Available online at www.sciencedirect.com

ScienceDirect

IFAC PapersOnLine 51-24 (2018) 680–687

Actuator Fault Tolerant Control Proposal for PI Controlled SISO Systems ⋆

Ignacio Peñarrocha-Alós * **Ester Sales-Setién** * **David Tena** *

* *Departament d'Enginyeria de Sistemes Industrials i Disseny. Universitat Jaume I de Castelló, Spain (e-mail: {ipenarro,esales,david.tenatena}@uji.es)*

Abstract: In this work, we develop a control structure which can be added to an existing closed loop in order to mitigate the effect of actuator faults. We analyze the initial performance of the closed loop in terms of robustness and time response under references, faults and measurement noises. In the design of the proposed active fault tolerant control structure, we keep the initial robustness and time response to references. At the same time, we try to improve some performance indices under faults at the cost of a higher control action activity caused by the measurement noises that affect the system. The design of the controller depends on a unique parameter so it can be easily understood. We show that although the response under step faults becomes oscillatory, the active fault tolerant structure reduces the integral of the absolute value of the tracking error under step faults and it attenuates the effect of ramp faults in steady state. Several examples show the goodness and drawbacks of the approach and show some aspects to be considered in the design.

Keywords: Active fault tolerant control, process control industry.

1. INTRODUCTION

The PI controller is unquestionably the most commonly used control algorithm in the process control industry (Desborough and Miller (2002)). The main reason is its relatively simple structure, which can be easily understood and implemented in practice (Wang and Shao (2000)). These controllers are able to track step references and mitigate constant disturbances (Åström and Hägglund (2006)) and, thus, if actuator faults are considered as input disturbances, PI control can be viewed as a fixed control structure that has some fault tolerant properties. More concisely, according to the classification of fault tolerant strategies included in Blanke et al. (2006) and Patton (2015), a PI controller can be viewed as a passive fault tolerant controller.

One known strategy to mitigate faults when considered as exogenous measured signals affecting the process output is to perform a feed-forward strategy, which is an active fault tolerant control strategy known as fault accommodation (Blanke et al. (2006)). If the fault is not measured, an option is to observe it in order to feedforward a fault estimate. This is the base of fault accommodation based on fault estimation (Zhang et al. (2012); Shen et al. (2017); Simani and Castaldi (2014); Shi and Patton (2015); Rotondo et al. (2014)). In the case of industrial control systems, control structures are generally implemented in basic function blocks that can only lead with PID controllers or simple lead-lag transfer functions. Provided that common fault estimators entail more complex structures in state-space representation (Gao et al. (2015)), we propose the direct computation of a control action to be feedforwarded in order to mitigate faults without an intermediate estimation of the faults.

In this work, we study if there exists a simple feedforward structure that can be added to an existing control system to improve the response under faults without affecting the original behavior under reference changes. In fault estimation there is a trade-off between the time response of the estimator and the noise amplification. When used in fault diagnosis, this trade-off is translated into a trade-off between the time needed to diagnose faults, the minimum diagnosable faults and false alarm rates (Zhang and Ding (2008); Peñarrocha et al. (2015); Sales-Setién et al. (2016)). When performing direct fault accommodation without intermediate estimation of the faults, this trade-off is translated into a new trade-off between noise amplification and fault attenuation.

The main objective of this work is to propose a simple active FTC structure to be added to a nominal closed loop which is controlled by a PI controller. The new fault tolerant controller only depends on one parameter and for the definition of its structure and design, we assume that the model of the system under control is known. The design does not depend on the initial PI controller acting on the system. We quantify the trade-offs between fault mitigation and actuator effort requirements caused by noise amplification and we present a performance-based design of the fault tolerant controller.

The remainder of this paper is as follows. First, we state the problem in Section 2. Section 3 deals with the proposal of a simple active FTC scheme and Section 4 presents a

⋆ This work has been supported by grant FPU14/01592 from MECD and by projects P11B2015-42 from Universitat Jaume I de Castelló and MINECO project number TEC2015-69155-R.

10.1016/j.ifacol.2018.09.649

proposal for the structure and design of the new controller in the FTC scheme. Section 5 presents some simulation results for theoretical cases of study and, finally, some concluding remarks are reported in Section 6.

2. PROBLEM STATEMENT

In this work, we consider a LTI SISO system in the form of

$$y(s) = G(s)(u(s) + f(s)), \quad (1)$$

where $G(s)$ is the transfer function that defines the behavior of the system and $y(s)$, $u(s)$ and $f(s)$ are the Laplace transform of the output, input and additive actuator fault signal, which can be considered as an input disturbance. The output is measured by a sensor which is affected by an additive noise and this measurement is given by

$$y_m(s) = y(s) + n(s), \quad (2)$$

with $n(s)$ being the noise signal. The control action is initially computed by a feedback controller in the form of

$$u_c(s) = C(s)e_m(s),\ e_m(s) = r(s) - y_m(s) \quad (3)$$

where $r(s)$ is the reference signal, $u_c(s)$ is the control action computed by the controller, $C(s)$ is the controller transfer function and $e_m(s)$ is the measured tracking error. The input of the system is, thus, initially given by $u(s) = u_c(s)$, and the output tracking error $e(s) = r(s) - y(s)$ and control action behavior in closed loop become

$$u(s) = \frac{C(s)r(s) - G(s)C(s)f(s) - C(s)n(s)}{1 + G(s)C(s)}, \quad (4)$$

$$e(s) = \frac{r(s) - G(s)f(s) + G(s)C(s)n(s)}{1 + G(s)C(s)}. \quad (5)$$

We assume that the controller has been designed and implemented to fulfill the some requirements over robustness, reference tracking behavior, disturbance rejection and noise attenuation. In order to characterize these initial performance indices, we assume that the controller $C(s)$ is a PI satisfying

$$C(s) = K_p + \frac{K_i}{s}, \quad (6)$$

with K_p and K_i being the proportional and integral gains. We quantify the performance indices as follows:

A1. Robustness. In order to characterize the robustness, we use the maximum value of the sensitivity function given by

$$M_s = \max_{\omega} \left| \frac{1}{1 + G(j\omega)C(j\omega)} \right|.$$

A2. Steady-state error under step references. It can be obtained from the static gain between the reference r and the tracking error signal e, which is given by

$$\lim_{s\to 0} \frac{1}{1 + G(s)C(s)} = 0.$$

This limit is zero due to the integral term of the PI controller.

A3. Integral of the error under step references or steady-state error under ramp references. Both are given by the limit

$$\lim_{s\to 0} \frac{1}{1 + G(s)C(s)} \frac{1}{s} = \frac{1}{K_i G(0)}.$$

A4. Steady-state error under step faults. It can be obtained from the static gain between the fault and the tracking error, which is given by

$$\lim_{s\to 0} \frac{-G(s)}{1 + G(s)C(s)} = 0.$$

Again, as the controller has an integral term, this value is zero.

A5. Steady-state error under ramp faults or integral of the error under step faults in absolute value. Both are given by the limit

$$\left| \lim_{s\to 0} \frac{-G(s)}{1 + G(s)C(s)} \frac{1}{s} \right| = \frac{1}{K_i}.$$

A6. Integral of the error under ramp faults. This can be obtained from the limit

$$\lim_{s\to 0} \frac{C(s)}{1 + G(s)C(s)} \frac{1}{s^2} = \infty.$$

As the steady-state error under ramp faults is not zero, this integral blows to infinity.

A7. Absolute value of the noise amplification from the sensor to the actuator. It can be characterized by the direct gain from the noise to the control action, i.e,

$$\left| \lim_{s\to\infty} \frac{-C(s)}{1 + G(s)C(s)} \right| = C(\infty) = K_p,$$

where we have assumed that $G(\infty) = 0$. In the case of a PI controller, this leads to an instantaneous change in the control action due to unitary abrupt changes in the noise signal of $C(\infty) = K_p$.

The goal of this work is to develop a new structure to be added to an standard closed loop whose controller $C(s)$ has been already designed and implemented.

3. FAULT TOLERANT CONTROL SCHEME

3.1 FTC Scheme

We propose to compute the control action with the addition of two terms

$$u(s) = u_c(s) + u_f(s), \quad (7)$$

where $u_c(s)$ comes from the initial controller $C(s)$ and $u_f(s)$ is the one in charge of the fault tolerant objectives and comes from a fault tolerant active structure. We initially propose to estimate the fault by means of

$$\hat{f}(s) = H_u(s)u(s) + H_y(s)y_m(s), \quad (8)$$

and then compute a feedforward action with

$$u_f(s) = C_f(s)\hat{f}(s). \quad (9)$$

In order to achieve an easier implementation, we propose to directly compute the feedforward action with the following strategy

$$u_f(s) = C_u(s)u(s) + C_y(s)y_m(s), \quad (10)$$

The transfer functions $C_u(s)$ and $C_y(s)$ would come from $C_u(s) = C_f(s)H_u(s)$ and $C_y(s) = C_f(s)H_y(s)$. In all, the proposed fault tolerant control structure is depicted in Fig.1. The tracking error and control action behavior in this new closed loop are given by

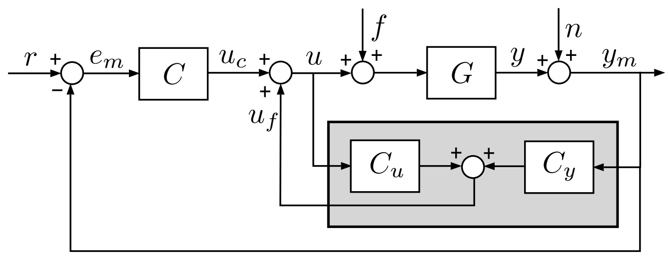

Fig. 1. Proposed actuator fault tolerant control structure.

$$u(s) = \frac{C(s)r(s) - G(s)(C(s) - C_y(s))f(s)}{1 + G(s)C(s) - C_u(s) - G(s)C_y(s)} - \frac{(C(s) - C_y(s))n(s)}{1 + G(s)C(s) - C_u(s) - G(s)C_y(s)}, \quad (11)$$

$$e(s) = \frac{(1 - C_u(s) - C_y(s)G(s))r(s) - G(s)(1 - C_u(s))f(s)}{1 + G(s)C(s) - C_u(s) - G(s)C_y(s)} + \frac{G(s)(C(s) - C_y(s))n(s)}{1 + G(s)C(s) - C_u(s) - G(s)C_y(s)}. \quad (12)$$

The objective of this work is to deal with the direct computation of $C_u(s)$ and $C_y(s)$ and to decide these transfer functions so as to:

- keep the robustness of the initial closed loop,
- keep the reference tracking performance of the initial closed loop,
- improve the behavior under actuator faults,
- guarantee certain limit for the noise amplification from the sensor to the actuator,
- obtain realizable and implementable transfer functions.

3.2 FTC Constraints

In order to fulfill the aforementioned requirements we have the following constraints:

B1. To keep the robustness, we must choose the transfer functions such that the sensitivity margin of

$$M_s' = \max_\omega \left| \frac{1}{1+G(j\omega)C(j\omega)-C_u(j\omega)+G(j\omega)C_y(j\omega)} \right|$$

remains the same as the one in A1, i.e., $M_s' = M_s$.

B2. To keep the steady state error under step references, we must guarantee

$$\lim_{s\to 0} \frac{(1 - C_u(s) - C_y(s)G(s))}{1 + G(s)C(s) - C_u(s) + G(s)C_y(s)} = 0,$$

B3. To keep the steady state error under ramp references or the integral error under step references, we must keep

$$\lim_{s\to 0} \frac{1 - C_u(s) - C_y(s)G(s)}{1 + G(s)C(s) - C_u(s) + G(s)C_y(s)} \frac{1}{s} = \frac{1}{K_i G(0)},$$

B4. In order to keep the steady state error under step faults we must keep the limit

$$\lim_{s\to 0} \frac{-G(s)(1 - C_u(s))}{1 + G(s)C(s) - C_u(s) + G(s)C_y(s)} = 0,$$

B5. In order to improve the steady state error under ramp faults or the integral of the error under step faults in absolute value, we look for transfer functions that lead to

$$\left| \lim_{s\to 0} \frac{-G(s)(1 - C_u(s))}{1 + G(s)C(s) - C_u(s) + G(s)C_y(s)} \frac{1}{s} \right| \le \frac{1}{K_i}.$$

B6. To improve the integral of the error under ramp faults we must obtain a finite value for the following limit

$$\left| \lim_{s\to 0} \frac{-G(s)(1 - C_u(s))}{1 + G(s)C(s) - C_u(s) + G(s)C_y(s)} \frac{1}{s^2} \right| < \infty.$$

B7. In order to attenuate the effect of the noise measurement, we must look for an appropriate transfer function in

$$\left| \lim_{s\to \infty} \frac{-C(s) + C_y(s)}{1 + G(s)C(s) - C_u(s) + G(s)C_y(s)} \right| < \gamma_1,$$

such that it attenuates the effect of the high frequency noises.

B8. To achieve a realizable and implementable solution, the transfer functions $C_u(s)$ and $C_y(s)$ must be causal and stable.

4. FAULT TOLERANT CONTROLLER STRUCTURE AND DESIGN

4.1 FT Controller Structure Discussion

Constraints B1 to B3 can be fulfilled if we choose $C_u(s)$ and $C_y(s)$ such that

$$C_u(s) + G(s)C_y(s) = 0. \quad (13)$$

In that case, the characteristic equation will remain the same as in the initial control loop, i.e.,

$$1 + G(s)C(s) = 1 + G(s)C(s) - C_u(s) - G(s)C_y(s),$$

and so will the transfer functions from the reference to the control action and the tracking error. Therefore, we fix this constraint in the following.

If we choose $C_u(s) = 1$, we reach zero values in the performance indices in B4 to B6; thus, we fulfill the desire of improving the behavior under faults. However, this value together with constraint (13), leads to a transfer function $C_y(s) = -G(s)^{-1}$ which is generally non-realizable and unstable (i.e., B8 is not satisfied). Furthermore, it leads to an algebraic loop in the definition of u and u_f in (7) and (10) and, thus, this structure becomes non-implementable. In all, we deduce that $C_u(s) = 1$ is not a feasible solution.

On the other hand, if we choose $C_y(s) = C(s)$ we mitigate the noise effect in B7. This choice also has several drawbacks. One is that the transfer function $C(s)$ is unstable because it has an integrator and this makes the operation $C_y(s)y(s)$ to wind up in a general scenario because B8 is not satisfied. On the other hand, this value together with constraint (13), leads to a transfer function $C_u(s)$ in the form of $C_u(s) = -G(s)C(s)$. With this transfer function, the tracking error under faults becomes the same as in an open-loop scenario, i.e., $e(s) = -G(s)f(s)$, and the transfer function from the fault to the control action is then zero; thus, the tracking ability under any kind of fault is lost. Therefore, $C_y(s) = C(s)$ is not a feasible solution either.

Note that, if we choose $C_u(s)$ such that it has unitary gain (i.e., $C_u(0) = 1$), we achieve zero values in indices B4 and

B5 and finite values in B6. Thus, we can choose $C_u(s)$ to be described by the simple filter function $C_u(s) = \frac{1}{1+\tau s}$, where, intuitively, τ needs to be small so as to enhance a fast actuator fault mitigation. This election together with constraint (13) leads to $C_y(s) = \frac{-G(s)^{-1}}{1+\tau s}$. This transfer function $C_y(s)$ is not always realizable because the function $G(s)^{-1}$ may have more zeros than poles. We can be face this problem by increasing the order of the filter in $C_u(s)$. Moreover, this expression of $C_y(s)$ is non-implementable whenever $G(s)$ has non-minimum phase zeros or delays because the inverse of $G(s)$ will have unstable poles or will require the knowledge of the future (i.e., a term e^{Ts}). In order to avoid the inversion of all the terms in $G(s)$, we can include in $C_y(s)$ only the inverse of the terms in $G(s)$ whose inverse is stable and add the other terms in $C_u(s)$. Provided this motivation, we include below the details of the chosen transfer functions.

4.2 FT Controller Structure Proposal

From the ideas presented in the previous discussion, we now propose the fault tolerant controller structure. We have seen that we must take account on the invertible and non-invertible (in terms of stability of the inverse); thus, we first decompose the system transfer function as follows

$$G(s) = \frac{K\prod_{i=1}^{m}(1+\beta_i s)\prod_{i=1}^{n}(1-\delta_i s)e^{-Ts}}{s^k\prod_{i=1}^{p}(1+\tau_i s)},$$

where $k = 1$ if the system has an integrator and $k = 0$ if not. K denotes the static gain (without the integrator, if it is the case), β_i stands for the half-left zeros in the complex plane ($\Re\{\beta_i\} > 0$) while δ_i stands for the half-right ones ($\Re\{\delta_i\} > 0$). e^{-Ts} takes account on the delay of the system, and, finally, $\tau_i s$ are the stable and unstable poles in $G(s)$. Note that β_i, δ_i and τ_i can be complex numbers characterizing the existence of oscillatory modes. Let us now rewrite the transfer function as

$$G(s) = K\, G_I(s)G_N(s),$$

with

$$G_I(s) = \frac{\prod_{i=1}^{m}(1+\beta_i s)}{s^k\prod_{i=1}^{p}(1+\tau_i s)},\ G_N(s) = \prod_{i=1}^{n}(1-\delta_i s)e^{-Ts},$$

in order to easily consider the parts whose inverse leads to an stable system ($G_I(s)$) and the terms which not ($G_N(s)$). Note that $G_N(0) = 1$ and $\lim_{s\to 0} G_I(s) = \lim_{s\to 0}\frac{1}{s^k}$, i.e., $G_I(0) = 1$ if the system does note have an integrator.

In order to prevent $C_y(s)$ from having non-implementable terms, we include in $C_u(s)$ the terms of $G(s)$ which would make $C_y(s)$ unstable and we include the other terms in $C_y(s)$. Taking this into account and provided that, as stated in B8, the FT controller must be realizable, we propose to use the following transfer functions:

$$C_u(s) = \frac{G_N(s)}{(1+\tau\, s)^{p+k-m}}, \tag{14}$$

$$C_y(s) = -\frac{G_I^{-1}(s)}{K(1+\tau\, s)^{p+k-m}}, \tag{15}$$

where τ is a design parameter and where $p+k-m$ is the relative degree of $G_I(s)$. The term $(1+\tau\, s)^{p+k-m}$ in the denominator makes $C_y(s)$ biproper. Assuming that the transfer function $G(s)$ is strictly proper (as usually in real systems), we have that $p+k-m-n > 0$. Then, the relative degree of $C_u(s)$ is $p+k-m-n$ and $C_u(s)$ is strictly proper. Furthermore, note that the constraint (13) is fulfilled as

$$\frac{G_N(s)}{(1+\tau\, s)^{p+k-m}} - K\, G_I(s)G_N(s)\frac{G_I^{-1}(s)}{K(1+\tau\, s)^{p+k-m}} = 0.$$

Remark 1. Note that the proposed FT controller does not depend on the original controller $C(s)$, which is an important advantage of the proposed fault tolerant structure.

4.3 Performance Analysis of the FTC Scheme

With the proposal (14)-(15), we have the closed-loop properties detailed below, where we have used a, b, c and d defined as follows:

$$a = \prod_{i=1}^{p}\tau_i,\ b = \prod_{i=1}^{m}\beta_i,\ c = \sum_{i=1}^{n}\delta_i,\ d = p+k-m.$$

C1. The sensitivity transfer function remains the same and so does the sensitivity margin in A1.

C2. The transfer function from reference to error remains the same and therefore, the steady state-error under step references remains the same as in A2.

C3. The steady-state error under ramp references or the integral error under step references is also the same as in A3.

C4. The steady-state error under step faults remains null.

C5. The steady-state error under ramp faults or the integral of the error under step faults in absolute value now goes to zero, i.e.,

$$\left|\lim_{s\to 0}\frac{-G(s)\left(1-\frac{G_N(s)}{(1+\tau\, s)^d}\right)}{1+G(s)C(s)}\frac{1}{s}\right| = 0.$$

As we have fixed the static gain of the transfer function $G_N(s)$ to be unitary (i.e., $G_N(0) = 1$), the function $1-\frac{G_N(s)}{(1+\tau\, s)^d}$ has a derivative term s and thus, the previous limit is zero.

C6. The integral error under ramp faults now becomes finite and is given by

$$\left|\lim_{s\to 0}\frac{-G(s)\left(1-\frac{G_N(s)}{(1+\tau\, s)^d}\right)}{1+G(s)C(s)}\frac{1}{s^2}\right| = \frac{d\,\tau+T+c}{K_i}.$$

C7. The amplification for high frequency noises in the control action now becomes

$$\left|\lim_{s\to\infty}\frac{-C(s)-\frac{G_I^{-1}(s)}{(1+\tau\, s)^d}}{1+G(s)C(s)}\right| = K_p + \frac{a}{b\,\tau^d}.$$

C8. The proposed FTC scheme is implementable because the transfer functions $C_u(s)$ and $C_y(s)$ are stable and causal.

Remark 2. The fact that the integral of the error becomes zero under step faults as shown in C5 means that the output will oscillate around the reference whenever step faults occur. This is one of the main differences w.r.t. the response under step disturbances in a PI control. In the case of a control loop with a PI controller, the error under ramp faults is finite and, if the sensitivity margin is small enough (i.e., below 1.6), the response under step faults is not oscillatory and the tracking error has mainly the same sign during all the transient response. The fact that our proposal can lead to an oscillatory response under step faults can be a drawback in some applications.

In the performance analysis detailed in C1-C8, one can deduce that the tuning parameter τ plays a role in both the time response of the closed loop under faults and in the noise amplification. When τ is set a low value, we get a fast response under ramp faults, but the measurement noise is amplified. In the following, we present a performance-based design of the FT controller which take account on this trade-off.

4.4 FT Controller Design

The existing trade-off between the time response and the noise amplification leads us to one of the following strategies:

D1. Minimize the time response under faults and guarantee certain noise amplification γ_1, i.e.,

$$\min_{\tau} \quad \tau \qquad s.t. \quad K_p + \frac{a}{b\,\tau^d} < \gamma_1 \tag{16}$$

The solution of problem (16) is

$$\tau = \left(\frac{a}{b\,(\gamma_1 - K_p)}\right)^{\frac{1}{d}} \tag{17}$$

If we express the constraint γ_1 in relative terms of the initial noise amplification K_p, i.e., $\gamma_1 = K_p(1+\alpha)$ with $\alpha > 0$, the solution of (16) becomes

$$\tau = \left(\frac{a}{b\,K_p\alpha}\right)^{\frac{1}{d}} \tag{18}$$

for certain allowed relative amplification α. The integral of the tracking error under ramp faults becomes

$$\frac{n\left(\frac{a}{b\,K_p\alpha}\right)^{\frac{1}{d}} + T + c}{K_i}.$$

D2. Minimize the noise amplification and guarantee certain response under faults characterized by γ_2:

$$\max_{\tau} \quad \tau \qquad s.t. \quad \frac{d\,\tau + T + c}{K_i} < \gamma_2 \tag{19}$$

The solution of (19) is

$$\tau = \frac{1}{d}(\gamma_2\,K_i - T - c), \tag{20}$$

and the achieved noise amplification is given by

$$K_p + \frac{a}{b\left(\frac{1}{d}(\gamma_2 K_i - T - c)\right)^d}.$$

4.5 Remarks on the Extension to Sensor FTC

The previous approach considered that the system is prone to actuator faults but no sensor faults may affect it. If we consider the existence of a sensor fault, signal n in (2) will be composed by both a measurement noise and a slow time-varying signal representing the sensor fault, i.e.,

$$n = v + f_s,$$

where v denotes the noise and f_s the sensor fault. The tracking error due to sensor faults in the initial control loop is then given by

$$\frac{e(s)}{f_s(s)} = -\frac{G(s)C(s)}{1+G(s)C(s)},$$

while in the proposed fault tolerant control scheme is

$$\frac{e(s)}{f_s(s)} = -\frac{G(s)(C(s)-C_y(s))}{1+G(s)C(s)-C_u(s)-G(s)C_y(s)}.$$

The fault tolerant control design would now also require the constraints in B1 to B3 (i.e., $C_u(s) + G(s)C_y(s) = 0$), B7 and B8. However, the constraints in B4 to B7 must be reformulated so that they demand a better performance w.r.t. sensor faults. The steady-state output tracking error under constant sensor faults in the initial configuration is given by

$$\lim_{s\to 0} \frac{-G(s)C(s)}{1+G(s)C(s)} = -1,$$

what means that a sensor fault in the form of a bias will be directly translated into a tracking error. On the other hand, with the proposed approach and provided the fulfilment of $C_u(s) + G(s)C_y(s) = 0$, we have that this error is

$$\lim_{s\to 0} \frac{-G(s)(C(s)-C_y(s)}{1+G(s)C(s)}.$$

In order to mitigate constant sensor faults (i.e., sensor biases) one must choose $C_y(s)$ such that it has the same steady-state gain as $C(s)$, i.e., one must include an integrator in $C_y(s)$; but, as explained before, its implementation is unstable. This induces us to the use of an alternative control structure different form the one in (7)-(10). The main difference would consist on adding some term to the initial measurement y_m instead of adding a new term to the initial control action u_c as we propose in the present work. This is the starting point to develop an alternative approach for sensor fault tolerant control. A combination of the presented alternative and the one to be developed will lead to a fault tolerant control scheme for both sensor and actuator faults.

5. CASES OF STUDY

In this section, we present different theoretical cases of study which show the goodness and drawbacks of the proposed approach. First, we compare the behaviour for different systems when the relative noise amplification is fixed. Second, we compare the behaviour in a fixed system depending on the allowed relative noise amplification.

5.1 Case of Study 1

In this case of study, we compare the behavior of three different systems with PI controllers which have been designed to have a phase margin of 60^o and a maximum value of K_i. These systems and controllers are

$$G_1(s) = \frac{e^{-0.1s}}{1+0.9s}, \qquad C_1(s) = 1.37 + \frac{3}{s},$$

$$G_2(s) = \frac{e^{-0.5s}}{1+0.5s}, \qquad C_2(s) = 0.833 + \frac{1.2}{s},$$

$$G_3(s) = \frac{e^{-0.9s}}{1+0.1s}, \qquad C_3(s) = 0.275 + \frac{0.733}{s}.$$

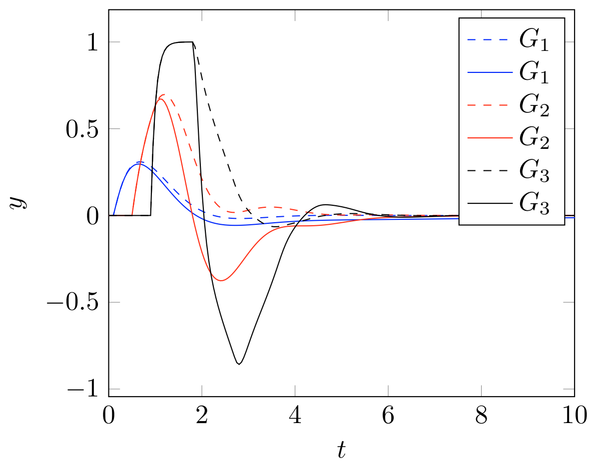

Fig. 2. Output response under unitary step faults. Dashed: PI control; Solid: FTC.

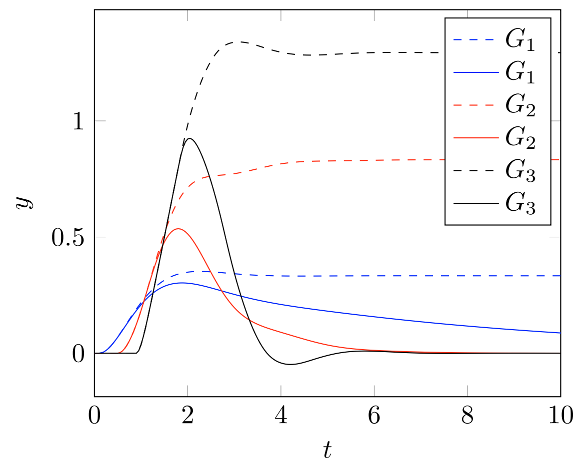

Fig. 3. Output response under unitary ramp faults. Dashed: PI control; Solid: FTC.

We design the fault tolerant control such that the noise amplification is allowed to achieve $\gamma_1 = 1.5$. Nota that this leads to a different relative amplification of the noise amplification, being $\alpha = 0.095$, $\alpha = 0.801$, and $\alpha = 4.455$ for each system, respectively. Then, the resulting controllers for FTC are

$$C_{u,1}(s) = \frac{e^{-0.1s}}{1+6.781s}, \qquad C_{y,1}(s) = -\frac{1+0.9s}{1+6.781s},$$
$$C_{u,2}(s) = \frac{e^{-0.5s}}{1+0.7498s}, \qquad C_{y,2}(s) = -\frac{1+0.5s}{1+0.7498s},$$
$$C_{u,3}(s) = \frac{e^{-0.9s}}{1+0.08161s}, \qquad C_{y,3}(s) = -\frac{1+0.1s}{1+0.08161s}.$$

Figure 2 shows the response under step faults of both the original control system and the fault tolerant proposed scheme. We appreciate an improvement for the first system, in which the delay is not dominant. However, we do not appreciate that improvement in the other two systems. The integral value of the error in absolute value ($IAE = \int |e|dt$) for each of the systems without and with the FTC proposal, and the maximum achieved error $e_{\max}$ are show in table 1. We appreciate that the proposed strategy improves the behavior under step faults in systems where the delay is low with respect the dynamics, that coincides with the systems where there is more room to amplify the noise w.r.t. the original controller (i.e., a larger value in α).

Table 1. Behavior comparison under step faults

System	G_1	G_2	G_3
original IAE	0.3731	0.8332	1.4092
IAE with FTC	0.5852	1.0717	1.9671
original $e_{\max}$	0.3093	0.6970	1
$e_{\max}$ with FTC	0.2968	0.6721	1
α	0.095	0.801	4.455

Figure 3 shows the behavior under ramp faults, where we can see that with the proposed approach we can mitigate the faults in steady state. The response time under these faults, as well as the maximum error, depends again on the predominance of the delay on the system. In figure 4 we show the behavior under a sinusoidal fault of unitary amplitude and frequency 0.4π rad/sec. We see that in the system with low delay, our approach diminish the effect of the fault, but our approach is not able to

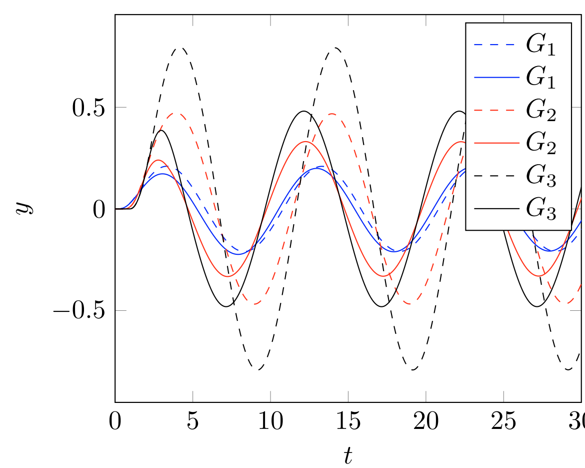

Fig. 4. Output response under unitary sinusoidal faults. Dashed: PI control; Solid: FTC.

improve the behavior in that scenario with higher delays. In figure 5 we show the magnitude of the frequency response of the transfer function that goes from fault f to output y in both the standard controlled system, and the one with our FTC approach. We see how the slope changes from $-20dB/dec$ to $-40dB/dec$ meaning that the original one can mitigate step fault signals, and our approach ramp fault ones. We also see that the behaviour remains the same for high frequencies. We see, however, that our approach can lead to a higher magnitude at some intermediate frequencies. Finally, in figure 6 we show the magnitude of the frequency response of the transfer function that goes from measurement noise v to control action u, showing that the FTC design have the same high frequency magnitude (as it has been the design criteria in this example). We also observe a higher magnitude for medium frequencies than the one with the PI controller. Finally, we can observe $0dB$ at low frequencies, what shows that this approach does not modify the behaviour under sensor faults w.r.t. PI controller.

5.2 Case of Study 2

In this example we analyze the effect of the parameter α and, therefore, parameter τ on the achieved performance

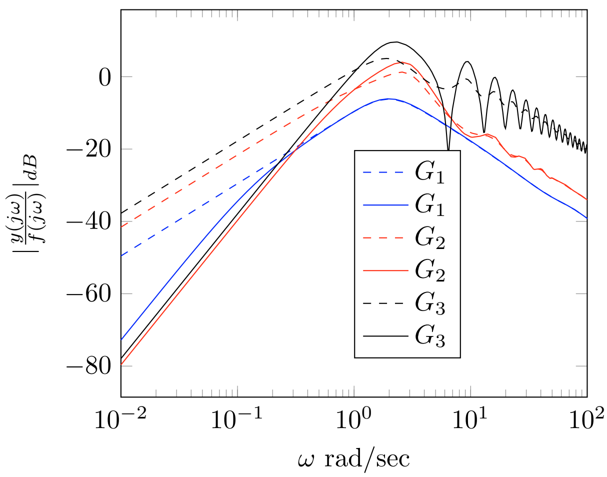

Fig. 5. Bode magnitude of the output due to faults. Dashed: PI control; Solid: FTC.

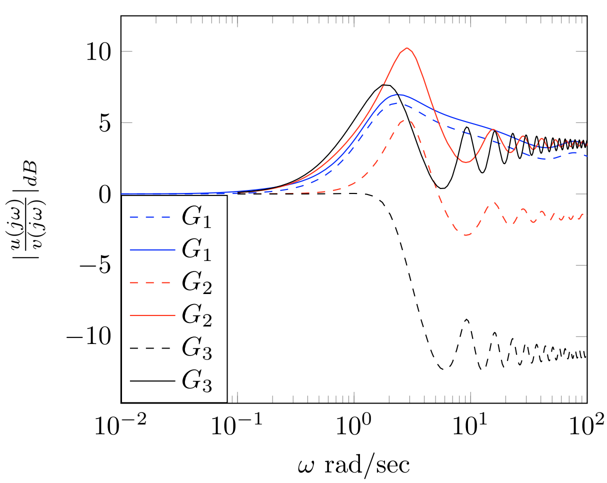

Fig. 6. Bode magnitude of actuator activity due to measurement noise. Dashed: PI control; Solid: FTC.

with the proposal of this paper. We consider now the system and controller

$$G(s) = \frac{e^{-s}}{(1+s)^2}, \qquad C(s) = 2.01 + \frac{0.787}{s},$$

and we design several fault tolerant controllers with an allowed noise amplifications in the set $\alpha = \{0.1, 1, 4\}$, i.e., allowing an increment of a 10% in the amplification, twice the original amplification, and 5 times the original one, respectively. This leads to the controllers indicated in table 2. We can see in the table the obtained τ (2.23, 0.706 and 0.353, respectively) as well as the achieved IAE and maximum error under step faults. The response time under step faults can be observed in figure 7. We observe that when we allow a low noise amplification ($\alpha = 0.1$), the IAE is not improved and the maximum error just decrease a 5%. When higher noises amplifications are allowed, the results are improved. We can see in figure 8 the achieved IAE under step faults, and we deduce that the IAE is improved with the proposed approach if $\alpha > 0.2$ w.r.t the IAE without the proposed fault tolerant controller (IAE= 1.2675). Figure 9 shows the response time under ramp faults. We can see that the response is improved when α is increased. We can appreciate in figures 10 and 11 how the frequency response of the output from faults and of the control action from noises changes as a function of α. We see how higher values of α lead to higher frequencies for which the fault input is mitigated, but at the cost of a higher measurement noise amplification at the actuator.

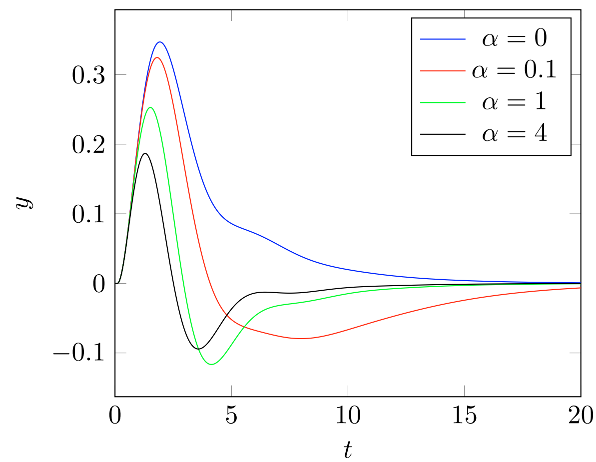

Fig. 7. Output response under unitary step faults.

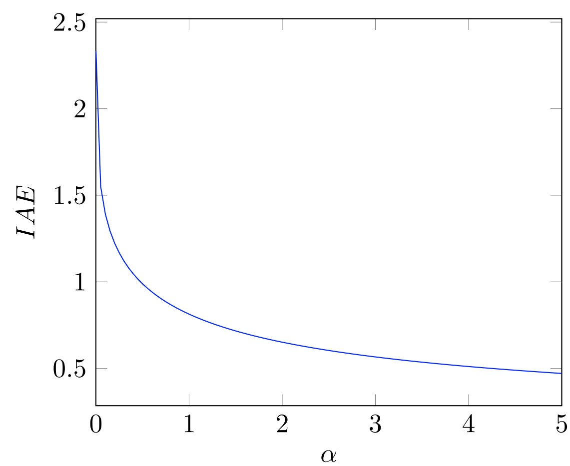

Fig. 8. IAE dependency on α.

Table 2. Behavior comparison under step faults

α	0	0.1	1	4
$C_u(s)$	0	$\frac{e^{-0.1s}}{(1+2.23s)^2}$	$\frac{e^{-0.1s}}{(1+0.706s)^2}$	$\frac{e^{-0.1s}}{(1+0.353s)^2}$
$C_y(s)$	0	$\frac{-(1+s)^2}{(1+2.23s)^2}$	$\frac{-(1+s)^2}{(1+0.706s)^2}$	$\frac{-(1+s)^2}{(1+0.353s)^2}$
IAE	1.2675	1.3725	0.8116	0.5100
$e_{\max}$	0.3469	0.3246	0.2528	0.1868

6. CONCLUSIONS

In this work, we have addressed the problem of incorporating actuator fault tolerant control ability in a control system that is initially running with a simple PI controller. We have looked for a FTC structure that is easy to implement and that does not modify the closed-loop behavior under references nor the robustness. We have proposed an structure which does not depend on the PI controller of the system and that is easy to implement because it only depends on a single parameter. This parameter allows us to decide a given trade-off between noise amplification and time response improvement under faults. With the

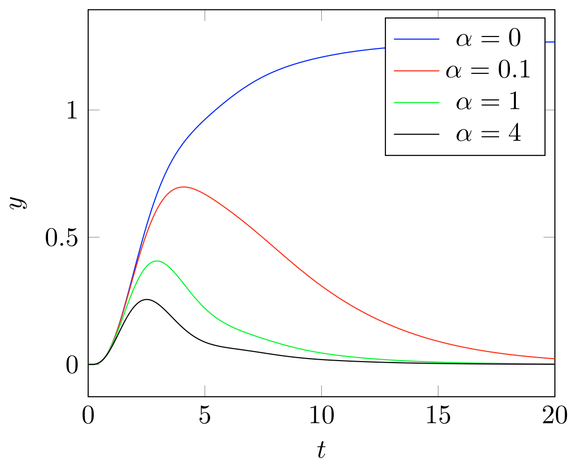

Fig. 9. Output response under unitary ramp faults.

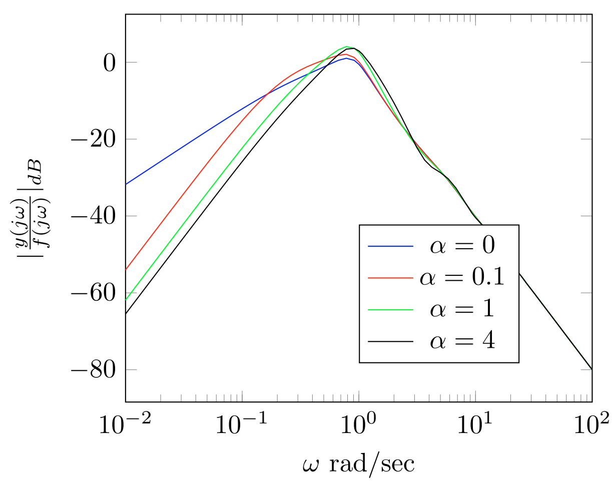

Fig. 10. Bode magnitude of the output due to faults.

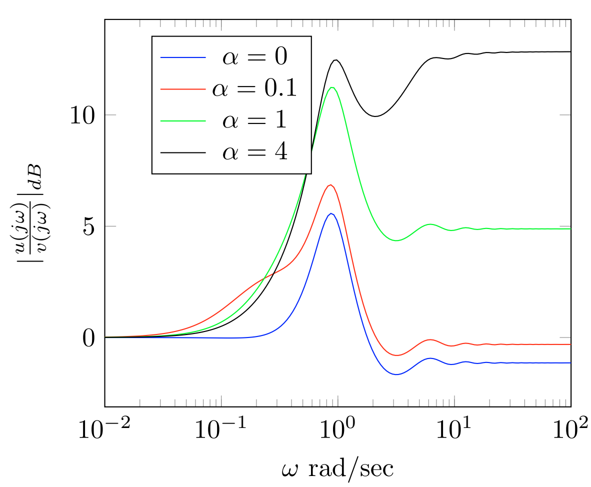

Fig. 11. Bode magnitude of actuator activity due to measurement noise.

proposed structure we can achieve null steady-state error under ramp faults, a property that cannot be achieved by a simple PI controller. But we have shown in the examples that the proposal can lose its properties if the system has long delays or if high noise amplification must be avoided. We have analyzed wether the approach is extensible to sensor fault tolerant control and we have showed that it would require an implementation of an unstable controller. In this sense, future work will include alternative structures that can be applicable for sensor fault tolerant control and the extension of these additive approaches to multivariable control systems.

REFERENCES

Åström, K.J. and Hägglund, T. (2006). *Advanced PID control*. ISA-The Instrumentation, Systems and Automation Society.

Blanke, M., Kinnaert, M., Lunze, J., Staroswiecki, M., and Schröder, J. (2006). *Diagnosis and fault-tolerant control*, volume 691. Springer Science & Business Media.

Desborough, L. and Miller, R. (2002). Increasing customer value of industrial control performance monitoring-honeywell's experience. In *AIChE symposium series*, 326, 169–189. New York; American Institute of Chemical Engineers; 1998.

Gao, Z., Cecati, C., and Ding, S.X. (2015). A survey of fault diagnosis and fault-tolerant techniquespart i: Fault diagnosis with model-based and signal-based approaches. *IEEE Transactions on Industrial Electronics*, 62(6), 3757–3767.

Patton, R.J. (2015). Fault-tolerant control. *Encyclopedia of Systems and Control*, 422–428.

Peñarrocha, I., Dolz, D., and Sanchis, R. (2015). Performance trade-offs for networked jump observer-based fault diagnosis. *IEEE Transactions on Signal Processing*, 63(10), 2692–2703.

Rotondo, D., Nejjari, F., and Puig, V. (2014). A virtual actuator and sensor approach for fault tolerant control of lpv systems. *Journal of Process Control*, 24(3), 203–222.

Sales-Setién, E., Peñarrocha, I., Dolz, D., and Sanchis, R. (2016). Performance-based design of PI observers for fault diagnosis in LTI systems under gaussian noises. In *3rd Conference on Control and Fault-Tolerant Systems (SysTol), 2016*, 407–412. IEEE.

Shen, Q., Jiang, B., and Shi, P. (2017). Active fault-tolerant control against actuator fault and performance analysis of the effect of time delay due to fault diagnosis. *International Journal of Control, Automation and Systems*, 15(2), 537–546. doi:10.1007/s12555-015-0307-5. URL `https://doi.org/10.1007/s12555-015-0307-5`.

Shi, F. and Patton, R. (2015). An active fault tolerant control approach to an offshore wind turbine model. *Renewable Energy*, 75, 788–798.

Simani, S. and Castaldi, P. (2014). Active actuator fault-tolerant control of a wind turbine benchmark model. *International Journal of Robust and Nonlinear Control*, 24(8-9), 1283–1303.

Wang, Y.G. and Shao, H.H. (2000). Optimal tuning for PI controller. *Automatica*, 36(1), 147–152.

Zhang, K., Jiang, B., and Shi, P. (2012). *Observer-based fault estimation and accomodation for dynamic systems*, volume 436. Springer Science & Business Media.

Zhang, P. and Ding, S.X. (2008). An integrated trade-off design of observer based fault detection systems. *Automatica*, 44(7), 1886–1894.