Model-free active fault tolerant control for sensor fault

Meziane Ait Ziane, Michel Zasadzinski, Cédric Join and Marie-Cécile Péra

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. The developed method is validated via numerical simulations for both stable and unstable linear systems. The performances of the developed active fault tolerant control procedure for unstable systems are evaluated with and without saturation of the control input.**

I. 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], [2], [3], [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 strategy is to ensure an acceptable level of system behavior in the presence of a fault [6].

Fault control strategies are 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. Theses informations are used by the controller to accommodate the fault in order to ensure a consistent behavior of system operation [10]. In general, the design of AFTC is based on the knowledge of the analytical model of the controlled system. There are few methods in the literature that introduce AFTC without an a priori knowledge of the system's analytical model.

Model-free control is one of the most efficient methods available to control linear and nonlinear systems without using the analytical model of the system [11], [12]. This model-free control can be considered as a PFTC that allows tolerating actuator fault and disturbances that affect the system [13], [14], [15] without fault estimation. However, this approach cannot tolerate sensor fault.

In this paper, we address an AFTC approach to accommodate the sensor fault in a model-free framework. 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. The ultra local model used in model-free control allows to generate a residual to detect and estimate the sensor fault. This estimate allows to adapt the model-free controller in order to achieve the sensor fault accommodation. The performances of the developed AFTC is checked for both stable and unstable systems. In addition, two cases are treated for unstable systems: with and without saturation of the control input.

This paper is structured as follows. The model-free control is recalled in Section II-A. The model-free sensor fault detection and estimation are developed in Section II-B. The adaptation of the model-free control law is given in Section II-C. The numerical examples to validate the AFTC strategy are presented in Section III-A for stable systems and in Section III-B for unstable systems.

II. Model-free active fault tolerant control

The closed loop system considered in this paper is given by

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

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, $y_d(t) \in \mathbb{R}$ is the desired trajectory and $f(t) \in \mathbb{R}$ is the sensor fault.

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

This work was not supported by any organization

M. Ait Ziane is with Group of Research in Electrical Engineering of Nancy (GREEN), Université de Lorraine, F-54000 Nancy, France. meziane.ait-ziane@univ-lorraine.fr

M. Zasadzinski and C. Join are with Research Center for Automatic Control of Nancy (CRAN), UMR CNRS 7039, Université de Lorraine, F-54000 Nancy, France michel.zasadzinski@univ-lorraine.fr, cedric.join@univ-lorraine.fr

M.C. Péra is with FEMTO-ST Instiute, FCLAB, Université Bourgogne Franche-Comte, CNRS, rue Thierry-Mieg, Belfort, 90010, Fraance marie-cecile.pera@univ-fcomte.fr

$$\forall \varepsilon \in \mathbb{R}^+, \exists T_1 \in \mathbb{R}^+ \text{ such that } |y_m(t) - y_d(t)| \leqslant \varepsilon \quad \forall t > T_1 \tag{2}$$

$$\forall \varepsilon \in \mathbb{R}^+, \exists T_2 \in \mathbb{R}^+, \exists c \in \mathbb{R} \text{ such that } |u(t) - cy(t)| \leqslant \varepsilon \quad \forall t > T_2 \tag{3}$$

In the property given by (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, while the regulated output y converges to y_d only if $f = 0$. In the property given by (3), the constant c means that there is an "almost linear" relationship between control u and output y for any reached steady-state. Note that the constant c always exists for linear systems.

A. Model-free control

In [11], [12], 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

$$y_m^{(\nu)}(t) = F(t) + \alpha u(t) \tag{4}$$

where ν refers to the derivative order of the measured output y_m, α is a non-pyhisical parameter defined 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

$$u(t) = \frac{1}{\alpha}\left(-\hat{F}(t) + \dot{y}_d(t) + k_p e(t)\right) \tag{5}$$

where

- $e = y_d - y_m$ is the tracking error,
- k_p is a tuning proportional gain,
- $\hat{F}$ is the estimation of F given by [16]

$$\hat{F}(t) = \frac{-3!}{T^3}\int_{t-T}^{t} \left((T - 2\tau)y_m(\tau) + \alpha\tau(T-\tau)u(\tau)\right) \mathrm{d}\tau \tag{6}$$

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

In [12], [17], 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 (2).

B. Model-free sensor fault detection and estimation

Model-free fault detection is introduced for fault detection in [14], for actuator fault in [15], for process and sensor faults in [18]. The main idea is to estimate the output of the controlled system from the ultra-local model given by (4) using the estimation of F in (6) and the control law in (5).

In this paper, the estimation of the magnitude of the sensor fault used in the AFTC procedure is provided in two steps: first, a residual is generated to detect the sensor fault in Section II-B.1 and, second, the magnitude of this fault is estimated by processing this residual in Section II-B.2.

1) Residual generation for sensor fault detection: The use of the ultra local model (4) leads to

$$y_m(t) = \int_0^t \left(F(\tau) + \alpha u(\tau)\right) \mathrm{d}\tau + y_m(0) \tag{7}$$

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

$$\hat{y}_m(t) = \int_0^t \left(\hat{F}(\tau) + \alpha u(\tau)\right) \mathrm{d}\tau + \hat{y}_m(0) \tag{8}$$

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

It is now a matter of correcting the estimated measured output $\hat{y}_m$ 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

$$r(t) = y_m(t) - \beta\hat{y}_m(t) \tag{9}$$

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

$$\beta(t) = \frac{y_m(t)}{\hat{y}_m(t)} = \frac{y_m(t)}{\int_0^t \left(\hat{F}(\tau) + \alpha u(\tau)\right) \mathrm{d}\tau + \hat{y}_m(0)} \tag{10}$$

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

Now we will prove that β is constant for systems having linear static characteristic as in (3). To do that, we need to use a numerical integration method for the integral in (10). 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

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

and

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

where T_e is the sampling time. □

Applying the approximation defined in Remark 1 to (8) leads to

$$\hat{y}_m(k) = \mathcal{I}\left(\hat{F}(k) + \alpha u(k)\right) + \hat{y}(0) \tag{11}$$

where the estimation of F in (4) is expressed by

$$\hat{F}(k) = \delta_{y_m}(k) - \alpha u(k-1) \tag{12}$$

instead of by relation (6). It should be noted that $u(k)$ cannot be used in (12) due to causality.

Using the definition of $\delta_{y_m}(k)$ given in Remark 1 and inserting (12) in (11) give

$$\hat{y}_m(k) = \mathcal{I}\left(\delta_{y_m}(k) - \alpha u(k-1) + \alpha u(k)\right) + \hat{y}_m(0) \tag{13}$$

where

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

$$+ \ldots + \frac{y_m(k) - y_m(k-1)}{T_e} \times T_e$$
$$= y_m(k) - y_m(0)$$

and

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

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

$$\hat{y}_m(k) = y_m(k) + \alpha T_e u(k) - y_m(0) + \hat{y}_m(0) \quad (14)$$

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

$$u(k) = cy(k) = cy_m(k) \quad (15)$$

where c is defined in (3). Substituting (15) in (14), the estimated measured output $\hat{y}_m$ becomes

$$\hat{y}_m(k) = y_m(k)\left(1 + T_e \alpha c(k)\right) - y_m(0) + \hat{y}_m(0) \quad (16)$$

Since the initial condition is assumed to be approximately known, i.e. $\hat{y}_m(0) \simeq y_m(0)$, inserting (16) in (10) yields

$$\beta(k) = \frac{y_m(k)}{y_m(k)\left(1 + T_e \alpha c\right)} = \frac{1}{1 + T_e \alpha c} = \beta \quad (17)$$

This proves that, for any change in desired trajectory, the parameter $\beta(k)$ converges to the same value β. So in the absence of sensor fault, the residual signal $r(t)$ in (9) is null in steady-state with β defined in (17).

2) Sensor fault estimation: Consider now that the measured output is affected by a sensor fault. Then the regulated output is expressed as follows

$$y = y_m - f, \qquad f \neq 0 \quad (18)$$

where f is an additive sensor fault. It is important to remember that β was determined before the fault occurred, i.e. when the measured output y_m was equal to the regulated output y. So (15) becomes

$$u(k) = cy(k) = c(y_m(k) - f(k)) \quad (19)$$

and (14) is expressed as

$$\hat{y}_m(k) = y_m(k) + \alpha T_e c(y_m(k) - \hat{f}(k)) \quad (20)$$

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

To determine the estimation $\hat{f}$, we proceed as follows. Inserting β given by (17) and $\hat{y}_m$ given by (20) in relation (9) where $\hat{f}$ is replaced by f, we obtain

$$\begin{aligned} r(k) &= y_m(k) - \frac{1}{1 + T_e \alpha c}\left(y_m(k)(1 + \alpha T_e c) - \alpha T_e c f(k)\right) \\ &= \frac{\alpha T_e c f(k)}{1 + \alpha T_e c} \\ &= f(k)\left(1 - \frac{1}{1 + \alpha T_e c}\right) = f(k)(1 - \beta) \quad (21) \end{aligned}$$

Using (21), the best estimation of the sensor fault f is given by

$$\hat{f}(k) = \frac{r(k)}{1 - \beta} \quad (22)$$

where $\beta \neq 1$ due to (17).

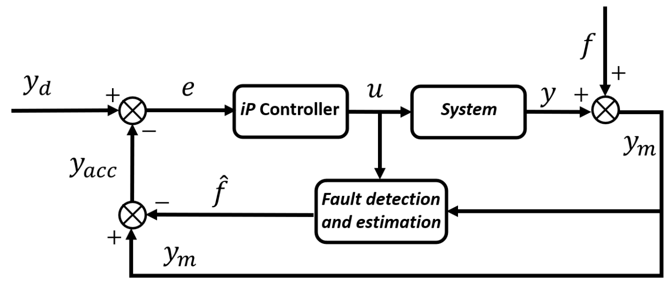


Fig. 1: Model-free active sensor fault tolerant control

C. Fault tolerant control strategy

The proposed AFTC strategy is based on the estimation of the sensor fault given by (22). Once the residual signal $r(t)$ in (9) exceeds a given threshold, the sensor fault is detected and the control law given by (5) is adapted to tolerate the fault according to the diagram shown in Fig. 1: the tracking error in (5) becomes

$$e(t) = y_d(t) - y_{acc}(t) \quad (23a)$$
$$y_{acc}(t) = y_m(t) - \hat{f}(t) \quad (23b)$$

III. Numerical simulations

The validation of the proposed model-free AFTC strategy is performed on two academic examples for stable and unstable linear systems where the measurements are impacted by a noise.

A. Stable system

Consider a stable linear system described by

$$\dot{x}(t) = Ax(t) + Bu(t) \quad (24a)$$
$$y(t) = Cx(t) + Du(t) \quad (24b)$$
$$y_m(t) = y(t) + f(t) + w(t) \quad (24c)$$

where

$$A = \begin{bmatrix} -6 & 1.5 & -0.5 \\ 1 & -7 & 0 \\ 0.5 & 0.2 & -3 \end{bmatrix}, B = \begin{bmatrix} 5 \\ -1 \\ 0 \end{bmatrix}, C = \begin{bmatrix} 0.5 & 1 & 0 \end{bmatrix}, D = 0$$

and $w(t)$ is a zero-mean white Gaussian noise.

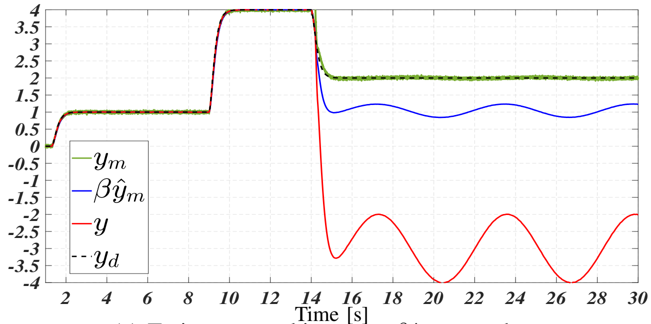
The sensor fault f is introduced as follows

$$f(t) = \begin{cases} 0, & t < 14.2\,\text{s} \\ \sin(t) + 5, & t \geqslant 14.2\,\text{s} \end{cases}$$

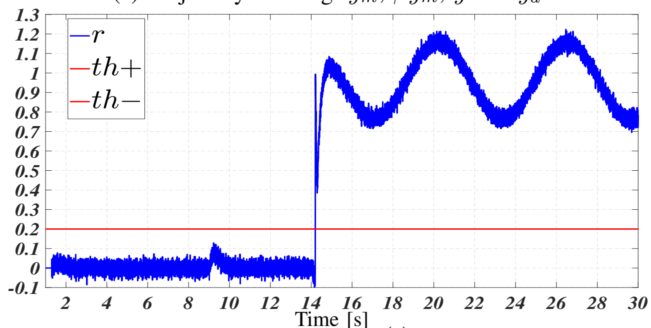
The parameters of the iP controller are $k_p = 14$ and $\alpha = 1.5$ with the sampling time $T_e = 0.001\,\text{s}$. After the first change of the desired trajectory y_d in Fig. 2(a), the signals y_m and $\hat{y}_m$ are used to determine the parameter β as in (10). The obtained value of β is 0.8076.

In fault free case, $y = y_m$ in Fig. 2(a), so the iP controller ensures the tracking of the desired trajectory.

At $t = 14.2\,\text{s}$, the occurrence of a fault impacts the measured output y_m. As the iP controller is robust to disturbances and corrects any difference between the set

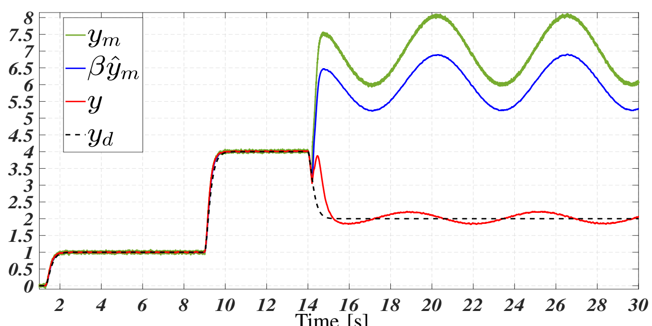


(a) Trajectory tracking: y_m, $\beta\hat{y}_m$, y and y_d

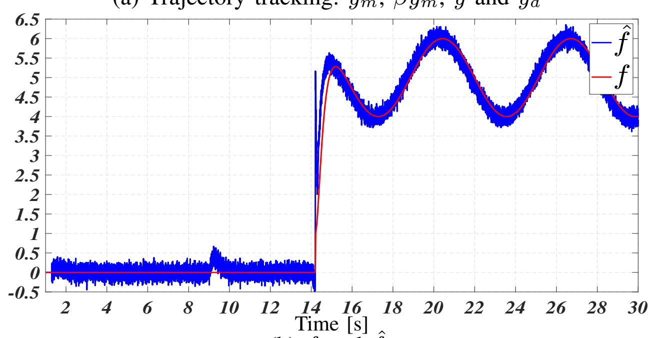


(b) Residual $r(t)$

Fig. 2: Sensor fault without AFTC (stable case)



(a) Trajectory tracking: y_m, $\beta\hat{y}_m$, y and y_d



(b) f and $\hat{f}$

Fig. 3: Sensor fault with AFTC (stable case)

point and the measured output, y_m is maintained at the desired trajectory y_d (see Fig. 2(a)).

However, the corrected estimated measured output $\beta\hat{y}_m$ differs from the measured output y_m after the time of the fault occurrence, which provides a non-zero residual $r(t)$, as shown in Fig. 2(b). The threshold for the residual $r(t)$ is defined empirically in Fig. 2(b) (there is no fault if $|r(t)| < th+ = -th- = 0.2$) . When the threshold is exceeded, the fault is considered to be present. The iP controller does not tolerate the sensor fault since the regulated output y is never controlled to the desired trajectory y_d at time $t > 14.2\,$s as shown in Fig. 2(a). The iP controller only ensures a correct control of the measured output y_m which is affected by the sensor fault. We can therefore conclude that the AFTC is mandatory to ensure an acceptable system behavior in the presence of a sensor fault.

The proposed AFTC is based on the estimation of the sensor fault using (22). Once the sensor fault is detected, the formula of the iP controller is adapted by using the tracking error defined in (23). Fig. 3(a) shows that the regulated output y is maintained at the desired trajectory y_d after the occurrence of the sensor fault. This confirms the ability of the proposed method to tolerate the sensor fault. However, the measured output y_m is never regulated to the desired trajectory y_d. As shown in Fig. 3(b), the fault f is well estimated despite the measurement noise: the AFTC strategy therefore compensates the effect of the sensor fault.

B. Unstable system

Consider an unstable linear system given by (24), where

$$A = \begin{bmatrix} -3 & 2.5 \\ 4 & 0 \end{bmatrix}, B = \begin{bmatrix} 2 \\ 0 \end{bmatrix}, C = \begin{bmatrix} 1 & 0.75 \end{bmatrix}, D = 0$$

and $w(t)$ is a zero-mean white Gaussian noise.

The sensor fault f is introduced as follows

$$f(t) = \begin{cases} 0, & t < 17\,\text{s} \\ \mu(t), & t \geqslant 17\,\text{s} \end{cases}$$

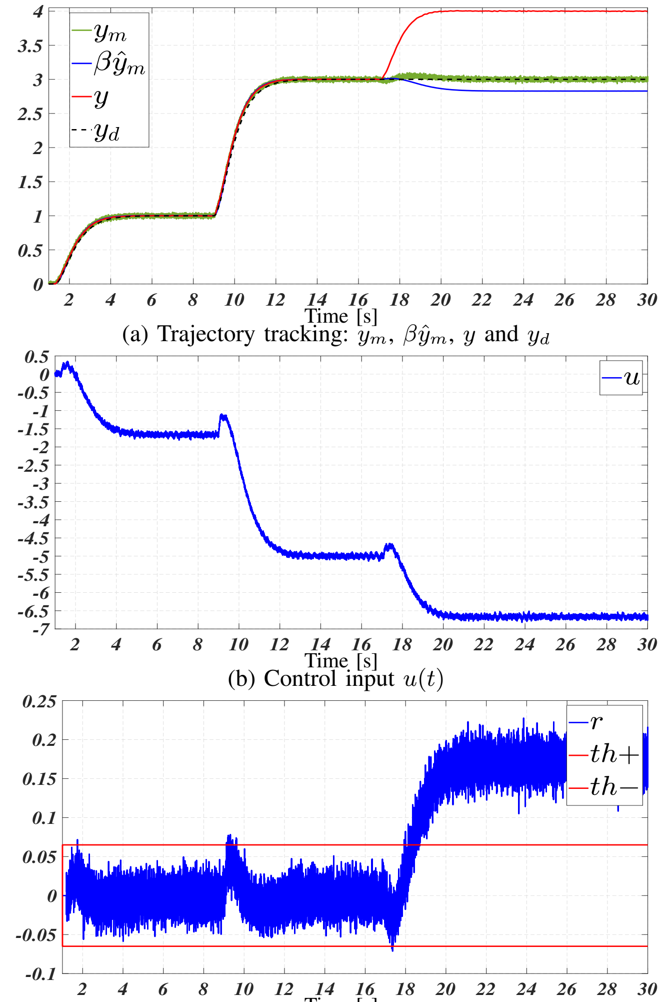
where $\mu(t)$ is the unitary step response of transfer function $\dfrac{-1}{(0.55s+1)^2}$ at time instant 17 s.

The parameters of the iP controller are $k_p = 1.5$ and $\alpha = 0.8$ with the sampling time $T_e = 0.001\,$s. After the first change of the desired trajectory y_d in Fig. 4(a) or Fig. 6(a), the signals y_m and $\hat{y}_m$ are used to determine the parameter β as in (10). The obtained values of β is 1.01723. For this unstable system two cases are considered:

- **(1)** The control input is not saturated, then the iP controller is stabilizing without AFTC procedure.
- **(2)** The control input is saturated, then the iP controller is not stabilizing without AFTC procedure.

1) Without saturation: By analyzing Fig. 4(a,c), we can make the same analysis as in Section III-A: first, the iP controller only ensures correct control of the measured output y_m which is affected by the sensor fault, second, the AFTC is mandatory to ensure an acceptable system behavior in the presence of this fault (there is no fault if $|r(t)| < th+ = -th- = 0.06$ in Fig. 4(c)).

Looking at Fig. 5(a,c), we can make the same analysis as in section III-A: the sensor fault is well estimated and the AFTC strategy compensates the effect of the sensor fault. Consequently, the proposed AFTC strategy works as well for stable as for unstable systems in the absence of input saturation.

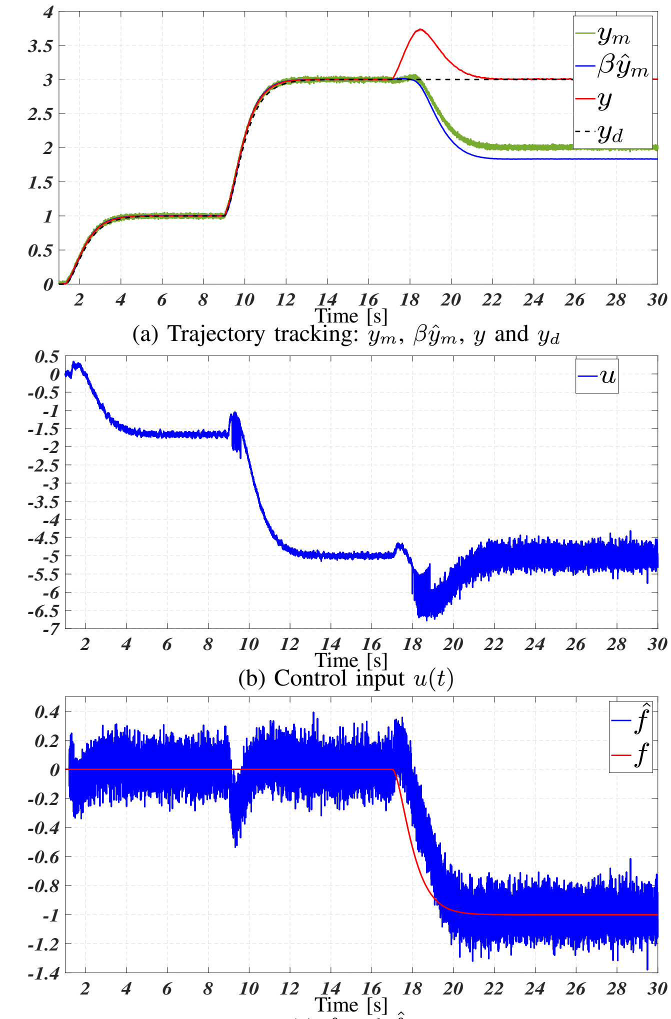


(a) Trajectory tracking: y_m, $\beta\hat{y}_m$, y and y_d

(b) Control input $u(t)$

(c) Residual $r(t)$

Fig. 4: Sensor fault without AFTC (unstable case without saturation)



(a) Trajectory tracking: y_m, $\beta\hat{y}_m$, y and y_d

(b) Control input $u(t)$

(c) f and $\hat{f}$

Fig. 5: Sensor fault with AFTC (unstable case without saturation)

By comparing Fig. 4(b) and Fig. 5(b), we can see how the AFTC procedure acts on the control input u.

2) With saturation: The control input $u(t)$ is saturated as follows: if $u(t) \leqslant -6.2$ than $u(t) = -6.2$, else $u(t)$ is not saturated. This saturation of the control input has a big effect in the closed loop behavior since the control input $u(t)$ converges to values inferior to -6.2 in the non saturated case as can be seen in Fig. 4(b). That the reason why the closed loop is unstable after the control input saturation as shown in Fig. 6(a,b) with the same thresholds for $r(t)$ as in Fig. 4(c)).

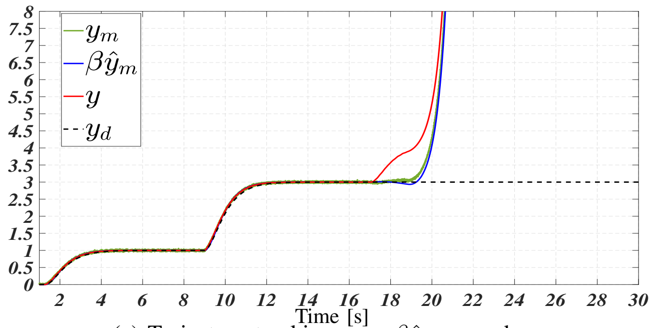
Fig. 7(b) shows that the AFTC acts as soon as the sensor fault is detected, but this generates saturation of the control input $u(t)$ (i.e. $u(t)$ falls below the saturation threshold). However, the proposed AFTC strategy is able to overcome the saturation of the input as can be seen in Fig. 7(b) at time instant $t > 20\,\mathrm{s}$: in Fig. 5(b) and Fig. 7(b) the control input $u(t)$ is approximately the same.

Comparing Fig. 5(a,b,c) and Fig. 7(a,b,c) after $t > 20\,\mathrm{s}$, the AFTC procedure provides the same closed loop behavior. This means that, despite saturation of the control input $u(t)$ applied to an unstable system, the proposed AFTC strategy is able to

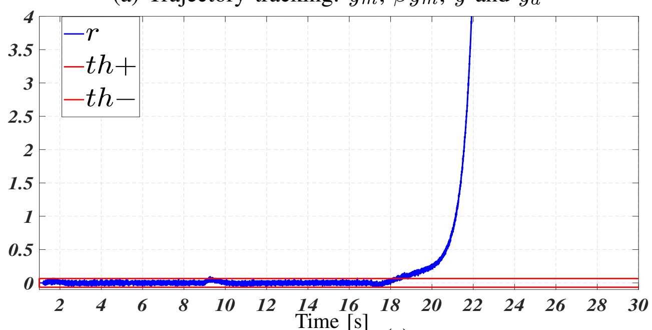
- ensure the closed loop stability,
- satisfy the tracking objective of the regulated output y,
- compensate the effect of a sensor fault.

IV. Conclusion

A model-free active sensor fault tolerant control approach is presented and evaluated in this paper. The proposed method ensures the accommodation of the sensor fault without knowledge of the analytical system model. The proposed AFTC procedure 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 AFTC technique to accommodate the sensor fault for both stable and unstable systems. To analyze the performance of the developed AFTC procedure applied to unstable systems, two cases are considered: with and without control input saturation. In



(a) Trajectory tracking: y_m, $\beta\hat{y}_m$, y and y_d



(b) Residual $r(t)$

Fig. 6: Sensor fault without AFTC (unstable case with saturation)

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

REFERENCES

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

[2] J. Lunze and J. Richter, "Reconfigurable fault-tolerant control: a tutorial introduction," *European J. Contr.*, vol. 14, pp. 359–386, 2008.

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

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

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

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

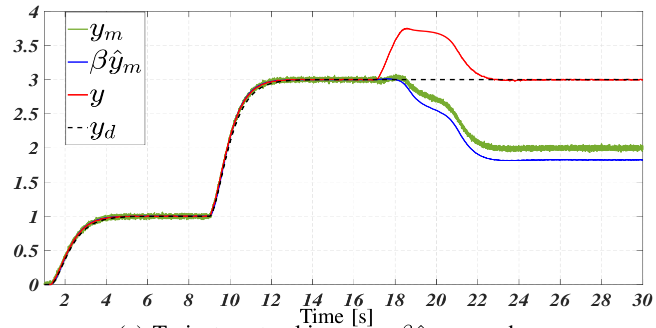
[7] R. Wang and J. Wang, "Passive actuator fault-tolerant control for a class of overactuated nonlinear systems and applications to electric vehicles," *IEEE Trans. Vehicular Technology*, vol. 62, pp. 972–985, 2012.

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

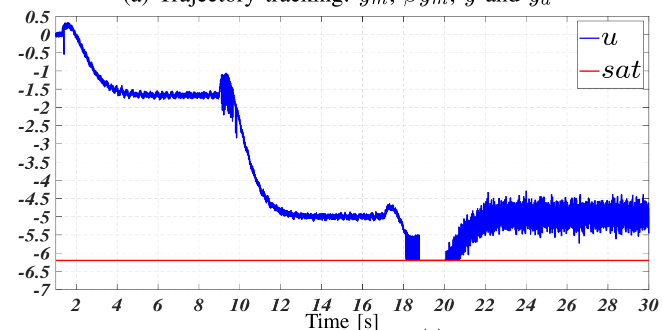
[9] J. Jiang and X. Yu, "Fault-tolerant control systems: a comparative study between active and passive approaches," *Annual Reviews in Control*, vol. 36, pp. 60–72, 2012.

[10] A. Abbaspour, S. Mokhtari, A. Sargolzaei, and K. Yen, "A survey on active fault-tolerant control systems," *Electronics*, vol. 9, p. ID 1513, 2020.

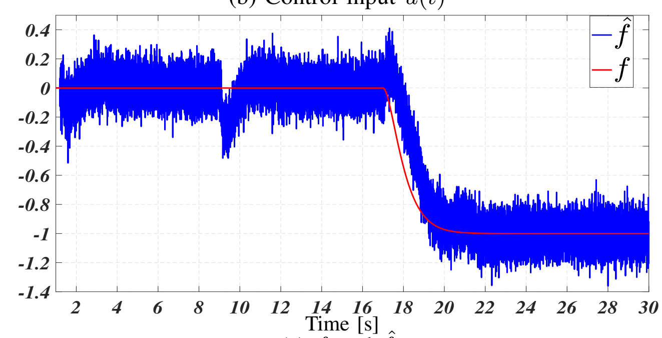
[11] M. Fliess and C. Join, "Model-free control," *Int. J. Control*, vol. 86, pp. 2228–2252, 2013.



(a) Trajectory tracking: y_m, $\beta\hat{y}_m$, y and y_d



(b) Control input $u(t)$



(c) f and $\hat{f}$

Fig. 7: Sensor fault with AFTC (unstable case with saturation)

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

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

[14] M. Ait Ziane, C. Join, M. Péra, N. Yousfi Steiner, M. Benne, and C. Damour, "A new method for fault detection in a free model context," in *Proc. IFAC SAFEPROCESS Symp.*, (Pafos, Cyprus), 2022.

[15] M. Ait Ziane, N. Yousfi Steiner, C. Join, M. Benne, C. Damour, and M. Péra, "Model-free fault detection: application to polymer electrolyte fuel cell system," in *Proc. IEEE Int. Conf. on Systems and Control*, (Marseille, France), 2022.

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

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

[18] M. Ait Ziane, C. Join, M. Benne, C. Damour, N. Yousfi Steiner, and M. Péra, "A new concept of water management diagnosis for a PEM fuel cell system," *Energy Conversion and Management*, vol. 47, p. ID 116986, 2023.