RESEARCH ARTICLE

Model-Based Fault Diagnosis Algorithms for Robotic Systems

AGUS HASAN [1], **(Senior Member, IEEE), MARYAMSADAT TAHAVORI** [2]**, (Member, IEEE), AND HENRIK SKOV MIDTIBY** [3]**, (Member, IEEE)**

[1]Department of ICT and Natural Sciences, Norwegian University of Science and Technology, 6025 Ålesund, Norway
[2]Department of Engineering Technology and Didactics, Technical University of Denmark, 2750 Ballerup, Denmark
[3]Mærsk McKinney Møller Institute, University of Southern Denmark, 5230 Odense, Denmark

Corresponding author: Agus Hasan (agus.hasan@ntnu.no)

ABSTRACT In this paper, three model-based fault diagnosis algorithms for robotic systems are designed, compared, simulated, and implemented. The first algorithm is based on a nonlinear adaptive observer (NLAO), where a sufficient condition for the convergence of the estimator is derived in terms of linear matrix inequality (LMI) under persistence of excitation condition. The second algorithm is based on an adaptive extended Kalman filter (AEKF). Unlike traditional approaches, where the fault parameters are considered as augmented state variables, the AEKF directly estimates the fault parameters from measurement data. The third algorithm is based on a cascade of a nonlinear observer (NLO) and a linearized adaptive Kalman filter (LAKF), called the adaptive exogenous Kalman filter (AXKF). The pros and cons for each algorithm are discussed. The performance of the algorithms is compared in a single-link joint robot system. Furthermore, the algorithms are implemented in a ball-balancing robot to detect and estimate the magnitude of the actuator faults.

INDEX TERMS Fault diagnosis, nonlinear observer, Kalman filter, robotics.

I. INTRODUCTION

Robotic systems are vulnerable to various types of faults. If faults appear in such systems, it can cause harm or injury for people and other adverse consequences. Fault diagnosis algorithms can be used as a first step to mitigate these problems [1]. Fault diagnosis consists of three steps: detection, localization, and estimation. While fault detection and localization algorithms have been quite well-established, fault estimation received less attention since the problem is more difficult to solve [2].

Two approaches for fault diagnosis of dynamical systems can be found in the literature. The first approach includes methods in which the system model is not needed. Instead of the system model, the measurement data is required for fault diagnosis. The main drawback of this approach is the large set of data that is required for proper operation. Neural networks, pattern recognition, variational mode

The associate editor coordinating the review of this manuscript and approving it for publication was Guillermo Valencia-Palomo [ID].

decomposition, Bayesian network method, statistical and expert methods are the examples of this approach [3], [4], [5], [6]. Recent advancement based on this approach is by using machine learning methods, such as deep learning [7], [8], long short-term memory (LSTM) [9], and support vector machine [10]. The second approach includes methods in which the mathematical model of the system is needed (model-based approach) [11]. This paper focuses on the second approach and is an extension of the first author's two IEEE conference papers presented in [12] and [13]. The extension includes design of fault diagnosis algorithms based on nonlinear adaptive observer (NLAO) and adaptive extended Kalman filter (AEKF). In particular, we design, compare, simulate, and implement the two aforementioned model-based fault diagnosis algorithms with the adaptive exogenous Kalman filter (AXKF). In contrast with the existing model-based algorithms, where the fault parameters are augmented as additional state variables, the algorithms presented in this paper estimate the faults directly from measurement data. To show the performance of the algorithms,
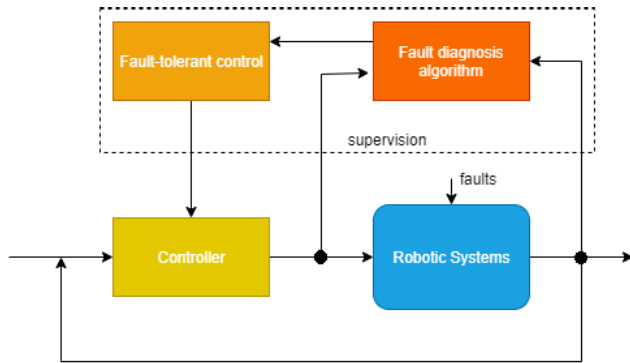
FIGURE 1. Schematic diagram of fault diagnosis for robotic systems.

we perform numerical simulations based on a single-link flexible joint robot system and a real experiment in a ball-balancing robot.

To outline the contribution of this paper into a general perspective, the reader can refer to Fig. 1. In this case, the contribution is the development of fault diagnosis algorithms for robotic systems. The information regarding fault diagnosis can be utilized as a decision support system. If the fault is detected, the robotic systems can either decide to stop or continue the operation. When the operation is decided to continue, the information regarding the magnitude of the fault (fault estimation) can be used by a fault-tolerant control algorithm to compensate the fault. A possible algorithm for the fault-tolerant control is by using model predictive control (MPC) [14]. In this case, the fault estimation is used to predict the behaviour of the robotic systems in the prediction horizon interval.

A. LITERATURE REVIEW

In principle, model-based fault diagnosis exploits state estimation methods to determine if the system outputs are consistent with the occurrence of a fault. Many methods for state estimation can be found in the literature. For example, Kalman filter (KF) has been proven as an optimal solution for state estimation of linear dynamic systems [15]. KF has been widely used in different applications due to its simplicity, tractability, and robustness. However, linear KF can not be used for systems with severe nonlinearity. Since most of the systems are nonlinear, sub-optimal state estimation techniques can be used. Extended Kalman filter (EKF) is one of these sub-optimal techniques in which measurements and system model equations are linearized [16], such that the linear KF algorithm can be implemented. However, linearization in the EKF causes instability of this method particularly for extremely nonlinear systems. Unscented Kalman filter (UKF) was presented in [17] to improve the deficiency of the EKF. The UKF uses a collection of sigma points to estimate the mean and covariance matrix propagation [18], [19]. Another group of model-based state estimation techniques is based on nonlinear observer (NLO). This method takes a global asymptotic or exponential stability as a design starting point,

and then employ tuning parameters to follow desired performance [20], [21], [22]. The aforementioned state estimation techniques have been widely used in different applications such as lithium-ion battery state of charge estimation [23], [24], attitude estimation and fault diagnosis of unmanned aircraft systems (UAVs) [25], autonomous mapping and exploring [26], locomotion planning of humanoid robots [27], and state estimation for cyber-physical systems [28].

B. CONTRIBUTION OF THIS PAPER

Fault diagnosis methods based on state and parameter estimation techniques presented in the literature review treated the fault parameters as additional state variables of the systems. This approach, however, can cause the augmented system to be unstable. Thus, there is an incentive to design estimation methods that can estimate the fault parameters directly from the measurement data. In this paper, we present three model-based actuator fault diagnosis algorithms for robotic systems, for which the fault parameters are estimated directly from the sensor data. The first algorithm is based on an NLAO, where a sufficient condition for the NLAO is derived in terms of linear matrix inequality (LMI). The second algorithm is based on an AEKF. In contrast with the traditional approach in which the fault parameters are modelled as additional state variables, the method estimates the fault parameters under persistence of excitation condition. The third algorithm is based on a cascade of nonlinear observer (NLO) and linearized adaptive Kalman filter (LAKF), called the AXKF. The algorithms are tested in numerical simulations and implemented in a ball-balancing robot.

C. ORGANIZATION OF THIS PAPER

This paper is organized as follows. The problem formulation is presented in Section II. The NLAO, AEKF, and AXKF algorithms are presented in Section III, IV, and V, respectively. The performance of the algorithms is compared in Section VI by implementing the algorithms for a single-link flexible joint robot model. In section VII, the algorithms are used to estimate the fault parameters in a ball-balancing robot. Finally, the conclusion is presented in Section VIII.

II. PROBLEM FORMULATION

Before we formulate the problem, we define symbols and notations used throughout this paper. We denote $\mathbb{R}^n$ as the n-dimensional real Euclidean space, I_n as the n-dimensional identity matrix, and 0_n as the n-dimensional null matrix. The symbol $\langle \cdot, \cdot \rangle$ represents the inner product in $\mathbb{R}^n$. This means for $x, y \in \mathbb{R}^n$, $\langle x, y \rangle = x^\mathsf{T} y$, where x^T denotes the transpose of x. The notation $\|x\|$ denotes the Euclidean norm of vector x in $\mathbb{R}^n$. The norm of matrix A is defined as $\|A\| = \sqrt{\lambda_{max}(A^\mathsf{T} A)} = \sigma_{max}(A)$, where λ_{max} and σ_{max} denote the largest eigenvalue of $A^\mathsf{T} A$ and the largest singular value of A, respectively. Furthermore, $A > 0$ means A is positive definite (all eigenvalues are positive), while $A < 0$ means A is negative definite (all eigenvalues are negative).

Let us consider dynamic models of robotic systems, which can be transformed into the following nonlinear discrete-time form:

$$x_{k+1} = Ax_k + f(x_k) + Bu_k + \Phi_k \theta + w_k \qquad (1)$$
$$y_k = Cx_k + v_k \qquad (2)$$

Remark that system (1) consists of a linear term Ax_k, a nonlinear term $f(x_k)$, a control term Bu_k, a faulty term $\Phi_k\theta$, and an uncertainty term w_k. Specifically, $x_k \in \mathbb{R}^n$ denotes the state vector and $A \in \mathbb{R}^{n \times n}$ denotes the state transition matrix. The nonlinear term is represented by $f : \mathbb{R}^n \to \mathbb{R}^n$. The input matrix and input vector are denoted by $B \in \mathbb{R}^{n \times l}$ and $u_k \in \mathbb{R}^l$, respectively. The output vector is denoted by $y_k \in \mathbb{R}^m$ and the measurement matrix is denoted by $C \in \mathbb{R}^{m \times n}$. The faults are assumed to be at the actuator and are represented by the term $\Phi_k\theta$. Here, the matrix sequence $\Phi_k \in \mathbb{R}^{n \times p}$ is assumed to be known. The value of the fault is represented by the constant or piecewise constant with rare jumps vector parameter $\theta \in \mathbb{R}^p$ and is unknown. If the fault is time-varying, then the algorithm may not converge. If $p = l$, then the matrix sequence Φ_k is given by

$$\Phi_k = -B\mathrm{diag}(u_k) \qquad (3)$$

In this case, when faults occur then the nominal control term Bu_k becomes

$$Bu_k - B\mathrm{diag}(u_k)\theta = B(I_l - \mathrm{diag}(\theta))u_k \qquad (4)$$

To model uncertainty both in the model and the measurement, we are adding noise w_k and v_k, respectively. The noises are assumed to be zero mean Gaussian white noise with covariance matrices Q_k^F and R_k^F, i.e., $w_k \sim (0, Q_k^F)$, $v_k \sim (0, R_k^F)$. We assume A, B, C, Φ_k, Q_k^F, and R_k^F are upper bounded, the pair (A, C) is uniformly completely observable, and the pair $(A, \sqrt{Q_k^F})$ is uniformly completely controllable. Furthermore, we limit the scope of the paper by assuming no time delay on the measurement.

The fault diagnosis problem is to estimate the state vector x_k and the unknown vector parameter θ using measurement data y_k under the process and measurement noise w_k and v_k. To answer this problem, we present three fault diagnosis algorithms in Section III, IV, and V, respectively. The schematic diagram of the three algorithms are presented in Fig. 2. The NLAO-based algorithm is simple and straightforward. Measurement data is utilized by the algorithm after we design an appropriate observer gain. The observer gain can normally be obtained by solving an LMI. The AEKF-based algorithm uses local approximation of nonlinear models to calculate the covariance estimate. The covariance is then used to update the state and parameter estimate of the nonlinear system. The AXKF-based algorithm added an NLO into the AEKF. This will result in global convergence and more accurate estimates.

III. ALGORITHM BASED ON NONLINEAR ADAPTIVE OBSERVER (NLAO)

In this section, we derive a sufficient condition for the NLAO-based fault diagnosis algorithm in terms of LMI.
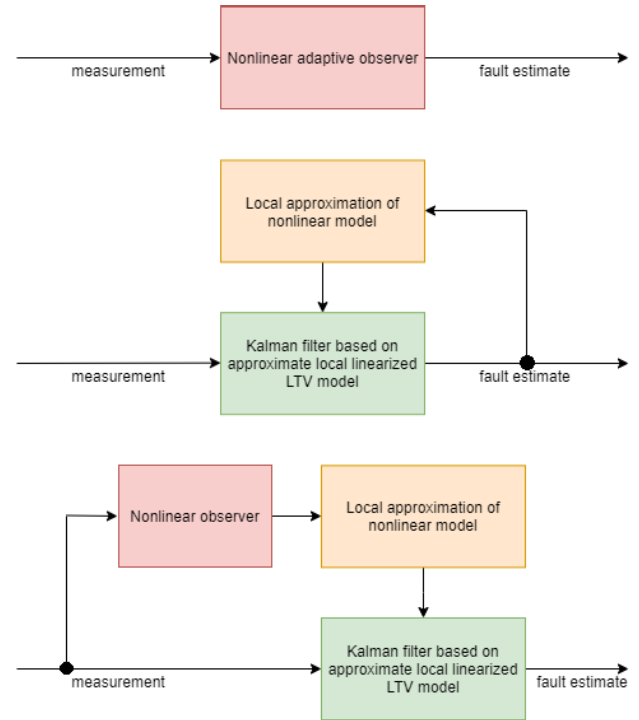

FIGURE 2. Schematic diagram of the NLAO (top), the AEKF (middle), the AXKF (bottom).

Deriving an NLAO for general nonlinearity f is not trivial, thus, we restrict our analysis only for Lipschitz nonlinearity. In particular, we consider f as a one-sided nonlinear Lipschitz function that satisfies a quadratic inner-boundedness condition. To this end, let us assume the nonlinear function f satisfies

$$\|f(x_k) - f(\check{x}_k)\| \le \gamma \|x_k - \check{x}_k\| \qquad (5)$$

$\forall x_k, \check{x}_k \in \mathbb{R}^n$. The constant parameter $\gamma > 0$ is called the Lipschitz constant. If there is a constant $\rho \in \mathbb{R}$ such that the inequality

$$\langle f(x_k) - f(\check{x}_k), x_k - \check{x}_k \rangle \le \rho \|x_k - \check{x}_k\|^2 \qquad (6)$$

is satisfied, then the nonlinear function f is called a one-sided Lipschitz function, i.e., for $\epsilon_1 > 0$, we have

$$\epsilon_1 \begin{pmatrix} \check{x}_k \\ \Delta f_k \end{pmatrix}^\mathsf{T} \begin{pmatrix} \rho I_n & -\dfrac{I_n}{2} \\ -\dfrac{I_n}{2} & 0_n \end{pmatrix} \begin{pmatrix} \check{x}_k \\ \Delta f_k \end{pmatrix} \ge 0 \qquad (7)$$

where $\check{x}_k = x_k - \check{x}_k$ and $\Delta f_k = f(x_k) - f(\check{x}_k)$. Furthermore, if there are constants $\beta, \eta \in \mathbb{R}$ such that

$$\Delta f_k^\mathsf{T} \Delta f_k \le \beta \|x_k - \check{x}_k\|^2 + \eta \langle x_k - \check{x}_k, \Delta f_k \rangle \qquad (8)$$

then f satisfies the quadratic inner-boundedness condition, i.e., for $\epsilon_2 > 0$, we have

$$\epsilon_2 \begin{pmatrix} \check{x}_k \\ \Delta f_k \end{pmatrix}^\mathsf{T} \begin{pmatrix} \beta I_n & \dfrac{\eta I_n}{2} \\ \dfrac{\eta I_n}{2} & -I_n \end{pmatrix} \begin{pmatrix} \check{x}_k \\ \Delta f_k \end{pmatrix} \ge 0 \qquad (9)$$

To guarantee the exponential convergence of both the state and the fault estimation errors, persistence of the excitation condition is employed. In this case, we assume there exist positive constants $c_2 \geq c_1 > 0$, such that for $N \in \mathbb{Z}$

$$c_1 \boldsymbol{I}_p \leq \sum_{i=k}^{k+N-1} \boldsymbol{\Phi}_i^\mathsf{T} \boldsymbol{\Phi}_i \leq c_2 \boldsymbol{I}_p, \quad \forall k \tag{10}$$

After defining the nonlinearity, we are ready to design the NLAO. Let us assume all states are measurable, the NLAO can be designed as follows

$$\bar{\boldsymbol{x}}_{k+1} = A\bar{\boldsymbol{x}}_k + \boldsymbol{f}(\bar{\boldsymbol{x}}_k) + \boldsymbol{B}\boldsymbol{u}_k + \boldsymbol{\Phi}_k\bar{\boldsymbol{\theta}}_k + \boldsymbol{J}^{-1}\boldsymbol{Y}\breve{\boldsymbol{y}}_k \tag{11}$$

$$\bar{\boldsymbol{\theta}}_{k+1} = \bar{\boldsymbol{\theta}}_k + \boldsymbol{\Phi}_k^\mathsf{T} \boldsymbol{\Xi}_k \left(\breve{\boldsymbol{x}}_{k+1} - \boldsymbol{K}\breve{\boldsymbol{x}}_k - \Delta \boldsymbol{f}_k \right) \tag{12}$$

where the pair $(\bar{\boldsymbol{x}}_k, \bar{\boldsymbol{\theta}}_k)$ denotes the state and fault estimation from the nonlinear adaptive observer, while $\breve{\boldsymbol{x}}_k = \boldsymbol{x}_k - \bar{\boldsymbol{x}}_k$, $\Delta \boldsymbol{f}_k = \boldsymbol{f}(\boldsymbol{x}_k) - \boldsymbol{f}(\bar{\boldsymbol{x}}_k)$, and $\breve{\boldsymbol{y}}_k = \boldsymbol{y}_k - \bar{\boldsymbol{x}}_k$. The observer gain $\boldsymbol{J}$ and $\boldsymbol{Y}$ are to be determined later. The fault estimation gains are given by

$$\boldsymbol{K} = \boldsymbol{A} - \boldsymbol{J}^{-1}\boldsymbol{Y} \tag{13}$$

$$\boldsymbol{\Xi}_k = 2\left(\boldsymbol{\Phi}_k \boldsymbol{\Phi}_k^\mathsf{T} + \boldsymbol{Q} \right)^{-1} \tag{14}$$

where $\boldsymbol{Q} = \boldsymbol{Q}^\mathsf{T}$ is a positive definite matrix. Thus, $\boldsymbol{\Xi} = \boldsymbol{\Xi}^\mathsf{T} > 0$. Remark that, $\boldsymbol{\Xi}$ is uniformly bounded by $0 < \frac{2}{\Phi^2 + q_b} \leq \boldsymbol{\Xi}(k) \leq \frac{2}{q_s}, \forall k$. Here, q_s denotes the smallest eigenvalue of $\boldsymbol{Q}$, while q_b denotes the largest eigenvalue of $\boldsymbol{Q}$. Let $\tilde{\boldsymbol{\theta}} = \boldsymbol{\theta} - \bar{\boldsymbol{\theta}}$ denotes the fault parameter error. The error dynamics of the state estimation are given by

$$\breve{\boldsymbol{x}}_{k+1} = \boldsymbol{K}\breve{\boldsymbol{x}}_k + \Delta \boldsymbol{f}_k + \boldsymbol{\Phi}_k\tilde{\boldsymbol{\theta}}_k \tag{15}$$

$$\tilde{\boldsymbol{\theta}}_{k+1} = \left(\boldsymbol{I}_p - \boldsymbol{\Phi}_k^\mathsf{T} \boldsymbol{\Xi}_k \boldsymbol{\Phi}_k \right) \tilde{\boldsymbol{\theta}}_k \tag{16}$$

The main result of this section can be stated as follows.

Theorem 1: If there exist a matrix $\boldsymbol{J} = \boldsymbol{J}^\mathsf{T} > 0 \in \mathbb{R}^{n \times n}$ and a matrix $\boldsymbol{Y} \in \mathbb{R}^{n \times n}$ such that the following LMI is satisfied

$$\begin{pmatrix} -\boldsymbol{J} + a_1 \boldsymbol{I}_n & \boldsymbol{L}^\mathsf{T} + a_2 \boldsymbol{I}_n & \boldsymbol{L}^\mathsf{T} \\ \boldsymbol{L} + a_3 \boldsymbol{I}_n & \boldsymbol{J} + a_4 \boldsymbol{I}_n & \boldsymbol{0}_n \\ \boldsymbol{L} & \boldsymbol{0}_n & -\boldsymbol{J} \end{pmatrix} < 0 \tag{17}$$

where $\boldsymbol{L} = \boldsymbol{J}\boldsymbol{A} - \boldsymbol{Y}$, $a_1 = \nu + \epsilon_1\rho + \epsilon_2\beta$, $a_2 = \frac{\eta\epsilon_2 - \epsilon_1}{2}$, $a_3 = \frac{\eta\epsilon_2 - \epsilon_1}{2}$, $a_4 = -\epsilon_2$ for $\epsilon_1, \epsilon_2, \nu > 0$ and $\eta, \rho, \beta \in \mathbb{R}$, then the equilibrium $\left(\breve{\boldsymbol{x}}_k, \tilde{\boldsymbol{\theta}}_k \right) = \boldsymbol{0}$ of the error dynamics (15)-(16) is globally uniformly asymptotically stable.

Proof of Theorem 1 can be obtained by modifying steps in the first author's conference paper [13]. Remark that the adaptive observer (11)-(12) has three tuning matrices: $\boldsymbol{J}$ and $\boldsymbol{Y}$, which can be obtained from Theorem 1, and $\boldsymbol{Q} > 0$.

The main advantage of using the NLAO is that the algorithm is guaranteed to converge to the actual value. Furthermore, the algorithm is easy to implement with minimum computational effort compared to other methods. However, for general nonlinearity the algorithm may not be suitable and the observer gains that satisfy (17) may not be found. Another disadvantage is that the algorithm is not designed to handle process and measurement noise. In many cases,

---

Algorithm 1 Nonlinear Adaptive Observer

Initialization $\bar{\boldsymbol{x}}_0, \bar{\boldsymbol{\theta}}_0$
Determine $\boldsymbol{J}$ and $\boldsymbol{Y}$ based on (17), select $\boldsymbol{Q} = \boldsymbol{Q}^\mathsf{T} > 0$
Recursions for $k = 0, 1, 2, \cdots$
Calculate the fault estimation gains $\boldsymbol{K}$ and $\boldsymbol{\Xi}_k$
based on (13)-(14)
Calculate the vector state $\bar{\boldsymbol{x}}_{k+1}$ and fault $\bar{\boldsymbol{\theta}}_{k+1}$
based on (11)-(12)

it also requires measurement of all state variables, which may not be available. In summary, the fault diagnosis algorithm based on the NLAO is robust with respect to different initial conditions and Lipschitz nonlinearity, but not robust with respect to noise.

IV. ALGORITHM BASED ON ADAPTIVE EXTENDED KALMAN FILTER (AEKF)

In this section, we extend the work of [29] to nonlinear systems. Linearizing (1) at $\hat{\boldsymbol{x}}_k$, we have

$$\boldsymbol{x}_{k+1} = \boldsymbol{F}_k(\hat{\boldsymbol{x}}_k)\boldsymbol{x}_k + \boldsymbol{E}_k(\hat{\boldsymbol{x}}_k) + \boldsymbol{B}\boldsymbol{u}_k + \boldsymbol{\Phi}_k\boldsymbol{\theta} + \boldsymbol{w}_k \tag{18}$$

where

$$\boldsymbol{F}_k(\hat{\boldsymbol{x}}_k) = \boldsymbol{A} + \left. \frac{\partial \boldsymbol{f}(\boldsymbol{x}_k)}{\partial \boldsymbol{x}_k} \right|_{\hat{\boldsymbol{x}}_k} \tag{19}$$

$$\boldsymbol{E}_k(\hat{\boldsymbol{x}}_k) = \boldsymbol{f}(\hat{\boldsymbol{x}}_k) - \left. \frac{\partial \boldsymbol{f}(\boldsymbol{x}_k)}{\partial \boldsymbol{x}_k} \right|_{\hat{\boldsymbol{x}}_k} \hat{\boldsymbol{x}}_k \tag{20}$$

For simplicity, let $\boldsymbol{F}_k = \boldsymbol{F}_k(\hat{\boldsymbol{x}}_k)$ and $\boldsymbol{E}_k = \boldsymbol{E}_k(\hat{\boldsymbol{x}}_k)$. The Kalman gain $\boldsymbol{K}_{k+1}$ and the error covariance matrix $\boldsymbol{P}_{k+1}^+$ are computed using the following recursion

$$\boldsymbol{P}_{k+1}^- = \boldsymbol{F}_k\boldsymbol{P}_k^+\boldsymbol{F}_k^\mathsf{T} + \boldsymbol{Q}_k^F \tag{21}$$

$$\boldsymbol{\Sigma}_{k+1} = \boldsymbol{C}\boldsymbol{P}_{k+1}^-\boldsymbol{C}^\mathsf{T} + \boldsymbol{R}_k^F \tag{22}$$

$$\boldsymbol{K}_{k+1} = \boldsymbol{P}_{k+1}^-\boldsymbol{C}^\mathsf{T}\boldsymbol{\Sigma}_{k+1}^{-1} \tag{23}$$

$$\boldsymbol{P}_{k+1}^+ = [\boldsymbol{I}_n - \boldsymbol{K}_{k+1}\boldsymbol{C}]\boldsymbol{P}_{k+1}^- \tag{24}$$

while the fault estimation gains $\boldsymbol{\Gamma}_{k+1}$ and $\boldsymbol{\Upsilon}_{k+1}$ are computed using the following recursion

$$\boldsymbol{\Upsilon}_{k+1} = (\boldsymbol{I}_n - \boldsymbol{K}_{k+1}\boldsymbol{C})\boldsymbol{F}_k\boldsymbol{\Upsilon}_k + (\boldsymbol{I}_n - \boldsymbol{K}_{k+1}\boldsymbol{C})\boldsymbol{\Phi}_k \tag{25}$$

$$\boldsymbol{\Omega}_{k+1} = \boldsymbol{C}\boldsymbol{F}_k\boldsymbol{\Upsilon}_k + \boldsymbol{C}\boldsymbol{\Phi}_k \tag{26}$$

$$\boldsymbol{\Lambda}_{k+1} = \left[\lambda\boldsymbol{\Sigma}_{k+1} + \boldsymbol{\Omega}_{k+1}\boldsymbol{S}_k\boldsymbol{\Omega}_{k+1}^\mathsf{T} \right]^{-1} \tag{27}$$

$$\boldsymbol{\Gamma}_{k+1} = \boldsymbol{S}_k\boldsymbol{\Omega}_{k+1}^\mathsf{T}\boldsymbol{\Lambda}_{k+1} \tag{28}$$

$$\boldsymbol{S}_{k+1} = \frac{1}{\lambda}\boldsymbol{S}_k - \frac{1}{\lambda}\boldsymbol{S}_k\boldsymbol{\Omega}_{k+1}^\mathsf{T}\boldsymbol{\Lambda}_{k+1}\boldsymbol{\Omega}_{k+1}\boldsymbol{S}_k \tag{29}$$

where λ denotes the forgetting factor, which determines the convergence rate of the fault estimator. The fault parameter and state variable are calculated using the following formula

$$\hat{\boldsymbol{\theta}}_{k+1} = \hat{\boldsymbol{\theta}}_k + \boldsymbol{\Gamma}_{k+1}\tilde{\boldsymbol{y}}_k \tag{30}$$

$$\hat{\boldsymbol{x}}_{k+1} = \boldsymbol{A}\hat{\boldsymbol{x}}_k + \boldsymbol{f}(\boldsymbol{x}_k) + \boldsymbol{B}\boldsymbol{u}_k + \boldsymbol{\Phi}_k\hat{\boldsymbol{\theta}}_k + \boldsymbol{K}_{k+1}\tilde{\boldsymbol{y}}_k + \boldsymbol{\Upsilon}_{k+1}[\hat{\boldsymbol{\theta}}_{k+1} - \hat{\boldsymbol{\theta}}_k] \tag{31}$$

where

$$\tilde{y}_k = y_k - C\hat{x}_k \qquad (32)$$

Algorithm 2 Adaptive Extended Kalman Filter

Initialization $\hat{x}_0$, $\hat{\theta}_0$, P_0^+, Υ_0, S_0, λ

Recursions for $k = 0, 1, 2, \cdots$

  Calculate the Jacobian matrix $\left.\frac{\partial f(x_k)}{\partial x_k}\right|_{\hat{x}_k}$

  Calculate the observer gains Γ_{k+1}, K_{k+1}, Υ_{k+1}
  based on (21)-(29)

  Calculate the fault $\hat{\theta}_{k+1}$ and the vector state $\hat{x}_{k+1}$
  based on (30)-(31)

The main advantage of using the AEKF is the method applicable for general nonlinearity, once the Jacobian matrix has been computed. Furthermore, the method can handle process and measurement noise. However, due to linearization, the estimates may not converge to the actual values. This limitation is addressed in the third algorithm below, where we add an NLO to guarantee the stability. In summary, the fault diagnosis algorithm based on the AEKF is robust with respect to noise, but not robust with respect to severe nonlinearity.

V. ALGORITHM BASED ON ADAPTIVE EXOGENOUS KALMAN FILTER (AXKF)

The third algorithm for fault diagnosis is based on the AXKF. The idea of the AXKF is to cascade an NLO and an LAKF. The NLO is used to guarantee the convergence of the state variable estimate to the actual value, while the LAKF is used to estimate the magnitude of the fault and to handle the process and measurement noise.

Let us assume there are m measurements, such that $C \in \mathbb{R}^{m \times n}$ and $y_k = Cx_k$. The NLO is designed as follow

$$\bar{x}_{k+1} = A\bar{x}_k + f(\bar{x}_k) + Bu_k + \Phi_k\hat{\theta}_k + J^{-1}Y\check{y}_k \quad (33)$$

Remark that in (33), the vector parameter fault is obtained from the LAKF at k. This is to simplify the analysis since estimating the vector parameter fault θ using a nonlinear adaptive observer is very challenging. The error dynamic is given by

$$\check{x}_{k+1} = \left(A - J^{-1}YC\right)\check{x}_k + \Delta f_k + \Phi_k\check{\theta}_k \quad (34)$$

Here, the NLO is used to estimate only the state variables. The fault parameter θ_k is estimated using the LAKF in the second stage. Following the NLAO design in Section 3, we have the following result.

Theorem 2: If there exist a matrix $J = J^\top > 0 \in \mathbb{R}^{n \times n}$ and a matrix $Y \in \mathbb{R}^{n \times m}$ such that

$$\begin{pmatrix} -J + a_1 I_n & H^\top + a_2 I_n & H^\top \\ H + a_3 I_n & J + a_4 I_n & 0_n \\ H & 0_n & -J \end{pmatrix} < 0 \quad (35)$$

where $H = JA - YC$, $a_1 = \epsilon_1\rho + \epsilon_2\beta$, $a_2 = \frac{\eta\epsilon_2 - \epsilon_1}{2}$, $a_3 = \frac{\eta\epsilon_2 - \epsilon_1}{2}$, $a_4 = -\epsilon_2$ for $\epsilon_1, \epsilon_2 > 0$ and $\eta, \rho, \beta \in \mathbb{R}$,

then the equilibrium $\left(\check{x}_k, \check{\theta}_k\right) = 0$ of the error dynamic (34) is globally uniformly asymptotically stable.

Proof of Theorem 2 can be obtained by modifying the steps in the first author's conference paper [12]. The difference between Theorem 1 and Theorem 2 is that the dimension of the measurement matrix C and the constant a_1. In this case, full state measurement is not required for the AXKF algorithm.

For the LAKF, let us linearize the nonlinear term $f(\bar{x}_k)$ about the state estimation from the observer $\bar{x}_k$. Linearizing (1) at $\bar{x}_k$, we have

$$x_{k+1} = F_k(\bar{x}_k)x_k + E_k(\bar{x}_k) + Bu_k + \Phi_k\theta + w_k \quad (36)$$

where

$$F_k(\bar{x}_k) = A + \left.\frac{\partial f(x_k)}{\partial x_k}\right|_{\bar{x}_k} \qquad (37)$$

$$E_k(\bar{x}_k) = f(\bar{x}_k) - \left.\frac{\partial f(x_k)}{\partial x_k}\right|_{\bar{x}_k}\bar{x}_k \qquad (38)$$

For simplicity, let $F_k = F_k(\bar{x}_k)$ and $E_k = E_k(\bar{x}_k)$. In this case, the fault and state are calculated using the following formula

$$\hat{\theta}_{k+1} = \hat{\theta}_k + \Gamma_{k+1}\tilde{y}_k \qquad (39)$$

$$\hat{x}_{k+1} = F_k\hat{x}_k + E_k + Bu_k + \Phi_k\hat{\theta}_k$$
$$+ K_{k+1}\tilde{y}_k + \Upsilon_{k+1}[\hat{\theta}_{k+1} - \hat{\theta}_k] \qquad (40)$$

Algorithm 3 Adaptive Exogenous Kalman Filter

Initialization $\bar{x}_0$, $\bar{\theta}_0$, $\hat{x}_0$, $\hat{\theta}_0$, P_0^+, Υ_0, S_0, λ

Determine J and Y based on (35), select $Q = Q^\top > 0$

Recursions for $k = 0, 1, 2, \cdots$

First stage estimation:

  Calculate the vector state $\bar{x}_{k+1}$ based on (33)

Second stage estimation:

  Calculate the Jacobian matrix $\left.\frac{\partial f(x_k)}{\partial x_k}\right|_{\bar{x}_k}$

  Calculate the observer gains Γ_{k+1}, K_{k+1}, Υ_{k+1}
  based on (21)-(29)

  Calculate the fault $\hat{\theta}_{k+1}$ and the vector state $\hat{x}_{k+1}$
  based on (39)-(40)

The advantage of using this two-stage estimation is that the algorithm is stable and can handle the process and measurement noise. However, similar to the NLAO, the observer gains may be difficult to find. In summary, the fault diagnosis algorithm based on the AXKF is both robust with respect to nonlinearity and noise.

VI. NUMERICAL SIMULATIONS

To test and compare the algorithms, we perform numerical simulations based on a single-link flexible joint model. The aims of the numerical simulations are threefold:

- To test the convergence rate and the transient response of the algorithms

- To compare the performance of the fault estimation algorithms
- To investigate the effect of the forgetting factor λ

The dynamics of the robot can be described by using the following first-order nonlinear ordinary differential equations

$$\dot{\phi}_m = \omega_m \tag{41}$$

$$\dot{\omega}_m = \frac{k}{J_m}(\phi_l - \phi_m) - \frac{B}{J_m}\omega_m + \frac{K_\tau}{J_m}(1-\theta)u \tag{42}$$

$$\dot{\phi}_l = \omega_l \tag{43}$$

$$\dot{\omega}_l = -\frac{k}{J_l}(\phi_l - \phi_m) - \frac{mgh}{J_l}\sin(\phi_l) \tag{44}$$

where ϕ_m and ϕ_l denote the rotation angles of the motor and the link, while ω_m and ω_l denote the angular velocities of the motor and the link, respectively.

Remark that the only nonlinearity is the sinusoidal function in (44), which is a Lipschitz continuous function. All parameters (k, J_m, B, K_τ, J_l, m, g, and h) for this simulation are taken from [30]. Applying the Euler method and taking the sampling time $\Delta t = 0.001$s to ensure the numerical scheme is stable and capture the transient response, the system resembles (1)-(2) with

$$A = \begin{pmatrix} 1 & 0.001 & 0 & 0 \\ -0.0486 & 0.9988 & 0.0486 & 0 \\ 0 & 0 & 1 & 0.001 \\ 0.0195 & 0 & -0.0195 & 1 \end{pmatrix}$$

$$B = \begin{pmatrix} 0 & 0.0216 & 0 & 0 \end{pmatrix}^\mathsf{T}$$

$$f(x_k) = \begin{pmatrix} 0 & 0 & 0 & -0.00333\sin(\phi_{l,k}) \end{pmatrix}^\mathsf{T}$$

The fault transition matrix is given by

$$\Phi_k = -\begin{pmatrix} 0 & 0.0216 & 0 & 0 \end{pmatrix}^\mathsf{T} u_k \tag{45}$$

For the NLAO-based algorithm, first we choose $Q = 0.0001I$. Solving the linear matrix inequality (17) for J and Y with $\nu = 0.0001$, $\epsilon_1 = 1$, and $\epsilon_2 = 100$, we obtain

$$J = \begin{pmatrix} 46.2 & 0.5 & -0.5 & 0.05 \\ 0.5 & 22.27 & -11.34 & 0.1725 \\ -0.5 & -11.34 & 27.28 & 0.19 \\ 0.05 & 0.17 & 0.19 & 0.5075 \end{pmatrix} \tag{46}$$

and

$$Y = \begin{pmatrix} 16.96 & 0.17 & 0 & 0.002 \\ -8.63 & 16.96 & 0 & 0.002 \\ 4.11 & -1.57 & 20 & 0.002 \\ 0.012 & -0.05 & 0 & 0.01 \end{pmatrix} \tag{47}$$

For the AEKF-based algorithm, we choose $S_0 = 0.01$ and $\Upsilon_0 = 0$. The covariance matrices Q_k^F and R_k^F are updated using the following formulas [31]

$$Q_{k+1}^F = aQ_k^F + (1-a)(K_k\tilde{y}_k\tilde{y}_k^\mathsf{T}K_k) \tag{48}$$

$$R_{k+1}^F = aR_k^F + (1-a)(\tilde{y}_k\tilde{y}_k^\mathsf{T} + CP_k^+C^\mathsf{T}) \tag{49}$$
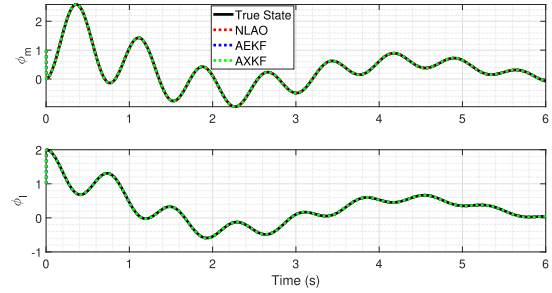

FIGURE 3. Comparison of the state estimation of ϕ_m and ϕ_l from the three algorithms.
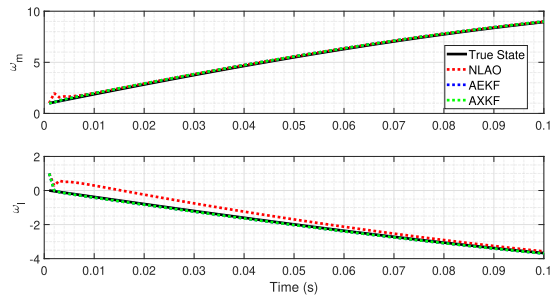

FIGURE 4. Comparison of the transient response from the three algorithms.

where $0 < a \leq 1$. For the AXKF-based algorithm, the procedure to find the parameters is similar to the two aforementioned methods. In this case, solving the (35) we obtain

$$J = \begin{pmatrix} 0.0295 & 0 & 0 & 0 \\ 0 & 0.0756 & 0.0402 & 0.1253 \\ 0 & 0.0402 & 0.0835 & 0.3002 \\ 0 & 0.1253 & 0.3002 & 3.5489 \end{pmatrix} \tag{50}$$

and

$$Y = \begin{pmatrix} 0.7500 & -0.0015 & 0 & 0 \\ 0.1944 & 0.3934 & 0.0486 & 0 \\ 0.0000 & -0.2663 & 1 & 0.001 \\ -0.0777 & -0.7336 & -0.0195 & 1 \end{pmatrix} \tag{51}$$

The simulation is running for 6s, in which the fault is introduced after 2s. We assume the fault is constant. The result of the state estimation for the motor and link angle can be seen from Fig. 3. Here, the three algorithms converge to the actual values. The transient responses of the three algorithms are presented in Fig. 4. Since the nonlinearity is not severe, the estimation from the AEKF performs well. The convergence rate for the NLAO can be increased by choosing different values for J and Y.

Table 1 shows the principal characteristics of the three algorithms, which include the accuracy through the Root Mean Square Error (RMSE), computational time, and computational complexity. We note that the computational complexity of the AEKF and the AXKF is $\mathcal{O}(n^3)$ since the calculation involves 3 matrix multiplication.

From Table 1, we can observe that the AXKF provides better accuracy but takes more computational effort. The

TABLE 1. Performance indicators of the algorithms.

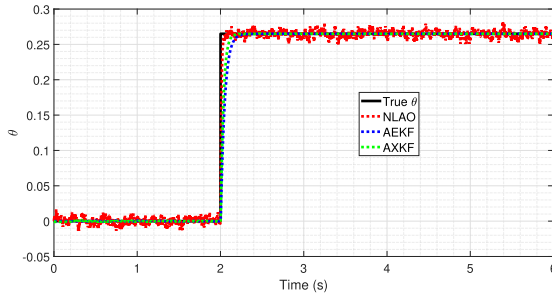| Algorithm	RMSE	comp. time (s)	comp. complexity
NLAO	0.587	0.875	$\mathcal{O}(n)$
AEKF	0.198	1.8438	$\mathcal{O}(n^3)$
AXKF	0.0573	2.1562	$\mathcal{O}(n^3)$

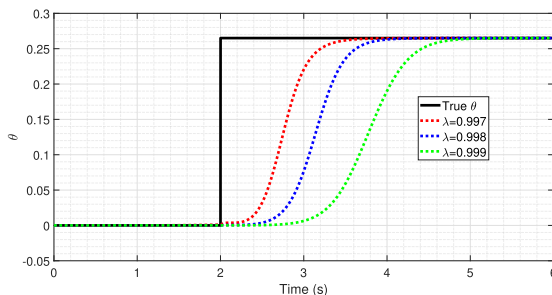
FIGURE 5. Fault estimation from the three algorithms.

FIGURE 6. Comparison of different forgetting factors λ in the AXKF algorithm.

actuator fault estimation from the three algorithms can be seen in Fig. 5. It can be observed that the AXKF algorithm converges faster than the AEKF algorithm. On the other hand, the AXKF algorithm is also better than the NLAO in terms of noise reduction. This result can be seen as a compromise solution between the convergence rate and the noise reduction. The convergence rate of the AXKF can be adjusted by changing the forgetting factor λ, as can be seen from Fig. 6. Smaller λ causes the algorithm to converge faster. However, choosing λ too small can cause the estimation result to be noisy.

VII. EXPERIMENTS IN A BALL-BALANCING ROBOT

The ball-balancing robot has four DC motors powered by a separate power supply consisting of 6 AA batteries connected in series and is designed to move in the x and y direction, as can be seen in Fig. 8 (left). The robot is completed with four omni-wheels which are positioned in-line, each actuated with its own motor. This design simplifies the control problem and makes sure that the system will always be supported on a driving wheel. A motor driver shield expansion board is used to interface the motors with the Arduino microcontroller. The motor driver consists of 2 L293D, a 4 half H-bridge driver, which acts as four full H-bridges, allowing up to
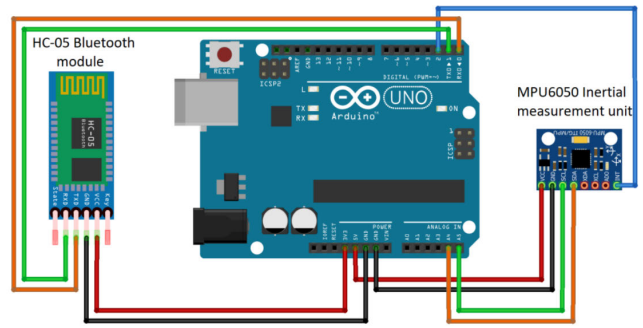

FIGURE 7. Circuit diagram of the IMU and Bluetooth.

four DC-motors to run in both directions. These drivers are connected to 8-bit PWM I/O pins of the microcontroller. All communication and debugging were done using a wireless connection. In this case, we use an HC 05 Bluetooth master and slave module (Fig. 7). The algorithms are then written in the microcontroller using the Arduino Integrated Development Environment (IDE).

The dynamics of the ball-balancing robot can be modelled as an inverted pendulum, as can be seen from Fig. 8 (left), where the vector diagram is presented in Fig. 8 (right). The configuration of the motors can be seen from Fig. 9. If we denote $x_1 = \psi$, $x_2 = \dot{x}_1$, $x_3 = x$, and $x_4 = \dot{x}_3$, then the dynamic model for the ball-balancing robot in the x direction is given by

$$\dot{x}_1 = x_2 \tag{52}$$

$$\dot{x}_2 = \frac{(M+m)g\sin(x_1) - u}{M} \tag{53}$$

$$\dot{x}_3 = x_4 \tag{54}$$

$$\dot{x}_4 = \frac{-mg\sin(x_1) + u}{M+m} \tag{55}$$

where u is the motor torque, which can be adjusted by controlling the current. Discretizing the above equation using Euler's method, the system resembles (1)-(2) with

$$A = \begin{pmatrix} 1 & 0.001 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0.001 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

$$f(x_k) = \begin{pmatrix} 0 \\ \dfrac{(M+m)g\sin(x_{1,k})}{M} \\ 0 \\ \dfrac{-mg\sin(x_{1,k})}{M+m} \end{pmatrix}, \quad B = \begin{pmatrix} 0 \\ -\dfrac{1}{M} \\ 0 \\ \dfrac{1}{M+m} \end{pmatrix}$$

In this experiment, the mass of the robot (as shown in Figure 9) is $M = 1.94$ Kg, while the mass of the pendulum (the structure above the robot) is $m = 0.81$ Kg. Furthermore, we assume a fault has occurred in the actuator system and can be modelled as a drastic gain loss θ. The experiment is performed for 40s and the fault, which is introduced into the ball-balancing robot as a gain loss of 10%, occurs after 20s. An MPU6050 Inertial Measurement Unit (IMU) sensor
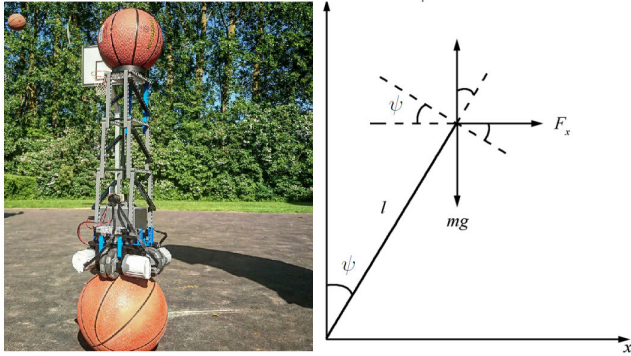
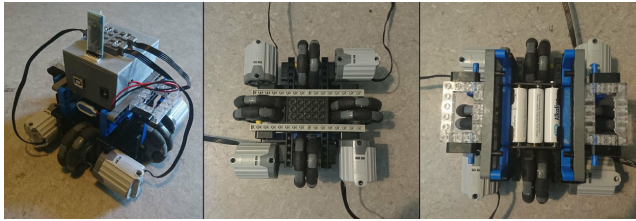**FIGURE 8.** SDU ball-balancing robot (left) and simplified vector diagram in *x* direction (right).

FIGURE 9. Motor configuration.

is used to measure the angle ψ and the position x of the robot. The sensor combines a microelectromechanical systems gyroscope and accelerometer and uses a standard I2C bus for data communication. Since we only able to measure two variables associated with the angle ψ and the position x, we have

$$C = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix} \quad (56)$$

The fault is estimated using the three algorithms. To this end, for the AXKF the observer gains are selected as follow

$$J = \begin{pmatrix} 2 & 0 & 0 & 0 \\ 0 & 0.001 & 0 & 0 \\ 0 & 0 & 2 & 0 \\ 0 & 0 & 0 & 0.001 \end{pmatrix}, \quad Y = \begin{pmatrix} 2 & 0 \\ 0 & 0 \\ 0 & 2 \\ 0 & 0 \end{pmatrix} \quad (57)$$

In the second stage estimation using the LAKF, the Jacobian matrix is given by

$$F(\bar{x}_k) = \begin{pmatrix} 0 & 0 & 0 & 0 \\ \dfrac{(M+m)g\cos(\bar{x}_{1,k})}{M} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ \dfrac{-mg\cos(\bar{x}_{1,k})}{M+m} & 0 & 0 & 0 \end{pmatrix} \quad (58)$$

The position in *xy*-plane from the three algorithms can be seen in Fig. 10, while the fault parameter can be seen in Fig. 11. The ground truth is determined by optical tracking. It can be observed that the AXKF algorithm performs better than the NLAO and the AEKF. In this experiment, we show that the AXKF can be used to estimate the magnitude of the actuator fault accurately with guaranteed stability.
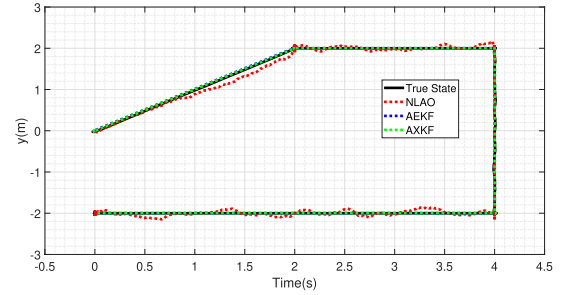

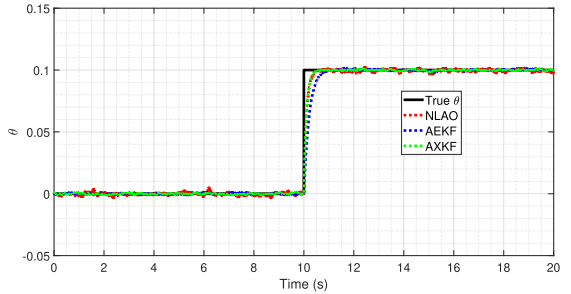**FIGURE 10.** Position of the ball-balancing robot in *xy*-plane.

FIGURE 11. Fault parameter.

VIII. CONCLUSION

In this paper, we present three model-based algorithms that can be used to estimate the magnitude of the fault parameters. The algorithms are based on the NLAO, the AEKF, and the AXKF, where the latter can be considered as a compromise solution to the former. The main strength of the NLAO is its stability, while the main strength of the AEKF is its ability to handle noise. We show all algorithms are able to estimate the fault magnitude accurately. Limitations of the algorithm include: (i) the model relies on accurate representation of the system, which may not be easy to find, (ii) the LMIs (17) and (35) may not have solution, and (iii) the algorithms only work for constant/piecewise constant fault. The information of the fault magnitude can be used in fault-tolerant control algorithms, e.g., using model predictive control (MPC). Future works include development of fault-tolerant MPC considering time delay on the measurement.

REFERENCES

[1] A. Hasan and T. A. Johansen, "Model-based actuator fault diagnosis in multirotor UAVs," in *Proc. Int. Conf. Unmanned Aircr. Syst. (ICUAS)*, Jun. 2018, pp. 1017–1024.

[2] J. Lan and R. J. Patton, "A new strategy for integration of fault estimation within fault-tolerant control," *Automatica*, vol. 69, pp. 48–59, Jul. 2016.

[3] Y. Hou, A. Yang, W. Guo, E. Zheng, Q. Xiao, Z. Guo, and Z. Huang, "Bearing fault diagnosis under small data set condition: A Bayesian network method with transfer learning for parameter estimation," *IEEE Access*, vol. 10, pp. 35768–35783, 2022.

[4] H. Cui, Y. Guan, and H. Chen, "Rolling element fault diagnosis based on VMD and sensitivity MCKD," *IEEE Access*, vol. 9, pp. 120297–120308, 2021.

[5] A. Mouzakitis, "Classification of fault diagnosis methods for control systems," *Meas. Control*, vol. 46, no. 10, pp. 303–308, Dec. 2013.

[6] B. Safarinejadian, P. Ghane, and H. Monirvaghefi, "Fault detection in nonlinear systems based on type-2 fuzzy logic," *Int. J. Syst. Sci.*, vol. 46, no. 3, pp. 394–404, Feb. 2015.

[7] X. Liu, X. Lu, and Z. Gao, "A deep learning-based fault diagnosis of leader-following systems," *IEEE Access*, vol. 10, pp. 18695–18706, 2022.

[8] M. Mansouri, M. Trabelsi, H. Nounou, and M. Nounou, "Deep learning-based fault diagnosis of photovoltaic systems: A comprehensive review and enhancement prospects," *IEEE Access*, vol. 9, pp. 126286–126306, 2021.

[9] H. Zhao, S. Sun, and B. Jin, "Sequential fault diagnosis based on LSTM neural network," *IEEE Access*, vol. 6, pp. 12929–12939, 2018.

[10] W. Tuerxun, X. Chang, G. Hongyu, J. Zhijie, and Z. Huajian, "Fault diagnosis of wind turbines based on a support vector machine optimized by the sparrow search algorithm," *IEEE Access*, vol. 9, pp. 69307–69315, 2021.

[11] R. Isermann, *Fault-Diagnosis Systems: An Introduction From Fault Detection to Fault Tolerance*. Cham, Switzerland: Springer, 2006.

[12] A. Hasan, "Adaptive eXogenous Kalman filter for actuator fault diagnosis in robotics and autonomous systems," in *Proc. 7th Int. Conf. Control, Mechatronics Autom. (ICCMA)*, Nov. 2019, pp. 162–167.

[13] A. Hasan, "EXogenous Kalman filter for state estimation in autonomous ball balancing robots," in *Proc. IEEE/ASME Int. Conf. Adv. Intell. Mechatronics (AIM)*, Jul. 2020, pp. 1522–1527.

[14] A. Kiselev, G. R. Catuogno, A. Kuznietsov, and R. Leidhold, "Finite-control-set MPC for open-phase fault-tolerant control of PM synchronous motor drives," *IEEE Trans. Ind. Electron.*, vol. 67, no. 6, pp. 4444–4452, Jun. 2020.

[15] R. E. Kalman, "A new approach to linear filtering and prediction problems," *J. Fluids Eng.*, vol. 82, no. 1, pp. 35–45, Mar. 1960.

[16] L. Ljung, "Asymptotic behavior of the extended Kalman filter as a parameter estimator for linear systems," *IEEE Trans. Autom. Control*, vol. AC-24, no. 1, pp. 36–50, Feb. 1979.

[17] S. Julier, J. Uhlmann, and H. F. Durrant-Whyte, "A new method for the nonlinear transformation of means and covariances in filters and estimators," *IEEE Trans. Autom. Control*, vol. 45, no. 3, pp. 477–482, Mar. 2000.

[18] S. J. Julier and J. K. Uhlmann, "Unscented filtering and nonlinear estimation," *Proc. IEEE*, vol. 92, no. 3, pp. 401–422, Mar. 2004.

[19] B. Safarinejadian and E. Kowsari, "Fault detection in non-linear systems based on GP-EKF and GP-UKF algorithms," *Syst. Sci. Control Eng.*, vol. 2, no. 1, pp. 610–620, Dec. 2014.

[20] G. Besançon, *Nonlinear Observers and Applications*, vol. 363. Cham, Switzerland: Springer, 2007.

[21] H. Nijmeijer and T. I. Fossen, *New Directions in Nonlinear Observer Design*, vol. 244. Berlin, Germany: Springer, 1999.

[22] T. A. Johansen and T. I. Fossen, "The eXogenous Kalman filter (XKF)," *Int. J. Control*, vol. 90, no. 2, pp. 161–167, Feb. 2017.

[23] T. Wang, S. Chen, H. Ren, and Y. Zhao, "Model-based unscented Kalman filter observer design for lithium-ion battery state of charge estimation," *Int. J. Energy Res.*, vol. 42, no. 4, pp. 1603–1614, Mar. 2018.

[24] M. Ye, H. Guo, R. Xiong, and R. Yang, "Model-based state-of-charge estimation approach of the lithium-ion battery using an improved adaptive particle filter," *Energy Proc.*, vol. 103, pp. 394–399, Dec. 2016.

[25] V. Gudmundsson, H. Kristinsson, S. Petersen, and A. Hasan, "Robust UAV attitude estimation using a cascade of nonlinear observer and linearized Kalman filter," in *Proc. Dyn. Syst. Control Conf.*, Sep. 2018, pp. 1–10.

[26] A. A. Ravankar, A. Ravankar, Y. Kobayashi, and T. Emaru, "Autonomous mapping and exploration with unmanned aerial vehicles using low cost sensors," *Proceedings*, vol. 4, no. 1, p. 44, 2018.

[27] S. Kuindersma, R. Deits, M. Fallon, A. Valenzuela, H. Dai, F. Permenter, T. Koolen, P. Marion, and R. Tedrake, "Optimization-based locomotion planning, estimation, and control design for the atlas humanoid robot," *Auton. Robots*, vol. 40, no. 3, pp. 429–455, Mar. 2016.

[28] Y. Shoukry, P. Nuzzo, A. Puggelli, A. L. Sangiovanni-Vincentelli, S. A. Seshia, and P. Tabuada, "Secure state estimation for cyber-physical systems under sensor attacks: A satisfiability modulo theory approach," *IEEE Trans. Autom. Control*, vol. 62, no. 10, pp. 4917–4932, Oct. 2017.

[29] Q. Zhang, "Adaptive Kalman filter for actuator fault diagnosis," *Automatica*, vol. 93, pp. 333–342, Jul. 2018.

[30] S. Ibrir, W. F. Xie, and C.-Y. Su, "Observer-based control of discrete-time Lipschitzian non-linear systems: Application to one-link flexible joint robot," *Int. J. Control*, vol. 78, no. 6, pp. 385–395, Apr. 2005.

[31] S. Akhlaghi, N. Zhou, and Z. Huang, "Adaptive adjustment of noise covariance in Kalman filter for dynamic state estimation," in *Proc. IEEE Power Energy Soc. Gen. Meeting*, Chicago, IL, USA, Jul. 2017, pp. 1–5.

AGUS HASAN (Senior Member, IEEE) received the B.Sc. degree in mathematics from the Department of Mathematics, Bandung Institute of Technology, and the Ph.D. degree in control systems from the Department of Cybernetics Engineering, Norwegian University of Science and Technology (NTNU). He is a Professor of cyber-physical systems with the Department of ICT and Natural Sciences, NTNU. His research interests include cyber-physical systems, system dynamics, digital twin, and autonomous systems. He is an IEEE Technical Committee Member on Aerial Robotics and Unmanned Aerial Vehicles and an IFAC Technical Committee Member on Distributed Parameter Systems.
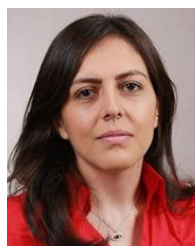
MARYAMSADAT TAHAVORI (Member, IEEE) received the Ph.D. degree from Aalborg University, Denmark, in 2014. She was a Visiting Researcher at Imperial College London and an Assistant Professor at the Center for Energy Informatics and SDU UAS Centre, University of Southern Denmark, from 2014 to 2020. She is currently an Associate Professor with the DTU Engineering Technology, Technical University of Denmark. She has worked on many projects (both EU and national projects). Her research interests include mathematical modeling, optimization, control, and data analysis.

HENRIK SKOV MIDTIBY (Member, IEEE) received the Ph.D. degree in robotics from the University of Southern Denmark, in 2012. He is an Associate Professor with the SDU UAS Center, University of Southern Denmark. He works with computer vision and has a special interest in interpreting images and videos from biological domains acquired by UAVs.

• • •