Calibration-free IMU-based Kinematic State Estimation for Robotic Manipulators*

Michael Fennel, Lukas Driller, Antonio Zea, Uwe D. Hanebeck

Abstract—The precise knowledge of a robot manipulator’s kinematic state including position, velocity, and acceleration is one of the base requirements for the application of advanced control algorithms. To obtain this information, encoder data could be differentiated numerically. However, the resulting velocity and acceleration estimates are either noisy or delayed as a result of low-pass filtering. Numerical differentiation can be circumvented by the utilization of gyroscopes and accelerometers, but these suffer from a variety of measurement errors and nonlinearity regarding the desired quantities. Therefore, we present a novel, real-time capable kinematic state estimator based on the Extended Kalman filter with states for the effective sensor biases. This way, the handling of arbitrary inertial sensor setups is made possible without calibration on manipulators composed of revolute and prismatic joints. Simulation experiments show that the proposed estimator is robust towards various error sources and that it outperforms competing approaches. Moreover, the practical relevance is demonstrated using a real manipulator with two joints.

I. INTRODUCTION

In theory, estimating a manipulator’s kinematic state consisting of joint position, velocity, and acceleration is a straightforward task when joint position measurements are available. In practice, however, noise and discretization of the joint encoders will result in poor signal quality and delayed signals after the numerical differentiation required for velocity and acceleration estimates. To compensate for the noise introduced, the signals can be filtered. However, this introduces phase delays for higher frequencies resulting in further delayed estimates. While this is not a serious issue in classical industrial robot applications, it represents a challenge in the context of collaborative robots, where an accurate and non-delayed kinematic state estimation is among the necessary information for fast collision detection [1]. Beyond that, kinematic state data is required in a variety of robotic disciplines, ranging from model identification [2] over trajectory tracking [3] to force control [4].

A promising option to overcome the limitations of pure encoder sensor setups regarding the estimation of velocities and accelerations is the addition of inertial measurement units (IMUs), consisting of gyroscopes and accelerometers, to the manipulator. Driven by reduced costs and the increased availability, the integration and utilization of such sensors has become an actively researched topic in the last two decades:

*This work was supported by the ROBDEKON project of the German Federal Ministry of Education and Research.

All authors are with the Intelligent Sensor-Actuator-Systems Laboratory (ISAS), Institute for Anthropomatics and Robotics, Karlsruhe Institute of Technology (KIT), Germany michael.fennel@kit.edu, lukas.driller@student.kit.edu, antonio.zea@kit.edu, uwe.hanebeck@kit.edu.

In [5], Zhu et al. presented two fusion approaches that utilize frequency weighting to combine acceleration measurements with encoder signals to estimate the end effector velocity of a manipulator in 3D space. In a similar paper, Hedberg et al. [6] presented an algorithm to fuse angular rate and linear acceleration measurements from end effector mounted sensors with the pose estimate from forward kinematics to improve information about the end effector trajectory. Although both authors utilize acceleration measurements, they do not provide smoothed acceleration estimates beyond the raw, noisy acceleration measurements. Furthermore, all estimates are provided in Cartesian space instead of joint space, even though the latter is more universal due to the unambiguity of forward kinematics. This limitation is compensated in the work of Chen et al. [7], where an end effector accelerometer is used to estimate the joint states of a robot with elastic actuators that do not possess load side encoders. For this, a dynamic model of the robot is paired with an optimization problem that has an acceleration constraint and a set of decoupled Kalman filters. It is obvious to deploy multiple IMUs if an estimation in the joint space instead of the Cartesian space is desired. For this reason, Vihonen et al. [8] proposed a sensor configuration where one triaxial gyroscope is attached to each link. The authors also show how the joint acceleration can be determined by adding one additional triaxial and three single-axis accelerometers. Similar to this, Rotella et al. introduced a setup in [9] with one triaxial gyroscope and two triaxial accelerometers attached to each link to calculate joint velocities and accelerations directly from raw measurements in a first step. In a second step, two Kalman filters are used to smooth the data and to ensure consistency between the position data and its derivatives.

Although a large number of sensors can result in improved accuracy, it can be beneficial to reduce the number of sensors for economical and constructive reasons. For this reason, Birjandi et al. presented an estimator in [10], [11] that utilizes one accelerometer and optionally one gyroscope per link in combination with an extended Kalman filter (EKF) to estimate all joint velocities and accelerations.

All solutions up to this point share the disadvantage of requiring an accurate sensor calibration regarding placement, bias, and scale factor, which is hard to achieve, especially in industrial environments. In contrast, the method in [12] uses one uncalibrated accelerometer per link to estimate a bias-polluted joint acceleration vector. Thereafter, the smoothened (and therefore delayed), but bias-free accelerations calculated from encoder data are used to compensate for the current acceleration bias. Although this method works well in
the given acceleration control application, it can neither incorporate gyroscope data nor provide velocity estimates beyond filtered position derivatives. This becomes a practical limitation if the last accelerometer is situated close to the revolute axis of the last joint.

This paper aims to overcome the presented limitations of the current state of the art by contributing a novel, unified, EKF-based framework for estimating the kinematic state of robot manipulators, which

- is hard real-time capable (i.e., execution frequency $\geq 1\, \text{kHz}$),
- does not depend on accurate sensor calibration information, such as exact pose, scale factor error, bias, and cross-axis-sensitivity,
- can be used with arbitrary inertial sensor setups, and
- provides good estimation quality regarding all kinematic quantities over a high bandwidth.

The remainder of this paper is structured as follows: Based on the notation from Section II, the problem is formalized in Section III. In Section IV, the state estimation including system and measurement models is introduced. Finally, a detailed evaluation based on simulations and experiments on real hardware is presented in Section V and Section VI, respectively, before the paper is concluded in Section VII.

II. NOTATION

Throughout this paper, vectors are printed underlined and matrices are printed in bold. Positions, linear velocities and linear accelerations are denoted with $\vec{x}$, $\vec{v}$, and $\vec{a}$, respectively. For rotations, rotation matrices $\mathbf{R}$ are used in combination with angular velocities $\vec{\omega}$ and angular accelerations $\vec{\alpha}$. Furthermore, $\vec{q}$ is the generalized joint configuration vector. The unit or zero vectors $\vec{r}$ and $\vec{t}$ define the rotational and translational axes of joints, respectively. This means that $\vec{r} = \vec{0}_3$ holds for prismatic joints and $\vec{t} = \vec{0}_3$ for revolute joints. Superscripts denote the resolving coordinate frame in which a quantity is expressed. The subscripts denote the reference frame (if applicable) and the object frame. For example, $x_{BC}^A$ represents the position of point $C$ relative to frame $B$, given in coordinates of frame $A$. Similarly, $R_{AB}$ describes the orientation of frame $B$ relative to frame $A$, resulting in a set of mathematical rules that are explained in depth in [13]. For brevity, the cross product $\vec{\omega} \times \vec{x}$ can be written as $\Omega \vec{x}$, where the uppercase $\Omega = [\vec{\omega} \times ]$ is the skew-symmetric matrix of the lowercase $\vec{\omega}$. This definition is also valid for other vectors, e.g., $\vec{r}$. To reference the $i$-th element of a vector, $[i]$ is appended to the subscript. Likewise, the notation $[i; i]$ can be used to address the first to $i$-th element of a vector. All quantities are given in SI units unless otherwise specified.

In the remainder, the coordinate frames as illustrated in Fig. 1 are used. The world frame $W$ is considered an inertial frame and marks the stationary base of the robot. Each joint $i$ of the robot is located between the end frame of the previous link $\beta_{i-1}$ and the begin frame of the next link $\beta_i$.

![Fig. 1: Definition of the frames on the considered kinematic chain with the link begin frames, the link end frames, and the joint axes. An inertial sensor $S$ is attached to the link after joint 3 in this example.](image)

III. PROBLEM FORMULATION

In this paper, a serial kinematic chain with $n$ joints, which are either prismatic or revolute joints, is considered. It is assumed that the forward kinematics is known precisely and that each joint provides noisy (load-side) position measurements from an encoder. Along the kinematic chain, $n_a \geq 0$ triaxial accelerometers $A = \{A_1, \ldots, A_{n_a}\}$ and $n_g \geq 0$ triaxial gyroscopes $G = \{G_1, \ldots, G_{n_g}\}$ are distributed in an arbitrary fashion.\(^1\) For each sensor $S \in G \cup A$, the mounting pose of the sensing frame relative to the parent link begins frame $p(S)$, i.e., $\vec{R}_p(S) \in \mathbb{R}^3$ and $\Omega_p(S) \in \mathbb{R}^3$, is known within typical manufacturing tolerances. However, the exact calibration of the sensors, including bias, scale-factor error, cross-axis-sensitivity, nonlinearity, and pose offset (e.g., originating from soldering), is not known.

The goal in this setup is to find accurate and undelayed estimates of the real joint positions $q \in \mathbb{R}^n$, velocities $\dot{q} \in \mathbb{R}^n$, and accelerations $\ddot{q} \in \mathbb{R}^n$ based on the raw measurements from encoders $\dot{q} \in \mathbb{R}^n$, gyroscopes $\vec{\omega}_{WG_1}, \ldots, \vec{\omega}_{WG_{n_g}} \in \mathbb{R}^3$, and accelerometers $\vec{a}_{WA_1}, \ldots, \vec{a}_{WA_{n_a}} \in \mathbb{R}^3$. Here, the term undelayed encompasses not only phase delays but also the calculation under real-time constraints.

IV. ESTIMATOR DESIGN

In this section, the estimator that combines all available measurements into a single kinematic state is explained. We chose an EKF architecture for its ability to handle nonlinear measurement functions, avoiding the introduction of sensor placement constraints as seen in [8] and [9]. Furthermore, the Kalman filter facilitates intuitive parameter tuning.

A. System Model

For the filter, a linear constant jerk motion model including all joints at once is combined with a constant bias model

\(^1\)Although the distribution can be arbitrary, it is preferred to have a configuration that effectively permits the observation of joint velocities and accelerations from the inertial sensor measurements.
resulting in the time-discrete system model

\[
\begin{pmatrix}
  \dot{q} \\
  \dot{\tilde{g}}_k \\
  \dot{\tilde{g}}_k \\
  \tilde{b}_k
\end{pmatrix}_{k+1} =
\begin{pmatrix}
  I & \Delta t I & \frac{1}{2} \Delta t^2 I & \frac{1}{6} \Delta t^3 I & 0 & 0 \\
  0 & I & \Delta t I & \frac{1}{2} \Delta t^2 I & 0 & 0 \\
  0 & 0 & I & \frac{1}{2} \Delta t^2 I & 0 & 0 \\
  0 & 0 & 0 & 0 & 0 & I \\
  b_{g} & 0 & 0 & 0 & 0 & 0
\end{pmatrix}
\begin{pmatrix}
  \dot{q} \\
  \dot{\tilde{g}}_k \\
  \dot{\tilde{g}}_k \\
  \tilde{b}_k
\end{pmatrix}_{k} + w_k
\]

(1)

with the sampling time \(\Delta t\). Here, the stacked biases

\[
\begin{align*}
  \tilde{b}_g &= (b_{G_1}^T, \ldots, b_{G_{n_g}}^T)^T \in \mathbb{R}^{3n_g} \quad \text{and} \\
  \tilde{b}_a &= (b_{A_1}^T, \ldots, b_{A_{n_a}}^T)^T \in \mathbb{R}^{3n_a}
\end{align*}
\]

(2)

(3)

describe the current bias values of each triaxial gyroscope and accelerometer, respectively, which are equal to the actual inertial sensor biases under ideal conditions. Under real conditions, the gyroscope and accelerometer biases will be approximated by the current total bias. The zero-mean, Gaussian process noise \(w_k\) is assumed to exclusively affect the joint jerk and the bias states, i.e.,

\[
\text{Cov}(w_k) = \text{diag}(0^T I_n 0^T I_n 0^T I_n 0^T I_n 0^T I_n 0^T I_n, \sigma_f^2)
\]

(4)

where \(\sigma_f\) is the respective standard deviation.

Note that other formulations of the system model, which have inputs based on controller setpoints or forward dynamics, might be more accurate due to the incorporation of additional information. However, this would impose non-trivial constraints on the robot architecture, violating the promise of a unified kinematic state estimation.

B. Measurement Model

The measurement vector \(\tilde{y} = \begin{pmatrix} \tilde{q} \\ \tilde{\omega}_{WG_1} \\ \vdots \\ \tilde{\omega}_{WG_{n_g}} \\ \tilde{a}_{A_1} \\ \vdots \\ \tilde{a}_{A_{n_a}} \end{pmatrix} \) stacks the measurements from all available sensors. The measured angular rates \(\tilde{\omega}_{WG_i}\) are assumed to be the sum of the actual angular rates \(\omega_{WG_i}\), the current total bias \(b_{G_i}\), and the corresponding entries of the measurement noise \(v \in \mathbb{R}^{n+3n_g+3n_a}\). The composition of the measured linear accelerations \(\tilde{a}_{Ai}\) follows the same scheme. With the assumption of uncorrelated sensor noise, the covariance matrix of \(v\) is diagonal and its values correspond to the actual noise intensity of the sensors.

In contrast to [10], this approach does not rely on a minimum number of gyros and accelerometers per link nor are the joints considered in a decoupled fashion. As a result, the initially stated requirement for flexibility can be met. Furthermore, inertial sensors can be disabled at runtime, e.g., when a sensor is faulty or reaches its range limits.

C. Measurement Jacobian

The measurement model (5) is nonlinear. To apply the EKF in this case, the Jacobian of the measurement function \(\tilde{y}\) with respect to the state variables from (1) is required. A simple solution for this involves numerical derivatives, but as \(\omega_{WG_i}\) is assumed to exclusively affect the joint jerk and the bias states, i.e.,

\[
\begin{align*}
  \frac{\partial \omega_{WG_i}}{\partial \dot{q}} &= 0 \\
  \frac{\partial \omega_{WG_i}}{\partial \dot{\tilde{g}}_k} &= \frac{\partial \omega_{WG_i}}{\partial \dot{\tilde{g}}_k} + b_{G_i} \\
  \frac{\partial \omega_{WG_i}}{\partial \dot{\tilde{g}}_k} &= \frac{\partial \omega_{WG_i}}{\partial \dot{\tilde{g}}_k} + b_{G_i} \\
  \frac{\partial \omega_{WG_i}}{\partial \tilde{b}_k} &= 0
\end{align*}
\]

(7)

by applying

\[
\begin{align*}
  \frac{\partial \omega_{WG_i}}{\partial \dot{q}} &= 0 \\
  \frac{\partial \omega_{WG_i}}{\partial \dot{\tilde{g}}_k} &= \frac{\partial \omega_{WG_i}}{\partial \dot{\tilde{g}}_k} + b_{G_i} \\
  \frac{\partial \omega_{WG_i}}{\partial \dot{\tilde{g}}_k} &= \frac{\partial \omega_{WG_i}}{\partial \dot{\tilde{g}}_k} + b_{G_i} \\
  \frac{\partial \omega_{WG_i}}{\partial \tilde{b}_k} &= 0
\end{align*}
\]

(8)

(9)

(10)

(11)

is calculated recursively. The angular velocity of the gyroscope \(G_i\) is determined from the angular velocity of the

\[
\begin{align*}
  \omega_{WG_i} &= \omega_{WG_i} + b_{G_i} \\
  \frac{\partial \omega_{WG_i}}{\partial \dot{q}} &= 0 \\
  \frac{\partial \omega_{WG_i}}{\partial \dot{\tilde{g}}_k} &= \frac{\partial \omega_{WG_i}}{\partial \dot{\tilde{g}}_k} + b_{G_i} \\
  \frac{\partial \omega_{WG_i}}{\partial \dot{\tilde{g}}_k} &= \frac{\partial \omega_{WG_i}}{\partial \dot{\tilde{g}}_k} + b_{G_i} \\
  \frac{\partial \omega_{WG_i}}{\partial \tilde{b}_k} &= 0
\end{align*}
\]

(7)

Consequently, the partial derivative of \(\omega_{WG_i}\) with respect to \(q[i]\) is defined recursively as

\[
\frac{\partial \omega_{WG_i}}{\partial q[i]} = \begin{cases} 
  C_{\omega_{WG_i}q[i]}^{\beta_{\omega_{WG_i}}} & i < j \\
  -R_{\omega_{WG_i}q[i]}^{\beta_{\omega_{WG_i}}} & i = j \\
  0 & i > j
\end{cases}
\]

(10)

With the same argumentation, the partial derivative with respect to \(\dot{q}[i]\) is

\[
\frac{\partial \omega_{WG_i}}{\partial \dot{q}[i]} = \begin{cases} 
  C_{\omega_{WG_i}\dot{q}[i]}^{\beta_{\omega_{WG_i}}} & i < j \\
  -R_{\omega_{WG_i}\dot{q}[i]}^{\beta_{\omega_{WG_i}}} & i = j \\
  0 & i > j
\end{cases}
\]

(11)
parent link begin frame \( k = p(G_i) \) via

\[
\omega_{WG_i}^{G_i} = C_{\beta_k}^{G_i} \omega_{Wj}^{\beta_k}.
\] (12)

Finally, the desired partial derivatives of the gyroscope measurement with respect to the state variables are

\[
\frac{\partial \omega_{WG_i}^{G_i}}{\partial q} = C_{\beta_k}^{G_i} \frac{\partial \omega_{Wj}^{\beta_k}}{\partial q} \quad \text{and} \quad \frac{\partial \omega_{WG_i}^{G_i}}{\partial q} = C_{\beta_k}^{G_i} \frac{\partial \omega_{Wj}^{\beta_k}}{\partial q}.
\] (14)

This scheme can be applied to linear accelerations as well. Unfortunately, the resulting formulas are lengthy and we refer the reader to the supplemental material\(^2\) for a detailed derivation. For this reason, we just outline the key formulas at this point. The recursive formula for the position of the link begin frame \( \beta_j \) in the world frame is

\[
x_W^{Wj} = x_W^{Wj-1} + C_{\beta_j}^{Wj-1} (x_{\beta_j-1} - x_{\beta_j-1} + \omega_{\beta_j-1} t_{\beta_j-1}, \omega_{\beta_j-1}, q[j]) .
\] (15)

If this is differentiated twice with respect to time and then resolved in the link begin frame \( \beta_j \),

\[
\alpha_{W\beta_j}^{\beta_j} = C_{\beta_j}^{\beta_j} \left[ 2 \Omega_{W\beta_j}^{\beta_j} \alpha_{\beta_j-1} + \left( \Omega_{W\beta_j}^{\beta_j} \times \right) + \Omega_{W\beta_j}^{\beta_j} \right] (q[j])
\]

\[
= C_{\beta_j}^{\beta_j} \left[ \left( x_{\beta_j-1} + \omega_{\beta_j-1} t_{\beta_j-1}, \omega_{\beta_j-1}, q[j] \right) + \left[ \Omega_{W\beta_j}^{\beta_j} \times \right] \left( q[j], \omega_{\beta_j-1} \right) \left( q[j], \omega_{\beta_j-1} \right) \right] (q[j])
\]

\[
\left( x_{\beta_j-1} + \omega_{\beta_j-1} t_{\beta_j-1}, \omega_{\beta_j-1}, q[j] \right) + \left[ \Omega_{W\beta_j}^{\beta_j} \times \right] \left( q[j], \omega_{\beta_j-1} \right) \left( q[j], \omega_{\beta_j-1} \right)
\]

is obtained. This expression can be differentiated recursively as demonstrated before. The occurring angular accelerations and their derivatives are calculated with the help of

\[
\alpha_{W\beta_j}^{\beta_j} = C_{\beta_j}^{\beta_j} \left[ 2 \Omega_{W\beta_j}^{\beta_j} \alpha_{\beta_j-1} + \left( \Omega_{W\beta_j}^{\beta_j} \times \right) + \Omega_{W\beta_j}^{\beta_j} \right] (q[j])
\]

\[
= C_{\beta_j}^{\beta_j} \left[ \left( x_{\beta_j-1} + \omega_{\beta_j-1} t_{\beta_j-1}, \omega_{\beta_j-1}, q[j] \right) + \left[ \Omega_{W\beta_j}^{\beta_j} \times \right] \left( q[j], \omega_{\beta_j-1} \right) \left( q[j], \omega_{\beta_j-1} \right) \right] (q[j])
\]

which in turn is obtained by differentiating (6) with respect to time and changing the resolving frame to \( \beta_j \). Note, that the effect of gravity is already fully covered if \( \alpha_{W\beta_0} \) is set appropriately. In the last step, the Jacobian of the accelerometer \( A_i \) mounted on link \( k = p(A_i) \) is calculated using the partial derivatives of

\[
\omega_{W\beta_k}^{A_i} = C_{\beta_k}^{A_i} \left( \omega_{W\beta_k} + \Omega_{W\beta_k} \Omega_{W\beta_k} \right) \omega_{\beta_k} A_i,
\] (18)

which is a modified version of (16).

\(^2\)https://iasr.iar.kit.edu/media/pdf/Fennel_MIT22_SupplementalMaterial.pdf

**D. Implementation**

The resulting EKF and the presented algorithm for the efficient calculation of the measurement Jacobian were implemented in C++. To be robot-agnostic, the robot model can be configured at runtime by providing a URDF-file and a list of available sensors. The necessary forward kinematic calculations as well as the parsing of the URDF-file are carried out by the pinocchio-library [15].

**V. EVALUATION IN SIMULATION**

This section covers various simulation experiments to characterize the behavior of the proposed estimator.

**A. Simulation Environment**

For the following simulations, performed with a sampling frequency of 1 kHz, the manipulator as depicted in Fig. 2 with 2 prismatic joints, 6 revolute joints, and 8 IMUs was used. The corresponding Denavit–Hartenberg (DH) parameters are listed in Table I. For the test movements, all joints follow a pre-defined sinusoidal trajectory with a given frequency. To ensure a steady state at the beginning and the end as in real scenarios, the sinusoidal signal is modulated with a cosine window of length 10 s. The position amplitudes are chosen so that a maximum acceleration of 20 m s\(^{-2}\) or rad s\(^{-2}\) occurs, as long as the necessary position amplitude of 30° or 0.52 m is not exceeded. To cover different circumstances, 30 initial configurations with \( |q[0]| < 0.52 \) were drawn once from an uniform distribution. Furthermore, each initial configuration is associated with a set of random phase offsets for the sinusoidal signals.

For the simulation of the encoder data, the groundtruth position signals are distorted with zero-mean, white Gaussian noise with standard deviation \( 4.0 \times 10^{-4} \) and discretized with resolution \( 1.2 \times 10^{-5} \) (19-bit encoders). The simulation model of the gyroscope is depicted in Fig. 3. It comprises the scale-factor error and cross-axis sensitivity matrix \( S_{G_i} \), the additive Gaussian noise \( \nu_{G_i} \), a constant bias \( b_{G_i, const} \), a temperature sensitive bias with coefficient \( \nu_{G_i} \), and a quantization. If not stated differently, the standard deviation of the noise is 0.32 \( \pm 1 \). All other parameters are drawn once from uniform random distributions, whose limits are derived from the typical values in the datasheet of an InvenSense MPU-9250 [16]. Additionally, a fixed pose calibration error is introduced for each gyroscope, for which translation errors are drawn uniformly from \([-2,2] \) mm and Euler orientation errors are drawn uniformly from \([-2,2] \) ° before the first simulation. The simulation model of the accelerometer follows the same scheme, whereas the noise standard deviation is 9.5 \( \times 10^{-3} \) m s\(^{-2}\). For the temperature \( T \), a deterministic sinusoidal trajectory with a frequency of 0.1 Hz and amplitude of 5 °C is assumed.

**B. Estimator Performance**

The proposed estimator was tuned for the presented simulation scenario. For this, the measurement noise covariance was set according to the sensor specifications. The process noise covariance was adjusted manually to
very close to the rotation axis of the last revolute joint.

For very high frequencies, the estimation errors become worse, especially regarding the last joint. The reason for this is the reduced observability of the last joint’s acceleration because the responsible acceleration sensor is mounted very close (50 mm) to the rotation axis and no further acceleration sensors are observing the last joint. For the same reason, the combined acceleration RMSE for all joints looks unsatisfying at the first glance for high frequencies. Nevertheless, this quantity facilitates a fast comparison of different estimators and is therefore used hereinafter.

In the simulations, a full iteration of the estimator took 560μs on average on a standard laptop with an Intel Core i7-9750H CPU. From this, we can conclude that our estimator is real-time capable for manipulators with ≤ 8 joints and control frequencies ≤ 1 kHz. Furthermore, the evaluation of the Jacobian required 6μs on average, proving that our proposed recursive algorithm from Section IV-C is highly efficient.

C. Influence of Environmental Variations

To test the influence of different environmental effects and increasing errors, the RMSEs over frequency were determined in the following setups:

1) Normal noise, but ideal sensor calibration and no bias.
2) Like 1, but with sensor bias terms.
3) Like 2, but with a scale factor error.
4) Reference setup from Section V-A with all errors.
5) Like 4, but with 5-fold increased inertial sensor noise.
6) Like 4, but only with sensors at every other link.

\[
\text{Cov}(w_i) = \text{diag}(0_6^T, 0_6^T, 0_6^T, 12.5^2 1_8^T, 0.0012^2 1_{24}^T, 0.012^2 1_{24}^T).
\]

Here, the large jerk noise was chosen to facilitate rapid acceleration changes. As stated in Section IV-A, the bias states have to compensate for all inertial sensor errors. For very high frequencies, the estimation errors become worse, especially regarding the last joint. The reason for this is the reduced observability of the last joint’s acceleration because the responsible acceleration sensor is mounted very close (50 mm) to the rotation axis and no further acceleration sensors are observing the last joint. For the same reason, the combined acceleration RMSE for all joints looks unsatisfying at the first glance for high frequencies. Nevertheless, this quantity facilitates a fast comparison of different estimators and is therefore used hereinafter.

In the simulations, a full iteration of the estimator took 560μs on average on a standard laptop with an Intel Core i7-9750H CPU. From this, we can conclude that our estimator is real-time capable for manipulators with ≤ 8 joints and control frequencies ≤ 1 kHz. Furthermore, the evaluation of the Jacobian required 6μs on average, proving that our proposed recursive algorithm from Section IV-C is highly efficient.

C. Influence of Environmental Variations

To test the influence of different environmental effects and increasing errors, the RMSEs over frequency were determined in the following setups:

1) Normal noise, but ideal sensor calibration and no bias.
2) Like 1, but with sensor bias terms.
3) Like 2, but with a scale factor error.
4) Reference setup from Section V-A with all errors.
5) Like 4, but with 5-fold increased inertial sensor noise.
6) Like 4, but only with sensors at every other link.
we want to emphasize that the estimation also works without
the given system model. The curves for setup 2 are
Accordingly, the curves of setup 1 in Fig. 5 represent a perfect
number of sensors cannot be reduced arbitrarily without
errors are added as in setup 3, the estimation error of velocity
and acceleration increases approximately by a factor of two
in the range between 1 Hz and 10 Hz. A further doubling of
these errors occurs when rotation and translation errors as in
setup 4 are included in the simulation, yielding the previously
discussed data from Fig. 4. Consequently, we deduce that the
calibration of the inertial sensors impacts the overall system
performance. Especially, the knowledge of scale factor errors,
cross-axis sensitivity, and alignment errors has the potential
to improve the estimation quality significantly. Nevertheless,
we want to emphasize that the estimation also works without
calibration as claimed in the introduction.

In real scenarios, the sensor noise might be increased. The
resulting curves from setup 5 in Fig. 5 suggest that the
proposed estimator is robust against such changes, as long
as a proportional increase of velocity and acceleration errors
for frequencies < 1 Hz is tolerated. Higher frequencies are
less affected by the increased sensor noise.

With the last setup, the claimed modularity regarding the
sensor configuration was tested. As one can see from setups
4 and 6 in Fig. 4, the missing sensors do not significantly
diminish the estimation quality. In practice, however, the
number of sensors cannot be reduced arbitrarily without
performance losses because the full observability based on
acceleration sensors will get lost at some point.

D. Comparison with other Estimators

In this section, a comparison of the proposed full Kalman
Filter (KF-F) with four kinematic state estimators from the
literature is carried out in simulation.

Fig. 5: RMSEs of the proposed estimator under different conditions. The
dashed lines represent the RMS values of the excitation signals.

Accordingly, the curves of setup 1 in Fig. 5 represent a perfect
sensor model and thus a lower bound for the estimation error
with the given system model. The curves for setup 2 are
coincident with setup 1, indicating that constant and slowly
varying sensor biases are handled as expected. If scale factor
errors are added as in setup 3, the estimation error of velocity
and acceleration increases approximately by a factor of two
in the range between 1 Hz and 10 Hz. A further doubling of
these errors occurs when rotation and translation errors as in
setup 4 are included in the simulation, yielding the previously
discussed data from Fig. 4. Consequently, we deduce that the
calibration of the inertial sensors impacts the overall system
performance. Especially, the knowledge of scale factor errors,
cross-axis sensitivity, and alignment errors has the potential
to improve the estimation quality significantly. Nevertheless,
we want to emphasize that the estimation also works without
calibration as claimed in the introduction.

In real scenarios, the sensor noise might be increased. The
resulting curves from setup 5 in Fig. 5 suggest that the
proposed estimator is robust against such changes, as long
as a proportional increase of velocity and acceleration errors
for frequencies < 1 Hz is tolerated. Higher frequencies are
less affected by the increased sensor noise.

With the last setup, the claimed modularity regarding the
sensor configuration was tested. As one can see from setups
4 and 6 in Fig. 4, the missing sensors do not significantly
diminish the estimation quality. In practice, however, the
number of sensors cannot be reduced arbitrarily without
performance losses because the full observability based on
acceleration sensors will get lost at some point.

D. Comparison with other Estimators

In this section, a comparison of the proposed full Kalman
Filter (KF-F) with four kinematic state estimators from the
literature is carried out in simulation.

Fig. 5: RMSEs of the proposed estimator under different conditions. The
dashed lines represent the RMS values of the excitation signals.

Accordingly, the curves of setup 1 in Fig. 5 represent a perfect
sensor model and thus a lower bound for the estimation error
with the given system model. The curves for setup 2 are
coincident with setup 1, indicating that constant and slowly
varying sensor biases are handled as expected. If scale factor
errors are added as in setup 3, the estimation error of velocity
and acceleration increases approximately by a factor of two
in the range between 1 Hz and 10 Hz. A further doubling of
these errors occurs when rotation and translation errors as in
setup 4 are included in the simulation, yielding the previously
discussed data from Fig. 4. Consequently, we deduce that the
calibration of the inertial sensors impacts the overall system
performance. Especially, the knowledge of scale factor errors,
cross-axis sensitivity, and alignment errors has the potential
to improve the estimation quality significantly. Nevertheless,
we want to emphasize that the estimation also works without
calibration as claimed in the introduction.

In real scenarios, the sensor noise might be increased. The
resulting curves from setup 5 in Fig. 5 suggest that the
proposed estimator is robust against such changes, as long
as a proportional increase of velocity and acceleration errors
for frequencies < 1 Hz is tolerated. Higher frequencies are
less affected by the increased sensor noise.

With the last setup, the claimed modularity regarding the
sensor configuration was tested. As one can see from setups
4 and 6 in Fig. 4, the missing sensors do not significantly
diminish the estimation quality. In practice, however, the
number of sensors cannot be reduced arbitrarily without
performance losses because the full observability based on
acceleration sensors will get lost at some point.

D. Comparison with other Estimators

In this section, a comparison of the proposed full Kalman
Filter (KF-F) with four kinematic state estimators from the
literature is carried out in simulation.

Fig. 5: RMSEs of the proposed estimator under different conditions. The
dashed lines represent the RMS values of the excitation signals.

Accordingly, the curves of setup 1 in Fig. 5 represent a perfect
sensor model and thus a lower bound for the estimation error
with the given system model. The curves for setup 2 are
coincident with setup 1, indicating that constant and slowly
varying sensor biases are handled as expected. If scale factor
errors are added as in setup 3, the estimation error of velocity
and acceleration increases approximately by a factor of two
in the range between 1 Hz and 10 Hz. A further doubling of
these errors occurs when rotation and translation errors as in
setup 4 are included in the simulation, yielding the previously
discussed data from Fig. 4. Consequently, we deduce that the
calibration of the inertial sensors impacts the overall system
performance. Especially, the knowledge of scale factor errors,
cross-axis sensitivity, and alignment errors has the potential
to improve the estimation quality significantly. Nevertheless,
we want to emphasize that the estimation also works without
calibration as claimed in the introduction.

In real scenarios, the sensor noise might be increased. The
resulting curves from setup 5 in Fig. 5 suggest that the
proposed estimator is robust against such changes, as long
as a proportional increase of velocity and acceleration errors
for frequencies < 1 Hz is tolerated. Higher frequencies are
less affected by the increased sensor noise.

With the last setup, the claimed modularity regarding the
sensor configuration was tested. As one can see from setups
4 and 6 in Fig. 4, the missing sensors do not significantly
diminish the estimation quality. In practice, however, the
number of sensors cannot be reduced arbitrarily without
performance losses because the full observability based on
acceleration sensors will get lost at some point.

D. Comparison with other Estimators

In this section, a comparison of the proposed full Kalman
Filter (KF-F) with four kinematic state estimators from the
literature is carried out in simulation.

Fig. 5: RMSEs of the proposed estimator under different conditions. The
dashed lines represent the RMS values of the excitation signals.

Accordingly, the curves of setup 1 in Fig. 5 represent a perfect
sensor model and thus a lower bound for the estimation error
with the given system model. The curves for setup 2 are
coincident with setup 1, indicating that constant and slowly
varying sensor biases are handled as expected. If scale factor
errors are added as in setup 3, the estimation error of velocity
and acceleration increases approximately by a factor of two
in the range between 1 Hz and 10 Hz. A further doubling of
these errors occurs when rotation and translation errors as in
setup 4 are included in the simulation, yielding the previously
discussed data from Fig. 4. Consequently, we deduce that the
calibration of the inertial sensors impacts the overall system
performance. Especially, the knowledge of scale factor errors,
cross-axis sensitivity, and alignment errors has the potential
to improve the estimation quality significantly. Nevertheless,
we want to emphasize that the estimation also works without
calibration as claimed in the introduction.

In real scenarios, the sensor noise might be increased. The
resulting curves from setup 5 in Fig. 5 suggest that the
proposed estimator is robust against such changes, as long
as a proportional increase of velocity and acceleration errors
for frequencies < 1 Hz is tolerated. Higher frequencies are
less affected by the increased sensor noise.

With the last setup, the claimed modularity regarding the
sensor configuration was tested. As one can see from setups
4 and 6 in Fig. 4, the missing sensors do not significantly
diminish the estimation quality. In practice, however, the
number of sensors cannot be reduced arbitrarily without
performance losses because the full observability based on
acceleration sensors will get lost at some point.

D. Comparison with other Estimators

In this section, a comparison of the proposed full Kalman
Filter (KF-F) with four kinematic state estimators from the
literature is carried out in simulation.

Fig. 5: RMSEs of the proposed estimator under different conditions. The
dashed lines represent the RMS values of the excitation signals.

Accordingly, the curves of setup 1 in Fig. 5 represent a perfect
sensor model and thus a lower bound for the estimation error
with the given system model. The curves for setup 2 are
coincident with setup 1, indicating that constant and slowly
varying sensor biases are handled as expected. If scale factor
errors are added as in setup 3, the estimation error of velocity
and acceleration increases approximately by a factor of two
in the range between 1 Hz and 10 Hz. A further doubling of
these errors occurs when rotation and translation errors as in
setup 4 are included in the simulation, yielding the previously
discussed data from Fig. 4. Consequently, we deduce that the
calibration of the inertial sensors impacts the overall system
performance. Especially, the knowledge of scale factor errors,
cross-axis sensitivity, and alignment errors has the potential
to improve the estimation quality significantly. Nevertheless,
we want to emphasize that the estimation also works without
calibration as claimed in the introduction.

In real scenarios, the sensor noise might be increased. The
resulting curves from setup 5 in Fig. 5 suggest that the
proposed estimator is robust against such changes, as long
as a proportional increase of velocity and acceleration errors
for frequencies < 1 Hz is tolerated. Higher frequencies are
less affected by the increased sensor noise.

With the last setup, the claimed modularity regarding the
sensor configuration was tested. As one can see from setups
4 and 6 in Fig. 4, the missing sensors do not significantly
diminish the estimation quality. In practice, however, the
number of sensors cannot be reduced arbitrarily without
performance losses because the full observability based on
acceleration sensors will get lost at some point.

D. Comparison with other Estimators

In this section, a comparison of the proposed full Kalman
Filter (KF-F) with four kinematic state estimators from the
literature is carried out in simulation.

Fig. 5: RMSEs of the proposed estimator under different conditions. The
dashed lines represent the RMS values of the excitation signals.
The filter orders have similar effects as in Fig. 6. The parameter $f_c$ was omitted as it had very little effect.

Encoder noise was identified to be less serious than assumed abruptly by triggering the mechanical brakes of the joints. The reversal points. After 11 cycles in 25 s back-and-forth movement in Cartesian space with a maximum 1 kHz of end effector. All calculations are carried out with a frequency every joint and an uncalibrated InvenSense MPU-9250 at the manipulator is equipped with a 19-bit load-side encoder in medium frequency range, which is mostly owed to the bad observability of the last joint as the joint-wise RMS errors $\text{RMSE} = (0.14, 0.19, 0.42, 0.65, 0.91, 0.92, 0.76, 6.41)$ and $\text{RMSE} = (0.10, 0.17, 0.21, 0.30, 0.26, 0.49, 0.44, 0.48)$ for 2 Hz suggest. A potential drawback of our method is the increased computational effort, as shown in Table II.

![Fig. 7: RMSEs of PI for acceleration estimation with different parameters.](image)

![Fig. 8: Estimation error of the compared kinematic state estimators.](image)

![Fig. 9: A planar manipulator with 2 joints in SCARA configuration was used for the evaluation on real hardware.](image)

Towards inaccurate calibration data. The acceleration estimates of PI are much less precise than those of KF-F in the medium frequency range, which is mostly owed to the bad observability of the last joint as the joint-wise RMS errors $\text{RMSE} = (0.14, 0.19, 0.42, 0.65, 0.91, 0.92, 0.76, 6.41)$ and $\text{RMSE} = (0.10, 0.17, 0.21, 0.30, 0.26, 0.49, 0.44, 0.48)$ for 2 Hz suggest. A potential drawback of our method is the increased computational effort, as shown in Table II.

### VI. Evaluation on Real Hardware

The practical applicability of the proposed estimator was demonstrated on real hardware.

#### A. Setup

The test setup is depicted in Fig. 9. A two joint SCARA manipulator is equipped with a 19-bit load-side encoder in every joint and an uncalibrated InvenSense MPU-9250 at the end effector. All calculations are carried out with a frequency of 1 kHz. The manipulator is programmed to perform a linear back-and-forth movement in Cartesian space with a maximum velocity of $0.5 \text{ m s}^{-1}$ and a distance of 0.5 m between the reversal points. After 11 cycles in 25 s, the motion is stopped abruptly by triggering the mechanical brakes of the joints. The encoder noise was identified to be less serious than assumed in the simulations, while the IMU errors were increased significantly. Hence, the filter has been retuned with the parameters $\text{Cov}(\bar{w}_k) = \text{diag}(0^T, 12.5^2, 0.005^2, 0.05^2, 1^T)$ and $\text{Cov}(\dot{v}) = \text{diag}(0.05^2, 1^T, 0.188^2, 1^T)$. The practical applicability of the proposed estimator was demonstrated on real hardware.

#### B. Results

Fig. 10 shows the velocity and acceleration estimates of our approach (KF-F) for both joints. The oscillations are not noise, but true mechanical oscillations originating from finite stiffness as reported in [12]. Before the estimation quality can be assessed, the reference signals must be discussed due to the lack of real groundtruth information in our setup.

It is obvious that numerical differentiation combined with an acausal smoothing step during post-processing can yield noise-free and undelayed estimates. However, the resulting signals are only reliable up to a specific frequency as the amplification of the encoder noise increases with frequency. Furthermore, the encoders have additional errors beyond additive noise and the manipulator’s structural elements are not infinitely stiff, resulting in high-frequent movements not captured by the encoders. For this reason, a fair comparison without dedicated acceleration and velocity groundtruth sensors is limited to a comparatively low frequency range.

In our case, an analysis of the raw IMU data and the differentiated, but unfiltered encoder data suggests that a frequency limitation to 10 Hz is reasonable. This was implemented using a 4th-order Butterworth filter in a forward-backward setup to achieve delay-free results. The resulting low-pass filtered numerical derivatives (reference) and the low-pass filtered estimates of our approach (KF-F, band-limited) are included in Fig. 10. Since both are very close, the deviations, which can be interpreted as an estimation error, are plotted separately in Fig. 11. For the whole trajectory with an RMS of 1.253 for velocity and 5.727 for acceleration, the corresponding RMSEs are 0.005 and 0.430, respectively.

![Fig. 10: Estimation error of the compared kinematic state estimators.](image)

### Table II: Processing times of the compared kinematic state estimators.

<table>
<thead>
<tr>
<th>Estimator</th>
<th>Mean runtime in µs</th>
</tr>
</thead>
<tbody>
<tr>
<td>KF-F</td>
<td>560</td>
</tr>
<tr>
<td>ND</td>
<td>&lt; 1</td>
</tr>
<tr>
<td>PI</td>
<td>6</td>
</tr>
<tr>
<td>KF-D</td>
<td>113</td>
</tr>
<tr>
<td>KF-T</td>
<td>24</td>
</tr>
</tbody>
</table>
We outperformed competing approaches significantly in all real-life scenarios, even in situations where very abrupt aspects except for runtime. The model has a comparatively large state dimension, real-time constant jerk model for the movement with constant bias joints was presented. To achieve this, an EKF combines a recursive expression for the Jacobian of the prediction and the update step of the EKF. Thereafter, the application of the proposed estimator on more complex real robots is planned. This also raises the question of which sensor setups are optimal regarding the number of sensors and the observability. Furthermore, the generalization to setups where the joint position cannot be measured directly with sufficient precision, i.e., human motion tracking, is considered for future research.

Fig. 10: Excerpt of the velocity and acceleration estimates of the proposed estimator over time in comparison with the band-limited numerical differentiation. The brake is actuated at $t \approx 25\,\text{s}$.

Fig. 11: Deviations between the proposed estimator and the reference from numerical differentiation over time, when a band-limit of 10 Hz is applied. The estimation errors are increased significantly when the brakes are triggered. Nevertheless, the error magnitude is kept at a fraction of the actual signal magnitude.

From this, it is evident that the proposed estimator also works well in real-life scenarios, even in situations where very abrupt acceleration and velocity changes occur.

VII. CONCLUSIONS

In this paper, a new calibration-free kinematic state estimator for robot manipulators with revolute and prismatic joints was presented. To achieve this, an EKF combines a constant jerk model for the movement with constant bias models for the inertial sensors. Although the resulting system model has a comparatively large state dimension, real-time capable and robot-agnostic execution is made possible by means of an efficient recursive expression for the Jacobian of the measurement function. As demonstrated in simulations and experiments with real hardware, the proposed estimator is able to incorporate nearly all information from a given sensor setup, while still being robust towards parameter changes. We outperformed competing approaches significantly in all aspects except for runtime.

Future work will involve runtime optimizations exploiting the sparsity of the system and measurement Jacobians during the prediction and the update step of the EKF. The observability. Furthermore, the generalization to setups where the joint position cannot be measured directly with sufficient precision, i.e., human motion tracking, is considered for future research.

REFERENCES