

Document Version

Final published version

Licence

CC BY

Citation (APA)

Osman, H., de Kanter, D., Boelens, J., Kok, M., & Seth, A. (2026). A Tightly Coupled Multibody Dynamics and Multi-Sensor Fusion Algorithm for Simultaneous Kinematics and Kinetics Estimation. *Sensors*, 26(12), Article 3697. <https://doi.org/10.3390/s26123697>

Important note

To cite this publication, please use the final published version (if applicable).
Please check the document version above.

Copyright

In case the licence states "Dutch Copyright Act (Article 25fa)", this publication was made available Green Open Access via the TU Delft Institutional Repository pursuant to Dutch Copyright Act (Article 25fa, the Taverne amendment). This provision does not affect copyright ownership.
Unless copyright is transferred by contract or statute, it remains with the copyright holder.

Sharing and reuse

Other than for strictly personal use, it is not permitted to download, forward or distribute the text or part of it, without the consent of the author(s) and/or copyright holder(s), unless the work is under an open content license such as Creative Commons.

Takedown policy

Please contact us and provide details if you believe this document breaches copyrights.
We will remove access to the work immediately and investigate your claim.

Article

A Tightly Coupled Multibody Dynamics and Multi-Sensor Fusion Algorithm for Simultaneous Kinematics and Kinetics Estimation

Hassan Osman ^{1,*} , Daan de Kanter ¹, Jelle Boelens ^{1,2}, Manon Kok ²  and Ajay Seth ¹ ¹ Department of Biomechanical Engineering, Delft University of Technology, 2628 CD Delft, The Netherlands; daandekanter1996@gmail.com (D.d.K.); a.seth@tudelft.nl (A.S.)² Delft Center for Systems and Control, Delft University of Technology, 2628 CD Delft, The Netherlands; m.kok-1@tudelft.nl

* Correspondence: h.osman@tudelft.nl

Abstract

Inertial Measurement Units (IMUs) enable portable, multibody motion capture in diverse environments beyond the laboratory, making them a desirable choice for diagnosing mobility disorders and supporting rehabilitation in clinical or home settings. However, challenges associated with IMU measurements, including magnetic distortions and errors due to integration drift, complicate their broader use for motion capture. In this work, we propose a tightly coupled motion-capture approach that directly integrates IMU measurements with multibody dynamic models via an iterated extended Kalman filter to simultaneously estimate the system's kinematics and kinetics. By enforcing the complete multibody system dynamics and utilizing only accelerometer and gyroscope data, our method accurately estimates joint kinematics and kinetics. Our algorithm is designed to fuse different sensor data, such as optical motion-capture measurements and joint torque readings, to further enhance estimation accuracy. We validated our approach using highly accurate ground-truth data from a 3-degree-of-freedom pendulum and a 6-degree-of-freedom collaborative robot. We demonstrate a maximum root-mean-square difference of 3.75° in the pendulum's computed joint angles with respect to the marker motion-capture inverse kinematics. For the robot, we observed a maximum joint angle root-mean-square difference of 3.24° with respect to the joint encoders, while the maximum joint angle root-mean-square difference of the optical motion-capture inverse kinematics with respect to the encoders was 1.16°. With regard to kinetic estimates, we report a maximum joint torque root-mean-square difference of 3.02 Nm in the pendulum with respect to the marker motion-capture inverse dynamics and 4.27 Nm in the robot relative to its joint torque sensors.

Keywords: sensor fusion; motion capture; iterated extended Kalman filter (IEKF); kinematics; kinetics; IMU; inertial sensors; movement tracking



Academic Editor: Pietro Picerno

Received: 10 April 2026

Revised: 15 May 2026

Accepted: 1 June 2026

Published: 10 June 2026

Copyright: © 2026 by the authors.

Licensee MDPI, Basel, Switzerland.

This article is an open access article distributed under the terms and conditions of the [Creative Commons Attribution \(CC BY\)](https://creativecommons.org/licenses/by/4.0/) license.

1. Introduction

Motion-capture systems are used in a wide range of applications, such as sports, film animation, video games and, importantly, the diagnosis and treatment of human movement disorders. They provide a method to analyze multibody movements [1–4]. Analysis of human movement begins with motion-capture systems [5] that provide measurements from a collection of sensors, from which the subject's kinematics and kinetics can be evaluated. Motion-capture measurements typically take the form of 3D marker positions, 3D sensor

orientations from Inertial Measurement Units (IMUs), or joint positions from markerless video-based systems [6,7].

IMUs allow multibody motion capture in more varied environments and settings compared to optical marker-based systems, which are typically restricted to controlled laboratory conditions [6,8]. This flexibility makes IMU-based systems particularly valuable for applications such as patient rehabilitation, where portability, accessibility and privacy are important [3]. However, due to limitations in accuracy, interpretability, and sensor-to-body calibration, IMU-based systems have not been widely adopted in clinical applications [9,10]. In this work, we focus on an estimation framework based on a tightly coupled motion-capture algorithm that leverages knowledge of the multibody system's dynamics, which are directly related to the IMU's accelerometer and gyroscope measurements, to simultaneously estimate system kinematics and kinetics, expressed in biomechanical motion axes [11]. By enforcing dynamic consistency between the estimated kinematics and kinetics, our approach aims to improve the accuracy of IMU-based motion estimation.

IMUs capture linear acceleration and angular velocity using accelerometers and gyroscopes [12]. In addition, IMUs typically incorporate a magnetometer to measure the local magnetic field intensity. Using these measurements, the orientation of the IMU can be estimated [12]. Inaccuracies in IMU-derived orientations arise from multiple sources: magnetic distortions (especially indoors), gyroscope bias and noise, and the difficulty of separating inertial acceleration from gravitational acceleration during motion. Consequently, orientation estimates contain errors that propagate to subsequent joint angle estimates. While these angular estimates are often used directly for kinematic analysis, in cases where kinetics are also required, they are typically low-pass-filtered and then differentiated twice and filtered to compute the joint torques using inverse dynamics (ID) [13]. Due to this process, the resulting kinetics are often inconsistent with the kinematic constraints imposed by joint definitions and with external forces measured by additional sensors such as force-plates [12–18].

Musculoskeletal modeling software such as OpenSim [19–21] enables joint angle estimation while imposing joint mobility constraints through inverse kinematics (IK). OpenSense [22], an open-source algorithm within OpenSim (version 4.4), utilizes IMU orientations in conjunction with OpenSim's internal coordinate formulation [23] to estimate multibody joint kinematics via IK. In this method, virtual IMUs are placed on the model's body segments with the same initial orientations as the real IMUs. In each time frame of movement, IK is solved for the model's joint angles that best reproduce the experimentally measured IMU orientations by minimizing the angular distance (error) between the orientations of the virtual and experimental IMUs. Because the measured IMU orientations are directly used to estimate joint angles, errors in IMU orientation estimates propagate to the resulting kinematics, making accurate IMU orientation estimation critical for this approach. Alternatively, kinematically coupled approaches directly integrate joint kinematic definitions with raw IMU measurements, bypassing the need for precomputed orientations. For example, Weygers et al. [24] enforce joint kinematic constraints by requiring consistency in acceleration at the joint center shared by adjacent body segments, as measured by the IMUs mounted on each segment. However, while this approach improves kinematic consistency, joint kinetics are not estimated directly.

Incorporating the multibody dynamics of the system, along with appropriate joint definitions, has been found to improve the accuracy of the estimated kinematics and kinetics [25]. Similar to the works of Dorschky et al. [25], Haraguchi and Hase [26], Nitschke et al. [27], and Koelewijn et al. [28], we leverage dynamic models to estimate joint kinematics and kinetics directly from IMU measurements (linear accelerations and angular velocities). However, our approach differs in several key aspects. First, unlike Dorschky et al. [25],

who employed a planar model, we use the full multibody system dynamics to enforce the system equations of motion as part of the estimation. Second, contrary to Nitschke et al. [27], we validate our method using real IMU measurements rather than relying on simulated inertial data generated from marker-based motion capture. Third, unlike trajectory optimization approaches such as that of Haraguchi and Hase [26], which require predefined movement parameterizations (e.g., fixed nodes per gait cycle) and are limited to periodic movements with known cycle boundaries, our method performs state estimation without requiring prior knowledge of movement type, duration, or periodicity. Similarly, Koelewijn et al. [28] require a predefined time discretization over a fixed movement window and estimate the entire trajectory simultaneously, whereas our approach is sequential and makes no such assumptions. We present an iterated extended Kalman filter (IEKF)-based algorithm that enables the use of different sensor modalities and measurement types. We demonstrate this extensibility by integrating marker-based motion capture and zero-torque updates (described in detail in Section 2) along with IMU measurements. We tested our algorithm using real sensor measurements on two experimental setups: a 6-DoF KUKA robot (KUKA AG, Augsburg, Germany) and a 3-DoF pendulum, each of which provided accurate ground-truth data.

2. Methodology

To exploit the underlying multibody dynamics in the movement estimation problem, we developed an algorithm to directly couple the multibody system's equations of motion with IMU sensor measurements (Figure 1). Estimating the system's kinematic and kinetic states directly from the system dynamics requires formulating a continuous nonlinear state-space model that describes the measurements in terms of the system states [12,29–31]. In this work, we leverage OpenSim (version 4.4) [20], an open-source software for musculoskeletal modeling and simulation, to evaluate the equations of motion and kinematic Jacobians of the multibody system, and we use it to derive different sensor measurement models [32].

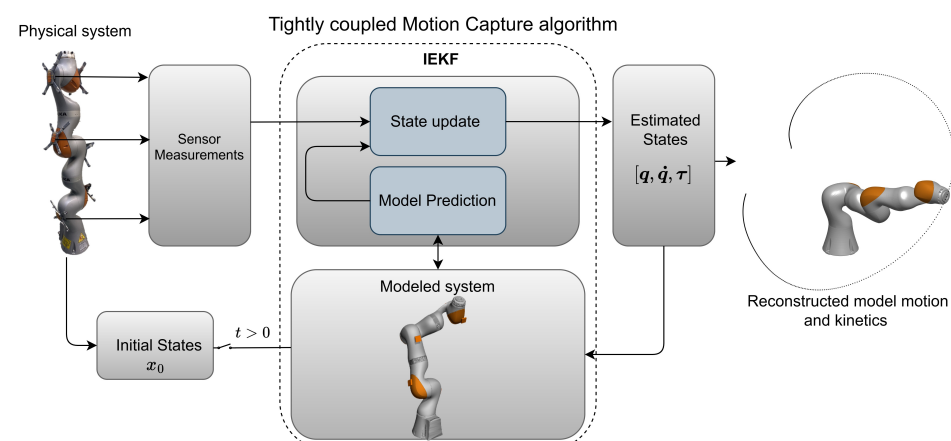


Figure 1. Overview of the tightly coupled motion-capture algorithm (TCMA).

The motion of a multibody system with N_J joints, N_B body segments and N_D joint angles at each time instant t is estimated in terms of joint angles $\mathbf{q} = (q_1, q_2, \dots, q_{N_D})^T$, joint angular speeds $\dot{\mathbf{q}} = (\dot{q}_1, \dot{q}_2, \dots, \dot{q}_{N_D})^T$ and joint torques $\boldsymbol{\tau} = (\tau_1, \tau_2, \dots, \tau_{N_D})^T$. Hence, the state vector $\mathbf{x}$ that we are interested in estimating is given by

$$\mathbf{x}^T = (\mathbf{q}^T \quad \dot{\mathbf{q}}^T \quad \boldsymbol{\tau}^T), \quad (1)$$

where $\mathbf{q} \in \mathbb{R}^{N_D}$, $\dot{\mathbf{q}} \in \mathbb{R}^{N_D}$ and $\boldsymbol{\tau} \in \mathbb{R}^{N_D}$.

2.1. General Multibody State-Space Model

The multibody system dynamics describes the dynamics of the states in (1) as

$$\tau = \bar{\mathbf{M}}(q)\ddot{q} + \underbrace{\mathbf{C}(q, \dot{q}) + \mathbf{G}(q)}_{\tau^I(q, \dot{q})}, \quad (2)$$

for which $\bar{\mathbf{M}}(q)$ is a symmetric, positive definite mass matrix, $\mathbf{C}(q, \dot{q})$ is a function that groups all the velocity terms, and $\mathbf{G}(q)$ is a function that represents the gravity force component [19,33]. Furthermore, $\tau^I(q, \dot{q})$ is a function representing the torque that results from gravity, the Coriolis force and the centrifugal force. Using (2), the joint angular accelerations can be expressed as

$$\ddot{q} = \bar{\mathbf{M}}(q)^{-1}(\tau - \tau^I(q, \dot{q})). \quad (3)$$

The system's dynamic model is expressed to include process noise, which accounts for uncertainties and disturbances in the system as follows:

$$\dot{\mathbf{x}} = \mathbf{g}(\mathbf{x}) + \mathbf{e}_p. \quad (4)$$

where $\mathbf{e}_p$ is zero-mean Gaussian noise with covariance $\mathbf{Q} \in \mathbb{R}^{3N_D \times 3N_D}$. The joint torques are assumed to evolve in a random-walk process [34] with $\dot{\tau} = \mathbf{e}_{p_\tau}$, where $\mathbf{e}_{p_\tau} \sim \mathcal{N}(0, \mathbf{Q}_\tau)$. Since the process noise is assumed to only affect the torque states, the process noise covariance matrix $\mathbf{Q}$ has a block-diagonal structure with zero blocks for the generalized coordinates and velocities and $\mathbf{Q}_\tau$ for the joint torques.

The system's measurement model relates the states to the measurements as follows:

$$\mathbf{y} = \mathbf{h}(\mathbf{x}) + \mathbf{e}_{\text{meas}}. \quad (5)$$

Here, $\mathbf{h} : \mathbb{R}^{3N_D} \rightarrow \mathbb{R}^{N_y}$ is a function that maps the states to the N_y sensor measurements, and $\mathbf{e}_{\text{meas}} \sim \mathcal{N}(0, \mathbf{R})$ is the measurement error, where $\mathbf{R} = \text{blkdiag}(\Sigma_{\omega, s_1}, \Sigma_{\dot{v}, s_1}, \Sigma_{p, m_1}, \Sigma_{\tau, \phi_1}, \dots)$. These measurements consist of IMU data, optical markers, and zero-torque updates, which will be described in Sections 2.2 and 2.3.

2.2. IMU Sensor Measurement Model

We assume that each body segment has at least one IMU attached. The total number of IMUs on the system's body segments is N_I . Each IMU s , with $s = 1, \dots, N_I$, has a gyroscope that measures the angular velocity ω_s^s and an accelerometer that measures linear acceleration $\dot{v}_s^s$, where the subscript indicates the measurement of sensor s , and the superscript denotes the frame in which the measurement is expressed. This notation will be used throughout the rest of this paper. The axes of the frame s are assumed to be aligned with the sensor's axes, and its origin is assumed to be located at the center of the accelerometer triad. The measurement model for the gyroscope angular velocity is modeled by

$$\mathbf{y}_\omega = \omega_s^s + \mathbf{e}_\omega^s, \quad (6)$$

where $\mathbf{y}_\omega \in \mathbb{R}^{3 \times 1}$ is the measured angular velocity, and $\mathbf{e}_\omega^s \sim \mathcal{N}(0, \Sigma_{\omega, s})$ is the measurement noise. Measured angular velocities can be expressed in terms of the body segment's angular velocities, which can be expressed in terms of the system's current generalized coordinates and velocities [33] as

$$\mathbf{y}_\omega = \mathbf{R}^{\text{sb}} \mathbf{R}^{\text{bn}}(q) \omega_b^n(q, \dot{q}) + \mathbf{e}_\omega^s. \quad (7)$$

Here, two rotation matrices are introduced. First, $\mathbf{R}^{sb}$ is the constant rotation matrix between the body frame b and the sensor frame s , which is assumed to be known. Second, $\mathbf{R}^{bn}$ is the rotation matrix that describes the orientation transformation from the navigation frame n to the body segment frame b as a function of the joint angles. Note that the body frame and sensor frame have the same angular velocity, provided that the velocities are expressed in a common frame.

The IMU's accelerometer measures the linear accelerations that the sensor experiences, including the acceleration due to gravity $\mathbf{g}$. The measured accelerations depend on the current generalized joint angles, angular velocities and angular accelerations and can be expressed as

$$\mathbf{y}_{\hat{v}} = \mathbf{R}^{sb} \mathbf{R}^{bn}(\mathbf{q})(\mathbf{a}_s^n(\mathbf{x}) - \mathbf{g}^n) + \mathbf{e}_{\hat{v}}^s. \quad (8)$$

Here, $\mathbf{y}_{\hat{v}} \in \mathbb{R}^{3 \times 1}$ is the measured linear acceleration, and $\mathbf{g}^n \in \mathbb{R}^{3 \times 1}$ is the gravity vector expressed in the navigation frame. We model the accelerometer measurement noise as $\mathbf{e}_{\hat{v}}^s \sim \mathcal{N}(0, \Sigma_{\hat{v}})$. The acceleration $\mathbf{a}_s^n(\mathbf{x})$ is defined in terms of the body's angular velocity, angular acceleration and linear acceleration as

$$\mathbf{a}_s^n(\mathbf{x}) = \mathbf{a}_b^n(\mathbf{x}) + [\boldsymbol{\alpha}_b^n(\mathbf{x}) \times] \left(\mathbf{R}^{nb}(\mathbf{q}) \mathbf{r}_{bs}^b \right) + [\boldsymbol{\omega}_b^n(\mathbf{q}, \dot{\mathbf{q}}) \times]^2 \left(\mathbf{R}^{nb}(\mathbf{q}) \mathbf{r}_{bs}^b \right). \quad (9)$$

Here, $\mathbf{a}_b^n$, $\boldsymbol{\alpha}_b^n$ and $\boldsymbol{\omega}_b^n$ are respectively the body segment's linear acceleration, angular acceleration and angular velocity. These kinematic quantities are computed by propagating the joint angles, velocities, and accelerations, $\mathbf{q}$, $\dot{\mathbf{q}}$, and $\ddot{\mathbf{q}}$, recursively through the multibody chain from the fixed base, where $\ddot{\mathbf{q}}$ depends on $\boldsymbol{\tau}$ through (2) and is evaluated by OpenSim. Furthermore, $\mathbf{r}_{bs}^b$ is a translation vector expressed in the body frame b that locates the position of the IMU frame s to the body frame b , as shown in Figure 2. The rotation matrix $\mathbf{R}^{nb}(\mathbf{q}) = (\mathbf{R}^{bn}(\mathbf{q}))^T$ maps the body frame to the navigation frame.

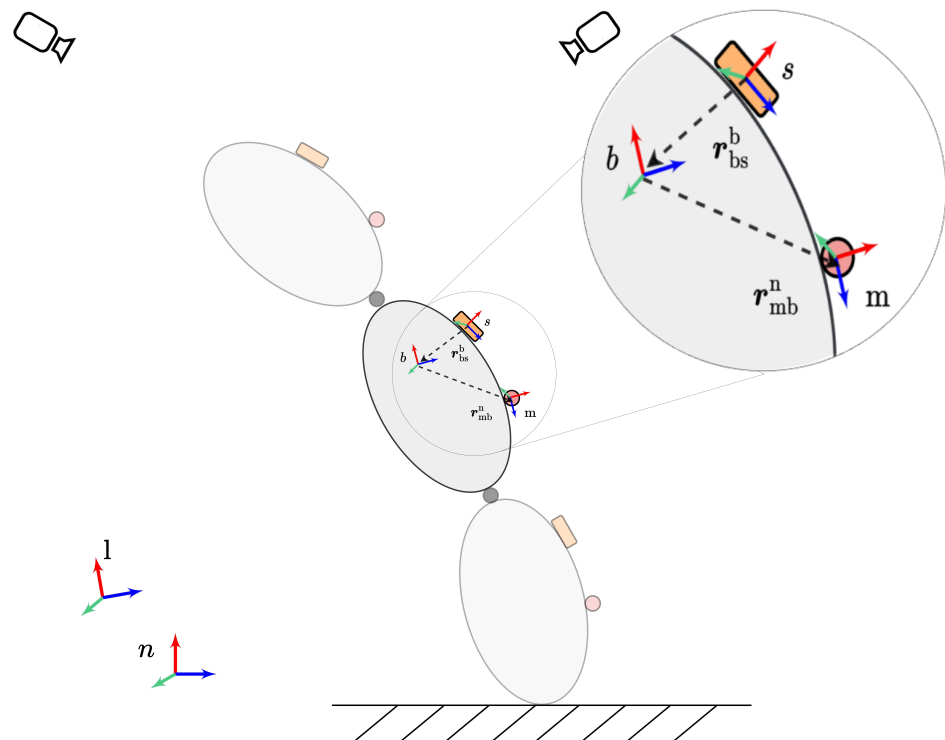


Figure 2. Definitions of the navigation frame n , the laboratory frame l , the body frame b , the IMU frame s , and the marker frame m . It also includes the translation vectors $\mathbf{r}_{bs}^b$ and $\mathbf{r}_{mb}^n$.

2.3. Marker-Based Motion Capture and Zero-Torque Update Measurement Models

Marker-based motion-capture systems use multiple high-speed cameras fitted within a laboratory. After calibration, these systems provide marker position measurements with respect to a laboratory frame l . Each body segment can have any number of markers attached for a total number of markers N_M . We denote each individual marker by $m = 1, \dots, N_M$. The marker position with respect to the lab frame can then be defined as

$$\mathbf{y}_p = \mathbf{p}_m^l + \mathbf{e}_p^l, \quad (10)$$

where $\mathbf{y}_p \in \mathbb{R}^{3 \times 1}$ is the marker position measured in the laboratory frame, and $\mathbf{e}_p^l \sim \mathcal{N}(0, \Sigma_{p,m})$ is the position measurement noise. Marker position measurements can be written in terms of the body position in the navigation frame as

$$\mathbf{y}_p = \mathbf{R}^{lb}(\mathbf{q})\mathbf{R}^{bn}(\mathbf{q})(\mathbf{p}_b^n + \mathbf{r}_{mb}^n(\mathbf{q})) + \mathbf{e}_p^l. \quad (11)$$

The translation vector $\mathbf{r}_{mb}^n(\mathbf{q}) \in \mathbb{R}^{3 \times 1}$ is a vector that maps the position of the body segment frame b to the marker frame m expressed in the navigation frame n , as shown in Figure 2. The two matrices $\mathbf{R}^{bn}(\mathbf{q}) \in \mathbb{R}^{3 \times 3}$ and $\mathbf{R}^{lb}(\mathbf{q}) \in \mathbb{R}^{3 \times 3}$ map the navigation frame n to the body frame b and the body frame b to the laboratory frame l , respectively.

We also consider the occasional availability of zero-torque measurement updates. For this, virtual joint torque sensors $\phi = 1, \dots, N_D$ that assume joint torque measurements of zero are expressed as

$$\mathbf{y}_\tau = 0 + \mathbf{e}_\tau^\phi, \quad (12)$$

where $\mathbf{y}_\tau \in \mathbb{R}^{1 \times 1}$ is the joint zero-torque virtual measurement, and $\mathbf{e}_\tau^\phi \sim \mathcal{N}(0, \Sigma_{\tau,\phi})$ is the virtual torque measurement noise.

2.4. Tightly Coupled Kinematics and Kinetics Estimation Algorithm

To couple the system dynamics with the measurement models (above), we employ an iterated extended Kalman filter (IEKF), specified as follows (see Algorithm 1). The multi-body system dynamics are defined by the OpenSim model, which includes body segment masses, inertial properties, joint types, and sensor attachment locations. OpenSim provides the computational framework to evaluate the mass matrix $\mathbf{M}(\mathbf{q})$, velocity-dependent terms $\mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})$, and gravitational forces $\mathbf{G}(\mathbf{q})$ in the process model $\mathbf{g}(\mathbf{x})$ from (4). Additionally, OpenSim computes the kinematic quantities for the measurement model $\mathbf{h}(\mathbf{x}_t)$ in (14), including rotation matrices $\mathbf{R}^{bn}(\mathbf{q})$ in (7) and (8), body segment accelerations $\mathbf{a}_b^n$, angular velocities $\boldsymbol{\omega}_b^n$, and angular accelerations $\boldsymbol{\alpha}_b^n$. Furthermore, OpenSim enables the computation of the process and measurement Jacobians required for the IEKF [23,35]. The dynamic model is discretized at the sensor measurement frequency (100 Hz) to predict the future state at time $t + 1$ by numerically integrating the equations of motion using OpenSim's Runge–Kutta integrator [35], resulting in a discrete-time update equation

$$\mathbf{x}_t = \mathbf{f}(\mathbf{x}_{t-1}, \Delta t) + \mathbf{w}_t, \quad (13)$$

where $\mathbf{x}_t$ denotes the state at time instance t , $\mathbf{f}(\mathbf{x}_t, \Delta t)$ represents the numerical integration of the continuous dynamics in (4) over the time step Δt , and $\mathbf{w}_t \sim \mathcal{N}(0, \mathbf{Q}_\Delta)$ is the discrete-time process noise. The algorithm measurement model that includes all the previously discussed sensor measurement models can then be expressed as

$$\mathbf{y}_t = \begin{pmatrix} \mathbf{y}_{\omega,t,s_1} \\ \mathbf{y}_{\dot{v},t,s_1} \\ \mathbf{y}_{p,t,m_1} \\ \mathbf{y}_{\tau,t,\phi_1} \\ \vdots \\ \mathbf{y}_{\omega,t,s_{N_I}} \\ \mathbf{y}_{\dot{v},t,s_{N_I}} \\ \mathbf{y}_{p,t,m_{N_M}} \\ \mathbf{y}_{\tau,t,\phi_{N_D}} \end{pmatrix} = \begin{pmatrix} \mathbf{h}_{\omega,t,1}(\mathbf{x}_t) \\ \mathbf{h}_{\dot{v},t,1}(\mathbf{x}_t) \\ \mathbf{h}_{p,t,1}(\mathbf{x}_t) \\ \mathbf{h}_{\tau,t,1}(\mathbf{x}_t) \\ \vdots \\ \mathbf{h}_{\omega,t,N_I}(\mathbf{x}_t) \\ \mathbf{h}_{\dot{v},t,N_I}(\mathbf{x}_t) \\ \mathbf{h}_{p,t,N_M}(\mathbf{x}_t) \\ \mathbf{h}_{\tau,t,N_D}(\mathbf{x}_t) \end{pmatrix} + \mathbf{e}_{\text{meas},t}. \quad (14)$$

$\underbrace{\hspace{10em}}_{\mathbf{h}(\mathbf{x}_t)}$

Here, $\mathbf{h}(\mathbf{x}_t)$ is the measurement model, $\mathbf{e}_{\text{meas},t} \sim \mathcal{N}(\mathbf{0}, \mathbf{R})$ is the total measurement error, and $\mathbf{R} \in \mathbb{R}^{(6N_I+3N_M+N_D) \times (6N_I+3N_M+N_D)}$ is the measurement error covariance.

Algorithm 1: Tightly coupled motion-capture algorithm (TCMA)

Input: Sensor measurements $\mathbf{y}_t$, an initial state estimate $\hat{\mathbf{x}}_0$ with covariance $\hat{\mathbf{P}}_0$, process and measurement noise covariance matrices $\mathbf{Q}$ and $\mathbf{R}$; see (4), (13) and (14).

Output: Estimates of the system states $\hat{\mathbf{x}}_t$ and the respective covariance $\hat{\mathbf{P}}_t$ for $t = 1, \dots, T$.

1: **for** $t = 1, \dots, T$ **do**

2: **Time update** Compute the state prediction by numerically integrating the noise-free dynamics in (4) from the previous estimate $\hat{\mathbf{x}}_{t-1}$ over one time step Δt ,

$$\check{\mathbf{x}}_t = \mathbf{f}(\hat{\mathbf{x}}_{t-1}, \Delta t). \quad (15)$$

Compute the predicted state covariance as

$$\check{\mathbf{P}}_t = \mathbf{F}_{t-1} \hat{\mathbf{P}}_{t-1} \mathbf{F}_{t-1}^\top + \mathbf{Q}, \quad (16)$$

where the discrete-time state transition Jacobian is approximated by discretizing the continuous-time Jacobian as $\mathbf{F}_{t-1} \approx \exp(\mathbf{G}_{t-1} \Delta t)$, with $\mathbf{G}_{t-1} = \left. \frac{\partial \mathbf{g}(\mathbf{x})}{\partial \mathbf{x}} \right|_{\mathbf{x}=\hat{\mathbf{x}}_{t-1}}$ computed numerically.

3: **Measurement update**

4: **for** $k = 1, \dots, \epsilon$ **until** $\bar{\mathbf{x}}_t^{k+1} \approx \bar{\mathbf{x}}_t^k$ **do**

5: Update the state with measurements $\mathbf{y}_t$ as

$$\bar{\mathbf{x}}_t^{k+1} = \check{\mathbf{x}}_t + \mathbf{K}_t^k \left(\mathbf{y}_t - \mathbf{h}_t(\bar{\mathbf{x}}_t^k) - \mathbf{H}_t^k (\check{\mathbf{x}}_t - \bar{\mathbf{x}}_t^k) \right), \quad (17)$$

$$\text{where } \mathbf{K}_t^k = \check{\mathbf{P}}_t (\mathbf{H}_t^k)^\top \left(\mathbf{H}_t^k \check{\mathbf{P}}_t (\mathbf{H}_t^k)^\top + \mathbf{R} \right)^{-1} \text{ and } \mathbf{H}_t^k = \left. \frac{\partial \mathbf{h}_t(\mathbf{x}_t)}{\partial \mathbf{x}_t} \right|_{\mathbf{x}_t=\bar{\mathbf{x}}_t^k}.$$

6: Compute the filtered state and covariance

$$\hat{\mathbf{x}}_t = \bar{\mathbf{x}}_t^{k+1}, \quad \hat{\mathbf{P}}_t = \left(\mathbf{I} - \mathbf{K}_t^k \mathbf{H}_t^k \right) \check{\mathbf{P}}_t. \quad (18)$$

The process of estimating the system states $\hat{\mathbf{x}}_t$ and their covariance $\hat{\mathbf{P}}_t$ using an IEKF is described in Algorithm 1. The first step is prediction, in which the state and covariance are predicted as $\check{\mathbf{x}}_t$ and $\check{\mathbf{P}}_t$ respectively. In this step the Jacobian $\mathbf{G} = \frac{\partial \mathbf{g}(\mathbf{x})}{\partial \mathbf{x}} \in \mathbb{R}^{3N_D \times 3N_D}$ is calculated to compute $\check{\mathbf{P}}_t$. In the second step the algorithm uses the measurements to iteratively update the values of the state and covariance. In this step the Jacobian

$H = \frac{\partial h(x_t)}{\partial x_t} \in \mathbb{R}^{(6N_I+3N_M+N_D) \times 3N_D}$ is calculated to compute the Kalman gain K_t , the updated states $\hat{x}_t$ and updated state covariance $\hat{P}_t$. To handle the nonlinearities of the measurement model, the IEKF re-linearizes around the most recent state estimate at each iteration within the same time step, refining the state estimate until convergence, as shown in Algorithm 1. This reduces the linearization error compared to the standard EKF, which linearizes only once around the predicted state. The discrete-time state transition Jacobian F_{t-1} is approximated via the matrix exponential of the continuous-time Jacobian rather than by perturbing the numerical integrator directly, as the latter would require $3N_D$ additional integrations per time step with no observable gain in accuracy.

2.5. Real-World Testing on Physical Systems

To validate our approach, we performed two separate sets of experiments on two different systems. In the first setup we used a 3-degree-of-freedom (DoF) physical four-link pendulum consisting of aluminum segments connected by roller-bearing hinge joints, as shown in Figure 3. Each body segment was equipped with an APDM Opal inertial sensor (APDM Wearable Technologies, Portland, OR, USA) and a set of optical motion-capture markers tracked by a Vicon motion-capture system (Vicon Motion Systems Ltd., Oxford, UK). Experiments were conducted on this passive multibody system, in which the lowest body segment, shown in silver in Figure 3, was raised to a certain height and then released to swing under its own weight; the applied joint torques are assumed to be zero upon release. We evaluated Algorithm 1 (TCMA) using three different combinations of sensor measurements to estimate the pendulum motion:

- IMU measurements only (TCMA_I);
- IMU and marker measurements (TCMA_{I&M});
- IMU, marker, and virtual zero-torque measurements (TCMA_{I,M&0T}).

We compared the output of our algorithm with these different measurement combinations against OpenSim marker-based IK/ID (IK_M/ID_M) and OpenSense IMU orientation-based IK/ID (IK_{Io}/ID_{Io}) across six trials, each with different initial generalized joint angles. While marker-based IK/ID is commonly used as a reference standard, we additionally include TCMA_{I&M} as a reference. This allows a comparison of OpenSense (which uses only IMU orientation estimates) against a method that integrates both measurement modalities.

The second experimental setup was a KUKA LBR 7 iiwa R800 robot (Figure 4) consisting of 7 bodies and 6 revolute joints. Markers tracked by an OptiTrack motion-capture system (NaturalPoint, Inc., Corvallis, OR, USA) and Xsens MTw2 IMUs (Movella/Xsens, Enschede, The Netherlands) were fitted on all robot body segments except the base and the end-effector. Five separate trials were conducted in which we extracted the markers' 3D positions, IMU accelerations and angular velocities, as well as ground-truth joint angles and torques from the robot's internal encoders and sensors. We applied the TCMA to estimate the robot motion using two measurement configurations: IMU-only measurements in 4 trials and combined IMU and marker-based measurements in 1 trial. For the combined measurement trial, we also compared results with IK_M/ID_M. The OpenSense IK_{Io}/ID_{Io} method was not feasible due to magnetic disturbances from the robot's metallic structure and motors, which compromise magnetometer-based orientation estimates [15]. For the KUKA robot setup, all IMUs were pre-calibrated before each trial to remove accelerometer and gyroscope biases.

The measurement noise covariance matrix R was constructed using the KUKA robot setup by computing the sample covariance of residuals between sensor measurements and the corresponding encoder-derived robot states, treating the encoder measurements as ground truth given their high accuracy [36]. The process noise covariance Q_τ was set as a diagonal matrix, with elements manually tuned to approximately 1 (Nm)^2 to balance

dynamic flexibility and estimation accuracy. These same R and Q_τ values were used for both the KUKA and pendulum experiments without further adjustment, as the algorithm exhibited low sensitivity to the precise values of R , and the chosen Q_τ provided adequate performance across both experimental setups.

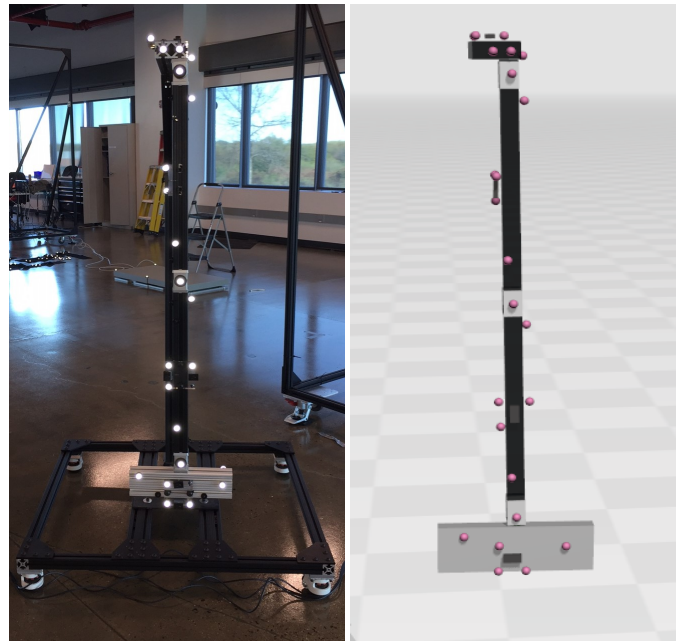


Figure 3. Experimental setup of a four-segment 3-DoF pendulum setup from Johns Hopkins Applied Physics Laboratory (**left**), where optical markers and an IMU are attached to each body segment. On the **right** is an OpenSim model with corresponding IMU and marker placements.

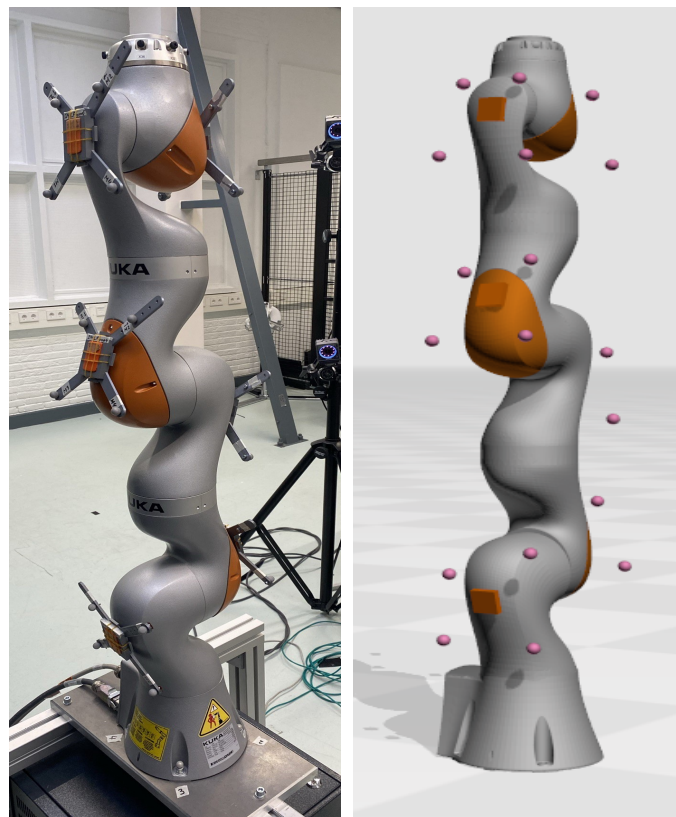


Figure 4. The experimental setup for the robot experiment using the KUKA LBR 7 iiwa R800 collaborative robot, where a set of 4 markers and an IMU were placed on each body segment (**left**). On the **right**, the OpenSim model that describes the system is shown.

3. Results

We present the motion estimated by the TCMA using the sensor measurement combinations described in Section 2.5. For each experimental setup, we report root-mean-square difference (RMSD) values across all trials and illustrate the estimated joint angles and torques for a representative trial.

Figure 5 shows representative results from a randomly selected pendulum trial (trial 3). At time 0 s, the pendulum was in its initial position, as shown in Figure 3. The pendulum was raised and then released to swing freely under its own weight from 1.2 s to 7.8 s. All other trials followed the same procedure with different initial joint angles. The virtual zero-torque measurements shown in Figure 5 were only applied to TCMA_{I,M&OT} after the pendulum was released. Table 1 presents RMSD values across all six pendulum trials, comparing estimates from the TCMA using the different measurement combinations (TCMA_I, TCMA_{I&M}, and TCMA_{I,M&OT}), alongside IK_{I0}/ID_{I0}, evaluated against IK_M/ID_M and TCMA_{I&M} as reference standards. Estimates from the TCMA showed small deviations in joint angles ($\leq 4.23^\circ$) and joint torques (≤ 3.02 Nm) when compared to IK_M/ID_M. However, a steady offset of approximately 2° is observed in the q_2 estimate from TCMA_{I,M&OT} after 2.4 s compared with the other q_2 estimates (Figure 5).

Table 1. Joint angle RMSD ($^\circ$) and joint torque RMSD (Nm) across six pendulum trials, with IK_M/ID_M and TCMA_{I&M} as references respectively (RMSD_{(IK_M/ID_M) | TCMA_{I&M}). Bold values indicate the highest joint angle and joint torque RMSD values per trial across all methods, reported separately for the two RMSD references.}

Trial	Metric	RMSD vs. IK _M /ID _M RMSD vs. TCMA _{I&M}				
		TCMA _I	TCMA _{I&M}	TCMA _{I,M&OT}	IK _{I0} /ID _{I0}	IK _M /ID _M
1	q_1 ($^\circ$)	1.27 0.10	1.28 –	1.29 0.10	0.92 2.13	– 1.28
	q_2 ($^\circ$)	0.91 0.12	0.93 –	0.99 0.09	0.92 1.28	– 0.93
	q_3 ($^\circ$)	3.49 0.12	3.54 –	4.23 0.93	6.56 3.32	– 3.54
	τ_1 (Nm)	1.05 0.03	1.07 –	0.92 0.47	0.78 1.56	– 1.07
	τ_2 (Nm)	0.39 0.01	0.40 –	0.34 0.28	0.39 0.66	– 0.40
	τ_3 (Nm)	0.07 0.00	0.07 –	0.11 0.07	0.15 0.16	– 0.07
2	q_1 ($^\circ$)	1.10 0.11	1.11 –	1.11 0.10	1.54 2.26	– 1.11
	q_2 ($^\circ$)	0.94 0.17	0.97 –	1.02 0.11	2.21 2.60	– 0.97
	q_3 ($^\circ$)	3.75 0.10	3.72 –	4.16 0.75	5.50 4.05	– 3.72
	τ_1 (Nm)	1.70 0.05	1.70 –	1.33 0.82	1.70 1.91	– 1.70
	τ_2 (Nm)	1.01 0.01	1.00 –	0.79 0.48	0.75 1.09	– 1.00
	τ_3 (Nm)	0.18 0.00	0.18 –	0.21 0.12	0.12 0.26	– 0.18
3	q_1 ($^\circ$)	1.34 0.10	1.35 –	1.38 0.10	1.00 1.67	– 1.35
	q_2 ($^\circ$)	0.76 0.16	0.79 –	0.87 0.13	1.10 1.27	– 0.79
	q_3 ($^\circ$)	1.64 0.09	1.66 –	1.86 0.40	3.18 2.81	– 1.66
	τ_1 (Nm)	1.23 0.05	1.24 –	1.25 0.66	1.45 1.52	– 1.24
	τ_2 (Nm)	0.63 0.01	0.63 –	0.62 0.39	0.64 0.71	– 0.63
	τ_3 (Nm)	0.14 0.00	0.14 –	0.18 0.09	0.06 0.16	– 0.14
4	q_1 ($^\circ$)	1.28 0.13	1.27 –	1.29 0.11	1.38 2.57	– 1.27
	q_2 ($^\circ$)	1.19 0.38	1.11 –	1.22 0.13	0.65 1.07	– 1.11
	q_3 ($^\circ$)	2.75 0.28	2.69 –	3.08 0.56	3.38 3.91	– 2.69
	τ_1 (Nm)	1.53 0.18	1.52 –	1.42 0.71	1.34 2.01	– 1.52
	τ_2 (Nm)	0.76 0.05	0.76 –	0.69 0.43	0.60 0.89	– 0.76
	τ_3 (Nm)	0.14 0.00	0.14 –	0.18 0.10	0.08 0.18	– 0.14
5	q_1 ($^\circ$)	1.05 0.12	1.04 –	1.05 0.11	0.86 1.51	– 1.04
	q_2 ($^\circ$)	1.24 0.58	1.18 –	1.18 0.18	2.09 2.29	– 1.18
	q_3 ($^\circ$)	2.67 0.68	2.65 –	2.94 0.56	2.37 2.60	– 2.65
	τ_1 (Nm)	1.85 0.07	1.85 –	1.74 0.69	1.18 2.06	– 1.85
	τ_2 (Nm)	1.02 0.07	1.02 –	0.95 0.41	0.60 1.10	– 1.02
	τ_3 (Nm)	0.26 0.02	0.26 –	0.26 0.09	0.05 0.26	– 0.26
6	q_1 ($^\circ$)	1.45 0.80	1.27 –	1.29 0.19	1.09 1.48	– 1.27
	q_2 ($^\circ$)	1.75 0.43	1.64 –	1.67 0.24	1.43 1.51	– 1.64
	q_3 ($^\circ$)	3.72 1.64	3.08 –	3.04 1.11	3.23 2.80	– 3.08
	τ_1 (Nm)	3.02 0.52	2.92 –	2.51 0.84	1.40 2.85	– 2.92
	τ_2 (Nm)	1.72 0.22	1.68 –	1.47 0.50	0.64 1.59	– 1.68
	τ_3 (Nm)	0.48 0.03	0.48 –	0.47 0.10	0.09 0.49	– 0.48

The KUKA robot experiments show similar performance. Joint angle and torque estimates from the TCMA, using either IMU-only measurements (TCMA_I) or combined IMU and marker measurements (TCMA_{I&M}), are compared with IK_M/ID_M and the robot's

internal encoders and torque sensors (Figure 6). The robot's initial pose is shown in Figure 7. The shaded regions in Figure 6 indicate the robot torque sensor accuracy ($\pm 2\%$ of maximum torque, ranging from 176 Nm for joint 1 to 40 Nm for joint 6 [36,37]), providing context for evaluating whether our estimates fall within sensor uncertainty. The robot torque measurements shown are smoothed in Figure 6 using a 20-sample moving average filter to reduce high-frequency noise.

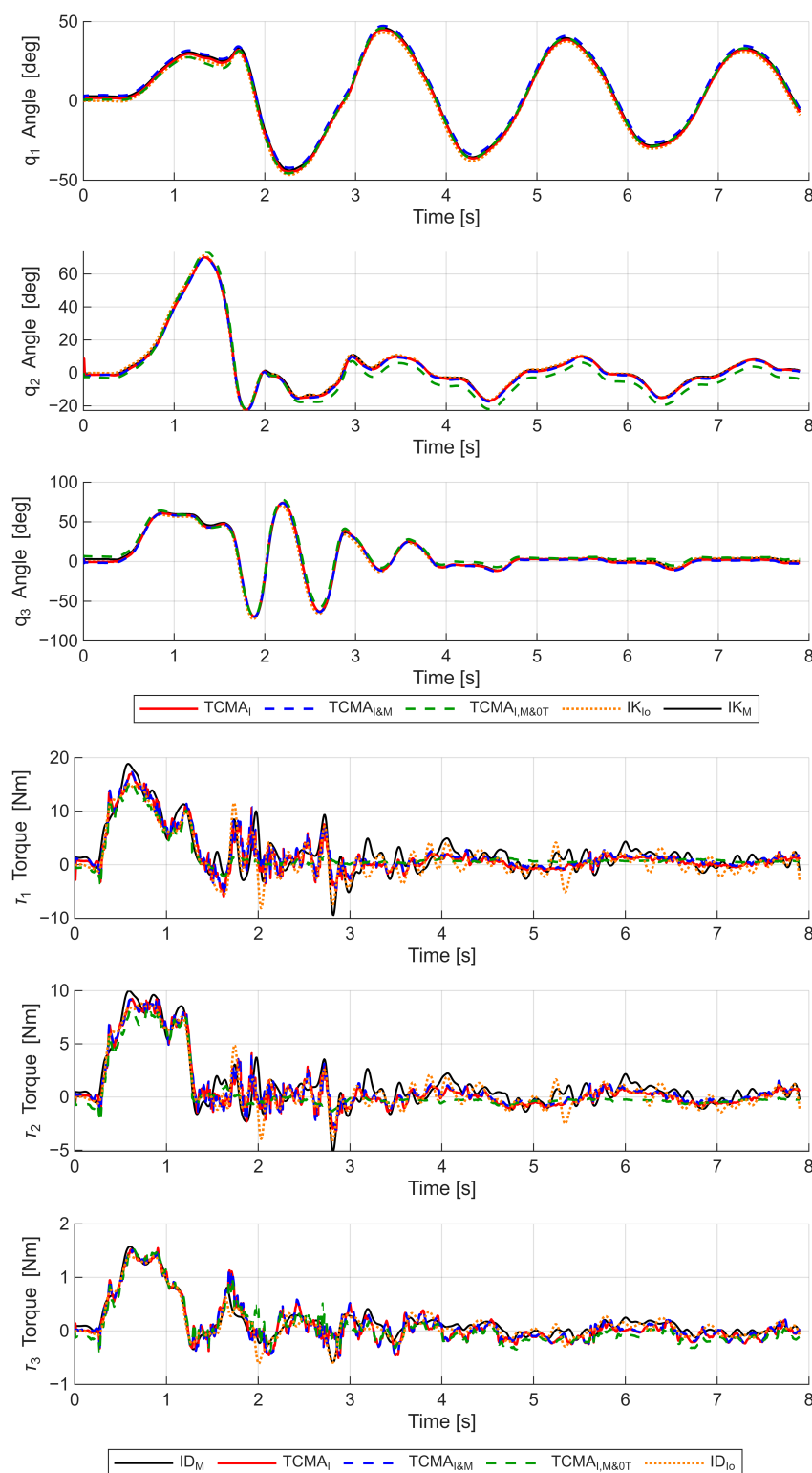


Figure 5. Estimated joint angles (**top**) and torques (**bottom**) for a randomly selected pendulum trial (trial 3) using $TCMA_I$, $TCMA_{I\&M}$, and $TCMA_{I,M\&0T}$, plotted against IK_{I_0}/ID_{I_0} and IK_M/ID_M .

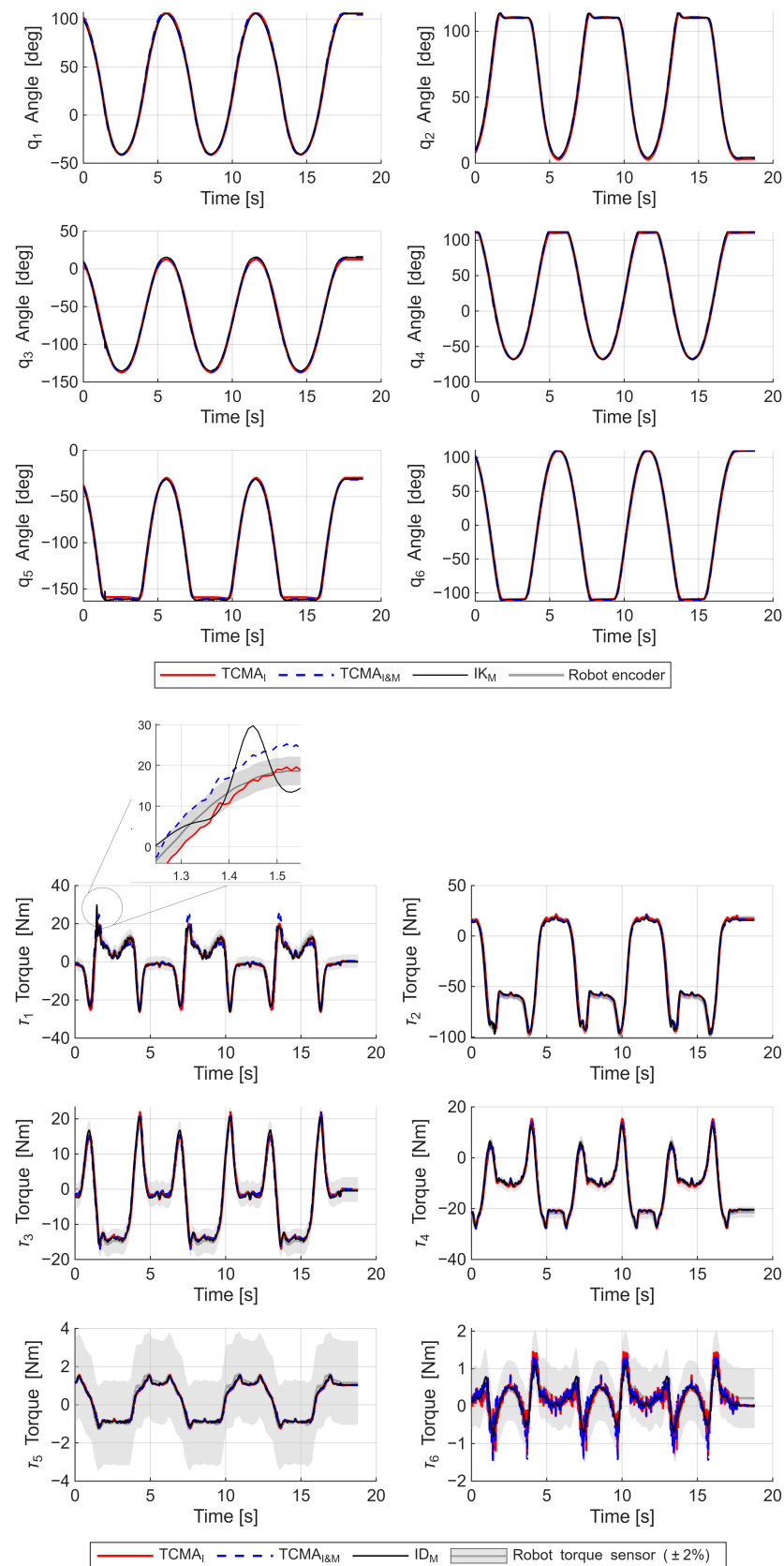


Figure 6. Estimated joint angles (top) and torques (bottom) for trial 1 using TCMA_I and TCMA_{I&M}, plotted against IK_M/ID_M and the robot measurements. The robot torque is shown as the moving-average mean with a shaded $\pm 2\%$ uncertainty band for the maximum rated joint torque.

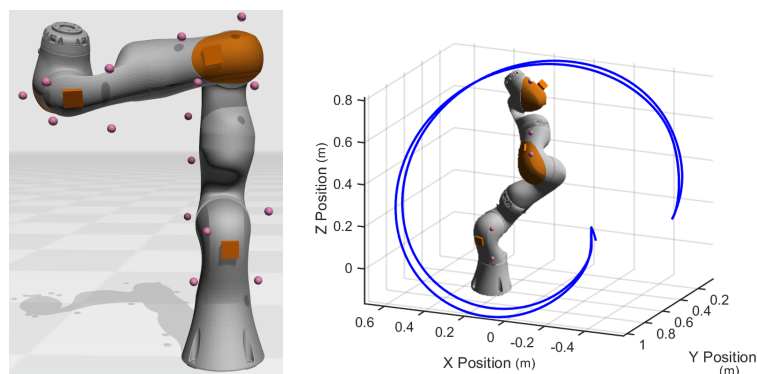


Figure 7. Robot trial 1: (left) initial pose and (right) end-effector motion path.

In the trial shown in Figure 6, all joints except the end-effector were simultaneously actuated. This was consistent across all five trials. Figure 7 shows the resulting end-effector motion path for trial 1. For trial 1, TCMA_I showed a maximum joint angle error of 3.24°, while TCMA_{I&M} showed a maximum joint angle error of 2.84°. Torque errors were below 4.27 Nm for TCMA_I and 3.71 Nm for TCMA_{I&M} (Table 2). Similarly, across trials 2 to 5, TCMA_I maintained maximum errors of 3.01° for joint angles and 2.33 Nm for joint torques (Table 3). Processing the experimental data with the TCMA took approximately 21.5 times longer than the data duration for the robot experiments (e.g., 412 s for a 19.2 s trial) and 9.8 times longer for the pendulum experiments (e.g., 77 s for a 7.9 s trial). Computations were performed using MATLAB R2025b on a Dell Latitude 7440 laptop equipped with an Intel Core i7-1365U processor (13th Gen, 1.80 GHz) with 16 GB RAM.

Table 2. Joint angle and torque RMSDs for robot trial 1, comparing TCMA_I, TCMA_{I&M}, and IK_M/ID_M against the robot measurements. Bold values indicate the highest joint angle and joint torque RMSD values across all methods, with the robot measurements as the reference.

Joint	Angle RMSD (°) Torque RMSD (Nm)		
	TCMA _I	TCMA _{I&M}	IK _M /ID _M
q_1 τ_1	1.76 1.99	1.37 2.23	0.33 1.67
q_2 τ_2	1.54 4.27	0.84 3.71	0.66 2.97
q_3 τ_3	1.44 1.41	1.45 1.13	1.16 1.01
q_4 τ_4	2.31 1.64	1.45 1.51	0.75 1.25
q_5 τ_5	2.20 0.35	1.22 0.34	0.48 0.33
q_6 τ_6	3.24 0.37	2.84 0.38	1.16 0.33

Table 3. Joint angle and torque RMSDs for robot trials 2–5 for TCMA_I versus the robot measurements. Bold values indicate the highest joint angle and joint torque RMSD values per trial, with the robot measurements as the reference.

Joint	Angle RMSD (°) Torque RMSD (Nm)			
	Trial 2	Trial 3	Trial 4	Trial 5
q_1 τ_1	2.72 0.66	0.99 0.64	1.23 0.46	1.58 0.46
q_2 τ_2	1.86 2.33	0.39 1.38	0.87 1.65	0.43 1.81
q_3 τ_3	3.01 0.82	2.00 0.46	2.12 0.33	2.24 0.34
q_4 τ_4	2.06 0.45	0.77 0.32	1.24 0.36	0.95 0.38
q_5 τ_5	1.52 0.08	1.57 0.08	0.86 0.07	1.00 0.07
q_6 τ_6	1.73 0.23	1.42 0.23	1.77 0.22	1.26 0.23

4. Discussion

We developed a tightly coupled motion-capture algorithm (TCMA) that is based on an iterated extended Kalman filter (IEKF) to tightly couple IMU measurements and a multibody system model to estimate the system kinematics and kinetics directly. Unlike traditional IMU-based methods such as OpenSense [22], our formulation does not depend on computing sensor orientations. Hence, it can be used in indoor applications, where there are often strong ferromagnetic disturbances affecting the magnetometer, as demonstrated in the robot experiment.

The results of the trials conducted on the pendulum experimental setup indicate that, with respect to IK_M , the TCMA_I joint angle RMSD values were on average 19% lower than those from IK_{Io} (Table 1). The maximum RMSD between the TCMA_I joint angle estimates and IK_M was 3.75° , compared to 6.56° for IK_{Io} (Table 1).

Across all six pendulum trials, torque estimates from the TCMA with different measurement combinations were within ~ 3 Nm RMSD of the marker-based ID (Table 1). The TCMA with different measurement combinations, particularly the virtual zero-torque measurements, produced torque estimates that fluctuated around zero from about 3 s onward (Figure 5, Table 1). This behavior aligns with the expected physical dynamics of the pendulum, where only the low friction of the joints is present during free passive swinging. However, the lack of joint friction in the model may have caused the observed offset in TCMA_{I,M&OT}'s q_2 estimate (Figure 5). Since the virtual zero-torque measurements pull the estimated torques toward zero, any unmodeled friction torque cannot be absorbed by the torque state and instead shows up as a small bias in the joint angles. This effect is most visible at q_2 , which is the only intermediate joint and therefore the only one coupling two freely swinging segments, so friction contributions from motion on both sides affect it.

The robot trials further demonstrate the ability of our approach to provide accurate motion estimates in environments with strong magnetic distortions, where using the OpenSense approach was not possible. Five motion paths were selected to exploit the high maneuverability of the robot (Figure 7), showcasing the capability of our method to track the corresponding kinematics and kinetics (Figure 6). The TCMA_I joint angle estimates were consistent with measurements in the pendulum trials, demonstrating joint angle RMSD values below 3.25° with respect to the joint encoders (Tables 2 and 3).

The torque estimates produced by TCMA_I were within 4.27 Nm RMSD of the robot's integrated torque sensor measurements across all trials (Tables 2 and 3). Furthermore, the TCMA tracked abrupt changes in the torque profile. For example, around 1.4 s in the τ_1 estimate, the motion capture ID overshoot by approximately 12 Nm, whereas TCMA_I did not (Figure 6). In the worst-case scenario, the RMSD of TCMA_I exceeded the torque sensors' uncertainty by 0.5% in the estimate of τ_2 in trial 1, where the torque RMSD was 4.27 Nm (Table 2).

A key advantage of our approach is the elimination of kinematic filtering requirements inherent to traditional ID methods. Conventional ID approaches (both OpenSense ID and marker-based ID) require low-pass filtering of kinematic data to reduce high-frequency artifacts introduced by inverse kinematics. Hence, selecting an appropriate cutoff frequency presents a significant challenge, as different body segments may move at varying frequencies, making a single cutoff frequency suboptimal across all joints [38]. This issue is particularly relevant in motions involving contact events, where making and breaking contact with the environment introduces higher-frequency dynamics, while limb movements remain predominantly low-frequency [39]. Our algorithm mitigates this limitation by directly estimating kinetics from the dynamic model without requiring explicit cutoff frequency selection, thereby reducing sensitivity to segment-dependent motion speeds and arbitrary filtering choices. This was evident in both the pendulum experiment, where

torque estimates naturally fluctuated around zero during free swinging (Figure 5), and the robot experiment, where abrupt torque changes were accurately tracked (Figure 6).

The inclusion of markers in the TCMA measurement model improved accuracy, with a 15–40% reduction versus the robot joint angle encoder RMSD (Table 2). Additionally, the virtual zero-torque measurements applied during the pendulum experiment produced the smoothest torque profile (Figure 5). These results show the effectiveness of combining different measurement models to enhance accuracy. For example, IMU measurements can be combined with pressure insoles or other force reaction measurements to capture human kinematics and kinetics outside the laboratory.

While our results from multibody mechanical systems are very encouraging and highlight distinct advantages, they rely on several assumptions that may limit the broader applicability of our approach. First, the locations of the IMUs were assumed to be rigid (i.e., free of soft tissue artifacts) and accurately known a priori, which is generally not the case in practice. In this work, we used motion-capture measurements to calibrate the sensor-to-body locations during our experiments. Second, although the TCMA exhibited low sensitivity to the measurement noise covariance matrix R , it was more sensitive to the torque process noise Q_τ . However, the development of adaptive process noise estimation methods lies beyond the scope of this work. Although the TCMA is sequential and thus potentially real-time-capable, currently, it does not yet meet real-time computational requirements.

Future improvements could address the sensor-to-body calibration limitation by incorporating an automatic sensor registration algorithm, which would solve for the sensor orientation and translation with respect to the body segment using the model's joint properties in addition to the sensor measurements [40]. The algorithm could also include an adaptive process and measurement covariance estimator to improve estimation accuracy, particularly when applied to different multibody systems in different environments. Additionally, estimating the gyroscope bias within the system states could further enhance accuracy. The TCMA also facilitates integration of additional sensor modalities, such as markerless motion-capture systems and force sensors, by defining corresponding measurement models in terms of the system states. This could enable estimation of additional quantities, such as joint contact forces and loading, while also improving overall motion estimation accuracy. An automatically differentiable model would provide exact Jacobians instead of numerical perturbations, potentially improving both TCMA accuracy and computational efficiency. Since TCMA uses the dynamics in OpenSim, the same approach can be applied to other models, including human musculoskeletal models that use joint and coordinate definitions commonly used by movement scientists and clinicians. This compatibility with existing biomechanical frameworks could facilitate broader adoption of IMU-based motion capture in clinical settings.

5. Conclusions

We presented a tightly coupled (IEKF-based) motion-capture algorithm (TCMA) that integrates the system's dynamic model with different combinations of sensor measurements (primarily IMU accelerometer and gyroscope measurements) to directly estimate both the kinematics and kinetics of a multibody system. Comparing our IMU TCMA estimates against the ground truth from optical motion capture and robot sensors, we demonstrate an accuracy of less than 3.8° compared to marker-based optical motion capture, which is prevalent in human biomechanical studies.

Integrating multibody dynamics and related constraints into the estimation process ensures that the estimated states are inherently consistent with both kinematic and kinetic relationships, thereby mitigating the inconsistencies typically observed in two-step approaches. Our TCMA enabled reliable performance in magnetically disturbed envi-

ronments, as demonstrated via the robot experiment, where magnetometer-dependent methods such as OpenSense were not feasible. Furthermore, unlike traditional inverse dynamics methods that require low-pass filtering with arbitrary cutoff frequencies, our algorithm provides direct kinetic estimates from the dynamic model. Our algorithm also facilitates integration of additional sensing modalities such as markerless motion-capture systems, enabling further enhancement of estimation accuracy. These results are promising for deploying IMU-based motion capture in real-world environments where optical systems are impractical.

Author Contributions: Conceptualization, H.O., D.d.K., J.B., M.K. and A.S.; methodology, H.O., D.d.K., J.B., M.K. and A.S.; software, H.O. and D.d.K.; validation, H.O., D.d.K. and J.B.; formal analysis, H.O., M.K. and A.S.; investigation, H.O.; resources, H.O.; data curation, D.d.K., J.B., M.K. and A.S.; writing—original draft preparation, H.O.; writing—review and editing, M.K. and A.S.; visualization, H.O. and J.B.; supervision, M.K. and A.S. All authors have read and agreed to the published version of the manuscript.

Funding: The authors received financial support from the Convergence Health & Technology (Convergence Human Mobility Centre), a flagship initiative funded by the Convergence Alliance (TU Delft, Erasmus MC, Erasmus University Rotterdam).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The TCMA is publicly available at <https://github.com/ComputationalBiomechanicsLab/Tightly-Coupled-Motion-capture-Algorithm-TCMA> (accessed on 1 May 2026).

Acknowledgments: We would like to thank Eyal Bar-Kochba from the Johns Hopkins Applied Physics Laboratory for sharing the pendulum experimental data, as well as Luka Peternel from TU Delft for providing access to the robot and data.

Conflicts of Interest: The authors declare no conflicts of interest. The funders had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript; or in the decision to publish the results.

References

1. Menache, A. *Understanding Motion Capture for Computer Animation*; Morgan Kaufmann: Cambridge, MA, USA, 2010.
2. Suo, X.; Tang, W.; Li, Z. Motion Capture Technology in Sports Scenarios: A Survey. *Sensors* **2024**, *24*, 2947. [\[CrossRef\]](#)
3. Liao, Y.; Vakanski, A.; Xian, M.; Paul, D.; Baker, R. A review of computational approaches for evaluation of rehabilitation exercises. *Comput. Biol. Med.* **2020**, *119*, 103687. [\[CrossRef\]](#)
4. van Basten, B.J.H.; Jansen, S.E.M.; Karamouzas, I. Exploiting motion capture to enhance avoidance behaviour in games. In *Proceedings of the Motion in Games*; Springer: Berlin/Heidelberg, Germany, 2009.
5. Salisu, S.; Ruhaiyem, N.I.R.; Eisa, T.A.E.; Nasser, M.; Saeed, F.; Younis, H.A. Motion Capture Technologies for Ergonomics: A Systematic Literature Review. *Diagnostics* **2023**, *13*, 2593. [\[CrossRef\]](#)
6. Das, K.; de Paula Oliveira, T.; Newell, J. Comparison of Markerless and Marker-Based Motion Capture Technologies through Simultaneous Data Collection during Gait: Proof of Concept. *Sci. Rep.* **2023**, *13*, 22880. [\[CrossRef\]](#)
7. Hindle, B.R.; Keogh, J.W.L.; Lorimer, A.V. Inertial-Based Human Motion Capture: A Technical Summary of Current Processing Methodologies for Spatiotemporal and Kinematic Measures. *Appl. Bionics Biomech.* **2021**, *2021*, 6628320. [\[CrossRef\]](#)
8. Liu, S.; Zhang, J.; Zhang, Y.; Zhu, R. A wearable motion capture device able to detect dynamic motion of human limbs. *Nat. Commun.* **2020**, *11*, 5615. [\[CrossRef\]](#) [\[PubMed\]](#)
9. Fang, Z.; Woodford, S.; Senanayake, D.; Ackland, D. Conversion of Upper-Limb Inertial Measurement Unit Data to Joint Angles: A Systematic Review. *Sensors* **2023**, *23*, 6535. [\[CrossRef\]](#) [\[PubMed\]](#)
10. Cho, Y.; Jang, S.; Cho, J.; Kim, M.; Lee, H.; Lee, S.; Moon, S. Evaluation of Validity and Reliability of Inertial Measurement Unit-Based Gait Analysis Systems. *Ann. Rehabil. Med.* **2018**, *42*, 872–883. [\[CrossRef\]](#)
11. Wu, G.; Van Der Helm, F.C.; Veeger, H.E.; Makhsous, M.; Van Roy, P.; Anglin, C.; Nagels, J.; Karduna, A.R.; McQuade, K.; Wang, X.; et al. ISB recommendation on definitions of joint coordinate systems of various joints for the reporting of human joint motion—Part II: Shoulder, elbow, wrist and hand. *J. Biomech.* **2005**, *38*, 981–992. [\[CrossRef\]](#) [\[PubMed\]](#)

12. Kok, M.; Hol, J.D.; Schön, T.B. Using inertial sensors for position and orientation estimation. *Found. Trends Signal Process.* **2017**, *11*, 1–153. [\[CrossRef\]](#)
13. Kuo, A.D. A Least-Squares Estimation Approach to Improving the Precision of Inverse Dynamics Computations. *J. Biomech. Eng.* **1998**, *120*, 148–159. [\[CrossRef\]](#)
14. Sturdy, J.T.; Silverman, A.K.; Pickle, N.T. Automated optimization of residual reduction algorithm parameters in OpenSim. *J. Biomech.* **2022**, *137*, 111087. [\[CrossRef\]](#)
15. de Vries, W.; Veeger, H.; Baten, C.; van der Helm, F. Magnetic distortion in motion labs, implications for validating inertial magnetic sensors. *Gait Posture* **2009**, *29*, 535–541. [\[CrossRef\]](#)
16. van Dijk, M.; Kok, M.; Berger, M.; Hoozemans, M.; Veeger, D. Machine Learning to Improve Orientation Estimation in Sports Situations Challenging for Inertial Sensor Use. *Front. Sport. Act. Living* **2021**, *3*, 670263. [\[CrossRef\]](#) [\[PubMed\]](#)
17. Kok, M.; Schön, T.B. A Fast and Robust Algorithm for Orientation Estimation Using Inertial Sensors. *IEEE Signal Process. Lett.* **2019**, *26*, 1673–1677. [\[CrossRef\]](#)
18. Ojeda, J.; Martínez-Reina, J.; Mayo, J. The effect of kinematic constraints in the inverse dynamics problem in biomechanics. *Multibody Syst. Dyn.* **2016**, *37*, 291–309. [\[CrossRef\]](#)
19. Delp, S.L.; Anderson, F.C.; Arnold, A.S.; Loan, P.; Habib, A.; John, C.T.; Guendelman, E.; Thelen, D.G. OpenSim: open-source software to create and analyze dynamic simulations of movement. *IEEE Trans. Biomed. Eng.* **2007**, *54*, 1940–1950. [\[CrossRef\]](#)
20. Seth, A.; Hicks, J.L.; Uchida, T.K.; Habib, A.; Dembia, C.L.; Dunne, J.J.; Ong, C.F.; DeMers, M.S.; Rajagopal, A.; Millard, M.; et al. OpenSim: Simulating musculoskeletal dynamics and neuromuscular control to study human and animal movement. *PLoS Comput. Biol.* **2018**, *14*, e1006223. [\[CrossRef\]](#) [\[PubMed\]](#)
21. Seth, A.; Sherman, M.; Reinbolt, J.A.; Delp, S.L. OpenSim: A musculoskeletal modeling and simulation framework for in silico investigations and exchange. *Procedia IUTAM* **2011**, *2*, 212–232. [\[CrossRef\]](#) [\[PubMed\]](#)
22. Al Borno, M.; O'Day, J.; Ibarra, V.; Dunne, J.; Seth, A.; Habib, A.; Ong, C.; Hicks, J.; Uhlich, S.; Delp, S. OpenSense: An open-source toolbox for inertial-measurement-unit-based measurement of lower extremity kinematics over long durations. *J. Neuroeng. Rehabil.* **2022**, *19*, 22. [\[CrossRef\]](#)
23. Seth, A.; Sherman, M.; Eastman, P.; Delp, S. Minimal formulation of joint motion for biomechanisms. *Nonlinear Dyn.* **2010**, *62*, 291–303. [\[CrossRef\]](#)
24. Weygers, I.; Kok, M.; De Vroey, H.; Verbeest, T.; Versteyhe, M.; Hallez, H.; Claeys, K. Drift-Free Inertial Sensor-Based Joint Kinematics for Long-Term Arbitrary Movements. *IEEE Sens. J.* **2020**, *79*, 7969–7979. [\[CrossRef\]](#)
25. Dorschky, E.; Nitschke, M.; Seifer, A.K.; van den Bogert, A.J.; Eskofier, B.M. Estimation of gait kinematics and kinetics from inertial sensor data using optimal control of musculoskeletal models. *J. Biomech.* **2019**, *95*, 109278. [\[CrossRef\]](#)
26. Haraguchi, N.; Hase, K. Prediction of ground reaction forces and moments and joint kinematics and kinetics by inertial measurement units using 3D forward dynamics model. *J. Biomech. Sci. Eng.* **2024**, *19*, 23-00130. [\[CrossRef\]](#)
27. Nitschke, M.; Dorschky, E.; Leyendecker, S.; Eskofier, B.M.; Koelewijn, A.D. Estimating 3D kinematics and kinetics from virtual inertial sensor data through musculoskeletal movement simulations. *Front. Bioeng. Biotechnol.* **2024**, *12*, 1285845. [\[CrossRef\]](#) [\[PubMed\]](#)
28. Koelewijn, A.D.; Nitschke, M.; Dorschky, E.; Gambietz, M.; Weiss, A.; Eskofier, B.M.; van den Bogert, A.J. BioMAC-Sim-Toolbox: A MATLAB toolbox for biomechanical motion analysis and creation through simulation. *J. Open Source Softw.* **2025**, *10*, 7945. [\[CrossRef\]](#)
29. Kok, M.; Hol, J.D.; Schön, T.B. An optimization-based approach to human body motion capture using inertial sensors. In *Proceedings of the 19th World Congress The International Federation of Automatic Control*, Cape Town, South Africa, 24–29 August 2014.
30. Escalona, J.L.; Urda, P.; Muñoz, S. A Track Geometry Measuring System Based on Multibody Kinematics, Inertial Sensors and Computer Vision. *Sensors* **2021**, *21*, 683. [\[CrossRef\]](#) [\[PubMed\]](#)
31. Ligris, U.; Pérez-Soto, M.; Michaud, F.; Cuadrado, J. Human motion capture, reconstruction, and musculoskeletal analysis in real time. *Multibody Syst. Dyn.* **2024**, *60*, 3–25. [\[CrossRef\]](#)
32. Vivian, M.; Tagliapietra, L.; Sartori, M.; Reggiani, M. Dynamic Simulation of Robotic Devices Using the Biomechanical Simulator OpenSim. In *Proceedings of the 13th International Conference IAS-13*; Springer: Cham, Switzerland, 2016.
33. Featherstone, R. *Rigid Body Dynamics Algorithms*; Springer: New York, NY, USA, 2008.
34. Simon, D. *Optimal State Estimation: Kalman, H Infinity, and Nonlinear Approaches*; John Wiley & Sons: Hoboken, NJ, USA, 2006.
35. Sherman, M.A.; Seth, A.; Delp, S.L. Simbody: Multibody dynamics for biomedical research. *Procedia IUTAM* **2011**, *2*, 241–261. [\[CrossRef\]](#)
36. KUKA. *KUKA LBR iiwa Specifications—Accuracy Information*; KUKA: Augsburg, Germany, 2024.
37. KUKA. *KUKA LBR iiwa Technical Data*; KUKA: Augsburg, Germany, 2015.
38. Winter, D.A. *Biomechanics and Motor Control of Human Movement*, 2nd ed.; John Wiley & Sons: Hoboken, NJ, USA, 1990.

39. Bisseling, R.W.; Hof, A.L. Handling of impact forces in inverse dynamics. *J. Biomech.* **2006**, *39*, 2438–2444. [[CrossRef](#)]
40. Taetz, B.; Lorenz, M.; Miezal, M.; Bleser-Taetz, G.; Stricker, D. JointTracker: Real-time inertial kinematic chain tracking with joint position estimation. *Open Res. Eur.* **2025**, *4*, 33. [[CrossRef](#)]

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.