Quadratic Programming for Multirobot and Task-Space Force Control

Karim Bouyarmane, *Member, IEEE*, Kevin Chappellet, Joris Vaillant, and Abderrahmane Kheddar, *Senior Member, IEEE*

***Abstract*—We have extended the task-space multiobjective controllers that write as quadratic programs (QPs) to handle multirobot systems as a single centralized control. The idea is to assemble all the "robots" models and their interaction task constraints into a single QP formulation. By multirobot, we mean that whatever entities a given robot will interact with (solid or articulated systems, actuated, partially or not at all, fixed-base or floating-base), we model them as clusters of robots and the controller computes the state of each cluster as an overall system and their interaction forces in a physically consistent way. By doing this, the tasks specification simplifies substantially. At the heart of the interactions between the systems are the contact forces; methodologies are provided to achieve reliable force tracking by our multirobot QP controller. The approach is assessed by a large panel of experiments on real complex robotic platforms (full-size humanoid, dexterous robotic hand, fixed-base anthropomorphic arm) performing whole-body manipulations, dexterous manipulations, and robot–robot comanipulations of rigid floating objects and articulated mechanisms, such as doors, drawers, boxes, or even smaller mechanisms like a spring-loaded click pen.**

***Index Terms*—Humanoid robot manipulation, manipulation force control, multirobot control, robot–robot comanipulation, task specification.**

I. Introduction

TASK-SPACE sensory control [1] has reached a considerable level of maturity and has diverse implementations in kinematics and inverse dynamics [2], [3]. It has been ported in a large variety of robots, especially redundant ones [4], [5], achieving multiobjective complex tasks under various constraints.

Recent implementations formulate the task-space control as a quadratic program (QP) where multiple objective tasks are ordered by means of a weighted, strict, or hybrid strict-weighted priority. The controller reduces to a QP solver for a problem that is built online, at each control loop, e.g., [6]–[10], and where the tasks are expressed as a part of the QP cost function or part of its constraints, e.g., [11], [12].

In our previous work [13], [14], we have devised a multicontact *planner* that considers robots and objects as multirobot systems. It also gathers nongaited locomotion and manipulation in a single multicontact planning framework. However, until now we have not proposed a *controller* that can deal with generated plans, nor had we experimented with common ground planning on real robots. We propose to extend the QP control methods to encompass the idea that other objects and entities can be integrated as parts of a single controller when they interact with the robot. We have introduced this idea and demonstrated its applicability in graphic animation of avatars [15], summarized in Section II-B. In this paper, we present its adaptation to robotics. The contribution of this paper over the material presented in [15] is to add all missing components for implementation on real robots, namely contact sensing and force control, and to demonstrate its applicability in real-application, real-world, real-robots scenarios with thorough experimentation on various platforms.

We believe that this idea will be largely adopted in robotics as, first, it is easy to implement—we provide the software implementation of this framework in open-source[1]—and, second, it allows us to ease task specification to its simplest expression, i.e., at the level of interactions. For example, when a robot has to open a fridge, our method does not ask to build specific geometric constraints [16] or virtual mechanisms [17]–[20], nor to implement a specific planning or control strategy [21]–[25]. Instead, we model the fridge as a "robot" with as many degrees of freedom (DoFs) as possible. The user must design the fridge model (e.g., as a ROS `urdf` file) and our controller integrates it with that of the robot and considers interaction tasks as defined through areas of interaction (contacts). The core idea here is that the model already embeds the constraints (the kinematics

Manuscript received February 13, 2018; revised August 20, 2018; accepted September 13, 2018. Date of publication November 9, 2018; date of current version February 4, 2019. This paper was recommended for publication by Associate Editor M. Schwager and Editor T. Murphay upon evaluation of the reviewers' comments. This work was supported in part by the EU H2020 COMANOID project www.comanoid.eu and in part by the JSPS Kakenhi B No. 16H02886. *(Corresponding author: Abderrahmane Kheddar.)*

K. Bouyarmane is with the Université de Lorraine, CNRS, Inria Nancy-Grand Est, Loria UMR 7503, Larsen team, Vandoeuvre-les-Nancy 54506, France (e-mail: karim.bouyarmane@gmail.com).

K. Chappellet and A. Kheddar are with the CNRS-UM LIRMM Interactive Digital Humans group, Montpellier 34090, France, and also with the CNRS-AIST Joint Robotics Laboratory, UMI3218/RL, Tsukuba 305-8560, Japan (e-mail: chappellet.kevin@gmail.com; kheddar@lirmm.fr).

J. Vaillant is with the Navya, Paris 75009, France (e-mail: joris.vaillant@gmail.com).

This paper has supplementary downloadable multimedia material available at http://ieeexplore.ieee.org provided by the authors. This includes a video that describes all the experiments achieved in multi-robot and force serving using QP. The robots that are showcased are the humanoid robot HRP-4, the Romeo Arm from SoftBak Robotics and the Shadow dextrous hand. This material is 45.9 MB in size.

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TRO.2018.2876782

[1]Available at github.com/jrl-umi3218/Tasks and a more complete version is available at https://gite.lirmm.fr/multi-contact/mc_rtc upon request

and the dynamics ones) instead of explicitly defining them as in [26] and [27].

Integrating the kinematic model of the manipulated mechanism has been proposed in previous work [23], [24]. However, they remain at the geometric level, and only the kinematics of the planned mechanisms are accounted for. The planned configurations and motions in these studies do not account for the dynamics and inertia of manipulated mechanism(s), although these will influence robot balance through motion. In [25], the dynamics of the articulated mechanism is accounted for, yet they restrict the study to one-DoF mechanisms, and robot balance is not an issue (manipulator), we also refer the reader to the references therein for a review of previous door and drawer opening studies and their limitations. A Cartesian impedance method for the opening of a door is proposed in [21] for a mobile manipulator without balancing issues and with the door opening motion being designed for the specific task at hand.

Instead, we compute desired states that have coherent contact interaction forces. Many of the intended manipulation and comanipulation applications rely on friction (manipulation of a free-floating box for example) and necessitates the generation of the right amount of normal and tangential forces. Therefore, it is important to master force control under a QP controller framework, even on position-controlled robots, which has not been previously proposed to our knowledge. Hence, we propose it as another contribution in this paper.

The rest of this paper is organized as follows. Section II recalls the multirobot QP formalism from [15]. Section III introduces QP force control to track the manipulation forces resulting from multirobot QP when applied on position-controlled robots. Finally, Section IV presents experimentation results where our new controller is applied in very challenging scenarios involving Kawada's HRP-4 humanoid robot, Softbank's ROMEO arm, and Shadow's dextrous hand.

II. Multirobot QP

A. QP Control: A Brief Background

QP control has been proposed in the robotics and computer graphics communities to solve the control problem of multibody systems with floating bases subject to friction limitations. The approach appeared particularly suited to humanoid robots and humanoid virtual characters that typically feature such properties. A QP is instantiated at every control/simulation time-step minimizing the error of multiple desired task accelerations under all physical and structural constraints of the robot, which have the characteristic of being linear in the optimization vector variable composed of the control torques, contact force coefficients along the linearized friction cones, and joint accelerations. The multitask problem is cast as a multiobjective optimization program that can be solved with a weighted-sum scalarization or a lexicographic ordering scheme, among other possible multiobjective optimization or multicriteria decision making resolution techniques. Of the studies that opted for the weighted-sum scalarization, the work in [28] is worth citing in the field of computer animation as one of the firsts that proposed the method for tracking in physics simulation a motion capture data clip with a standing humanoid character in a multicontact posture. In [29], the approach is enhanced by accounting for bilateral grasp contact and more complex balancing strategies; Lasa *et al.* [30] combined the QP controller with higher-level finite state machine and used it for locomotion with cyclic feet contact switching, whereas Bouyarmane and Kheddar [6] applied the approach for humanoid robot in acyclic multicontact locomotion, applied later in DARPA robotics challenge-like scenarios in simulation in [31] and to real robot HRP-2 climbing a vertical ladder in [12]. In [32] and [33], the continuous task activation/deactivation within this scheme is proposed by continuous variation of weights. There are many other studies that use the QP in other schemes, such as force control distribution, e.g., [34]–[36]. In the following, we extend this framework to multicontact manipulation of articulated mechanisms and floating objects by humanoids, and to multirobot collaboration (e.g., robot–robot comanipulation).

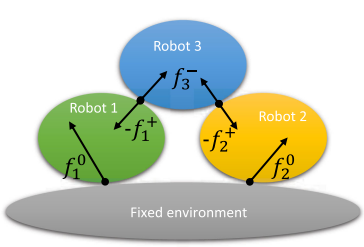
B. Multirobot QP Formalism

In this section, we briefly recall the multirobot QP formalism from [15]. Let us consider a system of n "robots" that can be actual robots, free-floating rigid objects, or passive articulated mechanisms, such as a door, a drawer, or a valve for example. A typical minimal manipulation system would consist of $n = 2$ "robots": the actual manipulating robot and the manipulated object or mechanism; a typical minimal collaboration system would consist of $n = 3$ "robots": the two collaborating robots and the collaboratively manipulated object; a dexterous hand with m fingers manipulating a rigid object would consist of $n = m + 1$ "robots," each finger and the object. We use the unified term "robot" here to refer to all these systems, since they are all instances of the general multibody model. Indeed, each of these systems $i \in \{1, \ldots, n\}$ can be modeled as a fixed base or free-floating base kinematic tree structure for which the DoFs q_i obey the following equation of motion (EoM)

$$M_i(q_i)\ddot{q}_i + N_i(q_i, \dot{q}_i) = J_i^T f_i + S_i \tau_i \,. \tag{1}$$

Equation (1) encompasses all types of robots and accounts for all underactuation possibilities (free-floating base for humanoids and for free-floating rigid objects, nonactuated joints of passive mechanisms) through the actuation-to-DoFs mapping matrix S_i. Note that we use Newton–Euler-based algorithms for the derivation of (1) in our implementation [37]. In this framework, the parts of $\dot{q}_i$ and $\ddot{q}_i$ corresponding to a free-floating link (i.e., the whole object in case of a free-floating rigid object or the base link of a humanoid) are abusive notations for V_i and $\dot{V}_i$, respectively, where V_i is the $\mathrm{SE}(3)$ velocity of the free-floating link.

The vector f_i stacks all point contact forces applied on the surfaces of robot i. These contact forces are either applied by the fixed inertial environment (e.g., at the feet of humanoids) or by another robot j (e.g., the forces applied on the hands of a humanoid by a manipulated object). The latter forces come in pairs of action/reaction forces among the system of robots according to Newton's third law, and opposite forces applied by the robot i on the robot j appear inside vector f_j. We, thus, decompose the forces f_i as $f_i = (f_i^0, f_i^-, -f_i^+)$ such that f_i^0 stacks the forces applied by the fixed environment on the robot



$$F^0 = \begin{bmatrix} f_1^0 \\ f_2^0 \end{bmatrix} \in \mathbb{R}^6$$
$$F^- = [f_3^-] \in \mathbb{R}^6$$
$$F^+ = \begin{bmatrix} f_1^+ \\ f_2^+ \end{bmatrix} \in \mathbb{R}^6$$
$$F^+ = F^- = (\Pi \otimes I_3)F^-,\ \Pi = I_2$$

Fig. 1. Force notation illustration.

i, f_i^- stacks the forces applied by the robots $j < i$ on robot i, and f_i^+ stacks forces applied by robot i on robots $j > i$. We then denote F^0, F^-, F^+, respectively, the vectors stacking all the vectors f_i^0, f_i^-, f_i^+. Let K be the total number of forces in F^-, i.e., such that $F^- \in \mathbb{R}^{3K}$. By virtue of Newton's third law, there exists a permutation matrix $\Pi \in \mathbb{R}^{K\times K}$ such that

$$F^+ = (\Pi \otimes I_3)F^- \tag{2}$$

where $\otimes$ denotes the Kronecker product, see Fig. 1.

We denote $\Psi = \Pi \otimes I_3$ (itself a permutation matrix). Let K_i be the number of forces in f_i^+, i.e., such that $f_i^+ \in \mathbb{R}^{3K_i}$. The permutation matrix Ψ is decomposed into selection matrix blocks $\Psi_i \in \mathbb{R}^{3K_i\times 3K}$ in the form

$$\Psi = \begin{pmatrix} \Psi_1 \\ \vdots \\ \Psi_n \end{pmatrix} \tag{3}$$

such that for each i we can write $f_i^+ = \Psi_i F^-$. Finally the equations of motions (1) take the form

$$M_i(q_i)\ddot{q}_i + N_i(q_i,\dot{q}_i) = J_{i,0}^T f_i^0 + J_{i,-}^T f_i^- - J_{i,+}^T \Psi_i F^- + S_i\tau_i \tag{4}$$

where $J_{i,0}$ and $J_{i,-}$ and $J_{i,+}$ are the matrices obtained by extracting from J_i the columns corresponding to the positions of f^0, f^-, f^+ in f, respectively.[1] We stack together all the equations (4) with the following matrices and vectors:

$$q = (q_1,\dots,q_n) \tag{5}$$
$$\tau = (\tau_1,\dots,\tau_n) \tag{6}$$
$$M(q) = \text{blockdiag}(M_1(q_1),\dots,M_n(q_n)) \tag{7}$$
$$J_0(q) = \text{blockdiag}(J_{1,0}(q_1),\dots,J_{n,0}(q_n)) \tag{8}$$
$$J_+(q) = \text{blockdiag}(J_{1,+}(q_1),\dots,J_{n,+}(q_n)) \tag{9}$$
$$J_-(q) = \text{blockdiag}(J_{1,-}(q_1),\dots,J_{n,-}(q_n)) \tag{10}$$
$$S = \text{blockdiag}(S_1,\dots,S_n) \tag{11}$$
$$N(q,\dot{q}) = \left(N_1(q_1,\dot{q}_1)^T \cdots N_n(q_n,\dot{q}_n)^T\right)^T \tag{12}$$

to get our synthetic Newton's third law-consistent EoM for the whole system of robots

$$M(q)\ddot{q} + N(q,\dot{q}) = J_0^T F^0 + \left(J_- - \Psi^T J_+\right)^T F^- + S\tau\,. \tag{13}$$

[1] we use the index notations $0, +, -$ in the superscript of vectors and subscript of matrices, to avoid conflict with the transpose notation of matrices.

The kinematic constraint that expresses the coincidence of the contacts points corresponding to an action/reaction pair can be synthetically written using the matrix Ψ and the principle of virtual work as

$$J_+\dot{q} = \Psi J_-\dot{q} \tag{14}$$

which is equivalent to, given that a permutation matrix is orthogonal $\Psi^T\Psi = I_{3K}$

$$\left(J_- - \Psi^T J_+\right)\dot{q} = 0\,. \tag{15}$$

This latter form of the constraint is consistent with the fact that F^- can be interpreted as the constraint's Lagrange multiplier in (13). This constraint has to be complemented with the fixed environment contact kinematic constraint that writes

$$J_0\dot{q} = 0 \tag{16}$$

for which F^0 also appears as the corresponding Lagrange multiplier in (13).

Note that the proposed *mathematical* Lagrange multiplier interpretations of F^- and F^0 do not oppose the fact that both F^- and F^0 consist of *physical* contact forces (as they had been initially constructed earlier in the section by concatenation of point contact forces). As a consequence of their physical nature, F^- and F^0 are indeed the correct subjects of the Coulomb friction cone constraints $F^- \in \mathcal{C}_-$ and $F^0 \in \mathcal{C}_0$ (which would not have been necessarily a justified hypothesis if we had derived (13) directly using a Lagrangian approach on the whole system made from all of the robots). These friction cones are then approximated as polyhedral cones with generators stacked as columns of matrices denoted C_- and C_0, respectively [38]. The coefficients of F^- and F^0 along the generators are denoted λ^- and λ^0, respectively, such that $F^- = C_-\lambda^-$ and $F^0 = C_0\lambda^0$. These coefficients are constrained to be nonnegative component wise

$$\lambda = (\lambda^-,\lambda^0) \geq 0\,. \tag{17}$$

The constraints of the problem are completed with the appending of joint limits, velocity limits, torque limits, and velocity-damper-based collision avoidance constraints between any links l_a and l_b, all of the initial forms

$$q_{\min} \leq q \leq q_{\max} \tag{18}$$
$$\dot{q}_{\min} \leq \dot{q} \leq \dot{q}_{\max} \tag{19}$$
$$\tau_{\min} \leq \tau \leq \tau_{\max} \tag{20}$$
$$\dot{\text{dist}}(l_a,l_b) \geq \xi\frac{\text{dist}(l_a,l_b) - \delta_s}{\delta_i - \delta_s} \tag{21}$$

where the parameters ξ, δ_i, and δ_s represent, respectively, the velocity damping coefficient, the influence distance between links below that the constraint starts to act, and the security distance between links that the constraint ensures will never be reached. The constraints (18), (19), and (21) are rewritten in

terms of constraints on $\ddot{q}$ as follows:

$$\frac{\dot{q}_{\min} - \dot{q}}{\Delta t} \leqq \ddot{q} \leqq \frac{\dot{q}_{\max} - \dot{q}}{\Delta t} \tag{22}$$

$$\frac{q_{\min} - q - \dot{q}\Delta t}{\frac{1}{2}\Delta t^2} \leqq \ddot{q} \leqq \frac{q_{\max} - q - \dot{q}\Delta t}{\frac{1}{2}\Delta t^2} \tag{23}$$

$$\ddot{\text{dist}} \geq \frac{1}{\Delta t}\left(-\xi\frac{\text{dist} - \delta_s}{\delta_i - \delta_s} - \dot{\text{dist}}\right). \tag{24}$$

These formulations allow us to write the control problem for the system of robots as a single QP

$$\min_{\ddot{q},\tau,\lambda} \sum_{k=1}^{M} w_k ||\ddot{g}_k - \ddot{g}_k^d||^2$$

$$\text{subject to (13), (15)–(17), (20), (22), (23), (24)} \tag{25}$$

where g_k denote the tasks (possibly multidimensional) and $\ddot{g_k}^d$ the desired task accelerations that can for example take the following form:

$$\ddot{g}_k^d = \ddot{g}_k^{\text{ref}} - P_k e_k - D_k \dot{e}_k\,,\; e_k = g_k - g_k^{\text{ref}} \tag{26}$$

with P_k and D_k denoting the task gain matrices designed such that $\begin{pmatrix} 0 & I \\ -P_k & -D_k \end{pmatrix}$ is a stable (Hurwitz) matrix, and where g_k^{ref} is a reference trajectory or a fixed set-point of the task [39].

Once a contact state for the system of robots has been specified, the effectiveness of the formulation (25) lies in the fact that a task can be specified for any feature of any single robot or group of robots of the system in a uniform way. For illustration, it is sufficient to specify a task in terms of position and orientation of a free-floating manipulated object; the control commands for the manipulating robot (or the comanipulating robots) will automatically be induced from the contact constraints through (25), without the need of explicitly specifying any task for the manipulating end-effectors. Similarly, if it is a mechanism that is being manipulated, it is sufficient to specify a task in terms of the configuration of the mechanism (opening angle of a door, rotation angle of a valve) rather than tasks for the manipulating end-effectors. As an further illustration of the expressiveness of (25), the balance of a biped robot manipulating an object with a nonnegligible mass can be written in terms of a single task on the center of mass (CoM) of the whole system.

III. QP Force Control

The QP controller (see Section II) outputs accelerations $\ddot{q}$, forces coefficients λ, and joint torques τ for the robots. We use it in our applications with position-controlled robots (see Section IV), by double integrating the output $\ddot{q}$ and feeding the resulting q to the low-level motor position controller. However, in view of the effective use of the multirobot QP in interaction tasks (e.g., robots comanipulations), it is necessary to ensure that the planned manipulation contact forces are adequately matched during the execution, even when the framework is applied on position-controlled robots. As demonstrated in Section II, the formulation (25) does produce accelerations and, hence, position commands, which are consistent with the *QP-predicted* contact forces λ at a given control time-step. However, there are following two issues with this prediction:

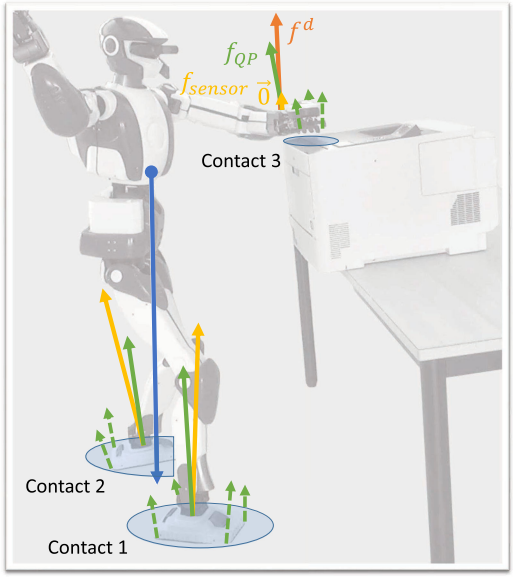


Fig. 2. Predicted forces (green) in planned contact state {Contact 1, Contact 2, Contact 3} versus sensor forces (yellow) in real contact state {Contact 1, Contact 2} (the predicted forces are resultant at the sensor locations of the point forces in dashed lines computed at the vertices of the contact prints). In this situation, the QP controller assumes that the robot is in the planned contact state and, therefore, predicts forces that do not correspond to the actual force repartition, since the hand contact (Contact 3) is not yet established.

1) it is based on the QP-used models of the robots and the objects, implying that any discrepancy in these models would result in inexact predicted forces;
2) it supposes that the robot is in a given contact state that was *planned* beforehand, without actually knowing whether the robot has effectively reached that contact state and whether the contact has been established. If not, the QP would still base its calculations on the assumption that the robot is in its planned contact state and will output contact forces that are in reality null, see Fig. 2.

Therefore, we need a method that feeds back the information from the force sensors and realizes the tracking of the predicted forces by the sensored ones. Such a tracking method should also be able to ensure that the actual contact states match the planned ones by making sure that any planned contact has effectively been established in the current contacts state.

Force control has been extensively studied in robotics, see a thorough review in the monographs [40], [41] and in the handbook of robotics [27]. Force control in the task space for fixed-base robots was also developed and experimented in [26], [42], and [43]. Contrarily to [16] and [26], task specification including force control is simplified and made straightforward with the QP built-in multirobot constraints specification, since interaction forces are part of the QP decision variables. Active force control can be achieved either directly, through explicit closure on the force, or indirectly through compliance, impedance, or admittance control [27]. The multirobot QP control framework allows us to have both, and also allows us to consider floating-base under-actuated or fixed-base robots. As compared to exiting controllers, the added value of the QP control is not in a structural way or in how basic force control is made, but

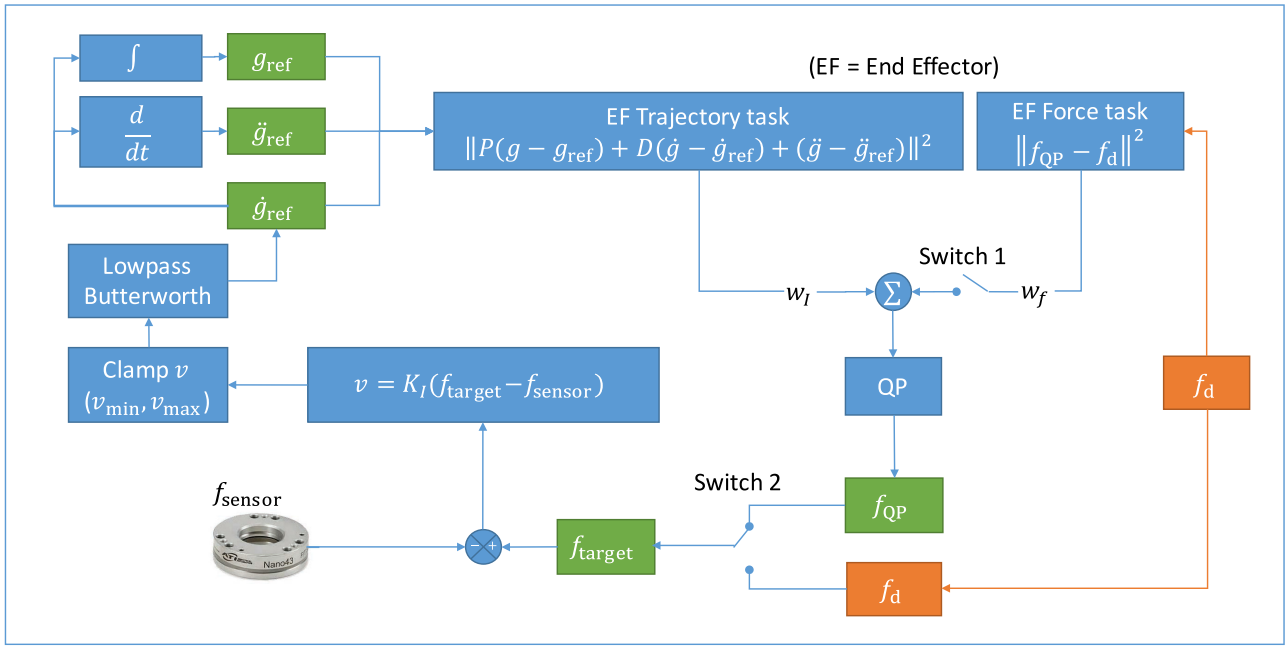


Fig. 3. Block diagram for force control with the QP. The error between the target force and the sensed force is converted into velocity with diagonal matrix gain K_I. This velocity is low-pass filtered into a reference velocity for the end-effector QP tracking task. The target force comes either from the force output of the QP (switch 2 up) or from a external user command (switch 2 down). That external user command can also alternatively be incorporated inside the QP (switch 1 closed) in order to influence the force output of the QP when using the latter as a target force (i.e., with switch 2 up).

rather in the way force control is integrated into the whole multiobjective task-space control, accounting for limitations in the force wrench explicitly and in full multi-unilateral-contact settings. For example, prohibiting sliding (unilateral contact) is made simply by adding built-in nonsliding constraint task; limiting the force or moment in any direction is easy. Yet, the real added-value with respect to existing force control frameworks, is the ability for the controller to suggest (i.e., plan) the references forces to be used in a given set of tasks and current state configuration and use that output to close the loop on such generated reference forces. For example, consider a multicontact setting where we ask a humanoid robot to move one leg and its body in multi-unilateral-contacts using the other foot and one hand as supporting contacts. In order to shift the CoM and moving the leg, we may let the QP decide what force to generate on the supporting hand under torque and nonsliding constraints and close the loop on such computed force wrench during the movement. We could also suggest a threshold force on the hand or a desired force behavior that must be met at best. If not, we would have to select by hand what the force trajectory is (for all components) that need to be generated. As we will see later, our controller can servo on a user-specified or otherwise planed desired force wrenches, or on its own computed force-wrenches (if no desired force wrench is specified) and finally, it can servo on both (i.e., on user-specified or planed desired force wrenches while driving the controller to generate a behavior and consequently contact forces as close as possible to the desired ones, but keeping physical and constraint consistencies).

To achieve the previously described force control behaviors in the multirobot QP controller, we propose the scheme represented in Fig. 3. For a given end-effector (or more generally any link) of the robot equipped with a force/torque sensor/observer, we proceed with an admittance controller that takes as an input the error between a target force f_{target} and the corresponding sensed force f_{sensor}, and transforms it into a QP end-effector task through the following stages. First, we convert the force error into a velocity command with a diagonal gain matrix K_I (inverse of a damping)

$$v = K_I\left(f_{\text{target}} - f_{\text{sensor}}\right) \tag{27}$$

then we clamp that value between $(v_{\min}, v_{\max})$ to prevent the end-effector from moving too fast nearby the contact surface if not reached yet (i.e., if $f_{\text{sensor}} = 0$, which happens when the end-effector is "searching" the surface it is supposed to be in contact with), this is a guarded motion with

$$v_{\text{clamp}} = \min\left(\max\left(v_{\min}, v\right), v_{\max}\right) \tag{28}$$

to which we apply a low-pass filter (order 3 and cutoff frequency 20 Hz Butterworth in our implementations). The filtered signal $\tilde{v}$ is converted into a QP end-effector task by taking it as the reference velocity trajectory ($\dot{g}_{\text{ref}} = \tilde{v}$) and by deriving it from the reference position g_{ref} and acceleration $\ddot{g}_{\text{ref}}$ trajectories. The latter are then sent to the QP as an end-effector trajectory tracking task (here, similar to an impedance). Note that the contact zero-acceleration constraint is dropped for any link that is subject to the admittance task.

We retained three possible strategies to incorporate the admittance scheme in our framework, depending on the states of the switches 1 and 2 that appear in Fig. 3. Fig. 4 illustrates a simplified representation of the different switch combinations in the block diagram of Fig. 3.

The configuration as it appears in the case of Fig. 3, i.e., with switch 1 open and switch 2 up, implements an autonomous behavior where the controller tracks the force output by the QP

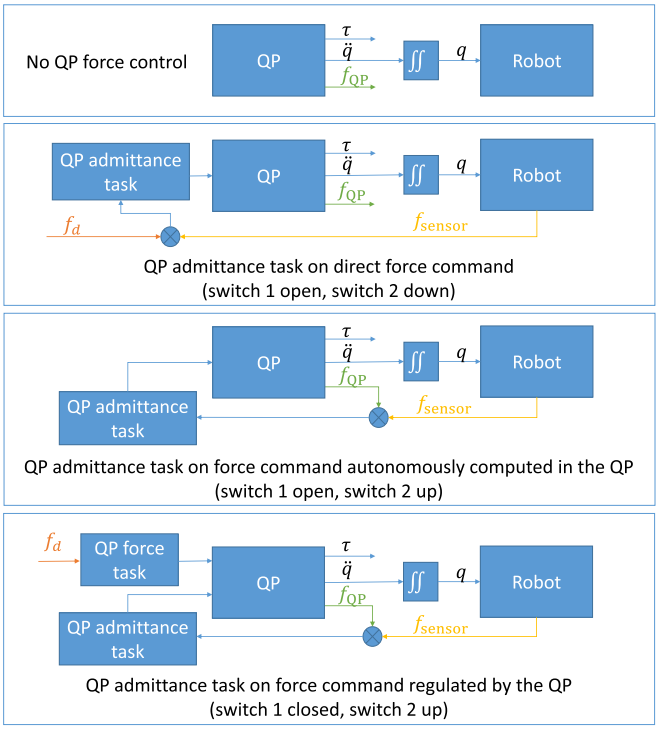


Fig. 4. Simplified representations of the different switch positions in the block diagram of Fig. 3. From top to bottom: Both switches 1 and 2 open, switch 1 open and switch 2 down, switch 1 closed and switch 2 up, switch 1 closed and switch 2 down.

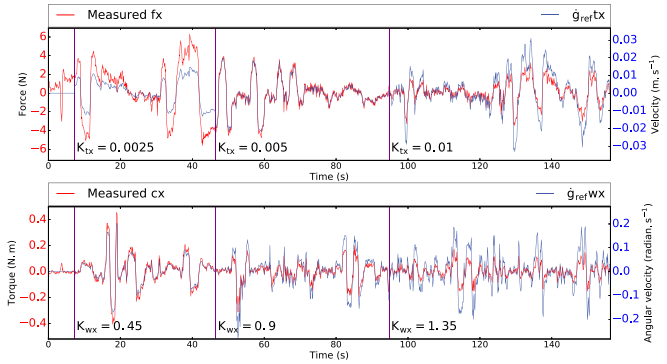


Fig. 5. Effect of the gains for x-axis in $\mathrm{diag}(K_I)$ on the force time response of the HRP-4 right hand to human interactive manipulation. The same behavior are observed for the other axis and also in both hands and feet that are force-control to zero-force reference ($f_d = 0$ in all components). The admittance gains are changed online three times during the manipulation from its initial (high damped) value to double and double again from the values. The higher the admittance gain, the faster the response, hence the more interactive the robot is to human guidance.

TABLE I
ADMITTANCE RANGE FOR FORCE CONTROL: (SLOW/SAFE) LOWER BOUND; FAST/INTERACTIVE (HIGHER BOUND)

Force sensor	Forces	Moments
hands	$[0.001 - 0.01]$	$[0.1 - 1.8]$
feet	$[0.001 - 0.01]$	$[0.001 - 0.01]$

as the QP figures it out from the other tasks of the problem. However, the user might want to have some control on interaction forces, that might not turn out to be satisfactory for them (typically, in the applications and experiments of Section IV, we considered that the forces output by the QP on the hands of the HRP-4 robot can be too important given the relative fragility of the hands, regardless of the nominal manufacturer's torque limits, so we wanted to produce less force on the hand). Hence, we offer the user the possibility to specify a desired force f_d that can be used in two ways. The simplest one is with switch 1 open and switch 2 down, this allows the sensor force to track f_d independently of the other physical constraints of the robot. This is not a safe strategy as the user might specify unrealistic forces f_d given the current configuration and contact state of the robot, it can be used as a last resort only if the user is sure that the specified f_d is safe/consistent. The other way to use f_d is through the QP, with switch 1 closed and switch 2 up by adding the term $||f - f_d||^2$ to the cost function of the QP, this we call a *QP force task*. This ensures that the user-specified force f_d is filtered through physical constraints that are taken into account in the QP and produces an f_{target} that is as close as possible to f_d while remaining physically consistent.

First, we conducted preliminary QP force tasks experiments with a humanoid HRP-4 equipped with built-in 6-DoF force/torque sensors on each foot; in addition, our robot has customized 6-DoF force/torque sensor on each hand (Nano 40 from ATI). In these experiments (that can be made with any robot), the reference force is put to zero $f_d = 0$ for each limb and in all directions. The goal is to determine, first, the range of gains matrices K_I and, second, the clamping speeds and direction for each controlled limb.

Each value of K_I correlates to the response time of the desired force/moment. To not account for dynamic equilibrium constraints, we hung the robot in the air, fixed the waist, and manipulated interactively and simultaneously (two operators are needed) every terminal point (both hands and feet). The higher the admittance gain for a given component of force, the faster the response, this is noticeable in Fig. 5. The task gains for the EF trajectory task (impedance on resulting position/orientation errors g in Fig. 3) are found to be $P = 0.1$ and $D = 10$ for all the limbs and all directions. This was the case for the interactive experiments and in the trials and final multicontact experiments we conducted later. Increasing task PD gains do not have a predominant effect w.r.t to the admittance gains K_I. For the admittance gains, the Table I sums-up the values for the controlled limbs (they are in fact the same for the hands and for each foot).

Once the gain ranges are determined, the robot is put in four contacts configuration, the CoM is shifted to lie within the center of the right foot. Fig. 6 shows the experimental setup that was used to assess the QP force control. It is put near a table in a half-sitting initial posture (i.e., the knees are slightly bent). A posture is computed so that both hands are just hovering above the table while the feet are firmly on the ground. The force control goal is specified for each hand and the left foot independently, but simultaneously. The right foot is kept as a supporting noncontrolled contact for the coherency of the force control. As we only have unilateral contacts, we must insure that the forces are not controlled in a nonconsistant way. Obviously, one cannot control

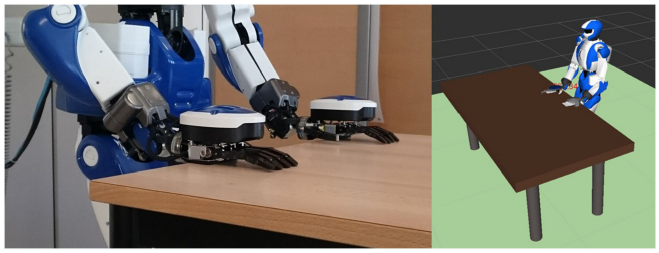


Fig. 6. Base experiment for comparing the different proposed QP force control paradigms. Each hand is controlled with a different paradigm.

a force f_i of a given contact C_i and, at the same time-control, a force in another contact $C_{j\neq i}$ that lies in the opposite direction of $f_i = -f_{j\neq i}$, to support the reaction. Another example, with only feet as the contact, controlling the force along the z-axis (normal to the ground aligned with the gravity field vector) of both feet at the same time will result in oscillatory behaviors and none convergence up to instability; this is because the CoM shift that results in controlling the wrench of one foot, will have a direct consequence on the other, based on the action-reaction principle. This problem is not encountered in force control of fixed-based robots. This is the reason why accounting for the QP computed forces and encompassing nonsliding and other bound forces in the multirobot QP allows us to keep the concistency of the force control in multi-unilateral-contacts in most cases.

Fig. 7 illustrates the result of the simultaneous force control in all directions for the remaining three limbs. For these experiments, we used the admittance gains $\mathrm{diag}(K_I) = [f_x = 0.001, f_y = 0.001, f_z = 0.002, m_x = 0.15, m_y = 0.15, m_z = 0.15]$ for the hands and $\mathrm{diag}(K_I) = [f_x = 0.001, f_y = 0.001, f_z = 0.002, m_x = 0.007, m_y = 0.007, m_z = 0.007]$ for the left foot. Given that we are in a *multi-unilateral contacts* setting, desired forces are kept small to avoid sliding (we also identified the sliding coefficients prior to the experiments) and also the time response is purposely set to be moderately slow (in the order of hundreds of milliseconds) to avoid overshooting in the force control. Overshooting in force response is not problematic, but it may excite the robot internal flexibilities and this may result in nondesired damped oscillations throughout the body of the robot. Also each time a reference force wrench is changed for a given contact, its effects are perceived through a light instant deviation of the remaining contact forces (due to light changes in the postures); this is highlighted in Fig. 7.

We previously mentioned the friction issue in unilateral contacts. In our experiments, we could easily identify the friction coefficients of the feet and hands in all directions using incremental reference translational forces. When we provide the QP with desired forces that lie outside the identified friction cone for a given contact, the latter slides and this is noticeable when we plot the measured forces: the desired forces cannot be reached. Torsional frictions are more difficult to identify. We could experimentally validate in part the observations in [44]: the torsional friction and bounds on rotational sliding depend not only on the normal force and the Coulomb friction coefficient, but also on the distance between the center of pressure and its distance to the closest vertex of the contact surface (that we cannot estimate in practice). For instance, we found a torsional friction of the left leg (around z, the vertical axis aligned with gravity and the contact normal to ground) to be $\simeq 0.5$; but this z-axis torsional friction decreased to 0.4 and up to 0.3 when we asked the feet to also be controlled with reference moments on the x- or y-axes.

All in all, nonsliding constraints and bounds on forces can be integrated as part of the whole-body controller explicitly. After experiments with the reference force wrenches in force tasks, i.e., without using the QP computed forces, the plots in Figs. 8 and 9 illustrate a comparative study of all combinations that can be achieved with our controller. The transient phase to converge to the desired command is due to the fact that the same unified control is used both for searching for the contact and for regulating the magnitude of the contact force once contact has been established. As for the previous canonical experiments, the speed of convergence can be mainly tuned with the admittance gain matrix K_I but also to some extent with task admittance gains P and D (see Fig. 3). The later also correlate to the time response as for any other tasks of the QP. Fig. 9 illustrates how the regulation scheme does not track forces when they are not consistent with the constraints and automatically compute the closest desired force wrench to the user-defined one.

IV. Experimentations

We experimented with the multirobot QP controller on various challenging scenarios that were recorded in an accompanying video. The scenarios use three different robots: Kawada HRP-4 humanoid robot, SoftBank Robotics ROMEO arm (with a hand), and Shadow dextrous multifinger hand. In all three scenarios, the control was performed in real-time, as the multirobot QP was consistently solved in times below 5 ms per iteration.

A. Box Manipulation Experiment

We instantiate the multirobot QP setting on a box manipulation experiment in Fig. 10. The box is modeled as a one-link free-floating robot with unknown mass and CoM, only the geometric model of the box is known to the controller. The force-control scheme with autonomous QP (switch 1 open, switch 2 up combination) is used to ensure that the robot applies sufficient force to avoid box slippage. The box is made with cardboard and is filled with various arbitrary objects tightly occupying all the space inside the box (to have a constant CoM). We used an Extended Kalman Filter method to estimate the mass and CoM position of the box online while manipulating it.

Once the dynamic parameters of the box are identified, the contact forces during manipulation can be computed from and controlled by the MQP to achieve the same trajectory of the box with less internal forces instead of using the user-defined contact forces.

B. Manipulation of Articulated Mechanisms

In these experiments, we illustrate the capabilities of the controller to manipulate every-day-life objects with articulated

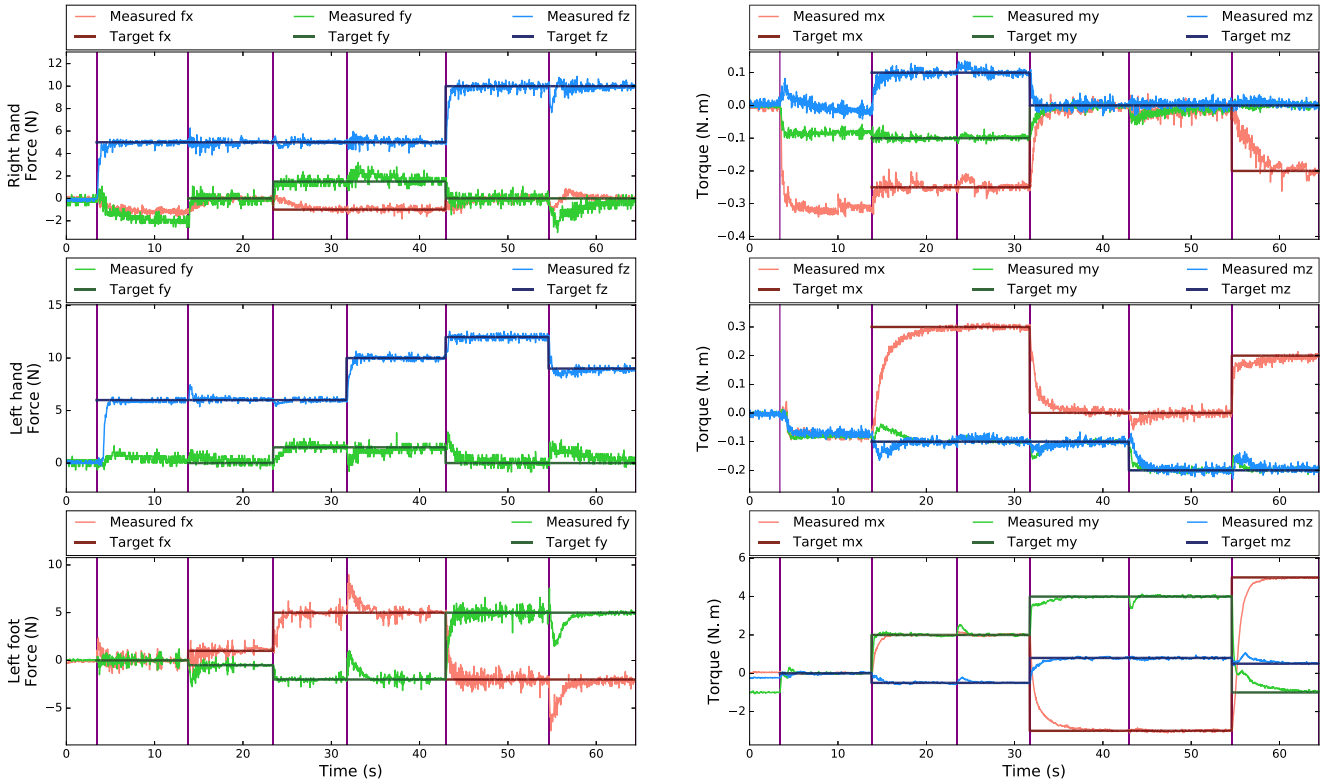


Fig. 7. Simultaneous multi-unilateral-contact force control of the humanoid robot HRP-4 in the setting of Fig. 6 (here the desired force wrench f_d is given directly to the QP and f_{QP} is not used). The vertical bar indicates the moment where a desired force wrench f_d is changed from either one of the hands or feet. Horizontal lines indicate the desired values and continuous curves are measured taking their values from the appropriate force sensor. When there is no dotted lines, it means that there is no explicit desired force sent to the robot. From up to down: left, right hands, and left foot three components of forces and moments.

mechanisms. Two manipulation scenarios were used in these experiments: door opening and printer tray opening. The robot is the HRP-4 humanoid that is provided with the `urdf` models of the objects. The door is a regular (not self-closing) door of the laboratory room intended for everyday use. It is modeled in the multirobot QP as a "robot" with a two-DoF fixed-base mechanism. One DoF is the passive revolute joint at the hinges of the door; the second one is a spring-loaded revolute joint at the handle of the door. The printer is a commercial printer (model Canon i-sensys LPB7680Cx). It is also modeled in the multirobot QP as "robot" with a fixed-base mechanism (although it could have been more accurately modeled as a floating-base mechanism in unilateral contact with a support table as in [45]) with one passive prismatic joint for the tray (the model can also include the other nonused trays and also all the buttons as prismatic joints).

In the printer experiment, the user provides a desired force $f_d = 10\,\text{N}$ along the local z-axis on the left hand in order to prevent its slippage (to compensate for the modeling approximation that we make consisting in defining a planar surface on the hand of HRP-4 that is not perfectly planar) and a desired force $f_d = 5\,\text{N}$ along the local z-axis on the right hand to firmly insert it inside the tray handle prior to the tray pulling motion, and $f_d = 0\,\text{N}$ for the remaining axes, see Fig. 11. Both force commands are sent using the "switch 1 closed, switch 2 up" combination of the controller in Fig. 3. Putting the left hand on the printer is suggested by the planner [14] to create a closed kinematic chain so as to pull the tray without causing equilibrium or force application problems.

Fig. 12 illustrates snapshots from the door opening experiment with the same controller. The accompanying video shows an additional door opening experiment using a position control scheme of the handle, with another robot posture where the door is opened with the left arm, pushed with the right one and finally crossed using a walking controller [46]. This is an example of sequencing the multirobot QP controller with other controllers such as a walking controller in this case.

C. *Robot–Robot Comanipulation*

We experimented with the multirobot QP paradigm for actual multirobot collaborative tasks between a humanoid robot ROMEO's left arm from SoftBank Robotics and the humanoid robot HRP-4. The two robots use different and unrelated low-level control and communication architectures, as well as different control frequencies (respectively, 100 Hz and 200 Hz), constituting a challenging setting for the multirobot QP controller, see Fig. 13.

The multirobot QP controller computations were completed on an external computer and sent to both robots using dedicated communication architecture. We plan in the future to embark the multirobot QP control computations on one of the robots

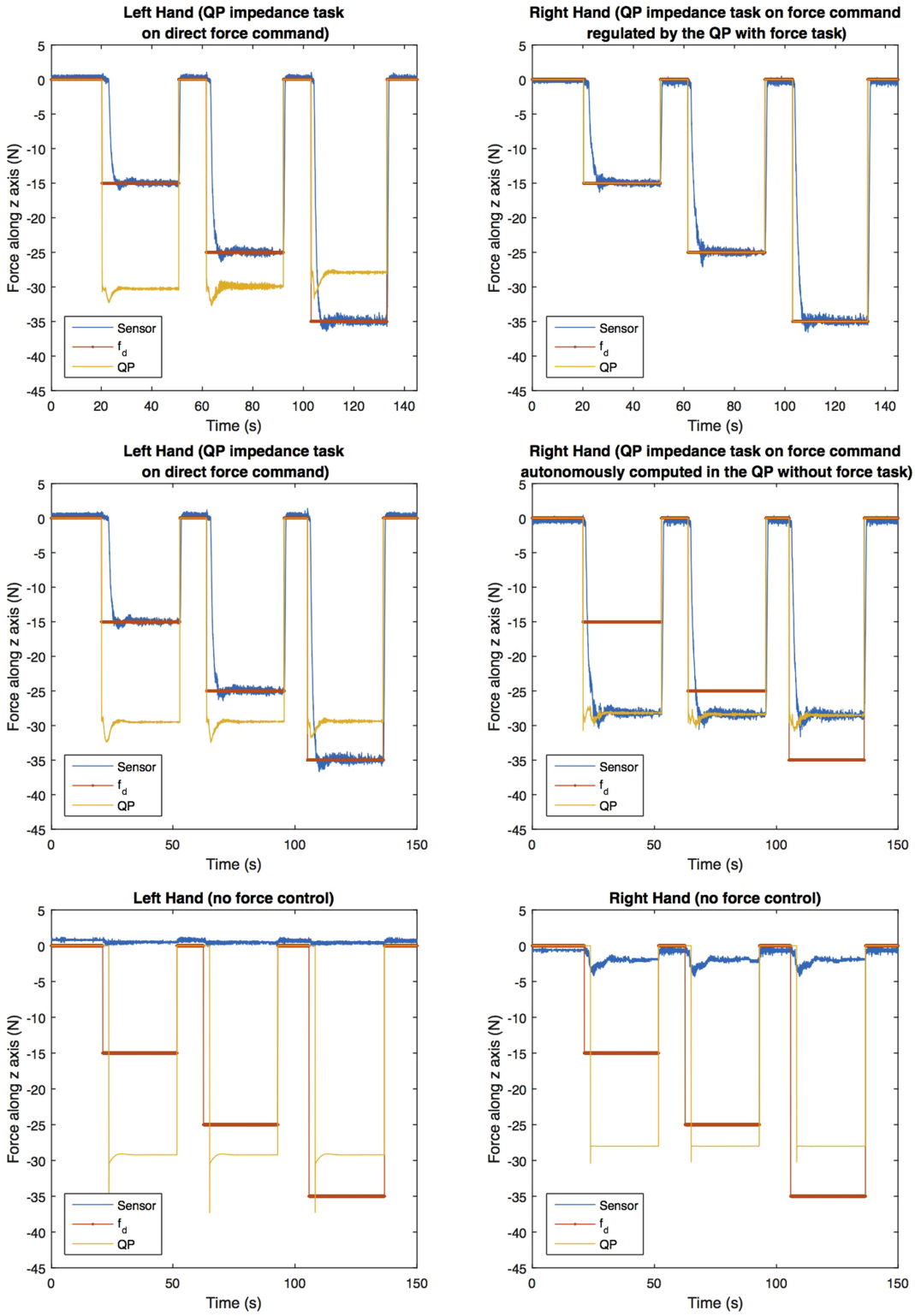


Fig. 8. Three different executions of the experiment in Fig. 6 to compare the different proposed QP force control methods. Each row represents the data for one run of the experiment.

and use the other robot's computational resources for auxiliary tasks such as vision for example.

The task consisted in a collaborative pick-and-place operation, collaboratively lifting a box (a random parcel package delivered by the post containing electronic parts that were not removed from the box) and putting it down in a different location. The task was only specified in terms of positions of the box using three way points: lifting up, moving to the right (of HRP-4), putting down. When putting down, we specified a slightly lower height than the lifted height to ensure that the contact

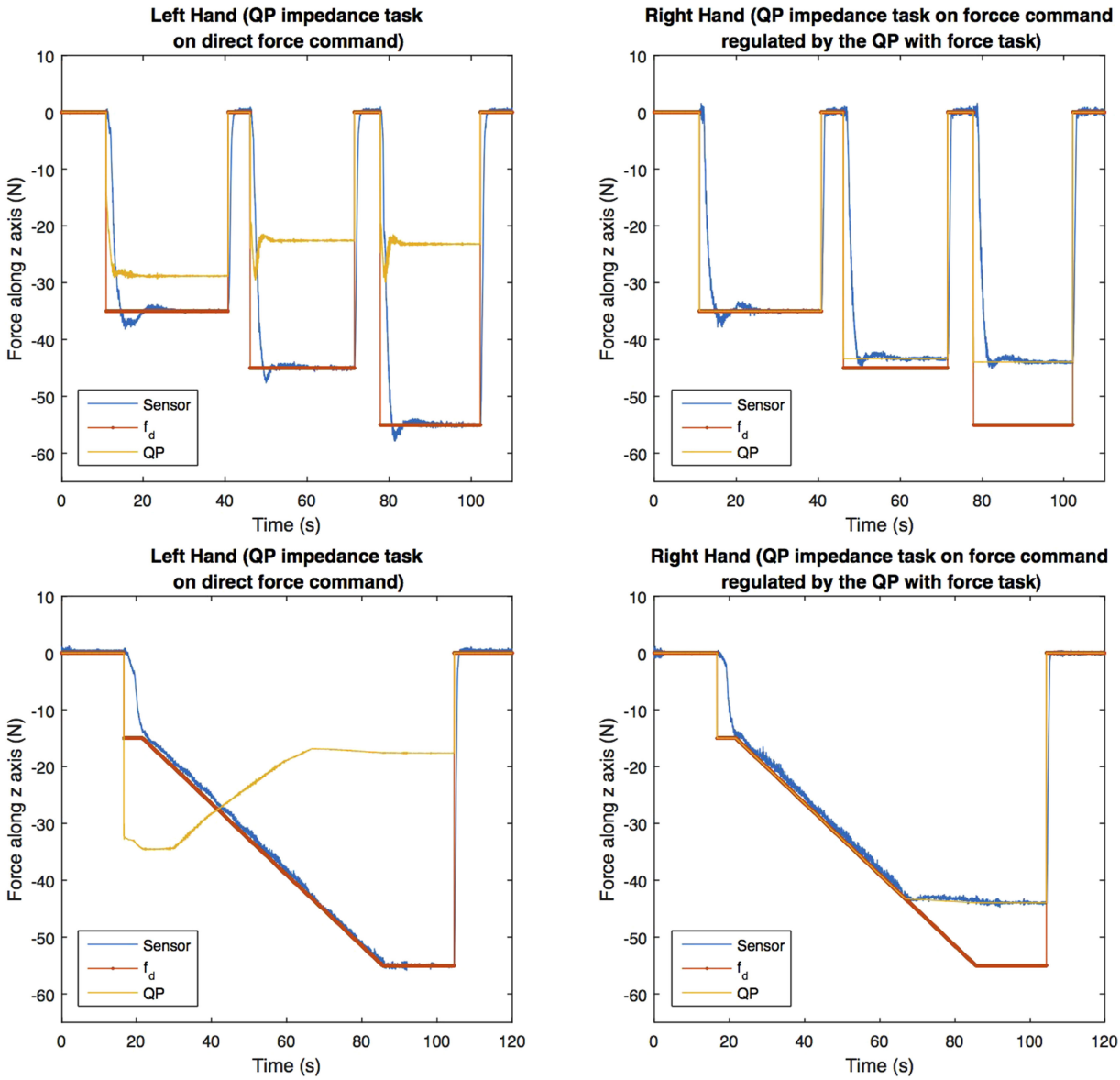


Fig. 9. Comparison between the two methods that account for a desired force command f_d, in two additional instances of the experiment in Fig. 6. In the first method, the robot follows the user command but reaches torque limits (over-torque errors appeared on the robot during both executions). The second method (right hand), the QP autonomously "saturates" the exaggerated force command to keep the robot within the torque limit constraint. The second method is thus the safest for the robot.

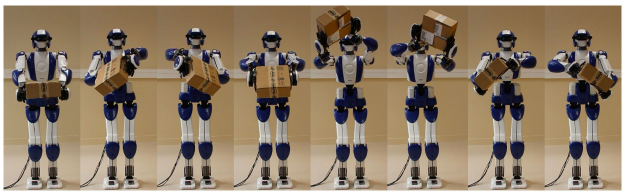


Fig. 10. Example of online dynamic inertial parameter estimation experiment for a manipulated box. Each image shows a posture way-point.

between the box and the table has been well established and to avoid dropping the box from a nonzero height. This one-task specification is illustrative of the multirobot QP coordination capabilities, since no explicit task is needed for the hands of the two robots, the unilateral contact constraints between the hands and the box being sufficient. Force control was additionally implemented in this experiment to maintain the contact force between the HRP-4 hand and the box, and the ROMEO hand and the box. Not using the QP force control scheme resulted in slippage of the box and the robots were unable to lift it. Fig. 14 tracks the motions of fixed points on the contact surface frames of the hands of the robots in comparison with the motion of a fixed point in the box frame representative of the task.

D. Dexterous Manipulation

Both HRP-4 and ROMEO robots are equipped with gripper mechanisms at the hand that do not implement anthropomorphic dexterous hand capabilities. So we chose to demonstrate the multirobot QP applicability to dexterous manipulation problems on a Shadow dexterous hand with 19 DoFs. We chose an

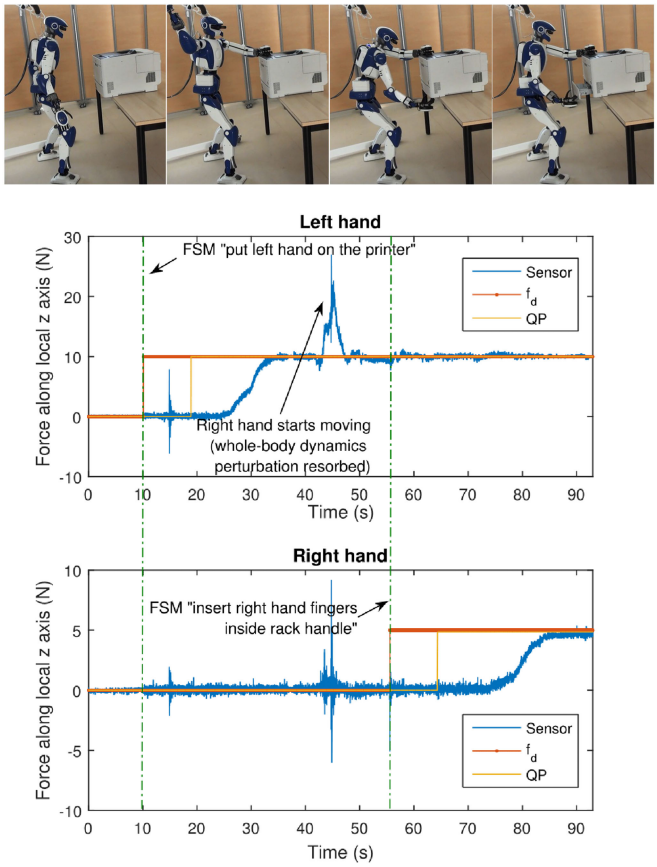


Fig. 11. Printer tray opening with HRP-4.

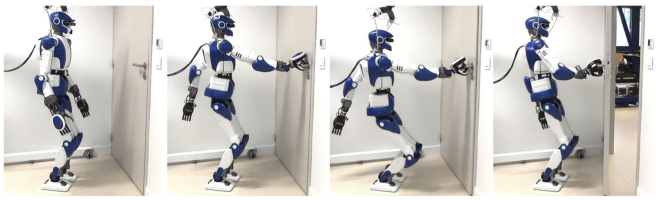


Fig. 12. Door opening with HRP-4 using the force control scheme of the multirobot QP.

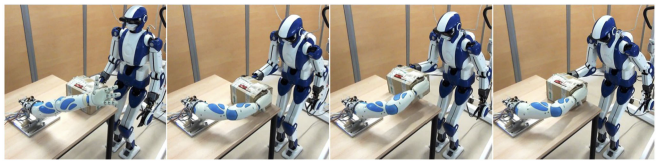


Fig. 13. Multirobot collaborative manipulation between HRP-4 and ROMEO's left arm.

illustrative manipulation problem where the manipulated object, a click pen, is again an articulated mechanism, but as opposed to the door and printer this time it is a free-floating base mechanism. The cardboard boxes in the previous experiments were also free floating but without articulations. Hence, with this last example, we cover all typologies of manipulated objects. The clicking articulation is modeled with a spring-loaded prismatic joint in the multirobot QP. The contact surfaces on the fingertips and the cylindrical body of the pen were faceted (approximated with planar surface patches).

The task in this example was specified on the configuration of the pen "robot" such that the position of the clicking part reaches its joint limit to trigger the exit of the writing tip (see Figs. 15 and 16).

E. Practical Considerations

All the experiments presented are performed with a consistent methodology: we use the MQP control framework without customization for all the experiments and we use the same desired force specification tasks and coding template with the switches in Fig. 3 for each robot/contact. The differences lie in the `urdf` of the robots in play and eventually the tasks (CoM, posture, force, etc.) for each scenario. For the previous experiments, Table II provides a list of the tasks that have been used (all coded as templates).

Since objects in the environment that are interacting with robots for given tasks are considered as "robots," an increasing number of objects come with an increase in the computational cost. This is partly true because of the following reasons.

1) We do not systematically include all robots—or objects considered as such, of a scene in a single MQP. Instead, we define clusters of MQPs on the basis of effective interactions between the robots/objects of interest; each of which is computed separately. Given the application and the context, we may consider the MQP computation as being remote, cloud computed, or distributed;
2) The MQP is sparse; therefore, we can benefit from any QP solver that exploits the sparsity of the problem; this is being currently investigated and since our framework is independent from the QP solver *per se*, any efficient sparse QP solver can be used with little development efforts.

The box manipulation experiment highlights the benefits in having an estimation of the parameters such as inertia and the pose of objects that are considered as "robots," whereas they are not. For the case of the box experiment—or any well structured articulated rigid objects, it is possible to add the inertia parameters identification in the task space (as they can write as a QP [47]) that allow more precise contact force computation and control during the manipulation when this is possible. As for the pose and configuration, such objects do not have encoders and, hence, estimating their configuration can be made using the robot embedded camera as we did very recently in [45]. If such two identification features can be gathered, then we may consider extending our MQP framework to deal with human–robot cooperative tasks.

Although our control framework is able to identify a large number of nonconsistent multicontact force control, there is no guarantee, as for now, that we handle all of them properly. For example, if we ask a humanoid robot to achieve a desired same f_d force on both foot in the same x-axis direction (front), the robot will bend behind up to falling in trying to fulfill such force references if there are no tasks to constraint the CoM. Moreover, we need through an explicit formalism to detect in a multi-unilateral-contacts setting if we are asking in specific

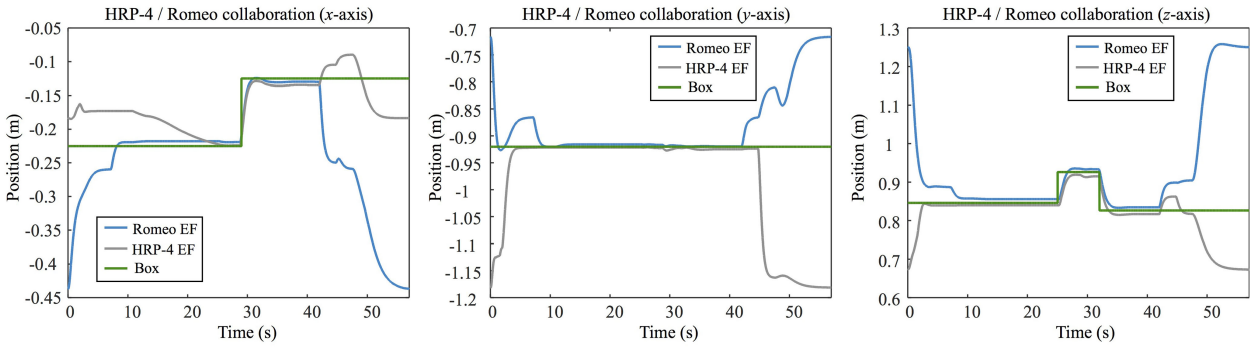


Fig. 14. Robot–robot comanipulation motion. Resulting coordinated motion (position of HRP-4 and that of ROMEO's hand link frames) from single task command (position of box frame).

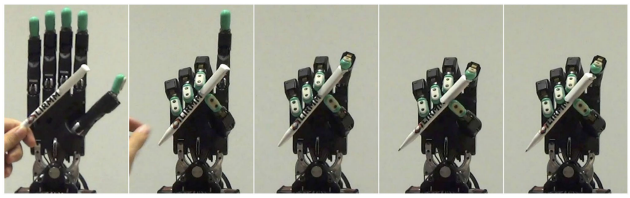


Fig. 15. Dexterous hand clicking a pen.

Fig. 16. Green spots are the predefined contact areas.

TABLE II
LIST OF TASKS FOR EACH EXPERIMENT

Experiment	List of tasks
Box	6D position and orientation of the Box
	2D CoM of {robot, box}
	Force tasks on both hands
Printer	6D position and orientation of both hands before contacts
	Force task on both hands just before planned contacts
	Force task on left hand after contact
	2D CoM of robot
	1D position of the printer tray after grasp
Door	6D position of right hand before contact
	2D CoM of robot
	Force task on right hand just before contact
	and when pressing the handle
	1D angle of the handle after grasp
	1D angle of the hinge after handle task complete
Romeo	6D position and orientation of the HRP-4 hand before contact
	Force task on HRP-4 hand after contact
	2D CoM of HRP-4
	6D position and orientation of Romeo arm before contact
	6D position and orientation of the box after contact
Pen	19D Grasp configuration of the hand before contact
	1D position of the click joint of the pen after grasp
All	Full configuration "rest" (or zero) task
	(low weight task to ensure QP is positive definite)

directions for force control of both the action and its reaction. Automating the detection of such user-specified inconsistencies is an interesting issue to investigate further.

V. CONCLUSION

We have shown the benefit of integrating task-space QP controllers as a single problem to handle multirobot interactions. By multirobot, we mean that the controller can deal with any cluster of objects or robots or mechanisms that are passive, partially passive, or totally capable of interaction. Not only does such an approach ease the specification of the tasks (as a complement to planning) to its simplest expression (i.e., the interaction level), but it also computes physically consistent contact interaction forces. Subsequently, we devised force control algorithms for QP controllers and show that we can achieve reliable closed-loop force control where the QP can track at best the desired forces (and does its best when they are not feasible) but also plan them if not specified. The implementation code of the multirobot controller is open source; it has been interfaced with vRep and Gazebo simulators. The code is already distributed to several teams worldwide, it is sustained, and has been implemented on other humanoid and robotic platforms (e.g., ARMAR, Nao, Pepper, HRP-2Kai, KuKa arms, etc.).

QP controllers are currently emerging as a gold standard to handle multiobjective tasks in redundant robots, such as humanoids. We have proven that they are capable of controlling position, torque and now multirobots and force. Recently our MQP controller has been enhanced with visual servoing [45] that makes it a multimodal controller as vision, force, impedance/admittance, and position tasks can be specified together [48]. Investigations in terms of stability have been conducted in [39]. Singularity problems are also considered in [49]. We are also conducting promising research using QP control as an adaptive controller, where gains of the actuators [50] and tasks are also part of decision variables. Additional issues appear to be important for further investigations. For example, what is the granularity of the objects that need to be considered and integrated as "robots" when we consider tasks such as gathering small pieces in a box (more linked to perception); how to extend the MQP framework to deal with deformable objects; and finally, how the formalism can be extended to other types of robots, such as complex wheeled robots, flying and sea robots, and cable robots.

REFERENCES

[1] C. Samson, M. Le Borgne, and B. Espiau, *Robot Control—The Task Function Approach*. Oxford, U.K.: Clarendon, 1991.

[2] N. Mansard, O. Khatib, and A. Kheddar, "A unified approach to integrate unilateral constraints in the stack of tasks," *IEEE Trans. Robot.*, vol. 25, no. 3, pp. 670–685, Jun. 2009.

[3] L. Sentis, J. Park, and O. Khatib, "Compliant control of multi-contact and center-of-mass behaviors in humanoid robots," *IEEE Trans. Robot.*, vol. 26, no. 3, pp. 483–501, Jun. 2010.

[4] Y. Nakamura, H. Hanafusa, and T. Yoshikawa, "Task-priority based redundancy control of robot manipulators," *Int. J. Robot. Res.*, vol. 6, no. 2, pp. 3–15, Jun. 1987.

[5] B. Siciliano and J.-J. E. Slotine, "A general framework for managing multiple tasks in highly redundant robotic systems," in *Proc. Int. Conf. Adv. Robot.*, Pisa, Italy, Jun. 19–22, 1991, vol. 2, pp. 1211–1216.

[6] K. Bouyarmane and A. Kheddar, "Using a multi-objective controller to synthesize simulated humanoid robot motion with changing contact configurations," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, San Fransico, CA, USA, Sep. 25–30, 2011, pp. 4414–4419.

[7] O. Kanoun, F. Lamiraux, and P.-B. Wieber, "Kinematic control of redundant manipulators: Generalizing the task-priority framework to inequality task," *IEEE Trans. Robot.*, vol. 27, no. 4, pp. 785–792, Aug. 2011.

[8] P. M. Wensing and D. E. Orin, "Generation of dynamic humanoid behaviors through task-space control with conic optimization," in *Proc. IEEE Int. Conf. Robot. Automat.*, Karlsruhe, Germany, May 2013, pp. 3088–3094.

[9] A. Escande, N. Mansard, and P.-B. Wieber, "Hierarchical quadratic programming: Fast online humanoid-robot motion generation," *Int. J. Robot. Res.*, vol. 33, no. 7, pp. 1006–1028, 2014.

[10] S. Kuindersma, F. Permenter, and R. Tedrake, "An efficiently solvable quadratic program for stabilizing dynamic locomotion," in *Proc. IEEE Int. Conf. Robot. Automat.*, Hong Kong, 2014, pp. 2589–2594.

[11] S. Feng, X. Xinjilefu, W. Huang, and C. G. Atkeson, "Optimization-based full body control for the DARPA robotics challenge," *J. Field Robot.*, vol. 32, no. 2, pp. 293–312, Mar. 2015.

[12] J. Vaillant *et al.*,"Multi-contact vertical ladder climbing by an HRP-2 humanoid," *Auton. Robots*, vol. 40, no. 3, pp. 561–580, Mar. 2016.

[13] K. Bouyarmane and A. Kheddar, "Multi-contact stances planning for multiple agents," in *Proc. IEEE Int. Conf. Robot. Automat.*, Shanghai, China, May 9–13, 2011, pp. 5246–5253.

[14] K. Bouyarmane and A. Kheddar, "Humanoid robot locomotion and manipulation step planning," *Adv. Robot.*, vol. 26, no. 10, pp. 1099–1126, 2012.

[15] J. Vaillant, K. Bouyarmane, and A. Kheddar, "Multi-character physical and behavioural interactions controller," *IEEE Trans. Vis. Comput. Graph.*, vol. 23, no. 6, pp. 1650–1662, Jun. 2017.

[16] G. Borghesan, E. Scioni, A. Kheddar, and H. Bruyninckx, "Introducing geometric constraint expressions into robot constrained motion specification and control," *IEEE Robot. Automat. Lett.*, vol. 1, no. 2, pp. 1140–1147, Jul. 2016.

[17] K. Kosuge, T. Itoh, T. Fukuda, and M. Otsuka, "Tele-manipulation system based on task-oriented virtual tool," in *Proc. IEEE Int. Conf. Robot. Automat.*, Nagoya, Japan, May 21–27, 1995, pp. 351–356.

[18] L. D. Joly and C. Andriot, "Imposing motion constraints to a force reflecting telerobot through real-time simulation of a virtual mechanism," in *Proc. IEEE Int. Conf. Robot. Automat.*, Nagoya, Japan, May 21–27, 1995, pp. 357–362.

[19] A. Kheddar, "Teleoperation based on the hidden robot concept," *IEEE Trans. Syst. Man Cybern. A*, vol. 31, no. 1, pp. 1–13, Jan. 2001.

[20] S. A. Bowyer, B. L. Davies, and F. Rodriguez y Baena, "Active constraints/virtual fixtures: A survey," *IEEE Trans. Robot.*, vol. 30, no. 4, pp. 138–157, Feb. 2014.

[21] C. Ott, B. Bäuml, C. Borst, and G. Hirzinger, "Employing cartesian impedance control for the opening of a door: A case study in mobile manipulation," in *Proc. IFAC Symp. Intell. Auton. Veh.*, Toulouse, France, 2007, pp. 349–354.

[22] S. Dalibard, A. Nakhaei, F. Lamiraux, and J.-P. Laumond, "Manipulation of documented objects by a walking humanoid robot," in *Proc. IEEE RAS Int. Conf. Humanoid Robots*, Nashville, TN, USA, Dec. 6–8, 2010, pp. 518–523.

[23] D. Berenson, S. Srinivasa, and J. Kuffner, "Task space regions: A framework for pose-constrained manipulation planning," *Int. J. Robot. Res.*, vol. 30, no. 12, pp. 1435–1460, 2011.

[24] F. Burget, A. Hornung, and M. Bennewitz, "Whole-body motion planning for manipulation of articulated objects," in *Proc. IEEE Int. Conf. Robot. Automat.*, Karlsruhe, Germany, May 6–10, 2013, pp. 1656–1662.

[25] Y. Karayiannidis, C. Smith, F. E. V. Barrientos, P. Ögren, and D. Kragic, "An adaptive control approach for opening doors and drawers under uncertainties," *IEEE Trans. Robot.*, vol. 32, no. 1, pp. 161–175, Feb. 2016.

[26] H. Bruyninckx and J. De Schutter, "Specification of force-controlled actions in the "task frame formalism"—A synthesis," *IEEE Trans. Robot. Automat.*, vol. 12, no. 4, pp. 581–589, Aug. 1996.

[27] L. Villani and J. De Schutter, "Force control," in *Handbook of Robotics*. Berlin, Germany: Springer-Verlag, 2008, pp. 161–185.

[28] Y. Abe, M. da Silva, and J. Popović, "Multiobjective control with frictional contacts," in *Proc. Eurographics/ACM SIGGRAPH Symp. Comput. Animation.*, San Diego, CA, USA, Aug. 2–4, 2007, pp. 249–258.

[29] C. Collette, A. Micaelli, C. Andriot, and P. Lemerle, "Dynamic balance control of humanoids for multiple grasps and non coplanar frictional contacts," in *Proc. IEEE/RAS Int. Conf. Humanoid Robots*, Pittsburgh, PA, USA, Nov. 29–Dec. 1, 2007, pp. 81–88.

[30] M. de Lasa, I. Mordatch, and A. Hertzmann, "Feature-based locomotion controllers," *ACM Trans. Graph. (SIGGRAPH)*, vol. 29, no. 4, pp. 1–10, 2010.

[31] K. Bouyarmane, J. Vaillant, F. Keith, and A. Kheddar, "Exploring humanoid robots locomotion capabilities in virtual disaster response scenarios," in *Proc. IEEE RAS Int. Conf. Humanoid Robots*, Osaka, Japan, Nov. 2012, pp. 337–342.

[32] J. Salini, S. Barthélemy, and P. Bidaud, *LQP-Based Controller Design for Humanoid Whole-Body Motion*. Berlin, Germany: Springer, 2010, pp. 177–184.

[33] J. Salini, V. Padois, and P. Bidaud, "Synthesis of complex humanoid whole-body behavior: A focus on sequencing and tasks transitions," in *Proc. IEEE Int. Conf. Robot. Automat.*, Shanghai, China, May 9–13, 2011, pp. 1283–1290.

[34] A. Herzog, L. Righetti, F. Grimminger, P. Pastor, and S. Schaal, "Balancing experiments on a torque-controlled humanoid with hierarchical inverse dynamics," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, Chicago, IL, USA, Sep. 2014, pp. 981–988.

[35] M. Liu, A. Micaelli, P. Evrard, A. Escande, and C. Andriot, "Interactive virtual humans: A two-level prioritized control framework with wrench bounds," *IEEE Trans. Robot.*, vol. 28, no. 6, pp. 1309–1322, Dec. 2012.

[36] L. Righetti and S. Schaal, "Quadratic programming for inverse dynamics with optimal distribution of contact forces," in *Proc. IEEE RAS Int. Conf. Humanoid Robots*, Osaka, Japan, Nov. 2012, pp. 538–543.

[37] R. Featherstone, *Rigid Body Dynamics Algorithms*. New York, NY, USA: Springer, 2008.

[38] D. Stewart and J. C. Trinkle, "An implicit time-stepping scheme for rigid body dynamics with coulomb friction," in *Proc. IEEE Int. Conf. Robot. Automat.*, 2000, vol. 1, pp. 162–169.

[39] K. Bouyarmane and A. Kheddar, "On weight-prioritized multitask control of humanoid robots," *IEEE Trans. Autom. Control*, vol. 14, no. 6, pp. 2343–2352, Jun. 2018.

[40] D. M. Gorinevsky, A. M. Formalsky, and A. Y.-U. Schneider, *Force Control of Robotics Systems*. Boca Raton, FL, USA: CRC Press, 1997.

[41] B. Siciliano and L. Villani, *Robot Force Control (The Springer International Series in Engineering and Computer Science 540)*. New York, NY, USA: Springer, 1999.

[42] J. De Schutter *et al.*, "Estimating first-order geometric parameters and monitoring contact transitions during force-controlled compliant motion," *Int. J. Robot. Res.*, vol. 18, no. 12, pp. 1161–1184, 1999.

[43] J. De Schutter *et al.*, "Constraint-based task specification and estimation for sensor-based robot systems in the presence of geometric uncertainty," *Int. J. Robot. Res.*, vol. 26, no. 5, pp. 433–455, 2007.

[44] S. Caron, Q.-C. Pham, and Y. Nakamura, "Stability of surface contacts for humanoid robots: closed-form formulae of the contact wrench for rectangular support areas," in *Proc. IEEE RAS Int. Conf. Robot. Automat.*, Seattle, WA, USA, May 26–30, 2015, pp. 5107–5112.

[45] A. Paolillo, K. Chappellet, A. Bolotnikova, and A. Kheddar, "Interlinked visual tracking and robotic manipulation of articulated objects," *IEEE Robot. Automat. Lett.*, vol. 3, no. 4, pp. 2746–2753, Oct. 2018.

[46] D. J. Agravante, A. Sherikov, P.-B. Wieber, A. Cherubini, and A. Kheddar, "Walking pattern generators designed for physical collaboration," in *Proc. IEEE Int. Conf. Robot. Automat.*, Stockholm, Sweden, May 16–21, 2016, pp. 1573–1578.

[47] J. Jovic, A. Escande, K. Ayusawa, E. Yoshida, A. Kheddar, and G. Venture, "Humanoid and human inertia parameter identification using hierarchical optimization," *IEEE Trans. Robot.*, vol. 32, no. 3, pp. 726–735, Jun. 2016.

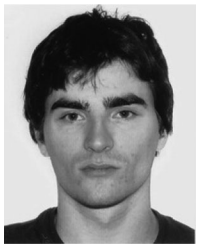
[48] A. Bolotnikova *et al.*, "A circuit-breaker use-case operated by a humanoid in aircraft manufacturing," in *Proc. IEEE Conf. Automat. Sci. Eng.*, X'ian, China, Aug. 20–23, 2017, pp. 15–22.
[49] K. Pfeiffer, A. Escande, and A. Kheddar, "Singularity resolution in equality and inequality constrained hierarchical task-space control by adaptive nonlinear least squares," *IEEE Robot. Automat. Lett.*, vol. 3, no. 4, pp. 3630–3637, Oct. 2018.
[50] V. Samy, K. Bouyarmane, and A. Kheddar, "QP-based adaptive-gains compliance control in humanoid falls," in *Proc. IEEE Int. Conf. Robot. Automat.*, Singapore, May 29–Jun. 3, 2017, pp. 4762–4767.

Karim Bouyarmane (M'17) received the double Ingénieur Diploma from the École Polytechnique, Palaiseau, France, in 2007 and from the École Nationale Supérieure des Mines de Paris, Paris, France, in 2008, and the Ph.D. degree from the University of Montpellier, Montpellier, France, in 2011.

Before receiving the Ph.D. degree, he completed a full-time Ph.D. program in the Joint Robotics Laboratory, National Institute of Advanced Industrial Science and Technology, Tsukuba, Japan. He subsequently held a Japan Society for the Promotion of Science (JSPS) Postdoctoral Fellowship with the Computational Neuroscience Laboratories Department, Advanced Telecommunications Research Institute International, Kyoto, Japan, working on brain–robot interfaces. He conducted a CNRS fixed-term contract researcher with the Laboratory of Informatics, Robotics and Microelectronics of Montpellier (LIRMM). He is currently an Assistant Professor with the University of Lorraine, Lorraine, France.

Kevin Chappellet received the Master's degree in computer sciences and applied mathematics (Geometry, Image, and CAD) from Joseph Fourier University, Grenoble, France, in 2014. He is currently pursuing the Ph.D. degree in manufacturing humanoid technologies from the Interactive Digital Human Group, CNRS-UM Laboratory of Informatics, Robotics, and Microelectronics, Montpellier, France.

He is currently a Research Engineer with the Interactive Digital Human Group, CNRS-UM Laboratory of Informatics, Robotics, and Microelectronics, where he is responsible for the software development and experiments on the HRP-4 humanoid robot. He was on an internship with the French National Institute for computer science and applied mathematics, Grenoble, France. He was with the IMAGINE group working on the calculation of the pressure forces of a garment on a character in a WebGL rendering.



Joris Vaillant received the Master's degree in interactive systems from Paul Sabatier University, Toulouse, France, in 2010 and the Ph.D. degree in robotics from the University of Montpellier, Montpellier, France, in May 2015.

He conducted his Ph.D. research at the Interactive Digital Human group, CNRS-UM Laboratory of Informatics, Robotics, and Microelectronics, Montpellier, France. He was with Abolis Biotechnologies, Paris, and he is currently Research and Development Engineer with Navya, Paris, France. His research interests include contact planning and whole body motion for humanoids and virtual avatars.

Abderrahmane Kheddar (SM'12) received the B.S. in computer science from the Institut National d'Informatique (ESI), Bab Ezzouar, Algeria, and the M.Sc. and Ph.D. degrees in robotics, from the University of Pierre and Marie Curie, Paris 6, Paris, France in 1993 and 1997.

He is currently the Director of Research with the CNRS, Paris, France. He is the Co-Director of the CNRS-AIST Joint Robotic Laboratory (JRL), UMI3218/RL, Tsukuba, Japan, and a Leader of the Interactive Digital Humans (IDH) team, CNRS-UM LIRMM, Montpellier, France. His research interests include humanoid robotics, haptics, and thought-based control using brain machine interfaces.

Dr. Kheddar is currently a founding member of the IEEE/RAS chapter on haptics (acting also as a senior advisor), the Co-Chair and co-founding member of the IEEE/RAS Technical committee on model-based optimization, and a member of the steering committee of the IEEE Brain Initiative. He is currently an Editor of the IEEE TRANSACTIONS ON ROBOTICS, the *Journal of Intelligent and Robotic Systems*, and *Frontiers in Bionics*; he is a founding member of the IEEE TRANSACTIONS ON HAPTICS and served on its editorial board during three years (2007–2010), he was also an Associate Editor for the *MIT Press Presence journal*. He is currently a member of the National Academy of Technologies of France (NATF), knight in the National Order of Merit, and a Senior Member of the IEEE Society.