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

Mehdi Benallegue\textsuperscript{1}, Pierre Gergondet\textsuperscript{2}, Hervé Audren\textsuperscript{2,3}, Alexis Mifsud\textsuperscript{4,5}, Mitsuharu Morisawa\textsuperscript{1}, Florent Lamiraux\textsuperscript{4,5}, Abderrahmane Kheddar\textsuperscript{2,3} and Fumio Kanehiro\textsuperscript{1}

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.

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.
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, 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 \( \mathcal{I} M_b \). 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 \textit{a priori} not aware of this deformation, this is why we were referring to this frame as the \textit{control frame} in our former publications [1], [18], [2]. The real configuration can be described as a transformation matrix denoted \( \mathcal{W} M_b \).

The transformation between the ideal and the world frame can be called \textit{deviation} and can be represented by the matrix \( \mathcal{W} M_I = \mathcal{W} M_b \mathcal{I} M_b^{-1} \). 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

\[
\forall i \in \{0, \ldots, n\}, \mathcal{W} M_i = \mathcal{W} M_I \mathcal{I} M_i, \quad (1)
\]

where \( n \) is the number of limbs, \( \mathcal{W} M_i \) is the configuration of the \( i \)-th body in the world frame, and \( \mathcal{I} M_i \) 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 \( t_I \). 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 (i) 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 dis-
placement of contact points in the reconstructed kinematic-
ics. 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 to 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

\[
f_{c,i} = -K_{f,p}t_{c,i} - K_{f,d}\dot{t}_{c,i} \tag{2}
\]

where

\[
t_{c,i} = p_{c,i} - z_{p_{c,i}} = R_{I}T p_{c,i} + t_{z} - z_{p_{c,i}}, \tag{3}
\]

\[
\dot{t}_{c,i} = [\omega_{I}] \times R_{I}T p_{c,i} + \dot{t}_{I}. \tag{4}
\]

with \(\omega_{I}\) is the angular velocity vector of the deviation, i.e. \(R_{I} = [\omega_{I}] \times R_{I}\) where \([\cdot]_{\times}\) is the skew symmetric operator

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

and \(K_{f,p}\) and \(K_{f,d}\) are \(3 \times 3\) symmetric positive de-
finite matrices representing elasticity and viscosity coeffi-
cients [19]. We sum the forces over the \(n_{c}\) contacts to obtain
the resultant contact force \(f_{c} = \sum_{i=1}^{n_{c}} f_{c,i}\).

The contact moment is the combination of the reaction
torque of the torsional damped spring and the moment of
the forces \(f_{c,i}\). This gives the following expression

\[
m_{c,i} = -K_{m,p}\Omega_{I} - K_{m,d}\omega_{I} + [p_{c,i}]_{\times} f_{c,i}. \tag{5}
\]

Where \(\Omega_{I}\), also called rotation vector, is the minimum norm
vector in \(\mathbb{R}^{3}\) so that \(R_{I} = \exp([\omega_{I}]_{\times})\) 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 \(m_{c,i}\) over the contacts to
obtain the resultant moment: \(m_{c} = \sum_{i=1}^{n_{c}} m_{c,i}\).

C. The feed-forward state dynamics

Let us first define our state vector

\[
x = (t_{I}^{T} \Omega_{I}^{T} \dot{t}_{I}^{T} \omega_{I}^{T} f_{c,1}^{T} m_{c,1}^{T} \ldots f_{c,n_{c}}^{T} m_{c,n_{c}}^{T} f_{e}^{T} m_{e}^{T})^{T} \tag{6}
\]

where \(t\) upper-script stands for transpose, and \(f_{e}\) and \(m_{e}\) 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 \(f_{c,i}\) and \(m_{c,i}\)
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:

\[
f_{c} + f_{e} - gm_{u_{z}} = \frac{d}{dt} \left( \sum_{i=1}^{n_{c}} m_{c,i} \dot{e}_{i} \right), \tag{7}
\]

\[
m_{c} + m_{e} - [R_{I}c_{i}]_{\times} gm_{u_{z}} = \sigma
\]

\[
ed = \frac{d}{dt} \left( \sum_{i=1}^{n_{c}} \left( R_{i}I_{i} R_{i}^{T} \omega_{i} + m_{c,i} [c_{i}]_{\times} \dot{e}_{i} \right) \right) \tag{9}
\]

where \(u_{z} = [0 \ 0 \ 1]^{T}\) is the vertical unit vector along
the z-axis, \(g\) the standard gravity, \(c_{i}\) is the position of the
CoM of the \(i\)-th body, \(I_{i}\) and \(R_{i}\) and \(\omega_{i}\) are the inertia matrix,
the orientation of the \(i\)-th body and its angular velocity
respectively.

In the Ideal frame \(I\) 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, \(c_{I}, \dot{c}_{I}, \ddot{c}_{I}\),
respectively, the 3 \(\times\) 3 tensor of inertia of the robot \(I_{z}\)
and its time-derivative \(\dot{I}_{z}\), and the angular momentum \(\sigma_{z}\)
and its derivative \(\dot{\sigma}_{z}\). 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:

\[
\omega_{I} = R_{I}(I_{z} + m[c_{I}]_{\times})^{-1} R_{I}^{T}
\]

\[
(m_{c} + m_{e} - [Rcz_{I} + t_{z}]_{\times} (\dot{c}_{e} + f_{e})
\]

\[
- \left( (w_{z})_{\times} R_{I}R_{I}^{T} R_{I}^{T} + R_{I} \sigma_{z} [w_{z}]_{\times} R_{I} \sigma_{z} \right)
\]

\[
+ m[Rcz_{I}] (Rcz_{I} + 2[w_{z}]_{\times} Rcz_{I} + [w_{z}]_{\times}^{2} Rcz_{I})
\]

and:

\[
\dot{I}_{z} = \frac{1}{m} (f_{e} + f_{c}) - (Rcz_{I} + 2[w_{z}]_{\times} Rcz_{I} + [w_{z}]_{\times}^{2} Rcz_{I})
\]

\[
+ [Rcz_{I}] \omega_{I} \tag{11}
\]

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 \(f_{e}\) and \(m_{e}\) is that they are constant in time, which
allows filtering their noise. This gives the model
\[ \dot{t}_s = 0 \]  
\[ \dot{m}_c = 0 \]  

(12) 

(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:

\[ y_{f_s} = \mathcal{T} R_t^s \omega_s + \mathcal{T} R_t^s R_t^w \omega_t \]  
\[ y_{m_s} = \mathcal{T} R_t^s \mathcal{P}_s \]  

(14) 

(15)

where \( y_{f_s} \) is the force measurement and \( y_{m_s} \) is the moment measurement of the \( i \)-th contact and \( \mathcal{T} R_t^s \) 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 force sensor in the ideal frame.

\[ \omega_I \]

\[ \dot{x}_k = \frac{1}{2} \left( T \dot{x}_k + T \dot{\theta}_k \right) \]  
\[ V_{log}(\exp[\Theta \omega_x]_x \exp[\Theta \omega_z]_x) \]  
\[ \dot{\theta}_k = \frac{1}{2} \left( T \dot{\theta}_k + T \dot{\theta}_z \right) \]  
\[ \omega_I + T \omega_I \]  
\[ -K_{f,p} \omega_{c,k+1} - K_{f,p} \omega_{c,k+1} \]  
\[ -K_{m,p} \omega_{c,k+1} - K_{m,d} \omega_{c,k+1} + [\mathbf{p}_{c,i,k+1}]_x f_{c,i,k+1} \]  
\[ \ldots \]  
\[ -K_{f,p} \omega_{c,n,c,k+1} - K_{f,p} \omega_{c,n,c,k+1} \]  
\[ -K_{m,p} \omega_{c,n,c,k+1} - K_{m,d} \omega_{c,n,c,k+1} + [\mathbf{p}_{c,n,c,k+1}]_x f_{c,n,c,k+1} \]  
\[ f_{c,k} \]  
\[ m_{c,k} \]  

(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 \( R = \exp\left([V_{log}(R)]_x\right) \), and \( \dot{x}_k \) and \( \omega_I \) are computed using (11) and (10) respectively using values in \( x_k \). 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

\[ f = R_s y_a - \sum_{i=1}^{n_s} R_{c_i} y_{f_i}, \]

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. Never-
theless 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 \( p_e \). This
distance gives an evaluation of the error in the estimation of
the contact axis. It is obtained by the following expression.
TABLE I
SUMMARY OF THE RESULTS - INCLUDING SENSOR-BASED ESTIMATION (GREYED) [14].

<table>
<thead>
<tr>
<th>Measure</th>
<th>Model-based Force (N)</th>
<th>Model-based Moment (N.m)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>x</td>
<td>y</td>
</tr>
<tr>
<td>Simulation</td>
<td>-27.98</td>
<td>3.06</td>
</tr>
<tr>
<td>RMSE</td>
<td>2.90</td>
<td>0.026</td>
</tr>
<tr>
<td>Real robot</td>
<td>-25.50</td>
<td>4.80</td>
</tr>
<tr>
<td>RMSE</td>
<td>3.82</td>
<td>1.95</td>
</tr>
</tbody>
</table>

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.

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 p_c 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.

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

REFERENCES


