



HAL
open science

Residual-based contacts estimation for humanoid robots

Fabrizio Flacco, Antonio Paolillo, Abderrahmane Kheddar

► **To cite this version:**

Fabrizio Flacco, Antonio Paolillo, Abderrahmane Kheddar. Residual-based contacts estimation for humanoid robots. Humanoids 2016 - 16th IEEE-RAS International Conference on Humanoid Robots, Nov 2016, Cancun, Mexico. pp.409-415, 10.1109/HUMANOIDS.2016.7803308 . hal-01546332

HAL Id: hal-01546332

<https://hal.science/hal-01546332>

Submitted on 23 Jun 2017

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Residual-based contacts estimation for humanoid robots

Fabrizio Flacco, Antonio Paolillo, Abderrahmane Kheddar

Abstract—The residual method for detecting contacts is a promising approach to allow physical interaction tasks with humanoid robots. Nevertheless, the classical formulation, as developed for fixed-base robots, cannot be directly applied to floating-base systems. This paper presents a novel formulation of the residual based on the *floating-base dynamics* modeling of humanoids. This leads to the definition of the *internal* and *external* residual. The first estimates the joints effort due to the external perturbation acting on the robot. The latter is an estimation of the external forces acting on the floating-base of the robot. The potential of the method is shown proposing a simple internal residual-based reaction strategy, and a procedure for estimating the contact point that combines both the internal and external residuals.

I. INTRODUCTION

Humanoids must be embedded with capabilities of multi-contact interaction with their environment or with the human. Indeed, to operate in confined spaces, it is necessary to rely on multi-contact motion and detect unexpected and undesired contacts on the robot originated from the environment or the colocated workers. Motion planning, with knowledge of the robot dynamics, the environment model, and the contacts, gives the *nominal* plan to accomplish the desired task, e.g. open doors [1], climb stairs or ladders [2], move across uneven terrain [3], etc.

Visual techniques to evaluate the appearance of the environment improved substantially, e.g. positioning and classification of objects [4], identification of possible contact points [3], localization of the robot [5] and mapping of the environment [6]— thanks to the development of new affordable and effective sensors, such as RGB-D technology.

Meanwhile, the estimation of the interaction forces between the humanoid robot and its surrounding, as well as the identification of how these forces affect the robot *internal* (joint level) and *external* (floating-base level) motions, rely on embedded force/torque (F/T) sensors, or on whole-body tactile sensing technologies that are still not yet readily available [7] for a whole-body coverage under humanoid constraints. F/T sensors measurements—that are found only at terminal points (feet or wrists), include (interfere with) the forward links load. In such cases, the interaction forces acting on the robot can be indirectly reconstructed, e.g. by looking at the equilibrium status of the robot [8].

The authors are with CNRS-University of Montpellier, Laboratoire d'Informatique de Robotique et de Microélectronique de Montpellier (LIRMM), IDH Group, 34095 Montpellier Cedex 5, France. Email: {paolillo,kheddar}@lirmm.fr.

This work is supported by the European Commission, within the H2020 COMANOID project (www.comanoid.eu), and by the ROMEO 2 project, (www.projetromeo.com), bpifrance in the framework of the Structuring Projects of Competitiveness Clusters (PSPC).

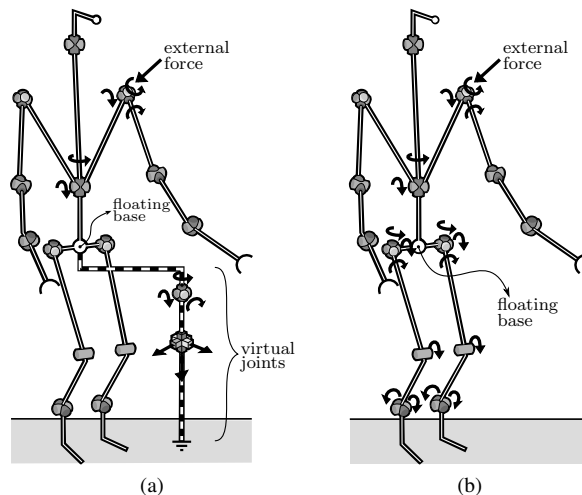


Fig. 1. Residual computation for humanoid robots. (a) Using a classical dynamic model, the effort due to an external force is unloading to the ground through the 6-D virtual joints composing the floating-base. (b) The use of a floating-base dynamic model, more appropriate to the nature of humanoids, allows to correctly compute the joint effort due to the external perturbation.

The measurement of the perturbation on the internal and external state of the robot is of crucial importance for achieving physical interaction tasks. In fact, these information can be used to (i) detect if the contact point is slipping, (ii) detect unforeseen collisions and react to them, (iii) to evaluate the dynamic characteristic of a contact, (iv) etc.

In this work we estimate the joint level effects due to external forces using proprioceptive sensors only: the momentum-based residual [9], that we name *classical residual*. This technique, initially developed for fixed-base manipulators, is outlined in II-B. In Sec. II-C, we show that its use for humanoid robots is not proper (see Fig. 1).

Our main contributions are: a novel *internal residual* that estimates correctly the effect of external forces at the joint level and its computation algorithms, which outperform that of the classical residual (Sec. III); a new *external residual* that estimates the effect of the sum of all external forces on the motion of the floating-base (Sec. V); the techniques for merging the internal and external residual to estimate the external contact forces (Sec. V) and even the location when there is a single contact point (Sec. VI); an application using the internal residual as feedback in a joint admittance controller to comply with an external perturbation (Sec. IV).

Throughout the manuscript, we present a series of Matlab simulation results using the dynamic model of a 34 degree of freedom robot HRP-4. We use the Spatial Vector and Rigid-

Body Dynamics library¹. The simulated low-level torque control is composed by gravity compensation plus a PD control of the joint positioning error. Contacts with the ground are simulated as a simple spring-damper system.

II. BACKGROUND

A. Spatial algebra

We briefly recall spatial vector algebra and notation. The reader may refer to [10], [11] and [12] for more details.

Consider the i -th (rigid) link of the robot kinematic chain: We can express its motion and the force that acts on it with the so-called *spatial motion* and *force* vectors, that are elements of two dual vector spaces, \mathcal{M}^6 and \mathcal{F}^6 . With reference to Fig. 2, $\mathbf{v} \in \mathcal{M}^6$ and $\mathbf{f} \in \mathcal{F}^6$ represent the spatial velocity of the body and the spatial force acting on it. It is convenient to express the coordinate vector of a spatial vector. For example, ${}^i\mathbf{v} \in \mathbb{R}^6$ and ${}^i\mathbf{f} \in \mathbb{R}^6$ are the coordinate vectors representing \mathbf{v} and \mathbf{f} in \mathcal{F}_i , the i -th link reference frame attached to the i -th joint. The vector ${}^i\mathbf{v}$ collects the angular and linear velocity around and along the axes of the frame \mathcal{F}_i . Similarly, ${}^i\mathbf{f}$ is filled with the moment and the force acting on the body, and expressed with respect to \mathcal{F}_i .

The coordinate vectors can be easily transformed from a frame \mathcal{F}_A to a frame \mathcal{F}_B :

$${}^B\mathbf{v} = {}^B\mathbf{X}_A {}^A\mathbf{v}, \text{ and } {}^B\mathbf{f} = {}^B\mathbf{X}_A^* {}^A\mathbf{f}. \quad (1)$$

By virtue of the duality existing between the motion and force vector spaces, it is ${}^B\mathbf{X}_A^* = {}^B\mathbf{X}_A^{-T}$. The transform matrix can be expressed as follows:

$${}^B\mathbf{X}_A = \begin{bmatrix} {}^B\mathbf{E}_A & \mathbf{0} \\ -\mathbf{t}_{AB} \times {}^B\mathbf{E}_A & {}^B\mathbf{E}_A \end{bmatrix} \quad (2)$$

where ${}^B\mathbf{E}_A$ is the rotation matrix from \mathcal{F}_A to \mathcal{F}_B , and $\mathbf{t}_{AB} \times$ is the skew symmetric matrix associated to the position vector of the origin of \mathcal{F}_B in \mathcal{F}_A ; $\mathbf{0}$ is a 3×3 zero matrix².

Finally, the spatial inertia \mathbf{I} of the link is a 6×6 matrix that includes information about the mass of the body, the position of the center of mass expressed in \mathcal{F}_i and the rotational inertia around the center of mass. The spatial inertia can be easily transformed from frame \mathcal{F}_A to \mathcal{F}_B :

$${}^B\mathbf{I} = {}^B\mathbf{X}_A^* {}^A\mathbf{I} {}^A\mathbf{X}_B. \quad (3)$$

The spatial inertia of two bodies i and j can be merged in a composite rigid body, simply by summing their spatial inertia expressed in a common frame ${}^A\mathbf{I}^c = {}^A\mathbf{I}_i + {}^A\mathbf{I}_j$.

B. Residual estimator

The dynamic model of a rigid fixed-base robot is

$$\mathbf{H}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{c}(\mathbf{q}, \dot{\mathbf{q}}) + \mathbf{g}(\mathbf{q}) = \boldsymbol{\tau} - \boldsymbol{\tau}_\mu(\mathbf{q}, \dot{\mathbf{q}}) + \boldsymbol{\tau}_c, \quad (4)$$

where $\mathbf{H}(\mathbf{q})$ is the robot inertia matrix, $\mathbf{c}(\mathbf{q}, \dot{\mathbf{q}})$ the centrifugal and Coriolis term, and $\mathbf{g}(\mathbf{q})$ accounts for the gravity; \mathbf{q} , $\dot{\mathbf{q}}$ and $\ddot{\mathbf{q}}$ are the vector of the n joint generalized coordinates and its first and second time derivative, respectively³. $\boldsymbol{\tau}$ are

¹Online: royfeatherstone.org/spatial/v2/index.html

²We denote respectively with $\mathbb{1}$ and $\mathbf{0}$ the identity matrix and the zero matrix –dimension has to be inferred from the context.

³In the rest of the paper we get rid of the dependence to the generalized coordinates (e.g. $\mathbf{H}(\mathbf{q}) \rightarrow \mathbf{H}$) as much as possible.

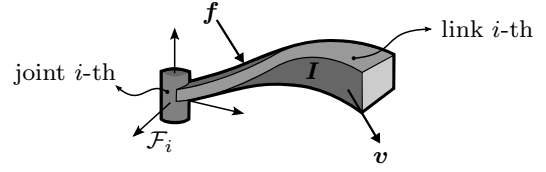


Fig. 2. Schematic representation of the i -th robot link and variables \mathbf{v} (the spatial velocity), \mathbf{f} (spatial force), and \mathbf{I} (spatial inertia matrix), expressed w.r.t. the reference frame \mathcal{F}_i , placed at the i -th joint.

the joint torques applied by the motors, $\boldsymbol{\tau}_\mu(\mathbf{q}, \dot{\mathbf{q}})$ is torques due to friction and $\boldsymbol{\tau}_c$ represents those due to external contact forces. Considering k external forces \mathbf{f}_c , with $c = 1 \dots k$, their action on the joints is given by

$$\boldsymbol{\tau}_c = \sum_{c=1}^k \mathbf{J}_c^T \mathbf{f}_c. \quad (5)$$

The $n \times m$ Jacobian \mathbf{J}_c can be associated either to the contact point ($m = 3$), at which the three-dimensional (3-D) contact force \mathbf{f}_c acts, or to the body where the six-dimensional (6-D) spatial force \mathbf{f}_c is applied ($m = 6$). In both cases, the Jacobian matrix takes into account the kinematic chain from the contact location to the robot base frame. Thus, the external forces \mathbf{f}_c are expressed in the base frame.

If the robot has F/T sensors to measure some of the external forces acting on the robot $\mathbf{f}_{c,\text{F/T}}$, it is possible to divide $\boldsymbol{\tau}_c$ in a measured part $\bar{\boldsymbol{\tau}}_c$ and an unknown part $\hat{\boldsymbol{\tau}}_c$

$$\boldsymbol{\tau}_c = \bar{\boldsymbol{\tau}}_c + \hat{\boldsymbol{\tau}}_c = \sum_{c=1}^{\bar{k}} \mathbf{J}_c^T \mathbf{f}_{c,\text{F/T}} + \sum_{c=1}^{\hat{k}} \mathbf{J}_c^T \mathbf{f}_c, \quad (6)$$

with $k = \bar{k} + \hat{k}$.

The knowledge of the robot dynamic model, and the measurement of \mathbf{q} , $\dot{\mathbf{q}}$ and $\boldsymbol{\tau}_m = \boldsymbol{\tau} - \boldsymbol{\tau}_\mu + \bar{\boldsymbol{\tau}}_c$ enables the computation of the so-called *classical residual* vector [9]

$$\mathbf{r} = K_i \left[\mathbf{H}\dot{\mathbf{q}} - \int_0^t (\boldsymbol{\tau}_m + \mathbf{C}^T \dot{\mathbf{q}} - \mathbf{g} + \mathbf{r}) ds \right] \quad (7)$$

where $K_i > 0$ is the residual gain, and $\mathbf{H}\dot{\mathbf{q}}$ is the generalized angular momentum of the robot. The centrifugal and Coriolis term has been factorized with the form $\mathbf{c}(\mathbf{q}, \dot{\mathbf{q}}) = \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}}$, that is built from Cristoffel symbols of second kind. By evaluating the time derivative of \mathbf{r} , and exploiting the well known relation $\dot{\mathbf{H}} = \mathbf{C} + \mathbf{C}^T$, it is straightforward to show that the residual \mathbf{r} is a filtered version of $\hat{\boldsymbol{\tau}}_c$ (ideally, $K_i \rightarrow \infty \Rightarrow \mathbf{r} = \hat{\boldsymbol{\tau}}_c$) and can be used for estimating the perturbation of the robot joints due to external forces. Using the gain K_i it is possible to control the bandwidth of the residual. With large K_i the signal will be more reactive, but also more sensitive to noise. By lowering K_i , it is possible to filter the noise, but the response will be slower. A discrete time implementation of the residual is presented in [13]. The factorized centrifugal and Coriolis term \mathbf{C} can be computed with a modified version of the Newton-Euler method [14]. Note however that the total computational complexity for obtaining \mathbf{C} is $\mathcal{O}(n^2)$, since n instances of the Newton-Euler

routine needs to be called. Therefore, this procedure could be time consuming for robots with a large number of joints.

The residual approach proved to be efficient with fixed-base robots. Initially, it was proposed for detecting and isolating actuators faults [15], then it was used for collision detection and reaction [9], estimating the stiffness of variable stiffness actuators [13] and the external forces [16].

C. The use of the classical residual with floating-base robots

When the robot does not have a fixed base, the dynamic model (4) is still commonly used by considering six virtual joints (three prismatic and three revolute) connected to a point of the robot, called *floating-base* (FB). These virtual joints allow taking into account the attitude of the robot with respect to a virtual base (world) frame.

The virtual dynamic model is then described by

$$\mathbf{H}_v \ddot{\mathbf{q}}_v + \mathbf{C}_v \dot{\mathbf{q}}_v + \mathbf{g}(\mathbf{q}_v) = \boldsymbol{\tau}_v - \boldsymbol{\tau}_{v,\mu} + \boldsymbol{\tau}_c, \quad (8)$$

where subscript v indicates the virtual base version of the matrices and vectors characterizing the robot dynamics. In this model, the $n + 6$ generalized coordinate vector \mathbf{q}_v is composed by the 6 virtual joint coordinates \mathbf{q}_{fb} , and the n robot joint coordinates \mathbf{q} , i.e. $\mathbf{q}_v = (\mathbf{q}_{fb}^T \quad \mathbf{q}^T)^T$. The under-actuation of the floating-base is represented by having zero motor torques associated to the virtual joints $\boldsymbol{\tau}_v = (\mathbf{0}^T \quad \boldsymbol{\tau}^T)^T$. The joint torque due to external contacts $\boldsymbol{\tau}_c$ is split into measured $\bar{\boldsymbol{\tau}}_c$ and unknown $\hat{\boldsymbol{\tau}}_c$ torques as in (6).

The classical residual-based estimator of $\hat{\boldsymbol{\tau}}_c$ (7) can be applied with the model (8) by adapting the matrices used to the ones of the virtual floating-base model, e.g. $\mathbf{H}_v \dot{\mathbf{q}}_v$ in place of $\mathbf{H} \dot{\mathbf{q}}$. This estimated torque is similar to the fixed-base case (6). In this case, the external force acting on the contact point follows the kinematic chain to the FB; hence, it is grounded to the virtual base frame.

Although the dynamic model (8) is consistent from an *external* point-of-view, that is, it correctly models how the robot behaves under the action of external forces, it is not correct from an *internal* point-of-view. In fact, the perturbations on the joints torque due to the external forces are not considered in a proper way. For instance, assume an external force acting on the right shoulder. According to this model, the torque due to the external force (5) will have nonzero values for those joints of the kinematic chain between the shoulder and the FB, and zero values for the joints of the legs. Clearly, this is not a realistic situation (see Fig. 1a). In fact, the joint torque perturbations depend on the position of the FB point, meaning that if it was positioned right on the contact point, the force would be completely grounded to the virtual joints, and no torques would be accounted for the robot joints. Thus, the estimation computed with the classical residual (7) is not able to perceive the effects of the external forces at the joint level, and cannot be used in the same way it is used for fixed-base manipulators.

To show this effect, we simulated a force applied on the right shoulder of the HRP-4. The robot task is to maintain the initial configuration. The external forces due to the ground reactions, applied to the foot are measured using

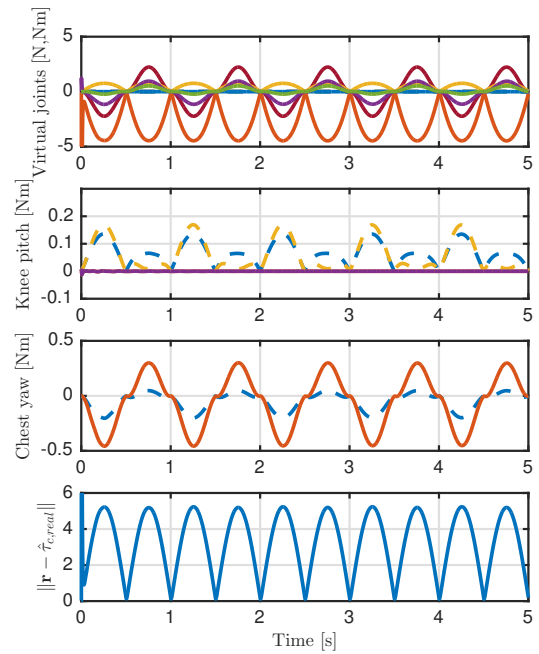


Fig. 3. Estimated joint force/torque due to the external contact (solid) and real values (dashed). The six virtual joints (top), the right and left knees pitch (middle up), the chest yaw (middle down) and norm of the error in estimating the real torque due to the unknown external force (bottom).

F/T sensors, therefore assumed as known. Figure 3 shows how the classical residual (7), applied on the model (8), is not able to estimate the real joint torque due to the external contact, that we obtained from the simulation ($\hat{\boldsymbol{\tau}}_{c,\text{real}}$). In fact, the classical residual estimates: torques/forces on the virtual joints, which are senseless; no torques on the legs, e.g. at the knees pitch joints; wrong values for the joints in the kinematic chain between the contact point and the floating-base point, e.g. at the chest yaw joint.

III. A NOVEL RESIDUAL FOR HUMANOID ROBOTS

In the previous Section, we showed that the classical residual estimator, based on the humanoid dynamic model with virtual joints for the floating-base, fails to estimate the internal joint torques due to external contacts. For overcoming this problem we have to consider the under-actuated nature of humanoid robots. To this end, we start from the floating-base dynamic model described in [12] (p. 181), which decouples the motion of the FB with the dynamic of the robot joints

$$\begin{bmatrix} \mathbf{I}_0^c & \mathbf{F} \\ \mathbf{0} & \mathbf{H}^{\text{fb}} \end{bmatrix} \begin{bmatrix} \mathbf{a}_0 \\ \ddot{\mathbf{q}} \end{bmatrix} + \begin{bmatrix} \hat{\mathbf{p}}_0^c \\ \mathbf{C}^{\text{fb}} \end{bmatrix} = \begin{bmatrix} \mathbf{0} \\ \boldsymbol{\tau} - \boldsymbol{\tau}_\mu \end{bmatrix} + \sum_{c=1}^k \begin{bmatrix} \mathbf{1} \\ \mathbf{J}_c^{\text{fb},T} \end{bmatrix}^0 \mathbf{X}_{\text{ref}}^* \mathbf{f}_c \quad (9)$$

where we highlighted the contribution of the k external forces \mathbf{f}_c , expressed in the world reference frame.

Let \mathcal{F}_0 be the frame attached to the FB, and \mathbf{a}_0 the acceleration of the FB. $\mathbf{H}^{\text{fb}} = \mathbf{H} - \mathbf{F}^T (\mathbf{I}_0^c)^{-1} \mathbf{F}$ is the *floating inertia matrix*, with \mathbf{H} the robot inertia matrix considering

Algorithm 1 Computing \hat{C} and \hat{p}_0^c

```
 $\mathbf{a}_0^{vp} = -{}^0\mathbf{a}_g$ 
for  $i = 1$  to  $n$  do
   $[\mathbf{X}_J, \mathbf{S}_i] = \text{jcalc}(\text{jtype}(i), q_i)$ 
   $\mathbf{v}_J = \mathbf{S}_i \dot{q}_i$ 
   ${}^i\mathbf{X}_{\lambda(i)} = \mathbf{X}_J \mathbf{X}_T(i)$ 
   $\mathbf{v}_i = {}^i\mathbf{X}_{\lambda(i)} \mathbf{v}_{\lambda(i)} + \mathbf{v}_J$ 
   $\mathbf{a}_i^{vp} = {}^i\mathbf{X}_{\lambda(i)} \mathbf{a}_{\lambda(i)}^{vp} + \mathbf{v}_i \times \mathbf{v}_J$ 
   $\mathbf{f}_i = \mathbf{I}_i \mathbf{a}_i^{vp} + \mathbf{v}_i \times^* \mathbf{I}_i \mathbf{v}_i$ 
end for
 $\mathbf{f}_0 = \mathbf{I}_0 \mathbf{a}_0^{vp} + \mathbf{v}_0 \times^* \mathbf{I}_0 \mathbf{v}_0$ 
for  $i = n$  to  $1$  do
   $\hat{C}_i = \mathbf{S}_i^T \mathbf{f}_i$ 
   $\mathbf{f}_{\lambda(i)} = \mathbf{f}_{\lambda(i)} + {}^i\mathbf{X}_{\lambda(i)}^T \mathbf{f}_i$ 
end for
 $\hat{p}_0^c = \mathbf{f}_0$ 
```

\mathcal{F}_0 as base, \mathbf{F} is the $6 \times n$ matrix whose columns are the spatial forces required at the floating-base to support unit acceleration about each joint variable and \mathbf{I}_0^c is the composite inertia of the whole robot. The 6-D vector \hat{p}_0^c is the spatial bias force for the composite rigid body comprising the whole floating-base system, and $\mathbf{C}^{\text{fb}} = \hat{C} - \mathbf{F}^T (\mathbf{I}_0^c)^{-1} \hat{p}_0^c$ is the *floating generalized bias force*, with \hat{C} joint level bias force⁴. All spatial quantities are expressed in \mathcal{F}_0 . For more details on how this model is derived, please refer to [12].

The last part of the model (9) represents the action of external forces that perturbs both the joints state and the motion of the FB. The coordinate transformation ${}^0\mathbf{X}_{\text{ref}}^*$ is used to transform external forces from the inertial reference frame (world) to \mathcal{F}_0 . The action of an external forces on the joint is guided by the floating Jacobian transposed $\mathbf{J}_c^{\text{fb},T} = \mathbf{J}_{0,c}^T - \mathbf{F}^T (\mathbf{I}_0^c)^{-1}$, where $\mathbf{J}_{0,c}$ is the $n \times m$ Jacobian of the contact point (body) with respect to the \mathcal{F}_0 .

As with the classical modeling, we can consider the joint torque contribution of known external forces as follows

$$\bar{\tau}_c = \sum_{c=1}^{\bar{k}} \mathbf{J}_c^{\text{fb},T} {}^0\mathbf{X}_{\text{ref}}^* \mathbf{f}_{c,\text{F/T}}. \quad (10)$$

To estimate the joint torques due to unknown external contacts, we build a new residual using the second line in (9)

$$\mathbf{r}_{\text{int}} = K_{i,\text{int}} \left[\mathbf{H}^{\text{fb}} \dot{\mathbf{q}} - \int_0^t (\tau_m + \dot{\mathbf{H}}^{\text{fb}} \dot{\mathbf{q}} - \mathbf{C}^{\text{fb}} + \mathbf{r}_{\text{int}}) ds \right] \quad (11)$$

with $K_{i,\text{int}} > 0$. This residual is also a first order filter of the joint torques due to external forces

$$\mathbf{r}_{\text{int}} \approx \hat{\tau}_c = \hat{\tau}_{c,\text{real}} = \sum_{c=1}^{\bar{k}} \mathbf{J}_c^{\text{fb},T} {}^0\mathbf{X}_{\text{ref}}^* \mathbf{f}_c. \quad (12)$$

Since this residual estimates the joint level effects of an external force, we call it *internal (floating-base) residual*.

⁴The hat superscription indicates that the bias action of external forces is not included, since we have expressed it separately.

Algorithm 2 Computing \mathbf{H} , \mathbf{F} , \mathbf{I}_0^c , $\dot{\mathbf{H}}$, $\dot{\mathbf{F}}$ and $\dot{\mathbf{I}}_0^c$

```
 $\mathbf{H} = \mathbf{0}$ ,  $\dot{\mathbf{H}} = \mathbf{0}$ 
for  $i = 1$  to  $n$  do
   $[\mathbf{X}_J, \mathbf{S}_i] = \text{jcalc}(\text{jtype}(i), q_i)$ 
   $\mathbf{v}_J = \mathbf{S}_i \dot{q}_i$ 
   ${}^i\mathbf{X}_{\lambda(i)} = \mathbf{X}_J \mathbf{X}_T(i)$ 
   ${}^i\dot{\mathbf{X}}_{\lambda(i)} = -\mathbf{v}_J \times {}^i\mathbf{X}_{\lambda(i)}$ 
   $\mathbf{I}_i^c = \mathbf{I}_i$ ,  $\dot{\mathbf{I}}_i^c = \mathbf{0}$ 
end for
 $\mathbf{I}_0^c = \mathbf{I}_0$ ,  $\dot{\mathbf{I}}_0^c = \mathbf{0}$ 
for  $i = n$  to  $1$  do
   $\mathbf{I}_{\lambda(i)}^c = \mathbf{I}_{\lambda(i)}^c + {}^i\mathbf{X}_{\lambda(i)}^T \mathbf{I}_i^c {}^i\mathbf{X}_{\lambda(i)}$ 
   $\dot{\mathbf{I}}_{\lambda(i)}^c = \dot{\mathbf{I}}_{\lambda(i)}^c + {}^i\mathbf{X}_{\lambda(i)}^T \dot{\mathbf{I}}_i^c {}^i\mathbf{X}_{\lambda(i)} +$ 
     ${}^i\dot{\mathbf{X}}_{\lambda(i)}^T \mathbf{I}_i^c {}^i\mathbf{X}_{\lambda(i)} + {}^i\mathbf{X}_{\lambda(i)}^T \mathbf{I}_i^c {}^i\dot{\mathbf{X}}_{\lambda(i)}$ 
   $\mathbf{F}_i = \mathbf{I}_i^c \mathbf{S}_i$ ,  $\dot{\mathbf{F}}_i = \dot{\mathbf{I}}_i^c \mathbf{S}_i$ 
   $\mathbf{H}_{ii} = \mathbf{S}_i^T \mathbf{F}_i$ ,  $\dot{\mathbf{H}}_{ii} = \mathbf{S}_i^T \dot{\mathbf{F}}_i$ 
   $j = i$ 
  while  $\lambda(j) \neq 0$  do
     $\dot{\mathbf{F}}_i = {}^j\mathbf{X}_{\lambda(j)}^T \dot{\mathbf{F}}_i + {}^j\dot{\mathbf{X}}_{\lambda(j)}^T \mathbf{F}_i$ ,  $\mathbf{F}_i = {}^j\mathbf{X}_{\lambda(j)}^T \mathbf{F}_i$ 
     $j = \lambda(j)$ 
     $\mathbf{H}_{ij} = \mathbf{S}_i^T \mathbf{F}_i$ ,  $\dot{\mathbf{H}}_{ij} = \mathbf{S}_i^T \dot{\mathbf{F}}_i$ 
     $\mathbf{H}_{ji} = \mathbf{H}_{ij}$ ,  $\dot{\mathbf{H}}_{ji} = \dot{\mathbf{H}}_{ij}$ 
  end while
   $\dot{\mathbf{F}}_i = {}^j\mathbf{X}_0^T \dot{\mathbf{F}}_i + {}^j\dot{\mathbf{X}}_0^T \mathbf{F}_i$ ,  $\mathbf{F}_i = {}^j\mathbf{X}_0^T \mathbf{F}_i$ 
end for
```

The time derivative of the floating-base inertia matrix $\dot{\mathbf{H}}^{\text{fb}}$ can be obtained analytically as

$$\dot{\mathbf{H}}^{\text{fb}} = \dot{\mathbf{H}} - \dot{\mathbf{F}}^T (\mathbf{I}_0^c)^{-1} \mathbf{F} - \mathbf{F}^T (\mathbf{I}_0^c)^{-1} \dot{\mathbf{F}} - \mathbf{F}^T \left(-(\mathbf{I}_0^c)^{-1} \dot{\mathbf{I}}_0^c (\mathbf{I}_0^c)^{-1} \right) \mathbf{F}. \quad (13)$$

Moreover, \hat{C} and \hat{p}_0^c can be computed using the Algorithm 1, and \mathbf{H} , \mathbf{F} , \mathbf{I}_0^c and their time derivative using Algorithm 2, which is an extension of the method proposed in [12] (p. 183). In these algorithms, ${}^0\mathbf{a}_g$ is the gravity acceleration expressed in frame 0, $\lambda(i)$ is the body index of the parent of body i ($0 \leq \lambda(i) < i$). $\text{jcalc}(\cdot)$ is a function that according to the type of joint $\text{jtype}(\cdot)$ (e.g. revolute or prismatic) returns the joint coordinate transform \mathbf{X}_J (for revolute joint it is a pure rotation of q_i around the z -axis), and the joint motion subspace vector \mathbf{S}_i that describe the motion axis of the joint (for revolute joint $\mathbf{S}_i = [0 \ 0 \ 1 \ 0 \ 0 \ 0]^T$).

Summing up, the information needed to compute the internal residual (11) are: joint generalized position \mathbf{q} and velocity $\dot{\mathbf{q}}$; FB rotation w.r.t the virtual base frame (for computing ${}^0\mathbf{a}_g$) and FB velocity \mathbf{v}_0 ; joint motor torque τ , joint friction torque τ_μ and joint torques due to known external forces $\bar{\tau}_c$, for computing τ_m .

Thanks to Algorithms 1 and 2, the internal residual (11) has a lower computational load w.r.t the classical residual (7). In fact, the algorithm complexity is $\mathcal{O}(n)$. For instance, the mean computational time to obtain the classical residual for HRP-4 in the previous simulative example is 0.0795 s, while

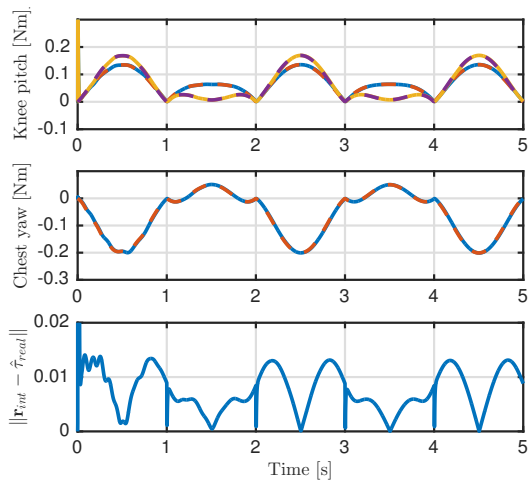


Fig. 4. Estimated joint torque due to the external force (solid) and real values (dashed). The right and left knees pitch (top) and the chest yaw (middle) and norm of the error in estimating the real torque (bottom).

it is 0.005 s for the internal residual, almost 16 times faster.

We simulated the same sinusoidal external force applied on the right shoulder of the HRP-4 presented in the previous section. Fig. 4 shows the estimated joint torques, using the presented internal residual. The estimated torque at the knees pitch and at the chest yaw joints are now correct, as well as all the other joints (that we omit for lack of space). Note that, compared to Fig. 3, the virtual joints are not presented as their values are identically zero.

IV. AN APPLICATION OF THE INTERNAL RESIDUAL

The estimation of the joint torques due to external perturbation, as computed by the internal residual (11) can be used in many ways, for example in the framework of physical human-robot interaction. In order to show the potential and the effectiveness of the proposed method, we propose a possible application where HRP-4 is requested to move on the direction of the external forces without knowing neither their number, the contact points nor the force vector. For this purpose, we implemented a simple joint admittance control that generates the desired joint configuration as

$$\dot{q}_d = \mathbf{K}_a \hat{\mathbf{r}}_{c,\text{real}}, \quad \mathbf{q}_d = \mathbf{q} + \dot{q}_d T \quad (14)$$

where \mathbf{K}_a is a diagonal matrix of gains and T is the simulation sampling time.

In this example, the robot stands at its position for 15 s; at $t = 5$ s, we applied two downwards forces on the robot shoulders, with a polynomial profile increasing till $t = 8.5$ s and then decreasing till $t = 12$ s. The maximum value of the forces was set to 5 N. As if a human push down the robot expecting that the robot bends for a seating configuration. We compared the results obtained by estimating $\hat{\mathbf{r}}_{c,\text{real}}$ with \mathbf{r} , as computed with (7), and \mathbf{r}_{int} , given by (11).

Fig. 5 shows snapshots of the simulation performing the control with the classic residual, while Fig. 6 shows snapshots of the simulation obtained using the internal residual. The entire motion can be seen in the accompanying video.

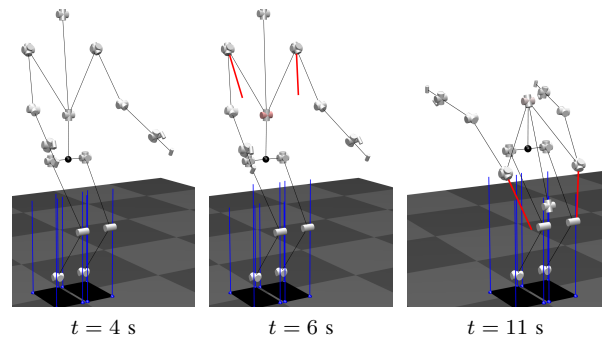


Fig. 5. Reaction strategy to external perturbations using the classic residual: simulation results. The joints effort due to the external forces (red lines) is felt only at the pitch joint of the chest (gradually highlighted in red in proportion to the effort). As a result, the admittance control acts only on this joints, preventing an effective reaction strategy.

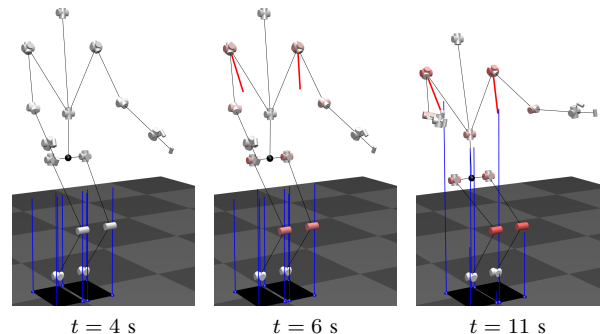


Fig. 6. Reaction strategy to external perturbations using the internal residual: simulation results. The joints effort due to the external forces (red lines) is felt at all the involved joints of the kinematic chain (gradually highlighted in red in proportion to the effort). The admittance control implements a whole-body motion, ensuring an effective reaction strategy.

As explained in Sec. II-B, the classical residual does not allow a whole-body reaction strategy, preventing a proper robot behavior when trying to comply with external forces. On the other side, the simulation that makes use of the internal residual confirms the expectation of Sec. III. It allows to perceive the effort due to the external force at any interested joints of the robot. As a result, the robot can comply with the perturbation with more degrees of freedom and ensure a more effective reaction strategy.

V. EXTERNAL FORCE ESTIMATION

Now, we consider one of the most important use of the residual: the estimation of the force applied on a contact point. In [17] it is described how the residual \mathbf{r} provides an estimation of the \hat{k} unmeasured external forces \mathbf{f}_c from the knowledge of the contact points location:

$$\begin{bmatrix} \mathbf{f}_1 \\ \vdots \\ \mathbf{f}_{\hat{k}} \end{bmatrix} = \begin{bmatrix} \mathbf{J}_1^T & \dots & \mathbf{J}_{\hat{k}}^T \end{bmatrix}^\# \mathbf{r}. \quad (15)$$

Successively, how this estimation can be exploited in the control is presented in [18]. The dynamic model of a floating-

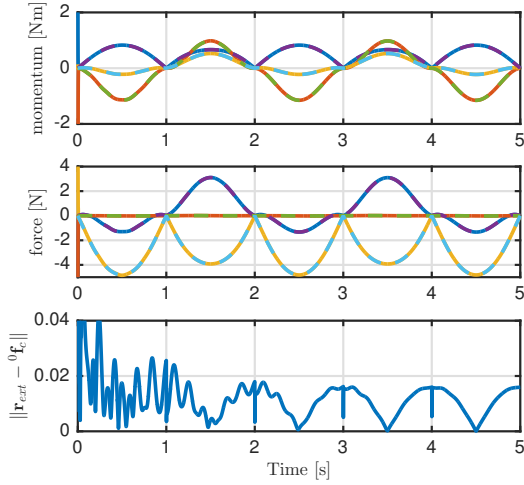


Fig. 7. Estimated total torque (top) and force (middle) due to the external contact (solid) and real values (dashed). The estimation error norm (bottom)

base robot with virtual joints (8) is not proper from an internal viewpoint (Sec. II-C), but the external behavior of the robot is correctly modeled. This is similarly observed in the estimation of the external forces. In fact, while the residual estimator (7) is not able to represent properly the internal torques, the estimated external forces using (15) are correct. An example of the use of the classical residual for estimating external forces on a humanoid robot is presented in [19], where a particle filter takes advantage of the assumption of the knowledge of the friction cone of the contact to estimate the contact point and the applied force. Yet, we show that the use of the floating-base dynamic model (9) is more flexible because it allows to decouple the internal and external robot behavior due to external forces. Thus, these information can be used separately or combined for estimating the external forces applied to the robot.

Considering the upper row of the decoupled dynamic model (9), we have a relation between the robot motion and the total external forces applied to the robot. Taking into account this 6-D information, we can build a novel residual that we call *external (floating-base) residual*:

$$\mathbf{r}_{\text{ext}} = K_{i,\text{ext}} \left[\mathbf{I}_0^c \mathbf{v}_0 + \mathbf{F} \dot{\mathbf{q}} - \int_0^t (\dot{\mathbf{I}}_0^c \mathbf{v}_0 + \dot{\mathbf{F}} \dot{\mathbf{q}} - \hat{\mathbf{p}}_0^c + \bar{\mathbf{f}}_{\text{sum}} + \mathbf{r}_{\text{ext}}) ds \right] \quad (16)$$

where $K_{i,a} > 0$ is a residual gain matrix, and $\bar{\mathbf{f}}_{\text{sum}} = \sum_{c=1}^k {}^0 \mathbf{X}_{\text{ref}}^* \mathbf{f}_{c,\text{F/T}}$ is the sum of all measured forces. Note that all matrices to compute the external residual (16) are obtained with Algorithms 1 and 2. Moreover, this residual is much more reliable, since it does not need the measure of the motors torque and the estimation of joints friction behavior, which are usually more difficult to obtain. As previously, it is simple to show that \mathbf{r}_{ext} is a first order filter of the sum of all unknown external forces applied on the robot (expressed

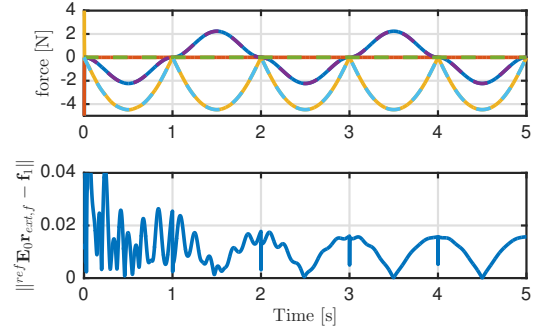


Fig. 8. External force due to a single external contact (top): estimated (solid) and real values (dashed). Norm of the estimation error (bottom)

in \mathcal{F}_0 frame)

$$\mathbf{r}_{\text{ext}} \approx \sum_{c=1}^k {}^0 \mathbf{X}_{\text{ref}}^* \mathbf{f}_c. \quad (17)$$

Using the external residual on the example initially proposed in Sec. II-C, where a single sinusoidal force is applied to the right shoulder, we obtained the estimation of the force (expressed in the floating-base frame) as shown in Fig. 7.

With this new tool, if one need to know just the resultant of all external forces applied to the robot, e.g. for push recovering, you can use it directly. If we assume to have a single unmeasured 3-D force \mathbf{f}_1 applied to one contact point, whose position \mathbf{p}_1 is known/detected, its estimation is given directly by the last three components of \mathbf{r}_{ext} opportunely rotated. In fact, it is

$$\mathbf{r}_{\text{ext}} = \begin{bmatrix} \mathbf{r}_{\text{ext},m} \\ \mathbf{r}_{\text{ext},f} \end{bmatrix} \approx {}^0 \mathbf{X}_{\text{ref}}^* \begin{bmatrix} \mathbf{p}_1 \times \mathbf{f}_1 \\ \mathbf{f}_1 \end{bmatrix} \quad (18)$$

thus,

$$\mathbf{f}_1 \approx {}^{\text{ref}} \mathbf{E}_0 \mathbf{r}_{\text{ext},f}. \quad (19)$$

Figure 8 shows the reconstructed external force applied to the robot shoulders in the previous example.

When two or more unmeasured 3-D forces are applied to known robot points, the external residual has to be combined with the internal residual for estimating the forces. The most straightforward method is to use pseudo-inversion

$$\begin{bmatrix} \hat{\mathbf{f}}_1 \\ \vdots \\ \hat{\mathbf{f}}_k \end{bmatrix} = \begin{bmatrix} \mathbf{1} & \cdots & \mathbf{1} \\ \mathbf{J}_1^{\text{fb},T} & \cdots & \mathbf{J}_k^{\text{fb},T} \end{bmatrix}^{\#} {}^{\text{ref}} \mathbf{X}_0^* \begin{bmatrix} \mathbf{r}_{\text{ext}} \\ \mathbf{r}_{\text{int}} \end{bmatrix}. \quad (20)$$

When the two residual are combined, we should be aware that they use different sensor information, affected by different source of noise. The quality of the internal residual is mainly disturbed by the noise on the measure of the motor torque and by the estimated values of the joint friction, while the external sensor depends on the estimation of the orientation and the spatial velocity of the floating-base. Thus, we suggest to use different values for $K_{i,\text{int}}$ and $K_{i,\text{ext}}$.

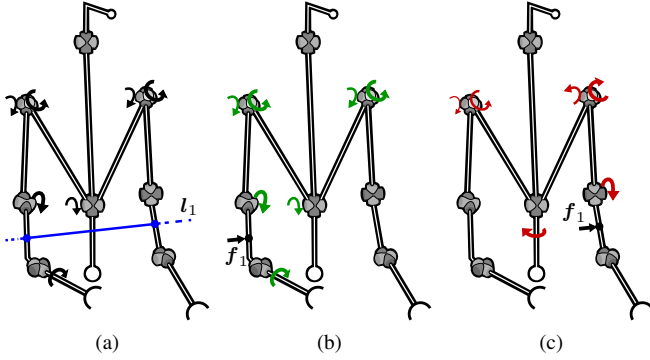


Fig. 9. Estimation of the external force and the contact point. The external residual estimates the contact force \mathbf{f}_1 and the line l_1 where it lies on. The intersections of l_1 with the robot links localize the possible contact points (a). These points are used to compute the Jacobian and, along with \mathbf{f}_1 , the hypothetic joint effort due to the external perturbation as it was acting on those points (b), (c). The real contact point corresponds to the one producing the joint efforts that matches with the internal residual.

VI. CONTACT POINTS DETECTION AND FUTURE WORKS

One of the more challenging issue is finding the contact position where the external forces are applied, knowing just the number of contact points, or even estimating it. State-of-the-art methods are based on using external sensors, such as tactile skin [20] or depth camera [16].

We explore the possibility to combine the internal and external residual to estimate the position of unforeseen contacts, without using any extra sensors. Meanwhile, we present here how the new two residual can be combined for estimating both the contact position and the applied force, under the assumption of one single unforeseen contact. This situation is the same proposed in the previous Section, with the external residual estimates (18), but in this case the contact position \mathbf{p}_1 is considered unknown. Note that eq. (19) gives an estimation of the applied force, even if the contact position is unknown. At this point, we consider

$$\mathbf{r}_{\text{ext}}^{\text{ref}} = \begin{bmatrix} \mathbf{r}_{\text{ext},m}^{\text{ref}} \\ \mathbf{r}_{\text{ext},f}^{\text{ref}} \end{bmatrix} = {}^0\mathbf{X}_{\text{ref}}^T \mathbf{r}_{\text{ext}}, \quad (21)$$

where, trivially $\mathbf{r}_{\text{ext},f}^{\text{ref}} = \mathbf{E}_0 \mathbf{r}_{\text{ext},f}^{\text{ref}} \approx \mathbf{f}_1$. From the upper row of (18), we know that the contact point lies on the line

$$\mathbf{l}_1 = \frac{\mathbf{r}_{\text{ext},f}^{\text{ref}} \times \mathbf{r}_{\text{ext},m}^{\text{ref}}}{\|\mathbf{r}_{\text{ext},f}^{\text{ref}}\|^2} + l \mathbf{r}_{\text{ext},f}^{\text{ref}}, \quad l \in \mathbb{R}. \quad (22)$$

This line is oriented as \mathbf{f}_1 and passes through \mathbf{p}_1 .

Using a CAD representation of the robot, or a simpler representation with primitive shapes, we can individuate all possible contact points on the robot surface that intersect line l_1 . Thus, for all of them we compute the joint torque perturbation we would have observed if the force $\mathbf{r}_{\text{ext},f}^{\text{ref}}$ had applied on that point. By comparing those joint torques with the internal residual \mathbf{r}_{int} , we can evaluate which among the possible contact points is compatible with the estimated internal behavior, thus we can discriminate the correct contact position. This procedure is illustrated in Fig. 9.

For the experimental validation of the presented method on the real HRP-4, we plan to use the two residuals to refine

the parameters of the robot dynamic model. In fact, when no unknown contact are present, the residuals give an estimation of the discrepancy between the robot dynamics and the used model. From another side, we plan to improve the estimation of the position and velocity of the floating-base by fusing IMU and visual information.

ACKNOWLEDGEMENT

The authors would like to thank Prof. Roy Featherstone for his advice on the realization of the Algorithm 2.

REFERENCES

- [1] N. Banerjee, X. Long, R. Du, F. Polido, S. Feng, C. G. Atkeson, M. Gennert, and T. Padiar, "Human-supervised control of the ATLAS humanoid robot for traversing doors," in *IEEE-RAS International Conference on Humanoid Robots*, 2015, pp. 722–729.
- [2] J. Vaillant, A. Kheddar, H. Audren, F. Keith, S. Brossette, A. Escande, K. Bouyarmane, K. Kaneko, M. Morisawa, P. Gergondet, E. Yoshida, S. Kajita, and F. Kanehiro, "Multi-contact vertical ladder climbing with HRP-2 humanoid," *Autonomous Robots*, vol. 40, no. 3, 2016.
- [3] M. F. Fallon, P. Marion, R. Deits, T. Whelan, M. Antone, J. McDonald, and R. Tedrake, "Continuous humanoid locomotion over uneven terrain using stereo fusion," in *IEEE-RAS International Conference on Humanoid Robots*, 2015, pp. 881–888.
- [4] A. Holzbach and G. Cheng, "Enhancing object recognition for humanoid robots through time-awareness," in *IEEE-RAS International Conference on Humanoid Robots*, 2013, pp. 246–251.
- [5] G. Oriolo, A. Paolillo, L. Rosa, and M. Vendittelli, "Humanoid odometric localization integrating kinematic, inertial and visual information," *Autonomous Robots*, vol. 40, no. 5, pp. 867–879, 2016.
- [6] M. Meilland and A. I. Comport, "On unifying key-frame and voxel-based dense visual SLAM at large scales," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2013, pp. 3677–3683.
- [7] R. S. Dahiya and M. Valle, *Robotic tactile sensing—Technologies and system*. Springer, 2013.
- [8] M. Bellaccini, L. Lanari, A. Paolillo, and M. Vendittelli, "Manual guidance of humanoid robots without force sensors: Preliminary experiments with NAO," in *IEEE Int. Conf. on Robotics and Automation*, 2014, pp. 1184–1189.
- [9] A. De Luca, A. Albu-Schäffer, S. Haddadin, and G. Hirzinger, "Collision detection and safe reaction with the DLR-III lightweight robot arm," in *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, 2006, pp. 1623–1630.
- [10] R. Featherstone, "A Beginner's Guide to 6-D Vectors (Part 1)," *IEEE Robotics Automation Magazine*, vol. 17, no. 3, pp. 83–94, 2010.
- [11] —, "A Beginner's Guide to 6-D Vectors (Part 2)," *IEEE Robotics Automation Magazine*, vol. 17, no. 4, pp. 88–99, 2010.
- [12] —, *Rigid Body Dynamics Algorithms*. Secaucus, NJ, USA: Springer-Verlag New York, Inc., 2008.
- [13] F. Flacco, A. De Luca, I. Sardellitti, and N. G. Tsagarakis, "On-line estimation of variable stiffness in flexible robot joints," *The Int. Journal of Robotics Research*, vol. 31, no. 13, pp. 1556–1577, 2012.
- [14] A. De Luca and L. Ferrajoli, "Exploiting robot redundancy in collision detection and reaction," in *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, 2008, pp. 3299–3305.
- [15] A. De Luca and R. Mattone, "Actuator failure detection and isolation using generalized momenta," in *IEEE Int. Conf. on Robotics and Automation*, 2003, pp. 634–639.
- [16] E. Magrini, F. Flacco, and A. De Luca, "Estimation of contact forces using a virtual force sensor," in *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, 2014, pp. 2126–2133.
- [17] A. De Luca and F. Flacco, "Integrated control for pHRI: Collision avoidance, detection, reaction and collaboration," in *Proc. IEEE Int. Conf. on Biomed. Robotics and Biomechanics*, 2012, pp. 288–295.
- [18] E. Magrini, F. Flacco, and A. De Luca, "Control of generalized contact motion and force in physical human-robot interaction," in *IEEE Int. Conf. on Robotics and Automation*, 2015, pp. 2298–2304.
- [19] L. Manuelli and R. Tedrake. (submitted in 2016) Localizing external contact using proprioceptive sensors: The contact particle filter.
- [20] A. Cirillo, F. Ficuciello, C. Natale, S. Pirozzi, and L. Villani, "A conformable force/tactile skin for physical human-robot interaction," *IEEE Robotics and Automation Letters*, vol. 1, no. 1, pp. 41–48, 2016.