

# Physically Consistent Online Inertial Adaptation for Humanoid Loco-manipulation

James Foster<sup>1</sup>, Stephen McCrory<sup>1,2</sup>, Christian DeBuys<sup>3</sup>, Sylvain Bertrand<sup>2</sup>, and Robert Griffin<sup>1,2,3</sup>

**Abstract**—The ability to accomplish manipulation and locomotion tasks in the presence of significant time-varying external loads is a remarkable skill of humans that has yet to be replicated convincingly by humanoid robots. Such an ability will be a key requirement in the environments we envision deploying our robots: dull, dirty, and dangerous. External loads constitute a large model bias, which is typically unaccounted for. In this work, we enable our humanoid robot to engage in loco-manipulation tasks in the presence of significant model bias due to external loads. We propose an online estimation and control framework involving the combination of a physically consistent extended Kalman filter for inertial parameter estimation coupled to a whole-body controller. We showcase our results both in simulation and in hardware, where weights are mounted on Nadia’s wrist links as a proxy for engaging in tasks where large external loads are applied to the robot.

## I. INTRODUCTION

State-of-the-art model-based controllers rely heavily on an internal dynamic model, usually used inside an optimization routine to repeatedly plan the best sequence of control actions given the current (estimated) state of the robot. However, most current model-based approaches in humanoid locomotion and manipulation utilize a dynamic model that is identified offline but remains unchanged online. This introduces several difficulties. Firstly, there is often significant engineering time spent tuning the dynamic model to compensate for inevitable model mismatch. While one would hope that a good controller is robust to some degree of model mismatch, a second and more practical issue is that a frozen dynamic model restricts the robot to performing manipulation and locomotion tasks where the model is known to some acceptable margin. This often results in manipulation tasks involving objects of trivial mass relative to the robot, and precludes locomotion tasks where any significant load is placed on the robot.

These restrictions stand in stark contrast to humans and other animals, which have been experimentally shown to adapt their motor behaviour in the presence of consistent model bias [1, 2]. For humanoids to engage in meaningful, non-trivial interaction with the environment, they must be capable of identifying changes in their internal dynamic model and updating it for use in downstream control routines. In this paper, we enable identification and adaptation in the presence



Fig. 1: Our humanoid robot Nadia actively estimating and adapting to 12lb dumbbells attached to the forearms. We use these weights as a proxy for inertially significant manipulation tasks, such as heavy tool use.

of large external loads on our humanoid robot Nadia, as seen in Fig. 1, by proposing a framework consisting of an extended Kalman filter (EKF) that streams inertial parameter estimates to a whole-body controller online. The proposed filter is lightweight and general in that it can be easily modified to estimate the parameters of any link in a given rigid body system, removing the dependence on a reference model and thus making it controller-agnostic. Furthermore, the proposed filter builds upon a recently introduced “physically consistent” inertial parameterization [3], guaranteeing that the streamed estimates are physically realizable and precluding catastrophic errors in the controller.

### A. Related Work

There are several excellent recent surveys on inertial identification and model learning. In [4], various concrete least-squares formulations, Kalman filtering approaches, neural network methods, and semidefinite programming approaches

<sup>1</sup>Intelligent Systems & Robotics, University of West Florida, USA

<sup>2</sup>Robotics Lab, Florida Institute for Human & Machine Cognition, USA

<sup>3</sup>Boardwalk Robotics, USA

jfl127@students.uwf.edu,  
{smccrory, sbertrand, rgriffin}@ihmc.org,  
christian.debuys@boardwalkrobotics.com

are compared, whereas [5] provides a more conceptual overview of the exploitable structure of rigid body dynamics and how to couple this structure with data-driven modeling techniques. Finally, [6] covers different metrics, sources of error, and recent advances in geometric methods for parameter identification.

Most inertial parameter estimation works leverage the finding that the inverse dynamics are linear with respect to inertial parameters [7]. Subsequent works have highlighted the additional structure the inertial parameters must obey [8]–[10]: beyond the obvious constraint on positive mass, the moment of inertia matrix must satisfy additional geometric conditions like positive definiteness and the principal moments of inertia must satisfy a triangle inequality. Unfortunately, for multi-link systems like manipulators, particularly for high degree of freedom systems like humanoids, uniquely identifying inertial parameters becomes challenging [11]. A lack of identifiability can be due to the kinematic structure of the robot [12] or from a lack of persistent excitation in the data [13], but methods exist to characterize the identifiable parameters either through numerical [14] or analytical [15] means. Extensive work has gone into methods for generating rich and informative trajectories for robot parameter identification [16] with several applications to humanoids [17]–[19]. Alternatively, several control-oriented works on humanoids [20, 21] have estimated gross robot quantities like center of mass and momentum online.

There are also adaptive control approaches that update parameters based on task error. The seminal work [22] has been extended in many ways: from the fully actuated to the underactuated case [23], modifying the adaptation law to account for the geometric properties of the inertial parameters [24], and embedding the adaptation law as a control-Lyapunov function inequality constraint in a model predictive controller [25]. Recent work in [26] applies L1 adaptive control to quadrupedal load carrying, another direct adaptive control approach with an adaptation law based on the difference in behavior between a reference model and the actual robot. In [27], a centroidal nonlinear MPC scheme on iCub is proposed where the attenuation of an external wrench from a payload is included as a task.

## B. Contributions

From our survey of existing work, we note that most approaches to parameter estimation take place offline in a system identification context, with notable online exceptions for estimating gross robot quantities like center of mass or momentum, which is useful for the locomotion task but less so for other tasks. Online task-error approaches require tight coupling with an existing reference model or controller, reducing their flexibility and complicating the overall design process. In this paper, we contribute:

- A lightweight, controller-agnostic extended Kalman filtering framework that guarantees physically consistent parameter estimates online.
- A compact problem formulation that assumes only a subset of the inertial parameters will be estimated

online, resulting in more efficient rigid body dynamics calculations inside the filter.

- Verification of the proposed filter in simulation, showing the benefit of physical consistency over a baseline Kalman filter, and improvements in control performance when our filter is included in an adaptive control framework versus a controller with no adaptation.
- Demonstrations on hardware of the robot adapting to loads on the forearm links, and an innovation gating method for rejecting undesirable transients in parameter estimates due to contact impulses during locomotion.

## II. BACKGROUND

We assume that the robot can be adequately modelled as a floating base rigid body system, with the equations of motion given by:

$$\mathbf{M}(\mathbf{q})\dot{\mathbf{v}} + \mathbf{c}(\mathbf{q}, \mathbf{v}) = \mathbf{S}^\top \boldsymbol{\tau} + \sum_{i \in \mathcal{C}} \mathbf{J}_i^\top(\mathbf{q}) \boldsymbol{\lambda}_i, \quad (1)$$

where  $\mathbf{q} \in \mathbb{R}^{n_q}$ ,  $\mathbf{v} \in \mathbb{R}^{n_v}$  are the generalized configuration and velocity of the robot,  $\boldsymbol{\tau} \in \mathbb{R}^{n_u}$  are the joint torques of the robot, and  $\mathbf{S}$  is the selection matrix for the actuated degrees of freedom. The pair  $\mathbf{J}_i$  and  $\boldsymbol{\lambda}_i$  are the Jacobian and wrench corresponding to the  $i$ th contact in the set of external contacts  $\mathcal{C}$ .  $\mathbf{M}$  is the mass matrix and  $\mathbf{c}$  collects the centripetal, Coriolis, and gravitational terms.

Denoting the number of rigid bodies in the system by  $N$ , we can follow [28] and write the inverse dynamics in a form where the inertial parameters of the robot appear in a linear fashion:

$$\mathbf{Y}(\mathbf{q}, \mathbf{v}, \dot{\mathbf{v}}) \boldsymbol{\pi} = \mathbf{S}^\top \boldsymbol{\tau} + \sum_{i \in \mathcal{C}} \mathbf{J}_i^\top(\mathbf{q}) \boldsymbol{\lambda}_i, \quad (2)$$

where  $\mathbf{Y} \in \mathbb{R}^{n_v \times 10N}$  is a state-dependent regressor matrix that maps the stacked inertial parameters  $\boldsymbol{\pi} = [\boldsymbol{\pi}_1^\top, \boldsymbol{\pi}_2^\top, \dots, \boldsymbol{\pi}_N^\top]^\top \in \mathbb{R}^{10N}$  into torque. The inertial parameters for the  $n$ th individual body  $\boldsymbol{\pi}_n \in \mathbb{R}^{10}$  consist of its mass  $m$ , first moment of mass  $\mathbf{h} \in \mathbb{R}^3$ , and the unique elements of the symmetric rotational inertia matrix  $\mathbf{I} \in \mathbb{R}^{3 \times 3}$ :

$$\boldsymbol{\pi}_n = [m, h_x, h_y, h_z, I_{xx}, I_{xy}, I_{xz}, I_{yy}, I_{yz}, I_{zz}]^\top. \quad (3)$$

As in [28], it is then possible to collect samples of robot motion and leverage the linear structure of the equations of motion to get parameter estimates by solving a least squares problem. Concretely, one gathers  $K$  samples of  $\mathbf{q}$ ,  $\mathbf{v}$ ,  $\dot{\mathbf{v}}$ ,  $\boldsymbol{\tau}$ , and  $\boldsymbol{\lambda}$ , and creates the following stacked problem data:

$$\underbrace{\begin{bmatrix} \mathbf{Y}_1 \\ \mathbf{Y}_2 \\ \vdots \\ \mathbf{Y}_K \end{bmatrix}}_{\bar{\mathbf{Y}}} \boldsymbol{\pi} = \underbrace{\begin{bmatrix} \mathbf{S}^\top \boldsymbol{\tau}_1 + \sum_i \mathbf{J}_1^\top \boldsymbol{\lambda}_1 \\ \mathbf{S}^\top \boldsymbol{\tau}_2 + \sum_i \mathbf{J}_2^\top \boldsymbol{\lambda}_2 \\ \vdots \\ \mathbf{S}^\top \boldsymbol{\tau}_K + \sum_i \mathbf{J}_K^\top \boldsymbol{\lambda}_K \end{bmatrix}}_{\bar{\mathbf{f}}} \quad (4)$$

where  $\sum_i \mathbf{J}_j^\top \boldsymbol{\lambda}_j$  is an abuse of notation denoting the sum of contact wrench torque contributions over contacts  $i \in \mathcal{C}$  for sample  $j$ . One then solves the linear least squares problem:

$$\min_{\hat{\boldsymbol{\pi}}} \frac{1}{2} \|\bar{\mathbf{f}} - \bar{\mathbf{Y}} \hat{\boldsymbol{\pi}}\|_{\mathbf{W}}^2, \quad (5)$$

where  $\mathbf{W}$  is an appropriately chosen weight matrix that captures the variance of the measurements and the characteristic scales of the different units (e.g. mass vs. inertia).

Though we do not follow the batch estimation procedure in (5), we include it to highlight several difficulties of the problem (see also [4]):

*Unobservability due to kinematic structure.* Oftentimes the kinematic structure of the robot renders certain parameters unobservable, or observable only in linear combinations [12]. This will cause  $\bar{\mathbf{Y}}$  to become rank-deficient.

*Lack of persistent excitation in motion data.* If the motion data is not sufficiently rich enough to excite all the degrees of freedom, again  $\bar{\mathbf{Y}}$  will be rank-deficient. In both cases, (5) can be regularized and solved by methods such as damped least squares [7] or other regularization techniques [29].

*Physically inconsistent estimates.* While (5) exploits the linearity of the inverse dynamics with respect to inertial parameters, it does not take into account additional problem structure [8, 9]; depending on the data, one may receive estimates where mass is negative, and moment of inertia matrices that are physically impossible. If a parameter estimation algorithm is run in a live control loop, as in this work, non-physical parameter estimates could yield catastrophic results when used in a downstream model-based controller.

### III. METHOD

#### A. Condensing the inertial estimation problem

Most existing work on inertial parameter estimation for humanoids has focused on system identification, where most (if not all) of the inertial parameters are considered to be poorly known and thus required to estimate.

In this work, we are more focused on online estimation and adaptation. We make the assumption that a prior offline system identification procedure has been performed, so that many of the inertial parameters are well-known and can be considered fixed. Furthermore, we assume knowledge of the tasks the robot will perform, such that we know what links will vary inertially. Therefore, only a subset of the robot's inertial parameter vector  $\boldsymbol{\pi}$  needs to be actively estimated online. Indeed, one can split the product of the regressor matrix and parameter vector into "estimate"  $\boldsymbol{\pi}^{\text{est}}$  and "nominal"  $\boldsymbol{\pi}^{\text{nom}}$  terms:

$$\mathbf{Y}\boldsymbol{\pi} = \mathbf{Y}^{\text{est}}\boldsymbol{\pi}^{\text{est}} + \mathbf{Y}^{\text{nom}}\boldsymbol{\pi}^{\text{nom}}, \quad (6)$$

where typically  $\dim(\boldsymbol{\pi}^{\text{est}}) \ll \dim(\boldsymbol{\pi}^{\text{nom}})$ .

This decomposition has multiple benefits. Firstly, it can help to resolve linear dependence issues by pinning the inertial parameters in  $\boldsymbol{\pi}^{\text{nom}}$  to certain values. Secondly, one can exploit some well-known concepts in the rigid body dynamics literature to efficiently calculate the two terms in (6). For the first term,  $\mathbf{Y}^{\text{est}}$  can be found via a modified recursive Newton-Euler algorithm (RNEA), where the forward pass is run just once independently, and spatial forces in the backward pass are linearly parameterized with respect to the inertial parameters, such an approach is implemented in multiple popular software libraries [30, 31]. For the second term, one can use the assumed-known  $\boldsymbol{\pi}^{\text{nom}}$ : the equivalent

torque  $\boldsymbol{\tau}^{\text{nom}}$  can be computed with a *single* RNEA call where any inertial parameters not in  $\boldsymbol{\pi}^{\text{nom}}$  are zeroed. This concept is similar to that mentioned in [32, p. 94], where the zeroing of certain parameters in RNEA yields certain dynamic quantities. These simplifications result in:

$$\mathbf{Y}\boldsymbol{\pi} = \mathbf{Y}^{\text{est}}\boldsymbol{\pi}^{\text{est}} + \boldsymbol{\tau}^{\text{nom}}. \quad (7)$$

We next show this efficient regressor calculation can be used as the measurement model in a Kalman Filter designed to estimate  $\boldsymbol{\pi}^{\text{est}}$ .

#### B. Inertial Kalman Filter

Following [33, Chapter 5.1], the standard form for the process model and measurement model of a discrete-time Kalman filter with no control input is:

$$\mathbf{x}_k = \mathbf{F}_k\mathbf{x}_{k-1} + \boldsymbol{\eta}_k, \quad \boldsymbol{\eta}_k \sim \mathcal{N}(\mathbf{0}, \mathbf{Q}_k), \quad (8)$$

$$\mathbf{z}_k = \mathbf{H}_k\mathbf{x}_k + \boldsymbol{\nu}_k, \quad \boldsymbol{\nu}_k \sim \mathcal{N}(\mathbf{0}, \mathbf{R}_k). \quad (9)$$

Within a single time instant  $k$ , the *predict* and *update* step proceed as:

$$\begin{aligned} \text{predict: } \mathbf{x}_k^- &= \mathbf{F}_k\mathbf{x}_{k-1} \\ \mathbf{P}_k^- &= \mathbf{F}_k\mathbf{P}_{k-1}\mathbf{F}_k^\top + \mathbf{Q}_k \end{aligned} \quad (10)$$

$$\begin{aligned} \text{update: } \tilde{\mathbf{y}}_k &= \mathbf{z}_k - \mathbf{H}_k\mathbf{x}_k^- \\ \mathbf{S}_k &= \mathbf{H}_k\mathbf{P}_k^-\mathbf{H}_k^\top + \mathbf{R}_k \\ \mathbf{K}_k &= \mathbf{P}_k^-\mathbf{H}_k^\top\mathbf{S}_k^{-1} \\ \mathbf{x}_k^+ &= \mathbf{x}_k^- + \mathbf{K}_k\tilde{\mathbf{y}}_k \\ \mathbf{P}_k^+ &= (\mathbf{I} - \mathbf{K}_k\mathbf{H}_k)\mathbf{P}_k^- \end{aligned} \quad (11)$$

Concerning the process model for inertial parameter estimation, we follow existing work in the literature (see [21] for a humanoid example, and [34, Chapter 5.1] for a more general discussion of estimating constant or slowly varying parameters) by assuming the only change in the parameters being estimated is driven by the white noise  $\boldsymbol{\eta}_k$ :

$$\boldsymbol{\pi}_k^{\text{est}} = \boldsymbol{\pi}_{k-1}^{\text{est}} + \boldsymbol{\eta}_k. \quad (12)$$

To form the measurement, we plug (7) into (2), replacing  $\mathbf{S}^\top\boldsymbol{\tau}$  with the observed torque from the robot  $\boldsymbol{\tau}^{\text{obs}}$ :

$$\mathbf{Y}^{\text{est}}\boldsymbol{\pi}^{\text{est}} + \boldsymbol{\tau}^{\text{nom}} = \boldsymbol{\tau}^{\text{obs}} + \sum_{i \in \mathcal{C}} \mathbf{J}_i^\top \boldsymbol{\lambda}_i \quad (13)$$

We then introduce the variable  $\Delta\boldsymbol{\tau}$  to isolate the contribution from the estimated parameters:

$$\Delta\boldsymbol{\tau} = \boldsymbol{\tau}^{\text{obs}} - \boldsymbol{\tau}^{\text{nom}} + \sum_{i \in \mathcal{C}} \mathbf{J}_i^\top \boldsymbol{\lambda}_i, \quad (14)$$

which leaves  $\Delta\boldsymbol{\tau} \approx \mathbf{Y}^{\text{est}}\boldsymbol{\pi}^{\text{est}}$ , justifying the use of the measurement model:

$$\Delta\boldsymbol{\tau}_k = \mathbf{Y}_k^{\text{est}}\boldsymbol{\pi}_k^{\text{est}} + \boldsymbol{\nu}_k. \quad (15)$$

It is worth noting that with assumed constant parameters  $\boldsymbol{\pi}^{\text{est}}$ , if we were to additionally set the process covariance  $\boldsymbol{\eta}_k$  to zero, the Kalman filter would reduce to a scheme similar to recursive least squares. We chose to retain the Kalman filter

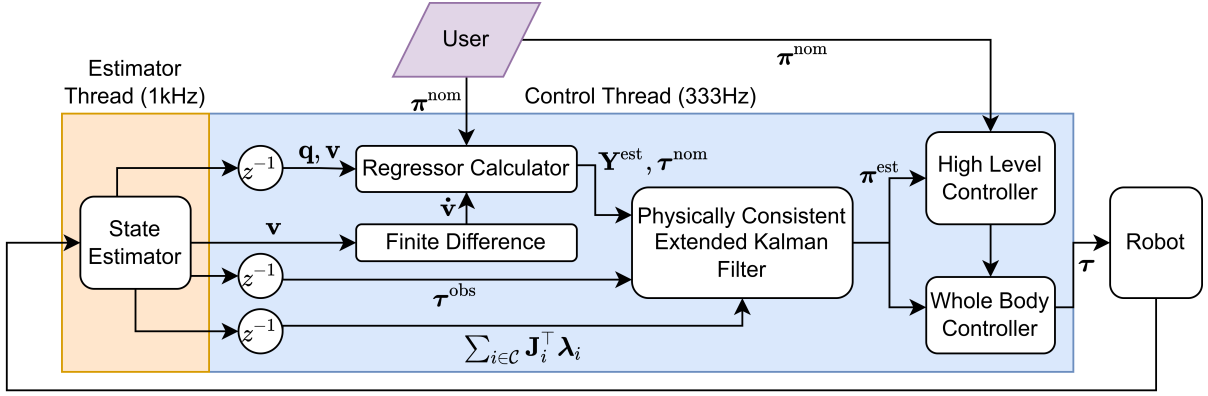


Fig. 2: Block diagram of proposed filter and its interface with the controller, as described in Section III-D.

formulation with a non-zero  $\eta_k$  as it allows one to tune the balance of  $\eta_k$  and  $\nu_k$  to achieve a slower, smoother response (process covariance  $\eta_k$  is relatively small) or a fast, noisier response (measurement covariance  $\nu_k$  is relatively small).

### C. Physically Consistent Inertial Extended Kalman Filter

The linear least squares problem in (5) only considers the inertial parameters as members of the Euclidean space  $\mathbb{R}^{10}$ . However, there are additional conditions that the inertial parameters must satisfy. [8] introduces the conditions of the mass  $m$  being positive and the rotational inertia matrix being positive definite. These conditions have since been termed *physical consistency*. Subsequently, [9] noted that the rotational inertia matrix being positive definite was a necessary but not sufficient condition. Coining the term *full physical consistency*, [9] shows in addition that a set of triangle inequalities between the principal moments of inertia must also be satisfied. Their solution method involves manifold optimization constraints, which are difficult to utilize in an online real-time control framework. Recently, [3] introduced an inertial parameterization that is fully physically consistent by construction, revolving around a log-Cholesky decomposition of the pseudo-inertia matrix. This new parameterization for the  $n^{\text{th}}$  rigid body is given by  $\theta_n \in \mathbb{R}^{10}$ :

$$\theta_n = [\alpha, d_1, d_2, d_3, s_{12}, s_{13}, s_{23}, t_1, t_2, t_3]^T, \quad (16)$$

where  $\alpha$  scales the density of a reference body, and  $d$ ,  $s$ , and  $t$  respectively provide a scaling, shearing, and translation of a reference body. One can therefore map from the log-Cholesky parameterization to the set of fully physically consistent parameters in the default parameterization via:

$$\pi_n = \mathbf{g}(\theta_n) = e^{2\alpha} \begin{bmatrix} t_1^2 + t_2^2 + t_3^2 + 1 \\ t_1 e^{d_1} \\ t_1 s_{12} + t_2 e^{d_2} \\ t_1 s_{13} + t_2 s_{23} + t_3 e^{d_3} \\ s_{12}^2 + s_{13}^2 + s_{23}^2 + e^{2d_2} + e^{2d_3} \\ -s_{12} e^{d_1} \\ -s_{13} e^{d_1} \\ s_{13}^2 + s_{23}^2 + e^{2d_1} + e^{2d_3} \\ -s_{12} s_{13} - s_{23} e^{d_2} \\ s_{12}^2 + e^{2d_1} + e^{2d_2} \end{bmatrix} \quad (17)$$

where the ordering is slightly different from [3] due to the use of a different ordering of  $\pi$  in this work.

As shown in [3], the mapping  $\mathbf{g}$  is smooth, unique, and invertible, allowing one to easily move back and forth between parameterizations of fully physically consistent parameters as required. Furthermore, the Jacobian of  $\mathbf{g}$  is closed-form, which we can exploit in our extended Kalman filtering framework. We define a similar process model to (12), where the parameters to estimate are assumed constant, but in the log-Cholesky parameterization:

$$\theta_k^{\text{est}} = \theta_{k-1}^{\text{est}} + \eta_k, \quad (18)$$

with the corresponding measurement model:

$$\Delta \tau_k = \underbrace{\mathbf{Y}_k^{\text{est}} \mathbf{g}(\theta_k^{\text{est}})}_{\mathbf{h}(\theta_k^{\text{est}})} + \nu_k. \quad (19)$$

Introducing the nonlinear measurement model  $\mathbf{h}(\theta^{\text{est}})$  requires the modification of the first two computations of the standard Kalman filter update step in (11). For the inertial estimation problem at hand, we have:

$$\begin{aligned} \Delta \tilde{\tau}_k &= \Delta \tau_k - \mathbf{h}(\theta_k^{\text{est}-}), \\ \mathbf{S}_k &= \underbrace{(\mathbf{Y}_k^{\text{est}} \mathbf{G}_k)}_{\mathbf{H}_k} \mathbf{P}_k^- \underbrace{(\mathbf{Y}_k^{\text{est}} \mathbf{G}_k)^T}_{\mathbf{H}_k^T} + \mathbf{R}_k, \end{aligned} \quad (20)$$

$$\mathbf{G}_k = \left. \frac{\partial \mathbf{g}}{\partial \theta} \right|_{\theta_k^{\text{est}-}} \quad \text{therefore} \quad \mathbf{H}_k = \left. \frac{\partial \mathbf{h}}{\partial \theta} \right|_{\theta_k^{\text{est}-}} = \mathbf{Y}_k^{\text{est}} \mathbf{G}_k \quad (21)$$

To summarise, these modifications require the forfeit of the linear least squares structure, mandating a move from the regular Kalman filter to the extended Kalman filter, but result in inertial parameters that are fully physically consistent by construction at every stage of the solve procedure.

### D. Integration with state estimator and controller

A block diagram showing the structure of our approach is given in Fig. 2. We run the inertial parameter estimator inside the control thread that is pinned at 333Hz. We read configuration  $\mathbf{q}$ , velocity  $\mathbf{v}$ , observed joint torques  $\tau^{\text{obs}}$ , and contact wrench information  $\lambda$  from the state estimator that runs at 1kHz. Accelerations  $\dot{\mathbf{v}}$  are computed inside the estimator

via finite differences, and the other signals are delayed to compensate for the lag introduced. The user supplies nominal parameters  $\pi^{\text{nom}}$ , and thus implicitly describes the parameters to be estimated  $\pi^{\text{est}}$ . The physically consistent EKF internally uses (18) and (19) before converting the estimate back to the standard inertial parameterization  $\pi^{\text{est}}$  which is passed through to the high-level controller and whole-body controller. The high-level controller is responsible for specifying motion tasks that are passed as objectives to the quadratic program solved in the whole-body controller [35]. An inverse dynamics solver is used to translate the output of the whole-body controller into commanded joint torques. While this solver does not explicitly require physically consistent inertial parameters, many state-of-the-art methods like nonlinear MPC rely on forward dynamics, which does have this explicit requirement (due to the need to invert  $M$ ).

#### IV. RESULTS

##### A. Simulation

Our simulation results utilize the open-source Simulation Construction Set 2 [36] which is integrated into our controller stack such that there is minimal change between the control code ran in simulation versus hardware. Internally, the simulator uses an efficient implementation of Featherstone’s forward dynamics algorithm [32] and has an API for modifying the inertial parameters of the robot online to simulate varying loads.

We first showcase the benefit of the physically consistent log-Cholesky parameterization from [3] in the Extended Kalman Filter (18) and (19) over a baseline Kalman Filter implementation (12) and (15). Fig. 3 shows the estimates of our robot’s torso link over time when the robot is instructed to walk forward. It is interesting to observe how the two parameterizations react to a lack of persistent excitation in the data; whilst both estimators recover the torso mass, the lack of excitation for identifying the inertia diagonals causes the baseline Kalman filter estimates for  $I_{xx}$ ,  $I_{yy}$ , and  $I_{zz}$  to drift and become physically inconsistent almost immediately, with  $I_{xx}$  becoming negative for a short time interval. In contrast, the lack of persistent excitation causes the inertia diagonals for the physically consistent EKF to vary periodically, but always remain physically consistent. The periodic oscillation warrants further investigation, but is perhaps a side-effect of the estimated parameters evolving on the log-Cholesky parameterization manifold structure when the provided data is not rich enough to uniquely identify the parameters.

We next show the benefit of integrating our inertia estimation pipeline with a downstream whole-body controller in an adaptive control manner. In Fig. 4, we compare the

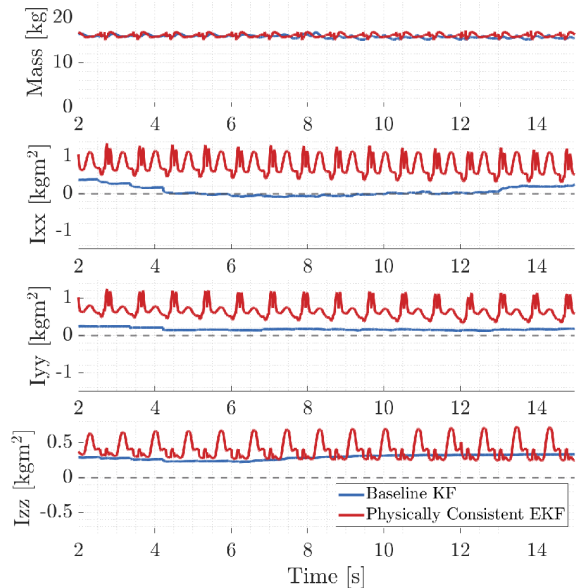


Fig. 3: Parameter estimates for the robot’s torso link from the baseline Kalman filter and the log-Cholesky extended Kalman filter.

performance on a pelvis height control task with and without adaptation of the torso link’s inertial parameters, where the mass of the torso link is varied to simulate a load-carrying task. In Fig. 5, we compare the performance on an elbow jointspace control task with and without adaptation of the forearm link’s inertial parameters. The mass of the forearm link is varied to simulate the manipulation of a heavy tool. The waveforms used for both tasks are detailed in Table I.

For the pelvis height control task, it can be observed that the controller tracks the desired pelvis height much more closely with an adaptive estimate of the torso inertial parameters than without. There are some ripples in the mass estimate that occur when the robot’s feet touchdown, but the estimator mostly captures the added load. For the elbow jointspace control task, we observe that an adaptive estimate is vital for the robot to maintain its “home pose” configuration in the presence of a time-varying load. The root-mean-square error over the two tasks, without and with adaptation, can be seen in Table II.

##### B. Hardware

The experimental procedure for hardware testing is complicated by the safety concerns that arise when fixing external loads to any powerful robot during operation. On the robot, the estimator is disabled until the robot is hoisted down to the ground and boots up into the walking state. We observed

TABLE I: Simulation mass variation waveforms.

Link	Waveform	Offset	Amplitude	Frequency
Torso	Square	30kg	5kg	0.2Hz
Forearm	Sine/Square	4.5kg	3kg	1Hz

TABLE II: Root-mean-squared error for simulated tasks.

Task	No adaptation	With adaptation
Pelvis height	0.0352m	0.0157m
Elbow angle	1.0956rad	0.1499rad

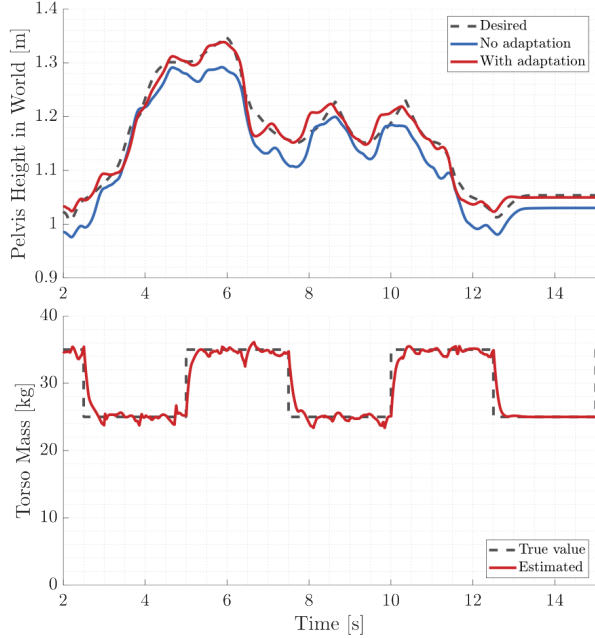


Fig. 4: Pelvis height task performance over rough terrain, with and without adaptation of the torso inertial parameters.

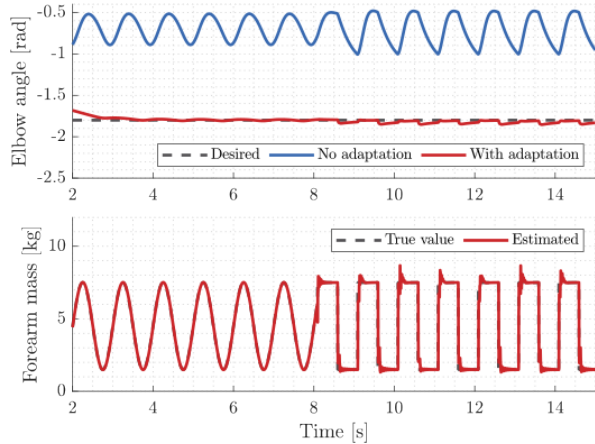


Fig. 5: Elbow jointspace task performance with and without adaptation of the forearm link inertial parameters. Around the 8s mark, the mass waveform is changed from a sine wave to a square wave.

an additional bias in the residuals due to imperfections in the nominal inertial parameters, which do not capture the added inertial effects of cables, wires, and even the safety hoist on the robot. Unfortunately, this does highlight a weakness in our approach, in that we assume these nominal parameters are well-known. We are actively working on improved off-line system identification routines to better characterize the nominal parameters. We have developed a calibration routine used on robot startup that computes the average residual over a recent time window, and then injects this constant bias as another term in (14). It is difficult to otherwise actively estimate this bias, as one can see by augmenting the filter  $\mathbf{x}'$  state with a measurement bias  $\mathbf{m}_b$  such that  $\mathbf{x}' =$



Fig. 6: Nadia curling a 12lb dumbbell after estimating the inertial parameters of the combined dumbbell and forearm link.

$[\boldsymbol{\pi}^{\text{est}}; \mathbf{m}_b]$ , resulting in a process model  $\mathbf{F}' = [\mathbf{I}, \mathbf{0}; \mathbf{0}, \mathbf{I}]$  and measurement model  $\mathbf{H}' = [\mathbf{Y}, \mathbf{I}]$  and noting that the resultant observability matrix is rank-deficient.

To test our combined estimation and control framework on hardware, we attached a 12lb dumbbell to the left arm, and then tasked the robot with performing bicep curls, as seen in Fig. 6. No prior information about the dumbbell was passed to the robot. For safety purposes, the dumbbell was attached to the robot before activation. When the estimator is first activated, the parameter estimates are fed through to the controller in a rate-limited manner. Taking mass as an example, we cap the rate at 3kg/s to prevent large initial step changes in controller objectives. The bicep curl is executed by commanding a desired elbow angle from a sine wave of 0.7rad amplitude and 0.333Hz frequency. The tracking performance of the controller and the corresponding mass estimate are shown in Fig. 7. The combined mass of the forearm link and the dumbbell is slightly overestimated, and ripples at the top and bottom of the curl. We believe this is primarily due to imperfections in other inertial parameters of the robot that are assumed known. As previously mentioned, we approximate these imperfections with a constant bias, whereas in actuality their effects are state-dependent. The elbow angle in Fig. 7 generally follows the reference for the curl, but with some phase lag and a deadband-like response at the bottom of the curl. It is worth noting that without the estimate being fed into the controller, the robot cannot even achieve the home pose, limply hanging the arm down by the robot's side.

When testing an early version of the estimator we noticed that significant spikes in the parameter estimates occur around contact times. During locomotion, foot impacts with the ground result in shocks in the acceleration, torque, and wrench signals, which immediately manifest in the filter residual, causing an impulse response in the filter estimate. Rather than trying to pre-process the input signals to the filter, we found the best solution was to use *innovation gating* (see [37, Chapter 7] and the monograph [38, Chapter 5.4]). We compute the *normalized innovation* from each measurement update and reject those above a normalization

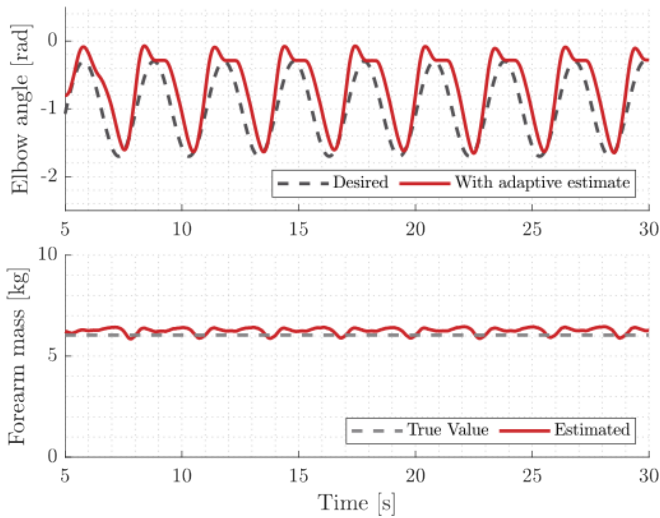


Fig. 7: Elbow joint tracking and adaptive estimate of the forearm link mass during bicep curls on hardware.

threshold  $\gamma$ . Therefore, for a measurement to be utilized, we require:

$$\Delta \tilde{\tau}_k^\top \mathbf{S}_k^{-1} \Delta \tilde{\tau}_k \leq \gamma. \quad (22)$$

There is a statistical interpretation of the normalized innovation varying as a  $\chi^2$  distribution with  $\gamma$  acting as the bound for a confidence interval. In our situation, we simply resorted to tuning the value online, and leave more rigorous investigation of this value, and adaptive schemes for changing it in the presence of extremely large estimate variations, to future work. The effect of this innovation gating can be seen in Fig. 8, where the robot is commanded to walk in place and the effect of the contact impulses on the torso link mass estimate is recorded. If the measurements are not gated, undesirable spikes are consistently seen in the estimate, whereas with innovation gating, the effect of the spikes is largely attenuated.

Unfortunately, we do not yet have the experimental apparatus in place to measure changes in the torso inertial parameters in a verifiable manner, as we do with the arms. However, we show that we can reliably recover a mass estimate close to the nominal mass of the robot in our URDF. We stress that this URDF value is likely to be underestimating the true mass of the link, as the URDF value is pulled directly from CAD, where the mass contributions from wires and other fixings to the torso are not captured. Therefore, we consider our estimate to be reasonable.

## V. CONCLUSIONS AND FUTURE WORK

In this work we have introduced an estimation and control framework that facilitates online adaptation of the inertial parameters of a humanoid robot. We leveraged a recently proposed inertial parameterization that is fully physically consistent by construction—and therefore does not require any constraints—but does introduce nonlinearity, mandating that an extended Kalman Filter is used. By assuming the completion of an offline system identification phase, and that the robot will be engaging in tasks involving certain links, we showed that the problem can be reduced in size significantly,

mitigating identifiability issues and permitting more efficient computation in the rigid body dynamics algorithms used inside the filter.

We demonstrated the effectiveness of the proposed filter both in simulation and on hardware, showing that Nadia is able to identify and accommodate a 12lb dumbbell being attached to the wrist link while performing bicep curls. We also showed the robot’s ability to closely estimate the mass of the torso link while stepping in place and introduced an innovation gating mechanism in the EKF measurement update to reject spikes in the estimates due to contact impulses.

## VI. ACKNOWLEDGMENTS

We would like to thank Khizar Mohammed and Nick Kitchel for robot maintenance and experimental assistance; Bhavyansh Mishra, Geoffrey Clark, and Carlos Mastalli for insightful discussions and feedback; Achintya Mohan for graphics assistance; and the team at Boardwalk Robotics for fabricating the dumbbell mounts. We additionally thank the anonymous reviewers for helpful suggestions and clarifications that improved the paper.

## REFERENCES

- [1] G. Orbán and D. M. Wolpert, “Representations of uncertainty in sensorimotor control,” *Current Opinion in Neurobiology*, vol. 21, no. 4, pp. 629–635, 2011.
- [2] D. McNamee and D. M. Wolpert, “Internal models in biological control,” *Annual Review of Control, Robotics, and Autonomous Systems*, vol. 2, pp. 339–364, 2019.
- [3] C. Rucker and P. M. Wensing, “Smooth parameterization of rigid-body inertia,” *IEEE Robotics and Automation Letters*, vol. 7, no. 2, pp. 2771–2778, 2022.
- [4] Q. Leboutet, J. Roux, A. Janot, J. R. Guadarrama-Olvera, and G. Cheng, “Inertial parameter identification in robotics: A survey,” *Applied Sciences*, vol. 11, no. 9, p. 4303, 2021.
- [5] A. R. Geist and S. Trimpe, “Structured learning of rigid-body dynamics: A survey and unified view from a robotics perspective,” *GAMM-Mitteilungen*, vol. 44, no. 2, p. e202100009, 2021.
- [6] T. Lee, J. Kwon, P. M. Wensing, and F. C. Park, “Robot model identification and learning: A modern perspective,” *Annual Review of Control, Robotics, and Autonomous Systems*, vol. 7, 2023.
- [7] C. G. Atkeson, C. H. An, and J. M. Hollerbach, “Estimation of inertial parameters of manipulator loads and links,” *The International Journal of Robotics Research*, vol. 5, no. 3, pp. 101–119, 1986.
- [8] K. Yoshida, K. Osuka, H. Mayeda, and T. Ono, “When is the set of base-parameter values physically impossible?” *Journal of the Robotics Society of Japan*, vol. 14, no. 1, pp. 122–130, 1996.
- [9] S. Traversaro, S. Brossette, A. Escande, and F. Nori, “Identification of fully physical consistent inertial parameters using optimization on manifolds,” in *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2016, pp. 5446–5451.
- [10] P. M. Wensing, S. Kim, and J.-J. E. Slotine, “Linear matrix inequalities for physically consistent inertial parameter identification: A statistical perspective on the mass distribution,” *IEEE Robotics and Automation Letters*, vol. 3, no. 1, pp. 60–67, 2017.
- [11] K. Ayusawa, G. Venture, and Y. Nakamura, “Identifiability and identification of inertial parameters using the underactuated base-link dynamics for legged multibody systems,” *The International Journal of Robotics Research*, vol. 33, no. 3, pp. 446–468, 2014.
- [12] R. Bellman and K. J. Åström, “On structural identifiability,” *Mathematical Biosciences*, vol. 7, no. 3–4, pp. 329–339, 1970.
- [13] M. Gautier and W. Khalil, “Exciting trajectories for the identification of base inertial parameters of robots,” *The International Journal of Robotics Research*, vol. 11, no. 4, pp. 362–375, 1992.
- [14] —, “Identification of the minimum inertial parameters of robots,” in *1989 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE Computer Society, 1989, pp. 1529–1530.

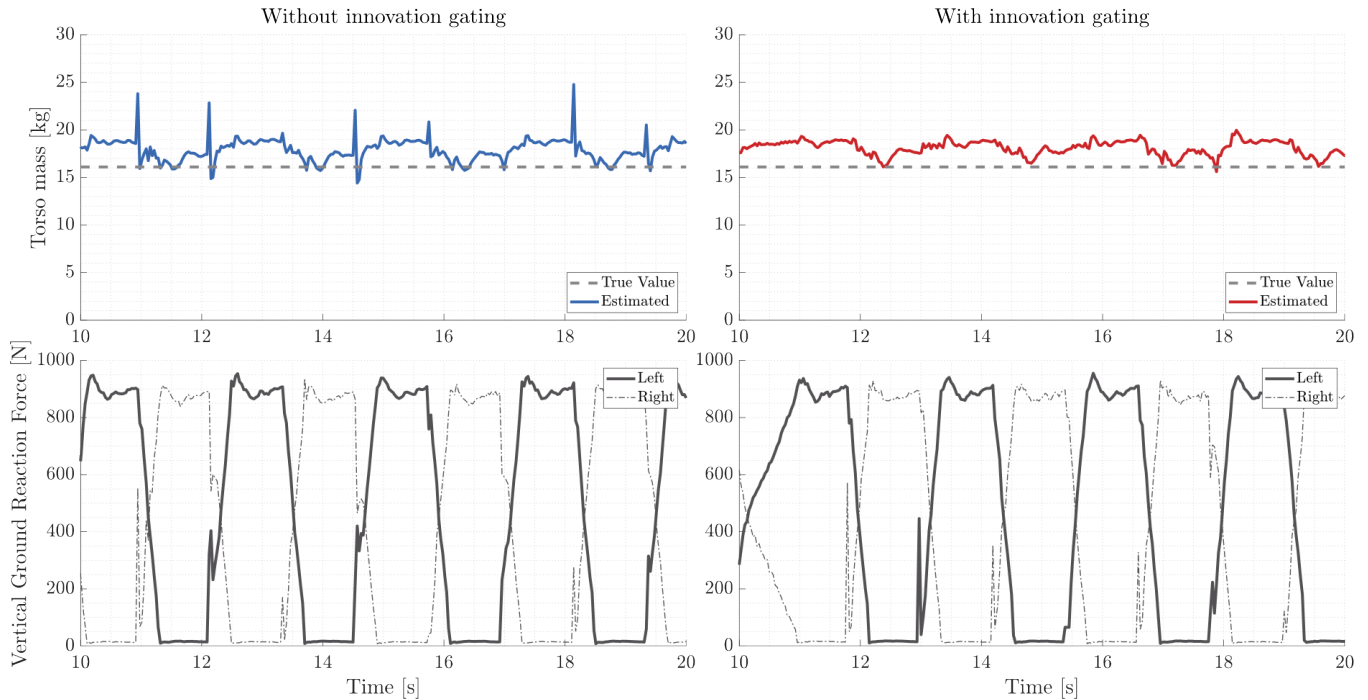


Fig. 8: Comparison of torso link mass estimates before and after introducing innovation gating during hardware tests, along with the corresponding vertical ground reaction forces. We believe that the “true value” of the torso mass drawn directly from CAD likely underestimates the mass of the link when cabling and other features are added.

- [15] P. M. Wensing, G. Niemeyer, and J.-J. E. Slotine, “Observability in inertial parameter identification,” *arXiv preprint arXiv:1711.03896*, 2017.
- [16] J. Swevers, C. Ganseman, D. B. Tukul, J. De Schutter, and H. Van Brussel, “Optimal robot excitation and identification,” *IEEE Transactions on Robotics and Automation*, vol. 13, no. 5, pp. 730–740, 1997.
- [17] A. C.-H. Lee, H.-K. Hsu, and H.-P. Huang, “Optimized system identification of humanoid robots with physical consistency constraints and floating-based exciting motions,” *International Journal of Humanoid Robotics*, vol. 19, no. 05, p. 2250015, 2022.
- [18] J. Jovic, A. Escande, K. Ayusawa, E. Yoshida, A. Kheddar, and G. Venture, “Humanoid and human inertia parameter identification using hierarchical optimization,” *IEEE Transactions on Robotics*, vol. 32, no. 3, pp. 726–735, 2016.
- [19] V. Bonnet, A. Crosnier, G. Venture, M. Gautier, and P. Fraithe, “Inertial parameters identification of a humanoid robot hanged to a fix force sensor,” in *2018 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2018, pp. 4927–4932.
- [20] K. Mori, K. Ayusawa, and E. Yoshida, “Online center of mass and momentum estimation for a humanoid robot based on identification of inertial parameters,” in *2018 IEEE-RAS 18th International Conference on Humanoid Robots (Humanoids)*. IEEE, 2018, pp. 1–9.
- [21] 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)*. IEEE, 2015, pp. 556–563.
- [22] J.-J. E. Slotine and W. Li, “On the adaptive control of robot manipulators,” *The International Journal of Robotics Research*, vol. 6, no. 3, pp. 49–59, 1987.
- [23] D. Pucci, F. Romano, and F. Nori, “Collocated adaptive control of underactuated mechanical systems,” *IEEE Transactions on Robotics*, vol. 31, no. 6, pp. 1527–1536, 2015.
- [24] T. Lee, J. Kwon, and F. C. Park, “A natural adaptive control law for robot manipulators,” in *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2018, pp. 1–9.
- [25] M. V. Minniti, R. Grandia, F. Farshidian, and M. Hutter, “Adaptive clf-mpc with application to quadrupedal robots,” *IEEE Robotics and Automation Letters*, vol. 7, no. 1, pp. 565–572, 2021.
- [26] M. Sombolstan and Q. Nguyen, “Adaptive force-based control of dynamic legged locomotion over uneven terrain,” *arXiv preprint arXiv:2307.04030*, 2023.
- [27] M. Elobaid, G. Romualdi, G. Nava, L. Rapetti, H. A. O. Mohamed, and D. Pucci, “Online non-linear centroidal mpc for humanoid robots payload carrying with contact-stable force parametrization,” *arXiv preprint arXiv:2305.10917*, 2023.
- [28] M. Mistry, S. Schaal, and K. Yamane, “Inertial parameter estimation of floating base humanoid systems using partial force sensing,” in *2009 9th IEEE-RAS International Conference on Humanoid Robots*. IEEE, 2009, pp. 492–497.
- [29] T. Lee, P. M. Wensing, and F. C. Park, “Geometric robot dynamic identification: A convex programming approach,” *IEEE Transactions on Robotics*, vol. 36, no. 2, pp. 348–365, 2019.
- [30] J. Carpentier, F. Valenza, N. Mansard *et al.*, “Pinocchio: fast forward and inverse dynamics for poly-articulated systems,” <https://stack-of-tasks.github.io/pinocchio>, 2015–2021.
- [31] P. Wensing, S. Singh, and H. Li, “spatial.v2.extended,” <https://github.com/ROAM-Lab-ND/spatial.v2.extended>, 2020–2024.
- [32] R. Featherstone, *Rigid Body Dynamics Algorithms*. Springer, 2014.
- [33] D. Simon, *Optimal State Estimation: Kalman, H infinity, and Nonlinear Approaches*. John Wiley & Sons, 2006.
- [34] T. D. Barfoot, *State Estimation for Robotics*. Cambridge University Press, 2024.
- [35] T. Koolen, S. Bertrand, G. Thomas, T. De Boer, T. Wu, J. Smith, J. Engelsberger, and J. Pratt, “Design of a momentum-based control framework and application to the humanoid robot atlas,” *International Journal of Humanoid Robotics*, vol. 13, no. 01, 2016.
- [36] S. Bertrand *et al.*, “Simulation Construction Set 2,” <https://github.com/iHmRobotics/simulation-construction-set-2>, 2019–2024.
- [37] M. S. Grewal and A. P. Andrews, *Kalman filtering: Theory and Practice with MATLAB*. John Wiley & Sons, 2014.
- [38] Y. Bar-Shalom, X. R. Li, and T. Kirubarajan, *Estimation with Applications to Tracking and Navigation: Theory, Algorithms and Software*. John Wiley & Sons, 2004.