2020 59th IEEE Conference on Decision and Control (CDC)
Jeju Island, Republic of Korea, December 14-18, 2020

# Detection and isolation of actuator faults and collisions for a flexible robot arm

Claudio Gaz, Andrea Cristofaro, Alessandro De Luca

***Abstract*— This paper presents a unified approach to detection and isolation of both actuator faults and unexpected collisions for a two-link robot with a flexible forearm. The proposed approach is sensorless, i.e., no dedicated exteroceptive sensors are considered, and is based on the design of residuals to be used as monitoring filters. The method has been tested by extensive simulations on the Flexarm robot used as case study. The reported results show the efficacy in detecting and isolating faults in the actuators or collisions on the robot links.**

## I. Introduction

The demand for light and adaptable mechanical structures in cutting-edge industrial sectors, such as aerospace or biomedical, has led to an increasing interest for systems with flexible components. Such enhanced paradigm for mechanical systems had a remarkable impact on the robotic research too, with the creation of the new branch of soft robotics [1]. Robots with elastic joints or flexible links are usually lighter and energy efficient, increase the safety level in collisions, and may provide better adaptive characteristics, for example in grasping tasks.

Because of these promising features, many control problems have been investigated for flexible manipulators [2], [3], [4], [5], [6]. While flexibility enables new abilities and desirable performance, it introduces also a larger complexity in the dynamical modeling and control design, e.g., by requiring the definition and evaluation of additional state variables. Indeed, the general model of a flexible robotic arm is described by an infinite-dimensional system, but two finite-dimensional model types are commonly used. The first one is based on finite-element approximations of the distributed deformation [7], while the second one consists in selecting a finite number of terms, referred to as *deformation modes*, from the series expression of the distributed solution [8].

As for other mechanical systems, flexible manipulators are subject to actuator faults that may alter the produced torques, with a consequent loss of performance or even instability. To prevent and avoid unsafe operational conditions, monitoring schemes are usually implemented with the aim of detecting, and possibly isolating, such faults. The risk of collisions is an issue also for flexible manipulators. As in the case of actuator faults, collision detection is typically performed by dedicated filters [9], referred to as *residuals*, that supervise the difference between the commanded torques at the joints and the actual ones, without the use of torque sensors. In particular, our method for residual design relies on physical principles and leads to an estimation of the extra torques at the joint level, reflecting the unknown external force applied on the link structure due to a collision/contact. As such, it has been used already as a valuable tool for rigid and massive robots in industrial contexts, both for safety and in human-robot collaborative tasks [10]. To compute these model-based residuals, the values of some state variables are needed, and these can be provided by direct measurements (e.g., through encoders), obtained by numerical derivation, or estimated by dynamic observers. The use of residuals for rigid robots, or for robots with elastic joints, leverages a well established setup. However, the generalization to robots with flexible links is still open.

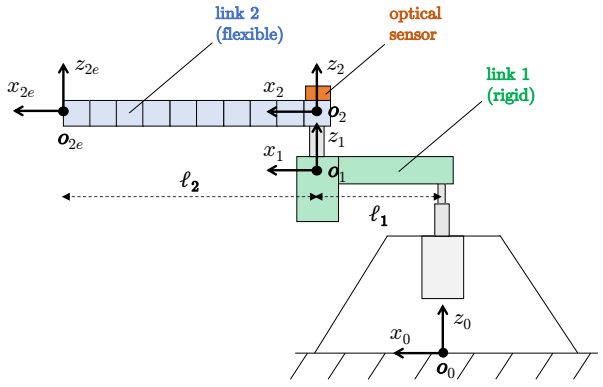


Fig. 1. Sketch of the Flexarm robot.

The aim of this paper is to present the design and the application of residuals to actuator faults and collision detection/isolation in a two-link robotic manipulator with a flexible forearm. For this, we will consider a finite-dimensional model of the flexible arm based on a finite number of deflection modes. We assume in the following that the entire robot state can be measured, typically through a combination of encoders, optical sensors, and strain gauges, while the commanded torques at the two joints are also easily available. We shall see that the designed residuals are able to detect and identify actuator faults on both joints and collisions occurring on either link of the robot. Furthermore, the same residuals allow to distinguish between an actuator fault and a collision on the second link.

The paper is organized as follows. In Sec. II, a short description of the Flexarm robot is given, presenting also the structure of its dynamic model. In Sec. III, the formulation of a momentum-based residual signal is presented in this context, separating the rigid from the flexible contributions. In Sec. IV, simulation results are reported, considering several situations of actuator faults and collisions. Conclusions and future work are discussed in Sec. V.

The authors are with the Department of Computer, Control and Management Engineering, Sapienza University of Rome, Via Ariosto 25, 00185 Rome, Italy. Corresponding author email: gaz@diag.uniroma1.it.

## II. Flexible Robot Arm Model

### A. Motivating example

In [11], a two-link planar manipulator with revolute joints and a flexible forearm has been presented, called *Flexarm*. Despite of its limited complexity, the peculiarity of this robot is that it includes the most relevant nonlinear and interacting dynamic effects of interest. The mechanism is mounted on a fixed horizontal basement, as sketched in Fig. 1. The upper link is very stiff with respect to the forearm, whose structure is designed in such a way to have flexibility only in the horizontal plane of motion, being relatively stiff in the vertical plane and with respect to torsion [11]. The arm is actuated by two DC motors located at the joints, in a direct-drive arrangement. No gear-boxes are used to couple the motors to the links, as they would introduce in the system undesired effects like backlash, friction and joint elasticity.

### B. The dynamic model

Inspired by the previous example, we aim at deriving a general model for a two-link robot with flexible forearm.

The first link is rigid and we denote with $\theta_1$ the related joint angle. In order to study the deflection of the flexible arm during motion, i.e., to compute the low frequency modes, the forearm link is modeled as an Euler-Bernoulli beam of length $\ell_2$, uniform density, and constant elastic properties. With reference to Fig. 2, for a link point $s \in [0, \ell_2]$, $w(s,t)$ is the bending deflection measured from the axis passing through the rotation axis of joint 2 and the Center of Mass (CoM) of the forearm. Accordingly, $\theta_2$ is the angle between this axis and the first rigid link. An approximation of order $n$ of the link deflection can be expressed as

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

with the time-varying coordinates $\delta_i(t)$ associated to the mode shapes $\phi_i(s)$ (for further details, see [11], [8]).

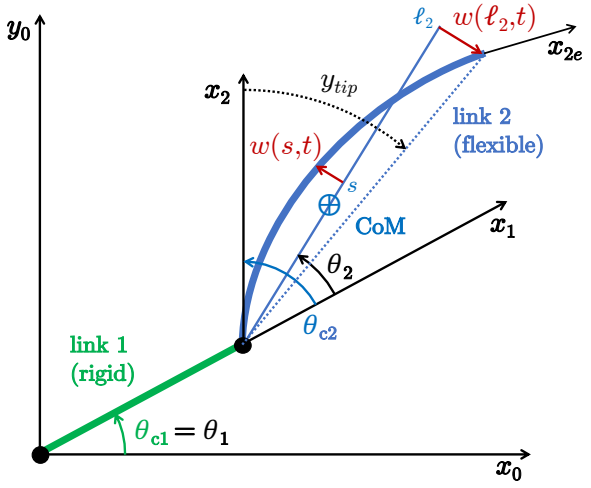


Fig. 2. Sketch of the Flexarm angles and deflection.

The dynamic model of the robot (neglecting gravity, which has no influence on this planar robot) can be expressed according to the Euler-Lagrange formulation as

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

where $\boldsymbol{q} = \begin{pmatrix} \boldsymbol{\theta}^T & \boldsymbol{\delta}^T \end{pmatrix}^T = \begin{pmatrix} \theta_1 & \theta_2 & \delta_1 & \dots & \delta_n \end{pmatrix}^T \in \mathbb{R}^{2+n}$ is the vector of the generalized coordinates, $\boldsymbol{M}(\boldsymbol{q}) > 0$ is the inertia matrix, $\boldsymbol{c}(\boldsymbol{q},\dot{\boldsymbol{q}})$ is the Coriolis and centrifugal vector, and $\boldsymbol{K} \geq 0$ is the stiffness matrix of the system. Joint viscous friction and modal damping coefficients are arranged on the diagonal of matrix $\boldsymbol{D} \geq 0$, while the input matrix $\boldsymbol{G}$ transforms the motor torques $\boldsymbol{u} \in \mathbb{R}^2$ into generalized forces performing work on $\boldsymbol{q}$.

Let the constant terms $\phi_{ie}$ and $\phi'_{i0}$ 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

$$\boldsymbol{G} = \begin{pmatrix} \boldsymbol{I}_{2\times 2} \\ \boldsymbol{G}_\delta \end{pmatrix}, \qquad \boldsymbol{G}_\delta = \begin{pmatrix} 0 & \phi'_{10} \\ 0 & \phi'_{20} \\ \dots & \dots \\ 0 & \phi'_{n0} \end{pmatrix}, \tag{4}$$

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

$$\boldsymbol{K} = \begin{pmatrix} \boldsymbol{0}_{2\times 2} & \boldsymbol{0}_{2\times n} \\ \boldsymbol{0}_{n\times 2} & \boldsymbol{K}_\delta \end{pmatrix}, \quad \boldsymbol{K}_\delta = \mathrm{diag}\begin{pmatrix} \omega_1^2 & \dots & \omega_n^2 \end{pmatrix}, \tag{5}$$

where $\boldsymbol{K}_\delta \in \mathbb{R}^{n\times n}$ contains the angular eigenfrequencies $\omega_i$, $i = 1, \dots, n$, of the flexible forearm. Modal damping is included by specifying

$$\boldsymbol{D} = \begin{pmatrix} \boldsymbol{0}_{2\times 2} & \boldsymbol{0}_{2\times n} \\ \boldsymbol{0}_{n\times 2} & \boldsymbol{D}_\delta \end{pmatrix}, \ \boldsymbol{D}_\delta = \mathrm{diag}\begin{pmatrix} 2\zeta_1\omega_1 & \dots & 2\zeta_n\omega_n \end{pmatrix}, \tag{6}$$

where the first two zeros on the diagonal of $\boldsymbol{D}$ reflect the fact that friction at the joints is very small and can be neglected.

The relevant outputs for this system that are typically considered for control are the angular positions $\theta_{c1}$ and $\theta_{c2}$ of the two motors ('clamped' to the link bases), which can be measured by encoders, and the tip deflection $y_{\text{tip}}$ of the forearm, which is measured in the Flexarm by an optical sensor mounted at the base of the forearm. These quantities can be expressed as linear combinations of the components of $\boldsymbol{q}$, namely as

$$\boldsymbol{\theta}_c = \begin{pmatrix} \theta_{c1} \\ \theta_{c2} \end{pmatrix} = \begin{pmatrix} \theta_1 \\ \theta_2 + \sum_{i=1}^{n} \phi'_{i0}\delta_i \end{pmatrix}, \tag{7}$$

and

$$y_{\text{tip}} = \frac{w(\ell_2,t)}{\ell_2} + \theta_2 - \theta_{c2} = \sum_{i=1}^{n} \left(\frac{\phi_{ie}}{\ell_2} - \phi'_{i0}\right)\delta_i. \tag{8}$$

## III. Residual for Actuator Fault/Collision Detection and Isolation

During motion, actuator faults and/or collisions between the manipulator and the environment (or a human operating in the robot workspace) may occur as unforeseen events.

In order to automatically recognize when an actuator fails or when a collision happens (without resorting to exteroceptive sensors, such as a tactile skin or force/torque sensors), the physically oriented design of the *residual* signal in [9], [12] can be exploited also in the present case, both

for *detecting* the occurrence of a fault/collision as well as for *isolating* the interested joint/link.

With reference to the Flexarm, we assume to have at disposal information about the full state $\boldsymbol{x}$ of the system, namely

$$\begin{aligned}\boldsymbol{x} &= \left( \boldsymbol{q}^T \;\; \dot{\boldsymbol{q}}^T \right)^T \\ &= \left( \theta_1 \;\; \theta_2 \;\; \delta_1 \;\; \ldots \;\; \delta_n \;\; \dot{\theta}_1 \;\; \dot{\theta}_2 \;\; \dot{\delta}_1 \;\; \ldots \;\; \dot{\delta}_n \right)^T.\end{aligned} \tag{9}$$

We will define in general a residual vector for capturing an unknown (in amplitude and sign) extra torque $\boldsymbol{u}_F \in \mathbb{R}^2$ that may appear in the dynamic model (2), as a result of an actuator fault or a collision. We would have then

$$\boldsymbol{M}(\boldsymbol{q})\ddot{\boldsymbol{q}} + \boldsymbol{c}(\boldsymbol{q}, \dot{\boldsymbol{q}}) + \boldsymbol{K}\boldsymbol{q} + \boldsymbol{D}\dot{\boldsymbol{q}} = \boldsymbol{G}\left(\boldsymbol{u} + \boldsymbol{u}_F\right). \tag{10}$$

For what follows, it is useful to decompose the inertia matrix $\boldsymbol{M}(\boldsymbol{q})$ in blocks as

$$\boldsymbol{M}(\boldsymbol{q}) = \begin{pmatrix} \boldsymbol{M}_{\theta\theta}(\theta_2, \delta_1, \ldots, \delta_n) & \boldsymbol{M}_{\theta\delta}(\theta_2) \\ \boldsymbol{M}_{\theta\delta}^T(\theta_2) & \boldsymbol{I}_{n\times n} \end{pmatrix}, \tag{11}$$

with $\boldsymbol{M}_{\theta\theta} \in \mathbb{R}^{2\times 2}$ and $\boldsymbol{M}_{\theta\delta} \in \mathbb{R}^{2\times n}$. Moreover, the Coriolis and centrifugal vector $\boldsymbol{c}(\boldsymbol{q}, \dot{\boldsymbol{q}})$ can be rewritten in the factorized form

$$\boldsymbol{c}(\boldsymbol{q}, \dot{\boldsymbol{q}}) = \boldsymbol{S}(\boldsymbol{q}, \dot{\boldsymbol{q}})\dot{\boldsymbol{q}} = \begin{pmatrix} \boldsymbol{S}_{\theta\theta} & \boldsymbol{S}_{\theta\delta} \\ \boldsymbol{S}_{\delta\theta} & \boldsymbol{S}_{\delta\delta} \end{pmatrix}, \tag{12}$$

where the (non-symmetric) matrix $\boldsymbol{S}$ is obtained by symbolic derivation of the elements of the inertia matrix $\boldsymbol{M}$ using the Christoffel's symbols [13] and is such that $\dot{\boldsymbol{M}} - 2\boldsymbol{S}$ is skew-symmetric (or, equivalently, $\dot{\boldsymbol{M}} = \boldsymbol{S} + \boldsymbol{S}^T$). According to the inertia decomposition, we have the dimensions of the blocks $\boldsymbol{S}_{\theta\theta} \in \mathbb{R}^{2\times 2}$, $\boldsymbol{S}_{\theta\delta} \in \mathbb{R}^{2\times n}$, $\boldsymbol{S}_{\delta\theta} \in \mathbb{R}^{n\times 2}$ and $\boldsymbol{S}_{\delta\delta} \in \mathbb{R}^{n\times n}$.

The generalized momentum $\boldsymbol{p}$ of the flexible manipulator is defined (and decomposed similarly) as:

$$\boldsymbol{p} = \begin{pmatrix} \boldsymbol{p}_\theta \\ \boldsymbol{p}_\delta \end{pmatrix} = \boldsymbol{M}(\boldsymbol{q})\dot{\boldsymbol{q}} \;\Rightarrow\; \begin{aligned} \boldsymbol{p}_\theta &= \boldsymbol{M}_{\theta\theta}\dot{\boldsymbol{\theta}} + \boldsymbol{M}_{\theta\delta}\dot{\boldsymbol{\delta}} \\ \boldsymbol{p}_\delta &= \boldsymbol{M}_{\theta\delta}^T\dot{\boldsymbol{\theta}} + \boldsymbol{I}\dot{\boldsymbol{\delta}} \end{aligned}, \tag{13}$$

Its time derivative is computed as:

$$\begin{aligned}\dot{\boldsymbol{p}} &= \boldsymbol{M}\ddot{\boldsymbol{q}} + \dot{\boldsymbol{M}}\dot{\boldsymbol{q}} = \\ &= \left(\boldsymbol{G}(\boldsymbol{u} + \boldsymbol{u}_F) - \boldsymbol{S}\dot{\boldsymbol{q}} - \boldsymbol{K}\boldsymbol{q} - \boldsymbol{D}\dot{\boldsymbol{q}}\right) + \left(\boldsymbol{S} + \boldsymbol{S}^T\right)\dot{\boldsymbol{q}} \\ &= \boldsymbol{S}^T\dot{\boldsymbol{q}} - \boldsymbol{D}\dot{\boldsymbol{q}} - \boldsymbol{K}\boldsymbol{q} + \boldsymbol{G}(\boldsymbol{u} + \boldsymbol{u}_F).\end{aligned} \tag{14}$$

In particular, we can split eq. (14) in

$$\begin{aligned}\dot{\boldsymbol{p}}_\theta &= \boldsymbol{S}_{\theta\theta}^T\dot{\boldsymbol{\theta}} + \boldsymbol{S}_{\delta\theta}^T\dot{\boldsymbol{\delta}} + \boldsymbol{u} + \boldsymbol{u}_F \\ \dot{\boldsymbol{p}}_\delta &= \boldsymbol{S}_{\theta\delta}^T\dot{\boldsymbol{\theta}} + \boldsymbol{S}_{\delta\delta}^T\dot{\boldsymbol{\delta}} - \boldsymbol{D}_\delta\dot{\boldsymbol{\delta}} - \boldsymbol{K}_\delta\boldsymbol{\delta} + \boldsymbol{G}_\delta(\boldsymbol{u} + \boldsymbol{u}_F).\end{aligned} \tag{15}$$

Although a 'complete' residual $\boldsymbol{r} \in \mathbb{R}^{2+n}$ could be designed, for our purposes it is sufficient to consider only the dynamics of $\boldsymbol{p}_\theta$ for the definition of an useful residual. We define $\boldsymbol{r}_\theta \in \mathbb{R}^2$, with as many elements as the number of robot joints, as

$$\boldsymbol{r}_\theta(t) = \boldsymbol{K}_r \left( \boldsymbol{p}_\theta - \int_0^t \left( \boldsymbol{S}_{\theta\theta}^T\dot{\boldsymbol{\theta}} + \boldsymbol{S}_{\delta\theta}^T\dot{\boldsymbol{\delta}} + \boldsymbol{u} + \boldsymbol{r}_\theta \right) d\tau \right), \tag{16}$$

where $\boldsymbol{K}_r > 0$ is a $2 \times 2$ diagonal gain matrix. It can be shown [9], [12] that the dynamics of the residual is

$$\dot{\boldsymbol{r}}_\theta = \boldsymbol{K}_r(\boldsymbol{u}_F - \boldsymbol{r}_\theta). \tag{17}$$

Thus, the residual $\boldsymbol{r}_\theta(t)$ asymptotically recovers the value of $\boldsymbol{u}_F$ (when constant) or follows closely the dynamics of the fault/collision torque.

Faults and collisions are detected when the residual signal exceeds a given threshold. Moreover, a simple rule that also isolates the fault/collision is as follows:

- **actuator faults**: the fault is located at the joint where the residual signal exceeds the threshold;
- **collisions**: the collision is located on the last link (downstream from base to end effector) whose corresponding joint has a residual exceeding the threshold.

This heuristic rule for isolation relies on the fact that actuator faults and collisions affect the components of the residual vector in an intrinsically different way, as will be shown in the next section.

## IV. Simulation Results

We have simulated our detection and isolation method considering single or concurrent actuator faults and unexpected collisions. The chosen platform was the planar robot with two joints and a flexible link presented in Sec. II, with only the first two deflection modes $\delta_1(t)$ and $\delta_2(t)$ included the series expansion (1). The robot parameters used for the simulations are those reported in [11].

To generate a torque input for a robot motion task, we chose to control the clamped joint angles

$$\boldsymbol{\theta}_c = \begin{pmatrix} \theta_{c1} \\ \theta_{c2} \end{pmatrix} = \begin{pmatrix} \theta_1 \\ \theta_2 + \phi'_{i0}\delta_1 + \phi'_{20}\delta_2 \end{pmatrix} \tag{18}$$

along a desired trajectory $\boldsymbol{\theta}_{c,\text{des}}(t)$ by means of a simple decentralized PD law

$$\boldsymbol{u} = \boldsymbol{K}_P\left(\boldsymbol{\theta}_{c,\text{des}} - \boldsymbol{\theta}_c\right) + \boldsymbol{K}_D\left(\dot{\boldsymbol{\theta}}_{c,\text{des}} - \dot{\boldsymbol{\theta}}_c\right), \tag{19}$$

with gains

$$\boldsymbol{K}_P = \text{diag}\begin{pmatrix} 3 & 1 \end{pmatrix} > 0, \quad \boldsymbol{K}_D = \text{diag}\begin{pmatrix} 1.5 & 1 \end{pmatrix} > 0. \tag{20}$$

The value of $\dot{\boldsymbol{\theta}}_c$ in (19) is computed from (18), using the state variables $\dot{\theta}_1$, $\dot{\theta}_2$, $\dot{\delta}_1$ and $\dot{\delta}_2$.

The results of four illustrative case studies are reported next. In the first three, actuator faults are simulated, while in the last one we applied an external contact force on link 1 and on link 2 during different intervals of time. In all cases, the desired joint trajectories have been chosen as

$$\theta_{c1,\text{des}}(t) = 2\sin(0.1\pi t), \quad \theta_{c2,\text{des}}(t) = 2\sin(0.2\pi t). \tag{21}$$

A fault/collision event is detected when the following threshold condition is valid:

$$\exists\, i \in \{1, 2\} \quad \text{s.t.} \quad |r_i| \geq r_{\text{th}} = 0.01\,[\text{Nm}]. \tag{22}$$

### A. Actuator faults

In the first case study, a fault on the first motor (moving the rigid link) occurs at time $t_{F_1} = 5$ s, when the motor abruptly loses 90% of its effectiveness. In Fig. 3, it is clear that the local PD law at joint 1 is no longer able to make $\theta_{c1}$ follow its desired trajectory after the fault has occurred. The top plot in Fig. 4 shows the actual torque that the first motor delivers to the robot (red line) and the commanded torque computed by the controller (blue line).

The residual vector $\boldsymbol{r}$, computed according to (16), is shown in Fig. 5. Indeed, the fault is perfectly detected and

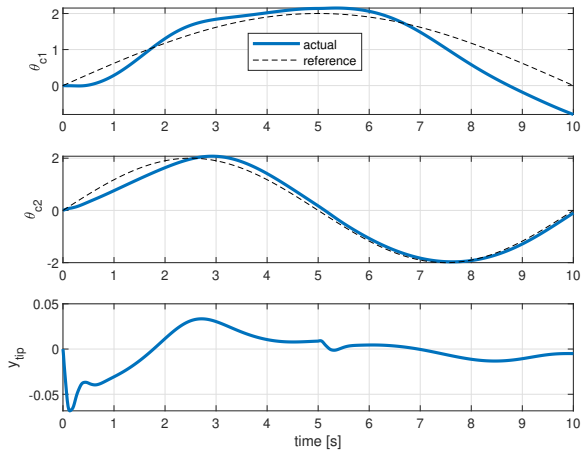


Fig. 3. Case study 1. System outputs with a fault on first motor.

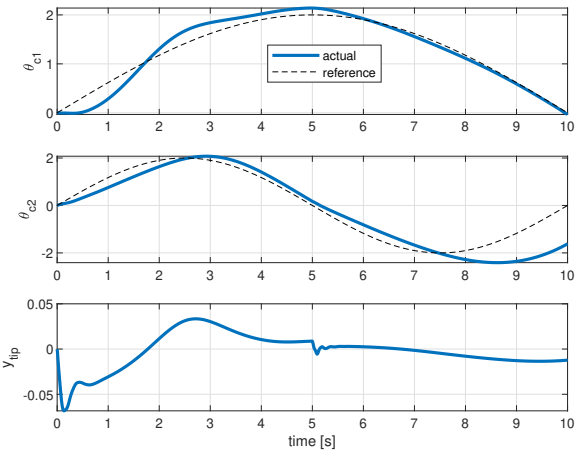


Fig. 6. Case study 2. System outputs with a fault on second motor.

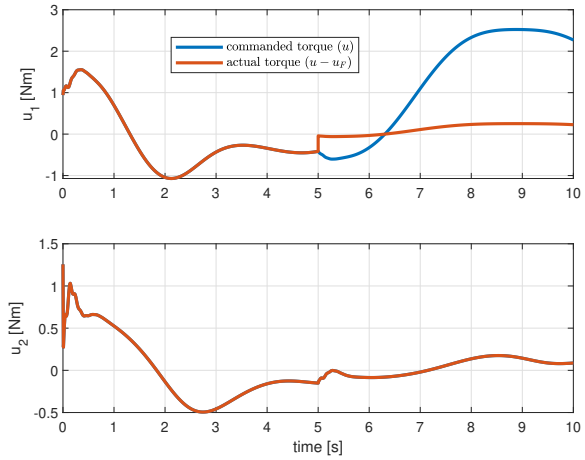


Fig. 4. Case study 1. Commanded vs. actual torques.

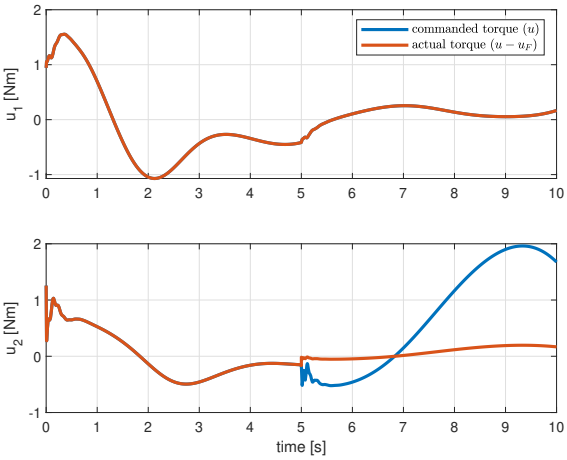


Fig. 7. Case study 2. Commanded vs. actual torques.

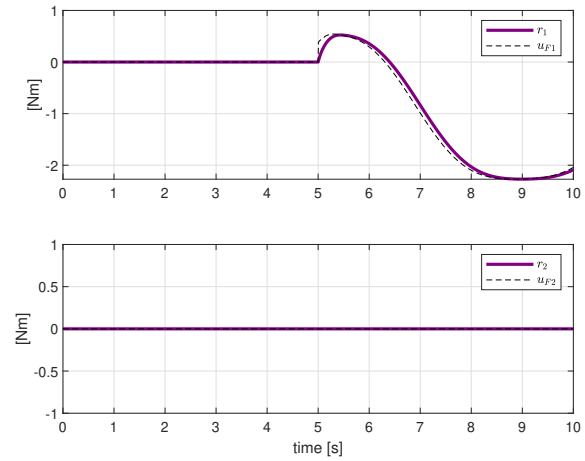


Fig. 5. Case study 1. Residuals with a fault on first motor.

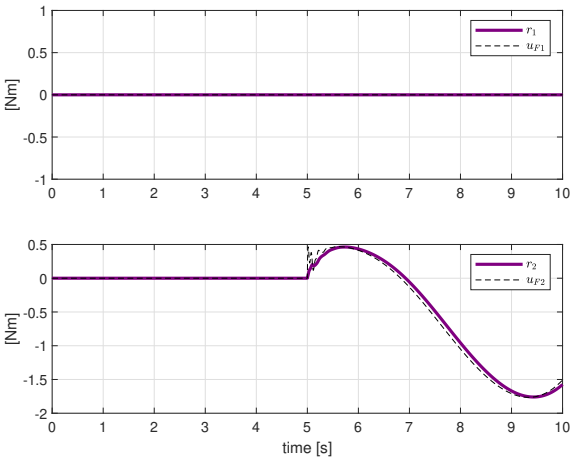


Fig. 8. Case study 2. Residuals with a fault on second motor.

isolated. In fact, $\boldsymbol{r}(t) = 0$ as long as both motors are fully operational, while the residual component $r_1$ (purple line) increases in magnitude and exceeds the threshold $r_{\text{th}}$ as soon as the motor loses power, allowing the monitoring rule (22) to detect the fault. Moreover, the fault is easily isolated at the first joint, due to the fact that only $|r_1|$ is above threshold. In addition, note that the residual vector, as stated in eq. (17), asymptotically converges componentwise to the fault torque $\boldsymbol{u}_F = \boldsymbol{u} - \boldsymbol{u}_{\text{act}}$ (see the dashed black line in Fig. 5 for joint 1), where $\boldsymbol{u}$ is the torque commanded by the controller and

$\boldsymbol{u}_{\text{act}}$ is the actual torque delivered by the actuators.

The second case study is a dual case with respect to first one. At time $t_{F_2} = 5$ s, the second motor becomes faulty, loosing instantaneously 90% of its power. The motion of the relevant outputs of the robot is shown in Fig. 6, while Fig. 7 compares the commanded torque (blue line) with the actual torque (red line). As before, the fault is perfectly detected and isolated ($|r_2| > r_{\text{th}}$, whereas $r_1 = 0$), as shown in Fig. 8.

The third case refers to a partly concurrent failure of both actuators. Starting from time $t_{F_1} = 5$ s, the first motor actuates only 80% of the commanded torque; the second motor exhibits the same type of faulty behavior, but starting from time $t_{F_2} = 10$ s (see Fig. 10). It is worth to note that, despite the deviation of the outputs from their references is rather limited (Fig. 9) and the missing torques are quite small (Fig. 10), the residual in Fig. 11 allows to properly detect and isolate the faults on each single motor (in a decoupled way).

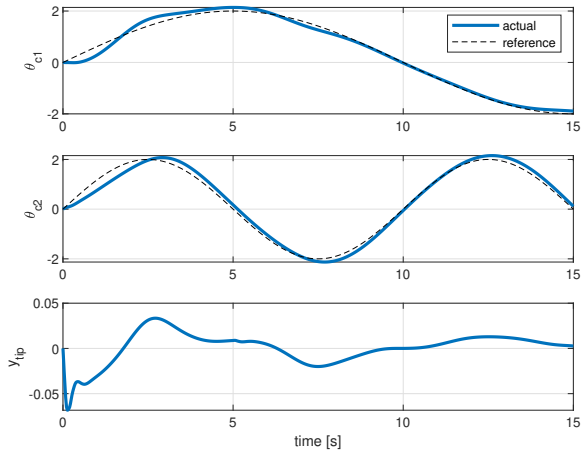


Fig. 9. Case study 3. System outputs with faults on both motors.

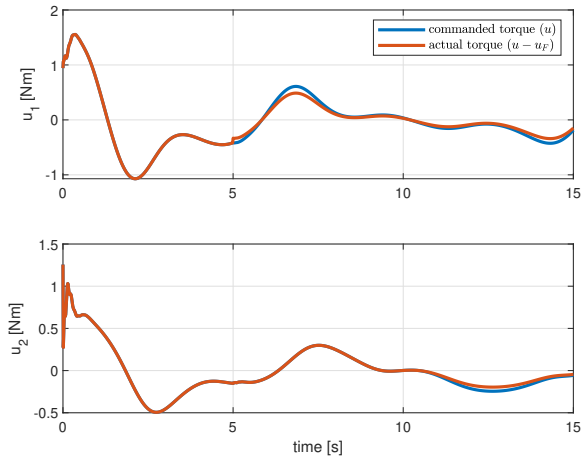


Fig. 10. Case study 3. Commanded vs. actual torques.

### B. Collision on both links

The residual vector signal (16) returns a reliable estimation of the external torques (subtracted or added to the commanded ones) that are acting at the level of the robot joints. These 'external' torques are detected when an actuator

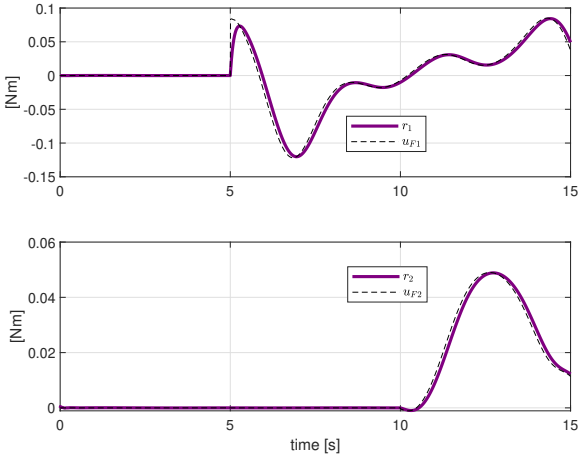


Fig. 11. Case study 3. Residuals with faults on both motors.

fails (in multiple ways, as seen in the previous examples) or when a unknown force is exerted on the robot links from the environment (e.g., in case of an unexpected collision). In this fourth case study, an external force $\boldsymbol{F}_{\text{ext}} = (1\ 1)^T$, acting in the plane of robot motion, is separately applied to both links, in particular at the origin of the two frames $\boldsymbol{o}_1$ and $\boldsymbol{o}_{2e}$ (see Fig. 1). The force $\boldsymbol{F}_{\text{ext}}$ is applied to link 1 during the time interval from $t_{F_{1,\text{init}}} = 10$ s to $t_{F_{1,\text{fin}}} = 12$ s, while the same force is applied to link 2 between $t_{F_{2,\text{init}}} = 25$ s and $t_{F_{2,\text{fin}}} = 27$ s. These external forces produce torques $\boldsymbol{u}_{\text{ext}}$ that act on the robot joints according to the usual formula

$$\boldsymbol{u}_{\text{ext}} = -\boldsymbol{J}_P(\boldsymbol{q})^T \boldsymbol{F}_{\text{ext}}, \tag{23}$$

where $\boldsymbol{J}_P(\boldsymbol{q})$ is the robot Jacobian relative to the actual position of the contact point $P$ along the kinematic chain, which depends in turn on the current robot configuration $\boldsymbol{q}$.

The three system outputs $\theta_{c1}$, $\theta_{c2}$, and $y_{\text{tip}}$, together with the control reference values for the first two, are shown in Fig. 12, while the commanded and actual torques acting on the joints are reported in Fig. 13. The behavior of the residual vector in Fig. 14 confirms that the first collision (on link 1) affects only the residual at the first joint, while the effect of the second collision (on link 2) propagates over both joints. Therefore, even in this case, it is possible to distinguish the two instances of contact.

*Remark 1:* Although with the residual analysis it is possible to isolate the link on which a contact occurred, it is typically unfeasible (without an extra sensor) to derive from the residual an accurate information about where the collision occurred on the link. This is because both the external force $\boldsymbol{F}_{\text{ext}}$ and the Jacobian $\boldsymbol{J}_P(\boldsymbol{q})$ are unknown.

## V. Conclusions

In this paper, we addressed the problem of detection and isolation of actuator faults and of collisions for a robot with flexible links, without the need of any dedicated exteroceptive sensors. To this end, we have extended to flexible robots the mathematical formulation of the momentum-based residual, a reliable method already widely used in rigid robots, so as to improve safety in human-robot physical interaction and collaboration. By separating the residual in

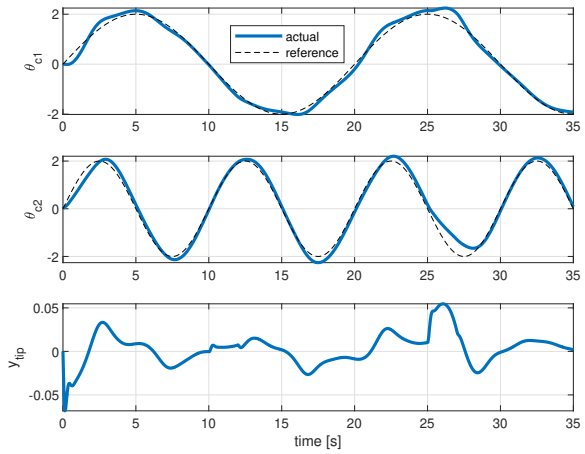


Fig. 12. Case study 4. System outputs with collisions on both links.

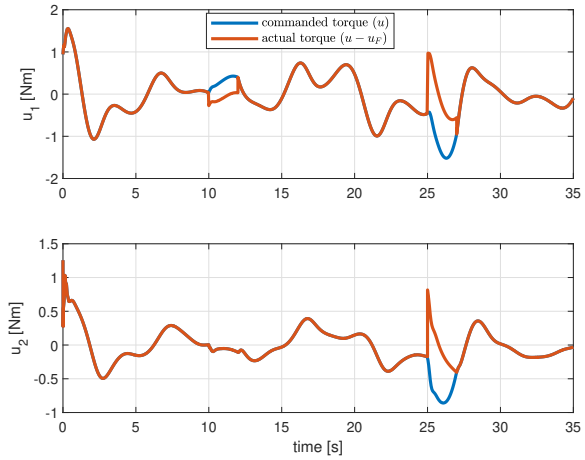


Fig. 13. Case study 4. Commanded vs. actual torques.

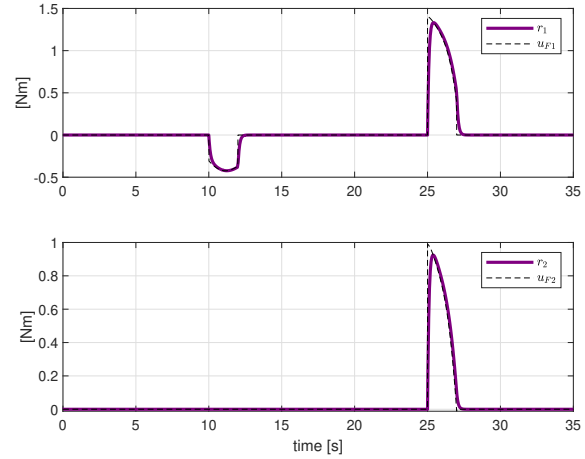


Fig. 14. Case study 4. Residuals with collisions on both links.

two different parts, we exploited for our purposes only the component pertaining to the rigid dynamics. We have tested our algorithm through extensive simulations performed on the Flexarm, a two-link planar manipulator having a rigid upper link and a flexible forearm, with arbitrarily assigned reference trajectories for the joints. Thanks to the extended version of the residual, we were able to effectively detect and isolate actuator faults as well as link collisions. Moreover, faults and collisions can be distinguished according to a simple rule, by reasoning on the joints whose residual signal exceeds a given threshold. In the context of the present paper, since only simulation results are reported and the dynamic model is assumed to be known, the threshold has been set to zero for all joints; nevertheless, a crucial task to be considered in possible real applications is the proper tuning of such threshold. In fact, when model uncertainties or disturbances are present, the residual may diverge from zero even if no faults or collisions occur.

Although relatively simple, the Flexarm platform is already a challenging benchmark for our purposes. As a consequence, the proposed detection/isolation method is expected to be valid for any robot with flexible links, provided a good knowledge of its dynamic model parameters is available.

In the present work, we assumed that the full system state is accessible for measure. However, this situation rarely occurs in practice, especially when dealing with robots with flexible links, for which a direct measure of the deflection modes is often unavailable. To overcome this issue, we are currently working on an extended version of the method that integrates a suitable nonlinear state observer.

## REFERENCES


[1] 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.

[2] 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.

[3] 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.

[4] A. De Luca and L. Lanari, "Achieving minimum phase behavior in a one-link flexible arm." in *Proc. Int. Symposium on Intelligent Robots*, 1991, pp. 224–235.

[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] 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.

[10] C. Gaz, E. Magrini, and A. De Luca, "A model-based residual approach for human-robot collaboration during manual polishing operations," *Mechatronics*, vol. 55, pp. 234–247, 2018.

[11] 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.

[12] 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.

[13] B. Siciliano, L. Sciavicco, L. Villani, and G. Oriolo, *Robotics: Modeling, Planning and Control*, 3rd ed. London: Springer, 2008.