A discrete-time framework for fault diagnosis in robotic manipulators

Fabrizio Caccavale†, Senior Member, IEEE, Alessandro Marino†, Member, IEEE, Giuseppe Muscio‡, and Francesco Pierrir, Member, IEEE

Abstract—In this paper, a discrete-time framework for diagnosis of faults of joint sensors, wrist-mounted force/torque sensors and actuators of robotic manipulators is devised. It is assumed that redundant joint sensor measurements are available. Sensor measurements, together with the estimates computed by two isolation observers, are processed by a decision making system, providing detection and isolation of the faults of the joint sensors as well as healthy measurements. Then, healthy measurements are used to feed a bank of diagnostic observers aimed at detecting, isolating and identifying faults of joint actuators and force/torque sensors. The framework is experimentally tested on a cooperative industrial setup, composed by two six-degrees-of-freedom industrial robots performing a cooperative task.

Index Terms—Fault diagnosis, manipulators, observers, adaptive estimation.

I. INTRODUCTION

Robotic manipulators are often employed in scenarios requiring a high degree of autonomy (e.g., space and underwater missions) and/or a tight interaction with humans (e.g., robotic co-workers in manufacturing systems). In such scenarios, failures could result in very critical and unpredictable behavior of the robotic system, thus leading to unsafe operations. Therefore, in order to guarantee suitable levels of safety and reliability, early diagnosis of failures is of the utmost importance, making crucial to develop Fault Diagnosis (FD) systems. The main goal of a FD scheme is to monitor the robotic system so as to detect the occurrence of faults (fault detection), recognize the faulty components (fault isolation) and identify the fault time evolution (fault identification) [1].

Fault diagnosis methodologies based on the concept of analytical redundancy, exploit functional relationships between the variables of the system [2], [3]. Among the analytical redundancy approaches, the most widely adopted are the observer-based methods [1], [4], [5], [6], based on the adoption of diagnostic observers to estimate a set of relevant variables characterizing the system. Then, a set of variables sensitive to the occurrence of a failure (the so-called residuals) are computed on the basis of the measured variables and those predicted via the diagnostic observer. In order to counteract the effect of model uncertainties and disturbances, adaptive and/or robust approaches have been proposed for nonlinear systems [5], [7], [8].

Early approaches to FD for robotic systems were based on model parameter estimation [9], [10], while, more recently, approaches based on learning and adaptive methodologies [11], discrete-time observers [12], [13], [14], [15], robust [16], [17], [18] and adaptive [19] observers, as well as on the so-called generalized momenta [20], have been proposed. Also, soft computing methods have been developed to improve the performance of FD schemes for uncertain systems ([21], [22] and references therein). In [23] joint torque sensors are exploited to achieve notable robustness properties of the FD scheme against unmodeled dynamics.

The above recalled approaches are usually focused on either failures affecting the sensors or those occurring at the actuators (eventually including unexpected collisions). At the best of authors’ knowledge, only in [12], [13], [17], [23] diagnosis schemes capable of detecting and isolating both sensor and actuator failures have been considered. The results in [12], [13] do not consider, however, physical interaction of the manipulators with the external environment and/or with other robots (e.g., cooperative manipulators, robotic hands). In [15], a diagnosis scheme for robotic manipulators interacting with the external environment and/or with other manipulators is developed, but only sensor faults are considered. Moreover, differently from [17] and [23], the proposed framework does not resort to either joint torque sensors or vision sensors.

In this paper, a general framework for fault diagnosis in robotic manipulators, eventually interacting with the external environment, other robots or humans is devised. It is assumed that each manipulator is equipped with both proprioceptive sensors (i.e., joint sensors providing position and velocity measurements) and a wrist-mounted force/torque sensor. The approach takes into account faults affecting the joint sensors, the force/torque sensor and the actuators (i.e., joint motors and their driving systems). Redundant joint sensors have been considered, i.e., it is assumed that, at each joint, a couple of identical sensors provides both joint position and joint velocity readings. Redundant measurements, together with the adoption of suitably designed diagnostic observers, are used for proprioceptive sensor fault detection and isolation. Then, an analytical redundancy method is adopted to perform detection, isolation and identification of faults affecting the force/torque sensor and the actuators.

In detail, a couple of diagnostic observers is adopted, in conjunction with redundant measurements, in order to obtain proprioceptive sensor fault isolation. A Decision Making System (DMS) votes a healthy measure, that is used to feed a bank of observers aimed at detecting, isolating and identifying
The state space equations are then given by the term interaction forces at the joints. Modeling uncertainties and \( \varsigma \) end-effector, matrix, \( C \) vector \( g \) in the presence of unmodeled dynamics and disturbances.

It is worth remarking that, since the proposed approach takes into account not only position sensors but also force sensors, it allows fault diagnosis for robotic manipulators interacting with the external environment and other manipulators (e.g., cooperative multi-manipulator systems). Hence, the developed framework can be applied in more complex and realistic operating scenarios. It is worth remarking that the adoption of redundant sensors is widely accepted in application scenarios requiring a high level of autonomy and/or reliability (planetary exploration, underwater missions, service robotics), while it is not considered cost-effective in industrial settings; however, duplication of some critical components will become a viable choice in future industrial scenarios requiring high levels of safety (e.g., involving close interaction between robots and humans, according to the so-called robotic co-worker concept) or in medical applications. Moreover, duplication of proprioceptive sensors allows voting of a healthy measure to be used for the diagnosis of force/torque sensor faults and actuator faults; also, the voted healthy measure can be used to feed the motion control loops.

The proposed FD scheme is experimentally tested on an industrial setup composed by two six-degrees-of-freedom COMAU industrial robots grasping and manipulating a rigid cardboard box by means of an impedance control scheme [25]. The results demonstrate the effectiveness of the approach even in the presence of unmodeled dynamics and disturbances.

II. Modeling

The dynamic model of an open chain \( n \) degree-of-freedom manipulator can be written as

\[
M(q)\ddot{q} + C(q, \dot{q})\dot{q} + F \ddot{q} + g(q) + \varsigma(q, \dot{q}, \tau) = \tau - J^T(q)h,
\]

where \( q \in \mathbb{R}^n \) (\( \dot{q} \) and \( \ddot{q} \)) is the joint positions (velocities and accelerations) vector, \( \tau \in \mathbb{R}^{n_T} \) is the joint torques vector, \( M(q) \in \mathbb{R}^{n \times n} \) is the symmetric and positive definite inertia matrix, \( C(q, \dot{q}) \in \mathbb{R}^{n \times n} \) is the centrifugal and Coriolis term matrix, \( F \in \mathbb{R}^{n \times n} \) is the matrix modeling viscous friction and \( g(q) \in \mathbb{R}^n \) is the vector of gravity terms. The \( (6 \times 1) \) vector \( h \in \mathbb{R}^6 \) collects the generalized forces acting at the end-effector, \( J(q) \in \mathbb{R}^{6 \times n} \) is the geometric Jacobian and the term \( J^T(q)h \) represents the effect of the generalized interaction forces at the joints. Modeling uncertainties and disturbances (e.g., low velocity friction, motor electromagnetic disturbances, noise) are explicitly taken into account by the term \( \varsigma(q, \dot{q}, \tau) \in \mathbb{R}^n \). In order to represent the above system in the state-space form, a suitable choice for the state variables is the following

\[
x = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} q \\ \dot{q} \end{bmatrix} \in \mathbb{R}^{2n},
\]

The state space equations are then given by

\[
\begin{align*}
\dot{x} &= A_e x + B_e(x)u + n_e(x) + V_e(x)h + \eta_e(x, u) \\
y &= x,
\end{align*}
\]

where \( u = \tau \) and \( y \) is the output vector, which is assumed to coincide with the state. The matrices \( A_e, B_e \) and \( V_e \) in (2) are defined as follows:

\[
A_e = \begin{bmatrix} \mathbf{O}_{n \times n} & I_{n \times n} \\ \mathbf{O}_{n \times n} & \mathbf{O}_{n \times n} \end{bmatrix}, \quad B_e(x) = \begin{bmatrix} \mathbf{O}_{n \times n} \\ \mathbf{O}_{n \times n} \end{bmatrix}, \quad V_e(x) = \begin{bmatrix} \mathbf{O}_{n \times 6} \\ \mathbf{-M}^{-1}(x_1) J^T(x_1) \end{bmatrix},
\]

where \( I_{n \times n} \) and \( \mathbf{O}_{n \times m} \) denote, respectively, the identity matrix and the null matrix of dimension \( n \times m \). The other terms in (2) are given by

\[
n_e(x) = \begin{bmatrix} 0_n \\ -\mathbf{M}^{-1}(x_1) (C(x_1, x_2) x_2 + F x_2 + g(x_1)) \end{bmatrix},
\]

\[
\eta_e(x, u) = \begin{bmatrix} 0_n \\ -\mathbf{M}^{-1}(x_1) \varsigma(x, u) \end{bmatrix},
\]

where \( 0_n \) is the \( n \times 1 \) null vector.

The control system of a robotic manipulator is implemented in discrete-time. Hence, a digital implementation of the whole diagnostic scheme must be considered as well. As usual in digital control systems, it is assumed that the measurements of the output variables and the generalized forces acting at the end-effector are sampled at fixed sampling time \( T \), while the input torques are assumed to be constant over each sampling interval \([kT, (k+1)T]\), where \( k \) denotes the positive integer discrete-time index. Thus, it is worth looking for a discrete-time equivalent model of (2). A simple discrete-time model, based on the first-order Euler method [26], can be devised

\[
\begin{align*}
x(k+1) &= \mathbf{A}x(k) + \mathbf{B}(x(k))u(k) + n(x(k)) + V(x(k))h(k) + \eta(x(k), u(k)) \\
y(k) &= x(k),
\end{align*}
\]

where \( \mathbf{A} = \mathbf{I}_{2n \times 2n} + T \mathbf{A}_e, \quad \mathbf{B} = T \mathbf{B}_e, \quad n = T n_e, \quad V = TV_e, \quad \eta = T \eta_e + \epsilon_e \) and \( \epsilon_e \) is the vector collecting the errors due to discretization of the continuous-time dynamics.

The Euler method is characterized by a global accumulated error of the same order of magnitude as \( T \) [26]. More accurate discrete-time models might be derived. However, such models are either unsuitable for direct dynamics (i.e., computation of motion from torque inputs) [27] or too complex and computationally demanding than the Euler method. On the other hand, the discretization error can be seen as a model uncertainty to be counteracted by the use of properly tuned state observers and uncertainty interpolators, as described in the following. Thus, additional complexity introduced by more accurate discretization approaches is unnecessary. Finally, it is worth pointing out that the choice of \( T \) affects the effectiveness of the diagnosis scheme, since the smaller is \( T \) the smaller is the discretization error \( \epsilon_e \).

A. Fault modeling

Sensor faults can affect both proprioceptive sensors (i.e., sensors providing joint position and velocity) and the force/torque sensor mounted at the manipulator’s wrist, measuring the generalized forces acting at the end-effector. A
sensor fault occurs when the sensor readings do not coincide with the true values of the variables.

In detail, a fault on a joint position or velocity sensor can be modeled as an additive unknown term in the output equation, i.e.,

$$y(k) = x(k) + \delta x(k),$$  \hspace{1cm} (4)

where $\delta x$ is the unknown fault vector of the form

$$\delta x(k) = \begin{bmatrix} \delta x_1(k) \\ \delta x_2(k) \end{bmatrix} = \begin{bmatrix} \delta \theta(k) \\ \delta \dot{\theta}(k) \end{bmatrix}.$$ \hspace{1cm} (5)

It is worth noticing that $\delta x$ in (4) includes sensor noise as well. Hence, sensor readings are always affected by faults, since noise is always superimposed to the measured variable in real sensors. According to the user requirements, the distinction between noise and faults is established by setting suitable thresholds on the relevant decision variables (i.e., the residuals): in other words, the noise itself might become a fault if it exceeds a level considered acceptable by the user, where such a level is indirectly fixed by choosing the thresholds.

Similarly, a fault occurring on the force/torque sensor mounted at the manipulator’s wrist can be modeled as an unknown term added to the true force/torque vector $h$, i.e.,

$$\overrightarrow{h}(k) = h(k) + \delta h(k),$$ \hspace{1cm} (6)

where $\delta h$ is the unknown fault vector.

Actuator faults for robot manipulators can be defined as the failures occurring either in the joint motors or in the corresponding mechanical transmission. This class of failures can be represented as an unknown additive term, $\delta u$, on the commanded torques, hereafter referred as nominal torques, $\overrightarrow{u}$, i.e.,

$$u(k) = \overrightarrow{u}(k) + \delta u(k).$$ \hspace{1cm} (7)

The manipulator dynamics (3), in the presence of faults, becomes

$$\begin{cases} x(k+1) = Ax(k) + B(x(k))\overrightarrow{u}(k) + n(x(k)) + V(x(k))\overrightarrow{h}(k) + \eta(k, x(k), u(k)) + f_h(k) + f_u(k) \\ y(k) = x(k) + \delta x(k), \end{cases}$$ \hspace{1cm} (8)

where $f_h$ and $f_u$ represent the effects on the system dynamics of faults affecting the force/torque sensor and the actuators, respectively,

$$f_h(x, k) = -V(x(k))\delta h(k),$$ \hspace{1cm} (9)

$$f_u(x, k) = B(x(k))\delta u(k).$$ \hspace{1cm} (10)

It is worth noticing that actuator and force/torque sensor faults affect directly only the dynamics of the last six components of the state vector (i.e., the state variable $x_2$), since the first six components of $f_h$, $f_u$ and $\eta_c$ are null.

### III. Fault Diagnosis Scheme

The FD scheme is devised by assuming that two independent measures of the whole state are available, i.e., both joint position and velocity sensors are duplicated. Moreover, the presence of a force/torque sensor (not duplicated) measuring the generalized forces acting on the end-effector is considered.

In detail, two sets of sensors (hereafter labeled as $S^1$ and $S^2$) are present, each of them including $n$ position and $n$ velocity sensors. Therefore $S^j_i$ ($j = 1, 2$ and $i = 1, \ldots, 2n$) will denote the $i$th sensor of the $j$th set. Sensor redundancy is used in conjunction with two diagnostic observers in order to achieve fault detection and isolation. A suitably designed Decision Making System (DMS) detects the occurrence of a fault, isolates the faulty sensor and votes the healthy measure used for actuator and force sensor fault diagnosis. Noticeably, voting of a healthy measure is possible thanks to the duplication of proprioceptive sensor.

The healthy measure feeds a bank of $n + 7$ observers, providing fault detection and isolation for the force/torque sensor and the actuators. The first observer plays the role of detection observer, i.e., determines the occurrence of actuators and/or force/torque sensor faults. Then, a bank of $n$ observers, one for each joint, is used for isolation and identification of actuator faults, while 6 observers (one for each component of the sensor output: 3 force components and 3 moment components) are aimed at isolating and identifying force/torque sensor faults.

In order to decouple sensor fault diagnosis from faults affecting the force/torque sensor and the actuators, an estimate of the fault obtained by the bank of isolation observers is used into the proprioceptive sensor isolation observers.

Figure 1 shows a block diagram representation of the overall architecture.

### IV. Fault diagnosis for proprioceptive sensors

Thanks to sensor redundancy, two vectors of independently measured data are available: $y_{S^1}$, given by the sensors in set $S^1$, and $y_{S^2}$, given by the sensors in set $S^2$. Fault detection can be directly achieved by comparing the measures of the same output variable provided by the duplicated sensors $y_{S^1}$ and $y_{S^2}$; hence, $2n$ detection residuals can be defined

$$r_{S^i} = y_{S^i} - y_{S^j}, \hspace{1cm} i = 1, \ldots, 2n.$$ \hspace{1cm} (11)

If a sensor reading is affected by a fault, the value of the corresponding residual, $r_{S^i}$, is expected to be non-zero, otherwise it is expected to be zero. In practice, the residuals are always non-zero due to the presence of random measurement noise on sensor readings.

In order to define a set of isolation residuals, two observers are designed: the first observer is fed by the output of the first set of sensors, $y_{S^1}$, while the second observer uses the output of the second set of sensors, $y_{S^2}$. Both the observers have the following form ($j = 1, 2$)

$$\begin{cases} \hat{\bar{y}}_{S^j}(k+1) = A\hat{\bar{y}}_{S^j}(k) + B(y_{S^j}(k))\overrightarrow{u}(k) + n(y_{S^j}(k)) + V(y_{S^j}(k))\overrightarrow{h}(k) + \tilde{\eta}(y_{S^j}(k), \hat{\bar{y}}_{S^j}(k), \overrightarrow{u}(k)) + K_s\tilde{e}_{S^j}(k) \\ \hat{y}_{S^j}(k) = \hat{\bar{y}}_{S^j}(k), \end{cases}$$ \hspace{1cm} (12)
where \( \hat{\eta} \) denotes the estimated variables, \( e_{S_j} = y_{S_j} - \hat{y}_{S_j} \) is the output estimation error and the matrix gain \( K_o \in \mathbb{R}^{2n \times 2n} \) is chosen such that \( F = A - K_o \) has all its eigenvalues in the unit circle. The term \( \hat{\eta}_{S_j} \) is an estimate of the vector of uncertainties, \( \eta \). In view of (8) and (12), the estimation error dynamics, in the presence of only proprioceptive sensor faults (i.e., \( y_{S_j} = x + \delta x, \ u = \bar{u} \) and \( h = \bar{h} \)), is given by

\[
e_{S_j}(k+1) = Fe_{S_j}(k) + \hat{\eta}_{S_j}(k) + f_{S_j}(k),
\]

where \( \hat{\eta}_{S_j} = \eta - \tilde{\eta}_{S_j} \) and

\[
f_{S_j}(k) = \delta x_{S_j}(k+1) - A \delta x_{S_j}(k) + \left[ B(x(k)) - B(y_{S_j}(k)) \right] u(k) + \left[ n(x(k)) - n(y_{S_j}(k)) \right] + \left[ V(x(k)) - V(y_{S_j}(k)) \right] h(k).
\]

The isolation residual is the vector

\[
r_{S_j}(k+1) = e_{S_j}(k+1) - Fe_{S_j}(k) = \hat{\eta}_{S_j}(k) + f_{S_j}(k).
\]

Since \( r_{S_j} \) is affected only by \( \hat{\eta}_{S_j} \) and \( f_{S_j} \), if an accurate estimation of \( \eta \) is achieved, the residual will be mainly affected by the vector of fault signatures \( f_{S_j}(k) \).

A. Uncertainties estimation

In order to obtain an estimate, \( \hat{\eta}_{S_j} \), of \( \eta \), different approaches can be considered, including neural networks and recursive or adaptive estimation. If a sufficiently wide set of experimental data are available, a universal interpolator (e.g., a neural network) can be efficiently trained along a set of fault-free trajectories in such a way that its output accurately estimates the uncertainties vector [22]. A different way is the use of recursive algorithms [13] that allow to estimate the uncertainties on the basis of the observed error.

Several well-known results in the literature propose adoption of on-line interpolators [28] to obtain a good approximation of \( \eta \) via parametric models. Namely, a parametric model of \( \eta \) can be adopted by assuming that it depends linearly on a set of \( p \) unknown parameters collected in the vector \( \mu \), i.e.,

\[
\eta(k, x(k), u(k)) = \Phi(k, x(k), u(k))\mu + \epsilon(k),
\]

where the \( \Phi \) is the \((2n \times p)\) regressor matrix and \( \epsilon \) is the interpolation error. Of course, not all uncertainties can be rigorously characterized by a linear-in-the-parameters structure. However, this modeling assumption is not too restrictive, since it has been demonstrated that a wide class of functions can be effectively interpolated by a linear-in-the-parameters model [14], [28]. An estimate of \( \eta \) can be indirectly obtained through the adaptive estimation of \( \mu \), via the following update law [13], [14]

\[
\tilde{\mu}(k+1) = \tilde{\mu}(k) + \Phi^T(k) \Gamma_{\mu} r_{S_j}(k+1),
\]

where \( \Gamma_{\mu} = 2 (\Phi \Phi^T + Q)^{-1} \), \( Q \) is a positive definite symmetric matrix and the arguments \( x(k), u(k) \) in \( \Phi \) and \( \Gamma_{\mu} \) have been dropped for notation compactness. Matrix \( \Gamma_{\mu} \) is symmetric and positive definite, and its adoption corresponds to the so-called projection algorithm [29]. The dynamics of the state estimation error \( e_{S_j} \) and of the parameter estimation error, \( \tilde{\mu} = \mu - \mu \), are given by

\[
\begin{align*}
e_{S_j}(k+1) &= Fe_{S_j}(k) + \Phi(\tilde{\mu}(k) + \epsilon(k)) + f_{S_j}(k) \\
\tilde{\mu}(k+1) &= G(k)\tilde{\mu}(k) + H(k)(\epsilon(k) + f_{S_j}(k)),
\end{align*}
\]

where the arguments \( x(k), u(k) \) in \( \Phi, G \) and \( H \) have been dropped for notation compactness, \( G = I_p \times p - \Phi^T \Gamma_{\mu} \Phi \) and \( H = -\Phi^T \Gamma_{\mu} \).

A good estimate of \( \eta \) can be obtained via RBF networks that are known to be effective universal interpolators [24]. Namely, the estimate of the \( i \)-th component of \( \eta \) has the following structure

\[
\hat{\eta}_{S_j,i} = \sum_{h=1}^{N_\eta} \phi_{i,h} \tilde{\mu}_{i,h} = \phi_i^T \hat{\mu}_i, \quad i = 1, \ldots, 2n,
\]

where \( N_\eta \) is the number of radial basis functions, \( \tilde{\mu}_i = [\tilde{\mu}_{i,1} \ldots \tilde{\mu}_{i,N_\eta}]^T \) and \( \phi_i = [\phi_{i,1} \ldots \phi_{i,N_\eta}]^T \). The radial basis functions have the form

\[
\phi_{i,h}(x, u) = \exp \left( - \frac{||z - c_{i,h}||}{2\sigma^2} \right),
\]
where $z = [x^T\ u^T]^T \in \mathbb{R}^{3n}$, $c_{i,k}$ and $\sigma$ are the centroids and the width of the functions, respectively. According to the Universal Interpolation Theorem [24, 30], the continuous mapping $\eta_i(k, x(k), u(k))$ can be approximated (in the $L_p$-norm, $p \in [1, \infty]$) by a newtork $\tilde{\eta}_{S_i,k}(k, x(k), u(k))$ of the form (18) with suitably chosen centroids and a common width, provided that the basis functions $\phi_{i,k}(x(k), u(k))$ are continuous, bounded, integrable and with non-null integral over $\mathbb{R}^{3n}$. Since $\eta_i$ can be reasonably assumed continuous and the RBFs satisfy the conditions of the theorem, there exists a RBF network (18), i.e., suitable sets of weights, widths and centroids, capable of approximating $\eta_i$ to any degree of accuracy. Thus, the interpolation error $\epsilon_i$ in (18) can be reasonably assumed norm bounded, i.e., $\|\epsilon_i(k)\| \leq \tau_i$, $\forall k \in K$, where $K$ is the observation discrete-time interval. As for the choice of the centroids, they can be computed via a clustering algorithm [24] applied to a set of data collected along trajectories not affected by faults. The weights of the RBFs are, then, adaptively tuned online via the update law (16) with

$$
\Phi = \text{diag}\left\{\phi_1^T, \phi_2^T, \ldots, \phi_{2n}^T\right\} \quad \text{and} \quad \mu = \begin{bmatrix} \mu_1^T \mu_2^T \cdots \mu_{2n}^T \end{bmatrix}^T
$$

(i.e., $p = 2nN\eta$).

By adopting the same arguments used in [14], in the absence of faults (i.e., $r_{S_i}(k) = 0$), it is possible to prove that the error dynamics (17) is exponentially convergent to $0$ in the case of null interpolation error (i.e., $\epsilon_i(k) = 0$), provided that the Persistence of Excitation (PE) condition is satisfied [14], [29]:

$$
\beta_M I_{p \times p} \geq \sum_{l=k}^{k+N-1} \Phi(l)^T \Phi(l) \geq \beta_m I_{p \times p},
$$

(20)

for all $k$ and for some $\beta_M > \beta_m > 0$, $N > 0$. Although a reliable estimate of the convergence rate cannot be obtained, assessing exponential convergence is important to ensure that, in the presence of bounded interpolation error (i.e., $\|\epsilon_i(k)\| \leq \tau_i$), $r_{S_i}$ is guaranteed to be norm-bounded too, i.e., it exists $\overline{\eta}(\|\mu(k_0)\|) > 0$ such that

$$
\|r_{S_i}(k+1)\| = \|\tilde{\eta}_{S_i,k}(k)\| \leq \overline{\eta}(\|\mu(k_0)\|), \quad \forall k \geq k_0,
$$

(21)

where $k_0$ is the initial time step and the value of the bounding constant, $\overline{\eta}$, depends on the initial estimation error ($\|\mu(k_0)\|$) and the upper bound of the interpolation error [14].

On the other hand, if a fault occurs for $k > k_f$, the residual $r_{S_i}(k+1) = \tilde{\eta}_{S_i,k}(k) + f_{S_i,k}(k)$ can be lower bounded as follows

$$
\|r_{S_i}(k+1)\| \geq \|\tilde{\eta}_{S_i,k}(k) + f_{S_i,k}(k)\| - \overline{\eta}(\|\mu(k_f)\|),
$$

(22)

where it is $\tilde{\eta}_{S_i,k}(k,f) = \sum_{l=k_f}^{k-1} R(k,l) f_{S_i,k}(l)$; in addition

$$
R(k,l) = \Phi(k) \Psi(k,l+1) G(l)
$$

and

$$
\Psi(k,l+1) = \begin{cases} 
I_{p \times p}, & k = l + 1 \\
G(k-1)G(k-2) \cdots G(l+1), & k > l + 1.
\end{cases}
$$

B. Decision Making System

The use of residuals (11) and (14) allows detection and isolation of a proprioceptive sensor fault. In detail, residuals in (11) allow detection of the fault and determination of the couple of sensors providing different measurements, while the residuals in (14) allow to determine the failed set.

In order to detect sensor faults, a threshold on the residuals (11) must be set to discriminate faults from normal operating conditions. Then, if $r_{S_i}$ ($i = 1, \ldots, 2n$) in (11) exceeds its threshold, i.e.,

$$
|r_{S_i}| > \rho_{S_i},
$$

(23)

a fault of the couple $(S_1^i, S_2^i)$ is declared. In a similar way, if the norm of the corresponding isolation residual (14) exceeds a suitably selected threshold $\rho_{S_i}$ ($j = 1, 2$)

$$
\|r_{S_i}(k)\| > \rho_{S_i},
$$

(24)

a fault is declared on sensor $S_j^i$.

As already stated, thresholds can be considered user-defined parameters aimed at determining the border between faults and normal operating conditions. In other words, the threshold level sets the maximum tolerated effect of all unmodeled phenomena on the system’s behavior: if their influence is such that the residuals keep below the user-defined thresholds, they should not be considered as faults, i.e., as potential menaces for the system.

If a large set of historical data (recorded during normal operating conditions, i.e., in the absence of failures) is available, the extrema of the residuals can be determined in various healthy conditions. Then, the values of the extrema can be used to determine thresholds. In principle, if enough a priori information about the structure of the uncertainties is available, upper bounds on the magnitude of the fault-free residuals could be derived and used to set the thresholds (e.g., a possible choice for $\rho_{S_i}$ could be given by a reliable estimate of the upper bound on the residual in (21)). However, such bounds can be usually computed on the basis of conservative assumptions, leading to very conservative thresholds and, thus, to low fault sensitivity. Hence, the first (empirical) approach to threshold selection will be pursued.

Equation (11) and inequality (22) lead to the following sufficient conditions ensuring detection and isolation of a fault on sensor $S_j^i$:

$$
\exists k \geq k_f : \|y_{S_i} - y_{S_i}^j\| > \rho_{S_i} \quad \text{and} \quad \| \tilde{\eta}_{S_i}(k_f, k) + f_{S_i,k}(k) \| \geq \rho_{S_i} + \overline{\eta}(\|\mu(k_f)\|).
$$

Thanks to redundant measurements, a healthy signal can be computed when sensor faults occur. To this purpose, the following logic has been considered:

(i) If the detection residuals (11) do not exceed the thresholds, no fault is declared and the signal given by the average of the two sensors (standard duplex measure) is voted.

(ii) If a threshold is exceeded, a fault is declared, then the isolation residuals (14) must be considered so as to decide if the faulty signal can be isolated.

(iii) If only one of the residuals (14) exceeds the threshold, a fault is isolated and the measurement provided by the healthy sensor is voted.

(iv) If both the isolation residuals (14) remain below their thresholds or exceed their thresholds, a missed isolation
is declared and the signal given by the weighted average of the signal output by the two sensors and the two estimates provided by the observers is voted.

In the presence of multiple sensor faults (not occurring on the same couple of sensors), it is possible to use the same isolation observers, provided that a compensation term is added. Namely, once a fault is isolated, the identification of $\delta \bar{x}$ is achieved by comparing the value measured by the healthy sensor with the measure given by the faulty sensor. Therefore, after a fault is correctly isolated, a compensation term is added to the output of the observer (12)

$$\hat{y}_{S_1}(k) = \hat{x}_{S_1}(k) + \delta \bar{x}(k),$$

where $\delta \bar{x} = y_{S_1} - y_{S_2}$, if the fault is on the sensor set $S^2$, or $\delta \bar{x} = y_{S_2} - y_{S_1}$, if the fault is on the sensor set $S^1$. This modification allows using the same observers for the isolation of faults affecting different sensors.

C. Discussion

It is worth discussing the case of simultaneous faults, i.e., faults occurring at the same time step on different sensors. In such a case, detection can be achieved via the residuals in (11), while isolation could not be achieved in some particular cases. In detail, two different cases are worth discussing:

- Two or more sensors of the same set fail simultaneously, i.e., at the same time step. In this case, the faults can be isolated, since only the isolation residual (14) corresponding to the failed set is expected to exceed its threshold.
- Two or more sensors belonging to different sets fail simultaneously, i.e., at the same time step. In this case, isolation could not be achieved since all the isolation residuals are likely to exceed their respective thresholds during the same time interval. In order to isolate the faults, one or more additional observers should be adopted, each corresponding to a different combination of possible simultaneous faults. On the other hand, if the faults do not occur exactly at the same time step, i.e., the fault times are very close to each other (but not equal), the faults can be correctly isolated, provided that the difference between the fault times is large enough to allow the residual corresponding to the earlier fault to return below the threshold.

V. Fault Diagnosis for Force/Torque Sensor and Actuators

The healthy measure voted by the DMS is used to feed a bank of observers providing fault detection and isolation for force/torque sensor and actuators. The first observer is designed for detection purposes. The other observers of the bank are aimed at determining the faulty actuators and/or the faulty components of the readings provided by the force/torque sensor. In detail, $n$ possible actuator fault types are considered, each corresponding to a vector $\delta u$ in (7) with all components zero but that referred to the faulty joint actuator. Moreover, 6 different fault types are considered (3 for the components of the measured force and 3 for the components of the measured torque) for the force/torque sensor, each corresponding to a vector $\delta h$ in (6) having all components zero but that corresponding to the faulty component.

A. Detection

The detection observer has the same form as the observer (12), the only difference is that the former is fed by the healthy measure computed by the DMS, hereafter denoted by $y$.

$$\begin{align*}
\hat{x}_d(k+1) &= A\hat{x}_d(k) + B(y(k))\bar{u}(k) + n(y(k)) + V(y(k))\bar{h}(k) + \eta_d(k, \hat{x}_d(k), \bar{u}(k)) + K_d e_d(k) \\
\hat{y}_d(k) &= \hat{x}_d(k),
\end{align*}$$

(26)

where $e_d = y - \hat{y}_d$ and $\eta_d$ is an estimate of $\eta$ obtained by using the same adaptive interpolator (16)–(19). The following detection residual can be considered

$$r_d(k+1) = e_d(k+1) - F e_d(k).$$

(27)

In the presence of an actuator or force/torque sensor fault, the residual (27) becomes

$$r_d(k+1) = \tilde{\eta}_d(k) + f_*(k),$$

(28)

where $\tilde{\eta}_d = \eta - \tilde{\eta}_d$ and $f_*$ ($*=h,u$) has been defined in (9) and (10). Therefore, an actuator or force/torque sensor fault can be detected if the norm of the residual vector exceeds a suitably defined threshold, $\rho_d$, i.e.,

$$\|r_d(k)\| > \rho_d.$$ 

(29)

By using the same arguments used in Section IV-B, the following sufficient condition for detectability can be obtained ($*=h,u$)

$$\exists k \geq k_f: \|\tilde{f}_*(k_f, k) + f_*(k)\| \geq \rho_d + \tilde{\eta}(\|\tilde{\mu}(k_f)\|),$$

where $\tilde{f}_*, \tilde{\eta}$ are defined as in (22) and $k_f$ is the fault time. Of course, thresholds selection is achieved by following the same guidelines given in Section IV-B.

B. Isolation and identification

The isolation scheme is activated only after a fault has been detected. In detail, a bank of $n+6$ nonlinear adaptive observers are designed in such a way to be sensitive to all types of fault but one: $n$ observers are aimed at isolating actuator faults and 6 observers are for the force/torque sensor faults. The $l$th observer (hereafter $l = 1, \ldots, n+6$) has the form

$$\begin{align*}
\tilde{x}_l(k+1) &= A\tilde{x}_l(k) + B(y(k))\bar{u}(k) + n(y(k)) + V(y(k))\bar{h}(k) + f_l(k, \tilde{x}_l(k)) + K_l e_l(k) \\
\tilde{y}_l(k) &= \tilde{x}_l(k),
\end{align*}$$

(30)

where $f_l$ is an estimate of the effect of the $l$th fault on the system dynamics. Hence, the $l$th observer compensates only the effects of the $l$th type of fault: therefore, the residual vector remains sensitive to the other faults. In the presence of a fault belonging to the $l$th type, $r_l$ becomes

$$r_l(k+1) = e_l(k+1) - F e_l(k) = \eta(k) + \tilde{f}_l(k),$$

(31)
where \( k > k_d \), \( \tilde{f}_l = f_l - \hat{f}_l \) and \( k_d \) is the detection time, i.e., the time step in which the detection residual \( r_d \) exceeds the threshold. The residual output of the \( j \)th observer (\( j \neq l \)) becomes

\[
\begin{align*}
\mathbf{r}_j(k+1) &= \mathbf{e}_j(k+1) - \mathbf{F}_1 \mathbf{e}_j(k) \\
&= \mathbf{\eta}(k) + \mathbf{f}_j(k) - \tilde{f}_j(k, \tilde{x}_j(k)),
\end{align*}
\]

with \( k > k_d \). Therefore, in the presence of the fault \( f_l, r_l \) is expected to remain below a suitably selected threshold, \( \rho_l \), since it is biased only by the uncertainties and the fault estimation error. Instead, the residual output by the \( j \)th observer (for all \( j \neq l \)) is biased by the uncompensated fault effect as well, it is expected to exceed the corresponding threshold, \( \rho_j \).

From (31) and (32), the following inequalities can be derived

\[
\begin{align*}
\|\mathbf{r}_l(k+1)\| &\leq \|\mathbf{\eta}(k)\| + \|\mathbf{f}_l(k)\|, \\
\|\mathbf{r}_j(k+1)\| &\geq \|\mathbf{f}_j(k) - \tilde{f}_j(k, \tilde{x}_j(k))\| - \|\mathbf{\eta}(k)\|, \quad j \neq l.
\end{align*}
\]

Of course, if \( \|\mathbf{\eta}(k)\| \) can be upper bounded by a known constant or a known function of time, the above inequalities can be further completed by using such a bound in lieu of \( \|\mathbf{\eta}(k)\| \). Hence, sufficient conditions for isolability for the \( l \)th fault affecting force/torque sensor or actuators are given by the two inequalities

\[
\|\tilde{f}_l(k)\| \leq \rho_l - \|\mathbf{\eta}(k)\|, \quad \forall k > k_d,
\]

\[
\exists k > k_d: \|\mathbf{f}_l(k) - \tilde{f}_l(k, \tilde{x}_l(k))\| \geq \rho_l + \|\mathbf{\eta}(k)\|, \quad \forall j \neq l.
\]

The above conditions depend both on the capability of the \( l \)th observer to estimate the \( l \)th fault, as well as on the sensitivity of the other isolation observers to the effect of the \( l \)th fault. It can be argued that such a sensitivity depends on different factors, such as structural similarities of two distinct faults; hence, (36) can be interpreted as a condition ensuring that the two faults have a sufficiently different structure. Moreover, (35) and (36) depend on the model uncertainties instead of the uncertainties estimation error; this implies that thresholds wider than those used for detection could be necessary for isolation purposes. As for thresholds selection, the discussion in Section IV-B applies in this case as well.

C. Parametric model of the faults

According to (31), an accurate estimation of the fault effects on manipulator dynamics, \( \tilde{f}_l(k, \tilde{x}_l(k)) \), is required to achieve fault isolation. Hence, in view of (6) and (7), an estimation of the fault vector \( \delta \mathbf{h} \) or \( \delta \mathbf{u} \) is needed.

It is worth remarking that it is not assumed any \textit{a priori} knowledge on the time evolution of the fault. As in Section IV-A for the uncertainties estimation, a linear-in-the-parameters structure is adopted to model the unknown faults. It is worth remarking that such a modeling choice is necessary, since a general model capturing all possible fault behaviors cannot be adopted, while a wide class of functions can be effectively interpolated by a linear-in-the-parameters model \cite{14, 28}. In detail, the following parametric model is assumed

\[
\begin{align*}
\delta \mathbf{h}(k) &= \mathbf{\Omega}_h(k, x(k)) \mathbf{\theta}_h + \mathbf{e}_h(k), \quad (37) \\
\delta \mathbf{u}(k) &= \mathbf{\Omega}_u(k, x(k)) \mathbf{\theta}_u + \mathbf{e}_u(k), \quad (38)
\end{align*}
\]

where \( \mathbf{\Omega}_h (\ast = u, h) \) are known matrices of radial basis functions chosen as in (19), \( \mathbf{\theta}_h \) are unknown vectors of parameters and \( \mathbf{e}_h \) are norm-bounded interpolation errors. Hence, the adaptive approach (16)–(19) can be adopted to achieve the estimates of the fault vectors \( \hat{\delta} \mathbf{h} \) and \( \hat{\delta} \mathbf{u} \), by updating the weights vectors \( \mathbf{\theta}_h \) as follows

\[
\begin{align*}
\hat{\mathbf{h}}(k+1) &= \hat{\mathbf{h}}(k) + \mathbf{\Omega}_h^T(k) \mathbf{\Gamma}_h(k) V^1 \mathbf{r}_l(k+1), \\
\hat{\mathbf{u}}(k+1) &= \hat{\mathbf{u}}(k) + \mathbf{\Omega}_u^T(k) \mathbf{\Gamma}_u(k) B^1 \mathbf{r}_l(k+1),
\end{align*}
\]

where \( \mathbf{\Gamma}_h, \mathbf{\Gamma}_u \) are computed as in Section IV-A, \( V^1 = \mathbf{O}_{6 \times n} - T^{-1} \mathbf{J}^T \mathbf{M} \) and \( B^1 = \mathbf{O}_{n \times n} T^{-1} \mathbf{M} \). Then, \( \tilde{f}_l \) can be estimated by considering (9) and (10). If the fault affects the \( l \)th component of the force/torque sensor (\( l = 1, \ldots, 6 \)), then, the estimate \( \hat{f}_l \) is computed as

\[
\hat{f}_l(k) = v_l(y(k)) \hat{\delta} h_l(k),
\]

where \( v_l \) is the \( l \)th column of \( V \) and \( \hat{\delta} h_l \) is the \( l \)th component of \( \hat{\delta} h \). Otherwise, if the fault affects the \((l-6)\)th joint actuator (\( l = 7, \ldots, n+6 \)), the estimate \( \hat{f}_l \) is computed as

\[
\hat{f}_l(k) = b_{l-6}(y(k)) \hat{\delta} u_{l-6}(k),
\]

where \( b_{l-6} \) is the \((l-6)\)th column of \( B \) and \( \hat{\delta} u_{l-6} \) is the \((l-6)\)th component of \( \hat{\delta} u \).

By invoking the results in \cite{14}, \( \tilde{f}_l \) can be assumed to be norm-bounded, if the interpolation error is bounded (i.e., \( \| \mathbf{e}_h \| \leq \epsilon_M \)), i.e.,

\[
\exists \hat{f}_{l, M} > 0: \| \tilde{f}_l(k) \| \leq \hat{f}_{l, M}, \quad \forall k \geq k_0.
\]

Again, the discussion in Section IV-B related to thresholds selection applies as well.

In order to make the observers adopted for isolation of proprioceptive sensor faults insensitive to force/torque sensor and actuator faults, the estimate \( \hat{f}_l \) of the isolated force/torque sensor (or actuator) fault can be introduced in (12). Hence, if the \( l \)th fault type (affecting the force/torque sensor or the actuators) occurs and is correctly detected and isolated, then the fault estimate (41) or (42) is added to the observer dynamics in order to compensate the effect of the fault. Therefore, provided that a bounded (possibly small) error on the fault estimation is achieved, \( e_{S_l} \) is only marginally influenced by the faults affecting either the force sensor or the actuators.

D. Discussion

The adoption of a bank of \( n + 6 \) isolation observers, in addition to the detection observer, is a standard approach fully accepted in the research community (see, e.g., \cite{28}). Moreover, the most computationally heavy nonlinear terms in the observers (i.e., \( Bu, n, Vh \)) are exactly the same in both the detection observer and the \( n + 6 \) isolation observers, while
they differ only for the (much lighter) linear \((A\ddot{\textbf{x}}, K_o,e)\) and adaptive \((\hat{\eta}_d, \hat{f}_t)\) terms. Hence, the common terms can be computed once, with considerable computational savings. Finally, it must be remarked that the bank of isolation observers is activated only in the presence of detection. In conclusion, the computational load is compatible with nowadays CPU computational power.

In addition, it is worth discussing the effect of a missed sensor fault detection on the performance of the actuators and force sensor diagnosis scheme. A missed sensor fault detection might occur only if the two readings coming from the failed sensors in the same redundant couple are very close to each other (i.e., the difference is below the detection threshold). This can be due to the following causes:

- The faults are of very small magnitude, i.e., the fault signatures are not large enough to drive the residuals over the thresholds. In such a case, of course, the sensor fault diagnosis system will not declare any fault. However, if the user-defined thresholds are properly chosen, it is expected that the consequences of a small magnitude sensor fault will not affect the overall control and diagnosis performance in a remarkable way.
- Two simultaneous faults having (almost) the same time history occur on the same couple of sensors, hence, the corresponding sensor detection residual \((11)\) remains below the threshold. In such a case, since both the sensors in the couple fail, a reliable voted measure could not be obtained. Thus, actuator fault diagnosis is likely to provide poor performance. On the other hand, force sensor and actuator fault detection is still possible, since the corresponding detection observer would be characterized by large errors. It must be remarked, however, that such an event is expected to occur very rarely and might lead to potentially catastrophic consequences if system operation is not stopped.

VI. EXPERIMENTS

A. The experimental setup

The proposed approach has been tested on a real setup composed by two six-revolute-joint anthropomorphic industrial robots Comau SMART-3S with C3G 9000 control units, available in the PRISMA Lab. Two six-axis force/torque sensor ATI FT30-100 with force range of \(\pm 130\) N and torque range of \(\pm 10\) Nm are mounted at either arm’s wrist.

The robotic system is equipped with an open controller architecture described in [31].

The FD framework developed in the previous sections has been tested in a complex cooperative manipulation operation previously designed [25]. In the experiments, the manipulators grasp a \(0.24 \times 0.24 \times 0.45\) m cardboard box, with \(m_e = 1.3792\) kg mass. The grasp is achieved via two couples of rubber discs; each couple consists of two discs mounted, respectively, on the end-effector of the manipulator and on one of the object’s faces at a given grasp area. The environment interacting with the cooperative system is a planar wooden horizontal surface, whose position and orientation is unknown to the controller.

The assigned task has a total duration of \(115\) s and is composed by three phases:

- in the first phase (grasping), with a duration of \(9\) s \([0,9]\) s interval), the manipulators, moving from their initial configuration, grasp the object using the two rubber disks;
- in the second phase (free-space motion), with a duration of \(30\) s \([9,39]\) s interval), the manipulators lift the commonly held object and change the object orientation of an angle of \(0.23\) rad about the axis between the two end-effectors positions, then, the object is moved down so as to reach the contact with the environment;
- in the last phase (constrained motion), with a duration of \(76\) s \([39,115]\) s interval), the object is commanded to follow a vertical path of \(0.011\) m with constant orientation, pushing against the environment.

As the standard industrial manipulators are not equipped with duplicated joint sensors, hardware redundancy has been emulated via software. Namely, the set \(S^1\) is composed by the physical joint sensors, while the set \(S^2\) is composed by virtual sensors whose readings are equal to the readings from set \(S^1\) with an additional Gaussian noise \(\mathcal{N}(0,10^{-5})\). Moreover, since the manipulators are not equipped with velocity sensors, joint velocity readings have been obtained by numerical differentiation. In all the case studies presented below, in order to safely emulate the faults, an additive signal has been superimposed to the measured experimental data, i.e., the sequences \(\delta x, \delta h\) and \(\delta u\) have been simply added to the fault-free measured data.

The following matrix gain \(K_o\) has been used for all the state observers

\[
K_o = \begin{bmatrix} K_1 & TI_6 \\ O_6 & K_2 \end{bmatrix}, \quad K_1 = K_2 = 0.9I_6,
\]

which has been selected by testing various choices on available trajectories without faults.

In order to tune the residual thresholds, a number of experiments in the absence of faults have been performed and the corresponding residuals recorded, then, the thresholds have been set on the basis of the maximum norm of the obtained residuals. The trajectories used to compute the residuals have been selected so as to maximize the effect of uncertain terms (e.g., Coriolis/centrifugal terms, unmodeled frictional effects). In detail, all the thresholds used for the detection and the isolation of proprioceptive sensor faults have been set to be \(2.5 \cdot 10^{-4}\) rad and will be denoted by \(\rho_S\). With regards to the actuator/force sensor faults, \(\rho_d = 2 \cdot 10^{-4}\) has been assumed for the detection residuals, while the thresholds \(\rho_u = 7 \cdot 10^{-4}\) and \(\rho_h = 3.6 \cdot 10^{-3}\) are used for actuators and force sensors isolation, respectively. It is worth noticing that, as remarked in Section V-B, it is advisable to choose the thresholds for isolation residuals larger than those chosen for the detection residuals. Finally, concerning the RBF interpolators introduced in Section IV-A (for uncertainties estimation) and in Section V-C (for fault estimation), 20 radial basis functions have been employed, the centroids have been evenly distributed in the domain to which the input vector belongs, the widths have been set to \(10^2\) and \(Q = 10^2I_{12}\).
The sampling time adopted for the diagnosis scheme \((T = 1\ \text{ms})\) is the smallest one supported by the available hardware/software architecture. However, the approach has been tested by using larger sample times: 2 and 4 ms, respectively. In the first case \((T = 2\ \text{ms})\), very small adjustments of some gains and thresholds are necessary, while in the second case \((T = 4\ \text{ms})\) larger thresholds \((10-15\%\) larger than the previous case) must be adopted while keeping the same gains.

In the following, residuals are normalized to their respective thresholds: in the plots all the thresholds will be set to 1.

For the sake of clarity, as all the faults relative to the case studies appear in a limited time window, the time histories of the variables will be reported only in the relevant time range.

B. Uncertainties estimation

The accuracy of the RBF-based uncertainties estimation approach described in Section IV-A is first experimentally investigated. To the purpose, the uncertainties vector \(\eta\) is computed as in [12]

\[
\eta(k, x(k), u(k)) = x(k + 1) - [ Ax(k) + B(x(k))u(k) + n(x(k)) + V(x(k))\hat{h}(k)] ,
\]

using the measured values of the state variables and of the input torque along one of the trajectories executed by the manipulators in the absence of faults. The results, not shown here for the sake of brevity, confirm that \(\eta\) can be assumed continuous and norm bounded. Namely, the peak value of the norm of the interpolation error, \(\|\eta\| = \|\eta - \hat{\eta}\| = \|\Phi\hat{\mu} + \epsilon\|\), is less than five per cent of the magnitude of the peak value of \(\|\eta\|\) and is bounded by a small constant \((about 5 \cdot 10^{-5}\ \text{rad/s})\).

C. First case study

The first manipulator is affected by two position sensor faults and a force sensor fault, while an actuator fault affects the second manipulator. All the faults occur during the grasping and the free-motion phases.

In detail, as for the two position sensor faults, the first involves sensor \(S^1_2\) (position sensor at joint 2 belonging to the first set)

\[
\delta x_{S^1_2}(k) = -0.5 \left( 1 - e^{-(kT - k_1T)} / \tau_1 \right) 1(k - k_1) \ \text{rad},
\]

where \(k_1T = 5\ \text{s}\) represents the fault time, \(\tau_1 = 0.01\ \text{s}\) is the fault time scale and \(1(k)\) is the discrete-time step function. The second sensor fault involves sensor \(S^1_3\) (position sensor at joint 3 belonging to the first set)

\[
\delta x_{S^1_3}(k) = 0.6 \left( 1 - e^{-(kT - k_2T)} / \tau_2 \right) 1(k - k_2) \ \text{rad},
\]

where \(k_2T = 15\ \text{s}\) and \(\tau_2 = 0.02\ \text{s}\). The force sensor fault on the first manipulator occurs on the third component

\[
\delta h_3(k) = 30 \left( 1 - e^{-(kT - k_3T)} / \tau_3 \right) 1(k - k_3) \ \text{N},
\]

where \(k_3T = 10\ \text{s}\) and \(\tau_3 = 0.03\ \text{s}\). Finally, a fault on the third actuator of the second manipulator is injected

\[
\delta u_3(k) = 80 \left( 1 - e^{-(kT - k_4T)} / \tau_4 \right) 1(k - k_4) \ \text{Nm},
\]

with \(k_4T = 12\ \text{s}\) and \(\tau_4 = 0.01\ \text{s}\).

The detection residuals for the first manipulator are shown in Figure 2: both the position sensor faults (top) and the actuator/force sensor fault (bottom) are correctly detected. Thanks to the compensation terms added to the observers (12), it is possible to detect multiple sensor faults (occurring not on the same couple), together with a force sensor or actuator fault.

![Figure 2](image.png)

**Fig. 2.** First case study. Detection residuals (first manipulator). Top: detection residuals for each couple of redundant position sensors. Bottom: detection residuals for the force/torque sensor and the actuators.

The isolation residuals for the joint sensor faults relative to the first manipulator are reported in Figure 3: the top figure shows that the sensor faults are correctly isolated on the first sensor set. Due to the fault compensation, the residuals exceed the threshold only for a short amount of time.

![Figure 3](image.png)

**Fig. 3.** First case study. Isolation residuals for position sensor faults (first manipulator). Top: isolation residuals for the first sensor set. Bottom: isolation residuals for the second sensor set.

In Figure 4 the voted signals are shown. As it can be seen, as soon as the faults occur the measures output by the DMS for the faulty joints coincide with the measurements of the healthy sensors.

The norm of the isolation residuals corresponding to the force/torque sensor and the actuators of the first manipulator are reported in Figures 5 and 6, respectively, showing that correct isolation is achieved. Indeed, only the residual corresponding to the third component of the force sensor remains below the threshold, since it is insensitive to this fault, while the other residuals exceed the thresholds immediately after the fault occurrence (Figure 5). Moreover, Figure 6 shows that the norm of the isolation residuals corresponding to the actuator faults overcome the thresholds.

Finally, Figure 7 shows that a fairly accurate identification of the fault magnitude is achieved.
In Figure 8 the detection residuals for both the joint position (top) and actuator/force sensor faults (bottom) for the second manipulator are shown, showing that position sensor faults are obviously not detected, while a force sensor or actuator fault is detected at $kT = 12$ s.

All the residuals corresponding to the force/torque sensor faults, shown in Figure 9, exceed the corresponding thresholds, as expected. On the other hand, correct isolation of actuator faults is achieved, as shown in Figure 10, since only the residual corresponding to the third actuator remains below the threshold, while the other residuals exceed their thresholds immediately after the fault occurrence.

Finally, as shown in Figure 11, a fairly accurate identification of the actuator fault magnitude is achieved.

The detection and isolation times for the first case study are reported in Table I. It can be noticed that all faults are almost instantaneously detected. Isolation of joint sensor faults is achieved after a very few time steps, due to the direct effect of the fault on the residual (14). On the other hand, actuator and force sensor faults are filtered by the system’s dynamics and weighted by matrices $B$ and $V$, respectively. Therefore, the effects of actuator and force sensor faults on observer dynamics could be less evident (as in this case, where $\|B\| < 1$ and $\|V\| < 1$) than the effect of proprioceptive sensor ones. Thus, larger isolation time are experienced in this case.
D. Second case study

In the second case study, an abrupt position sensor fault and a slowly varying actuator fault have been considered on the first manipulator, while a slow position sensor fault and a slow force sensor fault have been considered on the second manipulator. In addition, the force sensor fault on the second manipulator is characterized by a more erratic (sinusoidal) behavior. All the faults occur during the free-motion phase and the interaction of the object with the horizontal plane.

For the first manipulator, the position fault occurs on the sensor of the fifth joint belonging to the first set

$$\delta x_{S_5^1}(k) = 0.5 \left( 1 - e^{-(kT-k_1T)/\tau_1} \right) 1(k-k_1) \text{ rad},$$  

where $k_1T = 37$ s is the fault time and $\tau_1 = 0.03$ s is the fault time constant. A slowly varying actuator fault affects the actuator of the second joint

$$\delta u_2(k) = 60 \left( 1 - e^{-(kT-k_2T)/\tau_2} \right) 1(k-k_2) \text{ Nm},$$

where $k_2T = 45$ s and $\tau_2 = 0.3$ s. As for the second manipulator, a slowly varying position fault affects the sensor of the sixth joint belonging to the second set

$$\delta x_{S_6^2}(k) = 0.5 \left( 1 - e^{-(kT-k_3T)/\tau_3} \right) 1(k-k_3) \text{ rad},$$

where $k_3T = 35$ s and $\tau_3 = 0.2$ s. Finally, the force sensor fault is given by

$$\delta h_1(k) = (50+10\sin(\omega kT)) \left( 1 - e^{-(kT-k_4T)/\tau_4} \right) 1(k-k_4) \text{ N},$$

where $k_4T = 36$ s and $\tau_4 = 0.2$ and $\omega = 2\pi/3$ rad/s.

The detection residuals for the first manipulator are reported in Figure 12: a position sensor fault is correctly detected (top) as well as an actuator/force sensor faults (bottom).

Figure 13 shows that a fault on the first position sensor set of the first manipulator is correctly isolated.

Since the position sensor fault occurs on the fifth position sensors of the first set ($S_5^1$), the measures output by the DMS, not shown here for brevity, coincide with the measurements of the second position sensor set.

All the residuals related to the force sensor, not reported for brevity, overcome the respective thresholds, and thus force sensor faults for the first manipulator are not declared. On the contrary, the residual corresponding to the second actuator remains below the threshold (Figure 14), while the other residuals exceed their corresponding thresholds.

Finally, Figure 15 shows that a good estimate of the fault is computed.

As for the second manipulator, Figure 16 shows that the position sensor fault is correctly detected at $kT = 35$ s on the
Isolation time \( k \) of the fault occurring at the second joint (top), while the residual relative to the actuator/force sensor faults detection observer exceeds the threshold (bottom) at \( kT = 45 \) s.

Figure 17 correctly reports a fault on the second position sensor set of the second manipulator.

Since the fault occurs on the sixth position sensor of the second set \( S_2^2 \), the output of the DMS relative to this joint, not shown for brevity, coincides with the measurement of the first position sensor set.

The isolation residuals for the force/torque sensor are shown in Figure 18. Namely, the residual corresponding to the first component remains below the threshold, while the other residuals overcome the thresholds. All the residuals corresponding to the actuator faults, not reported for brevity, overcome the thresholds, since actuator faults do not occur.

Finally, Figure 19 shows that a good estimate of the force sensor fault is computed.

The detection and isolation times for the second case study are reported in Table II. Detection and isolation of joint sensor faults is again almost instantaneous while, due to larger time constants with respect to the previous case study, the detection of actuator and force sensor faults needs a few more steps. The same considerations apply to the isolation times.

**E. Third case study**

In order to show the ability of the FDI scheme to detect and isolate joint sensor faults occurring at time steps very close to each other, two joint position faults with a relative delay of 0.2 s are considered. The two faults occur during the free-motion and the interaction of the object with the horizontal plane.

In detail, the first fault involves the position sensor at joint 2 belonging to the first set

\[
\delta x_{S^2_2}(k) = 0.5 \left(1 - e^{-(kT-k_1T)/\tau_1}\right) 1(k-k_1) \text{ rad,} \quad (52)
\]

\[\begin{array}{c|c|c|c}
\text{Fault type} & \text{Fault time} & \text{Detection time} & \text{Isolation time} \\
\hline
S^2_1 & k_1T & (k_1 + 1)T & (k_1 + 2)T \\
S^2_2 & k_3T & (k_3 + 27)T & (k_3 + 53)T \\
\hline
u_2 & k_4T & (k_4 + 49)T & (k_4 + 87)T \\
h_1 & k_3T & (k_3 + 49)T & (k_3 + 87)T \\
\end{array}\]

**TABLE II**

SECOND CASE STUDY. DETECTION AND ISOLATION TIMES.
where \( k_1 T = 35 \text{ s} \) and \( \tau_1 = 0.02 \text{ s} \). The second fault involves the position sensor at joint 3 belonging to the second set

\[
\delta x_{S_2^3}(k) = 0.6 \left( 1 - e^{-(kT-k_2 T)/\tau_2} \right) 1(k - k_2) \text{ rad},
\]

where \( k_2 T = 35.2 \text{ s} \) and \( \tau_2 = 0.03 \text{ s} \).

Figure 20 shows that the two sensor faults (top) are detected, while the detection residuals for the force sensor and the actuators (bottom) correctly do not report any fault occurrence.

Figure 21 shows that both faults are correctly isolated. In Figure 22 the time history of the voted signals is reported: as soon as the faults occur the measures output by the DMS for the faulty joints coincide with the measurements of the healthy sensors, thus providing a correct measure.

Finally, the case when the above defined faults are exactly simultaneous (i.e., \( k_1 = k_2 \)) is discussed. In this case, both the isolation observers exceed their thresholds at the same time instant (Figure 23). Therefore, the DMS cannot provide the correct signal and thus the voted measures at joints 2 and 3 are the weighted average of readings and estimates (see Section IV-B). Hence, the detection residual for the force/torque sensor and the actuators fault erroneously overcomes its threshold, since the observer (26) is fed by a wrong voted measure. Thus, a false alarm is generated.

![Detection residuals for the force sensor and the detection residuals for each couple of redundant position sensors](image)

**ACKNOWLEDGMENT**

This work was supported by MIUR (Italian Ministry of University and Research) under Grant PRIN 2009 n. 20094WTJ29 (Cooperative and Collaborative Robotics).

**REFERENCES**


Fig. 23. Detection residuals in the presence of exactly simultaneous faults (first manipulator). Top: detection residuals for each couple of redundant position sensors. Bottom: detection residual for the force/torque sensor and the actuators.


Fabrizio Caccavale was born in Naples, Italy, on November 14, 1965. He received the Laurea degree and the Ph.D. degree in Electronic Engineering from the University of Naples in 1993 and 1997, respectively. From 1999 to 2001 he has been Assistant Professor at the Department of Computer and Systems Engineering of the University of Naples. He is currently Associate Professor at the Department of Environmental Engineering and Physics of the University of Basilicata. His research interests include manipulator inverse kinematics techniques, cooperative robot manipulation, fault diagnosis and nonlinear control of mechanical systems. He has published more than 100 journal and conference papers. He is co-author of the books “Fondamenti di Sistemi Dinamici” (McGraw-Hill 2003, in Italian) and “Control and Monitoring of Chemical Batch Reactors” (Springer, 2011); he is co-editor of the book “Fault Diagnosis for Mechatronic Systems: Recent Advances” (Springer, 2002) and co-author of the chapter “Cooperative Manipulators” for the Springer Handbook of Robotics, (Springer, 2008). He has been in the program committee of several international conferences and workshops. He is Associate Editor of the journals Robotica and IEEE Transactions on Control Systems Technology. He is Senior Member of IEEE.

Alessandro Marino was born in Potenza, Italy, on March 1, 1982. He received the Laurea degree in Computer Engineering from University of Naples Federico II in 2006 and the Ph.D. degree in Industrial and Innovation Engineering from University of Basilicata in 2010. From April to October 2006 he has been recipient of a research contract at the Department of Computer and Systems Engineering of the University of Naples. From April to September 2008 he has been a visiting scholar at the Department of Computer Science of the University of Tennessee, Knoxville, TN. Since July 2010 to September 2011 he was a post-doc at the University of Cassino. Since December 2011 he is Assistant Professor at the Department of Electronic Engineering and Computer Science of University of Salerno, Italy. His research interests include cooperative robot manipulation, trajectory planning and control algorithms for platoons of robots moving in an unknown environment. He is Member of IEEE.

Giuseppe Muscio was born in Potenza, Italy, on June 11, 1983. He received the Bachelor degree cum laude in 2006 and the Master degree cum laude in 2010, both in Mechanical Engineering from the University of Basilicata. Since November 2010 he is PhD student in Industrial and Innovation Engineering at the Department of Environmental Engineering and Physics of the University of Basilicata. His research interests are focused on control of dual arm/hand dexterous manipulators, cooperative manipulators, fault detection and fault tolerant control.

Francesco Pierri was born in Tramutola, Italy, on November 18, 1977. He received the Laurea Degree in Mechanical Engineering and the PhD degree in Environmental Engineering from the University of Basilicata in 2003 and 2007, respectively. From March to September 2006 he has been a visiting scholar at the Department of Automatic Control of the Lund University, Sweden. Since December 2008 he is Assistant Professor at Department of Environmental Engineering and Physics of the University of Basilicata. His research interests include process control, fault diagnosis and fault tolerant control for nonlinear systems, and control of robotics hands. He is author of about twenty journal and conference papers. He is Member of IEEE.