Model-free active sensor fault tolerant control for systems with linear static characteristic

First Meziane Ait Ziane[1], Second Michel Zasadzinski[2*], Third Cédric Join[2], Fourth Marie-Cécile Péra[3†]

[1]Group of Research in Electrical Engineering of Nancy (GREEN), Université de Lorraine, Street, Vandœuvre-lès-Nancy, 54506, France.
[2]Research Center for Automatic Control of Nancy (CRAN), UMR CNRS 7039, Université de Lorraine, Street, Vandœuvre-lès-Nancy, 54506, France.
[3]CNRS, FEMTO-ST Institute, , FCLAB, Université de Marie et Louis Pasteur, rue Thierry-Mieg, Belfort, 90010, France.

*Corresponding author(s). E-mail(s): michel.zasadzinski@univ-lorraine.fr, cedric.join@univ-lorraine.fr;
Contributing authors: meziane.ait-ziane@univ-lorraine.fr; cedric.join@univ-lorraine.fr; marie-cecile.pera@univ-fcomte.fr;
†These authors contributed equally to this work.

Abstract

An active model-free sensor fault tolerant control approach is presented in this paper. The proposed method is based on a model-free controller that has demonstrated an effective ability to work without any analytical model knowledge. The active fault tolerant control procedure has three stages: firstly, the model-free controller is designed using an ultra-local model; secondly, this ultra-local model is used to detect and estimate the sensor fault; thirdly, the obtained estimation is used to adapt the control law according to the sensor fault. The aim of the proposed active fault tolerant control procedure is to ensure that the regulated output, but not the measured one, tracks the desired trajectory despite the occurrence of a sensor fault. Additive or multiplicative sensor fault are considered for systems that have a linear relationship between control input and system output for any reached steady-state. This kind of system includes linear ones and some specific nonlinear systems. The developed methods are validated via numerical simulations for unstable linear and nonlinear systems, with and without saturation of the control input.

Keywords: Sensor fault accommodation, Sensor fault estimation, Additive sensor fault, Multiplicative sensor fault, Model-free approach

Acknowledgements. The authors would like to thank Professor Michel Fliess for his helpful comments and suggestions.

Author Contributions. All authors contributed equally to this work.

Funding. No funds, grants, or other support was received.

Declarations

- **Data Availability** Not applicable.
- **Code Availability** Not applicable.
- **Conflict of Interest** Not applicable.

Model-Free Active Sensor Fault Tolerant Control For Systems With Linear Static Characteristic

Abstract

An active model-free sensor fault tolerant control approach is presented in this paper. The proposed method is based on a model-free controller that has demonstrated an effective ability to work without any analytical model knowledge. The active fault tolerant control procedure has three stages: firstly, the model-free controller is designed using an ultra-local model; secondly, this ultra-local model is used to detect and estimate the sensor fault; thirdly, the obtained estimation is used to adapt the control law according to the sensor fault. The aim of the proposed active fault tolerant control procedure is to ensure that the regulated output, but not the measured one, tracks the desired trajectory despite the occurrence of a sensor fault. Additive or multiplicative sensor fault are considered for systems that have a linear relationship between control input and system output for any reached steady-state. This kind of system includes linear ones and some specific nonlinear systems. The developed methods are validated via numerical simulations for unstable linear and nonlinear systems, with and without saturation of the control input.

Keywords: Sensor fault accommodation, Sensor fault estimation, Additive sensor fault, Multiplicative sensor fault, Model-free approach

1 Introduction

In the broadest sense of the term, a fault can be considered as an unexpected event that occurs and has an impact on the behavior of a system, preventing it from performing its nominal operation. This event occurs in the system itself, the actuator or the sensor [1–4]. The literature presents a wealth of fault detection methods that are employed to provide the user informations about the operating status of the system [5]. The presence of a fault usually leads to undesirable consequences such as system performance degradation. The aim of a fault tolerant control (FTC) strategy is to ensure an acceptable level of system behaviour in the presence of a fault [6].

Fault control strategies can be categorized into two main approaches: passive fault tolerant control (PFTC) [7] and active fault tolerant control (AFTC) [8] strategies. The PFTC strategy allows the fault to be tolerated without any information about the type and magnitude of the fault. When designing a PFTC law, the controller can tolerate only a few faults, although not all faults that may impact the system [9].

However, the AFTC can tolerate all faults that are detected and isolated via the fault detection and isolation module. The latter gives more informations about the fault and enables to identify its nature and amplitude. These informations are used by the controller to accommodate the fault in order to ensure a consistent behaviour of system operation [10].

In this paper, we address an AFTC approach to accommodate the sensor fault in a model-free framework, and the sensor fault accommodation is defined as follows: the regulated output, but not the measured one, tracks the desired trajectory despite the occurrence of a sensor fault. Generally, the design of AFTC for sensor fault in the literature is based on the knowledge of the analytical model of the controlled system and can be classified according to kind of sensor fault or accommodation objective.

The considered sensor faults are either additive like in [11–19], or additive and multiplicative like in [20–22], or the measurement matrix changes due to the occurrence of a sensor fault [23–25], or described by a loss of sensor information like in [26, 27]. One of the aims of AFTC procedure is to regulate the system output to the setpoint despite the fault on the measurement as in [11,13,18–20]. In [12,22,24,25], the tracking objective is on a combination of the state components. Other aims of AFTC strategy is to ensure that some optimization criteria (like induced norms on signals) are satisfied like in [14, 16, 17, 21, 23, 26]. In addition, only the closed loop stability is guaranteed in [15, 27].

In the literature, there are few methods that introduce AFTC strategy without an a priori knowledge of the system's analytical model. In this paper, an AFTC procedure based on model-free control is developed to accommodate sensor fault. Model-free control is a very efficient approach to control linear and nonlinear systems without using the analytical model of the system [28,29]. This model-free control can be seen as a PFTC that can tolerate actuator faults and disturbances affecting the system [30–32] without the need to estimate the actuator fault. However, this approach is unable to tolerate sensor faults. In [33], a model-free AFTC method is proposed to tolerate an additive sensor fault for linear systems.

In this paper, we address an AFTC approach that considers both additive and multiplicative sensor faults for linear and nonlinear systems exhibiting a linear static characteristic in a model-free framework. The sensor fault accommodation is defined as follows: the regulated output, but not the measured one, should tracks the desired trajectory despite the occurrence of a sensor fault. The ultra-local model used in model-free control allows generating a residual to detect and estimate the sensor fault. This estimate is used to adapt the model-free control law in order to achieve the sensor fault accommodation. The performances of the developed AFTC is validated for both unstable linear and nonlinear systems with input control saturation.

This paper is structured as follows. The problem of model-free active sensor fault tolerant control for systems with linear static characteristic is stated in Section 2. Model-free control design is shortly presented in Section 3 and is used to residual generation for sensor fault detection in Section 4. The sensor fault estimation in model-free framework is treated in Section 5 where Subsections 5.1 and 5.2 are devoted to additive and multiplicative cases, respectively. The sensor fault accommodation is given in Section 6. The AFTC strategy is applied to an unstable linear system in Section 7, where the additive and multiplicative cases are treated in Subsections 7.1 and 7.2, respectively. The AFTC for unstable nonlinear systems is illustrated in Sections 8 and 9 for additive and multiplicative cases, respectively. In Section 10, the numerical simulation results are analysed and the model-free framework for AFTC is discussed. The conclusions of this work are presented in Section 11.

2 Problem statement for closed loop steady state sensor fault accommodation

The closed loop system considered in this paper is given by

$$\dot{x}(t) = g(x(t), u(t)) \tag{1a}$$
$$y(t) = h(x(t), u(t)) \tag{1b}$$
$$u(t) = \gamma(y_m(t), y_d(t)) \tag{1c}$$
$$y_m(t) = \rho(y + f_a(t)) + (1-\rho) f_m(t) y(t) \tag{1d}$$

where $x(t) \in \mathbb{R}^n$ is the state, $u(t) \in \mathbb{R}$ is the control input, $y(t) \in \mathbb{R}$ is the regulated output, $y_m(t) \in \mathbb{R}$ is the measured output and $y_d(t) \in \mathbb{R}$ is the desired trajectory, while $f_a(t) \in \mathbb{R}$ and $f_m(t) \in]0;1] \subset \mathbb{R}$ are the additive and multiplicative sensor fault, respectively, with $\rho \in \{0,\ 1\}$.

Assumption 1. *The stability of the closed loop system Eq. (1) is guaranteed by the control law $u(t)$ and the two following properties are satisfied*

$$\forall \varepsilon \in \mathbb{R}^+,\ \exists T_1 \in \mathbb{R}^+ \text{ such that } |y_m(t) - y_d(t)| \leqslant \varepsilon, \qquad \forall t > T_1, \tag{2}$$
$$\forall \varepsilon \in \mathbb{R}^+,\ \exists T_2 \in \mathbb{R}^+, \exists c \in \mathbb{R} \text{ such that } |u(t) - c y(t)| \leqslant \varepsilon, \qquad \forall t > T_2. \tag{3}$$

In the property given by Eq. (2), the measured output y_m follows the desired trajectory y_d when steady-state is reached regardless of the occurrence of the sensor fault f_a or f_m, while the regulated output y converges to y_d only if either $f_a = 0$ and $\rho = 1$ or $f_m = 1$ and $\rho = 0$. In the property given by Eq. (3), the constant c means that there exists a linear relationship between control u and output y for any reached steady-state. Note that the constant c exists for all linear systems, but only for special kinds of nonlinear systems (see the examples in Sections 8 and 9).

So it follows that the regulated output y does not converge to the desired trajectory y_d in presence of sensor fault since $y \neq y_m$ and the closed loop works under faulty situation.

Problem 1. *The AFTC procedure consists on replacing the measurement signal y_m by an accommodation signal y_{acc} to vanish the effect of sensor fault on the regulated output y, i.e. to guarantee that y converges to y_d. So the control input in Eq. (1c) is replaced by*

$$u(t) = \gamma(y_{acc}(t), y_d(t)). \tag{4}$$

The accommodation signal y_{acc} is obtained through three steps named sensor fault detection (see Section 4), sensor fault estimation (see Section 5) and sensor fault accommodation (see Section 6). These three steps are based on the model-free control design described in the following section.

Since the model-free control has been proposed in the literature, only a short recall is given in Sections 3.

3 Model-free control

In [28,29], the model-free control is supported by an ultra-local model which replaces the global mathematical model of the system to be controlled. This ultra-local model is represented as follows

$$y_m^{(\nu)}(t) = F(t) + \alpha u(t) \tag{5}$$

where ν refers to the derivative order of the measured output y_m, α is parameter chosen by the user and F is a function that includes all the unknown part of the system. In this paper, ν is equal to 1. When $\nu = 1$, the model-free control is called iP controller and the control law is given by

$$u(t) = \frac{1}{\alpha}\left(-\widehat{F}(t) + \dot{y}_d(t) + k_p e(t)\right) \tag{6}$$

where

- $e = y_d - y_m$ is the tracking error,
- k_p is a tuning proportional gain,
- $\widehat{F}$ is the estimation of F given by [34]

$$\widehat{F}(t) = \frac{-3!}{T^3}\int_{t-T}^{t}((T-2\tau)y_m(\tau) + \alpha\tau(T-\tau)u(\tau))\,\mathrm{d}\tau \tag{7}$$

  where $T > 0$ might be small and $[t - T;\ t]$ denotes the sliding windows of the integration interval.

In [28, 29] the determination of the parameter T results in trade-off between the sampling period of the numerical integration and the intensity of noise that may affect the measurement (see remark 13 in [28]). As explained in [35], the unmodeled dynamics are estimated by F from the measurements y_m and the control input u, and this estimate is updated for each integration time interval $[t-T;\ t]$ with a backward horizon strategy. The above trade-off leads to choosing a small integration window in order to have an acceptable short time to estimate F, but a window large enough to guarantee low-order filter properties to attenuate noise that may affect the measurement y_m.

In [29, 36], it is shown that the iP controller, i.e. when $\nu = 1$, ensures that the measurement y_m tracks the desired trajectory y_d, as required by property in Eq. (2).

The reader is referred to [28–30, 35–52] for further explanations of the model-free control approach and its applications to various processes.

4 Model-free residual generation for sensor fault detection in steady-state

4.1 Determination of steady-state behavior

Since the AFTC procedure described below is based on the constant c given in Eq. (3) characterizing the steady-state behaviour of the closed loop system, the start and final instants of steady-state behaviour should be determined:

- the start instant t_{s_i} of steady-state behaviour corresponds to the instant when $|\dot{y}_d|$ and $|e|$ are "sufficiently" small during a given duration,
- the final instant t_{f_i} of steady-state behaviour is the instant when $|\dot{y}_d|$ becomes not "sufficiently" small during a given duration.

So to apply the approach, the user should determine where the process is transient or steady state behaviour. In view of the two previous items some parameters are chosen in order to decide when the process is in steady-state. These parameters are as follows:

- Let $\kappa^{y_d} > 0$ and $\kappa^e > 0$ be the parameters that define the tolerated variations of y_d and e to consider that we are in steady state, i.e. $|\dot{y}_d| < \kappa^{y_d}$ and $|e| < \kappa^e$.
- The conditions $|\dot{y}_d| < \kappa^{y_d}$ and $|e| < \kappa^e$ must be of sufficiently long duration to be considered steady-state. The parameters characterizing these two durations are d^{y_d} for $|y_d|$ and d^e for $|e|$.

The determination of d^e and κ^e is strongly related to the closed loop specifications, such as the standard deviation of the noise affecting the measurement, the closed loop time constants, ...

The choice of d^{y_d} and κ^{y_d} depends of the construction of the desired trajectory to be tracked.

All the steps in the AFTC procedure described in the next sections are applied between instants t_{s_i} and t_{f_i}, bearing in mind that several steady-state behaviours may succeed one another during closed-loop operation.

4.2 Residual generation

Model-free fault detection is introduced for actuator fault detection in [31, 32], while for process and sensor faults in [51]. The main idea is to generate a residual signal that is used as a fault indicator. This signal is designed via an estimation of the output y_m of the controlled system. To this end, we use the ultra-local model given by Eq. (5), the estimation of F in Eq. (7) and the control law u in Eq. (6).

The ultra-local model Eq. (5) leads to

$$y_m(t) = \int_0^t (F(\tau) + \alpha u(\tau))\,\mathrm{d}\tau + y_m(0) \tag{8}$$

where $y_m(0)$ is the initial condition. By replacing F by $\widehat{F}$, the measured estimated output $\widehat{y}_m$ is estimated as follows

$$\widehat{y}_m(t) = \int_0^t \left(\widehat{F}(\tau) + \alpha u(\tau)\right) \mathrm{d}\tau + \widehat{y}_m(0) \tag{9}$$

Assuming that $\widehat{F}(t) = F(t)$ and $\widehat{y}_m(0) \simeq y_m(0)$ implies $\widehat{y}_m \simeq y_m$ for any operating system state. However, in practical case, the estimation $\widehat{F}$ is never equal to F, which implies $\widehat{y}_m \neq y_m$ in the presence or absence of a sensor fault f_a or f_m.

It is now a matter of correcting the estimated output $\widehat{y}_m$ so that it is equal to the measured output y_m in the absence of a sensor fault. The sensor fault detection is based on the residual signal given by

$$r(t) = y_m(t) - \beta \widehat{y}_m(t) \tag{10}$$

where β is a parameter to be determined. In order to obtain a residual equal to 0 in absence of sensor fault, this parameter β is defined by

$$\beta(t) = \frac{y_m(t)}{\widehat{y}_m(t)} = \frac{y_m(t)}{\int_0^t \left(\widehat{F}(\tau) + \alpha u(\tau)\right) \mathrm{d}\tau + \widehat{y}_m(0)} \tag{11}$$

where steady-state values of the signals y_m and $\widehat{y}_m$ are used in the absence of the fault.

Theorem 1. *β is constant for systems having linear static characteristic described by Eq. (3).*

To prove this theorem, we need to introduce a continuous integral discretization method for the calculation in Eqs. (9) and (11). This is made in the following remark.

Remark 1. *The approximation $\mathcal{I}(q(k))$ of the temporal integral of the function q by the rectangle method is defined as*

$$\int_0^t q(\tau) d(\tau) \simeq \mathcal{I}(q(k)) = \sum_{i=1}^{k} q(i) T_e$$

and

$$\dot{q}(t) \simeq \delta_q(i) = \frac{q(i) - q(i-1)}{T_e}$$

where T_e is the sampling time. □

Proof. Applying the approximation defined in Remark 1 to Eq. (9) leads to

$$\widehat{y}_m(kT_e) = \mathcal{I}\left(\widehat{F}(kT_e) + \alpha u(kT_e)\right) + \widehat{y}_m(0) \tag{12}$$

where the estimation of F in Eq. (5) is expressed by

$$\widehat{F}(kT_e) = \delta_{y_m}(kT_e) - \alpha u((k-1)T_e) \tag{13}$$

instead of by relation Eq. (7). It should be noted that $u(kT_e)$ cannot be used in Eq. (13) due to causality.

Using the definition of $\delta_{y_m}(kT_e)$ given in Remark 1 and inserting Eq. (13) in Eq. (12) give

$$\widehat{y}_m(kT_e) = \mathcal{I}\left(\delta_{y_m}(kT_e) - \alpha u((k-1)T_e) + \alpha u(kT_e)\right) + \widehat{y}_m(0) \tag{14}$$

where

$$\mathcal{I}(\delta_{y_m}(kT_e)) = \mathcal{I}\left(\frac{y_m(kT_e) - y_m((k-1)T_e)}{T_e}\right)$$
$$= \frac{y_m(T_e) - y(0)}{T_e} \times T_e + \frac{y_m(2T_e) - y_m(T_e)}{T_e} \times T_e + \ldots + \frac{y_m(kT_e) - y_m((k-1)T_e)}{T_e} \times T_e$$
$$= y_m(kT_e) - y_m(0)$$

and

$$\mathcal{I}(\alpha u(kT_e) - \alpha u((k-1)T_e)) = \alpha T_e(u(kT_e) - u(0))$$

Choosing the initial condition $u(0) = 0$ gives

$$\widehat{y}_m(kT_e) = y_m(kT_e) + \alpha T_e u(kT_e) - y_m(0) + \widehat{y}_m(0) \tag{15}$$

Since model-free control guarantees the stability of the controlled system, a steady-state is still achieved and the control input $u(k)$ in steady-state can be expressed as follows

$$u(kT_e) = cy(kT_e) = cy_m(kT_e) \tag{16}$$

where c is defined in Eq. (3). Substituting Eq. (16) in Eq. (15), the estimated measured output $\widehat{y}_m$ becomes

$$\widehat{y}_m(kT_e) = y_m(kT_e)\left(1 + T_e\alpha c(kT_e)\right) - y_m(0) + \widehat{y}_m(0) \tag{17}$$

Since the initial condition is assumed to be approximately known, i.e. $\widehat{y}_m(0) \simeq y_m(0)$, inserting Eq. (17) in Eq. (11) yields

$$\beta(kT_e) = \frac{y_m(kT_e)}{y_m(kT_e)\left(1 + T_e\alpha c\right)} = \frac{1}{1 + T_e\alpha c} = \beta \tag{18}$$

This proves that, for any change in desired trajectory, the parameter $\beta(kT_e)$ converges to the same value β. So in the absence of sensor fault, the residual signal $r(t)$ in Eq. (10) is null in steady-state with β given by Eq. (18). $\square$

5 Model-free sensor fault estimation

Unlike the second step of sensor AFTC corresponding to the sensor fault detection described in Section 4 works both for additive and multiplicative sensor faults, the third step corresponding to sensor fault estimation should be treated separately for additive and multiplicative cases.

5.1 Additive sensor fault estimation

Consider that the measured output is affected by an additive sensor fault, that means that $\rho = 1$ in Eq. (1d). Then the regulated output is expressed as follows

$$y = y_m - f_a, \qquad f_a \neq 0 \tag{19}$$

where f_a is an additive sensor fault. It is important to remember that β was determined before the fault occurred as in Eq. (11), i.e. when the measured output y_m was equal to the regulated output y. So Eq. (16) becomes

$$u(kT_e) = cy(kT_e) = c(y_m(kT_e) - f_a(kT_e)) \tag{20}$$

and Eq. (15) is expressed as

$$\widehat{y}_m(kT_e) = y_m(kT_e) + \alpha T_e c(y_m(kT_e) - \widehat{f}_a(kT_e)) \tag{21}$$

where the initial condition is chosen by $\widehat{y}_m(0) \simeq y_m(0)$, and $\widehat{f}_a$ is the estimation of the sensor fault f_a.

To determine the estimation $\widehat{f}_a$, we proceed as follows. Inserting β given by Eq. (18) and $\widehat{y}_m$ given by Eq. (21) in Eq. (10) where $\widehat{f}_a$ is replaced by f_a, we obtain

$$\begin{aligned} r(kT_e) &= y_m(kT_e) - \frac{1}{1 + T_e\alpha c}\Big(y_m(kT_e)(1 + \alpha T_e c) - \alpha T_e c \widehat{f}_a(kT_e)\Big) \\ &= \frac{\alpha T_e c \widehat{f}_a(kT_e)}{1 + \alpha T_e c} \\ &= \widehat{f}_a(kT_e)\left(1 - \frac{1}{1 + \alpha T_e c}\right) = \widehat{f}_a(kT_e)(1 - \beta) \end{aligned} \tag{22}$$

Using Eq. (22), the best estimation of sensor fault f_a is then given by

$$\widehat{f}_a(kT_e) = \frac{r(kT_e)}{1 - \beta} \tag{23}$$

where $\beta \neq 1$ due to Eq. (18).

The determination of the threshold of $r(kT_e)$ is made with respect to the chosen minimal absolute value $\widetilde{f}_a$ of the sensor fault f_a to be detected and estimated. So using Eq. (23), this threshold th is computed as

$$th(kT_e) = \widetilde{f}_a\,|1 - \beta| \tag{24}$$

Remark 2. *For additive sensor fault f_a, the calculated th is constant for any reached steady-states. In the sequel, the following notations are used $th^+ = th$ and $th^- = -th$.*

5.2 Multiplicative sensor fault estimation

Consider now that $\rho = 0$ in Eq. (1d). Then the regulated output is expressed by

$$y = \frac{y_m}{f_m} \tag{25}$$

and Eq. (16) becomes

$$u(kT_e) = \frac{c y_m(kT_e)}{f_m(kT_e)} \tag{26}$$

When the initial condition is chosen by $\widehat{y}_m(0) \simeq y_m(0)$, the estimated output $\widehat{y}_m$ in Eq. (15) can be rewritten as

$$\widehat{y}_m(kT_e) = y_m(kT_e) + \alpha T_e \frac{c y_m(kT_e)}{\widehat{f}_m(kT_e)} \tag{27}$$

Inserting Eqs. (18) and (27) in Eq. (10), the residual r is expressed by

$$\begin{aligned} r(kT_e) &= y_m(kT_e) - \beta \left(y_m(kT_e) + \alpha T_e \frac{c y_m(kT_e)}{\widehat{f}_m(kT_e)} \right) \\ &= y_m(kT_e) \left(1 - \beta - \frac{(1-\beta)}{\widehat{f}_m(kT_e)} \right) \end{aligned} \tag{28}$$

Using Eq. (28), the best estimation of sensor fault f_m is then given by

$$\widehat{f}_m(kT_e) = \frac{y_m(kT_e)(\beta - 1)}{y_m(kT_e)(\beta - 1) + r(kT_e)} \tag{29}$$

The determination of the threshold of $r(kT_e)$ is made with respect to the chosen maximal value $0 < \widetilde{f}_m < 1$ of the sensor fault f_m to be detected and estimated. So using Eq. (29), this threshold th is computed as

$$\begin{cases} th(kT_e) = \dfrac{|(\beta-1)y_m(kT_e)|\,(1-\widetilde{f}_m)}{\widetilde{f}_m} & \text{if } kT_e \notin [t_{s_i}; t_{f_i}] \\ th(kT_e) = \dfrac{|(\beta-1)y_m(t_{s_i})|\,(1-\widetilde{f}_m)}{\widetilde{f}_m} & \text{if } kT_e \in [t_{s_i}; t_{f_i}] \end{cases} \tag{30}$$

where $y_m(t_{s_i})$ is the output measurement at the beginning of each reached steady-state (see Section 4.1).

Remark 3. *For multiplicative sensor fault f_m, the calculated th varies with respect to $y_m(t_{s_i})$. In the sequel, the following notations are used $th^+(kT_e) = th(kT_e)$ and $th^-(kT_e) = -th(kT_e)$. If $kT_e \in [t_{s_i}; t_{f_i}]$, we should check that $th(kT_e)$ is superior to ψ times the standard deviation of the noise affecting the residual $r(kT_e)$, where $\psi > 1$. If it is not the case the value of $\widetilde{f}_m$ should be reduced.*

Remark 4. *Outside the context of this work, the choice of* $\widehat{f}_a$ *in Eq.* (24) *and* $\widehat{f}_m$ *Eq.* (30) *or threshold is not trivial. Indeed, it plays an important role in the trade-off between non-detection and false alarm or, more generally, fault sensitivity [53].*

6 Sensor fault accommodation

The proposed AFTC strategy is based on the estimation of the sensor fault given by Eq. (23) for additive sensor fault and by Eq. (29) for multiplicative sensor fault.

Once the residual signal $r(t)$ in Eq. (10) exceeds a given threshold, the sensor fault is detected and the control law given by Eq. (6) is adapted to tolerate the sensor fault.

The tracking error used in the control input in Eq. (6) is replaced by

$$e(t) = y_d(t) - y_{acc}(t) \tag{31}$$

where $y_{acc}(t)$ is the accommodation signal given by

$$y_{acc}(t) = y_m(t) - \widehat{f}_a(t) \tag{32}$$

in the case of additive sensor fault (see Section 5.1) and by

$$y_{acc}(t) = \frac{y_m(t)}{\widehat{f}_m(t)} \tag{33}$$

in the case of multiplicative sensor fault (see Section 5.2).

The following diagram Figure 1 provides a complete summary of the proposed FDIA procedure, described in Sections 4 to 6.

7 First example: unstable linear system

Consider an unstable linear system described by

$$\dot{x}(t) = Ax(t) + Bu(t) \tag{34a}$$
$$y(t) = Cx(t) + Du(t) \tag{34b}$$
$$y_m(t) = \rho(y + f_a(t)) + (1-\rho)f_m(t)y(t) + w(t) \tag{34c}$$

where

$$A = \begin{bmatrix} -17.5 & -91 & 50 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}, B = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}, C = \begin{bmatrix} 0 & 0 & 100 \end{bmatrix}, D = 0$$

and $w(t)$ is a zero-mean white Gaussian noise with standard deviation $\sigma_w = 0.01416$.

The parameters of the iP controller are $k_p = 0.145$ and $\alpha = 2$ with the sampling time $T_e = 0.001\,\mathrm{s}$.

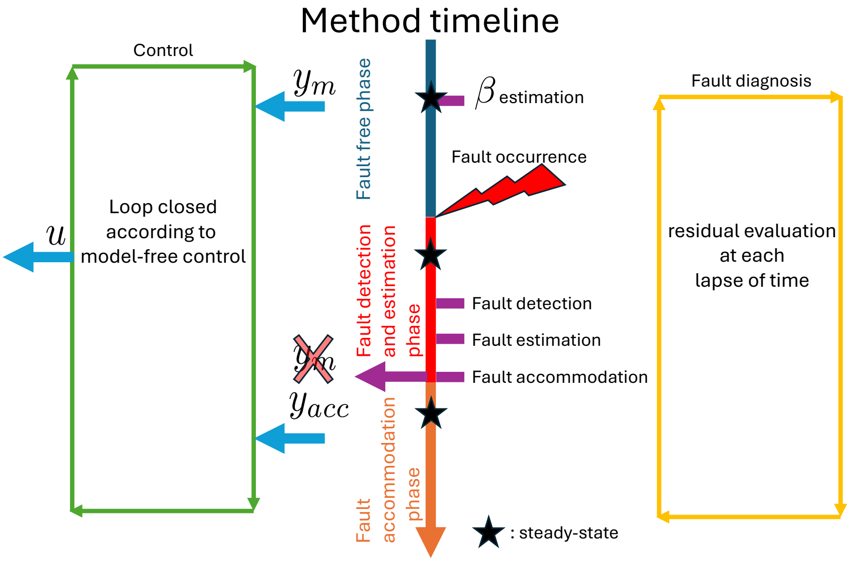

Fig. 1 Timeline of the proposed model-free FDIA method

To determine the steady-state behaviour in which the AFTC works, the following parameters have been chosen: $d^{y_d} = 60T_e$, $d^e = 300T_e$, $\kappa^{y_d} = 9 \times 10^{-4}$ and $\kappa^e = 3.5\sigma_w$.

After the first change of the desired trajectory y_d either in Figure 2 or in Figure 12, the signals y_m and $\widehat{y}_m$ are used to determine the parameter β as in Eq. (11). The obtained value of β is 1.111.

To attenuate the effects of the noise w, the signals r, $\widehat{f}_a$ and $\widehat{f}_m$ are filtered with a filter having the following transfer function $\dfrac{4 \times 10^4}{s^2 + 400s + 4 \times 10^4}$ and the obtained filtered noise has a standard deviation $\sigma_{w_f} = 0.0031418$.

In Subsections 7.1.2, 7.1.3, 7.2.2 and 7.2.3, the control input $u(t)$ is saturated as follows: if $u(t) \leqslant -2.2$ than $u(t) = -2.2$, else $u(t)$ is not saturated.

Remark 5. *In the sequel of Section 7 and in Sections 8 and 9, the following convention is made: in the figures associated with the residual $r(t)$, either the fault f_a or f_m, areas colored green do not correspond to steady-state behaviors in which AFTC is not applied, while areas colored white correspond to steady-state behaviors in which the AFTC procedure is performed.*

7.1 Additive fault case

The additive sensor fault f_a in Eq. (34c) with $\rho = 1$ is introduced as follows

$$f_a(t) = \begin{cases} 0, & t < t_f \\ \mu(t), & t \geqslant t_f \end{cases}$$

where $\mu(t)$ is the unitary step response of transfer function $\dfrac{-1.3}{(s+1)^2}$ at time instant $t_f = 33\,\mathrm{s}$.

We chose $\widetilde{f}_a = 0.27$, using Eq. (24) the obtained $th = 0.02997$. This value is compatible with the filtered noise since $th \approx 9.54\sigma_{w_f}$.

7.1.1 Sensor fault detection and estimation without accommodation and without control input saturation

Figures 2, 3, 4 and 5 deal with the situation where there is no sensor fault accommodation and no saturation on the control input.

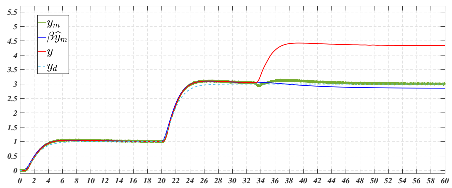

Fig. 2 Trajectory tracking: y_m, $\beta\widehat{y}_m$, y and y_d

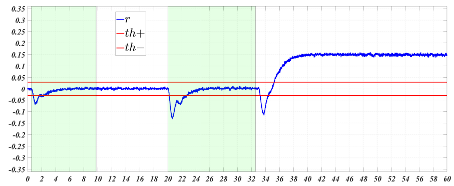

Fig. 3 Residual $r(t)$

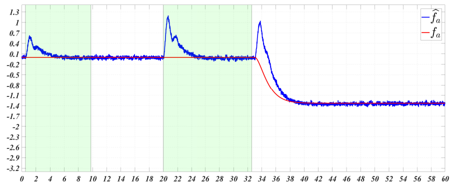

Fig. 4 Fault f_a and its estimate $\widehat{f}_a$

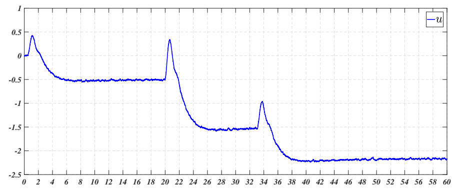

Fig. 5 Control input $u(t)$

7.1.2 Sensor fault detection and estimation without accommodation and with control input saturation

Figures 6 and 7 deal with the situation where there is no sensor fault accommodation, but where the control input is saturated.

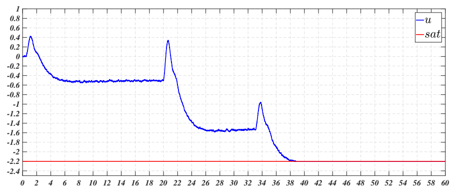

Fig. 6 Saturated control input $u(t)$

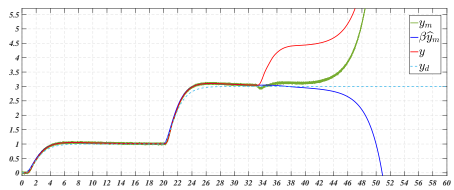

Fig. 7 Trajectory tracking: y_m, $\beta\widehat{y}_m$, y and y_d

7.1.3 Sensor fault detection and estimation with accommodation and control input saturation

Figures 8, 9, 10 and 11 deal with the situation where there is sensor fault accommodation and the control input is saturated.

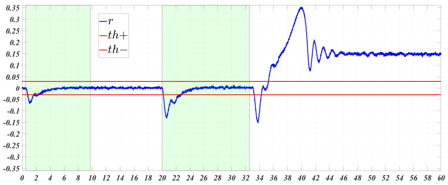

Fig. 8 Residual $r(t)$

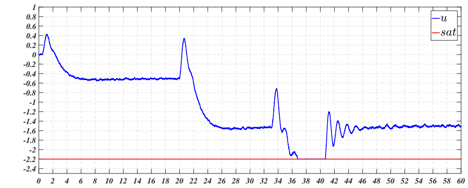

Fig. 9 Saturated control input $u(t)$

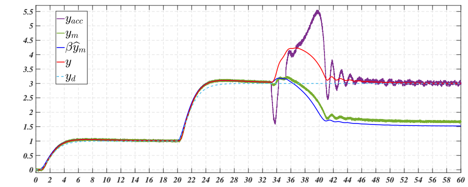

Fig. 10 Trajectory tracking: y_m, $\beta\widehat{y}_m$, y and y_d

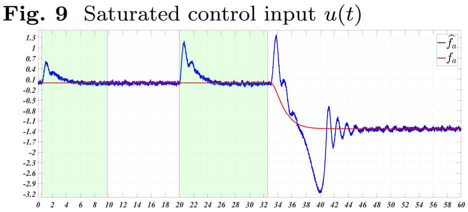

Fig. 11 Fault f_a and its estimate $\widehat{f}_a$

7.2 Multiplicative fault case

The multiplicative sensor fault f_m in Eq. (34c) with $\rho = 0$ is introduced as follows

$$f_m(t) = \begin{cases} 0, & t < t_f \\ \mu(t), & t \geqslant t_f \end{cases}$$

where $\mu(t)$ is the unitary step response of transfer function $\dfrac{0.7}{(s+1)^2}$ at time instant $t_f = 33\,\mathrm{s}$.

We chose $\widetilde{f}_m = 0.92$, using Eq. (30) the obtained $th \geqslant \psi\sigma_{w_f}$ where $\psi > 4$.

7.2.1 Sensor fault detection and estimation without accommodation and without control input saturation

Figures 12, 13, 14 and 15 deal with the situation where there is no sensor fault accommodation and no saturation on the control input.

7.2.2 Sensor fault detection and estimation without accommodation and with control input saturation

Figures 16 and 17 deal with the situation where there is no sensor fault accommodation, but where the control input is saturated.

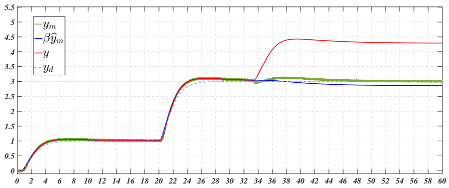

Fig. 12 Trajectory tracking: y_m, $\beta\widehat{y}_m$, y and y_d

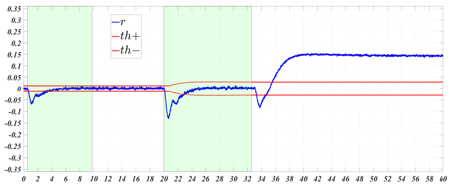

Fig. 13 Residual $r(t)$

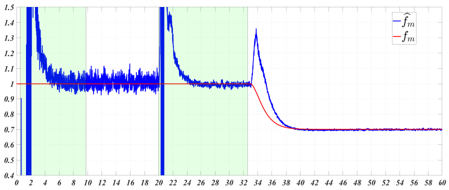

Fig. 14 Fault f_m and its estimate $\widehat{f}_m$

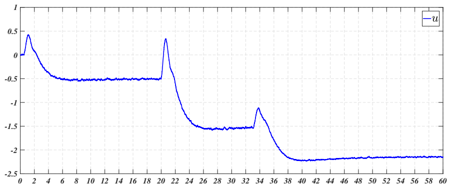

Fig. 15 Control input $u(t)$

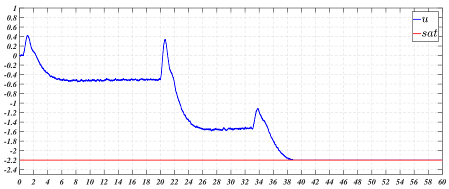

Fig. 16 Saturated control input $u(t)$

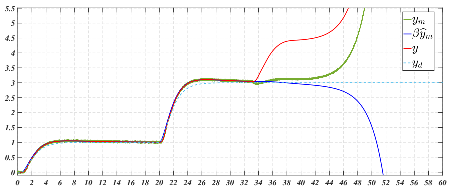

Fig. 17 Trajectory tracking: y_m, $\beta\widehat{y}_m$, y and y_d

7.2.3 Sensor fault detection and estimation with accommodation and control input saturation

Figures 18, 19, 20 and 21 deal with the situation where there is sensor fault accommodation and the control input is saturated.

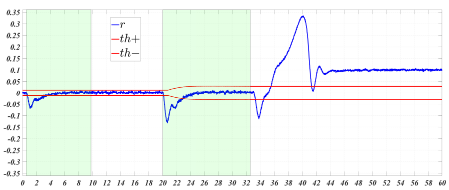

Fig. 18 Residual $r(t)$

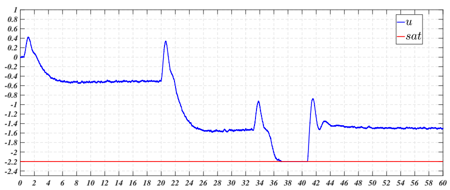

Fig. 19 Saturated control input $u(t)$

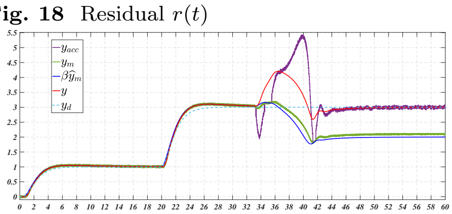

Fig. 20 Trajectory tracking: y_m, $\beta\widehat{y}_m$, y and y_d

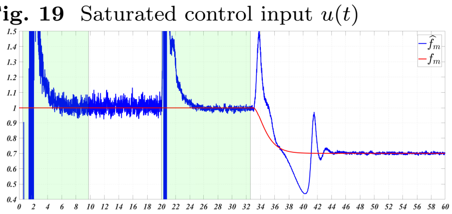

Fig. 21 Fault f_m and its estimate $\widehat{f}_m$

8 Second example: unstable nonlinear system with additive sensor fault

Consider an unstable nonlinear system described by

$$\dot{x}^3(t) = 2x(t) + 5u(t) \tag{35a}$$
$$y(t) = x(t) \tag{35b}$$
$$y_m(t) = y + f_a(t) + w(t) \tag{35c}$$

where $\rho = 1$ in Eq. (1d) and $w(t)$ is a zero-mean with standard deviation $\sigma_w = 0.01416$.

The additive sensor fault f_a in Eq. (35c) is introduced as follows

$$f_a(t) = \begin{cases} 0, & t < t_f \\ \mu(t), & t \geqslant t_f \end{cases}$$

where $\mu(t)$ is the unitary step response of transfer function $\dfrac{-2}{(s+1)^2}$ at time instant $t_f = 33\,\mathrm{s}$.

The parameters of the iP controller are $k_p = 3$ and $\alpha = 1$ with the sampling time $T_e = 0.001\,\mathrm{s}$.

To determine the steady-state behaviour in which the AFTC works, the following parameters have been chosen: $d^{y_d} = 60T_e$, $d^e = 300T_e$, $\kappa^{y_d} = 9 \times 10^{-4}$ and $\kappa^e = 3.5\sigma_w$.

After the first change of the desired trajectory y_d in Figure 22, the signals y_m and $\widehat{y}_m$ are used to determine the parameter β as in Eq. (11). The obtained value of β is 1.2308.

To attenuate the effects of the noise w, the signals r and $\widehat{f}_a$ are filtered with a filter having the following transfer function $\dfrac{10^4}{s^2 + 200s + 10^4}$ and the obtained filtered noise has a standard deviation $\sigma_{w_f} = 0.0022345$.

In Subsections 8.2 and 8.3 the control input $u(t)$ is saturated as follows: if $u(t) \leqslant -9.5$ than $u(t) = -9.5$, else $u(t)$ is not saturated.

We chose $\widetilde{f}_a = 0.3$, using Eq. (24) the obtained $th = 0,06924$. This value is compatible with the filtered noise since $th \approx 30.98\sigma_{w_f}$.

8.1 Sensor fault detection and estimation without accommodation and without control input saturation

Figures 22, 23, 24 and 25 deal with the situation where there is no sensor fault accommodation and no saturation on the control input.

8.2 Sensor fault detection and estimation without accommodation and with control input saturation

Figures 26 and 27 deal with the situation where there is no sensor fault accommodation, but where the control input is saturated.

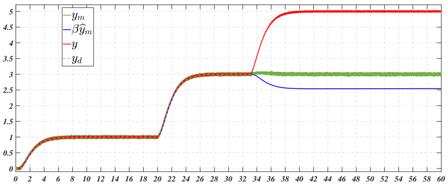

Fig. 22 Trajectory tracking: y_m, $\beta\widehat{y}_m$, y and y_d

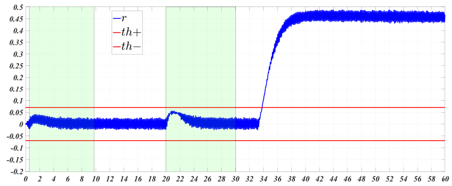

Fig. 23 Residual $r(t)$

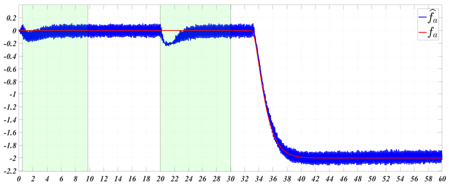

Fig. 24 Fault f_a and its estimate $\widehat{f}_a$

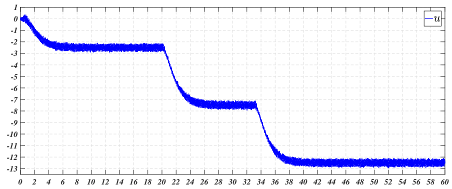

Fig. 25 Control input $u(t)$

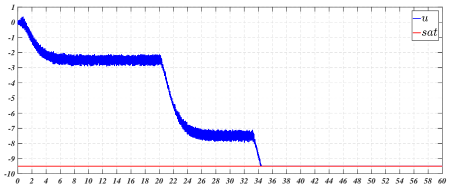

Fig. 26 Saturated control input $u(t)$

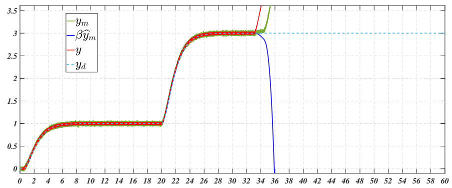

Fig. 27 Trajectory tracking: y_m, $\beta\widehat{y}_m$, y and y_d

8.3 Sensor fault detection and estimation with accommodation and control input saturation

Figures 28, 29, 30 and 31 deal with the situation where there is sensor fault accommodation and the control input is saturated.

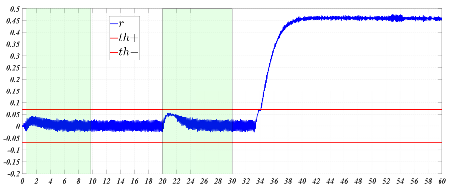

Fig. 28 Residual $r(t)$

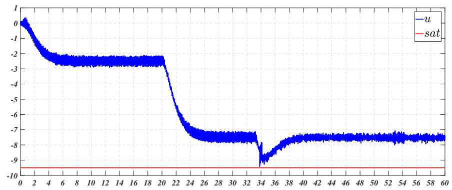

Fig. 29 Saturated control input $u(t)$

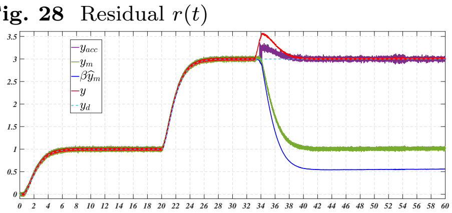

Fig. 30 Trajectory tracking: y_m, $\beta\widehat{y}_m$, y and y_d

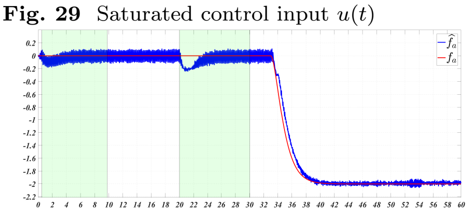

Fig. 31 Fault f_a and its estimate $\widehat{f}_a$

9 Third example: unstable nonlinear system with multiplicative sensor fault

Consider an unstable nonlinear system described by

$$\dot{x}(t) = 1.5^3 x^3(t) + 1.5u^3(t) \tag{36a}$$
$$y(t) = x(t) \tag{36b}$$
$$y_m(t) = f_m(t)y(t) + w(t) \tag{36c}$$

where $\rho = 0$ in Eq. (1d) and $w(t)$ is a zero-mean white Gaussian noise with standard deviation $\sigma_w = 0.01416$.

The additive sensor fault f_m in Eq. (36c) is introduced as follows

$$f_m(t) = \begin{cases} 0, & t < t_f \\ \mu(t), & t \geqslant t_f \end{cases}$$

where $\mu(t)$ is the unitary step response of transfer function $\dfrac{0.55}{(s+1)^2}$ at time instant $t_f = 33\,\mathrm{s}$.

The parameters of the iP controller are $k_p = 10$ and $\alpha = 2.5$ with the sampling time $T_e = 0.001\,\mathrm{s}$.

To determine the steady-state behaviour in which the AFTC works, the following parameters have been chosen: $d^{y_d} = 60T_e$, $d^e = 300T_e$, $\kappa^{y_d} = 9 \times 10^{-4}$ and $\kappa^e = 3.5\sigma_w$.

After the first change of the desired trajectory y_d in Figure 22, the signals y_m and $\widehat{y}_m$ are used to determine the parameter β as in Eq. (11). The obtained value of β is 1.97.

To attenuate the effects of the noise w, the signals r and $\widehat{f}_m$ are filtered with a filter having the following transfer function $\dfrac{10^4}{s^2 + 200s + 10^4}$ and the obtained filtered noise has a standard deviation $\sigma_{w_f} = 0.0022345$.

In Subsections 9.2 and 9.3 the control input $u(t)$ is saturated as follows: if $u(t) \leqslant -5$ than $u(t) = -5$, else $u(t)$ is not saturated.

We chose $\widetilde{f}_m = 0.92$, using Eq. (30) the obtained $th \geqslant \psi\sigma_{w_f}$ where $\psi > 4$.

9.1 Sensor fault detection and estimation without sensor fault accommodation and without control input saturation

Figures 32, 33, 34 and 35 deal with the situation where there is no sensor fault accommodation and no saturation on the control input.

9.2 Sensor fault detection and estimation without accommodation and with control input saturation

Figures 36, 37 and 38 deals with the situation where there is no sensor fault accommodation, but where the control input is saturated.

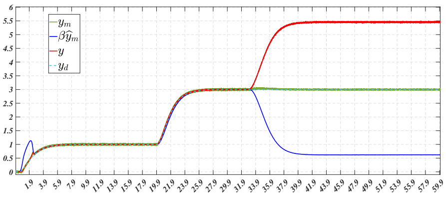

Fig. 32 Trajectory tracking: y_m, $\beta\widehat{y}_m$, y and y_d

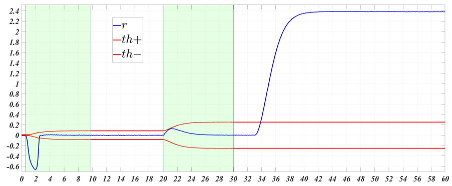

Fig. 33 Residual $r(t)$

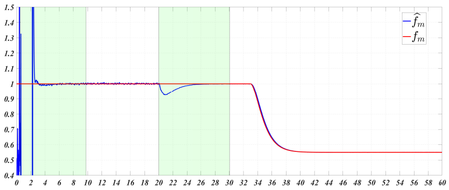

Fig. 34 Fault f_m and its estimate $\widehat{f}_m$

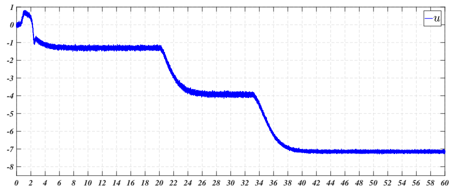

Fig. 35 Control input $u(t)$

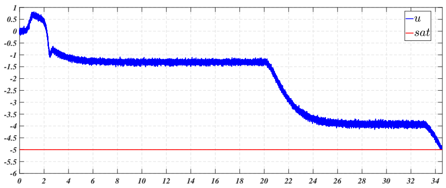

Fig. 36 Saturated control input $u(t)$

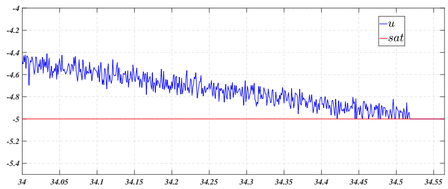

Fig. 37 Zoom on Figure 36

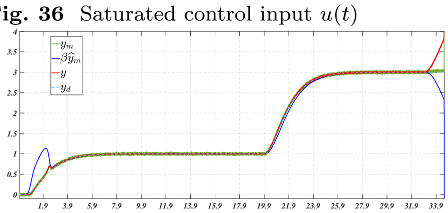

Fig. 38 Trajectory tracking: y_m, $\beta\widehat{y}_m$, y and y_d

9.3 Sensor fault detection and estimation with accommodation and control input saturation

Figures 39, 40, 41 and 42 deal with the situation where there is sensor fault accommodation and the control input is saturated.

10 Discussion

For systems considered in Sections 7, 8 and 9, which are unstable, the simulation scenario is divided into three parts where the sensor fault is either additive or multiplicative:

- in the 1st part, the effects of the sensor fault on the closed loop tracking objectives are illustrated when the fault accommodation procedure is not used and where the control input is not saturated (see Subsection 10.1);

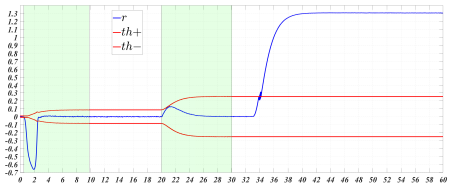

Fig. 39 Residual $r(t)$

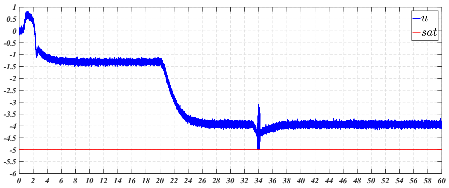

Fig. 40 Saturated control input $u(t)$

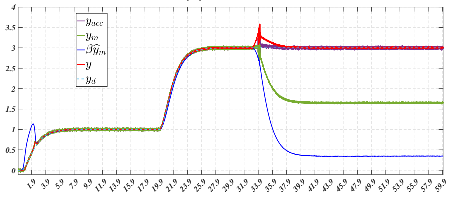

Fig. 41 Trajectory tracking: y_m, $\beta\widehat{y}_m$, y and y_d

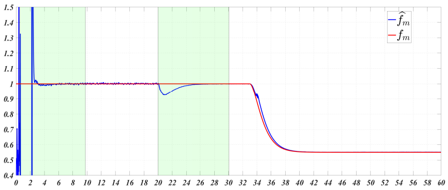

Fig. 42 Fault f_m and its estimate $\widehat{f}_m$

- in the 2nd part, the effects of the sensor fault with control input saturation on the closed loop behaviour are shown when the fault accommodation technique is not applied (see Subsection 10.2);
- in the 3rd part, the proposed fault accommodation procedure is applied where the control input is saturated (see Subsection 10.3).

In the three parts of the scenario, the 1st and 2nd steps of the AFTC procedure, i.e. the sensor faut detection and estimation (see Sections 4 and 5), respectively, are applied, while the 3rd step of the AFTC procedure given by the accommodation of the control law (see Sections 6) is used in association with the 1st and 2nd steps of the AFTC approach.

In the figures associated with the residual $r(t)$, either the fault f_a or f_m, areas colored green do not correspond to steady-state behaviors in which AFTC is not applied, while areas colored white correspond to steady-state behaviors in which the AFTC procedure is performed. These areas are obtained using parameters d^{y_d}, d^e, κ^{y_d} and κ^e determined from the closed loop response generated by the model-free controller.

10.1 1st part: sensor fault detection and estimation without accommodation and without control input saturation

This part of this discussion concerns Subsubsections 7.1.1 and 7.2.1 for linear cases and Subsections 8.1 and 9.1 for nonlinear cases.

- Figures 2, 12, 22 and 32: Before the sensor fault occurrence ($t < t_f$) the iP controller works well: the regulated output y tracks the desired trajectory y_d with very slight overshoot. Once the sensor fault occurred at time instant t_f, the iP controller only ensures correct control of the measured output y_m which tracks the desired trajectory y_d, but the regulated output y is not maintained at the desired trajectory y_d, i.e. the control objective is not satisfied in this situation and the accommodation step of the AFTC is mandatory.

- Figures 3, 13, 23 and 33: Before sensor fault occurrence ($t < t_f$) the residual r returns to 0 after the transient due to the change of the desired trajectory y_d. The residual r exceeds the threshold at $t = t_f + \delta_{t_f}$ which means that the sensor fault is detected at this instant: $\beta\widehat{y}_m \neq y_m$ at $t = t_f + \delta_{t_f}$ in Figures 2, 12, 22 and 32, where $\delta_{t_f} \leqslant 1\,\mathrm{s}$.
- Figures 4, 14, 24 and 34: The proposed method of sensor fault estimation works well: the sensor fault estimation converges to the actual sensor fault value.
- Figures 5, 15, 25 and 35: The effects of the sensor fault on the control input u at instant t_f can be viewed.

10.2 2nd part: sensor fault detection and estimation without accommodation and with control input saturation

This part of this discussion concerns Subsubsections 7.1.2 and 7.2.2 for linear cases and Subsections 8.2 and 9.2 for nonlinear cases.

- Figures 6, 16, 26, 36 and 37: Note that the control input u crosses saturation threshold after the occurrence of the sensor fault (see Figures 5, 15, 25 and 35), so the control input u becomes saturated. To see the saturation on the control input in Figure 36, a zoom is performed in Figure 37 since the measured output y_m and the regulated output y go so fast towards infinity in Figure 38 that the simulation stopped at $t = 34.55\,\mathrm{s}$.
- Figures 7, 17, 27 and 38: Since the systems to be controlled are unstable, the regulated output y diverges after the saturation of the control input u. So, it is mandatory to stabilize the closed loop and to ensure the tracking objective.

10.3 3rd part: sensor fault detection and estimation with accommodation and with control input saturation

This part of this discussion concerns Subsubsections 7.1.3 and 7.2.3 for linear cases and Subsections 8.3 and 9.3 for nonlinear cases.

- Figures 8, 18, 28 and 39: For the residual signal $r(t)$, the same comments as in Figures 3, 13, 23 and 33 can be made (see Subsection 10.1).
- Figures 9, 19, 29 and 40: The accommodation step of the AFTC acts as soon as the sensor fault is detected, but this generates saturation of the control input $u(t)$ (i.e. $u(t)$ falls below the saturation threshold). However, the proposed AFTC strategy is able to overcome this saturation at time instant $t > t_f + 1\,\mathrm{s}$ and the control input $u(t)$ is approximately the same before saturation after a short transient behaviour.
- Figures 10, 20, 30 and 41: The whole AFTC procedure (i.e with the accommodation step) works well since the closed loop is stabilized and the regulated output y converges to the desired trajectory y_d with a short stabilization time as soon as the control input u returns to the unsaturated zone.
- Figures 11, 21, 31 and 42: The convergence of the estimate of the sensor fault to the actual value confirms the efficiency of the proposed model-free sensor fault estimation procedure.

10.4 On the model-free framework for AFTC

In papers [11] to [27] cited in the introduction section, the AFTC strategies are based on the knowledge of the process model (use of an observer bank, an unknown input observer bank, a parity space, an analytical redundancy, ...), by considering two cases either with or without unstructured uncertainties. The efficiency of these AFTC procedures are therefore linked to the accuracy of the model parameters and the determination of uncertainty limits. In our model-free sensor AFTC approach, only the knowledge of the static characteristic is needed and this one can be easily obtained by the available closed-loop signals (i.e. u and y_m before the occurrence of the fault). This makes the proposed approach well suitable for industrial applications, where the process model is often unknown or almost unknown, and where the bounds of uncertainty are difficult to assess.

11 Conclusion

In this paper, Problem 1 stated in Section 2 has been solved where all the steps of the sensor fault AFTC procedure are designed in a model-free framework. This approach allows avoiding the use of the analytical system model, knowing that the latter is unknown or subjected to uncertainties in almost practical industrial situations. Only the static characteristic obtained from the control input and measurement signals is used. The proposed AFTC method works for both additive and multiplicative sensor fault and ensures that the regulated output, but not the measured one, tracks the desired trajectory despite the occurrence of a sensor fault.

The results of the numerical simulations illustrate the ability of the proposed model-free AFTC technique to accommodate the additive or multiplicative sensor fault for both unstable linear and nonlinear systems with control input saturation.

In future work, the proposed AFTC method will be extended to simultaneous actuator and sensor faults for multi-input multi-output systems.

References

[1] Jiang, J. Fault-tolerant control systems - an introductory overview. *Acta Automatica Sinica* **31**, 161–174 (2005).

[2] Lunze, J. & Richter, J. Reconfigurable fault-tolerant control: a tutorial introduction. *European J. Contr.* **14**, 359–386 (2008).

[3] Gao, Z., Cecati, C. & Ding, S. A survey of fault diagnosis and fault-tolerant techniques. Part I: fault diagnosis with model-based and signal-based approaches. *IEEE Trans. Industrial Electronics* **62**, 3757–3767 (2015).

[4] Gao, Z., Cecati, C. & Ding, S. A survey of fault diagnosis and fault-tolerant techniques. Part II: fault diagnosis with knowledge-based and hybrid/active approaches. *IEEE Trans. Industrial Electronics* **62**, 3768–3774 (2015).

[5] Park, Y., Fan, S. & Hsu, C. A review on fault detection and process diagnostics in industrial processe. *Processes* **8**, ID 1123 (2020).

[6] Noura, H., Theilliol, D., Ponsart, J. & Chamseddiner, A. *Fault-Tolerant Control Systems: Design and Practical Applications* (Springer-Verlag, London, 2009).

Table 1 Variables and signals

x	state
u	control input
c	parameter defining linear static characteristic
y	regulated output
y_m	measured output
ν	derivative order of the measured output y_m
α	iP controller parameter
$\widehat{y}_m$	estimated output
y_{acc}	accommodated output
y_d	desired trajectory
β	parameter used to correct the estimated output $\hat{y}_m$
f_a	additive sensor fault
$\widehat{f}_a$	estimated additive sensor f_a
$\widetilde{f}_a$	minimal value of the sensor fault f_a to be detected and estimated
f_m	multiplicative sensor fault
$\widehat{f}_m$	estimated multiplicative sensor f_m
$\widetilde{f}_m$	minimal value of the sensor fault f_m to be detected and estimated
e	tracking error
k_p	tuning proportional gain
T	windows of integration
T_e	sampling time
t_{s_i}	start instant of steady-state behaviour
t_{f_i}	final instant of steady-state behaviour
t_f	time of sensor fault occurrence
κ^{y_d}	magnitude of $\dot{y}_d$ variations used to define the steady-state behaviour
d^{y_d}	chosen duration on $\dot{y}_d$ used to define the steady-state behaviour
κ^e	magnitude of e variations used to define the steady-state behaviour
d^e	chosen duration on e used to define the steady-state behaviour
r	residual signal used for fault detection
th^+	maximum threshold
th^-	minimum threshold
w	measurement noise
σ_w	noise standard deviation

[7] Wang, R. & Wang, J. Passive actuator fault-tolerant control for a class of over-actuated nonlinear systems and applications to electric vehicles. *IEEE Trans. Vehicular Technology* **62**, 972–985 (2012).

[8] Amin, A. & Hasan, K. A review of fault tolerant control systems: advancements and applications. *Measurement* **143**, 58–68 (2019).

[9] Jiang, J. & Yu, X. Fault-tolerant control systems: a comparative study between active and passive approaches. *Annual Reviews in Control* **36**, 60–72 (2012).

[10] Abbaspour, A., Mokhtari, S., Sargolzaei, A. & Yen, K. A survey on active fault-tolerant control systems. *Electronics* **9**, ID 1513 (2020).

[11] Gao, Z., Han, B., Jiang, G., Lin, J. & Xu, D. Active fault tolerant control design approach for the flexible spacecraft with sensor faults. *J. of the Franklin Institute* **354**, 8038–8056 (2017).

[12] Zhu, Z., Chen, T., Zeng, Y. & Wang, C. Deterministic learning-based sensor fault detection, isolation and accommodation of nonlinear systems. *Int. J. Control* (2023). Https://doi.org/10.1080/00207179.2023.2260904.

[13] Du, M. & Mhaskar, P. Isolation and handling of sensor faults in nonlinear systems. *Automatica* 1066–1074 (2014).

[14] Davoodi, M., Meskin, N. & Khorasani, K. A single dynamic observer-based module for design of simultaneous fault detection, isolation and tracking control scheme. *Int. J. Control* **91**, 508–523 (2018).

[15] Tahoun, A. Fault-tolerant control for a class of quantised networked control of nonlinear systems with unknown time-varying sensor faults. *Int. J. Control* **93**, 619–628 (2020).

[16] Seo, C. & Kim, B. Design of robust reliable $\mathcal{H}_\infty$ output feedback control for a class of uncertain linear systems with sensor failure. *Int. J. Syst. Sci.* **27**, 963–968 (1996).

[17] Chen, L., Alwi, H. & Edwards, C. On the synthesis of an integrated active LPV FTC scheme using sliding modes. *Automatica* **110**, ID 108536 (2019).

[18] Qin, L., He, X. & Zhou, D. Fault-tolerant cooperative output regulation for multi-vehicle systems with sensor faults. *Int. J. Control* **90**, 2227–2248 (2017).

[19] Theilliol, D., Noura, H. & Ponsart, J. Fault diagnosis and accommodation of a three-tank system based on analytical redundancy. *ISA Transactions* **41**, 365–382 (2002).

[20] Khebbache, H., Tadjine, M. & Labiod, S. Adaptive sensor-fault tolerant control for a class of mimo uncertain nonlinear systems: Adaptive nonlinear filter-based dynamic surface control. *J. of the Franklin Institute* **353**, 1313–1338 (2016).

[21] Cai, X. & Wu, F. Robust parameter-dependent fault-tolerant control for actuator and sensor faults. *Int. J. Control* **83**, 1475–1484 (2010).

[22] Yetendje, A., De Doná, J. & Seron, M. Multisensor fusion fault tolerant control. *Automatica* **47**, 1461–1466 (2011).

[23] Richter, J., Heemels, W., van de Wouw & Lunze, J. Reconfigurable control of piecewise affine systems with actuator and sensor faults: stability and tracking. *Automatica* **47**, 678–691 (2011).

[24] Seron, M., De Doná, J. & Richter, J. Integrated sensor and actuator fault-tolerant control. *Int. J. Control* **86**, 689–708 (2013).

[25] Seron, M. & De Doná, J. Robust fault estimation and compensation for LPV systems under actuator and sensor faults. *Automatica* **52**, 294–301 (2015).

[26] Wu, H. & Zhang, H. Reliable mixed $\mathcal{L}_2/\mathcal{H}_\infty$ fuzzy static output feedback control for nonlinear systems with sensor faults. *Automatica* **41**, 1925–1932 (2005).

[27] Seron, M., Zhuo, X., De Doná, J. & Martínez, J. Multisensor switching control strategy with fault tolerance guarantees. *Automatica* **44**, 88–97 (2008).

[28] Fliess, M. & Join, C. Model-free control. *Int. J. Control* **86**, 2228–2252 (2013).

[29] Fliess, M. & Join, C. An alternative to proportional-integral and proportional-integral-derivative regulators: Intelligent proportional-derivative regulators. *Int. J. Robust & Nonlinear Contr.* **32**, 9512–9524 (2022).

[30] Lafont, F., Balmat, J., Pessel, N. & Fliess, M. A model-free control strategy for an experimental greenhouse with an application to fault accommodation. *Computers and Electronics in Agriculture* **110**, 139–149 (2015).

[31] Ait Ziane, M. *et al.* A new method for fault detection in a free model context (Pafos, Cyprus, 2022). *Proc. IFAC SAFEPROCESS Symp.*

[32] Ait Ziane, M. *et al.* Model-free fault detection: application to polymer electrolyte fuel cell system (Marseille, France, 2022). *Proc. IEEE Int. Conf. on Systems and Control.*

[33] Ait Ziane, M., Zasadzinski, M., Join, C. & Péra, M. Model free active fault tolerant control for sensor fault (Valletta, Malta, 2024). *Proc. IEEE Int. Conf. Control, Decision and Information Technologies.*

[34] Fliess, M. & Sira-Ramírez, H. in *Closed-loop parametric identification for continuous-time linear systems via new algebraic technique* (eds Garnier, H. & Wang, L.) *Identification of Continuous-time Models from Sampled Data* 362–391 (Springer, London, 2008).

[35] Barth, J. *et al.* Model-free control algorithms for micro air vehicles with transitioning flight capabilities. *International Journal of Micro Air Vehicles* **12**, 1–22 (2020).

[36] Polack, P., Delprat, S. & d'Andréa-Novel, B. Brake and velocity model-free control on an actual vehicle. *Control Engineering Practice* **92**, ID 104072 (2019).

[37] Fliess, M., Join, C., Moussa, K., Djouadi, S. & Alsager, M. *Toward simple in silico experiments for drugs administration in some cancer treatments* (Ghent, Belgium, 2021).

[38] Riachy, S., Fliess, M. & Join, C. *High-order sliding modes and intelligent PID controllers: first steps toward a practical comparison* (Milano, Italy, 2011).

[39] Wang, Z., Zhou, X., Cosio, A. & Wang, J. Ground vehicle lane-keeping assistance system via differential flatness output feedback control and algebraic derivative estimation. *Control Engineering Practice* **137**, ID 105576 (2023).

[40] Villagra, J. & Herrero-Perez, D. A comparison of control techniques for robust docking maneuvers of an AGV. *IEEE Trans. Control Syst. Techn.* **4**, 1116–1123 (2012).

[41] Scherer, P., Othmane, A. & Rudolph, J. Combining model-based and model-free approaches for the control of an electro-hydraulic system. *Control Engineering Practice* **133**, ID 105453 (2023).

[42] Mounier, H., Join, C., Delaleau, E. & Fliess, M. Active queue management for alleviating internet congestion via a nonlinear differential equation with a variable delay. *Annu. Rev. Control* **55**, 61–69 (2023).

[43] Join, C., d'Onofrio, A. & Fliess, M. Toward more realistic social distancing policies via advanced feedback control. *Automation* **3**, 286–301 (2022).

[44] Michel, L. *et al.* Model-free control of the dynamic lift of a wind turbine blade section: experimental results. *J. of Physics: Conference Series* **2265**, ID 032068 (2022).

[45] Menhour, L., d'Andréa-Novel, B., Fliess, M. & Mounier, H. *Multivariable decoupled longitudinal and lateral vehicle control: a model-free design* (Firenze, Italy, 2013).

[46] Moreno-Gonzalez, M., Artuñedo, A., Villagra, J., Join, C. & Fliess, M. Speed-adaptive model-free path-tracking control for autonomous vehicles: analysis and design. *Vehicles* **5**, 1698–717 (2023).

[47] Fliess, M. & Join, C. Two competing improvements of PID controllers: A comparison. arXiv (2018). 1801.10399v1.

[48] Fliess, M. & Join, C. *Model-free control and intelligent PID controllers: towards a possible trivialization of nonlinear control* (Saint-Malo, France, 2009).

[49] De Miras, J., Join, C., Fliess, M., Riachy, S. & Bonnet, S. *Active magnetic bearing: a new step for model-free control* (Firenze, Italy, 2013).

[50] Choi, S., d'Andréa-Novel, B., Fliess, M., Mounier, H. & Villagra, J. *Model-free control of automotive engine and brake for stop-and-go scenarios* (Budapest, Hungary, 2009).

[51] Ait Ziane, M. *et al.* A new concept of water management diagnosis for a PEM fuel cell system. *Energy Conversion and Management* **47**, ID 116986 (2023).

[52] Ait Ziane, M. *et al.* On-line implementation of model free controller for oxygen stoichiometry and pressure difference control of polymer electrolyte fuel cell. *Int. J. of Hydrogen Energy* **47**, 38311–38326 (2022).

[53] Basseville, M. & Nikiforov, I. *Detection of Abrupt Changes: Theory and Application* (Prentice Hall, Englewood Cliffs, New Jersey, 1994).