



# Model-Based External Force/Moment Estimation For Humanoid Robots With No Torque Measurement

Mehdi Benallegue, Pierre Gergondet, Hervé Audren, Alexis Mifsud, Mitsuharu Morisawa, Florent Lamiraux, Abderrahmane Kheddar, Fumio Kanehiro

## ► To cite this version:

Mehdi Benallegue, Pierre Gergondet, Hervé Audren, Alexis Mifsud, Mitsuharu Morisawa, et al.. Model-Based External Force/Moment Estimation For Humanoid Robots With No Torque Measurement. ICRA 2018 - 35th IEEE International Conference on Robotics and Automation, May 2018, Brisbane, Australia. pp.3122-3129, 10.1109/ICRA.2018.8460809 . hal-01746095

**HAL Id: hal-01746095**

**<https://hal.science/hal-01746095>**

Submitted on 29 Mar 2018

**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.

# Model-Based External Force/Moment Estimation For Humanoid Robots With No Torque Measurement

Mehdi Benallegue<sup>1</sup>, Pierre Gergondet<sup>2</sup>, Hervé Audren<sup>2,3</sup>, Alexis Mifsud<sup>4,5</sup>, Mitsuharu Morisawa<sup>1</sup>, Florent Lamiraux<sup>4,5</sup>, Abderrahmane Kheddar<sup>2,3</sup> and Fumio Kanehiro<sup>1</sup>

**Abstract**—The dynamics of a humanoid robot cannot be correctly described independently from the external forces acting on it. These forces have to be reconstructed to enable the robot to control them or to compensate for them. Force sensors are usually used to measure these forces, but because of their cost, they are often put only on the ankle/feet and possibly the wrists. This paper addresses the issue of the estimation of external forces and moments that apply at any part of a robot without *direct* force measurements and without torque measurements. The sensors used are the regular force sensors and the IMUs of the robot. The method relies on a model-based estimator able to make the fusion between these sensors and the whole body dynamics. The estimator reconstructs a single state vector containing the floating-base kinematics, a filtered measurement of contact force and an additional estimation external force that we evaluate in this paper. Validation is performed on HRP-2 in a multi-contact motion.

## I. INTRODUCTION

Legged robots are underactuated, in the sense that they must use actuated joints to generate contact forces with the environment on which they rely to counteract the effects of gravity, to modify the trajectory of their center of mass and to effectively control their orientation [23]. Therefore, besides the joint torques, contact forces become control variables. On the other hand, external forces can be the cause of disturbances to which the robot has to react in order to keep balance and to perform the required tasks [14]. Finally, and more importantly, the robot is often required to limit its external forces, especially when interacting with humans [8].

Therefore, most robots, especially the biggest ones, are equipped with force sensors to control these forces. These sensors are usually located at the feet level to ensure balance. Others sensors, if any, are commonly located at the end-effector of the arms to measure force during manipulation, interaction and multi-contact locomotion [21]. However, it is clear that the coverage of these sensors is very limited compared to the shape of the robot [9]. Unexpected contacts may happen anywhere on the robot [12], and even multi-contact trajectories may resort to using a different part of the body because of geometry or to minimize energy costs [15].

In the presented references, the contact forces achieved in open-loop, but this solution rapidly reaches its limitations.

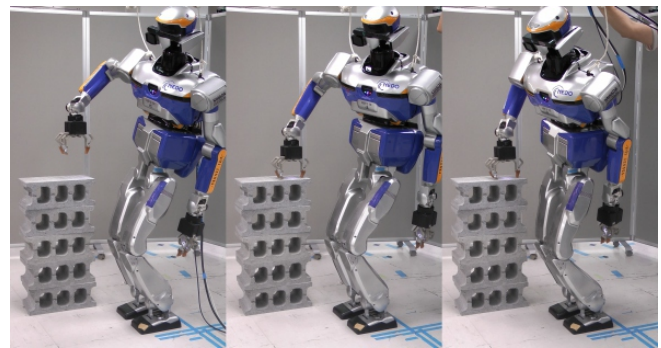
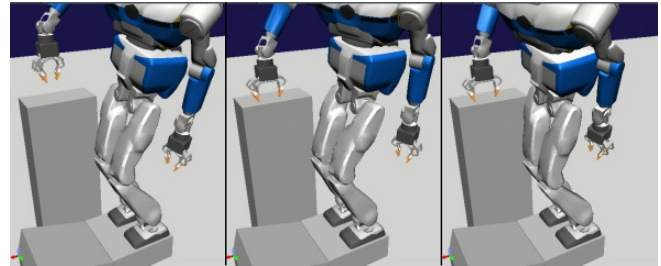


Fig. 1. The experimental setting. At the top a view of the simulation and at the bottom some screen-shots of the real experiment. The leftmost pictures show the robot before making the contact, the center pictures show the robot in contact and the rightmost pictures show the robot after the CoM has been moved forward.

The robot does not always achieve the desired forces when actually performing the actions, even for very similar kinematics [15]. This is critical when interacting with humans but is important even in more kinematics-based actions, for example in the presence of constrained or underactuated objects in the environment [5].

Most approaches deal with this problem, for humanoid robots and manipulator arms, by using the residual method. This is achieved either using torque sensors, a model of the stiffness of the joint or an inversion of the motor model regarding electrical current or tracking error [16], [6], [17], [11], [10], [22]. However, the torque data may not be always available, since torque sensors are still heavy and expensive to be equipped at all the joints of a humanoid robot, and motor model inversion can be insufficient especially for the case of non-backdrivable or high-friction actuators. Furthermore, as stated in [11] and [22], additional data are required when using residues method for floating-base robots, such as the orientation of the floating base, and its velocity.

Authors are with

- 1- Humanoid Research Group, AIST, Tsukuba, Ibaraki, Japan.
  - 2- CNRS-AIST Joint Robotics Laboratory, UMI3218/RL, Tsukuba, Japan.
  - 3- CNRS, Univ de Montpellier, LIRMM, IDH group, Montpellier, France.
  - 4- CNRS, LAAS, 7 avenue du colonel Roche, F-31400 Toulouse, France
  - 5- Univ de Toulouse, LAAS, F-31400 Toulouse, France
- email: mehdi.benallegue@aist.go.jp

Another clever method used the property that by assuming that the center of mass (CoM) of the robot and its inertial measurement unit (IMU) are at the same point, external forces can be reconstructed algebraically [14]. However, besides the approximation regarding the CoM position, this estimation provides only forces and not moments, it is very noisy and requires filtering, and the orientation of the IMU needs also to be reconstructed.

Indeed, the orientation and the velocity of the floating base are also subject to external forces, especially in the case of an unpredicted contact with the environment [18], [3]. In fact, several advanced methods for reconstructing the floating base kinematics actually use the data of external forces to reconstruct reliably the state of the robot [20], [2].

This coupling, instead of being a problem, can be seen actually as an advantage since the sensors tracking the robot dynamics carry then also data on these external forces. This is the base of our method. Instead of using explicit formulations to reconstruct the external force, we introduce this variable in a state estimator able to rebuild, in a single state vector, floating base kinematics, contact forces and external contacts. This vector would be optimized to be consistent with the dynamics, any contact model and sensor measurements.

In the next section, we expose our state estimator, based on extended Kalman filtering. Then we describe the experimental setting we used to test the estimator. Afterwards, the experimental results are shown. And finally, we give a short discussion and conclusion.

## II. DESCRIPTION OF THE ESTIMATOR

The state estimator is very similar to our previous version presented in [2]. In this former work, we have shown that the fusion between the IMU, contact forces, Newton-Euler dynamics and a contact model into a single state, provides at the same time an estimation which is not only consistent with the dynamics but also near-optimally consistent with all the measurement. This is due to the large part of redundancy which lies in these measurements. Indeed, we have shown that the IMU alone is able to reconstruct the center of pressure of contact forces relatively accurately when coupled with our model [18]. The idea here is to take profit from this redundancy in order to rebuild an additional variable, the overall external force and moment which are not directly sensed. This is different from the reconstruction of the center of pressure presented above because it was using a viscoelastic model of the contact forces. This time we do not introduce any model of the external force.

The estimator is based on Extended Kalman filtering (EKF), which requires a state definition, a state dynamics and a measurement dynamics.

### A. Model of the kinematics

A humanoid robot is aware of its joint configuration thanks to its joint encoder, but there is usually no measurement on the position and the orientation of the floating base. Instead,

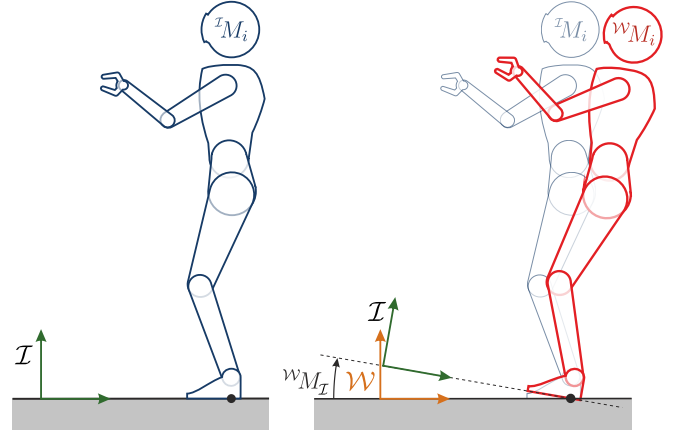


Fig. 2. Definition of the frames.  ${}^I M_i$  is the position of the  $i$ -th body in the ideal frame  $\mathcal{I}$ .  ${}^W M_i$  is the position of the same body in the world reference frame  $\mathcal{W}$ .  ${}^W M_I$  is the homogeneous matrix between these two frames representing the deviation.

there is often a modeled position of this configuration described in an inertial frame. This is usually obtained through a model of the environment, of the contact positions and orientations and using joint encoders similarly to a fixed-base kinematic chain. This configuration can be described by the transformation matrix between the base and an ideal frame  $\mathcal{I}$ . Let's denote this matrix  ${}^I M_b$ . In the ideal case, the ideal frame and the world frame  $\mathcal{W}$  are identical. However, even when contact positions and orientations are perfectly known, no robot is perfectly stiff and the compliance of the robot leads the real configuration to differ. A robot controller is *a priori* not aware of this deformation, this is why we were referring to this frame as the *control frame* in our former publications [1], [18], [2]. The real configuration can be described as a transformation matrix denoted  ${}^W M_b$ .

The transformation between the ideal and the world frame can be called *deviation* and can be represented by the matrix  ${}^W M_I = {}^W M_b {}^I M_b^{-1}$ . In this work, we make the following approximation: we consider that the deviation is a rigid transformation applied on the whole body, in other words the deviation is the same for all the limbs of the robot. This can be written as

$$\forall i \in \{0, \dots, n\}, {}^W M_i = {}^W M_I {}^I M_i. \quad (1)$$

where  $n$  is the number of limbs,  ${}^W M_i$  is the configuration of the  $i$ -th body in the world frame, and  ${}^I M_i$  is its position in the ideal frame, obtained with forward kinematics and contact data (see Figure 1).

In the following developments, we will omit the superscript  $\mathcal{W}$  to simplify notations, therefore, the deviation matrix is  $M_I$ . Let's decompose this matrix into rotation matrix  $R_I$  and translation  $\mathbf{t}_I$ . It is clear that if we manage to rebuild this transformation, we will be able to locate every limb in the space. The advantage of this representation is (i) its independence from the number of contacts, and (ii) its easy interpretation since the ideal value is the identity. In the following, we describe how we predict the dynamics of

this state, with regard to the external forces and the internal kinetic variables.

### B. Contact wrenches model

The assumption of the transformation being rigid neglects the deformation of each limb and leads to a possible displacement of contact points in the reconstructed kinematics. However, the known contact positions provide a good anchor with the environment that should be exploited to take profit from the couplings happening between rotations and translations [1]. In order to not lose this advantage, we model contact forces penalizing the deviations of contact points. This can be a very good model when the feet or the ground are flexible [13], but can also be considered as an approximation of the contact wrench generated by the reaction force of the compliant links.

We model this wrench by viscoelastic forces and moments in response the 6D deviation of each contact. The forces are responses to translations and the moments to rotations. The forces are applied at the contact points, so they contribute to the resulting total moment applied on the robot.

Accordingly, the forces are expressed as

$$\mathbf{f}_{c,i} = -K_{f,p}\mathbf{t}_{c,i} - K_{f,d}\dot{\mathbf{t}}_{c,i} \quad (2)$$

where

$$\begin{aligned} \mathbf{t}_{c,i} &= \mathbf{p}_{c,i} - {}^{\mathcal{I}}\mathbf{p}_{c,i} \\ &= R_{\mathcal{I}}\mathcal{I}\mathbf{p}_{c,i} + \mathbf{t}_{\mathcal{I}} - {}^{\mathcal{I}}\mathbf{p}_{c,i}, \end{aligned} \quad (3)$$

$$\dot{\mathbf{t}}_{c,i} = [\boldsymbol{\omega}_{\mathcal{I}}]_{\times} R_{\mathcal{I}}\mathcal{I}\mathbf{p}_{c,i} + \dot{\mathbf{t}}_{\mathcal{I}}. \quad (4)$$

with  $\boldsymbol{\omega}_{\mathcal{I}}$  is the angular velocity vector of the deviation, i.e.  $\dot{R}_{\mathcal{I}} = [\boldsymbol{\omega}_{\mathcal{I}}]_{\times} R_{\mathcal{I}}$  where  $[\cdot]_{\times}$  is the skew symmetric operator

$$\text{such that } \begin{bmatrix} x \\ y \\ z \end{bmatrix}_{\times} = \begin{bmatrix} 0 & -z & y \\ z & 0 & -x \\ -y & x & 0 \end{bmatrix}.$$

and  $K_{f,p}$  and  $K_{f,d}$  are  $3 \times 3$  symmetric positive definite matrices representing elasticity and viscosity coefficients [19]. We sum the forces over the  $n_c$  contacts to obtain the resultant contact force  $\mathbf{f}_c = \sum_{i=1}^{n_c} \mathbf{f}_{c,i}$ .

The contact moment is the combination of the reaction torque of the torsional damped spring and the moment of the forces  $\mathbf{f}_{c,i}$ . This gives the following expression

$$\mathbf{m}_{c,i} = -K_{m,p}\Omega_{\mathcal{I}} - K_{m,d}\boldsymbol{\omega}_{\mathcal{I}} + [\mathbf{p}_{c,i}]_{\times} \mathbf{f}_{c,i}. \quad (5)$$

Where  $\Omega_{\mathcal{I}}$ , also called rotation vector, is the minimum norm vector in  $\mathbb{R}^3$  so that  $R_{\mathcal{I}} = \exp([\boldsymbol{\omega}_{\mathcal{I}}]_{\times})$  with  $\exp$  is the matrix exponential and  $K_{m,p}$  and  $K_{m,d}$  are  $3 \times 3$  symmetric positive definite matrices.

We finally sum the moments  $\mathbf{m}_{c,i}$  over the contacts to obtain the resultant moment:  $\mathbf{m}_c = \sum_{i=1}^{n_c} \mathbf{m}_{c,i}$ .

### C. The feed-forward state dynamics

Let us first define our state vector

$$x = (\mathbf{t}_{\mathcal{I}}^t \ \Omega_{\mathcal{I}}^t \ \dot{\mathbf{t}}_{\mathcal{I}}^t \ \boldsymbol{\omega}_{\mathcal{I}}^t \ \mathbf{f}_{c,1}^t \ \mathbf{m}_{c,1}^t \ \dots \ \mathbf{f}_{c,n_c}^t \ \mathbf{m}_{c,n_c}^t \ \mathbf{f}_e^t \ \mathbf{m}_e^t)^t \quad (6)$$

where  $t$  upper-script stands for *transpose*, and  $\mathbf{f}_e$  and  $\mathbf{m}_e$  are respectively the non-modeled external force and moment at the origin that we estimate in this paper. This state definition is different from our previous estimators [18], [2], because it contains the contact forces and moments  $\mathbf{f}_{c,i}$  and  $\mathbf{m}_{c,i}$  instead of the second order derivatives of the kinematics. This choice may lead to a heavier state vector, but it allows to finely tune the modeling error made by the viscoelastic forces and moments. The fact that the state changes size when a contact is broken is not a problem in general for Kalman Filtering, but for our implementation, we just chose to put zero forces and moments for broken contacts.

The dynamics of the forces is described in equations (2) and (5). For the kinematics part; the prediction comes from the kinetics. We have through Newton and Euler equations:

$$\mathbf{f}_c + \mathbf{f}_e - g m \mathbf{u}_z = \frac{d}{dt} \left( \sum_{i=1}^n m_i \dot{\mathbf{c}}_i \right), \quad (7)$$

$$\mathbf{m}_c + \mathbf{m}_e - [R_{\mathcal{I}}\mathbf{c}]_{\times} g m \mathbf{u}_z = \dot{\sigma} \quad (8)$$

$$= \frac{d}{dt} \sum_{i=1}^n \left( R_i I_i R_i^t \boldsymbol{\omega}_i + m_i [\mathbf{c}_i]_{\times} \dot{\mathbf{c}}_i \right) \quad (9)$$

where  $\mathbf{u}_z = [0 \ 0 \ 1]^t$  is the vertical unit vector along the  $z$ -axis,  $g$  the standard gravity,  $\mathbf{c}_i$  is the position of the CoM of the  $i$ -th body,  $I_i$ ,  $R_i$  and  $\boldsymbol{\omega}_i$  are the inertia matrix, the orientation of the  $i$ -th body and its angular velocity respectively.

In the Ideal frame  $\mathcal{I}$  many values are available just through joint encoders and its time-derivatives. In order to predict the dynamics of the state some of these data need to be computed. These required data are: the mass  $m$ , the position, velocity and acceleration of the CoM,  $\mathbf{c}_{\mathcal{I}}$ ,  $\dot{\mathbf{c}}_{\mathcal{I}}$ ,  $\ddot{\mathbf{c}}_{\mathcal{I}}$ , respectively, the  $3 \times 3$  tensor of inertia of the robot  $I_{\mathcal{I}}$  and its time-derivative  $\dot{I}_{\mathcal{I}}$ , and the angular momentum  $\sigma_{\mathcal{I}}$  and its derivative  $\dot{\sigma}_{\mathcal{I}}$ . These variables allow decoupling the kinematics of Newton-Euler equations in the world frame between those in the ideal frame and the proper motion of the ideal in the world frame, and lead to the prediction state-dynamics given by the following equations:

$$\begin{aligned} \dot{\boldsymbol{\omega}}_{\mathcal{I}} &= R_{\mathcal{I}}(I_{\mathcal{I}} + m[\mathbf{c}_{\mathcal{I}}]_{\times})^{-1} R_{\mathcal{I}}^t \\ &\quad (\mathbf{m}_c + \mathbf{m}_e - [R_{\mathcal{I}}\mathbf{c}_{\mathcal{I}} + \mathbf{t}_{\mathcal{I}}]_{\times} (\mathbf{f}_c + \mathbf{f}_e) \\ &\quad - (([\boldsymbol{\omega}_{\mathcal{I}}]_{\times} R_{\mathcal{I}} I_{\mathcal{I}} R_{\mathcal{I}}^t + R_{\mathcal{I}} \dot{I}_{\mathcal{I}} R_{\mathcal{I}}^t) \boldsymbol{\omega}_{\mathcal{I}} + R_{\mathcal{I}} \dot{\sigma}_{\mathcal{I}} + [\boldsymbol{\omega}_{\mathcal{I}}]_{\times} R_{\mathcal{I}} \sigma_{\mathcal{I}}) \\ &\quad + m[R_{\mathcal{I}}\mathbf{c}_{\mathcal{I}}] (R_{\mathcal{I}} \ddot{\mathbf{c}}_{\mathcal{I}} + 2[\boldsymbol{\omega}_{\mathcal{I}}]_{\times} R_{\mathcal{I}} \dot{\mathbf{c}}_{\mathcal{I}} + [\boldsymbol{\omega}_{\mathcal{I}}]_{\times}^2 R_{\mathcal{I}} \mathbf{c}_{\mathcal{I}})) \end{aligned} \quad (10)$$

and:

$$\begin{aligned} \ddot{\mathbf{t}}_{\mathcal{I}} &= \frac{1}{m} (\mathbf{f}_c + \mathbf{f}_e) - (R_{\mathcal{I}} \ddot{\mathbf{c}}_{\mathcal{I}} + 2[\boldsymbol{\omega}_{\mathcal{I}}]_{\times} R_{\mathcal{I}} \dot{\mathbf{c}}_{\mathcal{I}} + [\boldsymbol{\omega}_{\mathcal{I}}]_{\times}^2 R_{\mathcal{I}} \mathbf{c}_{\mathcal{I}} \\ &\quad + g \mathbf{u}_z) + [R_{\mathcal{I}} + \dot{\mathbf{c}}_{\mathcal{I}}]_{\times} \boldsymbol{\omega}_{\mathcal{I}} \end{aligned} \quad (11)$$

Details of these developments are described in [18]. These equations enable to predict the future dynamics of the deviation.

Finally, the dynamical model of the external non-modeled forces  $\mathbf{f}_e$  and  $\mathbf{m}_e$  is that they are constant in time, which allows filtering their noise. This gives the model

$$\dot{\mathbf{f}}_e = 0 \quad (12)$$

$$\dot{\mathbf{m}}_e = 0 \quad (13)$$

#### D. Sensors

This paper uses both force sensors and IMU. The model of the measurements of the force sensors is straightforward, since contact forces are already in the state vector, and can be described as follows:

$$\mathbf{y}_{f_i} = {}^{\mathcal{I}}R_{c_i}^t R_{\mathcal{I}}^t \mathbf{f}_{c,i}^t \quad (14)$$

$$\mathbf{y}_{m_i} = {}^{\mathcal{I}}R_{c_i}^t R_{\mathcal{I}}^t \mathbf{m}_{c,i}^t, \quad (15)$$

where  $y_{f_i}$  is the force measurement and  $y_{m_i}$  is the moment measurement of the  $i$ -th contact and  ${}^{\mathcal{I}}R_{c_i}^t$  is the orientation of the force sensor in the ideal frame.

The IMU contains an accelerometer and a gyrometer. The accelerometer measures the gravity and linear accelerations of the sensor and the gyrometer measures the angular velocity of the sensor. Both sensors produce their measurements in their local frame. This gives the following model

$$\mathbf{y}_g = {}^{\mathcal{I}}R_s^t \boldsymbol{\omega}_s + {}^{\mathcal{I}}R_s^t R_{\mathcal{I}}^t \boldsymbol{\omega}_{\mathcal{I}} \quad (16)$$

where  ${}^{\mathcal{I}}R_s^t$  and  ${}^{\mathcal{I}}\boldsymbol{\omega}_s$  are the orientation and the angular velocity of the IMU in the ideal frame.

$$\begin{aligned} \mathbf{y}_a = & {}^{\mathcal{I}}R_s^t R_{\mathcal{I}}^t (([\dot{\boldsymbol{\omega}}_{\mathcal{I}}]_{\times} + [\boldsymbol{\omega}_{\mathcal{I}}]_{\times}^2) R_{\mathcal{I}}^t \mathbf{p}_s + 2[\boldsymbol{\omega}_{\mathcal{I}}]_{\times} R_{\mathcal{I}}^t \dot{\mathbf{p}}_s) \\ & + {}^{\mathcal{I}}R_s^t ({}^{\mathcal{I}}\ddot{\mathbf{p}}_s + R_{\mathcal{I}}^t \ddot{\mathbf{t}}_{\mathcal{I}}) + {}^{\mathcal{I}}R_s^t R_{\mathcal{I}}^t g \mathbf{u}_z \end{aligned} \quad (17)$$

where  ${}^{\mathcal{I}}\mathbf{p}_s$  is the position of the sensor in the ideal frame.

#### E. Extended Kalman Filtering

EKF techniques proceeds in two steps, the prediction which allows using a model as a feed forward predictor for the state, and the update comparing predicted measurement and actual ones to correct for modeling errors and external disturbances. The outcome is a state vector  $\mathbf{x}$  reconstructed in a single optimization process, taking into account all the interactions between the components and all the redundancy and coupling that lie in the sensor data.

Although the dynamics is mostly described in continuous time, our implementation of the Kalman filter is discrete time. Using (2), (5), (12) and (13), the state prediction takes the following form:

$$\mathbf{x}_{k+1} = \begin{pmatrix} \mathbf{t}_{\mathcal{I},k} + T\dot{\mathbf{t}}_{\mathcal{I},k} \\ V_{\log} \left( \exp [T\boldsymbol{\omega}_{\mathcal{I},k}]_{\times} \exp [\boldsymbol{\Omega}_{\mathcal{I},k}]_{\times} \right) \dot{\mathbf{t}}_{\mathcal{I},k} + T\ddot{\mathbf{t}}_{\mathcal{I},k} \\ \boldsymbol{\omega}_{\mathcal{I},k} + T\dot{\boldsymbol{\omega}}_{\mathcal{I},k} \\ -K_{f,p}\mathbf{t}_{c,1,k+1} - K_{f,d}\dot{\mathbf{t}}_{c,1,k+1} \\ -K_{m,p}\boldsymbol{\Omega}_{\mathcal{I},k+1} - K_{m,d}\dot{\boldsymbol{\omega}}_{\mathcal{I},k+1} + [\mathbf{p}_{c,1,k+1}]_{\times} \mathbf{f}_{c,1,k+1} \\ \vdots \\ -K_{f,p}\mathbf{t}_{c,n_c,k+1} - K_{f,d}\dot{\mathbf{t}}_{c,n_c,k+1} \\ -K_{m,p}\boldsymbol{\Omega}_{\mathcal{I},k+1} - K_{m,d}\dot{\boldsymbol{\omega}}_{\mathcal{I},k+1} + [\mathbf{p}_{c,n_c,k+1}]_{\times} \mathbf{f}_{c,n_c,k+1} \\ \mathbf{f}_{e,k} \\ \mathbf{m}_{e,k} \end{pmatrix} \quad (18)$$

where  $k$  is used for time indexation of variables,  $T$  is the sampling time,  $V$  is defined such that for any rotation matrix  $R$ , we have  $V_{\log}(R) = V(\log(R))$  such that  $\log$  is the matrix logarithm and  $V$  is the inverse of the skew-symmetric operator, or in other words  $R = \exp([V_{\log}(R)]_{\times})$ , and  $\ddot{\mathbf{t}}_{\mathcal{I},k}$  and  $\dot{\boldsymbol{\omega}}_{\mathcal{I},k}$  are computed using (11) and (10) respectively using values in  $\mathbf{x}_k$ . The set of equations (14), (15), (16) and (17) represent the sensing model for predicting the measurements.

Of course, the limitation of EKF is that it remains only optimal in the linear approximation of these nonlinear interactions and with Gaussian white noise assumptions on the disturbances. Nevertheless, hereinafter we describe the obtained result on a multi-contact task in simulation and on a real humanoid robot.

### III. EXPERIMENTAL SETTING

#### A. Motion generation

Multi-contact locomotion is the main objective of our state estimator, intended to allow for closed-loop control of contact force for any application point on the body. For this, we use HRP-2Kai Robot in a position-controlled motion. The control is mainly in open-loop regarding the actual contact forces, except for contact detection.

The motion was generated using a task-based inverse dynamics controller [4], [21]. Such a controller solves, in real-time, a constrained Quadratic Program in generalized coordinates accelerations and contact forces. Using the robot model, it aims at minimizing a set of weighted objectives under physical constraints.

At all times we satisfy the following constraints:

- Dynamic consistency
- Joint limit constraints (position, velocity, torque)
- Non-sliding contacts

In the objectives, we always use a low-weight posture objective as a means of regularizing the joint accelerations. Then a small Finite-State-Machine allows us to first bring the hand above the bricks, then reach the contact and finally oscillate along the sagittal plane.

In order to bring the hand above the bricks, we used a 6D position task. Then, we used an admittance task to bring the gripper in contact with the top surface of the bricks. This was achieved by targeting a pure normal force of 15N at

the hand level. Once contact was established, we enforced the non-sliding constraint at the hand. Then, we added a CoM task to our program. The target of this CoM task was cyclically moved by 10 cm to repeatedly move the robot forward and backward (see Figure (1)).

Our estimator only uses the force sensor at the feet level and the IMU, but since HRP-2Kai is equipped with a force sensor at the hand we have a direct measurement of the actual force. We do not provide the estimator with any additional knowledge about the external forces. For instance, neither the point of application nor the nature of the force are known.

### B. Comparison with sensor-based approach

We compare our results with the estimation obtained by the method described in [14], because not only it uses the same data as our method but it provides a good view on the advantage of model-based approach over sensor-based ones. This method assumes that the CoM and the IMU are located at the same point, then the force can be approximated with the following algebraic expression

$$\mathbf{f}_e \approx R_s y_a - \sum_{i=1}^{n_c} R_{c_i} y_{f_i}$$

where  $R_s$  and  $R_{c_i}$  are the orientations of the IMU and the  $i$ -th force sensor in the world frame respectively. Therefore, this method requires the estimation of the attitude of the floating base, so we provide it with the estimation coming from our state observer.

The data provided by this algebraic estimator are very noisy, so we filter the signals with a first-order filter with the same cutoff frequency as for the reference work.

Since this estimation is deeply rooted in the sensor-space, we call it sensor-based estimation. Our estimator will be referred to as a model-based one.

Finally, note that only the force is provided using this approach, so we do not have a reference for this estimation regarding moments.

## IV. RESULTS

The numerical evaluation of the estimators is available at Table I. The first studied parameter is the bias, which is the estimated value of the force when we know that no actual force is applied in the beginning. The bias indicates how the measurement is reliable when we do not know whether a force is applied at the beginning of the experiment or not, it may also be variable depending on the configuration of the robot, so a lower bias is important for the detection of contacts. This bias can come from modelling error of contact positions or orientations, and biases in the sensors such as force sensors and gyrometers. Then this bias is removed from all the signals and the estimations are compared with the measurements, summarized in the Root Mean Square Error RMSE, computed during the whole experiment, and showed in the table for each dimension and for overall estimation (norm). The plots show the comparison between

the estimations and the measurements after the initial bias being removed.

Hereinafter the details of the results in simulation and on the real robot.

### A. Simulation

The estimator has been tested in simulation and on the real robot. The simulation was achieved on Choreonoid, a dynamical simulator allowing to have perfect measurements but not a perfect model. Table I shows that the model based estimator has very low bias and a small RMSE. The value of the bias along  $x$  axis is larger for an unknown reason, and actually removing the initial bias increased the error since this bias disappears during the contact phase. The sensor-based estimation has much larger biases because there is no model to correct for them. When the bias is removed the estimation is precise but less than the model-based estimation.

Figure 3 shows the evolution of the estimations together with the measurements. It is clear that the estimation is of good quality, except for the force on  $y$  axis for sensor-based approach, where oscillations were recorded. The model-based approach did not suffer from this effect.

### B. Real robot

The experiment was run on the real robot as well, and there were, of course, more sources of errors and noises. Nevertheless I shows that the biases did not increase significantly, except for the force along  $y$  axis. This is due to the internal forces of the robot between its feet which gave a resulting force of about -10 N for an unknown reason, which seems to be intrinsic to the force sensors and could not be removed by displacing the robot. Similarly, the internal moments create also a moment bias around  $x$  axis.

We can see also on Figure 4 the time-evolution of the bias-free estimations compared to the ground truth measurements. All the dimensions seem to be nicely estimated, except for the force on  $y$  axis, and moment on  $x$  axis, mainly for the same reason of internal forces being incorrectly measured by force sensors at the foot. It is interesting to see how the force estimation along  $x$  axis is nicely tracked by the model-based estimator compared to the sensor-based one.

### C. Beyond force/moment estimation

1) *Contact axis*: Following [17], we may assume that the external force has no moment at its point of application. This may happen, for example, when the force comes from a point contact. The reconstruction of forces and moments allows building a straight line in the Cartesian space along which the moment and the force are aligned. This axis has the minimum moment and is sometimes called the wrench central axis [7], which approximates the direction of the contact and the application point. We can compute the distance between the central axis and the real point of application  $\mathbf{p}_e$ . This distance gives an evaluation of the error in the estimation of the contact axis. It is obtained by the following expression.



		Sensor-based Force (N)				Model-based Force (N)				Model-based Moment (N.m)			
Simulation	Bias	<i>x</i>	<i>y</i>	<i>z</i>	norm	<i>x</i>	<i>y</i>	<i>z</i>	norm	<i>x</i>	<i>y</i>	<i>z</i>	norm
	RMSE	6.89	12.73	2.90	14.76	7.35	1.95	1.93	7.85	3.54	4.43	2.27	6.11
Real robot	Bias	-35.56	-25.50	4.80	44.03	-4.04	10.98	-1.90	11.85	-14.93	-8.93	0.22	17.40
	RMSE	12.35	25.76	3.82	28.83	3.01	15.02	3.87	15.80	7.07	2.17	1.49	7.54

TABLE I  
SUMMARY OF THE RESULTS - INCLUDING SENSOR-BASED ESTIMATION (GREYED) [14].

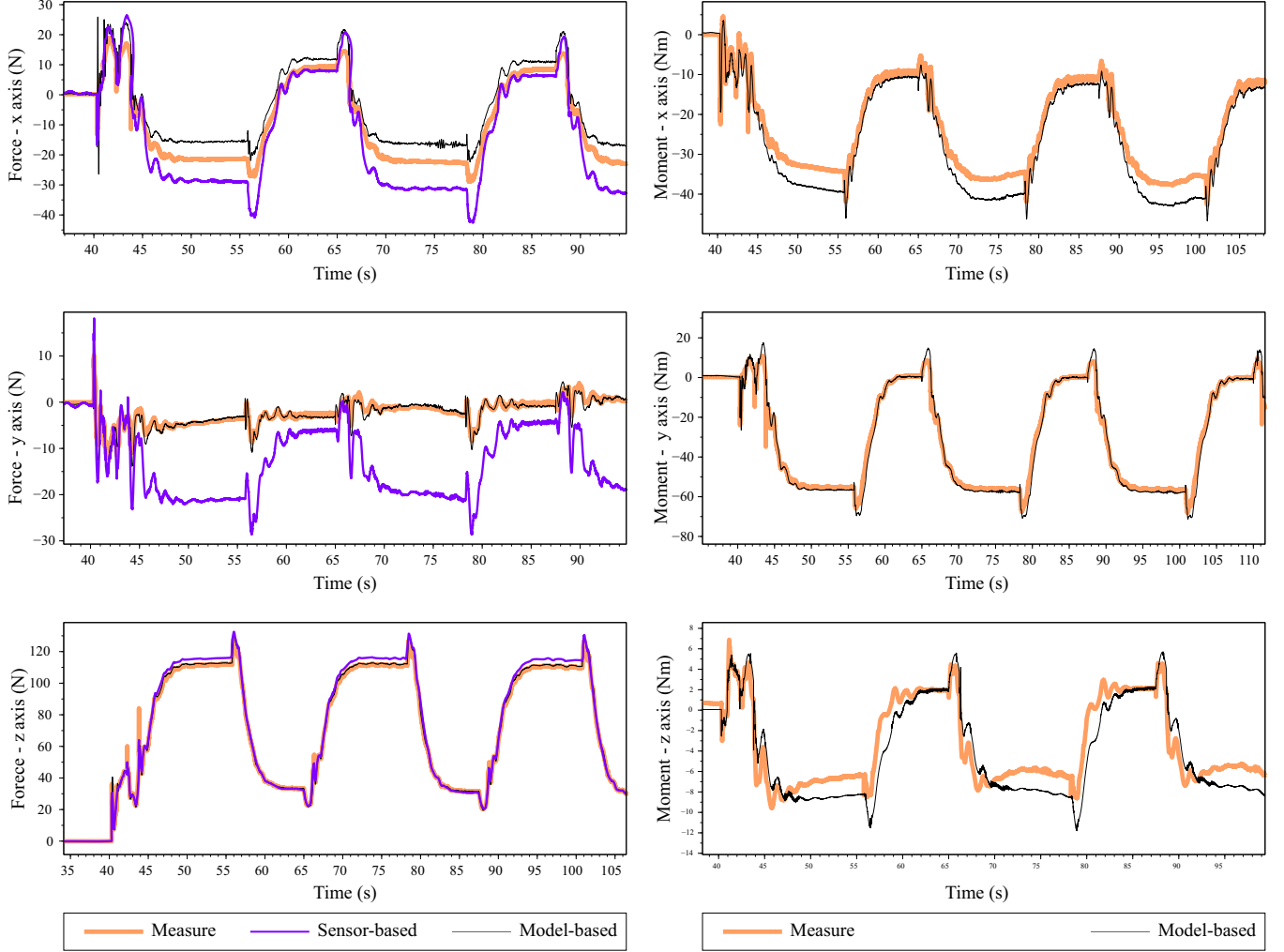


Fig. 3. Plots showing estimations of forces and moments at the origin during simulation. The thin dark line is the result of our model-based estimation, the thick light line is the result of the measurement of the force sensor at the hand and the medium line is the sensor-based estimator [14]. All the initial biases have been removed.

$$d_{\mathbf{p}_e} = \frac{\| [\mathbf{f}_e]_{\times} (\mathbf{m}_e + [\mathbf{f}_e]_{\times} \mathbf{p}_e) \|}{\| \mathbf{f}_e \|^2} \quad (19)$$

In Figure 5 we show this error for our estimator and the value for the hand force sensor. The precision increases when the normal force increases because the signal is more clear and less sensitive to noise. The best precision reached with our estimator is about 7 cm, which is the approximate size of the hand. The hand sensor reaches obviously much higher precision of less than 3 mm error.

2) *No force sensor*: We have tested our estimator when deactivating all force sensors. In that case, the estimator heavily relies on the models, but we can see interesting results in 6. The estimator was able to reconstruct the moments around *x* and very precisely around *y*. This is consistent with the former study where we show that the center of pressure can be rebuilt with only the IMU and the model [18]. The other variables were badly rebuilt and are omitted here. Instead, we tried to see if the additional knowledge of contact position  $\mathbf{p}_e$  could provide data on the forces.

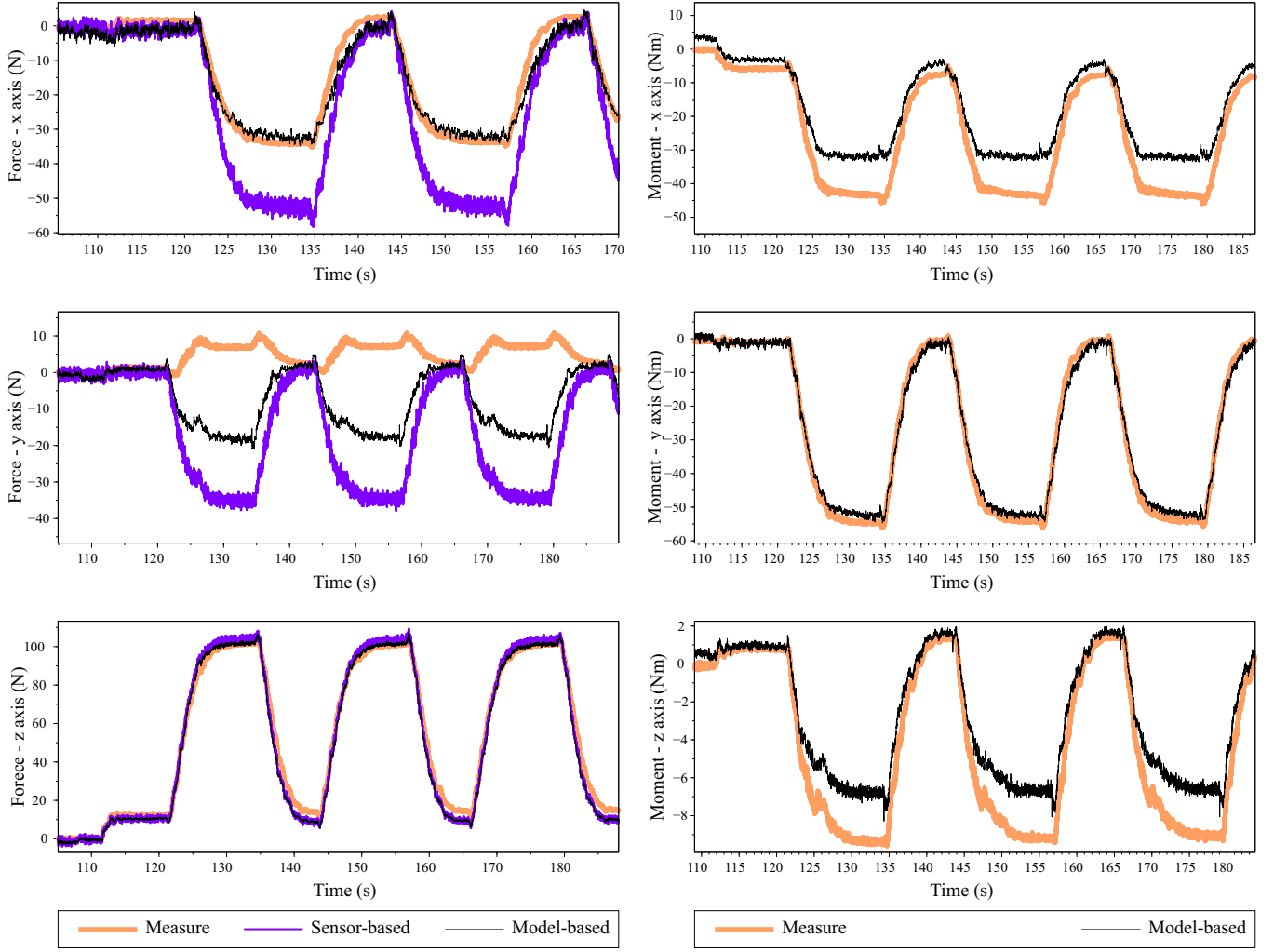


Fig. 4. Plots showing estimations of forces and moments at the origin during the real experiment. The thin dark line is the result of our model-based estimation, the thick light line is the result of the measurement of the force sensor at the hand and the medium line is the sensor-based estimator [14]. All the initial biases have been removed.

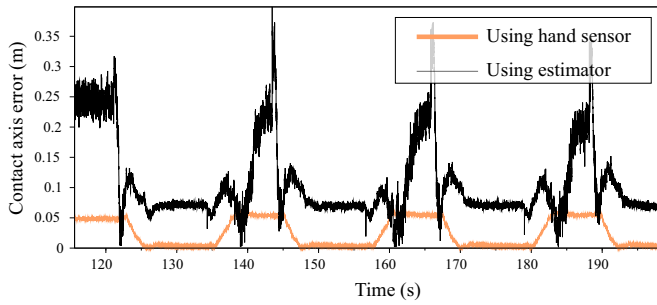


Fig. 5. Plot showing the distance between the real contact point and the central axis obtained with the forces and moments. The thin dark line is the result of our estimation and the thick light line is the result of the measurement of the force sensor at the hand.

Using a pseudoinverse, the result was a bad estimation, except for  $x$  direction where the force was estimated with a scale factor. We believe that with a proper model we can improve this estimation.

This last result may seem of poor quality compared to the

previous ones, but it underlines nicely the contribution of the model in the estimation, improving not only the accuracy by reducing biases and errors, but also giving smoother responses in the estimations without introducing phase shifts.

## V. CONCLUSION

In this paper, we have presented a state estimator capable to reconstruct the external force without direct measurement and without resorting to a torque-based estimator. The estimator showed good performances with low biases and tracking errors. The estimator provides at the same time an estimation of the floating base kinematics, and a model-based filtered estimation of the sensed contact forces. The reconstructed state is then consistent with all the measurements and the dynamics of the system. The showed performances seem good enough to be applied for closed-loop control, which will be one of the next objectives of our research activities.

The introduction of torque data can also drastically improve the quality of the estimation, a fusion between all the



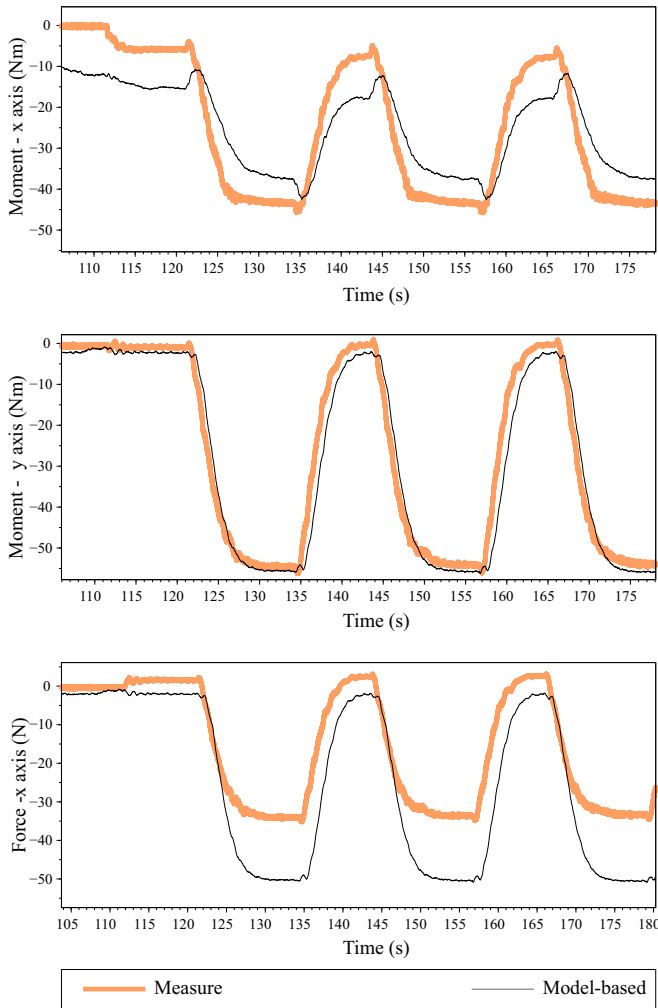


Fig. 6. At the top and the middle, a plot showing the estimations of moments around  $x$  and  $y$  axis during real experiment without using any force sensor. At the bottom, a plot showing the reconstructed force along  $x$  axis using the moments and the contact point information. The thin dark line is the estimation and the thick light line is the measurement obtained with the force sensor at the hand.

available data is indeed the next topic of state estimation for humanoid robots.

## REFERENCES

- [1] M. Benallegue and F. Lamiraux. Humanoid flexibility deformation can be efficiently estimated using only inertial measurement units and contact information. In *14th {IEEE-RAS} International Conference on Humanoid Robots, Humanoids 2014, Madrid, Spain, November 18-20, 2014*, pages 246–251, 2014.
- [2] M. Benallegue, A. Mifsud, and F. Lamiraux. Fusion of force-torque sensors, inertial measurements units and proprioception for a humanoid kinematics-dynamics observation. In *IEEE-RAS International Conference on Humanoid Robots (Humanoids)*, 2015.
- [3] M. Bloesch, M. Hutter, M. Hoepflinger, S. Leutenegger, C. Gehring, C. D. Remy, and R. Siegwart. State Estimation for Legged Robots - Consistent Fusion of Leg Kinematics and {IMU}. In *Proceedings of Robotics: Science and Systems*, Sydney, Australia, jul 2012.
- [4] K. Bouyarmane and A. Kheddar. Using a Multi-Objective Controller to Synthesize Simulated Humanoid Robot Motion with Changing Contact Configurations. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, San Fransico, CA, 2011.
- [5] K. Bouyarmane, J. Vaillant, K. Chappellet, and A. Kheddar. Multi-robot and task-space force control with quadratic programming. *Submitted to IEEE TRANSACTIONS ON ROBOTICS*, 2017.
- [6] G. Buondonno and A. De Luca. Combining real and virtual sensors for measuring interaction forces and moments acting on a robot. In *IEEE International Conference on Intelligent Robots and Systems*, volume 2016-Novem, pages 794–800. IEEE, oct 2016.
- [7] J. Carpentier, M. Benallegue, N. Mansard, and J. P. Laumond. Center-of-mass estimation for a polyarticulated system in contact: A spectral approach. *IEEE Transactions on Robotics*, 32(4):810–822, Aug 2016.
- [8] A. De Luca and L. Ferrajoli. Exploiting robot redundancy in collision detection and reaction. In *2008 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 3299–3305, 2008.
- [9] Salman Faraji and Auke Jan Ijspeert. Designing a virtual whole body tactile sensor suit for a simulated humanoid robot using inverse dynamics. In *Intelligent Robots and Systems (IROS), 2016 IEEE/RSJ International Conference on*, pages 5564–5571. Ieee, 2016.
- [10] F. Flacco and A. Kheddar. Contact Detection and Physical Interaction for Low Cost Personal Robots. In *RO-MAN: Robot and Human Interactive Communication*, Lisbon, Portugal, August 2017. Conference theme: Human-Robot Collaboration and Human Assistance for an Improved Quality of Life.
- [11] F. Flacco, A. Paolillo, and A. Kheddar. Residual-based contacts estimation for humanoid robots. In *2016 IEEE-RAS 16th International Conference on Humanoid Robots (Humanoids)*, pages 409–415, 2016.
- [12] Sami Haddadin and Alessandro De Lucaband Alin Albu-Schäffer. Robot collisions: A survey on detection, isolation, and identification. *IEEE Transactions on Robotics*, 2017.
- [13] N. Kanehira, T. Kawasaki, S. Ohta, T. Ismumi, T. Kawada, F. Kanehiro, S. Kajita, and K. Kaneko. Design and experiments of advanced leg module (HRP-2L) for humanoid robot (HRP-2) development. In *Intelligent Robots and Systems*, volume 3, pages 2455–2460, 2002.
- [14] K. Kaneko, F. Kanehiro, M. Morisawa, E. Yoshida, and J.-P. Laumond. Disturbance observer that estimates external force acting on humanoid robots. *2012 12th IEEE International Workshop on Advanced Motion Control (AMC)*, 1(c):1–6, 2012.
- [15] S. Lengagne, J. Vaillant, E. Yoshida, and A. Kheddar. Generation of whole-body optimal dynamic multi-contact motions. *The International Journal of Robotics Research*, 32(9-10):1104–1119, 2013.
- [16] E. Magrini, F. Flacco, and A. De Luca. Estimation of contact forces using a virtual force sensor. *2014 IEEE/RSJ International Conference on Intelligent Robots and Systems, (Iros):2126–2133*, 2014.
- [17] L. Manuelli and R. Tedrake. Localizing external contact using proprioceptive sensors: The Contact Particle Filter. In *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 5062–5069, 2016.
- [18] A. Mifsud, M. Benallegue, and F. Lamiraux. Estimation of Contact Forces and Floating Base Kinematics of a Humanoid Robot Using Only Inertial Measurement Units. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2015)*, page 6p., Hamburg, Germany, sep 2015.
- [19] Y. Mikami, T. Moulard, E. Yoshida, and G. Venture. Identification of hrp-2 foot’s dynamics. In *2017 IEEE/RSJ International Conference on, pages 927–932*. IEEE, 2014.
- [20] N. Rotella, A. Herzog, S. Schaal, and L. Righetti. Humanoid momentum estimation using sensed contact wrenches. In *Proceedings of the 2015 IEEE-RAS International Conference on Humanoid Robots*, 2015.
- [21] 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 an HRP-2 humanoid. *Autonomous Robots*, 40(3):561–580, 2016.
- [22] J. Vorndamme, M. Schappler, and S. Haddadin. Collision detection, isolation and identification for humanoids. In *2017 IEEE International Conference on Robotics and Automation (ICRA)*, pages 4754–4761, May 2017.
- [23] P.-B. Wieber, R. Tedrake, and S. Kuindersma. *Modeling and Control of Legged Robots*, pages 1203–1234. Springer International Publishing, Cham, 2016.