



FP7-600716

Whole-Body Compliant Dynamical Contacts in Cognitive Humanoids

D3.3

Local solver in compliant and unforeseen cases

Editor(s)	Vincent Padois ^{1,2} , Daniele Pucci ³
Responsible Partner	UPMC
Affiliations	¹ Sorbonne Universités, UPMC Paris 06, UMR 7222, Institut des Systèmes Intelligents et de Robotique (ISIR), F-75005, Paris, France. ² CNRS, UMR 7222, Institut des Systèmes Intelligents et de Robotique (ISIR), F-75005, Paris, France. ³ Istituto Italiano di Tecnologia (IIT) Via Morego 30, Genoa, Italy.
Status-Version:	Final-1.0
Date:	Feb. 28, 2017
EC Distribution:	Consortium
Project Number:	600716
Project Title:	Whole-Body Compliant Dynamical Contacts in Cognitive Humanoids

Title of Deliverable:	Local solver in compliant and unforeseen cases
Date of delivery to the EC:	28/2/2017

Workpackage responsible for the Deliverable	WP3
Editor(s):	Vincent Padois, Daniele Pucci
Contributor(s):	Morteza Azzad, Mingxing Liu, Michael Mistry, Francesco Nori, Vincent Padois, Daniele Pucci, Francesco Romano, Silvio Traversaro
Reviewer(s):	
Approved by:	All Partners
Abstract	<p>This deliverable provides an overview of the whole-body control strategies proposed within the framework of the CoDyCo project for balancing by means of contacts and unforeseen robot interactions. We first recall the general structure of the whole-body controller used in the CoDyCo project. This controller is written as a quadratic multi-objective optimization problem under linear constraints, and priorities between the control objectives can be dealt with through strict or soft hierarchy. The control architecture is then modified in order to deal with unforeseen cases. More precisely, we assume that a user can interact with the robot at specific locations, namely, the robot forearms. Then, we propose a passivity-based modification to the controller in order to take into account the robot interaction according to its desired motion. This deliverable is strongly related to Deliverable 5.5 [?] which discusses the technical details and choices for the implementation of the year-4 validation scenario.</p>
Keyword List:	Whole-body controllers, Non-rigid contacts, Multi-contacts

Document Revision History

Version	Date	Description	Author
v. 0.1	Jan. 19, 2016	Initial creation of the file	Vincent Padois
v. 0.9	Feb. 25, 2016	Final version	Vincent Padois
v. 1.0	Feb. 27, 2016	Proofread version	Vincent Padois

Table of Contents

1	Introduction	5
2	General structure of the whole-body controller in the CoDyCo project	6
2.1	Formulation	7
2.2	Dealing with multiple objectives	8
2.3	Direct and two-stage whole-body control approaches	9
3	Balancing on compliant contacts: model-free control approach	10
4	Conclusion	10

Index of Figures

1 Introduction

Under-actuated robots, such as free-floating humanoid robots, usually need to make contacts with their environments to achieve some goal directed whole-body movements. Most researches on whole-body control assume that the environment of the robot is rigid. This means that no adaptation to the environment compliance is needed for controllers. In reality, there is no completely rigid surface even if in practice we can assume a surface to be rigid if it is stiff enough (i.e. deflection is negligible). Unfortunately, many objects in human environments cannot be considered as stiff enough and their compliance has to be accounted for (e.g. a soft cushion, a sofa, a yoga carpet). Indeed, a controller that does not take into account the compliant properties of the contact material may not be sufficient: the compliance of the contact has to be considered by the controller, otherwise the robot may fail to properly balance and fall over. For example, pushing too weakly against a compliant object may not provide the robot with enough reaction forces to support its whole-body tasks. The problem is complex as the rigidity of the object in contact is unknown a priori to robotic controllers, which is usually the case in many scenarios.

The humanoid whole-body control problem has been addressed by different types of whole-body controllers, using analytical approaches [?, ?, ?], constrained quadratic programming [?, ?, ?, ?, ?], or a mixture of them [?, ?]. These controllers are either developed for rigid environments, or validated only in rigid contact scenarios. In general, a valid set of contact forces during whole-body task control can be found by solving a multi-objective problem with a set of elementary task objectives as well as constraints, such as whole-body dynamics, friction cone constraints for non-sliding contacts, and linear complementarity conditions [?, ?, ?, ?], which implies zero relative motions between two bodies in contact when normal contact force is non-negative. In the case of rigid contact with static environment, the linear complementarity condition implies two constraints: (i) the motion of the contact point is zero and (ii) the contact force along the normal to the contact surface is non-negative. The zero motion constraint may not necessarily be true in the case of non-rigid contacts, since the velocities or accelerations of contact points may be non-zero, although the relative motion between the two contact points remains zero. In this case, hybrid control methods [?] that control forces and motions in orthogonal directions are not applicable. Therefore, the controller should take into account the dynamic relation between the contact point position and the contact force, rather than just control the contact force alone.

Such physical interaction dynamics is taken into account in impedance control [?] with the idea of controlling the relation between the contact point motion and the reaction force. Traditional impedance control [?, ?, ?] computes the target impedance of the robot according to the estimated impedance of the environment, which requires high quality measurement of interaction forces. In [?, ?], learning approaches are applied to optimize the robot impedance. Such approaches do not require interaction force sensing and can be adaptable to variable environment impedance. However, the application of such approaches in the context of humanoid balance control with non-rigid contacts is not suitable. First, these methods rely on trajectory-based learning and adaptation algorithms, whereas there is not necessarily a reference motion trajectory for each support contact in the whole-body balancing context

considered here. Furthermore, they need to explore the entire state-action space if a globally optimal solution is to be found, which is impossible for high dimensional robots such as humanoids.

The problem of humanoid balance control with deformable contact support was addressed in [?], which proposed a posture planning approach assuming that the contact material properties are known. Compliant contacts between robots and their environments have been studied by some researchers in other areas such as grasping [?] and animated characters [?]. However, there has not been much research efforts on balancing legged robots on compliant surfaces.

This deliverable provides an overview of the whole-body control strategies proposed within the framework of the CoDyCo project for balancing by means of compliant contacts. It is organized as follows. We first recall the general structure of the whole-body controller used in the CoDyCo project. This controller is written as a quadratic multi-objective optimization problem under linear constraints and priorities between the objectives can be dealt with through strict or soft hierarchy. This controller has to be modified in order to deal with compliant cases and two cases are then distinguished. In the first one, no model of the environment is assumed to be known and an adaptive force regulation task is added to the original “rigid contact” controller in order to account for the compliance of the environment. In the second case, a contact model is supposed to be known and using that knowledge a control strategy is derived.

2 General structure of the whole-body controller in the CoDyCo project

Even though it is not often formulated as such, control of dynamical systems is an optimization problem. Within the framework of the CoDyCo project, the whole-body controller is written as a quadratic multi-objective optimization problem under linear constraints where priorities between the objectives can be dealt with through strict or soft hierarchy. Deliverable 3.1 [?] exposes the reasons why it is preferable to express controllers in this way. The logic behind this choice can be briefly summarized in a very straightforward way:

1. the equation of motion and joint space to task spaces mappings can be written as equalities but they are not sufficient to describe the overall dynamics and physical behaviour of a robot.
2. Indeed, other intrinsic physical constraints have to be accounted for at the joint level as well as in Cartesian space. These constraints do not solely describe relationships between physical quantities but also limits which cannot (control input saturation) or should never be crossed in order to maintain the robot and its environment in proper working conditions.
3. These limits translate into inequalities.
4. Assuming a convex solution space, the optimal solution of the control problem lies at the boundary of the feasible (constraint compliant) solution space.

5. Finding the optimal solution thus boils down to finding the active constraint set, *i.e.* on which boundary it lies.
6. Optimization problem solvers are designed to optimally choose this subset of constraints that should be considered when computing the optimal solution of the control problem.
7. The strong mathematical background in convex optimization is such that optimization based methods mostly outperform analytical methods attempting to heuristically activate constraints.

2.1 Formulation

Based on this formulation choice, the reactive control problem aims at finding at each control instant the actuation torque minimizing $T(\mathbb{X})$, some tasks related function to minimize, given the equation of motion of the multi-body system as well as other equality and inequality constraints. This can be written

$$\tau^* = \underset{\mathbb{X}}{\operatorname{argmin}} \quad T(\mathbb{X}) \quad (1a)$$

$$\text{subject to} \quad M(q)\dot{\nu} + C(q, \nu)\nu + G(q) = B\tau + \underbrace{\sum_{k=1}^{n_c} J_{C_k}^\top(q) f_k}_{J^\top(q)f} \quad (1b)$$

$$A(q, \nu)\mathbb{X} = b(q, \nu) \quad (1c)$$

$$D(q, \nu)\mathbb{X} \leq h(q, \nu) \quad (1d)$$

where:

- $\tau \in \mathbb{R}^n$ is the internal actuation torque with $n + 1$ the number of rigid bodies – called links – connected by n actuated joints with one degree of freedom each.
- $q \in \mathbb{R}^3 \times SO(3) \times \mathbb{R}^n$ is the generalized coordinates that parametrizes the configuration of the free-floating system. q is a triplet composed of the origin and orientation of the base frame expressed in the inertial frame $({}^I p_B, {}^I R_B)$ and the n joint angles q_j .
- $\nu \in \mathbb{R}^{n+6}$ is the system velocity, a triplet concatenating the floating-base twist $({}^I \dot{p}_B, {}^I \omega_B)$ and the joint velocities \dot{q}_j .
- $J(q) = [J_{C_1}^\top(q) \ \dots \ J_{C_k}^\top(q)]^\top$ is the contact Jacobian matrix for all k contact points.
- $f = [f_1^\top \ \dots \ f_k^\top]^\top$ is the vector of external contact wrenches applied by the environment on the links.
- $\mathbb{X} = (\dot{\nu}, \tau, f)$ gathers the dynamic variables of the multi-body system.
- $M \in \mathbb{R}^{(n+6) \times (n+6)}$ is the mass matrix.
- $C \in \mathbb{R}^{(n+6) \times (n+6)}$ is the Coriolis and centrifugal effects matrix.

- $G \in \mathbb{R}^{n+6}$ is the gravity term.
- $B = (0_{n \times 6}, 1_n)^\top$ is a selection matrix.
- $A(q, \nu)\mathbb{X} = b(q, \nu)$ gathers kinematics constraints related to the velocity of the contact points.
- $D(q, \nu)\mathbb{X} \leq h(q, \nu)$ gathers inequality constraints related to joint limits (position and velocity), control input saturation, contact forces (existence and friction limits) and potentially obstacle avoidance.

2.2 Dealing with multiple objectives

The control problem is often multi-objective and the tasks-related function $T(\mathbb{X})$ can actually be written

$$T(\mathbb{X}) = T(\lambda_1, T_1(\mathbb{X}), \lambda_2, T_2(\mathbb{X}), \dots, \lambda_{n_t}, T_{n_t}(\mathbb{X})) \quad (2)$$

where $T_i(\mathbb{X})$ is the i -th task among n_t operational tasks to be achieved with $J_i(q)$ its associated task Jacobian and λ_i its priority level. T_i is generally of three types:

- Operational space acceleration $T_i = J_i(q)\dot{\nu} + \dot{J}_i(q_t, \nu)\nu - \ddot{x}^d$
 - Joint space acceleration $T_i = \dot{\nu} - \dot{\nu}^d$
 - Operational space force $T_i = f_{C_i} - f_{C_i}^d$
- (3)

where \ddot{x}^d , $\dot{\nu}^d$ and $f_{C_i}^d$ are desired Cartesian space acceleration, configuration space acceleration and contact wrench respectively, the desired value itself being the outcome of some higher level control architecture (Proportional–Derivative–Integral regulators and/or Momentum regulators and/or Model Predictive Controllers and/or Trajectory planners providing a feedforward reference, etc).

T_i usually appears in T under the form of a weighted, euclidean norm thus leading to a quadratic cost associated to each task. This quadratic form and the linearity of the constraints allows to resort to the convex optimization techniques, more particularly Linear Quadratic Programs. Then, depending of the type of retained prioritization scheme, the optimization problem 1 can be solved:

1. at once using a single LQP and soft prioritization where T is written as a weighted sum of quadratic costs and where λ s play the role of weights for each task, see [?, ?] for examples;
2. using a cascade of LQPs thus inducing strict prioritization between tasks (in that case λ s are used to defined a lexicographic order), see [?, ?, ?] for examples;
3. using a representation able to convey both soft and strict prioritization, see [?, ?].

2.3 Direct and two-stage whole-body control approaches

In CoDyCo, these three types of prioritization scheme coexist and are used indifferently as they allow to produce very similar types of behaviours¹. A distinction has still to be made between two ways to approach the whole-body control problem.

The first, direct, approach does not separate the problem of computing the contact forces from the one of computing the joint torques. Given some task space desired acceleration (among which a center of mass one), the resolution of the control problem described by Equation (1) directly provides the optimal torques. The contact forces come as a by-product of the problem resolution. This approach has the very nice property of being able to account for the all constraints in a straightforward way. Its drawback is that it hides some of the essence of the balance control problem and while it may be preferable to use such a solution for practical implementation, a two-stage solution may be preferred to ease the analysis of the system (for example in terms of its controllability).

The second, two-stage approach, uses contact forces as intermediate controls and first compute the desired contact forces given the desired acceleration of the center of mass. This can be done using the Newton-Euler equation for the floating-base system, written at the center of mass of the system. It can be written

$$\begin{pmatrix} m(\ddot{x} - g) \\ \dot{H}_\omega \end{pmatrix} = \underbrace{\sum_{k=1}^{n_c} \begin{pmatrix} 1_3 & 0_{3 \times 3} \\ S(p_{c_k} - x) & 1_3 \end{pmatrix} f_k}_{X_{cf}} \quad (4)$$

where m is the total mass of the system, $x \in \mathbb{R}^3$ is the position of the center of mass expressed in the inertial frame, \dot{H}_ω is the derivative of the angular momentum, g is the acceleration induced by gravity expressed in the inertial frame, p_{c_k} is the k -contact point and $S(u) \in \mathbb{R}^{3 \times 3}$ is the skew-symmetric matrix such that $S(u)v = u \times v$, where \times denotes the cross product operator in \mathbb{R}^3 .

Equation (4) plainly displays the relation between contact wrenches and the dynamics of the center of mass and f can be computed based on it in order to achieve some desired center of mass linear acceleration \ddot{x}^d and to maintain the angular momentum null (by choosing $\dot{H}_\omega = 0$). The solution for f can be computed using an LQP as well. This allows to account for inequality constraints related to friction. Given these desired contact forces, the whole-body control problem can then be solved using the general formulation provided by Equation (1). This control approach, often referred as *momentum-based balance controller*, has extensively been used in the recent years [?, ?, ?] and is the one used in the CoDyCo demonstrations.

The next sections describe how this generic controller structure can be exploited and adapted to deal with compliant contacts.

¹A formal comparison of their intrinsic properties is out of the scope of this deliverable but would constitute a work of interest for the community

3 Balancing on compliant contacts: model-free control approach

4 Conclusion

In this deliverable, an overview of the work performed, within the framework of the CoDyCo project, on balancing with compliant contacts is provided. This work is the result of tasks T3.2, T3.3 and T5.3.

The adaptive approach described in [?] has the advantage of providing a way to accommodate compliant contacts without the need for a model. Its major disadvantage lies in the fact that it aims at reaching a state where, after a transient period, compliant contacts are fully compressed and can be used as rigid ones. Unfortunately, in highly compliant environment this is never the case and one actually has to balance with contact points that remain compliant. A second disadvantage is that the proposed algorithm tends to require more forces from the most compliant contact points. This may be critical if the corresponding contacts corresponds to fragile surfaces which may break under high contact forces.

The two other proposed approaches [?], [?] assume that a model of the compliant contact is known. While the approach described in Deliverable 5.3 [?] is less sensitive to the quality of the estimation of this model, the identification problem remains a complex one. At this stage estimation is performed off-line through a tedious procedure. In practical cases where the environment may not be known in advance, online estimation techniques would be needed. However, these methods are still to be developed, in a context where estimating dynamic quantities (for example the floating-base state) for rigid multi-body systems remains a recent research topic [?], [?].