

A Tightly Coupled IMU-Based Motion Capture Approach for Estimating Multibody Kinematics and Kinetics

Hassan Osman^{1,*}, Daan de Kanter², Jelle Boelens^{1,2}, Manon Kok², and Ajay Seth¹

¹Department of Biomechanical Engineering, Delft University of Technology, the Netherlands

²Delft Center for Systems and Control, Delft University of Technology, the Netherlands

*h.osman@tudelft.nl

ABSTRACT

Inertial Measurement Units (IMUs) enable portable, multibody motion capture (MoCap) in diverse environments beyond the laboratory, making them a practical choice for diagnosing mobility disorders and supporting rehabilitation in clinical or home settings. However, challenges associated with IMU measurements, including magnetic distortions and drift errors, complicate their broader use for MoCap. 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 (IEKF) to simultaneously estimate the system's kinematics and kinetics. By enforcing kinematic and kinetic properties and utilizing only accelerometer and gyroscope data, our method improves IMU based state estimation accuracy. Our approach is designed to allow for incorporating additional sensor data, such as optical MoCap 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 (DoF) pendulum and a 6 DoF Kuka robot. We demonstrate a maximum Root Mean Square Difference (RMSD) in the pendulum's computed joint angles of 3.75° compared to optical MoCap Inverse Kinematics (IK), which serves as the gold standard in the absence of internal encoders. For the Kuka robot, we observe a maximum joint angles RMSD of 3.24° compared to the Kuka's internal encoders while the maximum joint angles RMSD of the optical MoCap IK compared to the encoders was 1.16°. Additionally, we report a maximum joint torque RMSD of 2 Nm in the pendulum compared to optical MoCap Inverse Dynamics (ID), and 3.73 Nm in the Kuka robot relative to its internal torque sensors.

Introduction

Measurements from Motion Capture (MoCap) systems are used to capture and analyze multibody movements^{1,2}. This multibody movement analysis is essential for a wide range of applications, including human rehabilitation, sports, film animation, and video games^{3–5}. To achieve this, MoCap systems provide measurements from a collection of sensors, from which multibody kinematics and kinetics can be calculated. These measurements can take the form of 3D body marker positions in marker-based MoCap, 3D body orientations in Inertial Measurement Unit (IMU)-based MoCap, or joint center positions in markerless optical MoCap⁶.

Inertial sensors allow multibody MoCap in various environments and settings, unlike optical marker-based systems, which are typically restricted to controlled laboratory conditions^{7,8}. This flexibility makes IMU-based MoCap particularly valuable for applications such as patient rehabilitation, where portability and accessibility are important⁴. However, due to accuracy and reliability limitations, IMU-based MoCap has not been widely used in clinical applications^{9,10}. Our aim is to develop a tightly coupled IMU-based MoCap approach that leverages knowledge of the multibody system's properties alongside IMU measurements to estimate kinematics and kinetics. Our approach also integrates complementary measurements, such as optical MoCap data and zero-torque updates, to enhance estimation accuracy.

IMUs capture linear acceleration and angular velocity using accelerometers and gyroscopes¹¹. Some IMUs also incorporate a magnetometer to measure magnetic field intensity. Using these measurements, the orientation of the IMU can be computed¹¹. Inaccuracies in IMU-derived orientations arise from multiple sources: magnetic distortions (especially indoors), gyroscope bias and noise, and the difficulty of separating inertial accelerations from gravitational acceleration during motion. Consequently, orientation estimates contain errors that propagate through subsequent analyses^{11–14}.

OpenSense is an open-source algorithm that utilizes IMU orientations together with the system's kinematic model to estimate multibody kinematics via inverse kinematics (IK)¹⁵. In this method, virtual IMUs are placed on the model's body segments with the same initial orientations as the real IMUs. Joint angles are then computed by minimizing the error between virtual and experimental IMU orientations during motion. However, accurate IMU orientation estimates are critical for reliable

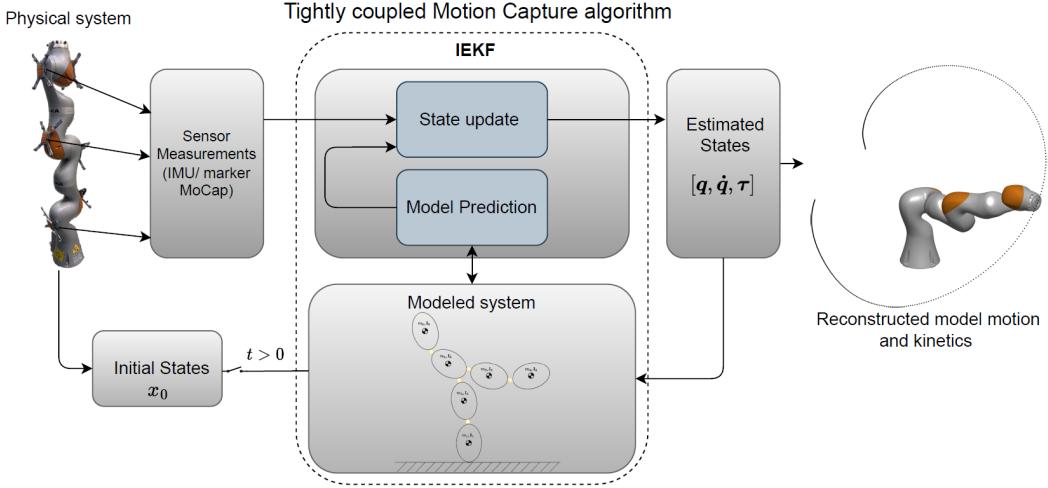


Figure 1. Overview of the tightly coupled sensor and model dynamics approach. Given the initial states, IEKF couples sensor measurements and the multibody system model to estimate generalized joint angles, velocities, and torques.

kinematics with OpenSense.

Joint kinematics can be coupled with IMU measurements by enforcing that the acceleration at the joint center shared by two segments is identical for the IMUs mounted on each segment¹⁶. This coupling leverages the system's kinematic constraints alongside sensor data to improve motion estimates. Since kinetics are not directly estimated, a common approach is to use the estimated kinematics to compute kinetics via inverse dynamics (ID) in a second step^{17–20}. However, this two-step process allows kinematic errors to propagate into the kinetics estimates^{18,21}. Instead, we our pipeline jointly estimates both kinematics and kinetics by integrating sensor measurements and the system's dynamic model.

Incorporating a dynamic model has been previously found to improve the accuracy of the estimated kinematics and kinetics during MoCap. One way to achieve this is by formulating an optimal control problem that computes movement and control trajectories using dynamic models and raw inertial data (accelerometer and gyroscope measurements)^{22–24}. However, these approaches either employed a planar model rather than a full 3D model or relied on simulated inertial measurements generated from marker-based MoCap data. Our approach employs an Iterated Extended Kalman Filter (IEKF) algorithm to integrate an accurate 3D dynamic model with actual IMU measurements. This framework can be extended to incorporate different sensor modalities and measurements, enabling more accurate kinematics and kinetics estimates. Our pipeline was tested using real sensor measurements on two experimental setups—a 6 DoF Kuka robot and a 3 DoF pendulum—each of which provided accurate ground truth data.

1 Methodology

The general outline of the algorithm shown in Figure 1 illustrates the motion estimation process where the sensors measurements are coupled with the system multibody model through an Iterative Extended Kalman Filter (IEKF). The IEKF algorithm utilizes the system multibody model along with the initial or the previous states to predict the current states then both the sensor measurements and the measurement model are used to iteratively update this prediction to estimate the current states. In this work we leverage OpenSim which is an open source software used to model and simulate musculoskeletal systems to model and evaluate the multibody system's equations of motion and kinematic jacobians during the estimation process^{25–28}. We also use OpenSim to derive the measurement model by integrating the sensors measurements with the multibody system. We demonstrate such integration by discussing the general multibody dynamics model, the IMU, marker MoCap and zero torque updates measurement models and the IEKF approach used to couple between them to estimate the motion's kinematics and kinetics simultaneously.

The motion is estimated for a dynamic chain with N_J joints, N_B body segments and N_D generalized coordinates represented as joint angles at each time instance t . The estimated motion states are expressed as the generalized joint angles $\mathbf{q} = (q_1, q_2, \dots, q_{N_D})^\top$, the joints angular speeds $\dot{\mathbf{q}} = (\dot{q}_1, \dot{q}_2, \dots, \dot{q}_{N_D})^\top$ and the generalized forces corresponding to the joint torques $\boldsymbol{\tau} = (\tau_1, \tau_2, \dots, \tau_{N_D})^\top$. Hence the state vector x that we are interested in estimating is given by

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

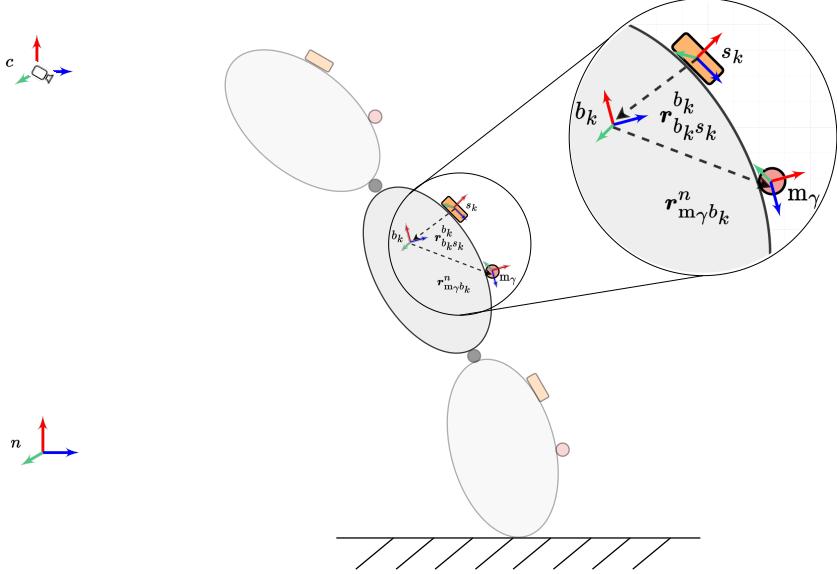


Figure 2. Displays the definitions of the navigation frame n , the optical MoCap frame c , the body frame b_k , the IMU frame s_k , and the marker frame m_γ . It also includes the translation vectors $r_{b_k s_k}^b$ and $r_{m_\gamma b_k}^n$

where $q \in \mathbb{R}^{N_D}$, $\dot{q} \in \mathbb{R}^{N_D}$ and $\tau \in \mathbb{R}^{N_D}$. Figure 2 illustrates the relation between the IMUs and the markers with the body segments where we assume that the sensors are rigidly attached to the body.

1.1 Multibody dynamic model

OpenSim framework was used to model the multibody system and capture the dynamics that describes the states in (1). The system dynamics are presented as

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

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. $\tau^I(q, \dot{q})$ is a function representing the torque that results from gravity, Coriolis force and centrifugal force. Hence, the joints angular accelerations can be expressed as,

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

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

$$\dot{\mathbf{x}} = g(q, \dot{q}, \tau) + e_p, \quad (4)$$

where $e_p \sim \mathcal{N}(0, \mathbf{Q})$ is the process noise and $\mathbf{Q} \in \mathbb{R}^{3N_D \times 3N_D}$ is the process noise covariance matrix.

1.2 General measurement model

The process of estimating the system states (1) via directly coupling the sensor measurements with the system dynamics requires formulating a measurement model. This model describes the system states in terms of the measurements^{11,29–31}. Hence, the system measurement model could be expressed as,

$$y = h(x) + e_{meas} \quad (5)$$

where $e_{meas} \sim N(0, \mathbf{R})$ is the measurement error and $h(x) \in \mathbb{R}^n$ is a function that maps the states to the n sensor measurements. Using the states in (1) the measurement model is expressed as

$$y = \begin{pmatrix} h_1(q, \dot{q}, \tau) \\ h_2(q, \dot{q}, \tau) \\ \vdots \\ h_n(q, \dot{q}, \tau) \end{pmatrix} + e_{meas}. \quad (6)$$

1.3 IMU sensor measurement model

The total number of IMUs fitted on the system's body segments is N_I . Each IMU s_k , $k = 1, \dots, N_I$ has both a gyroscope that measures angular velocities $\omega_{s_k}^{s_k}$ and an accelerometer that measures linear accelerations $\dot{v}_{s_k}^{s_k} = a_{s_k}^{s_k} - g$ where the subscript s_k donates the measurement frame and the superscript s_k donates the frame at which the measurement is expressed in. This notation will be used throughout the rest of this paper. The frame s_k axes are assumed to be aligned with the sensor's axes, with its origin located at the center of the accelerometer triad. The measurement model for the gyroscope angular velocity is given as

$$y_{\omega_k} = \omega_{s_k}^{s_k} + e_{\omega}^{s_k}, \quad (7)$$

where $y_{\omega} \in \mathbb{R}^{3 \times 1}$ is the measured angular velocities and $e_{\omega}^{s_k} \sim \mathcal{N}(0, \Sigma_{\omega,k})$ is the measurement noise. Measured angular velocities can be expressed in terms of the body segments angular velocities which are dependent on the system current generalized coordinates and velocities³² as,

$$y_{\omega_k} = R^{s_k b_k} R^{b_k n}(q) \omega_{b_k}^n(q, \dot{q}) + e_{\omega}^{s_k}, \quad (8)$$

here two rotation matrices are introduced. The constant rotation matrix $R^{s_k b_k}$ between the body frame b_k and the sensor frame s_k is assumed to be known, $R^{b_k n}$ is the rotation matrix that describes the orientation transformation from the navigation frame n to the body segment frame b_k for different generalized joint angles.

The IMU's accelerometer is capable of measuring the linear accelerations that it undergoes including the acceleration due to gravity g^n . The measured acceleration is expressed as

$$y_{\dot{v}} = R^{s_k n}(q)(a_{s_k}^n(q, \dot{q}, \ddot{q}) - g^n) + e_{\dot{v}}^{s_k}, \quad (9)$$

here $y_{\dot{v}} \in \mathbb{R}^{3 \times 1}$ is the measured linear accelerations. The rotation matrix $R^{s_k n}$ is used due to the fact that although the gravitational acceleration is constant in the navigation frame n while its not constant in the senor frame since, the sensor orientation is changing due to the motion. The accelerometer measurement noise is defined as $e_{\dot{v}}^{s_k} \sim \mathcal{N}(0, \Sigma_{\dot{v},k})$. The measured accelerations are dependent on the current generalized joint angles, angular velocities and angular accelerations. Therefore, using (3) and (1) the measured accelerations can be expressed in terms of the system states as,

$$y_{\dot{v}} = R^{s_k b_k} R^{b_k n}(q)(a_{s_k}^n(x) - g^n) + e_{\dot{v}}^{s_k}. \quad (10)$$

The Acceleration $a_{s_k}^n(x)$ is defined in terms of the body's angular velocity, angular acceleration and linear acceleration as

$$\begin{aligned} a_{s_k}^n(x) = & a_{b_k}^n(x) + [\alpha_{b_k}^n(x) \times] \left(R^{n b_k}(q) r_{b_k s_k}^{b_k} \right) + \\ & [\omega_{b_k}^n(q, \dot{q}) \times]^2 \left(R^{n b_k}(q) r_{b_k s_k}^{b_k} \right). \end{aligned} \quad (11)$$

Where $a_{b_k}^n$, $\alpha_{b_k}^n$ and $\omega_{b_k}^n$ are respectively the body segement's linear acceleration, angular acceleration and angular velocity and $r_{b_k s_k}^{b_k}$ is a translation vector expressed in the body frame b_k that maps the position of the IMU frame s_k to the body frame b_k shown in figure 2.

1.4 Marker based and zero torque update measurement models

Marker-based MoCap systems provide marker position information with respect to the system camera frame c . Each body segment can have any number of markers attached for a total number of markers N_M . We denote each individual marker as m_γ where $\gamma = 1, \dots, N_M$. The marker position with respect to the camera frame can then be defined as

$$y_p = P_{m_\gamma}^c + e_p^{m_\gamma}, \quad (12)$$

where $y_p \in \mathbb{R}^{3 \times 1}$ is the measured marker position and $e_p^{m_\gamma} \sim \mathcal{N}(0, \Sigma_{p,\gamma})$ is the position measurement noise. Marker position measurements can then be written in terms of the body position as

$$y_p = R^{c b_k}(q) R^{b_k n}(q) (P_{b_k}^n + r_{m_\gamma b_k}^n(q)) + e_p^{m_\gamma}. \quad (13)$$

The translation vector $r_{m_\gamma b_k}^n \in \mathbb{R}^{3 \times 1}$ is a vector that maps the position of the body segment frame b_k to the marker frame m_γ with respect to the navigation frame n shown in figure 2. This vector and the two matrices $R^{b_k n} \in \mathbb{R}^{3 \times 3}$ and $R^{c b_k} \in \mathbb{R}^{3 \times 3}$ are dependent on the generalized joint angles since the bodies and markers frames change orientations with respect to the camera frame c and the navigation frame n during motion.

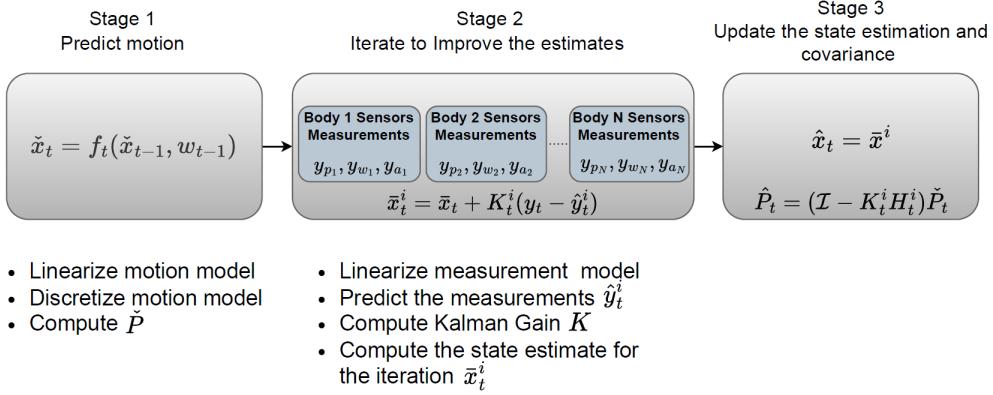


Figure 3. IEKF process: (1) Predict the states using the prior states and the OpenSim model, (2) Refine the state predictions by iteratively linearizing the measurement model either from the IMUs only or the combination of IMUs and other sensors to account for non-linearities then compute the Kalman gain K , (3) Update the state estimation and covariance.

To incorporate zero torque updates to the pipeline, we model virtual joint torque sensors ϕ_j , $j = 1, \dots, N_D$ that assume joint torque measurements of zero. This measurements are expressed as,

$$y_\tau = 0 + e_\tau^{\phi_j}, \quad (14)$$

where $y_\tau \in \mathbb{R}^{3 \times 1}$ is the joint zero torque virtual measurement and $e_{\tau,i}^{\phi_j} \sim \mathcal{N}(0, \Sigma_{\tau,j})$ is the virtual torque measurement noise.

1.5 Tightly Coupled Kinematics and Kinetics estimation algorithm

Our frame-work integrates the dynamic model described in 1.1 with the measurement models described in 1.2, 1.3 and 1.4 using an IEKF pipeline, as shown in Figure 3. The kinematic and dynamic properties of the system are defined using OpenSim^{33,34}. By modeling the sensors within this framework and incorporating them into the system dynamic model, we couple the sensor measurements to the states of the system.

The dynamic model is discretized to predict the future state at time $t + 1$ by numerically integrating the equations of motion using OpenSim's Runge-Kutta-Merson integrator³⁴, resulting in an update equation of the form,

$$x_{t+1} = x_t + \Delta t \cdot g(x_t). \quad (15)$$

The algorithm measurement model that includes all the previously discussed sensors measurement models can then be expressed as

$$y = \begin{pmatrix} y_{\omega,s_1} \\ y_{\dot{v},s_1} \\ y_{p,m_1} \\ y_{\tau,\phi_1} \\ \vdots \\ y_{\omega,s_{N_I}} \\ y_{\dot{v},s_{N_I}} \\ y_{p,m_{N_M}} \\ y_{p,\phi_{N_D}} \end{pmatrix} + e_{\text{meas}}, \quad (16)$$

where $e_{\text{meas}} \sim N(0, \mathbf{R})$ is total measurement error and $\mathbf{R} \in \mathbb{R}^{3(2N_I+N_M+N_D) \times 3(2N_I+N_M+N_D)}$ is the measurement error covariance. The set of functions in (6) that maps the states (1) to the sensor measurements using the sensors measurement models and the

dynamic model discussed in 1 is hence expressed as

$$y = \begin{pmatrix} h_{\omega,1}(x) \\ h_{\dot{\omega},1}(x) \\ h_{p,1}(x) \\ h_{\dot{p},1}(x) \\ \vdots \\ h_{\omega,N_I}(x) \\ h_{\dot{\omega},N_I}(x) \\ h_{p,N_M}(x) \\ h_{\dot{p},N_D}(x) \end{pmatrix} + e_{\text{meas}}. \quad (17)$$

The process of estimating the system states \hat{x}_t and their covariance \hat{P}_t using an IEKF is described in Alg. 1. The first step is the prediction in which the state and covariance are predicted as \check{x}_t and \check{P}_t respectively. In this step the Jacobian $G = \frac{\partial g(\check{x}_t)}{\partial x_t} \in \mathbb{R}^{3N_D \times 3N_D}$ is calculated to compute \check{P}_t . For the second step the algorithm uses the measurements to iteratively update the values of the state and covariance in this step the jacobian $H = \frac{\partial h(x_t)}{\partial x_t} \in \mathbb{R}^{3N_D \times (6N_I+3N_M+N_D)}$ is calculated to compute K_t , \hat{x}_t and \hat{P}_t . In this work, both Jacobians are calculated numerically by perturbations³⁵. To deal with the complex non-linear functions associated with the estimation problem, the algorithm refines the state predictions iteratively as shown in Figure 3 and Alg. 1. The number of these iterations ε is tuned to optimize both the accuracy and the computational efficiency of the algorithm³⁶. We assume that there is no cross-coupling between the predicted states. Hence, we assume an identity process noise distribution matrix $L = \mathbf{I}_{3N_D \times 3N_D}$.

Alg. 1: Tightly Coupled Multibody Kinematics and Kinetics estimation algorithm

Input: Sensor measurements y_t shown in (16), an initial state estimate \hat{x}_0 with covariance \hat{P}_0 , process and measurement noise covariances matrices Q and R demonstrated in (4) and (16) respectively and the number of iterations ε .

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

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

2: **Time update**

 Compute the state prediction \check{x}_t using (15) and \hat{x}_{t-1} . Also compute the state covariance \check{P}_t as

$$\check{P}_t = G_{t-1} \hat{P}_{t-1} G_{t-1}^T + L Q L^T, \quad (18)$$

 with $G_{t-1} = \left. \frac{\partial g(\check{x}_{t-1})}{\partial x_{t-1}} \right|_{x_{t-1}=\hat{x}_{t-1}}$.

3: **Measurement update**

4: **for** $k = 1, \dots, \varepsilon$ **do**

5: Update the state with measurements y_t as

$$\bar{x}_t^{k+1} = \check{x}_t + K_t^k \left(y_t - h_t(\bar{x}_t^k) - H_t^k (\check{x}_t - \bar{x}_t^k) \right), \quad (19)$$

 where $K_t^k = \check{P}_t (H_t^k)^T (H_t^k \check{P}_t (H_t^k)^T + R)^{-1}$ and $H_t^k = \left. \frac{\partial h_t(x_t)}{\partial x_t} \right|_{x_t=\bar{x}_t^k}$.

6: Compute the filtered state and covariance

$$\hat{x}_t = \bar{x}_t^{\varepsilon+1}, \quad \hat{P}_t = (I - K_t^\varepsilon H_t^\varepsilon) \check{P}_t. \quad (20)$$

Both stationary and pre-known dynamic sensor measurements were recorded and then compared against ground truth measurements discussed further in 2. The error between them was used to compute the measurement covariance matrix R . The number of iterations, ε , was chosen experimentally when it was noticed that there was no significant variation between consecutive iterations beyond that value.



Figure 4. An experimental setup with 3DOF is shown on the 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. Sensor locations on the model were registered using Optical MoCap measurements.

1.6 Real world testing on physical systems

Two separate sets of experiments were performed on two different systems to validate our approach. The experimental setups in this study were chosen to ensure accurate ground truth data to validate our proposed algorithm. Therefore, we omitted testing on humans, since precise ground truth for human motion cannot be reliably established.

The first experimental setup used was a 3DOF aluminum pendulum, shown in Figure 4. The pendulum consisted of four bodies and three pin joints. Each body segment was equipped with an IMU and a set of active light-emitting markers. Experiments were conducted on this passive multibody system, in which the lowest body segment shown in silver in Figure 4 was raised to a certain height and then released to swing under its own inertia. Our proposed approach is evaluated for estimating the pendulum motion with three different measurement models:

- IMU measurements
- IMU and optical MoCap measurements
- IMU, optical MoCap, and virtual zero torque measurements

These measurement models demonstrate the ability of our algorithm to integrate different sensor measurements. We compare the output of these measurement models with OpenSim marker-based and IMU-based (OpenSense) Inverse Kinematics (IK) and Inverse Dynamics (ID) across six trials, each with different initial generalized joint angles.

The second experimental setup used was a KUKA LBR 7 iiwa R800 robot shown in Figure 5. Apart from the end effector, the robot consists of 7 bodies and 6 revolute joints. Markers and IMUs were fitted on all the robot body segments except the base and the end effector. We use our approach to estimate the motion of the robot using two different measurement models. The IMU measurements only, and both the IMU measurements and the MoCap measurements. All IMUs were pre-calibrated before each trial in this experiment. IK in this experiment was only conducted using the optical MoCap data. Due to the magnetic disturbances, performing IK using the IMU orientations was not feasible. A total of 5 different trials were conducted, 1 using both the IMU and the MoCap measurements, and 4 using only the IMU measurements.

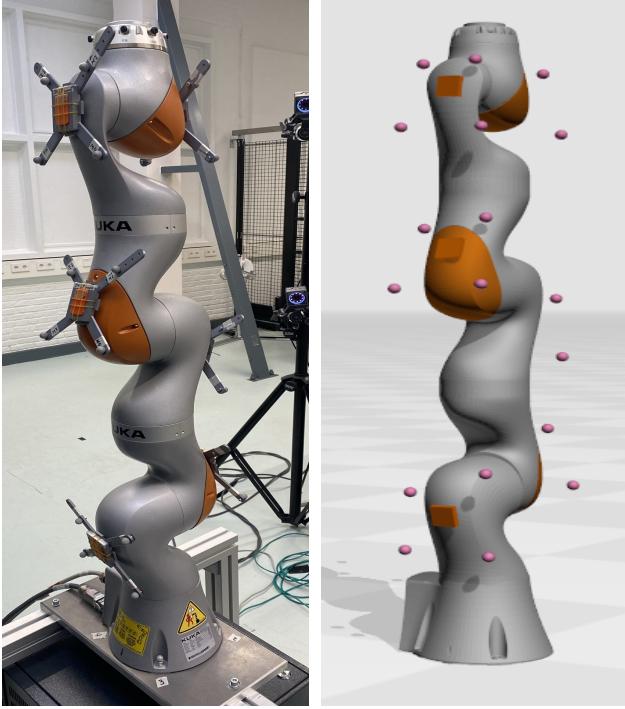


Figure 5. On the left the actual KUKA LBR iiwa R800 that was used where a set of 4 markers and an IMU were placed on each body segment. On the right an OpenSim model that describes the system is shown.

2 Results

In this section, we present the results and findings of the trials conducted on the two experimental setups discussed in 1. For each experimental setup, we illustrate the output of the different measurement models by showing the estimated joint angles and torques plot against the groundtruth for one trial and the Root Mean Squared Difference (RMSD) values for all the trials.

Figure 6 presents the estimated joint angles and torques for trial 3 conducted on the pendulum experimental setup using different measurement models. At time 0 s, the pendulum was in its initial position shown in Figure 4. the pendulum was raised and then left to swing freely under its own inertia from time 1.2 s to 7.8 s. The plots in the figure indicates that the joint angle estimates from the different measurement models closely follow the same trend. However, a steady offset of approximately 2° is observed in the IMU marker-zero torque update q_2 estimates after 2.4 s compared with the other measurement models. The joint torque estimates from the different measurement models also appear to follow the same trend, although each model exhibits a slightly different oscillation pattern. Notably, both the marker and IMU-based Inverse Dynamics (ID) estimates show higher peaks around zero after time 3 s in both τ_1 and τ_2 .

Table 1 presents the generalized joint angles and torques RMSD values of the different measurement models compared to the marker based IK and ID for the six pendulum trials. The table also shows RMSD values of the different measurement models compared with the IMU marker IEKF estimates. In trial 1, the estimate of IMU-based IK (OpenSense-IK) for q_3 had an RMSD of 6.56° compared to 3.49° achieved by IMU-IEKF, indicating a 46.8% marker-IK RMSD reduction. Similarly, in trials 2, 3, and 4, the IMU-IEKF demonstrated RMSD reductions of 31.7%, 48.6%, and 18.6%, respectively, over the OpenSense-IK estimate for q_3 . While in trial 6 OpenSense-IK estimate for q_3 had an RMSD of 3.23° compared to 3.72° achieved by IMU-IEKF showing a reduction of 13.17%. The RMSD values of both the IMU-marker IEKF and IMU-IEKF, calculated with respect to the marker-IK estimates, are closely matched, with a maximum difference of 0.64° . This difference reflects a 17.2% reduction in RMSD for the IMU-marker IEKF q_3 estimate in trial 6.

As for the torque Marker-ID RMSD, the maximum difference between OpenSense-ID and IMU-IEKF was 1.62 Nm in τ_1 during trial 6. The IMU-IEKF and IMU-Marker IEKF RMSD values were close with a maximum difference of 0.10 Nm in τ_1 during trial 6.

Figure 7 presents the estimated generalized joint angles and torques for the Kuka robot trial 1. The initial pose of the robot during this trial is shown in Figure 8. The total trial time was about 18 s where all joints were simultaneously actuated except for the end effector. The end effector motion path of this trial is shown in figure 9. The estimated joint angles of the different measurement models shown in the figure appear to have small differences. However, the RMSD values calculated with respect

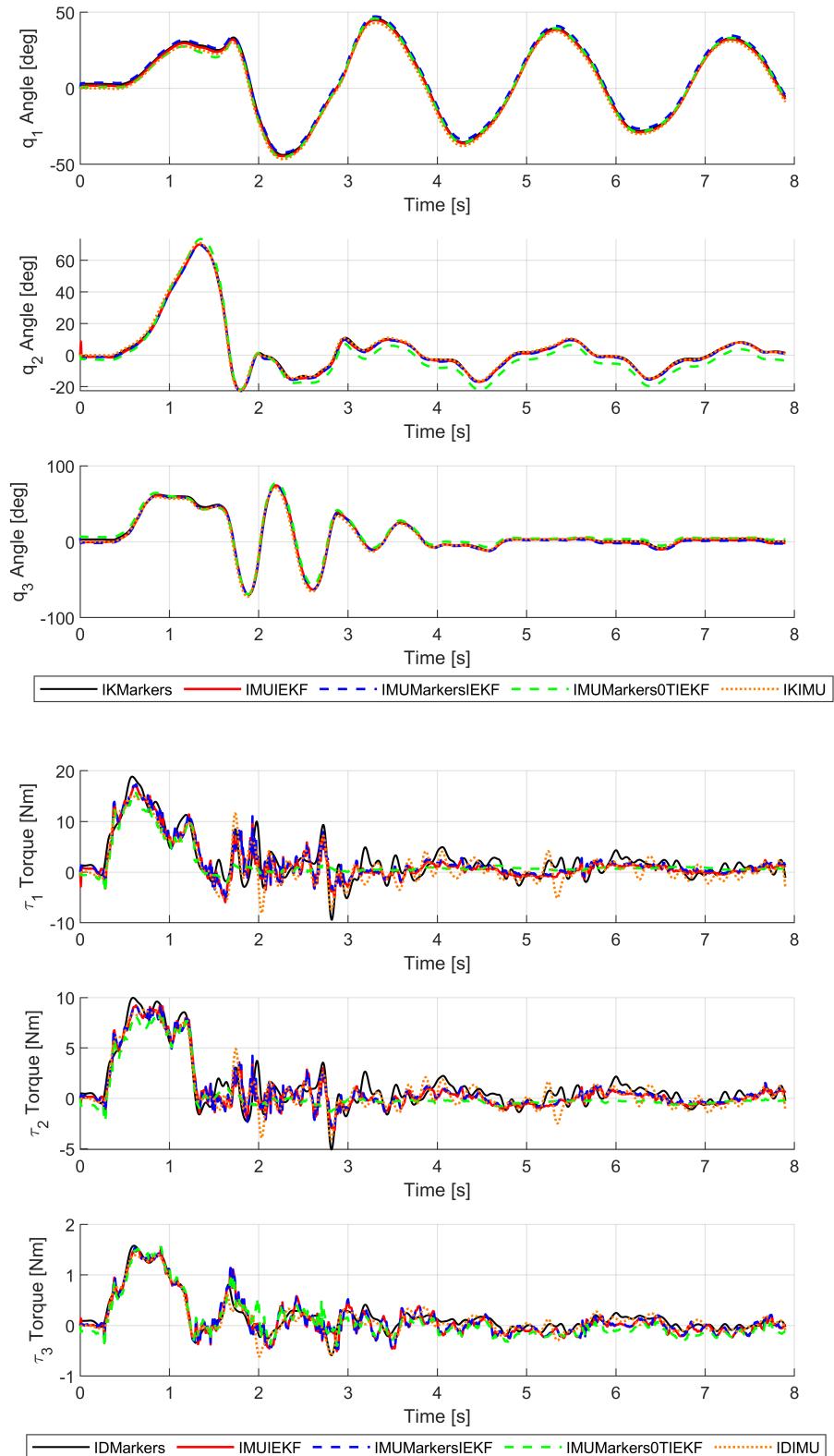


Figure 6. Shows trial 3 estimated generalized joint angles and torques of the 3DOF pendulum using the IMU data only, The IMU and Marker data and The IMU and Marker data with zero torque update all within the proposed tightly coupled IEKF approach. This is plotted against OpenSim's IK/ID.

T	Metric	IMU	IMU Markers	IMU Markers 0T	OS	Markers
1	q_1	1.27—0.10	1.28—	1.29—0.10	0.92—2.13	— 1.28
	q_2	0.91—0.12	0.93—	0.99—0.09	0.92—1.28	— 0.93
	q_3	3.49—0.12	3.54—	4.23—0.93	6.56—3.32	— 3.54
	τ_1	1.05—0.03	1.07—	0.92—0.47	0.78—1.56	— 1.07
	τ_2	0.39—0.01	0.40—	0.34—0.28	0.39—0.66	— 0.40
	τ_3	0.07—0.00	0.07—	0.11—0.07	0.15—0.16	— 0.07
2	q_1	1.10—0.11	1.11—	1.11—0.10	1.54—2.26	— 1.11
	q_2	0.94—0.17	0.97—	1.02—0.11	2.21—2.60	— 0.97
	q_3	3.75—0.10	3.72—	4.16—0.75	5.50—4.05	— 3.72
	τ_1	1.70—0.05	1.70—	1.33—0.82	1.70—1.91	— 1.70
	τ_2	1.01—0.01	1.00—	0.79—0.48	0.75—1.09	— 1.00
	τ_3	0.18—0.00	0.18—	0.21—0.12	0.12—0.26	— 0.18
3	q_1	1.34—0.10	1.35—	1.38—0.10	1.00—1.67	— 1.35
	q_2	0.76—0.16	0.79—	0.87—0.13	1.10—1.27	— 0.79
	q_3	1.64—0.09	1.66—	1.86—0.40	3.18—2.81	— 1.66
	τ_1	1.23—0.05	1.24—	1.25—0.66	1.45—1.52	— 1.24
	τ_2	0.63—0.01	0.63—	0.62—0.39	0.64—0.71	— 0.63
	τ_3	0.14—0.00	0.14—	0.18—0.09	0.06—0.16	— 0.14
4	q_1	1.28—0.13	1.27—	1.29—0.11	1.38—2.57	— 1.27
	q_2	1.19—0.38	1.11—	1.22—0.13	0.65—1.07	— 1.11
	q_3	2.75—0.28	2.69—	3.08—0.56	3.38—3.91	— 2.69
	τ_1	1.53—0.18	1.52—	1.42—0.71	1.34—2.01	— 1.52
	τ_2	0.76—0.05	0.76—	0.69—0.43	0.60—0.89	— 0.76
	τ_3	0.14—0.00	0.14—	0.18—0.10	0.08—0.18	— 0.14
5	q_1	1.05—0.12	1.04—	1.05—0.11	0.86—1.51	— 1.04
	q_2	1.24—0.58	1.18—	1.18—0.18	2.09—2.29	— 1.18
	q_3	2.67—0.68	2.65—	2.94—0.56	2.37—2.60	— 2.65
	τ_1	1.85—0.07	1.85—	1.74—0.69	1.18—2.06	— 1.85
	τ_2	1.02—0.07	1.02—	0.95—0.41	0.60—1.10	— 1.02
	τ_3	0.26—0.02	0.26—	0.26—0.09	0.05—0.26	— 0.26
6	q_1	1.45—0.80	1.27—	1.29—0.19	1.09—1.48	— 1.27
	q_2	1.75—0.43	1.64—	1.67—0.24	1.43—1.51	— 1.64
	q_3	3.72—1.64	3.08—	3.04—1.11	3.23—2.80	— 3.08
	τ_1	3.02—0.52	2.92—	2.51—0.84	1.40—2.85	— 2.92
	τ_2	1.72—0.22	1.68—	1.47—0.50	0.64—1.59	— 1.68
	τ_3	0.48—0.03	0.48—	0.47—0.10	0.09—0.49	— 0.48

Table 1. Joint angle RMSD (°) and joint torque RMSD (Nm) with respect to marker-based inverse kinematics (IK) and inverse dynamics (ID), as well as IMU-Marker IEKF, across six trials. Columns: **IMU** = IMU-only IEKF, **IMU Markers** = IMU markers IEKF, **IMU Markers 0T** = IMU markers with zero torque updates IEKF, **OS** = OpenSense, **Markers** = marker-based IK or ID. Values are formatted as *marker-based RMSD*—*IMU-Marker IEKF RMSD*.

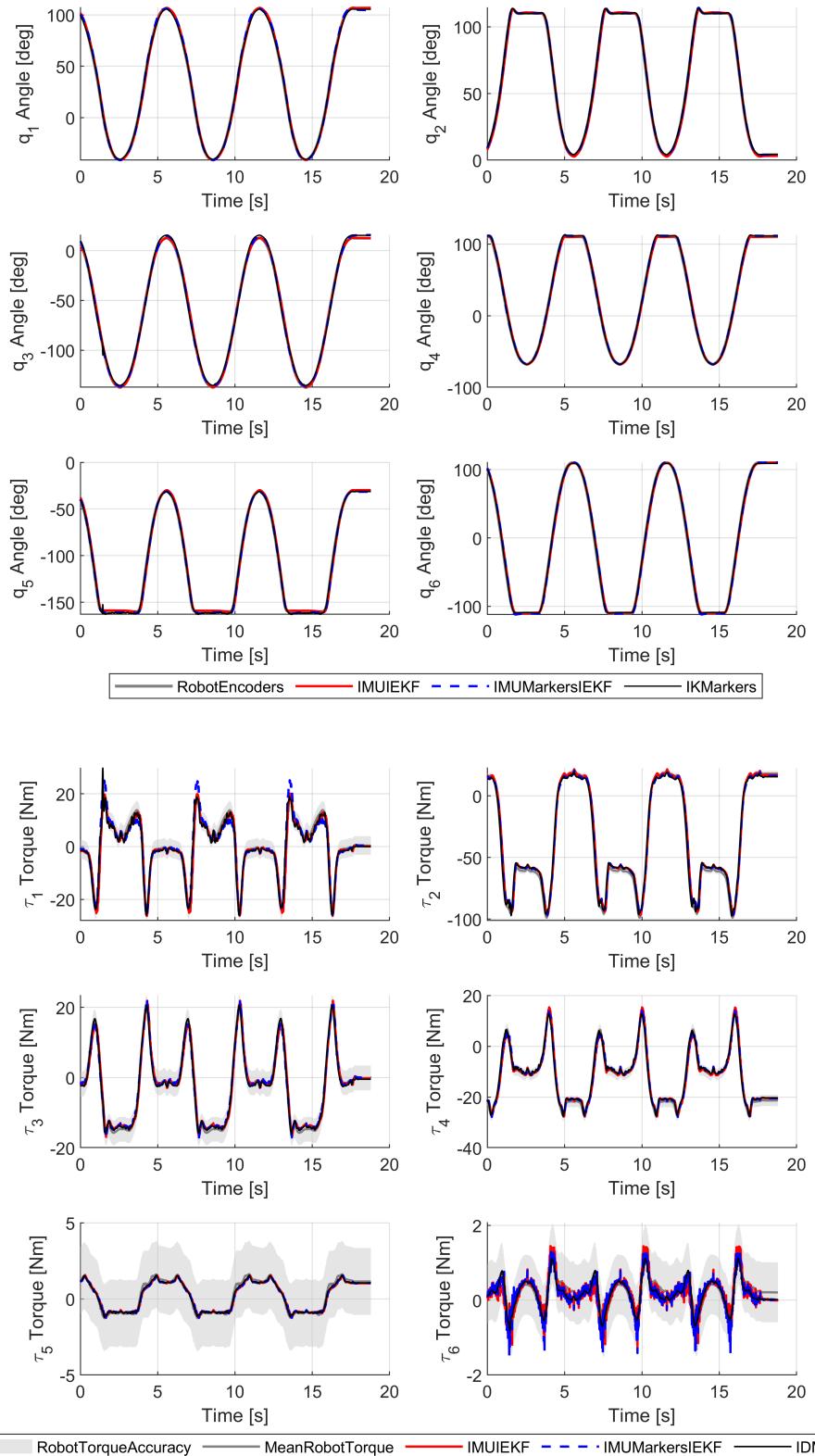


Figure 7. Shows trail 1 estimated joint angles and torques using our proposed IMU-IEKF measurement model, the IMU-Marker IEKF. This is plotted against OpenSim's IK/ID and the robot's own encoders and torque sensors.

Joint	Angle RMSD ($^{\circ}$) / Torque RMSD (Nm)		
	IMU-IEKF	IMU-M-IEKF	Marker-IK/ID
$q_1 - \tau_1$	1.76 — 1.99	1.37 — 2.23	0.33 — 1.67
$q_2 - \tau_2$	1.54 — 4.27	0.84 — 3.71	0.66 — 2.97
$q_3 - \tau_3$	1.44 — 1.41	1.45 — 1.13	1.16 — 1.01
$q_4 - \tau_4$	2.31 — 1.64	1.45 — 1.51	0.75 — 1.25
$q_5 - \tau_5$	2.20 — 0.35	1.22 — 0.34	0.48 — 0.33
$q_6 - \tau_6$	3.24 — 0.37	2.84 — 0.38	0.36 — 0.33

Table 2. Kuka experiment 1 joint angles and torques RMSD between different measurement models and the robot encoders and torque sensors.

to the Kuka encoders in Table 2 show a maximum RMSD of 3.24° for the IMU-IEKF q_6 estimate, 2.84° for the IMU-Marker IEKF q_6 estimate and 1.16° for the marker-IK q_3 estimate. The joint torque estimates of the different models also show small differences. However, The Marker ID τ_1 estimate shows a difference of about 4.2 Nm compared to the other models at time 1.2 s. The RMSD values calculated with respect to the Kuka torque sensors show a maximum of 4.27 Nm for IMU-IEKF, 3.71 Nm for IMU-Marker IEKF and 2.97 Nm in the Marker ID τ_2 estimates.

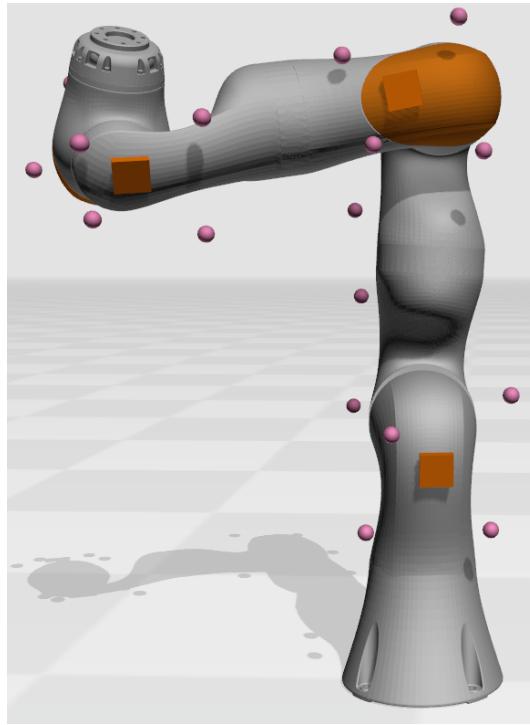


Figure 8. Initial pose of the Kuka in Trial 1.

Table 2 shows both the estimated joint angles and torques for trials 2 to 5 conducted on the Kuka robot. These RMSD values are calculated with respect to the IMU-IEKF estimates and the robot encoders and torque sensors measurements. The total time of these trials ranged between 60 s for trial 2 and 30 s for trial 5. The joint angles RMSD values for these experiments ranged between 3.01° in trial 2 q_3 estimate and 0.39° in trial 3 q_2 estimate. As for the joint torques the RMSD values ranged between 2.33 Nm in trial 2 τ_2 estimate and 0.07 Nm in trial 4 and 5 τ_5 estimate.

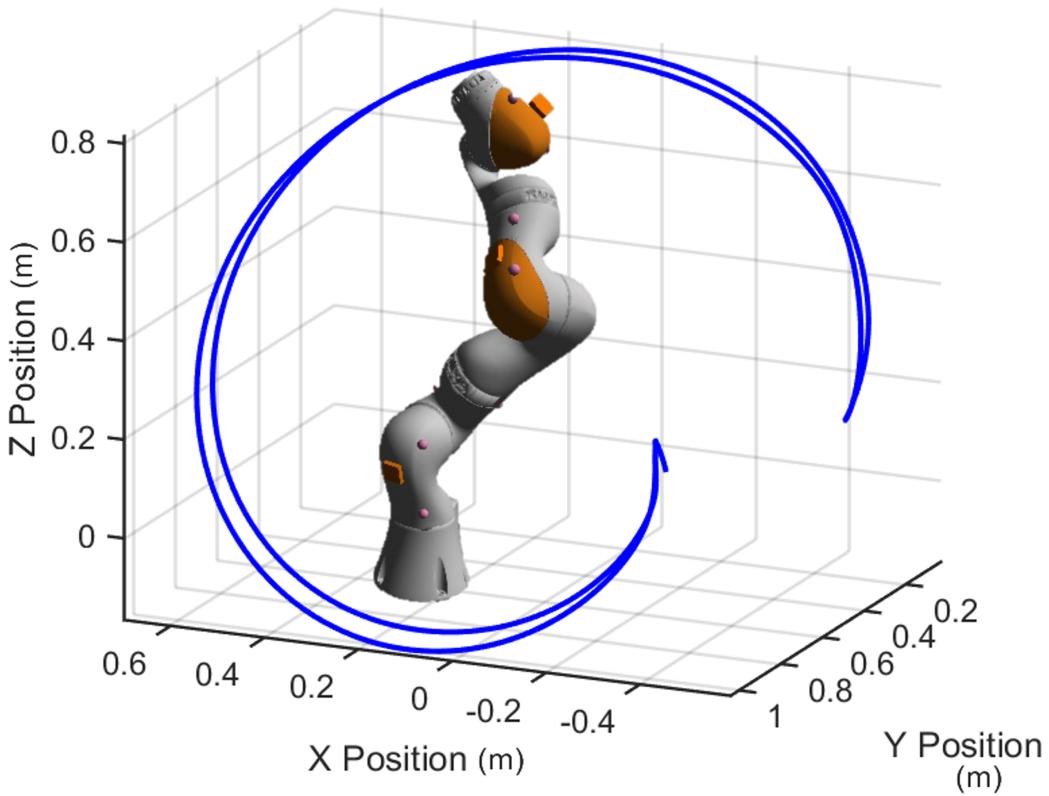


Figure 9. 3D visualization of the end effector motion path for the Kuka Robot trial 1.

Joint	Angle RMSD ($^{\circ}$) / Torque RMSD (Nm)			
	Trial 2	Trial 3	Trial 4	Trial 5
$q_1 - \tau_1$	2.72 — 0.66	0.99 — 0.64	1.23 — 0.46	1.58 — 0.46
$q_2 - \tau_2$	1.86 — 2.33	0.39 — 1.38	0.87 — 1.65	0.43 — 1.81
$q_3 - \tau_3$	3.01 — 0.82	2.00 — 0.46	2.12 — 0.33	2.24 — 0.34
$q_4 - \tau_4$	2.06 — 0.45	0.77 — 0.32	1.24 — 0.36	0.95 — 0.38
$q_5 - \tau_5$	1.52 — 0.08	1.57 — 0.08	0.86 — 0.07	1.00 — 0.07
$q_6 - \tau_6$	1.73 — 0.23	1.42 — 0.23	1.77 — 0.22	1.26 — 0.23

Table 3. Kuka trials 2 to 5 joint angles and torques RMSD between IMU based IEKF and the robot encoders and torque sensors.

3 Discussion

The results of the trials conducted on the pendulum experimental setup indicate that our algorithm generally provided more accurate joint angle estimates than the OpenSense approach. The maximum joint angle RMSD for our algorithm did not exceed 3.75° , while the OpenSense approach reached a maximum RMSD of 6.57° , exceeding the 5° RMSD benchmark suggested for human upper extremity analysis^{37,38}. However, in some trials, the OpenSense estimates showed a reduction in RMSD errors over the IMU-IEKF. Although this reduction did not exceed 0.48° , it may be attributed to a slight misalignment in the sensors translation registration. Since, OpenSense primarily depends on an accurate orientation alignment only, which was the same for both approaches. The results also indicated that the zero torque virtual measurements slightly impacted the accuracy of the generalized joint angles. This effect is likely due to the presence of dynamic friction in the joint bearings, which introduces non-zero torques. Across all 6 trials, torque estimates from the various measurement models showed strong agreement. In contrast to methods that rely on ID, which typically require low-pass filtering to mitigate noise. Our approach provides estimates without being affected by such processing steps. As shown in Figure 6, the measurement models, particularly the zero-torque update IEKF, closely converge toward zero torque from around 3 s onward. This behavior aligns well with the expected physical dynamics of the pendulum, where minimal torque is anticipated during this phase of the motion.

The Kuka results show the ability of our approach to provide accurate motion estimates in environments of high magnetic distortions where this was not possible with the OpenSense approach. The 5 trials motion paths were chosen to utilize the high maneuverability of the Kuka robot, demonstrating the ability of our proposed approach to track the corresponding kinematics and kinetics. The RMSD values presented in Table 2 and Table 3 show joint angles RMSD that are consistent with the pendulum trials with values not exceeding 3.24° . As for the torque estimates the results illustrates the ability of the IMU-IEKF to closely follow the trajectory providing torque estimates that are aligned with robot's torque sensors given that the Kuka's torque sensors have an accuracy of $\pm 2\%$ of the maximum torque which ranges between 176 Nm and 40 Nm for joints 1 to joint 6^{39,40}. This torque accuracy is shown in Figure 7 where it can be noticed that nearly all the torque estimates were between the accuracy bounds of the torque sensors shown in gray. Figure 7 demonstrates that our proposed pipeline is able to track the abrupt changes in the torque profile accurately. In τ_1 estimate at time 1.2 s the already filtered inverse dynamics had an overshoot of 4.2 Nm. However, both the IMU-IEKF and the IMU-Marker IEKF were able to track the robot's torque sensor measurements. The maximum torque RMSD was 4.27 Nm which constitutes approximately 2.5% of the maximum joint torque which is comparable to the $\pm 2\%$ accuracy of the Kuka's torque sensor. The inclusion of markers in the measurement model improved accuracy, by a 15%-40% reduction in the joint angles RMSD values when using the IMU-Marker IEKF compared with the IMU-IEKF. these results show the effectiveness of combining different measurement models to enhance accuracy. Although the marker-based IK and ID estimates show lower RMSD values than the IMU-IEKF, it's important to note the accuracy of the torque sensor. Additionally, IMUs are more cost-effective and can be used in a wider range of environments and settings.

Certain assumptions were made in this work, highlighting potential limitations. First, the locations of the IMUs were assumed to be rigid (i.e., no soft tissue artifacts) and accurately known a priori, which may not always hold in practical applications. Second, the process noise Q should ideally be estimated based on the model and the current system state, but this remains challenging. The highly nonlinear nature of the system model complicates the computation of the measurement model Jacobian H_t . Although the IEKF mitigates these challenges by iterating the linearization process, this results in a computationally expensive algorithm.

Future improvements could address these limitations 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 kinematic properties in addition to the sensors measurements. The algorithm can also include an adaptive process and measurement covariance estimator to improve the estimation accuracy specially if applied on different multibody systems in different environments. Additionally, estimating the gyroscope bias within the system states could enhance accuracy. The use of a differentiable model could also improve Jacobian calculation, potentially balancing both accuracy and computational efficiency.

3.1 conclusions and future work

In this work, we presented a tightly coupled IMU-based motion capture (MoCap) algorithm that integrates various measurement models to estimate both the kinematics and kinetics of a multibody system using an IEKF and the system's dynamic model. The proposed approach was validated using two different multibody systems: a non-actuated pendulum and an actuated robotic arm. Our results demonstrated that the algorithm can accurately estimate both kinematics and kinetics solely from IMU measurements. This capability enables motion capture beyond constrained laboratory environments, facilitating accessible human kinematic and kinetic estimation.

Integrating system kinematics and dynamic constraints into the estimation process was shown not only to enhance accuracy but also to mitigate the effects of omitting magnetometer measurements, thereby ensuring reliable estimates even in environments with strong magnetic disturbances. Additionally, the algorithm's ability to integrate multiple sensors further improved estimation accuracy.

References

1. Menache, A. *Understanding Motion Capture for Computer Animation* (Morgan Kaufmann, 2010).
2. Moeslund, T. B., Hilton, A. & Krüger, V. A survey of advances in vision-based human motion capture and analysis. *Comput. Vis. Image Underst.* (2006).
3. Suo, X., Tang, W. & Li, Z. Motion capture technology in sports scenarios: A survey. *Sensors* (2024).
4. Liao, Y., Vakanski, A., Xian, M., Paul, D. & Baker, R. A review of computational approaches for evaluation of rehabilitation exercises. *Comput. Biol. Medicine* (2020).
5. van Basten, B. J. H., Jansen, S. E. M. & Karamouzas, I. Exploiting motion capture to enhance avoidance behaviour in games. In *Motion in Games* (Springer Berlin Heidelberg, 2009).
6. Salisu, S. *et al.* Motion capture technologies for ergonomics: A systematic literature review. *Diagnostics* (2023).
7. 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. Reports* (2023).
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).
9. Fang, Z., Woodford, S., Senanayake, D. & Ackland, D. Conversion of upper-limb inertial measurement unit data to joint angles: A systematic review. *Sensors* **23** (2023).
10. Cho, Y. *et al.* Evaluation of validity and reliability of inertial measurement unit-based gait analysis systems. *Annals Rehabil. Medicine* (2018).
11. Kok, M., Hol, J. D. & Schön, T. B. *Using inertial sensors for position and orientation estimation* (IEEE, 2017).
12. 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).
13. 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. Sports Active Living* (2021).
14. Kok, M. & Schön, T. B. A fast and robust algorithm for orientation estimation using inertial sensors. *IEEE Signal Process. Lett.* (2019).
15. Al Borno, M. *et al.* Opensense: An open-source toolbox for inertial-measurement-unit-based measurement of lower extremity kinematics over long durations. *J. NeuroEngineering Rehabil.* (2022).
16. Weygers, I. *et al.* Drift-Free Inertial Sensor-Based Joint Kinematics for Long-Term Arbitrary Movements. *IEEE Sensors J.* (2020).
17. Winter, D. A. *Biomechanics and Motor Control of Human Movement* (John Wiley & Sons, 1990), 2nd edn.
18. Kuo, A. D. A least-squares estimation approach to improving the precision of inverse dynamics computations. *J. Biomech. Eng.* **120** (1998).
19. Riemer, R. & Hsiao-Wecksler, E. T. Improving joint torque calculations: Optimization-based inverse dynamics to reduce the effect of motion errors. *J. Biomech.* (2008).
20. Faber, H., van Soest, A. J. & Kistemaker, D. A. Inverse dynamics of mechanical multibody systems: An improved algorithm that ensures consistency between kinematics and external forces. *PLoS One* (2018).
21. Ojeda, J., Martínez-Reina, J. & Mayo, J. The effect of kinematic constraints in the inverse dynamics problem in biomechanics. *Multibody Syst. Dyn.* (2016).
22. 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.* **95** (2019).
23. 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).
24. 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).
25. Delp, S. L. *et al.* Opensim: open-source software to create and analyze dynamic simulations of movement. *IEEE Transactions on Biomed. Eng.* (2007).

- 26.** Seth, A. *et al.* Opensim: Simulating musculoskeletal dynamics and neuromuscular control to study human and animal movement. *PLoS Comput. Biol* (2018).
- 27.** Seth, A., Sherman, M., Reinholt, J. A. & Delp, S. L. Opensim: a musculoskeletal modeling and simulation framework for in silico investigations and exchange. *Procedia IUTAM* **2** (2011).
- 28.** 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* (2016).
- 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* (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).
- 31.** Lugrís, U., Pérez-Soto, M., Michaud, F. & Cuadrado, J. Human motion capture, reconstruction, and musculoskeletal analysis in real time. *Multibody Syst. Dyn.* (2024).
- 32.** Featherstone, R. *Rigid Body Dynamics Algorithms* (Springer, New York, 2008).
- 33.** Seth, A., Sherman, M., Eastman, P. & Delp, S. Minimal formulation of joint motion for biomechanisms. *Nonlinear Dyn.* (2010).
- 34.** Sherman, M. A., Seth, A. & Delp, S. L. Simbody: multibody dynamics for biomedical research. *Procedia IUTAM* (2011).
- 35.** Nian-dong, L., Kun, D., Jia-peng, T. & Wei-xin, D. Analytical Solution of Jacobian Matrices of WDS Models. *Procedia Eng.* (2017). XVIII International Conference on Water Distribution Systems, WDSA2016.
- 36.** Havlík, J. & Straka, O. Performance evaluation of iterated extended Kalman filter with variable step-length. *J. Physics: Conf. Ser.* (2015).
- 37.** Morrow, M. M. B., Lowndes, B., Fortune, E., Kaufman, K. R. & Hallbeck, M. S. Validation of inertial measurement units for upper body kinematics. *J. Appl. Biomech.* (2017).
- 38.** Fang, Z., Woodford, S., Senanayake, D. & Ackland, D. Conversion of upper-limb inertial measurement unit data to joint angles: A systematic review. *Sensors (Basel)* (2023).
- 39.** AG, K. *KUKA LBR iiwa Technical Data* (2015).
- 40.** AG, K. *KUKA LBR iiwa Specifications - Accuracy Information* (2024).