

Article

Gain Scheduled Fault Detection Filter for Markovian Jump Linear System with Nonhomogeneous Markov Chain

Leonardo Carvalho ^{1,*} , Jonathan M. Palma ² , Cecília F. Morais ³ , Bayu Jayawardhana ⁴ 
and Oswaldo L. V. Costa ¹ 

- ¹ Departamento de Engenharia de Telecomunicações e Controle, Escola Politécnica na Universidade de São Paulo, São Paulo 05508-010, SP, Brazil
- ² Facultad de Ingeniería, Universidad de Talca, Curico 3340000, Maule, Chile
- ³ Pontifical Catholic University of Campinas (PUC-Campinas), Center for Exact, Environmental and Technological Sciences (CEATEC), Postgraduate Program in Telecommunication Networks Management, Campinas 13086-900, SP, Brazil
- ⁴ Engineering and Technology Institute Groningen, Faculty of Science and Engineering, Rijksuniversiteit Groningen, 9747 AG Groningen, The Netherlands
- * Correspondence: carvalho.lp@usp.br

Abstract: In a networked control system scenario, the packet dropout is usually modeled by a time-invariant (homogeneous) Markov chain (MC) process. However, from a practical point of view, the probabilities of packet loss can vary in time and/or probability parameter dependency. Therefore, to design a fault detection filter (FDF) implemented in a semi-reliable communication network, it is important to consider the variation in time of the network parameters, by assuming the more accurate scenario provided by a nonhomogeneous jump system. Such a premise can be properly taken into account within the linear parameter varying (LPV) framework. In this sense, this paper proposes a new design method of \mathcal{H}_∞ gain-scheduled FDF for Markov jump linear systems under the assumption of a nonhomogeneous MC. To illustrate the applicability of the theoretical solution, a numerical simulation is presented.



Citation: Carvalho, L.; Palma, J.M.; Morais, C.F.; Jayawardhana, B.; Costa, O.L.V. Gain Scheduled Fault Detection Filter for Markovian Jump Linear System with Nonhomogeneous Markov Chain. *Mathematics* **2023**, *11*, 1713. <https://doi.org/10.3390/math11071713>

Academic Editors: Adrian Olaru and Gabriel Frumusanu

Received: 4 March 2023

Revised: 28 March 2023

Accepted: 29 March 2023

Published: 3 April 2023



Copyright: © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

Keywords: fault-detection filter; Markovian jump linear system; \mathcal{H}_∞ norm; LMI relaxations; nonhomogeneous Markov chains

MSC: 93E10

1. Introduction

In order to keep a manufacturing process as lean as possible, there are several aspects that must be considered. The monitoring capability is one of the features receiving the major spotlight in industrial operations since it is crucial to guarantee that the process is safe for the personnel involved. Among the procedures that constitute the monitoring systems, one that is worth mentioning is the fault-detection (FD) process [1,2].

A fault can be seen as the first indication of more harsh problems. It is any type of unwanted minor behavior that was not expected from the system. It can be caused, for instance, by extended wear due to long periods of time without maintenance. As a consequence of inadequately fixed wear, malfunctions or failures can cause a breakage [3].

In this sense, FD is a model-based process in which any abnormal/unexpected behavior is detected by a two-step procedure. The first step in the FD process is the residue generation, performed by an observer. The second step is the evaluation process, where the residue signal, generated by the observer, is treated by an evaluation function and compared with a predetermined threshold. We assume that a fault has occurred if the evaluation function surpasses the threshold; otherwise, we consider that the system is working as intended [4].

Currently, an important assumption that must be taken into account in FD systems is that the communication between the components is made via semireliable networks, which are associated with occasional packet dropout. These dropouts are caused by different sources as package collision due to high network congestion [5]. The distinction between a dropout and a fault is an essential aspect of the FD process since it makes easier to locate a fault. A viable way to model a packet dropout in a network is to employ a Markov jump linear system (MJLS) framework. This representation is appropriate to handle systems whose dynamic behavior is subject to random abrupt changes, like those caused by network packet dropouts. In this scenario, a Markov chain (MC) is used to model the jumping between the modes of operation of the system [6].

In the literature, there are plenty of examples of FD approaches that consider network behavior in their design. For instance, in [7,8] linear matrix inequality (LMI)-based constraints are provided to design fault detection filters (FDF) by using the \mathcal{H}_∞ norm as a performance index. In [9], the authors have developed an FD approach for underactuated manipulators modeled by MJLS. In [10], an FD method for networked control systems (NCS) under the assumption of the existence of a variable delay between the signals received by the system components is tackled. In [11], a fault detection filter under the MJLS formulation was applied to a control moment gyroscope. In [12], a fault-detection filtering problem is tackled under the Markov switching memristive neural networks. In [13], an observed-based sliding mode control problem based on the event-triggered protocol under the Markovian jump systems framework was presented. In [14], a fault-detection filter for discrete time Markov jump Lur'e systems with bounded sector condition. Observing all the above examples, one fundamental premise in the MJLS context is that the Markov chain (MC) is considered to be homogeneous [15], which means that it does not vary in time. However, since the packet dropout sources (collision, congestion, networked-induced delay) change in time, we consider that a fixed transition probability between the Markovian operation modes does not properly model the network behavior. A way to handle the particularity of a time-varying MC was presented in [16], where the author has proposed new LMI constraints to evaluate the stability of MJLS governed by a nonhomogeneous MC. A particular case of the proposal presented by [16], which allows for designing FDF for MJLS systems affected by nonhomogeneous MCs, consists of using a linear parameter varying (LPV)-based representation for the time-varying transition probability matrix [17,18]. There are several works in the literature that deal with the problem of control (or filter) synthesis for nonlinear systems by using different approaches. For example, regarding the design of fault-tolerant controllers, there are strategies based on fuzzy systems [19] capable of modeling system nonlinearities by using Takagi–Sugeno models, so that if the probability of actuator failure is small, the control mode is normal, and if the probability is high, the control is changed to fault-tolerant mode. Another strategy to deal with nonlinearities that can be found in the literature arises in the context of sliding mode control [20]. In this case, the class of discrete-time nonlinear systems with delays and uncertainties that is considered is the conic type, where the nonlinear terms satisfy the constraint that lies in a known hypersphere with an uncertain center. However, the proposed approach, in addition to considering the loss of packets in the communication network via the Markov chain, deals with the nonlinearity of the systems by using a different strategy from those previously discussed, in which the modes of operation are considered linear but depend on time-varying parameters. Such modeling allows the use of convex optimization methods and LMI-based tools to solve the filtering problem without adding extra levels of complexity.

In view of the above works, the main contribution of the present work are

- the proposition of a new design technique of gain-scheduled FDF for MJLS with nonhomogeneous MC, and
- the numerical simulation to reinforce the usability of the proposed theoretical solution.

The proposed approach describes the nonhomogeneous MC using linear time-varying parameters to model those variations, assuming that these parameters are known or at

least measurable. Another important assumption made is that the probability varies in time arbitrarily. Hence, the probability parameter for the following instant $(k + 1)$ does not depend on present instant k , which grants the ability to disassociate the Lyapunov function in two distinct simplexes. Based on this assumption, we propose the design of a gain-scheduled fault-detection filter where the scheduling parameter implemented is the one that dictates the variation of the MC. One advantage of the proposed approach, when compared with others found in the literature, is that the design conditions assure the system stability for the entire parameter-varying range since the FDF is scheduled in terms of time-varying parameters modeling the network variation. The major novelty of the proposed technique is the higher level of fidelity in the representation of the network influence in the system model. Since FD is a model-based approach, a more accurate representation of the system can lead to better performance in practice.

The paper is organized as follows. Sections 2 and 4 present the necessary theoretical fundamentals. Section 3 shows how to model the nonhomogeneous Markov Chain by using LPV. Section 5 introduces the problem formulation and the main contributions. Section 6 illustrates the feasibility of applying the proposed technique, by means of a numerical simulation, and Section 7 concludes the paper with some final remarks.

Notation

The real Euclidean space is denoted by \mathbb{R}^n where n represents its dimension and a real matrix with n rows and m columns is represented by $\mathbb{R}^{n \times m}$. The symbol I_n stands for an $n \times n$ identity matrix (or, for simplicity, just I , with an appropriate dimension, whenever no confusion arises) and the symbol $(\cdot)'$ denotes the transpose of a matrix. The operator $\text{Her}(\cdot)$ is used to express the symmetric sum as in $\text{Her}(X) = X + X'$, while the operator $\text{diag}(\cdot)$ represents a diagonal matrix. The symbol \bullet denotes a symmetric block in a partitioned symmetric matrix. The expected value operator is represented by $\mathbb{E}(\cdot)$ and the conditional expected operator is denoted by $\mathbb{E}(\cdot|\cdot)$. The fundamental probability space is described by $(\Omega, \mathcal{F}, \{\mathcal{F}_k\}, \text{Pr}(\cdot))$. The space \mathcal{L}^2 is the Hilbert space formed by \mathcal{F}_k -measurable random sequences $\{z_k\}_{k=0}^\infty$ such that $\|z\|_2 \triangleq [\sum_{k=0}^\infty \mathbb{E}\{|z(k)|^2\}]^{1/2} < \infty$.

2. Preliminaries

A generic discrete-time MJLS is given by

$$\mathcal{G} \equiv \begin{cases} x(k+1) = A_{\theta_k}x(k) + J_{\theta_k}w(k) \\ z(k) = C_{\theta_k}x(k) + D_{\theta_k}w(k)' \end{cases} \tag{1}$$

where $x(k) \in \mathbb{R}^{n_x}$ is for the state vector, $w(k) \in \mathbb{R}^{n_w}$ is the exogenous input vector, and $z(k) \in \mathbb{R}^{n_z}$ is the output signal. The state-space matrices of system (1) depend on the index θ_k , which represents a discrete-time Markov chain belonging to a finite set of modes $\mathbb{K} = \{1, \dots, \sigma\}$, whose switching is ruled by a time-varying transition probability matrix

$$\mathbb{P}(k) = \begin{bmatrix} \rho_{11}(k) & \cdots & \rho_{1\sigma}(k) \\ & \ddots & \\ \rho_{\sigma 1}(k) & \cdots & \rho_{\sigma\sigma}(k) \end{bmatrix}. \tag{2}$$

The entries $\rho_{ij}(k)$ of $\mathbb{P}(k)$ are such that $\rho_{ij}(k) = \text{Pr}(\theta_{k+1} = j | \theta_k = i)$, $\forall k \geq 0, \rho_{ij}(k) \geq 0$, and $\sum_{j=1}^\sigma \rho_{ij}(k) = 1$. We recall that whenever the transition matrix is time-invariant, that is, $\mathbb{P}(k) = \mathbb{P}$, the associated Markov chain is said to be homogeneous; otherwise, it is called nonhomogeneous (meaning that the probabilities vary in time) [15,21]. It is assumed that $\rho_{ij}(k)$ varies within the following interval: $0 \leq \underline{\rho}_{ij} \leq \rho_{ij}(k) \leq \bar{\rho}_{ij} \leq 1$, where $\underline{\rho}_{ij}$ represents lower bound and $\bar{\rho}_{ij}$ denotes the upper bound. Another important assumption is that the upper and lower bounds of the transition probability are known, and the transition probability variation is instantly measurable. Therefore, all the parameters in (2) may

vary in a known range with $\rho_{ij}(k) \in [\underline{\rho}_{ij}, \bar{\rho}_{ij}]$. There are several ways to determine the values of the upper ($\bar{\rho}_{ij}$) and lower ($\underline{\rho}_{ij}$) bounds of $\rho_{ij}(k)$. Those values can be obtained via mathematical modeling, observation, estimation, simulation, or based on the a priori knowledge of the system, such that the estimate can vary among systems and depends on the type of variation to which the system is subjected.

From the constraints $\sum_{j=1}^{\sigma} \rho_{ij}(k) = 1$ and $0 \leq \underline{\rho}_{ij} \leq \rho_{ij}(k) \leq \bar{\rho}_{ij} \leq 1$, the transition matrix (2) can be described by N polytopic intervals, where N depends on the number of transition probabilities that are time-varying. From these polytopic intervals, some techniques can be applied to obtain a gain-scheduled FDF.

In order to exemplify these N polytopic intervals and how to define a time-varying transition matrix by using LPV, let us assume that $\sigma = 5$ and the parameters $\rho_{14}(k)$ and $\rho_{15}(k)$ vary in time; hence, the first row of the transition matrix (2) can be written as

$$[\rho_{11} \quad \rho_{12} \quad \rho_{13} \quad \rho_{14}(k) \quad \rho_{15}(k)], \tag{3}$$

and from this row, two polytopic intervals ($N = 2$) are obtained:

$$[\rho_{11} \quad \rho_{12} \quad \rho_{13} \quad \underline{\rho}_{14} \quad \bar{\rho}_{15}], \quad [\rho_{11} \quad \rho_{12} \quad \rho_{13} \quad \bar{\rho}_{14} \quad \underline{\rho}_{15}]. \tag{4}$$

The polytopic intervals obey the constraints $\sum_{j=1}^{\sigma} \rho_{ij}(k) = 1$ and $0 \leq \underline{\rho}_{ij} \leq \rho_{ij}(k) \leq \bar{\rho}_{ij} \leq 1$ simultaneously. The following notation will be used to represent a time-varying row of $\mathbb{P}(k)$ as in (3) with the lower and upper bounds as in (4):

$$[\rho_{11} \quad \rho_{12} \quad \rho_{13} \quad [\underline{\rho}_{14}, \bar{\rho}_{14}] \quad [\underline{\rho}_{15}, \bar{\rho}_{15}]]. \tag{5}$$

The main novelty in this paper is the usage of the same time-varying parameters that coordinate the nonhomogeneous MC variation as gain-scheduled parameters for the design and implementation of the FDF. This concept will be carefully described in Section 3.

Although the time variation that affects the probability matrix is generally represented by modeling $\mathbb{P}(k)$ as belonging to a polytope, in this paper we choose to use another approach, which describes each time-varying row of $\mathbb{P}(k)$ in terms of a linear time-varying parameter vector $\alpha_r(k)$ belonging to a distinct unit simplex Λ_{N_r} , $r = 1, 2, \dots, m$. The definition of the unit simplex is given by

$$\Lambda_{N_r} \equiv \left\{ \zeta \in \mathbb{R}^{N_r} \mid \sum_{i=1}^{N_r} \zeta_i = 1, \zeta_i \geq 0, i = 0, 1, \dots, N_r \right\}, \tag{6}$$

where m is the number of time-varying rows in the probability matrix. In order to group up all the time-varying parameters of $\mathbb{P}(k)$ in a single domain, we perform a Cartesian product of m simplexes, each one of dimension N_r , in a single domain called multisimplex, and represent it by $\Lambda_N = \Lambda_{N_1} \times \Lambda_{N_2} \times \dots \times \Lambda_{N_m}$, with the index N given by $N = (N_1, \dots, N_m)$. For ease of notation \mathbb{R}^N represents the space $\mathbb{R}^{N_1+N_2+\dots+N_m}$. In this sense, a given element $\alpha(k) \in \Lambda_N$ is a vector belonging to \mathbb{R}^N and can be decomposed as $(\alpha_1(k), \alpha_2(k), \dots, \alpha_m(k))$ according to the structure of Λ_N . Subsequently, each $\alpha_r(k) \in \Lambda_r \subset \mathbb{R}^{N_r}$, $r = 1, \dots, m$, is decomposed in the form $(\alpha_{r1}, \alpha_{r2}, \dots, \alpha_{rN_r})$. This approach follows the one adopted in [22].

Hereafter, the transition probability will be denoted by $\rho_{ij}(\alpha(k))$, where the term $\alpha(k) \in \Lambda_N$ represents the time-varying parameter responsible to model the probability of the nonhomogeneous Markov chain at time k .

3. Modeling the Nonhomogeneous Markov Chain by Using the Linear Parameter Varying Approach

We next present the definition of a matrix $O_i(\iota_k)$, where i represents the MC mode, and $\iota_k = (\iota_1(k), \dots, \iota_m(k))$ denotes a generic LPV parameter. It is assumed that $O_i(\iota_k)$ is affinely dependent on the time-varying parameters $\iota_j(k)$, as described below:

$$O_i(\iota_k) = O_{i_{i_0}} + \sum_{j=1}^m \iota_j(k) O_{i_{i_j}}. \tag{7}$$

The matrix in the affine form (7) can be interpreted in the following manner: matrix $O_{i_{i_0}}$ represents the time-invariant part of the filter dynamics. The remaining matrices $O_{i_{i_j}}, j = 1, \dots, m$ denote the time-varying dynamic that depends on the parameters $\iota_j(k)$. To illustrate this particular structure, consider the example presented below, for an MJLS with three operation modes, whose time-varying probability matrix is given by

$$\mathbb{P}(k) = \begin{bmatrix} 0.5 & [0.1, 0.3] & [0.2, 0.4] \\ [0, 0.4] & [0.5, 0.9] & 0.1 \\ 0.2 & 0.6 & 0.2 \end{bmatrix}, \tag{8}$$

where elements $\rho_{12}(k), \rho_{13}(k), \rho_{21}(k)$ and $\rho_{22}(k)$ vary in a known interval $[\underline{\rho}_{ij}, \bar{\rho}_{ij}]$. Since each uncertain row of $\mathbb{P}(k)$ can be represented by a polytopic interval, the representation of the first row is given by

$$[0.5 \ 0.1 \ 0.4] \alpha_{11}(k) + [0.5 \ 0.3 \ 0.2] \alpha_{12}(k) \tag{9}$$

and the second row is

$$[0 \ 0.9 \ 0.1] \alpha_{21}(k) + [0.4 \ 0.5 \ 0.1] \alpha_{22}(k) \tag{10}$$

with $\alpha_1(k) = (\alpha_{11}(k), \alpha_{12}(k)) \in \Lambda_2, \alpha_2(k) = (\alpha_{21}(k), \alpha_{22}(k)) \in \Lambda_2$, and $\alpha(k) = (\alpha_1(k), \alpha_2(k)) \in \Lambda_2 \times \Lambda_2$. On the other hand, the representation of $\mathbb{P}(k)$, in terms of parameter $\iota(k)$ used in the affine structure, can be done as follows,

$$\mathbb{P}(k) = \underbrace{\begin{bmatrix} 0.5 & 0.2 & 0.3 \\ 0.2 & 0.7 & 0.1 \\ 0.2 & 0.6 & 0.2 \end{bmatrix}}_{\mathbb{P}_{i_0}} + \underbrace{\begin{bmatrix} 0 & 1 & -1 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}}_{\mathbb{P}_{i_1}} \iota_1(k) + \underbrace{\begin{bmatrix} 0 & 0 & 0 \\ 1 & -1 & 0 \\ 0 & 0 & 0 \end{bmatrix}}_{\mathbb{P}_{i_2}} \iota_2(k), \tag{11}$$

where $\iota_1(k) \in [-0.1, 0.1]$ and $\iota_2(k) \in [-0.2, 0.2]$. Although the modeling seems to be different, note that a simple change of variables can recover the multisimplex modeling from the affine representation, since

$$\iota_r(k) = \underline{\iota}_r \alpha_{r1}(k) + \bar{\iota}_r \alpha_{r2}(k), \tag{12}$$

where $\iota_r(k) \in [\underline{\iota}_r, \bar{\iota}_r], \alpha_r(k) = (\alpha_{r1}(k), \alpha_{r2}(k)) \in \Lambda_2, r = 1, 2$.

In order to clarify how to write a time-varying matrix in the affine form, consider the following affine matrix as

$$\underbrace{\begin{bmatrix} 5 & 0.3 + \iota_1(k) \\ 12 & -2 + 0.5 \iota_2(k) \end{bmatrix}}_{O_i(\iota_j)} = \underbrace{\begin{bmatrix} 5 & 0.3 \\ 12 & -2 \end{bmatrix}}_{O_{i_{i_0}}} + \underbrace{\begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix}}_{O_{i_{i_1}}} \iota_1(k) + \underbrace{\begin{bmatrix} 0 & 0 \\ 0 & 0.5 \end{bmatrix}}_{O_{i_{i_2}}} \iota_2(k), \tag{13}$$

where $\iota_1(k) \in [\underline{\iota}_1, \bar{\iota}_1]$ and $\iota_2(k) \in [\underline{\iota}_2, \bar{\iota}_2]$. By using the multisimplex formulation, $\iota_r(k) = \underline{\iota}_r \alpha_{r1}(k) + \bar{\iota}_r \alpha_{r2}(k)$ for $r = 1, 2$, we recover the representation with $\alpha(k) \in \Lambda_N = \Lambda_2 \times \Lambda_2$, where $N = (2, 2)$. This procedure can be extended for all matrices throughout this paper. Bearing this in mind, in what follows, whenever we write $P_i(\alpha(k))$ for $\alpha(k) \in \Lambda_N$, we mean

a representation, as in (7), in terms of generic LPV parameters or, equivalently, in terms the multisimplex parameter $\alpha(k)$.

4. Bounded Real Lemma

The concept of stability for the nonhomogeneous Markov chain is different from its homogeneous counterpart. This discrepancy is caused by the arbitrary variation of the transition probability. Therefore, an upper bound of the \mathcal{H}_∞ norm can only be obtained if system (1) under the assumption of $w(k) \equiv 0$ is exponentially stable in the mean square sense with conditioning of type I (ESMS-CI). This concept was first introduced in [23] and is also presented in [16]. In this sense, before introducing the main results of this paper, some fundamental definitions are presented next.

Definition 1 ([16]). *Assuming that system (1) is ESMS-CI, and $x(0) = 0$, its \mathcal{H}_∞ norm is given by*

$$\|\mathcal{G}\|_\infty = \sup_{w \in \mathcal{L}, \|w\|_2 \neq 0} \frac{\|z\|_2}{\|w\|_2}. \tag{14}$$

Next, we present a sufficient condition version of the bounded real lemma adapted from [16], which allows us to deal with nonhomogeneous MJLS with arbitrarily fast time-varying parameters, where the parameters are modeled by using the multisimplex domain Λ_N . For that, it is assumed that the condition H_1 in Proposition 1 in [16] is satisfied; that is, $\Pr(\theta_k = i) > 0$ for all $i \in \mathbb{K}$ and $k \geq 0$.

Remark 1. *In order to draw the results presented in Lemma 1, it is necessary to consider the assumption that the variation of the probabilities $\rho_{ij}(k)$ is arbitrarily fast. Under this assumption, there is no need to bound the variation limit.*

Lemma 1. *System (1) is ESMS-CI and satisfies $\|\mathcal{G}\|_\infty < \gamma$ if there exist symmetric positive definite matrices $P_i(\alpha(k))$, such that, for each $i \in \mathbb{K}$ and for all $\alpha(k), \alpha(k + 1) \in \Lambda_N$, the parameter-dependent LMIs*

$$\underbrace{\begin{bmatrix} A_i & J_i \\ C_i & D_i \end{bmatrix}' \begin{bmatrix} \mathbb{E}_i(P)(\alpha(k), \alpha(k+1)) & 0 \\ 0 & I \end{bmatrix} \begin{bmatrix} A_i & J_i \\ C_i & D_i \end{bmatrix} - \begin{bmatrix} P_i(\alpha(k)) & 0 \\ 0 & \gamma^2 I \end{bmatrix}}_{\Omega_i(\alpha(k))} < 0 \tag{15}$$

are satisfied, where $\mathbb{E}_i(P)(\alpha(k), \alpha(k + 1)) = \sum_{j=1}^\sigma \rho_{ij}(\alpha(k)) P_j(\alpha(k + 1))$.

Proof. Here is a sketch of the proof for Lemma 1. Assuming that there exist $P_i(\alpha(k)) = P'_i(\alpha(k))$ such that Equation (15) holds, we have, from Proposition 1 in [16], that system (1) is ESMSC1. Define the cost function as

$$\mathcal{J}_\tau^\gamma = \sum_{k=0}^\tau E \left[z(k)'z(k) - \gamma^2 w(k)'w(k) \right]. \tag{16}$$

Observe that $\|\mathcal{G}\|_\infty < \gamma \iff \mathcal{J}_\infty^\gamma \leq -e^2 \|w\|_2^2, \forall \|w\|_2 \neq 0$ and for some $e \neq 0$, where $\mathcal{J}_\infty^\gamma$ represent the cost function for $\tau \rightarrow \infty$. Considering the Lyapunov function $\mathcal{V}_{\theta_k}(k, x(k)) \triangleq x(k)'P_{\theta_k}(\alpha(k))x(k)$, one has

$$\begin{aligned} \mathcal{J}_\tau^\gamma &= \sum_{k=0}^\tau E \left[z(k)'z(k) - \gamma^2 w(k)'w(k) - \mathcal{V}_{\theta_k}(k, x(k)) + \mathcal{V}_{\theta_{k+1}}(k + 1, x(k + 1)) \right] \\ &\quad + \sum_{k=0}^\tau E \left[\mathcal{V}_{\theta_k}(k, x(k)) - \mathcal{V}_{\theta_{k+1}}(k + 1, x(k + 1)) \right] \end{aligned}$$

$$\begin{aligned}
 &= \sum_{k=0}^{\tau} E \left[z(k)'z(k) - \gamma^2 w(k)'w(k) - \mathcal{V}_{\theta_k}(k, x(k)) \right] \\
 &\quad + \sum_{k=0}^{\tau} E \left[E \left[\mathcal{V}_{\theta_{k+1}}(k+1, x(k+1)) | \mathcal{F}_k \right] \right. \\
 &\quad \left. - E \left[\mathcal{V}_{\tau+1}(\tau+1, x(\tau+1)) \right] + E \left[\mathcal{V}_{\theta_0}(0, x(0)) \right] \right] \\
 &= \sum_{k=0}^{\tau} E \left[z(k)'z(k) - \gamma^2 w(k)'w(k) - \mathcal{V}_{\theta_k}(k, x(k)) \right] \\
 &\quad + \sum_{k=0}^{\tau} E \left[x(k+1)' E \left[P_{\theta_{k+1}}(\alpha(k+1)) | \mathcal{F}_k \right] x(k+1) \right. \\
 &\quad \left. - E \left[\mathcal{V}_{\tau+1}(\tau+1, x(\tau+1)) \right] + E \left[\mathcal{V}_{\theta_0}(0, x(0)) \right] \right] \\
 &= \sum_{k=0}^{\tau} E \left[[x(k)' w(k)'] \Omega_{\theta_k}(\alpha(k)) [x(k)' w(k)']' \right. \\
 &\quad \left. - E \left[\mathcal{V}_{\tau+1}(\tau+1, x(\tau+1)) \right] + E \left[\mathcal{V}_{\theta_0}(0, x(0)) \right] \right],
 \end{aligned}$$

where $\Omega_i(\alpha(k))$ is presented in Equation (15). Recalling that $x(0) = 0$ so that $\mathcal{V}_{\theta_0}(0, x(0)) = 0$, we have $\forall k \geq 0$ and some $e \neq 0$ that

$$\begin{aligned}
 \mathcal{J}_{\tau}^{\gamma} &= \sum_{k=0}^{\tau} E \left[[x(k)' w(k)'] \Omega_{\theta_k}(\alpha(k)) \bullet \right] - E \left[\mathcal{V}_{\tau+1}(\tau+1, x(\tau+1)) \right] \\
 &\leq \sum_{k=0}^{\tau} E \left[[x(k)' w(k)'] \Omega_{\theta_k}(\alpha(k)) \bullet \right] \leq - \sum_{k=0}^{\tau} e^2 E \left[\|w(k)\|^2 \right]. \tag{17}
 \end{aligned}$$

Inequality (17) yields, as $\tau \rightarrow \infty$, that $\mathcal{J}_{\infty}^{\gamma} \leq -e^2 \|w\|_2^2 \forall \|w\|_2 \neq 0$, showing the desired result. \square

5. Problem Formulation and Main Result

The block diagram of Figure 1 illustrates the FD scheme considered in this paper. Note that there are three elements composing the diagram: the system itself (\mathcal{G}_{θ_k} , which represents the plant subjected to a fault), the controller K_{θ_k} , and the gain-scheduled FDF block \mathcal{F}_{θ_k} .

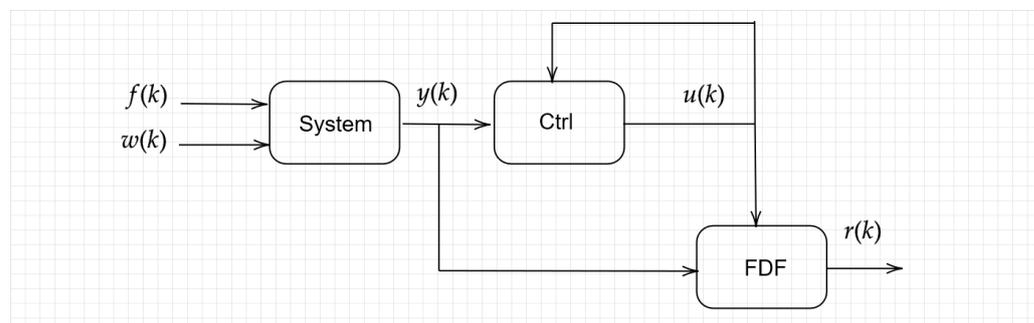


Figure 1. Graphic description of the FD scheme used to design the gain-scheduled fault detection filter.

The main purpose of this section is to design the FDF. Consider that the nonhomogeneous MJLS \mathcal{G}_{θ_k} is defined as

$$\mathcal{G}_{\theta_k} \equiv \begin{cases} x(k+1) = A_{\theta_k}x(k) + B_{\theta_k}u(k) + J_{\theta_k}w(k) + F_{\theta_k}f(k) \\ y(k) = C_{\theta_k}x(k) + D_{\theta_k}w(k) + E_{\theta_k}f(k) \end{cases}, \tag{18}$$

where $x(k) \in \mathbb{R}^{n_x}$ denotes the system states vector, $u(k) \in \mathbb{R}^{n_u}$ represents the control input, $w(k) \in \mathbb{R}^{n_w}$ is the exogenous/noise input, $y(k) \in \mathbb{R}^{n_y}$ represents the measurement signals, and $f(k) \in \mathbb{R}^{n_f}$ is the fault signal that should be detected. We also assume that $w(k), f(k) \in \mathcal{L}^2$. Finally, when regarding the output-feedback control law, consider the following expression

$$u(k) = K_{\theta_k}y(k). \tag{19}$$

The controller K_{θ_k} is assumed to be designed beforehand.

Remark 2. Although the formulation presented in this paper considers an output-feedback control law (19) that is mode-dependent (depends on θ_k) but is parameter-independent (does not depend on α_k or the time-varying probabilities $\rho_{ij}(k)$), the synthesis of the FD filter presented next can be extended to deal with a parameter-dependent control law. Regardless of that, for both situations, it is imperative that the controller is designed a priori. The major implementation difference in the latter case is that the controller would be defined as gain-scheduled.

We define the gain-scheduled FDF under the aforementioned conditions as

$$\mathcal{F}_{\theta_k} \equiv \begin{cases} \eta(k+1) = \mathcal{A}_{\eta\theta_k}(\alpha(k))\eta(k) + \mathcal{M}_{\eta\theta_k}(\alpha(k))u(k) + \dots \\ \quad \mathcal{B}_{\eta\theta_k}(\alpha(k))y(k) \\ r(k) = \mathcal{C}_{\eta\theta_k}(\alpha(k))\eta(k) \end{cases}, \tag{20}$$

where $\eta(k) \in \mathbb{R}^{n_\eta}$ is the filter states, $r(k) \in \mathbb{R}^{n_r}$ is the residue signal, $u(k)$ is the control law given by (19), and $y(k)$ represents the measurement output. The FDF is scheduled in terms of the time-varying parameter $\alpha(k)$, which represents the variation in time of the MC. Consequently, it is assumed that the time-varying behavior of the MC is known or at least measurable. The purpose of the gain-scheduled FDF in (20) is to generate a residue signal $r(k)$, which is used to detect the fault.

Remark 3. All matrices that compose the filter (20) are written in the affine form as in (7), that is

$$\mathcal{A}_{\eta i}(\beta_k) = \mathcal{A}_{\eta i \beta_0} + \sum_{j=1}^m \beta_j(k)\mathcal{A}_{\eta i \beta_j}, \tag{21}$$

and similarly for $\mathcal{M}_{\eta i}(\beta_k), \mathcal{B}_{\eta i}(\beta_k), \mathcal{C}_{\eta i}(\beta_k)$. Recall that the goal of this paper is to design the gain-scheduled FDF (20) where the schedule parameter represents the variation in the nonhomogeneous MC, as explained in Section 3.

We define $e(k) = r(k) - f(k)$ and the augmented system as

$$\mathcal{G}_{aug} \equiv \begin{cases} \tilde{x}(k+1) = \tilde{A}_{\theta_k}(\alpha(k))\tilde{x}(k) + \tilde{J}_{\theta_k}(\alpha(k))\tilde{w}(k) \\ e(k) = \tilde{C}_{\theta_k}(\alpha(k))\tilde{x}(k) + \tilde{D}_{\theta_k}(\alpha(k))\tilde{w}(k) \end{cases}, \tag{22}$$

where $\tilde{x}(k) = [x(k) \ \eta(k)]$, and $\tilde{w}(k) = [u(k) \ w(k) \ f(k)]$. The matrices that compose the augmented system are

$$\tilde{A}_i(\alpha(k)) = \begin{bmatrix} A_i & 0 \\ \mathcal{B}_{\eta i}(\alpha(k))C_i & \mathcal{A}_{\eta i}(\alpha(k)) \end{bmatrix},$$

$$\begin{aligned} \tilde{J}_i(\alpha(k)) &= \begin{bmatrix} B_i & J_i & F_i \\ \mathcal{M}_{\eta_i}(\alpha(k)) & \mathcal{B}_{\eta_i}(\alpha(k))D_i & \mathcal{B}_{\eta_i}(\alpha(k))E_i \end{bmatrix}, \text{ and} \\ \tilde{C}_i(\alpha(k)) &= [0 \ C_{\eta_i}(\alpha(k))], \quad \tilde{D}_i(\alpha(k)) = [0 \ 0 \ -I]. \end{aligned} \tag{23}$$

In order to provide an FDF (20), which generates a residue signal $r(k)$ that is robust against noise and associated with a fast response whenever a fault occurs, we set our goal as the synthesis of a proper FD filter by minimizing an upper bound γ for the \mathcal{H}_∞ norm of the augmented system (22), such that

$$\sup_{w(k) \in \mathcal{L}^2, \|w(k)\|_2 \neq 0} \frac{\|e(k)\|_2}{\|w(k)\|_2} < \gamma. \tag{24}$$

In the following, we present our main result of the paper, which provides parameter-dependent bilinear matrix inequalities (BMI) for the design of an FDF (20) for system (18) with \mathcal{H}_∞ guaranteed cost. For compactness, hereafter, the dependence on time of parameter $\alpha(k)$ and $\alpha(k + 1)$ is omitted, such that they will be respectively replaced by α and α^+ .

Theorem 1. *If there exist, for all $i \in \mathbb{K}$, symmetric positive definite parameter-dependent matrices $Z_i(\alpha)$, $R_i(\alpha)$, $\bar{Z}_1(\alpha^+)$, $\bar{R}_i(\alpha^+)$, matrices $W_i(\alpha)$, $\bar{W}_i(\alpha^+)$, $X_i(\alpha)$, $Y_i(\alpha)$, $\mathcal{O}_i(\alpha)$, $\mathcal{B}_{\eta_i}(\alpha)$, $\mathcal{M}_{\eta_i}(\alpha)$, $\nabla_i(\alpha)$ with appropriate dimensions, and a scalar $\xi \in (0, 2)$, such that the BMIs (25) hold for all $\alpha, \alpha^+ \in \Lambda_N$ and $i \in \mathbb{K}$,*

$$\begin{bmatrix} \Pi_{11} & \bullet \\ \Pi_{21} & \Pi_{22} & \bullet \\ 0 & 0 & -\gamma^2 I & \bullet \\ 0 & 0 & 0 & -\gamma^2 I & \bullet & \bullet & \bullet & \bullet & \bullet & \bullet \\ 0 & 0 & 0 & 0 & -\gamma^2 I & \bullet & \bullet & \bullet & \bullet & \bullet \\ \Pi_{61} & \Pi_{62} & 0 & 0 & 0 & \Pi_{66} & \bullet & \bullet & \bullet & \bullet \\ \Pi_{71} & \Pi_{72} & 0 & 0 & 0 & \Pi_{76} & \Pi_{77} & \bullet & \bullet & \bullet \\ \Pi_{81} & \Pi_{82} & Y_i^a(\alpha)'(B_i) & Y_i^a(\alpha)'(J_i) & Y_i^a(\alpha)'(F_i) & \xi \Pi_{81} & \xi \Pi_{82} & -\bar{Z}(\alpha^+) & \bullet & \bullet \\ \Pi_{91} & \Pi_{92} & Y_i^b(\alpha)'\mathcal{M}_{\eta_i}(\alpha) & \Pi_{94} & \Pi_{95} & \xi \Pi_{91} & \xi \Pi_{92} & -\bar{W}(\alpha^+) & -\bar{R}(\alpha^+) & \bullet \\ \nabla_i(\alpha) & \nabla_i(\alpha) & 0 & 0 & -I & \xi \nabla_i(\alpha) & \xi \nabla_i(\alpha) & 0 & 0 & -I \end{bmatrix} < 0, \tag{25}$$

where

$$\begin{aligned} Y_i^a(\alpha) &= [\rho_{i1}^{1/2}(\alpha)I_{n_x} \ \dots \ \rho_{i\sigma}^{1/2}(\alpha)I_{n_x}], \quad Y_i^b(\alpha) = [\rho_{i1}^{1/2}(\alpha)I_{n_\eta} \ \dots \ \rho_{i\sigma}^{1/2}(\alpha)I_{n_\eta}], \\ Y_i^c(\alpha) &= [\rho_{i1}^{1/2}(\alpha)I_{n_x+n_\eta} \ \dots \ \rho_{i\sigma}^{1/2}(\alpha)I_{n_x+n_\eta}], \quad \Pi_{11} = Z_i(\alpha) - Her(X_i(\alpha)), \\ \Pi_{21} &= W_i(\alpha) - Y_i(\alpha) - X_i(\alpha)', \quad \Pi_{61} = Z_i(\alpha) - \xi X_i(\alpha)' - X_i(\alpha), \\ \Pi_{71} &= W_i(\alpha) - \xi X_i(\alpha)' - Y_i(\alpha), \quad \Pi_{81} = Y_i^a(\alpha)' A_i X_i(\alpha), \\ \Pi_{91} &= Y_i^b(\alpha)'(\mathcal{B}_{\eta_i}(\alpha)C_i X_i(\alpha) + \mathcal{O}_i(\alpha)), \quad \Pi_{22} = R_i(\alpha) - Her(Y_i(\alpha)), \\ \Pi_{62} &= W_i(\alpha)' - \xi Y_i(\alpha)' - X_i(\alpha), \quad \Pi_{72} = R_i(\alpha) - \xi Y_i(\alpha)' - Y_i(\alpha), \\ \Pi_{82} &= \Pi_{81}, \quad \Pi_{92} = \Pi_{91}, \quad \Pi_{94} = Y_i^b(\alpha)'\mathcal{B}_{\eta_i}(\alpha)D_i, \\ \Pi_{95} &= Y_i^b(\alpha)'\mathcal{B}_{\eta_i}(\alpha)E_i, \quad \Pi_{66} = -\xi Her(X_i(\alpha)), \\ \Pi_{76} &= -\xi(Y_i(\alpha) - X_i(\alpha)'), \quad \Pi_{77} = -\xi Her(Y_i(\alpha)), \end{aligned}$$

and

$$\begin{aligned} \bar{Z}(\alpha^+) &= \text{diag}(\bar{Z}_1(\alpha^+), \dots, \bar{Z}_\sigma(\alpha^+)), \quad \bar{W}^+(\alpha^+) = \text{diag}(\bar{W}_1^+(\alpha^+), \dots, \bar{W}_\sigma(\alpha^+)), \\ \bar{R}_i(\alpha^+) &= \text{diag}(\bar{R}_1(\alpha^+), \dots, \bar{R}_\sigma(\alpha^+)), \end{aligned} \tag{26}$$

then γ is an upper bound for the \mathcal{H}_∞ norm of the augmented system (22), where the matrices that compose the FDF in the form of (20) are given by $\mathcal{A}_{\eta_i}(\alpha) = \mathcal{O}_i(\alpha)Y_i(\alpha)^{-1}$, $\mathcal{B}_{\eta_i}(\alpha)$, $\mathcal{M}_{\eta_i}(\alpha)$, $\mathcal{C}_{\eta_i}(\alpha) = \nabla_i Y_i(\alpha)^{-1}(\alpha)$ for all $i \in \mathbb{K}$.

Proof of Theorem 1. Consider the augmented matrices as from (23), and define the matrices $P_i(\alpha)$ and $H_i(\alpha)$ according to

$$P_i(\alpha) = \begin{bmatrix} Z_i(\alpha) & \bullet \\ W_i(\alpha) & R_i(\alpha) \end{bmatrix}, \quad H_i(\alpha) = \begin{bmatrix} X_i(\alpha) & X_i(\alpha) \\ Y_i(\alpha) & Y_i(\alpha) \end{bmatrix}. \tag{27}$$

In addition, we define the matrices $\bar{P}_i(\alpha^+)$ and $\bar{P}(\alpha^+)$ as follows:

$$\bar{P}_i(\alpha^+) = \begin{bmatrix} \bar{Z}_i(\alpha^+) & \bullet \\ \bar{W}_i(\alpha^+) & \bar{R}_i(\alpha^+) \end{bmatrix}, \quad \bar{P}(\alpha^+) = \text{diag}(\bar{P}_i(\alpha^+), \dots, \bar{P}_\sigma(\alpha^+)). \tag{28}$$

Consider the following change of variables: $\mathcal{O}_i(\alpha) = \mathcal{A}_{\eta i}(\alpha)Y_i(\alpha)$, $\nabla_i(\alpha) = \mathcal{C}_{\eta i}(\alpha)Y_i(\alpha)$. Then we obtain the following identities:

$$\begin{aligned} \tilde{A}_i(\alpha)H_i(\alpha) &= \begin{bmatrix} A_i+B_iK_i(\alpha)C_i & 0 \\ \mathcal{M}_{\eta i}(\alpha)K_i(\alpha)C_i+\mathcal{B}_{\eta i}(\alpha)C_i & \mathcal{A}_{\eta i}(\alpha) \end{bmatrix} \begin{bmatrix} X_i(\alpha) & X_i(\alpha) \\ Y_i(\alpha) & Y_i(\alpha) \end{bmatrix} \\ &= \begin{bmatrix} (A_i+B_iK_i(\alpha)C_i)X_i(\alpha) & (A_i+B_iK_i(\alpha)C_i)X_i(\alpha) \\ \Xi_i(\alpha)+\mathcal{A}_{\eta i}(\alpha)Y_i(\alpha) & \Xi_i(\alpha)+\mathcal{A}_{\eta i}(\alpha)Y_i(\alpha) \end{bmatrix}, \\ \tilde{C}_i(\alpha)H_i(\alpha) &= \begin{bmatrix} 0 & \mathcal{C}_{\eta i}(\alpha) \end{bmatrix} \begin{bmatrix} X_i(\alpha) & X_i(\alpha) \\ Y_i(\alpha) & Y_i(\alpha) \end{bmatrix} \\ &= \begin{bmatrix} \mathcal{C}_{\eta i}(\alpha)Y_i(\alpha) & \mathcal{C}_{\eta i}(\alpha)Y_i(\alpha) \end{bmatrix}, \end{aligned} \tag{29}$$

where $\Xi_i(\alpha) = (\mathcal{M}_{\eta i}(\alpha)K_iC_i + \mathcal{B}_{\eta i}(\alpha)C_i)X_i(\alpha)$. Consequently, the BMI (25) can be rewritten as

$$\begin{bmatrix} P_i(\alpha)-\text{Her}(H_i(\alpha)) & \bullet & \bullet & \bullet & \bullet \\ 0 & -\gamma^2 I & \bullet & \bullet & \bullet \\ P_i(\alpha)-\xi H_i(\alpha)'-H_i(\alpha) & 0 & -\xi \text{Her}(H_i(\alpha)) & \bullet & \bullet \\ Y_i^c(\alpha)'\tilde{A}_i(\alpha)H_i(\alpha) & Y_i^c(\alpha)'\tilde{J}_i(\alpha) & \xi Y_i^c(\alpha)'\tilde{A}_i(\alpha)H_i(\alpha) & -\bar{P}(\alpha^+) & \bullet \\ \tilde{C}_i(\alpha)H_i(\alpha) & \tilde{D}_i(\alpha) & \xi \tilde{C}_i(\alpha)H_i(\alpha) & 0 & -I \end{bmatrix} < 0. \tag{30}$$

By using the projection lemma (see [24]), (30) can be rewritten as follows,

$$\mathcal{D} + \mathcal{U}'H_i\mathcal{V} + \mathcal{V}H_i'\mathcal{U} < 0, \tag{31}$$

where

$$\mathcal{D} = \begin{bmatrix} P_i & \bullet & \bullet & \bullet & \bullet \\ 0 & -\gamma^2 I & \bullet & \bullet & \bullet \\ P_i & 0 & 0 & \bullet & \bullet \\ 0 & Y_i^c(\alpha)'\tilde{J}_i & 0 & -\bar{P}^+ & \bullet \\ 0 & \tilde{D}_i & 0 & 0 & -I \end{bmatrix}, \quad \mathcal{U}' = \begin{bmatrix} -I \\ 0 \\ -I \\ Y_i^c(\alpha)'\tilde{A}_i \\ \tilde{C}_i \end{bmatrix}, \quad \mathcal{V}' = \begin{bmatrix} I \\ 0 \\ \xi I \\ 0 \\ 0 \end{bmatrix}. \tag{32}$$

(Observe that in the remaining of the proof, the time-varying parameter α is omitted for notation simplicity, as well as the dependence on α^+ , which will be replaced by the superscript index “+”.)

By taking the following basis for the null space of \mathcal{U} and \mathcal{V}

$$\mathcal{N}_{\mathcal{U}} = \begin{bmatrix} I & 0 & 0 & 0 \\ 0 & I & 0 & 0 \\ -I & 0 & \tilde{A}_i'Y_i^c & \tilde{C}_i' \\ 0 & 0 & I & 0 \\ 0 & 0 & 0 & I \end{bmatrix}, \quad \mathcal{N}_{\mathcal{V}} = \begin{bmatrix} \xi I & 0 & 0 & 0 \\ 0 & I & 0 & 0 \\ -I & 0 & 0 & 0 \\ 0 & 0 & I & 0 \\ 0 & 0 & 0 & I \end{bmatrix} \tag{33}$$

and by applying the equivalence conditions of the projection lemma, we get

$$N_{\mathcal{U}}'\mathcal{D}N_{\mathcal{U}} = \begin{bmatrix} -P_i & \bullet & \bullet & \bullet \\ 0 & -\gamma^2 I & \bullet & \bullet \\ Y_i^c(\alpha)'\tilde{A}_iP_i & Y_i^c(\alpha)'\tilde{J}_i & -\bar{P}^+ & \bullet \\ \tilde{C}_iP_i & \tilde{D}_i & 0 & -I \end{bmatrix} < 0, \tag{34}$$

$$N_{\mathcal{V}}'\mathcal{D}N_{\mathcal{V}} = \begin{bmatrix} (\xi^2-2\xi)P_i & \bullet & \bullet & \bullet \\ 0 & -\gamma^2 I & \bullet & \bullet \\ 0 & Y_i^c(\alpha)'\tilde{J}_i & -\bar{P}^+ & \bullet \\ 0 & \tilde{D}_i & 0 & -I \end{bmatrix} < 0. \tag{35}$$

Notice that from the first term in (35) we can state that $0 < \xi < 2$. Now, pre- and postmultiplying (34) by $\text{diag}(X_i, I, \bar{\mathcal{X}}^+, I)$, where $X_i = P_i^{-1}$, $\bar{\mathcal{X}}^+ = (\bar{P}^+)^{-1}$, we obtain

$$\begin{bmatrix} -X_i & \bullet & \bullet & \bullet \\ 0 & -\gamma^2 I & \bullet & \bullet \\ \bar{\mathcal{X}}^+ Y_i^c \tilde{A}_i & \bar{\mathcal{X}}^+ Y_i^c \tilde{J}_i & -\bar{\mathcal{X}}^+ & \bullet \\ \tilde{C}_i & \tilde{D}_i & 0 & -I \end{bmatrix} < 0. \tag{36}$$

Notice that $\hat{\mathcal{X}}^+$ represents a block diagonal matrix given by

$$\hat{\mathcal{X}}^+ = \text{diag}(\bar{X}_1^+, \dots, \bar{X}_\sigma^+), \tag{37}$$

with $\bar{X}_i^+ = (\bar{P}_i^+)^{-1}$. By using the Schur complement in (36) and noticing that

$$Y_i^c \bar{\mathcal{X}}^+ Y_i^{c'} = \sum_{j=1}^{\sigma} \rho_{ij} \bar{X}_j^+, \tag{38}$$

we have the condition (15) satisfied, so that the result follows from the bounded real lemma, presented in Lemma 1, for the nonhomogenous MJLS. \square

Remark 4. Observe that the conditions of Theorem 1 constitute of infinite-dimensional problems, which can be solved by using homogeneous polynomial approximations for the optimization variables (LMI relaxations) and then testing the positivity of the polynomial matrix inequalities by means of a finite set of LMIs. For this purpose, the authors strongly recommend the use of the toolbox robust LMI parser (ROLMIP), whose tutorial can be found in [25].

Remark 5. Theorem 1 can be adapted to handle the FDF synthesis problem for homogeneous MJLS with constant or uncertain but time-invariant probability matrix by simply making $\alpha(k + 1) = \alpha(k) = \alpha$.

The constraints presented in Theorem 1 are BMIs, which means that is necessary to use appropriate tools in order to solve them. Among a number of techniques available in the literature [26–28], we employ the coordinate descend algorithm (CDA), since it is a well-known and widely used tool to solve such issues. Accordingly, an iterative procedure based in CDA is given below to solve Theorem 1.

In Algorithm 1, ϕ represents the stop criteria and t_{\max} is the maximum number of iterations allowed. Observe that if a solution is found in the first iteration of CDA, the iterative procedure will converge to an optimized solution or at least keep the same solution found in the first iteration. The CDA is better detailed in [26,27].

Algorithm 1 Coordinate descent algorithm.

Coordinate descent algorithm (CDA):

Input: $\mathcal{B}_{\eta i}, \gamma, t_{\max}, \phi$.

Output: $\mathcal{A}_{\eta i}, \mathcal{B}_{\eta i}, \mathcal{M}_{\eta i}, \mathcal{C}_{\eta i}$.

Initialization:

While: $\frac{\gamma^{t-1} - \gamma^t}{\gamma^{t-1}} \leq \eta$ or $t \leq t_{\max}$ **do:**

Step 1: Find a solution for the LMI constraint (25) obtaining the values of X using as an input \mathcal{B}_i , which can be obtained by using any method, for example, the one in Theorem 1 in [7].

Step 2: Now find a solution for the same LMI constraint (25) to obtain $\mathcal{A}_{\eta i}, \mathcal{B}_{\eta i}, \mathcal{M}_{\eta i}, \mathcal{C}_{\eta i}$, but this time by using X as an input. Also obtain the value of γ .

6. Numerical Example

To illustrate the applicability of the theoretical results, a numerical example of a coupled tanks model (see Figure 2) is presented next. In this example, we assume that the plant itself is time-invariant, nevertheless, there are some time-varying parameters associated with the model of the communication network, which is represented by a nonhomogeneous Markov chain.

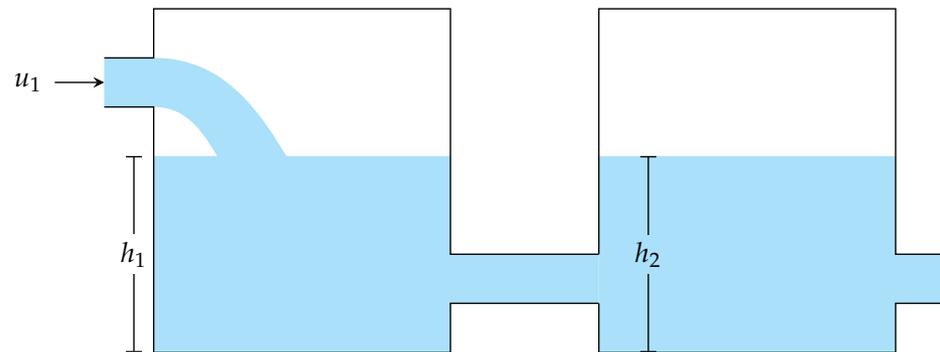


Figure 2. Coupled tanks model considered in the numerical example. The level on each tank h_1, h_2 , measured independently from each other, are the system states, while the control input is the inlet flow u_1 into the first tank.

6.1. Simulation Setup

The parameters and modeling of the coupled tanks system were extracted from [29], such that the continuous-time state-space matrices are given by

$$A = \begin{bmatrix} -0.0239 & -0.0127 \\ 0.0127 & -0.0285 \end{bmatrix}, B = \begin{bmatrix} 0.71 \\ 0 \end{bmatrix},$$

$$J = \begin{bmatrix} 0.0071 \\ 0 \end{bmatrix}, F = \begin{bmatrix} 0.071 \\ 0 \end{bmatrix}, K = [-1.03 \quad -0.33]. \tag{39}$$

The sampling time used is $T_s = 1s$. Note that in order to represent the fault as an abnormal input on the first tank, the matrix associated with the fault signal F is 10% of the control input matrix B .

Regarding the network modeling, we assume that each tank is far away from the other; therefore, data gathered from each sensor is transmitted via two distinct networks. Network 1 transmits the measurement of the first tank, and Network 2 transmits the measurements of the second tank. The transmission of the measurement signals through a semireliable communication network is modeled by using a simplified Gilbert–Elliot model, as done in [30] while the packet dropout is represented by the zero-input approach from [31], meaning that when a packet loss occurs, we assume that the value of the received signal is null. Hence, the complete network behavior is represented by four distinct operation modes, as illustrated by Figure 3. The first one is that where all the measurements are correctly transmitted (called “Ok Ok” in Figure 3); the second one considers that the measurement on the first tank is successfully transmitted, but occurs a packet dropout of the measurement from the second tank (called “Ok Drop” in Figure 3); the third case is the opposite of the second one (called “Drop Ok” in Figure 3); and the last mode represents the case where all measurements were lost (called “Drop Drop” in Figure 3).

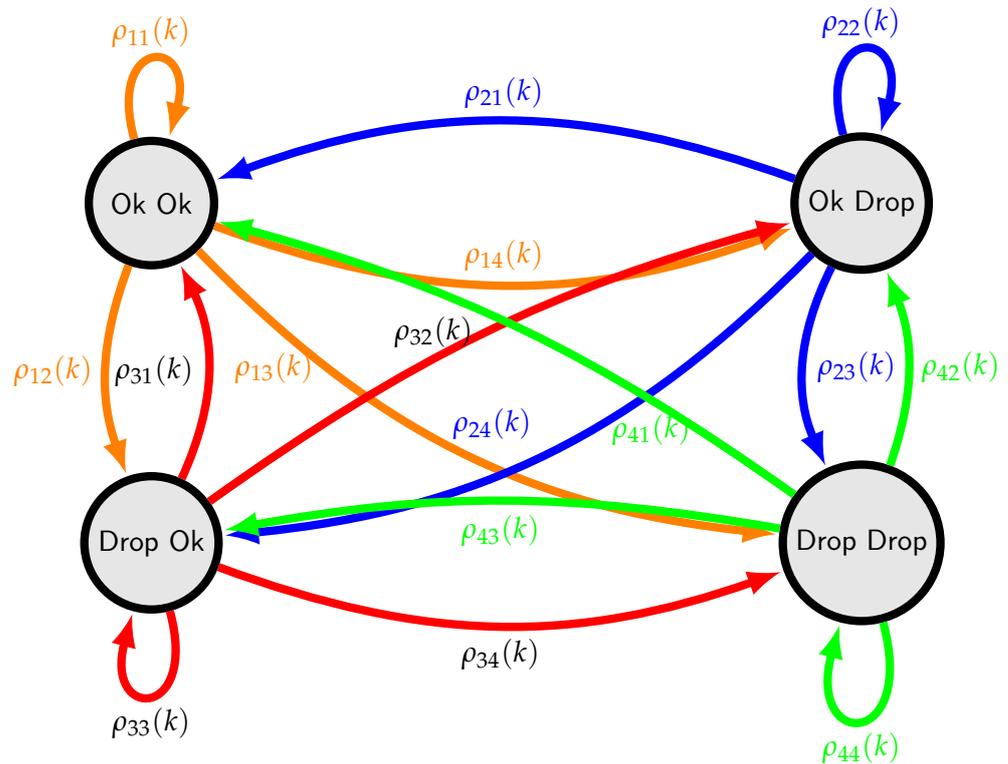


Figure 3. Graphic representation of the network states modeled as a Markov chain where the mode Ok Ok denotes that the networks responsible for transmitting the measurement of both sensors are operating on the nominal state, the mode Ok Drop represents the situation where there is a problem in the network in charge of transmitting the measurements of the second tank, the mode Drop Ok represents a failed transmission of the first tank measurements, and the mode Drop Drop stands for the case when both networks present some issues.

Remark 6. It is essential to point out that the proposed solution allows modeling the network packet dropout rate varying in time. However, the feasibility of the proposed solution is dictated by three main features, the system dynamic, the number of Markov modes, and range variation. Therefore, it is important to keep in mind that an overly complex Markov chain with a high range of variation will request more computational effort, and in some situations, a feasible solution may not be achieved.

From the previous network description, we can write the matrices that represent the measurement signal $y(k)$ are denoted by (the subindex is associated with the network operation mode)

$$\begin{aligned}
 C_1 &= \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, & C_2 &= \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix}, & C_3 &= \begin{bmatrix} 0 & 0 \\ 0 & 1 \end{bmatrix}, & C_4 &= \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix}, \\
 D_1 &= \begin{bmatrix} 0 & 0.01 \\ 0 & 0.01 \end{bmatrix}, & D_2 &= \begin{bmatrix} 0 & 0.01 \\ 0 & 0 \end{bmatrix}, & D_3 &= \begin{bmatrix} 0 & 0 \\ 0 & 0.01 \end{bmatrix}, & D_4 &= \begin{bmatrix} 0 \\ 0 \end{bmatrix}, \\
 E_{1,2,3,4} &= \begin{bmatrix} 0 \end{bmatrix}
 \end{aligned} \tag{40}$$

Observe that we imposed E equal to zero in all modes because despite the fault in the example representing an abnormal input on the first tank, the sensors are assumed to be healthy throughout the simulation.

To illustrate the flexibility of the proposed approach, a particular scenario was tested. Assume that we solely know the boundary of the transition probability matrix that governs the jumps among the four network modes. Another important assumption considered in the design process is that in the fourth mode, Drop Drop, where there is no sensor information, the scheduling parameter is not accessible, implying that the matrices that compose the FDF for this mode are designed in the robust form instead of the affine form.

The time-varying transition probability matrix is given by

$$\mathbb{P}(k) = \underbrace{\begin{bmatrix} 0.3 & 0.2 & 0.1 & 0.4 \\ 0.1 & 0.4 & 0.3 & 0.2 \\ 0.1 & 0.2 & 0.4 & 0.3 \\ 0.1 & 0.2 & 0.1 & 0.6 \end{bmatrix}}_{\mathbb{P}_{\beta_0}} + \underbrace{\begin{bmatrix} 0 & 0 & 0 & 0 \\ 1 & -1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}}_{\mathbb{P}_{\beta_1(k)}} \beta_1(k) + \underbrace{\begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 1 & -1 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}}_{\mathbb{P}_{\beta_2(k)}} \beta_2(k), \tag{41}$$

where $\beta_1(k) \in [0, 0.3], \beta_2(k) \in [0, 0.3]$, and by consequence $\beta_r(k) = 0\alpha_{r1}(k) + 0.3\alpha_{r2}(k), r = 1, 2$, where $\alpha(k) \in \Lambda_N = \Lambda_2 \times \Lambda_2, N = (2, 2)$.

Additionally, the transition probability among network states represented in Figure 3 by $\rho_{ij}(k)$ corresponds to an entry in the transition probability matrix $\mathbb{P}(k)$. Those transition probabilities depend on time as follows,

$$\rho_{ij}(k) = \rho_{0ij} + \beta_1(k)\rho_{1ij} + \beta_2(k)\rho_{2ij},$$

where ρ_{0ij}, ρ_{1ij} and ρ_{2ij} represent the elements of i th row and j th column from the following matrices $\mathbb{P}_{\beta_0}(k), \mathbb{P}_{\beta_1}(k)$ and $\mathbb{P}_{\beta_2}(k)$ of Equation (41). Furthermore, the time evolution of the time-varying parameters $\beta_1(k)$ and $\beta_2(k)$ is illustrated by Figure 4.

By using the above numerical values of the plant and the network, we are now able to apply Theorem 1 to provide a solution for the FDF, so that the filter matrices in (20) (with $\zeta = 0.9$ in Theorem 1) are given by

$$\begin{aligned} \mathcal{A}_{\eta 1\beta_0} &= \begin{bmatrix} 0.56 & 0.00 \\ -0.00 & -0.00 \end{bmatrix}, & \mathcal{A}_{\eta 1\beta_1} &= \begin{bmatrix} -0.01 & -0.00 \\ 0.00 & -0.00 \end{bmatrix}, & \mathcal{A}_{\eta 1\beta_2} &= \begin{bmatrix} -0.01 & -0.00 \\ 0.00 & -0.00 \end{bmatrix}, \\ \mathcal{A}_{\eta 2\beta_0} &= \begin{bmatrix} 0.12 & 0.00 \\ 0.00 & 0.00 \end{bmatrix}, & \mathcal{A}_{\eta 2\beta_1} &= \begin{bmatrix} -0.01 & -0.00 \\ -0.00 & -0.00 \end{bmatrix}, & \mathcal{A}_{\eta 2\beta_2} &= \begin{bmatrix} -0.01 & -0.00 \\ -0.00 & -0.00 \end{bmatrix}, \\ \mathcal{A}_{\eta 3\beta_0} &= \begin{bmatrix} 0.00 & -0.02 \\ -0.00 & -0.00 \end{bmatrix}, & \mathcal{A}_{\eta 3\beta_1} &= \begin{bmatrix} -0.00 & 0.03 \\ 0.00 & 0.00 \end{bmatrix}, & \mathcal{A}_{\eta 3\beta_2} &= \begin{bmatrix} -0.00 & 0.03 \\ 0.00 & 0.00 \end{bmatrix}, \\ \mathcal{A}_{\eta 4\beta_0} &= \begin{bmatrix} -0.00 & -0.00 \\ 0.00 & -0.00 \end{bmatrix}, \\ \mathcal{B}_{\eta 1\beta_0} &= \begin{bmatrix} 0.04 & 0.00 \\ -0.00 & 0.00 \end{bmatrix}, & \mathcal{B}_{\eta 1\beta_1} &= \begin{bmatrix} 0.01 & -0.00 \\ -0.00 & 0.00 \end{bmatrix}, & \mathcal{B}_{\eta 1\beta_2} &= \begin{bmatrix} 0.01 & -0.00 \\ -0.00 & 0.00 \end{bmatrix}, \\ \mathcal{B}_{\eta 2\beta_0} &= \begin{bmatrix} 0.03 & 0.00 \\ 0.00 & 0.00 \end{bmatrix}, & \mathcal{B}_{\eta 2\beta_1} &= \begin{bmatrix} 0.01 & 0.00 \\ 0.00 & 0.00 \end{bmatrix}, & \mathcal{B}_{\eta 2\beta_2} &= \begin{bmatrix} 0.01 & 0.00 \\ 0.00 & 0.00 \end{bmatrix}, \\ \mathcal{B}_{\eta 3\beta_0} &= \begin{bmatrix} 0.00 & 0.03 \\ 0.00 & 0.00 \end{bmatrix}, & \mathcal{B}_{\eta 3\beta_1} &= \begin{bmatrix} 0 & 0.01 \\ 0 & 0.00 \end{bmatrix}, & \mathcal{B}_{\eta 3\beta_2} &= \begin{bmatrix} 0 & 0.01 \\ 0 & 0.00 \end{bmatrix}, \\ \mathcal{B}_{\eta 4\beta_0} &= \begin{bmatrix} 0.00 & 0.00 \\ 0.00 & 0.00 \end{bmatrix}, \\ \mathcal{M}_{\eta 1\beta_0} &= \begin{bmatrix} 0.04 \\ 0.00 \end{bmatrix}, & \mathcal{M}_{\eta 1\beta_1} &= \begin{bmatrix} 0.01 \\ 0.00 \end{bmatrix}, & \mathcal{M}_{\eta 1\beta_2} &= \begin{bmatrix} 0.01 \\ 0.00 \end{bmatrix}, \\ \mathcal{M}_{\eta 2\beta_0} &= \begin{bmatrix} 0.03 \\ 0.00 \end{bmatrix}, & \mathcal{M}_{\eta 2\beta_1} &= \begin{bmatrix} 0.01 \\ 0.00 \end{bmatrix}, & \mathcal{M}_{\eta 2\beta_2} &= \begin{bmatrix} 0.01 \\ 0.00 \end{bmatrix}, \\ \mathcal{M}_{\eta 3\beta_0} &= \begin{bmatrix} 0.03 \\ 0.00 \end{bmatrix}, & \mathcal{M}_{\eta 3\beta_1} &= \begin{bmatrix} 0.01 \\ 0.00 \end{bmatrix}, & \mathcal{M}_{\eta 3\beta_2} &= \begin{bmatrix} 0.01 \\ 0.00 \end{bmatrix}, \\ \mathcal{M}_{\eta 4\beta_0} &= \begin{bmatrix} 0.01 \\ 0.00 \end{bmatrix}, \\ \mathcal{C}_{\eta 1\beta_0} &= [1.20 \ -0.12], & \mathcal{C}_{\eta 1\beta_1} &= [-0.01 \ -0.01], & \mathcal{C}_{\eta 1\beta_2} &= [-0.01 \ -0.01], \\ \mathcal{C}_{\eta 2\beta_0} &= [0.32 \ 0.01], & \mathcal{C}_{\eta 2\beta_1} &= [-0.00 \ -0.00], & \mathcal{C}_{\eta 2\beta_2} &= [-0.01 \ -0.00], \\ \mathcal{C}_{\eta 3\beta_0} &= [0.02 \ -0.02], & \mathcal{C}_{\eta 3\beta_1} &= [-0.00 \ 0.01], & \mathcal{C}_{\eta 3\beta_2} &= [-0.00 \ 0.01], \\ \mathcal{C}_{\eta 4\beta_0} &= [-0.01 \ -0.00]. \end{aligned} \tag{42}$$

6.2. Simulation Result

In this section, for comparison purposes, we present simulation results with one design of FDF that assumes complete knowledge of the modes and the other design that uses the results from [8], where an \mathcal{H}_∞ guaranteed cost is used as the performance criterion. This comparison is important to show that the consideration of the nonhomogeneous MC impacts the FDF performance. This particular paper was chosen to be compared with the proposed approach since both are based on the MJLS framework and are based on the H_∞ norm. Another critical piece of information that can be gathered from this comparison is the complexity of the problem versus performance gain. We remind that the parameters $\beta_1(k), \beta_2(k)$ are assumed to be instantly measurable. Hence, the parameters $\beta_1(k), \beta_2(k)$ vary during the simulation according to the information provided by Figure 4.

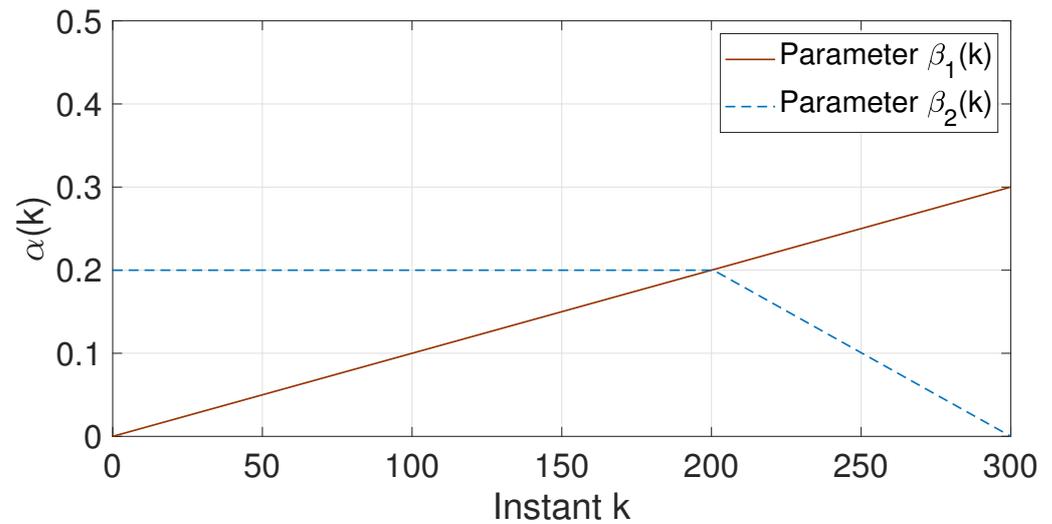


Figure 4. The behavior of the parameters $\beta_1(k)$, $\beta_2(k)$ during each simulation. The parameters $\beta_1(k)$, $\beta_2(k)$ are responsible for representing the time-varying aspect of the transition matrix $\mathbb{P}(k)$ in (41).

There are three faults that the system is subjected to during simulation. Those faults, presented in Figure 5, were applied on the first tank, and both affect the signal multiplying matrix F in (39). A Monte Carlo simulation with 100 interactions was performed, and Figures 6 and 7 illustrate the obtained results.

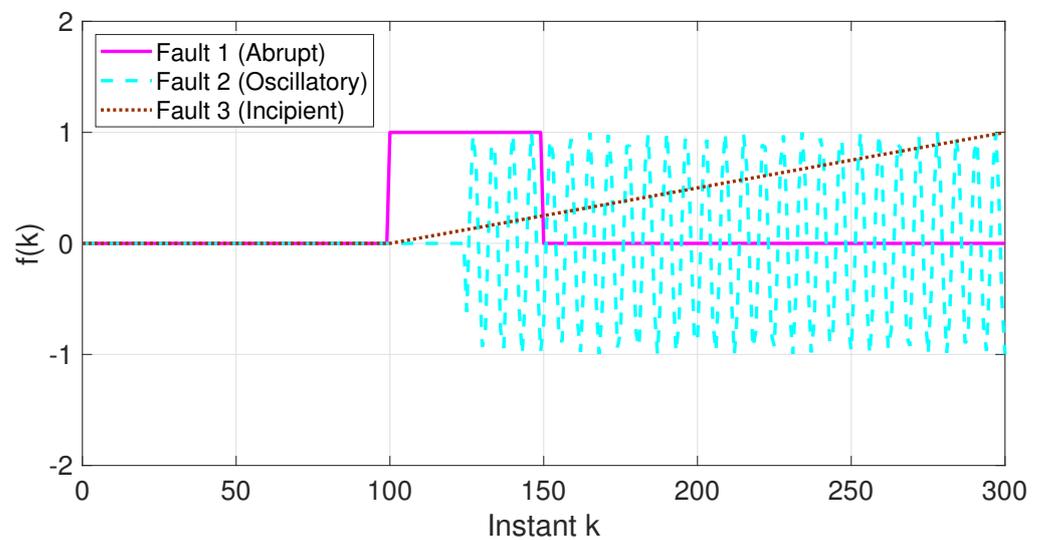


Figure 5. The representation of the three faults that were inflicted in the system during simulations. The magenta curve represents an abrupt fault, the cyan curve denotes an oscillatory fault, and the brown curve is the incipient fault.

Figure 6 presents the system’s states when subjected to all three fault cases, and also a case without fault. From Figure 6, it is possible to observe that the faults do not surpass 10% of the nominal state value.

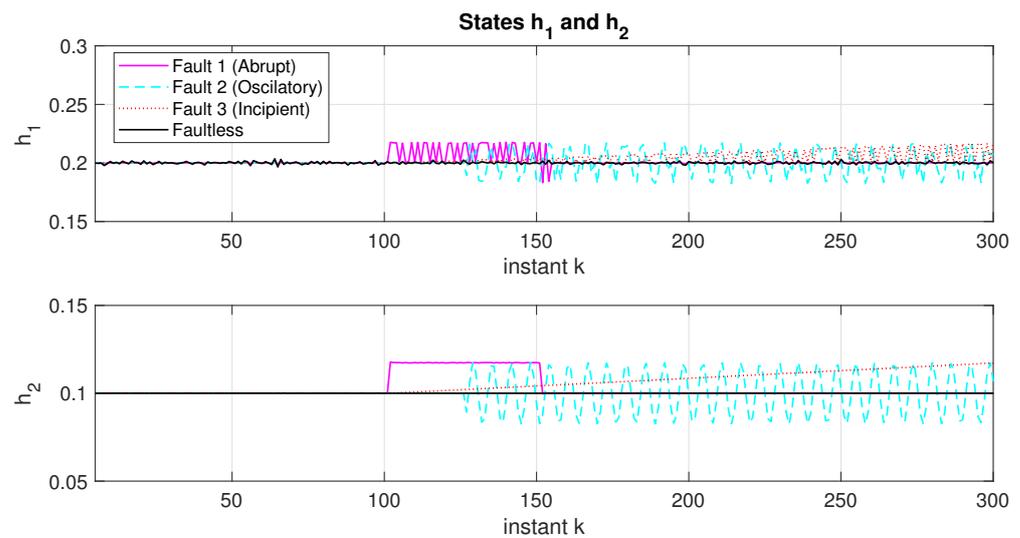


Figure 6. System states when subjected to all three faults, where the magenta curves denote the states with an abrupt fault, the cyan curves represent the states for the oscillatory fault, and brown curves are the states results for the incipient fault. The black curves represent the state without faults.

The residue signal generated by the FDFs for each type of fault, and also the case without fault are presented in Figure 7. We may state that all the FDF worked as expected, have been affected by the faults when it occurs, and that when there is not a fault the residue kept close to zero. Note that the results obtained by the FDF designed with the complete knowledge of the modes show more sensitivity against the fault, which helps to detect the fault faster.

A fault-detection procedure has two major stages: the residue generation and evaluation process. In Figure 7, the residue signals obtained via simulation were presented. To execute the next stage, it is necessary to define two tools: the evaluation function ($EVAL(k)$) and the threshold (TH). The definition of these two tools are a deep issue that will not be tackled here; therefore, for a more detailed discussion, please refer to [1,4]. Consider the following definition for the evaluation function:

$$EVAL(k) \triangleq \sqrt{\sum_{i=k-L}^k r(i)'r(i)}, \tag{43}$$

where L represents the evaluation window, which, in this particular simulation is assumed to be $L = 250$. The threshold TH is used to assess the evaluation function $EVAL(k)$. If $EVAL(k) > TH$ there is a fault, and for the opposite case there is no fault. For the simulations, the value of the threshold was arbitrarily set to $TH = 5$. From the aforementioned, the evaluation functions for all the FDF considering each fault were calculated and presented in Figure 8.

As can be seen in Figure 8, all FDFs are able to detect the different faults; however, there is a clear difference in the time that it takes for each FDF, as indicated by Table 1. Note that the proposed approach presented a faster detection for all three faults. This occurred because the proposed solution considers a more trustworthy model. Thus, an abnormal change can be detected more quickly.

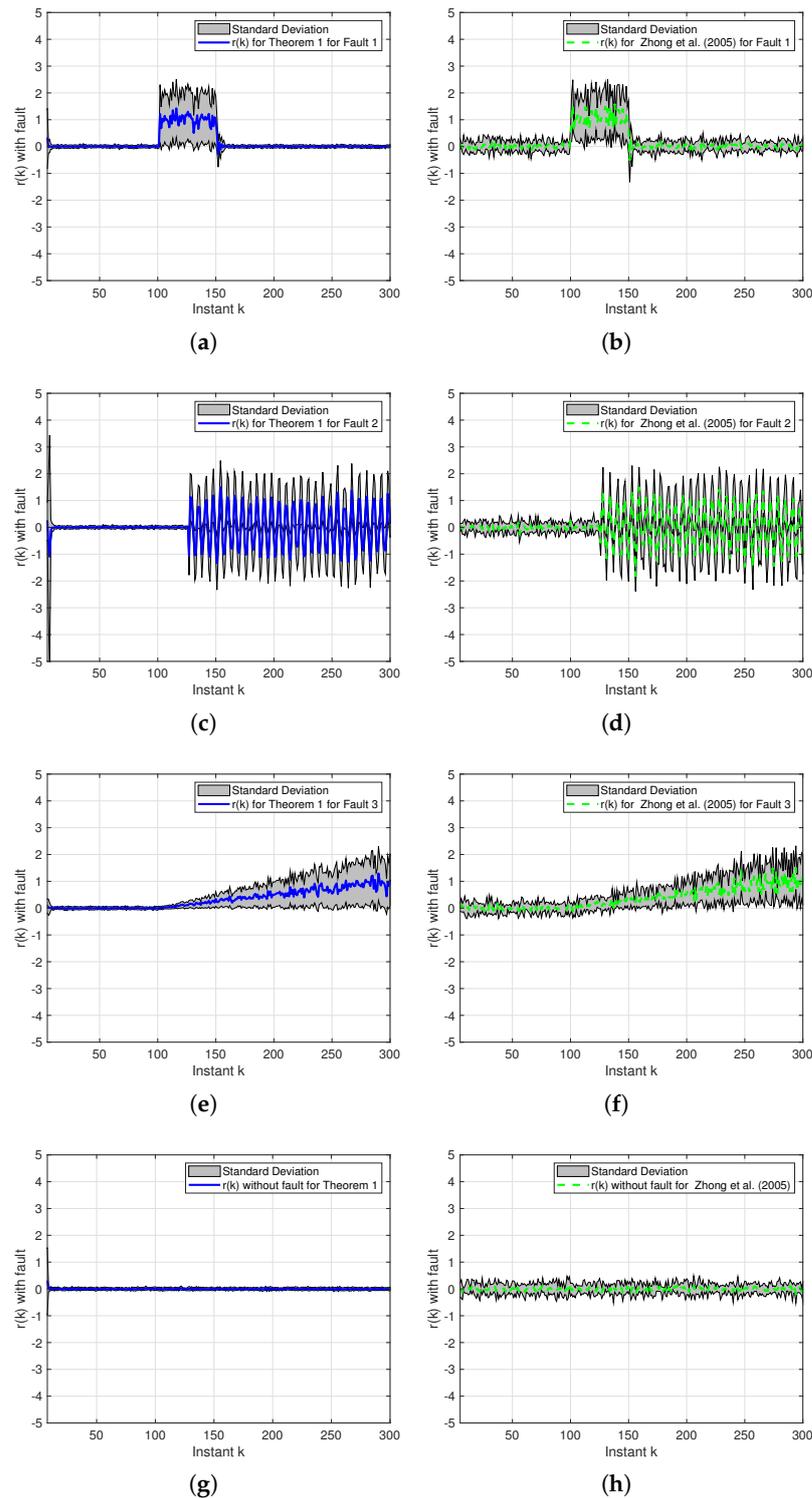


Figure 7. The residue signal obtained for FDF designed by using Theorem 1 under the assumption of knowledge of all modes (to the left), and the FDF provided by [8] (to the right). Both designs are subjected to three type of faults, and also the situation without fault. (a) Residue signal for FDF designed via Theorem 1 subjected to Fault 1. (b) Residue signal for FDF designed via [8] subjected to Fault 1. (c) Residue signal for FDF designed via Theorem 1 subjected to Fault 2. (d) Residue signal for FDF designed via [8] subjected to Fault 2. (e) Residue signal for FDF designed via Theorem 1 subjected to Fault 3. (f) Residue signal for FDF designed via [8] subjected to Fault 3. (g) Residue signal for FDF designed via Theorem 1 without fault. (h) Residue signal for FDF designed via [8] without fault.

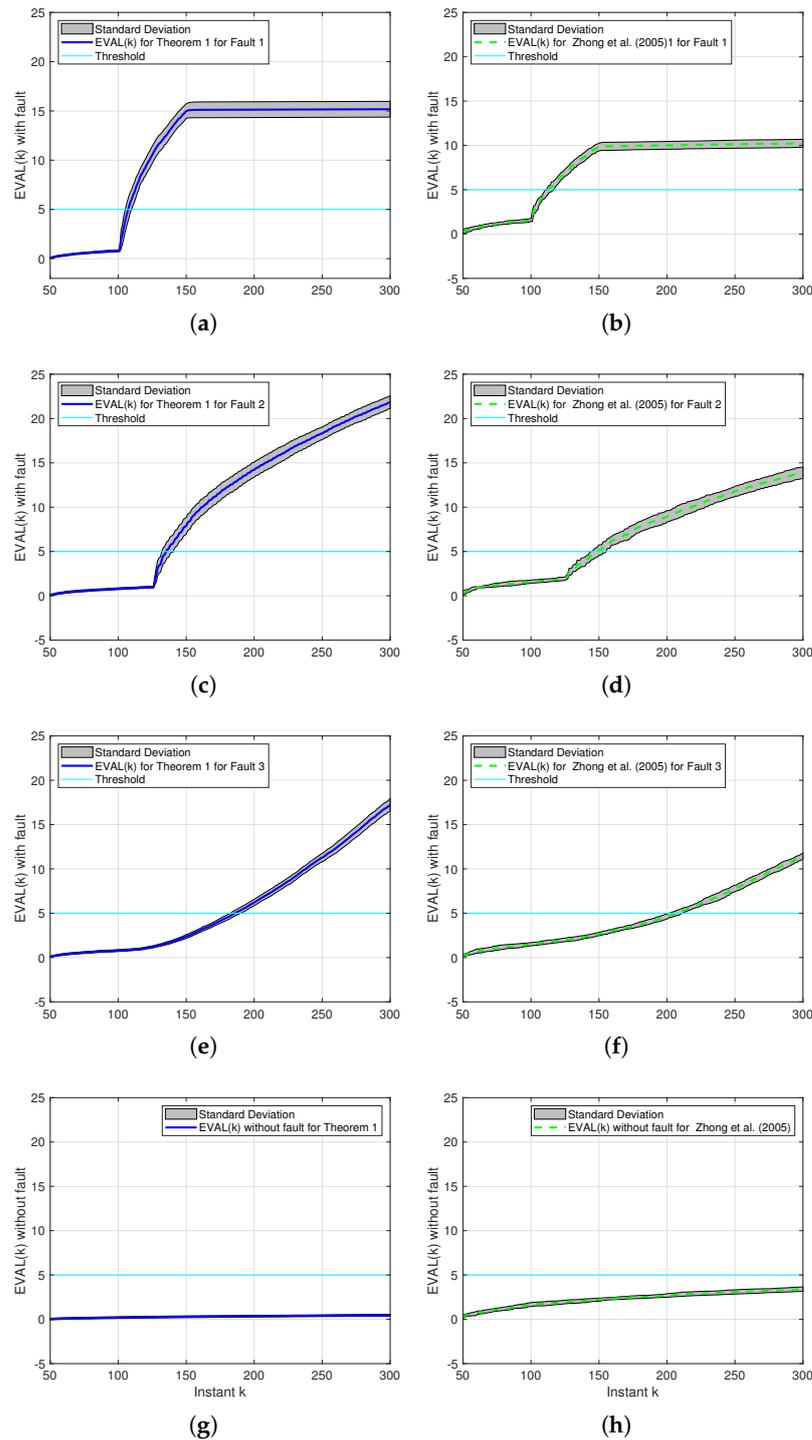


Figure 8. The evaluation function obtained for FDF designed by using Theorem 1 under the assumption of knowledge of all modes (to the left), and the FDF provided by [8] (to the right). Both designs are subjected to three type of faults, and also the situation without fault. (a) Evaluation function for FDF designed via Theorem 1 subjected to Fault 1. (b) Evaluation function for FDF designed via [8] subjected to Fault 1. (c) Evaluation function for FDF designed via Theorem 1 subjected to Fault 2. (d) Evaluation function for FDF designed via [8] subjected to Fault 2. (e) Evaluation function for FDF designed via Theorem 1 subjected to Fault 3. (f) Evaluation function for FDF designed via [8] subjected to Fault 3. (g) Evaluation function for FDF designed via Theorem 1 without fault. (h) Evaluation function for FDF designed via [8] without fault.

Table 1. The detection window for all FDF considering each fault after the Monte Carlo simulation.

Design Method	Fault 1 (k)	Fault 2 (k)	Fault 3 (k)
Theorem 1 (All)	[105 114]	[133 140]	[183 188]
[8]	[112 117]	[145 156]	[202 212]

Another interesting aspect is that the FDF designed by using [8] (the FDF under the assumption that the MC is homogeneous) is affected by the variation even when there is no fault occurrence. This particularity can be observed in Figure 8h, where the evaluation value drifts away from zero. This is an issue that affects the performance of the FDF, since it may cause false alarms. Note that this phenomenon does not occur on the FDF designed by using the proposed approach, showing that the consideration of the nonhomogeneous MC is useful to increase the performance of the FDF, especially for the situation where a nonhomogeneous MC is used to model characteristics that are inherently time-varying, such as a wireless sensor network. The last piece of information gathered from the simulation is that even though the proposed solution was based on more assumptions and a higher number of LMI constraints, which is clearly more computationally costly compared to the results in [8], the proposed approach provides superior performance.

7. Conclusions

The main contribution of this paper is the development of a new design method for gain-scheduled FDF considering that the plant measurement signals are transmitted through a network whose dropouts are modeled by using the MJLS theory, with the Markov chain being nonhomogeneous. The premise of nonhomogeneous MC is tackled by using the LPV modelling, specifically applied to the probability transition matrix, allowing us to design the FDF under this particular circumstance. The FDF synthesis results are obtained by using parameter-dependent LMI constraints that employ \mathcal{H}_∞ norm as performance index. Since the gain-scheduled FDF varies according to the probability transition matrix variation, the proposed FDF is optimal for all the variation range of the Markov chain. To illustrate this, a comparison between the proposed technique with another method from the literature that does not contemplate the nonhomogeneous MC assumption was made. The simulation results show that the nonhomogeneous assumption positively impacts the FDF performance, allowing the proposed technique to detect the fault in a smaller time window, which can be seen in Table 1. Furthermore the FDF does not present false variations in the evaluation function, that could be identified as faults, when a fault does not occur, as can be seen in Figure 8g,h. A possible next step along this research line would be to include a sensitivity \mathcal{H}_- index in the design to further improve the FDF performance.

Author Contributions: Conceptualization, L.C., J.M.P. and C.F.M.; Methodology, C.F.M. and L.C.; Validation, C.F.M., B.J. and O.L.V.C.; Investigation, L.C. and J.M.P.; Writing—original draft, L.C.; Writing—review & editing, J.M.P., C.F.M., B.J. and O.L.V.C.; Visualization, L.C.; Supervision, B.J. and O.L.V.C.; Project administration, O.L.V.C.; Funding acquisition, J.M.P. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by ANID-FONDECYT grant number 11201049 and 1220903. And The APC was funded by ANID-FONDECYT. The first, second, and third authors were funded by the Chilean National Agency for Research and Development, project ANID-FONDECYT Iniciacion finance code N° 11201049 and ANID-FONDECYT 1220903. The fifth author was supported by Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPq), process No. 304149/2019-5, and by Instituto Nacional de Ciência e Tecnologia para Sistemas Autônomos Cooperativos (INSAC), process CNPq/INCT -465755/2014-3 and FAPESP/INCT-2014/50851-0.

Data Availability Statement: Not applicable.

Conflicts of Interest: The authors declare no conflict of interest.

Abbreviations

The following abbreviations are used in this manuscript:

Fault Detection	FD
Markov Jump Linear Systems	MJLS
Markov Chain	MC
Networked Control System	NCS
Fault Detection Filter	FDF
Linear Parameter Varying	LPV
Exponentially Stable in the Mean Square Sense with Conditioning of Type 1	ESMS-CI
Linear Matrix Inequality	LMI
Coordinate Descend Algorithm	CDA

References

- Chen, J.; Patton, R.J. *Robust Model-Based Fault Diagnosis for Dynamic Systems*; Springer Science & Business Media: Berlin/Heidelberg, Germany, 2012; Volume 3.
- Patton, R.J.; Frank, P.M.; Clark, R.N. *Issues of Fault Diagnosis for Dynamic Systems*; Springer Science & Business Media: Berlin/Heidelberg, Germany, 2013.
- Isermann, R.; Schwarz, R.; Stolzl, S. Fault-tolerant drive-by-wire systems. *IEEE Control Syst.* **2002**, *22*, 64–81.
- Frank, P.M.; Ding, X. Survey of robust residual generation and evaluation methods in observer-based fault detection systems. *J. Process. Control* **1997**, *7*, 403–424. [[CrossRef](#)]
- Al-Karaki, J.N.; Kamal, A.E. Routing techniques in wireless sensor networks: A survey. *IEEE Wirel. Commun.* **2004**, *11*, 6–28. [[CrossRef](#)]
- Wu, J.; Chen, T. Design of networked control systems with packet dropouts. *IEEE Trans. Autom. Control* **2007**, *52*, 1314–1319. [[CrossRef](#)]
- de Paula Carvalho, L.; de Oliveira, A.M.; do Valle Costa, O.L. Robust fault detection H_∞ filter for Markovian jump linear systems with partial information on the jump parameter. *IFAC-PapersOnLine* **2018**, *51*, 202–207. [[CrossRef](#)]
- Zhong, M.; Ye, H.; Shi, P.; Wang, G. Fault detection for Markovian jump systems. *IEE Proc.-Control Theory Appl.* **2005**, *152*, 397–402. [[CrossRef](#)]
- Wu, L.; Luo, W.; Zeng, Y.; Li, F.; Zheng, Z. Fault detection for underactuated manipulators modeled by Markovian jump systems. *IEEE Trans. Ind. Electron.* **2016**, *63*, 4387–4399. [[CrossRef](#)]
- Zhai, D.; An, L.; Li, J.; Zhang, Q. Fault detection for stochastic parameter-varying Markovian jump systems with application to networked control systems. *Appl. Math. Model.* **2016**, *40*, 2368–2383. [[CrossRef](#)]
- de Paula Carvalho, L.; Toriumi, F.Y.; Angélico, B.A.; do Valle Costa, O.L. Model-based fault detection filter for Markovian jump linear systems applied to a control moment gyroscope. *Eur. J. Control* **2021**, *59*, 99–108. [[CrossRef](#)]
- Lin, A.; Cheng, J.; Park, J.H.; Yan, H.; Qi, W. Fault Detection Filtering of Nonhomogeneous Markov Switching Memristive Neural Networks with Output Quantization. *Inf. Sci.* **2023**, *632*, 715–729. [[CrossRef](#)]
- Zhao, L.; Zhang, H.; Hu, J.; Xu, L. Event-based Sliding Mode Control for Markovian Jump Systems with Time-varying Delays: An Observer Method. *Int. J. Control Autom. Syst.* **2023**, 1–10. [[CrossRef](#)]
- de Paula Carvalho, L.; Jayawardhana, B.; do Valle Costa, O.L. Fault Detection Filter for Discrete-Time Markov Jump Lur'e Systems. In Proceedings of the 2021 European Control Conference (ECC), Delft, The Netherlands, 29 June–2 July 2021; pp. 1826–1831.
- Ross, S.M. *Introduction to Probability Models*; Academic Press: Cambridge, MA, USA, 2014.
- Aberkane, S. Bounded real lemma for nonhomogeneous Markovian jump linear systems. *IEEE Trans. Autom. Control* **2012**, *58*, 797–801. [[CrossRef](#)]
- Palma, J.M.; Morais, C.F.; Oliveira, R.C.L.F. H_2 gain-scheduled filtering for discrete-time LPV systems using estimated time-varying parameters. In Proceedings of the 2018 American Control Conference, Milwaukee WI, USA, 27–29 June 2018; pp. 4367–4372. [[CrossRef](#)]
- Palma, J.M.; Morais, C.F.; Oliveira, R.C. A less conservative approach to handle time-varying parameters in discrete-time linear parameter-varying systems with applications in networked control systems. *Int. J. Robust Nonlinear Control* **2020**, *30*, 3521–3546. [[CrossRef](#)]
- Wu, J.; Shi, H.; Jiang, X.; Su, C.; Li, P. Stochastic fuzzy predictive fault-tolerant control for time-delay nonlinear system with actuator fault under a certain probability. *Optim. Control Appl. Methods* **2022**. [[CrossRef](#)]
- He, S.; Lyu, W.; Liu, F. Robust H_∞ Sliding Mode Controller Design of a Class of Time-Delayed Discrete Conic-Type Nonlinear Systems. *IEEE Trans. Syst. Man Cybern. Syst.* **2021**, *51*, 885–892. [[CrossRef](#)]
- Iosifescu, M. *Finite Markov Processes and Their Applications*; Courier Corporation: North Chelmsford, MA, USA, 2014.
- Morais, C.F.; Braga, M.F.; Oliveira, R.C.; Peres, P.L. H_2 control of discrete-time Markov jump linear systems with uncertain transition probability matrix: Improved linear matrix inequality relaxations and multi-simplex modelling. *IET Control Theory Appl.* **2013**, *7*, 1665–1674. [[CrossRef](#)]

23. Dragan, V.; Morozan, T.; Stoica, A.M. *Mathematical Methods in Robust Control of Discrete-Time Linear Stochastic Systems*; Springer: Berlin/Heidelberg, Germany, 2010.
24. Boyd, S.; El Ghaoui, L.; Feron, E.; Balakrishnan, V. *Linear Matrix Inequalities in System and Control Theory*; SIAM: Philadelphia, PA, USA, 1994.
25. Agulhari, C.M.; Felipe, A.; Oliveira, R.; Peres, P.L. Manual of “The Robust LMI Parser”, Version 3.0. 2019. Available online: https://www.researchgate.net/profile/Mohamed-Mourad-Laffi/post/Can_anyone_help_me_to_find_the_maximum_number_of_linear_matrix_inequality_LMI_that_can_be_solved_by_using_YALMIP_or_LMI_toolbox/attachment/5cb8ae87cfe4a7df4ae9bc27/AS%3A749084939923456%401555607175083/download/Manual+of+%E2%80%9CThe+Robust+LMI+Parser%E2%80%9D+%E2%80%93+Version+3.0.pdf (accessed on 3 March 2023).
26. Simon, E.; R-Ayerbe, P.; Stoica, C.; Dumur, D.; Wertz, V. LMIs-based coordinate descent method for solving BMIs in control design. *IFAC Proc. Vol.* **2011**, *44*, 10180–10186. [[CrossRef](#)]
27. Wang, Y.; Zemouche, A.; Rajamani, R. A sequential LMI approach to design a BMI-based multi-objective nonlinear observer. *Eur. J. Control* **2018**, *44*, 50–57. [[CrossRef](#)]
28. de Paula Carvalho, L.; Rosa, T.; Jayawardhana, B.; Costa, O. Fault accommodation controller under Markovian jump linear systems with asynchronous modes. *Int. J. Robust Nonlinear Control* **2020**, *30*, 8503–8520. [[CrossRef](#)]
29. Feedback Instruments, Ltd. *FeedBack Coupled Tanks Control Experiments 33-041S (For Use with MATLAB)*, 1st ed.; Feedback Instruments, Ltd.: Crowborough, UK, 2013; pp. 1–49.
30. Gilbert, E.N. Capacity of a burst-noise channel. *Bell Syst. Tech. J.* **1960**, *39*, 1253–1265. [[CrossRef](#)]
31. Schenato, L. To zero or to hold control inputs with lossy links? *IEEE Trans. Autom. Control* **2009**, *54*, 1093–1099. [[CrossRef](#)]

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.