

# On the Use of Torque Measurement in Centroidal State Estimation

Shahram Khorshidi<sup>1,5</sup>, Ahmad Gazar<sup>1</sup>, Nicholas Rotella<sup>2</sup>, Maximilien Naveau<sup>3</sup>,  
Ludovic Righetti<sup>1,4</sup>, Maren Bennewitz<sup>5</sup>, Majid Khadiv<sup>1</sup>

**Abstract**—State-of-the-art legged robots are either capable of measuring torque at the output of their drive systems, or have transparent drive systems which enable the computation of joint torques from motor currents. In either case, this sensor modality is seldom used in state estimation. In this paper, we propose to use joint torque measurements to estimate the centroidal states of legged robots. To do so, we project the whole-body dynamics of a legged robot into the nullspace of the contact constraints, allowing expression of the dynamics independent of the contact forces. Using the constrained dynamics and the centroidal momentum matrix, we are able to directly relate joint torques and centroidal states dynamics. Using the resulting model as the process model of an Extended Kalman Filter (EKF), we fuse the torque measurement in the centroidal state estimation problem. Through real-world experiments on a quadruped robot executing different gaits, we demonstrate that the estimated centroidal states from our torque-based EKF drastically improve the recovery of these quantities compared to direct computation.

## I. INTRODUCTION

State estimation for legged robots plays a crucial role in the successful application of modern state feedback controllers to this domain. The estimation problem is especially difficult for legged robots, as they are inherently underactuated and experience uncertain, intermittent contacts with the environment during motion. Furthermore, their dynamics are highly nonlinear, which makes the design of estimators nontrivial.

The majority of legged robot estimation works to date have focused on base state estimation. This problem is particularly difficult for floating base systems, as the base pose cannot be measured directly. One of the most widely used choices for base state estimation is the Extended Kalman Filter (EKF) [1], [2], [3]. While there have been recent attempts to use more advanced approaches based on the use of factor graphs [4] and invariant Kalman Filters [5], [6], the EKF framework is commonly used due to its compromise between simplicity, efficiency, and performance [7], [8].

For base state estimation, most works have fused Inertial Measurement Unit (IMU) data with leg odometry to estimate the base position and velocity as well as its orientation [1],

[2]. In order to compensate for the drift of the unobservable base position while simultaneously mapping the environment, sensor modalities like LiDAR and cameras are often added [3], [7], [9].

Force measurement at the endeffectors of legged robots is another useful source of information that can be used to better estimate [10], [11] and handle contact switch events [12], [13] as well as aid in estimation of the centroidal states [14], [4]. However, since it is highly important for legged robots to have low leg inertia for agile movements, using sensorized feet [13] can drastically degrade the range of dynamic movements they can perform, especially for quadrupeds. Thus, most of works in this area first estimate the contact forces without direct sensing, and then use the estimated forces for state estimation [12], [4].

While legged robots often lack endeffector force sensing, they are commonly equipped with joint torque sensors. In this work, we propose to fuse joint torque measurements with the nonlinear dynamic model in the EKF framework to estimate the centroidal states – namely, the center of mass position and momenta – of a legged robot. We argue that since most state-of-the-art quadrupedal [15], [16], [17] and bipedal [18], [19], [20] platforms have the capability to measure or estimate joint torque, this sensing modality can be used to perform centroidal state estimation regardless of contact force measurement capabilities. Model-based centroidal state computations are usually very noisy due to noise in measured joint velocities, and low-pass filtering can introduce delays on the order of tens of milliseconds; such large delays can destabilize high rate controllers [14]. By using joint torque measurements for centroidal estimation, we can avoid this issue without necessitating contact force sensing. To the best of our knowledge, this is the first time that this sensing modality is used to estimate the centroidal states of a legged robot. Notably, the proposed framework is general enough to be used on any legged robot, not just a quadruped.

The main contributions of this work are as follows:

- We propose an EKF-based framework which fuses the whole body dynamics of a legged robot projected into the nullspace of the contact constraints with joint torque measurements in order to perform centroidal state estimation.
- We demonstrate through an extensive set of experiments with different gaits on the real quadruped robot Solo12 [17], that the use of torque measurements provides accurate centroidal state estimation with minimal noise and time delay.

<sup>1</sup>Max-Planck Institute for Intelligent Systems, Tübingen, Germany. {firstname.lastname}@tuebingen.mpg.de

<sup>2</sup>Agility Robotics, Oregon, USA. nicholas.rotella@gmail.com

<sup>3</sup>LAAS-CNRS, Toulouse. mnaveau@laas.fr

<sup>4</sup>Tandon School of Engineering, New York University, USA. ludovic.righetti@nyu.edu

<sup>5</sup>Humanoid Robots Lab, University of Bonn, Germany. maren@cs.uni-bonn.de, khorshidi@cs.uni-bonn.de

This work was supported by the Max-Planck Institute for Intelligent Systems' Grassroots program (M10338 and M10343) and the National Science Foundation (CMMI-1825993).

## II. FUNDAMENTALS

### A. Notation

- The  $\oplus$  operator denotes the proper vector composition of  $SE(3) \times \mathbb{R}^n$ .
- $\nabla(\cdot)$  is the derivative operator with respect to (w.r.t.) corresponding variable, and  $I$  is the identity matrix with the proper dimension.
- Throughout the paper, we use small letters to specify scalars and scalar-valued functions, bold small letters for vectors and vector-valued functions, and capital letters for matrices.

### B. Floating-Base Dynamics

The dynamics of a floating-base system can be written as

$$M(\mathbf{q})\dot{\mathbf{v}} + \mathbf{n}(\mathbf{q}, \mathbf{v}) = B\boldsymbol{\tau} + J_c^\top \boldsymbol{\lambda}, \quad (1)$$

where  $M \in \mathbb{R}^{(n+6) \times (n+6)}$  is the mass-inertia matrix,  $\mathbf{q} \in SE(3) \times \mathbb{R}^n$  denotes the configuration space,  $\mathbf{v} \in \mathbb{R}^{n+6}$  encodes the vector of generalized velocities (or more precisely quasi-velocities [21] §7.6), and  $\mathbf{n} \in \mathbb{R}^{n+6}$  is a concatenation of nonlinear terms including centrifugal, Coriolis and gravitational effects.  $B \in \mathbb{R}^{(n+6) \times n}$  is a selection matrix that separates the actuated and unactuated Degrees of Freedom (DoFs),  $\boldsymbol{\tau} \in \mathbb{R}^n$  is the vector of actuated joint torques,  $J_c \in \mathbb{R}^{3m \times (n+6)}$  is the Jacobian of  $m$  feet in contact, and finally  $\boldsymbol{\lambda} \in \mathbb{R}^{3m}$  is the vector of contact forces (here we assume point-contact feet for quadrupeds, however all results hold for humanoids with flat feet as well).

### C. Motion and Constraint Dynamics

Assuming rigid non-slipping contact with the environment, and using the orthogonal projection operator  $N = I - J_c^\dagger J_c$  (where  $\dagger$  is the Moore-Penrose inverse) which projects into the nullspace of the contact Jacobian, we can divide (1) into the following set of equations [22]

$$NM(\mathbf{q})\dot{\mathbf{v}} + N\mathbf{n}(\mathbf{q}, \mathbf{v}) = NB\boldsymbol{\tau}, \quad (2a)$$

$$(I - N)(M(\mathbf{q})\dot{\mathbf{v}} + \mathbf{n}(\mathbf{q}, \mathbf{v})) = (I - N)B\boldsymbol{\tau} + J_c^\top \boldsymbol{\lambda}. \quad (2b)$$

Equation (2a) encodes the dynamics of the robot motion independent of the constraint forces, while (2b) yields the dynamics of the system in the contact constraint space [22]. In other words,  $NB\boldsymbol{\tau}$  is used to move the system on a desired trajectory without violating the contact constraint, while  $(I - N)B\boldsymbol{\tau}$  preserves the stationary contact. Interestingly, since (2a) is independent of contact forces, we can use it to relate the measured joint torques to the joint accelerations without knowledge of the contact forces. This is especially useful for agile legged robots whose endeffectors are often not equipped with contact force/torque sensors.

Since the feet in contact are assumed stationary, we can write down the following constraint and differentiate both sides w.r.t. time, yielding

$$(I - N)\mathbf{v} = \mathbf{0} \Rightarrow (I - N)\dot{\mathbf{v}} = \dot{N}\mathbf{v}. \quad (3)$$

Defining the constraint-consistent mass-inertia matrix as  $M_c = NM + I - N$  and substituting it along with (3) into (2a), we have

$$M_c(\mathbf{q})\dot{\mathbf{v}} - \dot{N}\mathbf{v} + N\mathbf{n}(\mathbf{q}, \mathbf{v}) = NB\boldsymbol{\tau}. \quad (4)$$

Now, we can invert  $M_c$  and solve the constraint-consistent equations of motion for  $\dot{\mathbf{v}}$  to yield

$$\dot{\mathbf{v}} = M_c^{-1}(\dot{N}\mathbf{v} - N\mathbf{n} + NB\boldsymbol{\tau}). \quad (5)$$

### D. Relation between Joint Torques and Centroidal States

Denoting the centroidal momentum vector by  $\mathbf{h}_G \in \mathbb{R}^6$ , we can write down its relation with generalized velocities using

$$\mathbf{h}_G = [\mathbf{l}_G, \mathbf{k}_G]^\top = A_G(\mathbf{q})\mathbf{v}, \quad (6)$$

where  $A_G(\mathbf{q})$  is the Centroidal Momentum Matrix [23]. Taking the derivative of (6) w.r.t. time yields

$$\dot{\mathbf{h}}_G = A_G(\mathbf{q})\dot{\mathbf{v}} + \dot{A}_G(\mathbf{q})\mathbf{v}. \quad (7)$$

Substituting (5) into (7), we have

$$\dot{\mathbf{h}}_G = A_G(\mathbf{q})(M_c^{-1}(\dot{N}\mathbf{v} - N\mathbf{n} + NB\boldsymbol{\tau})) + \dot{A}_G(\mathbf{q})\mathbf{v}. \quad (8)$$

Assuming that  $\mathbf{q}, \mathbf{v}$  can be measured using joint encoders, and the base state is estimated separately using a base estimator, we can write down the momentum dynamics in the following compact form:

$$\dot{\mathbf{h}}_G = D(\mathbf{q}, \mathbf{v})\boldsymbol{\tau} + \mathbf{b}(\mathbf{q}, \mathbf{v}), \quad (9)$$

where

$$D(\mathbf{q}, \mathbf{v}) \triangleq A_G(\mathbf{q})M_c^{-1}NB, \quad (10a)$$

$$\mathbf{b}(\mathbf{q}, \mathbf{v}) \triangleq A_G(\mathbf{q})M_c^{-1}(\dot{N}\mathbf{v} - N\mathbf{n}) + \dot{A}_G(\mathbf{q})\mathbf{v}. \quad (10b)$$

## III. CENTROIDAL STATE ESTIMATION

We aim to estimate the centroidal states  $\mathbf{x} \in \mathbb{R}^9 = [\mathbf{c}_G, \mathbf{l}_G, \mathbf{k}_G]^\top$  using joint torque measurements and the above centroidal momentum dynamics. The process model is

$$\begin{aligned} \dot{\mathbf{c}}_G &= \frac{1}{m}\mathbf{l}_G, \\ [\dot{\mathbf{l}}_G, \dot{\mathbf{k}}_G]^\top &= D\hat{\boldsymbol{\tau}} + \mathbf{b}, \end{aligned} \quad (11)$$

where  $\hat{\boldsymbol{\tau}}$  is the measured joint torques,  $m$  is the total mass of the robot,  $\mathbf{c}_G \in \mathbb{R}^3$  is the center of mass (CoM) position, and  $\mathbf{l}_G \in \mathbb{R}^3$  and  $\mathbf{k}_G \in \mathbb{R}^3$  are the linear and angular components of centroidal momentum respectively. These values are computed using the results of base state estimation, measured generalized joint states, kinematics, and inertial properties of the robot. Specifically, we use (6) to compute the momenta while the CoM position is computed using the estimated base state and joint measurements as

$$\mathbf{c}_G = \mathbf{g}(\mathbf{q}). \quad (12)$$

The measurement model is

$$\begin{aligned} \mathbf{y} &: (SE(3) \times \mathbb{R}^n, \mathbb{R}^{n+6}) \mapsto \mathbb{R}^9, \\ \mathbf{y}(\mathbf{q}, \mathbf{v}) &= [\mathbf{g}(\mathbf{q}), \mathbf{h}_G]^\top. \end{aligned} \quad (13)$$

As shown in [14], although the centroidal states can be computed directly, they are subject to considerable noise and modeling errors. By fusing the computed states with a torque-based process model, we similarly obtain low-noise estimates with minimal delay which is suitable for high-bandwidth control.

**Remark 1:** It is important to note that the right hand side of the process models (11) is an implicit function of  $c_G$ ,  $l_G$ , and  $k_G$ . Hence, we need to compute its Jacobian to be used inside the EKF formulation. We show in Sec. III-C how this Jacobian can be computed. One simpler approach than computing these Jacobians and using EKF would be to treat (11) as a time-varying linear model where  $D$  and  $\mathbf{b}$  are evaluated at each configuration. This would be a special case of the more general EKF framework we are proposing in this paper.

**Remark 2:** For both base and centroidal state estimations, a contact detection mechanism for each foot is employed. One might ask since we estimate the contact forces for contact detection, then why not use a centroidal state estimator based on estimated contact forces (for instance as in [14]) which would render the process model much simpler compared to (11)? The answer to this question is threefold: 1) contact detection is normally performed by setting a threshold only on the *estimated normal force* separately for each leg neglecting the dynamics. This force measurement is not accurate enough to be used in the centroidal state estimation problem. 2) Instead of using the whole-body dynamics to estimate the contact forces and then use it to estimate the centroidal states, our approach estimates those quantities directly from torque measurements. 3) Many legged robots are endowed with binary contact sensors at their endeffectors [17], [18] that can directly detect contact events, hence there is no need to estimate endeffector forces.

#### A. Extended Kalman Filtering

We choose to implement our estimator as an Extended Kalman Filter (EKF). The EKF estimates the mean  $\boldsymbol{\mu}$  and the error covariance matrix  $P$  over the state  $\mathbf{x}_k$ , where both process and measurement models are corrupted by additive zero-mean Gaussian noise. The discrete-time stochastic system evolves as:

$$\mathbf{x}_k = \mathbf{f}(\mathbf{x}_{k-1}, \mathbf{u}_k) + \mathbf{w}_k, \quad (14a)$$

$$\mathbf{z}_k = \mathbf{h}(\mathbf{x}_k) + \mathbf{v}_k, \quad (14b)$$

where the input  $\mathbf{u}_k = \boldsymbol{\tau}_k$  is the vector of joint torques and  $\mathbf{z}_k$  is the measurement. The process noise  $\mathbf{w}_k \sim \mathcal{N}(\mathbf{0}, Q_k)$  and measurement noise  $\mathbf{v}_k \sim \mathcal{N}(\mathbf{0}, R_k)$  are parameterized by the corresponding noise covariance matrices  $Q_k$  and  $R_k$ . These matrices constitute the main tuning parameters of the filter. Note that the process noise is chosen to be purely additive (rather than based on the uncertainty in torque sensing) for the sake of simplicity in filter implementation. In other words, we represent the effects of joint torque measurements and dynamic model errors as additive noise in the process and measurement models respectively.

In the prediction step, having determined the set of feet in contact using estimated contact forces, the mean is propagated using the discrete-time nonlinear process model, and the process model Jacobian  $F_k = \left. \frac{\partial \mathbf{f}(\mathbf{x})}{\partial \mathbf{x}} \right|_{\mathbf{x}=\boldsymbol{\mu}_k^+}$  is used to propagate the error covariance matrix as

$$\boldsymbol{\mu}_k^- = \mathbf{f}(\boldsymbol{\mu}_{k-1}^+, \mathbf{u}_k), \quad (15a)$$

$$P_k^- = (F_{k-1} P_{k-1}^+ F_{k-1}^T) + Q_k, \quad (15b)$$

where the minus superscript denotes the *a priori* (before measurement update) estimate, and the plus superscript denotes the *a posteriori* (after measurement update).

In the update step, the measurements  $\mathbf{z}_k$  are integrated into the EKF using the *a priori* estimates and the measurement noise covariance as follows:

$$K_k = P_k^- H_k^T (H_k P_k^- H_k^T + R_k)^{-1}, \quad (16a)$$

$$\boldsymbol{\mu}_k^+ = \boldsymbol{\mu}_k^- + K_k (\mathbf{z}_k - \mathbf{h}(\boldsymbol{\mu}_k^-)), \quad (16b)$$

$$P_k^+ = (I - K_k H_k) P_k^-, \quad (16c)$$

where  $K_k$  is the Kalman gain, and  $H_k = \left. \frac{\partial \mathbf{h}(\mathbf{x})}{\partial \mathbf{x}} \right|_{\mathbf{x}=\boldsymbol{\mu}_k^-}$  is the measurement Jacobian matrix.

#### B. Discrete-Time Nonlinear Process Model

In the prediction step, the mean of the state is propagated using the discretized nonlinear process model. We discretize (11) using a first-order explicit Euler integration as follows:

$$\boldsymbol{\mu}_k = \begin{bmatrix} c_{G[k-1]} \\ l_{G[k-1]} \\ \mathbf{k}_{G[k-1]} \end{bmatrix} + \begin{bmatrix} \frac{1}{m} \mathbf{l}_{G[k-1]} \Delta t \\ [D \hat{\boldsymbol{\tau}}_k + \mathbf{b}] \Delta t \end{bmatrix}, \quad (17)$$

where  $\Delta t$  is the discretization time step, and  $\hat{\boldsymbol{\tau}}_k$  are the measured joint torques.

#### C. Discrete-Time Linearized Process Model

In order to propagate the state covariance in the prediction step (15b), we first linearize then discretize the dynamics at the filtering frequency. Since the nonlinear dynamic model of the momentum states  $[l_G, \mathbf{k}_G]^T$  is expressed implicitly as a function of the centroidal states, we resort to finite differencing for the computation of the continuous-time process model Jacobian  $F_c$  in (9).

To compute the partial derivative of  $\dot{\mathbf{h}}_G$  w.r.t.  $c_x$ , we first find the deviation of  $\delta \mathbf{q}$  corresponding to the variation of CoM position in the  $x$ -direction. By considering the first three rows (linear momentum part) of  $A_G(\mathbf{q})$  as  $A'_G(\mathbf{q})$ , we can write down the variation of the CoM position as a function of variation of generalized coordinates as

$$\delta c_G = \frac{1}{m} A'_G(\mathbf{q}) \delta \mathbf{q}. \quad (18)$$

Finding the variation of  $\mathbf{q}$  only in the  $x$ -direction of the CoM position simply becomes

$$\delta \mathbf{q} = m A_G^{\dagger}(\mathbf{q}) [\delta c_x, 0, 0]^T. \quad (19)$$

Using  $\mathbf{q}^+ = \mathbf{q} \oplus \delta \mathbf{q}$ , we can find the part of the Jacobian matrix corresponding to  $c_x$  as



Fig. 1: Snapshots of the first motion scenario (moving the base while balancing, with all feet in contact).

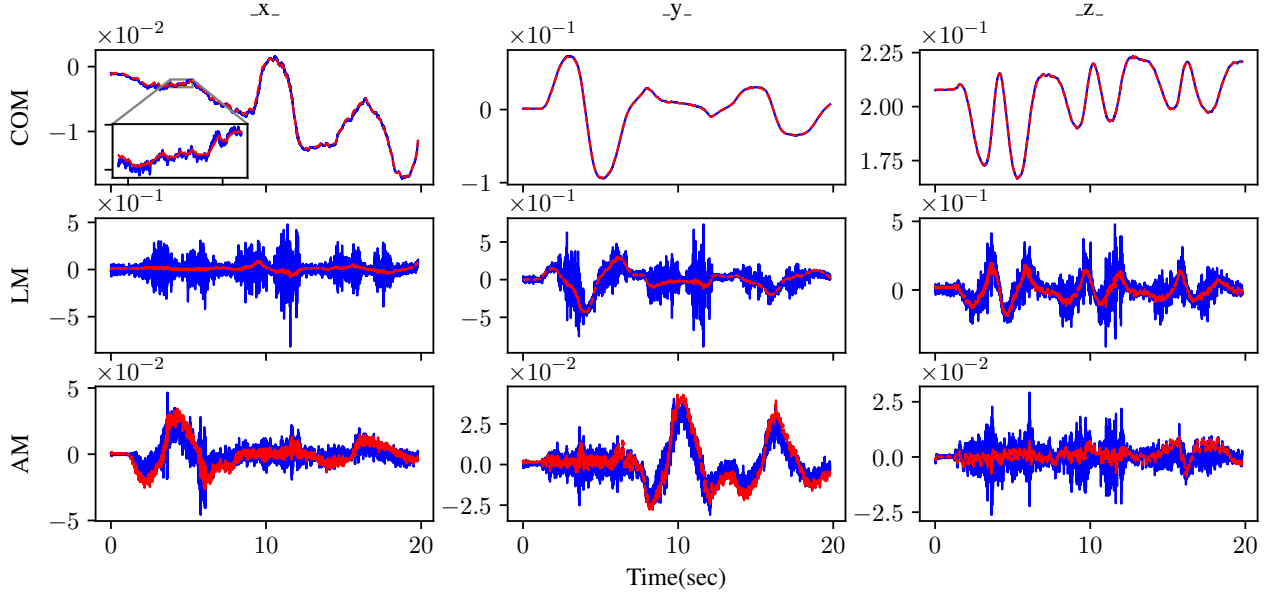


Fig. 2: Motion #1: Moving the base while balancing. Estimation of the CoM position ( $m$ ) (top row), linear momentum ( $\frac{kgm}{s}$ ) (middle row) and angular momentum ( $\frac{kgm^2}{s}$ ) (bottom row); computed centroidal states in blue, and EKF in red.

$$\Delta \dot{\mathbf{h}}_G = \dot{\mathbf{h}}_G(\mathbf{q}^+, \mathbf{v}, \hat{\boldsymbol{\tau}}) - \dot{\mathbf{h}}_G(\mathbf{q}, \mathbf{v}, \hat{\boldsymbol{\tau}}), \quad (20a)$$

$$\Delta \mathbf{c}_G = \mathbf{g}(\mathbf{q}^+) - \mathbf{g}(\mathbf{q}), \quad (20b)$$

$$\frac{\partial \dot{\mathbf{h}}_G}{\partial c_x} = \frac{\Delta \dot{\mathbf{h}}_G}{\Delta c_x}, \quad (20c)$$

where  $\Delta c_x \in \mathbb{R}$  is the  $x$  component of (20b). Following the same procedure above, we compute the derivative w.r.t.  $c_y$  and  $c_z$  by considering small variation as  $[0, \delta c_y, 0]^\top$  and  $[0, 0, \delta c_z]^\top$  respectively.

Similarly, to find the derivative of  $\dot{\mathbf{h}}_G$  w.r.t. linear momentum  $l_x$ , we compute the deviation of  $\delta \mathbf{v}$  corresponding to the variation of linear momentum only in the  $x$ -direction. To do so, we rewrite (7) as

$$\delta \mathbf{h}_G = A_G(\mathbf{q})\delta \mathbf{v} + \delta A_G(\mathbf{q})\mathbf{v}. \quad (21)$$

By considering  $A_G(\mathbf{q})$  constant for small variations (i.e.  $\delta A_G(\mathbf{q}) = 0$ ), we can find a small variation of  $\mathbf{v}$  corresponding to the small variation of linear momentum in the  $x$ -direction being

$$\delta \mathbf{v} = A_G^\dagger(\mathbf{q}) [\delta l_x, 0, 0, 0, 0, 0]^\top. \quad (22)$$

Given  $\mathbf{v}^+ = \mathbf{v} + \delta \mathbf{v}$ , we compute the part of Jacobian matrix corresponding to  $l_x$  as

$$\Delta \dot{\mathbf{h}}_G = \dot{\mathbf{h}}_G(\mathbf{q}, \mathbf{v}^+, \hat{\boldsymbol{\tau}}) - \dot{\mathbf{h}}_G(\mathbf{q}, \mathbf{v}, \hat{\boldsymbol{\tau}}), \quad (23a)$$

$$\Delta \mathbf{l}_G = \mathbf{l}_G(\mathbf{q}, \mathbf{v}^+) - \mathbf{l}_G(\mathbf{q}, \mathbf{v}), \quad (23b)$$

$$\frac{\partial \dot{\mathbf{h}}_G}{\partial l_x} = \frac{\Delta \dot{\mathbf{h}}_G}{\Delta l_x}, \quad (23c)$$

where  $\Delta l_x \in \mathbb{R}$  is the  $x$  component of (23b). Following the same procedure above, we compute the rest of the remaining partial derivatives of the linear and angular momenta w.r.t.  $l_y, l_z, k_x, k_y,$  and  $k_z$ . Finally, we assemble the continuous-time process model Jacobian matrix  $F_c \in \mathbb{R}^{9 \times 9}$  as

$$F_c = \begin{bmatrix} 0_{3 \times 3} & \frac{1}{m} I_{3 \times 3} & 0_{3 \times 3} \\ \Omega_{6 \times 3} & \Lambda_{6 \times 3} & \Gamma_{6 \times 3} \end{bmatrix}, \quad (24)$$

where  $\Omega \triangleq \left[ \frac{\Delta \dot{\mathbf{h}}_G}{\Delta c_x}, \frac{\Delta \dot{\mathbf{h}}_G}{\Delta c_y}, \frac{\Delta \dot{\mathbf{h}}_G}{\Delta c_z} \right]$ ,  $\Lambda \triangleq \left[ \frac{\Delta \dot{\mathbf{h}}_G}{\Delta l_x}, \frac{\Delta \dot{\mathbf{h}}_G}{\Delta l_y}, \frac{\Delta \dot{\mathbf{h}}_G}{\Delta l_z} \right]$ ,  $\Gamma \triangleq \left[ \frac{\Delta \dot{\mathbf{h}}_G}{\Delta k_x}, \frac{\Delta \dot{\mathbf{h}}_G}{\Delta k_y}, \frac{\Delta \dot{\mathbf{h}}_G}{\Delta k_z} \right]$ . In the discrete-time form, the Jacobian is truncated at its first-order Taylor approximation for simplicity, yielding  $F_k \approx I_{9 \times 9} + F_c \Delta t$ , and the discretized process noise covariance matrix is likewise approximated as  $Q_k \approx F_k Q_c F_k^T \Delta t$ . Similarly, we can find the measurement Jacobian in the update step, by taking corresponding derivatives w.r.t. our state. By following the procedure described for computing prediction Jacobian, and defining proper variations for  $\mathbf{q}$ , and  $\mathbf{v}$  we get  $H_k = I_{9 \times 9}$ , which is consistent with the result of [14]. We highlight that the measurement Jacobian is the identity matrix since the state is directly measured. This also validates the finite difference method proposed in this work for computing the process and measurement Jacobians.

#### D. Observability Analysis

The process model of the proposed estimator is a highly-nonlinear function of the states; in general, we may investigate the observability of a nonlinear estimator by forming the

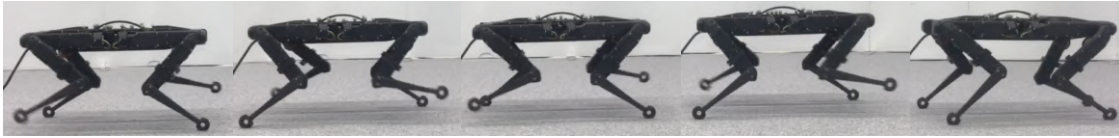


Fig. 3: Snapshots of the second motion scenario (forward trotting).

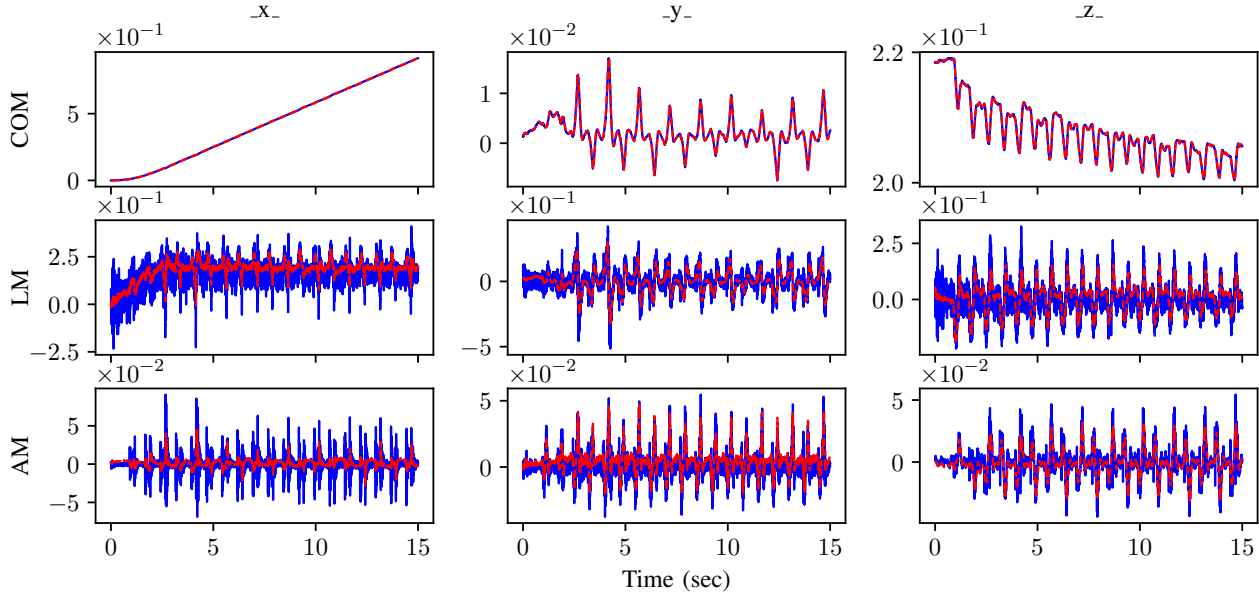


Fig. 4: Motion #2: Forward trotting with contact switching. Estimation of the CoM position ( $m$ ) (top row), linear momentum ( $\frac{kgm}{s}$ ) (middle row) and angular momentum ( $\frac{kgm^2}{s}$ ) (bottom row); computed centroidal states in blue, and EKF in red.

nonlinear observability matrix [24]. The state of the estimator is observable if the observability matrix has full rank. In our case, the process model cannot be expressed explicitly in terms of the states, preventing us from computing Jacobians of  $f(x)$ . However, because we directly measure the full state, we have  $\nabla h(x) = H_k = I_{9 \times 9}$ . This already means that the observability matrix has full column rank, so the state is observable. This is in agreement with the results of [14], where the observability matrix is computed for a similar momentum estimator.

#### IV. RESULTS

In this section, we present the experimental results of performing estimation of the centroidal states both with and without the use of torque measurements, for a variety of different gaits on the quadruped robot Solo12 [17]. Specifically, we consider three different behaviors: moving the base while balancing, trotting, and jumping. The trajectories are generated using the trajectory optimization framework in [25], and the whole-body controller in [17] is used to track these trajectories while satisfying friction cone constraints. The process and measurement models covariance matrices are tuned as  $Q_k = \text{Diag}(10^{-7}, 10^{-7}, 10^{-7}, 10^{-5}, 10^{-5}, 10^{-5}, 10^{-4}, 10^{-4}, 10^{-4})$  and  $R_K = (10^{-5}) * I_9$  respectively. Note that we do not close the control loop around the estimated centroidal states; this is left to future work. Note also that the measured torques are computed by simply multiplying the measured current to the motor constant. While this does not give a

precise torque measurement at the output of the joints due to the unmodeled drive system (e.g. friction, elasticity, etc.), we show in the following experiments that this is not an issue for the estimator. Note that since we cannot measure directly the centroidal states, we compute them using measurements from the motion capture system and joint encoders. Given these base and joint state measurements, we employ (6) and (12) to construct the full centroidal states.

##### A. Motion #1: Moving the Base while Balancing

In the first scenario we consider a wobbling motion of the base without contact switching (Fig. 1). The main goal of this test is to evaluate the quality of the estimated states in isolation from uncertainties introduced by contact switching as well as contact detection. Fig. 2 shows the computed centroidal states in blue, compared against the estimated states from our proposed torque measurement-based EKF in red. As it can be clearly seen in this figure, the use of torque measurements in the proposed EKF significantly filter the centroidal momentum without adding delays.

##### B. Motion #2: Forward Trotting

In the second scenario, we consider a more complex motion involving intermittent contact switching. In this test, the robot moves forward approximately one meter in a trotting gait (Fig. 3). Contact detection is performed by projecting the joint torques of each leg through the Jacobian into the space of endeffector forces, and applying a force

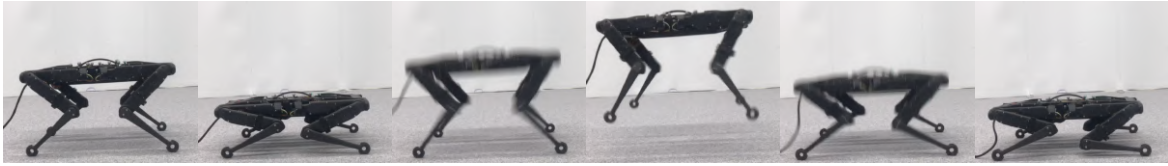


Fig. 5: Snapshots of the third motion scenario (jump in place).

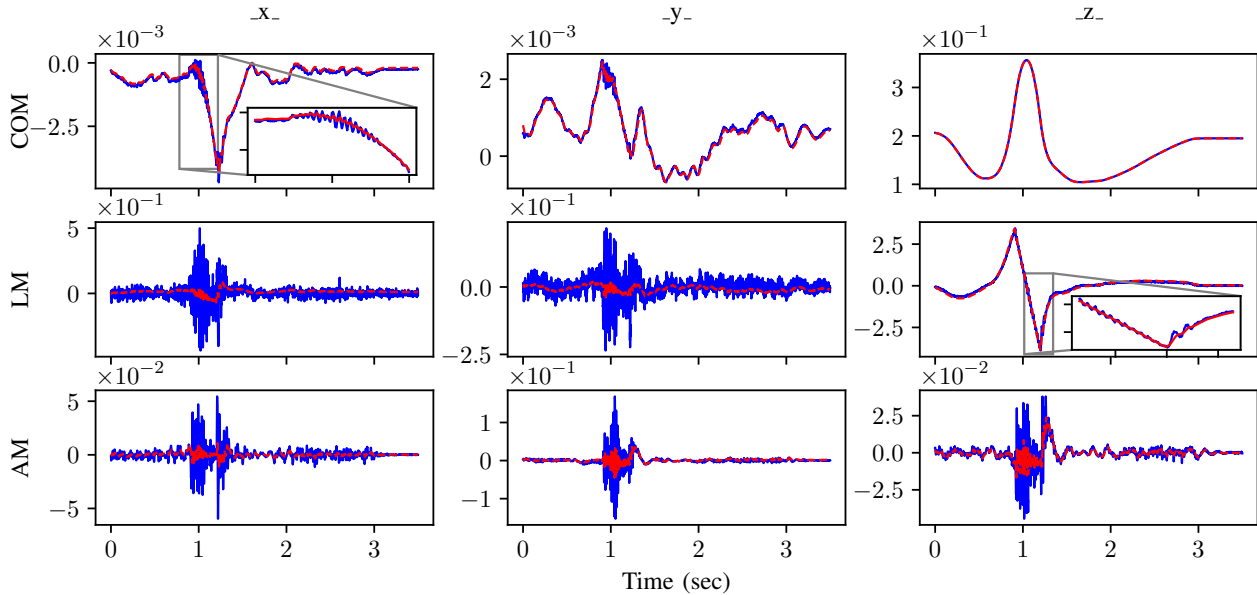


Fig. 6: Motion #3: Jump in place with contact switching. Estimation of the CoM position ( $m$ ) (top row), linear momentum ( $\frac{kgm}{s}$ ) (middle row) and angular momentum ( $\frac{kgm^2}{s}$ ) (bottom row); computed centroidal states in blue, and EKF in red.

threshold to determine whether each foot is in contact. As shown in Fig. 4, in this case there are large discontinuities in the computed states (blue) due to the contact switches; in practice, these anomalies could destabilize a control loop if these states are used for feedback. However, the proposed torque measurement-based EKF (red) reduces the magnitude of these discontinuities, resulting in state estimates which are more suitable to use in a control loop.

### C. Motion #3: Jump in Place

The third motion we consider is a jump in place (Fig. 5). This motion is particularly challenging due to the large impact induced during landing that could be problematic if a direct force/torque sensor at the endeffector is used to estimate the centroidal states as in [14]. However, since we are instead using torque measurements in the proposed estimator, we benefit from the fact that Solo12’s structural and drive system damping filters out the effect of the impact from measured torque considerably. This helps prevent the estimator from diverging in the presence of large impacts. As demonstrated by the results shown in Fig. 6, the proposed estimator is not significantly affected by the impact force during landing, and can thus filter out the measurement noise even in the presence of large impact forces during dynamic maneuvers.

## V. CONCLUSIONS AND FUTURE WORK

In this paper, we proposed to use joint torque measurements for the centroidal momentum state estimation

using an EKF for legged robots. First, we exploited the relationship between the joint torques and centroidal states by projecting the whole-body dynamics of the robot into the nullspace of the contact constraints. Then, we used the resulting dynamics as the process model of an EKF with joint torque input to estimate the centroidal states. One of the key advantages of our approach is that we can estimate the required quantities without relying on Force/Torque sensors as direct measurements. We evaluated the performance of our centroidal momentum estimator for three different gaits on the quadruped robot Solo12. The results of these experiments showed that the estimated states from our approach have considerably less noise when compared to the naive computation of the centroidal states from the measured encoder values and estimated base states while maintaining minimal delay.

In the future, we would like to test the performance of closed-loop momentum control using the estimated centroidal states from the proposed EKF. Furthermore, we would like to mitigate the effects of dynamic model uncertainties by simultaneously estimating the centroidal states and performing online model identification. We are also interested in extending the estimation problem to enable identification of the (non-rigid) contact model parameters, which could be used to adapt the controller when the surface properties are changed.

## REFERENCES

- [1] M. Bloesch, M. Hutter, M. A. Hoepflinger, S. Leutenegger, C. Gehring, C. D. Remy, and R. Siegwart, “State estimation for legged robots-

- consistent fusion of leg kinematics and imu,” *Robotics*, vol. 17, pp. 17–24, 2013.
- [2] N. Rotella, M. Bloesch, L. Righetti, and S. Schaal, “State estimation for a humanoid robot,” in *2014 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 952–958, IEEE, 2014.
- [3] M. F. Fallon, M. Antone, N. Roy, and S. Teller, “Drift-free humanoid state estimation fusing kinematic, inertial and lidar sensing,” in *2014 IEEE-RAS International Conference on Humanoid Robots*, pp. 112–119, IEEE, 2014.
- [4] M. Fourmy, T. Flayols, P.-A. Léziart, N. Mansard, and J. Solà, “Contact forces preintegration for estimation in legged robotics using factor graphs,” in *2021 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 1372–1378, IEEE, 2021.
- [5] R. Hartley, M. Ghaffari, R. M. Eustice, and J. W. Grizzle, “Contact-aided invariant extended kalman filtering for robot state estimation,” *The International Journal of Robotics Research*, vol. 39, no. 4, pp. 402–430, 2020.
- [6] P. Ramadoss, G. Romualdi, S. Dafarra, F. J. A. Chavez, S. Traversaro, and D. Pucci, “Diligent-kio: A proprioceptive base estimator for humanoid robots using extended kalman filtering on matrix lie groups,” in *2021 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 2904–2910, IEEE, 2021.
- [7] M. Camurri, M. Ramezani, S. Nobili, and M. Fallon, “Pronto: A multi-sensor state estimator for legged robots in real-world scenarios,” *Frontiers in Robotics and AI*, vol. 7, p. 68, 2020.
- [8] T.-Y. Lin, R. Zhang, J. Yu, and M. Ghaffari, “Legged robot state estimation using invariant kalman filtering and learned contact events,” in *5th Annual Conference on Robot Learning*, 2021.
- [9] S. Piperakis, D. Kanoulas, N. G. Tsagarakis, and P. Trahanias, “Outlier-robust state estimation for humanoid robots,” in *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 706–713, 2019.
- [10] N. Rotella, S. Schaal, and L. Righetti, “Unsupervised contact learning for humanoid estimation and control,” in *2018 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 411–417, 2018.
- [11] G. Fink and C. Semini, “Proprioceptive sensor fusion for quadruped robot state estimation,” in *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 10914–10920, 2020.
- [12] G. Bledt, P. M. Wensing, S. Ingersoll, and S. Kim, “Contact model fusion for event-based locomotion in unstructured terrains,” in *2018 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 4399–4406, IEEE, 2018.
- [13] G. Valsecchi, R. Grandia, and M. Hutter, “Quadrupedal locomotion on uneven terrain with sensorized feet,” *IEEE Robotics and Automation Letters*, vol. 5, no. 2, pp. 1548–1555, 2020.
- [14] N. Rotella, A. Herzog, S. Schaal, and L. Righetti, “Humanoid momentum estimation using sensed contact wrenches,” in *2015 IEEE-RAS 15th International Conference on Humanoid Robots (Humanoids)*, pp. 556–563, IEEE, 2015.
- [15] M. Hutter, C. Gehring, D. Jud, A. Lauber, C. D. Bellicoso, V. Tsounis, J. Hwangbo, K. Bodie, P. Fankhauser, M. Bloesch, *et al.*, “Anymal-a highly mobile and dynamic quadrupedal robot,” in *2016 IEEE/RSJ international conference on intelligent robots and systems (IROS)*, pp. 38–44, IEEE, 2016.
- [16] B. Katz, J. Di Carlo, and S. Kim, “Mini cheetah: A platform for pushing the limits of dynamic quadruped control,” in *2019 International Conference on Robotics and Automation (ICRA)*, pp. 6295–6301, IEEE, 2019.
- [17] F. Grimmering, A. Meduri, M. Khadiv, J. Viereck, M. Wüthrich, M. Naveau, V. Berenz, S. Heim, F. Widmaier, T. Flayols, *et al.*, “An open torque-controlled modular robot architecture for legged locomotion research,” *IEEE Robotics and Automation Letters*, vol. 5, no. 2, pp. 3650–3657, 2020.
- [18] M. Chignoli, D. Kim, E. Stanger-Jones, and S. Kim, “The mit humanoid robot: Design, motion planning, and control for acrobatic behaviors,” in *2020 IEEE-RAS 20th International Conference on Humanoid Robots (Humanoids)*, pp. 1–8, IEEE, 2021.
- [19] “Agility robotics cassie.” <https://www.agilityrobotics.com/robots/#cassie>.
- [20] E. Daneshmand, M. Khadiv, F. Grimmering, and L. Righetti, “Variable horizon mpc with swing foot dynamics for bipedal walking control,” *IEEE Robotics and Automation Letters*, vol. 6, no. 2, pp. 2349–2356, 2021.
- [21] H. Baruh, *Analytical dynamics*. WCB/McGraw-Hill Boston, 1999.
- [22] M. Mistry and L. Righetti, “Operational space control of constrained and underactuated systems,” *Robotics: Science and systems VII*, pp. 225–232, 2012.
- [23] D. E. Orin and A. Goswami, “Centroidal momentum matrix of a humanoid robot: Structure and properties,” in *2008 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 653–659, IEEE, 2008.
- [24] R. Hermann and A. J. Krener, “Nonlinear controllability and observability,” *Automatic Control, IEEE Transactions on*, vol. 22, pp. 728–740, Oct 1977.
- [25] B. Ponton, M. Khadiv, A. Meduri, and L. Righetti, “Efficient multicontact pattern generation with sequential convex approximations of the centroidal dynamics,” *IEEE Transactions on Robotics*, vol. 37, no. 5, pp. 1661–1679, 2021.