*Robotics and Advanced Manufacturing Group, CINVESTAV-IPN, Mexico
+CNRS-AIST Joint Robotics Laboratory, UM13218/CRT, Japan

Time-Constrained Whole Body Control with Smooth Task Transitions

G. Jarquín* G. Arechavaleta* A. Escande+ E. Yoshida+

Whole-Body Control for Robots in the Real World

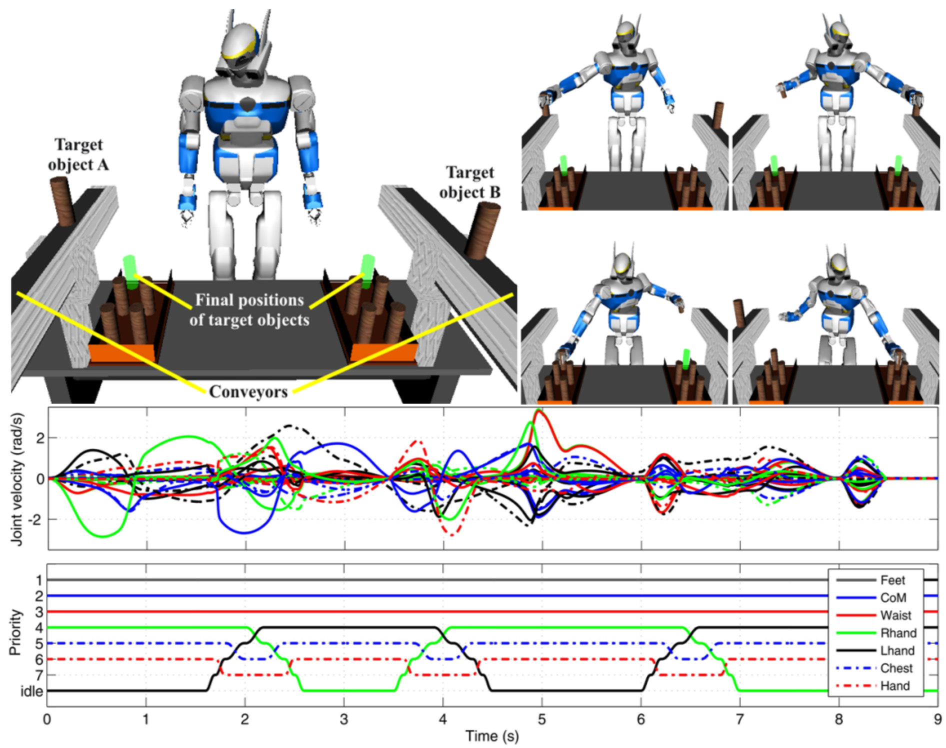
IEEE/RSJ International Conference on Intelligent Robots and Systems
Chicago, Illinois, September 2014

Time-Constrained Whole-Body Control with Smooth Task Transitions

Gerardo Jarquín*, Gustavo Arechavaleta*, Adrien Escande+ and Eiichi Yoshida+

* Robotics and Advanced Manufacturing Group, CINVESTAV-IPN, Mexico
+ Joint Robotics Laboratory UMI3218/CRT, CNRS-AIST, Japan

- Desired time convergence of hierarchical tasks is achieved.
- The control strategy smoothly handles task transitions in the stack of tasks at running time.
- The calculation is performed in real-time because it preserves the HIK solver's computational cost.



Time-constrained humanoid reaching motion

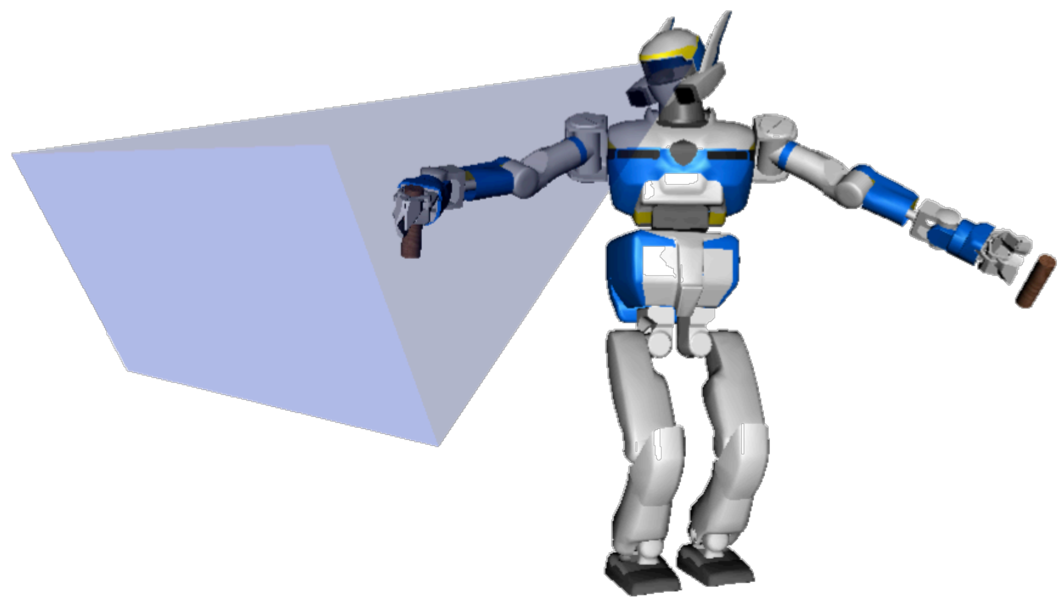
G. Jarquín, G. Arechavaleta, A. Escande, E. Yoshida

Hierarchical Inverse Kinematics (HIK)
with Desired Time Convergence (DTC) of Tasks

Smooth task transitions

Real-time execution

$$SoT = \begin{cases} \dot{\boldsymbol{e}}_A(\boldsymbol{q}) \\ \dot{\boldsymbol{e}}_B(\boldsymbol{q}) \\ \dot{\boldsymbol{e}}_C(\boldsymbol{q}) \\ \dot{\boldsymbol{e}}_D(\boldsymbol{q}) \\ \vdots \\ \dot{\boldsymbol{e}}_p(\boldsymbol{q}) \end{cases} \Rightarrow \begin{cases} \dot{\boldsymbol{e}}_C(\boldsymbol{q}) \\ \dot{\boldsymbol{e}}_A(\boldsymbol{q}) \\ \dot{\boldsymbol{e}}_D(\boldsymbol{q}) \\ \dot{\boldsymbol{e}}_E(\boldsymbol{q}) \\ \vdots \\ \dot{\boldsymbol{e}}_{p-1}(\boldsymbol{q}) \end{cases}$$



Inputs:

- The set of tasks and DTC of each task

$$\dot{\mathbf{e}}_A(\mathbf{q},\alpha_A(t)) \quad \dot{\mathbf{e}}_B(\mathbf{q},\alpha_B(t)) \quad \dot{\mathbf{e}}_C(\mathbf{q},\alpha_C(t)) \quad \dot{\mathbf{e}}_D(\mathbf{q},\alpha_D(t)) \quad \dot{\mathbf{e}}_E(\mathbf{q},\alpha_E(t))$$

- The initial Stack of Tasks (SoT).

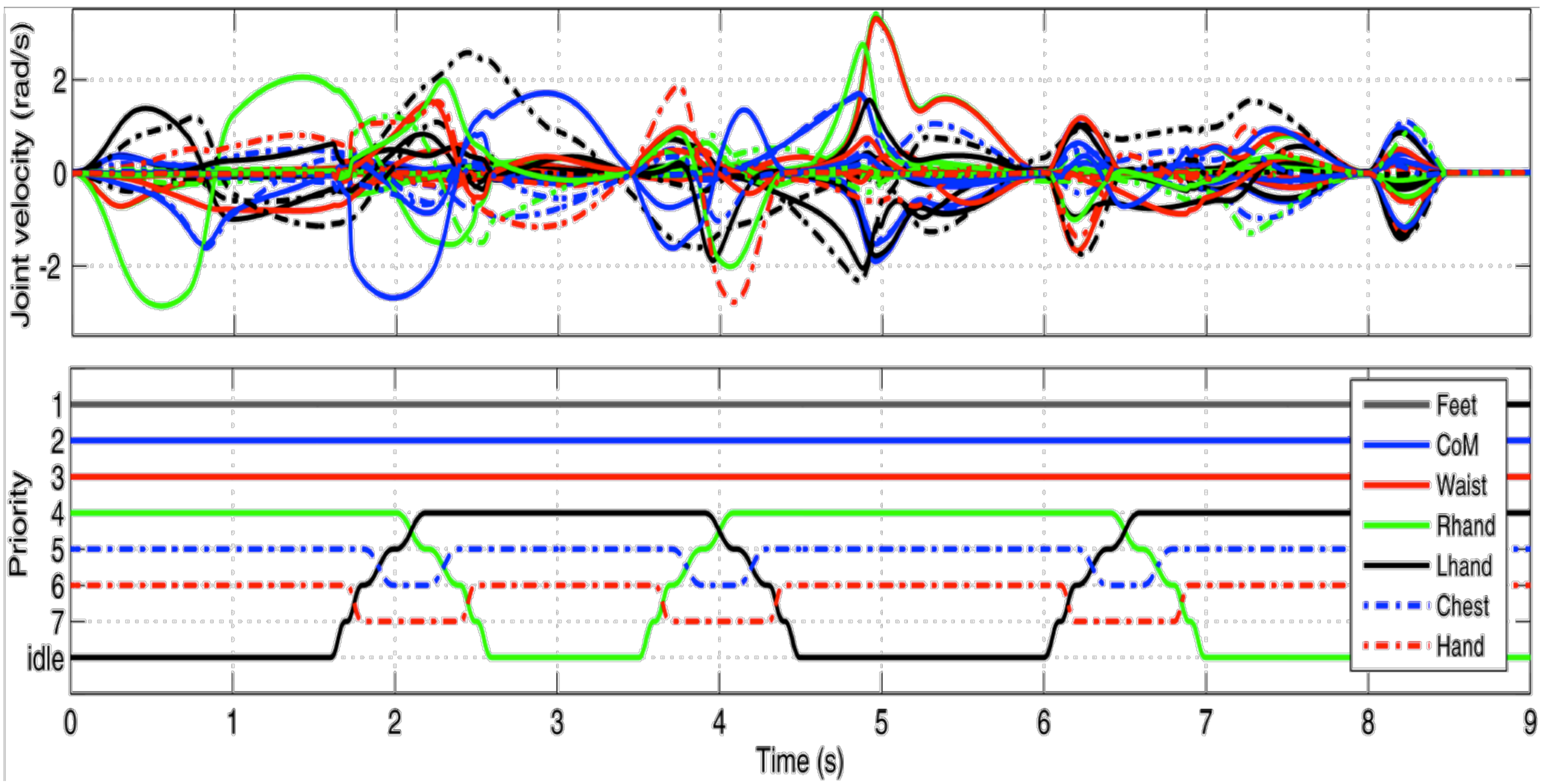
$$SoT_0 = \begin{cases} \dot{\mathbf{e}}_B(\mathbf{q},\alpha_B(t)) \\ \dot{\mathbf{e}}_E(\mathbf{q},\alpha_E(t)) \\ \dot{\mathbf{e}}_D(\mathbf{q},\alpha_D(t)) \end{cases}$$

- The transitions to be performed and the time at which they should occur.

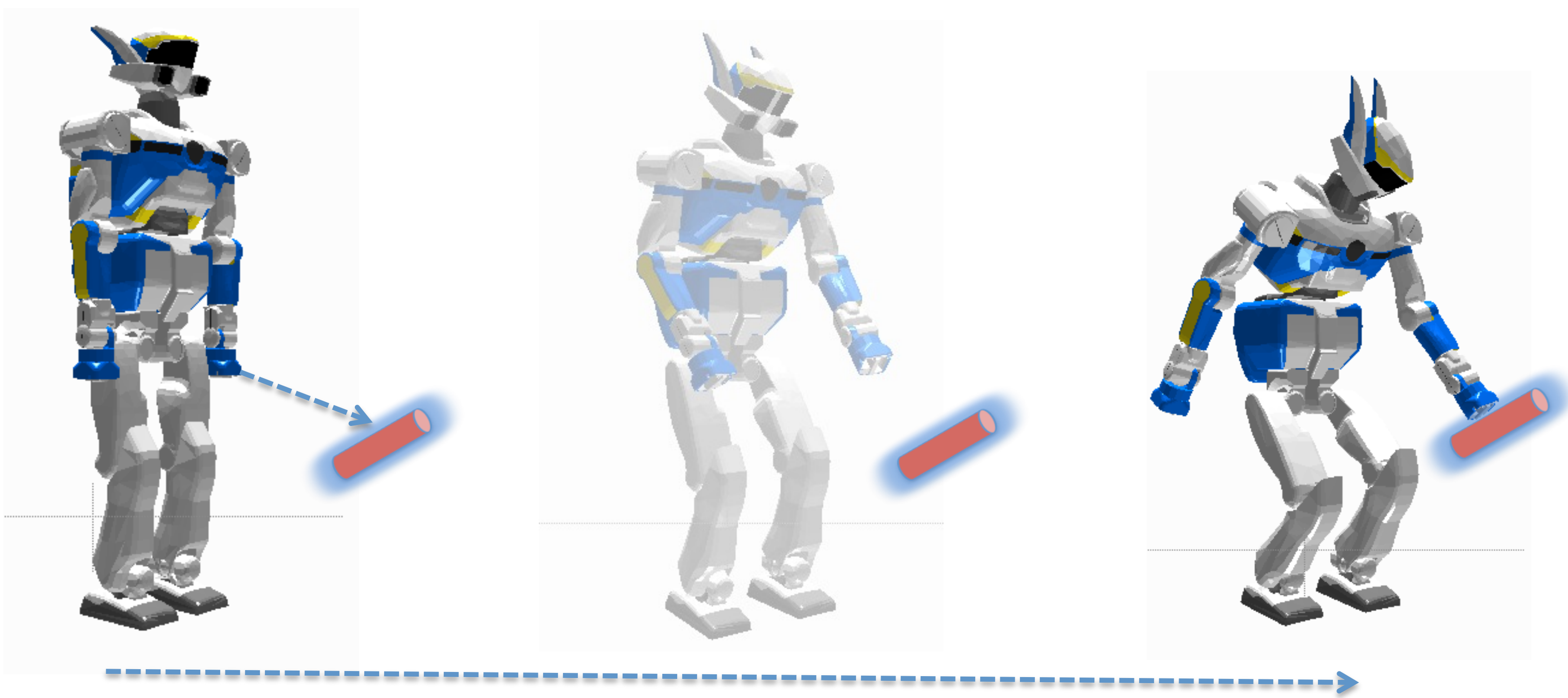
$$tt_k(t) = swap(\dot{\mathbf{e}}_B(\mathbf{q},\alpha_B(t)), \dot{\mathbf{e}}_E(\mathbf{q},\alpha_E(t)), t_{tr})$$

Output:

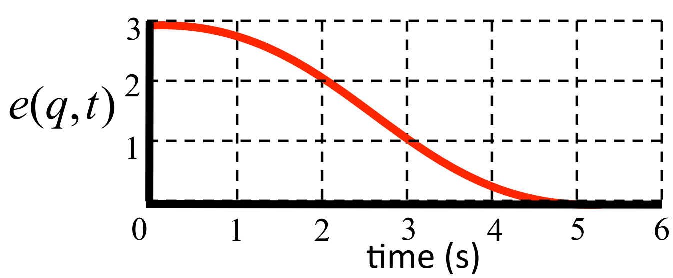
- The feasible velocity trajectories according to the input requirements.



Whole body control based on terminal attractors



5 seconds to reach the object



For handling DTC of tasks we propose the following task error behavior

$$\dot{\boldsymbol{e}}(t) = -\alpha(t)\boldsymbol{e}$$

such that

$$\alpha(t) = \frac{\dot{\xi}(t)}{1 - \xi(t) + \delta}$$

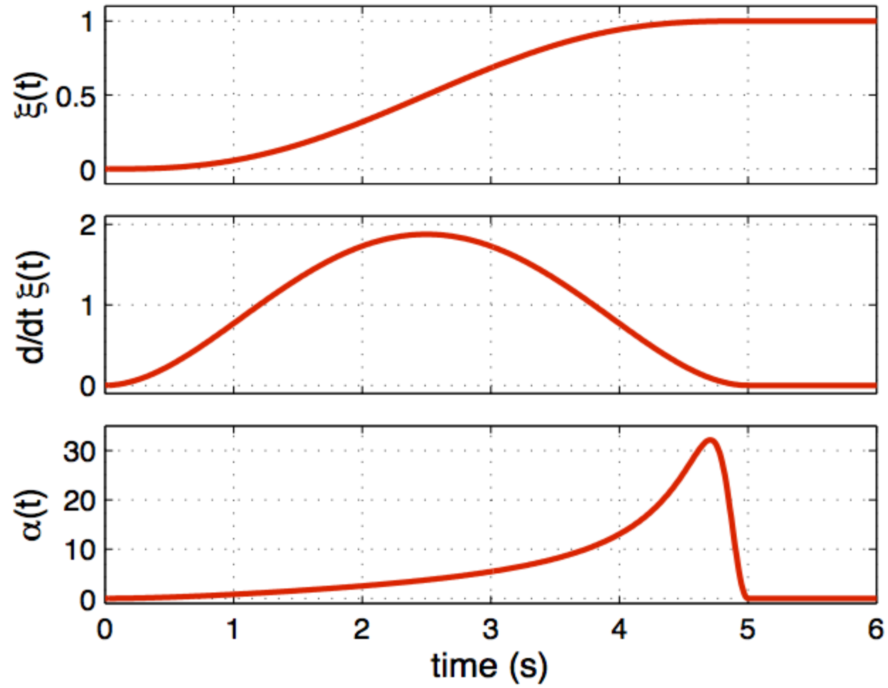
is a time base generator (TBG) gain and its solution is

$$\boldsymbol{e} = \boldsymbol{e}(t_0)\left[1 - \xi(t) + \delta\right]$$

The rate of convergence can be modified

G. Jarquín, G. Arechavaleta and V. Parra-Vega,"Time Parametrization of Prioritized Inverse Kinematics Based on Terminal Attractors" IEEE/RSJ IROS 2011

Characteristics of TBG



1. The discontinuity occurred at $t = t_0$ is avoided.

2. The time derivative of the TBG at $t = t_0$ and $t = t_f$ is 0.

3. $\xi(t)$, $\dot{\xi}(t)$ and $\alpha(t)$ reach their final values at the desired convergence time.

G. Jarquín, G. Arechavaleta and V. Parra-Vega,"Time Parametrization of Prioritized Inverse Kinematics Based on Terminal Attractors" IEEE/RSJ IROS 2011

In the HIK formalism, the *k* task is projected onto the null-space of the *k-1* one

$$\dot{\boldsymbol{q}}_k = \dot{\boldsymbol{q}}_{k-1} + (J_k P_{k-1})^+ (\dot{\boldsymbol{e}}_k - J_k \dot{\boldsymbol{q}}_{k-1})$$

$$\dot{\boldsymbol{q}}_1 = J_1^+ \dot{\boldsymbol{e}}_1$$

where

$$P_1 = I_n - J_1^+ J_1$$

$$P_k = P_{k-1} - (J_k P_{k-1})^+ J_k P_{k-1}$$

This solution is analog to solve the constrained Quadratic Program (QP):

$$\min_{\dot{\boldsymbol{q}}_k, \boldsymbol{w}_k} \frac{1}{2} \|\boldsymbol{w}_k\|^2$$

$$s.\ t. \quad J_k \dot{\boldsymbol{q}}_k = \dot{\boldsymbol{e}}_k + \boldsymbol{w}_k$$

$$\bar{J}_{k-1} \dot{\boldsymbol{q}}_k = \bar{\dot{\boldsymbol{e}}}_{k-1} + \bar{\boldsymbol{w}}^*_{k-1}$$

with $\bar{X}_{k-1} = \begin{bmatrix} X_1 \\ \vdots \\ X_{k-1} \end{bmatrix}$

B. Siciliano and J.-J. Slotine , "A general framework for managing multiple tasks in highly redundant robotic systems". *IEEE ICAR, 1991*
P. Baerlocher and R. Bulic, "An inverse kinematics architecture enforcing an arbitrary number of strict priority levels". *IJCG, 2004*
A. Escande, N. Mansard, P.-B. Wieber, "Hierarchical quadratic programming: Fast online humanoid-robot motion generation". *IJRR, 2014*

A Stack of Tasks (SoT) is a set of active tasks ordered with decreasing priority

$$SoT = \begin{cases} \dot{\boldsymbol{e}}_A(\boldsymbol{q}) \\ \dot{\boldsymbol{e}}_B(\boldsymbol{q}) \\ \vdots \\ \dot{\boldsymbol{e}}_p(\boldsymbol{q}) \end{cases}$$

Decreasing priority level

Characteristics of the task priority framework

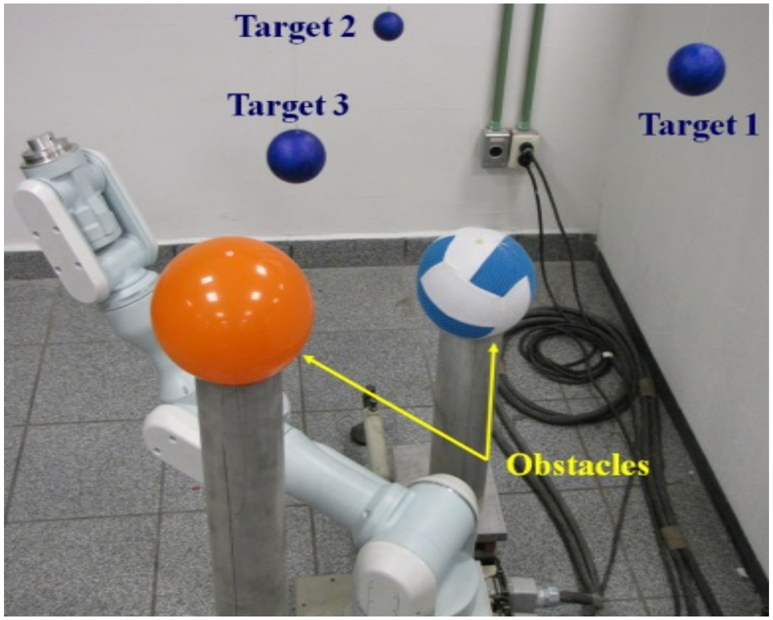
1. Multiple task managing on redundant systems.
2. Solutions may be used as input reference trajectories for dynamic controllers
3. Recursive solution.

Examples of HIK schemes

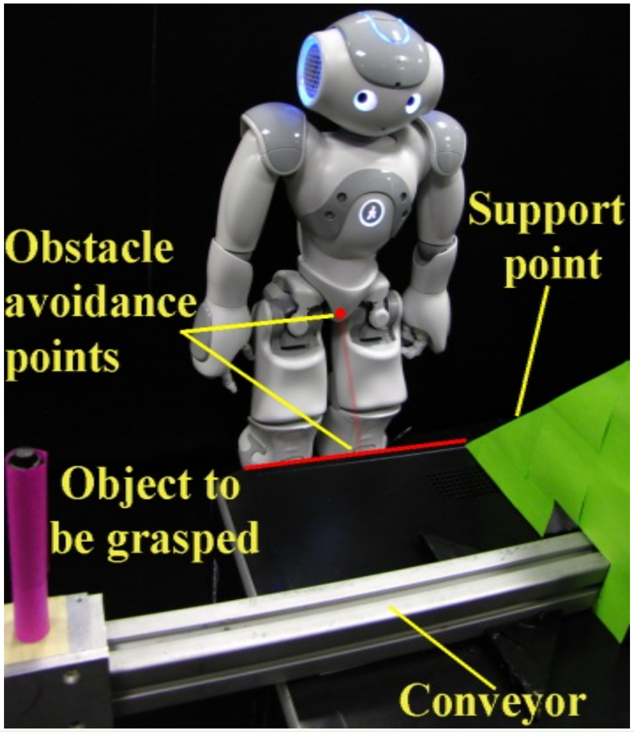
- [P. Baerlocher and R. Boulic, 2001] (least squares)
- [A. Escande et al, 2010] (Complete Orthogonal Decomposition)
- [O. Kanoun et al, 2011] (mixed QP at any priority level)
- [O. Kanoun 2011] (QR decompositions at each level of hierarchy)

N. Mansard and F. Chaumette, "Task sequencing for high-level sensor-based control". I*EEE T-RO,* 2007
H. Hanafusa, T. Yoshikawa and Y. Nakamura, "Analysis and control of articulated robot with redundancy". *IFAC,* 1981.

Experiments with a redundant robot PA-10 and a humanoid robot NAO.



- A NAO humanoid grasps an object in 5 seconds.



- A PA-10 robot sequentially reaches 3 targets at a desired time while avoiding static obstacles

G. Jarquín, G. Arechavaleta and V. Parra-Vega, "Time Parametrization of Prioritized Inverse Kinematics Based on Terminal Attractors" *IEEE/RSJ IROS 2011*
G. Jarquín, G. Arechavaleta and V. Parra-Vega, "Continuous Kinematic Control with Terminal Attractors for Handling Task Transitions of Redundant Robots". *IEEE ICRA 2013.*

Task Transitions:

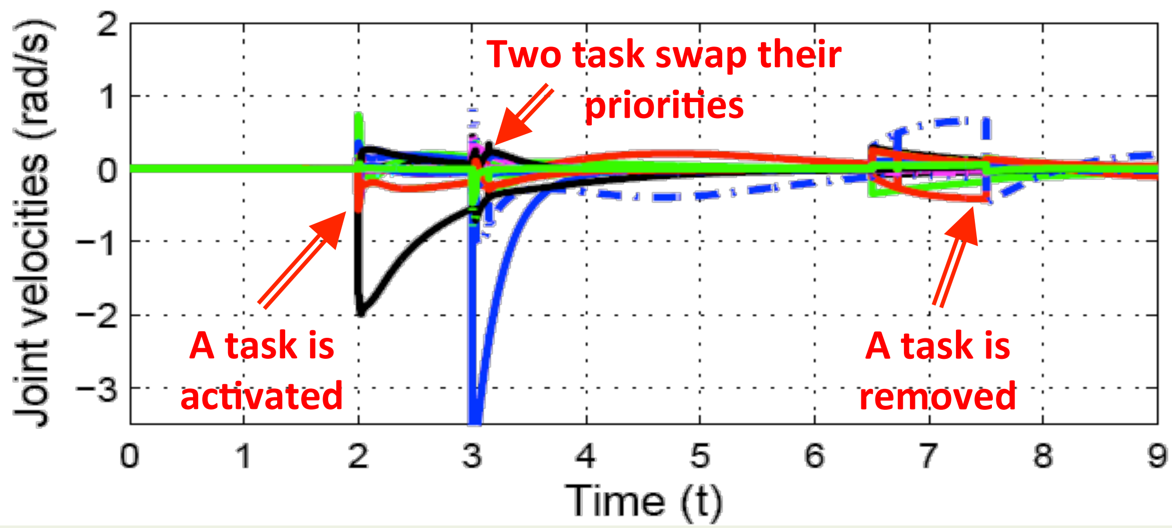
A task is removed or inserted

$$SoT = \begin{cases} \dot{e}_A(q) \\ \dot{e}_B(q) \\ \vdots \\ \dot{e}_p(q) \end{cases} \Rightarrow \begin{cases} \dot{e}_B(q) \\ \dot{e}_C(q) \\ \vdots \\ \dot{e}_p(q) \end{cases}$$

Two tasks swap their priorities

$$SoT = \begin{cases} \dot{e}_A(q) \\ \dot{e}_B(q) \\ \vdots \\ \dot{e}_p(q) \end{cases} \Rightarrow \begin{cases} \dot{e}_B(q) \\ \dot{e}_A(q) \\ \vdots \\ \dot{e}_p(q) \end{cases}$$

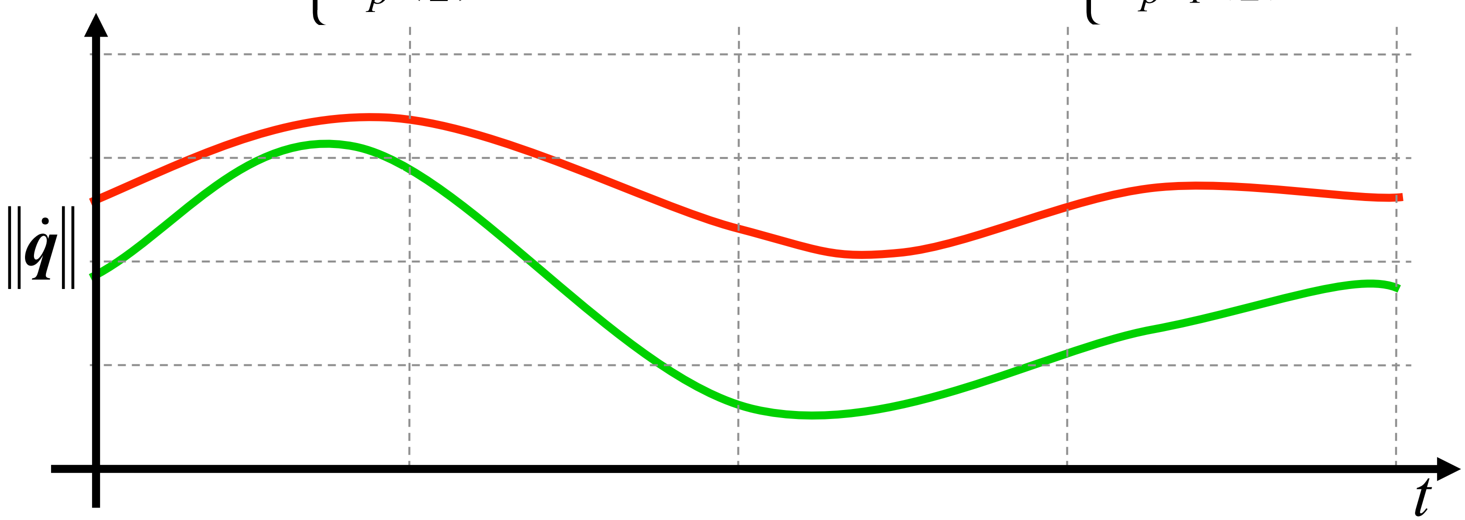
A task transition performed at motion execution may produce a discontinuity



F. Keith, P.-B. Wieber, N. Mansard, and A. Kheddar, "Analysis of the discontinuities in prioritized task-space control under discreet task scheduling operations" IROS 2011

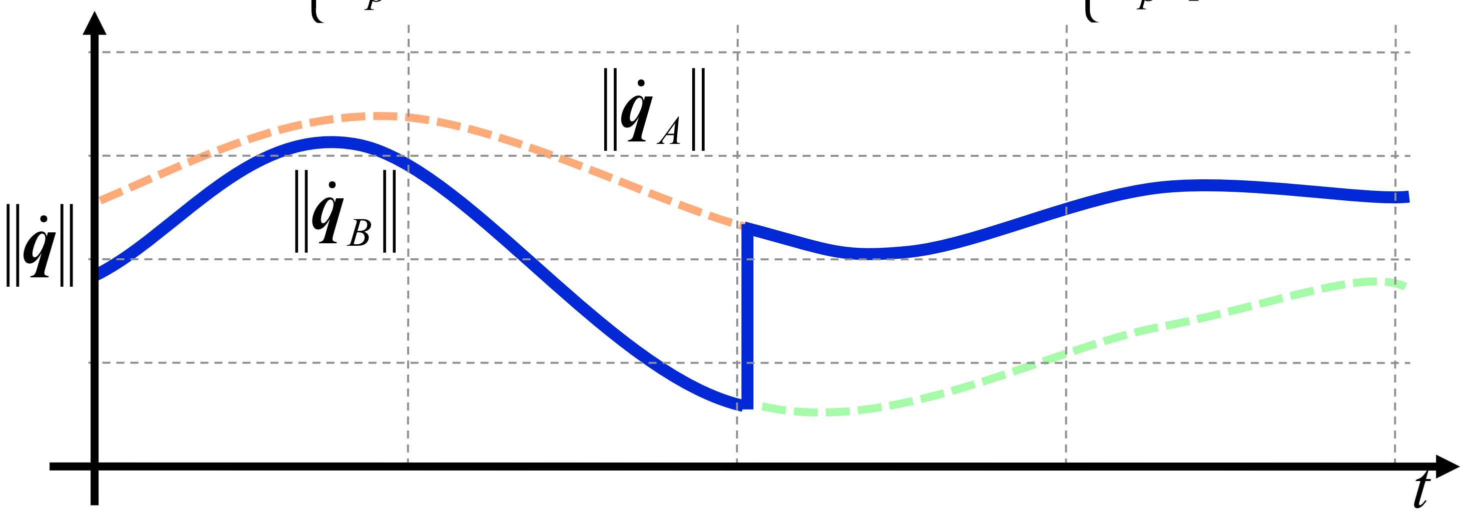
The discontinuity arises because, in general, two SoT produce different control signals.

$$SoT_A = \begin{cases} \dot{\boldsymbol{e}}_A(\boldsymbol{q}) \\ \dot{\boldsymbol{e}}_B(\boldsymbol{q}) \\ \dot{\boldsymbol{e}}_C(\boldsymbol{q}) \\ \dot{\boldsymbol{e}}_D(\boldsymbol{q}) \\ \vdots \\ \dot{\boldsymbol{e}}_p(\boldsymbol{q}) \end{cases} \qquad SoT_B = \begin{cases} \dot{\boldsymbol{e}}_C(\boldsymbol{q}) \\ \dot{\boldsymbol{e}}_A(\boldsymbol{q}) \\ \dot{\boldsymbol{e}}_D(\boldsymbol{q}) \\ \dot{\boldsymbol{e}}_E(\boldsymbol{q}) \\ \vdots \\ \dot{\boldsymbol{e}}_{p-1}(\boldsymbol{q}) \end{cases}$$



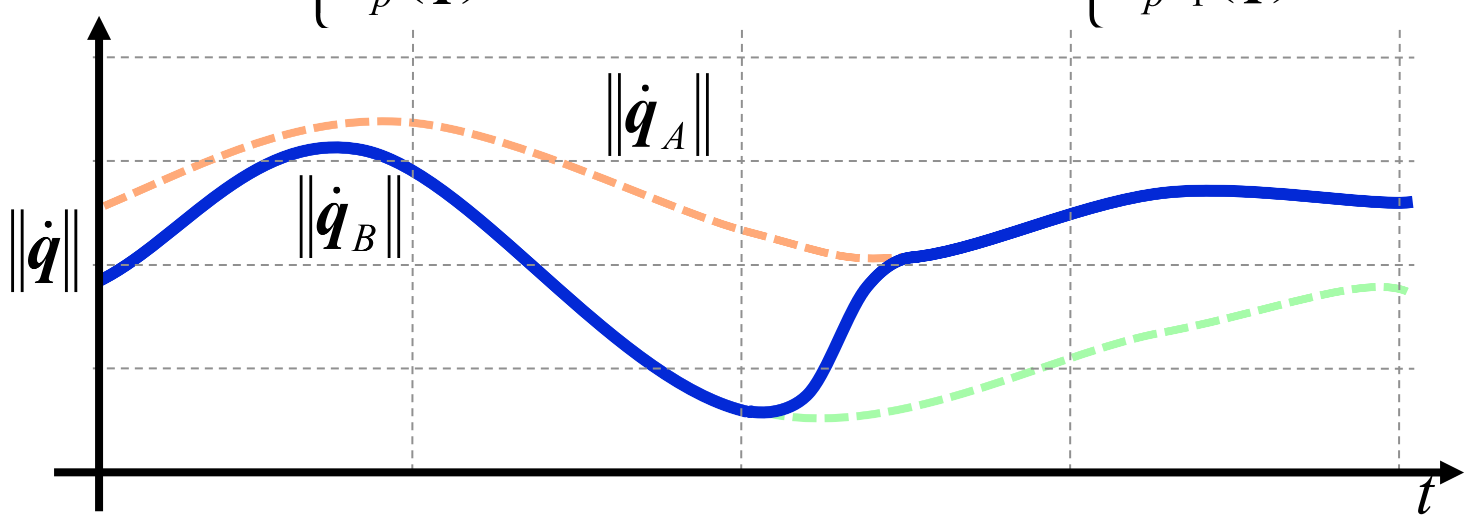
A discontinuity is produced when the SoT_B is replaced by the SoT_A

$$SoT_A = \begin{cases} \dot{e}_A(q) \\ \dot{e}_B(q) \\ \dot{e}_C(q) \\ \dot{e}_D(q) \\ \vdots \\ \dot{e}_p(q) \end{cases} \qquad SoT_B = \begin{cases} \dot{e}_C(q) \\ \dot{e}_A(q) \\ \dot{e}_D(q) \\ \dot{e}_E(q) \\ \vdots \\ \dot{e}_{p-1}(q) \end{cases}$$



The objective is then to smooth the transition

$$SoT_A = \begin{cases} \dot{\boldsymbol{e}}_A(\boldsymbol{q}) \\ \dot{\boldsymbol{e}}_B(\boldsymbol{q}) \\ \dot{\boldsymbol{e}}_C(\boldsymbol{q}) \\ \dot{\boldsymbol{e}}_D(\boldsymbol{q}) \\ \vdots \\ \dot{\boldsymbol{e}}_p(\boldsymbol{q}) \end{cases} \qquad SoT_B = \begin{cases} \dot{\boldsymbol{e}}_C(\boldsymbol{q}) \\ \dot{\boldsymbol{e}}_A(\boldsymbol{q}) \\ \dot{\boldsymbol{e}}_D(\boldsymbol{q}) \\ \dot{\boldsymbol{e}}_E(\boldsymbol{q}) \\ \vdots \\ \dot{\boldsymbol{e}}_{p-1}(\boldsymbol{q}) \end{cases}$$



Our strategy is based on the results reported in [Van Loan, 1985] which states that:

$$\min_{\dot{q}_k, {}^A w_k} \frac{1}{2} \left\| {}^A w_k \right\|^2$$

$$\begin{aligned} s.\ t.\quad {}^A J_k \dot{q}_k &= {}^A \dot{e}_k + {}^A w_k \\ {}^B J_{k-1} \dot{q}_k &= {}^B \dot{e}_{k-1} + {}^B w^*_{k-1} \\ \bar{J}_{k-2} \dot{q}_k &= \bar{\dot{e}}_{k-2} + \bar{w}^*_{k-2} \end{aligned}$$

$$\approx$$

$$\min_{\dot{q}_k, {}^A w_k, {}^B w_k} \frac{1}{2} \left\| \begin{bmatrix} {}^A w \\ \beta\, {}^B w \end{bmatrix}_k \right\|^2$$

$$\begin{aligned} s.\ t.\quad \begin{bmatrix} {}^A J \\ \beta\, {}^B J \end{bmatrix}_k \dot{q}_k &= \begin{bmatrix} {}^A \dot{e} \\ \beta\, {}^B \dot{e} \end{bmatrix}_k + \begin{bmatrix} {}^A w \\ \beta\, {}^B w \end{bmatrix}_k \\ \bar{J}_{k-1} \dot{q}_k &= \bar{\dot{e}}_{k-1} + \bar{w}^*_{k-1} \end{aligned}$$

where β regulates the weight of task B over task A such that if β is large enough the quadratic program on the left hand is equivalent to the right one.

C. Van Loan "On the Method of Weighting for Equality-Constrained Least-Squares Problems", SIAM J. Numer. Anal.,1985

The idea is to introduce a transition period where the weights of the tasks in transition smoothly change the priority order:

Before the Swap

$$\min_{\dot{q}_k, {}^A w_k} \frac{1}{2} \left\| {}^A w_k \right\|^2$$

$s.t.$

$${}^A J_k \dot{q}_k = {}^A \dot{e}_k + {}^A w_k$$

$${}^B J_{k-1} \dot{q}_k = {}^B \dot{e}_{k-1} + {}^B w^*_{k-1}$$

$$\bar{J}_{k-2} \dot{q}_k = \bar{\dot{e}}_{k-2} + \bar{w}^*_{k-2}$$

Transition period

$$\min_{\dot{q}_k, {}^A w_k, {}^B w_k} \frac{1}{2} \left\| \begin{bmatrix} \gamma_1 {}^A w \\ \gamma_2 {}^B w \end{bmatrix}_k \right\|^2$$

$s.t.$

$$\begin{bmatrix} \gamma_1 {}^A J \\ \gamma_2 {}^B J \end{bmatrix}_k \dot{q}_k = \begin{bmatrix} \gamma_1 {}^A \dot{e} \\ \gamma_2 {}^B \dot{e} \end{bmatrix}_k + \begin{bmatrix} \gamma_1 {}^A w \\ \gamma_2 {}^B w \end{bmatrix}_k$$

$$\bar{J}_{k-1} \dot{q}_k = \bar{\dot{e}}_{k-1} + \bar{w}^*_{k-1}$$

After the Swap

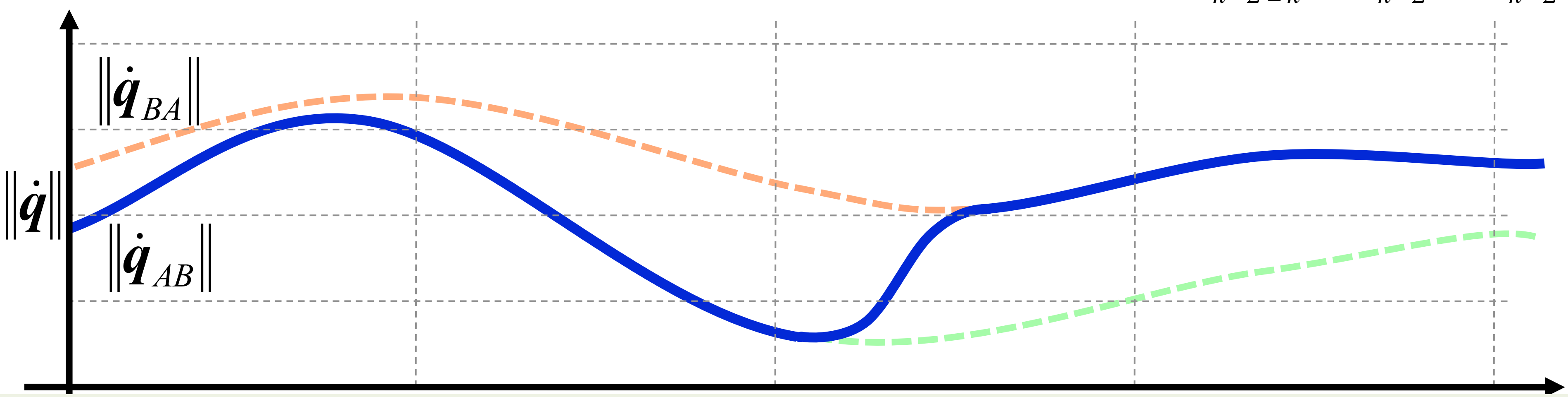
$$\min_{\dot{q}_k, {}^B w_k} \frac{1}{2} \left\| {}^B w_k \right\|^2$$

$s.t.$

$${}^B J_k \dot{q}_k = {}^B \dot{e}_k + {}^B w_k$$

$${}^A J_{k-1} \dot{q}_k = {}^A \dot{e}_{k-1} + {}^A w^*_{k-1}$$

$$\bar{J}_{k-2} \dot{q}_k = \bar{\dot{e}}_{k-2} + \bar{w}^*_{k-2}$$



G. Jarquín, A. Escande, G. Arechavaleta, T. Moulard, E. Yoshida and V. Parra-Vega,"Real-Time Smooth Task Transitions for Hierarchical Inverse Kinematics" ICHR 2013

Our strategy allows to use a damping factor to avoid high velocities near singularities since no task is shadowed by the damping factor during the transition.

$$\min_{\dot{q}_k, {}^A w_k, {}^B w_k} \frac{1}{2} \left\| \begin{bmatrix} \gamma_1 {}^A w \\ \gamma_2 {}^B w \end{bmatrix}_k \right\|^2 + \delta \|\dot{q}\|^2$$

$$\forall \quad 0 < \delta << 1$$

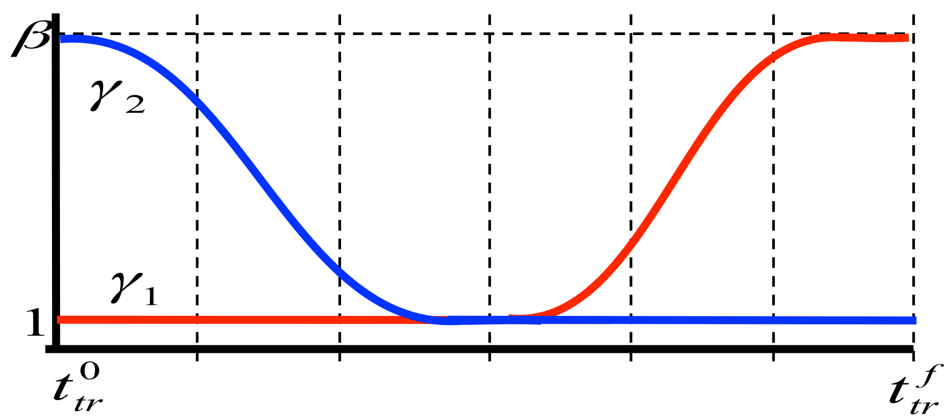
$s.t.$

$$\begin{bmatrix} \gamma_1 {}^A J \\ \gamma_2 {}^B J \end{bmatrix}_k \dot{q}_k = \begin{bmatrix} \gamma_1 {}^A \dot{e} \\ \gamma_2 {}^B \dot{e} \end{bmatrix}_k + \begin{bmatrix} \gamma_1 {}^A w \\ \gamma_2 {}^B w \end{bmatrix}_k$$

$$\bar{J}_{k-1} \dot{q}_k = \bar{\dot{e}}_{k-1} + \bar{w}^*_{k-1}$$

Note that:

$$\gamma_1, \gamma_2 \geq 1 \ \forall \ t_{tr}^0 \leq t \leq t_{tr}^f$$



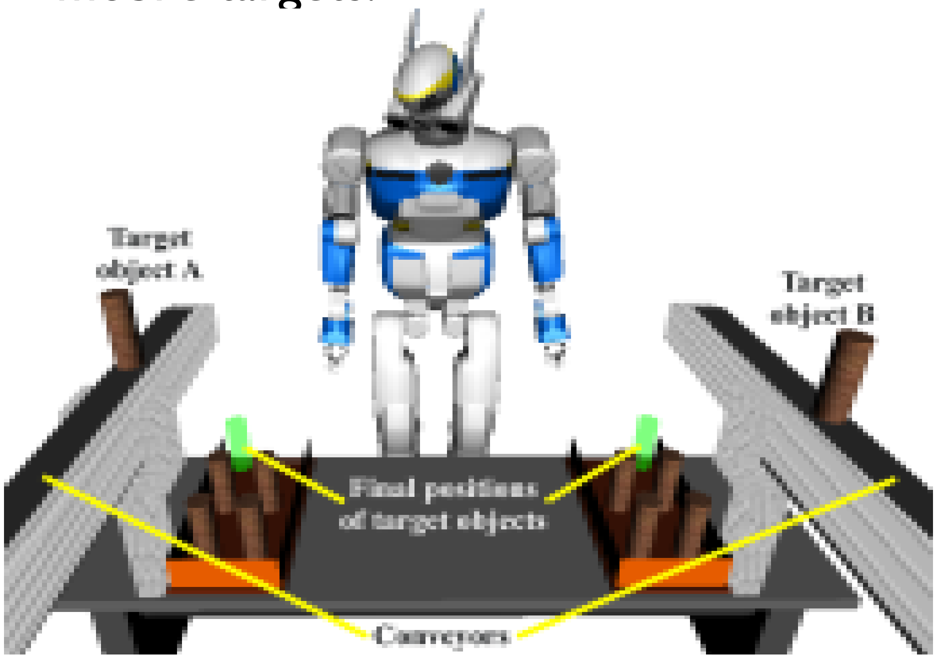
G. Jarquín, A. Escande, G. Arechavaleta, T. Moulard, E. Yoshida and V. Parra-Vega,"Real-Time Smooth Task Transitions for Hierarchical Inverse Kinematics" ICHR 2013

Main features of our strategy:

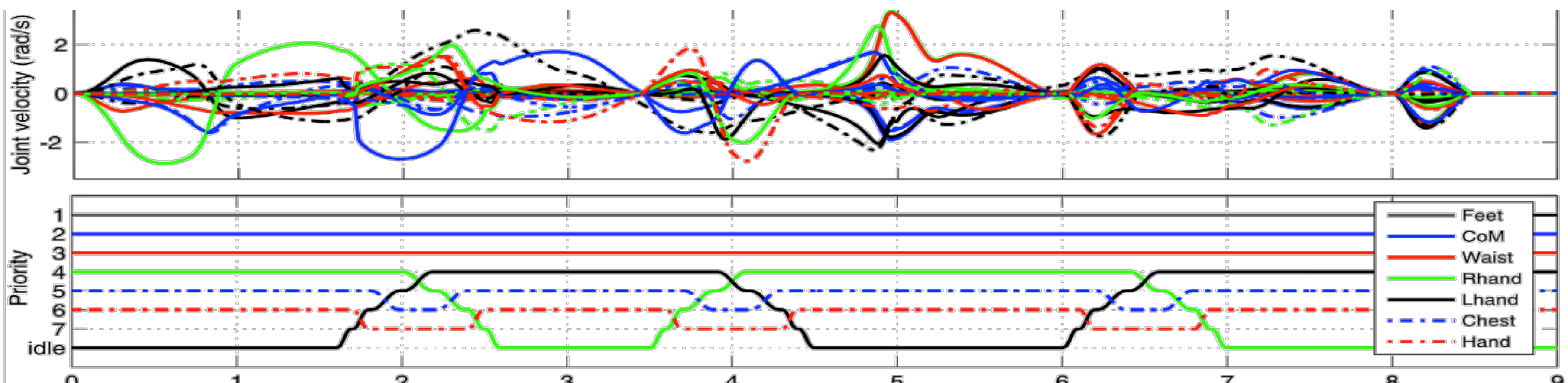
1. The basic operation is the priority swap of two tasks in consecutive priority levels.
2. Insertion and removal of tasks is performed by a sequence of swaps
3. It is not necessary to modify the HIK solver to implement the strategy.
4. The computational cost is not increased, then our strategy is adequate to be implemented in real-time applications with real humanoid robots.
5. It is possible to use the classical damping factor to avoid high velocities near algorithmic and task singularities.

G. Jarquín, A. Escande, G. Arechavaleta, T. Moulard, E. Yoshida and V. Parra-Vega,"Real-Time Smooth Task Transitions for Hierarchical Inverse Kinematics" ICHR 2013

Simulation and experiment with a humanoid robot HRP-2 in an environment with mobile targets.



- Targets on conveyors must be reached before they fall.
- After reach one target, it should be placed in the boxes on the desk
- Task transitions must be executed in order to provide reaching tasks with higher priority than placing tasks.



CONCLUDING REMARKS

- We introduced a strategy to handle smooth task transitions between a set of prioritized tasks in real-time.
- In order to apply our method, it is not needed any modification of the HIK solver.
- The strategy is based in a mechanism to smoothly swap the priorities of two tasks in adjacent priority levels.
- Insertion and removal of tasks is achieved by a set of sequential swaps.
- We provided a simulation and a experiment with a humanoid robot HRP-2 performing a complex grasping task with mobile targets.

Thank you for your attention.

Questions?