DIAGNOSTYKA, 2024, Vol. 25, No. 3

e-ISSN 2449-5220
DOI: 10.29354/diag/192181

FAULT TOLERANT CONTROL AND FAULT DIAGNOSIS METHODS INTEGRATED USING INTELLIGENT CONTROLLER FOR HYBRID DYNAMIC SYSTEMS

Mohammed Said ACHBI [1, 2], Boubakeur ROUABAH [1], Mohamed Abdelbasset MAHBOUB [1], Bilal BENARABI [1], Mohamed Redouane KAFI [1], Sihem KECHIDA [2]

[1] Université Kasdi Merbah, Ouargla, Algérie
[2] Laboratoire d'Automatique et Informatique de Guelma (LAIG lab.). Université 8 Mai 1945 Guelma, BP 401, Guelma 24000, Algérie

Abstract

This research aims to provide a fault diagnosis approach for Hybrid Dynamic (SDHs), Systems and Fault-Tolerant Control synthesis, while also ensuring the smooth operation of industrial settings. This study is a part of the larger topic of Hybrid Dynamic System control and fault diagnosis. The primary focus is on modelling strategies designed expressly for Hybrid Dynamic Systems, with a concentration on combining continuous and event-driven components. Much work is devoted to developing a model that can incorporate both kinds of elements. A system model that can track several modes without explicit identification can be created by utilizing Neuro-Fuzzy Networks, providing a thorough overview. On the basis of this synthesized model, an AI-based fault diagnosis method is subsequently developed.

Keywords: hybrid dynamic systems, modeling, fault diagnosis, fault tolerant control, neural fuzzy systems.

1. INTRODUCTION

A system comprises elements that work together to achieve a specific task, with connections to external factors through inputs (control and disturbances) and outputs (reactions or responses). Systems come in various types, such as static or unchanging systems, dynamic systems that evolve over time, systems dealing with one or multiple variables, continuous or uninterrupted systems, discrete event systems, linear or non-linear systems, causal systems, and invariant or unchanging systems, among others [1].

Real-world or industrial systems tend to be intricate, exhibiting dynamics that are modeled by a mix of discrete and continuous phenomena when viewed on a larger scale. As a result of combining discrete and continuous elements, a new class of systems known as "Hybrid Dynamic Systems" (HDS) emerges. From the 1990s onward, significant emphasis within the scientific community has been placed on investigating these hybrid systems, as evidenced by various studies [2], [3], [4], [5], [6].

Numerous modeling approaches have been introduced to address the simultaneous integration of continuous and discrete aspects in these systems.

The goal of enhancing system automation is to enhance its efficiency. However, the pursuit of improved performance has resulted in the creation of increasingly intricate systems, thereby escalating the potential for malfunctions that could jeopardize both the system and its surroundings.

Consequently, in numerous applications, it becomes essential to establish a monitoring or diagnostic system to identify, locate, and recognize faults. Accomplishing this task necessitates a comprehensive understanding of a fault model. Numerous techniques have been developed in the field of automation to locate and diagnose problems in hybrid dynamic systems. Every method is distinguished by the necessary comprehension of the fundamental procedure. Certain strategies emphasize an exhaustive understanding of the system, relying on detailed prior knowledge to pinpoint faults. Conversely, alternative methods prioritize fault detection and diagnosis without comprehensive prior insights, employing adaptive algorithms or machine learning techniques to deduce faults in these systems. The spectrum of these approaches underscores the significance of prior knowledge and its influence on fault detection strategies in hybrid dynamic systems.

Received .2023-11-26; Accepted 2024-08-09; Available online 2024-08-22

The diagnostic technique for hybrid dynamic systems proposed in reference [7] combines hybrid automata with neural-fuzzy systems.

References [8], [9] and [10] concentrate on a modelling strategy tailored for systems with hybrid dynamics. Finding a good model that combines continuous and event aspects is the goal. Next, a fault diagnosis method utilizing Artificial Intelligence (AI) techniques is developed.

The ability of a system to continue operating normally in the event that one or more of its components fail is known as system integrity. Many academics investigated this issue in their early research [11] and [12]. An additive control law is created by combining integral action and state feedback control in the Fault Tolerant Control (FTC) proposition. This approach relies on an extended data-driven projection method (EDPM) that estimates faults using input and output measures, eliminating the necessity for mathematical models, as outlined in reference [13].

The Fault Tolerant Control (FTC) method ensures the integrity of the multicellular converter's structure even in the event of flying capacitors failure. This is achieved through the implementation of robust sliding mode control, effectively managing various fault scenarios, as detailed in reference [14].

A study that proposes Fault Tolerant Control (FTC) and a machine learning-based fault diagnosis approach is presented in Reference [15]. In this study and [16], two power converters, one using the Maximum Power Point Tracking (MPPT) algorithm and the other a three-cell multi-cellular converter, are used to power a photovoltaic water pumping system. The DC motor is connected to a submerged pump through this converter.

The work in this paper is dedicated to the modeling of a hybrid dynamic system, employing a two-part approach.

- Firstly, Neuro-Fuzzy models are developed to represent the continuous aspect of the system during standard operation.
- Simultaneously, a hybrid automata model is formulated to capture the discrete aspect of the system.
- For system diagnosis, an offline diagnostic tool based on Neuro-Fuzzy models is proposed, specifically tailored for these systems.
- Additionally, the study presents an approach to fault accommodation by utilizing pre-computed Neuro-Fuzzy control laws, thereby introducing a fault-tolerant command mechanism equipped with a compensation term to address faults effectively. This innovative methodology leverages Fuzzy Logic to facilitate the selection of pre-computed Neuro-Fuzzy control laws, ensuring robust fault accommodation within the system.

Through simulation, the proposed techniques are set to be validated, considering the intricate nonlinear dynamics of the model. By integrating Fuzzy Logic into the selection process, the study not only enhances fault tolerance but also demonstrates the adaptability and effectiveness of these techniques in addressing complex system faults.

2. MODELING HYBRID DYNAMIC SYSTEM

Conventional representations of the behavior of dynamic systems rely on a model in which the form (continuous or discrete) is directly related to the underlying state variables and the temporal features that characterize the system. In order to model hybrid dynamic systems, two different behaviors must be described: First, the discrete dynamics represented by a set of states and transitions; and second, the continuous dynamics usually represented by a system of differential and algebraic equations. Reconciling these distinct and continuous parts has led to a substantial formalization.

A variety of hybrid models are used in the literature to describe physical processes that have traits of both continuous and event-driven behaviors [17], [18], [19] and [20].

2.1. Modelling Hybrid Dynamic System

Real systems frequently exhibit complex, non-stationary, and nonlinear characteristics, which presents modeling difficulties. Despite this intricacy, these elements should be taken into consideration while creating a predictive tool. For prediction challenges, a variety of artificial intelligence algorithms have been tested, showing better performance than conventional approaches [21], [22], [23]. This study highlights how well-suited Neuro-Fuzzy networks, in particular, Jang's invention, ANFIS, the Adaptive Neuro Fuzzy Inference System, are for handling such complexity [24]. More precisely, the work we do focus on investigating ANFIS's capabilities in this situation.

2.1.1. Adaptive Neuro-Fuzzy Inference System

The five-layer neural network structure used by ANFIS, or the Adaptive Network-Based Fuzzy Inference System, represents a stage in a Takagi-Sugeno type fuzzy inference system.

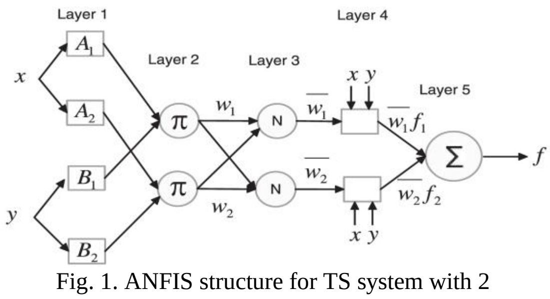

Fig. 1. ANFIS structure for TS system with 2 inputs/one output

Let's take a look at a fuzzy inference system that has two inputs (*x* and *y*) and an output (*f*) in order to simplify. Using two fuzzy Takagi-Sugeno rules as the rule base, the TS model applies these rules in a "If-Then" manner as follows:

R_1: $If\ x\ is\ A_1\ and\ y\ is\ B_1,$

$$Then\ y1 = f1\ (x, y) = p1\ x + q1\ y + r1 \quad (1)$$

R_2: $If\ x\ is\ A_2\ and\ y\ is\ B_2$,

$Then\ y1 = f2\,(x, y) = p2\,x + q2\,y + r2$ (2)

Each input x and y is connected to two nodes, representing the membership functions A_1, A_2, for x and the two inputs B_1, and B_2 for y.

The parameters p_1, q_1, r_1, and p_2, q_2, r_2 are linear coefficients associated with the output in the Takagi–Sugeno fuzzy inference model.

The core idea of the five-layer ANFIS architecture is the combination of the multilayer feed-forward neural network's supervised learning capabilities with the explicit knowledge of the Takagi-Sugeno (TS) fuzzy inference system. This combination defines the ANFIS technique.

Describe the ANFIS architecture that is standard:

The first layer's primary goal is to produce the node output that matches the membership values connected to the premise section:

$$O_i^1 = \mu_{A_i}(x), \quad i = 1{,}2 \tag{3}$$

In the second layer, each node takes on a fixed role denoted as π. These nodes perform a multiplication operation on the incoming signals:

$$O_i^2 = w_i = \mu_{A_i}(x) \times \mu_{B_i}(x), \quad i = 1{,}2 \tag{4}$$

In the third layer, every node is labeled as N and functions as a constant element for the purpose of normalization. Its role is to compute the proportion of the firing strength associated with the i-th rule in relation to the overall sum of firing strengths across all rules:

$$O_i^3 = \bar{w}_i = \frac{w_i}{w_1 + w_2}, \quad i = 1{,}2 \tag{5}$$

In the fourth layer, each node is adaptive in nature. These nodes perform a specific function, which is:

$$O_i^4 = \bar{w}_i \times f_i = \bar{w}_i(p_i x + q_i y + r_i), i = 1{,}2 \tag{6}$$

In this context, w_i signifies the output of layer 3, while p_i, q_i, $r_i$are the parameters associated with the first-order Sugeno rule. The network's output is computed using these parameters and the rules defined within the system.

$$O_i^5 = f = \sum_i \bar{w}_i \times f_i \tag{7}$$

To learn and update parameters in the ANFIS model, Jang's hybrid learning algorithm is utilized. This algorithm combines the least squares approach and the gradient descent method. The purpose of this algorithm is to update the linear consequent parameters in layer 4 and the nonlinear premise parameters in layer 1.

There are two paths that the algorithm takes: the forward path and the backward path. Recursive least square estimator (RLSE) technique is applied to update the consequent parameters in the fourth layer in the forward path, while the premise parameters in layer 1 stay fixed. The use of RLSE is motivated by the consequent parameters' linearity, which aims to accelerate convergence during the hybrid learning process.

On the other hand, the gradient descent algorithm is used in the backward path to update the premise parameters in layer 1 while keeping the consequent parameters constant. The ANFIS model's dual-path strategy maximizes the learning process for both kinds of parameters.

As part of the learning process, an error is generated that shows the difference between the desired and actual output and is then propagated back to the first layer. This backward path makes sure that the premise parameters are changed in accordance with the errors that are observed, which helps the model learn and become more refined.

2.2. The discrete event part modelling

A collection of systems with continuous dynamics interacting with one or more discrete event systems is typically used to model hybrid systems.

Many methods exist for modeling hybrid dynamic systems. What unites them is that distinct events have an impact on the ongoing evolution. Hybrid automata are one type of discrete part modeling tool used in hybrid dynamic systems [25].

Hybrid automata are frequently utilized in modeling and control applications due to their ability to accurately represent systems with both continuous and discrete behavior.

In the context of the study mentioned, where a hybrid dynamic system is being modeled, the choice of hybrid automata is likely driven by the necessity to capture the diverse aspects of the system's behavior [26].

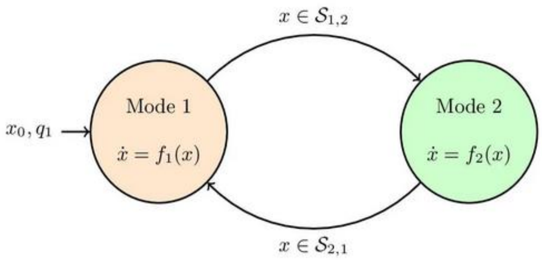

Fig. 2. Hybrid automata

2.2.1. Hybrid Automata

In a single formalism, differential equations for continuous change and transitions for discrete change are combined in a mathematical model for HDS known as a hybrid automaton. A hybrid automaton is a type of finite state machine that consists of a finite number of continuous variables, each represented by a set of ODEs for its value. A hybrid automaton is defined by the following tuple [27]:

$$G = (Q, \Sigma, X, flux, Init, \delta) \tag{8}$$

Q: is a representation of the system's hybrid model states;

The set of system events is denoted by Σ.

X: represents a limited collection of continuous variables that characterize the system's continuous dynamics;

$flux$: $Q \times X \rightarrow< n$ is a function that describes how X continuously evolves dynamically in each state q;

The system's state transition function is denoted by $\delta : Q \times \Sigma \rightarrow Q$.

A shift from state q to state $q+$ following the occurrence of a discrete event $e \in \Sigma$ is represented by a transition $\delta(q,e)=q+$;
The initial conditions are given by the set
$Init = (q1 \in Q, X(q1), flux(q1)$.

3. FAULT DETECTION AND EVALUATION

Making a diagnosis is frequently a difficult task, and traditional analytical techniques frequently fall short of offering workable solutions for design problems. For this reason, artificial intelligence techniques like fuzzy logic and neural networks are becoming more and more common in industrial diagnostic applications.

These techniques yield results that are simple to interpret and provide valuable data for the stage of decision-making. The diagnostic task consists of two phases: residual creation and decision-making (residuals evaluation).

By creating residuals, fault indicators can be obtained from the current inputs and outputs. The generation process is based on comparing the observed behavior of the system with the expected reference behavior (derived from model predictions). However, categorizing the defects that have been found through residue analysis is a step in the decision-making process.

Generally speaking, there should be almost no residue (no defects). In contrast, if there are flaws, the value of this residue won't be zero [27,28].

3.1. Model based residual generation

The need for a precise mathematical model stands out as a major limitation of analytical methods applied in the diagnostic field. Conventional fault diagnosis and isolation techniques "FDI" rely on a mathematical model, making them notably susceptible to modeling errors, variations in parameters, noise, and disturbances.

To address these challenges, it becomes essential to opt for FDI algorithms that are better suited for real systems, mitigating some of the limitations associated with relying solely on mathematical models.

Analytical models and residual creation are still based on the same fundamental idea. It entails contrasting the process's outputs with the estimators. To compute the estimates, a Neuro-Fuzzy model is used in this instance.

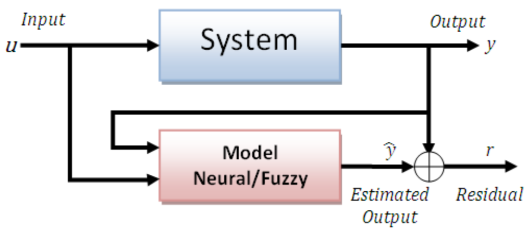

Fig. 3. Generation of residuals by the Neuro-Fuzzy model

The difference between the actuator output vector, *y(t)*, and the output vector from the Neuro-Fuzzy model, *ŷ(t)*, yields the residue vector, *r(t)*...

$$r(t) = y(t) - \hat{y}(t) \quad (9)$$

The construction of the Neuro-Fuzzy model involves several key steps:

3.1.1. Creation of a Database

Begin by offline compilation of a database based on expert knowledge. This database should encompass vital process characteristics such as operating points, stability, noise levels, etc. It's then split into two segments: a significant portion for learning purposes and the remainder for validation. This forms the foundation for further model development.

3.1.2. Choice of Model Structure

Selecting the appropriate structure for the Neuro-Fuzzy model is crucial. The NARX (Nonlinear AutoRegressive with eXogenous inputs) structure is commonly favored, especially for deterministic or minimally noisy systems. This choice helps circumvent stability issues that might arise in other structures like NNARMAX.

3.1.3. Training

Weights and biases are first determined at random and subsequently modified through the use of a learning algorithm that reduces quadratic error. The Levenberg-Marquardt algorithm is widely used for this in our context.

3.1.4. Validation

Following the training of the network, the ultimate weight and bias values are determined. An evaluation phase is crucial to verify that the network satisfies predefined criteria. Various tests are performed on the network, and if it falls short of expectations, adjustments may involve modifying the network structure (e.g., adjusting input or output orders) or increasing the learning phase's iteration count to get enough network parameter convergence.

3.2. Neuro-Fuzzy model based residual evaluation

The basis of fuzzy logic, also referred to as approximate reasoning, is the use of linguistic variables to express fuzzy rules in natural language. Fuzzy reasoning, fuzzy adaptive threshold, and fuzzy classification are some common forms that these fuzzy rules take. A lot of Fault Detection and Isolation (FDI) techniques use fuzzy logic, especially when evaluating residuals.

$If\ (x \in A) and\ (y \in B)\ Then (z \in C),$ With, $A, B\ and\ C$ fuzzy sets. (10)

3.2.1. Fuzzification

This procedure involves creating fuzzy membership functions for each input and output in order to convert raw data values into fuzzy input values. Every residue is given a membership function, which is typically represented by a triangle or trapezoid shape, indicating the degree of its involvement in a failure.

3.2.2. Inference

This phase serves as the foundation for establishing rules that discern fault conditions from non-faulty states within the system. For instance:

"If residue1 = 0 and residue2 = 0, then no failure is detected."

"If residue1 > 0 and residue2 < 0, then fault1 is detected."

Validation of these rules can be challenging if they don't align with an operator's experience or expertise.

3.2.3. Defuzzification

The inference sets' raw output values are produced at this step. This output goes beyond a binary declaration to indicate the level of fault presence in the system. It indicates both the scale of the existing fault and the confidence in its presence. Each considered fault receives such an output.

During this phase, raw output values are generated based on the inference sets. The output reflects the degree of fault presence in the system, surpassing a straightforward binary declaration [29] and [30].

4. FAULT TOLERANT CONTROL

Fault diagnosis is undoubtedly crucial, yet it shouldn't stand alone. Considering faults during the system control law's design phase is prudent. The significance of diagnosis and fault tolerance in automated systems is evident, aiming to:

- Enhance safety for people and systems.
- Optimize maintenance procedures.
- Enhance production quality and efficiency.

Over the years, fault tolerance has been approached from various perspectives, underscoring its crucial role in preventing catastrophic consequences and ensuring system safety. Main goal of fault tolerance techniques is to maintain accuracy and stability while keeping the system operating normally.

Preventing the propagation of faults that could lead to system-wide failure is of paramount importance. Therefore, fault tolerance holds a central position in the synthesis of control laws and system design. By reducing or eliminating the effects of faults, a fault-tolerant control law seeks to preserve stability and nominal system performance in the face of failures.

There are two primary types of fault-tolerant control systems: active and passive methods. The passive method uses strong control laws that show resistance to particular faults without the need for reconfiguration or detection schemes.

On the other hand, the active approach incorporates fault detection schemes and strategies for adjusting the control law while ensuring system stability and performance are maintained.

Within the active approach, two strategies, fault accommodation and system reconfiguration, are prominent. Accommodation involves fault estimation enabled by the Fault Detection and Identification (FDI) stage. If fault estimation isn't viable, system reconfiguration becomes essential, involving component subsets or replacing faulty elements [31], [32] and [33].

An online fault accommodation approach integrates fault diagnosis modules for tolerant control. This module, a bank of controllers, compensates for control law faults. It selects the appropriate control law from pre-calculated offline controllers, maintaining input/output relationships with the system.

These controllers vary in structure, including neuronal, fuzzy, or Neuro-Fuzzy designs. Depending on the process phase identified by the diagnostic step, a fuzzy mode selector picks the suitable control mode—adjusting control parameters or switching between controller structures—to meet performance requirements.

Figure (4) outlines a general diagram of fault-tolerant control utilizing neural networks and fuzzy logic.

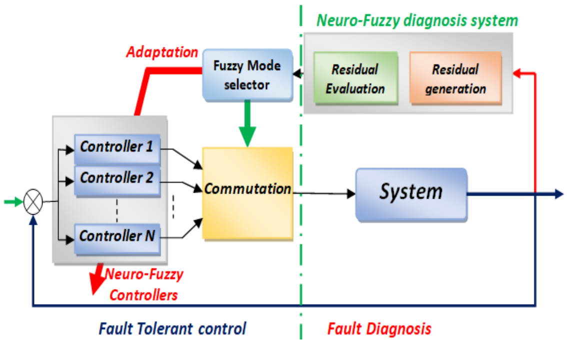

Fig. 4. General diagrams of Fault Tolerant Control using Neural Networks and Fuzzy Logic

This extensive and interdisciplinary research area encompasses various domains, including stability concerns, control methods, modeling techniques, and fault diagnosis for Systems with Discrete Dynamics (HDS). The underlying philosophy of these approaches predominantly centers on fault modeling and adjusting control laws based on the magnitudes of detected faults.

Regarding fault-tolerant control applied to SDH, only a restricted number of studies have been conducted [33],[34] and [35]. Furthermore, even in normal operation, a typical assumption for discrete control in these systems is the continuous awareness of the system's current mode.

However, this assumption can be quite stringent and might not always hold in industrial setups. It necessitates abundant, efficient, and sometimes costly sensor instrumentation. The determination of the current mode becomes an additional functionality that the monitoring software layer must provide.

Figure (5) illustrates a typical diagram of fault-tolerant control systems, comprising four primary components:

- A real-time fault diagnosis block that provides information.
- A reconfiguration mechanism.
- A reconfigurable regulator.
- A reference applied to the system.

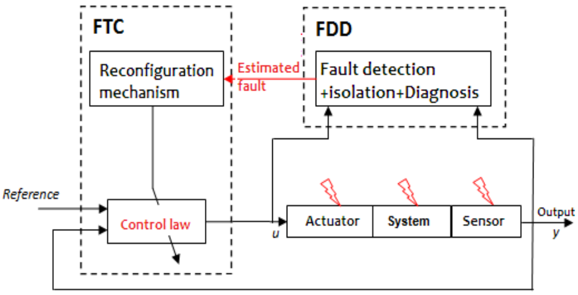

Fig. 5. Principle of an active FTC control law

5. APPLICATION

The hydraulic system, depicted in Figure (6), comprises two cylindrical tanks with identical cross-sectional areas $S = 0.0154\, m^2$. These tanks are connected by pipes C_2, C_3, positioned at levels $b = 0\, m$ and $h = 0.5\, m$ respectively. Valves V_1 and V_4, attached to pipes C_1 and C_4, facilitate liquid evacuation for usage. Pipes C_2 and C_3 are equipped with valves V_2 and V_3.

The system utilizes pump P_1 to regulate flow Q_p affecting the level of tank 1. Level sensors monitor the levels h_1 and h_2 in the respective tanks.

For simplicity in analysis, valves V_1 and V_2 and V_3 are assumed to remain open consistently. Additionally, the pump operates in an on-off manner to maintain h_2 within a predetermined range.

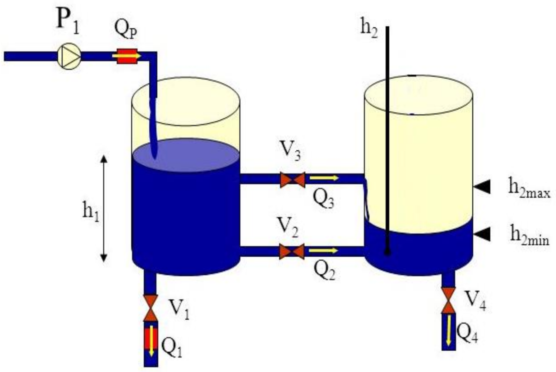

Fig. 6. Two tank system

The pump generates zero flow when it's not operational. However, when activated, it produces a flow denoted as $Q_p = Q_0 = 0.001\, m^3/h$. The pump logic is as follows:

- The pump is initially on.
- It is turned off when $h_2 \geq 0.2m$.
- It starting when $h_2 \leq 0.1m$.

The valve V_4 is manually operated, allowing users to open or close it as needed. The system considers two discrete states for C_3 pipe: "Empty" (denoted as V) or "Full" (denoted as P), and for valve V_4: "Open" (O) or "Closed" (F). Therefore, four distinct modes define the system's behavior. Each mode is characterized by discrete states (the state of pipe C_3 and valve V_4, specific state equations, and inequality constraints. Torricelli's law provides expressions for the flows:

$$\begin{cases} Q_1(t) = A_1.\sqrt{2g.h_1(t)} \\ Q_2(t) = A_2.sign(h_1(t) - h_2(t)).\sqrt{2.g.|h_1(t) - h_2(t)|} \\ Q_4(t) = A_4.\sqrt{2.g.h_2(t)} \end{cases} \tag{11}$$

The pipe sections $C_i (i = 1, \dots, 4)$,
$g = 9.81\, m/s^2$, $A_1 = \cdots = A_4 = 3.6 \times 10^{-5} m^2$.
Q_3: Three expressions can be provided based on the liquid level in the tanks:

$$Q_3 = \begin{cases} A_3.\sqrt{2g.(h_1(t) - h(t))}, & if\ h_1 \geq h\ et\ h_2 < h \\ -A_3.\sqrt{2g.(h_2(t) - h(t))}, & if\ h_1 < h\ et\ h_2 > h \\ A_3.sign(h_1(t) - h_2(t)).\sqrt{2.g.|h_1(t) - h_2(t)|}, & if\ h_1 \geq h\ et\ h_2 > h \end{cases} \tag{12}$$

To simplify the writing, we rewrite $Q_3(t)$ with the following expression:

$$Q_3(t) = B.\sqrt{2.g.|H_1(h_1) - H_2(h_2)|} \tag{13}$$

H_1, H_2 are functions of h_1 and h_2 respectively:

$$H_1(h_1) = \begin{cases} 0 & if\ h_1 < h \\ h_1 - h & if\ h_1 \geq h \end{cases}; \tag{14}$$

$$H_2(h_2) = \begin{cases} 0 & if\ h_2 < h \\ h_2 - h & if\ h_2 \geq h \end{cases}; \tag{15}$$

$$B = A_3.sign(H_1(h_1) - H_2(h_2)) \tag{16}$$

Expressions flows become:

$$\begin{cases} Q_1 = A.\sqrt{2g}.\sqrt{h_1} \\ Q_2 = A.\sqrt{2g}.\sqrt{|h_1 - h_2|} \\ Q_3 = A.\sqrt{2g}.\sqrt{|h_1 - h|} \\ Q_4 = A.\sqrt{2g}.\sqrt{h_2} \end{cases} \tag{17}$$

This system encompasses two event types:

1. Controlled Events: are connected to the ON/OFF commands of the valves in this context. More specifically, the opening and closing of valve V_4 at times t = 240 s and t = 380 s, respectively, are represented by e_1 and e_2.

2. Spontaneous Events: These events are internally triggered. They appear when the water levels in tanks h_1 and h_2 exceed or fall below certain thresholds.

The pump initiates when $h_2 = h_{2min} = 0.1m$ and halts when $h_2 = h_{2max} = 0.2m$.

To construct a diagnostic model for this hybrid system, a modelling approach using a hybrid automaton will be presented. Neuro-fuzzy models are employed to generate fault indicators. This strategy is specifically utilized in scenarios such as diagnosing hydraulic systems. The proposed Fault Detection and Identification (FDI) technique is outlined in Figure 7.

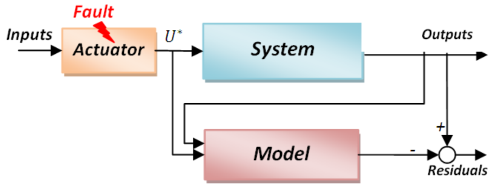

Fig. 7. Residuals generation

The following figure shows the hybrid automata that represent the system under typical operating conditions:

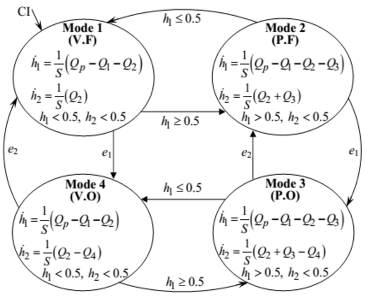

Fig. 8. Hybrid Automata

For a total of 500 s, the simulation is run with the following starting conditions: For $h_1$andh_2, the liquid levels are shown in the following figure. The following starting conditions are established, and the simulation has a 500 s total simulation time.

The simulation is run with the following initial conditions for a total simulation time of 500 s: The next figure provides the liquid levels for.
The simulation has 500 s of total simulation time, and the following initial conditions are set: $h_{1,0} = 0.4\ m\ and\ h_{2,0} = 0\ m.$ The liquid levels h_1 and h_2 are illustrated in the figure below:

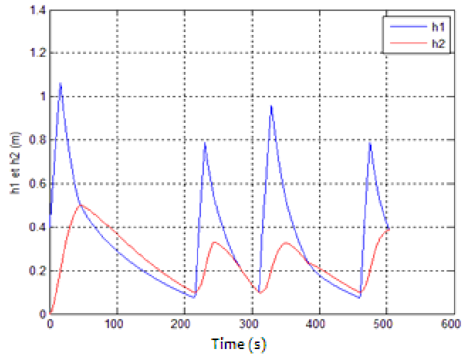

Fig. 10. The evolution of the levels $h_1\ et\ h_2$

The chronogram of the modes, depicting the evolution over time, is presented in the following figure:

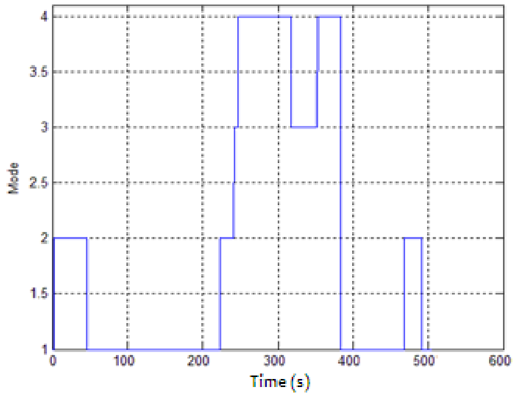

Fig. 11. Modes Evolution

5.1. Modelling of the system by ANFIS

The techniques developed in this study are applicable to multivariable systems that can be modeled using neural network-based models and fuzzy inference systems. We utilized Neuro-Fuzzy modeling to effectively capture and address a wide

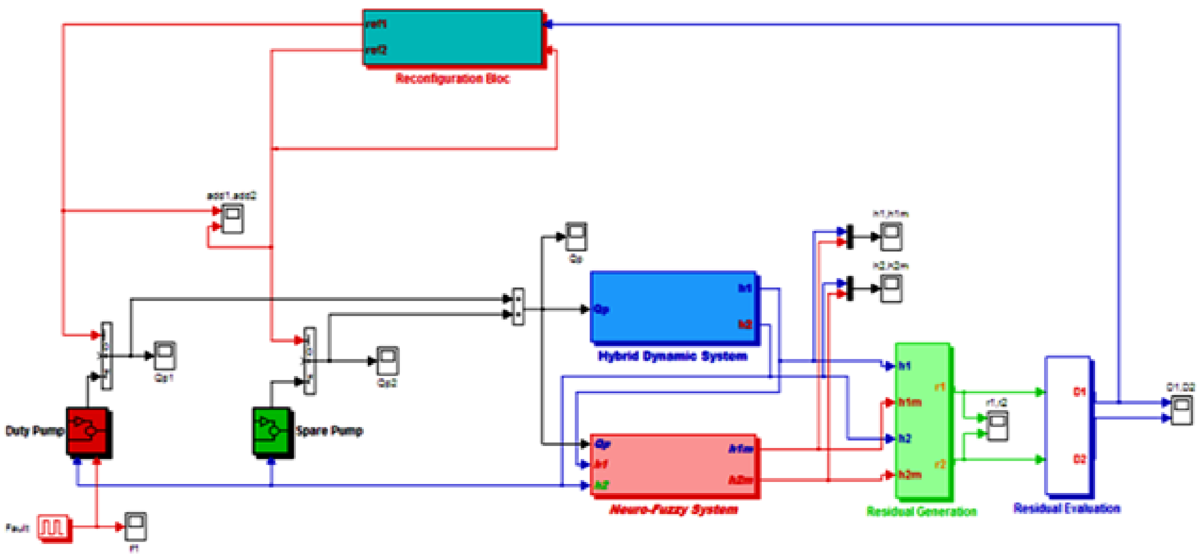

Fig. 9. Simulation model

range of model uncertainties, as well as abnormalities in parameter variations caused by system faults.

Selecting model architecture often involves considering the system's functioning and structure. After conducting various tests, we opted for a model comprising two ANFIS networks:

$$\hat{h}_1(k) = F_1\big(Q_p(k-1), Q_p(k-2), h_1(k-1), h_1(k-2)\big) \tag{18}$$

$$\hat{h}_2(k) = F_2\big(Q_p(k-1), Q_p(k-2), h_2(k-1), h_2(k-2)\big) \tag{19}$$

Q_p: The system input,
h_1 : The system output,
h_2 : The system output,
$\hat{h}_1$: The estimated output of h_1,
$\hat{h}_2$: The estimated output of h_2.

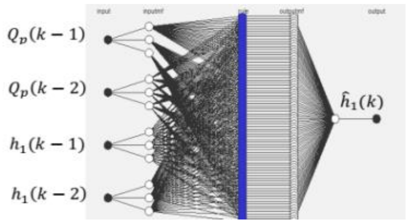

Fig. 12. ANFIS 1 Network ($\hat{h}_1$)

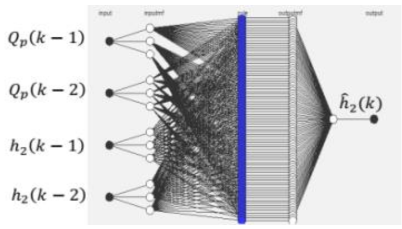

Fig. 13. ANFIS 2 Network ($\hat{h}_2$)

Note: The system simulation was conducted with a simulation step size of Te=10 milliseconds. Therefore, all figures are in terms of the discrete time $k = t/Te$.

Initially, we conducted several experiments (in the absence of faults) to test the reliability of the system. The residues are given in Figure (14) (if there are no defects).

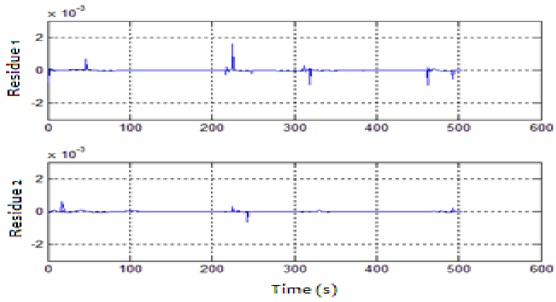

Fig. 14. Modeling errors (residuals: No defect)

In a Hybrid Dynamic System, a residue should ideally equate to zero when no failures are present. Following the modeling phase, which involves generating residuals, the subsequent step involves residue assessment. This evaluation phase aims to appraise the residuals generated and their values.

After the modeling step, specifically after generating the residuals, the next step is the evaluation of these residuals. They are utilized to diagnose the faults present within the system and provide relevant information for conducting the diagnosis.

In order to demonstrate the effectiveness of the proposed Neuro-Fuzzy model, we investigated two scenarios:

1. In the first scenario, we initially demonstrate the effectiveness of the diagnostic system (without reconfiguration). Subsequently, we present the proposed fault-tolerant control strategy.
2. The simulation results are obtained from experiments conducted on the hydraulic process, which are illustrated in the parts below.

5.1. Fault Diagnosis strategy

The diagnostic approach introduced in this research relies on real-time fault estimation, facilitating direct fault diagnosis such as the isolation and identification of detected faults.

This method enables the identification of all faults, including simultaneous ones, and provides immediate insights into the nature and severity of each fault.

Fault scenarios with the prefix "f" are taken into account to illustrate the recommended methodology and confirm the reliability and effectiveness of the diagnostic system.

For example, one false fault scenario suggests a decrease in pump actuator efficiency. Such a system failure has an instantaneous effect on the differential equations that govern the system's dynamics.

$$\begin{cases} S.\dot{h}_1 = Q_p - Q_1 - Q_2 - Q_3 - f \\ \quad S.\dot{h}_2 = Q_2 + Q_3 - Q_4 \end{cases} \tag{20}$$

The proposed method classifies defects using a fuzzy reasoning model. Three membership functions, two trapezoidal and one triangular, have been selected for each residue. Setting the parameters for these functions required extensive testing that included a range of flaws.

Table 1. Membership functions of the residuals

	Residual 1	Residual 2
N	$[-1\ \ -1\ \ -0.011\ \ -0.011]$	$[-1\ \ -1\ \ -0.0021\ \ -0.0021]$
Z	$[-0.01\ \ -0.01\ 0.01\ 0.01]$	$[-0.002\ \ -0.002\ 0.002\ 0.002]$
P	$[0.011\ 0.011\ 1\ 1]$	$[0.0021\ 0.0021\ 1\ 1]$

One simulated scenario involves a 20% efficiency loss in the first actuator input, initiated at time 3100 and lasting until 3400. Subsequently, we observe the system's output, associated residues, and the decision model produces outputs to evaluate the impact of this fault on the system.

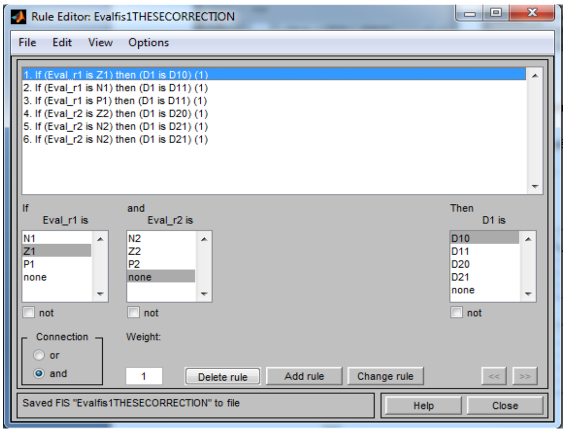

Fig. 15. Fuzzification of the residuals

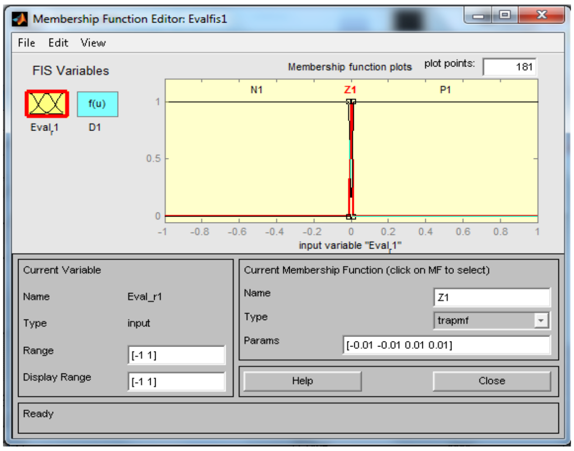

Fig. 16. Inference (Fuzzy Rules)

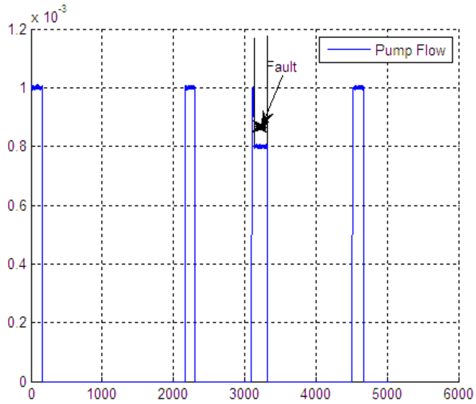

Fig. 17. Evolution of Pump Flow with fault

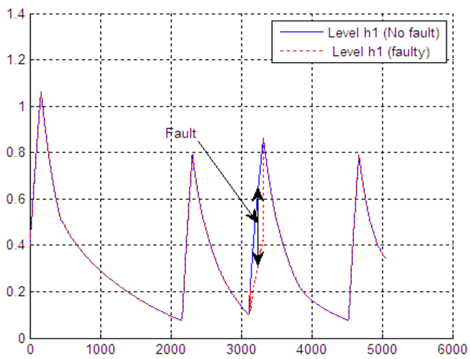

Fig. 18. The evolution of the level h_1 and estimated level h_1 in diagnosis strategy

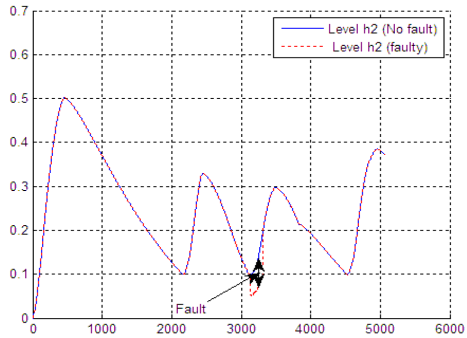

Fig. 19. The evolution of the level h_2 and estimated level h_2 in diagnosis strategy

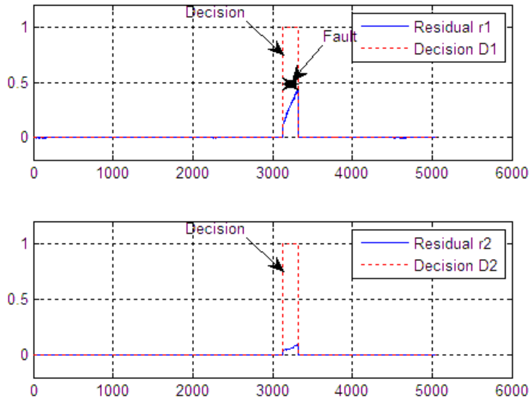

Fig. 20. The evolution of residuals and decisions responses in diagnosis strategy

Figures 17 to 19 depict the trends of pump flow, tank levels, residuals, and decisions, showcasing a decline in input and output values from time 3100 until 3400.

Figure 20 illustrates that the residue outputs remain steady at zero until time 3100. Subsequently, due to the fault introduced in the system's input until instant 3400, the residue values change. Additionally, the evolution of decisions reflects a consistent value of one (1) during this period. Notably, the diagnostic system consistently generates positive decisions when faults influence the system input.

Numerous additional experiments were conducted, consistently affirming the capability of this diagnostic approach to identify and acknowledge all possible faults affecting the system.

5.2. Fault Tolerant Control strategy

In contrast, the proposed control strategy adopts a cooperative framework that integrates the nominal control system, the diagnostic module, and the regulators responsible for ensuring fault tolerance. This integrated approach enhances system robustness by effectively coordinating control and diagnostic functionalities within a single structure.

To confirm the fault-tolerant control system's dependability and efficiency, we ran a test that simulated a 20% reduction in efficiency in the first actuator's input from instant 3100 to 3400. To evaluate the impact, we then looked at the system outputs, related residues, and decision model outputs.

The goal of the suggested fault-tolerant control approach is to keep the system operating at the targeted performance levels. In order to make up for the drop in pump efficiency, a spare pump has been added. The plan calls for quickly shutting down the malfunctioning pump and turning on the backup pump to maintain system functionality.

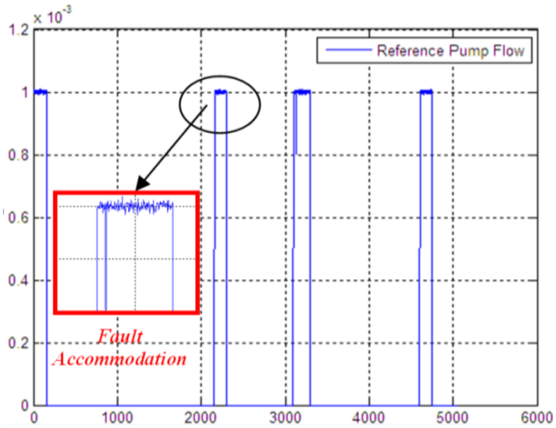

Fig. 21. Evolution of Pump Flow with FTC strategy

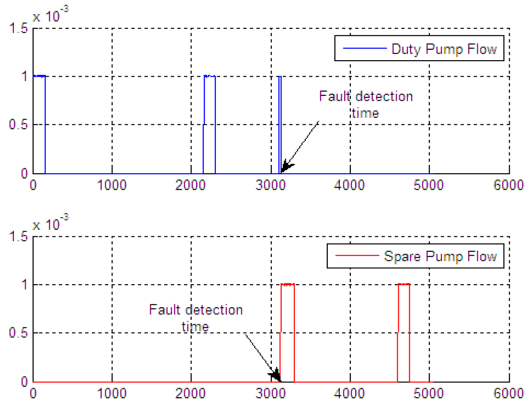

Fig. 22. Evolution of Pumps Flow with FTC strategy

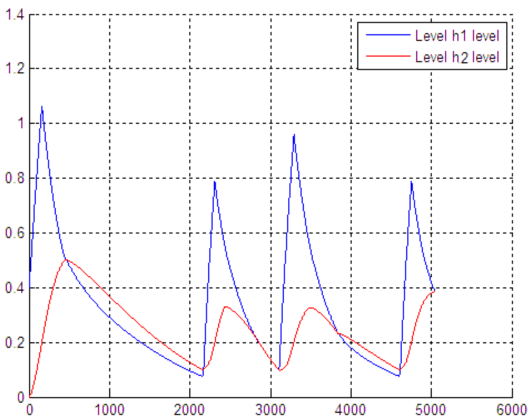

Fig. 23. The evolution of the levels h_1 *et* h_2 with FTC strategy

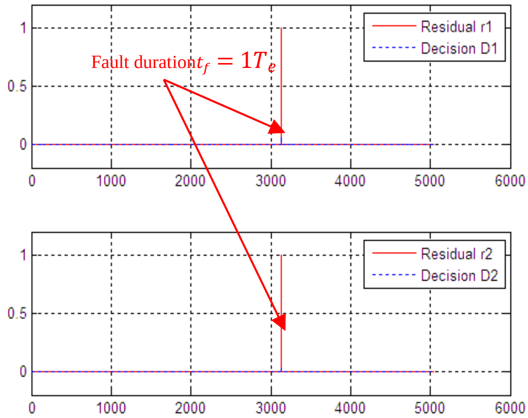

Fig. 24. The evolution of residuals and decisions responses with FTC strategy

The employed fault-tolerant control strategy showcases excellent results, ensuring the system's outputs remain unaffected by the input fault. Within the decision system, there's a minor deviation followed by a swift return to zero. This brief delay, termed as the detection time$T_d = 1.T_e$, signifies the activation of the tolerant control reconfiguration mechanism after one sample time of the fault's onset. This duration typically aligns with the system reaching its steady state.

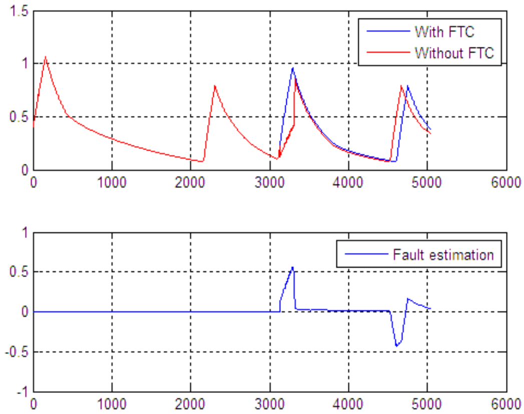

Fig. 25. The evolution of the level $h_1$and estimated level h_1 with FTC strategy

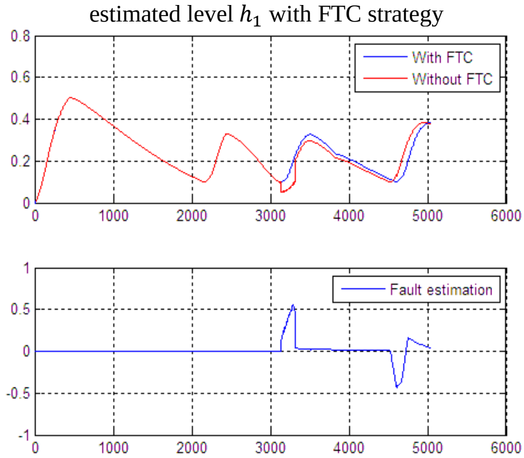

Fig. 26. The evolution of the level $h_2$and estimated level $h_2$with FTC strategy

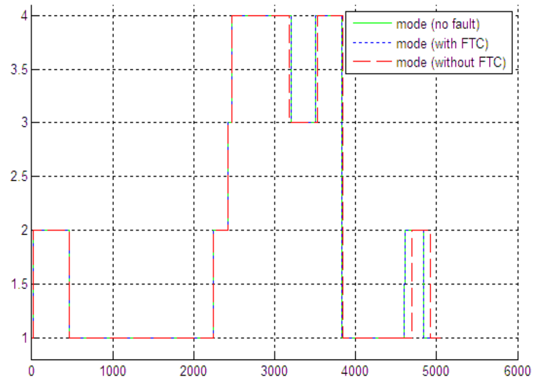

Fig. 27. Evolution of modes in: no fault case, Diagnosis case, FTC case

Figures 25 to 27 offer a comparison between two strategies: the diagnosis strategy (without reconfiguration) and the fault-tolerant control strategy (with reconfiguration). These illustrations highlight the effectiveness of the fault-tolerant control approach, showcasing its ability to maintain system stability and functionality despite faults.

The comparison between the diagnosis strategy and the fault-tolerant control strategy reveals significant differences in output behaviour. While the diagnosis strategy leads to gradual divergence of outputs from the nominal values, the fault-tolerant control strategy ensures that all outputs remain unchanged. These findings highlight the critical function and importance of using fault-tolerant control strategies.

The purpose of this simulation was to assess how actuator faults that might impair system functionality would compensate. The outcomes of the simulation validate the desired properties of fault-tolerant control in nullifying the detrimental effects associated with the occurrence of one or more faults within the system.

6. CONCLUSION

The primary objective of our work is to demonstrate that the diagnostic method presented herein relies on the online estimation of faults. This enables direct diagnosis, involving the isolation and identification of detected faults. It facilitates the isolation of all faults while providing immediate insights into the fault's nature and its severity (magnitude). Concurrently, the proposed control strategy operates through a cooperative framework that integrates the nominal control system, diagnostic module, and responsible regulators to ensure fault tolerance.

The fault-tolerant control strategy involves three pivotal steps. First, a modeling method utilizing Neuro-Fuzzy systems and hybrid automata is employed to represent Hybrid Dynamic System dynamics, encompassing both continuous and discrete aspects. The second step involves generating residuals used in residue evaluation, employing a fuzzy reasoning model to classify detected defects. Finally, the third step selects pre-established Neuro Fuzzy control laws, introducing fault-tolerant control capable of compensating for faults by integrating a compensation term. The proposed tolerant control laws assume a perfect estimation of actuator faults.

This application demonstrates the usefulness of Neuro-Fuzzy models in managing complex system dynamics by demonstrating their suitability for fault-tolerant control as well as diagnosis of hybrid dynamic systems.

Source of funding: *This research received no external funding.*

Author contributions: *research concept and design, M.S.A., B.R., M.A.M., M.R.K., S.K.; Collection and/or assembly of data, M.S.A., B.R., M.A.M., M.R.K., S.K.; Data analysis and interpretation, M.S.A., B.R., M.A.M., B.B., M.R.K., S.K.; Writing the article, M.S.A., B.R., M.A.M., M.R.K., S.K.; Critical revision of the article, M.S.A., B.R., M.A.M., B.B., M.R.K., S.K.; Final approval of the article, M.S.A., B.R., M.A.M., B.B., M.R.K., S.K.*

Declaration of competing interest: *The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.*

REFERENCES

1. Schaft A, Schumacher H. An introduction to hybrid dynamical systems. Springer 2000; 251.
2. Kechida S, Cocquempot V. Méthodologies de diagnostic des systèmes dynamiques: Théories et exemples. EUE, 2012.
3. Goebel R, Sanfelice RG, Teel AR. Hybrid dynamical systems. IEEE control systems magazine 2009; 29(2): 28-93.
4. Tomlin CJ, Sastry S. Computing piecewise affine controllers for hybrid systems. Automatica 2000; 36(3): 343-361.
5. Cassandras CG, Lafortune S. Introduction to discrete-event systems. Springer Science & Business Media 1999.
6. Mhamdi L, Said AM, Dhouibi H, Kechida S. Diagnosis of hybrid systems through bond graph, observers and timed automata. Diagnostyka 2020; 21(3): 113–25. https://doi.org/10.29354/diag/126444.
7. Said AM, Sihem K. Methodology for monitoring and diagnosing faults of hybrid dynamic systems: a case study on a desalination plant. Diagnostyka 2020;21(1): 27–33. https://doi.org/10.29354/diag/116076.
8. Sayed-Mouchaweh M. Fault diagnosis of hybrid dynamic and complex systems. Springer International Publishing 2018. https://doi.org/10.1007/978-3-319-74014-0.
9. Achbi MS, Kechida S, Mhamdi L, Dhouibi H. A neural-fuzzy approach for fault diagnosis of hybrid dynamical systems: demonstration on three-tank

system. Acta Mechanica et Automatica 2021; 15(1): 1–8. https://doi.org/10.2478/ama-2021-0001.
10. Mhamdi L, Said AM, Dhouibi H, Kechida S. Diagnosis of hybrid systems through bond graph, observers and timed automata. Diagnostyka 2020; 21(3): 113–25. https://doi.org/10.29354/diag/126444.
11. Bristol E. On a new measure of interaction for multivariableprocess control. IEEE Trans. Automat. Contr 1966; 11(1): 133–134. https://doi.org/10.1109/TAC.1966.1098266.
12. Dimirovski GM, Barnett S, Kleftouris DN, Gough NE. An inputoutput package for MIMO nonlinearcontrol systems. IFAC Proceedings Volumes 1979; 12(3): 265–273. https://doi.org/10.1016/S14746670(17)658130
13. Salwa Y, Saida B, Abderrahim K. Diagnosis and fault tolerant control against actuator fault for a class of hybrid dynamic systems. Measurement and Control 2023; 56(7–8): 1240–50. https://doi.org/10.1177/00202940221143584.
14. Zahaf A, Bououden S, Chadli M, Chemachema M. Robust fault tolerant optimal predictive control of hybrid actuators with time-varying delay for industrial robot arm. Asian Journal of Control 2022; 24(1): 1–15. https://doi.org/10.1002/asjc.2444.
15. Rouabah B, Toubakh H, Djemai M, Ben-Brahim L, Kafi MR. New active fault tolerant control of multicellular converter. 2022 7th International Conference on Environment Friendly Energies and Applications (EFEA) 2022; 1–6. https://doi.org/10.1109/EFEA56675.2022.10063825.
16. Rouabah B, Toubakh H, Djemai M, Ben-Brahim L, Ghandour R. Fault diagnosis based machine learning and fault tolerant control of multicellular converter used in photovoltaic water pumping system. IEEE Access 2023; 11: 39013–23. https://doi.org/10.1109/ACCESS.2023.3266522.
17. Van Der Schaft AJ, Schumacher H. An introduction to hybrid dynamical systems. Springer 2007; 251.
18. Engell S, Frehse G, Schnieder E. Modelling, analysis and design of hybrid systems. Springer 2003; 279.
19. Nadjm-Tehrani, Simin, Stromberg JE. Modelling and verification of hybrid systems: A case study. 1993 4th Annual Conference on AI, Simulation and Planning in High Autonomy Systems. IEEE 1993.
20. Yang H, Mao ZH, Jiang B. Model-based fault tolerant control for hybrid dynamic systems with sensor faults. Acta Automatica Sinica 2006; 32: 680–5.
21. Rajaee T, Khani S, Ravansalar M. Artificial intelligence-based single and hybrid models for prediction of water quality in rivers: A review. Chemometrics and Intelligent Laboratory Systems 2020;200: 103978. https://doi.org/10.1016/j.chemolab.2020.103978.
22. Zhang C, Lu Y. Study on artificial intelligence: the state of the art and future prospects. Journal of Industrial Information Integration 2021; 23: 100224. https://doi.org/10.1016/j.jii.2021.100224.
23. Neu D, Lahann J, Fettke P. A systematic literature review on state-of-the-art deep learning methods for process prediction. 2021
24. Jang JS. ANFIS adaptive-network-based fuzzy inference system. systems, man and cybernetics, IEEE Transactions on 1993; 23: 665–85. https://doi.org/10.1109/21.256541.
25. Raskin JF. an introduction to hybrid automata. 2005; 491–517. https://doi.org/10.1007/0-8176-4404-0_21.
26. Bak S, Beg OA, Bogomolov S, Johnson TT, Nguyen LV, Schilling C. Hybrid automata: from verification to implementation. International Journal on Software Tools for Technology Transfer 2019; 21(1): 87–104. https://doi.org/10.1007/s10009-017-0458-1.
27. Alur R, Courcoubetis C, Henzinger TA, Ho PH. Hybrid automata: An algorithmic approach to the specification and verification of hybrid systems. Hybrid Systems 1993; 209–29. https://doi.org/10.1007/3-540-57318-6_30.
28. Vento J, Travé-Massuyès L, Puig V, Sarrate R. An incremental hybrid system diagnoser automaton enhanced by discernibility properties. IEEE Transactions on Systems, Man, and Cybernetics: Systems 2015; 45: 788–804. https://doi.org/10.1109/TSMC.2014.2375158.
29. Achbi MS, Mhamdi L, Kechida S, Dhouibi H. Methodology to knowledge discovery for fault diagnosis of hybrid dynamical systems: demonstration on two tanks system. Diagnostyka 2020; 21(4): 115–23. https://doi.org/10.29354/diag/130617.
30. Zouari T. Diagnostic des systèmes dynamiques hybrides à modes non linéaires. Doctoral dissertation, Lille 1, France 2013.
31. Yang H, Jiang B, Cocquempot V. Fault tolerant control and hybrid systems. fault tolerant control Design for Hybrid Systems 2010; 1–9. https://doi.org/10.1007/978-3-642-10681-1_1.
32. Jiang B, Yang H, Cocquempot V. Results and perspectives on fault tolerant control for a class of hybrid systems. International Journal of Control 2011; 84: 396–411. https://doi.org/10.1080/00207179.2011.557395.
33. Amin AA, Hasan K. A Review of fault tolerant control systems: advancements and applications. Measurement 2019; 143: 58–68. https://doi.org/10.1016/j.measurement.2019.04.083.

Mohammed Said ACHBI is a Doctor in Industrial Automation, Doctor in Department of electronic and communications, Kasdi Merbah Ouargla University, Algeria. His research activities deal mainly with Fault Diagnosis, Fault tolerant control of Hybrid Dynamical Systems and industrial supervision. He is interested also in Artificial Intelligence. Team Research, Diagnostic et Sûreté de Fonctionnement at Laboratoire d'automatique et informatique de Guelma.
E-mail: achbi.mohammedsaid@univ-guelma.dz

Boubakeur ROUABAH Received his Engineer from University of Bordj Bou Arriridj in 2009, Magister from university of Setif, Algeria in 2012, and PhD from university of setif-1 algeria in 2021. In 2020 until now, he was a researcher at University of Kasdi Merbah Ouargla, Algeria.
E-mail : rouabah.boubakeur@univ-ouargla.dz

Mohamed Abdelbasset MAHBOUB
Received his Engineer from University of setif in 2009, Magister from university of Setif, Algeria in 2012, and PhD from university of Batna algeria in 2017. In 2019 until now, he was a researcher at University of Kasdi Merbah Ouargla, Algeria.
E-mail : mahboub.abdelbasset@univ-ouargla.dz

Bilal BENARABI
received the Ph.D. degree in microwave electronics from the University of Saint-Etienne, France in 2016 & the Master's degree in optoelectronics and microwaves from University of Montpellier, France in 2011. His research interests include microwave measurements, electromagnetic and the microwave characterization of materials.
E-mail : benarabi.bilal@univ-ouargla.dz

Mohamed Redouane Kafi
is a Professor at Department of electronic and communications Université Kasdi Merbah Ouargla. Kafi does research in Aeronautical Engineering, Control Systems Engineering and Electronic Engineering.

E-mail : kafi.redouane@univ-ouargla.dz

Sihem KECHIDA
is a Professor in Industrial Automation, Professor at Guelma University, Algeria. Team Research Leader Diagnostic et Sûreté de Fonctionnement at Laboratoire d'automatique et informatique de Guelma.
She is the supervisor of many PhD Students and she is the coordinator of several industrial research projects within the applied automatic diagnostics and reliability of industrial systems. She also works on Fault Detection and Isolation (FDI) for Hybrid Dynamical Systems and Transportation Systems. She is active as an expert in several national and internationals committees and collaboration research activities. She has participated in several international research projects and has led several national research projects.
E-mail : kechida.sihem@univ-guelma.dz