A note on frequency spectrum analysis and finite-time stability of the time trajectory-based active fault-tolerant control

Tushar Jain * **Joseph J. Yamé** **

* *Laboratory of Process Control and Automation, School of Chemical Technology, Aalto University, Kemistintie 1, FI-00076 Aalto, Finland. (e-mail: tushar.jain@aalto.fi).*
** *Centre de Recherche en Automatique de Nancy, CRAN-CNRS, UMR 7039, Université de Lorraine, Campus Sciences, 54506 Vandoeuvre-lès-Nancy, France. (e-mail: joseph.yame@univ-lorraine.fr)*

Abstract: The aims of this paper are two folds. First, we aim to gain some insights at the time Trajectory-based Active Fault-Tolerant Control (TAFTC) using the frequency spectrum analysis. Secondly, we introduce the concept of finite-time stability (FTS) in the context of AFTC systems. Generally, in TAFTC strategy, the desired specifications are defined in terms of time-domain characteristics under the mathematical framework of behavioral system theory. In this novel fault-tolerant mechanism, we do not work in the traditional input/output setting, i.e. the frequency-domain, at the outset. Instead, we use the system time-trajectories. As an extension, we report some results within this TAFTC strategy showing what frequency-domain properties are actually satisfied by the closed loop. Since the TAFTC scheme is based on the trajectories generated by the system, where no *a priori* information regarding the plant is known at a run-time, the theory of FTS renders a more practical insight than is provided by the classical theory to study the "behavior" of systems.

Keywords: fault-tolerant control, finite-time stability, frequency spectrum, behavioral theory.

1. INTRODUCTION

A fault, in general, is defined as un-permitted dynamics that changes the dynamics of a closed-loop system in such a way it no longer satisfies the desired specifications. Thus, the aim of fault tolerant control (FTC) is to counteract those altered dynamics by applying a suitable control law such that the system *encore* achieves the prescribed specifications. Predominately, a traditional process to re-establish the desired specifications undergoes the following two cascade stages: Fault Detection and Diagnoses (FDD), and controller reconfiguration (CR). The purpose of FDD is to use available signals to detect, identify, and isolate possibly the sensor faults, actuator faults, and any other system faults. Subsequently, the CR module reckons to-be-required actions so the system can still continue to operate safely even under faulty conditions Blanke et al. [2003].

Focusing on some of the disadvantages, described in Jain et al. [2011], of the above classical approach to address an FTC problem, the authors have proposed a novel technique in Jain et al. [2012a]. This approach is developed by taking the time-trajectory viewpoint of behavioral system theory Polderman and Willems [1997]. We term this FTC approach as time trajectory-based active fault-tolerant control (TAFTC). In this approach, no *a priori* information about the plant is used in real-time. In addition, neither the model of the plant is determined. Hence, no use of FDD module is seen in this approach. Instead, TAFTC strategy uses only the real-time measurements generated by the system. Finally, based on the desired specifications, the controller is reconfigured online such that an occurred fault can be accommodated.

While putting forward this novel approach, the stability issues of the overall scheme were not addressed. A preliminary work on the stability issue was addressed in Jain et al. [2012c]. In this paper, we use the concept of finite-time stability (FTS) that dates back to the sixties Weiss and Infante [1967]. Since then various version of FTS were introduced in the literature Dorato [2006], e.g. in the context of state-space equations, in the context of transfer functions, for time-varying systems, etc. Basically, the FTS concept differs from classical stability in two important ways. First, it deals with systems whose operation is limited to a fixed finite interval of time. Second, FTS requires prescribed bounds on system variables. Most of the results related to stability and performance in the AFTC research community are defined over an infinite time interval. This could only be feasible if a model of the plant is available in real-time. Unlikely, the TAFTC strategy deals with the behavior of the system over a fixed finite time interval. Therefore, FTS offers practical advantages. In addition, the concept of FTS has never been presented or dealt within the literature of fault-tolerant systems.

The second contribution of this paper is to present the spectral analysis of a TAFTC system. The key concept within the TFTC strategy contains only the time-domain specifications. At the outset, it is difficult to realize that which desired specifications in frequency-domain are achieved by the closed-loop system. Consequently, in addition to proving the FTS, another contribution is to investigate the frequency-domain specifications achieved by this novel TAFTC scheme. This helps one to gain more insights and to infer explicit advantages offered by this scheme.

2. TIME TRAJECTORY-BASED ACTIVE FAULT-TOLERANT CONTROL

We view a dynamical system as an exclusion law that indicates which trajectories are admissible for the system. A trajectory is a vector-valued function $s : \mathbb{T} \to \mathbb{S}, t \mapsto s(t)$ that take its values in the signal space $\mathbb{S}$ where $\mathbb{T} \subseteq \mathbb{R}, \mathbb{S} \subseteq \mathbb{R}^{\mathtt{s}}$ with $\mathtt{s}$ denoting the dimension of $s(t)$. The behavior of such systems can be expressed as the set of solutions of a system of linear, constant-coefficient differential equations. The set of all linear differential systems with $\mathtt{s}$ variables will be denoted by $\mathfrak{L}^{\mathtt{s}}$. The system is defined by a linear differential equation

$$R_0 s + R_1 \frac{d}{dt} s + R_2 \frac{d^2}{dt^2} s + \ldots + R_{\mathtt{n}} \frac{d^{\mathtt{n}}}{dt^{\mathtt{n}}} s = 0, \qquad (1)$$

where $R_i, i = 0, 1, 2, \ldots, \mathtt{n}$ are real constant matrices belonging to $\mathbb{R}^{\bullet \times \mathtt{s}}$ with finite number of rows and $\mathtt{s}$ columns. Equation (1) can be compactly written as

$$R\left(\frac{d}{dt}\right) s = 0, \quad R(\xi) \in \mathbb{R}^{\bullet \times \mathtt{s}}[\xi], \qquad (2)$$

with $R(\xi) = R_0 + R_1\xi + R_2\xi^2 + \ldots + R_{\mathtt{n}}\xi^{\mathtt{n}}$ where $\mathbb{R}^{\bullet \times \mathtt{s}}[\xi]$ denotes the set of $\bullet \times \mathtt{s}$ polynomial matrices with real coefficients and indeterminate ξ. Then the behavior $\mathcal{B}$ is given by the set

$$\mathcal{B} = \{s \in (\mathbb{R}^{\mathtt{s}})^{\mathbb{R}} | s \text{ satisfies } (2)\}. \qquad (3)$$

The representation used in (2) is called the kernel representation of $\mathcal{B}$, and we often write it as $\mathcal{B} = \mathtt{ker}(R(\frac{d}{dt}))$. From the above, clearly a dynamical system is now represented by a set of operating signals given in (3). The shift from representing a dynamical system as an input/output processor standpoint to an equivalent set of solutions is the key idea within the novel time trajectory-based FTC approach.

2.1 Feedback Interconnection of Dynamical Systems

The concept of interconnection plays the central role in modeling and control of system in the behavioral framework. By an interconnected system, we mean a system that consists of interacting subsystems. Here, we deal within a special type of interconnection, termed as the feedback interconnection. Let $\mathcal{P} \in \mathfrak{L}^{\mathtt{s}}$ denotes the behavior of the plant and $\mathcal{C} \in \mathfrak{L}^{\mathtt{s}}$ denotes the behavior of the controller, where $s = \begin{bmatrix} r^T & y^T & u^T \end{bmatrix}^T$, whose values lie in the signal space $\mathbb{S}$ having the dimension $\mathtt{s} = \mathtt{r} + \mathtt{y} + \mathtt{u}$. In the sequel, we denote this column vector by $s = \mathtt{col}(r, y, u)$. See Fig. 1 for a pictorial description of $\mathcal{P}$ and $\mathcal{C}$. Taking the behavioral point of view, we can now define the trajectory-based behaviors of the plant as

$$\mathcal{P} = \left\{ s = \mathtt{col}(r, y, u) \in \mathbb{S}^{\mathbb{T}} \middle| R\left(\frac{d}{dt}\right) s = 0 \right\}, \qquad (4)$$

where $R(\xi) = [0_{\mathtt{r}} \;\; D_p(\xi) \;\; -N_p(\xi)]$ with $D_p(\xi) \in \mathbb{R}^{\bullet \times \mathtt{y}}[\xi], N_p(\xi) \in \mathbb{R}^{\bullet \times \mathtt{u}}[\xi]$ being co-prime polynomials, and $0_{\mathtt{r}}$ representing the zero matrix of dimension $\mathtt{r}$. From the classical input/output point of view, y is considered as the output of the plant and u as the input. With this partition of inputs and outputs, evidently $D_p(\xi)^{-1} N_p(\xi) = G(\xi)$ defines a proper rational matrix with $D_p(\xi) \neq 0$ [Willems 1991, Section VIII]. In a similar way, the behavior of the controller is given by

$$\mathcal{C} = \left\{ s = \mathtt{col}(r, y, u) \in \mathbb{S}^{\mathbb{T}} \middle| C\left(\frac{d}{dt}\right) s = 0 \right\}, \qquad (5)$$

where $C(\xi) = [N_c(\xi) \;\; -N_c(\xi) \;\; -D_c(\xi)]$ with $D_c(\xi) \in \mathbb{R}^{\bullet \times \mathtt{u}}[\xi], N_c(\xi) \in \mathbb{R}^{\bullet \times \mathtt{y}}[\xi]$ being co-prime polynomial, and $D_c(\xi)^{-1} N_c(\xi) = H(\xi)$ representing a proper rational matrix with $D_c(\xi) \neq 0$. In this controller configuration, u is the output of the controller, and (r, y) are the inputs. The interconnection of $\mathcal{P}$ and $\mathcal{C}$ through the shared variable s results in a system in which this shared variable satisfies the dynamics of both $\mathcal{P}$ and $\mathcal{C}$. The behavior of this interconnection system is termed as the controlled behavior or the *implemented behavior* $\mathcal{K}$, and this is given by the set $\mathcal{K} = \mathcal{P} \cap \mathcal{C}$, which is equivalent to

$$\mathcal{K} = \{s = \mathtt{col}(r, y, u) \,|\, s \in \mathcal{P} \text{ and } s \in \mathcal{C}\}, \qquad (6)$$

where the symbol '$\cap$' denotes the interconnection operation. Note that the interconnection symbol '$\cap$' is akin to the interconnection of sets. The following definition formalizes the above concept of implementability.

Definition 1. (Implementability) Let $\mathcal{P} \in \mathfrak{L}^{\mathtt{s}}$ be a linear differential system, $\mathcal{C} \in \mathfrak{L}^{\mathtt{s}}$ be a controller, and $\mathcal{K} \in \mathfrak{L}^{\mathtt{s}}$. Whenever $\mathcal{K}$ is obtained by interconnecting $\mathcal{P}$ and $\mathcal{C}$, we say that "$\mathcal{C}$ implements $\mathcal{K}$". In addition, for a given $\mathcal{K} \in \mathfrak{L}^{\mathtt{s}}$ whenever there exists $\mathcal{C} \in \mathfrak{L}^{\mathtt{s}}$ such that $\mathcal{C}$ implements $\mathcal{K}$, $\mathcal{K}$ is said to be implementable by the interconnection.

The above definition implies that $\mathcal{K}$ is the restricted behavior satisfying the dynamics of both $\mathcal{P}$ and $\mathcal{C}$. Accordingly, a given $\mathcal{K} \in \mathfrak{L}^{\mathtt{s}}$ is implementable by an interconnection with respect to $\mathcal{P}$ if and only if $\mathcal{K} \subseteq \mathcal{P}$.

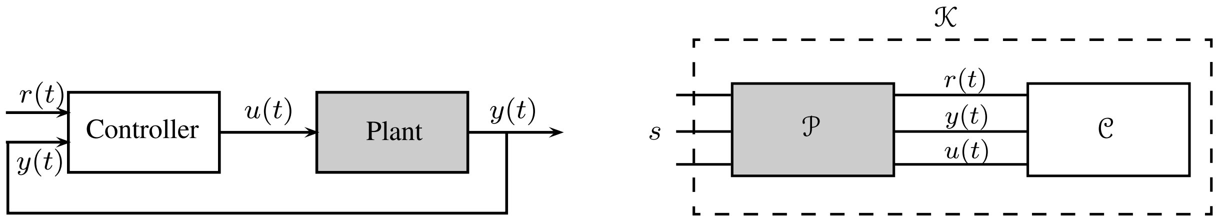
2.2 Effect of Faults

The real-time notion of controlling a faulty system is that the operating plant must achieve the control objectives at anytime, i.e. regardless of any occurrence of a fault. In this respect, we can single out a subset of plants' behavior as desirable. We call it the desired behavior, denoted by $\mathcal{D} \in \mathfrak{L}^{\mathtt{s}}$. The underlying principle is that the set of solutions belonging to the desired behavior satisfy the control objectives. We define the desired behavior as

$$\mathcal{D} = \left\{s = \mathtt{col}(r, y, u) \in \mathbb{S}^{\mathbb{T}} \,|\, J(s) \leq \lambda \right\}, \qquad (7)$$

where $J : (\mathbb{R}^{\mathtt{s}})^{\mathbb{R}} \to \mathbb{R}, s \mapsto J(s)$ defines the control performance functional with $\lambda \in \mathbb{R}$ denoting the threshold limit below which the performance is considered satisfactory.

Faults affect the dynamics of the system in a way that the control specifications are not satisfied. However, in some cases, the operating controller in the feedback control loop is extremely robust making a fault tolerable within an FTC system. Hence, no change in the control law would



(a) In classical sense (b) In behavioral sense

Fig. 1. Feedback interconnection : shaded behavior indicates that no *a priori* information is available in real-time

be required. With the above considerations, we define two classes of faults, namely *minor faults*, and *major faults* Jain [2012].

Definition 2. (Minor Faults). A fault is said to be a minor fault whenever there is no need of reconfiguring the controller in the closed-loop.

Definition 3. (Major Faults). A fault is said to be a major fault whenever $\mathcal{K} \nsubseteq \mathcal{D}$.

The real-time FTC problem we are dealing with is posed in the following way. Given the desired behavior $\mathcal{D}$, the problem is to find an appropriate controller $\mathcal{C}$, without using any *a priori* knowledge about the system in "real-time", which have the suitable control actions such that the controlled behavior $\mathcal{K}$ satisfy the desired behavior $\mathcal{D}$ at anytime.

2.3 Design and Implementation of TAFTC

In the traditional structure of AFTC systems, it is a fact that a precise knowledge of running plant is required during the fault diagnosis operation. On the other side, the novelty of utilizing the behavioral approach lies in its time-trajectories outlook of approaching an FTC problem, where no such knowledge is required. Nevertheless, the first stage for developing a fault-tolerant system requires Failure Mode and Effective Analysis (FMEA) Blanke et al. [2003]. FMEA's objective is to forecast systematically how fault effects on elements relate to faults at inputs, or outputs within the elements, and what reactions should be imposed on the system when a certain fault appears. Therefore, a mandatory prerequisite for achieving fault-tolerance is to have an effective FMEA of the system. We termed this phase as the Analysis & Development (AD) phase, which aims to provide a complete coverage of possible occurring faults in the closed-loop as well as the corresponding remedial measures. From the AD phase, it is assumed that a finite set of controllers

$$\mathbf{C} = \{\mathcal{C}_1, \mathcal{C}_2, \ldots, \mathcal{C}_N\} \tag{8}$$

is constructed, which makes the desired behavior $\mathcal{D}$ implementable. An approach to perform this analysis procedure is discussed in Wu [2004] and the references therein.

The Behaviors All modeling assumptions about the operating plant are embedded within the $R(\xi)$ matrix given in (4). At a run time, this matrix will not be determined. We form a measurement set $\mathcal{M}$, which is non-empty subset of $\mathbb{S}^{\mathbb{T}}$. From the real-time viewpoint, this is formalized in the following definition.

Definition 4. (Experimental plant's behavior). Given a vector space of time-dependent signals $\mathbb{S}^{\mathbb{T}}$, a dynamical system $\Sigma_{\mathcal{P}} = (\mathbb{T}, \mathbb{S}, \mathcal{P})$, and a measurement set $\mathcal{M} \subseteq \mathbb{S}^{\mathbb{T}}$, the behavior of the plant $\mathcal{P}$ is a superset of the (experimental) measurement on time intervals, i.e.

$$\mathcal{M}_\tau \subseteq \mathsf{O}_\tau(\mathcal{P}) \tag{9}$$

where O_τ is the time truncation operator given as

$$[\mathsf{O}_\tau(x)](t) = \begin{cases} x(t), & t_n - \tau \le t < t_n \ ; \\ 0, & \text{otherwise.} \end{cases}$$

where $t_n = n\tau, \forall n = 1, 2, \ldots$.

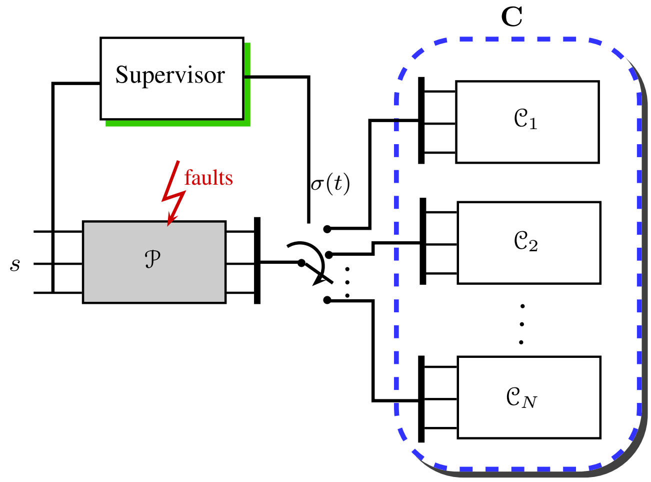


Fig. 2. System Architecture of time trajectory-based Active FTC system in the behavioral context

The role of introducing the time-truncation operator is to produce time-dependent subsets $\mathcal{M}_\tau$ of $\mathcal{M}$ for the interval of length $\tau \in \mathbb{R}$. From the above definition, for any controller $\mathcal{C}$ together with the behavior of the plant, we have

$$\mathsf{O}_\tau(\mathsf{O}_\tau^{-1}(\mathcal{M}_\tau) \cap \mathcal{C}) \subseteq \mathsf{O}_\tau(\mathcal{K}), \tag{10}$$

where $\mathsf{O}_\tau^{-1}(\mathcal{M}_\tau)$ denotes the pre-image of $\mathcal{M}_\tau$. It is interesting to note from (10) that the controlled behavior, by construction, is formulated independently of the matrix $R(\xi)$.

Set-up of the TAFTC Architecture The structure of the real-time fault-tolerant control in the behavioral context is provided in Fig. 2. Indeed, this structure is similar to what is usually used in the traditional projection-based approach. However, the novelty lies in the construction of the supervisor. Note that the plant in the figure is shaded as we do not have any *a priori* knowledge of it in real-time. Conversely, in the traditional approach, the supervisor is constructed by assuming the plant's knowledge which

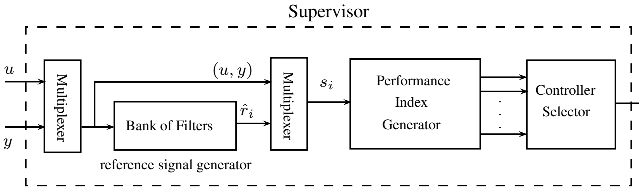


Fig. 3. Internal structure of the supervisor within the time trajectory-based FTC scheme of Fig. 2

performs the "when-which" task, i.e. *when* to change the control law, and *which* controller should be switched in the closed-loop. To perform this task without the aforesaid assumptions, we construct a supervisor as illustrated in Fig. 3, whose job is to select and switch in run time the correct controller within the closed-loop in one-shot, i.e. in a single switch. This supervisor is constructed using a bank of filters, a Performance Index Generator (PIG) block and a controller selector block.

2.3.2.1. Bank of Filters Since no knowledge of the plant's model is available, we only obtain a measurement set $\mathcal{M}_\tau$ during the interval of length τ, which is composed of the trajectories $u(t)$ and $y(t)$ produced by the plant. If a controller $\mathcal{C}$ were in the loop when the plant produced the trajectories $\mathtt{col}(y, u)$, then the restrictions imposed by the controller behavior (5) would be

$$D_c(\xi)u(t) = N_c(\xi)r(t) - N_c(\xi)y(t), t_n - \tau \leq t < t_n$$

for some $r \in (\mathbb{R}^{\mathtt{r}})^{\mathbb{R}}$ or equivalently,

$$N_c(\xi)r(t) = D_c(\xi)u(t) + N_c(\xi)y(t), t_n - \tau \leq t < t_n. \quad (11)$$

Assume that all controllers in the controller's bank (8) are stable causally left invertible, then based on the observed set $\mathcal{M}_\tau$ the corresponding reference trajectory $r(t)$ can be evaluated as

$$\hat{r}_i(t) = (N_{c_i}(\xi))^{-1}(D_{c_i}(\xi)u(t)+N_{c_i}(\xi)y(t)), t_n-\tau \leq t < t_n, \quad (12)$$

where $i \in \{1, 2, \ldots, N\}$. For the controller $\mathcal{C}$ in the actual loop, the trajectory $\hat{r}(t)$ asymptotically converges to $r(t)$, since subtracting (11) from (12) yields $N(\xi)(\hat{r}(t)-r(t)) = 0$, where $N(\xi)$ is the stable polynomial. Equation (11), in fact, yields the controlled behavior $\mathcal{K}$ as defined in (6), since here $(y, u) \in \mathcal{P}$ and $(\hat{r}, y, u) \in \mathcal{C}$. Consequently, for a measurement set $\mathcal{M}_\tau \subseteq \mathsf{O}_\tau(\mathcal{P})$, if there exists a trajectory $\hat{r}_i(t)$ corresponding to the i^{th} controller $\mathcal{C}_i, i = 1, 2, \ldots, N$, then it would yield the corresponding controlled behavior $\mathsf{O}_\tau(\mathcal{K}_i) \supseteq \mathsf{O}_\tau(\mathsf{O}_\tau^{-1}(\mathcal{M}_\tau) \cap \mathcal{C}_i), \forall i \in \{1, 2, \ldots, N\}$. Equation (12) defines a filter which reconstructs the virtual reference signal $\hat{r}(t)$ from the measurement set $\mathcal{M}_\tau$ Safonov and Tsao [1997]. From this, we have now determined the controlled behavior of all controllers with respect to $\mathcal{P}$ at run time, however, no knowledge of the plant's model is used here.

2.3.2.2. PIG block : The measurements generated by the plant together with the virtual reference, i.e. $\hat{s} = \mathtt{col}(\hat{r}, y, u) \in \mathbb{S}^{\mathbb{T}}$, are then fed to the PIG block. This block yields N performance indices

$$\{J(\hat{s}_i), i = 1, 2, \ldots, N\}, \quad (13)$$

for the corresponding N controllers, which are evaluated by considering the signal $\hat{s}$ during the interval of length τ.

2.3.2.3. Controller Selection : The controller selector block subsequently produces a piecewise constant signal (the switching signal) $\sigma(t)$ based on $\{J(\hat{s}_i)\}_{i=1}^N$ whose job is to select the controller that implements $\mathcal{D}$ from the bank of controllers. The switching signal is a map from the time axis $\mathbb{T}$ to the controllers index set $\{1, 2, ..., N\}$, i.e. $\sigma : \mathbb{T} \to \{1, 2, ..., N\}$. The control performance is evaluated during the interval of length τ, and if it requires switching of the controller, the switch will occur after time τ exclusively. Therefore, it imposes a lower bound on the length of intervals between successive switches. This minimum length of time in which a controller is active in the loop is known as the *dwell time* Morse [2008]. The control selection logic is then realized through $\sigma(t) = \sigma(t_n)$ for $t_n \leq t < t_{n+1}$ with the updating rule

$$\sigma(t_{n+1}) = \begin{cases} \sigma(t_n), & \text{if } \mathcal{K} \subseteq \mathcal{D}; \\ \mathtt{argmin}\{J(\hat{s}_i)\}_{i \neq \sigma(t_n)}, & \text{if } \mathcal{K} \nsubseteq \mathcal{D}. \end{cases} \quad (14)$$

The controller selector block contains the control selection algorithm given in (14).

3. FINITE-TIME STABILITY OF TAFTC

In the behavioral system theory, the stability is not considered as a property of a dynamical system, but of a *trajectory* generated by the system. However, for linear systems, the stability can be viewed as a property of the itself. The following definition formalizes the idea of stability within the class of linear systems.

Definition 5. A linear dynamical system is said to be stable whenever all elements of its behavior $\mathcal{B}$, i.e. $s = \mathtt{col}(r, y, u) \in \mathcal{B}$ are bounded on the half-line $[0, \infty)$. Otherwise, it is said to be unstable.

One can see that the above definition can be put in line to the classical concept of input/output stability, especially, bounded-input bounded-output stability. Indeed, this concept of stability considers either the trajectories characterized over an infinite time interval or the mathematical model of the system. Since, we do not have any *a priori* mathematical model of the system in real-time and we do not have trajectories up to time $t \to \infty$, we use the concept of finite-time stability to deal with AFTC systems.

Definition 6. Given positive scalars $T_1, T_2 \in \mathbb{R}, (T_2 > T_1)$, a linear dynamical system is said to be finite-time stable whenever all elements of its behavior $\mathcal{B}$, i.e. $s = \mathtt{col}(r, y, u) \in \mathcal{B}$ are bounded by a *prespecified* limit over the interval $t \in [T_1, T_2[$.

One can easily infer that the scalars defined above are in line with time limits specified in Definition 4. Note the difference between Definitions 5 and 6. First, the latter involves trajectories characterized over a finite time interval. Secondly, *quantitative bounds* are prespecified on trajectories unlike to the former definition. We want to guarantee that the novel TAFTC system is a finite-time stable system. To prove this, the desired behavior $\mathcal{D}$ that captures the dynamics of the closed-loop, should be finite-time stable. Otherwise, no controller can guarantee FTS of the closed-loop. First, we shall show when a closed-loop can be certified as a finite-time stable.

Proposition 1. Given a dynamical system $\Sigma_{\mathcal{C}} = (\mathbb{T}, \mathbb{S}, \mathcal{C})$, the desired behavior $\mathcal{D}$, and a measurement set $\mathcal{M}_\tau \subseteq$

$\mathsf{O}_\tau(\mathcal{P})$, the interconnection between $\mathcal{P}$ and $\mathcal{C}$ yields a 'stand-alone' finite-time stable system if

$$\mathsf{O}_\tau(\mathsf{O}_\tau^{-1}(\mathcal{M}_\tau) \cap \mathcal{C}) \subseteq \mathsf{O}_\tau(\mathcal{D}). \tag{15}$$

Proof Suppose the performance functional in the desired behavior $\mathcal{D}$ is chosen as

$$J(s = \texttt{col}(r, y, u)) = \frac{\|u\| + \|y\|}{\|r\| + \alpha_0},$$

where α_0 is a non-zero constant and $\| \bullet \|$ denotes the Euclidean norm. Within a finite-time interval of length τ, the stable desired behavior would be defined by

$$\mathsf{O}_\tau(\mathcal{D}) = \left\{ s = \texttt{col}(r, y, u) | J(s_\tau) = \frac{\|u\|_\tau + \|y\|_\tau}{\|r\|_\tau + \alpha_0} \leq \lambda \right\}. \tag{16}$$

Whenever the interconnection between the controller $\mathcal{C}$ and the unknown plant $\mathcal{P}$ is made, then $(r, y, u)_\tau \in \mathsf{O}_\tau(\mathcal{C})$ and $(y, u)_\tau \in \mathsf{O}_\tau(\mathcal{P}) \supseteq \mathcal{M}_\tau$. Note that no knowledge about the plant is used; instead we use the experimental plant's behavior. If the above interconnected system is included in the desired behavior then trajectories satisfy the inequality (16), which is equivalent to

$$\|u\|_\tau \leq \lambda_1 \|r\|_\tau + \delta_1, \qquad \|y\|_\tau \leq \lambda_2 \|r\|_\tau + \delta_2.$$

with $\Sigma_{i=1}^2 \lambda_i = \lambda$ and $\delta_i = \lambda_i \alpha_0, i = 1, 2$. Hence, according to the Definition 6, the closed-loop is a 'stand-alone' finite-time stable system.

Indeed, at a runtime the FTS of the closed-loop system cannot be guaranteed in the event of fault occurrence. In the sequel, we shall show when an operating closed-loop system can be certified as a finite-time stable system at a run-time. As shown earlier, there exists a controller in the bank that can achieve the desired behavior, and hence the FTS. A trivial solution is that one can test every controller *sequentially* in the closed-loop such that the Proposition 1 can be applied. However, at a runtime illustrating FTS in such a manner can lead the system to an unrecoverable mode, i.e. the mode from where an occurred fault cannot be accommodated. Therefore, to guarantee the FTS of the overall TAFTC scheme, it is sufficient showing that the controller that can stabilize the system, in the sense of FTS, switches in one-shot into the closed-loop.

Proposition 2. Given the stable desired behavior $\mathcal{D}$, and a measurement set $\mathcal{M}_\tau \subseteq \mathsf{O}_\tau(\mathcal{P})$. For any occurrence of a fault, the time trajectory-based fault-tolerant control system is finite-time stable.

Proof Without any loss of generality, consider a bank of three controllers $\mathbf{C} = \{\mathcal{C}_1, \mathcal{C}_2, \mathcal{C}_3\}$ is constructed in the AD phase. The real-time operation is initiated with an unknown $\mathcal{P}$ interconnected with the $\mathcal{C}_1$. Thus, $\mathsf{O}_\tau(\mathsf{O}_\tau^{-1}(\mathcal{M}_\tau) \cap \mathcal{C}_1) = \mathsf{O}_\tau(\mathcal{K}_1) \subseteq \mathsf{O}_\tau(\mathcal{D})$. Suppose, a minor fault occurs into the system. Since the occurred minor fault does not change the behavior of the system, we would still have $\mathsf{O}_\tau(\mathcal{K}_1) \subseteq \mathsf{O}_\tau(\mathcal{D})$. This implies that there is no need to change the controller $\mathcal{C}_1$. Hence, the closed-loop system is finite-time stable. Consider now that a major fault occurs into the system. Indeed, this will change the behavior of the plant, and suppose, this new behavior is then given by $\mathcal{P}^f$. From the Definition 3, it implies that an occurrence of major fault causes $\mathcal{K}_1 \nsubseteq \mathcal{D}$. Therefore, whenever the last inclusion satisfies, the operating controller in the loop is invalidated implying that a fault has occurred.

Again, without any loss of generality, suppose, that controller is $\mathcal{C}_2$ and not $\mathcal{C}_3$. Using the measurement set $\mathcal{M}_\tau$ generated by the plant, two virtual reference signals are evaluated by (11), which gives two sets of trajectories: $\hat{s}_2 = \texttt{col}(\hat{r}_2, y, u)$, and $\hat{s}_3 = \texttt{col}(\hat{r}_3, y, u)$. As described in Section 2.3, for these set of trajectories, we would have two corresponding virtual interconnected system, namely $\hat{\mathcal{K}}_2$ and $\hat{\mathcal{K}}_3$, defined as

$$\hat{\mathcal{K}}_2 = \{(\hat{s}_2 = \texttt{col}(\hat{r}_2, y, u) | (y, u) \in \mathcal{P}^f \text{ and } (\hat{r}_2, y, u) \in \mathcal{C}_2\},$$
$$\hat{\mathcal{K}}_3 = \{(\hat{s}_3 = \texttt{col}(\hat{r}_3, y, u) | (y, u) \in \mathcal{P}^f \text{ and } (\hat{r}_3, y, u) \in \mathcal{C}_3\}.$$

Interestingly, we have obtained the virtual closed-loop behaviors with every controller without putting actually the controllers into the loop. From the above, $\mathcal{C}_2$ is supposed to be that right controller, not $\mathcal{C}_3$. This, indeed, satisfies $\mathsf{O}_\tau(\hat{\mathcal{K}}_2) \subseteq \mathsf{O}_\tau(\mathcal{D})$. Consequently, the controller $\mathcal{C}_2$ will be switched in to the closed-loop by the switching logic (14), instead of the controller $\mathcal{C}_3$. As shown earlier, $\hat{r} = r$, which implies that $\hat{\mathcal{K}}_2 = \mathcal{K}_2$ and hence $\mathsf{O}_\tau(\mathcal{K}_2) \subseteq \mathsf{O}_\tau(\mathcal{D})$. Therefore, the time trajectory-based fault tolerant strategy is finite-time stable.

Above proposition demonstrates that the stability is guaranteed for the overall TAFTC scheme. However, a source of stability violation could be because of the switching between the controllers. The lack of dynamical consistency between the "state trajectories" of the to-be-switched control and the unknown plant is shown as the prime cause of stability violation Yamé and Kinnaert [2007]. Due to this inconsistency, bump appears whenever a new controller is introduced in the closed-loop at a run-time. To address this issue, we have used the algorithm proposed in Jain et al. [2012b], which guarantees the *smooth interconnection* between the controller and the unknown plant. The concept of smooth interconnection is demonstrated by using the behavioral framework. The main idea is to enforce the dynamical consistency between the controller and the plant such that whenever a controller is introduced in the loop, no bumps would appear. In other words, we *reset* the "state" of the to-be-switched controller without using any knowledge about the plant in real-time.

4. FREQUENCY SPECTRUM ANALYSIS OF TAFTC

From Fig. 3, it can be seen that the performance evaluator block plays a significant role in the selection of one of the controllers from the bank. Generally, the performance functional in the PIG block captures the control objective or the desired specification. No doubt this functional is composed of the plant trajectories together with the virtual reference signal. In the existing literature, with respect to this latter signal, various types of performance functional exist, see Jain et al. [2012c] for a brief overview. In this section, we shall see which properties are depicted by this functional from the viewpoint of frequency-domain.

First, we shall discuss the integral squared error (ISE) as the performance functional chosen in the desired behavior. Indeed, ISE is a usual and common performance index evaluator in practical control engineering problems Jain et al. [2013]. The desired behavior is then given by

$$\mathcal{D} = \{s = \texttt{col}(r, y, u) | J(s) = \int_t \|\hat{e}_i(\varsigma)\|^2 d\varsigma \leq \lambda\} \tag{17}$$

where $\hat{e}_i = \hat{r}_i - y, i = 1, 2, \ldots, N$ is the virtual error. Taking the spectral viewpoint, the virtual reference signal is given by $\hat{r}_i(f) = H_i^{-1}(f)u(f) + y(f)$, where $H_i^{-1}(f)$ is the inverse spectrum of the $i-$th controller. Above equations yield $\hat{e}_i = H_i^{-1}(f)u(f)$. One should keep in mind that the plant is operating in *closed-loop*, i.e. the signal $u(f)$ depends on the plant output signal $y(f)$. The behavior of the running plant G is captured by the signals (u, y), and in the classical input/output approach, this is expressed as $y(f) = G(f)u(f)$, where $G(f)$ is the frequency spectrum of the plant. On the other hand, we can also formally write $u(f) = G^{-1}(f)y(f)$, where $G^{-1}(f)$ is the inverse spectrum of the plant. The above equations yield the virtual error signal as $\hat{e}_i = H_i^{-1}(f)G^{-1}(f)y(f), i = 1, 2, \ldots, N$. Since the signal y is same for all virtual closed-loops being tested, the performance formally depends on $H_i^{-1}(f)G^{-1}(f)$, and this can be written as

$$J = \int_f \|H_i^{-1}(\varsigma)G^{-1}(\varsigma)y(\varsigma)\|^2 d\varsigma \tag{18}$$

It is interesting to note that the plant G in the last expression is the *actual* plant operating mode, and any occurrence of a fault results in a change of the operating mode. One would have certainly recognized that

$$\mathfrak{L}_{H_i}(f) = G(f)H_i(f) \tag{19}$$

is the *loop transfer* function of the feedback system (H_i, G), specifically with respect to H_i since the plant is same for all the virtual loops. Clearly, the switching logic made the comparison on the basis of the inverse of the loop transfer $\mathfrak{L}_{H_i}^{-1}(f)$. Consequently, the virtual feedback loop having the highest loop transfer gain (integral (18) will be minimal in that case) will have its controller switched in the actual real-time closed-loop by the supervisor. In fact, recalling that sensitivity is defined by $S = (1 + \mathfrak{L}^{-1})$ where $\mathfrak{L}$ is the loop transfer, one can clearly see that the supervisor seeks for the feedback loop which gives a *better* sensitivity function.

Since any other functional that capturing the control objectives can be chosen in the desired behavior. Now we will carry out the similar analysis with another performance functional used in Safonov and Tsao [1997], i.e.

$$J(s) = \|w_1 * (y - \hat{r}_i)\|^2 + \|w_2 * u)\|^2 - \|\hat{r}_i\|^2 \tag{20}$$

where $*$ is the convolution operator, signals $(r(t), y(t), u(t))$ have their usual meanings and w_1, w_2 are dynamical weights. In this case, the desired behavior would be then expressed as

$$\mathfrak{D} = \{s = \texttt{col}(r, y, u) | J(s) \leq 0\} \tag{21}$$

Taking the spectral viewpoint, and putting $\hat{r}_i(f) = H_i^{-1}(f)u(f) + y(f), u(f) = G^{-1}(f)y(f)$ in (20) we get

$$\frac{\|w_1(f)H_i^{-1}(f)G^{-1}(f)\|^2}{\|H_i^{-1}(f)G^{-1}(f) + 1\|^2} + \frac{\|w_2(f)G^{-1}(f)\|^2}{\|H_i^{-1}(f)G^{-1}(f) + 1\|^2} \leq 1$$

$$\frac{\|w_1(f)(GH_i)^{-1}(f)\|^2}{\|(GH_i)^{-1}(f) + 1\|^2} + \frac{\|w_2(f)G^{-1}(f)\|^2}{\|(GH_i)^{-1}(f) + 1\|^2} \leq 1$$

$$\frac{\|w_1(f)\|^2}{\|(GH_i)(f) + 1\|^2} + \frac{\|w_2(f)H_i(f)\|^2}{\|(GH_i)(f) + 1\|^2} \leq 1$$

$$\|w_1(f)S_i(f)\|^2 + \|w_2(f)H_i(f)S_i(f)\|^2 \leq 1$$

where $S_i(f) = \frac{1}{1+G(f)H_i(f)}$ is the sensitivity function with respect to $H_i(f), i = 1, 2, \ldots, N$. Again, one could have readily recognized that the supervisor selects a controller that satisfies an upper bound $w_1^{-1}(f)$ on $S_i(f)$. Specifically, it is desired that $|S_i(f)| \leq |w_1(f)|^{-1}$ for $f \geq 0$, while $w_2(f)$ aims to limit the control effort. The weighting function $w_1(f)$ gathers the desired specifications to be achieved.

From the above analysis, we conclude that without using any *a priori* model of the plant the supervisor selects one of the controllers that minimizes the sensitivity function. Roughly speaking, the performance depends on the controller-plant pairing (H_i, G) and the *well*-performing controller-plant pairings are pairings (H_j, G_j) where $j \in \{1, 2, ..., N\}$ is the index of the *actual* plant mode, that is, those pairings in which the controller matches the actual running mode for which it has been designed in the AD phase.

5. CONCLUSION

In this paper, we have gained some insights of the time trajectory-based fault-tolerant control. First, we showed the finite-time stability of an active fault-tolerant system. In the TAFTC approach, no *a priori* information about the plant is used in real-time. The desired behavior in this approach is generally defined in terms of time trajectories. We have showed what properties are actually satisfied by this novel FTC mechanism in the frequency-domain.

REFERENCES

Blanke, M., Kinnaert, M., Staroswiecki, M., and Lunze, J. (2003). *Diagnosis and Fault tolerant control*. Springer-Verlag.

Dorato, P. (2006). An overview of finite-time stability. In *Current Trends in Nonlinear Systems and Control*, 185–194. Springer.

Jain, T. (2012). *Behavioral system theoretic approach to fault tolerant control*. Ph.D. thesis, Université de Lorraine.

Jain, T., Yamé, J.J., and Sauter, D. (2011). Synergy of canonical control and unfalsified control concept to achieve fault tolerance. In *World Congress*, volume 18, 14832–14837.

Jain, T., Yamé, J.J., and Sauter, D. (2012a). Model-free reconfiguration mechanism for fault tolerance. *International Journal of Applied Mathematics and Computer Sciences*, 22(1), 125–137.

Jain, T., Yamé, J.J., and Sauter, D. (2012b). On implementing on-line designed controller for smooth interconnection in the behavioral framework. In *Robust Control Design*, volume 7, 424–429.

Jain, T., Yamé, J.J., and Sauter, D. (2012c). Role of performance evaluator in data-driven fault tolerant control. In *Fault Detection, Supervision and Safety of Technical Processes*, volume 8, 192–197.

Jain, T., Yamé, J.J., and Sauter, D. (2013). A real-time projection-based approach for fault accommodation in nrels 5mw wind turbine systems. In *IEEE American Control Conference*.

Morse, A. (2008). Lectures notes on logically switched dynamical systems. In *Nonlinear and Optimal Control Theory, Lectures Notes in Mathematics*. Springer-Verlag.

Polderman, J.W. and Willems, J.C. (1997). *Introduction to Mathematical Systems Theory: A Behavioral Approach.* Springer-Verlag.

Safonov, M. and Tsao, T.C. (1997). The unfalsified control concept and learning. *IEEE Transactions on Automatic Control*, 42(6), 843–847.

Weiss, L. and Infante, E.F. (1967). Finite time stability under perturbing forces and on product spaces. *International Transactions on Automatic Control*, 12, 54–59.

Willems, J.C. (1991). Paradigms and puzzles in the theory of dynamic systems. *IEEE Transactions on Automatic Control*, 36, 259–294.

Wu, N.E. (2004). Coverage in fault-tolerant control. *Automatica*, 40(4), 537 – 548.

Yamé, J.J. and Kinnaert, M. (2007). On bumps and reduction of switching transients in multicontroller systems. *Mathematical Problems in Engineering*, Volume 2007.