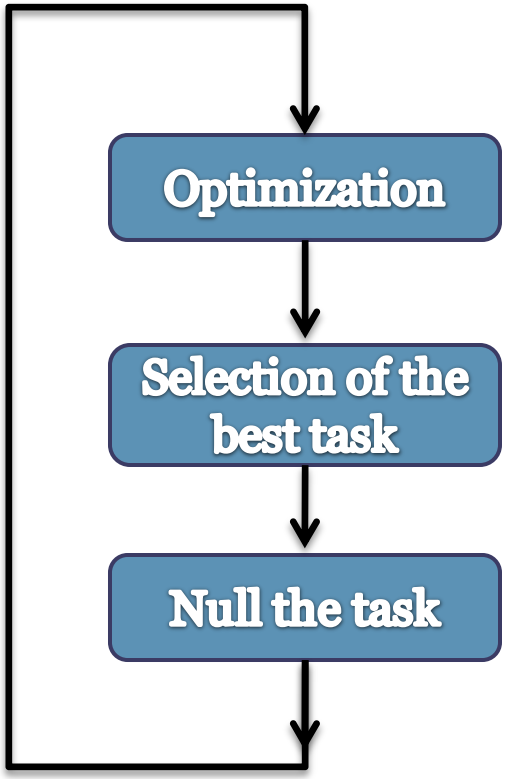
LAAS-CNRS

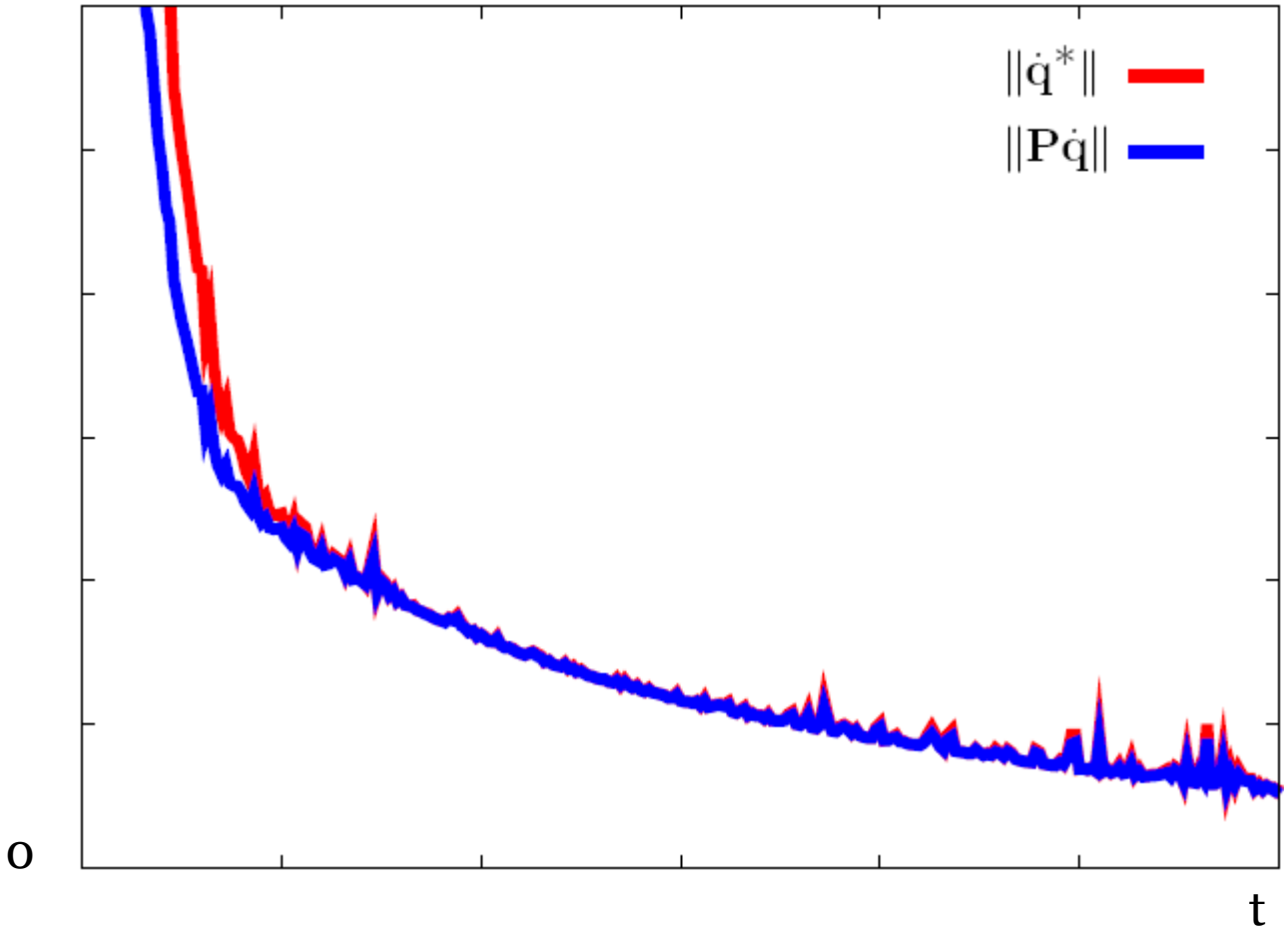
Tasks Recognition



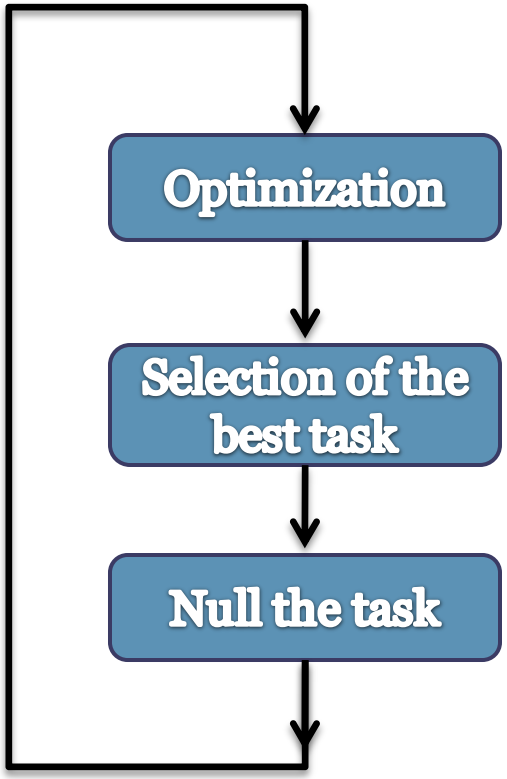


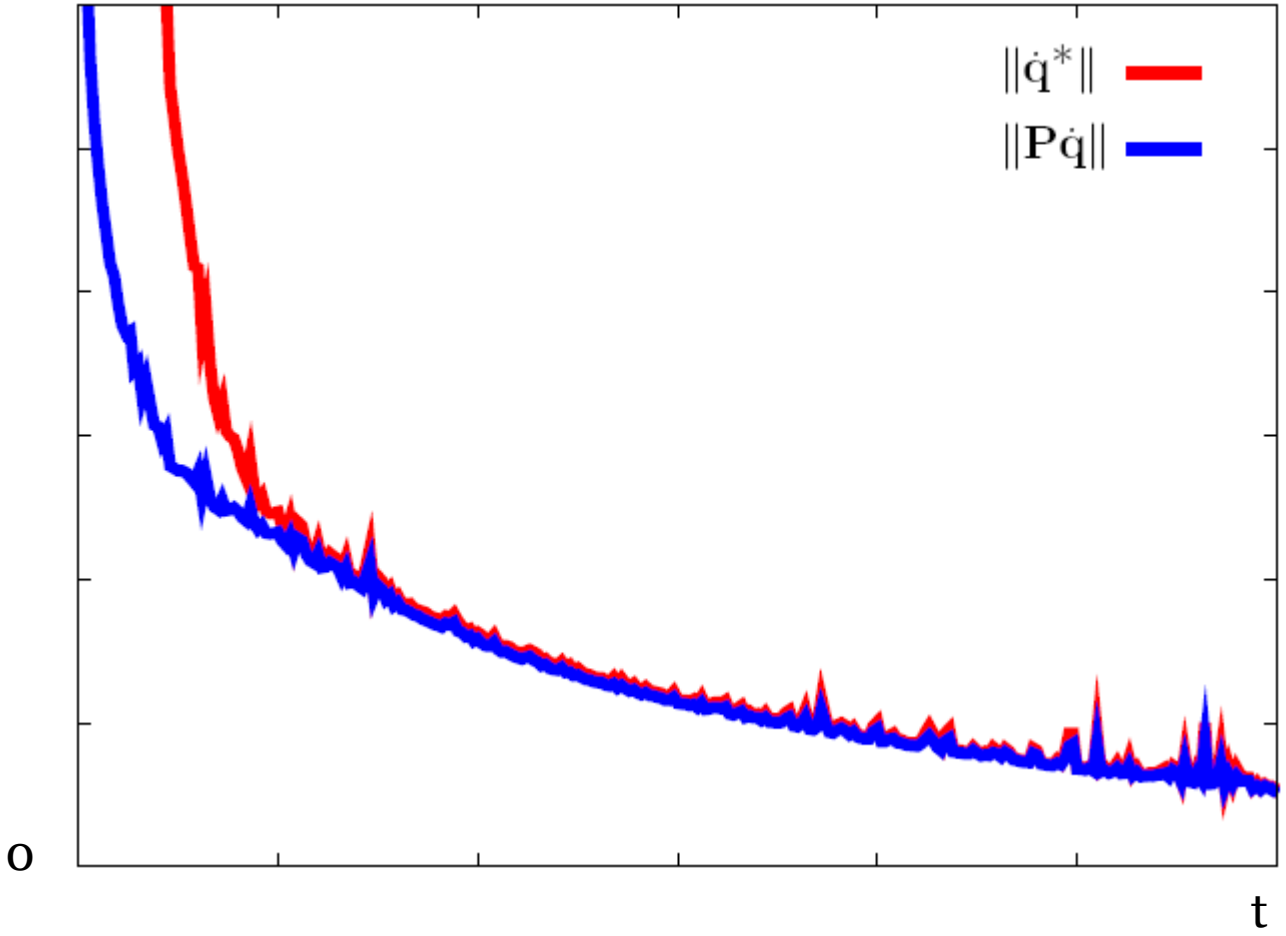
Tasks Recognition



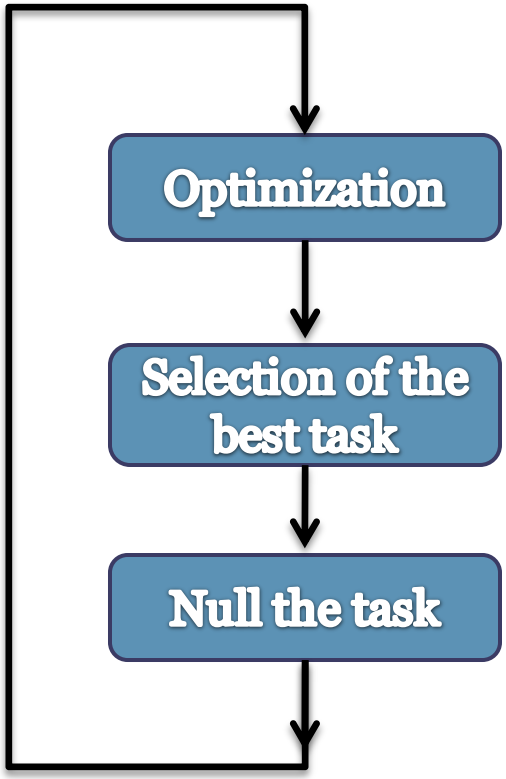


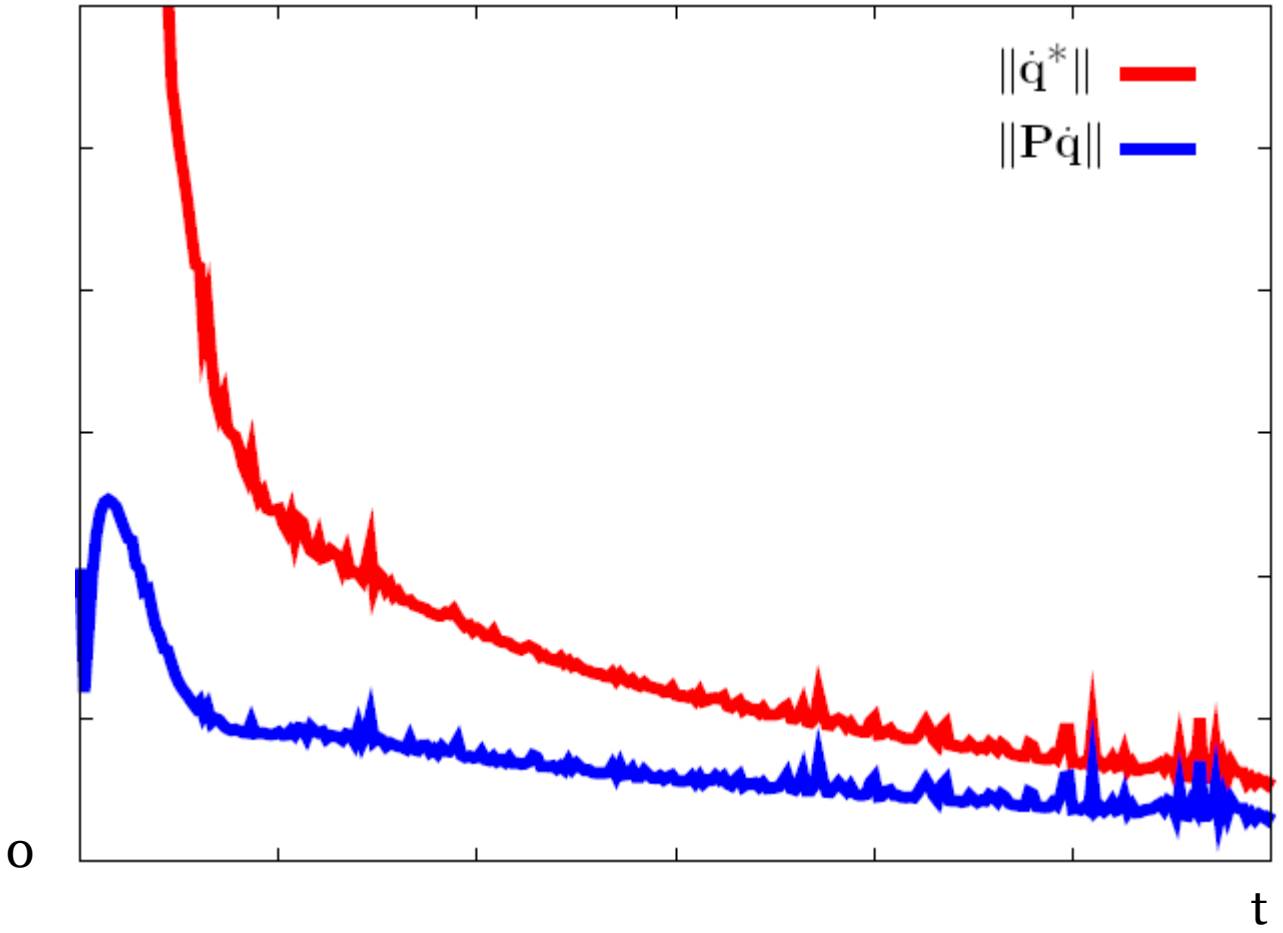
Tasks Recognition



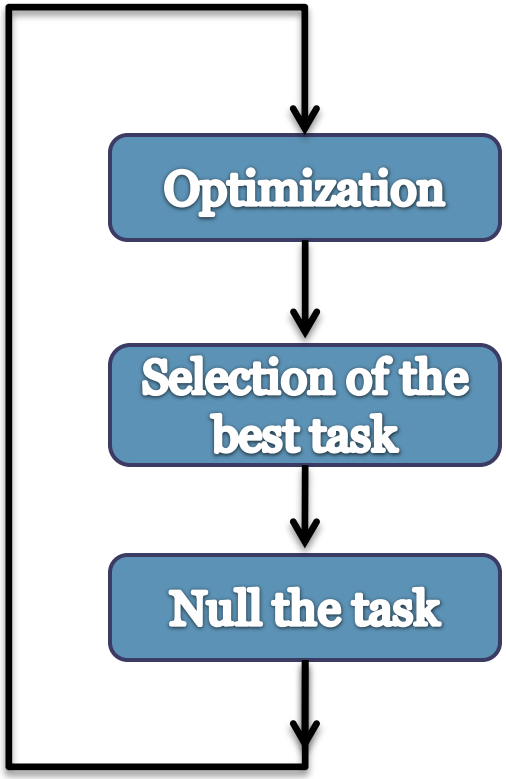


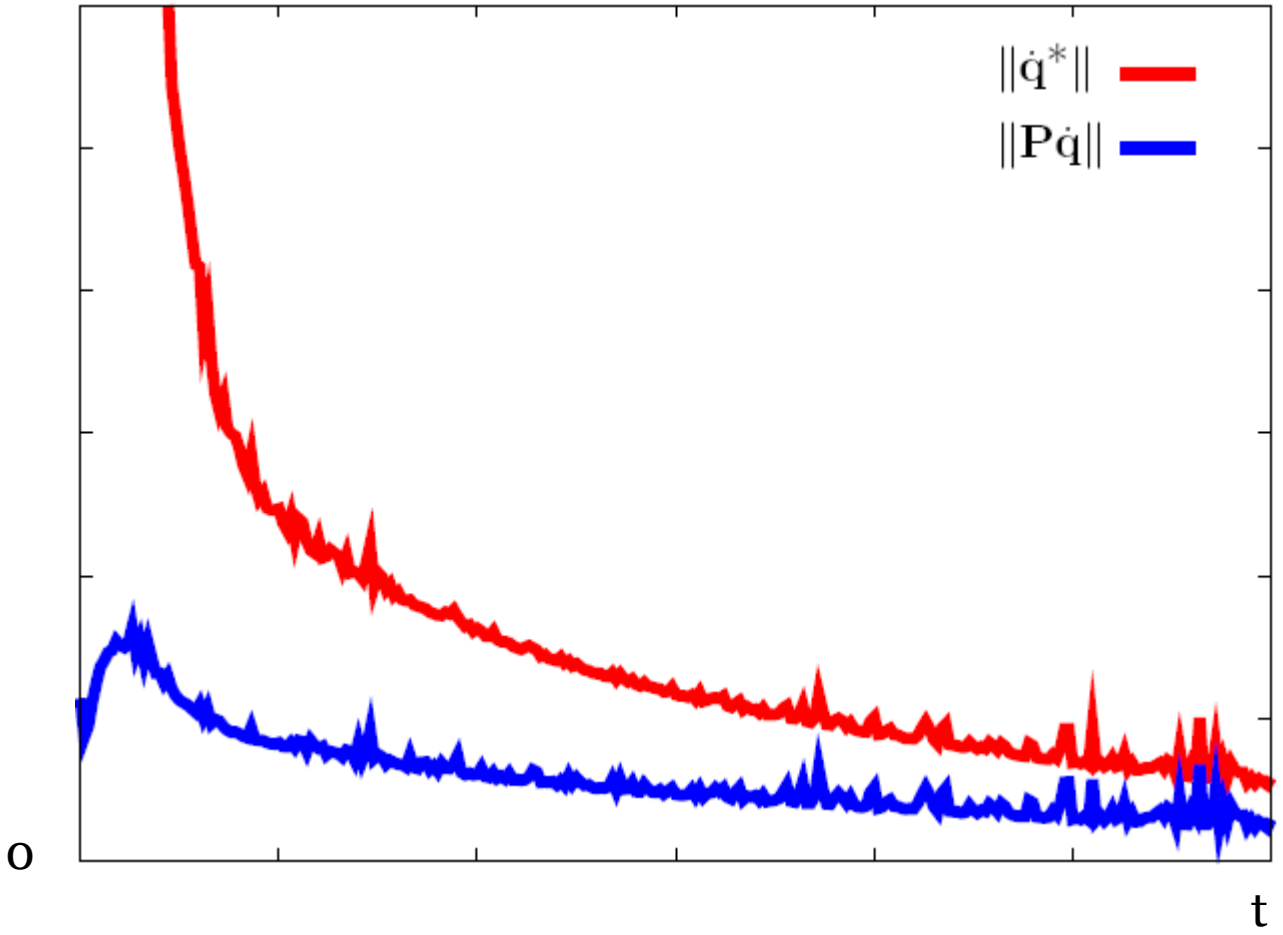
Tasks Recognition



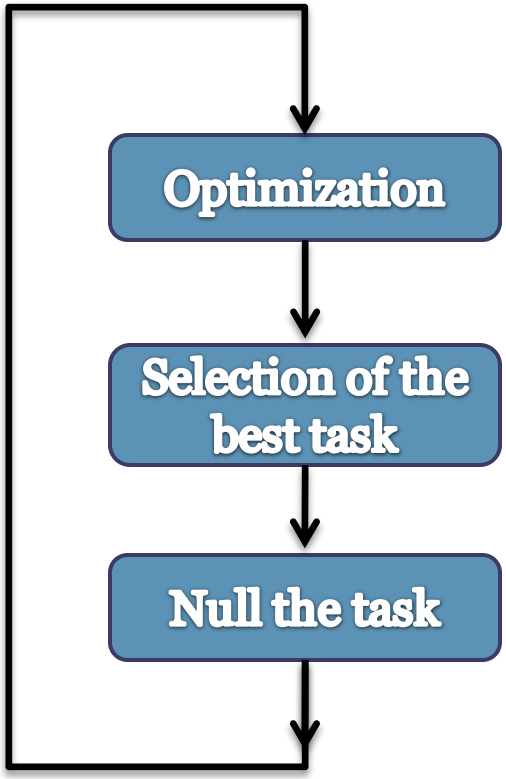


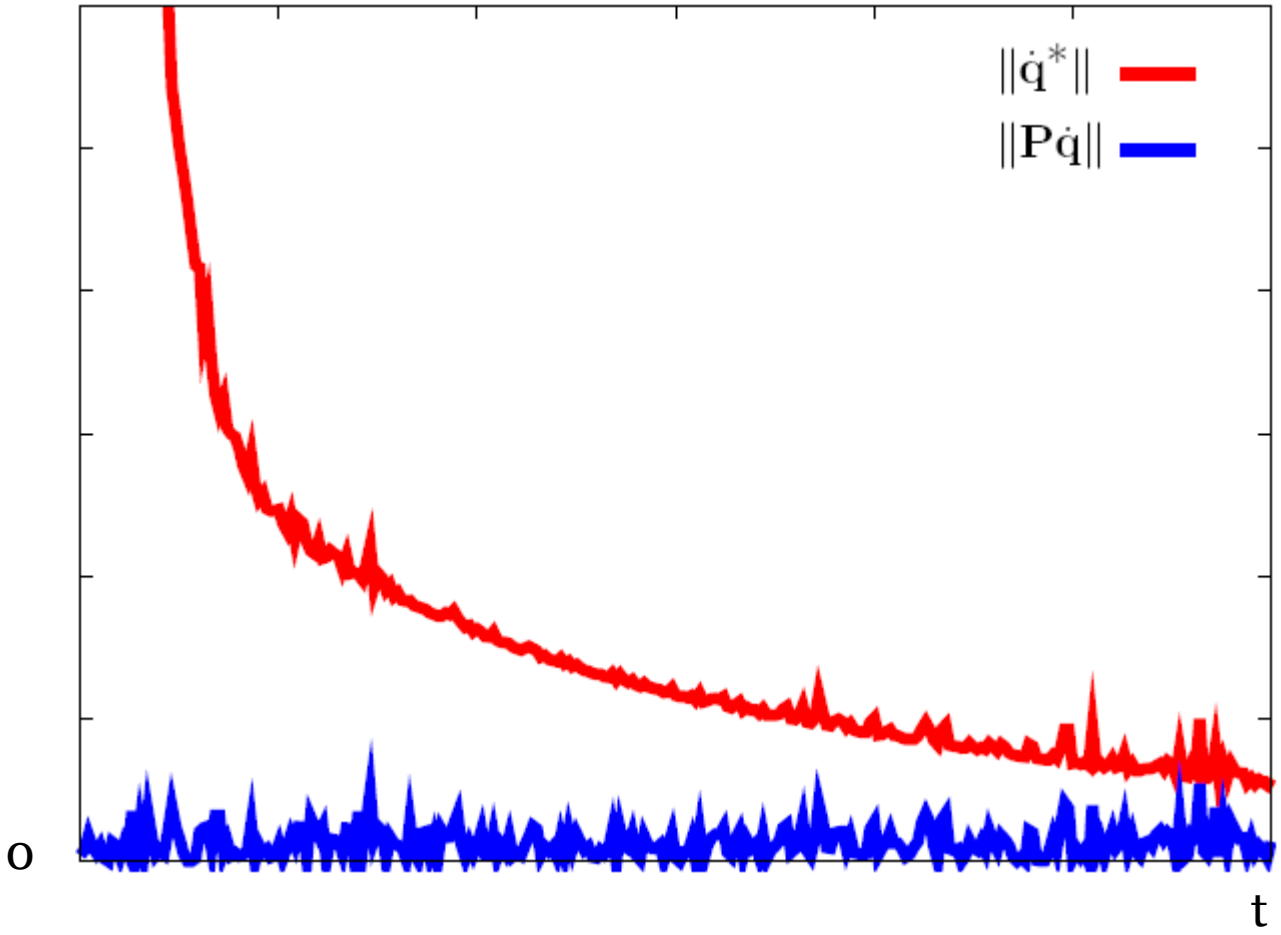
Tasks Recognition



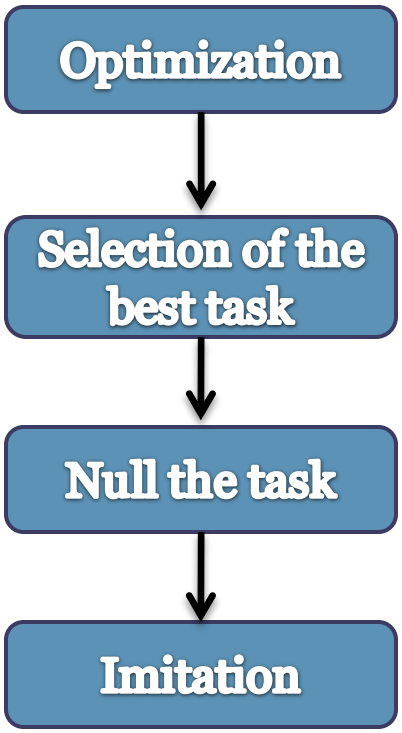


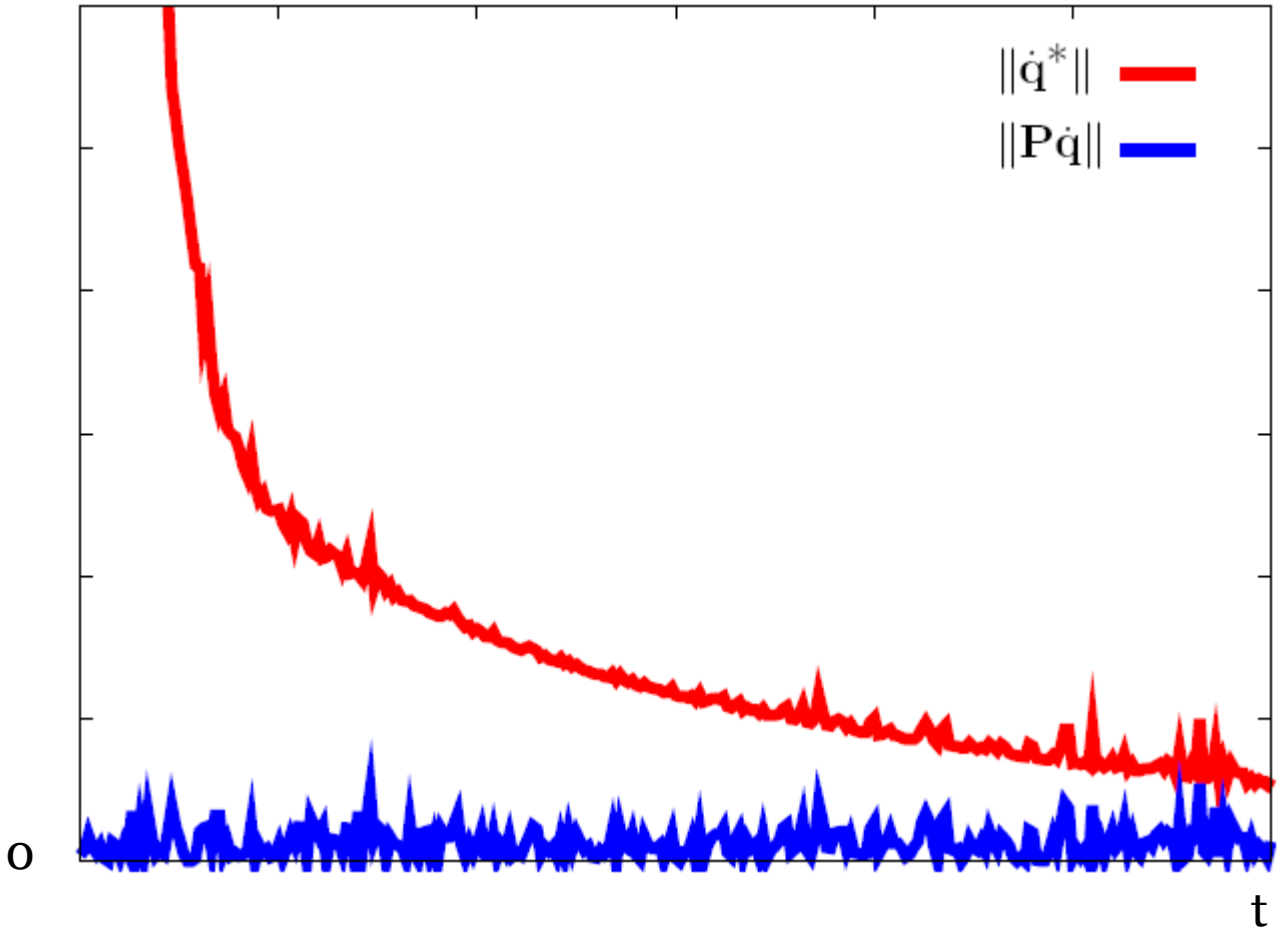
Tasks Recognition



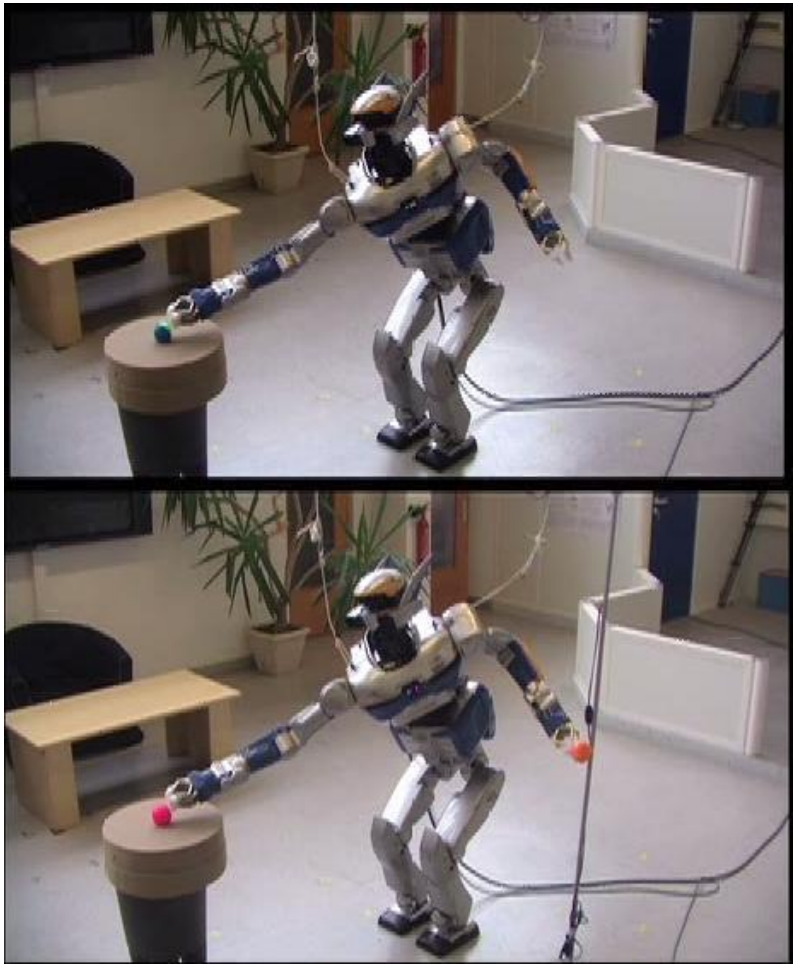


Tasks Recognition





Motion disambiguation (1)

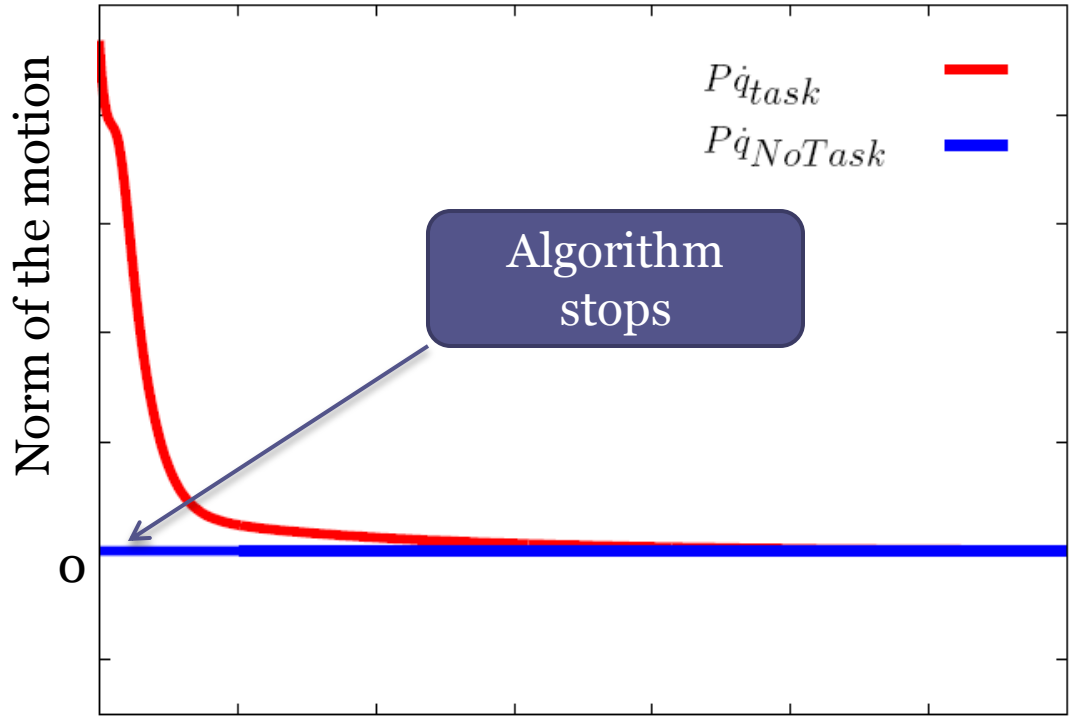


Motion disambiguation (2)

- Two different instances of a stack of tasks
  - With a right hand task (left hand moves due to COM compensation)
  - With a right hand and left hand task

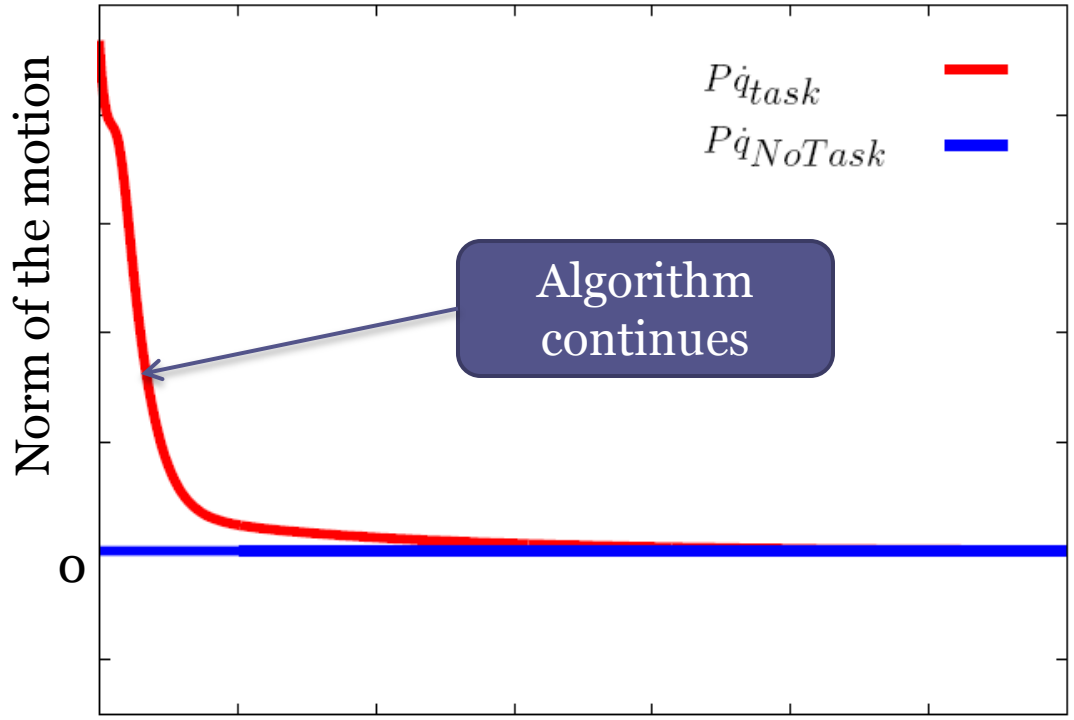
Motion disambiguation (3)

Projection of the motion in the right hand task nullspace



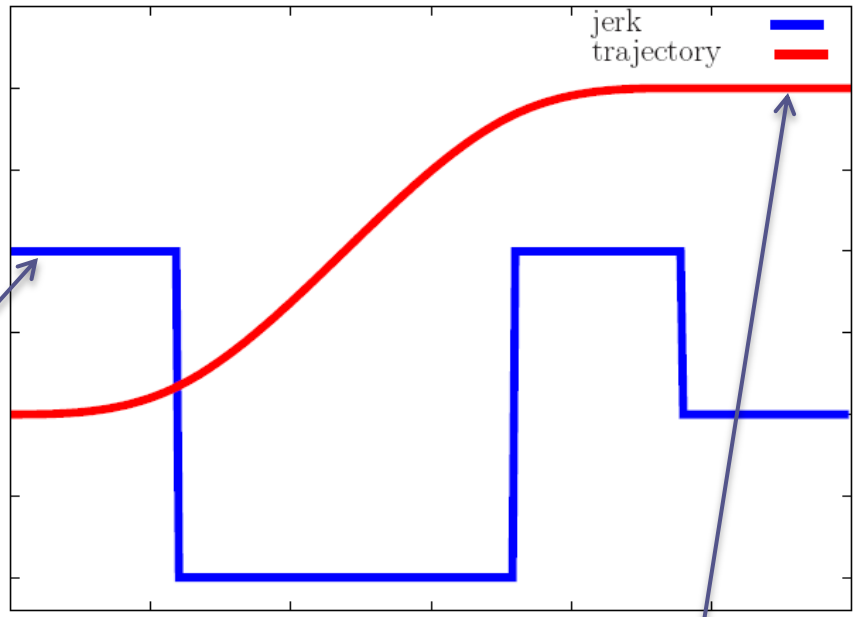
Motion disambiguation (3)

Projection of the motion in the right hand task nullspace



Human task modeling

- Using a minimum jerk motion

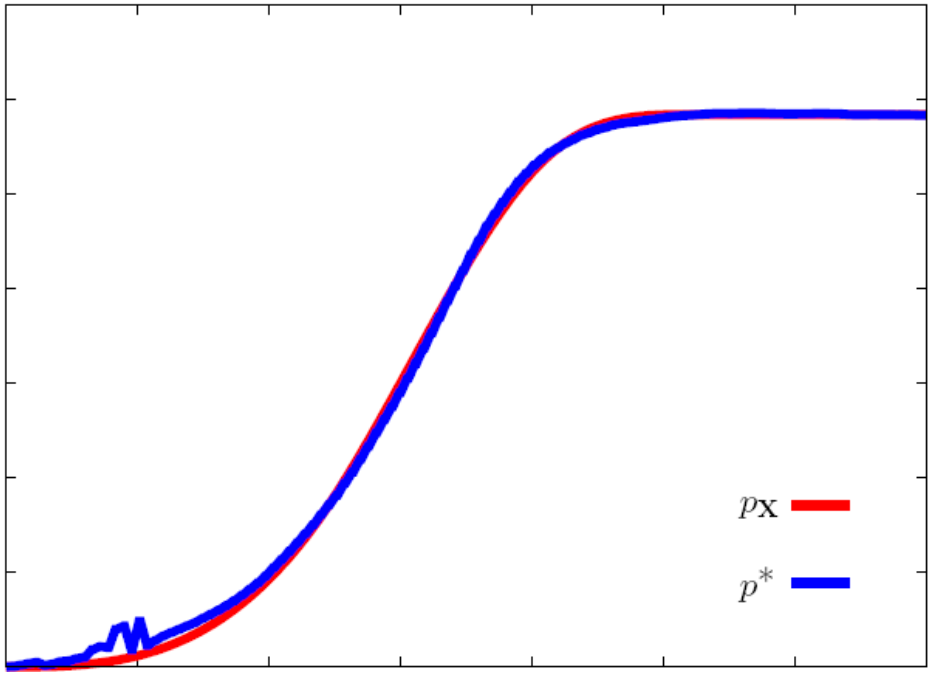


$$\text{jerk}(t) = \begin{cases} K_1 & if \quad 0 < t < \Delta t_1 \\ K_2 & if \quad dt1 < t < \Delta t_1 + \Delta t_2 \\ K_3 & if \quad dt2 < t < \Delta t_1 + \Delta t_2 + \Delta t_3 \\ 0 & if \quad t > \Delta t_1 + \Delta t_2 + \Delta t_3 \end{cases}$$

$$\mathbf{p_x}(t) = \overset{\mathbf{x}}{\iiint} \text{jerk}(t)\text{dt}$$

Human task modeling

- Fitting trajectory from a motion capture data



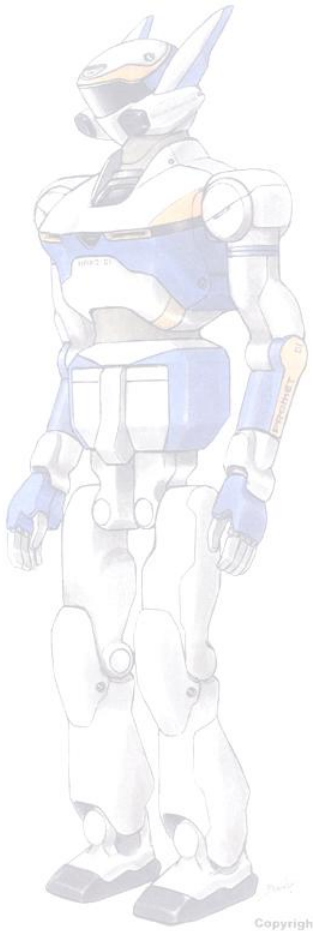
Trajectory Fitting by least square optimization

Application to the human motion

- Do non-controled members follow a minimum jerk trajectory ?
- Dealing with variability of human model to cancel a task

Outline

1. High Speed Decision Making
2. Motion Imitation
    1. Problem Statement
    2. Task Function Formalism
    3. Hypothesis
    4. Simultaneous recognition and control
    5. Example
3. Conclusion



Conclusion

1. Method for simultaneous recognition and control
   1. Working in the task space offers many advantages

2. Still some strong hypothesis
   1. The knowledge of the models
   2. The demonstrator must generate his motion from a stack of tasks