FP7-600716

Whole-Body Compliant Dynamical Contacts in Cognitive Humanoids

D5.2
Validation scenario 2: balancing on feet while performing goal directed actions

Editor(s)	Francesco Nori
Responsible Partner	IIT
Affiliations	[1] IIT
Status-Version:	Draft-1.0
Date:	Feb. 28, 2015
EC Distribution:	Consortium
Project Number:	600716
Project Title:	Whole-Body Compliant Dynamical Contacts in Cognitive Humanoids

Title of Deliverable:	Validation scenario 2: balancing on feet while performing goal directed actions
Date of delivery to the EC:	28/2/2015

Workpackage responsible for the Deliverable	deliv WP5
Editor(s):	Francesco Nori
Contributor(s):	Francesco Nori, Andrea Del Prete, Daniele Pucci, Serena Ivaldi, Vincent Padois, Francesco Romano, Jorhabib Eljaik, Silvio Traversaro
Reviewer(s):	
Approved by:	All Partners

Abstract	In the present deliverable we discuss the technical details and choices for the implementation of on the year-2 validation scenario of the CoDyCo project. This validation scenario consists on balancing with multiple rigid contact points while performing a goal directed action. Contact and trajectory planning is not part of the scenario. The final validation consists in controlling the iCub across the following sequence of control tasks. Multiple (force and position) tasks will be handled with strict priorities by adopting the theoretical framework known as stack of tasks [Del Prete, 2013]. First the robot is standing on two feet and, in order to maintain this posture, a controller takes care of controlling the left and right foot position and forces exchanged with the ground. Second, the robot establishes a different contact distribution which is necessary to lean forward and to reach a far distant object. Third, the robot performs a goal directed action exploiting the new contact distribution.
Keyword List:	Multiple, rigid, contacts, control, stability, tracking, forces, torques, reach, goal-directed.

Document Revision History

Version	Date	Description	Author
First draft	19 Feb 2015	In this version we simply write down a few considerations on the second year validation scenario as discussed after the mid-year CoDyCo meeting in Lubljana.	Francesco Nori

Table of Contents

1 Introduction

Similarly to the first year, the second year CoDyCo validation scenario deals with iCub balancing on multiple rigid contacts with the novelty of goal directed actions. The theoretical framework for dealing with these kind of situations has been extensively explored in a number of papers [Aghili, 2005, Righetti et al., 2013]. Taking advantage of our experience on the topic [Del Prete, 2013] the second year validation scenario will be implemented in the Task Space Inverse Dynamics (TSID) framework and in particular we will consider its application to floating base kinematic chains. The iCub will be torque controlled and the controller will assume that desired torques are exactly executed by a lower level torque control. Dynamics will be computed with a custom library, iDynTree[1], built on top of KDL[2]. Desired joint torques will be computed by the TSID algorithm given a set of tasks and their relative priorities. The definition of tasks and priorities is described in details in the following sections.

2 Executive Summary

The deliverable is organized as follows. Section 3 gives an high level presentation of the validation scenario to be presented at the second year review meeting. Section 4.1 discusses the numerical technique (TSID) used to implement the validation scenario as a prioritization of concurrent tasks. Section 4.2 discusses the set of control tasks that will be implemented in order to perform the validation scenario. Their sequencing in the form of a finite state machine is discussed in Section 4.3 and issues related to task switching discussed in Section 4.4. Priorities between tasks are summarized in Section 4.6.

3 Second Year Scenario Validation

The second year scenario aims at validating on the iCub[3] the theoretical results of the consortium in performing the task of balancing on multiple rigid contacts while performing goal directed actions. As clearly stated in the CoDyCo proposal the validation should not involve any planning and therefore the sequence of movements and trajectories will be predefined and kept constant across trials. During the mid-year meeting held in Paris, the sequence of movements was discussed and an agreement was found across the entire consortium. The scenario will begin with the iCub standing on two feet in front of an object that offers the possibility of additional contacts (e.g. a table where to place both hands). After some time the robot will start a movement of both hands towards the additional contact. As soon as the hands detect the additional contacts with the artificial skin, the iCub will start regulating the interaction forces at both hands trying to maintain a comfortable and stable posture. Finally, the robot will exploit the additional contacts to perform a goal directed movement which was not possible without the aforementioned additional contacts.

[1] `http://wiki.icub.org/codyco/dox/html/group__iDynTree.html`

[2] `http://www.orocos.org/kdl`

[3] Implementation is foreseen on the iCub platforms currently available at IIT and UPMC.

4 Second year scenario implementation

Recently we proposed a numerical technique [Del Prete, 2013] for solving the problem of controlling multiple concurrent tasks on floating base robots. Considered tasks include the control of contact forces and multiple motion tasks. Tasks are ordered according to a priority structure, with force tasks at the highest priority. The proposed solution, named task space inverse dynamics (TSID), is presented in the following section.

4.1 Task Space Inverse Dynamics (TSID)

Let's first recall how the force control problem is solved in the TSID framework in the context of floating base manipulators [Del Prete, 2013]. The framework computes the joint torques to match as close as possible a desired vector of forces at the contacts (1a) compatible with the system dynamics (1b) and with the contact constraints (1c):

$$\boldsymbol{\tau}^* = \text{arg} \min_{\boldsymbol{\tau} \in \mathbb{R}^n} \|\boldsymbol{f} - \boldsymbol{f}^*\|^2 \tag{1a}$$

$$s.t. \quad M\ddot{\boldsymbol{q}} + h - J_c^T \boldsymbol{f} = S^T \boldsymbol{\tau} \tag{1b}$$

$$J_c \ddot{\boldsymbol{q}} + \dot{J}_c \dot{\boldsymbol{q}} = 0 \tag{1c}$$

$$J_i \ddot{\boldsymbol{q}} + \dot{J}_i \dot{\boldsymbol{q}} = \ddot{\boldsymbol{x}}_i^*, \qquad i = 1, \ldots, N-1 \tag{1d}$$

where $\boldsymbol{q}$ is a vector of generalized coordinates describing the robot pose (includes both $\boldsymbol{q}_j$ and the floating base position), $\boldsymbol{\tau}$ is the vector of joint torques, M is the mass matrix, h contains both gravitational and Coriolis terms, S is the selection matrix which takes into account the fact that we do not have a direct control over the floating base variables, $\boldsymbol{f}$ is the contact spatial vector, $\boldsymbol{f}^*$ is its desired value and J_c is the contact Jacobian.

In the TSID formalism, the idea is to solve the force control task (represented as in (1a)) at the highest priority and to exploit the nullspace of this primary control task to perform $N-1$ motion tasks at lower priorities (represented as in (1d)). These tasks (indexed with $i = 1, \ldots,$ $N-1$) are all represented as the problem of tracking a given reference acceleration $\ddot{\boldsymbol{x}}_i^*$ for a variable $\boldsymbol{x}_i$ differentially linked to $\boldsymbol{q}$ by the Jacobian J_i, i.e. $\dot{\boldsymbol{x}}_i = J_i \dot{\boldsymbol{q}}$ and $\ddot{\boldsymbol{x}}_i = J_i \ddot{\boldsymbol{q}} + \dot{J}_i \dot{\boldsymbol{q}}$. Assuming that the force task has maximum priority the solution is the represented by the following expression:

$$\boldsymbol{\tau}^* = -(J_c \bar{S})^\top \boldsymbol{f}^* + N_j^{-1} \ddot{\boldsymbol{q}}_1^* + \bar{S}^\top h, \tag{2}$$

where $\bar{S} = M^{-1} S^\top \left(S M^{-1} S^\top \right)^{-1}$ and the term $\ddot{\boldsymbol{q}}_1^*$ is computed solving the following recursion for $i = N, \ldots, 1$:

$$\begin{aligned} \ddot{\boldsymbol{q}}_i =& \ddot{\boldsymbol{q}}_{i+1} + (J_i \bar{S} N_{p(i)})^\dagger + (\ddot{\boldsymbol{x}}_i^* - \dot{J}_i \dot{\boldsymbol{q}} + J_i (U^T M_b^{-1} (h_b - J_{cb}^T f) - \bar{S} \ddot{\boldsymbol{q}}_{i+1})) \\ N_{p(i)} =& N_{p(i+1)} - (J_{i+1} \bar{S} N_{p(i+1)})^\dagger J_{i+1} \bar{S} N_{p(i+1)}, \end{aligned} \tag{3}$$

which is initialized setting $\ddot{q}_{N+1} = 0$, $N_{p(N)} = I$, $J_N = J_c$ and $\ddot{\boldsymbol{x}}_N = 0$.

4.2 Set of admissible tasks

The CoDyCo second year scenario is implemented within the TSID framework which requires the definition of a suitable set of position ($\ddot{\boldsymbol{x}}_i^*$) and force ($\boldsymbol{f}^*$) tasks and their relative priority.

The set of admissible tasks is quite flexible also considering the flexibility of the underlying software libraries[4]. Nevertheless we list here a set of possible tasks, which we will use as a reference in the following sections. Quantities are defined as with a notation similar to the one used in [Featherstone, 2008]: $\boldsymbol{v}$ indicates a spatial velocity (a single vector for representing linear and angular velocities), $\boldsymbol{a}$ indicates a spatial acceleration (a single vector for representing linear and angular accelerations), $\boldsymbol{f}$ indicates a spatial force (a single vector for forces and torques), the index $i = 0, 1, \ldots, N_B - 1$ is used to reference the N_B rigid bodies representing the iCub body chain (0 being defined as the pelvis rigid link), the index w is used to represent the world reference frame, the superscripts and subscripts ^{lh}, ^{rh}, ^{lf}, ^{rf} indicate reference frames rigidly attached to the left hand, right hand, left foot and right foot respectively, the superscript ^{i} indicates the reference frame attached to the i-th rigid body, ${}^{k}\boldsymbol{X}_i$ represents the rigid roto-translation from the reference frame i to the reference frame k, $\boldsymbol{q}_j$ represent the position of the iCub joints.

Tasks will be thrown out of the following set of admissible tasks. For each task T_i we specify the reference values ($\ddot{\boldsymbol{x}}_i$ or $\boldsymbol{f}^*$) and associated Jacobians (J_i).

- $T^{rh}_{\boldsymbol{X}}$: right hand linear and angular accelerations.

$$\begin{aligned} \ddot{\boldsymbol{x}}_i^* &: \quad {}^{rh}\ddot{\boldsymbol{x}}_w =^{rh} \ddot{\boldsymbol{x}}_w^*, \quad {}^{rh}\dot{\boldsymbol{\omega}}_w = {}^{rh}\dot{\boldsymbol{\omega}}_w^*; \\ J_i &: \quad J_{rh}; \end{aligned}$$

- $T^{lh}_{\boldsymbol{X}}$: right hand linear and angular accelerations.

$$\begin{aligned} \ddot{\boldsymbol{x}}_i^* &: \quad {}^{lh}\ddot{\boldsymbol{x}}_w =^{lh} \ddot{\boldsymbol{x}}_w^*, \quad {}^{lh}\dot{\boldsymbol{\omega}}_w = {}^{lh}\dot{\boldsymbol{\omega}}_w^*; \\ J_i &: \quad J_{lh}; \end{aligned}$$

- $T^{com}_{\boldsymbol{X}}$: center of mass acceleration. Move the center of mass with the following acceleration:

$$\begin{aligned} \ddot{\boldsymbol{x}}_i^* &: \quad \ddot{\boldsymbol{x}}_{com} = \ddot{\boldsymbol{x}}_{com}^*; \\ J_i &: \quad J_{com}; \end{aligned}$$

- $T^{rf}_{\boldsymbol{f}}$: right foot force. Regulate the right foot interaction force to a predefined value:

$$\begin{aligned} \ddot{\boldsymbol{f}}_i^* &: \quad \boldsymbol{f}_{rf} = \boldsymbol{f}_{rf}^*; \\ J_i &: \quad J_{rf}; \end{aligned}$$

- $T^{lf}_{\boldsymbol{f}}$: left foot force. Regulate the left foot interaction force to a predefined value:

$$\begin{aligned} \ddot{\boldsymbol{f}}_i^* &: \quad \boldsymbol{f}_{lf} = \boldsymbol{f}_{lf}^*; \\ J_i &: \quad J_{lf}; \end{aligned}$$

[4] `http://wiki.icub.org/codyco/dox/html/group__iDynTree.html`

- $T_{\boldsymbol{f}}^{rh}$: right hand force. Regulate the right hand interaction force to a predefined value:

$$\begin{array}{rcl} \ddot{\boldsymbol{f}}_i^* & : & \boldsymbol{f}_{rh} = \boldsymbol{f}_{rh}^*; \\ J_i & : & J_{rh}; \end{array}$$

- $T_{\boldsymbol{f}}^{lh}$: left hand force. Regulate the left hand interaction force to a predefined value:

$$\begin{array}{rcl} \ddot{\boldsymbol{f}}_i^* & : & \boldsymbol{f}_{lh} = \boldsymbol{f}_{lh}^*; \\ J_i & : & J_{lh}; \end{array}$$

- T^q : postural task. Maintain the robot joints $\boldsymbol{q}_j$ close to certain reference posture $\boldsymbol{q}_j^*$:

$$\begin{array}{rcl} \ddot{\boldsymbol{x}}_q^* & : & \ddot{\boldsymbol{q}}_j = \ddot{\boldsymbol{q}}_j^*; \\ J_i & : & I. \end{array}$$

4.3 Sequencing of tasks

The set of tasks active at a certain instant of time is regulated by a finite state machine. In particular there are five different states $S_1, \ldots, S_5$ each characterized by a different set of active tasks $\mathcal{S}_1, \ldots, \mathcal{S}_5$.

- S_1 has the following set of active tasks $\mathcal{S}_1 = \left\{T_{\boldsymbol{X}}^{com}, T_{\boldsymbol{f}}^{rf}, T_{\boldsymbol{f}}^{lf}, T^q\right\}$.
- S_2 has the following set of active tasks $\mathcal{S}_2 = \mathcal{S}_1 \cup \left\{T_{\boldsymbol{X}}^{lh}\right\}$.
- S_3 has the following set of active tasks $\mathcal{S}_3 = \mathcal{S}_1 \cup \left\{T_{\boldsymbol{f}}^{lh}\right\}$.
- S_4 has the following set of active tasks $\mathcal{S}_4 = \mathcal{S}_1 \cup \left\{T_{\boldsymbol{f}}^{lh}, T_{\boldsymbol{X}}^{rh}\right\}$.

Transition between states is regulated by the following finite state machine where the sets C_{rh} and C_{lh} represent the number of taxels (tactile elements) activated at time t.

$$\begin{tikzcd} \text{start} \arrow[r] & S_1 \arrow[loop above, "t_0 \le t \le t_1"] \arrow[r, "t > t_1"] & S_2 \arrow[loop above, "C_{rh} = \emptyset"] \arrow[r, "C_{rh} \neq \emptyset"] & S_3 \arrow[r, "\|\boldsymbol{f}_{rh} - \boldsymbol{f}_{rh}^*\| \cong 0"] & S_4 \end{tikzcd}$$

In practice, at start ($t = t_0$) the robot is in state S_1 and controls the position and orientation of the pelvis with respect to a world reference frame ($T_{\boldsymbol{X}}^{com}$). Balance is maintained by controlling also the forces and torques exchanged at the right ($T_{\boldsymbol{f}}^{rf}$) and left foot ($T_{\boldsymbol{f}}^{lf}$). A postural tasks (T^q) guarantees that the system is not drifting and maintains a posture close to a reference posture. After a predefined amount ($t = t_1$) the system switches to the state S_2 by adding two tasks to the set of active tasks: a control of the position and orientation of the right hand ($T_{\boldsymbol{X}}^{rh}$). Successive transitions are triggered by the tactile sensors. If no contact are detected on the right hand ($C_{rh} = \emptyset$) the system remains in S_2. A transition from S_2 to

S_3 is performed when the system detects a contact on the right hand ($C_{rh} \neq \emptyset$): in this state the active tasks are the same active in S_2 but the right hand position and orientation control ($T^{rh}_{\boldsymbol{X}}$) is replaced with the force control ($T^{rh}_{\boldsymbol{f}}$). Once a stable force control is achieved or more precisely when the right hand contact force $\boldsymbol{f}_{rh}$ is close enough to the desired contact force ($\|\boldsymbol{f}_{rh} - \boldsymbol{f}^*_{rh}\| \cong 0$) a transition to S_4 is triggered, thus activating the cartesian control for the left hand ($T^{lh}_{\boldsymbol{X}}$) towards the desired target.

4.4 Task references

In this section we discussed how to compute the task references:

$$\boldsymbol{f}^*_{rh}, \quad \boldsymbol{f}^*_{rf}, \quad \boldsymbol{f}^*_{lf}, \quad {}^{rh}\ddot{\boldsymbol{x}}^*_w, \quad {}^{rh}\dot{\boldsymbol{\omega}}^*_w, \quad {}^{lh}\ddot{\boldsymbol{x}}^*_w, \quad {}^{lh}\dot{\boldsymbol{\omega}}^*_w, \quad \ddot{\boldsymbol{x}}^*_{com}.$$

The proposed definitions assume that for each reference we have an associated desired trajectory which is pre-specified:

$$\boldsymbol{f}^d_{rh}(t), \quad \boldsymbol{f}^d_{rf}(t), \quad \boldsymbol{f}^d_{lf}(t), \quad {}^{rh}\boldsymbol{x}^d_w(t), \quad {}^{rh}R^d_w(t), \quad {}^{lh}\boldsymbol{x}^d_w(t), \quad {}^{lh}R^d_w(t), \quad \boldsymbol{x}^d_{com}(t).$$

The definition of these desired trajectories is discussed in Section 4.5. In this section we discuss how task references are constructed given the task trajectories. Let's first discuss how the center of mass reference acceleration $\ddot{\boldsymbol{x}}^*_{com}$ is generated at each instant of time. Given the desired trajectory $\boldsymbol{x}^d_{com}(t)$ trajectory we define:

$$\ddot{\boldsymbol{x}}^*_{com}(t) = \ddot{\boldsymbol{x}}^d_{com}(t) + K^{com}_d \left(\dot{\boldsymbol{x}}^d_{com}(t) - \dot{\boldsymbol{x}}_{com}\right) + K^{com}_p \left(\boldsymbol{x}^d_{com}(t) - \boldsymbol{x}_{com}\right), \tag{4}$$

where K^{com}_p and K^{com}_d are arbitrary positive definite matrices. Similarly, the force reference trajectories $\boldsymbol{f}^*_\cdot(t)$ are obtained from a desired force trajectory $\boldsymbol{f}^d_\cdot(t)$:

$$\boldsymbol{f}^*_\cdot(t) = \boldsymbol{f}^d_\cdot(t) + K^f_i \int_0^t \left(\boldsymbol{f}^d_\cdot(\tau) - \boldsymbol{f}_\cdot(\tau)\right) d\tau, \tag{5}$$

where again K^f_i is an arbitrary positive definite matrix. It is slightly more complicated to define the hands acceleration reference trajectories $\ddot{\boldsymbol{x}}^*_w(t)$, $\dot{\boldsymbol{\omega}}^*_w(t)$. The linear acceleration trajectory $\ddot{\boldsymbol{x}}^*_w(t)$ can be handled exactly as it was done for the center of mass position exploiting a suitable desired trajectory $\boldsymbol{x}^d_w(t)$. Angular acceleration trajectories $\dot{\boldsymbol{\omega}}^*_w(t)$ are slightly more complicated in considerations of the fact that they are derived from a reference rotation matrix $R^d_w(t) \in SO(3)$. Within this context, we follow the definition in [Siciliano et al., 2009] and define an orientation error $\boldsymbol{e}_O$ between the desired pose $R^d_w(t) \in SO(3)$ and the current pose $R_w \in SO(3)$ as:

$$\boldsymbol{e}_O = \sin(\theta)\boldsymbol{r},$$

being θ and $\boldsymbol{r}$ the axis-angle representation of the error rotation $R^d_w(t)R^\top_w$. We have:

$$\dot{\boldsymbol{e}}_O = -L\omega + L^\top \omega_d,$$

with:

$$L = -\frac{1}{2}\left(S(\boldsymbol{n}_d)S(\boldsymbol{n}) + S(\boldsymbol{s}_d)S(\boldsymbol{s}) + S(\boldsymbol{a}_d)S(\boldsymbol{a})\right),$$

where $\boldsymbol{n}$, $\boldsymbol{s}$, $\boldsymbol{a}$ (and correspondingly $\boldsymbol{n}_d$, $\boldsymbol{s}_d$, $\boldsymbol{a}_d$) are the columns of R_w, i.e. $R_w = \begin{bmatrix}\boldsymbol{n} & \boldsymbol{s} & \boldsymbol{a}\end{bmatrix}$. Similarly, the second derivative of the angular error is given by:

$$\ddot{\boldsymbol{e}}_O \;=\; -\dot{L}\omega - L\dot{\omega} + \dot{L}^\top\omega_d + L^\top\dot{\omega}_d.$$

where the derivative of L can be expressed as follows (computations omitted):

$$\dot{L} = -\frac{1}{2}\left(S(\omega\times\mathbf{n}_d)S(\mathbf{n}) + S(\mathbf{n}_d)S(\omega\times\mathbf{n}) + S(\omega\times\mathbf{s}_d)S(\mathbf{s}) + \right.$$
$$\left. + \; S(\mathbf{s}_d)S(\omega\times\mathbf{s}) + S(\omega\times\mathbf{a}_d)S(\mathbf{a}) + S(\mathbf{a}_d)S(\omega\times\mathbf{a})\right).$$

The dynamics $\ddot{\boldsymbol{e}}_O + K_d^O\dot{\boldsymbol{e}}_O + K_p^O\boldsymbol{e}_O = 0$ can be rewritten as:

$$\dot{\omega} \;=\; L^{-1}\left(-\dot{L}\omega + \dot{L}^\top\omega_d + L^\top\dot{\omega}_d + K_d^O\dot{\boldsymbol{e}}_O + K_p^O\boldsymbol{e}_O\right),$$

which eventually gives us the following rule for choosing the right and left hand reference trajectories:

$$\begin{aligned}
{}^{rh}\dot{\boldsymbol{\omega}}_w^*(t) &= L^{-1}\left(-\dot{L}^{rh}\boldsymbol{\omega}_w + \dot{L}^{\top rh}\boldsymbol{\omega}_w^d + L^{\top rh}\dot{\boldsymbol{\omega}}_w^d + K_d^O\dot{\boldsymbol{e}}_O^{rh} + K_p^O\boldsymbol{e}_O^{rh}\right),\\
{}^{lh}\dot{\boldsymbol{\omega}}_w^*(t) &= L^{-1}\left(-\dot{L}^{lh}\boldsymbol{\omega}_w + \dot{L}^{\top lh}\boldsymbol{\omega}_w^d + L^{\top lh}\dot{\boldsymbol{\omega}}_w^d + K_d^O\dot{\boldsymbol{e}}_O^{lh} + K_p^O\boldsymbol{e}_O^{lh}\right).
\end{aligned}$$

4.5 Task trajectories

In this section we discuss how to compute the tasks reference trajectories:

$$\boldsymbol{f}_{rh}^d(t),\quad \boldsymbol{f}_{rf}^d(t),\quad \boldsymbol{f}_{lf}^d(t),\quad {}^{rh}\boldsymbol{x}_w^d(t),\quad {}^{rh}R_w^d(t),\quad {}^{lh}\boldsymbol{x}_w^d(t),\quad {}^{lh}R_w^d(t),\quad \boldsymbol{x}_{com}^d(t).$$

At this stage of the CoDyCo project, there is no interest in the planning movements but just in controlling them. Therefore, it is out of the scope of the current deliverable to find desired trajectories compatible with the system dynamics, joint limits, motor torque limits and so on. Hands ($T_{\boldsymbol{X}}^{rh}$ or $T_{\boldsymbol{X}}^{com}$) and center of mass ($T_{\boldsymbol{X}}^{rh}$) trajectories are predefined and assumed to be chosen according to the specific environment chosen for the validation scenario. In particular, the center of mass trajectory $\boldsymbol{x}_{com}^d(t)$ is left as a free parameter to be chosen during the validation. Hand trajectories ${}^{rh}\boldsymbol{x}_w^d(t), {}^{rh}R_w^d(t), {}^{lh}\boldsymbol{x}_w^d(t), {}^{lh}R_w^d(t)$ are constructed by specifying only the final posture to be reached: ${}^{rh}\bar{\boldsymbol{x}}_w^d, {}^{rh}\bar{R}_w^d, {}^{lh}\bar{\boldsymbol{x}}_w^d, {}^{lh}\bar{R}_w^d$. This posture is chosen so as to reach the hand contacts or the desired goal position, assumed to be fixed in the world reference frame[5]. Whenever either $T_{\boldsymbol{X}}^{rh}$ or $T_{\boldsymbol{X}}^{lh}$ is activated, a new reference trajectory is instantiated corresponding to an interpolation (e.g. a minimum jerk trajectory) from the current hand position to the contact locations; the trajectory duration is left as an open parameter.

The definition of the interaction force tasks $T_{\boldsymbol{f}}^{rf}$, $T_{\boldsymbol{f}}^{lf}$, $T_{\boldsymbol{f}}^{rh}$ and the associated reference force values $\boldsymbol{f}_{rh}^*$, $\boldsymbol{f}_{rf}^*$, $\boldsymbol{f}_{lf}^*$ requires more care. The motivation resides in the fact that these tasks are intertwined to the center of mass task $T_{\boldsymbol{X}}^{com}$ through the system dynamics. At each instant of time external forces determine the center of mass acceleration according to the following:

$$m\ddot{\boldsymbol{x}}_{com} = f_{rh} + f_{rf} + f_{lf}, \tag{6}$$

[5] In practice final postures are positions on the positions where the hand contact should occur.

being m the total mass of the robot and f_{rh}, f_{rf} and f_{lf} the Cartesian (not the spatial!) forces exchanged by the robot at the contact points. In view of this consideration we need to be careful in choosing reference interaction forces $\boldsymbol{f}^*_{rh}$, $\boldsymbol{f}^*_{rf}$, $\boldsymbol{f}^*_{lf}$ and reference center of mass acceleration $\ddot{\boldsymbol{x}}^*_{com}$. The notation is slightly complicated by the fact that in the different scenario states S_1, ..., S_5 the meaning of $\boldsymbol{f}$ (and consequently $\boldsymbol{f}^*$) in (1) changes. In particular we have:

$$S_1, S_2 : \boldsymbol{f} = \begin{bmatrix} \boldsymbol{f}_{rf} \\ \boldsymbol{f}_{lf} \end{bmatrix}, \quad S_3, S_4 : \boldsymbol{f} = \begin{bmatrix} \boldsymbol{f}_{rh} \\ \boldsymbol{f}_{rf} \\ \boldsymbol{f}_{lf} \end{bmatrix}, \tag{7}$$

where each spatial force has the usual structure $\boldsymbol{f}_i = \left[\mu_i^\top, f_i^\top\right]^\top$ being f_i the Cartesian force and μ_i the Cartesian torque always defined in the link reference frame. In the different states, the following constraints on $\boldsymbol{f}^*$ hold with the definitions below:

- S_1 and S_2: desired force $\boldsymbol{f}^*$ is such that:

$$C_{S_1}\boldsymbol{f}^* = C_{S_2}\boldsymbol{f}^* = f^*_{rf} + f^*_{lf} = m\ddot{\boldsymbol{x}}^*_{com}. \tag{8}$$

- S_3 and S_4: desired force $\boldsymbol{f}^*$ is such that:

$$C_{S_3}\boldsymbol{f}^* = C_{S_4}\boldsymbol{f}^* = f^*_{rh} + f^*_{rf} + f^*_{lf} = m\ddot{\boldsymbol{x}}^*_{com}. \tag{9}$$

where we defined:

$$\begin{aligned} C_{S_1} &= \begin{bmatrix} 0_{3\times3} & I_{3\times3} & 0_{3\times3} & I_{3\times3} \end{bmatrix}, \\ C_{S_2} &= \begin{bmatrix} 0_{3\times3} & I_{3\times3} & 0_{3\times3} & I_{3\times3} \end{bmatrix}, \\ C_{S_3} &= \begin{bmatrix} C_{S_1} & 0_{3\times3} & I_{3\times3} \end{bmatrix}, \\ C_{S_4} &= \begin{bmatrix} C_{S_1} & 0_{3\times3} & I_{3\times3} \end{bmatrix}. \end{aligned}$$

In general these constraints (deriving from the Newton equation) can be completed with additional constraints which derive from the angular momentum equilibrium, but we left this complete formulation out to simplify the notation. The interested reader should refer to [Orin et al., 2013]. Similarly to simplify the notation we neglected the effect of gravitational forces, which should be taken into account when balancing forces. Correct computations can be simply obtained by modifying all equations to include a term $m\boldsymbol{g}$ being $\boldsymbol{g}$ the gravitational acceleration. Otherwise all equations still hold referring to the system "proper acceleration".

In the TSID formalism, tasks $T^{rf}_{\boldsymbol{f}}$, $T^{lf}_{\boldsymbol{f}}$, $T^{rh}_{\boldsymbol{f}}$ can be solved solving (1a)-(1b)-(1c) at the first stage of the hierarchy (i.e. with maximum priority). However, it is worth observing that the solution of (1a)-(1b)-(1c) might be in conflict with the center of mass task $T^{com}_{\boldsymbol{X}}$ if it is specified in (1d) with lower priority with respect to the force task. Therefore, in order to enforce the compatibility of the tasks $T^{rf}_{\boldsymbol{f}}$, $T^{rh}_{\boldsymbol{f}}$, $T^{lh}_{\boldsymbol{f}}$ and $T^{com}_{\boldsymbol{X}}$, we pre-compute a reference force $\boldsymbol{f}^*$ compatible with the reference center of mass acceleration $\boldsymbol{x}^*_{com}$. Even if we impose the above constraints on $\boldsymbol{f}^*$, we do not identify a unique solution. Additional constraints or requirements needs to be imposed in order to properly define $\boldsymbol{f}^*$. In order to simplify this ambiguity, the following problem can be solved when at state S_i:

$$\boldsymbol{f}^* = \text{arg} \min_{\boldsymbol{f}} \|\boldsymbol{f} - \boldsymbol{f}_0\|^2_W \text{ s.t. } C_{S_i}\boldsymbol{f} = m\ddot{\boldsymbol{x}}^*_{com}, \tag{10}$$

where $\| \cdot \|_W$ denotes a norm weighted with the matrix $W = W^\top > 0$. The solution of this optimization is given by:

$$\boldsymbol{f}^* = mC^\dagger_{S_i,W}\ddot{\boldsymbol{x}}^*_{com} + (I - C^\dagger_{S_i,W}C_{S_i})\boldsymbol{f}_0, \tag{11}$$

having defined $C^\dagger_{S_i,W} = \left(C^\top_{S_i}WC_{S_i}\right)^{-1} C^\top_{S_i}W$, the weighted pseudo-inverse of $C_{S_i,W}$. The solution of this optimization gives a set of desired forces $\boldsymbol{f}^*$ which are compatible with the desired center of mass acceleration $\ddot{\boldsymbol{x}}^*_{com}$.

It was worth noticing that in the above optimization constraints involve only the Cartesian forces while the norm to be minimized is referred to the spatial forces, thus including both Cartesian forces and torques. As a consequence, (10) naturally imposes a solution with null Cartesian torques at all contacts. It is worth noting here that all Cartesian forces and torques are defined in a reference frame attached to the rigid link on which the contact is active. Additionally, if these reference frames are properly chosen within the contact support polygon, (10) naturally induces the center of pressure condition (sometimes called also the ZMP condition) for all the contacts which guarantees the stability of the contact itself.

Once a compatible $\boldsymbol{f}^*$ is computed solving (10), tasks $T^{rf}_{\boldsymbol{f}}$, $T^{lf}_{\boldsymbol{f}}$, $T^{rh}_{\boldsymbol{f}}$ and $T^{com}_{\boldsymbol{X}}$ are simultaneously computed solving the associated optimization (1a)-(1b)-(1c) which has maximum priority in the hierarchy. Alternatively, a possible solution in a single step consists in solving the following problem:

$$\boldsymbol{\tau}^* = \text{arg} \min_{\boldsymbol{\tau}\in\mathbb{R}^n} \|\boldsymbol{f}\|^2_W \tag{12a}$$

$$s.t. \quad M\ddot{\boldsymbol{q}} + h - J_c^T\boldsymbol{f} = S^T\boldsymbol{\tau} \tag{12b}$$

$$J_c\ddot{\boldsymbol{q}} + \dot{J}_c\dot{\boldsymbol{q}} = 0 \tag{12c}$$

$$J_{com}\ddot{\boldsymbol{q}} + \dot{J}_{com}\dot{\boldsymbol{q}} = \ddot{\boldsymbol{x}}^*_{com}. \tag{12d}$$

The solution of (12) has strong analogies with the solution of (10) and (1a)-(1b)-(1c) but a complete analysis of the underlying similarities is out of the scope of the present report.

In order to avoid discontinuities in the desired forces to be tracked by the controller we should chose the trajectories $\boldsymbol{f}^*_{rh}(t)$, $\boldsymbol{f}^*_{rf}(t)$, $\boldsymbol{f}^*_{lf}(t)$ so that at the moment of task switching the desired force equal the current one. In order to do so, we simply exploit the fact that at any time, measurements $\boldsymbol{f}$, $\ddot{\boldsymbol{x}}_{com}$ should satisfy the equilibrium condition $C_{S_i}\boldsymbol{f} = m\ddot{\boldsymbol{x}}_{com}$. Therefore, at the switching instant t_s we can simply solve (10) with $\boldsymbol{f}_0 = C_{S_i}\boldsymbol{f}(t_s)$ and $\ddot{\boldsymbol{x}}^*_{com} = \ddot{\boldsymbol{x}}_{com}(t_s)$ to obtain a solution $\boldsymbol{f}^* = \boldsymbol{f}(t_s)$. The values of $\ddot{\boldsymbol{x}}^*_{com}$ and $\boldsymbol{f}_0$ are then obtained as described in Section 4.4 as the output of a PID filter on the difference between desired trajectories $\ddot{\boldsymbol{x}}^d_{com}(t)$, $\boldsymbol{f}^d(t)$ and current values $\ddot{\boldsymbol{x}}_{com}$, $\boldsymbol{f}$. In particular, according the definitions that we gave in (4) and (5) the final reference for forces and accelerations will be:

$$\begin{aligned}
\ddot{\boldsymbol{x}}^*_{com}(t) &= \ddot{\boldsymbol{x}}^d_{com}(t) + K^{com}_d\left(\dot{\boldsymbol{x}}^d_{com}(t) - \dot{\boldsymbol{x}}_{com}\right) + K^{com}_p\left(\boldsymbol{x}^d_{com}(t) - \boldsymbol{x}_{com}\right), \\
\boldsymbol{f}_0(t) &= \boldsymbol{f}^d(t) + K^f_i\int_{t_s}^t \left(\boldsymbol{f}^d(\tau) - \boldsymbol{f}(\tau)\right)d\tau, \\
\boldsymbol{f}^* &= mC^\dagger_{S_i,W}\ddot{\boldsymbol{x}}^*_{com} + (I - C^\dagger_{S_i,W}C_{S_i})\boldsymbol{f}_0.
\end{aligned}$$

At the switching instant t_s we choose $\boldsymbol{f}^d = \boldsymbol{f}$, $\ddot{\boldsymbol{x}}^d_{com} = \ddot{\boldsymbol{x}}_{com}$, $\dot{\boldsymbol{x}}^d_{com} = \dot{\boldsymbol{x}}_{com}$, $\boldsymbol{x}^d_{com} = \boldsymbol{x}_{com}$ so as to obtain $\boldsymbol{f}^* = \boldsymbol{f}$ (remeber that $\boldsymbol{f}$, $\ddot{\boldsymbol{x}}_{com}$ are always compatible). Then we can choose

$\boldsymbol{f}^d(t)$ and $\ddot{\boldsymbol{x}}^d_{com}(t)$ arbitrarily with the guarantee that the resulting $\boldsymbol{f}^*$ and $\ddot{\boldsymbol{x}}^*_{com}$ will always be dynamically consistent.

4.6 Task prioritization

In the TSID framework, control tasks are executed with a lexicographic priority, i.e. tasks with higher priority should be satisfied at the expenses of the tasks with lower priority. In order to represent T_i with a lower lexicographic priority with respect to T_j we use the notation $T_i \prec T_j$. In consideration of what discussed above, tasks force tasks $T^{rf}_{\boldsymbol{f}}$, $T^{lf}_{\boldsymbol{f}}$, $T^{rh}_{\boldsymbol{f}}$ and center of mass tasks $T^{com}_{\boldsymbol{X}}$ are grouped in a unique optimization which we denote $T^{com}_{\boldsymbol{f}}$. In the different states, priorities are handled as follows:

- S_1 and S_2: $T^q \prec T^{com}_{\boldsymbol{f}}$.
- S_2: $T^q \prec T^{rh}_{\boldsymbol{X}} \prec T^{com}_{\boldsymbol{f}}$.
- S_3: $T^q \prec T^{com}_{\boldsymbol{f}}$.
- S_4: $T^q \prec T^{lh}_{\boldsymbol{X}} \prec T^{com}_{\boldsymbol{f}}$.

References

[Aghili, 2005] Aghili, F. (2005). A unified approach for inverse and direct dynamics of constrained multibody systems based on linear projection operator: applications to control and simulation. *Robotics, IEEE Transactions on*, 21(5):834–849.

[Del Prete, 2013] Del Prete, A. (2013). *Control of Contact Forces using Whole-Body Force and Tactile Sensors: Theory and Implementation on the iCub Humanoid Robot*. PhD thesis, Istituto Italiano di Tecnologia.

[Featherstone, 2008] Featherstone, R. (2008). *Rigid Body Dynamics Algorithms*. Springer.

[Orin et al., 2013] Orin, D. E., Goswami, A., and Lee, S.-H. (2013). Centroidal dynamics of a humanoid robot. *Auton. Robots*, 35(2-3):161–176.

[Righetti et al., 2013] Righetti, L., Buchli, J., Mistry, M., Kalakrishnan, M., and Schaal, S. (2013). Optimal distribution of contact forces with inverse dynamics control. *The International Journal of Robotics Research*.

[Siciliano et al., 2009] Siciliano, B., Sciavicco, L., and Villani, L. (2009). *Robotics: Modelling, Planning and Control*. Advanced Textbooks in Control and Signal Processing. Springer.