A Stable Adaptive Extended Kalman Filter for Estimating Robot Manipulators Link Velocity and Acceleration

Seyed Ali Baradaran Birjandi[1,*], Harshit Khurana[2], Aude Billard[2] and Sami Haddadin[1]

***Abstract*— One can estimate the velocity and acceleration of robot manipulators by utilizing nonlinear observers. This involves combining *inertial measurement units (IMUs)* with the motor encoders of the robot through a model-based sensor fusion technique. This approach is lightweight, versatile (suitable for a wide range of trajectories and applications), and straightforward to implement. In order to further improve the estimation accuracy while running the system, we propose to adapt the noise information in this paper. This would automatically reduce the system vulnerability to imperfect modelings and sensor changes. Moreover, viable strategies to maintain the system stability are introduced. Finally, we thoroughly evaluate the overall framework with a seven DoF robot manipulator whose links are equipped with IMUs.**

I. Introduction and State of the Art

Kalman filter has been widely used for different purposes. Indeed, the early success of Kalman filters in aerospace programs in the 1960s gave rise to industrial applications. However, the filter works well as long as both the model and noise statistics, which are provided to it, remain accurate. In fact, degradation in estimation precision or even divergence is noticeable once either the model or noise information is inaccurate [1]. This flaw became apparent soon after the Kalman filter emergence (look at, e.g., [2]). Governments spent millions of dollars on space programs in the 1960s, where the systems and the noise statistics were modeled with high fidelity. However, the industry has a limited budget and, thus, may face hardships when using Kalman filters in the intended applications [3].

Accurate modeling of dynamic systems is beyond the scope of Kalman filters and requires separate treatment. Moreover, the noise statistics can be *optimally estimated* offline in some special cases only [4]. Therefore, different approaches are proposed in the literature to overcome this difficulty by online *adapting* the noise information, which is given to the filter in terms of noise covariance. One of the first attempts in this direction is made by Mehra [5]. He suggests two iterative methods, namely maximum likelihood estimation and covariance matching, to adapt the covariance of the a priori process and measurement noise while running the estimator. The former estimates the noise statistics along the system states by parameterizing the noise covariance. While the latter strives to match the theoretical and observed innovations covariance, which corresponds to optimal filter performance [6]. Maximum likelihood method is generally utilized to adapt the noise covariance matrix in applications such as global positioning systems [6]–[8], power systems [9], object tracking [10] and robotics [11], [12].

In the sequel, we account for a noise covariance adaptation algorithm, which is based on maximum likelihood in conjunction with *extended Kalman filter (EKF)*. Thus far, we have used EKF to estimate robot manipulators link velocity

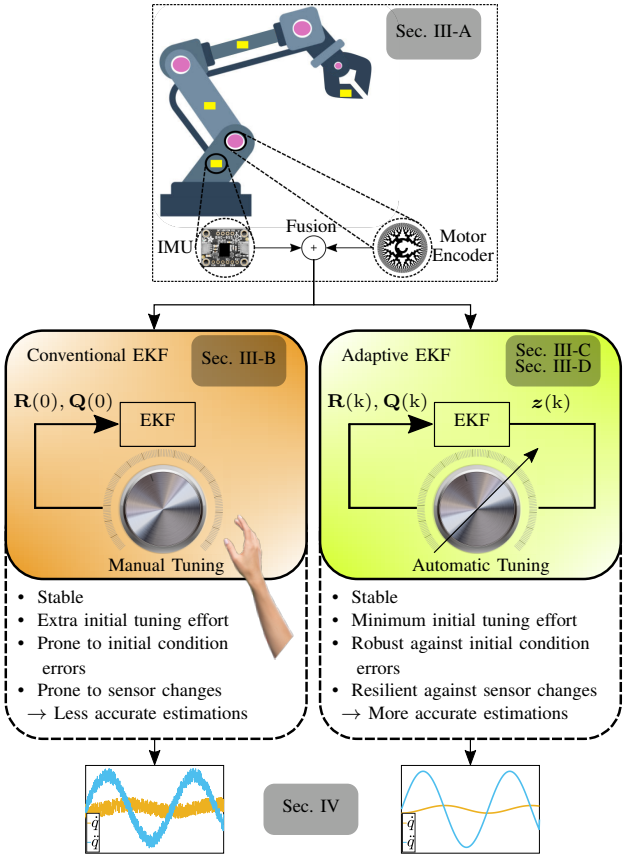

Fig. 1. Summary of the topics covered in the paper.

[1] Munich Institute of Robotics and Machine Intelligence, Technical University of Munich, Germany.

[2] LASA laboratory, Ecole Polytechnique Federale de Lausanne, Lausanne, Switzerland.

[*] Corresponding author. E-mail address: ali.baradaran@tum.de

We greatly acknowledge the funding of this work by of the Lighthouse Initiative Geriatronics by StMWi Bayern (Project X, grant no. IUK-1807-0007// IUK582/001) and LongLeif GaPa gGmbH (Project Y). This work was also supported by the European Union's Horizon 2020 research and innovation programme as part of the project I.AM. under grant no. 871899 and the project Darko under grant no. 101017274. We also acknowledge the financial support by the Bavarian State Ministry for Economic Affairs, Regional Development and Energy (StMWi) for the Lighthouse Initiative KI.FABRIK, (Phase 1: Infrastructure as well as the research and development program under, grant no. DIK0249). The authors also wish to thank Mr. Erfan Shahriari for his generous support. Please also note that S. Haddadin has a potential conflict of interest as shareholder of Franka Emika GmbH.

and acceleration (initially introduced in [13] and thoroughly evaluated in [14]). In fact, this approach has proved to be modular and computationally light. Furthermore, thanks to the fusion of various types of sensors, the technique benefits from high bandwidth. Therefore, the estimator is suitable for realistic robotic challenges such as collision detection [15], [16] and control [17]. However, tuning EKF demands large measurement samples and some extra effort. Moreover, the sensors characteristics may alter with time and environmental factors such as temperature or humidity, which can lead to imprecise estimations. Thus, to decrease the initial tuning effort as well as the estimation sensitivity to the sensor changes, we propose a stable method for adapting the noise covariance in this paper.

This paper is organized as follows. In Sec. II, the problem at hand is detailed. The estimator of manipulator link variables is reviewed in Sec. III-A, followed by outlining the preliminaries of general EKF in Sec. III-B. Subsequently, we explain a noise covariance adaptation law in Sec. III-C and propose some approaches for maintaining its stability in Sec. III-D. In Sec. IV, the setup framework is initially introduced, followed by experiential evaluation of the link variable estimator. Finally, the paper concludes in Sec. V. Figure 1 summarizes the paper topics.

II. Problem Statement and Contribution

Let us assume the reduced rigid body dynamics model of a serial-chain robot with n joints as [18]

$$\boldsymbol{M}(\boldsymbol{q})\ddot{\boldsymbol{q}} + \boldsymbol{C}(\boldsymbol{q},\dot{\boldsymbol{q}})\dot{\boldsymbol{q}} + \boldsymbol{g}(\boldsymbol{q}) = \boldsymbol{\tau}_{\mathrm{m}} - \boldsymbol{\tau}_{\mathrm{f}}, \tag{1}$$

where $\boldsymbol{q}, \dot{\boldsymbol{q}}, \ddot{\boldsymbol{q}} \in \mathbb{R}^n$ denote the link side joint position, velocity, and acceleration, $\boldsymbol{M}(\boldsymbol{q}) \in \mathbb{R}^{n\times n}$ the symmetric and positive definite inertia matrix, $\boldsymbol{C}(\boldsymbol{q},\dot{\boldsymbol{q}})\dot{\boldsymbol{q}} \in \mathbb{R}^n$ the centripetal and Coriolis vector, $\boldsymbol{g}(\boldsymbol{q}) \in \mathbb{R}^n$ the gravity vector, $\boldsymbol{\tau}_{\mathrm{m}} \in \mathbb{R}^n$ the active motor torque and $\boldsymbol{\tau}_{\mathrm{f}} \in \mathbb{R}^n$ the torque caused by external effects (e.g., frictions).

Link velocity and acceleration information $(\dot{\boldsymbol{q}}, \ddot{\boldsymbol{q}})$ can be used to improve robotics algorithms [13]. However, these physical quantities can not be directly measured. Therefore, we have to estimate them by fusing multiple sensors in a model-based observer/estimator. We have, in fact, developed and applied this estimator to various robotic use cases. Thus, the research question we strive to answer is as follows:

While manually tuning the estimator may be challenging, can we improve the estimation accuracy by automatically adjusting it online? This would also intrinsically decrease the estimator sensitivity to system/sensor changes over time. Therefore, the paper's contributions can be summarized in the following (see Fig. 1).

- We initially review the method for estimating manipulator link variables.
- Subsequently, we propose a method for adapting the estimator noise covariance matrices online. Moreover, some strategies are employed to maintain the observer stability.
- Finally, we evaluate the theory by performing experiments on a seven-DoF manipulator. For this, each manipulator link is equipped with one IMU. The results are also compared against a state-of-the-art algorithm based on covariance matching techniques.

III. Link Velocity and Acceleration Estimation

This section briefly reviews the method for robot link velocity and acceleration estimation. Also, we provide details on how to adapt extended Kalman filter to improve the estimation performance.

A. Link Velocity and Acceleration Observer

In order to be able to obtain the link i variables by fusing IMU m with the robot link-side measurements, an estimator is utilized with the following discrete dynamics model [15]

$$\begin{aligned}\hat{\mathbf{x}}_{\mathrm{i}}(\mathrm{k}+1) &= \boldsymbol{f}_{\mathrm{i}}(\hat{\mathbf{x}}_{\mathrm{i}}(\mathrm{k})) \\ &= \begin{pmatrix}1 & \Delta t & \Delta t^2/2 & \Delta t^3/6\\ 0 & 1 & \Delta t & \Delta t^2/2\\ 0 & 0 & 1 & \Delta t\\ 0 & 0 & 0 & 1\end{pmatrix}\begin{pmatrix}\hat{q}_{\mathrm{i}}(\mathrm{k})\\ \hat{\dot{q}}_{\mathrm{i}}(\mathrm{k})\\ \hat{\ddot{q}}_{\mathrm{i}}(\mathrm{k})\\ \hat{\dddot{q}}_{\mathrm{i}}(\mathrm{k})\end{pmatrix} + \boldsymbol{w}(\mathrm{k}),\\ \hat{\mathbf{y}}_{\mathrm{i}}(\mathrm{k}) &= \boldsymbol{h}_{\mathrm{i}}(\hat{\mathbf{x}}_{\mathrm{i}}(\mathrm{k})) = \begin{pmatrix}\hat{q}_{\mathrm{i}}(\mathrm{k})\\ {}^{\mathrm{m}}\boldsymbol{a}(\hat{q}_{\mathrm{i}}(\mathrm{k}),\hat{\dot{q}}_{\mathrm{i}}(\mathrm{k}),\hat{\ddot{q}}_{\mathrm{i}}(\mathrm{k}))\\ {}^{\mathrm{m}}\boldsymbol{\omega}(\hat{q}_{\mathrm{i}}(\mathrm{k}),\hat{\dot{q}}_{\mathrm{i}}(\mathrm{k}))\end{pmatrix} + \boldsymbol{v}(\mathrm{k}),\end{aligned} \tag{2}$$

with $\mathrm{k} \in \mathbb{Z}^+$ being the independent variable, Δt the system sampling time and $\mathbf{x}_{\mathrm{i}}(\mathrm{k}) = [q_{\mathrm{i}}(\mathrm{k})\ \dot{q}_{\mathrm{i}}(\mathrm{k})\ \ddot{q}_{\mathrm{i}}(\mathrm{k})\ \dddot{q}_{\mathrm{i}}(\mathrm{k})]^T \in \mathbb{R}^4$ the system states at epoch k ($\hat{\cdot}$ operator denotes the estimated counterparts). Furthermore, $\dddot{q}_{\mathrm{i}}$ denotes the link i jerk. Moreover, $\boldsymbol{f}_{\mathrm{i}} : \mathbb{R}^4 \to \mathbb{R}^4$ is the system transition function and $\boldsymbol{h}_{\mathrm{i}} : \mathbb{R}^3 \to \mathbb{R}^7$ is the measurement function. Accordingly, $\hat{\mathbf{y}}_{\mathrm{i}} \in \mathbb{R}^7$ denotes the estimated measurements. $\boldsymbol{w}(\mathrm{k}) \in \mathbb{R}^4$ and $\boldsymbol{v}(\mathrm{k}) \in \mathbb{R}^7$ are the white Gaussian process and measurement noise at epoch k, respectively. Also, IMU m is installed on link i and outputs Cartesian acceleration ${}^{\mathrm{m}}\boldsymbol{a}(q_{\mathrm{i}},\dot{q}_{\mathrm{i}},\ddot{q}_{\mathrm{i}}) : \mathbb{R}^3 \to \mathbb{R}^3$ and angular velocity ${}^{\mathrm{m}}\boldsymbol{\omega}(q_{\mathrm{i}},\dot{q}_{\mathrm{i}}) : \mathbb{R}^2 \to \mathbb{R}^3$ in frame $\{\mathrm{m}\}$. According to the algebraic (memory-less) kinematical equations for rigid bodies [19],

$${}^{\mathrm{i}}\boldsymbol{a}_{\mathrm{m}} = {}^{\mathrm{i}}\boldsymbol{a}_{l} + {}^{\mathrm{i}}\dot{\boldsymbol{\omega}}_{\mathrm{i}} \times {}^{\mathrm{i}}\boldsymbol{x}_{\mathrm{S_m}} + {}^{\mathrm{i}}\boldsymbol{\omega}_{\mathrm{i}} \times \left({}^{\mathrm{i}}\boldsymbol{\omega}_{\mathrm{i}} \times {}^{\mathrm{i}}\boldsymbol{x}_{\mathrm{S_m}}\right), \tag{3}$$

$${}^{\mathrm{m}}\boldsymbol{a} = {}^{\mathrm{m}}\boldsymbol{R}_{\mathrm{i}}\,{}^{\mathrm{i}}\boldsymbol{a}_{\mathrm{m}}, \tag{4}$$

$$\begin{pmatrix}0\\0\\\dot{q}_{\mathrm{i}}\end{pmatrix} = {}^{\mathrm{i}}\boldsymbol{\omega}_{\mathrm{i}} - {}^{\mathrm{i}}\boldsymbol{\omega}_{\mathrm{i}-1},\ \begin{pmatrix}0\\0\\\ddot{q}_{\mathrm{i}}\end{pmatrix} = {}^{\mathrm{i}}\dot{\boldsymbol{\omega}}_{\mathrm{i}} - {}^{\mathrm{i}}\dot{\boldsymbol{\omega}}_{\mathrm{i}-1}, \tag{5}$$

$${}^{\mathrm{m}}\boldsymbol{\omega} = {}^{\mathrm{i}}\boldsymbol{\omega}_{\mathrm{i}} \tag{6}$$

hold. Here, ${}^{\mathrm{i}}\boldsymbol{a}_{\mathrm{m}} \in \mathbb{R}^3$ and ${}^{\mathrm{i}}\dot{\boldsymbol{\omega}}_{\mathrm{i}} \in \mathbb{R}^3$ are the Cartesian acceleration and angular velocity at sensor location ${}^{\mathrm{i}}\boldsymbol{x}_{\mathrm{S_m}} \in \mathbb{R}^3$ in frame $\{\mathrm{i}\}$, respectively. Moreover, ${}^{\mathrm{m}}\boldsymbol{R}_{\mathrm{i}} \in \mathbb{R}^{3\times3}$ is the rotation matrix from frame $\{\mathrm{i}\}$ to frame $\{\mathrm{m}\}$. Since the angular velocity of a point in rigid bodies is independent of the location of measurements, (6) holds. According to (5), for an n-link robot manipulator, the states are estimated recursively from the first link $\mathrm{i} = 1$ to the last link $\mathrm{i} = n$.

Note that in the absence of the desired trajectory information, a constant jerk is assumed in (2). Although this assumption is generally not accurate, it will impose less numerical instability compared to constant acceleration or velocity

assumptions.
Also note that the state-transition function $\boldsymbol{f}_{\mathrm{i}}$ in (2) is linear. Furthermore, the algebraic measurement function $\boldsymbol{h}_{\mathrm{i}}$ is essentially a second-order polynomial w.r.t the system state. Therefore, the system (2) nonlinearities are minor, which makes EKF a suitable choice for estimating the system states ($\mathbf{x}_{\mathrm{i}}$). EKF follows the principles of linear Kalman filter by linearizing the nonlinear system around the currently estimated states ($\hat{\mathbf{x}}_{\mathrm{i}}(\mathrm{k})$). The linearization is generally based on Taylor expansion, which implies that EKF works best when the system nonlinearities are not severe. Therefore, we have used EKF in conjunction with system (2) for different applications thus far. We discuss the implementation preliminaries of extended Kalman filter in the next section.

B. Extended Kalman Filter

As briefly mentioned earlier, the underlying notion of EKF is based on the linearization of the nonlinear system around the current state. Let

$$\begin{aligned}\boldsymbol{F}_{\mathrm{i}}(\hat{\mathbf{x}}_{\mathrm{i}}(\mathrm{k})) &= \frac{\partial \boldsymbol{f}_{\mathrm{i}}}{\partial \mathbf{x}_{\mathrm{i}}}|_{\mathbf{x}_{\mathrm{i}}=\hat{\mathbf{x}}_{\mathrm{i}}(\mathrm{k})} \in \mathbb{R}^{4\times 4}\\ \boldsymbol{H}_{\mathrm{i}}(\hat{\mathbf{x}}_{\mathrm{i}}(\mathrm{k})) &= \frac{\partial \boldsymbol{h}_{\mathrm{i}}}{\partial \mathbf{x}_{\mathrm{i}}}|_{\mathbf{x}_{\mathrm{i}}=\hat{\mathbf{x}}_{\mathrm{i}}(\mathrm{k})} \in \mathbb{R}^{7\times 4}\end{aligned} \tag{7}$$

denote the Jacobian matrices of the state transition and measurement functions in (2) at $\hat{\mathbf{x}}_{\mathrm{i}}(\mathrm{k})$, respectively. In fact, $\boldsymbol{F}_{\mathrm{i}}(\cdot)$ is the same transition matrix introduced in (2). Moreover,

$$\begin{aligned}&\hat{\mathbf{x}}_{\mathrm{i}}(0) = \hat{\mathbf{x}}_{\mathrm{i}}(\mathrm{k}=0)\\ &\boldsymbol{P}_{\mathrm{i}}(0) = \mathbb{E}[(\mathbf{x}_{\mathrm{i}}(0)-\hat{\mathbf{x}}_{\mathrm{i}}(0))\,(\mathbf{x}_{\mathrm{i}}(0)-\hat{\mathbf{x}}_{\mathrm{i}}(0))^{T}] \in \mathbb{R}^{4\times 4}\end{aligned} \tag{8}$$

are the system initial state and the initial state estimation error covariance matrix, respectively. Here, $\mathbb{E}[\cdot]$ denotes the expected value. The initial process noise covariance $\mathbf{Q}_{\mathrm{i}}(0) = \mathbb{E}[\boldsymbol{w}\boldsymbol{w}^{T}] \in \mathbb{R}^{4\times 4}$ and measurement noise covariance $\mathbf{R}_{\mathrm{i}}(0) = \mathbb{E}[\boldsymbol{v}\boldsymbol{v}^{T}] \in \mathbb{R}^{7\times 7}$ for the i-th link, which are positive definite matrices, are also available. In practice, the robot can be assumed at rest, initially. Therefore, the initial state is given by

$$\hat{\mathbf{x}}_{\mathrm{i}}(0) = \begin{bmatrix} q_{\mathrm{i}}(0) \\ 0 \\ 0 \\ 0 \end{bmatrix}, \tag{9}$$

where $q_{\mathrm{i}}(0)$ is the measured link position at $\mathrm{k}=0$. Depending on the accuracy of $q_{\mathrm{i}}(0)$, the initial (diagonal) estimation error covariance matrix $\boldsymbol{P}_{\mathrm{i}}(0)$ can be set to small values. Subsequently, EKF undergoes two steps, namely, *prediction* and *correction* at each iteration k.

a) Prediction: The a priori estimates of the states and estimation error covariance are obtained via the a posteriori (corrected) estimates of the previous iteration, as

$$\hat{\mathbf{x}}_{\mathrm{i}}^{-}(\mathrm{k}) = \boldsymbol{f}_{\mathrm{i}}(\hat{\mathbf{x}}_{\mathrm{i}}^{+}(\mathrm{k}-1)) \tag{10}$$

$$\boldsymbol{P}_{\mathrm{i}}^{-}(\mathrm{k}) = \boldsymbol{F}_{\mathrm{i}}(\hat{\mathbf{x}}_{\mathrm{i}}^{-}(\mathrm{k}))\boldsymbol{P}_{\mathrm{i}}^{+}(\mathrm{k}-1){\boldsymbol{F}_{\mathrm{i}}}^{T}(\hat{\mathbf{x}}_{\mathrm{i}}^{-}(\mathrm{k})) + \mathbf{Q}_{\mathrm{i}}(\mathrm{k}). \tag{11}$$

Here $\cdot^{-}$ and $\cdot^{+}$ denote the a priori and a posteriori estimates, respectively.

b) Correction: Once the new measurements are fed to the filter, the a priori estimates can be corrected. For this innovation ($\boldsymbol{z}_{\mathrm{i}}(\mathrm{k}) \in \mathbb{R}^{7}$), which is the difference between the a priori predicted measurements and the real ones, as well as its covariance ($\boldsymbol{S}_{\mathrm{i}}(\mathrm{k}) \in \mathbb{R}^{7\times 7}$) is computed via

$$\boldsymbol{z}_{\mathrm{i}}(\mathrm{k}) = \mathbf{y}_{\mathrm{i}}(\mathrm{k}) - \boldsymbol{h}_{\mathrm{i}}(\hat{\mathbf{x}}_{\mathrm{i}}^{-}(\mathrm{k})) \tag{12}$$

$$\boldsymbol{S}_{\mathrm{i}}(\mathrm{k}) = \boldsymbol{H}_{\mathrm{i}}(\hat{\mathbf{x}}_{\mathrm{i}}^{-}(\mathrm{k}))\boldsymbol{P}_{\mathrm{i}}^{-}(\mathrm{k})\boldsymbol{H}_{\mathrm{i}}^{T}(\hat{\mathbf{x}}_{\mathrm{i}}^{-}(\mathrm{k})) + \mathbf{R}_{\mathrm{i}}(\mathrm{k}), \tag{13}$$

where $\mathbf{y}_{\mathrm{i}}(\mathrm{k})$ is the sensors output. Note that since we are dealing with a nonlinear measurement function, $\boldsymbol{S}_{\mathrm{i}}(\mathrm{k})$ is the approximation of the innovation covariance. In fact, the true innovation covariance is given by

$$\mathbb{E}[\boldsymbol{z}_{\mathrm{i}}\boldsymbol{z}_{\mathrm{i}}^{T}|\boldsymbol{Z}_{\mathrm{i}}(\mathrm{k})] = \boldsymbol{S}_{\mathrm{i}}(\mathrm{k}) + \tilde{\boldsymbol{S}}_{\mathrm{i}}(\mathrm{k}) \tag{14}$$

$$\text{with } \tilde{\boldsymbol{S}}_{\mathrm{i}}(\mathrm{k}) = \mathbb{E}[\frac{1}{4}\frac{\partial^{2}\boldsymbol{h}_{\mathrm{i}}}{\partial \mathbf{x}_{\mathrm{i}}^{2}}(\mathbf{x}_{\mathrm{i}}-\hat{\mathbf{x}}_{\mathrm{i}})^{2}(\mathbf{x}_{\mathrm{i}}-\hat{\mathbf{x}}_{\mathrm{i}})^{2^{T}}\frac{\partial^{2}\boldsymbol{h}_{\mathrm{i}}^{T}}{\partial \mathbf{x}_{\mathrm{i}}^{2}} + \cdots |\boldsymbol{Z}_{\mathrm{i}}(\mathrm{k})]. \tag{15}$$

Here, $\boldsymbol{Z}_{\mathrm{i}}(\mathrm{k}) = \{\boldsymbol{z}_{\mathrm{i}}(1), \cdots, \boldsymbol{z}_{\mathrm{i}}(\mathrm{k})\} \in \mathbb{R}^{7\times \mathrm{k}}$ is the time series of innovations and $\tilde{\boldsymbol{S}}_{\mathrm{i}}$ is the expected value of the remainder of Taylor series expansion.
Subsequently, the Kalman gain ($\boldsymbol{K}_{\mathrm{i}}(\mathrm{k}) \in \mathbb{R}^{4\times 7}$) for the i-th link is given by

$$\boldsymbol{K}_{\mathrm{i}}(\mathrm{k}) = \boldsymbol{P}_{\mathrm{i}}^{-}(\mathrm{k})\boldsymbol{H}_{\mathrm{i}}^{T}(\hat{\mathbf{x}}_{\mathrm{i}}^{-}(\mathrm{k}))\boldsymbol{S}_{\mathrm{i}}^{-1}(\mathrm{k}). \tag{16}$$

Now we are able to update the estimation and its error covariance with

$$\hat{\mathbf{x}}_{\mathrm{i}}^{+}(\mathrm{k}) = \hat{\mathbf{x}}_{\mathrm{i}}^{-}(\mathrm{k}) + \boldsymbol{K}_{\mathrm{i}}(\mathrm{k})\boldsymbol{z}_{\mathrm{i}}(\mathrm{k}) \tag{17}$$

$$\boldsymbol{P}_{\mathrm{i}}^{+}(\mathrm{k}) = \left(\boldsymbol{I}_{4} - \boldsymbol{K}_{\mathrm{i}}(\mathrm{k})\boldsymbol{H}_{\mathrm{i}}(\hat{\mathbf{x}}_{\mathrm{i}}^{+}(\mathrm{k}))\right)\boldsymbol{P}_{\mathrm{i}}^{-}(\mathrm{k}), \tag{18}$$

where $\boldsymbol{I}_{4}$ is the identity matrix of size 4.

C. Adaptive Extended Kalman Filter

Kalman filter is, in general, capable of estimating parameters (variables with zero dynamics) along with the system states. Moreover, all the components of (extended) Kalman filter itself, including the initial state and the corresponding error covariance, process, and measurement noise covariance, can be considered as parameters, too [20]. The method we use for adapting the estimator is based on (i) parameterizing and (ii) iteratively estimating the noise covariance matrices $\mathbf{Q}_{\mathrm{i}}$ and $\mathbf{R}_{\mathrm{i}}$. For this we use the innovation $\boldsymbol{z}_{\mathrm{i}}$ and its covariance, which contains both $\mathbf{Q}_{\mathrm{i}}$ and $\mathbf{R}_{\mathrm{i}}$ (substitute $\boldsymbol{P}_{\mathrm{i}}^{-}(\mathrm{k})$ in (13) with (11)). We formulate this as a maximum likelihood problem, which seeks to maximize the likelihood function of noise covariance given the $N \in \mathbb{Z}^{+}$ recent innovations, as [21]

$$\max_{\boldsymbol{\theta}_{\mathrm{i}}} \boldsymbol{L}(\boldsymbol{\theta}_{\mathrm{i}}|\boldsymbol{Z}_{\mathrm{i}}(N)), \tag{19}$$

with $\boldsymbol{\theta}_{\mathrm{i}} = \{\mathbf{Q}_{\mathrm{i}}, \mathbf{R}_{\mathrm{i}}\}$ being the parameters, which correspond to the noise covariance matrices of the i-th link, and $\boldsymbol{Z}_{\mathrm{i}}(N) = \{\boldsymbol{z}_{\mathrm{i}}(\mathrm{k}-N+1), \cdots, \boldsymbol{z}_{\mathrm{i}}(\mathrm{k})\} \in \mathbb{R}^{7\times N}$ the time series of N recent innovations. By maximizing $\boldsymbol{L}(\boldsymbol{\theta}_{\mathrm{i}}|\boldsymbol{Z}_{\mathrm{i}}(N))$ we increase the probability of the true parameters with the N given innovations.
The analytical and iterative solution to (19) is initially proposed in [5] for linear systems. This method, however,

may violate the positive-definiteness of the noise covariance matrices [22]. Therefore, we use a modified version proposed by Mohamed et al. [7], which under some conditions, has proven to be effective for EKF and nonlinear systems, as well [23]. Take the moving average (with length N) of innovation sequence $\bar{\boldsymbol{C}}_{\mathrm{i}}(\mathrm{k}) \in \mathbb{R}^{7\times 7}$ as

$$\bar{\boldsymbol{C}}_{\mathrm{i}}(\mathrm{k}) = \frac{1}{N} \sum_{\mathrm{j}=\mathrm{k}-N+1}^{\mathrm{k}} \boldsymbol{z}_{\mathrm{i}}(\mathrm{j}) \boldsymbol{z}_{\mathrm{i}}^{T}(\mathrm{j}), \quad \text{with } \mathrm{k} \geq N \tag{20}$$

the measurement noise covariance matrix is adapted via [7]

$$\mathbf{R}_{\mathrm{i}}(\mathrm{k}+1) = \bar{\boldsymbol{C}}_{\mathrm{i}}(\mathrm{k}) + \boldsymbol{H}_{\mathrm{i}}(\hat{\mathbf{x}}_{\mathrm{i}}^{+}(\mathrm{k})) \boldsymbol{P}_{\mathrm{i}}^{+}(\mathrm{k}) \boldsymbol{H}_{\mathrm{i}}^{T}(\hat{\mathbf{x}}_{\mathrm{i}}^{+}(\mathrm{k})). \tag{21}$$

Moreover, the process noise covariance matrix is *approximately* given by [6]

$$\mathbf{Q}_{\mathrm{i}}(\mathrm{k}+1) = \boldsymbol{K}_{\mathrm{i}}(\mathrm{k}) \bar{\boldsymbol{C}}_{\mathrm{i}}(\mathrm{k}) \boldsymbol{K}_{\mathrm{i}}^{T}(\mathrm{k}). \tag{22}$$

As the adaptation laws for the noise covariance matrices are formulated, we discuss the stability properties of the system in the next section.

D. Notes on the Adaptation Stability

The estimation error is exponentially bounded in terms of the mean square, and consequently, the stability of EKF can be proven if [24]

- the dynamics model fulfills the nonlinear observability criteria,
- the initial state estimation error and its covariance are bounded,
- and the positive-definite noise covariance matrices remain bounded.

The (local) observability of nonlinear systems is generally investigated through the observability matrix, obtained based on Lie derivatives of the nonlinear measurement function w.r.t. the state transition function [25]. It can be shown that system (2) is observable [17]. Moreover, with the practical approach for initializing EKF, explained in Sec. III-B, the initial condition error can also be assumed to be bounded. By carefully choosing the initial positive-definite noise covariance matrices $\mathbf{Q}_{\mathrm{i}}(0)$ and $\mathbf{R}_{\mathrm{i}}(0)$, all three criteria for a stable conventional (non-adaptive) EKF are fulfilled. However, with the adaptive laws (21) and (22) discussed in the previous section, the third stability criterion, which is the boundedness of the noise covariance, is in question. To the best of the author's knowledge, the stability analysis of the reviewed adaptive extended Kalman filter, in general terms, is only marginal in the literature. Therefore, we propose several strategies in this section to maintain the estimator stability, given the adaptive mechanism.

a) Adaptation at Steady-state: As mentioned earlier, the proposed *non-adaptive* EKF is proven to be stable and can reach a steady state in a finite time. Moreover, note that the process noise adaptation law (22), which respects the positive definiteness of the covariance matrix, is the modified (and reduced) version of the original one proposed in [7]. The reduced version is valid only at the EKF steady state [6], where the rate of change of Kalman gain $\boldsymbol{K}_{\mathrm{i}}$ as well as the estimation error covariance matrix $\boldsymbol{P}_{\mathrm{i}}^{+}$ is at minimum. Therefore, we propose to employ the adaptation mechanisms (21) and (22) at the EKF steady state. Accordingly, the adaptation may start after some reasonable delay or, alternatively, once the rate of change of estimation error covariance drops below a certain threshold $\delta_{\mathrm{ss}} \in \mathbb{R}^{+}$, as

$$\|\boldsymbol{P}_{\mathrm{i}}^{+}(\mathrm{k}) - \boldsymbol{P}_{\mathrm{i}}^{+}(\mathrm{k}-1)\|_2 < \delta_{\mathrm{ss}}. \tag{23}$$

Here, $\|\cdot\|_2$ denotes matrix Euclidean norm.

b) Tunable Adaptation Rate: In order to add a degree of freedom to the adaptation rate, a forgetting factor $\alpha \in [0, 1)$ is used in the literature in various formulations (e.g. look at [26]–[28]). We introduce the forgetting factor to the adaptation mechanism as

$$\begin{aligned} &\mathbf{R}_{\mathrm{i}}(\mathrm{k}+1) = \alpha \mathbf{R}_{\mathrm{i}}(\mathrm{k}) + \\ &(1-\alpha)\left(\bar{\boldsymbol{C}}_{\mathrm{i}}(\mathrm{k}) + \boldsymbol{H}_{\mathrm{i}}(\hat{\mathbf{x}}_{\mathrm{i}}^{+}(\mathrm{k})) \boldsymbol{P}_{\mathrm{i}}^{+}(\mathrm{k}) \boldsymbol{H}_{\mathrm{i}}^{T}(\hat{\mathbf{x}}_{\mathrm{i}}^{+}(\mathrm{k}))\right), \end{aligned} \tag{24}$$

$$\mathbf{Q}_{\mathrm{i}}(\mathrm{k}+1) = \alpha \mathbf{Q}_{\mathrm{i}}(\mathrm{k}) + (1-\alpha)\left(\boldsymbol{K}_{\mathrm{i}}(\mathrm{k}) \bar{\boldsymbol{C}}_{\mathrm{i}}(\mathrm{k}) \boldsymbol{K}_{\mathrm{i}}^{T}(\mathrm{k})\right). \tag{25}$$

Obviously, with $\alpha = 1$ no adaptation takes place.

c) Stopping Criterion: An important property of maximum likelihood estimate is its consistency, which means that the estimate converges, in a probabilistic sense, to the true value of the variable as the number of sampled data grows without bounds. The maximum likelihood estimate, however, will generally be biased for a small number of samples [7]. In other words, only with a large enough moving average window N, the noise covariance matrices may converge to reliable values.

On the one hand, with a small window N, the adaptation may diverge and result in instability. In order to investigate this phenomenon, let l be the first iteration at which the adaptation mechanism is employed. Since conventional EKF is assumed to be stable,

$\forall\, \mathrm{k} \in \{1, \cdots, \mathrm{l}\},\ \exists\, \beta_F(\mathrm{k}), \beta_H(\mathrm{k}), \lambda_{\mathrm{R}}(\mathrm{k}), \lambda_{\mathrm{Q}}(\mathrm{k}), \lambda_P(\mathrm{k}) \in \mathbb{R}^{+}$ such that

$$\begin{aligned} &\|\boldsymbol{F}_{\mathrm{i}}(\hat{\mathbf{x}}_{\mathrm{i}}(\mathrm{k}))\| \leq \beta_F(\mathrm{k}), \qquad &&\|\boldsymbol{H}_{\mathrm{i}}(\hat{\mathbf{x}}_{\mathrm{i}}(\mathrm{k}))\| \leq \beta_H(\mathrm{k}), \\ &\|\mathbf{R}_{\mathrm{i}}(\mathrm{k})\| \leq \lambda_{\mathrm{R}}(\mathrm{k}), &&\|\mathbf{Q}_{\mathrm{i}}(\mathrm{k})\| \leq \lambda_{\mathrm{Q}}(\mathrm{k}), \\ &\|\boldsymbol{P}_{\mathrm{i}}(\mathrm{k})\| \leq \lambda_P(\mathrm{k}), \end{aligned}$$

where $\|\cdot\|$ is the matrix induced norm. Note that the positive definite matrices are bounded by their current maximum eigenvalues, denoted by $\lambda(\cdot) \in \mathbb{R}$. Moreover, positive definite matrices are positive operators [29]. This property would let us benefit from the triangle inequality theorem to determine the bounds of the induced norms. In particular, for positive operators $\boldsymbol{A}$ and $\boldsymbol{B}$ and positive scalar $c \in \mathbb{R}^{+}$ triangle inequality is given by

$$\|\boldsymbol{A} \pm c\boldsymbol{B}\| \leq \|\boldsymbol{A}\| + c\|\boldsymbol{B}\|. \tag{26}$$

Therefore and according to (13), the induced norm of innovation covariance $\boldsymbol{S}_{\mathrm{i}}(\mathrm{k})$ is bounded by

$$\|\boldsymbol{S}_{\mathrm{i}}(\mathrm{k})\| \leq \beta_S(\mathrm{k}) = \beta_H^2(\mathrm{k})\left(\beta_F^2(\mathrm{k})\lambda_P(\mathrm{k}) + \lambda_{\mathrm{Q}}(\mathrm{k})\right) + \lambda_{\mathrm{R}}(\mathrm{k}). \tag{27}$$

Furthermore, $\|\tilde{\boldsymbol{S}}_{\mathrm{i}}(\mathrm{k})\| \leq \beta_{\tilde{S}}(\mathrm{k}) \in \mathbb{R}^{+}$ holds, as the remainder of Taylor expansion is bounded, too. According to (14) and (20), it can be shown that

$$\begin{aligned}\left\|\bar{\boldsymbol{C}}_{\mathrm{i}}(\mathrm{k})\right\| &\leq \frac{\mathrm{k}}{N}\left(\|\boldsymbol{S}_{\mathrm{i}}(\mathrm{k})\| + \|\tilde{\boldsymbol{S}}_{\mathrm{i}}(\mathrm{k})\|\right) \\ &+ \frac{\mathrm{k}-\mathrm{N}}{N}\left(\|\boldsymbol{S}_{\mathrm{i}}(\mathrm{k}-\mathrm{N})\| + \|\tilde{\boldsymbol{S}}_{\mathrm{i}}(\mathrm{k}-\mathrm{N})\|\right) \\ &\leq \frac{\mathrm{k}}{N}\beta_{C}(\mathrm{k}) \end{aligned} \tag{28}$$

holds, for some finite $\beta_{C}(\mathrm{k}) \in \mathbb{R}^{+}$. Note that the norm bound of the windowed innovation covariance $\bar{\boldsymbol{C}}_{\mathrm{i}}(\mathrm{k})$ scales up with the iteration number k. More specifically, with relatively small Ns, $\left\|\bar{\boldsymbol{C}}_{\mathrm{i}}(\mathrm{k})\right\|$ grows with each iteration. This might cause instability, as the adaption laws (21) and (22) are closely dependent on $\bar{\boldsymbol{C}}_{\mathrm{i}}(\mathrm{k})$.

On the other hand, finding large-enough N might be challenging. Moreover, large Ns can be computationally expensive, too. Therefore, in order to prevent potential instabilities, we propose a simple yet effective test to stop the adaptation, based on the condition number (the ratio between the matrix maximum and minimum singular values) of the adapted matrices, as

$$\kappa(\mathbf{R}_{\mathrm{i}}(\mathrm{k})) > \delta_{\text{cnd}} \tag{29}$$

$$\text{Or } \kappa(\mathbf{Q}_{\mathrm{i}}(\mathrm{k})) > \delta_{\text{cnd}}. \tag{30}$$

Here, $\kappa(\cdot) : \mathbb{R}^{\mathrm{p}\times\mathrm{q}} \to \mathbb{R}^{+}$ (with p and q being arbitrary variables) denotes matrix condition number. In other words, once the condition numbers exceed the threshold $\delta_{\text{cnd}} \in \mathbb{R}^{+}$, the adaption stops, and the system works as a conventional EKF.

Algorithm 1 summarizes the adaptive EKF, discussed in this section. Note that the adaptive EKF starting criterion (23) is examined only as long as it has not held true in this algorithm. Once the rate of change of $\boldsymbol{P}_{\mathrm{i}}^{+}$ drops below the threshold δ_{ss}, adaptive EKF is activated, and this criterion is no longer investigated. Similarly, once the condition number of either of noise covariance matrices exceeds the threshold δ_{cnd} according to (29) and (30), adaptive EKF stops.

Algorithm 1 Adaptive extended Kalman filter for link i

Require: $\hat{\mathbf{x}}_{\mathrm{i}}(0)$, $\boldsymbol{P}_{\mathrm{i}}(0)$, $\mathbf{R}_{\mathrm{i}}(0)$, $\mathbf{Q}_{\mathrm{i}}(0)$ ▷ Conventional EKF
Require: N, α, δ_{ss}, δ_{cnd} ▷ Adaptive EKF
  $\mathrm{k} \leftarrow 0$
  AEKF $\leftarrow$ false ▷ Do not start the adaptation
  while true **do**
    Predict $\hat{\mathbf{x}}_{\mathrm{i}}^{-}(\mathrm{k})$ (10)
    Predict $\boldsymbol{P}_{\mathrm{i}}^{-}(\mathrm{k})$ (11)
    Read current measurements $\mathbf{y}_{\mathrm{i}}(\mathrm{k})$
    Compute Kalman gain $\boldsymbol{K}_{\mathrm{i}}(\mathrm{k})$ (16)
    Correct $\hat{\mathbf{x}}_{\mathrm{i}}^{+}(\mathrm{k})$ (17)
    Correct $\boldsymbol{P}_{\mathrm{i}}^{+}(\mathrm{k})$ (18)
    if AEKF is false & (23) is true **then**
      AEKF $\leftarrow$ true ▷ Start the adaptation
    end if
    if $\mathrm{k} \geq N$ & AEKF is true & (29) is false & (30) is false **then**
      Adapt $\mathbf{R}_{\mathrm{i}}(\mathrm{k}+1)$ (24)
      Adapt $\mathbf{Q}_{\mathrm{i}}(\mathrm{k}+1)$ (25)
    else
      $\mathbf{R}_{\mathrm{i}}(\mathrm{k}+1) \leftarrow \mathbf{R}_{\mathrm{i}}(\mathrm{k})$
      $\mathbf{Q}_{\mathrm{i}}(\mathrm{k}+1) \leftarrow \mathbf{Q}_{\mathrm{i}}(\mathrm{k})$
    end if
    $\mathrm{k} \leftarrow \mathrm{k}+1$
  end while

IV. Experimental Evaluation

This section investigates the theories and topics covered in Sec. III. Initially, we explain the experiment setup, including the method to obtain the ground truth signal. Subsequently, the necessity of using excitation trajectory in the experiments as well as the implementation, are outlined. Finally, we evaluate estimator (2) with 7 IMUs installed on all robot links, with full-body actuation.

A. Experiment Setup

Here, the robot is equipped with one IMU per link. The acceleration and rotational data is acquired using seven onboard STMicroelectronics LSM6DSOTR IMU sensors. The full-scale range is set to 2 g for the accelerometer and 2000 $^{\circ}/s$ for the gyroscope. The output data rate is set to 6.67 kHz for both sensors. All IMU internal filters are deactivated. The data acquisition system (*DAQ*) includes an Analog Devices LTC2864-1 RS485 Full-Duplex transceiver on each IMU board and on the main board. The use of high-bandwidth RS485 transceivers between the IMUs and the main board allows for faster and more robust communication. The onboard microcontroller is capable of handling up to eight sensors and six channels (three for the accelerometer and three for the gyroscope) per sensor at 10 kS/s/ch. Consequently, the data is sent (based on the Master request) via the EtherCAT bus at a 1 kHz sampling rate. A Control PC (x64) running an Ubuntu 20.04 real-time system hosts the EtherCAT Master using Etherlab. Finally, the data is recorded in real-time via Matlab/Simulink in the control PC. Another real-time system provides the robot with desired velocity trajectory. The high-fidelity ground truth is obtained based on numerical differentiation, followed by lag-free filtering of the encoder measurements. The experimental setup is depicted in Fig. 2.

B. Excitation Trajectory

In order to fully evaluate the bandwidth of the estimator (2), we move the robot in a persistently exciting trajectory in full-dynamics motions. These trajectories are conventionally used for estimating the robot manipulators dynamics (see, e.g., [30]). The robot is actuated to follow a 30-second excitation trajectory. This periodic Fourier-like trajectory [31]

$$q_{\mathrm{i}}(t) = q_{\mathrm{i},0} + \sum_{\mathrm{r}=1}^{R} \frac{a_{\mathrm{r,i}}}{r\omega}\sin(\mathrm{r}\omega t) - \sum_{\mathrm{r}=1}^{R} \frac{b_{\mathrm{r,i}}}{r\omega}\cos(\mathrm{r}\omega t), \tag{31}$$

respects all robot joint limitations from the datasheet. The number of harmonics in this trajectory is $R = 60$, and

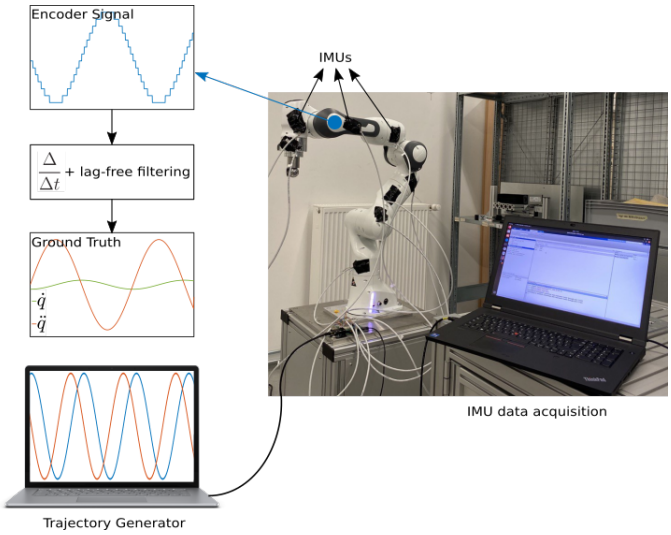

Fig. 2. The schematic of the experimental setup

the base frequency is $\omega = 2\pi/30$. These parameters are obtained via an optimization routine, where robot joint and torque limits are provided as constraints. These coefficients are optimized for minimal uncertainty in identifying robot base parameters [31]. Therefore, this trajectory persistently excites the system along the whole bandwidth and subsequently evaluates the estimator (2) performance.

C. Results

In order to fully evaluate the proposed method, the adaptive Kalman filter is employed with different settings. Table I summarizes these settings. EKF refers to conventional EKF with no adaptation. Moreover, we compare the performance of our method with the state-of-the-art one, introduced in [22] (abbreviated by *SotA* in table I). Instead of maximum likelihood estimation, the state-of-the-art algorithm employs covariance matching techniques to adapt the noise covariance. Similar to (13), the innovation covariance is obtained based on the linearization of the measurement function via Taylor approximation in SotA. Moreover, $\alpha = 0.3$ is proposed to be a stable choice in [22]. The initial condition, including initial state, state estimation error covariance, and noise covariance, are equal across different estimators. Note that the initial noise covariance is obtained based on large samples from different trajectories. Therefore, the conventional EKF is supposed to perform well. Due to the recursive nature of the estimator (2) and accumulation of the error from the first to the last (seventh) link, seventh link velocity and acceleration are most challenging to estimate. Also, we have observed the minimum (if any) improvement from the conventional EKF to the adaptive EKFs in the last link. Therefore, as the worst case scenario, Fig. 3 depicts the seventh link velocity and acceleration estimation based on different methods, briefed in Tab. I. Moreover, Fig. 4 compares the *normalized root-mean-square error (NRMSE)* of the estimations for different scenarios. According to Fig. 3 (a) and (b), the conventional EKF performs acceptable, given the finely-tuned initial conditions. Moreover, with the given trajectory, $N = 1000$ is a small sample size for adapting noise covariance. This has resulted in a biased maximum-likelihood estimation in (19), which degraded the estimation accuracy in AEKF1 (see Fig. 3 (c) and (d)). However, with the tight threshold $\delta_{\text{cnd}} = 10^8$, the estimator never diverges. On the other hand, with a looser $\delta_{\text{cnd}} = 10^{12}$ in AEKF2, the filter diverges for some iterations and consequently converges again (see Fig. 3 (e) and (f)). This illustrates the effectiveness of stopping criteria (29) and (30), where the filter stability is maintained, although N is poorly chosen. AEKF3 can reduce the noise effects, compared to conventional EKF (see Fig. 3 (a), (b), (g), (h), (k), and (l)). This is especially true in the estimated link acceleration (see Fig. 4 (b),(d)), where the noise effects are more severe. Note that for better visualization purposes, AKEF2 results are circumvented from Fig. 4 (a), (b), (c), and (d). Furthermore, covariance matching is not a suitable technique for adapting the system (2) noise covariance (see Fig. 3 (i) and (j)). This may be due to the fact that covariance matching relies mainly on the innovation covariance matrix given by (13). However, this variable loses precision in nonlinear systems because of the Taylor approximation (see (14) and (15)). On the other hand, although windowed, the *true* and *recent* innovation covariance is employed in maximum likelihood methods. Moreover, Fig. 4 (e) and (f) show the effects of the window size N, when the forgetting factor is constant. In these experiments, the stopping threshold is set to $\delta_{\text{cnd}} = 10^8$. Also, for the given trajectory, when the window size is large enough ($N = 5000$), the contribution of forgetting factor (α) is negligible (see Fig. 4 (g) and (h)). In summary and according to Fig. 4, the noise covariance adaptation method, which is discussed in this paper, performs more competently in the first links when the filter initial condition is tuned correctly.

TABLE I
DIFFERENT SETTINGS FOR ADAPTIVE EXTENDED KALMAN FILTER

Abbreviation	Adaptation Method	Window Size N	Forgetting Factor α	Threshold δ_{cnd} in (29) and (30)
EKF	-	-	-	-
AEKF1	(24) and (25)	1000	0.3	10^8
AEKF2	(24) and (25)	1000	0.3	10^{12}
AEKF3	(24) and (25)	5000	0.3	10^8
SotA	[22]	-	0.3	-

V. CONCLUSIONS

In this paper, we initially reviewed a method for estimating robot manipulators link velocity and acceleration. Traditionally, EKF is used for this purpose. However, EKF is prone to errors in the dynamics model and noise information provided to this estimator. Even with precise modeling of the process and noise, this approach is vulnerable to inevitable sensor changes caused by environmental effects. Therefore, we discussed a stable algorithm for adapting the noise covariance in the estimator. Finally, we evaluated our findings with a

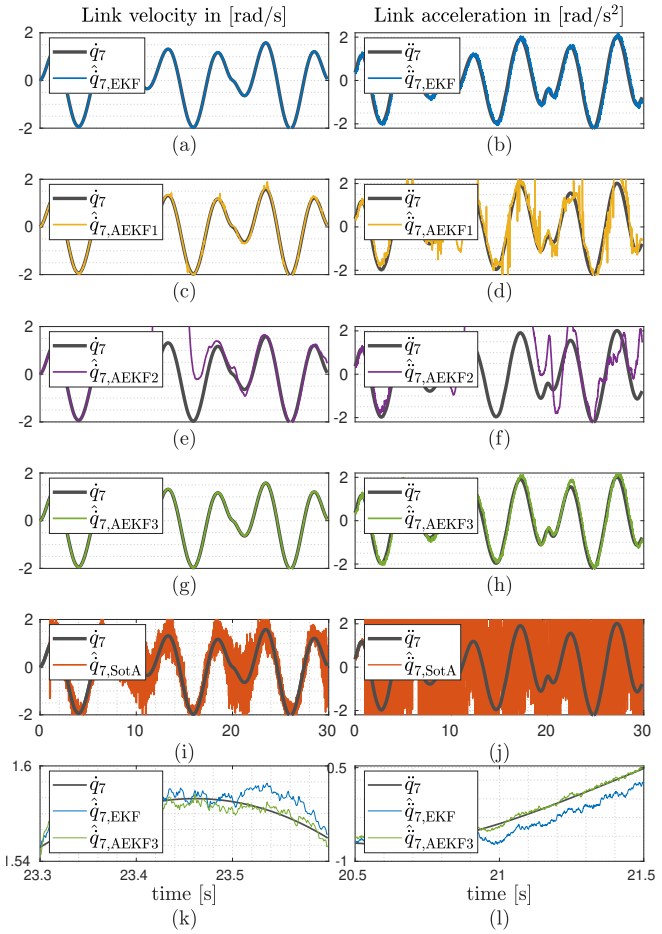

Fig. 3. Seventh link velocity and acceleration estimation via different methods, summarized in Tab. I. In this experiment, each robot link is equipped with one IMU. The robot moves in an excitation trajectory, and all kinematics states are estimated using estimator (2). Here, the ground truth ($\dot{q}_7$ and $\ddot{q}_7$) is the encoder signal, which is differentiated and filtered via lag-free filtering methods. Estimated link (a) velocity and (b) acceleration using conventional EKF. Estimated link (c) velocity and (d) acceleration using AEKF1. Estimated link (e) velocity and (f) acceleration using AEKF2. Estimated link (g) velocity and (h) acceleration using AEKF3. Estimated link (i) velocity and (j) acceleration using SotA. A close comparison of estimated link (k) velocity and (l) acceleration via EKF and AEKF3 in short time intervals.

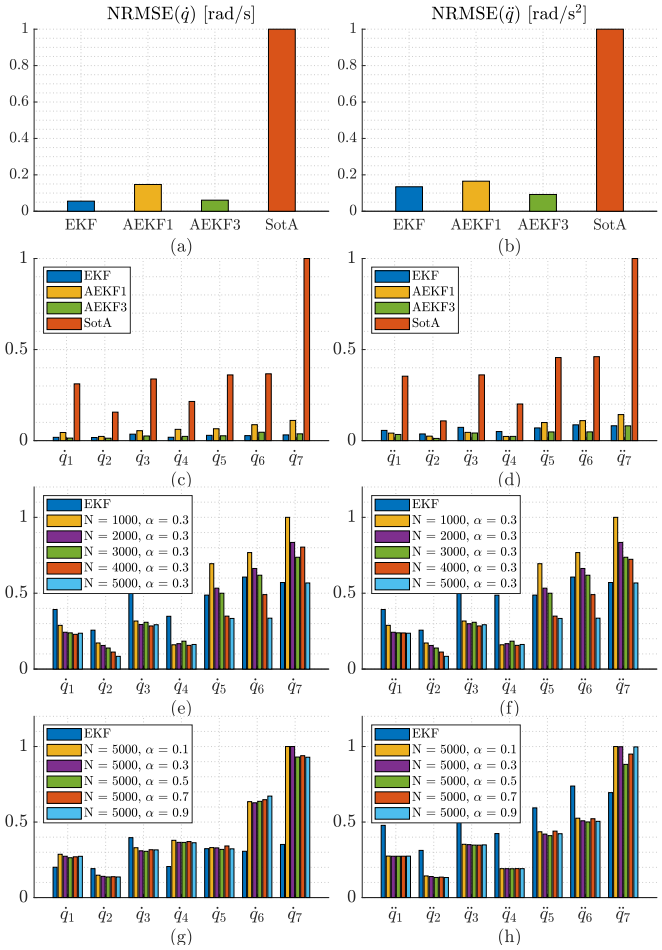

Fig. 4. (a) NRMSE of link velocity estimation via the methods, summarized in Tab. I and for all links. (b) NRMSE of acceleration estimation error via the methods, summarized in Tab. I and for all links. (c) NRMSE of link velocity estimation via the methods, summarized in Tab. I for each link, separately. (d) NRMSE of acceleration estimation error via the methods, summarized in Tab. I for each link separately. (e) NRMSE of link velocity estimation, with different window size N, when the forgetting factor is constant ($\alpha = 0.3$). (f) NRMSE of link acceleration estimation, with different window size N, when the forgetting factor is constant ($\alpha = 0.3$). (g) NRMSE of link velocity estimation, with different forgetting factor α, when the window size is constant ($N = 5000$). (h) NRMSE of link acceleration estimation, with different forgetting factor α, when the window size is constant ($N = 5000$).

seven-DoF manipulator. Every robot link is equipped with one IMU in this experiment. Consequently, EKF is initialized with relatively accurate values obtained from large samples. In the case of nearly precise initial noise covariance, the results corresponding to the proposed method are promising, especially for the first links.

REFERENCES

[1] D.-J. Jwo, M.-Y. Chen, C.-H. Tseng, and T.-S. Cho, "Adaptive and nonlinear Kalman filtering for GPS navigation processing," *Kalman Filter: Recent Advances and Applications*, vol. 19, 2009.

[2] T. Soong, "On a priori statistics in minimum-variance estimation problems," *Journal of basic engineering*, 1965.

[3] D. Simon, *Optimal state estimation: Kalman, H infinity, and nonlinear approaches*. John Wiley & Sons, 2006.

[4] K. Xiong, H. Zhang, and L. Liu, "Adaptive robust extended Kalman filter for nonlinear stochastic systems," *IET Control Theory & Applications*, vol. 2, no. 3, pp. 239–250, 2008.

[5] R. Mehra, "On the identification of variances and adaptive Kalman filtering," *IEEE Transactions on automatic control*, vol. 15, no. 2, pp. 175–184, 1970.

[6] C. T. Fraser and S. Ulrich, "Adaptive extended Kalman filtering strategies for spacecraft formation relative navigation," *Acta Astronautica*, vol. 178, pp. 700–721, 2021.

[7] A. Mohamed and K. Schwarz, "Adaptive Kalman filtering for INS/GPS," *Journal of geodesy*, vol. 73, pp. 193–203, 1999.

[8] B. Gao, S. Gao, G. Hu, Y. Zhong, and C. Gu, "Maximum likelihood principle and moving horizon estimation based adaptive unscented Kalman filter," *Aerospace Science and Technology*, vol. 73, pp. 184–196, 2018.

[9] Y. Xi, X. Tang, Z. Li, Y. Cui, T. Zhao, X. Zeng, J. Guo, and W. Duan, "Harmonic estimation in power systems using an optimised adaptive Kalman filter based on pso-ga," *IET Generation, Transmission & Distribution*, vol. 13, no. 17, pp. 3968–3979, 2019.

[10] R. P. Tripathi, S. Ghosh, and J. Chandle, "Tracking of object using optimal adaptive Kalman filter," in *2016 IEEE International Conference on Engineering and Technology (ICETECH)*. IEEE, 2016, pp. 1128–1131.

[11] S. Fakoorian, A. Santamaria-Navarro, B. T. Lopez, D. Simon, and A.-a. Agha-mohammadi, "Towards robust state estimation by boosting the maximum correntropy criterion Kalman filter with adaptive behaviors," *IEEE Robotics and Automation Letters*, vol. 6, no. 3, pp. 5469–5476, 2021.

[12] S. Karaçam, K. Yalçin, and T. S. Navruz, "An adaptive EKF algorithm with adaptation of noise statistic based on MLE, EM and ICE," in *Smart Applications with Advanced Machine Learning and Human-Centred Problem Design*. Springer, 2023, pp. 185–198.

[13] S. A. B. Birjandi, J. Kühn, and S. Haddadin, "Joint velocity and acceleration estimation in serial chain rigid body and flexible joint manipulators," in *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Nov 2019, pp. 7503–7509.

[14] S. A. B. Birjandi, E. Pozo Fortunić, and S. Haddadin, "Evaluation of robot manipulator link velocity and acceleration observer," in *22nd World Congress of the International Federation of Automatic Control (IFAC 2023)*. Elsevier, 2023, pp. 313–320.

[15] S. A. B. Birjandi, J. Kühn, and S. Haddadin, "Observer-extended direct method for collision monitoring in robot manipulators using proprioception and IMU sensing," *IEEE Robotics and Automation Letters*, vol. 5, no. 2, pp. 954–961, April 2020.

[16] S. A. B. Birjandi and S. Haddadin, "Model-adaptive high-speed collision detection for serial-chain robot manipulators," *IEEE Robotics and Automation Letters*, vol. 5, no. 4, pp. 6544–6551, 2020.

[17] S. A. B. Birjandi, N. Dehio, , A. Kheddar, and S. Haddadin, "Robust Cartesian kinematics estimation for task-space control system," in *2023 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Nov 2022.

[18] M. W. Spong, "Modeling and control of elastic joint robots," *Journal of dynamic systems, measurement, and control*, vol. 109, no. 4, pp. 310–318, 1987.

[19] B. Siciliano, O. Khatib, and T. Kröger, *Springer handbook of robotics*. Springer, 2008, vol. 200.

[20] B. P. Gibbs, *Advanced Kalman filtering, least-squares and modeling: a practical handbook*. John Wiley & Sons, 2011.

[21] A. H. Sayed, *Adaptive filters*. John Wiley & Sons, 2011.

[22] S. Akhlaghi, N. Zhou, and Z. Huang, "Adaptive adjustment of noise covariance in Kalman filter for dynamic state estimation," in *2017 IEEE power & energy society general meeting*. IEEE, 2017, pp. 1–5.

[23] M. Khodarahmi and V. Maihami, "A review on Kalman filter models," *Archives of Computational Methods in Engineering*, pp. 1–21, 2022.

[24] K. Reif, S. Gunther, E. Yaz, and R. Unbehauen, "Stochastic stability of the discrete-time extended Kalman filter," *IEEE Transactions on Automatic control*, vol. 44, no. 4, pp. 714–728, 1999.

[25] G. Besançon, *Nonlinear observers and applications*. Springer, 2007, vol. 363.

[26] K.-H. Kim, G.-I. Jee, C.-G. Park, and J.-G. Lee, "The stability analysis of the adaptive fading extended Kalman filter using the innovation covariance," *International Journal of Control, Automation, and Systems*, vol. 7, no. 1, p. 49, 2009.

[27] M. Song, R. Astroza, H. Ebrahimian, B. Moaveni, and C. Papadimitriou, "Adaptive Kalman filters for nonlinear finite element model updating," *Mechanical Systems and Signal Processing*, vol. 143, p. 106837, 2020.

[28] M. Wu, L. Qin, G. Wu, Y. Huang, and C. Shi, "State of charge estimation of power lithium-ion battery based on a variable forgetting factor adaptive Kalman filter," *Journal of Energy Storage*, vol. 41, p. 102841, 2021.

[29] R. Bhatia and F. Kittaneh, "Norm inequalities for positive operators," *Letters in Mathematical Physics*, vol. 43, no. 3, pp. 225–231, Feb 1998. [Online]. Available: https://doi.org/10.1023/A:1007432816893

[30] C. G. Atkeson, C. H. An, and J. M. Hollerbach, "Estimation of inertial parameters of manipulator loads and links," *The International Journal of Robotics Research*, vol. 5, no. 3, pp. 101–119, 1986.

[31] J. Swevers, C. Ganseman, D. B. Tukel, J. De Schutter, and H. Van Brussel, "Optimal robot excitation and identification," *IEEE transactions on robotics and automation*, vol. 13, no. 5, pp. 730–740, 1997.