

An Online Inertial Parameter Identification Framework with Complete Physical Consistency: Lower-Body Exoskeleton Use Case

Ahmed Fahmy Soliman¹, Patrick M. Wensing², and Barkan Ugurlu³

Abstract—This paper presents a real-time inertial parameter identification algorithm for physically consistent real-time human–robot system modeling. The algorithm is implemented for execution on on-board robot computers, producing physically plausible parameter estimates suitable for direct use in model-based exoskeleton–human interaction control. By combining noise-robust regression with a physically constrained optimization structure, the method remains effective across contact transitions and dynamically changing interactions. The identification problem is solved online through recursive semidefinite programming, allowing parameters to be updated continuously without sacrificing real-time performance or physical validity. The proposed algorithm was validated in both simulation and hardware experiments. In simulation, a 40 kg underactuated exoskeleton interacting with twelve anthropomorphic human models (66–102 kg) was evaluated. Hardware experiments further demonstrated performance during self-balancing squatting motions of varying amplitudes using a bipedal exoskeleton with a dummy manikin. Across all scenarios, the proposed framework achieved improved computational efficiency and estimation accuracy compared to existing methods, while consistently preserving the physical consistency of the identified parameters.

Note to Practitioners— Lower-limb exoskeletons are increasingly used in rehabilitation, industrial assistance, and logistics to reduce physical workload and improve mobility. A major challenge in deploying such systems at scale is the large variability in user body properties, which can significantly degrade the performance of model-based control algorithms when inaccurate models are used. To this end, this work presents a real-time inertial parameter identification method that estimates the human–exoskeleton model parameters in real-time during normal operation. The proposed approach runs in real-time on on-board hardware, requires no dedicated excitation motions, and maintains physically consistent model parameters. From a practitioner’s perspective, this enables faster system setup, reduces commissioning effort, and improves control robustness across different users. The method is particularly suitable for locomotion tasks, e.g., walking and squatting, in both assistive and power-augmentation scenarios. Beyond exoskeletons, the

method has strong potential to benefit robotic systems involving changing payloads or physical human–robot interaction.

Index Terms—Inertial parameter identification, Physical consistency, Wearable robots, Recursive least squares, Semidefinite programming.

I. INTRODUCTION

Advances in wearable technologies have led to the development of lower-limb exoskeletons for gait assistance, offering the potential to improve the quality of life for individuals in general [1], [2]. A major control challenge in these robots arises from increased modeling uncertainty due to the inertial properties of human users [3]. To address this issue, various robust controllers have been proposed [4], [5]; however, they are typically designed for specific functions, e.g., trajectory tracking, and may not fully accommodate other use cases for model-based control.

With the emergence of wearable robotics, the demand for user-specific model-based control techniques is expected to increase. Thus, this paper contributes to this area by addressing key challenges in robust and real-time inertial parameter identification for wearable robots, e.g., exoskeletons. The remainder of the paper is organized as follows. After a concise literature review and an outline of the paper’s contributions, Section II presents the preliminaries on inertial parameter identification. Section III introduces the computation of the filtered regressor, the identifiable parameters tracking algorithm (IPTA), and the development of a recursive optimization solver. The simulation and experimental results are presented and discussed in Section IV. Finally, the paper is concluded in Section V.

A. State-of-the-Art

In response to the increasing demand for personalized model based controllers for exoskeletons, inertial parameter identification techniques have gained renewed attention [6]. A seminal contribution by Atkeson et al. [7] established that the dynamics of a robot can be rearranged into a form that linearly relates measured joint forces to inertial parameters, i.e., mass, center of mass position, and inertia of individual links. These parameters could then be estimated using a least squares formulation. Building on this linear formulation, more recent works have introduced modern identification strategies that enhance robustness, numerical stability, and applicability to a wide range of robotic systems [8]–[14].

Historically, two main factors have posed challenges for parameter identification procedures. The first challenge is

¹A. F. Soliman is with the Robotics and Artificial Intelligence Collaboration (RAICo), The University of Manchester, Whitehaven CA28 6AU, United Kingdom. ahmed.soliman@manchester.ac.uk

²Patrick M. Wensing is with the Department of Aerospace and Mechanical Engineering, University of Notre Dame, Notre Dame, IN 46556, USA. pwensing@nd.edu

³B. Ugurlu is with the Department of Mechanical Engineering, Ozyegin University, Istanbul 34794, Türkiye. barkan.ugurlu@ozyegin.edu.tr

This work was supported by the Scientific and Technological Research Council of Türkiye (TÜBİTAK) under Project 215E138. The authors also acknowledge the Robotics and Artificial Intelligence Collaboration (RAICo) and The University of Manchester for providing computational resources.

the limitation of parameter excitability due to robot joint constraints [15] and identification trajectories [16], [17], which directly affects the determination of identifiable parameters. These parameters can be obtained either symbolically [18]–[20] or numerically using decomposition methods [7], [21]. The second challenge is the requirement for complete physical consistency [22], which ensures that inertia tensors are not only positive definite but also satisfy the so-called triangle inequalities. In [9], Wensing et al. addressed these inequalities using a linear matrix inequality (LMI) applied to a 4×4 pseudo-inertia matrix. A semidefinite programming (SDP) framework can then be employed to enforce the positive definiteness of this matrix.

A great majority of parameter identification algorithms face challenges when applied to legged robots. The absence of a fixed base introduces complexities such as unactuated virtual joints, contact phase variations [23], and the necessity of force sensing for all joints and contact points. In response to this matter, several identification algorithms have been developed to reduce the number of measurement requirements [24], [25]. While influential studies have demonstrated promising results [9], [13], [14], [22], [24], they are often implemented in offline settings, prioritizing accuracy over computational efficiency. Consequently, data collection must be performed prior to actual use of a given system. Though this approach may be suitable for humanoid and quadruped robots, it may not be practical for exoskeleton applications, as requiring each human user to complete a dedicated data collection session, particularly with specific excitation trajectories, can be impractical. To overcome this limitation, we envision an identification framework capable of continuously estimating human-robot inertial parameters during nominal operation, without requiring dedicated excitation trajectories or a separate calibration phase.

In response to this matter, Zhang et al. present a real-time, safety-critical system identification framework that combines real-time parameter estimation with ARMOUR-based receding-horizon excitation planning using degree-5 Bézier trajectories [26]. However, the method relies on a dedicated pre-task identification phase with purposefully exciting motions. Therefore, it may not address the need for estimating human-exoskeleton lumped inertial parameters continuously during nominal operation without requiring separate excitation trajectories or an identification stage.

In a seminal work, Ayusawa et al. proposed an algorithm in which each rigid body is approximated using a finite set of point masses at fixed locations relative to each body's body-fixed reference frame [8], [27]. By enforcing all masses must be positive, they ensure that the resulting parameters are physically consistent [8]. Their algorithm supports real-time implementation and has served as a basis for other studies where researchers estimated the robot payload [28] and implemented model-based locomotion controllers [25]. Despite yielding promising results, it has two inherent limitations: i) The selection of the number of point masses and their positions is heuristic. ii) While ensuring positive point masses guarantees compliance with the triangle inequalities [22], relying on point mass can be limiting. Increasing the

number of point masses can slow computations and lead to more challenging conditioning. We defend that working directly with the inertial parameters, without imposing a point mass decomposition a priori, can be more beneficial.

B. Motivation and Contributions

A key limitation of most existing inertial parameter identification methods is that they require either complete datasets to be collected before deployment for offline identification or dedicated excitation trajectories to be executed during a separate calibration phase. Consequently, parameter identification is typically not performed during normal operation. While such approaches may be acceptable for industrial manipulators or legged robots, they are impractical for wearable exoskeletons which must accommodate different users without exhaustive pre-use calibration procedures. This limitation is particularly restrictive for emerging self-balancing lower-limb exoskeletons [29]–[32], where reliable model identification during natural activities (e.g., walking, sit-to-stand, squatting) is essential for safe and adaptive human-robot interaction. As a result, there is a clear need for identification frameworks that estimate coupled human-robot inertial parameters during nominal operation.

In response to this matter, this paper proposes a real-time inertial parameter identification framework for floating-base systems under time-varying contact conditions. The proposed framework enables physically consistent and numerically stable estimation of coupled human-exoskeleton parameters during nominal operation, removing the need for offline calibration or dedicated excitation. Physical consistency constraints are explicitly enforced while maintaining computational tractability for real-time use. To the best of our knowledge, no existing approach **simultaneously** supports real-time execution, freedom from pre-task calibration, complete physical consistency, and ability to handle time-varying contacts. Therefore, this paper aims to fill this critical gap in the literature.

The proposed framework achieves this through a synergistic integration of three key components: (i) a novel identifiable-parameters tracking algorithm (IPTA) that monitors and improves regressor conditioning online to ensure numerical stability and convergence; (ii) a reformulated filtered regressor approach for real-time identification, eliminating the need for acceleration measurements and improving robustness to measurement noise and contact switching; and (iii) a recursive semidefinite programming (SDP) scheme solved via an interior-point method that enforces physical feasibility, including triangle inequalities, at every update step. These components are tightly integrated to handle the core challenges of real-time human-exoskeleton identification, namely intermittent and unknown contact conditions, limited excitation during natural motions, and the requirement for physically valid estimates from the initial time steps. The proposed approach is validated through simulation and hardware experiments with a particular focus on a lower-body exoskeleton platform [30].

II. PRELIMINARIES

A. Rigid-Body Dynamics

The dynamics of a floating-base rigid-body system can be expressed as

$$\mathbf{M}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{h}(\mathbf{q}, \dot{\mathbf{q}}) = \mathbf{f}, \quad (1)$$

where $\mathbf{M}(\mathbf{q}) \in \mathbb{R}^{n_d \times n_d}$ is the mass matrix, $\mathbf{h}(\mathbf{q}, \dot{\mathbf{q}}) \in \mathbb{R}^{n_d}$ collects Coriolis, centrifugal, and gravitational terms, and $\mathbf{f} \in \mathbb{R}^{n_d}$ denotes the generalized forces. The latter can be written as

$$\mathbf{f} = \mathbf{S}^\top \boldsymbol{\tau} + \mathbf{J}_c(\mathbf{q})^\top \boldsymbol{\lambda}_c, \quad (2)$$

where $\boldsymbol{\tau} \in \mathbb{R}^{n_r}$ are the actuated joint torques, $\boldsymbol{\lambda}_c \in \mathbb{R}^{6n_c}$ are external contact wrenches, $\mathbf{S}$ is the actuation selection matrix, and $\mathbf{J}_c(\mathbf{q})$ is the contact Jacobian.

Following Atkeson et al. [7], the dynamics in (1) are linear in the inertial parameter vector $\boldsymbol{\pi} \in \mathbb{R}^{10n_b}$. Stacking n_s measurements yields the regressor form

$$\mathbf{Y}(\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}})\boldsymbol{\pi} = \mathbf{f}_s, \quad (3)$$

where $\mathbf{Y}(\cdot)$ is the inertial regressor and $\mathbf{f}_s$ collects the generalized forces across samples.

Inertial parameter identification can then be formulated as the following constrained least-squares problem:

$$\begin{aligned} \hat{\boldsymbol{\pi}} = \operatorname{argmin}_{\boldsymbol{\pi} \in \mathbb{R}^{10n_b}} \|\mathbf{Y}(\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}})\boldsymbol{\pi} - \mathbf{f}_s\|^2 \\ \text{s.t. } m_b > 0, \quad \mathbf{I}_b^c(\boldsymbol{\pi}) \succ \mathbf{0}, \quad b = 1, \dots, n_b, \end{aligned} \quad (4)$$

where m_b and $\mathbf{I}_b^c$ denote the mass and the inertia tensor about the center of mass of body b , respectively. The vector $\boldsymbol{\pi} = [\pi_1^\top, \dots, \pi_{n_b}^\top]^\top$, stores the inertia parameters vector of each body, with each $\boldsymbol{\pi}_b \in \mathbb{R}^{10}$ giving the 10 inertial parameters:

$$\boldsymbol{\pi}_b = [I_{b,xx} \ I_{b,yy} \ I_{b,zz} \ I_{b,xy} \ I_{b,xz} \ I_{b,yz} \ h_{b,x} \ h_{b,y} \ h_{b,z} \ m_b]^\top, \quad (5)$$

where, m_b represents the mass of body b , $\mathbf{h}_b = [h_{b,x} \ h_{b,y} \ h_{b,z}]^\top$ is first-order mass moment vector, and $I_{b,xx}, I_{b,yy}, I_{b,zz}, I_{b,xy}, I_{b,xz}, I_{b,yz}$ the components of the inertia tensor $\mathbf{I}_b$ about the origin of a body-fixed coordinate frame.

B. Complete Physical Consistency

Beyond positive definiteness of the inertia tensor about the center of mass (CoM), physically consistent inertial parameters must also satisfy the triangle inequalities on the principal moments of inertia [22]. Since the CoM inertia matrix $\mathbf{I}_b^c$ is symmetric, it admits the eigen decomposition

$$\mathbf{I}_b^c = \mathbf{O}_b \boldsymbol{\Lambda}_b^c \mathbf{O}_b^\top, \quad (6)$$

where $\mathbf{O}_b$ is orthogonal and $\boldsymbol{\Lambda}_b^c = \operatorname{diag}(\lambda_{b,1}^c, \lambda_{b,2}^c, \lambda_{b,3}^c)$ contains the eigenvalues of $\mathbf{I}_b^c$.

The triangle inequalities require that

$$\begin{aligned} \lambda_{b,1}^c + \lambda_{b,2}^c > \lambda_{b,3}^c, \quad \lambda_{b,1}^c + \lambda_{b,3}^c > \lambda_{b,2}^c, \\ \lambda_{b,2}^c + \lambda_{b,3}^c > \lambda_{b,1}^c, \end{aligned} \quad (7)$$

which impose additional nonlinear constraints on the inertial parameters.

As shown by Wensing et al. [9], these inequalities can be equivalently enforced by requiring positive definiteness of the

pseudo-inertia matrix

$$\mathbf{J}_b(\boldsymbol{\pi}_b) = \begin{bmatrix} \frac{\operatorname{tr}(\mathbf{I}_b)}{2} \mathbf{1}_3 - \mathbf{I}_b & \mathbf{h}_b \\ \mathbf{h}_b^\top & m_b \end{bmatrix} \succ \mathbf{0}, \quad (8)$$

where $\operatorname{tr}(\bullet)$ denotes the trace operator. Enforcing (8) guarantees complete physical consistency, including positivity of mass, positive-definite CoM inertia, and satisfaction of the triangle inequalities.

C. Identifiable Inertial Parameters

Due to joint constraints and limited excitation, not all inertial parameters are identifiable from measured data [33]. Following standard practice [21], we restrict the estimation to the set of identifiable parameters, comprising the base parameters and their linearly dependent counterparts, while discarding non-identifiable parameters corresponding to zero columns in the regressor.

After removing zero columns and permuting the remaining ones, the regressor in (3) can be written as

$$\mathbf{Y}_i \boldsymbol{\pi}_i = \mathbf{f}_s, \quad (9)$$

where $\mathbf{Y}_i = [\mathbf{Y}_\beta \ \mathbf{Y}_d]$ is the identifiable regressor and $\boldsymbol{\pi}_i = [\boldsymbol{\pi}_\beta^\top \ \boldsymbol{\pi}_d^\top]^\top$ collects the base and linearly dependent parameters. A QR decomposition of $\mathbf{Y}_i$ yields [21]

$$\mathbf{Y}_i = \mathbf{Q}_\beta [\mathbf{R}_\beta \ \mathbf{R}_d], \quad (10)$$

where $\mathbf{Q}_\beta$ has orthonormal columns and $\mathbf{R}_\beta$ is upper triangular. This allows the regression model to be rewritten as

$$[\mathbf{1}_\beta \ \mathbf{K}] \boldsymbol{\pi}_i = \mathbf{R}_\beta^{-1} \mathbf{Q}_\beta^\top \mathbf{f}_s, \quad (11)$$

with $\mathbf{K} = \mathbf{R}_\beta^{-1} \mathbf{R}_d$. This transformation improves numerical conditioning and is exploited by the proposed real-time identification scheme.

Using (11) together with the physical consistency constraint (8), the identification problem becomes

$$\begin{aligned} \hat{\boldsymbol{\pi}}_i = \operatorname{argmin}_{\boldsymbol{\pi}_i \in \mathbb{R}^{n_i}} \|\boldsymbol{\Phi} \boldsymbol{\pi}_i - \mathbf{f}\|^2 \\ \text{s.t. } \mathbf{J}_b(\boldsymbol{\pi}_{i,b}) + \mathbf{J}_b^0 \succ \mathbf{0}, \quad b = 1, \dots, n_b, \end{aligned} \quad (12)$$

where

$$\boldsymbol{\Phi} = [\mathbf{1}_\beta \ \mathbf{K}], \quad \mathbf{f} = \mathbf{R}_\beta^{-1} \mathbf{Q}_\beta^\top \mathbf{f}_s. \quad (13)$$

The matrix $\mathbf{J}_b^0$ accounts for the contribution of non-identifiable parameters and is fixed throughout the optimization. The parameter n_i is the number of identifiable parameters.

III. THE PROPOSED IDENTIFICATION ALGORITHM

A. A Filtered Regressor

The presence of joint accelerations $\ddot{\mathbf{q}}$ in (3) requires numerical differentiation of $\dot{\mathbf{q}}$, which is severely affected by noise. This can be avoided by employing a low pass filter to the signals on both sides of (1) [34]. Thus, (1) can be expressed as below:

$$\frac{\gamma}{s + \gamma} (\mathbf{M}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{Y}_h(\mathbf{q}, \dot{\mathbf{q}})\boldsymbol{\pi}) = \mathbf{f}_f, \quad (14)$$

where $\mathbf{Y}_h(\mathbf{q}, \dot{\mathbf{q}})$ represents a linear mapping between, $\mathbf{h}(\mathbf{q}, \dot{\mathbf{q}})$ and the inertial parameters. The filtered force vector is $\mathbf{f}_f = \frac{\gamma}{s + \gamma} \mathbf{f}$. The parameter γ denotes the cutoff frequency. Considering the impulse response of the first-order filter $\eta(t) = \gamma e^{-\gamma t}$, the first term on the left-hand side of (14) is computed through

convolution

$$\frac{\gamma}{s + \gamma} [\mathbf{M}(\mathbf{q})\ddot{\mathbf{q}}] = \int_0^t \eta(t - \vartheta) \mathbf{M}(\mathbf{q}(\vartheta)) \ddot{\mathbf{q}}(\vartheta) d\vartheta. \quad (15)$$

Assuming zero initial velocities, integrating (15) by parts yields the following:

$$\begin{aligned} \frac{\gamma}{s + \gamma} [\mathbf{M}(\mathbf{q})\ddot{\mathbf{q}}] &= \gamma \mathbf{M}(\mathbf{q})\dot{\mathbf{q}} \\ &\quad - \frac{\gamma}{s + \gamma} (\dot{\mathbf{M}}(\mathbf{q})\dot{\mathbf{q}} + \gamma \mathbf{M}(\mathbf{q})\dot{\mathbf{q}}). \end{aligned} \quad (16)$$

The inertia matrix can be computed as below [35]:

$$\mathbf{M}(\mathbf{q}) = \mathbf{J}(\mathbf{q})^\top \mathbf{I} \mathbf{J}(\mathbf{q}), \quad (17)$$

where the matrix $\mathbf{J}(\mathbf{q}) \in \mathbb{R}^{6n_b \times n_d}$ stores the Jacobian matrices of the bodies; $\mathbf{J}(\mathbf{q}) = [\mathbf{J}_1(\mathbf{q})^\top \mathbf{J}_2(\mathbf{q})^\top \dots \mathbf{J}_{n_b}(\mathbf{q})^\top]^\top$. The matrix $\mathbf{I} \in \mathbb{R}^{6n_b \times 6n_b}$ is a constant block diagonal matrix concerning the spatial inertias of the bodies; $\mathbf{I} = \text{diag}(\mathbf{I}_{s,1}, \mathbf{I}_{s,2}, \dots, \mathbf{I}_{s,n_b})$ [35]. The spatial inertia $\mathbf{I}_{s,b}$ for each rigid body b can be represented linearly, in terms of the inertial parameters as follows:

$$\mathbf{I}_{s,b} = \begin{bmatrix} \mathbf{I}_b & \mathcal{S}(\mathbf{h}_b) \\ \mathcal{S}(\mathbf{h}_b)^\top & m_b \mathbf{1}_3 \end{bmatrix}. \quad (18)$$

The time derivative of (17) can be computed in terms of Jacobian derivatives $\dot{\mathbf{J}}(\mathbf{q})$ such that:

$$\dot{\mathbf{M}}(\mathbf{q}) = \mathbf{J}(\mathbf{q})^\top \mathbf{I} \dot{\mathbf{J}}(\mathbf{q}) + \dot{\mathbf{J}}(\mathbf{q})^\top \mathbf{I} \mathbf{J}(\mathbf{q}), \quad (19)$$

where the matrix $\dot{\mathbf{J}}(\mathbf{q})$ can be analytically computed using the method reported in [35]. Thus, the eqs. (19) and (17) can be represented linearly in terms of the inertial parameters. Accordingly, (16) can be expressed as follows:

$$\frac{\gamma}{s + \gamma} [\mathbf{M}(\mathbf{q})\ddot{\mathbf{q}}] = \gamma \mathbf{Y}_m(\mathbf{q}, \dot{\mathbf{q}}) \boldsymbol{\pi} - \left[\frac{\gamma}{s + \gamma} \mathbf{Y}_{mf}(\mathbf{q}, \dot{\mathbf{q}}) \right] \boldsymbol{\pi}, \quad (20)$$

where $\mathbf{Y}_m(\mathbf{q}, \dot{\mathbf{q}})$, and $\mathbf{Y}_{mf}(\mathbf{q}, \dot{\mathbf{q}})$ are regressor matrices concerning the terms in (16). Combining (14) and (20), we obtain the following equation in which the filtered forces are linearly related to the inertial parameters:

$$[\gamma \mathbf{Y}_m(\mathbf{q}, \dot{\mathbf{q}}) + \frac{\gamma}{s + \gamma} (\mathbf{Y}_h(\mathbf{q}, \dot{\mathbf{q}}) - \mathbf{Y}_{mf}(\mathbf{q}, \dot{\mathbf{q}}))] \boldsymbol{\pi} = \mathbf{f}_f. \quad (21)$$

For convenience, the matrix $\mathbf{Y}_f(\mathbf{q}, \dot{\mathbf{q}})$ is defined such that $\mathbf{Y}_f(\mathbf{q}, \dot{\mathbf{q}}) = \gamma \mathbf{Y}_m(\mathbf{q}, \dot{\mathbf{q}}) + \frac{\gamma}{s + \gamma} (\mathbf{Y}_h(\mathbf{q}, \dot{\mathbf{q}}) - \mathbf{Y}_{mf}(\mathbf{q}, \dot{\mathbf{q}}))$. Thus, the filtered dynamics model is given as below:

$$\mathbf{Y}_f(\mathbf{q}, \dot{\mathbf{q}}) \boldsymbol{\pi} = \mathbf{f}_f, \quad (22)$$

From now on the filtered regressor $\mathbf{Y}_f(\mathbf{q}, \dot{\mathbf{q}})$ given in (22) will replace the conventional regressor $\mathbf{Y}(\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}})$, as defined in (3), for the subsequent analyses and computations.

B. Identifiable Parameters Tracking Algorithm (IPTA)

We consider collecting windows of samples, each consisting of n_s samples. For every n_s samples, $\mathbf{Y}_f(\mathbf{q}, \dot{\mathbf{q}})$ and $\mathbf{f}_f$ are used to construct $\mathbf{Y}_{f,w}(\mathbf{q}, \dot{\mathbf{q}})$, and $\mathbf{f}_{f,w}$ for each window $w \in \{1, 2, \dots, W - 1, W\}$, where w is the window index and W is the total number of windows. Consequently, a larger regressor matrix $\bar{\mathbf{Y}}_{f,w}$ and filtered force vector $\bar{\mathbf{f}}_{f,w}$ are updated as follows:

$$\bar{\mathbf{Y}}_{f,w} \triangleq \begin{bmatrix} \bar{\mathbf{Y}}_{f,w-1} \\ \mathbf{Y}_{f,w} \end{bmatrix}, \quad \bar{\mathbf{f}}_{f,w} \triangleq \begin{bmatrix} \bar{\mathbf{f}}_{f,w-1} \\ \mathbf{f}_{f,w} \end{bmatrix},$$

where

$$\bar{\mathbf{Y}}_{f,w} \boldsymbol{\pi} = \bar{\mathbf{f}}_{f,w}. \quad (23)$$

The large regressor $\bar{\mathbf{Y}}_{f,w}$ stores the computed regressor $\mathbf{Y}_{f,w}(\mathbf{q}, \dot{\mathbf{q}})$ for each window w . Similarly, $\bar{\mathbf{f}}_{f,w}$ accumulates the corresponding filtered force vector $\mathbf{f}_{f,w}$ for each window w . In this paper, an overline is used to denote variables associated with the large accumulated regressor $\bar{\mathbf{Y}}_{f,w}$, the force vector across multiple windows $\bar{\mathbf{f}}_{f,w}$, the permutation matrix $\bar{\mathbf{P}}_w$, and the identifiable parameter vector $\bar{\boldsymbol{\pi}}_w$ identified through $\bar{\mathbf{Y}}_{f,w}$.

In the proposed approach, we do not directly utilize (23) in our optimization procedure to obtain the estimated inertial parameters $\hat{\boldsymbol{\pi}}$. We propose a sub-algorithm named Identifiable Parameters Tracking Algorithm (IPTA) to improve the optimization robustness and computation overhead. Unlike conventional optimization algorithms, IPTA enables the estimation of only the identifiable parameters, including both the base and linearly coupled components. In contrast to retaining unidentifiable parameters and optimizing the conventional torque/force residual, the resulting algorithm primarily improves the regressor condition number, leading to more robust optimization features.

IPTA starts by removing the non-identifiable parameters and excluding their corresponding columns from $\mathbf{Y}_{f,w}(\mathbf{q}, \dot{\mathbf{q}})$. Next, the columns corresponding to the base and linearly coupled identifiable parameters are rearranged using the following decomposition method [21]:

$$\mathbf{Y}_{f,w} \mathbf{P}_w = \mathbf{Y}_{i,w}. \quad (24)$$

The matrix $\mathbf{P}_w$ is a permutation matrix that is used to select the identifiable parameters and rearrange the columns of $\mathbf{Y}_{f,w}$ into the identifiable parameters regressor $\mathbf{Y}_{i,w}$. In the second step, the identifiable parameters regressor for the window w (24) is preconditioned via QR decomposition $QR(\mathbf{Y}_{i,w})$. Using (10), (11), and, (13) for every $\mathbf{Y}_{i,w}$, the corresponding $\bar{\boldsymbol{\Phi}}_w$ and $\bar{\mathbf{f}}_w$ are obtained. Consequently a large preconditioned regressor $\bar{\boldsymbol{\Phi}}_w$ and force vector $\bar{\mathbf{f}}_w$ can be updated as follows,

$$\bar{\boldsymbol{\Phi}}_w \triangleq \begin{bmatrix} \bar{\boldsymbol{\Phi}}_{w-1} \\ \boldsymbol{\Phi}_w \mathbf{P}_w \end{bmatrix}, \quad \bar{\mathbf{f}}_w \triangleq \begin{bmatrix} \bar{\mathbf{f}}_{w-1} \\ \mathbf{f}_w \end{bmatrix}, \quad (25)$$

when there is no change in the identifiable parameters. However, if new identifiable parameters emerge from the most recent window of data, a large preconditioned regressor $\bar{\boldsymbol{\Phi}}_w$ is defined as follows:

$$\bar{\boldsymbol{\Phi}}_w \triangleq \begin{bmatrix} \bar{\boldsymbol{\Phi}}_{w-1} & \mathbf{0} \\ \boldsymbol{\Phi}_w \mathbf{P}_w \end{bmatrix}, \quad (26)$$

where

$$\bar{\boldsymbol{\Phi}}_w \bar{\boldsymbol{\pi}}_w = \bar{\mathbf{f}}_w. \quad (27)$$

The vector $\bar{\boldsymbol{\pi}}_w$ stores all identifiable parameters across all $\boldsymbol{\Phi}_k$ for $k \in \{1, 2, \dots, w - 1, w\}$. The projection matrix $\bar{\mathbf{P}}_w$ is used to rearrange the columns of $\boldsymbol{\Phi}_w$ concerning the previous preconditioned regressors. If some parameters are non-identifiable in the final window $\boldsymbol{\Phi}_w$ but were identifiable in previous windows $\bar{\boldsymbol{\Phi}}_{w-1}$, the matrix $\bar{\mathbf{P}}_w$ is used to generate zero columns corresponding to these parameters in $\boldsymbol{\Phi}_w \bar{\mathbf{P}}_w$. The projection matrix $\bar{\mathbf{P}}_w$ and vector $\bar{\boldsymbol{\pi}}_w$ are then formulated to satisfy the following equalities.

$$\bar{\boldsymbol{\Phi}}_w \bar{\mathbf{P}}_w \bar{\boldsymbol{\pi}}_w = \bar{\boldsymbol{\Phi}}_w \boldsymbol{\pi}_{i,w}, \quad (28)$$

$$[\bar{\Phi}_{w-1} \mathbf{0}] \bar{\pi}_w = \bar{\Phi}_{w-1} \bar{\pi}_{w-1}, \quad (29)$$

The vector $\pi_{i,w}$ stores all the parameters identifiable by Φ_w . The matrix $\mathbf{0}$ is a zero matrix of proper size and is omitted when there is no change in the identifiable parameters. Using (28) and (29), the vector of identifiable parameters is updated as follows:

$$\bar{\pi}_w = [\bar{\pi}_{w-1}^T \pi_{w/w-1}^T]^T, \quad (30)$$

where $\pi_{w/w-1}$ represents the parameters that become identifiable in Φ_w but were not identifiable in $\bar{\Phi}_{w-1}$.

For each window w , an optimization problem is solved to obtain the estimated identifiable parameters, namely $\hat{\pi}_w$. After multiple optimization runs, it is crucial to determine which parameters stored in $\bar{\pi}_w$ correspond to the vector of predefined parameters, namely π . To achieve this, a selection matrix $\mathcal{P}_w$ is constructed as follows:

$$\bar{\pi}_w = \mathcal{P}_w^T \pi. \quad (31)$$

The vector of optimal inertial parameters, $\hat{\pi}$, is then obtained using a specially developed function, $Aug(\bullet)$, which merges the initial guess π_0 for the non-identifiable parameters with the vector $\mathcal{P}_w \hat{\pi}_w$. This yields $\hat{\pi} = Aug(\mathcal{P}_w, \hat{\pi}_w, \pi_0)$.

1) Recursive Construction of Projection Matrices

Using (28)-(31) and the permutation matrix $\mathbf{P}_w$, we developed a recursive algorithm to compute $\bar{\mathbf{P}}_w$, $\mathcal{P}_w$, and $Aug(\bullet)$. The matrix $\mathbf{P}_w$ is given by the relation between $\pi_{i,w}$ and π as follows:

$$\pi_{i,w} = \mathbf{P}_w^T \pi. \quad (32)$$

A set of remarks can be made to characterize $\mathbf{P}_w$ [16]:

$\mathcal{R}1$: $\mathbf{P}_w$ has dimensions $\mathbb{R}^{10n_b \times n_i}$, where $n_i \leq 10n_b$.

$\mathcal{R}2$: $\mathbf{P}_w$ is a binary matrix in which each column contains exactly one entry equal to 1, while each row either has a single 1 or consists entirely of zeros.

$\mathcal{R}3$: The indices of the zero rows correspond to the non-identifiable parameter indices in π .

$\mathcal{R}4$: $\mathbf{P}_w \mathbf{P}_w^T$ is a diagonal matrix with entries of either 1 or 0, where 1s indicate identifiable parameters in π , and 0s correspond to non-identifiable ones.

The matrices $\mathcal{P}_w$ and $\bar{\mathbf{P}}_w$ are two selection matrices which must satisfy the remarks $\mathcal{R}2$ - $\mathcal{R}4$. Considering this and (28), we used the matrix $\bar{\mathbf{P}}_w$ to select the elements of the vector $\pi_{i,w}$ from $\bar{\pi}_w$, which are then arranged to correspond with the columns of the matrix Φ_w . Thus, the matrix $\bar{\mathbf{P}}_w$ can be represented as:

$$\pi_{i,w} = \bar{\mathbf{P}}_w \bar{\pi}_w. \quad (33)$$

From (32) and (33), the following equation can be concluded:

$$\mathbf{P}_w^T \pi = \bar{\mathbf{P}}_w \bar{\pi}_w. \quad (34)$$

From (30) and (31), a recursive projection matrix to project $\bar{\pi}_w$ onto π can be formulated. The vector $\bar{\pi}_{w-1}$ in (30) is projected onto π using $\mathcal{P}_{w-1}$, while the vector $\pi_{w/w-1}$ is projected onto π using the selection matrix Γ_w whose transpose, when multiplied by π , yields $\pi_{w/w-1}$. Therefore, the projection matrix $\mathcal{P}_w$ is updated as follows:

$$\mathcal{P}_w = [\mathcal{P}_{w-1} \quad \Gamma_w]. \quad (35)$$

From (31), (34), and (35) the matrix $\bar{\mathbf{P}}_w$ is obtained as follows:

$$\bar{\mathbf{P}}_w = \mathbf{P}_w^T \mathcal{P}_w. \quad (36)$$

The function $Aug(\bullet)$ is utilized with (31) and the remark $\mathcal{R}4$ as below:

$$Aug(\mathcal{P}_w, \bar{\pi}_w, \pi_0) = \mathcal{P}_w \bar{\pi}_w + D_C(\mathcal{P}_w) \pi_0, \quad (37)$$

where $D_C(\bullet)$ returns a diagonal matrix whose diagonal elements are the complement of those in $\mathcal{P}_w \mathcal{P}_w^T$.

For the first window ($w = 1$), we have $\bar{\mathbf{P}}_1 = \mathbf{1}$, and $\mathcal{P}_1 = \mathbf{P}_1$. Accordingly, the IPTA algorithm is formulated as described in Algorithm 1.

Algorithm 1: IPTA($\mathbf{Y}_{f,w}$, $\mathbf{f}_{f,w}$, $\mathcal{P}_{w-1}$)

- 1 Perform the decomposition: $\mathbf{Y}_{f,w} \mathbf{P}_w = \mathbf{Y}_{i,w}$ [21].
 - 2 Implement the eqs. (10), (11), and (13) to the decomposed regressor $\mathbf{Y}_{i,w}$ to obtain Φ_w and $\mathbf{f}_w$.
 - 3 Calculate the matrix $\mathcal{P}_w$ using (35).
 - 4 Compose the matrix $\bar{\mathbf{P}}_w$ using (36).
 - 5 return $(\Phi_w, \mathbf{f}_w, \bar{\mathbf{P}}_w, \mathcal{P}_w)$.
-

C. Recursive Optimization

By leveraging the recursive structure of $\bar{\Phi}_w$ and $\bar{\mathbf{f}}_w$, the optimization problem in (12) can be formulated recursively. Using the equality in (27), (12) can be expressed as follows:

$$\hat{\pi}_w = \underset{\bar{\pi}_w \in \mathbb{R}^{\mathcal{I}_w}}{\operatorname{argmin}} \frac{1}{2} \|\bar{\Phi}_w \bar{\pi}_w - \bar{\mathbf{f}}_w\|^2 \quad \text{s.t. } \mathbf{J}_b(\bar{\pi}_{w,b}) + \mathbf{J}_b^0 > 0, \quad b \in \{1, 2, \dots, n_b\}. \quad (38)$$

where $\mathcal{I}_w$ refers to the size of identifiable parameter sets by the augmented regressor $\bar{\Phi}_w$. The optimization problem given in (38) can be transformed into the following quadratic form:

$$\hat{\pi}_w = \underset{\bar{\pi}_w \in \mathbb{R}^{\mathcal{I}_w}}{\operatorname{argmin}} \frac{1}{2} (\bar{\pi}_w)^\top \mathbf{Q}_w \bar{\pi}_w + \mathbf{c}_w^\top \bar{\pi}_w \quad \text{s.t. } \mathbf{J}_b(\bar{\pi}_{w,b}) + \mathbf{J}_b^0 > 0, \quad b \in \{1, 2, \dots, n_b\}. \quad (39)$$

Using (25) and the objective function in (38), the matrix $\mathbf{Q}_w \in \mathbb{R}^{\mathcal{I}_w \times \mathcal{I}_w}$, and the vector $\mathbf{c}_w \in \mathbb{R}^{\mathcal{I}_w}$, are updated recursively as follows:

$$\mathbf{Q}_w = \mathbf{Q}_{w-1} + (\Phi_w \bar{\mathbf{P}}_w)^\top \Phi_w \bar{\mathbf{P}}_w \quad (40)$$

$$\mathbf{c}_w = \mathbf{c}_{w-1} - \bar{\mathbf{f}}_w^\top \Phi_w \bar{\mathbf{P}}_w, \quad (41)$$

when there is no change in the identifiable parameters. If the number of excited parameters increases, the matrix $\mathbf{Q}_w$ is updated using (26) as follows:

$$\mathbf{Q}_w = \begin{bmatrix} \mathbf{Q}_{11} & \mathbf{Q}_{12} \\ \mathbf{Q}_{21} & \mathbf{Q}_{22} \end{bmatrix} + \epsilon \begin{bmatrix} \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{1}_{\delta_w} \end{bmatrix} \quad (42)$$

$$\mathbf{Q}_{11} = \mathbf{Q}_{w-1} + ((\Phi_w \bar{\mathbf{P}}_w)^\top \Phi_w \bar{\mathbf{P}}_w)[1 : \mathcal{I}_{w-1}][1 : \mathcal{I}_{w-1}] \quad (43)$$

$$\mathbf{Q}_{12} = ((\Phi_w \bar{\mathbf{P}}_w)^\top \Phi_w \bar{\mathbf{P}}_w)[1 : \mathcal{I}_{w-1}][\mathcal{I}_{w-1} + 1 : \mathcal{I}_w] \quad (44)$$

$$\mathbf{Q}_{21} = ((\Phi_w \bar{\mathbf{P}}_w)^\top \Phi_w \bar{\mathbf{P}}_w)[\mathcal{I}_{w-1} + 1 : \mathcal{I}_w][1 : \mathcal{I}_{w-1}] \quad (45)$$

$$\mathbf{Q}_{22} = ((\Phi_w \bar{\mathbf{P}}_w)^\top \Phi_w \bar{\mathbf{P}}_w)[\mathcal{I}_{w-1} + 1 : \mathcal{I}_w][\mathcal{I}_{w-1} + 1 : \mathcal{I}_w] \quad (46)$$

where $\delta_w = \mathcal{I}_w - \mathcal{I}_{w-1}$. The square brackets in (43)-(46) denote the matrix rows and columns, e.g., $\mathbf{A}[r_1 : r_n][c_1 : c_n]$

corresponds to matrix $\mathbf{A}$ from row r_1 to r_n and from column c_1 to c_n . The parameter ϵ is a small positive scalar value. For the first window, we have the following:

$$\mathbf{Q}_1 = (\Phi_1 \bar{\mathbf{P}}_1)^\top \Phi_1 \bar{\mathbf{P}}_1 + \epsilon \mathbf{1}_{\mathcal{I}_1}. \quad (47)$$

The last term in (47) is added to prevent singularity and acts similarly to regularization, pushing the estimate toward a zero guess. The structure of $\mathbf{Q}_w$ can be exploited to recursively compute $\mathbf{Q}_w^{-1}$ as function of $\mathbf{Q}_{w-1}^{-1}$. The algorithms used, as described in [36], are based on the Sherman-Morrison-Woodbury formula, which allows the recursive computation of the inverse of $\mathbf{Q}_w$. When the matrix order remains unchanged, the inverse update can be written compactly as follows:

$$\mathbf{Q}_w^{-1} = \text{recI}(\mathbf{Q}_{w-1}^{-1}, (\Phi_w \bar{\mathbf{P}}_w)). \quad (48)$$

Whereas in the presence of an increase in matrix order, the update is expressed as below:

$$\begin{aligned} \mathbf{Q}_w^{-1} &= \text{Add}(\mathbf{Q}_{11}^{-1}, \mathbf{Q}_w), \\ \mathbf{Q}_{11}^{-1} &= \text{recI}(\mathbf{Q}_{w-1}^{-1}, (\Phi_w \bar{\mathbf{P}}_w)). \end{aligned} \quad (49)$$

The explicit algorithmic realizations of these recursive updates are provided in the Supplementary Material.

D. Recursive SDP Interior Point Method (SDP-IPM)

1) SDP Interior Point Method

A recursive semidefinite programming framework is developed using the interior point method to solve (39) recursively, leveraging the recursive inversion formulae from (48) and (49); see [37], [38]. The inequality constraint given in (39), can be replaced by a block diagonal matrix: $\mathbf{J}_P^D(\bar{\pi}_w) = \text{diag}(\mathbf{J}_1(\bar{\pi}_{1,w}), \mathbf{J}_2(\bar{\pi}_{2,w}), \dots, \mathbf{J}_{n_b}(\bar{\pi}_{n_b,w}))$, where $\mathbf{J}_P^D(\bar{\pi}_{i,w}) \in \mathbb{R}^{4n_b \times 4n_b}$. By exploiting the LMI form, the matrix $\mathbf{J}_P^D(\bar{\pi}_w)$ can be represented linearly in terms of the components of $\bar{\pi}_w$ components as follows:

$$\mathbf{J}_P^D(\bar{\pi}_w) = \sum_{j=1}^{\mathcal{I}_w} \bar{\pi}_{w,j} \mathbf{G}_j \quad (50)$$

where, $\bar{\pi}_{j,w}$ represents the j^{th} component in $\bar{\pi}_w$. The set of $\mathbf{G}_j \in \mathbb{R}^{4n_b \times 4n_b} \forall j \in \{1, \dots, \mathcal{I}_w\}$ are linearly independent symmetric matrices.

Using (50), the optimization problem in (39) can be rewritten as:

$$\hat{\pi}_w = \underset{\bar{\pi}_w \in \mathbb{R}^{\mathcal{I}_w}}{\text{argmin}} \frac{1}{2} (\bar{\pi}_w)^\top \mathbf{Q}_w \bar{\pi}_w + \mathbf{c}_w^\top \bar{\pi}_w \quad (51)$$

$$\text{s.t.} \quad \sum_{j=1}^{\mathcal{I}_w} \bar{\pi}_{w,j} \mathbf{G}_j + \mathbf{J}_0^D = \mathbf{Z} \quad (52)$$

$$\mathbf{Z} \succ \mathbf{0} \quad (53)$$

where, $\mathbf{Z} \in \mathbb{R}^{4n_b \times 4n_b}$ represents a slack variable matrix. $\mathbf{J}_0^D \in \mathbb{R}^{4n_b \times 4n_b}$ is a block diagonal matrix given as: $\mathbf{J}_0^D = \text{diag}(\mathbf{J}_1^0, \mathbf{J}_2^0, \dots, \mathbf{J}_{n_b}^0)$. The Lagrangian associated with (51)-(53) can be computed as in the following:

$$\begin{aligned} \mathcal{L} &= \frac{1}{2} \bar{\pi}_w^\top \mathbf{Q}_w \bar{\pi}_w + \mathbf{c}_w^\top \bar{\pi}_w - \\ &\quad \left\langle \sum_{j=1}^{\mathcal{I}_w} \bar{\pi}_{w,j} \mathbf{G}_j + \mathbf{J}_0^D - \mathbf{Z}, \mathbf{X} \right\rangle - \mu \log \det(\mathbf{Z}), \end{aligned} \quad (54)$$

where a log-determinant barrier function, $\log \det(\bullet)$, is employed to enforce positive definiteness of $\mathbf{Z}$. The function $\langle \mathbf{A}, \mathbf{B} \rangle$ is the trace function of the product of any two arbitrary symmetric matrices $\mathbf{A}$ and $\mathbf{B}$. The matrix $\mathbf{X} \in \mathbb{R}^{4n_b \times 4n_b}$ is a Lagrange multiplier matrix that is positive definite and symmetric.

The parameter μ is a positive scalar that is introduced to incorporate the barrier function into the objective function. Initially, μ is set to a large value to prioritize resolution of the barrier function, preventing $\mathbf{Z}$ from approaching its boundary. Subsequently, during the optimization iterations, μ is gradually reduced to allow the algorithm to focus on minimizing the original objective function. Using (54), the first-order optimality conditions are obtained as follows:

$$\begin{aligned} \nabla_{\pi} \mathcal{L} &= \mathbf{Q}_w \bar{\pi}_w + \mathbf{c}_w - [\langle \mathbf{G}_1, \mathbf{X} \rangle \langle \mathbf{G}_2, \mathbf{X} \rangle \dots \langle \mathbf{G}_{\mathcal{I}_w}, \mathbf{X} \rangle]^\top \\ &= \mathbf{r}_{\pi} \end{aligned} \quad (55)$$

$$\nabla_{\mathbf{X}} \mathcal{L} = - \left(\sum_{j=1}^{\mathcal{I}_w} \bar{\pi}_{w,j} \mathbf{G}_j + \mathbf{J}_0^D \right) + \mathbf{Z} = \mathbf{R}_{\mathbf{X}} \quad (56)$$

$$\nabla_{\mathbf{Z}} \mathcal{L} = -\mu \mathbf{Z}^{-1} + \mathbf{X} = \mathbf{R}_{\mathbf{Z}} \quad (57)$$

where, $\mathbf{r}_{\pi} \in \mathbb{R}^{\mathcal{I}_w}$ is a vector that represent the residuals of the first optimality condition. The matrices $\mathbf{R}_{\mathbf{X}} \in \mathbb{R}^{4n_b \times 4n_b}$ and $\mathbf{R}_{\mathbf{Z}} \in \mathbb{R}^{4n_b \times 4n_b}$ are the residuals of the second and third optimality conditions. In the literature, the condition (57) is referred as the complementary condition, and the method used to solve it dictates the optimization approach [38]. To ensure that the optimization variables follow a central path satisfying the complementary condition, the matrices $\mathbf{X}$ and $\mathbf{Z}$ must satisfy the centering condition, which is given by:

$$\mathbf{XZ} = \mu \mathbf{1}_{4n_b}. \quad (58)$$

To this end, the matrices $\mathbf{X}$ and $\mathbf{Z}$ must commute along the central path [37]. In this work, the $\mathbf{XZ} + \mathbf{ZX}$ method is employed for the direct application of Newton's method [38]. In this approach, the complementary condition given by (57) is replaced by the following equality:

$$\mathbf{R}_{\mathbf{Z}} = \mathbf{XZ} + \mathbf{ZX} - 2\mu \mathbf{1}_{4n_b} \quad (59)$$

which satisfies the centering condition given in (58). To obtain a Karush-Kuhn-Tucker (KKT) system, the function $\mathbf{a} = \text{svec}(\mathbf{A})$ is introduced. This function maps an arbitrary symmetric matrix $\mathbf{A} \in \mathbb{R}^{n \times n}$ to a vector $\mathbf{a} \in \mathbb{R}^{\frac{n(n+1)}{2}}$. $\mathbf{A} = \text{smat}(\mathbf{a})$ is the inverse of $\text{svec}(\bullet)$ [38]; additional details are provided in the Supplementary Material. A Newton step that satisfies the first-order optimality condition for the $\mathbf{XZ} + \mathbf{ZX}$ method is obtained by solving the following KKT system:

$$\begin{bmatrix} \mathbf{Q}_w & -\mathcal{G} & \mathbf{0} \\ -\mathcal{G}^\top & \mathbf{0} & \mathbf{1}_{n_{\text{sym}}} \\ \mathbf{0} & \mathbf{E}_S & \mathbf{F}_S \end{bmatrix} \begin{bmatrix} \Delta \bar{\pi}_w \\ \Delta \mathbf{x} \\ \Delta \mathbf{z} \end{bmatrix} = \begin{bmatrix} -\mathbf{r}_{\pi} \\ -\mathbf{r}_{\mathbf{X}} \\ -\mathbf{r}_{\mathbf{Z}} \end{bmatrix} \quad (60)$$

where the matrix $\mathcal{G} \in \mathbb{R}^{\mathcal{I}_w \times n_{\text{sym}}}$ is yielded as follows: $\mathcal{G} = [\text{svec}(\mathbf{G}_1) \text{svec}(\mathbf{G}_2) \dots \text{svec}(\mathbf{G}_{\mathcal{I}_w})]^\top$. The matrices $\mathbf{E}_S \in \mathbb{R}^{n_{\text{sym}} \times n_{\text{sym}}}$ and $\mathbf{F}_S \in \mathbb{R}^{n_{\text{sym}} \times n_{\text{sym}}}$ are obtained using a symmetric Kronecker product math operation $\otimes_s$ such that:

$$\mathbf{E}_S = \mathbf{Z} \otimes_s \mathbf{1}_{n_{\text{sym}}}, \quad \mathbf{F}_S = \mathbf{X} \otimes_s \mathbf{1}_{n_{\text{sym}}}. \quad (61)$$

The definition of $\otimes_s$ follows [38]; further details are given in

the Supplementary Material. Note that $n_{\text{sym}} = 2n_b(4n_b + 1)$. The residuals $\mathbf{r}_x$, and $\mathbf{r}_z$ are transformed from their equivalent matrix forms as follows:

$$\mathbf{r}_x = \text{svec}(\mathbf{R}_x), \quad \mathbf{r}_z = \text{svec}(\mathbf{R}_z). \quad (62)$$

An interior point method (IPM) based optimization algorithm is employed to solve (51) by using the first order optimality conditions (55), (56), and (59), along with the KKT system (60). This algorithm is outlined in Algorithm 2. For the main-loop stopping condition in step 8, the relative residuals are used; see [37].

$$\tau_\pi = \frac{\|\mathbf{r}_\pi\|}{1 + \|\mathbf{c}_w\|} \quad (63)$$

$$\tau_x = \frac{\|\mathbf{r}_x\|}{1 + \|\mathbf{z}\|} \quad (64)$$

$$\tau_z = \frac{\langle \mathbf{X}, \mathbf{Z} \rangle / 4n_b}{1 + |\frac{1}{2}\bar{\pi}_w^\top \mathbf{Q}_w \bar{\pi}_w + \mathbf{c}_w^\top \bar{\pi}_w|} \quad (65)$$

The barrier matching parameter μ in the third and ninth steps is updated in accordance with the rule provided in [38]:

$$\mu = \delta \langle \mathbf{X}, \mathbf{Z} \rangle / 4n_b, \quad 0 < \delta < 1. \quad (66)$$

2) Line Search Step-Size Optimization Algorithm

A backtracking line-search procedure is employed to determine the optimal step sizes α^* and β^* , as indicated in line 12 of Algorithm 2. The search is invoked only when infeasibility is detected for a nominal step, reducing the number of iterations and the associated computational overhead. Within the main optimization loop, the step sizes, initialized as $\alpha = \alpha_0$ and $\beta = \beta_0$, are iteratively decreased in a nested loop by multiplying with the scaling factor $0 < \nabla < 1$. This nested loop implements a simple line-search technique: at each iteration, the step sizes are reduced whenever $\mathbf{X}$ or $\mathbf{Z}$ is not positive definite, until feasibility is achieved.

3) KKT system Resolution

The core of the recursive least-squares algorithm is to compute the necessary matrix inversions recursively. Therefore, the solution of (60) is determined in terms of $\mathbf{Q}_w^{-1}$, which can be recursively obtained using (48), (49). Given the KKT system in (60), the KKT matrix can be solved through elimination as follows:

$$\mathbf{H} \Delta \bar{\pi}_w = -\mathcal{G}^{-1}(\mathbf{r}_z - \mathbf{F}_S \mathbf{r}_x) - \mathbf{r}_\pi, \quad (67)$$

$$\Delta \mathbf{z} = \mathcal{G} \Delta \bar{\pi}_w - \mathbf{r}_x, \quad (68)$$

$$\Delta \mathbf{x} = -\mathbf{E}_S^{-1}(\mathbf{F}_S \mathcal{G}^\top \Delta \bar{\pi}_w + (\mathbf{r}_z - \mathbf{F}_S \mathbf{r}_x)), \quad (69)$$

where the matrix $\mathbf{H}$ is as formulated as follows: $\mathbf{H} = \mathbf{Q}_w + \mathcal{G} \mathbf{E}_S^{-1} \mathbf{F}_S \mathcal{G}^\top$. The inverse of the matrix $\mathbf{H}$ is obtained using the Kenneth-Miller inverse of sum of matrices algorithm as a function in $\mathbf{Q}_w^{-1}$; see [39]:

$$\mathbf{H}^{-1} = \mathbf{K} \mathbf{M}(\mathbf{Q}_w^{-1}, \mathcal{G} \mathcal{M}), \quad (70)$$

where, the matrix $\mathcal{M}$ is calculated as $\mathcal{M} = \mathbf{E}_S^{-1} \mathbf{F}_S \mathcal{G}^\top$. The explicit recursive realization of this update is provided in the Supplementary Material.

To compute the matrix $\mathcal{M}$, each j^{th} column, denoted as $\mathbf{m}_j$,

Algorithm 2: Semidefinite Programming Interior Point Method (SDP-IPM)

Initialization:

- 1 Feasibly initialize $\bar{\pi}_{w,0}$, $\mathbf{X}_0 > \mathbf{0}$, $\mathbf{Z}_0 > \mathbf{0}$
- 2 Compute the equivalent vector form $\mathbf{x}_0 = \text{svec}(\mathbf{X}_0)$, $\mathbf{z}_0 = \text{svec}(\mathbf{Z}_0)$
- 3 Initialize the barrier parameter $\mu_0 = \langle \mathbf{X}_0, \mathbf{Z}_0 \rangle / 4n_b$
- 4 Obtain the initial residuals $(\tau_\pi^0, \tau_x^0, \tau_z^0)$ using (63), (64), and (65).
- 5 Choose the initial step size, $0 < \alpha_0 < 1$, $0 < \beta_0 < 1$, and barrier reduction parameter, $0 < \delta < 1$
- 6 Select sufficiently small positive scalars as tolerance parameters $(\epsilon_\pi, \epsilon_x, \epsilon_z)$
- 7 Initialize the iteration index $h = 0$

Main Loop:

- 8 **while** $(\|\tau_\pi^h\| > \epsilon_\pi, \|\tau_x^h\| > \epsilon_x, \|\tau_z^h\| > \epsilon_z)$ **do**
 - 9 Update the barrier parameter $\mu_{h+1} = \delta \mu_h$
 - 10 Solve the KKT system in (60) to obtain the Newton step $(\Delta \bar{\pi}_w, \Delta \mathbf{x}, \Delta \mathbf{z})$
 - 11 Compute the equivalent matrices form $\Delta \mathbf{X} = \text{smat}(\Delta \mathbf{x})$, $\Delta \mathbf{Z} = \text{smat}(\Delta \mathbf{z})$
 - 12 Obtain the step size α^* and β^* via backtracking line search.
 - 13 Update the optimization variables $\bar{\pi}_{w,h+1} = \bar{\pi}_{w,h} + \alpha^* \Delta \bar{\pi}_w$
 $\mathbf{z}_{h+1} = \mathbf{z}_h + \alpha^* \Delta \mathbf{z}$
 $\mathbf{x}_{h+1} = \mathbf{x}_h + \beta^* \Delta \mathbf{x}$
 - 14 Compute the equivalent matrices form $\mathbf{X}_{h+1} = \text{smat}(\mathbf{x}_{h+1})$, $\mathbf{Z}_{h+1} = \text{smat}(\mathbf{z}_{h+1})$
 - 15 Obtain new residuals values $(\tau_\pi^{h+1}, \tau_x^{h+1})$ using (55, 56, and 62)
 - 16 Obtain the relative residuals' values $(\tau_\pi^{h+1}, \tau_x^{h+1}, \tau_z^{h+1})$ using (63, 64, and 65)
 - 17 Update the iteration index $h = h + 1$
 - 18 **end**
 - 19 **return** $(\hat{\pi}_w)$
-

is computed individually as follows:

$$\mathbf{Z} \mathbf{M}_j + \mathbf{M}_j \mathbf{Z} = \mathbf{X} \mathbf{G}_j + \mathbf{G}_j \mathbf{X} \quad (71)$$

$$\mathbf{m}_j = \text{svec}(\mathbf{M}_j). \quad (72)$$

The matrix $\mathbf{M}_j$ is obtained by solving the Lyapunov equation (71) using the eigen value method reported in [40].

E. Methodological Synthesis

To implement the proposed approach, the flowchart illustrated in Fig. 1 can be followed.

The Starting Engine block used to provide SDP-IPM with proper initial values. It composes the vector $\bar{\pi}_{w,0}$ using the previously estimated parameters $\hat{\pi}_{w-1}$ and the initial guess π_0 . The matrix $\mathbf{Z}_0$ is obtained using $\bar{\pi}_{w,0}$ and (52). The matrix $\mathbf{X}_0$ is defined as a positive definite matrix that commutes with $\mathbf{Z}_0$.

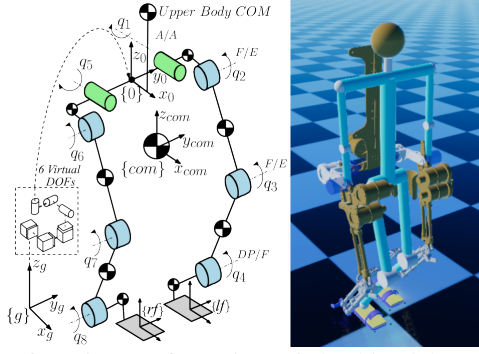


Fig. 2: Left: Joint configuration of the bipedal exoskeleton. Right: RaiSim simulation model.

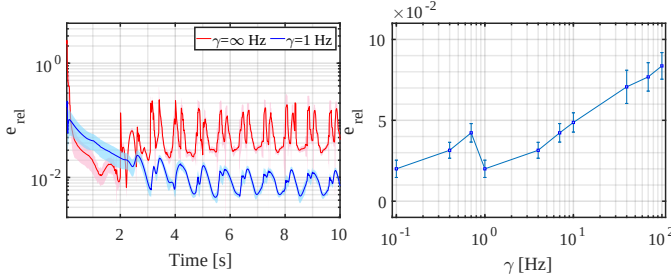


Fig. 3: (a) Relative error e_{rel} with and without filtered regressor. (b) Relative error e_{rel} for different filter cutoff frequencies γ .

were fixed. We denote each human-robot coupled model as M_l , where the subscript l represents the model index.

2) Forward Walking Pattern Generation

Twelve different human-robot coupled models were used to simulate forward dynamic walking motions. In each simulation run, the walking parameters were kept the same for the single and double support periods; they are given as 0.6 and 0.2 s, respectively. The maximum foot clearance was 4 cm.

The identification algorithm was evaluated based on the data collected every sampling period at $1kHz$. Multi-threading functions were used to execute the identification algorithm. The plots in this section are shown with a window time of $10ms$ (i.e., 10 samples per window). The algorithms were implemented using C/C++ and GNU Scientific Library [44].

3) Discussion

This subsection evaluates the efficacy of the proposed algorithm based on simulation results. The feasibility of the applied filtered regressor is depicted in Fig. 3 with a 12 different models performing walking motion with a speed of $0.1m/s$ and with varying cutoff frequencies (γ). The relative error e_{rel} is computed using (22) as below:

$$e_{rel} = \frac{\|\mathbf{Y}_f(\mathbf{q}_{act}, \dot{\mathbf{q}}_{act})\hat{\boldsymbol{\pi}} - \mathbf{f}_f\|}{\|\mathbf{f}_f\|}, \quad (75)$$

The subscript “act” denotes the actual measured signal. Fig. 3.a shows the acquired relative error e_{rel} for the different models as mean $\pm$ SD with shaded regions, and its average variation was denoted via solid lines. The solid blue and red lines show e_{rel} with and without filtered regressor, respectively. Examining Fig 3.a, the application of the filtered regressor significantly reduces the impact of contact transitions on the identification algorithm’s accuracy. To determine an appropri-

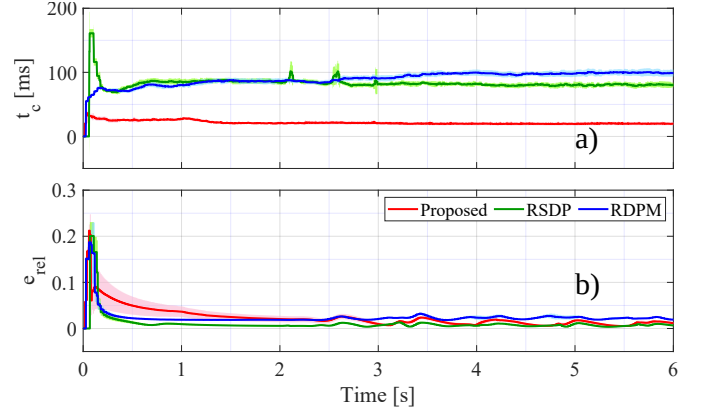


Fig. 4: The computation time t_c and relative error e_{rel} concerning all three methods.

ate value for γ , each model performed the walking motion for 10 different values of γ . The mean $\pm$ SD e_{rel} variation is illustrated in Fig. 3.b, showing that the effect of contact transitions on accuracy decreases as γ decreases. Based on this result, the value of $\gamma = 1 Hz$ is fixed for all subsequent simulation experiments.

Fig. 4 displays computation time, t_c , and relative error, e_{rel} for all three methods. Shaded blue, green, and red regions indicate mean $\pm$ SD variations for RDPM, RSDP, and the proposed method, respectively. Each solid line shows the average of 12 different anthropomorphic models walking with a speed of $0.1 m/s$.

Fig. 4.a illustrates the computation time of the three methods. The proposed method achieved the lowest average computation time ($t_c \approx 22ms$), outperforming both RSDP ($t_c \approx 80ms$) and RDPM ($t_c \approx 94ms$). This highlights the proposed method’s advantage in terms of computational efficiency and real-time applicability. Fig. 4.b presents the average relative error e_{rel} over time. While the proposed method exhibits a slightly slower convergence rate than RSDP, it still reaches the same level of accuracy within approximately three seconds. Overall, these results indicate that the proposed method offers a favorable trade-off between computational effort and convergence rate, making it well-suited for real-time optimization scenarios.

Furthermore, seventy-two simulation runs were conducted using twelve human-robot models walking at six different velocities ranging from 0.08 to 0.13 m/s. The complete set of simulation results, including computation-time and parameter-identification accuracy analyses, is provided in the Supplementary Material and the anonymized online repository¹. Across all simulated subjects, the relative-error RMSE remained low, ranging from 9.3×10^{-3} to 2.0×10^{-2} with standard deviations below 1.7×10^{-3} .

To qualitatively assess the performance of the proposed algorithm, Fig. 5 illustrates the agreement between the estimated joint torque vector $\hat{\boldsymbol{\tau}}$ and the actual joint torques $\boldsymbol{\tau}_{act}$. For comparison, the initial guess torque $\tilde{\boldsymbol{\tau}}$ —computed using the dynamic model with the initial guess of the inertial

¹<https://github.com/AhmedFahmySoliman/online-inertial-parameter-identification-data.git>

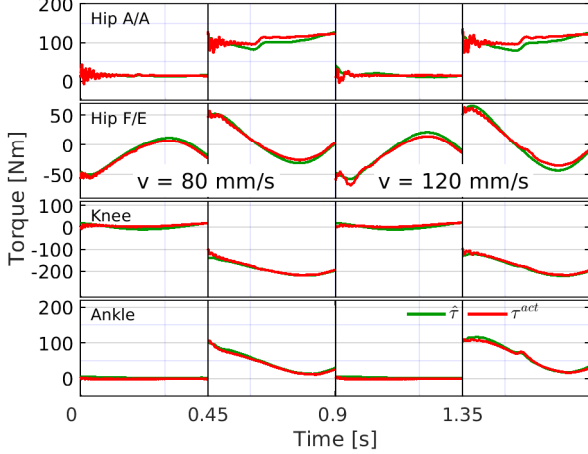


Fig. 5: The estimated (proposed) right leg joint torques $\hat{\tau}$ versus the actual torques τ^{act} .

parameters—is also included. Both $\hat{\tau}$ and $\tilde{\tau}$ are derived by solving the dynamic model equations with the identified and initial inertial parameters, respectively. The corresponding equations are provided below.

$$\begin{aligned}\hat{\tau} &= \mathbf{S}^\top (\hat{\mathbf{f}} - \mathbf{J}_c(\mathbf{q}_{\text{act}})^\top \hat{\lambda}_c), & \tilde{\tau} &= \mathbf{S}^\top (\tilde{\mathbf{f}} - \mathbf{J}_c(\mathbf{q}_{\text{act}})^\top \tilde{\lambda}_c) \\ \hat{\lambda}_c &= (\mathbf{S}_{fb} \mathbf{J}_c(\mathbf{q}_{\text{act}})^\top)^{-1} \mathbf{S}_{fb} \hat{\mathbf{f}}, & \tilde{\lambda}_c &= (\mathbf{S}_{fb} \mathbf{J}_c(\mathbf{q}_{\text{act}})^\top)^{-1} \mathbf{S}_{fb} \tilde{\mathbf{f}} \\ \hat{\mathbf{f}} &= \mathbf{Y}(\mathbf{q}_{\text{act}}, \dot{\mathbf{q}}_{\text{act}}, \ddot{\mathbf{q}}_{\text{ref}}) \hat{\boldsymbol{\pi}}, & \tilde{\mathbf{f}} &= \mathbf{Y}(\mathbf{q}_{\text{act}}, \dot{\mathbf{q}}_{\text{act}}, \ddot{\mathbf{q}}_{\text{ref}}) \boldsymbol{\pi}_0\end{aligned}\quad (76)$$

The vectors $\hat{\mathbf{f}} \in \mathbb{R}^{n_d}$ and $\tilde{\mathbf{f}} \in \mathbb{R}^{n_d}$ denote the proposed and initial guess generalized force vectors, respectively. The matrix $\mathbf{S}_{fb} \in \mathbb{R}^{6 \times n_d}$ selects the first six components of these vectors. The projected generalized contact force vectors onto the contact foot are represented by $\hat{\lambda}_c \in \mathbb{R}^6$ and $\tilde{\lambda}_c \in \mathbb{R}^6$. The current analysis is limited to the single support phase, during which the contact force has a dimension of 6×1 . This results in a square matrix $(\mathbf{S}_{fb} \mathbf{J}_c(\mathbf{q}_{\text{act}})^\top)^{-1}$, allowing for a well-defined direct inverse. The subscript “ref” and the superscript “act” are used to denote the reference and actual signals, respectively.

Fig. 5 presents the averaged torque data over 20 steps for the right leg across four single support phases. The four subplots correspond to the hip abduction/adduction torque, hip flexion/extension torque, knee torque, and ankle torque, arranged from top to bottom, respectively. Each subplot is divided into four columns: the first two represent the left and right single support phases at a forward velocity of $v = 0.08$ m/s, while the latter two show the corresponding phases at $v = 0.12$ m/s. The estimated torques $\hat{\tau}$ obtained using the proposed method are shown as solid green lines, the actual torques τ^{act} are shown in solid red. Fig. 5 illustrates a noticeable agreement between the estimated and actual torques.

C. Experimental results

The proposed identification algorithm is verified experimentally using a real-life exoskeleton, which is custom-built; see Fig. 6. The exoskeleton is equipped with eight series elastic actuators integrated into the active joints. Each joint features two encoders: one for measuring the joint angular position and

another for measuring spring deflection. Joint torques were obtained from measurements of deflection in calibrated stiff springs and pre-processed to compensate for sensor bias, drift, and scaling effects [45]. To capture ground reaction forces, a distributed architecture of force sensing elements is implemented across the foot sole plane, providing measurements of the vertical contact force component and the sagittal- and frontal-plane moments. Additionally, an inertial measurement unit (IMU) mounted on the robot’s pelvis provides measurements of angular velocity and linear acceleration of the floating base. For further details regarding this exoskeleton, refer to [30].

Due to the lack of direct measurements of the complete contact wrench, the contact-consistent dynamics formulation proposed in [46], [47] was adopted for the hardware experiments. Specifically, the dynamic model introduced in Section II-A was projected onto the null space of the contact Jacobian prior to executing the identification algorithm, yielding $\mathbf{N}(\mathbf{M}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{h}(\mathbf{q}, \dot{\mathbf{q}})) = \mathbf{N}\mathbf{S}^\top \boldsymbol{\tau}$. The matrix $\mathbf{N} = \mathbf{1} - \mathbf{J}_c^\dagger \mathbf{J}_c$ denotes the orthogonal projector onto the null space of the contact Jacobian $\mathbf{J}_c$ and $\tilde{\boldsymbol{\tau}} = \mathbf{N}\mathbf{S}^\top \boldsymbol{\tau}$ represents the null-space-projected joint torque vector. Accordingly, the contact wrench contribution is eliminated from the equations of motion. Consequently, the resulting regression model depends only on the measured joint torques and kinematic quantities, thereby enabling inertial parameter identification without direct contact-force measurements.

A total of 30 hardware experiments were conducted using three dummy configurations and two representative motion patterns: squatting and swaying. The squatting motions consisted of peak-to-peak vertical torso displacements of 6, 7, 8, and 10 cm performed at frequencies ranging from 0.2 Hz to 0.5 Hz. The swaying motions were executed at frequencies of 0.25 Hz, 0.33 Hz, and 0.5 Hz with torso roll amplitudes of 5°, 6°, and 7°. To evaluate the robustness of the identification framework under varying inertial conditions, the three dummy configurations corresponded to approximate total human–robot system masses ranging from 65 kg to 95 kg.

Since both robot feet remain in contact with the ground during the squatting and swaying experiments, the resulting redundancy in contact-force computation renders accurate estimation of the true contact forces infeasible [48]. Consequently, the experimental assessment is performed using the null-space-projected joint torque vector, denoted by $\tilde{\boldsymbol{\tau}}$. Fig. 7 presents the estimated null-space-projected joint torques, $\tilde{\boldsymbol{\tau}}^{\text{est}}$ and the measured null-space-projected joint torques, $\tilde{\boldsymbol{\tau}}^{\text{act}}$ shown by the solid green and red lines, respectively, for a representative squatting experiment. The results demonstrate that the estimated torques closely track the measured torques throughout the motion. The corresponding torque-matching results for the remaining hardware experiments are provided in the anonymized online repository¹.

Fig. 8 compares the proposed method, shown in red, against the benchmark RSDP method, shown in green. The RDPM method is excluded from this comparison because of its poor convergence behavior, which is quantitatively analyzed in Sections IV-D and IV-E. Fig. 8(a) presents the average absolute torque error across all squatting and swaying experiments.

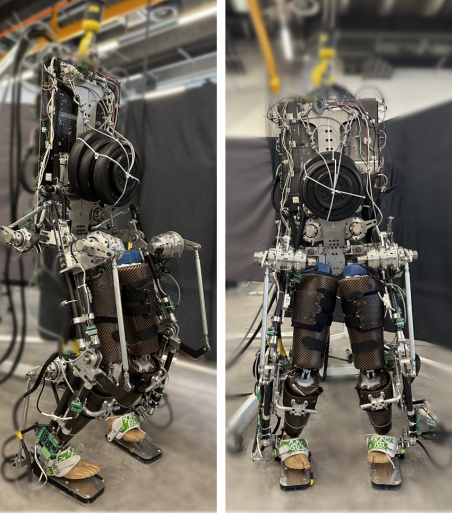


Fig. 6: Bipedal exoskeleton used for experimental verification.

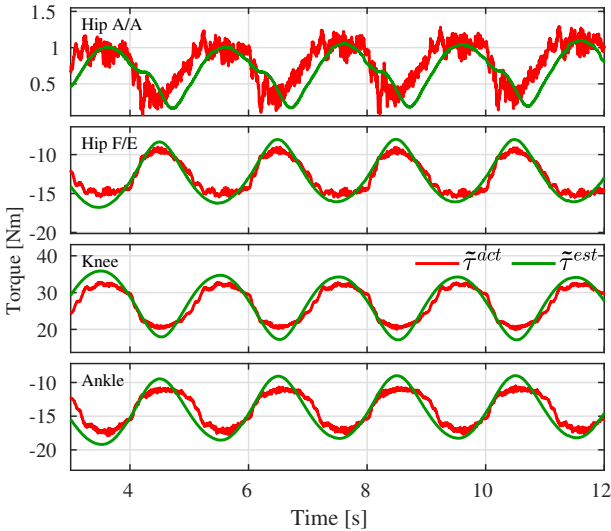


Fig. 7: Estimated (proposed) right leg null-space-projected joint torques $\hat{\tau}^{est}$ versus the measured null-space-projected joint torques $\hat{\tau}^{act}$.

The mean absolute torque error, denoted by $\bar{e}_\tau$, is represented by the solid curve, while the shaded region corresponds to $\bar{e}_\tau \pm \sigma(e_\tau)$. The absolute torque error for joint t is defined as $e_t = |\hat{\tau}_t^{est} - \hat{\tau}_t^{act}|$. The mean and standard deviation of the absolute torque error are computed as follows:

$$\bar{e}_\tau = \frac{\sum_{t=1}^8 e_t}{8}, \quad \sigma(e_\tau) = \text{std}([e_1, \dots, e_8]). \quad (77)$$

The function $\text{std}(\bullet)$ denotes the standard deviation of the input vector. From Fig. 8a, it is evident that the error convergence exhibits stable performance in the experimental data analysis.

The average computation time t_c for both experiments is shown in Fig. 8b. The mean t_c is represented by the solid blue line, while the mean $\pm$ SD is indicated by the shaded area. The results demonstrate that the computation time remains below 50 ms, making it suitable for online applications.

For completeness, the variation of the identified inertial parameters for both simulation and experimental results is presented in the Supplementary Material.

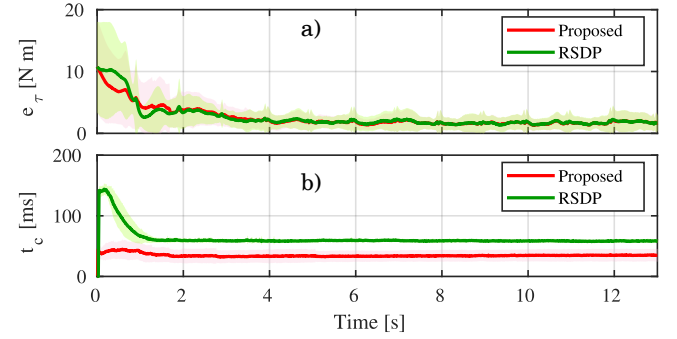


Fig. 8: a) Robot joints absolute torque error experimental result. b) Computation time t_c experimental result.

D. Performance Summary

Table I summarizes the overall performance of the investigated identification methods across the hardware squatting/swaying experiments and the walking simulation experiments. The reported values correspond to the mean and standard deviation of the torque RMSE, optimization computation time, and condition number obtained over all experimental trials.

For the hardware experiments, the proposed method achieved the lowest torque RMSE while requiring substantially lower computation time than RSDP and yielding significantly better numerical conditioning than both benchmark methods. In contrast, RDPM exhibited considerably larger torque RMSE and condition numbers, indicating reduced identification accuracy and numerical robustness.

For the walking simulation experiments, the proposed method achieved the lowest torque RMSE and computation time among all considered methods. Furthermore, the proposed method consistently produced the smallest condition numbers in both hardware and simulation studies, demonstrating improved numerical conditioning of the optimization problem. These results indicate that the proposed approach provides an effective balance between identification accuracy, computational efficiency, and numerical robustness.

Figure 9 illustrates the convergence behaviour and temporal evolution of the identified inertial parameters for representative simulation and hardware experiments. The inertial parameter estimates from all methods were processed using the same first-order low-pass filter with a cut-off angular frequency of $\omega_c = 10$ rad/s. The normalized parameter convergence metric is defined as

$$d_\pi(t) = \frac{\|\pi(t) - \pi_f\|_2}{\|\pi_f\|_2},$$

where π_f denotes the average of the final 10% of the identified parameter samples. Figures 9(a) and (c) show the evolution of the convergence metric for the simulation and hardware experiments, respectively, together with the corresponding settling times. Figures 9(b) and (d) present the temporal evolution of five representative inertial parameters identified by the proposed method, namely the masses of the floating base m_{fb} , hip m_h , thigh m_{th} , shank m_{sh} , and foot m_f . These parameters were selected to represent the temporal identification behaviour across different body segments without overcrowding the figure with all 50 inertial parameters. Despite the initial transients and small residual variations, the estimated masses

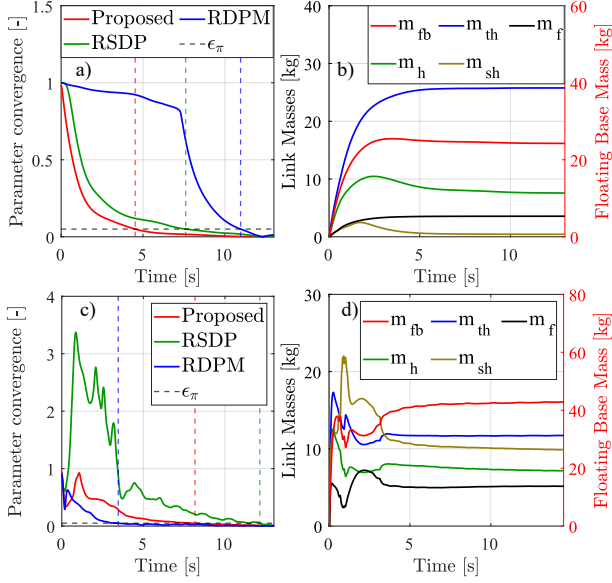


Fig. 9: Representative simulation and hardware experiments. (a,c) Parameter convergence metric $d_\pi(t)$, where the dashed lines indicate the settling times within the $\epsilon_\pi = 0.05$ band. (b,d) Temporal evolution of the floating-base, hip, thigh, shank, and foot masses identified by the proposed method.

TABLE I: Mean $\pm$ SD performance metrics for the hardware and simulation experiments. Lower values indicate better performance. Prop. denotes the proposed method.

Hardware Experiments			
Method	Torque RMSE	Computation Time [ms]	Condition Number
Prop.	2.99 ± 1.31	35.60 ± 9.52	5.94 ± 5.47
RSDP	3.80 ± 1.57	64.10 ± 2.82	$(2.12 \pm 0.90) \times 10^5$
RDPM	15.21 ± 9.05	39.23 ± 33.43	$(1.46 \pm 1.95) \times 10^7$
Walking Simulation Experiments			
Prop.	16.86 ± 0.82	23.12 ± 0.14	4404.6 ± 2.5
RSDP	20.78 ± 0.69	85.34 ± 2.13	$(2.15 \pm 0.01) \times 10^7$
RDPM	36.44 ± 3.90	91.19 ± 3.26	$(3.75 \pm 0.01) \times 10^6$

converge toward physically consistent steady-state values, in agreement with the convergence behaviour indicated by $d_\pi(t)$.

To further characterize the convergence behaviour of the identification algorithms, the average inertial-parameter settling time was also evaluated across all experiments. For the hardware experiments, the proposed method achieved an average settling time of 4.98 ± 1.84 [s], compared with 7.15 ± 1.77 [s] for RSDP and 11.87 ± 1.92 [s] for RDPM. For the walking simulation experiments, the proposed method achieved an average settling time of 7.83 ± 0.59 [s], compared with 11.71 ± 0.63 [s] for RSDP and 3.44 ± 1.00 [s] for RDPM. Although RDPM exhibited faster settling in simulation, this behavior was accompanied by substantially larger torque RMSE values and poorer numerical conditioning. Overall, the proposed method achieved the shortest settling times in the hardware experiments while maintaining superior identification accuracy and numerical robustness.

E. Statistical Analysis

To quantitatively compare the identification methods, statistical analyses were performed using the torque root-mean-square error (RMSE) and optimization computation time as performance metrics. The analyses considered the proposed method and the benchmark methods, RSDP and RDPM, using the results obtained from the squatting/swaying hardware experiments and the walking simulation experiments.

For each experiment, the torque RMSE was computed between the estimated and measured torque quantities and averaged across the considered joints. In the hardware experiments, the RMSE was evaluated using the null-space-projected joint torques, whereas in the walking simulation experiments it was evaluated using the actual joint torques obtained directly from the RAISIM model. Computation time was defined as the time required to execute a single optimization update. These metrics were selected because they directly reflect the identification accuracy and computational efficiency of each method.

A repeated-measures analysis of variance (ANOVA) was conducted to determine whether statistically significant differences existed among the three identification methods for each performance metric. The significance level was set to $\alpha = 0.05$. When a significant main effect was detected, Bonferroni-corrected post-hoc pairwise comparisons were performed to identify statistically significant differences between individual methods.

The results revealed a significant effect of the identification method on both torque RMSE and computation time. For the hardware experiments, significant differences were observed in torque RMSE ($F(2, 58) = 67.55$, $p < 0.001$) and computation time ($F(2, 58) = 23.05$, $p < 0.001$). Similarly, for the walking simulation experiments, significant differences were observed in torque RMSE ($F(2, 22) = 196.53$, $p < 0.001$) and computation time ($F(2, 22) = 4191.4$, $p < 0.001$).

The post-hoc comparisons reported in Tables II and III provide additional insight into the pairwise differences among the methods. In the hardware experiments, the proposed method achieved the lowest torque RMSE, significantly outperforming both RSDP and RDPM. Furthermore, the proposed method required significantly less computation time than RSDP while exhibiting computation times statistically comparable to RDPM. In the walking simulation experiments, the proposed method again achieved the lowest torque RMSE and computation time, significantly outperforming both benchmark methods. These results indicate that the proposed method improves computational efficiency while simultaneously enhancing torque reconstruction accuracy.

Fig. 10 summarizes the performance of the three identification methods using the mean and standard deviation obtained from all hardware and simulation experiments.

V. DISCUSSION CONCLUSION

This paper presented a real-time applicable inertial parameter identification algorithm that provided complete physically consistent estimates of the inertial parameters. The filtered regressor idea from adaptive control [34] was shown to im-

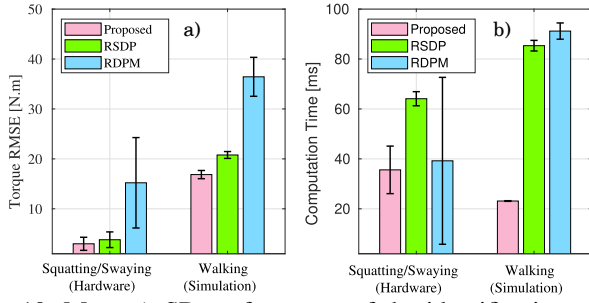


Fig. 10: Mean $\pm$ SD performance of the identification methods: (a) torque RMSE and (b) computation time.

TABLE II: Bonferroni-corrected post-hoc comparisons for the hardware squatting and swaying experiments.

Metric	Comparison	Δ Mean	p -value
Torque RMSE	Prop vs. RDPM	-12.219	< 0.001
	Prop vs. RSDP	-0.813	< 0.001
	RSDP vs. RDPM	-11.406	< 0.001
Computation Time [ms]	Prop vs. RDPM	-3.626	1.000
	Prop vs. RSDP	-28.499	< 0.001
	RDPM vs. RSDP	-24.873	< 0.001

prove the resilience of the method to the effects of contact transitions and signal noise while eliminating the need to estimate joint accelerations through finite differences. Additionally, an identifiable parameters tracking algorithm (IPTA) was developed to improve the regressor's condition number, leading to enhanced optimization features such as robustness and faster convergence rates. A semidefinite programming (SDP) algorithm based on the interior point method approach was modified using recursive inversion to enable a real-time SDP identification procedure.

The proposed algorithm was thoroughly tested and compared with two distinct identification algorithms in a realistic simulation environment and actual experiments. A human bipedal exoskeleton coupled model was used as a case study and simulated via the RaiSim dynamic simulator to verify the proposed method's applicability [4]. For thorough results, we used 12 distinct anthropomorphic models ($66 \sim 102kg$) coupled with a $40kg$ underactuated exoskeleton model. Each model performed dynamic walking at different velocities over several simulation runs. Furthermore, experimental verification was carried out using the actual bipedal exoskeleton performing squatting experiments with different amplitudes. As a result of the benchmarking, the proposed algorithm outperformed the other identification algorithms in terms of robustness and real-time applicability. Both experimental and simulation results demonstrated the complete physical consistency of the resultant inertial parameters obtained while using the proposed method.

In Table IV, a qualitative comparison is provided between recent state-of-the-art methods and the proposed framework. While existing approaches primarily focus on achieving complete physical consistency while maintaining accurate system identification, the literature still lacks a solution capable of operating in real time without requiring a dedicated trajectory excitation phase. Therefore, this table should not be interpreted

TABLE III: Bonferroni-corrected post-hoc comparisons for the walking simulation experiments.

Metric	Comparison	Δ Mean	p -value
Torque RMSE	Prop vs. RDPM	-19.581	< 0.001
	Prop vs. RSDP	-3.925	< 0.001
	RSDP vs. RDPM	-15.656	< 0.001
Computation Time [ms]	Prop vs. RDPM	-68.072	< 0.001
	Prop vs. RSDP	-62.221	< 0.001
	RSDP vs. RDPM	-5.851	< 0.001

TABLE IV: Qualitative comparison between the proposed method and representative inertial parameter identification approaches. RDPM denotes the discrete point-mass formulation of Ayusawa et al. [8] implemented with the proposed recursive optimization framework. The abbreviations RT, NDE, CPC, and TVC, indicate real time, no dedicated excitation, complete physical consistency, and time varying contacts, respectively.

Method	RT	NDE	CPC	TVC
Rucker & Wensing (2022) [13]	×	✓	✓	×
Zhang et al. (2025) [26]	✓	×	✓	×
Martinez et al. (2025) [14]	×	✓	✓	✓
RDPM [8]	✓	✓	✓	✓
Proposed Method	✓	✓	✓	✓

as a direct benchmark comparison; rather, it highlights the research gap addressed by the proposed method. In particular, the proposed framework offers a promising direction for emerging applications such as self-balancing exoskeletons, where real-time operation and minimal user intervention are critical.

A limitation of the proposed method is that it relies on a rigid body representation of the human-exoskeleton system and therefore does not explicitly capture the elastic and viscoelastic behavior of human soft tissues. Consequently, phenomena arising from tissue deformation, compliance at attachment interfaces, and complex contact transitions between the user and the exoskeleton are not directly identified by the proposed framework. Incorporating such effects would require more sophisticated biomechanical models and additional parameters, substantially increasing model complexity and computational burden, which could compromise the real time applicability of the identification process. Despite this limitation, the proposed method provides a physically meaningful estimate of the lumped inertial properties of the coupled human exoskeleton system during nominal operation. For many model based control applications, accurate inertial parameter estimates are of primary importance, and the resulting simplified model may therefore represent a practical tradeoff between physical fidelity and real time usability. Future work will investigate the performance of the proposed approach with human subjects and explore the integration of compliant interaction models to better account for soft tissue dynamics and user specific contact effects.

REFERENCES

- [1] C. Zou, Z. Peng, L. Zhang, F. Mu, R. Huang, and H. Cheng, "Optimization-based adaptive assistance for lower limb exoskeleton

- robots with a robotic walker via spatially quantized gait,” *IEEE Transactions on Automation Science and Engineering*, vol. 22, p. 2806–2818, 2025.
- [2] P. M. Kuber, M. M. Alemi, and E. Rashedi, “A systematic review on lower-limb industrial exoskeletons: Evaluation methods, evidence, and future directions,” *Annals of Biomedical Engineering*, vol. 51, no. 8, p. 1665–1682, May 2023.
 - [3] J. Babič, M. Laffranchi, F. Tessari, T. Verstraten, D. Novak, N. Šarabon, B. Ugurlu, L. Peternel, D. Torricelli, and J. F. Veneman, “Challenges and solutions for application and wider adoption of wearable robots,” *Wearable Technologies*, vol. 2, 2021.
 - [4] A. F. Soliman and B. Ugurlu, “Robust locomotion control of a self-balancing and underactuated bipedal exoskeleton: Task prioritization and feedback control,” *IEEE Robotics and Automation Letters*, vol. 6, no. 3, pp. 5626–5633, 2021.
 - [5] W.-Z. Li, G.-Z. Cao, and A.-B. Zhu, “Review on control strategies for lower limb rehabilitation exoskeletons,” *IEEE Access*, vol. 9, p. 123040–123060, 2021.
 - [6] J. Vantilt, K. Tanghe, M. Afschrift, A. K. Bruijnes, K. Junius, J. Geeroms, E. Aertbeliën, F. De Groote, D. Lefeber, I. Jonkers, and J. De Schutter, “Model-based control for exoskeletons with series elastic actuators evaluated on sit-to-stand movements,” *Journal of NeuroEngineering and Rehabilitation*, vol. 16, no. 1, Jun. 2019.
 - [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, Sep. 1986.
 - [8] K. Ayusawa, G. Venture, and Y. Nakamura, “Real-time implementation of physically consistent identification of human body segments,” in *2011 IEEE International Conference on Robotics and Automation*, 2011, pp. 6282–6287.
 - [9] 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, 2018.
 - [10] J.-A. Ting, M. N. Mistry, J. Peters, S. Schaal, and J. Nakanishi, “A bayesian approach to nonlinear parameter identification for rigid body dynamics,” in *Robotics: Science and systems*. Philadelphia, USA, 2006, pp. 32–39.
 - [11] Y. Huang, J. Ke, X. Zhang, and J. Ota, “Dynamic parameter identification of serial robots using a hybrid approach,” *IEEE Transactions on Robotics*, vol. 39, no. 2, pp. 1607–1621, 2023.
 - [12] G. Evangelisti and S. Hirche, “Data-driven momentum observers with physically consistent gaussian processes,” *IEEE Transactions on Robotics*, vol. 40, pp. 1938–1951, 2024.
 - [13] 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.
 - [14] S. Martinez, R. Griffin, and C. Mastalli, “Multi-contact inertial parameters estimation and localization in legged robots,” *IEEE Robotics and Automation Letters*, vol. 10, no. 5, p. 4730–4737, May 2025.
 - [15] R. Bellman and K. Åström, “On structural identifiability,” *Mathematical Biosciences*, vol. 7, pp. 329–339, 1970.
 - [16] M. Gautier and W. Khalil, “Exciting trajectories for the identification of base inertial parameters of robots,” in *1991 Proceedings of the 30th IEEE Conference on Decision and Control*, 1991, pp. 494–499 vol.1.
 - [17] J. Swevers, C. Ganseman, D. 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.
 - [18] B. F. Khalil W, “Symbolic calculation of the base inertial parameters of closed-loop robots,” *The International Journal of Robotics Research*, vol. 14, no. 2, pp. 112–128, 1995.
 - [19] H. Mayeda, K. Yoshida, and K. Osuka, “Base parameters of manipulator dynamic models,” *IEEE Transactions on Robotics and Automation*, vol. 6, no. 3, pp. 312–321, 1990.
 - [20] P. M. Wensing, G. Niemeyer, and J.-J. E. Slotine, “Observability in inertial parameter identification,” *International Journal of Robotics Research (to appear)*, 2024.
 - [21] M. Gautier, “Numerical calculation of the base inertial parameters of robots,” in *Proceedings., IEEE International Conference on Robotics and Automation*, 1990, pp. 1020–1025 vol.2.
 - [22] 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)*, 2016, pp. 5446–5451.
 - [23] 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.
 - [24] N. Y. Ayusawa K, Venture G, “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.
 - [25] J. Xu, Y. Zheng, X. Jiang, S. Yang, L. Xiang, and Z. Zhang, “Real-time inertial parameter identification of floating-base robots through iterative primitive shape division,” in *2022 International Conference on Robotics and Automation (ICRA)*, 2022, pp. 5960–5966.
 - [26] B. Zhang, Z. Zhou, and R. Vasudevan, “Provably-Safe, Online System Identification,” in *Proceedings of Robotics: Science and Systems*, Los Angeles, CA, USA, June 2025.
 - [27] K. Ayusawa, Y. Nakamura, and G. Venture, “Optimal estimation of human body segments dynamics using realtime visual feedback,” in *2009 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2009, pp. 1627–1632.
 - [28] P. Nadeau, M. Giamou, and J. Kelly, “Fast object inertial parameter identification for collaborative robots,” in *2022 International Conference on Robotics and Automation (ICRA)*, 2022, pp. 3560–3566.
 - [29] O. Harib, A. Hereid, A. Agrawal, T. Gurriet, S. Finet, G. Boeris, A. Duburcq, M. E. Mungai, M. Masselin, A. D. Ames, K. Sreenath, and J. W. Grizzle, “Feedback control of an exoskeleton for paraplegics: Toward robustly stable, hands-free dynamic walking,” *IEEE Control Systems Magazine*, vol. 38, no. 6, pp. 61–87, Dec. 2018.
 - [30] A. F. Soliman, S. Coruk, M. C. Yildirim, D. Ugur, S. C. Cevik, B. Ozkaynak, P. Sendur, and B. Ugurlu, “Design, development, and control for the self- stabilizing bipedal exoskeleton prototype Co-Ex,” *IEEE/ASME Transactions on Mechatronics*, vol. 30, no. 1, p. 458–468, Feb. 2025.
 - [31] C. Meijneke, G. van Oort, V. Sluiter, E. van Asseldonk, N. L. Tagliamonte, F. Tamburella, I. Pisotta, M. Masciullo, M. Arquilla, M. Molinari, A. R. Wu, F. Dzeladini, A. J. Ijspeert, and H. van der Kooij, “Symbitron exoskeleton: Design, control, and evaluation of a modular exoskeleton for incomplete and complete spinal cord injured individuals,” *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, pp. 1–1, 2021.
 - [32] D. Tian, W. Li, J. Li, F. Li, Z. Chen, Y. He, J. Sun, and X. Wu, “Self-balancing exoskeleton robots designed to facilitate multiple rehabilitation training movements,” *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 32, p. 293–303, 2024.
 - [33] K. Yoshida, K. Osuka, H. Mayeda, and T. Ono, “When is the set of base parameter values physically impossible?” in *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS’94)*, vol. 1, 1994, pp. 335–342 vol.1.
 - [34] W. Li and J.-J. Slotine, “Indirect adaptive robot control,” in *Proceedings. 1988 IEEE International Conference on Robotics and Automation*, 1988, pp. 704–709 vol.2.
 - [35] S. Ploen, “A skew-symmetric form of the recursive newton-euler algorithm for the control of multibody systems,” in *Proceedings of the 1999 American Control Conference (Cat. No. 99CH36251)*, vol. 6, 1999, pp. 3770–3773 vol.6.
 - [36] K. Engel and S. Engel, “Recursive least squares with linear inequality constraints,” *Optimization and Engineering*, vol. 16, no. 1, p. 1–26, 2015.
 - [37] J. Gondzio, “Interior point methods 25 years later,” *European Journal of Operational Research*, vol. 218, no. 3, pp. 587–601, 2012.
 - [38] F. Alizadeh, J.-P. A. Haeberly, and M. L. Overton, “Primal-dual interior-point methods for semidefinite programming: Convergence rates, stability and numerical results,” *SIAM Journal on Optimization*, vol. 8, no. 3, pp. 746–768, 1998.
 - [39] K. S. Miller, “On the inverse of the sum of matrices,” *Mathematics Magazine*, vol. 54, no. 2, pp. 67–72, 1981. [Online]. Available: <http://www.jstor.org/stable/2690437>
 - [40] G. P. Kirsten, “Comparison of methods for solving sylvester systems,” Ph.D. dissertation, Stellenbosch: Stellenbosch University, 2018.
 - [41] J. Hwangbo, J. Lee, and M. Hutter, “Per-contact iteration method for solving contact dynamics,” *IEEE Robotics and Automation Letters*, vol. 3, no. 2, pp. 895–902, 2018. [Online]. Available: www.raism.com
 - [42] W.-H. Ho, T.-Y. Shiang, C.-C. Lee, and S.-Y. Cheng, “Body segment parameters of young Chinese men determined with magnetic resonance imaging,” *Medicine & Science in Sports & Exercise*, vol. 45, no. 9, pp. 1759–1766, Sep. 2013.
 - [43] P. de Leva, “Adjustments to zatsiorsky-seluyanov’s segment inertia parameters,” *Journal of Biomechanics*, vol. 29, no. 9, pp. 1223–1230, Sep. 1996.
 - [44] M. Galassi, F. Rossi, B. Gough, M. Booth, G. Jungman, J. Theiler, and J. Davies, *GNU Scientific Library: Reference Manual*. Network

Theory Limited, 2001. [Online]. Available: <https://books.google.com.tr/books?id=w-cOAAAACAAJ>

- [45] M. C. Yildirim, P. Sendur, O. Bilgin, B. Gulek, G. G. Yapici, and B. Ugurlu, "An integrated design approach for a series elastic actuator: Stiffness formulation, fatigue analysis, thermal management," in *2017 IEEE-RAS 17th International Conference on Humanoid Robotics (Humanoids)*, 2017, pp. 384–389.
- [46] M. Mistry, J. Buchli, and S. Schaal, "Inverse dynamics control of floating base systems using orthogonal decomposition," in *Proceedings of Robotics: Science and Systems (RSS VII)*, Los Angeles, California, USA, 2011.
- [47] S. Khorshidi, M. Dawood, B. Nederkorn, M. Bennewitz, and M. Khadiv, "Physically-consistent parameter identification of robots in contact," *arXiv preprint arXiv:2409.09850*, 2024. [Online]. Available: <https://arxiv.org/abs/2409.09850>
- [48] M. Mistry, J. Nakanishi, G. Cheng, and S. Schaal, "Inverse kinematics with floating base and constraints for full body humanoid robot control," in *Humanoids 2008 - 8th IEEE-RAS International Conference on Humanoid Robots*. IEEE, 2008, pp. 22–27.



Ahmed Fahmy Soliman received the Ph.D. degree in mechanical engineering from Özyeğin University, Istanbul, Türkiye, in 2023. He is currently a Research Associate with The University of Manchester, United Kingdom, working within the Robotics and Artificial Intelligence Collaboration (RAICo). His research focuses on wearable and assistive robotics, human–robot interaction, robot dynamics and control, online inertial parameter identification, and physically consistent model-based methods for robotic systems.



Patrick M. Wensing (Senior Member, IEEE) received the B.S., M.S., and Ph.D. degrees in electrical and computer engineering from The Ohio State University, Columbus, OH, USA, in 2009, 2013, and 2014, respectively. He is currently the Wanzek Family Foundation Collegiate Associate Professor of Engineering with the Department of Aerospace and Mechanical Engineering, University of Notre Dame, Notre Dame, IN, USA, where he directs the Robotics, Optimization, and Assistive Mobility Laboratory. Before joining Notre Dame, he was a

Postdoctoral Associate with MIT, Cambridge, MA, USA. His current research interests include dynamics, optimization, and control toward advancing the mobility of legged robots and assistive devices. Dr. Wensing was the recipient of the NSF CAREER Award in 2020 and the Toshio Fukuda Young Professional Award in 2023, as well as several paper award recognitions from IEEE Robotics and Automation Letters, ICRA, Humanoids, and IJHR. He is a Senior Editor for the IEEE TRANSACTIONS ON ROBOTICS and serves as a member of the IEEE Robotics and Automation Society Executive Committee.



Barkan Ugurlu (Senior Member, IEEE) received the Ph.D. degree in electrical and computer engineering from Yokohama National University, Japan, in 2010. From 2010 to 2015, he worked as a researcher with the Istituto Italiano di Tecnologia, Italy, Toyota Technological Institute, Japan, and the Computational Neuroscience Laboratories of ATR, Japan. In 2015, he joined Özyeğin University, Istanbul, Türkiye, where he is currently an Associate Professor. His research interests include locomotion control of legged robotic systems.