8th IFAC Symposium on Fault Detection,
Supervision and Safety of Technical Processes (SAFEPROCESS)
August 29-31, 2012. Mexico City, Mexico

Role of Performance Evaluator in data-driven Fault Tolerant Control

Tushar Jain * **Joseph J. Yamé** ** **Dominique Sauter** ***

* *Centre de Recherche en Automatique de Nancy (CRAN), CNRS,*
UMR 7039, Université de Lorraine, Vandoeuvre-lès-Nancy 54500,
France (e-mail: tushar.jain@univ-lorraine.fr).
** *e-mail: joseph.yame@univ-lorraine.fr*
*** *e-mail: dominique.sauter@univ-lorraine.fr*

Abstract: In this note, we discuss about the performance evaluator tool that plays a significant role in the data-driven fault tolerant control (FTC) system. In our notion of data-driven approach, we do not have any access to the *a priori* plant model in real-time. Moreover, we do not equip any estimation algorithm to determine the model of the plant. Here, we use the trajectories generated by the system in real time. These trajectories, in fact, capture the behavior of the system and we directly evaluate the control performance of the closed-loop based on these trajectories. Whenever a fault occurs, this tool assists the supervisor to take necessary actions for the controller reconfiguration mechanism. Since no *a priori* knowledge about the plant is used online, this tool is shown here performing all the necessary roles, particularly fault detection, fault accommodation, and especially, stability assessment.

Keywords: fault tolerant systems, control performance, linear systems.

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 (Blanke et al. (2003)). 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 desired specifications. Predominately, the 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. Conversely, the CR module reckons the to-be-required actions so the system can still continue to operate safely even under the faulty conditions. In terms of condition monitoring or FDD, the existing methods are grouped into the following two categories:

1. Model based FDD (Chen and Patton (1999));
2. Data driven FDD including knowledge based FDD (Hong et al. (2009)).

In the early days (1980's onwards), a model-based FDD constituted the main stream of research, and a number of techniques were developed. Depending on whether the system model can be represented as either a state-space model or an input-output model, FDD can roughly be classified in the following two groups: observer based FDD (Jain et al. (2010), Blanke et al. (2003)) and system identification based FDD (Isermann (1984)). On comprising these two respective modules individually with the CR unit, it results in the following FTC strategies, namely *model-based FTC* and *data-driven FTC*.

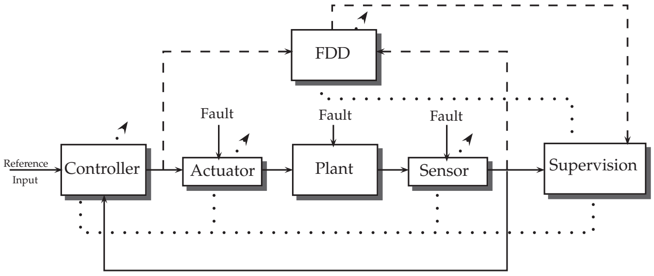

Fig. 1. Fault Tolerant Control system architecture with supervision sub-system Patton (1997).

An architectural description of an integrated FTC system with a supervisor unit is illustrated in Fig. 1. This figure shows a general functional scheme of a fault-tolerant control system with four prime components: the plant itself (including sensors and actuators), the fault detection and diagnosis (FDD) unit, the feedback (or feed-forward) controller, and the supervision sub-system. The main controller activities are represented by the solid line. The dashed line represents the operation of the FDD unit, and the dotted line represents the adaptation (tuning, scheduling, accommodation, and reconfiguration). In the faultless case, FDD unit remains inactive and the nominal feedback controller attenuates the disturbances and ensures set point following and other requirements on the closed-loop system. The FDD unit is responsible for providing to the supervision system with information about the onset, location, and severity of any occurring faults. On the supervision level, the diagnosis block simply

10.3182/20120829-3-MX-2028.00277

recognizes that the closed-loop system is faultless and no change of the control law is necessary. If a fault occurs, the diagnosis block identifies the fault. Based on the system inputs and outputs together with the fault decision information provided by the diagnosis module, the supervision sub-system reconfigures the sensor set and/or actuators to isolate the faults, and tune or adapt the controller to accommodate the fault effects so that the closed loop satisfies the performance specifications.

1.1 Existing Data-driven approach to FTC

In the existing literature, various data-driven approaches are studied to deal with a fault-tolerant control problem (Dong (2009), Hong et al. (2009), Jian-Xin and Zhong-Sheng (2009)). Primarily, the basic idea in most of the approaches is to construct an FDD module based on the collected (either online or offline) data. These approaches, however, lacks in satisfying the real-time aspects of an integrated FTC scheme (Zhang and Jiang (2008)). An FDD module reconstructs the system dynamics from the collected-data either by estimating the plant parameters or by matching the system trajectories with the off-line data. Therefore, the slow convergence issue exhibited by the system variables is always seen in such approaches. Concentrating on the timing issues involved in an integrated FTC system, we distribute the timing-intervals similar to a time-map studied in (Staroswiecki (2004)).

- $t \in [0, t_f[$: the system is in normal operation (model Σ_{nom}) and the applied control is the nominal one u_n.
- $t \in [t_f, t_{fd}[$: the system is faulty (model Σ_{faulty}), but the FDD algorithm has not yet detected, isolated and estimated the fault, and the control has not been reconfigured, therefore the nominal control u_n is still applied.
- $t \in [t_{fd}, t_{fdd}[$: the system is faulty (model Σ_{faulty}), and the FDD algorithm has detected the fault but has not yet isolated and estimated the fault, and the control has not been reconfigured, therefore the nominal control u_n is still applied.
- $t \in [t_{fdd}, t_R[$: the system is faulty (model Σ_{faulty}), the FDD algorithm has detected, isolated and estimated the fault, but the control has not yet been reconfigured, therefore the nominal control u_n is still applied.
- $t \in [t_R, \infty[$: the system is faulty (model Σ_{faulty}), the reconfigured control u_n^f has been computed and it is applied.

In (Staroswiecki (2004)), the only attention has been given towards reducing the controller reconfiguration time. However, the passage of time during the complete FDI/FDD operation is still there. As a result, our underlying aim is to reduce this passage of time as well, i.e. the time from the occurrence of a fault until the control reconfiguration. Stating otherwise, the elementary objective of our work is to reduce the fault accommodation time in comparison to that of seen in the classical model-based FTC scheme.

1.2 FTC problem formulation

Model-based FTC approaches have their own limitations to deal with model uncertainties in real-time. On the other hand, data-driven approaches that comprise the estimation of a plant model involve individual timing issues in fault diagnosis and fault accommodation. See (Jain et al. (2012b), Yamé and Sauter (2008)) for more details on these issues. It has been shown that the prime cause of these limitations is the use of the FDD unit for *reconfigurable* FTC systems. Therefore, our notion of data-driven approach to FTC does not even involve any use of an explicit FDD module.

The FTC problem is concerned with the control of the faulty system (Blanke et al., 2003, Definition 7.1), and our main central point takes into account the controller reconfiguration mechanism. We will show that in active FTC systems, the use of the online FDD module can be avoided providing the system can achieve the desired specifications by just changing the control law. Nevertheless, for other types of faults that require "reconfiguring the plant", i.e. the replacement of actuators or sensors while keeping the same (or even changing the) controller, one need an explicit FDI mechanism to identify the size and the location of a fault.

An FTC approach without utilizing an FDD module is also studied in (Ye and Yang (2006)). Unlike the (Ye and Yang (2006)), first we do not have the online estimates of an occurring fault. Secondly, we do not assume the availability of the system states at anytime. Here, our main objective is to re-configure the controller directly based on the trajectories generated by the system in *real-time*. This renders a fast and a reliable data-driven fault tolerant system. The presented FTC strategy lies under a broad category of projection-based active FTC mechanism. In our demonstrated control architecture for FTC in (Jain et al. (2012b)), the key role is played by the "performance evaluator". We directly evaluate the control performance of the closed-loop system unlike evaluating the estimator performance which is mostly seen in the projection-based FTC. Therefore, in this follow-up paper we discuss about the performance evaluator tool performing all the necessary roles, i.e. fault detection, controller reconfiguration and stability assessment.

2. MATHEMATICAL FRAMEWORK

In this section, we briefly introduce the mathematical framework used in our approach to achieve the fault tolerance property for the system. We employed the behavioral theory as one of the mathematical tools to support this approach (Polderman and Willems (1997)). Other tools are the virtual reference tool (Safonov and Tsao (1997)), and the norm based signals (Boyd and Barratt (1991)). As mentioned before, we are not equipped with any *a priori* knowledge of the plant model in real-time, following definition construes a dynamical system in a data-driven FTC.

Definition 1. A dynamical system Σ is represented by a triple $\Sigma = (\mathbb{T}, \mathbb{W}, \mathcal{B})$ where $\mathbb{T} \subseteq \mathbb{R}$ is the time axis, $\mathbb{W} \subseteq \mathbb{R}^{dim(\mathbf{w})}$ is the signal space with $\mathbf{w}$ as the signal, and $\mathcal{B} \subseteq \mathbb{W}^{\mathbb{T}}$ is *the behavior.* A trajectory is a function $\mathbf{w} : \mathbb{T} \to \mathbb{W},\ t \mapsto \mathbf{w}(t)$. $\square$

The set $\mathbb{W}$ is the space in which the system time-signals take on their values and the behavior $\mathcal{B} \subseteq \mathbb{W}^{\mathbb{T}}$ is a *family* of $\mathbb{W}$-valued time trajectories.