Unsupervised Fault Detection in a Controlled Conical Tank

Joaquín Ortega[1], Camilo Ramírez[2], Tomás Rojas[3], Ferhat Tamssaouet[4], Marcos Orchard[5], Jorge Silva[6]

[1,2,3,5,6] *Information and Decision Systems Group, University of Chile, Santiago, 8370451, Chile*
joaquin.ortega@ug.uchile.cl
camilo.ramirez@ug.uchile.cl
tomas.rojas.c@ug.uchile.cl
morchard@ing.uchile.cl
josilva@ing.uchile.cl

[4] *PROMES-CNRS, UPVD, Rambla de la Thermodynamique, Tecnosud, Perpignan, 66100, France*
ferhat.tamssaouet@univ-perp.fr

Abstract

Current trends in the Industrial Internet of Things (IIoT) have increased the sensorization of systems, thus increasing data availability to apply data-driven fault detection and diagnosis techniques to monitor these systems. In this work, we show the capabilities of an information-driven method for detecting and quantifying faults in a subsystem common among a broad range of industries: the conical tank. Our main experiment consists of using a simple black-box model (multi-layer perceptron – MLP) to capture the dynamics of a PID-controlled conical tank built in Simulink and then induce pump failures of different severities; the derived data-driven indicators that we developed increase with the severity of the fault validating its usefulness in this controlled setting. A complementary experiment is carried out to enrich our analysis; this consists of simulating an open-loop discrete-time version of the conical tank to explore a range of fault severity and analyze the distribution of the indicators across this range. All our results show the applicability of the data-driven fault monitoring method in conical tanks subjected to either open- or closed-loop operation.

1. Introduction

Fault detection and identification (FDI) and fault diagnosis are essential elements in ensuring the reliability and safety of systems, including those used in industrial processes such as conical tanks. FDI involves recognizing the presence of faults in a system, whereas fault diagnosis goes a step further by determining their location and nature (Abid, Khan, & Iqbal, 2021). These tasks are essential, as undetected faults can lead to system failure, reduced efficiency, safety risks, and increased operating costs. Early detection and accurate diagnosis of faults can prevent these problems and ensure continuous and safe operation. Common methods for FDI and fault diagnosis include model-based approaches, signal-processing techniques, and data-driven methods. Model-based approaches use mathematical models to detect deviations, signal processing analyzes output signals for anomalies, and data-driven methods leverage machine learning and historical data to identify patterns and correlations (Abid et al., 2021).

The majority of the works in the literature assume open-loop systems when identifying faults. Indeed, closed-loop control can degrade FDI and fault diagnosis performance. This is because the system's robustness can mask early or minor faults, lowering detection rates. Additionally, the feedback mechanism can cause faults to propagate and couple within the system, making fault identification more challenging (Sun, Wang, He, Zhou, & Gu, 2019; Talebi & Khorasani, 2012; Costa, Angelov, & Guedes, 2015). This raises a caveat, as, in most industrial and real-world settings, systems are subjected to a control loop.

Several strategies were developed to deal with system degradation. These strategies can be related to fault mitigation or failure prevention. In fault mitigation, the failure is taken into account in the design stage, which tends to increase the fault resilience of systems. Hence, domains such as system reconfiguration, fault tolerance (Amin & Hasan, 2019), self-repairing systems (Yang & Kwak, 2022), and self-healing (Ghosh, Sharman, Rao, & Upadhyaya, 2007) can be gathered under the name of fault mitigation. Despite the outstanding achievements of fault mitigation, failures cannot be eliminated; therefore, it is necessary to consider them as unavoidable events that have to be prevented. In practice, failure prevention can be performed through preventive maintenance policies, which ensure system safety and availability, and through usage profile adjustment to slow down degradation rates (Thuillier, Jha, Le Martelot, & Theilliol, 2024; Patel & Shah, 2019).

Among systems used in the process industry, there are conical tanks, which are commonly used due to their advantages in mixing, stirring, efficient cleaning, and ensuring complete drainage of contents (Vavilala, Thirumavalavan, & Chandrasekaran, 2020; Ramanathan, Mangla, & Satpathy, 2018). However, their conical shape leads to nonlinear dynamics, posing challenges in controlling the relevant variables for the various processes in which they are involved. There exists a vast corpus of literature addressing the control aspect of conical tanks with many different techniques, such as fractional order control (Vavilala et al., 2020), predictive control (Srinivasan, Sindhiya, & Devassy, 2016), and reinforcement learning (Ramanathan et al., 2018), to name a few.

Since the last century, there has been an increasing concern about engineering systems driven by the vast amounts of data they generate (Jieyang et al., 2023). Conical tanks, which are crucial pieces of equipment, have also attracted interest in this context, motivating Passive Fault-Tolerant Control System (PFTCS) schemes to control them (Patel & Shah, 2019). Although there are publications addressing fault-tolerant control, as stated above, we consider there is a gap in understanding and quantifying the faults in the context of this particular system; in turn, the work here opens the possibility for new Active Fault-Tolerant Control System (AFTCS) methods.

1.1. Our Contribution

In this work, we describe a method for detecting faults and indicators for quantifying them in conical tanks subjected to a closed-loop proportional-integral-derivative (PID) controller. Our methodology uses Mutual Information (MI) to detect faults. Although MI has previously been applied in the framework of FDI, it is typically used for feature selection (Lucke, Mei, Stief, Chioua, & Thornhill, 2019; Yin & Yan, 2019; Chen, Wang, Li, & Yang, 2024) or assessing statistical dependence with labeled faults (Lucke et al., 2019). In contrast, we use MI estimates themselves as fault indicators. This approach does not require labeled faulty data, making it an unsupervised approach in this regard and setting it apart from existing methods. We apply it specifically to conical tanks, which are the focus of this study, without a thorough modeling of the system. Moreover, we highlight how it is possible to enrich the fault analysis, in the case of having a system model, by emulating unobserved faults.

A key advantage of our method is its flexibility. For plants with existing operations, the model we introduce in this work can be seamlessly replaced by a more complex pre-existing model, such as one designed to improve fault isolability (Düştegör, Frisk, Cocquempot, Krysander, & Staroswiecki, 2006). This adaptability allows for the method to be easily replaced or upgraded without disrupting ongoing processes, offering a practical solution for both system diagnosis and enhancement.

1.2. Paper Outline

We start our work by introducing the model of the plant we will explore, as well as definitions, intuitions, and the methodology that will be used in the rest of the work, in Section 2. In Section 3, we discuss the black-box model (MLP) we used to monitor our system and show its corresponding results. We do the same for our explicit analytical model of the plant and show the results from this complementary analysis in Section 4. Finally, we give our conclusions and views on how to extend our work in Section 5.

2. PRELIMINARIES

In this section, we describe the system focused on our study – the controlled conical tank, as described in (Jáuregui, 2016) – and the fault detection methodology introduced in (Ramírez, Silva, Tamssaouet, Rojas, & Orchard, 2024) that we adapt to the mentioned setting.

2.1. The Controlled Conical Tank

This work focuses on a conical tank filled with water. A pump, positioned at a height of $20\ \mathrm{cm}$ from the bottom of the tank, introduces water into the tank while water is drained from the bottom, as depicted in Figure 1. As we do not have sufficient real data, we use a model developed in (Jáuregui, 2016), which has been experimentally validated to describe accurately the physics of the system under study. This approach, which can deviate from real-world conditions, has often been used in the literature to compensate for the lack of data (Raval, Patel, & Shah, 2021; Patel & Shah, 2019), but also to be able to carry out many test conditions that would be impossible to achieve in reality.

Usually, there is interest in controlling the height of the contents of the tank; hence, the controlled variable is the height of the water in the tank relative to its bottom, which is adjusted by a PID control loop.

The description of this process can be expressed by a set of five equations. Firstly, we have the model of the inflow and outflow given by

$$\mathrm{F}_{\text{in}} = \alpha_1 \cdot f + \alpha_2 \geq 0, \tag{1}$$

$$\mathrm{F}_{\text{out}} = \beta\sqrt{h_c}, \tag{2}$$

where $\alpha_1 = 543\ \mathrm{cm^3 s^{-1}}$ and $\alpha_2 = -78.23\ \mathrm{cm^3 s^{-1}}$

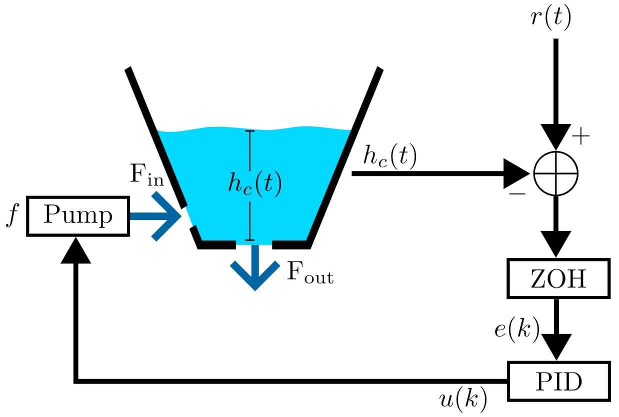


Figure 1. Diagram of the controlled conical tank system, where f is the percentage of pump utilization, $h_c(t) \geq 0$ is the noisy measurement of the height of the water inside the conical tank (measured in cm) at time instant t, F_{in} and F_{out} are the inflow and outflow of water in the tank, respectively (implicitly at t; both measured in cm^3/s). With respect to the control loop, $r(t)$ is the reference signal at t, and $e(k)$ and $u(k)$ are the error and control signals, respectively, at the discrete time-step k induced by the zero-order hold (ZOH).

are pump and pipe-dependent parameters and $\beta = 20.21\ \text{cm}^{5/2}\text{s}^{-1}$ is a parameter that depends on the specific geometry of the tank and the characteristics of the fluid involved. The fluid volume inside the tank (denoted as V) is the most important variable for mass balance; experiments were carried out in (Jáuregui, 2016), and the volume was adjusted to a cubic polynomial of the liquid height in the following fashion:

$$V = 0.21h_c^3 + 5.7h_c^2 + 17.1h_c + 290.7. \tag{3}$$

Then, we can perform a chain rule on the rate of volume change as a function of time by

$$\frac{\mathrm{d}V}{\mathrm{d}t} = \frac{\mathrm{d}V}{\mathrm{d}h_c} \cdot \frac{\mathrm{d}h_c}{\mathrm{d}t} = F_{\text{in}} - F_{\text{out}}, \tag{4}$$

which results in the following ordinary differential equation (ODE) for the height $h_c(t)$:

$$\frac{\mathrm{d}h_c}{\mathrm{d}t} = \frac{\mathrm{F_{in}} - \mathrm{F_{out}}}{0.63h_c^2 + 11.4h_c + 17.1}, \tag{5}$$

In consideration of Eqs. (1) and (2), Eq. (5) can be expressed as

$$\frac{\mathrm{d}h_c}{\mathrm{d}t} = \frac{\alpha_1 \cdot f + \alpha_2 - \beta\sqrt{h_c}}{0.63h_c^2 + 11.4h_c + 17.1}. \tag{6}$$

To emulate a real-world process, Gaussian noise was introduced to the measured water level (i.e., $h_c(t)$), simulating a pressure sensor with inherent noise. This noise was characterized by a mean of 0 and a standard deviation of 0.03 cm, reflecting a margin of error typical of such sensors.

This process is controlled by a closed-loop system where the manipulated variable is the percentage of pump utilization (f) and the controlled variable, as stated before, is the height of the liquid inside the tank (h_c). The control strategy used is a PID controller tuned by Particle Swarm Optimization (PSO).

PSO involves generating particles in the optimization space, which in this case represents the possible values of the parameters K_P, K_I, K_D of the PID controller ($\mathbb{R}^3$). The aim is to find a solution that minimizes a given cost function. Generally, we generate s particles with positions and velocities in the optimization space and use them to explore different solutions to evaluate the cost function. The following iterative process is applied to the velocity and position vectors of all the particles:

$$\begin{aligned} v_j^i(k+1) = \omega v_j^i(k) + c_1\varphi_1(k) \cdot [p_j^i(k) - x_j^i(k)] \\ + c_2\varphi_2(k) \cdot [g_j(k) - x_j^i(k)], \end{aligned} \tag{7}$$

$$x_j^i(k+1) = x_j^i(k) + v_j^i(k+1), \tag{8}$$

where, x_j^i and v_j^i represent the j-th component of the position ($j \in \{1,2,3\}$) and velocity of the i-th particle, respectively ($i \in \{1,\ldots,s\}$). The values of k and $k+1$ denote the algorithm's iteration indexes. The parameters are defined as follows: ω is the inertia factor, c_1 and c_2 are the cognitive and social constants, respectively, $\varphi_1(k)$ and $\varphi_2(k)$ are samples from a uniform distribution ($\mathcal{U}[0,1]$), $p_j^i(k)$ is the position of each particle that achieved the best performance according to the loss function, and $g_j(k)$ is the best position in history up to iteration k for all the particles.

The cost (loss) function for the algorithm is the total error of the controlled variable with respect to the reference; this is

$$J(t) = \int_0^t |e(\tau)|\,\mathrm{d}\tau. \tag{9}$$

Further details on the controller optimization procedure can be found in (Jáuregui, 2016). The main idea is that, following the presented rules, particles explore the optimization space with a compromise between individual and collective behavior. The PID parameters obtained using this approach are $K_P = 17.88$, $K_I = 9.41 \cdot 10^{-5}$ and $K_D = 4.44$; these parameters are incorporated into the following equation describing the action of the PID controller:

$$u(t) = K_P \cdot e(t) + K_I \cdot \int_0^t e(\tau)\,\mathrm{d}\tau + K_D \cdot \frac{\mathrm{d}e(t)}{\mathrm{d}t}. \tag{10}$$

Although Eq. (5) is in continuous time, the filling and emptying velocities of the tank allow for discretization with a reasonable resolution and timely response, eliminating the need for continuous observation. In particular, the filling time from an empty tank to a height of 50 cm using 100% of the pump capacity is approximately 150 seconds, while the emptying time is approximately 300 seconds. For this system, a sampling time of 15 seconds was selected, corresponding to approximately 10% of the filling rate. For this purpose, we adopt a Zero Order Hold (ZOH) technique to discretize the system, using the mentioned 15-second sampling interval. From this point on, we will consider t as a discrete variable, corresponding to the sampling described in Equation (11):

$$t = k \cdot T_{\mathrm{s}}, \tag{11}$$

where $k \in \mathbb{N}$ represents the number of samples since the beginning and $T_{\mathrm{s}} = 15$ s denotes the chosen sampling time.

2.2. The Information-Driven Fault Detection

We use the method for fault detection proposed in (Ramírez et al., 2024), which considers the system as a generative stochastic process of the inputs-outputs, which, in turn, can be reduced to a deterministic underlying mapping $\eta(\cdot)$. We approximate this underlying mapping with a model that is suited for the regression task of a variable of interest. The model can be of any type, whether it is a phenomenological model, a data-based (empirical) model (such as a neural network), or any other mapping capable of taking the same inputs as the system and generating (almost surely) the same outputs. The methodology is inspired by the orthogonality principle of least square error models, which states that the best model, in the sense of least square error, satisfies $\mathbb{E}[(\tilde{\eta}(X) - Y) \cdot X] = 0$, where $\tilde{\eta}(\cdot)$ is the system model, Y is the system's measured output, and X is the system's input. However, this principle does not align with the common intuition that the prediction error should be independent of the input, as the orthogonality principle does not guarantee independence. Therefore, we propose using the mutual information between the residual $R \equiv \tilde{\eta}(X) - Y$ and the input X to test for independence.

The mutual information of two arbitrary random variables X and Y, denoted as $I(X;Y)$, is a quantity derived from Information Theory, which quantifies the degree of dependence of the two random variables involved and has the particularity of being equal to 0 if and only if, X and Y are independent. For continuous random variables, mutual information can be expressed as follows:

$$I(X;Y) = \int_{\mathcal{X}\times\mathcal{Y}} \mu_{X,Y}(x,y) \log\left(\frac{\mu_{X,Y}(x,y)}{\mu_X(x)\cdot\mu_Y(y)}\right) \mathrm{d}x\, \mathrm{d}y, \tag{12}$$

where X and Y take values in $\mathcal{X}$ and $\mathcal{Y}$, respectively; $\mu_{X,Y}(x,y)$ is their joint probability density function (pdf), and $\mu_X(x)$ and $\mu_Y(y)$ are the marginal pdfs of X and Y, respectively, induced from $\mu_{X,Y}(x,y)$.

The expression in Eq. (12) needs the knowledge of the pdfs, which implies the necessity to use all distribution moments to compute it; this is in opposition to knowing just second-order moments, which is the requirement to compute the correlation to verify the orthogonality principle. The main cost associated with working with the mutual information is that knowing all distribution moments is not possible for unknown or non-trivial distributions; therefore, to implement the methodology, an estimator for the mutual information is needed.

The proposed method requires access to data from a system to monitor and a model of this system. In our case, we have a conical tank whose liquid level height we want to monitor. Contrary to what one could expect, the system is not just the tank; we also have to consider that there is a control loop that is an integral part of the system since, without it, the values and the distribution of the variables of interest would vary radically.

Once we have identified the system, we create a model that is based on it. During operation, the model predicts the target value; then, we calculate the prediction error, which we denominate the *residual* (denoted by uppercase R). To understand the relationship between the residual and the input (X), we estimate the mutual information between them. Since estimating mutual information requires multiple samples, we use a rolling window approach to collect the necessary data.

Finally, we assess the value of the mutual information estimation. If the value is greater than 0, we say our model and our system have *drifted* from their ideal fit. Since our model is fixed (after training), it cannot drift; hence, we attribute this discrepancy to unwanted alterations of the inner dynamics of the system, which we consider a fault. In consequence, detection is achieved.

An estimator has to be chosen for the calculation of mutual information. In this work, we will use the estimator presented in (Silva & Narayanan, 2012) following the study of its properties in the fault detection task studied in (Ramírez et al., 2024). This specific estimator, in conjunction with the method proposed, is formally proven to fulfill a set of convenient properties such as strong consistency, exponentially fast decision convergence on healthy systems, and guarantees on error convergence; the details can be found in (Ramírez et al., 2024).

The mentioned fault detection scheme is described in Figures 2 and 3.

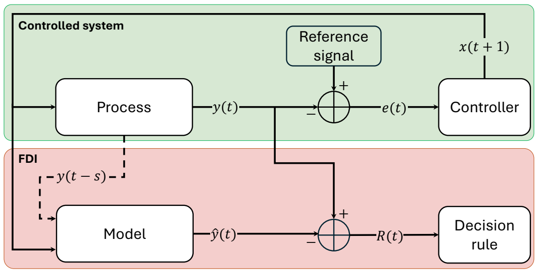


Figure 2. General schematic of model-based fault detection for controlled systems. Here, $s \geq 1$ denotes the number of historical system output values incorporated into the system model.

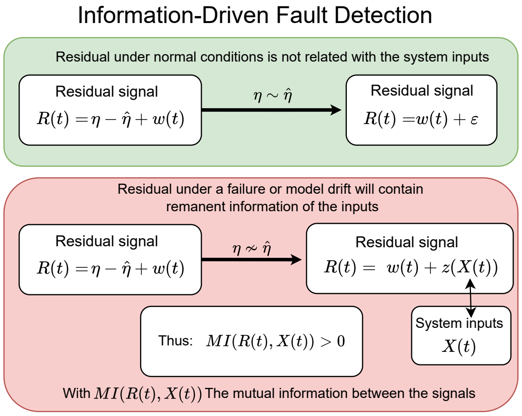

Information-Driven Fault Detection
Residual under normal conditions is not related with the system inputs
Residual signal
$R(t) = \eta - \hat{\eta} + w(t)$
$\eta \sim \hat{\eta}$
Residual signal
$R(t) = w(t) + \varepsilon$
Residual under a failure or model drift will contain remanent information of the inputs
Residual signal
$R(t) = \eta - \hat{\eta} + w(t)$
$\eta \nsim \hat{\eta}$
Residual signal
$R(t) = w(t) + z(X(t))$
Thus: $MI(R(t), X(t)) > 0$
System inputs
$X(t)$
With $MI(R(t), X(t))$ The mutual information between the signals

Figure 3. Main idea of the information-driven fault detection mechanism.

3. STUDY CASE — CONICAL TANK DESCRIBED BY A BLACK-BOX MODEL

In this section, we present several experiments in which a black-box model, fitted to the nominal system operation, supervises the PID-controlled conical tank. These experiments center around the phenomenological model of the conical tank operated by the described PID controller. Our objective is to emulate the system's behavior and use the information-driven fault detection method explained in the previous section to identify potential faults as they arise.

3.1. Nominal Black-Box Model Fitting

In this subsection, we detail the procedures performed to obtain the nominal black-box model for the system consisting of the PID-controlled conical tank.

3.1.1. Training Data

The data we used to train our black-box model consists of 90 hours of normal (nominal) plant operation, where the reference signal varied within normal operating limits to ensure a diverse and representative dataset. In Figure 4, a visualization of the initial four hours of nominal operation, which is part of the training dataset, is shown.

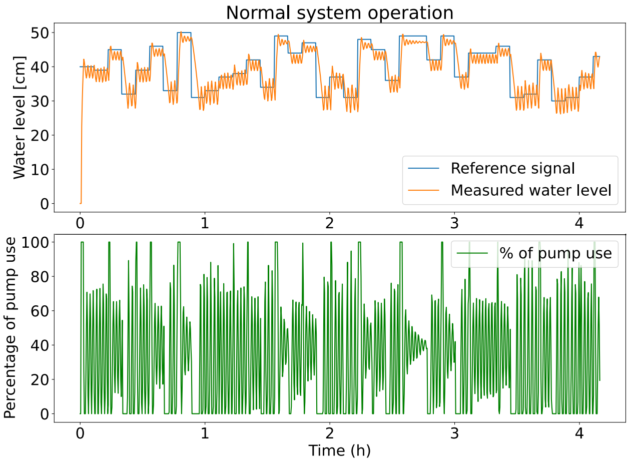


Figure 4. Snapshot of the initial 4 hours of training data. It reveals the structural composition of the dataset employed, as well as the impact of PID control.

3.1.2. Black-Box Model Specifications

Our black-box model is a Multi-Layer Perceptron (MLP) trained with data from the 90 hours of nominal plant operation described previously. Figure 5 illustrates how this model functions as a supervisor of the system. The MLP was chosen for its simplicity and ease of use, allowing us to demonstrate that the success of our method does not depend on model complexity. Using a straightforward model like the MLP helps us to visualize the innovative aspects of our method. Initially, the model is trained using data from nominal conditions; then, the fitted model emulates the system's output signal, which is compared with the actual signal to compute a residual. This residual signal is then used along with the system input for the information-driven fault detection testing.

The specific aim of this MLP is to predict the water level at the next discretized time step (i.e., at time instant t) based on the latest two observations of the water level and pump utilization percentage (i.e., $h_c(t-1)$, $f(t-1)$, $h_c(t-2)$, and $f(t-2)$). This information enables the network to capture insights about the system's inertia and dynamics while operating within its normal range, ultimately allowing it to estimate the water level for the next discretized time step when assuming normal conditions.

The MLP architecture consists of 4 input neurons followed by 2 hidden layers of 100 and 50 neurons, respectively; both hidden layers are equipped with ReLU activation functions, and there is only one output neuron that represents the predicted value of $h_c(t)$.

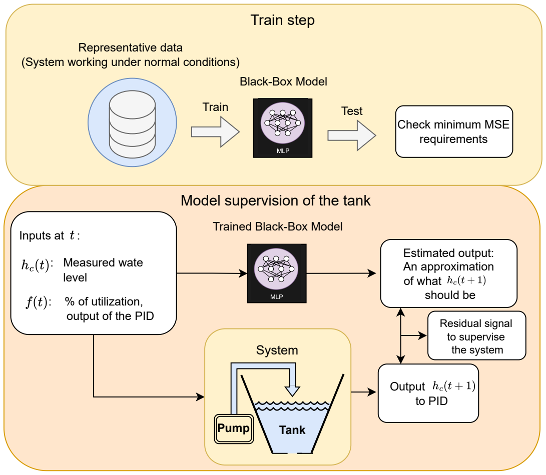

Train step
Representative data
(System working under normal conditions)
Black-Box Model
Train
MLP
Test
Check minimum MSE requirements
Model supervision of the tank
Inputs at t:
$h_c(t)$: Measured wate level
$f(t)$: % of utilization, output of the PID
Trained Black-Box Model
MLP
Estimated output: An approximation of what $h_c(t+1)$ should be
Residual signal to supervise the system
System
Pump
Tank
Output $h_c(t+1)$ to PID

Figure 5. Schematic diagram for both steps in our methodology. Top: training of the nominal black-box model. Bottom: System supervision using the nominal black-box model to perform fault detection.

3.1.3. Training Stage

Our first model was trained using a canonical training loop over 10,000 epochs. An ADAM optimizer was employed to minimize the Mean Squared Error (MSE) of the water level $h_c(t)$ prediction, with a learning rate set to 10^{-4}. To prevent overfitting and enhance generalization, a validation set was used during training.

The target MSE is determined by the *naive* no-change prediction approach (Armstrong, 2001, p. 308), which is one of the most straightforward forecasting techniques. This method posits that the forecast for the next time step is simply the most recent observed value and is used as a baseline for assessing more sophisticated forecasting methods – in this case, the naive prediction of $h_c(t)$ is given by $h_c(t-1)$. We set the target MSE to be at most the 1% of the MSE of the *naive* prediction (we will call it "*naive* MSE" for simplicity) to consider a model to be valid. A graphical representation of this kind of prediction can be seen in Figure 6.

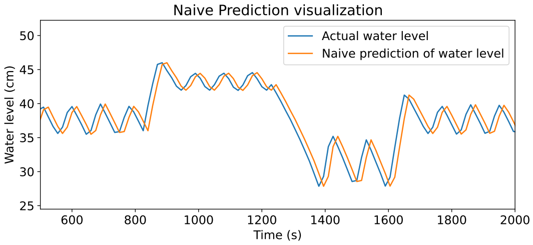


Figure 6. Visualization of the naive no-change prediction.

To ensure the robustness of our nominal black-box model, we trained four additional models using the same architecture. These models were only varied in the number of epochs and he employment of different learning rate (LR) schedulers. Their performances are compared in Table 1.

Table 1. MSE comparison between different models.

Model	N° of epochs	LR scheduler	MSE Test Set
Naive	N/a	N/a	2.46487
V1	10000	No	0.00649
V2	20000	No	0.00573
V3	30000	No	0.00261
V4	30000	Multiply the LR by 0.1 every 10000 epochs	0.00241
V5	50000	Multiply the LR by 0.9 every 2000 epochs	0.00186

As we can see in Table 1, all models achieve the target MSE. Given their similar performance, and to avoid redundancy, the results presented will focus on the best-performing model (i.e., model V5).

3.2. Modeling Pump Faults

In order to model a pump failure, we modified Eq. (1) to incorporate a perturbation coefficient δ, which weights a step change in the equation at time T_{fault}, explicitly induced by using the unit step function $s(\tau)$ which is 1 if $\tau \geq 0$ and 0 otherwise; this leads us to the following expression for the liquid inflow:

$$F_{\text{in}} = \alpha_1(1 - \delta \cdot s(t - T_{\text{fault}})) \cdot f + \alpha_2. \tag{13}$$

In our setting, all fault scenarios consist of a simulation of 90 hours with $T_{\text{fault}} = 45$ h, where we explore different values of δ. In Table 2, we show how different fault severities (i.e., δ values) impact the maximum possible liquid inflow of our system.

Table 2. Absolute impact of the simulated faults for 100% pump utilization.

Fault severity	Maximum fault impact (cm^3/s)
5%	-27.15
10%	-54.3
20%	-108.6
30%	-162.9
40%	-217.2

3.2.1. Application of the Mutual Information Test

The mutual information test was performed using a sliding window technique (see Figure 7), where the estimated mutual information (EMI) was calculated over a sliding window of 4 hours (which contains 960 samples).

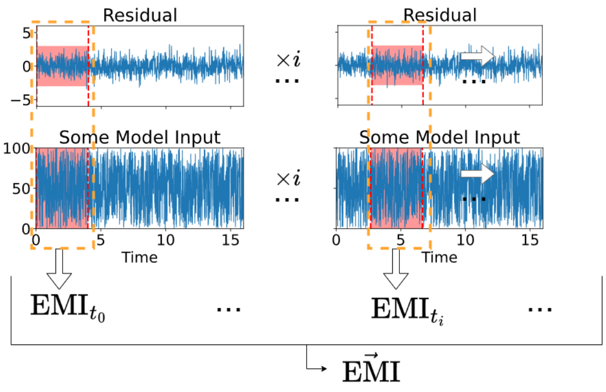


Figure 7. Mutual information estimation between the residual signal and some model input using a sliding window of 4 hours rolling over the whole time-series.

3.2.2. Results

In this section, we present the results of our experiments and analyses. We conducted the mutual information test over the residual signal $R(t) = h_c(t) - \hat{h}_c(t)$ and the inputs of the model that generated $\hat{h}_c(t)$. In Figures 8 and 9, we present the time evolution of the EMI value of $R(t)$ with $f(t-1)$ and $h_c(t-1)$, respectively, for all fault scenarios described in Table 2, represented in different colors. There is a clear increase in the EMI signal after the fault injection time (T_{fault}), depicted as vertical dashed red line, from which a transient in their values is observed up until all the data of the implemented rolling window is subjected to the fault (which is indicated as vertical dashed black line).

Once all the data of the rolling window used to calculate the EMI between the residual and the corresponding inputs contains only faulty data, the EMI stabilizes at increasing values which are monotonic with the severities of the introduced faults; this is clearly visible in Figure 10 where the EMI values from said times are averaged and their standard deviation is calculated. We also show these EMI statistics when no fault is introduced for comparison. It is clear that there is a monotonic relationship between the failure severity and the EMI, this could be exploited to easily detect faults in the system.

It is interesting to note that although the fault is located exclusively in the pump, the mutual information estimation between the residual and the water level input rises when the fault begins. This can be explained by the close connection between the percentage of pump usage and the water level caused by the PID control loop.

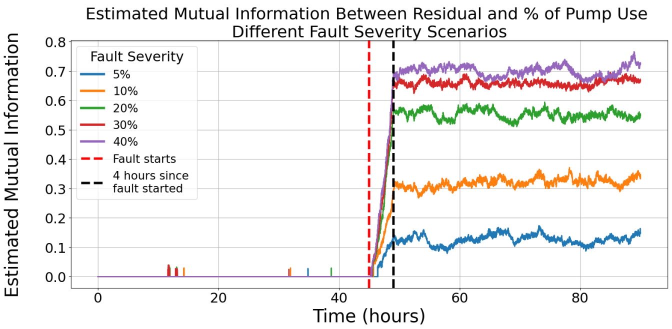


Figure 8. Estimated mutual information (EMI) between R and $f(t-1)$ – i.e., the residual and the percentage of pump use as a function of time – for different fault severity scenarios.

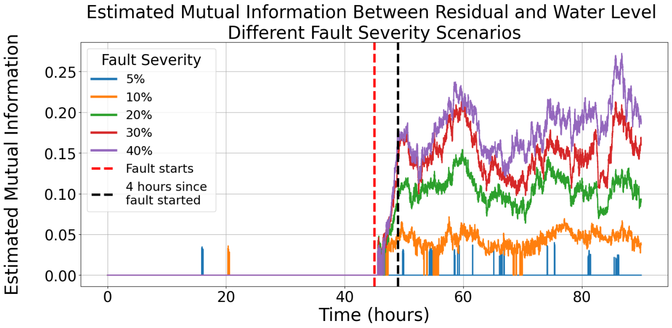


Figure 9. Estimated mutual information (EMI) between R and $h_c(t-1)$ – i.e., the residual and the water height level as a function of time for different fault severity scenarios.

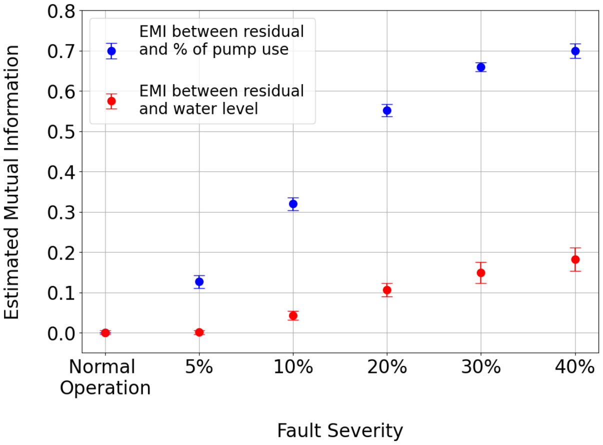


Figure 10. Average value and standard deviation of the mutual information estimations over a 40-hour window for normal system operation and different fault severities.

4. COMPLEMENTARY ANALYSIS — EXPERIMENT WITH A WHITE-BOX MODEL

To explain the results of the previous section, a *white box* experiment is performed. Here we are interested in enriching our previous analysis by simulating the plant with and without faults in an open-loop fashion and comparing the outputs of both faulty and faultless settings for the same inputs; this allows us to compare the performance of the methodology with the best possible model of the system, which is an exact simulation of the faultless system.

4.1. Experimental Setup

For this experiment, the original continuous-time variable system is discretized, and the ODEs are solved using Euler's method. The discretized set of equations, which are derived from Eq. (6), is the following:

$$h_{k+1} = \frac{(1-\delta)\cdot\alpha_1 f + \alpha_2 - \beta\sqrt{h_k}}{0.63k_k^2 + 11.4h_k + 17.1}\cdot T_s + h_k, \qquad (14)$$

where $k \in \mathbb{N}$ denotes the iteration step, $\delta \in [0,1]$ denote the fault severity, h_k and h_{k+1} denote the liquid height at steps k and $k+1$, respectively, and T_s is the sampling time which in this case was the same as the black-box experiment (15 s). Different levels of fault severity were explored: from 0% to 50%. The main differentiation of this analysis, with respect to the black-box analysis, is that we are working in an open-loop setting where the values of f are randomly generated from $\mathcal{U}[30\%, 40\%]$ instead of being obtained from a controller.

4.2. Results

In Figure 11, we can see that as the fault increases, the value of the input-residual EMI – residual information values (RIVs), as introduced in (Ramírez et al., 2024) – increases monotonically. This is in agreement with what was observed in the black-box analysis of the system.

We can see that for faults of low severity, there are no detections, and on the contrary, there is a point from which the detection rate is 100% (represented by the values of all RIVs being greater than 0). This is more clear to visualize in Figure 12, where the detection rate is plotted as a function of the fault's magnitude, where from fault severities higher than 5%, all 100% of the simulations are correctly detected as faulty; more importantly, Figure 12 also shows that our method does not incur in false positives (i.e., the false positive ratio – FPR – is 0).

To remark on the capability of the proposed method to quantify the fault's severy, Figure 13 shows, for different severities of fault, the histograms for the RIVs. It is possible to see that from a 10% severity, there are no false negatives; this means that there are no instances of faulty data wrongly labeled by our method as healthy. This is not the case for 5%, but the percentage of such occurrences is very low — 1.8%.

Although there is a discrepancy in the exact values of the RIVs between the black-box and white-box analyses, the overall behavior is consistent in both cases. The magnitude difference (approximately a factor of 2) can be attributed to the white-box analysis being an open-loop system and the different methods of solving the ODEs: a simple Euler method for the white-box analysis and Simulink as the solver for the black-box analysis.

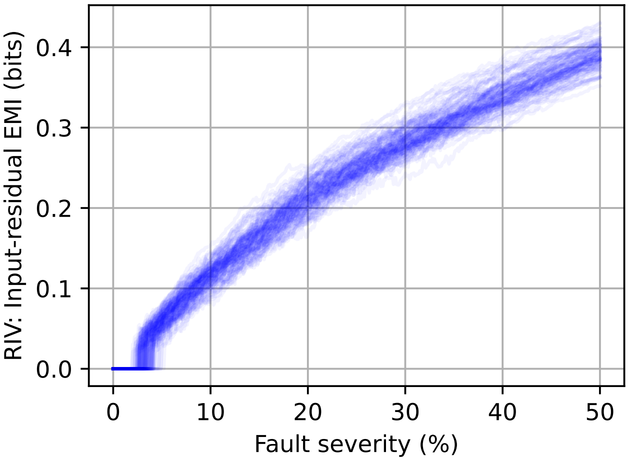


Figure 11. RIV values for different severities of the fault under study. For each fault level, 100 simulations were performed. For the estimation of the RIV, 1200 samples were used, which, at a sample time of 15 s, is equivalent to 5 hours.

5. CONCLUSION AND FUTURE WORK

In this paper, we demonstrated that our method effectively detects faults of varying severity using a black-box model, which can be pre-existing and not necessarily designed for fault detection purposes. Our methodology not only identifies faults but also provides an indicator of fault severity. To validate our approach, we replicated the setup in an open-loop system, where the model used is a white-box generative model of the data. The results were similar, further validating the effectiveness of our methodology even with a black-box model.

One key advantage of the presented methodology is its unsupervised nature for fault detection. It does not require prior access to fault data from the monitored system before beginning the detection process. This feature enhances its utility as a fault detection tool, with the resulting residual information value (RIV) serving as a critical component for Active Fault-Tolerant Control (Jiang & Yu, 2012).

As a potential improvement, more sophisticated models could be used to enhance the reliability of the experiments performed with the MLP. Additionally, running two parallel closed-loop simulations with identical inputs could serve as an ideal validation step, as it would represent a perfect model of the process. Finally, using a real system is the ultimate

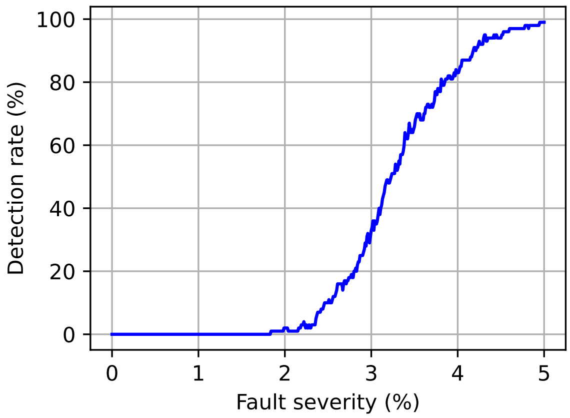


Figure 12. Detection Rate as a function of the severity of the fault. These rates were computed from 100 simulations with different random seeds.

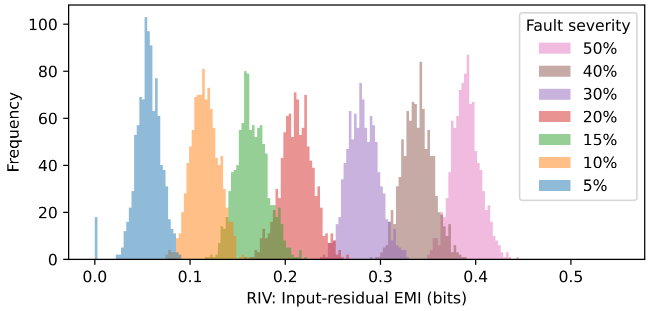


Figure 13. RIV histogram for different fault severities; these histograms were computed from 1000 simulations per fault severity with a unique random seed per simulation.

goal. Hence, this is another area where this work can be extended.

Finally, comparing our method with other FDI approaches is necessary to better understand its advantages and limitations.

ACKNOWLEDGEMENTS

This material is based on work supported by grants of CONICYT-Chile, Fondecyt 1210031 and Fondecyt 1210315, and the Advanced Center for Electrical and Electronic Engineering, Basal Project FB0008. Camilo Ramírez is supported by ANID-Subdirección de Capital Humano/Magíster-Nacional/2023 - 22230232 master's scholarship.

REFERENCES

Abid, A., Khan, M. T., & Iqbal, J. (2021). A review on fault detection and diagnosis techniques: basics and beyond. *Artificial Intelligence Review*, *54*(5), 3639–3664.

Amin, A. A., & Hasan, K. M. (2019). A review of fault tolerant control systems: advancements and applications. *Measurement*, *143*, 58–68.

Armstrong, J. S. (2001). *Principles of forecasting: a handbook for researchers and practitioners* (Vol. 30). Springer.

Chen, H., Wang, X.-B., Li, J.-m., & Yang, Z.-X. (2024). Dynamic focusing network for semisupervised mechanical fault diagnosis of rotating machinery. *IEEE Transactions on Industrial Informatics*.

Costa, B. S. J., Angelov, P. P., & Guedes, L. A. (2015). Fully unsupervised fault detection and identification based on recursive density estimation and self-evolving cloud-based classifier. *Neurocomputing*, *150*, 289–303.

Düştegör, D., Frisk, E., Cocquempot, V., Krysander, M., & Staroswiecki, M. (2006). Structural analysis of fault isolability in the damadics benchmark. *Control Engineering Practice*, *14*(6), 597–608.

Ghosh, D., Sharman, R., Rao, H. R., & Upadhyaya, S. (2007). Self-healing systems—survey and synthesis. *Decision support systems*, *42*(4), 2164–2185.

Jáuregui, C. (2016). *Evaluación de estrategias de sintonización de controladores fraccionarios para planta no lineal: sistema de estanques* (Master's thesis, Universidad de Chile). Retrieved from https://repositorio.uchile.cl/handle/2250/140963

Jiang, J., & Yu, X. (2012). Fault-tolerant control systems: A comparative study between active and passive approaches. *Annual Reviews in control*, *36*(1), 60–72.

Jieyang, P., Kimmig, A., Dongkun, W., Niu, Z., Zhi, F., Jiahai, W., . . . Ovtcharova, J. (2023). A systematic review of data-driven approaches to fault diagnosis and early warning. *Journal of Intelligent Manufacturing*, *34*(8), 3277–3304. doi: 10.1007/s10845-022-02020-0

Lucke, M., Mei, X., Stief, A., Chioua, M., & Thornhill, N. F. (2019). Variable selection for fault detection and identification based on mutual information of alarm series. *IFAC-PapersOnLine*, *52*(1), 673–678.

Patel, H. R., & Shah, V. A. (2019). Passive fault tolerant control system using feed-forward neural network for two-tank interacting conical level control system against partial actuator failures and disturbances. *IFAC-PapersOnLine*, *52*(14), 141–146.

Ramanathan, P., Mangla, K. K., & Satpathy, S. (2018). Smart controller for conical tank system using reinforcement learning algorithm. *Measurement*, *116*, 422–428. doi: 10.1016/j.measurement.2017.11.007

Ramírez, C., Silva, J. F., Tamssaouet, F., Rojas, T., & Orchard, M. E. (2024). Fault detection and monitoring using an information-driven strategy: Method, theory, and application. *arXiv preprint arXiv:2405.03667*. doi: 10.48550/arXiv.2405.03667

Raval, S., Patel, H. R., & Shah, V. A. (2021). Fault-tolerant controller comparative study and analysis for benchmark two-tank interacting level control system. *SN Computer Science*, *2*, 1–10.

Silva, J. F., & Narayanan, S. (2012). Complexity-regularized tree-structured partition for mutual information estimation. *IEEE transactions on information theory*, *58*(3), 1940–1952.

Srinivasan, K., Sindhiya, D., & Devassy, J. (2016). Design of fuzzy based model predictive controller for conical tank system. In *2016 ieee international conference on control and robotics engineering (iccre)* (pp. 1–6). doi: 10.1109/ICCRE.2016.7476135

Sun, B., Wang, J., He, Z., Zhou, H., & Gu, F. (2019). Fault identification for a closed-loop control system based on an improved deep neural network. *Sensors*, *19*(9), 2131.

Talebi, H. A., & Khorasani, K. (2012). A neural network-based multiplicative actuator fault detection and isolation of nonlinear systems. *IEEE Transactions on Control Systems Technology*, *21*(3), 842–851.

Thuillier, J., Jha, M. S., Le Martelot, S., & Theilliol, D. (2024). Prognostics aware control design for extended remaining useful life: Application to liquid propellant reusable rocket engine. *International Journal of Prognostics and Health Management*, *15*(1).

Vavilala, S. K., Thirumavalavan, V., & Chandrasekaran, K. (2020). Level control of a conical tank using the fractional order controller. *Computers & Electrical Engineering*, *87*, 106690. doi: 10.1016/j.compeleceng.2020.106690

Yang, J.-M., & Kwak, S. W. (2022). Self-repairing corrective control for input/output asynchronous sequential machines with transient faults. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*.

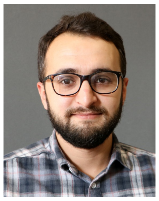
Yin, J., & Yan, X. (2019). Mutual information–dynamic stacked sparse autoencoders for fault detection. *Industrial & Engineering Chemistry Research*, *58*(47), 21614–21624.

BIOGRAPHIES

Joaquín Ortega received his degree in Electrical Engineering from the University of Chile, Santiago, in 2023, with a thesis focused on fault detection in conical tanks. He is currently pursuing an MSc in Computer Science at the same institution. Joaquín is also working as a Data Scientist at Banco de Créditos e Inversiones (BCI).

Camilo Ramírez received the B.Sc. degree in electrical engineering from the University of Chile, Santiago, Chile in 2022. He is currently pursuing the M.Sc. degree in electrical engineering from the University of Chile, has been a Research Assistant of the Information and Decision Systems (IDS) Systems group and the Advanced Laboratory for Geostatistical Supercomputing (ALGES) since 2021 and 2024, respectively, and was a Teaching Assistant of several graduate and undergraduate courses during 2019–2023. His research interests include prognostics and health management, machine learning, information theory, and statistics. He is a recipient of the National Master's Scholarship 2023 from the National Research and Development Agency of Chile.

Tomás Rojas obtained his degrees in Electrical Engineering and Physics from the University of Chile in 2024. From an early stage in his career, he developed a deep passion for research, gaining valuable experience at the European Southern Observatory (ESO), the Instituto de Ciencia de Materiales de Madrid (ICMM), and the Advanced Center for Electrical and Electronic Engineering (AC3E) at Universidad Técnica Federico Santa María (USM). He currently serves as a Research Assistant with the Information and Decision Systems (IDS) group, focusing on prognostics and health management, particularly in fault detection and identification. In parallel, he is also a Research Assistant at the Nanolaboratory (NanoLab), where his work emphasizes solid-state physics, specifically graphene-based devices.



Ferhat Tamssaouet received his Master's degree from the École Normale Supérieure Paris-Saclay (ENS-PS) in 2017. In 2020, he obtained his PhD from the Institut National Polytechnique (INP) de Toulouse in the field of industrial engineering. Following his PhD, he worked as an assistant lecturer at the École Nationale d'Ingénieurs de Tarbes (ENIT) for one year. From 2021 to 2024, he held the position of associate professor at the Université de Perpignan Via Domitia (UPVD) and the PROMES-CNRS laboratory. Since 2024, is an associate professor affiliated with the Université Toulouse III - Paul Sabatier and the LAAS-CNRS laboratory. His research interests include prognostics and health management of complex systems, discrete event systems, and renewable energy, with a particular focus on concentrated solar power.

Marcos Orchard received the M.S. and Ph.D. degrees in electrical and computer engineering from the Georgia Institute of Technology, Atlanta, GA, USA, in 2005 and 2007, respectively. Currently, he is a Professor with the Department of Electrical Engineering, Universidad de Chile, Santiago, Chile, and an Associate Researcher with the Advanced Center for Electrical and Electronic Engineering (UTFSM). He has authored and coauthored more than 100 papers on diverse topics, including the design and implementation of failure prognostic algorithms, statistical process monitoring, and system identification. His research work at the Georgia Institute of Technology was the foundation of novel real-time failure prognosis approaches based on particle filtering algorithms. His current research interests include the study of theoretical aspects related to the implementation of real-time failure prognosis algorithms, with applications to battery management systems, electromobility, mining industry, and finance. Dr. Orchard is a Fellow of the Prognostic and Health Management Society.

Jorge Silva received the M.Sc. and Ph.D. degrees in electrical engineering from the University of Southern California (USC), Los Angeles, CA, USA, in 2005 and 2008, respectively. From 2003 to 2008, he was a Research Assistant with the Signal Analysis and Interpretation Laboratory (SAIL), USC. He was also a Research Intern with the Speech Research Group, Microsoft Corporation, Redmond, WA, USA, in 2005. He is currently an Associate Professor with the Department of Electrical Engineering (EE), University of Chile, and a Principal Investigator with the Advanced Center of Electrical and Electronic Engineering, Valparaíso, Chile. He received the Outstanding Thesis Award for Theoretical Research of the Viterbi School of engineering, in 2009, the Viterbi Doctoral Fellowship, from 2007 to 2008, and the Simon Ramo Scholarship at USC, from 2007 to 2008. He was an Associate Editor of IEEE *Transactions on Signal Processing*, from 2016 to 2018.