Fault Detection and Isolation for High-End Industrial Printers

A hybrid model- and data-based approach

C.D. van Peijpe

Master of Science Thesis

TU Delft Delft University of Technology

Delft Center for Systems and Control

Fault Detection and Isolation for High-End Industrial Printers

A hybrid model- and data-based approach

Master of Science Thesis

For the degree of Master of Science in Systems and Control at Delft University of Technology

Supervisors:

P. Mohajerin Esfahani, Associate Professor

F. Ghanipoor, MSc

Y. de Loore, MSc
P.J.B. Hacking, MSc

C.D. van Peijpe

June 11, 2024

Faculty of Mechanical Engineering (ME) · Delft University of Technology

The writer was enabled by Canon Production Printing Netherlands B.V. to perform research that partly forms the basis for this report. Canon Production Printing B.V. does not accept responsibility for the accuracy of the data, opinions and conclusions mentioned in this report, which are entirely for the account of the writer.

Abstract

This report presents a hybrid model- and data-based method for detecting and isolating faults in the ink channel of a printer using the self-sensing capability of piezo actuators. Grey-box system identification is used to identify the parameters of the model of the ink channel. The model is used to construct the fault detection (FD) filter. The FD filter uses the piezo self-sensing signal of the printer as an input and puts out a residual signal, which is approximately zero when the system is healthy. Several methods to design the FD filter denominator are proposed. If the energy of the residual exceeds a threshold, the system is detected as faulty. The fault isolation (FI) filter uses linear regression, utilizing the residuals of the FD filter, to generate a probability vector. The entries of this vector correspond to the possible faults, and the highest entry is used to isolate the fault. Both filters are tested for a simulated and experimental dataset. For both datasets, the FD filter is shown to perform appropriately, as does the FI filter for experimental data. For simulated data, the FI filter is compared to other methods. The FI filter performs best when only regarding isolations with a high certainty.

Keywords **-** Fault Detection, Fault Isolation, System Identification, Linear Regression, Piezo Self-Sensing Actuators

Acknowledgements

I am thankful to my daily supervisor, Farhad Ghanipoor. Your hands-on guidance, detailed reviews, and day-to-day support have been crucial in navigating the complexities of my research. Our weekly meetings were crucial in pursuing new insights every day and your critical feedback continually challenged me to refine my approach and think more deeply about my work.

My gratitude extends to my supervisors from Canon, Youri de Loore and Pim Hacking. Your industry insights, practical perspectives, and professional expertise have significantly enriched my research. Our meetings always opened up new insights for me that I had not thought about, especially because of our different backgrounds and your knowledge of the complex systems I was working with. The guidance you provided helped me grasp concepts that would have otherwise taken me a lot of time to understand. This allowed me to make progress quickly without getting stuck. I also appreciate your patience in explaining topics that were new to me, even though they may have been trivial to you.

I would like to express my gratitude to my supervisor, Peyman Mohajerin Esfahani. Your expertise, insightful feedback, and enthusiasm have been invaluable. Considering your tight schedule, I feel honoured that you always tried to find time to guide me, often resulting in spontaneous meetings that helped me find both technical insights as well as motivation to make progress.

To all my supervisors and everyone who showed interest in my work in any way, your collective guidance, patience, and encouragement have been a cornerstone of my academic and professional development. I deeply appreciate the time, effort, and wisdom you have shared throughout this journey.

Thank you.

Delft, University of Technology
June 11, 2024

C.D. van Peijpe

Table of Contents

List of Figures

List of Tables

Chapter 1

Introduction

Over the past decades, fault diagnosis has been a focal point of research because of its vital role in maintaining the safety and reliability of various engineering systems. In this report, the systems of interest are the printers of Canon Production Printing (CPP). CPP is a Canon subsidiary company located in Venlo, the Netherlands, that develops, manufactures and sells high-end industrial printers [3]. The company's research and development department is dedicated to improving the quality of printing and other services. One of these improvements is accurate fault diagnosis for the ink channels of their printers. The ink channels are subject to different faults, such as empty ink channels, air bubble entrainment, partial or complete blockage of the nozzle or drying of the local ink [4]. All of these are examples that would degrade the image and printing quality.

Preventing a low-quality print has several advantages. Firstly, improving the overall print quality results in satisfied customers and possibly attracts new businesses. Secondly, avoiding poor prints can reduce the need for reprints, saving ink, paper, and running costs associated with customer complaints and repairs. Additionally, reducing waste ink and paper is not only cost-effective, but it's also environmentally responsible. Finally, a reliable and high-quality printer can give CPP a competitive advantage in the market. For these reasons, it is beneficial to detect a fault in advance and isolate faults in their early stages to compensate for a failing ink channel before it reduces printing quality.

Ensuring high print quality is thus a primary concern within the printing industry, and various general quality assurance mechanisms are employed for this purpose, one of which uses the self-sensing capability of piezo actuators. This technique can be leveraged to identify faults in jetting devices [5, 6, 7], and has the advantage of being able to distinguish the type of failure. Besides the printing industry, the utilization of piezo self-sensing for health monitoring is widespread across various other industries. Instances include avian influenza virus detection [8], structural health monitoring for concrete beams [9], health monitoring of aerospace structures [10], or damage monitoring of advanced 3D textile composites [11].

There are several reasons why addressing the problem of detecting and isolating faults in the ink channel of CPP's printers is challenging. Both the time scale and the dimension of

the system are microscopic, with a typical piezo self-sensing signal being tens of microseconds long [12], an actuation chamber of mere millimetres in length and drop ejection nozzles of just tens of micrometres in diameter. The combination of the small time and length scales results in very small physical quantities, making it hard to observe the system in real life. Only the piezo self-sensing signal can be used to get some sense of the system. Furthermore, in less than 100ms, more than 100.000 signals from different ink channels are collected. Dealing with those large amounts emphasises the need for a computationally efficient method. Moreover, the self-sensing signals contain noise, meaning a proposed method should be robust to noisy data. A further challenge regarding data collection is that the output signal is not captured during actuation, as the actuator is also the sensor of the system. For a model-based solution, input-output data is thus not available. This also means the initial state of the system cannot be assumed to be zero, which is generally the case in literature [13, 14, 15].

The frequency content of the self-sensing signals generated by the piezo actuator can be used for classification, either through straightforward threshold-based methods or by utilizing a neural network. The former is currently being used by CPP for the health monitoring of their printers and can be seen as a data-based method [16]. This method however does not directly take into account any of the physical changes happening during a fault. Besides, the thresholds for this method are manual margins based on knowledge, making the method non-systematic.

Motivated by the challenges of the aforementioned problem of detecting and isolating faults and the shortcomings of the current solution, a hybrid model- and data-based implementation is proposed that detects and isolates faults using piezo self-sensing signals. This implementation consists of two stages; Fault Detection (FD) and Fault Isolation (FI). The proposed method for FD is based on previous theoretical results [13, 14], where the task of FD involves generating a diagnostic signal sensitive to the occurrence of faults. This task is accomplished by designing a model-based filter with available signals as inputs and a scalar output that implements a non-zero mapping from the fault to the diagnostic signal, which is known as the residual. The filter formulation is modified to account for the input and output data not being available simultaneously. For FI, a variation on earlier research [17] is proposed, where an isolation filter is designed using a regression operator. Contrary to the previous work, faulty data is assumed to be available. Additionally, the filter is modified such that the result makes more sense for this use case, as only one fault can be present at a time.

The contributions of this work are summarized as follows:

- **Fault Detection for signals with non-zero initial state:** For any output signal, independent of the state at the time of acquisition, an FD filter, constructed using a healthy model, can be applied. The signal is detected as faulty when the energy of the FD filter output exceeds a predetermined threshold.
- **Fault Isolation, using a novel hybrid model- and data-scheme:** When a signal is detected as faulty by an FD filter, its residual is used and compared to the expected residual of every fault class, calculated by using known faulty signals. The probability of every fault being present is calculated and the fault with the highest probability is regarded as the present fault.

The content of the remainder of this report is as follows. The problem statement of this use case is described in Section 2. A solution approach to the problem is outlined in Section 3.

The results of this approach with simulated and experimental data are shown in Section 4. Lastly, the conclusions that can be drawn from these results are presented, together with recommendations for future work in Section 5.

Notation. *Sets ($\mathbb{R}_+$) $\mathbb{R}$, $\mathbb{N}$, and $\mathbb{R}^n$ denote all (non-negative) real, positive integers, and the space of n-dimensional real-valued vectors, respectively. Let $A \in \mathbb{R}^{n\times m}$ be an $n \times m$ matrix with real values, $A^\intercal \in \mathbb{R}^{m\times n}$ be its transpose, $\mathcal{N}(A)$ be its null space and $\det(A)$ be its determinant, only possible when $n = m$. The identity matrix with an appropriate dimension is denoted by I. Similarly, a matrix or vector composed of only zeros is denoted as $\mathbf{0}$. For a vector $v = [v_1, ..., v_n]$, the $\mathcal{L}_2$-norm is $||v||_2 = \sqrt{\sum_{i=1}^{n} v_i^2}$. s is the complex frequency parameter used in the Laplace transform. x_a is the acquisition state, i.e. the state $x(t_a)$, where t_a is the time of acquisition. $\hat{x}_a$ denotes the estimation of the acquisition state. $\mathcal{O}(n)$ denotes the order of magnitude of n, which is the smallest power of 10 used to represent that number.*

Chapter 2

Problem Description

In this section, the problem is described. First, a general way to describe the model is shown in Section 2-1. Then, the use case is described in Section 2-2.

2-1 Model description

The system can be described as

$$\begin{cases} \dot{x}(t) &= (A + \sum_{i=1}^{n_f} B_{f_i} f_i)x(t) + B_u u(t) + B_d d(t), \\ y(t) &= Cx(t) + D_u u(t) + D_d d(t), \end{cases} \tag{2-1}$$

where $t \in \mathbb{R}_+$, $x \in \mathbb{R}^n$, $y \in \mathbb{R}^m$, $u \in \mathbb{R}^{n_u}$ and $d \in \mathbb{R}^{n_d}$ are time, state, measured output, known input and unknown disturbance vectors, respectively, and $f_i \in \mathbb{R}_+$ is the i-th fault signal between 0 and 1. Matrices $(A, B_{f_i}, B_u, B_d, C, D_u, D_d)$ are of appropriate dimensions, $n, m, n_u, n_d \in \mathbb{N}$. For every fault, the term $B_{f_i} f_i$ is added to A, where B_{f_i} is equal to $A_{f_i} - A$, A_{f_i} being the A matrix describing the dynamics when fault i is present. B_{f_i} is different for every fault scenario and contains parameter changes because of the fault.

In what follows, the ink channels of CPP's printers are presented.

2-2 Use Case: Canon Production Printing

Printers of CPP consist of various components that work together to ensure their proper functioning. These components include the print bed, ink supply, and printhead. In this context, the focus is on the ink channels, of which up to hundreds to thousands are located in the printheads, depending on the printer. In Figure 2-1, a decomposition is shown that indicates where each component is located and how many of each component is inside the component to its left.

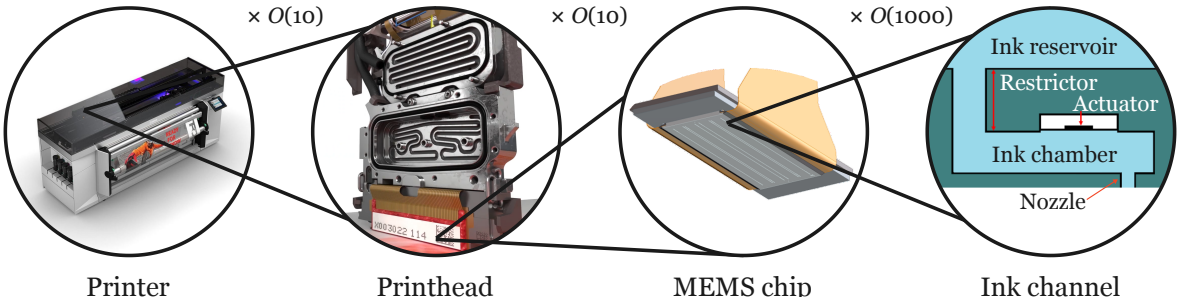

Figure 2-1: Decomposition of the printer where each component shown is smaller than and inside of the component to its left. The smallest component, the ink channel, also consists of components of which the names are shown. The orders show how many of the next components approximately are inside the previous component [1][2].

The smallest component, the ink channel, is considered as the system. When printing, the piezoelectric actuator is discharged, inducing the mechanical deformation of a membrane, and moves ink from the ink reservoir through the restrictor into the ink chamber. Microseconds later, the actuator is recharged, reverting the material to its original shape and moving ink from the ink chamber through the nozzle onto the paper [18]. A few microseconds after each actuation, the piezoelectric actuator can be used as a sensor. The flow rate in the ink chamber is proportional to the digital value of the sensor. An example of the process of both actuation and acquisition is plotted in Figure 2-2.

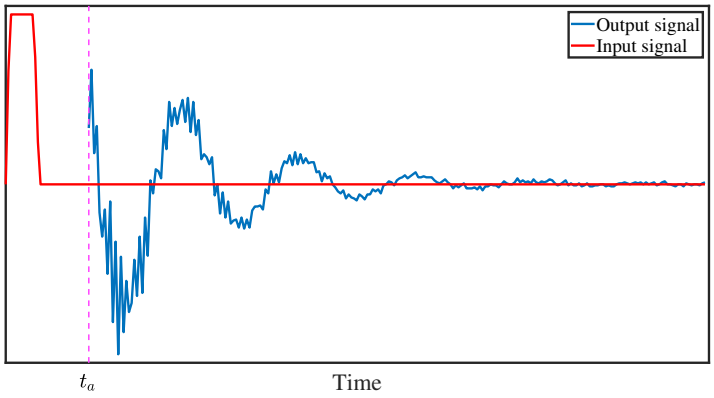

Figure 2-2: Displaying the input signal and an output signal, starting after the completion of the input, denoted by the vertical line at t_a. The order of the system output is $\mathcal{O}(10^2)$, the order of voltage input is $\mathcal{O}(10^1 V)$ and the order of time $\mathcal{O}(10^2 \mu s)$. The exact values are not given due to confidentiality. Note that although plotted together, the input and output differ in magnitude and unit.

Based on fluid dynamics the ink channel can be described as follows [12, 19]:

$$\underbrace{\begin{bmatrix}\dot{V}_r\\ \dot{V}_n\\ \ddot{V}_r\\ \ddot{V}_n\end{bmatrix}}_{\dot{x}} = \underbrace{\begin{bmatrix}0 & 0 & 1 & 0\\ 0 & 0 & 0 & 1\\ -\frac{1}{I_r B_t} & -\frac{1}{I_r B_t} & -\frac{R_r}{I_r} & 0\\ -\frac{1}{I_n B_t} & -\frac{1}{I_n B_t} & 0 & -\frac{R_n}{I_n}\end{bmatrix}}_{A}\underbrace{\begin{bmatrix}V_r\\ V_n\\ \dot{V}_r\\ \dot{V}_n\end{bmatrix}}_{x} + \underbrace{\begin{bmatrix}0\\ 0\\ \frac{b}{I_r B_t}\\ \frac{b}{I_n B_t}\end{bmatrix}}_{B_u} u,$$
$$y = \underbrace{\begin{bmatrix}0 & 0 & c & c\end{bmatrix}}_{C}\underbrace{\begin{bmatrix}V_r\\ V_n\\ \dot{V}_r\\ \dot{V}_n\end{bmatrix}}_{x}, \tag{2-2}$$

where V_r and V_n are the volumes of ink displaced through the restrictor and the nozzle, respectively. I_r and I_n are the inertances in the restrictor and the nozzle, which can be calculated approximately as

$$I_r = \frac{\rho L_r}{A_r}, \quad I_n = \frac{\rho L_n}{A_n}, \tag{2-3}$$

where ρ is the density of the ink and L_r and A_r are the length and cross-sectional area of the restrictor and L_n and A_n are the length and cross-sectional area of the nozzle [12]. R_r and R_n are the resistances in the restrictor and the nozzle, which can be calculated approximately as

$$R_r = \frac{8\pi\mu L_r}{A_r^2}, \quad R_n = \frac{8\pi\mu L_n}{A_n^2}, \tag{2-4}$$

where μ is the viscosity of the ink [12]. B_t is the total compliance of the system, which can be calculated approximately as

$$B_t = B_{act} + \frac{V_{ch}}{\rho c^2}, \tag{2-5}$$

where B_{act} is the compliance of the actuator, given by the manufacturer, V_{ch} is the total volume in the ink channel and c is the speed of sound in the ink. The input u is the voltage applied to the actuator. Actuator constant b captures the relation between the voltage input and the influence on the flow rate.

The actuator acts as a sensor after actuation. Whenever the output is collected, $u = 0$, as shown in Figure 2-2. Therefore the system is treated as autonomous with the non-zero state at the time of acquisition x_a, the acquisition state, thus B_u can be neglected. Acquisition constant c captures the relation between the flow rate in the ink chamber and the digital signal, i.e. the output of the system, and is given by the manufacturer.

Compared to the generic system described in equation (2-1), this system does not have an input. Output disturbance is present, but is not taken into account for the design process of the proposed solution as it can filter out disturbances. Faults cause the parameters to change, which are captured in B_{f_i}. The studied fault scenarios for the use case are as follows:

1) Empty ink channel (EC) Means an ink channel is not filled with ink. Every ink channel starts empty until it is filled with ink, which is not a trivial procedure. An ink channel can also be fully emptied when printing. An empty channel can be modelled as a decrease in ink density, i.e. a decrease in inertance, causing the actuator to oscillate at a higher frequency.

2) (Partially) blocked nozzle (PBN) Means a dirt particle is present at the outlet of the nozzle. Such a particle blocks a certain percentage of the nozzle's cross-sectional area, where 100% means the particle fully blocks the nozzle. This causes the inertance and resistance to increase, resulting in a lower frequency and higher damping.

3) Dried nozzle (DN) Means ink at the outlet of the nozzle has evaporated off some of its most volatile components, such as water. This causes the viscosity of the ink and thus the resistance to locally increase, meaning the damping increases. Different drying levels can be distinguished, such as slightly dried (SDN), intermediately dried (IDN) and deeply dried (DDN).

4) Air bubble (AB) Means air is entrapped in the ink chamber. Three different types of air bubbles can occur, which can be represented differently in the model.

At a high level, the problem aimed to be solved can now be stated.

Problem 1. (Fault Detection and Isolation) *Consider the system in* (2-1) *with specific dynamics of* (2-2)*. Design an FDI scheme such that:*
1) Fault Detection: *An FD filter distinguishes healthy and faulty systems using output data* $y(t)$ *and approximated acquisition state* $\hat{x}_a$*;*
2) Fault Isolation: *An FI filter isolates faulty systems using output data* $y(t)$ *and available faulty data.*

In what follows, the above problem is aimed to be solved.

Chapter 3

Solution Approach

In this section, the proposed solution to Problem 1 is given. Figure 3-1 presents a schematic overview of the proposed FDI scheme. From left to right, first a fault f enters the system dynamics, which can be one of several faults $f_1, f_2, \ldots, f_{n_f}$, where n_f is the number of possible faults. The system puts out signal y, which is used as an input for the FD filter. The other input is $\hat{x}_a$, which is the estimated acquisition state of the system. The output of the FD filter is the residual signal r, which should be close to zero when no fault is present in the system, and non-zero when there is a fault. If the energy of the residual exceeds a certain threshold, the system is seen as faulty and the residual signal is used as an input for the FI filter. This filter puts out a vector φ of which the entries correspond to the fault scenarios in f. The entry with the highest magnitude is chosen as the fault. In the figure, an interrupted arrow indicates an example of the signal is given. A continuous arrow indicates an actual input or output signal or number.

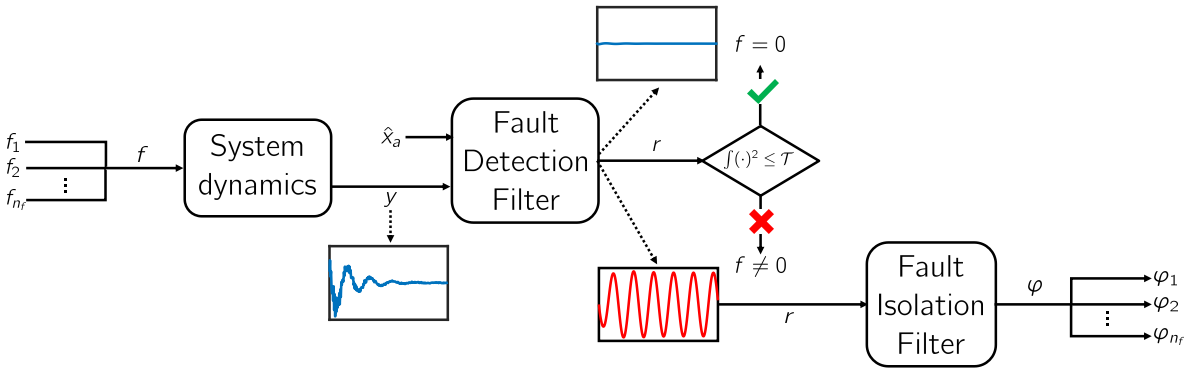

Figure 3-1: Configuration for FDI filter.

To design the proposed FDI scheme, first the preliminary step of system identification is shown in Section 3-1. Thereafter, the problems of fault detection and fault isolation are solved in Sections 3-2 and 3-3, respectively.

3-1 System Identification

A sufficiently accurate model and acquisition state must be available to use a model-based approach. The model can be found using grey-box system identification, which is solved by using the `greyest` function in MATLAB [20]. Grey-box system identification can be seen as an optimisation problem,

$$A, \hat{x}_a = \underset{A,\hat{x}_a}{\arg\min} \left|\left| y(t) - Ce^{At}\hat{x}_a \right|\right|_2, \tag{3-1}$$

where the parameters in A and approximated acquisition state $\hat{x}_a$ are calculated with known output $y(t)$ and C using an initial guess for A and a specified solver. The parameters of A are bounded by predetermined values. The output $y(t)$ used in (3-1) is the average of all output signals in the dataset, visualised in Figure 3-2.

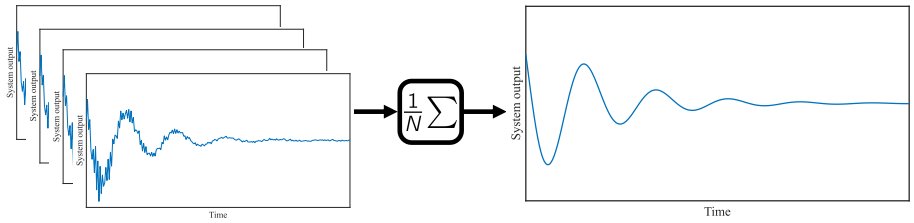

Figure 3-2: Visualisation of how a dataset containing output signals generates an average output, used for grey-box system identification.

In comparison to using only one output signal, averaging the signals has three benefits:

1. Although a healthy system is expected to always behave the same, small deviations in dynamics occur, even when a system is healthy. By taking the average of all signals, you do not get the bias you get when only using one signal;
2. There is a small chance the dataset contains a faulty signal that can be picked only using one signal, meaning the identified model already includes the fault. If the average is taken, the outlying faulty signal is neglected by the large amount of healthy signals;
3. Output signals contain noise, which is averaged out when enough signals are used. This way identification results are more accurate. Figure 3-2 shows this well, as all the raw signals contain noise, whereas the averaged signal is smooth.

Matrix C is assumed to be

$$C = \begin{bmatrix} 0 & 0 & 1 & 1 \end{bmatrix}.$$

Although this definition for C is not realistic since the order of magnitude of the flow rate and digital output differ significantly, it does not matter to the application as long as C is consistent. To successfully perform system identification, the initial guesses for the model parameters must be in the right order of magnitude due to non-convex optimization in (3-1). This is generally known within CPP, but can also be estimated using equations (2-3), (2-4) and (2-5) when the dimensions of the system and the properties of the used ink are known. The solver used for the optimisation problem is a constrained nonlinear solver, such as the

interior-point method. This search method is chosen since it results in improved estimation results for model structures where the loss function is a nonlinear function of the parameters or when bounds are imposed on the model parameters [21]. Both are the case for this system. After system identification is performed, the real output is compared to the simulated output using the identified model. Figure 3-3 shows an example of the system identification result. Here, a single piezo self-sensing signal from a healthy dataset is plotted with a simulated output signal using the model found. As can be seen, the result follows the real signal sufficiently accurately when disregarding the noise.

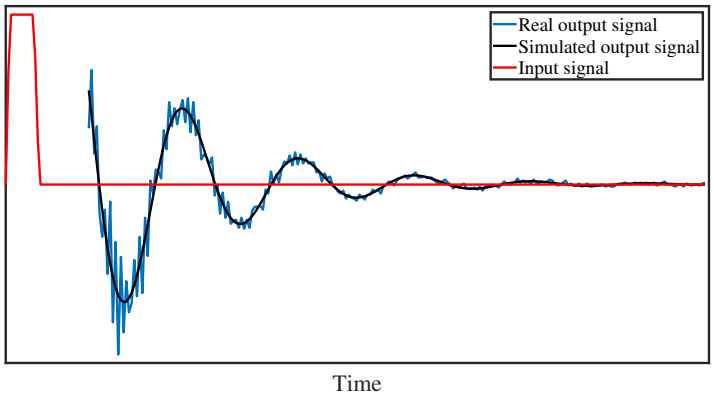

Figure 3-3: System identification results showing the simulated output of the identified model and a single raw signal for comparison. The input to the system is shown, though it is not used during identification.

When an accurate model is found, A, C and $\hat{x}_a$ are used for the model-based FD filter.

3-2 Fault Detection Filter

To construct the FD filter, first, the state-space representation of the system in (2-1) is converted from time-domain to frequency-domain as

$$\begin{cases} sX(s) - x_a & = AX(s) + \sum_{i=1}^{n_f} B_{f_i} f_i X(s), \\ Y(s) & = CX(s). \end{cases}$$

Here, s is the complex frequency parameter used in the Laplace transform. $X(s)$ and $Y(s)$ capture the frequency contents of $x(t)$ and $y(t)$. Combining the equations, this can be written as

$$\underbrace{\begin{bmatrix} -sI + A \\ C \end{bmatrix}}_{H(s)=H_0+H_1 s} X(s) + \begin{bmatrix} x_a \\ 0 \end{bmatrix} + \underbrace{\begin{bmatrix} 0 \\ -I \end{bmatrix}}_{L(s)} Y(s) + \underbrace{\begin{bmatrix} \sum_{i=1}^{n_f} B_{f_i} f_i \\ 0 \end{bmatrix}}_{F(s)} X(s) = 0, \tag{3-2}$$

where $H(s) \in \mathbb{R}^{(n+m)\times n}$, $L(s) \in \mathbb{R}^{(n+m)\times m}$ and $F(s) \in \mathbb{R}^{(n+m)\times n}$. This system representation is also known as the differential-algebraic equation (DAE) [13].

Now, the FD task is to design a filter whose input is the known signal y and whose output r differentiates whether the measurements are healthy, or influenced by the fault signal f. Formally speaking, the residual can be viewed as a function $r(x, f)$, and the FD design is ideally translated as the mapping requirements

$$x \mapsto r(x, 0) \equiv 0, \tag{3-3a}$$
$$f \mapsto r(x, f) \neq 0, \quad \forall x. \tag{3-3b}$$

To achieve this, first let

$$\bar{H} = \begin{bmatrix} H_0 & H_1 & & \\ & \ddots & \ddots & \\ & & H_0 & H_1 \end{bmatrix} \Big\} d_N,$$

where $\bar{H} \in \mathbb{R}^{d_N(n+m)\times n(d_N+1)}$ and d_N is the order of the filter. Generally, to create an FD filter, a linear programming network is solved in which the left null space of $\bar{H}$ is calculated, ensuring (3-3a) holds, that finds the solution that maximises the contribution of a fault, i.e. maximising (3-3b) [13]. However, for this specific system, the linear programming constraint equivalent to (3-3a) has only one solution, meaning the linear programming network is not needed. The easier method is thus to calculate the left null space of $\bar{H}$ as

$$\bar{N} = (\mathcal{N}(\bar{H}^\intercal))^\intercal = \begin{bmatrix} N_0 & N_1 & \cdots & N_{d_N} \end{bmatrix},$$

where $\bar{N} \in \mathbb{R}^{d_N(n+m)\times 1}$. To apply the result, $\bar{N}$ is converted to $N(s)$ using

$$N(s) := \sum_{i=0}^{d_N} N_i s^i.$$

The system equations in (3-2) are multiplied by $N(s)$, meaning $N(s)H(s)$ disappears by definition. The equations are divided by $\alpha(s)$, which is a (marginally) stable transfer function with sufficiently high order, such that a (strictly) proper transfer function is generated. Several methods to choose $\alpha(s)$ are described later. Equation (3-2) can now be modified as

$$\frac{N(s)}{\alpha(s)} \begin{bmatrix} x_a \\ 0 \end{bmatrix} + \frac{N(s)L(s)}{\alpha(s)} Y(s) + \frac{N(s)F(s)}{\alpha(s)} X(s) = 0.$$

The acquisition state x_a is assumed to be unknown but consistent for healthy signals and can be estimated as $\hat{x}_a$ using system identification, described in Section 3-1. With the availability of y, the presence of a fault can now be detected by defining the residual as

$$r(y) := \frac{N(s)L(s)}{\alpha(s)} y + \frac{N(s)}{\alpha(s)} \begin{bmatrix} \hat{x}_a \\ 0 \end{bmatrix},$$

where $r \in \mathbb{R}$ is the output of the FD filter.

For the use case, a non-zero residual does not necessarily imply the input signal is faulty, as a range of signals can be seen as healthy. Therefore, to determine whether a signal gets detected as faulty or not, a threshold on the energy is determined, i.e.

$$e(r) = \int_0^T ||r||_2^2 \, dt,$$

where $e \in \mathbb{R}_+$ is the residual energy and T is the finite time duration of the measurement. If this threshold is exceeded, the signal gets detected as faulty. To determine the threshold, a set of healthy training data is used to find the maximum energy for which a signal is healthy, which gets multiplied by a factor $\mu \in \mathbb{R}_+$, to be determined by the user, to make the detection method more robust. The threshold $\mathcal{T}$ can be formulated as

$$\mathcal{T} = \mu \cdot \max(e_h), \tag{3-4}$$

where $e_h \in \mathbb{R}_+$ is a vector containing all energies of residuals of healthy signals. Other metrics, such as the maximum of the residual, can also be used to determine whether a signal is faulty, and are possibly computationally faster. However, since the residual is an oscillating signal, the energy of the residual keeps increasing over time, highlighting the presence of a fault advantageously. The FD filter decision can now be described as

$$f = \begin{cases} 0, & e(r(y)) \leq \mathcal{T}, \\ 1, & e(r(y)) > \mathcal{T}. \end{cases}$$

The procedure of detecting faults is schematically shown in Figure 3-4.

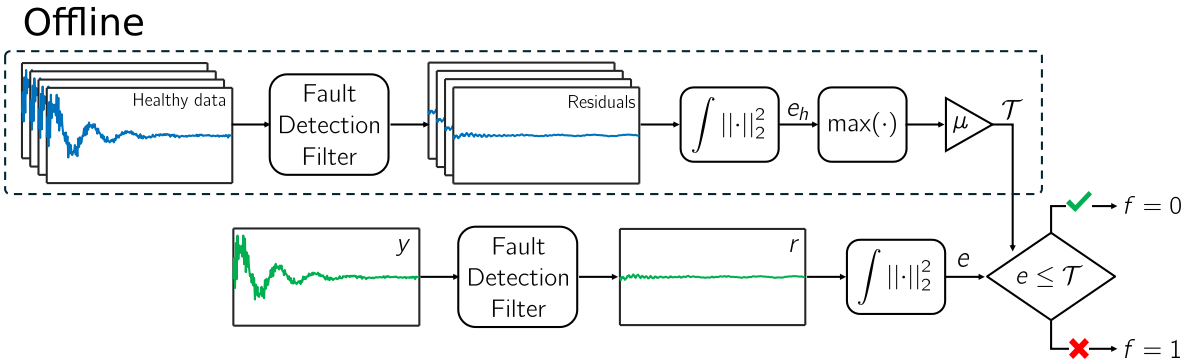

Figure 3-4: FD filter appliance. The offline area shows how threshold $\mathcal{T}$ is calculated, which is used to detect a fault in y.

In what follows are three design methods to determine the denominator $\alpha(s)$.

3-2-1 Fault Detection Filter Poles

In previous work, the FD filter denominator is chosen to be an arbitrary transfer function with sufficiently high order with roots with negative real parts to make the transfer function proper and stable [13]. However, the denominator can impact the performance of the FD filter when using thresholds to detect faults. Three different ways to choose the FD filter denominator are proposed.

I: Increasing residual energy

Since the energy of the residual is used to detect faults, one way to choose the FD filter denominator is to increase the energy of the residual. This can be done by placing two

marginally stable poles of the FD filter at $s = \pm\omega_r i$, where ω_r is the desired residual frequency. Since this pole pair does not have a negative real part, the residual does not converge to zero, thus the energy keeps increasing. This means the presence of a fault is amplified more as a faulty residual represents a sinusoid with a larger amplitude than a healthy residual. The frequency can be chosen such that a desired amount of oscillations happens within the known time frame, i.e.

$$\omega_r = \frac{n_o}{T},$$

where n_o is the desired number of oscillations. The other poles are chosen arbitrarily stable since they do not influence the long-term behaviour of the residual. The denominator can be written as

$$\alpha(s) = (s^2 + \omega_r^2) \cdot \beta(s),$$

where $\beta(s)$ is a stable transfer function with order $d_N - 2$, ensuring the denominator is of order d_N.

II: Eigenvalues of faulty system

The zeros in the numerator of the FD filter are exactly equal to the eigenvalues of the A-matrix of the healthy system, i.e.

$$N(s)L(s) = \det(A - sI).$$

This can be explained by the fact that the FD filter output should equal zero for these frequencies. In the same way, the filter's output is maximized when the denominator of the filter captures the dynamics of a fault, i.e.

$$\alpha(s) = \det(A_f - sI).$$

For every model-based fault, a faulty model can be constructed. The eigenvalues of the faulty model are used in the denominator.

III: Numerator enhancement

Since faulty data is available, it can be used to maximize the effect of a fault on the residual. This is done by adding extra zeros to the FD filter transfer function, designed such that the effect of a fault is maximized. The modified FD filter can be formulated as

$$r(y) := \frac{M(s)N(s)L(s)}{\alpha(s)}y + \frac{M(s)N(s)}{\alpha(s)}\begin{bmatrix} \hat{x}_a \\ 0 \end{bmatrix}.$$

As can be seen, the numerator is enhanced with $M(s)$. The denominator of the filter is chosen arbitrarily. Appendix A shows how $M(s)$ is constructed.

The first approach is chosen to determine the poles of the FD filter, as it is best suited for the use case and the chosen FD filter decision method. After a fault has been detected, the aim is to isolate the fault. For this, an FI filter is designed.

3-3 Fault Isolation Filter

In this section, two methods to construct an FI filter are described. First, a model-matching FI filter is explained in Section 3-3-1. Then, the residual matching filter is described in Section 3-3-2.

3-3-1 Model-matching FI filter

The model-matching FI filter is similar to the FD filter in construction. However, instead of using a model that represents the system's healthy dynamics, it employs a model that represents the dynamics of a particular fault. By utilizing faulty data, a model can be identified for each fault using the same identification techniques used for healthy data. A small residual is expected when the signal containing a specific fault is fed into the corresponding FD filter. A threshold can be chosen to determine when a signal corresponds to a specific fault. If the threshold is exceeded, the signal is likely healthy or may contain a different fault.

3-3-2 Residual matching FI filter

The residual matching FI filter uses estimates of the residual for every fault in a linear regression model [17]. To train the filter, faulty data with correct classifications must be available. First, each signal y in the dataset is subjected to the earlier constructed FD filter. Then, for every classification, the mean residual,

$$\rho = \frac{1}{N}\sum_{i=1}^{N} r_i,$$

is calculated, where $\rho \in \mathbb{R}$ and N is the number of signals used per classification. Using the mean residual per classification, an estimation matrix R is designed,

$$R = \begin{bmatrix} \rho_1 & \dots & \rho_{n_f} \end{bmatrix},$$

where $R \in \mathbb{R}^{n \times n_f}$ where $n = \frac{T}{\Delta t}$ is the length of the signal. Linear regression is used to determine what residual in R best resembles a given residual r. To achieve this, a probability vector

$$\varphi = \begin{bmatrix} \varphi_1 \\ \vdots \\ \varphi_{n_f} \end{bmatrix},$$

is used as a regressor, where $\varphi \in \mathbb{R}_+^{n_f}$. As a probability vector, the regressor must be greater than zero and sum to one. The linear regression model is formulated accordingly as

$$\begin{cases} \min_{\varphi} & ||R\varphi - r||_2 \\ \text{s.t.} & \sum_{i=1}^{n_f} \varphi_i = 1, \\ & \varphi \geq \mathbf{0}. \end{cases}$$

Here r is the residual of the signal y being isolated. The result is the vector φ that describes the contribution of every fault in terms of probability. The fault correlated with the highest

probability is assumed to be the present fault. A schematic visualisation can be seen in Figure 3-5, where the offline environment shows how the filter is constructed. Furthermore, an example of one signal that is isolated is shown, which happens online.

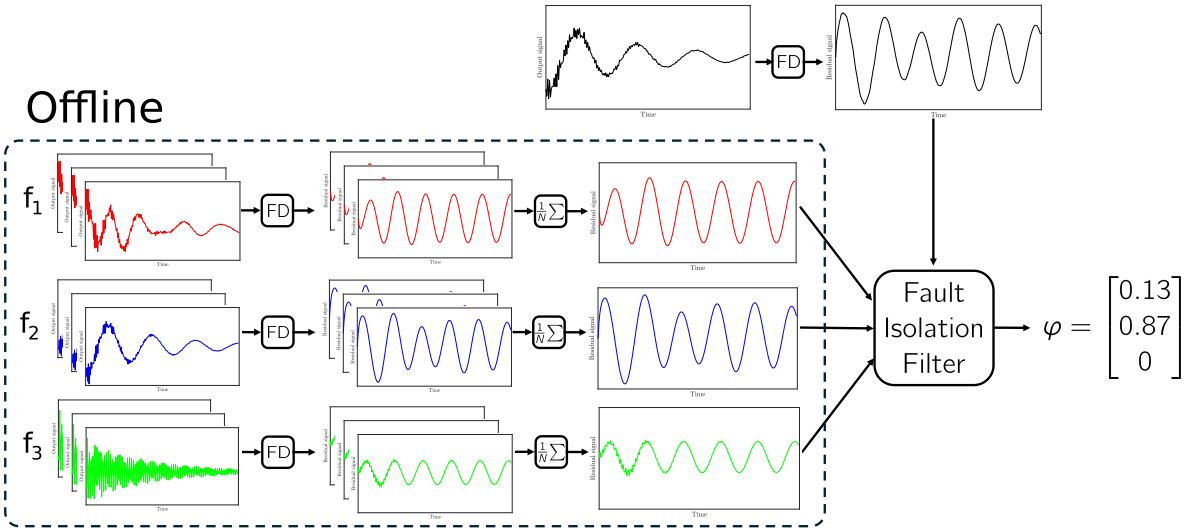

Figure 3-5: Schematic overview of the design and appliance of the FI filter.

The residual matching filter is chosen to isolate faults. When the FD filter and FI filter are designed, they can be tested using simulation and experimental data.

Chapter 4

Simulation and Experimental Results

In this section, the results of applying the FD and FI filters to simulated and experimental data are shown. First, the structure of the data is analysed in Section 4-1. Then the FD and FI filters are tested for the data in Sections 4-2 and 4-3, respectively.

4-1 Data analysis

To test both filters, simulated data and experimental data are used. Simulated data is generated based on knowledge of faults and the nature of experimental data. 4500 signals are generated, where 2475 are healthy signals and every fault class contains 225 signals, meaning nine fault classes are considered. The dataset can be visualised by plotting the amplitude, damping and frequency against each other for every type of fault, which are used to generate the signals. The result is shown in Figure 4-1.

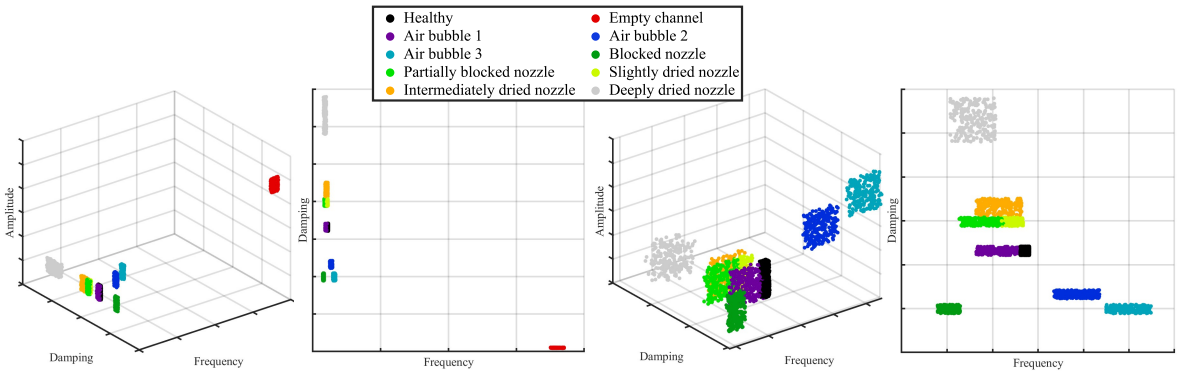

Figure 4-1: Simulated dataset visualisation. From left to right, the first plot shows the dominant amplitude, damping and frequency of every signal corresponding to a fault, the second plot is a 2D version showing only the dominant damping and frequency. The third and fourth plots are zoomed-in versions of the first two to show the difference between the signals without the empty channel (see red dots in the first plot).

Experimental data is collected by purposely drying the nozzle. The procedure to overcome a drying nozzle, called prefiring, is performed to collect data ranging from deeply dried to healthy. This way, three datasets are generated. The first is healthy, meaning enough prefires are performed to remove all dried ink from the nozzle. The second set contains intermediately dried nozzles. Here, some but not enough prefires are performed to remove some of the dried ink. The third set contains deeply dried nozzles, so no prefiring is performed. The same visualisation as with the simulated dataset is shown in Figure 4-2, where the key numbers of the signals were calculated using a proprietary CPP Fast Fourier Transform toolbox. The three clusters can be observed, where the intermediately dried nozzles form a transition between the healthy and deeply dried nozzles.

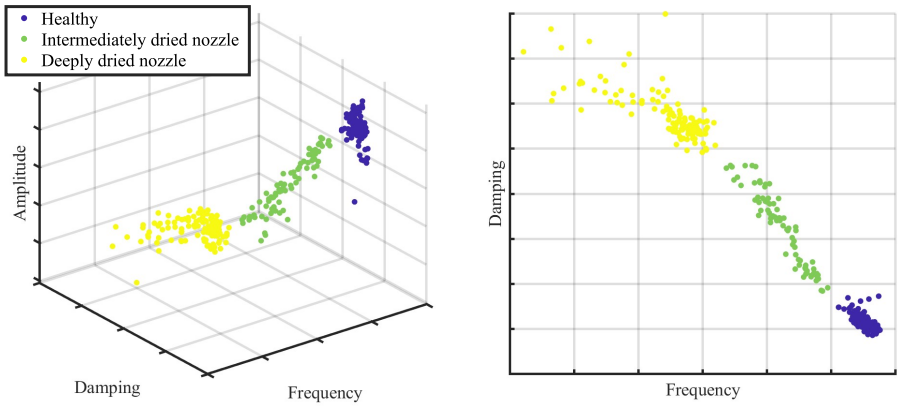

Figure 4-2: Experimental dataset visualisation. The left plot shows the dominant amplitude, damping and frequency of every signal corresponding to a fault. The right plot is a 2D version showing only the dominant damping and frequency.

4-2 Fault Detection Results

Some fault classes for the simulated dataset include signals that show characteristics similar to healthy signals. To prevent a large number of false positives, i.e. healthy signals being detected as faulty by the FD filter, the multiplication factor defined in (3-4) is chosen as $\mu = 1$, meaning the healthy signal with the highest energy in the training data is used as a threshold. For the experimental dataset, because healthy and faulty signals are better distinguished, a multiplication factor of $\mu = 2$ is chosen to prevent false negatives. This means that the threshold is determined by doubling the highest energy of all healthy signals.

To test the performance of the filter, a confusion matrix is generated using test data for simulated and experimental data.

Table 4-1: Confusion matrix for FD filter using simulated data.

		FD filter	
		Healthy	Faulty
Ground truth	Healthy	2474	1
	Faulty	92	1933

Table 4-2: Confusion matrix for FD filter using experimental data.

		FD filter	
		Healthy	Faulty
Ground truth	Healthy	48	0
	Faulty	0	99

As seen in Table 4-1, for the test data, only one healthy signal is detected as a fault by the filter. 92 faulty signals are detected as healthy, 83 of which are slightly dried nozzles and 9 of which are intermediately dried nozzles. This is because of the signal variance, meaning some slightly intermediately dried nozzles show characteristics of a healthy nozzle. The percentage of correct detections for the simulated dataset is 97.9%. For the current method used by CPP this is 94.8%, meaning the FD filter slightly outperforms. For experimental data, no false negatives nor any false positives were made, meaning this percentage is 100%, as can be seen in Table 4-2. CPP's current method was able to detect 91.8% of faults, meaning the FD filter performs better. The fact that the filter performs better for experimental data can be explained by the fact that healthy and faulty data are better distinguished for experimental data. For example, in the simulated dataset, slightly dried nozzles are generated such that they can easily be misinterpreted for healthy signals, especially due to noise. For experimental data, faults have higher magnitudes. To visualise the results, a scatter plot is created where the horizontal axis is the energy of every signal, and the vertical axis separates the fault class of every signal. The vertical line indicates the threshold for the energy. In Figure 4-3, this is done for simulated data.

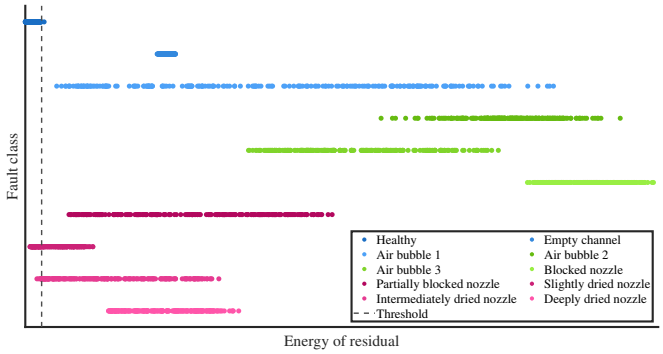

Figure 4-3: Visualisation of FD filter results for simulated data. The horizontal axis shows the energy of the residual for every signal. The vertical axis separates the different fault classes.

Figure 4-4 shows the same plot for experimental data.

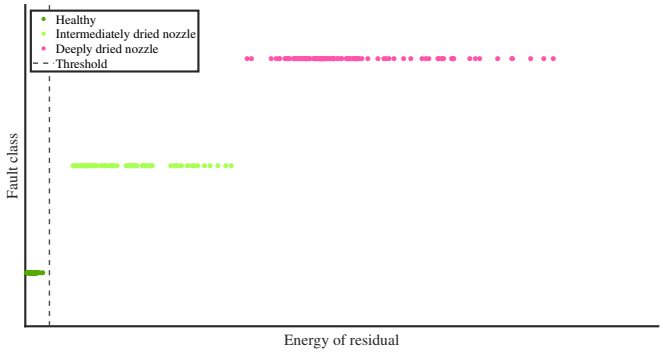

Figure 4-4: Visualisation of FD filter results for experimental data. The horizontal axis shows the energy of the residual for every signal. The vertical axis separates the different fault classes.

When a fault is detected by the FD filter, the aim is to isolate the fault. This is done by the FI filter, of which the results are shown in the next section.

4-3 Fault Isolation Results

The FI filter is tested for the simulated and experimental datasets. The confusion matrix can be seen in Table 4-3.

Table 4-3: Confusion matrix showing the FI filter results for simulated data. Abbreviations are explained in Section 2.

		FI filter								
		EC	AB1	AB2	AB3	BN	PBN	SDN	IDN	DDN
Ground truth	EC	154	0	0	0	0	0	27	0	44
	AB1	0	173	0	0	6	0	46	0	0
	AB2	0	0	222	2	1	0	0	0	0
	AB3	0	0	0	225	0	0	0	0	0
	BN	0	0	0	0	225	0	0	0	0
	PBN	0	1	0	0	0	172	52	0	0
	SDN	0	0	0	0	0	0	225	0	0
	IDN	0	0	0	0	0	99	117	9	0
	DDN	0	0	0	0	0	0	0	8	217

When testing the overall performance of the filter, the Harmonic Mean Average (HMA) is calculated which is a standard metric to evaluate the overall performance of fault isolation schemes [22]. For a square confusion matrix $\mathcal{A} \in \mathbb{R}^{n \times n}$, the HMA is calculated as

$$HMA(\mathcal{A}) = \frac{\sum_{i=1}^{n} \mathcal{A}_{ii}}{\sum_{i=1}^{n} \sum_{j=1}^{n} \mathcal{A}_{ij}},$$

which is the same as the trace of the matrix divided by the sum of all matrix entries. For the confusion matrix in Table 4-3, the HMA is 0.801, meaning 80.1% of faulty signals are isolated correctly. However, since the FI filter calculates a probability vector φ, a design choice can be to only regard signals where one of the entries in φ is 0.8 or larger, i.e. the filter is more than 80% sure the isolation is done correctly. This percentage is an arbitrary value and depending on the application this percentage can be varied. When considering this constraint, the FI filter can be tested again. Results are shown in Table 4-4.

Table 4-4: Confusion matrix showing the FI filter results for simulated data where only isolations with 80% or more certainty are isolated.

		FI filter								
		EC	AB1	AB2	AB3	BN	PBN	SDN	IDN	DDN
Ground truth	EC	125	0	0	0	0	0	0	0	0
	AB1	0	81	0	0	0	0	16	0	0
	AB2	0	0	146	0	0	0	0	0	0
	AB3	0	0	0	133	0	0	0	0	0
	BN	0	0	0	0	221	0	0	0	0
	PBN	0	0	0	0	0	16	0	0	0
	SDN	0	0	0	0	0	0	163	0	0
	IDN	0	0	0	0	0	52	62	0	0
	DDN	0	0	0	0	0	0	0	0	63

When only regarding signals with more than 80% isolation certainty, the HMA of the filter is 0.879, i.e. 87.9% correct isolations. When imposing this bound on the certainty, not all signals get isolated. Here, 53.2% of data is isolated. The other 46.8% of signals do not get isolated because the output of the FI filter did not contain an entry of 0.8 or higher. It can be observed that the intermediately dried nozzles are all isolated wrongly. When looking at the data, this can be explained by the fact that these signals are similar to the slightly dried nozzle and partially blocked nozzle. When disregarding the intermediately dried nozzle as a fault class in the table, the HMA is 0.983. Results can thus be improved by redefining similar fault classes, like combining slightly and intermediately dried nozzles as one fault class. Moreover, other features of the residual can be examined further to see if they can be used to better distinguish similar fault classes. CPP's current isolation method only correctly isolates 41.3% of faulty signals. It should be noted that the dried nozzle fault classes and air bubble 1 and 2 are not yet defined for this method, meaning this method would at most correctly isolate 44.4% of faulty signals. The fault classes that the current method can isolate thus get isolated correctly 93.0% of the time. To properly compare the two methods, the current method should be expanded to include all fault classes in the dataset.

The FI filter is also tested for experimental data. Results are shown in Table 4-5.

Table 4-5: Confusion matrix for FI filter using experimental data.

		FI filter	
		IDN	DDN
Ground truth	IDN	38	1
	DDN	0	60

The HMA of the FI filter for experimental data is 0.990, as only one signal was wrongly isolated. Therefore, no bound is imposed on the certainty. The current method of CPP correctly isolates 88.3% of faulty signals, mainly because intermediately dried nozzles get isolated as healthy, which is not possible in the FI filter. This is due to the current method performing FD and FI at once, instead of only isolating faulty signals. In what follows, the results are compared to other methods.

4-3-1 Comparison to other methods

To evaluate the results of the FI filter for both datasets, other methods are used on the same datasets for comparison. The proposed FI filter uses the residual signal r and determines what fault class the residual best belongs to using linear regression. However, the output signal y can be used to perform the same task. Moreover, the search method can be altered to a machine learning method, such as the k-nearest neighbours algorithm (KNN). KNN is a method where an object is classified by a majority vote of its k-nearest neighbours in the feature space [23]. Here, the object, which is a signal, is assigned to the class of that single nearest neighbour. Furthermore, the training size is decreased by 50% and 90% for all methods to evaluate the importance of the training size. The results for the comparison of the methods for simulated data are shown in Table 4-6.

Table 4-6: FI results when using linear regression (LR) with and without certainty bound and KNN with different training sizes for simulated data.

Training size	Signal used	LR	LR ($\varphi \geq 80\%$)	KNN
2025	r	0.801	0.879	0.816
	y	0.763	0.881	0.815
1013	r	0.797	0.879	0.815
	y	0.768	0.875	0.782
203	r	0.772	0.867	0.767
	y	0.661	0.820	0.729

As can be seen, linear regression with the certainty bound is more robust to a shortage of data than the other methods. Also, using r when having less data improves results in comparison to using y.

The experimental data is tested the same way. The results are shown in Table 4-7. As can be seen, all methods perform perfectly or close to perfect.

Table 4-7: FI results when using linear regression (LR) with and without certainty bound and KNN with different training sizes for experimental data.

Training size	Signal used	LR	LR ($\varphi \geq 80\%$)	KNN
147	r	0.990	1.00	0.990
	y	0.980	1.00	0.980
74	r	1.00	1.00	1.00
	y	0.980	1.00	0.980
15	r	0.990	1.00	0.990
	y	0.980	1.00	0.980

Chapter 5

Conclusion

This report has presented a method to detect and isolate faults in the ink channels of CPP's printers. To detect faults, a model-based FD filter has been designed. Grey-box system identification has been used to construct the model, which has been regarded as a successful method since the FD filter has been shown to work sufficiently accurately for both simulated and experimental data. The filter has outperformed the current methods of CPP to detect faults.

Based on results for simulated data for the FI filter, it has been concluded that the filter outperforms the current isolation method of CPP, which does not include five of nine fault classes. Using linear regression using a certainty bound outperforms other investigated methods regardless of the signal used and the training size. The difference between using r or y for this method is negligible, although using r in the case of a small training size has increased results by 5.7%. Not all test data gets isolated when using linear regression with the certainty bound. If it is desired to isolate all signals, one of the other methods should be used. For almost all cases, KNN has performed better than linear regression. Only when using a small training size and using r, linear regression has performed slightly better. Also, for all cases, using r has performed better than using y. It can be concluded that when it does not matter if all signals get isolated, using linear regression with a certainty bound has been the best option. When it is desired to isolate all signals, KNN has been a better option when enough training data is available, but it loses information about the certainty of the isolation process.

Based on the FI results using experimental data, it can be concluded that the filter has outperformed CPP's current method, mainly because of faulty signals being isolated as healthy. For this dataset, the method and training size have barely influenced the filter's performance, as all methods have been 98% or more accurate. A bigger dataset with more faults should be analysed to conclude whether the proposed or another method works best for experimental data. If results stay similar, the method with the lowest computational complexity where all data is isolated could be chosen, meaning linear regression with the certainty bound should not be chosen.

In future work, the first search direction is to apply both filters to more data. For simulated data, investigate the fidelity of the data, especially by evaluating if using uniform distributions

is an accurate representation. For experimental data, analyse more fault scenarios. The second search direction is to investigate the potential of the model-matching FI filter, as it is based on the same principle as the FD filter which is shown to work accurately. Moreover, the use case is suitable for fault estimation, which is done when a fault is isolated and the magnitude of the fault is desired to be known. The percentage of blockage or dryness of the nozzle or the size of an air bubble can be computed this way. Lastly, further investigate the role of the FD filter denominator and design a more systematic way of choosing it.

Appendix A

Numerator enhancement for FD filter

The process of constructing an enhanced numerator for the FD filter is shown. First, a faulty residual can be simulated using the FD filter,

$$r_f = \frac{N(s)L(s)}{\alpha(s)} y_f + \frac{N(s)}{\alpha(s)} \begin{bmatrix} \hat{x}_a \\ 0 \end{bmatrix},$$

where y_f is the average signal of a dataset of faulty signals belonging to the same fault class and $\alpha(s)$ is a randomly chosen stable transfer function of high enough order to enhance the numerator, i.e. $O(\alpha(s)) \geq d_N + d_M$. The linear program is formulated as

$$\begin{cases} \max_{\bar{M}} & \bar{M} r_f & \\ \text{s.t.} & \bar{N}\bar{H} & = 0, \\ & \|\bar{N}\|_\infty & \leq 1, \\ & \|\bar{M}\|_\infty & \leq 1 \end{cases}$$

Here, $\bar{M}$ is used to later enhance the numerator of the FD filter. The solution to the optimization program can be found by using the eigenvector corresponding to the maximum eigenvalue of Q_x [13], where

$$Q_x := \bar{D}\bar{D}^T,$$

where

$$\bar{D} := \begin{bmatrix} \beta^\star \\ \beta^\star D \\ \vdots \\ \beta^\star D^{d_M} \end{bmatrix},$$

where

$$D = \begin{bmatrix} 0 & 0 & 0 & \cdots & 0 & 0 \\ 0 & 0 & \omega & \cdots & 0 & 0 \\ 0 & -\omega & 0 & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & & 0 & 40\omega \\ 0 & 0 & 0 & \cdots & -40\omega & 0 \end{bmatrix},$$

and every i-th element of $\beta^\star$ is

$$\beta_i^\star = \int_0^T b_i(t) r_f(t) \mathrm{d}t,$$

where

$$b_i(t) := \begin{cases} \cos\left(\frac{i}{2}\omega t\right) & i: \text{ even} \\ \sin\left(\frac{i+1}{2}\omega t\right) & i: \text{ odd} \end{cases}, \quad \omega := \frac{2\pi}{T}, \quad i \in \{0, 1, \cdots, 80\}\ .$$

Finally, when Q_x is constructed, $M(s)$ can be found as

$$M(s) := \det(Q_x - sI).$$

This function is used to enhance the numerator of the FD filter.

Bibliography

[1] Canon Production Printing. Canon Colorado M-series. https://www.canon-europe.com/business/products/wide-format-printers/colorado-m-series/, November 2023.

[2] Canon Production Printing. UVgel. https://cpp.canon/products/uvgel/, November 2023.

[3] Canon Production Printing. https://cpp.canon/about/, November 2023.

[4] Arjan Fraters, Marc van den Berg, Youri de Loore, Hans Reinten, Herman Wijshoff, Detlef Lohse, Michel Versluis, and Tim Segers. Inkjet nozzle failure by heterogeneous nucleation: Bubble entrainment, cavitation, and diffusive growth. *Phys. Rev. Appl.*, Dec 2019.

[5] Johannes Mathieu Marie Simons and Mark Alexander Groninger. Printing apparatus with measuring circuit for diagnosis of condition of each electromechanical transducer. https://patents.google.com/patent/US6682162B2/, January 2004. Patent number: US6682162B2.

[6] Byung Hun Kim, Na Rae Park, and Hyun Ho Shin. Error detection apparatus of inkjet printer head and error detection method thereof. https://patents.google.com/patent/US8894178B2/, November 2014. Patent number: US8894178B2.

[7] Marko Mihailovic, Tjerk E. C. Hummel, Johannes M. M. Simons, Hylke Veenstra, Cornelis W. M. Venner, and Amol A. Khalate. Error detection apparatus of inkjet printer head and error detection method thereof. https://patents.google.com/patent/US10144215B2/, December 2018. Patent number: US10144215B2.

[8] Yawen Zhang, Feng Shi, Chenguang Zhang, Xin Sheng, Yunhao Zhong, Hui Chong, Zhanjun Yang, and Chengyin Wang. Detection of avian influenza virus h9n2 based on self-driving and self-sensing microcantilever piezoelectric sensor. *Chinese Chemical Letters*, 34(4):107700, 2023.

[9] Arka Ghosh, David John Edwards, M. Reza Hosseini, Riyadh Al-Ameri, Jemal Abawajy, and Wellington Didibhuku Thwala. Real-time structural health monitoring for concrete beams: A cost-effective 'industry 4.0' solution using piezo sensors. *International Journal of Building Pathology and Adaptation*, 39(2):283–311, 2020.

[10] Hassan Elahi. The investigation on structural health monitoring of aerospace structures via piezoelectric aeroelastic energy harvesting. *Microsystem Technologies*, 27(7):2605–2613, 2020.

[11] Jingyu Kang, Tao Liu, Yao Lu, Linlin Lu, Kai Dong, Shujuan Wang, Bo Li, Ying Yao, Yuan Bai, and Wei Fan. Polyvinylidene fluoride piezoelectric yarn for real-time damage monitoring of advanced 3d textile composites. *Composites Part B: Engineering*, 245:110229, 2022.

[12] Hans Reinten, Yogesh Jethani, Arjan Fraters, Roger Jeurissen, Detlef Lohse, Michel Versluis, and Tim Segers. Resonance behavior of a compliant piezo-driven inkjet channel with an entrained microbubble, Feb 2023.

[13] Peyman Mohajerin Esfahani and John Lygeros. A tractable fault detection and isolation approach for nonlinear systems with probabilistic performance. *IEEE Transactions on Automatic Control*, 61(3):633–647, 2016.

[14] Erik Frisk and Mattias Nyberg. Residual generation for fault diagnosis of systems described by general linear differential-algebraic equations. *IFAC Proceedings Volumes*, 35(1):137–142, 2002. 15th IFAC World Congress.

[15] Jingwei Dong, Kaikai Pan, Sergio Pequito, and Peyman Mohajerin Esfahani. Robust multivariate detection and estimation with fault frequency content information, 2024.

[16] P.J.B Hacking. Classification of jetting behavior based on self-sensing piezo actuators, 2019.

[17] Chris van der Ploeg. Fault detection and isolation for lateral control of an autonomous vehicle, 2018.

[18] Kye-Si Kwon and Wousik Kim. A waveform design method for high-speed inkjet printing based on self-sensing measurement. *Sensors and Actuators A: Physical*, 140(1):75–83, 2007.

[19] Detlef Lohse. Fundamental fluid dynamics challenges in inkjet printing. *Annual Review of Fluid Mechanics*, 54(Volume 54, 2022):349–382, 2022.

[20] Estimate ode parameters of linear grey-box model. https://nl.mathworks.com/help/ident/ref/greyest.html, May 2024.

[21] Option set for greyest. https://nl.mathworks.com/help/ident/ref/greyestoptions.html, March 2024.

[22] TAJ van Esch, F Ghanipoor, CG Murguia Rendon, F Takahashi, N van de Wouw, P Mohajerin Esfahani, and JJM van de Wijdeven. Hybrid model-based and data-based fault detection, isolation, and estimation for wafer handler robots.

[23] T. Cover and P. Hart. Nearest neighbor pattern classification. *IEEE Transactions on Information Theory*, 13(1):21–27, 1967.