

Nonlinear State Estimation for Humanoid Robot Walking*

Stylios Piperakis, Maria Koskinopoulou and Panos Trahanias

Abstract—This article presents a novel cascade state estimation framework for 3D Center of Mass (CoM) estimation of walking humanoid robots. The proposed framework, called SEROW (State Estimation Robot Walking) fuses effectively joint encoder, inertial, feet pressure and visual odometry measurements. Initially, we consider the humanoid’s Newton-Euler dynamics and rigorously derive the non-linear CoM estimator. The latter accurately estimates the 3D-CoM position, velocity and external forces acting on the CoM, while directly considering the presence of uneven terrain and the body’s angular momentum rate and thus effectively coupling the frontal with the lateral plane dynamics. Furthermore, we extend an established floating mass estimator to take into account the support foot pose, yielding in such a way the mandatory, for CoM estimation, affine transformations and forming a cascade state estimation scheme. Subsequently, we quantitatively and qualitatively assess the proposed scheme by comparing it to other estimation structures in terms of accuracy and robustness to disturbances, both in simulation and on an actual NAO robot walking outdoors over an inclined terrain. To facilitate further research endeavors, our implementation is offered as an open-source ROS/C++ package.

I. INTRODUCTION

Generating robust and stable omnidirectional gait for humanoid robots is a very challenging task. Difficulties arise due to the large number of degrees of freedom and the highly non-linear rigid-body dynamics of humanoids, the under-actuation experienced during the walking cycle, and the unilateral multi-contacts with the ground. In order to simplify gait planning and control, the dynamics are commonly approximated with mass concentrated models.

In early approaches, the Zero-Moment Point (ZMP) was regulated in order to achieve dynamic stable locomotion with feedback from the Center of Mass (CoM) [1], [2]. Other approaches considered step placing utilizing the Capture Point (CP) [3] [4], defined as a linear combination of CoM position and velocity, in order to maintain stability. Interestingly, the latter scheme was extended in three dimensions with the Divergent Component of Motion (DCM) [5] to allow for walking on uneven ground [6]. Such approaches proved robust and yielded a wide variety of omnidirectional walking gaits when accurate feedback was employed.

To this end, state estimation has a vital role in walking pattern generation and real-time gait control. Kuindersma et

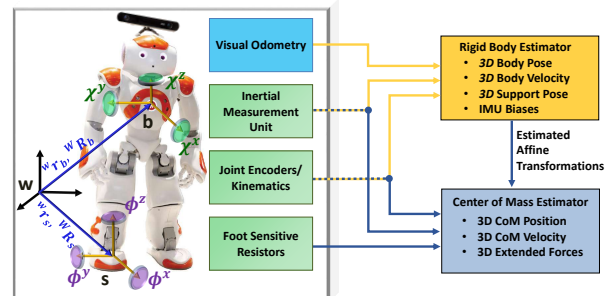


Fig. 1: Cascade state estimation scheme consisting of a rigid body estimator and a CoM estimator.

al. [7] proposed a rigid body estimator based on Newton-Euler dynamics of a floating mass for estimating the body position, orientation, and velocity utilizing an IMU, the robot kinematics, and a LIDAR sensor, yielding very low drift. This scheme was extended in [8] by considering the visually obtained terrain landscape, rendering an ATLAS robot able to walk continuously up and down staircases. A similar approach was proposed by Bloesch et al. [9] for quadruped robots, where the IMU and the kinematic measurements were used to estimate the base motion and the feet position. Subsequently, the latter scheme was appropriately adapted in [10] for humanoids while also considering the feet orientation.

In many popular walking pattern generators and real-time gait controllers, the 3D-CoM position and velocity is needed. Hence, Carpintieri et al. [11] used a complimentary filter for 3D-CoM estimation based on consistent dynamics. Rotela et al. [12] proposed a momentum estimator for 3D-CoM position, velocity and external wrench estimation. Nevertheless, both [11], [12] explicitly assume that 6D-Force/Torque (F/T) sensors are employed on the robot’s feet.

Stephens [13] demonstrated an approach based on the Linear Inverted Pendulum Mode (LIPM) dynamics to estimate the 2D-CoM position and velocity. The latter approach was also studied in [14] where two Kalman Filters (KFs), one based on the LIPM dynamics and one on the robot’s planar dynamics were compared. The planar KF performed more accurately, since it considered a more precise representation of the dynamics, but it was robot specific, harder to design, implement and tune contrasted to the LIPM KF.

However, when the LIPM dynamics are employed, one postulates that the dynamics in the x and y axes are independent and furthermore, that the CoM lies on a constant height plane. Presumably this is not the case in real conditions, and definitely not when the robot walks on uneven ground. Therefore, in our previous work [15], we presented a non-

*This work has been partially supported by the following funded projects: EU FET grant GA 641100 Timestorm, EU FET grant GA 754423 ENTIMENT, Greek National grand MIS 5002469 Advanced Research Activities in Biomedical and Agro alimentary Technologies, NSRF 2014-2020, and Greek National grand MIS 5030440 vipGPU.

The authors are with the Institute of Computer Science, Foundation for Research and Technology - Hellas (FORTH). {spiperakis, mkosk, trahania}@ics.forth.gr

linear extension of the LIPM based KFs, where by utilizing an Extended Kalman Filter (EKF), the full 3D-CoM position, velocity and modeling errors can be readily estimated.

In this work, we propose a robust non-linear state estimation framework for accurately estimating the 3D-CoM position, velocity and external forces acting on the CoM by effectively utilizing joint encoder, Foot Sensitive Resistor (FSR), and IMU measurements. Starting from the Newton-Euler humanoid dynamics, we rigorously derive a non-linear CoM estimator that uses as input the 3D Center of Pressure (COP), the vertical Ground Reaction Force (GRF), and the horizontal angular momentum rate. The output of the estimator is formulated as the 3D-CoM position along with the 3D-CoM acceleration. To the best of our knowledge, this is the first time that a CoM estimator explicitly considers the ground height and the angular momentum rate without relying on F/T sensors to yield, besides the 3D-CoM position and velocity, accurate 3D external force estimates. Contrasted to [15], the modeling errors in the acceleration level in our formulation represent exactly the external forces acting on the CoM and, furthermore, the angular momentum rate is taken into direct account. Thus, it is possible to provide more accurate estimates, when the motion is highly dynamic and the angular momentum rate is significant. In addition, this estimator can cope with cases of walking on uneven terrain, since the height of the ground is properly considered.

As it is standard practice in CoM estimators, all measurements before fused are transformed from their local frames to the world frame. Therefore, by extending the rigid body estimator in [7], we provide the indispensable transformations that link the robot's body and support foot to a world frame, by fusing the onboard joint encoders, IMU and the pose obtained with visual odometry. Contrasted to [7], our approach differs in that: (a) the 3D-support foot position and orientation are properly considered, (b) kinematically computed 3D-relative support foot position and orientation are fused, (c) visual odometry measurements are considered, and (d) the linearizations for the aforementioned quantities are derived. In addition, contrasted to [10], the proposed estimator: (a) maintains a robocentric state-space which improves the linearization accuracy and reduces drift [16], (b) incorporates visual odometry measurements, (c) considers only the support foot in the state which reduces the dimension of the filter by six, and (d) maintains rotational quantities directly as rotation matrices.

In summary, the proposed estimator comprises a modular, low-dimension, cascade state estimation scheme (Figure 1), consisting of two EKFs in tandem. This results in a low computational complexity and efficient implementation that is appropriate for onboard execution. Given that the proposed cascade framework is based on generic dynamics, and thus is readily amenable to generalization to other humanoids, we offer our implementation as an open-source ROS package termed SEROW (State Estimation Robot Walking) [17].

This article is organized as follows: in section II the proposed CoM estimator is mathematically established. Then, section III presents our extension to the rigid body estimator.

In section IV, the cascade scheme is quantitatively and qualitatively assessed. Finally, section V concludes the article.

II. CENTER OF MASS ESTIMATION

In this section, we formally derive a non-linear CoM state estimator and investigate its observability properties. In the following all quantities listed are in the world frame and the x, y, z superscripts indicate the corresponding vector coordinates. Consider the Newton-Euler equations of motion for a humanoid robot, where the ground contact forces \mathbf{f}_i are explicitly separated from the external forces \mathbf{f}_e acting on the CoM:

$$m(\ddot{\mathbf{c}} + \mathbf{g}) = \mathbf{f}_e + \sum_i \mathbf{f}_i \quad (1)$$

$$m\mathbf{c} \times (\ddot{\mathbf{c}} + \mathbf{g}) + \dot{\mathbf{L}} = \mathbf{c} \times \mathbf{f}_e + \sum_i \mathbf{s}_i \times \mathbf{f}_i \quad (2)$$

where \mathbf{c} is the CoM position, $\ddot{\mathbf{c}}$ is the CoM acceleration, $\dot{\mathbf{L}}$ is the rate of angular momentum, m is the mass of the robot, and $-\mathbf{g}$ is the gravity vector. Since \mathbf{s}_i are the position of the contact points, the COP is defined as:

$$\mathbf{p} = \left[\frac{\sum_i s_i^x f_i^z}{\sum_i f_i^z} \quad \frac{\sum_i s_i^y f_i^z}{\sum_i f_i^z} \quad s^z \right] \quad (3)$$

where we assume that in each foot, contact points are coplanar with respect to the local foot frame.

Then, by solving the first two equations of (2) for \ddot{c}^x and \ddot{c}^y while also considering (1), we get:

$$\ddot{c}^x = \frac{(c^x - p^x)(m(\ddot{c}^z + g^z) - f_e^z) - \dot{L}^y}{m(c^z - p^z)} + \frac{1}{m} f_e^x \quad (4)$$

$$\ddot{c}^y = \frac{(c^y - p^y)(m(\ddot{c}^z + g^z) - f_e^z) + \dot{L}^x}{m(c^z - p^z)} + \frac{1}{m} f_e^y \quad (5)$$

Examining the z component of (1) and introducing $\sum_i f_i^z = f_N$ as the vertical GRF, we get:

$$\ddot{c}^z = \frac{1}{m} (f_N + f_e^z) - g^z \quad (6)$$

By substituting (6) in (4) and (5), we readily obtain the non-linear dynamics that our CoM estimator is based on:

$$\ddot{c}^x = \frac{c^x - p^x}{m(c^z - p^z)} f_N - \frac{\dot{L}^y}{m(c^z - p^z)} + \frac{1}{m} f_e^x \quad (7)$$

$$\ddot{c}^y = \frac{c^y - p^y}{m(c^z - p^z)} f_N + \frac{\dot{L}^x}{m(c^z - p^z)} + \frac{1}{m} f_e^y \quad (8)$$

$$\ddot{c}^z = \frac{1}{m} (f_N + f_e^z) - g^z \quad (9)$$

A. CoM Estimator Process Model

For deriving the state-space needed in the EKF formulation, we assume a flying-wheel on the body with inertia \mathbf{I}_b . The latter is constantly computed based on the configuration of the limbs, to approximate the rate of angular momentum:

$$\dot{\mathbf{L}} = \mathbf{I}_b \dot{\boldsymbol{\omega}}_b + \boldsymbol{\omega}_b \times \mathbf{I}_b \boldsymbol{\omega}_b \quad (10)$$

where $\boldsymbol{\omega}_b$ is the gyro rate. Note that the second term in (10) accounts for the Coriolis and centrifugal effects. Subsequently, the following state vector is formulated:

$$\mathbf{x}_t^c = [c^x \ c^y \ c^z \ \dot{c}^x \ \dot{c}^y \ \dot{c}^z \ f_e^x \ f_e^y \ f_e^z]^\top$$

where the superscript c denotes the CoM estimator.

Furthermore, let the filter's input \mathbf{u}_t^c be the location of the COP \mathbf{p} in the 3D space with respect to the world frame, along with the vertical GRF f_N as measured by the FSRs. In addition, we compute the gyro acceleration $\dot{\omega}_b$ by numerical differentiation of the IMU's gyro rate. Since numerical differentiation amplifies noise, we filter the gyro acceleration with a small window moving average filter to avoid introducing significant delays and phase shifts.

To this end, the input vector is:

$$\mathbf{u}_t^c = [p^x \ p^y \ p^z \ f_N \ \dot{L}^x \ \dot{L}^y]^\top \quad (11)$$

and the process model assumes the standard non-linear form:

$$\dot{\mathbf{x}}_t^c = \mathbf{f}(\mathbf{x}_t^c, \mathbf{u}_t^c) + \mathbf{w}_t^c \quad (12)$$

where

$$\frac{d}{dt} \begin{bmatrix} c^x \\ c^y \\ c^z \\ \dot{c}^x \\ \dot{c}^y \\ \dot{c}^z \\ f_e^x \\ f_e^y \\ f_e^z \end{bmatrix} = \begin{bmatrix} \dot{c}^x \\ \dot{c}^y \\ \dot{c}^z \\ \frac{c^x - p^x}{m(c^z - p^z)} f_N - \frac{\dot{L}^y}{m(c^z - p^z)} + \frac{1}{m} f_e^x \\ \frac{c^y - p^y}{m(c^z - p^z)} f_N + \frac{\dot{L}^x}{m(c^z - p^z)} + \frac{1}{m} f_e^y \\ \frac{1}{m} (f_N + f_e^z) - g^z \\ 0 \\ 0 \\ 0 \end{bmatrix} + \mathbf{w}_t^c \quad (13)$$

with \mathbf{w}_t^c the process additive noise. The linearization of the state-space is straightforward. The state Jacobian of the dynamics is:

$$\mathbf{F}_t = \frac{\partial \mathbf{f}}{\partial \mathbf{x}} = \begin{bmatrix} \mathbf{0} & \mathbf{I} & \mathbf{0} \\ \mathbf{G}_t & \mathbf{0} & \mathbf{M}_t \\ \mathbf{0} & \mathbf{0} & \mathbf{0} \end{bmatrix} \quad (14)$$

where

$$\mathbf{G}_t = \begin{bmatrix} \frac{f_N}{m(c^z - p^z)} & 0 & -\frac{(c^x - p^x)f_N - \dot{L}^y}{m(c^z - p^z)^2} \\ 0 & \frac{f_N}{m(c^z - p^z)} & -\frac{(c^y - p^y)f_N + \dot{L}^x}{m(c^z - p^z)^2} \\ 0 & 0 & 0 \end{bmatrix}$$

$$\mathbf{M}_t = \begin{bmatrix} \frac{1}{m} & 0 & 0 \\ 0 & \frac{1}{m} & 0 \\ 0 & 0 & \frac{1}{m} \end{bmatrix}$$

B. CoM Estimator Measurement Model

The measurements fused in the update step are the kinematically computed CoM position c^{enc} and the IMU CoM acceleration \ddot{c}^{imu} , computed as in [18]. This approximation, as well as the approximation in (10), are valid as long as the actual CoM is located inside the same rigid link as the IMU, i.e. the body link.

Accordingly, since the CoM acceleration is not part of the state, the measurement model is also non-linear:

$$\mathbf{y}_t^c = \mathbf{h}(\mathbf{x}_t^c, \mathbf{u}_t^c) + \mathbf{n}_t^c \quad (15)$$

with

$$\mathbf{h}(\mathbf{x}_t^c, \mathbf{u}_t^c) = \begin{bmatrix} c^x \\ c^y \\ c^z \\ \frac{c^x - p^x}{m(c^z - p^z)} f_N - \frac{\dot{L}^y}{m(c^z - p^z)} + \frac{1}{m} f_e^x \\ \frac{c^y - p^y}{m(c^z - p^z)} f_N + \frac{\dot{L}^x}{m(c^z - p^z)} + \frac{1}{m} f_e^y \\ \frac{1}{m} (f_N + f_e^z) - g^z \end{bmatrix} \quad (16)$$

and \mathbf{n}_t^c the additive Gaussian noise. The measurement model linearization is derived similarly as in section II-A.

Notice, by using (7)-(9) both in the process and in the measurement model, the disturbance input noise correlates with the measurement noise. Still this has no effect on the estimation error itself but rather on the error covariance (see <https://goo.gl/aJwU4v> for a mathematical proof). Moreover, when the cross-correlation is zero, all expressions reduce to the EKF formulas. In all our walking experiments, including ones of sufficient duration, we haven't noticed any degradation in the estimation accuracy of the error covariance. Thus, the cross-correlation noise ought to be insignificantly small.

C. Non-Linear Observability Analysis

In this section, we investigate the observability properties of the proposed CoM estimator in terms of the local non-linear observability matrix. Following the approach in [19], that allows the non-linear observability analysis to take into account output dynamics that depend explicitly on the input \mathbf{u} , we define the following coordinates $(h_1, \varphi_1^1, h_2, \varphi_2^1, h_3, \varphi_3^1, h_4, h_5, h_6)$, on the current operating point $(^* \mathbf{x}_t^c, ^* \mathbf{u}_t^c)$ where h_j is the j -th row of $\mathbf{h}(^* \mathbf{x}_t^c, ^* \mathbf{u}_t^c)$ and $\varphi_i^1 = L_f h_i$ is the Lie derivative of h_i in the direction of the vector field $\mathbf{f}(^* \mathbf{x}_t^c, ^* \mathbf{u}_t^c)$. Using these coordinates, we form the map [19]:

$$\Phi(^* \mathbf{x}_t^c, ^* \mathbf{u}_t^c) = \begin{bmatrix} ^* c^x \\ ^* \dot{c}^x \\ ^* c^y \\ ^* \dot{c}^y \\ ^* c^z \\ ^* \dot{c}^z \\ \frac{^* c^x - ^* p^x}{m(^* c^z - ^* p^z)} ^* f_N - \frac{^* \dot{L}^y}{m(^* c^z - ^* p^z)} + \frac{1}{m} ^* f_e^x \\ \frac{^* c^y - ^* p^y}{m(^* c^z - ^* p^z)} ^* f_N + \frac{^* \dot{L}^x}{m(^* c^z - ^* p^z)} + \frac{1}{m} ^* f_e^y \\ \frac{1}{m} (^* f_N + ^* f_e^z) - g^z \end{bmatrix}$$

Subsequently, the components are re-ordered for convenience to form $\tilde{\Phi}$. Computing the Jacobian with respect to $^* \mathbf{x}_t$, we get the local non-linear observability matrix:

$$\mathcal{O} = \frac{\partial \tilde{\Phi}(^* \mathbf{x}_t, ^* \mathbf{u}_t)}{\partial ^* \mathbf{x}_t} = \begin{bmatrix} \mathbf{I} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{I} & \mathbf{0} \\ ^* \mathbf{G}_t & \mathbf{0} & ^* \mathbf{M}_t \end{bmatrix} \quad (17)$$

Ignoring the unrealistic case where $c^z = p^z$, meaningly the CoM lies exactly on the ground, we find that the non-linear local observability matrix is full rank and cannot drop rank, since $\det \mathcal{O} = \frac{1}{m^3}$. Thus, the dynamics in (12), (15) are locally observable in all cases.

D. The need for Rigid Body Estimation

All measurements fused by the CoM estimator must be in an inertial frame of reference. Still, the latter are typically obtained in local frames, i.e. the kinematically computed CoM ${}^b\mathbf{c}^{\text{enc}}$ is derived in the body frame and the measured by the FSR COP ${}^s\mathbf{p}^{\text{fsr}}$ is in the support foot frame. Accordingly, they must be transformed to the world frame as:

$${}^w\mathbf{c}^{\text{enc}} = {}^w\mathbf{r}_b + {}^w\mathbf{R}_b {}^b\mathbf{c}^{\text{enc}} \quad (18)$$

$${}^w\mathbf{p}^{\text{fsr}} = {}^w\mathbf{r}_s + {}^w\mathbf{R}_s {}^s\mathbf{p}^{\text{fsr}} \quad (19)$$

with ${}^w\mathbf{r}_b$ and ${}^w\mathbf{r}_s$ the position of the body and support foot with respect to the world frame, ${}^w\mathbf{R}_b$ and ${}^w\mathbf{R}_s$ the corresponding orientations expressed as rotation matrices. To this end, having reliable estimates of the body and support foot transformations is crucial in CoM estimation.

III. RIGID BODY ESTIMATION

In [7] a rigid body state estimator based on Newton-Euler dynamics of a floating mass was presented. Since, the transformation linking the support foot to the world frame is mandatory to use quantities measured in the support foot frame such as the GRFs and the COP, we appropriately extend the process and measurement models to be able to estimate the following state vector:

$$\mathbf{x}_t^r = [{}^b\mathbf{v}_b \quad {}^w\mathbf{R}_b \quad {}^w\mathbf{r}_b \quad {}^w\mathbf{R}_s \quad {}^w\mathbf{r}_s \quad \mathbf{b}_\omega \quad \mathbf{b}_\alpha]^\top$$

where the superscript r denotes the rigid body estimator, ${}^b\mathbf{v}_b$ is the body's velocity, and \mathbf{b}_ω , \mathbf{b}_α are the gyro and accelerometer biases, all expressed in the body local frame.

This EKF provides the necessary for CoM estimation rigid body transformations and accordingly preserves their affine properties. Hence, given Gaussian inputs the probability densities of the transformed output quantities remain Gaussians and thus the formed cascade estimation scheme does not give rise to inconsistencies during filtering.

A. Rigid Body Estimator Process Model

Let ${}^b\bar{\omega}_b = {}^b\omega_b^{\text{imu}} + \mathbf{b}_\omega$ and ${}^b\bar{\alpha}_b = {}^b\alpha_b^{\text{imu}} + \mathbf{b}_\alpha$, represent the IMU bias-removed gyro rate and linear acceleration, respectively. Then the non-linear state-space takes the form:

$${}^b\dot{\mathbf{v}}_b = -({}^b\bar{\omega}_b + \mathbf{w}_\omega) \times {}^b\mathbf{v}_b - {}^w\mathbf{R}_b^\top \mathbf{g} + {}^b\bar{\alpha}_b + \mathbf{w}_a \quad (20)$$

$${}^w\dot{\mathbf{R}}_b = {}^w\mathbf{R}_b ({}^b\bar{\omega}_b + \mathbf{w}_\omega)_{[\times]} \quad (21)$$

$${}^w\dot{\mathbf{r}}_b = {}^w\mathbf{R}_b {}^b\mathbf{v}_b \quad (22)$$

$${}^w\dot{\mathbf{R}}_s = {}^w\mathbf{R}_s \mathbf{w}_s_{[\times]} \quad (23)$$

$${}^w\dot{\mathbf{r}}_s = \mathbf{w}_{r_s} \quad (24)$$

$$\dot{\mathbf{b}}_\omega = \mathbf{w}_{b_\omega} \quad (25)$$

$$\dot{\mathbf{b}}_\alpha = \mathbf{w}_{b_\alpha} \quad (26)$$

where $[\times]$ denotes the wedge operation and (23), (24) have been introduced to model the support foot orientation and position as random walks, since the foot in contact may or may not be stationary due to possible slippage. Furthermore, \mathbf{w}_ω and \mathbf{w}_α are the IMU noise vectors for the gyro rate and the linear acceleration, respectively, \mathbf{w}_s and \mathbf{w}_{r_s} are the

support foot orientation and position noises and \mathbf{w}_{b_ω} , \mathbf{w}_{b_α} are the IMU bias noises.

To track the body's and support foot's orientation uncertainty we consider perturbation rotations in the corresponding local frames. Thus, if the true body and support rotation matrices are ${}^w\mathbf{R}_b$ and ${}^w\mathbf{R}_s$, then ${}^w\mathbf{R}_b = {}^w\hat{\mathbf{R}}_b e^{\chi_{[\times]}}$ and ${}^w\mathbf{R}_s = {}^w\hat{\mathbf{R}}_s e^{\phi_{[\times]}}$ where ${}^w\hat{\mathbf{R}}_b$, ${}^w\hat{\mathbf{R}}_s$ are the estimated rotation matrices and χ , ϕ are the corresponding error exponential coordinates. An illustration of those quantities is given in Figure 1, where the black frame is the world frame, the yellow frames indicate the local body and support foot frames, while green and purple circles represent the corresponding orientation errors.

Subsequently, the linearization of (20) - (26) is derived as:

$${}^b\delta\dot{\mathbf{v}}_b = -{}^b\bar{\omega}_b \times {}^b\delta\mathbf{v}_b - ({}^w\mathbf{R}_b^\top \mathbf{g}) \times \chi + {}^b\mathbf{v}_b \times (\delta\mathbf{b}_\omega + \mathbf{w}_\omega) + \delta\mathbf{b}_\alpha + \mathbf{w}_\alpha \quad (27)$$

$$\dot{\chi} = -{}^b\bar{\omega}_b \times \chi + \delta\mathbf{b}_\omega + \mathbf{w}_\omega \quad (28)$$

$${}^w\delta\dot{\mathbf{r}}_b = {}^w\mathbf{R}_b {}^b\delta\mathbf{v}_b - {}^w\mathbf{R}_b ({}^b\mathbf{v}_b \times \chi) \quad (29)$$

$$\dot{\phi} = \mathbf{w}_s \quad (30)$$

$$\delta\dot{\mathbf{r}}_s = \mathbf{w}_{r_s} \quad (31)$$

$$\delta\dot{\mathbf{b}}_\omega = \mathbf{w}_{b_\omega} \quad (32)$$

$$\delta\dot{\mathbf{b}}_\alpha = \mathbf{w}_{b_\alpha} \quad (33)$$

B. Rigid Body Estimator Measurement Model

The output model, formulated in [7], consists of the global body velocity using the robot's kinematics and the global body position and orientation using a LIDAR sensor and a Gaussian particle filter. To obtain the body velocity, the body position was computed using the filter's estimated orientation and the kinematically computed relative position of the support foot with respect to the body ${}^b\mathbf{r}_s^{\text{enc}}$ and then it was numerically differentiated. However, when using the estimated orientation (which is part of the state) for a measurement, correlation is induced to the filter. In addition, numerical differentiation commonly amplifies the noise and further filtering is needed.

Interestingly, it is possible to directly fuse ${}^b\mathbf{r}_s^{\text{enc}}$ since both the body and support foot position are available in our state. Moreover, the relative orientation ${}^b\mathbf{R}_s^{\text{enc}}$ must be also fused to render the support foot orientation observable:

$${}^b\mathbf{r}_s^{\text{enc}} = {}^w\mathbf{R}_b^\top ({}^w\mathbf{r}_s - {}^w\mathbf{r}_b) + \mathbf{n}_{r_s} \quad (34)$$

$${}^b\mathbf{R}_s^{\text{enc}} = {}^w\mathbf{R}_b^\top {}^w\mathbf{R}_s e^{\mathbf{n}_{r_{[\times]}}} \quad (35)$$

with \mathbf{n}_{r_s} , \mathbf{n}_r the kinematics measurement noise.

The previous measurements are typically available at a very fast rate. In this work, we also employ measurements of the global head position and orientation by mounting an external camera on the head of the robot and using a visual odometry algorithm. The latter are then kinematically transformed to obtain the global body position and orientation and fused as:

$${}^w\mathbf{r}_b^{\text{cam}} = {}^w\mathbf{r}_b + \mathbf{n}_{r_b} \quad (36)$$

$${}^w\mathbf{R}_b^{\text{cam}} = {}^w\mathbf{R}_b e^{\mathbf{n}_{b_{[\times]}}} \quad (37)$$

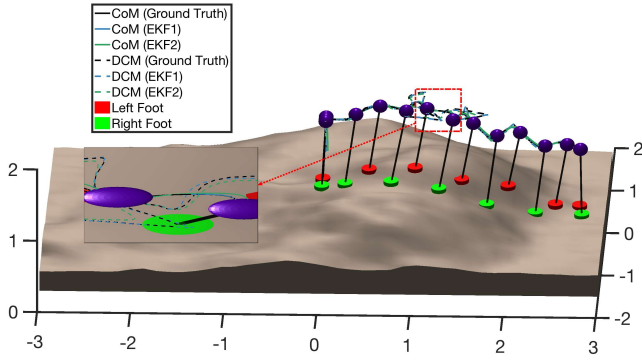


Fig. 2: CoM/DCM trajectories in the 3D space, black / black-dotted lines indicate the ground truth trajectories, blue / blue-dotted lines the EKF1 estimated trajectories, green / green-dotted lines the EKF2 estimated trajectories and red / green the left and right support foot respectively.

where \mathbf{n}_{r_b} , \mathbf{n}_b the camera measurement noise. This addition is essential, since leg odometry tends to drift and become inaccurate. Interestingly, this is also verified in the outdoors walking experiments presented in section IV-B.

For the linearization of the output model we consider the error exponential coordinates ζ^{enc} and ψ^{str} related with ${}^b\mathbf{R}_s^{\text{enc}}$ and ${}^w\mathbf{R}_b^{\text{cam}}$, respectively. To this end, the linearization of (34) - (37) is given by:

$${}^b\delta\mathbf{r}_s^{\text{enc}} = {}^w\mathbf{R}_b^{\text{T}} ({}^w\delta\mathbf{r}_s - {}^w\delta\mathbf{r}_b) + ({}^w\mathbf{R}_b^{\text{T}} ({}^w\mathbf{r}_s - {}^w\mathbf{r}_b)) \times \boldsymbol{\chi} + \mathbf{n}_{r_s} \quad (38)$$

$$\zeta^{\text{enc}} = - ({}^w\mathbf{R}_s^{\text{T}} {}^w\mathbf{R}_b) \boldsymbol{\chi} + \boldsymbol{\phi} + \mathbf{n}_r \quad (39)$$

$${}^w\delta\mathbf{r}_b^{\text{cam}} = {}^w\delta\mathbf{r}_b + \mathbf{n}_{r_b} \quad (40)$$

$$\psi^{\text{cam}} = \boldsymbol{\chi} + \mathbf{n}_b \quad (41)$$

IV. RESULTS

The proposed framework has been implemented and experimentally validated. In the next section we outline quantitative, simulation-based results, that demonstrate the accuracy and robustness of the proposed estimator in simulated gaits over uneven terrain. Subsequently, we present results on a NAO robot, and demonstrate accurate external force estimation and how drift affects the CoM estimation, highlighting the significance of the proposed cascade scheme. Given that disturbances tend to be sudden and discrete events, we employ in all experiments high process noise in order to facilitate fast convergence of the estimated external forces.

A. Simulation Experiments

1) *Humanoid Robot Walking over Rough Terrain:* In order to obtain quantitative assessment results, we simulated a humanoid robot walking over uneven terrain, while our non-linear CoM estimator is employed for feedback. The proposed CoM estimator, termed as EKF1, is contrasted to the non-linear estimator in [15], termed as EKF2, and to a Linear KF (LKF) variant of [18], which is the only linear scheme fusing CoM acceleration. The latter estimates a CoM offset instead of the external forces, thus the offsets are

transformed to forces as:

$$f_e^{x,y} = m \frac{g}{h_c} c_{\text{offset}}^{x,y} \quad (42)$$

where h_c is the nominal CoM height. The selection of EKF2 and LKF schemes for comparison is due to the fact that EKF2 has been shown to be an accurate 3D-CoM estimator [15] and LKF is broadly utilized in the literature [18].

For all employed filters ideal base/support state estimation was assumed and the same noise covariances \mathbf{Q} and \mathbf{R} are used. The 3D-step positions are computed based on the terrain's shape while the motion generation is based on the DCM $\boldsymbol{\xi}$ with continuous Double Support (DS) phases [6].

In this experiment, illustrated in Figure 2, the robot stands up, initializes its posture by taking two steps in place and starts to walk. During the third step and at $t = 6s$, a disturbance in the x axis of $2200N$ is introduced. After recovering within a step, another push happens in the y axis with intensity of $1500N$. Subsequently, in the following step the robot is perturbed in both x and y axes with $1800N$ and $1600N$, respectively. Due to this last push, early swing leg landing occurs causing a disturbance of approximately $1000N$ in the vertical axis. Finally, the robot manages to walk down the terrain unperturbed. Figure 3 shows both the 3D-CoM position (top) and velocity (middle) as estimated by the employed estimators contrasted to the ground-truth trajectories. We observe that the proposed CoM estimator yields more accurate estimates, which is due to the fact that the ground height in the denominators of (7) and (8) along with the angular momentum rates translate to modeling errors, yielding inaccuracies for all the estimated quantities of EKF2 and LKF, while in the EKF1 case they are directly considered. This is also evident in Figure 3 (bottom) illustrating the external forces, where as seen strong pushes, e.g. in x -axis, cause the robot to rotate about the y -axis, generating angular momentum and false appearing as external forces for EKF2 and LKF in that axis.

Moreover, EKF1 and EKF2, as expected, yield similar response in the z -axis since they are both based on (9). On the other hand, LKF yields no estimates since it assumes that CoM lies on a constant height plane.

Based on the above, in order to demonstrate the accuracy of the proposed CoM estimator, we conducted 100.000 simulations of 12 random omnidirectional steps each. In every run, random disturbances varying in magnitude from $1 - 1.5$, $0.5 - 1$, and $0.25 - 0.5$ times the weight of the robot in the x , y and z axes respectively, were introduced at random time instances during the gait. Figure 4 illustrates the Root Mean Square Error (RMSE) from the ground truth trajectories during the perturbation periods for each estimator employed. The external forces are scaled by 10^{-3} for clarity. As evident, we gained a significant boost in accuracy for all quantities of interest in the x and y axes, especially in external forces, for only 14.32% extra computational cost.

2) *Valkyrie Humanoid Walking over Uneven Terrain:*

Next, we employ the proposed cascade framework in Gazebo with NASA's Valkyrie humanoid using our ROS open-source

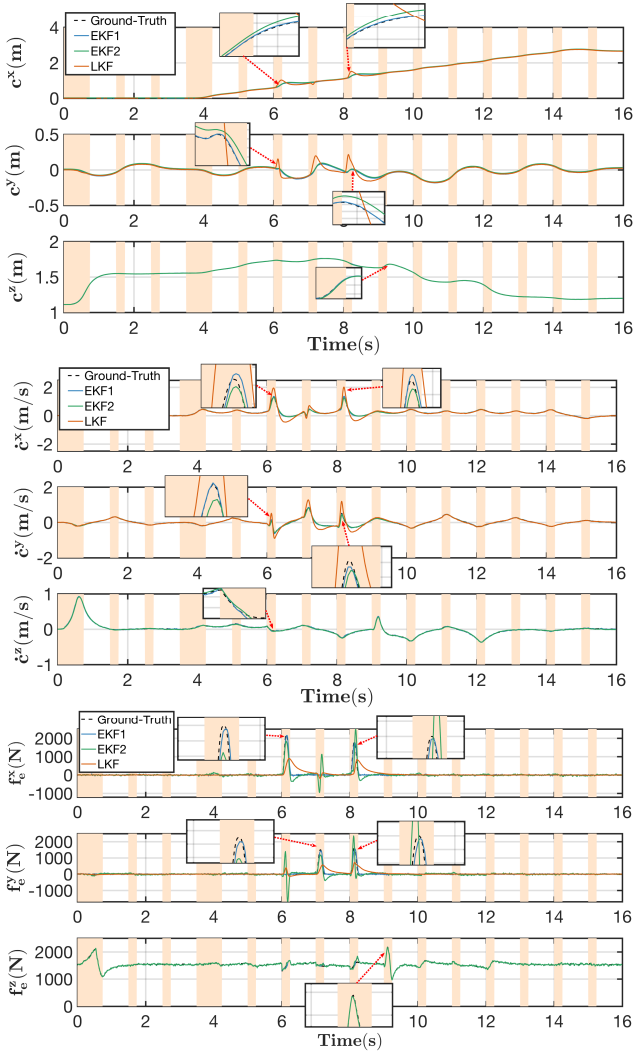


Fig. 3: **Top** 3D-CoM trajectories, **Middle** 3D-CoM velocities, **Bottom** 3D-External forces, light beige regions indicate the DS phases, black dotted lines indicate the ground truth trajectories, blue lines, green lines, and orange lines are the estimated trajectories by EKF1, EKF2, and LKF respectively.

implementation running in real-time every $2ms$, where also the parameters used for this experiment are listed in the Valkyrie configuration file [17]. For walking over the uneven terrain we utilized the control framework in [4]. The IMU measurements are available at $1kHz$ while the joint encoders and FSR measurements are obtained at $500Hz$. Furthermore, to compute the visual pose fused in our filter, we used the Semi-Direct Visual Odometry (SVO) [20] with the multisense stereo running at $40Hz$. In Figure 5 the 3D-Body position and velocity are illustrated. Notice the kinematically computed trajectories inevitably drift as the robot continuously walks, whereas, our rigid-body estimator, termed as EKF1, yielded accurate estimates for all quantities with respect to the ground-truth trajectories. Specifically, the RMSE for the body position were $0.0034m$, $0.0036m$ and $0.001m$, and for the body velocity $0.0139m/s$, $0.0159m/s$ and $0.0128m/s$ for the x , y and z axes, respectively.

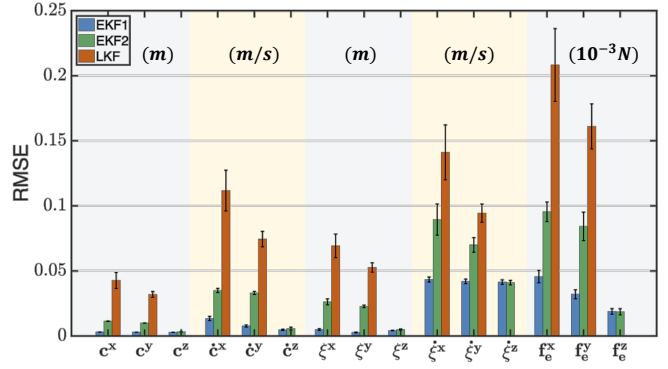


Fig. 4: RMSE for CoM, DCM, and external forces during perturbation periods for 100.000 simulations; blue bars indicate the EKF1s error, green bars the EKF2s error, orange bars the LKFs error, and black lines the standard deviation from the corresponding mean values.

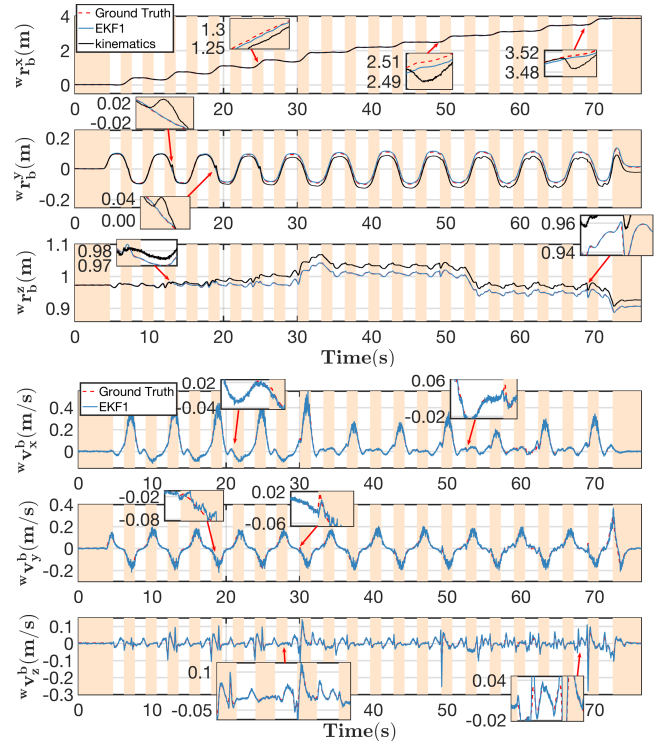


Fig. 5: **Top** 3D-Body trajectories, **Bottom** 3D-Body velocities, light beige regions indicate the DS phases, black dotted lines the ground truth trajectories, blue lines the estimated trajectories by EKF1 and black lines the leg odometry.

Since Valkyrie is employed with 6D-F/T sensors in the feet (as opposed to the simulated robot in section IV-A.1 and NAO in Section IV-B), we compare the proposed CoM estimator (termed as EKF1 for simplicity) to the Momentum Estimator (ME) with external wrenches [12]. For both filters the same base/support information and noise covariances for the measurements in common are used, whereas the torque and external torque covariances were fine tuned in the ME case. Figure 6 illustrates the 3D-CoM position and velocities as estimated by each method. As evident, for the 3D-CoM

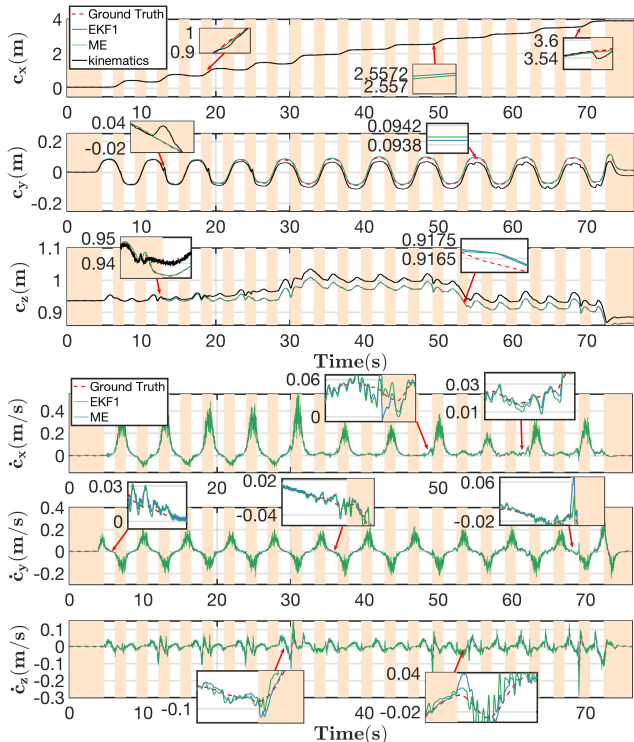


Fig. 6: **Top** 3D-CoM trajectories, **Bottom** 3D-CoM velocities, light beige regions indicate the DS phases, blue and green lines are the estimated trajectories by EKF1 and ME.

position both estimators yielded the same response, while small differences arose in the estimated 3D-CoM velocities. Table I summarizes the RMSE of all estimated quantities for both filters. In this static, low pace gait, Valkyrie experienced mostly co-planar contacts, thus the proposed CoM estimator yielded very accurate estimates. Nevertheless, we expect ME to provide more accurate estimates in the general case where the robot exhibits non co-planar contacts, but at the cost of employing 6D-F/T sensors at the robot’s end-effectors.

TABLE I: RMSE of Estimated Quantities.

	$c_x(m)$	$c_y(m)$	$c_z(m)$	$\dot{c}_x(\frac{m}{s})$	$\dot{c}_y(\frac{m}{s})$	$\dot{c}_z(\frac{m}{s})$
EKF1	0.0036	0.0037	0.0011	0.0179	0.0174	0.0123
ME	0.0036	0.0037	0.0011	0.0205	0.0212	0.0099

B. Real Robot Experiments

The proposed cascade framework was further implemented on a NAO v4.0 humanoid robot, running in real-time every 10ms. The joint encoder, IMU and FSRs measurements needed by the scheme are available at a 100Hz rate. For obtaining the pose, we used SVO with a ZED stereo camera, running on an Nvidia Jetson TX2 module, communicating with NAO through ethernet with a TCP/IP server. The latter was available to NAO at an average rate of 40Hz. The estimation parameters used can be found in the NAO configuration file in [17].

A first result regards estimation of the external forces, where the robot was disturbed and the pushes were accurately measured with an Alluris force gauge. The NAO robot was

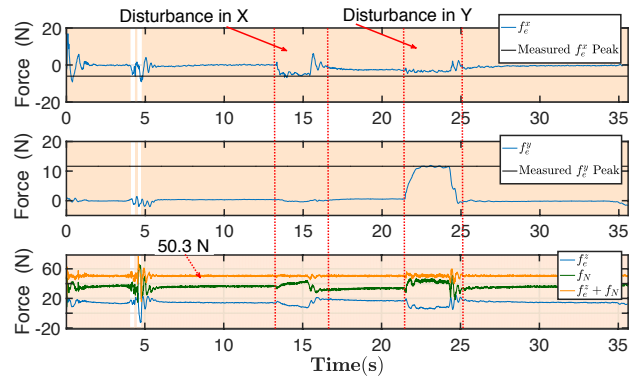


Fig. 7: 3D-External forces, light beige regions correspond to the DS phases, blue lines indicate the estimated external forces by the proposed CoM estimator, orange is the vertical resultant force $f_e^z + f_N$, green is the FSR measured vertical GRF f_N , and black are the measured external force peaks.

commanded to stand up, initialize its posture by making two steps, and then stand still. As we can see in Figure 7, from 6s to 13s where the NAO robot is unperturbed in the z -axis the external force counter balance the false measurement from the FSR for the total weight so that the resultant force ($f_N + f_e^z$) yields the mass of the robot which is approximately 5.19kg. Moreover, a disturbance in the x -axis is performed at 13s and settles at 16.6s. This disturbance was measured to have a peak magnitude of 5.96N, as also estimated by f_e^x . Finally, a constant lateral disturbance was enforced at 21.4s until 25.3s with peak at 11.64N making NAO tilt; again our estimator yielded an accurate estimate f_e^y .

To fair contrast the proposed cascade scheme, we constructed another serial state estimation scheme, based on the rigid body estimator of [7] and EKF2; for simplicity this is termed EKF2 in the sequel. Since a LIDAR sensor was not available, we used the camera pose for the global body position and orientation measurement. Furthermore, to remove the correlation explained in section III-B, we computed the body velocity in the world frame using kinematics.

In addition, the transformation of the support foot with respect to the world frame was computed using the kinematic relative transformation from the body to the support and the estimated body to world transformation, since it cannot be estimated directly, as in our approach.

Subsequently, we let our robot walk outdoors on a challenging inclined terrain where the slope in the forward and in the lateral directions was 16° and 5°, respectively. To accurately measure the final pose, since in outdoor environments ground truth data are not available, we used both conventional measuring tools and digital laser rangefinders to measure the final position and orientation (termed as Ground-Truth) at the end of the gait.

In order to observe the drift and how drift can affect CoM estimation, we ordered NAO to walk straight up the inclined road. Figure 8 (top) shows the 2D body pose as estimated by the employed schemes and as computed using the kinematics. Both estimators yielded pretty similar results

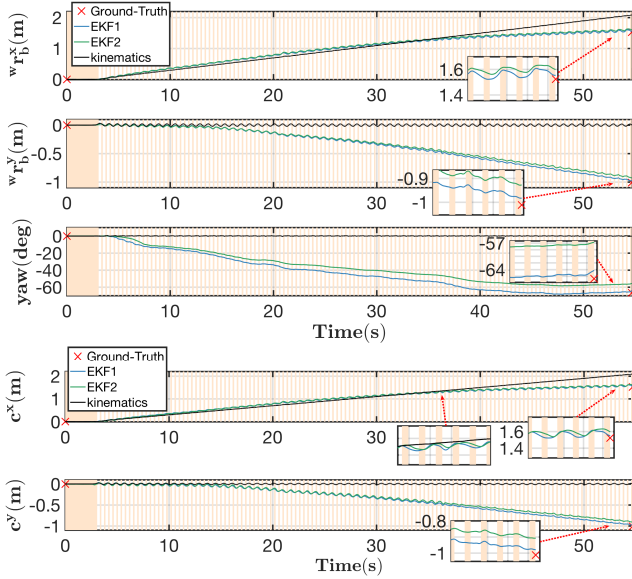


Fig. 8: **Top** 2D-Body pose trajectory, **Bottom** 2D-CoM trajectory, light beige regions indicate the DS phases, \times the ground truth positions, blue lines and green lines are the estimated trajectories with EKF1 and EKF2 respectively, and black lines are the kinematically computed trajectories.

while walking straight where the drift was negligible. Small differences arise from the fact that in the proposed cascade scheme the support foot dynamics also work as constraints for respecting the robot kinematic chains. Nevertheless, when the robot started to drift, EKF2 accuracy started to degrade, since the kinematically computed body velocity in the world frame is fused. EKF2 final pose error was 7.56cm and 8.43cm in the x and y -axes and 10.06° for the body yaw angle, while for the EKF1 it was 3.06cm and 2.88cm in the x and y -axes and 2.81° in the yaw angle. Notice, the kinematically computed odometry was completely off, since it shows that the robot had actually performed a straight gait. Accordingly, Figure 8 (bottom) shows that the same degradation in accuracy is inevitably inherited in the CoM estimation, demonstrating one more time that accurate rigid body estimation is vital to CoM estimation. In addition, we note that EKF1 yields a more oscillatory response, which is expected since when walking over inclined terrains early swing leg landing commonly occurs causing the robot to rotate and producing angular momentum, which is not considered in EKF2. All the presented experiments can be visualized in high resolution at <https://goo.gl/7kbcuf>.

V. CONCLUSIONS

In this paper we proposed a novel cascade estimation scheme that fuses IMU, joint encoders, FSR and visual input to provide with accurate estimates of important quantities in humanoid planning and control.

After implementing the proposed scheme both in simulation and on a real NAO robot, we demonstrated its accuracy, robustness to disturbances and efficacy in realistic scenarios. Given that the proposed cascade scheme is based on generic

dynamics, it is readily amenable to generalization to other humanoids. To this end, we released SEROW [17], an open-source ROS package to reinforce robotic research endeavors.

In future work we aim at investigating whether considering the swing foot dynamics improves the estimation accuracy. In addition, we plan to extend our planning and control schemes, to allow for more dynamic and agile motions effectively utilizing the accurate estimates.

REFERENCES

- [1] S. Kajita et al., “Biped Walking Pattern Generation by using Preview Control of Zero-Moment Point,” in *IEEE Intl. Conf. on Robotics and Automation (ICRA)*, 2003, pp. 1620–1626.
- [2] S. Piperakis, E. Orfanoudakis, and M. Lagoudakis, “Predictive Control for Dynamic Locomotion of Real Humanoid Robots,” in *IEEE/RSJ Intl. Conf. Intel. Robots and Systems*, 2014, pp. 4036–4043.
- [3] J. Pratt, J. Carff, S. Drakunov, and A. Goswami, “Capture Point: A Step toward Humanoid Push Recovery,” in *IEEE-RAS Intl. Conf. on Humanoid Robots*, 2006, pp. 200–207.
- [4] T. Koolen et al., “Design of a momentum-based control framework and application to the humanoid robot atlas,” *Intl. Journal of Humanoid Robotics*, vol. 13, p. 1650007, 2016.
- [5] T. Takenaka, T. Matsumoto, and T. Yoshiike, “Real time motion generation and control for biped robot -1st report: Walking gait pattern generation-,” in *IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems*, 2009, pp. 1084–1091.
- [6] J. Engelsberger et al., “Trajectory generation for continuous leg forces during double support and heel-to-toe shift based on divergent component of motion,” in *IEEE/RSJ Conf. Intel. Robots and Systems*, 2014.
- [7] S. Kuindersma, R. Deits, M. Fallon, A. Valenzuela, H. Dai, F. Permenter, T. Koolen, P. Marion, and R. Tedrake, “Optimization-based Locomotion Planning, Estimation, and Control Design for the Atlas Humanoid Robot,” *Autonomous Robots*, vol. 40, pp. 429–455, 2016.
- [8] M. F. Fallon et al., “Continuous Humanoid Locomotion over Uneven Terrain using Stereo Fusion,” in *IEEE-RAS Intl. Conf. on Humanoid Robots*, 2015, pp. 881–888.
- [9] M. Bloesch et al., “State Estimation for Legged Robots—Consistent Fusion of Leg Kinematics and IMU,” *Robotics Sci. and Sys.*, 2012.
- [10] N. Rotella, M. Bloesch, L. Righetti, and S. Schaal, “State Estimation for a Humanoid Robot,” in *IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems*, 2014, pp. 952–958.
- [11] J. Carpintieri, 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*, vol. 32, 2016.
- [12] N. Rotella, A. Herzog, S. Schaal, and L. Righetti, “Humanoid momentum estimation using sensed contact wrenches,” in *IEEE-RAS Intl. Conf. on Humanoid Robots*, 2015, pp. 556–63.
- [13] B. J. Stephens, “State Estimation for Force-controlled Humanoid Balance using Simple Models in the Presence of Modeling Error,” in *IEEE Intl. Conf. on Robotics and Automation*, 2011, pp. 3994–3999.
- [14] Xinjilefu and C. G. Atkeson, “State Estimation of a Walking Humanoid Robot,” in *IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems*, 2012, pp. 3693–3699.
- [15] S. Piperakis and P. Trahanias, “Non-linear ZMP based State Estimation for Humanoid Robot Locomotion,” in *IEEE-RAS Intl. Conf. on Humanoid Robots*, 2016, pp. 202–209.
- [16] R. Martinez-Cantin and J. A. Castellanos, “Bounding Uncertainty in EKF-SLAM: the Robocentric Local Approach,” in *Intl. Conf. on Robotics and Automation*, 2006, pp. 430–435.
- [17] S. Piperakis, “SEROW Humanoid Walking Estimator.” [Online]. Available: <https://github.com/mrsp/serow>
- [18] X. Xinjilefu, S. Feng, and C. G. Atkeson, “Center of Mass Estimator for Humanoids and its Application in Modelling Error Compensation, Fall Detection and Prevention,” in *IEEE-RAS Intl. Conf. on Humanoid Robots*, 2015, pp. 67–73.
- [19] D. Vecchio and R. Murray, “Observability and Local Observer Construction for Unknown Parameters in Linearly and Nonlinearly Parameterized Systems,” in *Amer. Control Conf.*, 2003, pp. 4748–53.
- [20] C. Forster, M. Pizzoli, and D. Scaramuzza, “SVO: Fast Semi-direct Monocular Visual Odometry,” in *IEEE Intl. Conf. on Robotics and Automation*, 2014, pp. 15–22.