Data-Driven and Adaptive Statistical Residual Evaluation for Fault Detection with an Automotive Application

Carl Svärd, Mattias Nyberg, Erik Frisk and Mattias Krysander

Original Publication:

Carl Svärd, Mattias Nyberg, Erik Frisk and Mattias Krysander, Data-Driven and Adaptive Statistical Residual Evaluation for Fault Detection with an Automotive Application, 2014, Mechanical systems and signal processing, (45), 1, 170-192.
http://dx.doi.org/10.1016/j.ymssp.2013.11.002

Data-Driven and Adaptive Statistical Residual Evaluation for Fault Detection with an Automotive Application

Carl Svärd[a], Mattias Nyberg[a], Erik Frisk[b], Mattias Krysander[b]

[a]*Systems Development, Scania CV AB, SE-15187 Södertälje, Sweden*
[b]*Department of Electrical Engineering, Linköping University, SE-58183 Linköping, Sweden*

Abstract

An important step in model-based fault detection is residual evaluation, where residuals are evaluated with the aim to detect changes in their behavior caused by faults. To handle residuals subject to time-varying uncertainties and disturbances, which indeed are present in practice, a novel statistical residual evaluation approach is presented. The main contribution is to base the residual evaluation on an explicit comparison of the probability distribution of the residual, estimated online using current data, with a no-fault residual distribution. The no-fault distribution is based on a set of a-priori known no-fault residual distributions, and is continuously adapted to the current situation. As a second contribution, a method is proposed for estimating the required set of no-fault residual distributions off-line from no-fault training data. The proposed residual evaluation approach is evaluated with measurement data on a residual for fault detection in the gas-flow system of a Scania truck diesel engine. Results show that small faults can be reliable detected with the proposed approach in cases where regular methods fail.

Keywords: fault diagnosis, fault detection, residual evaluation, statistical hypothesis testing, automotive diesel engine

1. Introduction

Fault diagnosis is becoming more and more important with the increasing demand for dependable technical systems, driven mostly by economical, environmental, and safety, incentives. One example is automotive systems, where good fault diagnosis is essential in order to meet customer demands regarding up-time, efficient repair and maintenance, and also to fulfill on-board diagnosis (OBD) legislative regulations.

Email addresses: carl.svard@scania.com (Carl Svärd), mattias.nyberg@scania.com (Mattias Nyberg), frisk@isy.liu.se (Erik Frisk), matkr@isy.liu.se (Mattias Krysander)

Model-based fault diagnosis typically comprises fault detection and isolation [13], and the fault detection part contains the essential steps residual generation and residual evaluation. In the first step, a model of the system is used together with measurements to generate residuals. In the second step, the residuals are evaluated with the aim to detect changes in the residual behavior caused by faults in the system. This works concerns the second step, residual evaluation.

Ideally, residuals are signals that are zero when no faults are present in the system, and non-zero otherwise. Due to the presence of uncertainties and disturbances, caused by for instance modeling errors, measurement noise, and unmodeled phenomena, residuals typically however deviate from zero even in the no-fault case. Moreover, due to changes in the operating mode of the system, the magnitude of these uncertainties and disturbances is time-varying, causing the behavior of residuals to be non-stationary. An illustration is given by Figure 1, where a residual for fault detection in the gas-flow system of a truck diesel engine is shown. Clearly, the residual is not zero in the no-fault case, and it is obvious that the residual exhibit non-stationary features. It can also be noted that the difference between the residual in the no-fault and fault cases is time-varying. Nevertheless, the fact that there is a difference implies that the present fault is potentially detectable. Note also that the residual does not exhibit any periodic time-variations.

There are two main approaches [22] for residual evaluation; statistical [60, 27, 7, 50, 3, 14, 59, 30, 48] and norm-based [23, 25, 26, 53, 18, 62, 63, 33, 4, 36, 1]. Statistical approaches exploits the framework of statistical hypothesis testing in order to detect changes in some parameter of the probability distribution of the residual, typically by means of likelihood ratio testing [28]. In norm-based approaches, residual evaluation is typically done by adaptive or constant thresholding of some norm of the residual.

Apparently, when encountering a residual as the one depicted in Figure 1, neither statistical-based approaches assuming stationary probability distributions, nor norm-based approaches using constant thresholds, would be successful. A potential solution is to consider adaptive thresholds [19, 24], and use a-priori knowledge, either qualitative [33, 62, 31, 23] or quantitative [53, 25, 47], to derive non-constant thresholds to take the time-varying uncertainties and disturbances into account. Furthermore since the residual depicted in Figure 1 is non-periodic, diagnosis approaches for machines working in non-stationary operating conditions [16, 52, 43, 51] are not applicable.

This paper instead proposes an adaptive statistical residual evaluation method, which exploits quantitative a-priori knowledge in the form of data.

The main contribution is to base the residual evaluation on an explicit comparison of the probability distribution of the residual, estimated on-line using current data, with a no-fault residual distribution. The no-fault distribution is based on a set of a-priori known no-fault distributions and to handle changes in the operating mode of the system, and thus time-varying residual features, it is continuously adapted to the current operating mode of the system. The comparison is done in the framework of statistical hypothesis testing by application of

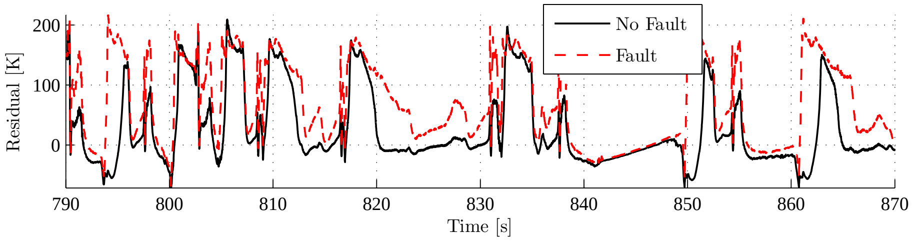


Figure 1: A residual for fault detection in the gas-flow system of a heavy-duty truck diesel engine in the no-fault (solid) and fault (dashed) cases.

the Generalized Likelihood Ratio (GLR). As a second contribution, a method is proposed for estimating the required set of no-fault residual distributions off-line from no-fault training data. Thus, using the method for distribution estimation, the overall residual evaluation method becomes fully data-driven and no assumptions regarding the properties of the probability distribution of the residual, nor the properties of the faults to be detected, are made.

The paper is organized as follows. Section 2 discusses and formalizes the problem setup and the residual evaluation problem is formulated in the framework of statistical hypothesis testing. In Section 3, the GLR is utilized to design a preliminary test statistic for the residual evaluation hypotheses, and the emerging likelihood maximization problems are considered. In Section 4, the preliminary test statistic is improved in terms of required computational effort, and a residual evaluation algorithm suitable for implementation in an online environment is given. Section 5 presents an off-line algorithm for learning no-fault residual distributions from no-fault training data. In Section 6 the proposed residual evaluation approach is applied to a residual for fault detection in the gas-flow system of a real Scania truck diesel engine. Finally, Section 7 concludes the paper.

2. Problem Formulation

The residual evaluation problem, as considered in this work, is formally stated in this section.

2.1. Prerequisites

A *residual*, r, is considered to be the output from a *residual generator*, taking measurements from a *system* as input. Typically, the measurements consists of the input and output of the system.

The system is considered to be subject to faults, and the intention is to detect if any fault is present in the system by monitoring the behavior of the residual. Note that if a set of residuals sensitive to different faults is used, faults can also be isolated, see for example [13].

The system typically operates in a number of different *operating modes*, and normal operation usually involves several of these modes. For an example, consider a heavy-duty truck diesel engine, for which a residual is shown in Figure 1. Naturally, this system is designed to operate in a number of different operating modes typically characterized by engine torque, engine speed, ambient temperature, ambient pressure, etc.

The considered setup most often contains uncertainties in the form of measurement noise or, in the case of a model-based residual generator, modeling errors. Typically, the magnitudes and nature of the uncertainties are different for different operating modes of the system. For example, a sensor may be more or less sensitive to noise in different operating modes, and a model may be more accurate in one operating mode than another. Since the operating mode of the system varies in time, so does the magnitudes and nature of the uncertainties. This is the cause of the non-ideal residual behavior illustrated in Figure 1.

It is assumed that during on-line operation, the current operating mode of the system is unknown. In addition, it is also assumed that the probability that the system is in a specific mode is unknown. In this sense, the system can be considered to be subject to an unknown, i.e., unmeasurable, input signal, determining the current operating mode. Regarding in particular the first assumption, it is considered to be hard to quantify and measure all factors, internal and external, that determine the current operating mode of a system. Furthermore, these factors may be different for different individuals of the system, or may change over time. However, even if its is possible to determine a set of measured signals that determines the operating mode, all signals may not be available for the residual evaluation scheme due to for example fault decoupling principles, or architectural constraints in the control system software. In addition, even if all signals are available, they may as well be subject to faults. The second assumption is mainly motivated by the fact that the operation of a system differs between different individuals of the same system, and may change over time or due to external unmeasurable factors.

2.2. Probabilistic Framework

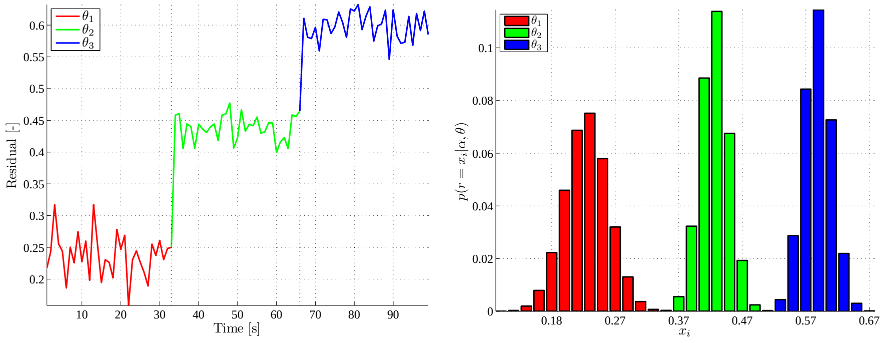
To handle the uncertain environment described above, a probabilistic framework is adopted. Let the discrete random variable R with range $\mathcal{X} = \{x_1, x_2, \ldots, x_M\}$, represent the discretized and sampled value of the residual, and let r denote a particular outcome of R.

For a given specific operating mode i of the system, the probability that $R = r$ is assumed to be characterized by the probability mass function (pmf)

$$p\left(r|\theta_i\right) = \Pr\left(R = r|\theta_i\right) = \theta_{ij}, \quad \text{if } r = x_j, \tag{1}$$

for $j = 1, \ldots, M$. The pmf (1) is fully parametrized by $\theta_i = (\theta_{i1}, \theta_{i2}, \ldots, \theta_{iM})$, where the θ_{ij} are required to fulfill

$$\theta_{ij} \geq 0, \quad j = 1, 2, \ldots, M \qquad \text{and} \qquad \sum_{j=1}^{M} \theta_{ij} = 1. \tag{2}$$



(a) Residual

(b) Distribution of the Residual

Figure 2: Example of a sample from a mixture distribution in the form (3) with 3 components θ_1, θ_2, and θ_3, and mixture weights $\alpha_1 = \alpha_2 = \alpha_3 = \frac{1}{3}$.

Under the assumption that there is in total K operating modes, the probability that $R = r$ can be characterized by the K-component mixture distribution given by the pmf

$$p\left(r|\alpha,\theta\right) = \sum_{i=1}^{K} \alpha_i \, p\left(r|\theta_i\right) \tag{3}$$

with $\alpha = (\alpha_1, \alpha_2, \ldots, \alpha_K)$ and

$$\theta = \begin{pmatrix} \theta_1 \\ \theta_2 \\ \vdots \\ \theta_K \end{pmatrix} = \begin{pmatrix} \theta_{11} & \theta_{12} & \cdots & \theta_{1M} \\ \theta_{21} & \theta_{22} & \cdots & \theta_{2M} \\ \vdots & \vdots & \vdots & \vdots \\ \theta_{K1} & \theta_{K2} & \cdots & \theta_{KM} \end{pmatrix}, \tag{4}$$

where α_i, $i = 1, 2, \ldots, K$, are referred to as mixture weights required to fulfill

$$\alpha_i \geq 0, \quad i = 1, 2, \ldots, K, \qquad \text{and} \qquad \sum_{i=1}^{K} \alpha_i = 1. \tag{5}$$

In the context of this work, the mixture weight α_i specifies the probability that the system is in mode i. As said in Section 2.1, the probability that the system is in a specified operating mode is considered to be unknown. Consequently, α_i, $i = 1, 2, \ldots, K$, are assumed to be unknown and will in the following be considered as nuisance parameters.

Figure 2(a) shows a set of residual samples with underlying distributions described by the pmf (3), with 3 components θ_1, θ_2, and θ_3, and mixture weights $\alpha_1 = \alpha_2 = \alpha_3 = \frac{1}{3}$ shown in Figure 2(b).

In the context of residual evaluation, it is assumed that the distribution of the residual is known in the no-fault case. Let θ^{NF} denote the no-fault distribution parameter, where the i-th row θ_i^{NF} describes the distribution of the residual in operating mode i of the no-fault system. Section 5 describes how the required parameters θ_i^{NF} can be learned from no-fault training data, without the need of any detailed a-priori knowledge of the system. For a different approach, utilizing expert knowledge regarding the system, see [57].

Typically, the distribution of the residual is different for all K operating modes of the no-fault system, which implies that the matrix θ^{NF} has full row rank. For the model (3) to make sense it is required that $M > K$, since otherwise θ^{NF} can be used to describe any residual distribution, including ones originating from faulty cases.

2.3. Residual Evaluation in the Framework of Hypothesis Testing

Consider now a set $\mathcal{R} = \{r_1, r_2, \ldots, r_N\}$ of sampled residual values. Given θ^{NF} and $\mathcal{R}$, the residual evaluation problem is, in the context of this work, to determine if the probability distribution of the residual samples in $\mathcal{R}$ can be characterized by the pmf (3) with $\theta = \theta^{\mathrm{NF}}$ for some $\alpha \in \Upsilon$, where

$$\Upsilon = \left\{ \alpha \in \mathbb{R}^K \; : \; \alpha_i \geq 0, \; \sum_{i=1}^{K} \alpha_i = 1 \right\}, \tag{6}$$

denotes the space of α as specified by (5).

The residual evaluation problem as described above can be formulated by means of the hypotheses

$$\begin{aligned} H_0 : &\quad \theta = \theta^{\mathrm{NF}}, \; \alpha \in \Upsilon \\ H_1 : &\quad \theta \neq \theta^{\mathrm{NF}}, \; \alpha \in \Upsilon \end{aligned} \tag{7}$$

where the null hypothesis H_0 corresponds to the no-fault case, i.e., when no fault is present in the system, and the alternative hypothesis H_1 to the faulty case, i.e., when one or several faults are present in the system. Next section deals with the problem of designing a test statistic for the hypotheses (7).

3. GLR Test Statistic

A standard approach when encountering composite hypotheses, is to utilize the Generalized Likelihood Ratio (GLR), see, e.g., [17, 7]. For testing hypothesis H_0 versus H_1 in (7), the GLR is

$$\Lambda\left(\mathcal{R}\right) = \frac{\max\limits_{\alpha \in \Upsilon} \; \mathcal{L}\left(\alpha, \theta^{\mathrm{NF}} | \mathcal{R}\right)}{\max\limits_{\alpha \in \Upsilon, \; \theta \in \Theta} \; \mathcal{L}\left(\alpha, \theta | \mathcal{R}\right)}, \tag{8}$$

where $\mathcal{L}\left(\theta, \alpha | \mathcal{R}\right) = p(\mathcal{R}|\theta, \alpha)$ is the *likelihood function* of α and θ, given the set $\mathcal{R}$ of residual samples, and

$$\Theta = \left\{ \theta \in \mathbb{R}^{K \times M} \; : \; \theta_{ij} \geq 0, \; \sum_{j=1}^{M} \theta_{ij} = 1 \right\}, \tag{9}$$

denotes the space of the distribution parameter θ as specified by (2).

Let c_j denote how many of the samples in $\mathcal{R}$ that have value x_j, i.e.,

$$c_j = \left|\{r_k \in \mathcal{R} : r_k = x_j,\ x_j \in \mathcal{X}\}\right|, \quad j = 1, 2, \dots, M. \tag{10}$$

By definition, it holds that $\sum_{j=1}^{M} c_j = N$. It is worth noting that the quantities $c_1, c_2, \dots, c_M$ can be obtained from a regular histogram, with M bins, calculated from $\mathcal{R}$.

By assuming that samples from (3) are independent and identically distributed (iid), the log-likelihood can be expressed as

$$\ell\left(\alpha, \theta|\mathcal{R}\right) = \log\left[\mathcal{L}\left(\alpha, \theta|\mathcal{R}\right)\right] = \sum_{j=1}^{M} c_j \log\left[\sum_{i=1}^{K} \alpha_i\, \theta_{ij}\right]. \tag{11}$$

Before proceeding, the following is assumed without loss of generality [55] regarding $c_1, c_2, \dots, c_M$, as specified by (10).

Assumption 1. $c_j > 0, \quad j = 1, 2, \dots, M.$

3.1. Likelihood Maximizations

This section is devoted to explore in detail how to solve the two maximization problems in the GLR (8). Both problems correspond to finding parameter values that maximize the log-likelihood function (11), given the residual samples in $\mathcal{R}$, i.e., finding Maximum Likelihood Estimators (MLE's).

3.1.1. Denominator MLE Problem

Consider first the MLE problem

$$\max_{\alpha\in\Upsilon,\ \theta\in\Theta} \quad \mathcal{L}\left(\alpha, \theta|\mathcal{R}\right), \tag{12}$$

in the denominator of (8). Under the iid assumption and by using the log-likelihood function (11) as well as the structure of the parameter spaces (9) and (6), the MLE problem (12) can be equivalently stated as

$$\begin{aligned} \max_{\alpha\in\mathbb{R}^K,\ \theta\in\mathbb{R}^{K\times M}} \quad & \sum_{j=1}^{M} c_j \log\left[\sum_{i=1}^{K} \alpha_i\, \theta_{ij}\right] \\ \text{subject to} \quad & \alpha_i \geq 0, \quad i = 1, 2, \dots, K, \\ & \theta_{ij} \geq 0, \quad i = 1, 2, \dots, K, j = 1, 2, \dots, M, \\ & \sum_{i=1}^{K} \alpha_i = 1, \\ & \sum_{j=1}^{M} \theta_{ij} = 1, \quad i = 1, 2, \dots, K, \end{aligned} \tag{13}$$

which is a general non-linear constrained maximization problem. It turns out that (13), and equivalently the MLE problem (12), can be solved explicitly. For a formal proof, see [55, Paper C].

Theorem 1. *Let $\mathcal{R}$ be a set of residual samples, define $c_1, c_2, \ldots, c_M$ according to (10), and let Assumption 1 be valid. Then, any $\alpha^\star \in \Upsilon$ and $\theta^\star \in \Theta$ such that*

$$\sum_{i=1}^{K} \alpha_i^\star \, \theta_{ij}^\star = \frac{c_j}{N}, \quad j = 1, 2, \ldots, M, \tag{14}$$

is a solution to the MLE problem (12).

3.1.2. Numerator MLE Problem

Consider now the MLE problem

$$\max_{\alpha \in \Upsilon} \quad \mathcal{L}\left(\alpha, \theta^{\mathrm{NF}} | \mathcal{R}\right), \tag{15}$$

in the numerator of the GLR (8). Note that (15) and (12) differs by that θ is fixed to θ^{NF} in (15).

With the notion of Section 2, the parameter θ^{NF} characterizes the set of distributions of the no-fault residual for all operating modes of the system. In this sense, the MLE problem (15) corresponds to finding a no-fault distribution that is most likely to fit the residual samples in $\mathcal{R}$.

By again using the iid assumption, the log-likelihood function (11), and exploiting the structure of the space (6) of the parameter α, the MLE problem (15) can be equivalently stated as the non-linear constrained maximization problem

$$\begin{aligned} \max_{\alpha \in \mathbb{R}^K} \quad & \sum_{j=1}^{M} c_j \log\left[\sum_{i=1}^{K} \alpha_i \, \theta_{ij}^{\mathrm{NF}}\right] \\ \text{subject to} \quad & \alpha_i \geq 0, \quad i = 1, 2, \ldots, K, \\ & \sum_{i=1}^{K} \alpha_i = 1. \end{aligned} \tag{16}$$

In the general case, it is unfortunately not possible to find an explicit expression for a solution to the maximization problem (16), or equivalently the MLE problem (15), as was the case with the MLE problem (12). There are however several efficient numerical approaches, see, e.g., [46].

By using similar arguments as in the proof of Theorem 1, it can be shown that also (16) is a concave maximization problem. The concavity property facilitates the numerical solving since it implies that if a local maximum can be found, then it is also a global maximum.

4. Online Residual Evaluation Algorithm

Typically, residual evaluation is to be done in an online environment subject to real-time constraints, i.e., computational times in order of micro- or milliseconds with strict deadlines. Unfortunately, it is in general not feasible to solve the non-linear MLE problem (15), or equivalently (16), under such conditions. In this section, a relaxed version of the MLE problem (15) is proposed. The relaxed problem requires less computational effort and results in a residual evaluation test that under certain conditions performs better than the residual evaluation test based on the original MLE problem.

4.1. Relaxed Problem

In light of Theorem 1, and since the problems (16) and (13) exhibit significant similarities, an intuitive solution to problem (16) is to, if possible, choose $\alpha \in \Upsilon$ so that

$$\sum_{i=1}^{K} \alpha_i \, \theta_{ij}^{\mathrm{NF}} = \frac{c_j}{N}, \quad j = 1, 2, \ldots, M. \tag{17}$$

However, since $K < M$, see Section 2.2, (17) corresponds to an overdetermined set of equations which in general has no solution. Motivated by this discussion, it makes sense to chose α so that each $\sum_{i=1}^{K} \alpha_i \, \theta_{ij}^{\mathrm{NF}}$ is as close as possible to $\frac{c_j}{N}$ for $j = 1, 2, \ldots, M$. Thus, the following relaxation of the problem (16) is considered

$$\begin{aligned} \min_{\alpha \in \mathbb{R}^K} \quad & \frac{1}{2} \| \sum_{i=1}^{K} \alpha_i \theta_i^{\mathrm{NF}} - \phi^\star \|_2^2 \\ \text{subject to} \quad & \alpha_i \geq 0, \quad i = 1, 2, \ldots, K, \\ & \sum_{i=1}^{K} \alpha_i = 1, \end{aligned} \tag{18}$$

where $\phi^\star = (\phi_1^\star, \phi_2^\star, \ldots, \phi_M^\star)$ with $\phi_j^\star = \frac{c_j}{N}$ for $j = 1, 2, \ldots, M$.

The relaxed problem (18) is equivalent to a linear least squares problem with equality and non-negative constraints. Solving (18) therefore typically requires less computational effort than solving the original general non-linear maximization problem (16). Solving of (18) will be further discussed in Section 4.3.

In order to compare the fault detection properties of the residual evaluation tests based on the relaxed problem (18) and the original MLE problem (16), the following result is given.

Lemma 1. *Let $c_1, c_2, \ldots, c_M$ fulfill Assumption 1, let $\theta^{NF} \in \Theta$, and*

$$\Phi^{NF} = \left\{ \phi \; : \; \phi = \sum_{i=1}^{K} \alpha_i \; \theta_i^{NF}, \; \forall \alpha \in \Upsilon \right\}. \tag{19}$$

Further, let $\phi^\star \in \Phi^{NF}$, and let α^O and α^R be solutions to the original problem (16) and relaxed problem (18), respectively. Then, it holds that

$$\sum_{i=1}^{K} \alpha_i^O \; \theta_i^{NF} = \sum_{i=1}^{K} \alpha_i^R \; \theta_i^{NF} = \phi^\star. \tag{20}$$

Proof. First note that $\phi^\star \in \Phi^{\mathrm{NF}}$ is equivalent to that the set

$$\Upsilon^{\mathrm{NF}} = \left\{ \alpha \in \Upsilon \ : \ \phi^\star = \sum_{i=1}^{K} \alpha_i \, \theta_i^{\mathrm{NF}} \right\}, \tag{21}$$

is non-empty. Assume that $\Upsilon^{\mathrm{NF}} \neq \emptyset$ and consider first the optimization problem (16). Since $\Upsilon^{\mathrm{NF}} \neq \emptyset$, it follows from [55, Paper C, Lemma 1], and the fact that $\log[\cdot]$ is an increasing function, that any optimal solution to (16) is contained in Υ^{NF}. In particular, this holds for α^{O} and thus $\phi^\star = \sum_{i=1}^{K} \alpha_i^{\mathrm{O}} \, \theta_i^{\mathrm{NF}}$. Consider next the optimization problem (18). Again $\Upsilon^{\mathrm{NF}} \neq \emptyset$ implies that any optimal solution to (18), in particular α^{O}, is contained in Υ^{NF}. Hence, $\phi^\star = \sum_{i=1}^{K} \alpha_i^{\mathrm{R}} \, \theta_i^{\mathrm{NF}}$ and the proof is complete. □

Consider the hypotheses in (7) and the GLR test statistic $\lambda(\mathcal{R}) = -2\log\Lambda(\mathcal{R})$ with $\Lambda(\mathcal{R})$ defined by (8). Define the test statistic

$$\lambda_{\mathrm{R}}(\mathcal{R}) = -2\log \frac{\mathcal{L}\left(\alpha^{\mathrm{R}}, \theta^{\mathrm{NF}} | \mathcal{R}\right)}{\mathcal{L}\left(\alpha^\star, \theta^\star | \mathcal{R}\right)}, \tag{22}$$

where $(\alpha^\star, \theta^\star)$ is a solution to the original MLE problem (12) as present in (8), but where α^{R} is a solution to the relaxed numerator MLE problem (18).

The power of the residual evaluation test $\lambda(\mathcal{R}) > J$ can be quantified by the *power function* [17]

$$\begin{aligned} \beta_\lambda(\alpha, \theta) &= \Pr\left(\text{reject } H_0 | \alpha, \theta\right) \\ &= \Pr\left(\lambda(\mathcal{R}) > J | \alpha, \theta\right), \end{aligned} \tag{23}$$

where J is a fixed threshold. If $\alpha \in \Upsilon$ and $\theta = \theta^{\mathrm{NF}}$ in (23), i.e., under H_0, the power function gives the probability of false detection, or *Type I error*. Otherwise, the power function gives the probability of detection for fixed α and θ, or equivalently the probability of missed detection or *Type II error*, by $1 - \beta_\lambda(\alpha, \theta)$.

Consider now the power function

$$\beta_{\lambda_{\mathrm{R}}}(\alpha, \theta) = \Pr\left(\lambda_{\mathrm{R}}(\mathcal{R}) > J | \alpha, \theta\right), \tag{24}$$

for the residual evaluation test $\lambda_{\mathrm{R}}(\mathcal{R}) > J$, based on the relaxed problem (18). The relation between the power functions (23) and (24) is given by the following result.

Theorem 2. *It holds that*

$$\beta_{\lambda_R}(\alpha, \theta) \geq \beta_\lambda(\alpha, \theta). \tag{25}$$

Proof. It is first noted that according to Theorem 1, it holds that $\phi_j^\star = \frac{c_j}{N}$, $j = 1, 2, \ldots, M$, and thus Lemma 1 is applicable. According to Lemma 1, it holds that $\phi^\star = \sum_{i=1}^{K} \alpha_i^{\mathrm{O}} \, \theta_i^{\mathrm{NF}} = \sum_{i=1}^{K} \alpha_i^{\mathrm{R}} \, \theta_i^{\mathrm{NF}}$ if $\phi^\star \in \Phi^{\mathrm{NF}}$. This implies that

$\mathcal{L}\left(\alpha^{\mathrm{O}},\theta^{\mathrm{NF}}|\mathcal{R}\right)=\mathcal{L}\left(\alpha^{\mathrm{R}},\theta^{\mathrm{NF}}|\mathcal{R}\right)$ if $\phi^\star\in\Phi^{\mathrm{NF}}$. Due to the concavity property of the likelihood function $\mathcal{L}\left(\alpha,\theta|\mathcal{R}\right)$, and the fact that α^{O} is a solution to the MLE problem (15), it follows that

$$\mathcal{L}\left(\alpha^{\mathrm{R}},\theta^{\mathrm{NF}}|\mathcal{R}\right)\leq\mathcal{L}\left(\alpha^{\mathrm{O}},\theta^{\mathrm{NF}}|\mathcal{R}\right),$$

with equality if $\phi^\star\in\Phi^{\mathrm{NF}}$. Thus, it holds that

$$\frac{\mathcal{L}\left(\alpha^{\mathrm{R}},\theta^{\mathrm{NF}}|\mathcal{R}\right)}{\mathcal{L}\left(\alpha^\star,\theta^\star|\mathcal{R}\right)}\leq\frac{\mathcal{L}\left(\alpha^{\mathrm{O}},\theta^{\mathrm{NF}}|\mathcal{R}\right)}{\mathcal{L}\left(\alpha^\star,\theta^\star|\mathcal{R}\right)},\tag{26}$$

and equivalent that $\lambda_{\mathrm{R}}\left(\mathcal{R}\right)\geq\lambda\left(\mathcal{R}\right)$, due to (22) and the definition of $\lambda\left(\mathcal{R}\right)$, again with equality if $\phi^\star\in\Phi^{\mathrm{NF}}$. The claim (25) then follows directly by definitions (23) and (24). □

The implication of Theorem 2 is that the residual evaluation test $\lambda_{\mathrm{R}}\left(\mathcal{R}\right)>J$, based on the relaxed problem (18), gives greater or equal probability for detection than the test $\lambda\left(\mathcal{R}\right)>J$, based on the original problem (16). Or equivalently, that the Type II error, i.e., the probability for missed detection, for the test $\lambda_{\mathrm{R}}\left(\mathcal{R}\right)>J$ always is smaller than, or equal to, the Type II error for the test $\lambda\left(\mathcal{R}\right)>J$.

In general, unfortunately, the test $\lambda_{\mathrm{R}}\left(\mathcal{R}\right)>J$ gives larger probability for false detection, i.e., Type I error, than the test $\lambda\left(\mathcal{R}\right)>J$. This is a direct consequence of Theorem 2. However, asymptotically the condition $\phi^\star\in\Phi^{\mathrm{NF}}$ holds under hypothesis H_0, i.e., in the no-fault case, which implies that also the probabilities for false detection becomes equal for the two tests. This fact is formalized in the following result.

Theorem 3. *Let N denote the number of residual samples in $\mathcal{R}$, and let H_0 in (7) be valid. Then, it holds that*

$$\lim_{N\to\infty}\beta_{\lambda_R}\left(\alpha,\theta\right)-\beta_{\lambda}\left(\alpha,\theta\right)=0.\tag{27}$$

Proof. Define $\phi=\sum_{i=1}^{K}\alpha_i\,\theta_i$ and note that from (7), it can be deduced that $\phi\in\Phi^{\mathrm{NF}}$ is equivalent to that $\alpha\in\Upsilon$ and $\theta=\theta^{\mathrm{NF}}$, i.e., that H_0 in (7) is valid. Thus, by assumption, it holds that $\phi\in\Phi^{\mathrm{NF}}$. Consider now $\phi^\star$ and note that due to the invariance property [17] of maximum likelihood estimates it holds that if $(\alpha^\star,\theta^\star)$ are the MLE of (α,θ), which indeed is true by assumption, then $\phi^\star=\sum_{i=1}^{K}\alpha_i^\star\,\theta_i^\star$ is the MLE of ϕ. Due to the consistenty properties of MLEs [17, Theorem 10.1.6] it can then be shown [55, Paper C, Lemma 5] that

$$\lim_{N\to\infty}\Pr\left(|\phi^\star-\phi|\geq\epsilon\right)=0,$$

for all $\epsilon>0$ and $\phi\in\Phi'$, with

$$\Phi'=\left\{\phi\in\mathbb{R}^M\;:\;\phi_j>0,\quad\sum_{j=1}^{M}\phi_j=1\right\}.\tag{28}$$

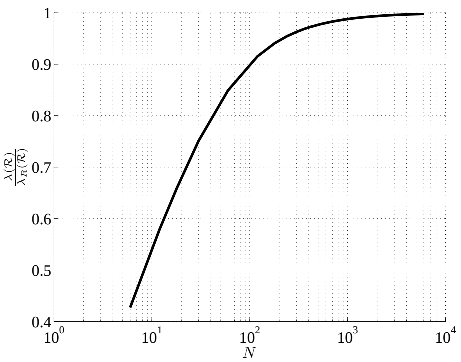


Figure 3: Comparison of test quantities $\lambda_{\mathrm{R}}(\mathcal{R})$ and $\lambda(\mathcal{R})$ under hypothesis H_0, by means of the quantity $\frac{\lambda(\mathcal{R})}{\lambda_{\mathrm{R}}(\mathcal{R})}$, for different values of the size N of the residual sample $\mathcal{R}$.

Since it holds that $\phi \in \Phi^{\mathrm{NF}}$ by assumption, it therefore holds that $\phi^\star \in \Phi^{\mathrm{NF}}$ when $N \to \infty$. Since $\phi^\star \in \Phi^{\mathrm{NF}}$ holds, (26) holds with equality which is equivalent to that $\lambda_{\mathrm{R}}(\mathcal{R}) = \lambda(\mathcal{R})$. By (23) and (24) this is equivalent to $\beta_{\lambda_{\mathrm{R}}}(\alpha, \theta) = \beta_{\lambda}(\alpha, \theta)$, and thus (27) holds. □

Theorem 3 is empirically illustrated in Figure 3, which shows a comparison of the test statistics $\lambda_{\mathrm{R}}(\mathcal{R})$ and $\lambda(\mathcal{R})$, under hypothesis H_0, as the size N of the residual sample $\mathcal{R}$ grows. In this particular case, the parameters $M = 80$ and $K = 25$ was used. The comparison is done by means of the quantity $\frac{\lambda(\mathcal{R})}{\lambda_{\mathrm{R}}(\mathcal{R})}$, and Figure 3 shows the average of 10,000 Monte Carlo simulations using synthetic data. It is clear that the test quantities $\lambda_{\mathrm{R}}(\mathcal{R})$ and $\lambda(\mathcal{R})$ are almost equal when N is large, in this case for $N > 1000$. Since both test $\lambda_{\mathrm{R}}(\mathcal{R}) > J$ and $\lambda(\mathcal{R}) > J$ are based on the same threshold J, the situation in Figure 3 implies that the power functions $\beta_{\lambda_{\mathrm{R}}}(\alpha, \theta)$ and $\beta_{\lambda}(\alpha, \theta)$ are almost identical under H_0 when N is sufficiently large.

To summarize, Theorem 2 implies that the test $\lambda_{\mathrm{R}}(\mathcal{R}) > J$, based on the relaxed problem (18), will result in greater or equal probability for detection than the GLR test $\lambda(\mathcal{R}) > J$, based on the original MLE problem (16). Moreover, according to Theorem 3, if N is sufficiently large, then also the probabilities for false detection will be almost equal for two tests.

In an application where computational effort is crucial, and when implementation matters limit usage of a "sufficiently large" N, a switch from the original MLE problem (16) to the relaxed problem (18), means trading probability of false detection against computational feasibility.

4.2. Residual Evaluation Algorithm

The proposed method for residual evaluation is summarized as an algorithm below. Input to the algorithm is a set of residual samples $\mathcal{R} = \{r_1, r_2, \ldots, r_N\}$, a no-fault residual distribution parameter θ^{NF}, and a detection threshold J.

Output is a decision whether to reject hypothesis H_0 in (7) or not, i.e., whether a fault is present in the system or not.

Step 1: Compute $c_1, c_2, \ldots, c_M$ according to (10).

Step 2: Obtain α^{R} by solving (18).

Step 3: Obtain (22) by computing

$$\lambda_{\mathrm{R}} = -2 \log \frac{\prod_{j=1}^{M} \left(\sum_{i=1}^{K} \alpha_i^{\mathrm{R}} \, \theta_{ij}^{\mathrm{NF}} \right)^{c_j}}{\prod_{j=1}^{M} \left(\frac{c_j}{N} \right)^{c_j}}. \tag{29}$$

Step 4: Reject H_0 if $\lambda_{\mathrm{R}} > J$.

Note that for use with sequential residual data, the samples in $\mathcal{R}$ may be collected by using a sliding window, i.e., at sampling instant t the set of residual samples $\mathcal{R}_t = \{r_{t-N+1}, r_{t-N+2}, \ldots, r_t\}$ is used, where r_t denotes the residual sample collected at instant t.

4.2.1. Parameter Choices

The parameters involved in the residual evaluation are the number N of residual samples in $\mathcal{R}$, the detection threshold J, and the no-fault distribution parameter θ^{NF}. The first two parameters, N and J, are discussed below. The parameter θ^{NF} is the topic of Section 5.

According to Theorem 3, the relaxation (18) of the MLE problem (15) is justified in terms of the probability for false detection if N is sufficiently large. The actual meaning of "sufficiently large" is application dependent and must be evaluated from case to case. This can for example be done by comparing the test quantities $\lambda_{\mathrm{R}}(\mathcal{R})$ and $\lambda(\mathcal{R})$, under hypothesis H_0, for different values of N in the same manner as in Figure 3.

In general, given that N is large enough to justify the relaxation, the choice of N is a trade-off between detection performance and complexity. A large N will give the test statistic smoothed, low-pass, characteristics. This makes it possible to detect small changes in the residual, but on the other hand a large N may increase the detection time. Computational and memory aspects will be discussed in Section 4.3.

The choice of detection threshold J is a trade-off between detection time, and test power, in terms of probability of false detection and probability of missed detection. The higher the threshold, the longer the detection time, the lower the probability of false detections but the higher the probability of missed detection. The actual selection of threshold may be aided by the fact that the test statistic based on the GLR, ideally, is Chi-squared distributed [60].

4.3. Implementation Issues and Computational Complexity

Typically, the residual evaluation algorithm outlined in Section 4.2 is implemented and executed in real-time in an online environment. This poses strict restrictions on the computational complexity of the algorithm, in terms of requirements of computing time and storage.

The main potential computational pitfall of the algorithm is related to Step 2, i.e., solving the equality and inequality constrained linear least square problem (18). A variety of numerical methods have been developed for solving this kind of problems, see, e.g., [29, 35, 12, 64]. Most methods are based on convex optimization [15], where primal-dual methods, including interior point methods [61] and the active set method [12], are of particular interest.

Convex optimization problems can be efficiently solved [15, 61], using for example algorithms with worst-case polynomial complexity [45]. State-of-the-art algorithms often exploits code-generation, where solvers are customized to a specific problem class. One such example is CVXGEN [42], which enables real-time, i.e., solving time scales in microseconds or milliseconds with strict deadlines, solving of modest-sized quadratic optimization problems [41].

The absolute requirements on memory and computation time for solving the linear least square problem (18) by using any of the above methods, depends on the dimension and structure of the $K \times M$ matrix θ^{NF}, where K denotes the number of considered operating modes of the system, and M the number of bins in the above mentioned histogram. The most crucial parameter of these two is K, which in this sense should be kept as low as possible. Implications of the value of this quantity, in the context of residual evaluation performance, is further discussed in Section 5.

It is worth noting that the complexity of the problem (18) does not depend on the number N of residual samples in $\mathcal{R}$. This is favorable since it is only justified to consider the relaxed problem (18) instead of the MLE problem (15) if N is sufficiently large, see Section 4.1.

5. Learning No-Fault Distribution Parameters

In previous sections, it was assumed that the distribution of the residual was known, by means of the parameter θ^{NF}, for K operating modes of the no-fault system. Given a set of residual samples, the problem was to determine if the set of samples originated from the distribution (3) with θ fixed to θ^{NF}. In the context of this section, however, the parameter θ^{NF}, as well as K, are considered to be unknown and the task at hand is to *learn*, i.e., estimate, these using a large set of residual samples, denoted training data.

It is important to stress that the learning is done in an off-line environment with less restrictions on computational complexity, while the actual residual evaluation, as considered in Section 4, typically, is performed online.

5.1. Problem Characterization

With the notion of Section 2, the distribution parameter θ_i^{NF}, i.e., the i-th row of the $K \times M$ matrix which constitutes the parameter θ^{NF}, characterizes the distribution of the no-fault residual when the considered system is in operating mode i. Thus, the value of K determines the number of considered operating modes of the system and θ^{NF} the set of no-fault residual distributions associated with these operating modes.

If the total number of operating modes of the system is known, this knowledge can be exploited and K set accordingly. In general, however, K is unknown and must be learned from the training data. The importance and meaning of the value of K is discussed next.

A large K allows for a complete description of the set of no-fault residual distributions as specified by θ^{NF}, which may be desirable. However, if K is too large, the set of distributions may become too large in the sense that any distribution in the form (3) can be characterized by θ^{NF}. This may reduce the fault detection performance of the residual evaluation test developed in previous sections, since almost any set of residual samples will be considered as generated from a no-fault system, which means no alarm, even if there is a fault present. In addition, a large K results in a θ^{NF} of large dimension, which affects the computational issues addressed in Section 4.3. So in this sense, K should be kept as low as possible. A too small K, on the other hand, may give an insufficient description of the set of all no-fault distributions. This typically also leads to decreased fault detection performance, either in the form of missed detections or false alarms, depending on the strategy used when setting the alarm threshold.

In conclusion, the choice of K and θ^{NF} is a trade-off between fault detection performance and computational effort. However, in order to take the fault detection performance into account, training data from a set of representative fault cases is needed. In the context of this work it is however assumed that only no-fault training data is available due to a number of reasons. First of all, the amount of available no-fault data is typically substantially larger than the available amount of fault data, since faults are rare. To create fault data, one alternative is to inject faults in the real system. This is however considered to be expensive, both in terms of time and money, since it typically require hardware modifications and active usage of the system. Another alternative is to create fault data by simulation. To give realistic results, this on the other hand requires models capable of describing the faulty system, which in turn require detailed knowledge regarding the behavior of the faulty system and possibly also its environment. This kind of information is seldom available for real applications.

Motivated by this discussion, fault detection performance will not be explicitly considered in the learning of K and θ^{NF}. Instead, the learning problem will be formulated as a trade-off between the ability of K and θ^{NF} to characterize the set of all no-fault residual distributions, i.e., model fit, and computational effort. The main motivation for this choice is that a good characterization of the no-fault case will hopefully make it possible to detect deviations from the no-fault case, meaning good fault detection performance. The resulting fault detection performance is however empirically studied in Section 6.

5.2. Problem Formulation

Consider a set $\mathcal{D} = (r_1, r_2, \ldots, r_{N_\mathcal{D}})$ of $N_\mathcal{D}$ residual samples ordered according to time. The residual samples in $\mathcal{D}$ will now be split into residual sample sets

$$\mathcal{R}_k = \left\{ r_{(k-1)n+1}, r_{(k-1)n+2}, \ldots, r_{k-n} \right\}, \tag{30}$$

containing n consecutive residual samples from $\mathcal{D}$. To this end, let $n < N_\mathcal{D}$, and define

$$\mathcal{T} = \left(\mathcal{R}_1, \mathcal{R}_2, \ldots, \mathcal{R}_{N_\mathcal{T}} \right), \tag{31}$$

where $N_\mathcal{T} = \left\lfloor \frac{N_\mathcal{D}}{n} \right\rfloor$, and $\mathcal{R}_k$ is given by (30) for $k = 1, 2, \ldots, N_\mathcal{T}$. The collection $\mathcal{T}$ of residual sample sets $\mathcal{R}_k$, will henceforth be referred to as the training data.

In the following, it is assumed that each $\mathcal{R}_k \in \mathcal{T}$ contains residual samples from only one operating mode. In practice, this can be achieved by choosing n such that the time it takes to collect a set of n residual samples is shorter than the time the system spends in one operating mode, as well as longer than the transition time between any two operating modes.

5.2.1. Formalization of Learning Problem

Let $V(\mathcal{T}, \theta)$ denote a metric that quantifies model fit, i.e., how well the set of distributions characterized by a given parameter θ is able to describe a data set $\mathcal{T}$ in the form (31). A general approach for enabling a trade-off between goodness of model fit and model complexity when identifying parameters in a model is to combine the model fit metric, in the present case $V(\mathcal{T}, \theta)$, with some metric that reflects the model complexity [37, 54].

In the context of this work, required computational effort rather than model complexity is of direct interest. As said in Section 4.2.1, the required computational effort for the residual evaluation algorithm presented in Section 4.2 is strongly dependent on the dimension $K \times M$ of θ^{NF}, and in particular the value of K. Since the larger the value of K, the higher the computational requirements, a function $C(K)$ that increases with K is suitable for quantification of the computational effort. Typically, the actual choice of $C(K)$ is implementation dependent. In general, there are many options, see, e.g., [37, 54]. One alternative is to exploit the information criteria due to Akaike [2].

Given $V(\mathcal{T}, \theta)$ and $C(K)$, the learning problem as stated in Section 5.1 can be formulated as the problem

$$\left(K^\star, \theta^{\mathrm{NF}} \right) = \arg \max_{K,\, \theta \in \Theta^{(K)}} \left(V(\theta, \mathcal{T}) - C(K) \right), \tag{32}$$

where the notation $\Theta^{(K)}$ for the space defined in (9) is introduced to stress the dependency between the space and K. The topic of the remaining of this section is to derive a suitable metric $V(\mathcal{T}, \theta)$ for quantification of model fit.

5.2.2. Quantification of Model Fit

To be able to exploit the developments in previous sections, a likelihood-based framework is adopted for quantification of model fit, and an expression for the (log)-likelihood $\ell\left(\theta|\mathcal{T}\right)$ is sought.

To this end, consider (31) and let c_{kj} denote the total number of residual samples in $\mathcal{R}_k$ that takes value x_j, c.f. (10). Similiar to (11), the log-likelihood of θ and α_k, $k = 1, 2, \ldots, N_{\mathcal{T}}$, given $\mathcal{T}$, can then be written as

$$\begin{aligned}\ell\left(\theta, \alpha_1, \alpha_2, \ldots, \alpha_{N_{\mathcal{T}}}|\mathcal{T}\right) &= \log p\left(\mathcal{T}|\theta, \alpha_1, \alpha_2, \ldots, \alpha_{N_{\mathcal{T}}}\right) \\ &= \sum_{k=1}^{N_{\mathcal{T}}}\sum_{j=1}^{M} c_{kj}\log\left[\sum_{i=1}^{K}\alpha_{ki}\theta_{ij}\right].\end{aligned} \tag{33}$$

The likelihood function $\ell\left(\theta, \alpha_1, \alpha_2, \ldots, \alpha_{N_{\mathcal{T}}}|\mathcal{T}\right)$ in (33) contains both the parameter of interest θ, and the nuisance parameters α_k, $k = 1, 2, \ldots, N_{\mathcal{T}}$. Thus, the nuisance parameters α_k must be *eliminated* from (33). There are mainly two standard approaches [8] for doing this. The first approach is to fix a prior probability distribution for the nuisance parameters, compute the posterior, and then integrate out the nuisance parameter from the posterior to arrive at the posterior marginal distribution of the parameter of interest, see for example [9]. The second approach is to replace the nuisance parameters in the original likelihood function with their conditional maximum likelihood estimates. The resulting function, which not indeed is a pure likelihood function anymore, is referred to as a *profile likelihood* or *maximized likelihood*, see, e.g., [49, 44].

In the context of this section, the mixture weight α_{ki} specifies the probability that the samples in $\mathcal{R}_k$ were collected when operating mode i was present. As said Section 2.1, this probability, and all other probabilities related to the nuisance parameters α_k are assumed to be unknown, which complicates the usage of the first approach mentioned above.

Motivated by this discussion, the second approach is adopted for elimination of α_k, $k = 1, 2, \ldots, N_{\mathcal{T}}$, from (33). The resulting profile likelihood of θ, given $\mathcal{T}$, takes the form

$$\begin{aligned}\hat{\ell}\left(\theta|\mathcal{T}\right) &= \max_{\alpha_1, \alpha_2, \ldots, \alpha_{N_{\mathcal{T}}} \in \Upsilon} \ell\left(\theta, \alpha_1, \alpha_2, \ldots, \alpha_{N_{\mathcal{T}}}|\mathcal{T}\right) \\ &= \max_{\alpha_1, \alpha_2, \ldots, \alpha_{N_{\mathcal{T}}} \in \Upsilon} \sum_{k=1}^{N_{\mathcal{T}}}\sum_{j=1}^{M} c_{kj}\log\left[\sum_{i=1}^{K}\alpha_{ki}\theta_{ij}\right] \\ &= \sum_{k=1}^{N_{\mathcal{T}}}\max_{\alpha_k \in \Upsilon}\sum_{j=1}^{M} c_{kj}\log\left[\sum_{i=1}^{K}\alpha_{ki}\theta_{ij}\right]\end{aligned} \tag{34}$$

Under the assumption that each $\mathcal{R}_k \in \mathcal{T}$ contains residual samples from only one operating mode, it holds that each α_k, $k = 1, 2, \ldots, N_{\mathcal{T}}$, contains one and only one non-zero element, equal to one. In this case,

$$\max_{\alpha_k \in \Upsilon}\sum_{j=1}^{M} c_{kj}\log\left[\sum_{i=1}^{K}\alpha_{ki}\theta_{ij}\right] = \max_{i \in \{1, 2, \ldots, K\}}\sum_{j=1}^{M} c_{kj}\log\theta_{ij},$$

and thus the (profile) likelihood (34) of θ, given $\mathcal{T}$, can be written as

$$\hat{\ell}\left(\theta|\mathcal{T}\right) = \sum_{k=1}^{N_{\mathcal{T}}} \max_{i\in\{1,2,\ldots,K\}} \sum_{j=1}^{M} c_{kj}\log\theta_{ij}. \tag{35}$$

Motivated by these developments, the metric

$$V\left(\mathcal{T},\theta\right) = \hat{\ell}\left(\theta|\mathcal{T}\right) = \sum_{k=1}^{N_{\mathcal{T}}} \max_{i\in\{1,2,\ldots,K\}} \sum_{j=1}^{M} c_{kj}\log\theta_{ij}, \tag{36}$$

will be used to quantify how well the set of distributions characterized by a given parameter θ is able to describe a data set $\mathcal{T}$.

5.3. Learning Algorithm

Consider now the learning problem as formulated in (32). According to Section 2.2 and the fact that it is required that $K < M$, the feasible set of $K^{\star}$ is bounded. Moreover, the quantity $C\left(K\right)$ is not dependent on θ. Thus, given that the problem $\max_{\theta\in\Theta^{(K)}} V\left(\mathcal{T},\theta\right)$ can be solved for a given K, the learning problem (32) can be solved by an exhaustive search over the feasible set of $K^{\star}$.

The key step when searching for $K^{\star}$ and θ^{NF} that solve (32), is therefore to find, for a given K, a $\theta^{(K)}$ that satisfies

$$\theta^{(K)} = \arg\max_{\theta\in\Theta^{(K)}} V\left(\mathcal{T},\theta\right). \tag{37}$$

This is the topic of the remainder of this section.

5.3.1. Method Outline

The basic idea of the proposed approach for finding $\theta^{(K)}$ is to first calculate a distribution parameter $\theta_k \in \Theta^{(1)}$ for each $\mathcal{R}_k \in \mathcal{T}$ by exploiting Theorem 1 and form the set

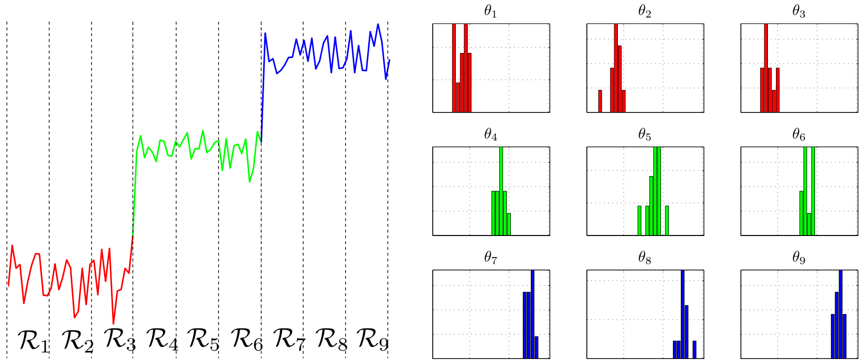
$$\Psi = \left(\theta_1, \theta_2, \ldots, \theta_{N_{\mathcal{T}}}\right), \tag{38}$$

where

$$\theta_k = \arg\max_{\theta\in\Theta^{(1)}} \ell\left(\theta|\mathcal{R}_k\right), \tag{39}$$

for $k = 1, 2, \ldots, N_{\mathcal{T}}$. Then group the distribution parameters in Ψ into K clusters $\mathrm{P}_1, \mathrm{P}_2, \ldots, \mathrm{P}_K$ according to their similarity, and finally calculate the distribution parameter $\theta_i^{\star}$, which constitute the i-th row of $\theta^{(K)}$, from the distribution parameters in cluster P_i.

For an illustration of the approach, consider the residual sample sets $\mathcal{T} = (\mathcal{R}_1, \mathcal{R}_2, \ldots \mathcal{R}_9)$ defined according to Figure 4(a). Note that the sets $\mathcal{R}_k$ in Figure 4(a) have been generated in an ideal way for the purpose of illustration. The set of corresponding distribution parameters $\Psi = (\theta_1, \theta_2, \ldots \theta_9)$ is illustrated in Figure 4(b), and the sought clusters are $\mathrm{P}_1 = \{\theta_1, \theta_2, \theta_3\}$, $\mathrm{P}_2 = \{\theta_4, \theta_5, \theta_6\}$, and $\mathrm{P}_3 = \{\theta_7, \theta_8, \theta_9\}$. The resulting distribution parameters $\theta_1^{\star}$, $\theta_2^{\star}$, and $\theta_3^{\star}$, calculated as the mean of the parameters in the clusters P_1, P_2, and P_3, respectively, are shown in Figure 5. Note the similarity between Figure 5 and Figure 2(b), where the latter in fact shows the true distribution parameters.



(a) Residual sample sets in $\mathcal{T}$

(b) Distribution parameters in Ψ

Figure 4: Illustration of the proposed learning algorithm. Figure 4(a) shows the residual sample sets in $\mathcal{T}$ and Figure 4(b) the corresponding distribution parameters in Ψ.

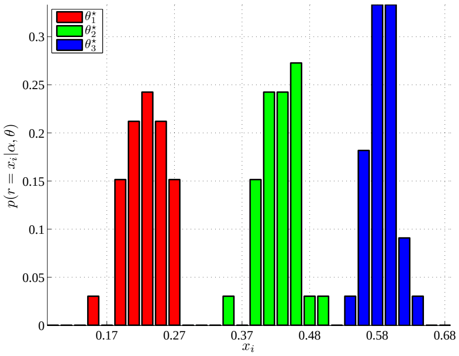


Figure 5: The distribution parameters learned from the training data set shown in Figure 4(a).

5.3.2. Algorithm

The general algorithm for finding a solution to (37) is given below. The input to the algorithm is a set of residual samples $\mathcal{D}$ and constants n and K. The output is a distribution parameter $\theta^{(K)}$.

In the algorithm, $D\left(p\left(r|\theta_k\right) \| p\left(r|\theta_i^\star\right)\right)$ denotes the Kullback-Leibler (KL) divergence [34] between the probability distributions characterized by $p\left(r|\theta_k\right)$ and $p\left(r|\theta_i^\star\right)$. The KL-divergence is one way to quantify the similarity of probability distributions and is properly defined in Section 5.4.

Step 1: Let $\mathcal{T}$ be defined by (31).

Step 2: Let Ψ be defined by (38).

Step 3: Partition Ψ into $\mathrm{P}^\star = (\mathrm{P}_1, \mathrm{P}_2, \ldots, \mathrm{P}_K)$ such that

$$\mathrm{P}^\star = \arg\min_{\mathrm{P}} \sum_{i=1}^{K} \sum_{\theta_k \in \mathrm{P}_i} D\left(p\left(r|\theta_k\right) \| p\left(r|\theta_i^\star\right)\right), \tag{40}$$

where

$$\theta_i^\star = \frac{1}{|\mathrm{P}_i|} \sum_{\theta_k \in \mathrm{P}_i} \theta_k, \quad i = 1, 2, \ldots, K. \tag{41}$$

Step 4: Let

$$\theta^{(K)} = \begin{pmatrix} \theta_1^\star \\ \theta_2^\star \\ \vdots \\ \theta_K^\star \end{pmatrix} = \begin{pmatrix} \theta_{11}^\star & \theta_{12}^\star & \cdots & \theta_{1M}^\star \\ \theta_{21}^\star & \theta_{22}^\star & \cdots & \theta_{2M}^\star \\ \vdots & \vdots & \vdots & \vdots \\ \theta_{K1}^\star & \theta_{K2}^\star & \cdots & \theta_{KM}^\star \end{pmatrix}. \tag{42}$$

The most crucial part of the above algorithm is Step 3, in which a particular partition of the set Ψ should be computed. This problem in fact corresponds to a hard K-means clustering problem [11], for which efficient heuristic methods exists [38]. Implementation issues are discussed in Section 5.5.

The justification of the algorithm, in terms of its ability to provide a solution to the problem (37), is given in next section.

5.4. Justification of Learning Algorithm

This section contains technical developments necessary for proving that the algorithm defined by Steps 1-4 in Section 5.3.2 indeed gives a solution to the problem (37) as output. This is done in the following manner. First, a sufficient condition for a solution to the problem (37) is given. The condition is given in terms of properties of a partition of the set $\mathcal{T}$, computed in Step 1 of the algorithm. Next, the sufficient condition is transformed into a condition on a partition of the set Ψ, defined in Step 2. Finally, it is verified that the partition of Ψ computed by means of K-means clustering in Step 3 satisfies this condition.

A sufficient condition for a solution to the problem (37) is given below.

Theorem 4. *Let $\mathcal{D}$ be a set of $N_{\mathcal{D}}$ iid residual samples, let $n < N_{\mathcal{D}}$, and let $\mathcal{T}$ be defined by (31). For a given positive integer K, if $\mathrm{T} = (\mathrm{T}_1, \mathrm{T}_2, \ldots, \mathrm{T}_K)$ is a partition of $\mathcal{T}$ such that for each block $\mathrm{T}_i \in \mathrm{T}$ and for each element $\mathcal{R}_k \in \mathrm{T}_i$, it holds that*

$$\ell\left(\theta_i^\star|\mathcal{R}_k\right) \geq \ell\left(\theta_p^\star|\mathcal{R}_k\right), \quad p = 1, 2, \ldots, K, \tag{43}$$

where

$$\theta_i^\star = \arg\max_{\theta \in \Theta^{(1)}} \sum_{\mathcal{R}_k \in \mathrm{T}_i} \ell\left(\theta|\mathcal{R}_k\right), \quad i = 1, 2, \ldots, K, \tag{44}$$

then

$$V\left(\mathcal{T}, \theta^{(K)}\right) = \max_{\theta \in \Theta^{(K)}} V\left(\mathcal{T}, \theta\right), \tag{45}$$

with $V(\mathcal{T}, \theta)$ and $\theta^{(K)}$ defined by (36) and (42), respectively.

Proof. From (11), and the fact that $\theta \in \Theta^{(1)}$ due to (44), which implies that $K = 1$ and $\alpha_1 = 1$ in (11), it holds that

$$\ell\left(\theta|\mathcal{R}_k\right) = \sum_{j=1}^{M} c_{kj} \log \theta_j, \tag{46}$$

where c_{kj}, $j = 1, 2, \ldots, M$, denotes the total number of samples in $\mathcal{R}_k$ that takes value x_j. Given is that $\mathrm{T} = (\mathrm{T}_1, \mathrm{T}_2, \ldots, \mathrm{T}_K)$ is a partition of $\mathcal{T}$, such that (43) is satisfied for each block $\mathrm{T}_i \in \mathrm{T}$ and for each element $\mathcal{R}_k \in \mathrm{T}_i$, with $\theta_i^\star$, $i = 1, 2, \ldots, K$, defined according to (44). From (43) and (46) it follows that for each $\mathrm{T}_i \in \mathrm{T}$ and for each $\mathcal{R}_k \in \mathrm{T}_i$, it holds that

$$\sum_{j=1}^{M} c_{kj} \log \theta_{ij}^\star \geq \sum_{j=1}^{M} c_{kj} \log \theta_{pj}^\star, \tag{47}$$

for $p = 1, 2, \ldots, K$. Due to (47) it holds that for each $\mathrm{T}_i \in \mathrm{T}$ and for each $\mathcal{R}_k \in \mathrm{T}_i$

$$\max_{p \in \{1,2,\ldots,K\}} \sum_{j=1}^{M} c_{kj} \log \theta_{pj}^\star = \sum_{j=1}^{M} c_{kj} \log \theta_{ij}^\star. \tag{48}$$

Due to (48) and the fact that $\mathrm{T} = (\mathrm{T}_1, \mathrm{T}_2, \ldots, \mathrm{T}_K)$ is a partition of $\mathcal{T} = (\mathcal{R}_1, \mathcal{R}_2, \ldots, \mathcal{R}_{N_{\mathcal{T}}})$, it holds that

$$\begin{aligned} V\left(\mathcal{T}, \theta^\star\right) &= \sum_{k=1}^{N_{\mathcal{T}}} \max_{p \in \{1,2,\ldots,K\}} \sum_{j=1}^{M} c_{kj} \log \theta_{pj}^\star \\ &= \sum_{i=1}^{K} \sum_{\mathcal{R}_k \in \mathrm{T}_i} \max_{p \in \{1,2,\ldots,K\}} \sum_{j=1}^{M} c_{kj} \log \theta_{pj}^\star \\ &= \sum_{i=1}^{K} \sum_{\mathcal{R}_k \in \mathrm{T}_i} \sum_{j=1}^{M} c_{kj} \log \theta_{ij}^\star \end{aligned} \tag{49}$$

By definition (44), it holds that $\theta_i^\star \in \Theta^{(1)}$, $i = 1, 2, \ldots, K$, and therefore that $\theta^{(K)} \in \Theta^{(K)}$ with $\theta^{(K)}$ defined by (45). To show that (45) is satisfied, it is sufficient to show that $V(\mathcal{T}, \theta^\star)$ is a maximum value. Since $\theta_i^\star = (\theta_{i1}^\star, \theta_{i2}^\star, \ldots, \theta_{iM}^\star)$ only is present in the term

$$\sum_{\mathcal{R}_k \in \mathrm{T}_i} \sum_{j=1}^{M} c_{kj} \log \theta_{ij}^\star, \tag{50}$$

in (49), it follows that $V(\mathcal{T}, \theta^\star)$ as given by (49) is a maximum if (50) is a maximum, for each $i = 1, 2, \ldots, M$. It is now noted that, due to (46), (44) is equivalent to

$$\theta_i^\star = \arg \max_{\theta \in \Theta^{(1)}} \sum_{\mathcal{R}_k \in \mathrm{T}_i} \sum_{j=1}^{M} c_{kj} \log \theta_j, \quad i = 1, 2, \ldots, K,$$

which completes the proof. □

The implication of Theorem 4, is that the solving of (37) can be reduced to finding a partition $\mathrm{T} = (\mathrm{T}_1, \mathrm{T}_2, \ldots, \mathrm{T}_K)$ of the set $\mathcal{T}$, defined according to (31), that fulfills (43). Next result, establishes a relation between the sought partition T of $\mathcal{T}$ and a partition P of the set Ψ computed in Step 2 of the algorithm.

To this end, KL-divergence needs to be properly defined. In general, for two distributions of a discrete random variable R with range $\mathcal{X}$ that are characterized by the pmf's $f_1(r)$ and $f_2(r)$, the KL-divergence between $f_1(r)$ and $f_2(r)$ is defined as

$$D\left(f_1(r) \| f_2(r)\right) = \sum_{x_k \in \mathcal{X}} f_1(x_k) \log \frac{f_1(x_k)}{f_2(x_k)}. \tag{51}$$

It follows that $D\left(f_1(r) \| f_2(r)\right) \geq 0$, with equality if and only if $f_1(r) \equiv f_2(r)$.

A transformation of the sufficient condition in Theorem 4 on a partition T of $\mathcal{T}$ to a partition P of the set Ψ is given by the following lemma. Proof of the result can be found in [55, Paper C, Lemma 3].

Lemma 2. *Let* $\mathrm{P}_i \subseteq \Psi$, *let*

$$\mathrm{T}_i = \{\mathcal{R}_k \in \mathcal{T} \ : \ \theta_k \in \mathrm{P}_i\} \tag{52}$$

and let all residual samples in all $\mathcal{R}_k \in \mathrm{T}_i$ *be iid. Then, for any* $\theta_p, \theta_q \in \Theta^{(1)}$ *and for each* $\mathcal{R}_k \in \mathrm{T}_i$ *it holds that*

$$\ell\left(\theta_p | \mathcal{R}_k\right) \geq \ell\left(\theta_q | \mathcal{R}_k\right), \tag{53}$$

if and only if for each $\theta_k \in \mathrm{P}_i$ *it holds that*

$$D\left(p\left(r|\theta_k\right) || p\left(r|\theta_q\right)\right) \leq D\left(p\left(r|\theta_k\right) || p\left(r|\theta_q\right)\right). \tag{54}$$

Moreover, it holds that

$$\arg \max_{\theta \in \Theta^{(1)}} \sum_{\mathcal{R}_k \in \mathrm{T}_i} \ell\left(\theta | \mathcal{R}_k\right) = \arg \min_{\theta \in \Theta^{(1)}} \sum_{\theta_k \in \mathrm{P}_i} D\left(p\left(r|\theta_k\right) || p\left(r|\theta\right)\right). \tag{55}$$

The problem of finding a partition T of $\mathcal{T}$ fulfilling the sufficient condition in Theorem 4 can with aid of Lemma 2 be equivalently stated as the problem of finding a partition P of Ψ fulfilling the condition (54). Next result verifies that a partition of Ψ computed in Step 3 of the algorithm indeed satisfies (54). Proof of the result can be found in [55, Paper C, Lemma 4].

Lemma 3. *Let $\mathcal{D}$ be a set of $N_{\mathcal{D}}$ residual samples fulfilling Assumption 1, let $n < N_{\mathcal{D}}$, let $\mathcal{T}$ be defined by (31), let Ψ be defined by (38), and let K be a positive integer. Further, let $\mathrm{P}^\star = (\mathrm{P}_1, \mathrm{P}_2, \ldots, \mathrm{P}_K)$ be a partition of Ψ such that (40) holds and $\theta_i^\star$, $i = 1, 2, \ldots, K$, satisfies (41). Then, it holds that*

$$\theta_i^\star = \arg \min_{\theta \in \Theta^{(1)}} \sum_{\theta_k \in \mathrm{P}_i} D\left(p\left(r|\theta_k\right) || p\left(r|\theta\right)\right), \tag{56}$$

for $i = 1, 2, \ldots, K$. Moreover, for each block $\mathrm{P}_i \in \mathrm{P}^\star$ and for each element $\theta_k \in \mathrm{P}_i$ it holds that

$$D\left(p\left(r|\theta_k\right) || p\left(r|\theta_i^\star\right)\right) \leq D\left(p\left(r|\theta_k\right) || p\left(r|\theta_j^\star\right)\right), \tag{57}$$

for $j = 1, 2, \ldots, K$.

With help of Theorem 4, Lemma 2, and Lemma 3, it can be proved that the output from the algorithm in Section 5.3.2 indeed is a solution to the problem (37).

Theorem 5. *Let $\mathcal{D}$ be a set of $N_{\mathcal{D}}$ iid residual samples, let $n < N_{\mathcal{D}}$, and let K be a positive integer. Further, let $\mathcal{D}$, n, and K, be input to the algorithm defined by Steps 1-4 in Section 5.3.2 and let $\theta^{(K)}$ be the output. Then, $\theta^{(K)}$ is a solution to (37).*

5.5. Implementation Issues

As said in Section 5.3.2, the most crucial part of the learning algorithm is Step 3, i.e., to find a partition P of Ψ by means of hard K-means clustering [11].

The complexity properties of the general K-means clustering problem depends on which similarity measure, in the present case the KL-divergence, that is used in (40). For instance, the problem is NP-hard [5] when the (squared) Euclidean distance is used, but can be solved in a polynomial time if a variance-based measure is used [32].

There are however a variety of heuristic algorithms available for solving the general clustering problem approximately. One widely used [10] and in practice often successful alternative, is the local search based K-Means algorithm [39, 38], which also is referred to as Lloyd's algorithm. For the particular, and present, case when the KL-divergence is used as similarity measure, an approximate solution to the clustering problem can be computed with the K-means algorithm in polynomial time [40]. For a general treatment of clustering problems with similarity measures based on Bregman divergences, including the KL-divergence, see [6].

As a remark, it is noted that the assignment and update steps in the K-means algorithm in fact [11] correspond to the Expectation and Maximization steps, respectively, in the EM-algorithm [21]. Thus, when the K-means algorithm is employed for solving the clustering problem in Step 3, the learning algorithm in a sense resembles the EM-algorithm.

It is also noted that in a practical implementation of the learning algorithm, the training data set $\mathcal{T}$ is preferably split into an estimation data set $\mathcal{E}$ and a validation data set $\mathcal{V}$, in order to avoid over-fitting, see, e.g., [37]. In this setting, the estimation data set $\mathcal{E}$ is used when solving (37) to obtain $\theta^{(K)}$, for a fixed K, and then the validation data set $\mathcal{V}$ is used to evaluate if the obtained solution $\theta^{(K)}$ and K satisfies (32).

5.5.1. Parameter Choices

The only parameter involved in the learning problem (32) is n, the number of residual samples used in each $\mathcal{R}_k$ when calculating the set $\mathcal{T}$ according to (31), which is done in Step 1 of the algorithm.

The choice of n is determined by the properties of the considered system. As said in Section 5.2, n should be chosen so that each $\mathcal{R}_k \in \mathcal{T}$ contains residual samples from only one operating mode of the system. In order to achieve this, n should be chosen so that the time it takes to collect a set of n residual samples is less than the average time that the system spends in one operating mode.

Before learning the parameter θ^{NF}, the quantization M of the residual, i.e., the size of the residual range space and thereby the resolution of the residual distribution (1), must be determined and the training data in $\mathcal{D}$ formated accordingly. Choosing M, in fact, corresponds to the well-studied, but nevertheless difficult, problem of choosing the number of bins in a regular histogram given a sample of data. Numerous approaches for solving this problem exist, see for example [20] and references therein.

Regardless of the method used to solve the problem, the choice of M is a trade-off between accuracy and computational complexity, in terms of time and storage. A larger M results in a more accurate discretization of the residual and higher resolution of the probability distributions. On the other hand, a large M requires more memory and involves more computations. The choice of M is also related to the choice of n and N, since a small n, or N, together with a large M will result in an inadequate estimation of the distribution, i.e., a sparse histogram.

The resolution of the residual also affects the fault detection performance in the sense that if the resolution is high, small deviations of the residual can be perceived and thereby small faults can be detected. As a guideline, the resolution of the residual can be matched to the size of the smallest fault that should be possible to detect.

6. Application Example

The proposed residual evaluation approach has been applied to the problem of fault detection in the gas-flow system of a Scania 6 cylinder, 13 liter, truck diesel engine equipped with Exhaust Gas Recirculation (EGR), Variable Geometry Turbine (VGT), and intake throttle. The overall purpose of the study was to evaluate and demonstrate the proposed on-line residual evaluation algorithm, as well as the off-line algorithm for learning no-fault residual distributions, using measurement data. In addition, it is also illustrated how the fault detection performance of the residual evaluation test is influenced by different values of the involved parameters, in particular the size N of the residual sample set $\mathcal{R}$, and the number K of no-fault distribution parameters in θ^{NF}.

6.1. Automotive Gas-Flow Diagnosis

The automotive gas-flow system, or rather the truck diesel engine itself, is a complex system that operates in a variety of different operating modes characterized by for instance ambient pressure and temperature, engine torque, engine speed, etc. Fault diagnosis of the gas-flow system consists of detecting and isolating faults in sensors that measure pressure, temperature, and mass-flow, actuators that control the EGR, VGT and intake throttle, as well as faults related to, e.g., manifold leakages and clogged air filters. The main incentives for gas-flow diagnosis are fault management by means of fault tolerant control, On-Board Diagnosis (OBD) regulations, and repair and maintenance.

The model of the gas-flow system, which is described in [58], relies on both fundamental first principle physics and gray-box modeling. For diagnosis of the gas-flow system, a set of model-based residual generators were designed with the sequential residual generation method described in [56]. Naturally, the model does not describe all aspects of the system, leading to that all residuals exhibit properties similar to those illustrated in Figure 1.

The particular residual considered in this study is sensitive to 10 faults: 3 leakages, 6 sensor faults, and 1 actuator fault. The value of the residual is based on a comparison of two modeled values of the temperature before the cylinders.

6.2. Learning of No-Fault Distribution Parameters

The data set used for the learning contains measurements from parts of a test drive, including both city and high-way driving, from Södertälje to Arvidsjaur in Sweden. The data set contains in total 156,912 measurements sampled at a rate of 0.1 s, which corresponds to more than 4 hours of driving. The measurements in the data set were used as input to the considered residual generator and the residual samples used in the study were computed off-line. In order to minimize the risk of over-fitting the no-fault distribution parameters to the training data, the set of residual samples was divided into an estimation data set, $\mathcal{E}$, and a validation data set, $\mathcal{V}$, of equal size.

6.2.1. Parameter Values

The value of the parameter M, i.e., the quantization of the residual samples, was chosen to be $M = 80$. This makes it theoretically possible to detect faults that cause deviations of the residual of about 3 kelvin. For this application,

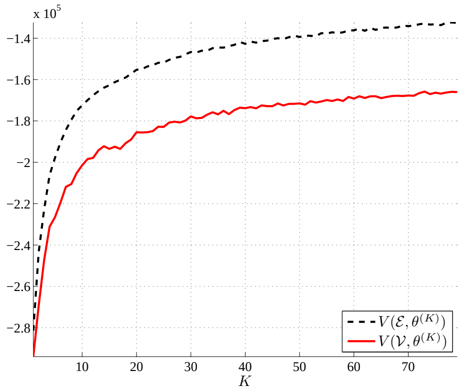


Figure 6: Evaluation of model fit metrics $V\left(\mathcal{E},\theta^{(K)}\right)$ (dashed, black) and $V\left(\mathcal{V},\theta^{(K)}\right)$ (solid, red) for different of values of K.

this is a good trade-off between complexity, in terms of required memory and computational effort, and accuracy.

By a brief analysis of the residual samples, it seems that the minimum time that the gas-flow system spends in one operating mode is approximately 4 s. This can be seen in Figure 1, which in fact shows a subset of the residual samples used in this study. Since the sample rate is 0.1 s, the parameter n, which specifies the number of residual samples in each $\mathcal{R}_k$ in the set $\mathcal{T}$ calculated in Step 1 of the algorithm, should be chosen to satisfy $n < 40$, see Section 5.5.1. Based on this, the parameter was chosen to be $n = 32$.

6.2.2. Results

The algorithm for learning no-fault distribution parameters described in Section 5.3.2, was implemented in Matlab. To solve the involved clustering problem, the K-means algorithm [39, 38] was employed. The algorithm was run with $K \in \{1, 2, \ldots, 79\}$.

Figure 6 shows the model fit metric (36) evaluated for the estimation data set $\mathcal{E}$ and validation data set $\mathcal{V}$, and with the parameters $\theta^{(K)}$, $K \in \{1, 2, \ldots, 79\}$, obtained as output from the algorithm. In Figure 6 it can first of all be seen that the quantitative behaviors of $V\left(\mathcal{E},\theta^{(K)}\right)$ and $V\left(\mathcal{V},\theta^{(K)}\right)$ are similar, but that $V\left(\mathcal{E},\theta^{(K)}\right)$ always is larger than $V\left(\mathcal{V},\theta^{(K)}\right)$. The latter seems natural since the data set $\mathcal{E}$ indeed was used as input to the learning algorithm. Second, it can also be noticed that the improvement in model fit as a function of K is larger for smaller K.

Based on the above observations, and with respect to the trade-off between model fit and required computational effort stated by (32), $K = 10$ was chosen. The 10 no-fault distribution parameters, i.e., the rows of $\theta^{(10)}$, are shown in Figure 7. Note that the characteristics of the learned distribution parameters are quite different, some are multi-modal and some have only one single mode. In addition, the distribution parameters are overlapping.

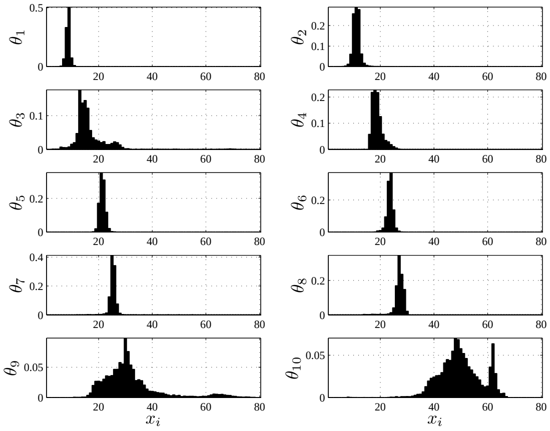


Figure 7: The no-fault distribution parameters contained in $\theta^{\mathrm{NF}} = \theta^{10}$.

6.3. Evaluation Setup

The set of residual samples used in the evaluation is based on the validation data set $\mathcal{V}$, which contains in total 78,456 residual samples. Note that this data set is different than the estimation data set used to learn the no-fault distribution parameters as described above.

6.3.1. Considered Fault

The fault considered in the evaluation is a fault in the boost pressure sensor. The relation between the boost pressure sensor signal $y_{p_{\mathrm{im}}}$ and the considered residual is dynamic, and the residual value r depends on the derivative of the boost pressure sensor signal, as well as the actual sensor signal, i.e., $r = F(y_{p_{\mathrm{im}}}, \dot{y}_{p_{\mathrm{im}}}, \ldots)$, where $F(\cdot)$ is a non-linear function. The considered fault scenario is a gain fault in the boost pressure sensor, that is, the sensor signal $y_{p_{\mathrm{im}}}$ fed to the residual generator is $y_{p_{\mathrm{im}}} = \delta \cdot p_{\mathrm{im}}$, where p_{im} is the actual boost pressure, and $\delta \neq 1$ indicates a gain fault. Gain faults in the range $\delta \in [0.2, 1.8]$ were implemented off-line by modification of the sensor signal.

6.3.2. Fault Detection Performance Metrics

The main metric considered in the evaluation is the power function, in this context defined as

$$\beta_{\lambda_{\mathrm{R}}}(\delta) = \Pr\left(\text{detection}|\delta\right) = \Pr(\lambda_{\mathrm{R}}\left(\mathcal{R}\right) > J|\delta), \tag{58}$$

for the test $\lambda_{\mathrm{R}}\left(\mathcal{R}\right) > J$, defined in Section 4. Note that $\delta = 1$ in the power function (58) corresponds to that $\alpha \in \Upsilon$ and $\theta = \theta^{\mathrm{NF}}$ in the power function (24).

To study another important aspect of the detection performance, the *Mean Time to Detection* (MTD) will also be considered. Note that the choices of the values of the parameters N and J, i.e., the size of the residual sample set $\mathcal{R}$

and the detection threshold, respectively, are a trade-off between the metrics measured by the power function and the MTD, see Section 4.2.1.

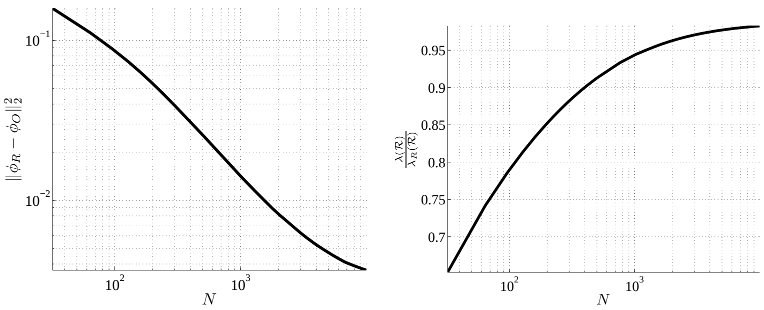
In order to be able to say something about the relative performance of the proposed residual evaluation approach, it will be compared to the often in practice used norm-based residual evaluation approach built upon the test statistic $s(\mathcal{R}) = \frac{1}{N}\sum_{r_k \in \bar{\mathcal{R}}} \bar{r}_k^2$ where $\bar{\mathcal{R}} = (\bar{r}_1, \bar{r}_2, \ldots, \bar{r}_N)$ is a low-passed filtered version of the sample $\mathcal{R}$. Note that the purpose of this comparison merely is to give a feeling of the relative performance of the proposed residual evaluation approach, and the comparison is not claimed to be exhaustive. The low-pass filtering was in this study performed with a first-order Butterworth filter and for comparison, four different cut-off frequencies, $f_1 = 0.005$ Hz, $f_2 = 0.05$ Hz, $f_3 = 0.5$ Hz, and $f_4 = 4.5$ Hz, were used. The corresponding test statistics are denoted s_1, s_2, s_3, and s_4. Recall that the residual is sampled at a rate of 0.1 s, corresponding to a frequency of $f_s = 10$ Hz.

6.3.3. Implementation Details

The residual evaluation algorithm described in Section 4.2, was implemented in Matlab. To solve the optimization problem (18), a tailored solver was generated using the software tool CVXGEN [42], see Section 4.3. With this solver, the optimization problem (18) in the setting of this study, could be solved in the time scale of 10^{-4} s. Solving the corresponding problem using the Matlab optimization toolbox results in solving times of the magnitude of 10^{-3} s. Solving the original numerator MLE problem (15) using the Matlab optimization toolbox however renders solving times of magnitude 10^{-1} s.

As said in Section 4, it is only justified, in terms of the probability of false detection, to consider the relaxed problem (18) instead of the original MLE problem (15) if the size N of the set of residual samples $\mathcal{R}$ is sufficiently large. To investigate the meaning of sufficiently large in the context of this study, Figure 8 shows a comparison of the solutions to the respective problems, as well as a comparison of the corresponding test statistics, for different values of N in the no-fault case. Figure 8(a) shows a comparison of the solution α^{R} to the relaxed problem (18) and the solution α^{O} to the original MLE problem (15), by means of the quantity $\|\phi_{\mathrm{R}} - \phi_{\mathrm{O}}\|_2^2$, where $\phi_{\mathrm{R}} = \sum_{i=1}^{K} \alpha_i^{\mathrm{R}} \theta_i^{\mathrm{NF}}$ and $\phi_{\mathrm{O}} = \sum_{i=1}^{K} \alpha_i^{\mathrm{O}} \theta_i^{\mathrm{NF}}$. Figure 8(b) shows a comparison of the test statistics $\lambda_{\mathrm{R}}(\mathcal{R})$, based on the relaxed problem and $\lambda(\mathcal{R})$, based on the original MLE problem, by means of the quantity $\frac{\lambda(\mathcal{R})}{\lambda_{\mathrm{R}}(\mathcal{R})}$. The results shown in Figure 8 are the average of 150,000 runs. Based on Figure 8, it was concluded that in the context of this study, $N > 1000$ is good enough to justify the switch to the relaxed problem. Recall from Section 4.3 that the complexity of the relaxed problem, in terms of computational time and memory, is independent of N.

The threshold J for the test $\lambda_{\mathrm{R}}(\mathcal{R}) > J$, as well as the thresholds for the norm-based tests, was computed based on the estimation data set used in the learning of the no-fault distribution parameters. All thresholds were computed in order to give a probability of false detection of 5 %. All residual sample sets were taken from the validation data set by using a sliding window, see Section 4.2.



(a) Comparison of α^{R} and α^{O}. (b) Comparison of $\lambda_{\mathrm{R}}(\mathcal{R})$ and $\lambda(\mathcal{R})$.

Figure 8: Investigation of how the relation between the solutions α^{R} and α^{O} to the relaxed (18) and original (15) MLE problems, respectively, as well as the corresponding test quantities, $\lambda_{\mathrm{R}}(\mathcal{R})$ and $\lambda(\mathcal{R})$, changes with the size N of the residual sample $\mathcal{R}$.

6.4. Evaluation Results

Figure 9 shows the residual and the test statistics $\lambda_{\mathrm{R}}(\mathcal{R})$ and $s_1(\mathcal{R})$, for size $N = 1024$ of the set of residual samples $\mathcal{R}$, in a test case when an abrupt fault occurs at time $t = 450$ s. The fault is a 10 % gain fault in the boost pressure sensor, which correspond to $\delta = 1.1$. For the test statistic $\lambda_{\mathrm{R}}(\mathcal{R})$, the parameter $\theta^{\mathrm{NF}} = \theta^{(10)}$ illustrated in Figure 7 was used.

It can be noted that, as in Figure 1, the residual in Figure 9 is non-zero in the no-fault case, i.e., for $t < 450$ s, and its distribution exhibit non-stationary features in both the no-fault and fault cases. Further, it can also be seen that the difference between the residual in the no-fault and fault cases are small, but that there is a significant difference between the test statistic $\lambda_{\mathrm{R}}(\mathcal{R})$ in the no-fault and fault cases. Since $\lambda_{\mathrm{R}}(\mathcal{R})$ is above the threshold in the fault case, the present fault can be detected. The fault can however not be detected in a reliable way with the test statistic $s_1(\mathcal{R})$, which in this case performed better than each of the test statistics $s_2(\mathcal{R})$, $s_3(\mathcal{R})$, and $s_4(\mathcal{R})$.

6.4.1. Power as Function of N

To illustrate how the power of the test $\lambda_{\mathrm{R}}(\mathcal{R}) > J$ varies with the number N of residual samples in $\mathcal{R}$, Figure 10 shows the power function for the test for different values of N and parameter $\theta^{\mathrm{NF}} = \theta^{(10)}$. Figure 10 clearly shows that the power of the test increases with N.

In Figure 10, it can be seen that as small faults as $\delta \approx 0.95$ and $\delta \approx 1.05$, corresponding to gain faults in the boost pressure sensor of about ± 5 %, may be possible to detect if N is sufficiently large. To further illustrate this, Figure 11 shows the Receiver Operating Characteristic (ROC) curve for different values of N, for a test case with $\delta = 1.05$. The ROC curve shows the relation between the True Positive Rate (TPR) of detection (y-axis), and the False Positive Rate (FPR) of detection (x-axis), i.e., the relation between correct detections and false detections, when the detection threshold J is varied. Figure 11 again shows that the detection performance increases with N, but also that the rate of false

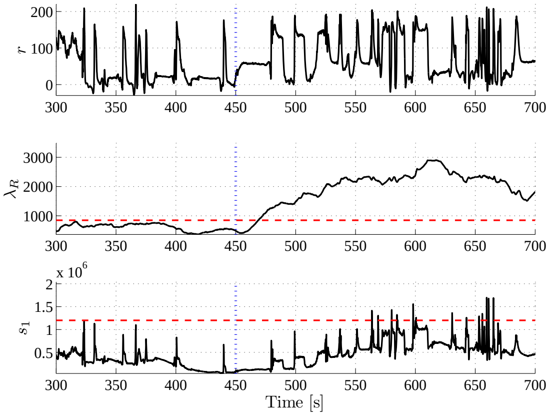


Figure 9: Residual r (top), test statistic $\lambda_{\mathrm{R}}(\mathcal{R})$ (middle), and test statistic $s_1(\mathcal{R})$ (bottom), when an abrupt fault occurs at $t = 450$ s. The fault is a 10 % gain fault in the boost pressure sensor, which corresponds to $\delta = 1.1$.

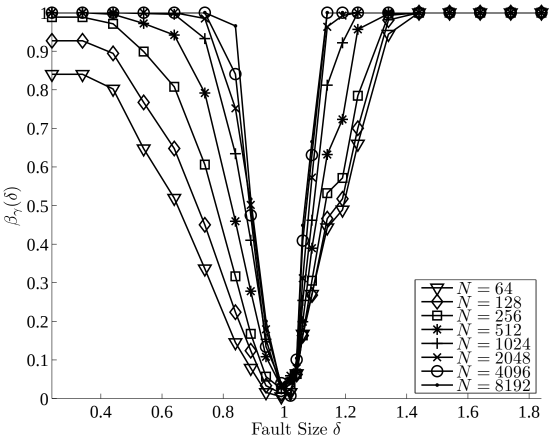


Figure 10: Power function $\beta_{\lambda_{\mathrm{R}}}(\delta)$ for the test $\lambda_{\mathrm{R}}(\mathcal{R}) > J$ for different sizes N of the sample $\mathcal{R}$. The power increases with N.

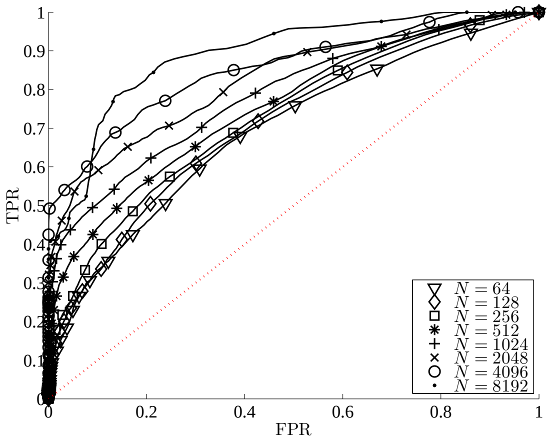


Figure 11: ROC for test $\lambda_{\mathrm{R}}(\mathcal{R}) > J$ when $\delta = 1.05$ for different sizes N of the sample $\mathcal{R}$.

detections can be made lower than the rate of actual detections even for moderate values of N.

6.4.2. Power as Function of K

To analyze how the power of the test $\lambda_{\mathrm{R}}(\mathcal{R}) > J$ varies with different values of the parameter $\theta^{\mathrm{NF}} = \theta^{(K)}$, specifying the set of no-fault residual distributions, or more specifically with K, i.e., the number of operating modes of the system, Figure 12 shows the power function for the test for different values of K. All considered parameters $\theta^{(K)}$ were obtained by means of the algorithm described in Section 5. To also see how the power of the test depends on the relation between K and N, Figure 12 shows how the power function depends on K for different values of N.

The general conclusion from the evaluation shown in Figure 12, is that for a given $256 \leq N \leq 1024$, the power of the test $\lambda_{\mathrm{R}}(\mathcal{R}) > J$ is almost equal for all considered K. For small N, e.g., $N = 64$, however, the power increases with K and for large N, e.g., $N = 4096$, the power increases as K decreases. The liable rationale behind this is that a small K results in a generic and averaged, in terms of operating modes, description of the set of no-fault residual distributions. A large set of residual samples typically means residual samples from a variety of operating modes, while a small set of residual samples on the other hand means residual samples from only a few operating modes. This means that a parameter θ^{NF} corresponding to a small K, typically can describe the distribution of a large set of no-fault residual samples, i.e., a large N, better than the distribution of a small set of no-fault residual samples, i.e., a small N. An accurate description of the no-fault residual distribution makes it possible to distinguish such from a faulty residual distribution, which indeed means good detection power.

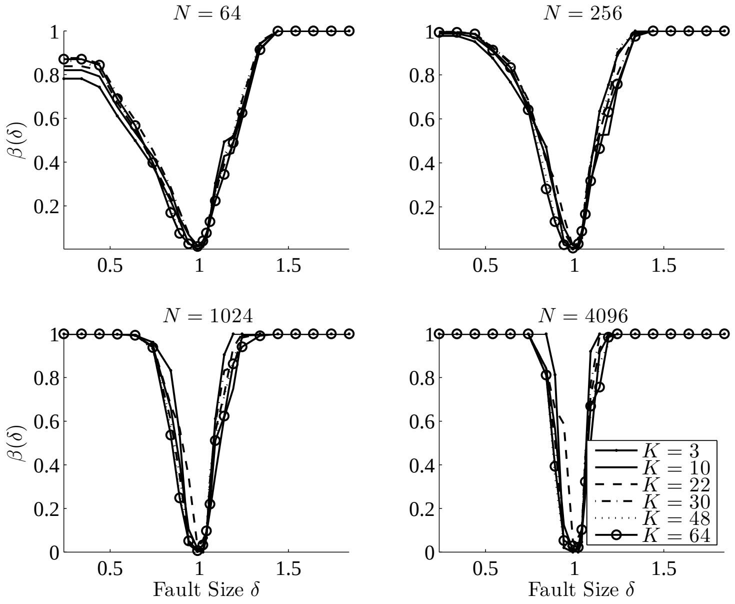


Figure 12: Comparison of power functions for the test based on $\lambda_{\mathrm{R}}(\mathcal{R})$ for a set of no-fault distribution parameters $\theta^{(K)}$ with different values of K.

6.4.3. Comparison of Tests

Figure 13 shows a comparison of the power functions for the tests based on the test statistics $\lambda_{\mathrm{R}}(\mathcal{R})$, $s_1(\mathcal{R})$, $s_2(\mathcal{R})$, $s_3(\mathcal{R})$, and $s_4(\mathcal{R})$, for different values of the parameter N, which specifies the number of residual samples in $\mathcal{R}$. For the test statistic $\lambda_{\mathrm{R}}(\mathcal{R})$, the parameter $\theta^{\mathrm{NF}} = \theta^{(10)}$ illustrated in Figure 7 was used.

Figure 13 shows that the powers of all tests increases with N and that the differences between the power of the tests seem to decrease with an increasing N. It can also be seen that the power function for the test based on $\lambda_{\mathrm{R}}(\mathcal{R})$ is near symmetric for all N, while the power functions for the other tests are asymmetric and tend to be less powerful for faults sizes $\delta < 1$. The difference in power for $\delta < 1$ is for example significant for $N = 64$.

The mean time to detection (MTD) for each of the tests based on $\lambda_{\mathrm{R}}(\mathcal{R})$, $s_1(\mathcal{R})$, $s_2(\mathcal{R})$, $s_3(\mathcal{R})$, and $s_4(\mathcal{R})$, is shown in Figure 14, for different sizes N of the sample $\mathcal{R}$.

In order to get comparable results, the MTD was computed as the mean of the detection time for the two largest faults, corresponding to $\delta = 0.2$ and $\delta = 1.8$, since all considered test statistics are able to detect these faults to some extent, see Figure 13. Each fault was injected in the test sequence at 10 time instances.

In Figure 14, it can be seen that the MTD's for all tests increase for $N > 256$. For $N < 256$, however, the MTD decreases with N for the norm-based tests and

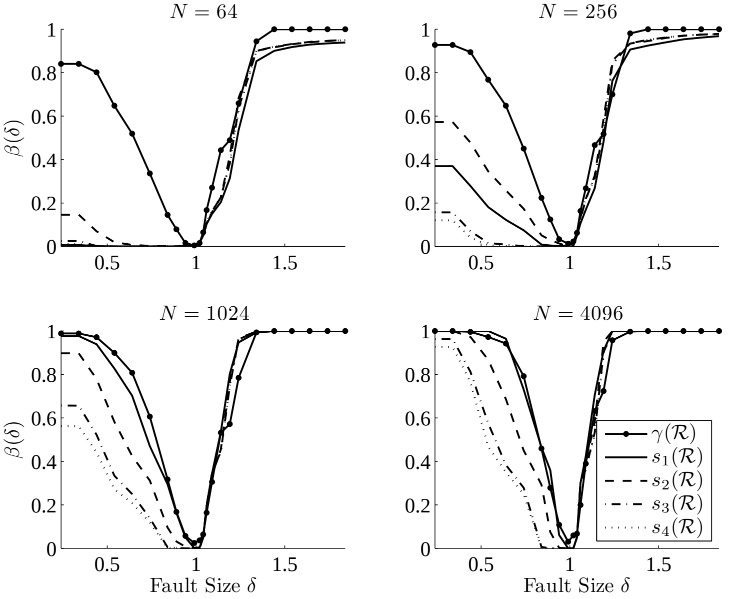


Figure 13: Comparison of power functions for the tests based on $\lambda_{\mathrm{R}}(\mathcal{R})$ (solid with dot markers), $s_1(\mathcal{R})$ (solid), $s_2(\mathcal{R})$ (dashed), $s_3(\mathcal{R})$ (dash-dotted), and $s_4(\mathcal{R})$ (dotted), for different sizes N of the sample $\mathcal{R}$.

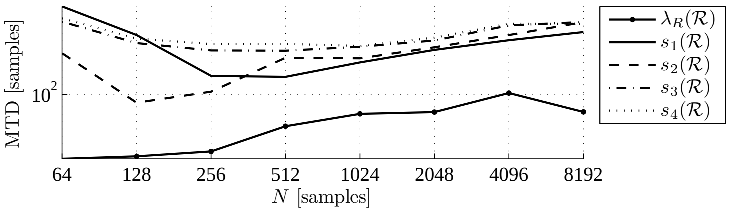


Figure 14: Comparison of the Mean Time to Detection (MTD) for the tests based on $\lambda_{\mathrm{R}}(\mathcal{R})$ (solid with dot markers), $s_1(\mathcal{R})$ (solid), $s_2(\mathcal{R})$ (dashed), $s_3(\mathcal{R})$ (dash-dotted), and $s_4(\mathcal{R})$ (dotted), for different sizes N of the sample $\mathcal{R}$.

increases with N for the test based on $\lambda_{\mathrm{R}}(\mathcal{R})$. It is worth noting that the MTD for the test based on $\lambda_{\mathrm{R}}(\mathcal{R})$ is smaller for all N than the MTD's for all other tests.

7. Conclusions

As illustrated by Figure 1, residuals in practice often deviate from zero even in the no-fault case due to uncertainties and disturbances caused by for example modeling errors, measurement noise, and unmodeled phenomena. In addition, due to changes in the operating mode of the underlying system, the magnitude of uncertainties and disturbances is time-varying, causing the behavior of residuals to be non-stationary. To handle these issues, a novel statistical residual evaluation approach has been proposed.

The main contribution is to base the residual evaluation on an explicit comparison of the probability distribution of the residual, estimated on-line using current data, with a no-fault residual distribution. The no-fault distribution is based on a set of a-priori known no-fault distributions, and is continuously adapted to the current operating mode of the system by means of the likelihood maximization problem (16). A computational efficient version of the residual evaluation test statistic suitable for online implementation has been derived by considering a properly chosen approximation (18) to the maximization problem (16). The fault detection properties of the resulting residual evaluation test have been analyzed by means of Theorems 2 and 3.

As a second contribution, a method has been proposed for learning the required set of no-fault residual distributions off-line from training data. Thus, by using this method, the overall residual evaluation method is data-driven and no assumptions regarding the properties of the probability distribution of the residual, nor the properties of the faults to be detected, are needed. The method was given by means of an algorithm based on K-means clustering, and was theoretically justified in Theorem 5.

The proposed residual evaluation method has been evaluated with measurement data on a residual for fault detection in the gas-flow system of a Scania truck diesel engine. The proposed test statistic performs well despite non-conventional properties of the considered residual. For instance, the method outperforms regular norm-based methods using constant thresholding in the sense that small faults can be detected in cases where these methods fail. It has been empirically investigated how the fault detection performance of the proposed method is influenced by different values of the involved parameters.

Acknowledgment

This work was sponsored by Scania and VINNOVA (Swedish Governmental Agency for Innovation Systems).

References

[1] M. Abid, W. Chen, S. X. Ding, and A.Q. Khan. Optimal residual evaluation for nonlinear systems using post-filter and threshold. *International Journal of Control*, 84(3):526 – 39, 2011.

[2] H. Akaike. A new look at the statistical model identification. *IEEE Transactions on Automatic Control*, 19(6):716 – 723, 1974.

[3] I. M. Al-Salami, S. X. Ding, and P. Zhang. Statistical based residual evaluation for fault detection in networked control systems. In *Proceedings of Workshop on Advances Control and Diagnosis*, Nancy, France, 2006. Nancy University.

[4] I. M. Al-Salami, K. Chabir, D. Sauter, and C. Aubrun. Adaptive thresholding for fault detection in networked control systems. In *Proceedings of the IEEE International Conference on Control Applications*, pages 446 – 451, Yokohama, Japan, 2010.

[5] D. Aloise, A. Deshpande, P. Hansen, and P. Popat. Np-hardness of euclidean sum-of-squares clustering. *Machine Learning*, 75:245–248, 2009.

[6] A. Banerjee, S. Merugu, I. S. Dhillon, J. Ghosh, and J. Lafferty. Clustering with bregman divergences. *Journal of Machine Learning Research*, 6(10): 1705 – 1749, 2005.

[7] M. Basseville and I. V. Nikiforov. *Detection of Abrupt Changes - Theory and Application.* Prentice-Hall, 1993.

[8] D. Basu. On the elimination of nuisance parameters. *Journal of the American Statistical Association*, 72(358):355–366, 1977.

[9] J. O. Berger, B. Liseo, and R. L. Wolpert. Integrated likelihood methods for eliminating nuisance parameters. *Statistical Science*, 14(1):1–22, 1999.

[10] P. Berkhin. Survey of clustering data mining techniques. *Techniques*, 10(c): 1–56, 2002.

[11] C. M. Bishop. *Pattern Recognition and Machine Learning.* Springer, 2006.

[12] A. Bjorck. *Numerical Methods for Least Squares Problems.* SIAM, Philadelphia, PA, 1996.

[13] M. Blanke, M. Kinnaert, J. Lunze, and M. Staroswiecki. *Diagnosis and Fault-Tolerant Control.* Springer, second edition, 2006.

[14] M. R. Blas and M. Blanke. Stereo vision with texture learning for fault-tolerant automatic baling. *Computers and Electronics in Agriculture*, 75(1): 159 – 68, 2011.

[15] S. Boyd and L. Vandenberghe. *Convex Optimization.* Cambridge Univ. Press, Cambridge, U.K, 2004.

[16] S. Braun and M. Feldman. Decomposition of non-stationary signals into varying time scales: Some aspects of the emd and hvd methods. *Mechanical Systems and Signal Processing*, 25:2608–2630, 2011.

[17] G. Casella and R. L. Berger. *Statistical Inference.* Duxbury Press, second edition, 2001.

[18] J. Chen and R. J. Patton. *Robust Model-Based Fault Diagnosis for Dynamic Systems.* MA: Kluwer, Boston, 1999.

[19] R. N. Clark. State estimation schemes for instrument fault detection. In R. J. Patton, P. M. Frank, and R. N. Clark, editors, *Fault Diagnosis in Dynamic Systems: Theory and Application*, chapter 2, pages 21–45. Prentice Hall, 1989.

[20] L. Davies, U. Gather, D. Nordman, and H. Weinert. A comparison of automatic histogram constructions. *ESAIM: Probability and Statistics*, 13: 181–196, 2009.

[21] A. P. Dempster, N. M. Laird, and D. B. Rubin. Maximum likelihood from incomplete data via the em algorithm. *Journal of the Royal Statistical Society*, 39(1):1–38, 1977.

[22] S. X. Ding, P. Zhang, and E. L. Ding. Fault detection system design for a class of stochastically uncertain systems. In Hong-Yue Zhang, editor, *Fault Detection, Supervision and Safety of Technical Processes 2006*, pages 705 – 710. Elsevier Science Ltd, 2007.

[23] A. Emami-Naeini, M. M. Akhter, and S. M. Rock. Effect of model uncertainty on failure detection: the threshold selector. *IEEE Transactions on Automatic Control*, 33(12):1106 –1115, 1988.

[24] P. M. Frank. Enhancement of robustness in observer-based fault-detection. *International Journal of Control*, 59(4):955–981, 1994.

[25] P. M. Frank. Residual evaluation for fault diagnosis based on adaptive fuzzy thresholds. In *IEE Colloquium on Qualitative and Quantitative Modelling Methods for Fault Diagnosis*, pages 401 –411, 1995.

[26] P. M. Frank and X. Ding. Survey of robust residual generation and evaluation methods in observer-based fault detection systems. *Journal of Process Control*, 7(6):403 – 424, 1997.

[27] J. J. Gertler. *Fault Detection and Diagnosis in Engineering Systems.* Marcel Dekker, 1998.

[28] F. Gustafsson. *Adaptive Filtering and Change Detection.* Wiley, 2000.

[29] K. H. Haskell and R. J. Hanson. An algorithm for linear least squares problems with equality and nonnegativity constraints. *Mathematical Programming*, 21(1):98–118, 1981.

[30] T. Heyns, S.J. Godsill, de Villiers J.P., and P.S. Heyns. Statistical gear health analysis which is robust to fluctuating loads and operating speeds. *Mechanical Systems and Signal Processing*, 27:651–666, 2012.

[31] T. Höfling and R. Isermann. Fault detection based on adaptive parity equations and single-parameter tracking. *Control Engineering Practice*, 4 (10):1361 – 1369, 1996.

[32] M. Inaba, N. Katoh, and H. Imai. Applications of weighted voronoi diagrams and randomization to variance-based k-clustering: (extended abstract). In *Proceedings of the tenth annual symposium on Computational geometry*, SCG '94, pages 332–339, New York, NY, USA, 1994. ACM.

[33] A. Ingimundarson, A. G. Stefanopoulou, and D. A. McKay. Model-based detection of hydrogen leaks in a fuel cell stack. *IEEE Transactions on Control Systems Technology*, 16(5):1004 –1012, 2008.

[34] S. Kullback and R. A. Leibler. On information and sufficiency. *Annals of Mathematical Statistics*, 22(1):79–86, 1951.

[35] C. L. Lawson and R. J. Hanson. *Solving Least Squares Problems.* Prentice-Hall, Englewood Cliffs, NJ, 1974.

[36] W. Li, Z. Zhu, and S. X. Ding. Fault detection design of networked control systems. *IET Control Theory and Applications*, 5(12):1439 – 49, 2011.

[37] L. Ljung. *System Identification - Theory for the User.* Prentice-Hall, Upper Saddle River, N.J., 2 edition, 1999.

[38] S. P. Lloyd. Least squares quantization in pcm. *IEEE Transactions on Information Theory*, 28(2):129–137, 1982.

[39] J. B. MacQueen. Some methods for classification and analysis of multivariate observations. In *Proceedings of 5th Berkeley Symposium on Mathematical Statistics and Probability*, pages 281–297. University of California Press, 1967.

[40] B. Manthey and H. Röglin. Worst-case and smoothed analysis of k-means clustering with bregman divergences. In Yingfei Dong, Ding-Zhu Du, and Oscar Ibarra, editors, *Algorithms and Computation*, volume 5878 of *Lecture Notes in Computer Science*, pages 1024–1033. Springer Berlin, Heidelberg, 2009.

[41] J. Mattingley and S. Boyd. Real-time convex optimization in signal processing. *IEEE Signal Processing Magazine*, 27(3):50–61, 2010.

[42] J. Mattingley and S. Boyd. CVXGEN: a code generator for embedded convex optimization. *Optimization and Engineering*, 13(1):1–27, 2012.

[43] G. Meltzer and N.P. Dien. Fault diagnosis in gears operating under non-stationary rotational speed using polar wavelet amplitude maps. *Mechanical Systems and Signal Processing*, 18:985–992, 2004.

[44] S. A. Murphy and A. W. van der Vaart. On profile likelihood. *Journal of the American Statistical Association*, 95(450):449–465, 2000.

[45] Y. Nesterov and A. Nemirovskii. *Interior Point Polynomial Algorithms in Convex Programming.* SIAM, Philadelphia, PA, 1994.

[46] J. Nocedal and S. J. Wright. *Numerical Optimization.* Springer, second edition, 2006.

[47] M. Nyberg and T. Stutte. Model based diagnosis of the air path of an automotive diesel engine. *Control Engineering Practice*, 12(5):513 – 525, 2004.

[48] A.W. Osburn, T.M. Kostek, and M.A. Franchek. Residual generation and statistical pattern recognition for engine misfire diagnostics. *Mechanical Systems and Signal Processing*, 20:2232–2258, 2006.

[49] W. M. Patefield. On the maximized likelihood function. *The Indian Journal of Statistics, Series B (1960-2002)*, 39(1):92–96, 1977.

[50] Y. Peng, A. Youssouf, P. Arte, and M. Kinnaert. A complete procedure for residual generation and evaluation with application to a heat exchanger. *IEEE Transactions on Control Systems Technology*, 5(6):542 – 555, 1997.

[51] Z.K. Peng and F.L. Chu. Application of the wavelet transform in machine condition monitoring and fault diagnostics: A review with bibliography. *Mechanical Systems and Signal Processing*, 18:199–221, 2004.

[52] A.G. Poulimenos and S.D. Fassois. Parametric time-domain methods for non-stationary random vibration modelling and analysis - a critical survey and comparison. *Mechanical Systems and Signal Processing*, 20:763–816, 2006.

[53] H. Sneider and P. M. Frank. Observer-based supervision and fault detection in robots using nonlinear and fuzzy logic residual evaluation. *IEEE Transactions on Control Systems Technology*, 4(3):274 –282, 1996.

[54] T. Söderström and P. Stoica. *System Identification.* Prentice-Hall Int., London, UK, 1989.

[55] C. Svärd. *Methods for Automated Design of Fault Detection and Isolation Systems with Automotive Applications.* PhD thesis, Linköping University, june 2012. Dissertation No. 1448.

[56] C. Svärd and M. Nyberg. Residual generators for fault diagnosis using computation sequences with mixed causality applied to automotive systems. *IEEE Transactions on Systems, Man and Cybernetics, Part A: Systems and Humans*, 40(6):1310–1328, 2010.

[57] C. Svärd, M. Nyberg, E. Frisk, and M. Krysander. Residual evaluation for fault diagnosis by data-driven analysis of non-stationary probability distributions. In *Proceedings of the 50th IEEE Conference on Decision and Control and European Control Conference (CDC-ECC 2011)*, 2011.

[58] J. Wahlström and L. Eriksson. Modeling diesel engines with a variable-geometry turbocharger and exhaust gas recirculation by optimization of model parameters for capturing non-linear system dynamics. *Proceedings of the Institution of Mechanical Engineers, Part D: Journal of Automobile Engineering*, 225(7), 2011.

[59] X. Wei, H. Liu, and Y. Qin. Fault diagnosis of rail vehicle suspension systems by using glrt. In *Control and Decision Conference (CCDC), 2011 Chinese*, pages 1932 –1936, 2011.

[60] A. Willsky and H. Jones. A generalized likelihood ratio approach to the detection and estimation of jumps in linear systems. *IEEE Transactions on Automatic Control*, 21(1):108 – 112, 1976.

[61] S. J. Wright. *Primal-Dual Interior-Point Methods.* SIAM, Philadelphia, PA, 1997.

[62] X. Zhang, M. M. Polycarpou, and T. Parisini. A robust detection and isolation scheme for abrupt and incipient faults in nonlinear systems. *IEEE Transactions on Automatic Control*, 47(4):576 –593, 2002.

[63] M. Zhong, H. Ye, S.X. Ding, and G. Wang. Observer-based fast rate fault detection for a class of multirate sampled-data systems. *IEEE Transactions on Automatic Control*, 52(3):520 – 525, 2007.

[64] Y. Zhu and X. Rong Li. Recursive least squares with linear constraints. *Communications in Information and Systems*, 7(3):287–312, 2007.