2022 IEEE 61st Conference on Decision and Control (CDC)
December 6-9, 2022. Cancún, Mexico

A Nonlinear Observer for a Flexible Robot Arm and its Use in Fault and Collision Detection

Claudio Gaz*,[1], Andrea Cristofaro[2], Pasquale Palumbo[3], Alessandro De Luca[2]

***Abstract*— A nonlinear observer is presented for the estimation of the joint velocities and of the deformation modes and their rates for a two-link flexible robot arm, using only motor encoders and a tip position sensor. The state observer can be used then for trajectory tracking control. By monitoring the mismatch between measured and estimated outputs, we can also detect the occurrence of permanent or intermittent system anomalies such as actuator faults or link collisions.**

I. Introduction

Mechanical systems with flexible elements have an increasing interest in applications such as aerospace, biomedical, and robotics where lightweight, compliant, and adaptable structures are desirable [1]. Lighter robots with elastic components are more human-friendly, increase operational safety against undesired collisions, allow a more efficient use of actuators for achieving high-speed response with low-energy burden, and provide adaptive characteristics in grasping tasks [2]. For these reasons, many control problems have been investigated over the years for manipulators with flexible links, see e.g., [3], [4], [5], [6]. In the face of its desirable features, distributed flexibility introduces a larger complexity in the mathematical model of the system, requiring additional state variables to describe the vibration dynamics and its interplay with the rigid motion. In a tradeoff between model accuracy and manageability for control design, two main finite-dimensional approximate modeling approaches are used to deal with the infinite-dimensional nature of a flexible link, either based on a finite-element decomposition of the distributed deformation [7] or by resorting to a finite number of assumed deformation modes [8].

In the typical setup of rigid robots, the joint angles are the only variables directly measured by the encoders mounted on the motors. The joint velocities that are needed in most control laws (from basic PD to feedback linearization) are either computed by numerical differentiation of the position measurements, or estimated by means of suitable state observers. Various solutions have been proposed for observing the full state of rigid robots. In [9], a nonlinear observer with a rather simple structure was proposed, providing a good, yet local, approximation of the robot state under the assumption of bounded velocity. In a similar setting, [10] designed a Luenberger-like observer, passivity arguments were investigated in [11], while sliding observers have been considered by [12]. On the other hand, the literature on state observers for flexible manipulators is more fragmentary. For a one-link flexible arm, infinite-dimensional observers have been developed in [13], [14], while [15] proposed an observer based on a finite-dimensional model.

In our recent work [16], we have considered the problem of fault and collision detection for robots with flexible links, by extending the state-of-the-art momentum-based residual method of [17], [18] that was originally proposed for rigid robots or for robots with elasticity concentrated at the joints. With this approach, we have shown that it is possible to detect and isolate possible actuator faults as well as accidental link collisions and even to discriminate between these two classes of events. A major drawback of our method, however, is that it relies on the knowledge of the full robot state, which is a strong requirement for flexible link robots.

In this work, the problem of state observation is investigated for a two-link planar manipulator with a flexible forearm (the *Flexarm* in [19]) using a finite-dimensional model. Inspired by [20], we design a nonlinear observer that estimates the joint velocities as well as all the deformation modes of the forearm and their derivatives, using only the two motor encoders at the joints and a tip position sensor. With the estimated state, one can successfully implement a trajectory tracking feedback control law. Further, the proposed observer enables to define monitoring signals that allow the detection of actuator faults and link collisions.

The rest of the paper is organized as follows. Section II recalls the dynamic model of the considered flexible manipulator. The design principle of the nonlinear state observer is described in Sec. III, while its implementation for the *Flexarm* is presented in Sec. IV. Section V reports simulation results on observer-based control and detection that support the validity of the proposed methods. Finally, conclusions are drawn in Sec. VI.

II. Dynamic Model of the Flexarm

The *Flexarm* in Fig. 1 is a two-link lightweight planar manipulator with two revolute joints, a first rigid link of length $\ell_1 = 0.3$ m, and a flexible forearm of $\ell_2 = 0.7$ m that bends only in the horizontal plane of motion [19].

The arm is actuated by two DC motors in direct-drive mode. The incremental encoders mounted on the motors measure the angle θ_1 of the rigid link and the (clamped) angle θ_{c2} of the second link. An optical sensor mounted at the base the forearm measures its deflection angle y_{tip} at the tip. Although simple, this platform is a good benchmark for the control of flexible structures since it includes already relevant nonlinear and coupling dynamic effects.

The forearm link is modeled as an Euler-Bernoulli beam of length ℓ_2, uniform density and constant elastic properties.

* Corresponding author.
[1] Faculty of Science, Engineering and Computing, Department of Mechanical Engineering, Kingston University, London, UK. Email: c.gaz@kingston.ac.uk
[2] Department of Computer, Control and Management Engineering, Sapienza University of Rome, Via Ariosto 25, 00185 Rome, Italy. Email: {cristofaro,deluca}@diag.uniroma1.it
[3] Department of Biotechnology and Biosciences, University of Milano Bicocca, Milan, Italy. Email: pasquale.palumbo@unimib.it.

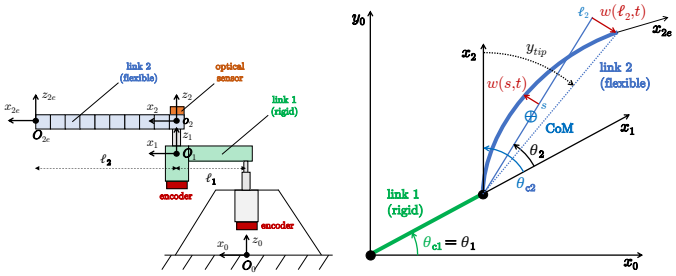

Fig. 1. Sketch of the *Flexarm* (left) and its angles and deflection (right).

With reference to Fig. 1, for a generic link point $s \in [0, \ell_2]$, $w(s,t)$ is the bending link deflection from the axis passing through the joint axis 2 and the Center of Mass (CoM) of the forearm, while θ_2 is the (rigid mode) angle between this same axis and the axis of the first link. The bending deflection of the forearm is expressed by the n-th order approximation

$$w(s,t) = \sum_{i=1}^{n} \phi_i(s)\delta_i(t), \tag{1}$$

with the deformation variables $\delta_i(t)$ associated to the mode shapes $\phi_i(s)$ (the eigenfunctions satisfying suitable dynamic boundary conditions). For further details, see [8], [19].

Following the Euler-Lagrange formulation, the dynamic model of our flexible manipulator takes the form

$$\boldsymbol{M}(\boldsymbol{q})\ddot{\boldsymbol{q}} + \boldsymbol{c}(\boldsymbol{q},\dot{\boldsymbol{q}}) + \boldsymbol{K}\boldsymbol{q} + \boldsymbol{D}\dot{\boldsymbol{q}} = \boldsymbol{G}\boldsymbol{u}, \tag{2}$$

where $\boldsymbol{q} = \begin{pmatrix} \boldsymbol{\theta}^T & \boldsymbol{\delta}^T \end{pmatrix}^T = \begin{pmatrix} \theta_1 & \theta_2 & \delta_1 & \dots & \delta_n \end{pmatrix}^T \in \mathbb{R}^{n+2}$ are the generalized coordinates, $\boldsymbol{M}(\boldsymbol{q}) > 0$ is the inertia matrix, $\boldsymbol{c}(\boldsymbol{q},\dot{\boldsymbol{q}})$ contains the Coriolis and centrifugal terms, $\boldsymbol{K} \geq 0$ and $\boldsymbol{D} \geq 0$ are the stiffness and damping (diagonal) matrices of the system, while the input matrix $\boldsymbol{G}$ transforms the motor torques $\boldsymbol{u} \in \mathbb{R}^2$ into generalized forces performing work on $\boldsymbol{q}$. Gravity terms have no influence on motion on the horizontal plane. Also, friction at the joints is very small due to the direct-drive actuation and can thus be neglected.

Let the constant terms ϕ_{ie} and ϕ'_{i0} associated to the mode shapes in (1) be defined as

$$\phi_{ie} = \phi_i(s)|_{s=\ell_2}, \quad \phi'_{i0} = \left.\frac{\partial \phi_i(s)}{\partial s}\right|_{s=0}, \quad i = 1,\dots,n. \tag{3}$$

Accordingly, the input matrix $\boldsymbol{G}$ takes the form

$$\boldsymbol{G} = \begin{pmatrix} \boldsymbol{I}_{2\times 2} \\ \boldsymbol{G}_\delta \end{pmatrix}, \quad \boldsymbol{G}_\delta = \begin{pmatrix} 0 & 0 & \dots & 0 \\ \phi'_{10} & \phi'_{20} & \dots & \phi'_{n0} \end{pmatrix}^T,$$

with $\boldsymbol{G}_\delta \in \mathbb{R}^{n\times 2}$. The stiffness matrix $\boldsymbol{K}$ is

$$\boldsymbol{K} = \begin{pmatrix} \boldsymbol{0}_{2\times 2} & \boldsymbol{0}_{2\times n} \\ \boldsymbol{0}_{n\times 2} & \boldsymbol{K}_\delta \end{pmatrix}, \quad \boldsymbol{K}_\delta = \mathrm{diag}\begin{pmatrix} \omega_1^2 & \dots & \omega_n^2 \end{pmatrix},$$

where $\boldsymbol{K}_\delta \in \mathbb{R}^{n\times n}$ contains the angular eigenfrequencies ω_i, $i = 1,\dots,n$, of the flexible forearm, while the modal damping matrix $\boldsymbol{D}$ is

$$\boldsymbol{D} = \begin{pmatrix} \boldsymbol{0}_{2\times 2} & \boldsymbol{0}_{2\times n} \\ \boldsymbol{0}_{n\times 2} & \boldsymbol{D}_\delta \end{pmatrix}, \quad \boldsymbol{D}_\delta = \mathrm{diag}\begin{pmatrix} 2\zeta_1\omega_1 & \dots & 2\zeta_n\omega_n \end{pmatrix}.$$

The outputs of the robotic system that are relevant for task definition and for control design are the two motor positions θ_{c1} and θ_{c2}, as measured by the encoders, and the tip deflection y_{tip} of the forearm, as measured by the available optical sensor. These quantities can be expressed as linear combinations of the components of $\boldsymbol{q}$, namely as

$$\boldsymbol{\theta}_c = \begin{pmatrix} \theta_{c1} \\ \theta_{c2} \end{pmatrix} = \begin{pmatrix} \theta_1 \\ \theta_2 + \sum_{i=1}^{n} \phi'_{i0}\delta_i \end{pmatrix} \tag{4}$$

and, using the definitions in (3),

$$y_{tip} = \frac{w(\ell_2,t)}{\ell_2} + \theta_2 - \theta_{c2} = \sum_{i=1}^{n}\left(\frac{\phi_{ie}}{\ell_2} - \phi'_{i0}\right)\delta_i. \tag{5}$$

Setting $\boldsymbol{x} = \begin{pmatrix} \boldsymbol{x}_1^T & \boldsymbol{x}_2^T \end{pmatrix}^T = \begin{pmatrix} \boldsymbol{q}^T & \dot{\boldsymbol{q}}^T \end{pmatrix}^T \in \mathbb{R}^{2(n+2)}$, the state-space representation of the dynamic model (2) is

$$\dot{\boldsymbol{x}} = \boldsymbol{f}(\boldsymbol{x}) + \begin{pmatrix} \boldsymbol{0}_{(n+2)\times 2} \\ \boldsymbol{G} \end{pmatrix}\boldsymbol{u}, \tag{6}$$

with

$$\boldsymbol{f}(\boldsymbol{x}) = \begin{pmatrix} \boldsymbol{x}_2 \\ -\boldsymbol{M}^{-1}(\boldsymbol{x}_1)\left(\boldsymbol{c}(\boldsymbol{x}) + \boldsymbol{K}\boldsymbol{x}_1 + \boldsymbol{D}\boldsymbol{x}_2\right) \end{pmatrix}. \tag{7}$$

III. State Estimation by a Nonlinear Observer

Equations (4–5) provide the measurable output $\boldsymbol{y} \in \mathbb{R}^3$ available for the considered flexible manipulator. However, the other state variables are also required for control and monitoring purposes, and this motivates the need of a full state observer. For this, we shall adopt the nonlinear observer structure proposed in [20], [21]. However, because of some critical points that naturally arise in the present framework, a slight modification of the original equations is needed. We present the rationale of the modified design in this section and will check then its efficacy through simulations performed under a suitable closed-loop feedback control.

In order to introduce the proposed observer equations, consider first a nonlinear autonomous system described by

$$\dot{\boldsymbol{x}} = \boldsymbol{f}(\boldsymbol{x}), \qquad \boldsymbol{y} = \boldsymbol{h}(\boldsymbol{x}), \tag{8}$$

with drift $\boldsymbol{f}\colon \mathbb{R}^\nu \to \mathbb{R}^\nu$ and output map $\boldsymbol{h}\colon \mathbb{R}^\nu \to \mathbb{R}^\mu$. The synthesis of the observer requires building a coordinate transformation in the state space that exploits the shape of the output equation.

To this end, take the (repeated) *Lie derivative* of the j-th output $y_j = h_j(\boldsymbol{x})$, $j \in \{1,\dots,\mu\}$, along vector field $\boldsymbol{f}(\boldsymbol{x})$

$$L_f h_j(\boldsymbol{x}) = \frac{dh_j}{d\boldsymbol{x}}\boldsymbol{f}(\boldsymbol{x}), \quad L_f^k h_j(\boldsymbol{x}) = L_f\Big(L_f^{k-1}h_j(\boldsymbol{x})\Big),$$

for $k \geq 1$ (with $L_f^0 h_j(\boldsymbol{x}) = h_j(\boldsymbol{x})$), and collect μ vectors of the type

$$\Phi_j^T(\boldsymbol{x}) = \begin{pmatrix} h_j(\boldsymbol{x}) & L_f h_j(\boldsymbol{x}) & \dots & L_f^{\nu_j-1}h_j(\boldsymbol{x}) \end{pmatrix}^T \in \mathbb{R}^{\nu_j} \tag{9}$$

such that $\nu_1 + \dots + \nu_\mu = \nu$ and

$$\boldsymbol{z} = \Phi(\boldsymbol{x}) = \begin{pmatrix} \Phi_1^T(\boldsymbol{x}) & \dots & \Phi_\mu^T(\boldsymbol{x}) \end{pmatrix}^T \tag{10}$$

is invertible in a given set $\Omega \subseteq \mathbb{R}^\nu$. The nonlinear map $\boldsymbol{\Phi}$ is an *observability map* in Ω if it is a diffeomorphism in an open set containing Ω. Then, the system is said to be *drift-observable* in Ω [21]. By virtue of the drift-observability property, the following observer can be written

$$\dot{\hat{\boldsymbol{x}}} = \boldsymbol{f}(\hat{\boldsymbol{x}}) + \boldsymbol{J}_{\boldsymbol{\Phi}}^{-1}(\hat{\boldsymbol{x}})\,\boldsymbol{\Gamma}\left(\boldsymbol{y} - \boldsymbol{h}(\hat{\boldsymbol{x}})\right) \tag{11}$$

where $\boldsymbol{J}_{\boldsymbol{\Phi}}$ is the Jacobian of $\boldsymbol{\Phi}$

$$\boldsymbol{J}_{\boldsymbol{\Phi}} = \left.\frac{\partial \boldsymbol{\Phi}}{\partial \boldsymbol{x}}\right|_{\boldsymbol{x}=\boldsymbol{\Phi}^{-1}(\boldsymbol{z})} \tag{12}$$

and $\boldsymbol{\Gamma} = \operatorname{diag}\{\boldsymbol{\Gamma}_1, \ldots, \boldsymbol{\Gamma}_\mu\} > 0$ is the block-diagonal observer gain matrix, with $\boldsymbol{\Gamma}_j \in \mathbb{R}^{\nu_j \times 1}$ designed in order to have eigenvalues with negative real part for $\boldsymbol{A}_{bj} + \boldsymbol{\Gamma}_j \boldsymbol{C}_{bj}$, where $(\boldsymbol{A}_{bj}, \boldsymbol{C}_{bj})$ is an ν_j-dimensional Brunowski pair.

In [20], [21] it has been proven that, provided that *(i)* the system is globally uniformly Lipschitz drift-observable, i.e., it is drift-observable in $\mathbb{R}^\nu$, with $\boldsymbol{\Phi}$ and $\boldsymbol{\Phi}^{-1}$ uniformly Lipschitz in $\mathbb{R}^\nu$ and *(ii)* functions $L_f^{\nu_j} h_j$, $j = 1, \ldots, \mu$, are uniformly Lipschitz in $\mathbb{R}^\nu$, then the gain $\boldsymbol{\Gamma}$ can be designed so as to ensure exponential convergence to zero of the state observation error, whatever the initial state estimate is. By weakening these properties, i.e., allowing them to hold only in a subset $\Omega \subset \mathbb{R}^\nu$, exponential convergence is only local and the available results concern semiglobal observers [21].

Differently from the autonomous system (8), consider next a nonlinear control system of the form

$$\dot{\boldsymbol{x}} = \boldsymbol{f}(\boldsymbol{x}) + \boldsymbol{g}(\boldsymbol{x})\boldsymbol{u}, \qquad \boldsymbol{y} = \boldsymbol{h}(\boldsymbol{x}), \tag{13}$$

with the additional input matrix $\boldsymbol{g}(\boldsymbol{x})$ whose ρ columns are the vector fields $\boldsymbol{g}_i \colon \mathbb{R}^\nu \to \mathbb{R}^\nu$, for $i = 1, \ldots, \rho$. In general, the design of a state observer for (13) is based on computing the relative degrees [22] for each output y_j, namely the integers $r_j \geq 1$, $j = 1, \ldots, \mu$, such that on a domain $\Omega \subseteq \mathbb{R}^\nu$:

$$\begin{aligned} &\forall \boldsymbol{x} \in \Omega, \quad L_g L_f^k h_j(\boldsymbol{x}) = 0, \qquad k = 0, 1, \ldots, r_j - 2, \\ &\exists \boldsymbol{x} \in \Omega\colon \quad L_g L_f^{r_j - 1} h_j(\boldsymbol{x}) \neq 0. \end{aligned}$$

The available theory concerning convergence properties of a state observer for the nonlinear control system (13) is based on the *vector relative degree* property [22] and requires (at least) that $r = r_1 + \cdots + r_\mu = \nu$. It is easy to verify that, for the dynamic model (2) of our flexible robot arm and with all $\mu = 3$ available outputs (4–5), this necessary condition is violated since

$$r_1 = r_2 = r_3 = 2 \quad \Longrightarrow \quad r = 6 < \nu = 8.$$

Nonetheless, we note that the drift-observability map could be written even without the vector relative degree property. However, the generalized form of the state transformation $\boldsymbol{\Phi}$ in (10), as applied to (13), will depend on the input $\boldsymbol{u}$ as well as on its time derivatives (except in special cases). This makes indeed the design of a state observer more critical.

In order to overcome such a drawback, we propose a simple heuristic that explicitly takes into account that the control input $\boldsymbol{u}$ in (13) is typically designed as $\boldsymbol{u}(\boldsymbol{x})$, namely as a suitable feedback law from the state $\boldsymbol{x}$. Then, we would obtain the new autonomous closed-loop system

$$\dot{\boldsymbol{x}} = \boldsymbol{f}(\boldsymbol{x}) + \boldsymbol{g}(\boldsymbol{x})\boldsymbol{u}(\boldsymbol{x}).$$

Indeed, a state observer is required because $\boldsymbol{u}(\boldsymbol{x})$ needs in general some state components that are not directly measured. Therefore, the closed-loop system will be rather written as

$$\dot{\boldsymbol{x}} = \boldsymbol{f}(\boldsymbol{x}) + \boldsymbol{g}(\boldsymbol{x})\boldsymbol{u}(\hat{\boldsymbol{x}}), \tag{14}$$

where $\hat{\boldsymbol{x}} \in \mathbb{R}^\nu$ is the estimated state. In analogy to (11), we propose then the following observer structure

$$\dot{\hat{\boldsymbol{x}}} = \boldsymbol{f}(\hat{\boldsymbol{x}}) + \boldsymbol{g}(\hat{\boldsymbol{x}})\boldsymbol{u}(\hat{\boldsymbol{x}}) + \boldsymbol{J}_{\boldsymbol{\Phi}}^{-1}(\hat{\boldsymbol{x}})\boldsymbol{\Gamma}(\boldsymbol{y} - \boldsymbol{h}(\hat{\boldsymbol{x}})), \tag{15}$$

where $\boldsymbol{\Phi}$ is built with the same selection of Lie derivatives used in (9–10), but replacing the vector field $\boldsymbol{f}(\boldsymbol{x})$ with $\tilde{\boldsymbol{f}}(\boldsymbol{x}) = \boldsymbol{f}(\boldsymbol{x}) + \boldsymbol{g}(\boldsymbol{x})\boldsymbol{u}(\boldsymbol{x})$.

A. A heuristic to overcome local singularities

To implement the state observer (15) we need to verify first the drift-observability of system (14), in particular that its generalized observability map $\boldsymbol{\Phi}$ is globally invertible. However, in many applications one can typically ensure only local drift-observability. As a result, the Jacobian $\boldsymbol{J}_{\boldsymbol{\Phi}}$ in (12) may become ill-conditioned close to some points of the operating space. In order to overcome this obstruction we replace the inverse of $\boldsymbol{J}_{\boldsymbol{\Phi}}$ in (15) with its Damped Least Squares (DLS) inverse $\boldsymbol{J}_{\boldsymbol{\Phi}}^{\dagger}(\hat{\boldsymbol{x}})$ when the Jacobian is too close to a singularity, namely when its (squared) smallest singular value $\lambda_\nu > 0$ drops below a small threshold $\eta > 0$. Using Algorithm 1 with a suitable damping factor[1] $\sigma > 0$ will limit the updating rate of $\hat{\boldsymbol{x}}$ in (15).

Algorithm 1: choose parameters $\eta > 0$ and $\sigma > 0$
input $\boldsymbol{J}_{\boldsymbol{\Phi}}(\hat{\boldsymbol{x}})$
let $\lambda_\nu = \min \lambda\{\boldsymbol{J}_{\boldsymbol{\Phi}}^T(\hat{\boldsymbol{x}})\boldsymbol{J}_{\boldsymbol{\Phi}}(\hat{\boldsymbol{x}})\}$
if $\lambda_\nu > \eta$, then $\boldsymbol{J}_{\boldsymbol{\Phi}}^{\dagger}(\hat{\boldsymbol{x}}) = \boldsymbol{J}_{\boldsymbol{\Phi}}^{-1}(\hat{\boldsymbol{x}})$
else $\boldsymbol{J}_{\boldsymbol{\Phi}}^{\dagger}(\hat{\boldsymbol{x}}) = (\sigma \boldsymbol{I}_{n\times n} + \boldsymbol{J}_{\boldsymbol{\Phi}}^T(\hat{\boldsymbol{x}})\boldsymbol{J}_{\boldsymbol{\Phi}}(\hat{\boldsymbol{x}}))^{-1}\boldsymbol{J}_{\boldsymbol{\Phi}}^T(\hat{\boldsymbol{x}})$
end
output $\boldsymbol{J}_{\boldsymbol{\Phi}}^{\dagger}(\hat{\boldsymbol{x}})$

This algorithm is effective whenever the singularities in the observability map are somewhat isolated and may be avoided by small perturbations of the system dynamics. In such cases, a global practical convergence of the observer (15) enhanced with the DLS inverse $\boldsymbol{J}_{\boldsymbol{\Phi}}^{\dagger}(\hat{\boldsymbol{x}})$ can be expected.

IV. Observer Design for the Flexarm

In this section, we design a nonlinear state observer (15) for the *Flexarm* robot arm of Sec. II. Only the first $n = 2$ deflection modes $\delta_1(t)$ and $\delta_2(t)$ in (1) will be considered, being the most relevant to capture the flexible system dynamics. The state equations (6–7) will be $\nu = 2(n+2) = 8$, with $\rho = 2$ inputs given by the motor torques.

The proposed observer design requires in the first place the choice of a state feedback control law. Following a standard practice in the field of flexible manipulators [1], we consider a desired trajectory tracking tasks $\boldsymbol{\theta}_{c,des}(t)$ (or a set-point regulation task, when $\boldsymbol{\theta}_{c,des}$ is constant) for the motor positions

$$\boldsymbol{\theta}_c = \begin{pmatrix} \theta_{c1} \\ \theta_{c2} \end{pmatrix} = \begin{pmatrix} \theta_1 \\ \theta_2 + \phi'_{10}\delta_1 + \phi'_{20}\delta_2 \end{pmatrix} = \boldsymbol{\theta}_c(\boldsymbol{x}), \tag{16}$$

to be achieved by a PD motor control law (without feedforward term, for simplicity)

$$\boldsymbol{u} = \boldsymbol{K}_P \left(\boldsymbol{\theta}_{c,des} - \hat{\boldsymbol{\theta}}_c\right) + \boldsymbol{K}_D \left(\dot{\boldsymbol{\theta}}_{c,des} - \dot{\hat{\boldsymbol{\theta}}}_c\right), \tag{17}$$

[1]The value of σ can be possibly made dependent on $\hat{\boldsymbol{x}}$ in order to achieve continuity of λ_ν, which is however not mandatory in the present context.

where $\boldsymbol{K}_P > 0$ and $\boldsymbol{K}_D > 0$ are diagonal gain matrices.

In order to obtain a reliable estimation $\hat{\boldsymbol{x}}$ of the state $\boldsymbol{x}$, we used all $\mu = 3$ measurable outputs (4–5) that are available in this robotic system. The observability map $\boldsymbol{\Phi} \in \mathbb{R}^8$ is given by the column vector

$$\begin{aligned} \boldsymbol{z} = \boldsymbol{\Phi}(\boldsymbol{x}) = \big(& \theta_{c1}(\boldsymbol{x}) \quad L_{\tilde{f}}\theta_{c1}(\boldsymbol{x}) \quad L^2_{\tilde{f}}\theta_{c1}(\boldsymbol{x}) \\ & \theta_{c2}(\boldsymbol{x}) \quad L_{\tilde{f}}\theta_{c2}(\boldsymbol{x}) \quad L^2_{\tilde{f}}\theta_{c2}(\boldsymbol{x}) \\ & y_{tip}(\boldsymbol{x}) \quad L_{\tilde{f}}y_{tip}(\boldsymbol{x}) \big)^T. \end{aligned} \tag{18}$$

Note that, due to the mechanical nature of the system, the first-order Lie derivatives of any positional output $h_j(\boldsymbol{x}_1)$ satisfy the identities $L_{\tilde{f}}h_j(\boldsymbol{x}) = L_f h_j(\boldsymbol{x})$, for $j = 1, \ldots, \mu$. The choice (18) provides a full rank Jacobian matrix $\boldsymbol{J}_\Phi$ in most regions of the explored state space.

The last design step is the choice of the observer gain matrix $\boldsymbol{\Gamma} \in \mathbb{R}^{8\times 3}$. According to the obtained drift indices $\nu_1 = 3$, $\nu_2 = 3$ and $\nu_3 = 2$, its structure has been chosen as

$$\boldsymbol{\Gamma} = \begin{pmatrix} \gamma_1 & \gamma_2 & \gamma_3 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \gamma_1 & \gamma_2 & \gamma_3 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & \gamma_4 & \gamma_5 \end{pmatrix}^T. \tag{19}$$

V. Simulation Results

The features of the proposed nonlinear observer are illustrated on our flexible robot arm by considering the performance of the PD control law (17) using only estimated states, as well as through a detection scheme for actuator faults or robot collisions that is based on the observer outputs.

The dynamic parameters of the *Flexarm* used in simulation are those reported in the experimental study [19]. The first two bending modes have eigenfrequencies $f_1 = 4.716$ and $f_2 = 14.395$ [Hz] ($\omega_i = 2\pi f_i$). The PD law (17) is applied to track the desired joint trajectories

$$\theta_{c1,des}(t) = 2\sin 0.05\pi t, \quad \theta_{c2,des}(t) = 2\sin 0.1\pi t, \tag{20}$$

with gain matrices chosen as

$$\boldsymbol{K}_P = \mathrm{diag}\left\{ 3 \quad 1 \right\}, \qquad \boldsymbol{K}_D = \mathrm{diag}\left\{ 1.5 \quad 1 \right\}. \tag{21}$$

These control gains are used also in the closed-loop nonlinear observer (15), with the observer gains in (19) tuned as

$$\gamma_1 = 6, \;\; \gamma_2 = 11, \;\; \gamma_3 = 6, \;\; \gamma_4 = 3, \;\; \gamma_5 = 2. \tag{22}$$

A. State observation under dynamic feedback control

In the first simulation, no actuator fault or link collision occur. The initial system state $\boldsymbol{x}_0 = \boldsymbol{x}(0)$ and observer state $\hat{\boldsymbol{x}}_0 = \hat{\boldsymbol{x}}(0)$ are

$$\begin{aligned} \boldsymbol{x}_0 &= \begin{pmatrix} \boldsymbol{q}_0^T & \dot{\boldsymbol{q}}_0^T \end{pmatrix}^T \\ &= \begin{pmatrix} \pi/2 & \pi/2 & 0.1 & 0.1 & 0.1 & 0.1 & 0.01 & 0.01 \end{pmatrix}^T, \\ \hat{\boldsymbol{x}}_0 &= \begin{pmatrix} \hat{\boldsymbol{q}}_0^T & \hat{\dot{\boldsymbol{q}}}_0^T \end{pmatrix}^T = \boldsymbol{0}. \end{aligned}$$

Figure 2 shows the evolution of the eight actual and estimated state variables, while the output evolution of the clamped joint angles and of the (uncontrolled) tip angle of the forearm are reported in Fig. 3. Convergence of the estimated robot state (in blue) to the actual state (in dashed black) occurs within the simulated time span. Once such convergence is achieved (in about 12 seconds), the chosen PD controller successfully drives the two controlled outputs (the clamped joint angles $\boldsymbol{\theta}_c$) along the desired trajectory.

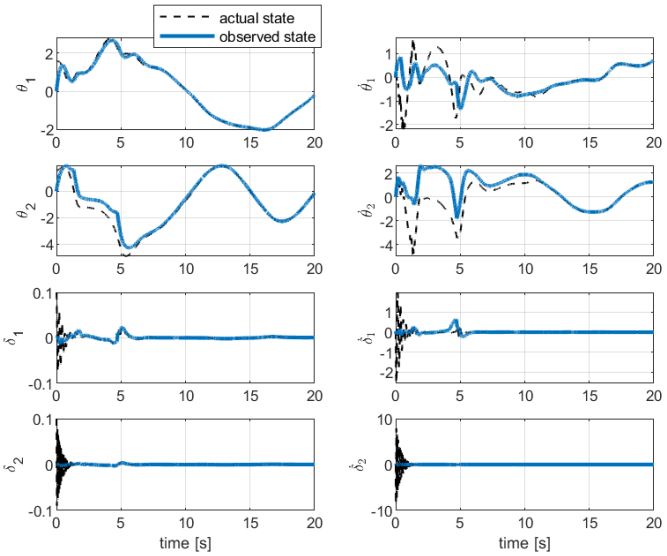

Fig. 2. Actual (dashed black) and estimated (blue) states of the *Flexarm*.

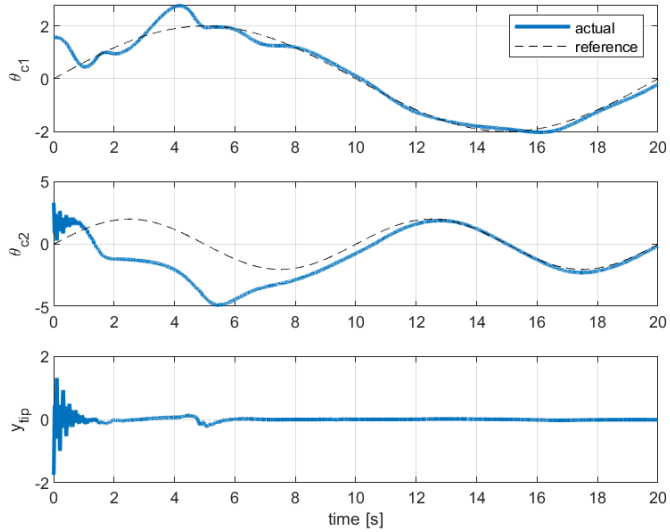

Fig. 3. Output variables of the *Flexarm*.

B. Monitoring anomalies

While the control algorithm runs, a routine for detecting anomalies and possibly transient, undesired events will monitor a scalar quantity $\epsilon \geq 0$ related to the state estimation error and defined as (see (16))

$$\epsilon = \left(\boldsymbol{\theta}_c - \hat{\boldsymbol{\theta}}_c\right)^T \left(\boldsymbol{\theta}_c - \hat{\boldsymbol{\theta}}_c\right), \tag{23}$$

For monitoring purposes, we have chosen to compare the measured and the estimated output in place of the desired and the actual one, so as to separate the performance of the anomaly diagnosis from the performance of the chosen control law.

When ϵ exceeds a given threshold $\bar{\epsilon}$, an anomaly is detected and highlighted by modifying a diagnosis flag. The three-valued flag is $F = 1$ when monitoring is not active, i.e., during the observer transient phase in which the state estimation error has not yet converged to zero. When the monitoring routine is active and the system is in normal

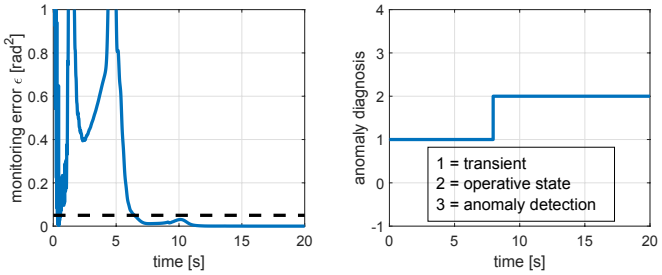

Fig. 4. Squared norm ϵ of the output estimation error [left] and diagnosis signal [right] in the no-anomaly case of Figs. 2–3.

operation, the flag is $F = 2$. When an anomaly is detected, i.e., when $\epsilon > \bar{\epsilon}$, the value of the flag becomes $F = 3$. To increase robustness with respect to noise and limit false positives/negatives, the switching between two flag values occurs when the signal ϵ remains for at least t_d seconds below (in the transitions $1 \to 2$ and $3 \to 2$) or above (in the transition $2 \to 3$) the threshold $\bar{\epsilon}$. Figure 4 shows the evolution of ϵ and the correct operation of the associated diagnosis signal for the no-anomaly case of Sec. V-A. Here and in the following, we have set a threshold $\bar{\epsilon} = 0.05$ [rad^2] and a dwell time $t_d = 1.5$ [s].

C. Control with faulty actuators

We consider next the possible occurrence of actuator faults. We tested our observer framework in the presence of single and concurrent actuator faults on both joints, and successfully detected the anomalies in all cases. Due to space limitations, only one typical result is reported here. Without loss of generality, the initial system state $\boldsymbol{x}_0$ is set closer to the initial observer state $\hat{\boldsymbol{x}}_0 = \mathbf{0}$,

$$\begin{aligned}\boldsymbol{x}_0 &= \begin{pmatrix} \boldsymbol{q}_0^T & \dot{\boldsymbol{q}}_0^T \end{pmatrix}^T \\ &= \begin{pmatrix} \pi/8 & -\pi/8 & 0.01 & 0 & \pi/20 & 0 & 0 & 0 \end{pmatrix}^T,\end{aligned}$$

just to reduce the initial transient time in the simulation. An abrupt fault occurs on the first motor (moving the rigid link) at time $t_{F_{m,1}} = 12$ [s], when the motor has a power loss and is no longer able to supply more than 10% of the torque requested to perform the desired trajectory (20), see Fig. 5. Although the applied control law (17) is decentralized in nature, both controlled outputs $\boldsymbol{\theta}_c$ are affected by this fault and diverge from their respective reference trajectory. This is due to the sudden divergence of the state estimated by the observer, which is needed in turn for retrieving both clamped angles $\hat{\boldsymbol{\theta}}_c$ used by the PD controller. For this reason, it is not possible to isolate the fault with the chosen monitoring signals, but only to obtain detection. Figure 6 shows the evolution of $\epsilon(t)$ in (23) and the diagnosis signal during the execution of the trajectory. It can be seen that the diagnosis flag switches from 2 (normal operation) to 3 (anomaly detection) almost 3 seconds after the beginning of the motor fault. Such delay can be reduced by decreasing the threshold $\bar{\epsilon}$. On the other hand, this may lead to an increase in the number of false positives being detected.

D. Control in the presence of link collisions

In case of unexpected collisions of the robot with the environment, the contact/impact forces exerted on the links

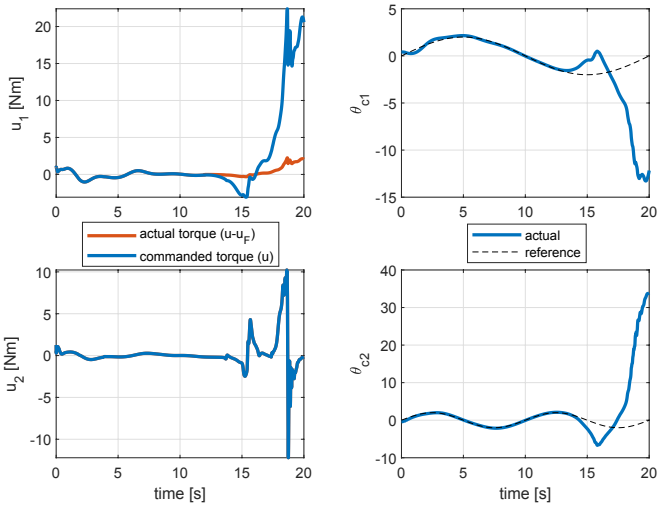

Fig. 5. Input torques and controlled outputs when a 90% power loss fault occurs on the first motor, starting at $t_{F_{m,1}} = 12$ [s].

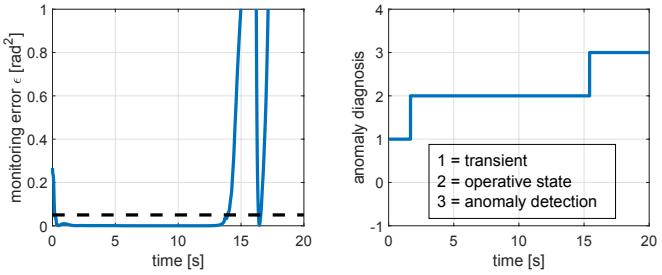

Fig. 6. Squared norm ϵ of the output estimation error [left] and diagnosis signal [right] when the fault on the first motor occurs.

of the *Flexarm* generate (unknown) extra joint torques that are added to the commanded ones. We have simulated the presence of an external force $\boldsymbol{F}_{ext} = -\begin{pmatrix} 1 & 1 \end{pmatrix}^T$ acting in the plane of robot motion and separately applied to both links, in particular at the two origins of the two frames O_1 and O_{2e}, see the left picture in Fig. 1. The force $\boldsymbol{F}_{ext}$ is applied to link 1 in the time interval from $t_{F_{1,init}} = 10$ [s] to $t_{F_{1,end}} = 12$ [s], while the same force is applied to link 2 between $t_{F_{2,init}} = 25$ [s] and $t_{F_{2,end}} = 27$ [s]. These external forces produce torques $\boldsymbol{u}_{ext}$ at the robot joints according to the standard mapping

$$\boldsymbol{u}_{ext} = \boldsymbol{J}_P^T(\boldsymbol{q})\boldsymbol{F}_{ext},$$

where $\boldsymbol{J}_P(\boldsymbol{q})$ is the configuration-dependent Jacobian matrix that relates the joint velocity $\dot{\boldsymbol{\theta}}_c \in \mathbb{R}^2$ to the linear velocity $\boldsymbol{v}_P \in \mathbb{R}^2$ of the contact point P on the robot body. Figure 7 shows the commanded and actual torques acting on the joints, together with the two measured and controlled outputs θ_{c1} and θ_{c2} (and their reference trajectories). The anomaly detection scheme successfully detects each of the two collisions within slightly less than 2 seconds from the application of the external force, see Fig. 8. The duration of these events, however, is estimated to be longer ($\simeq 5$ s) than their real value (2 s). This happens because the estimated state, which provides $\hat{\boldsymbol{\theta}}_c$ to the detection process, experiences new transient phases when the external forces are removed.

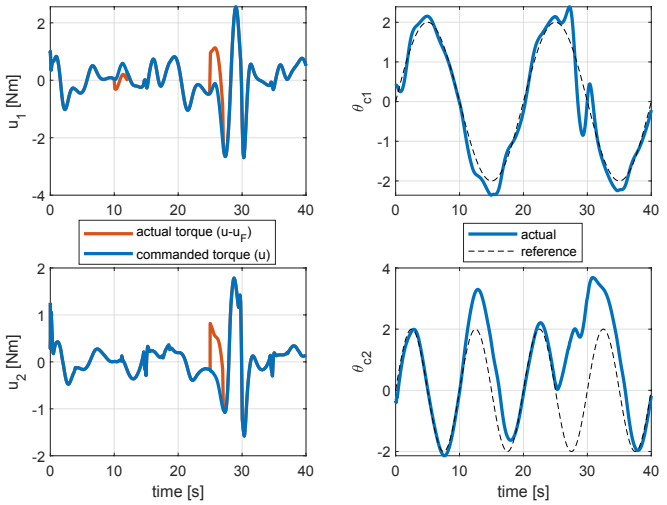

Fig. 7. Input torques and controlled outputs when when the two link collisions occur.

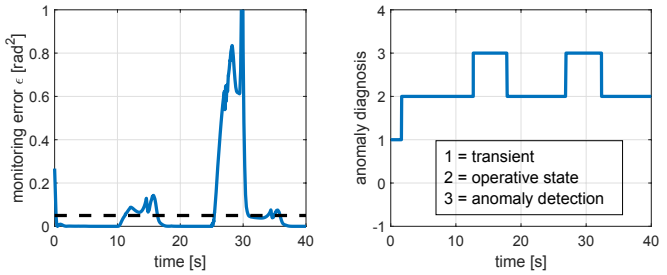

Fig. 8. Squared norm ϵ of the output estimation error [left] and diagnosis signal [right] when the two link collisions occur.

VI. Conclusions

In this paper, we have proposed a nonlinear state observer for the *Flexarm*, a two-link planar manipulator having a rigid upper link and a flexible forearm. Standard encoders at the joints and an optical sensor for the tip deformation are used as measurements to estimate the full state of the flexible robot, i.e., joint positions and velocities, and link deflections and their time derivative. The observer design is based on the drift-observability property, which is obtained in a closed-loop fashion when the robotic system is controlled by a simple PD feedback action at the joint level.

The observer-based dynamic feedback achieves an overall stable behavior even with large initial estimation errors, while good trajectory tracking performance are obtained upon convergence of the observation process. To cope with the semi-global nature of the observer and to prevent the negative effects of singularities, a regularization method can be included using the damped least squares inverse of the Jacobian of the drift-observability matrix. Using the output estimates of the observer, we addressed also the problem of monitoring the occurrence of possible anomalies during the motion of the flexible manipulator. In particular, the proposed scheme allows detection of single or concurrent actuator faults, as well as of link collisions along the structure, without the need of additional sensors.

Although relatively simple, the nonlinear and coupled dynamics of the *Flexarm* is already a challenging benchmark for state estimation and trajectory tracking control problems. As a result, the proposed observer design, together with the anomaly detection scheme, is expected to be valid even for more general robots with flexible links, and also for the new emerging class of soft manipulators. Our current work is on using the nonlinear observer to implement other residual-based methods (such as [16]) that allow detection *and* isolation (which is indeed not achievable with the procedure discussed in this paper) of actuator faults or link collisions in flexible manipulators when the full state is not available for measure.

References

[1] A. De Luca and W. Book, "Robots with flexible elements," in *Handbook of Robotics*, B. Siciliano and O. Khatib, Eds. Springer, 2016, pp. 243–282.

[2] A. Albu-Schäffer, O. Eiberger, M. Grebenstein, S. Haddadin, C. Ott, T. Wimbock, S. Wolf, and G. Hirzinger, "Soft robotics," *IEEE Robotics and Automation Mag.*, vol. 15, no. 3, pp. 20–30, 2008.

[3] J.-H. Park and H. Asada, "Design and control of minimum-phase flexible arms with torque transmission mechanisms," in *Proc. IEEE Int. Conf. on Robotics and Automation*, 1990, pp. 1790–1795.

[4] Ö. Morgül, "Orientation and stabilization of a flexible beam attached to a rigid body: Planar motion," *IEEE Trans. on Automatic Control*, vol. 36, no. 8, pp. 953–962, 1991.

[5] Z.-H. Luo, "Direct strain feedback control of flexible robot arms: New theoretical and experimental results," *IEEE Trans. on Automatic Control*, vol. 38, no. 11, pp. 1610–1622, 1993.

[6] M. Moallem, R. V. Patel, and K. Khorasani, "An inverse dynamics control strategy for tip position tracking of flexible multi-link manipulators," *J. of Robotic Systems*, vol. 14, no. 9, pp. 649–658, 1997.

[7] J. Liu and W. He, *Distributed Parameter Modeling and Boundary Control of Flexible Manipulators*. Springer, 2018.

[8] F. Bellezza, L. Lanari, and G. Ulivi, "Exact modeling of the flexible slewing link," in *Proc. IEEE Int. Conf. on Robotics and Automation*, 1990, pp. 734–739.

[9] S. Nicosia and P. Tomei, "Robot control by using only joint position measurements," *IEEE Trans. on Automatic Control*, vol. 35, no. 9, pp. 1058–1061, 1990.

[10] F. Celani, "A Luenberger-style observer for robot manipulators with position measurements," in *Proc. 14th Mediterranean Conf. on Control and Automation*, 2006, pp. 1–6.

[11] H. Berghuis and H. Nijmeijer, "A passivity approach to controller-observer design for robots," *IEEE Trans. on Robotics and Automation*, vol. 9, no. 6, pp. 740–754, 1993.

[12] C. Canudas De Wit and J.-J. Slotine, "Sliding observers for robot manipulators," *Automatica*, vol. 27, no. 5, pp. 859–864, 1991.

[13] T. D. Nguyen and O. Egeland, "Infinite dimensional observer for a flexible robot arm with a tip load," *Asian J. of Control*, vol. 10, no. 4, pp. 456–461, 2008.

[14] T. Jiang, J. Liu, and W. He, "A robust observer design for a flexible manipulator based on a PDE model," *J. of Vibration and Control*, vol. 23, no. 6, pp. 871–882, 2017.

[15] A. Zuyev and O. Sawodny, "Observer design for a flexible manipulator model with a payload," in *Proc. 45th IEEE Conf. on Decision and Control*, 2006, pp. 4490–4495.

[16] C. Gaz, A. Cristofaro, and A. De Luca, "Detection and isolation of actuator faults and collisions for a flexible robot arm," in *Proc. 59th IEEE Conf. on Decision and Control*, 2020, pp. 2684–2689.

[17] A. De Luca and R. Mattone, "Sensorless robot collision detection and hybrid force/motion control," in *Proc. IEEE Int. Conf. on Robotics and Automation*, 2005, pp. 999–1004.

[18] S. Haddadin, A. De Luca, and A. Albu-Schäffer, "Robot collisions: A survey on detection, isolation, and identification," *IEEE Trans. on Robotics*, vol. 33, no. 6, pp. 1292–1312, 2017.

[19] A. De Luca, L. Lanari, P. Lucibello, S. Panzieri, and G. Ulivi, "Control experiments on a two-link robot with a flexible forearm," in *Proc. 29th IEEE Conf. on Decision and Control*, 1990, pp. 520–527.

[20] G. Ciccarella, M. Dalla Mora, and A. Germani, "A Luenberger-like observer for nonlinear systems," *Int. J. of Control*, vol. 57, no. 3, pp. 537–556, 1993.

[21] M. Dalla Mora, A. Germani, and C. Manes, "Design of state observers from a drift-observability property," *IEEE Trans. on Automatic Control*, vol. 45, no. 8, pp. 1536–1540, 2000.

[22] A. Isidori, *Nonlinear Control Systems*, 3rd ed. Springer, 1995.