Combining model-based diagnosis and data-driven anomaly classifiers for fault isolation

Daniel Jung[1 a], Kok Yew Ng[b,c], Erik Frisk[a], Mattias Krysander[a]

[a] *Vehicular Systems, Linköping University, Linköping, Sweden*
[b] *School of Engineering, Ulster University, Newtownabbey, BT37 0QB, UK*
[c] *Electrical and Computer Systems Engineering, School of Engineering, Monash University Malaysia, Malaysia*

Abstract

Machine learning can be used to automatically process sensor data and create data-driven models for prediction and classification. However, in applications such as fault diagnosis, faults are rare events and learning models for fault classification is complicated because of lack of relevant training data. A hybrid diagnosis system design is proposed which combines model-based residuals with incremental anomaly classifiers. The proposed method is able to identify unknown faults and also classify multiple-faults using only single-fault training data. A physical model and data collected from an internal combustion engine are used to verify the proposed method.

Keywords: Fault diagnosis, fault isolation, machine learning, artificial intelligence, classification.

1. Introduction

Fault detection and isolation are important tasks in fault diagnosis systems to identify the root cause when faults occur in the system. This is complicated by the fact that there are often many possible diagnosis candidates (fault hypotheses) that can explain the system state. In a workshop, this can result in a mechanic having to troubleshoot several components in a system before identifying the true fault, which can be both costly and time-consuming [1].

Two common approaches in fault diagnosis are model-based [2] and data-driven [3]. Data-driven diagnosis, in general, isolates faults by using classifiers learned from training data using nominal data and data from different faults, see for example [4]. However, in many industrial applications, faults are rare events and available training data from faulty conditions is usually limited [5, 6]. Collecting a sufficient amount of data from relevant fault scenarios is a time-consuming and expensive process. Also, if there are faults that do not occur before several years of system operation time, they might not be considered during system development. Therefore, it is desirable that a diagnosis system is not only able to identify and localize known faults as they occur, but it should also be able to identify new types of faults and improve fault classification performance over time as new data are collected.

One solution to limited training data from different fault scenarios is the use of physical models. In model-based diagnosis, fault isolation is mainly performed by matching the set of triggered residual generators with the different fault signatures to compute diagnosis candidates [7]. An advantage of model-based methods, with respect to data-driven methods, is that fault isolation performance can be achieved without training data from different faults. Even though the fault has not been observed before, it is possible to point out likely fault locations based on residual information and model analysis [8]. However, there are often many diagnosis candidates that can explain the triggered residuals, meaning that it can still be difficult to identify the actual fault.

1.1. Problem motivation

A combined diagnosis system design has the potential of both model-based and data-driven diagnosis methodologies [9]. The objective of such a hybrid diagnosis system design is to improve fault isolation performance by utilizing both physical models and data collected from previous fault occurrences. Another advantage is that performance can improve over time by incrementally retrain the data-driven classifiers over time as

Email addresses: daniel.jung@liu.se (Daniel Jung), mark.ng@ulster.ac.uk (Kok Yew Ng), erik.frisk@liu.se (Erik Frisk), mattias.krysander@liu.se (Mattias Krysander)

new data are collected. This is relevant for example in the automotive industry where fleet data from connected vehicles can identify anomalies and help diagnose each individual vehicle. The idea is to first compute diagnosis candidates (fault hypotheses) that can explain the set of triggering residual generators and then rank the different candidates which is most likely using data-diven classifiers where each classifier models a different fault hypothesis. The proposed diagnosis system structure is illustrated in Figure 1. The purpose of the data-driven classifiers is not to reject any of the diagnosis candidates but to rank which of the computed candidates are more likely.

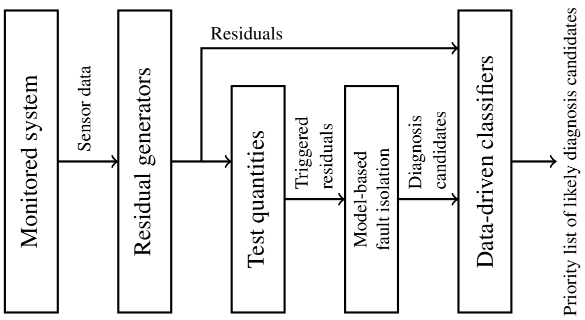

Figure 1: A schematic of the diagnosis system design. The data-driven fault isolation is used to rank diagnosis candidates computed by the consistency-based fault isolation.

This paper extends on the analysis and results of the proposed hybrid diagnosis system design presented in [10]. A framework is formulated for combining model-based and data-driven fault isolation. Also, with respect to the previous work, the performance and robustness of the proposed hybrid diagnosis system design are evaluated using a model of an internal combustion engine and collected data from a test bench.

1.2. Related research

Discussions regarding model-based and data-driven fault diagnosis methods can be found in, for example, [2], [11], and [12]. A survey of previous works combining model-based and data-driven fault diagnosis techniques is presented in [9], which also points out that there are potential advantages of applying a framework to integrate the model-based and data-driven methodologies. A hybrid framework is proposed in [13] where residuals, designed using bond graphs, and sensor data are combined using a Bayesian Network. In [14] and [15], different sets of test quantities are designed using model-based and data-driven methods. In [16], a model is estimated using data from a thermal power plant and a data-driven classifier is then used for fault isolation. Model-based residual selection is combined with training data in [17] to automatically identify important residuals and design test quantities, and in [18] residual detection performance is improved using machine learning to compensate for model uncertainties. In [19], a brief comparison is made between different hybrid approaches applied to monitor a wind turbine. Combined methods have also been proposed for prognostics and condition-based maintenance [20, 21]. With respect to previous works, this paper proposes a hybrid fault isolation strategy which takes advantage of both methodologies to improve fault isolation without increasing the risk of rejecting the true diagnosis.

2. Fault isolation and model-based diagnosis

The first step of the diagnosis system in Figure 1 follows a general model-based architecture where residuals are used to detect inconsistencies between model predictions and sensor data. In this section, it is summarized how the diagnosis candidates are computed as a set of fault hypotheses that can explain the set of triggered residuals.

In industrial systems, there are usually many potential faults that can occur that will have varying impact on the system and its performance. Let $\mathcal{F} = \{f_1, f_2, \ldots, f_{n_f}\}$ denote a set of n_f *known* types of faults to be monitored by a diagnosis system. However, note that the set $\mathcal{F} \subseteq \mathcal{F}^*$ only represents the known subset of all possible faults $\mathcal{F}^*$ that can occur in the system. Thus, the set $\mathcal{F}$ can increase over time as new types of faults are identified.

In many systems, it is possible that multiple faults can be present in the system at the same time. Therefore, to describe the system state the term *fault mode* is used which is defined as follows.

Definition 1 (Fault mode)**.** *A fault mode $F \subseteq \mathcal{F}$ is a set of faults that is present in the system.*

The nominal system state $F = \emptyset$, i.e. when the system is fault-free, is denoted the No Fault (NF) case.

2.1. Fault detection

In order to detect if a fault is present in the system, a set of residual generators $\mathcal{R} = \{r_1, r_2, \ldots, r_{n_r}\}$ is computed. A residual generator is a function of sensor and actuator data which ideally is zero in the fault-free case [22]. A residual generator is said to be sensitive to a fault f_i if that fault implies that the residual is non-zero, ideally. If a residual generator is not sensitive to fault f_i,

Table 1: Fault signature matrix.

Residual	f_{Waf}	f_{pim}	f_{pic}	f_{Tic}
r_1	X	X		
r_2	X		X	
r_3		X		X
r_4		X	X	
r_5	X			
r_6				X

is is also said that the fault is decoupled in that residual generator.

Note that the definitions of residual generators and fault sensitivity describe the ideal case. However, fault detection performance is complicated by model uncertainties and measurement noise. Therefore, a change in the residual output is usually determined by thresholding the residual. A function that evaluates if there is a change in a residual output or not is called a test quantity.

The different residual generators are designed to monitor different parts of the monitored system, i.e., to be sensitive to different subset of faults. The following definition of fault detectability for a given set of residual generators $\mathcal{R}$ is used [17].

Definition 2 (Fault mode detectability)**.** *A fault mode $F_i \subseteq \mathcal{F}$ is structurally detectable if there exists a residual generator $r_k \in \mathcal{R}$ that is sensitive to at least one fault $f \in F_i$.*

The relation between which residual generators are sensitive to which faults can be summarized in a Fault Signature Matrix (FSM). An example is shown in Table 1 where a mark at location (k, l) in the FSM indicates that residual r_k is sensitive to fault f_l. As an example, residual r_1 is sensitive to the faults f_{Waf} and f_{pim}, but not to f_{pic} and f_{Tic}.

2.2. Fault isolation

After a fault has been detected, i.e. when one or more residuals have triggered, the next step is to perform fault isolation. Fault isolation consists of computing diagnosis candidates, that can explain the set of triggered residual generators. There are different proposed methods for fault isolation, for example column matching [7], and consistency-based diagnosis [23]. The set of computed diagnoses can differ between different fault isolation algorithms. One reason is that the fault isolation algorithms are designed based on some fundamental assumptions on fault behavior [19]. Here, consistency-based diagnosis is used since it will not reject the true diagnosis candidate, as long as there are no false alarms.

Fault isolability between fault modes is defined for a set of residual generators $\mathcal{R}$ as follows [17].

Definition 3 (Fault mode isolability)**.** *A fault mode $F_i \subseteq \mathcal{F}$ is structurally isolable from another fault mode $F_j \subseteq \mathcal{F}$ if there exists a residual generator $r_k \in \mathcal{R}$ that is sensitive to at least one fault $f \in F_i$ but no fault $f \in F_j$.*

From this definition, the principles of consistency-based diagnosis for fault isolation can be summarized as follows. Initially, before any residuals have triggered, the set of possible diagnosis candidates $\mathcal{D}$ includes all possible subsets of $\mathcal{F}$, including the empty set representing that the system is fault-free. Since no diagnosis candidate $d \in \mathcal{D}$ has been rejected, they can all explain the current system state. When a residual triggers, diagnosis candidates that cannot explain the triggered residual are rejected, reducing the set of feasible candidates. All diagnosis candidates, where no subset of faults is a feasible candidate, are referred to as minimal diagnosis candidates. Since faults usually are rare events, the minimal diagnosis candidates represent the simplest but also the most likely explanations. Note that before any residual has triggered, the minimal diagnosis candidate is the fault-free case. As an example, consider the FSM in Table 1. Assume that residuals r_1 and r_2 have triggered alarms. Then, $\{f_{\text{Waf}}\}$ and $\{f_{\text{pim}}, f_{\text{pic}}\}$ are minimal diagnosis candidates. Note that, for example, $\{f_{\text{Waf}}, f_{\text{Tic}}\}$ is also a diagnosis candidate but it is not minimal since the subset $\{f_{\text{Waf}}\}$ is also a diagnosis candidate.

3. Fault classification using anomaly detection

Data-driven classifiers try to model data and find decision boundaries that best distinguish between different classes of data [24]. The different types of classifiers can broadly be divided based on how many classes of data that are used to train the classifier. Binary and multi-class classifiers are trained using data from multiple classes to determine decision boundaries that separate each class. Multi-class classifiers commonly extrapolate the decision boundaries to areas not represented by training data. If training data do not represent the different faulty scenarios, this can result in an unnecessary large set of mis-classifications.

One-class classifiers, usually referred to as anomaly classifiers [25], use data from only one class to identify if new data patterns belong to that class or not. There are many different types of data-driven anomaly classifiers proposed, for example, Principal Component Analysis (PCA), Partial Least Squares (PLS), k-means, Gaussian Mixture Models (GMM), and one-class Support Vector

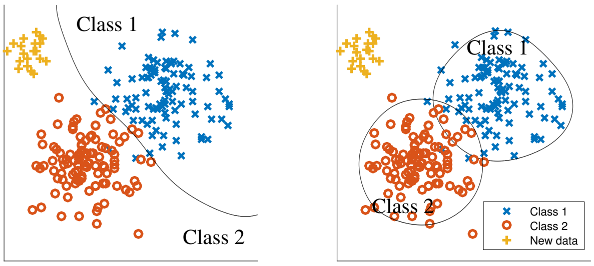

Figure 2: A comparison between a binary SVM classifier (left figure) trained using data from both class 1 and class 2, and two 1-SVM classifiers (right figure) trained for each class of data.

Machines (1-SVM) [24]. One-class classifiers are interesting alternatives to multi-class classifiers, both for fault detection and isolation since each fault mode can be modeled independently of each other. This is illustrated in Figure 2 where a binary Support Vector Machines (SVM) classifier and two 1-SVM classifiers are trained using two-dimensional data from two different classes [24]. The decision boundaries of the different classifiers are shown in the figures. When evaluated with new data, the binary SVM classifies the data as belonging to Class 2 while the two 1-SVM classifiers states that the new data does not belong to any of these classes. This can be used to identify unknown faults, i.e. new faults that have not been observed before.

3.1. Support Vector Data Description

The 1-SVM do not directly try to model the data distribution but its support. Two similar methods of 1-SVM are proposed in [26] and [27], respectively, referred to as ν-SVM and Support Vector Data Description (SVDD). The two methods utilize the kernel trick where ν-SVN uses a hyper-plane and SVDD a hyper-sphere to enclose the training data.

In order to handle large sets of data, incremental learning algorithms are necessary to reduce computational cost of updating the data-driven classifiers as new data is collected. In this paper, the incremental SVDD algorithm presented in [28] is used.

4. Merging information from model-based and data-driven fault isolation

When combining different algorithms for fault isolation, it is important to avoid contradictory conclusions, i.e. when diagnosis statements from the different algorithms are inconsistent. In [13] a Bayesian Network is used to merge information from different sources, which could contain contradictions, to determine the most likely diagnosis. The solution in this work is that consistency-based diagnosis is used to compute diagnosis candidates while a set of SVDD classifiers are used to evaluate how likely each diagnosis candidate is. Note that the data-driven classifiers are not used to reject any diagnosis candidate. To motivate the diagnosis system design in Figure 1, combining both consistency-based diagnosis and SVDD classifiers, the relation between the two fault isolation approaches is analyzed using the methodology in [19].

4.1. Modeling fault modes in consistency-based fault isolation

Different fault magnitudes and realizations of each set of faults will have different effects on the residual outputs. By assuming bounded residual uncertainties, the set of residual outputs that can be explained by a certain fault mode $F_l \subseteq \mathcal{F}$ is defined by a set $\Phi^*(F_l) \subseteq \mathbb{R}^{n_r}$. Different fault modes F_l can explain different sets of residual outputs $\Phi^*(F_l)$. Note that some residual outputs can be explained by multiple fault modes, i.e. $\Phi^*(F_{l1}) \cap \Phi^*(F_{l2}) \neq \emptyset$ is true for some F_{l1} and F_{l2}.

If the sets $\Phi^*(F_l)$ for all $F_l \subseteq \mathcal{F}$ are perfectly known, it is possible to avoid rejecting the true diagnosis candidate. However, this is rarely the case and different fault isolation algorithms try to approximate the sets $\Phi^*(F_l)$ to perform fault isolation. By comparing how each fault isolation algorithm approximates the different residual output sets $\Phi^*(F_l)$ gives information how to combine the different fault isolation methodologies to avoid inconsistencies and to improve isolation performance.

The approximation of each $\Phi^*(F_l)$ based on consistency-based diagnosis is denoted by $\Phi_{cb}(F_l)$ and is defined as follows: Let J_i be a threshold such that a residual r_i is said to have triggered if $|r_i| > J_i$. Then, the set $\Phi^*(F_l)$ is approximated as $\Phi_{cb}(F_l) = \mathbb{W}_1 \times \mathbb{W}_2 \times \ldots \times \mathbb{W}_i \times \ldots \times \mathbb{W}_{n_r}$ where

$$\mathbb{W}_i = \begin{cases} \mathbb{R} & \text{if } r_i \text{ is sensitive to any fault } f_j \in F_l \\ [-J_i, J_i] & \text{otherwise.} \end{cases}$$

Note that if $F_{l1} \subseteq F_{l2}$, then $\Phi_{cb}(F_{l1}) \subseteq \Phi_{cb}(F_{l2})$ [19]. This means that if fault mode F_{l2} is rejected, then all fault modes representing all subsets of faults are also rejected.

4.2. Comparison of fault isolation approaches

Let the set $\Phi_{\text{FIA}}(F_l)$ be an approximation of $\Phi^*(F_l)$ defined given a specific fault isolation algorithm FIA. Then, fault isolability based on the sets $\Phi_{\text{FIA}}(F_l)$ can be formulated as follows.

Definition 4 (Fault isolability given FIA). *A fault mode F_{l1} is said to be isolable from another fault mode F_{l2} given a fault isolation algorithm FIA if*

$$\Phi_{FIA}(F_{l1}) \nsubseteq \Phi_{FIA}(F_{l2}). \tag{1}$$

Some fault isolation algorithms are consequently over-estimating or under-estimating the true set $\Phi^*(F_l)$. The following definitions will be used to describe these two cases.

Definition 5. *A fault isolation algorithm FIA is called conservative if $\Phi^*(F_l) \subseteq \Phi_{FIA}(F_l)$ for all $F_l \subseteq \mathcal{F}$.*

Definition 6. *A fault isolation algorithm FIA is called optimistic if $\Phi_{FIA}(F_l) \subseteq \Phi^*(F_l)$ for all $F_l \subseteq \mathcal{F}$.*

To illustrate the difference between conservative and optimistic fault isolation algorithms consider the right plot in Figure 2. Assume that the two one-class classifiers are conservative, then the new data cannot belong to any of Class 1 or Class 2. However, if the one-class classifier for Class 1 is optimistic, the new data could belong to Class 1.

Note that if the test quantity for each residual in $R \subseteq \mathcal{R}$ is tuned such that false alarms can be neglected, consistency-based diagnosis is conservative. This is important since it implies that no diagnosis candidates, including the correct diagnosis candidate, are falsely rejected. However, this can result in unnecessary poor fault isolation performance since the number of computed diagnosis candidates might be larger than ideal.

On the other hand, data-driven fault isolation can be too optimistic if training data is not representative of all fault realizations, meaning that the true diagnosis candidate could be falsely rejected. Let the approximation of the set $\Phi^*(F_l)$ using SVDD be denoted $\Phi_{svdd}(F_l)$. Then, the relation between the consistency-based diagnosis and a SVDD classifier for fault mode $F_l \subseteq \mathcal{F}$ is given by

$$\Phi_{svdd}(F_l) \subsetapprox \Phi^*(F_l) \subseteq \Phi_{cb}(F_l) \tag{2}$$

The set $\Phi_{svdd}(F)$ is an approximate subset of $\Phi^*(F)$ because it depends on how tight the boundaries of the classifier are selected with respect to the training data.

If a SVDD classifier is trained for each fault mode, it can be used to count how many residual samples that belongs to that diagnosis candidate. By doing this for all diagnosis candidates, the candidates that have a high percentage of samples classified positive, i.e. belonging to that fault mode, are more likely compared to candidates where most samples are classified to not belong to that mode. Thus, the computed diagnosis candidates are not rejected by the SVDD classifiers but only ranked based on how many samples belongs to that mode, i.e. a higher rank corresponds to a more likely diagnosis candidate. Note that for such an approach, if more data are collected from mode F_l, the corresponding SVDD classifier can be updated meaning that $\Phi_{svdd}(F_l)$ better approximates $\Phi^*(F_l)$, and thus better classifies data from that fault mode.

5. A combined model-based and data-driven fault isolation algorithm

To improve the fault isolation performance, the proposed hybrid diagnosis system design, illustrated in Figure 1 uses a set of SVDD classifiers to rank the computed diagnosis candidates by counting the residual samples belonging to that fault mode.

5.1. Ranking diagnosis candidates using SVDD

Each fault mode F_l is modeled as an SVDD classifier

$$C^R_{F_l} : \mathbb{R}^{|R|} \to \{0, 1\}$$

where $R \subseteq \mathcal{R}$ is the set of residuals used by the classifier. Note that R can be a subset of the available residuals, however, for single-fault classification $R = \mathcal{R}$. Since SVDD models the data support and not the distribution, the minimal diagnosis candidates are ranked based on how many of the residual samples when a fault is detected are classified by each corresponding $C^R_{F_l}$. Let $\boldsymbol{r}_1, \boldsymbol{r}_2, \ldots, \boldsymbol{r}_N$, be N samples of the residuals when a fault is detected. If $F_l \in \mathcal{D}_{\min}$ is a minimal diagnosis candidate, its rank is computed as

$$\text{rank}(F_l) = \frac{1}{N} \sum_{k=1}^{N} C^R_{F_l}(\boldsymbol{r}_k), \tag{3}$$

i.e. the percentage of the samples that belongs to F_l. A higher rank(F_l) means that the diagnosis candidate F_l is ranked higher.

The use of SVDD classifiers to rank the minimal diagnosis candidates can be interpreted as evaluating new residual outputs using experience from previous faults. Some minimal diagnosis candidates should be prioritized if the residual data resembles previous observations of the fault mode. Note that even though available faulty data are limited initially, the performance of the SVDD classifiers can be improved over time as new data are collected.

5.2. Identifying unknown faults

It is also necessary to identify the likelihood that an unknown fault has occurred whereby if the residual output has not been observed before, then it does not belong to any existing fault mode. Let $D_{\min}$ denote the set of minimal diagnosis candidates, *excluding the unknown fault case*. Then, the ranking of an unknown fault $F_x = \{f_x\}$ is performed as

$$\text{rank}(F_x) = \frac{1}{N}\sum_{k=1}^{N}\left(1 - \bigwedge_{\forall F_l \in D_{\min}} C_{F_l}^{R}(\boldsymbol{r}_k)\right), \quad (4)$$

i.e. the rate of the samples that do not belong to any known fault mode. If an unknown fault has a high rank, possible locations of the fault can be identified by analyzing the model support of the triggered residuals.

5.3. Classifying multiple-faults using single-fault data

Training SVDD classifiers for all possible fault modes would require data from all combinations of different multiple-fault cases. However, since faulty data is rare, collecting data from all combinations of multiple-faults is not feasible. By using the information about fault sensitivity of the different residuals it is possible to train sets of SVDD classifiers for some multiple-fault modes using only single-fault training data.

The key observation is that a residual where a fault is decoupled will not change from its nominal behavior when that fault occurs. To evaluate if residual data belongs to a multiple-fault mode $F_l \subseteq \mathcal{F}$, a set of $|F_l|$ different residual sets, where all but one of the faults in F_l are decoupled, are utilized to train separate classifiers. Let $R_{F_l \setminus \{f_i\}} \subseteq \mathcal{R}$ denote the set of residuals where all faults $F_l \setminus \{f_i\}$ are decoupled, and at least one residual in the set is sensitive to f_i. Then, the multiple-fault mode is ranked by classifying each single-fault individually, and counting the number of samples belonging to all residual subset classifiers, as

$$\text{rank}(F_l) = \frac{1}{N}\sum_{k=1}^{N}\left(\prod_{\forall f_i \in F_l} C_{f_i}^{R_{F_l \setminus \{f_i\}}}(\boldsymbol{r}_k)\right) \quad (5)$$

Note that each classifier $C_{f_i}^{R_{F_l \setminus \{f_i\}}}(\boldsymbol{r}_k)$ only uses the subset of residuals $\boldsymbol{r}_k$ belonging to $R_{F_l \setminus \{f_i\}}$. The ability to classify multiple-faults, thus, depends on the fault sensitivities of the residuals in $\mathcal{R}$. It requires that the residual set can isolate each fault in the fault mode from the other faults.

5.4. Hybrid fault isolation summary

The diagnosis system algorithm presented in Figure 1 with the proposed hybrid fault isolation algorithm can be summarized as follows.

1. Based on the set of triggered residuals, compute a set of minimal diagnosis candidates using consistency-based diagnosis.
2. Rank each minimal diagnosis candidate by evaluating the residual outputs using an SVDD classifier trained with data from that fault mode.
   (a) If the minimal diagnosis candidate is multiple-faults, use a set of SVDD classifiers on the subset of residuals where each fault in the minimal diagnosis is decoupled and count the number of samples belonging to all SVDD classifiers.
3. Rank the unknown fault case by counting the samples not classified to any of the minimal diagnosis candidates.

6. Case study

To evaluate the hybrid diagnosis system design, a set of residuals are generated to monitor a four cylinder turbo charged internal combustion engine. The engine is mounted in a test bench, as can be seen in Fig. 3 and the available measurements represent a standard setup in a production vehicle including the following eight sensor signals: pressure before throttle y_{pic}, pressure in intake manifold y_{pim}, ambient pressure y_{pamb}, temperature before throttle y_{Tic}, ambient temperature y_{Tamb}, air mass flow after air filter y_{Waf}, engine speed y_{ω}, and throttle position y_{xpos}, and the two actuator signals: wastegate actuator u_{wg}, and injected fuel mass into the cylinders u_{mf} [17]. A mathematical model describing the air flow through the engine is used with a similar model structure as described in [29]. The right plot in Figure 3 shows a schematic illustration of the model. A set of six residual generators is selected and designed as described in [17]. The residuals are automatically generated based on a model of the engine using the Fault Diagnosis Toolbox [30].

6.1. Data collection

In this case study, data from four sensor faults have been collected: A fault in the sensor measuring the air mass flow f_{Waf}, the pressures at the intercooler f_{pic} and the intake manifold f_{pim}, and the temperature at the intercooler f_{Tic}. The FSM of the six residual generators with respect to these faults is shown in Table 1.

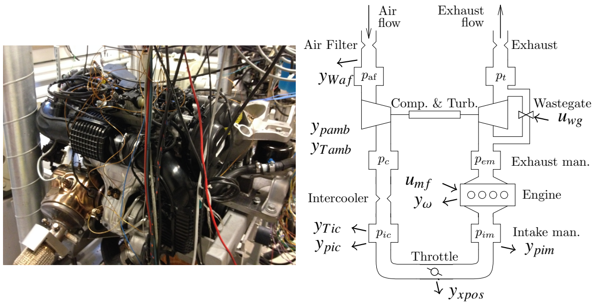

Figure 3: The left picture shows the engine test bench that is used for data collection. The right picture shows a schematic of the model of the air flow through the model. This figure is used with permission from [31].

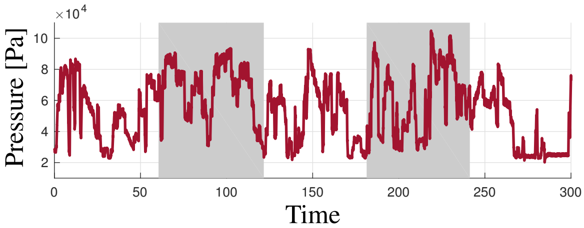

Figure 4: Intake manifold pressure sensor data y_{pim} with a highlighted intermittent fault f_{pim}.

Measurement data is generated when the engine is controlled to follow the EPA Highway Fuel Economy Test Cycle (HWFET) as a speed reference. Intermittent sensor faults are injected one by one into the engine control unit. The faults f_{Waf}, f_{pic}, and f_{pim}, are injected as multiplicative faults $y_i(t) = (1 + f_i)x_i(t)$ with a 20% change in the measured value while the fault f_{Tic} is induced as a sensor bias $y_{Tic}(t) = x_{Tic}(t) + f_{Tic}$ of 20°. Figure 4 shows an example of sensor data measuring the intake manifold pressure where a sensor fault f_{pim} occurs during the highlighted intervals.

Examples of the computed residual outputs are shown in Figures 5 and 6 including engine data from intermittent faults f_{Waf} and f_{pim}, respectively. The residuals that are sensitive to each fault are highlighted in red and the intervals when the fault is present are shaded in grey. It is visible that the effects of model uncertainties and measurement noise on the residual outputs cannot be neglected.

6.2. Evaluation

The evaluation of the hybrid diagnosis system design is performed as three different analyses. The first analysis considers a set of single-fault scenarios and is used to illustrate the advantage of using the incre-

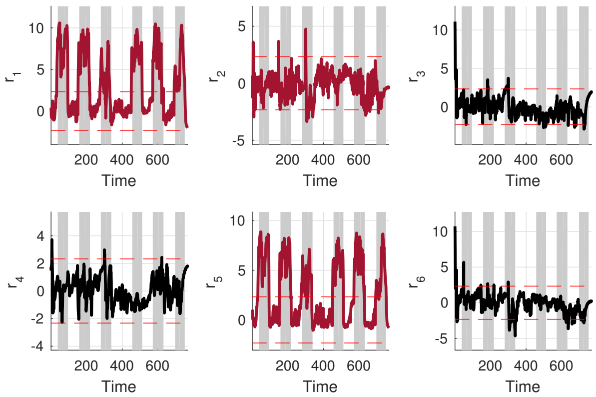

Figure 5: Evaluation of residuals to data with fault f_{Waf}. The grey areas represent intervals when the fault is present and residuals sensitive to the fault are colored red.

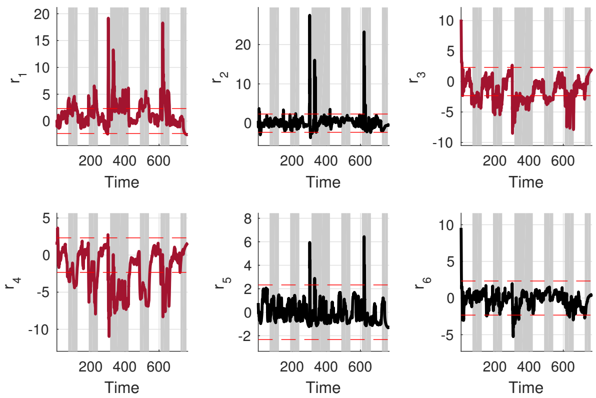

Figure 6: Evaluation of residuals to data with fault f_{pim}.

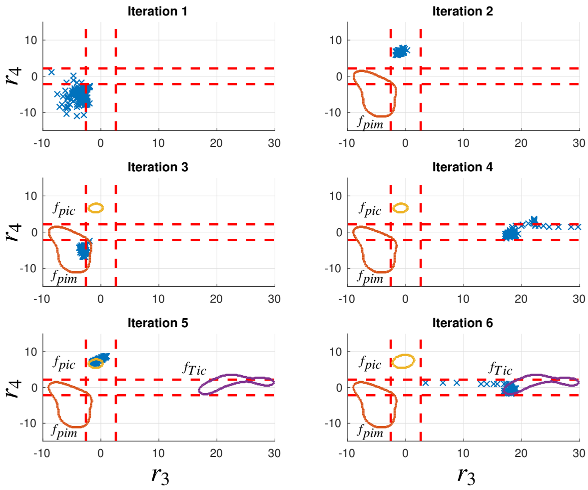

Figure 7: Residual data from r_3 and r_4 when evaluated for six fault scenarios. The dashed lines represent the thresholds for each residual used in the CUSUM tests. The colored areas represent the boundaries of each SVDD classifier for each single-fault mode in each iteration.

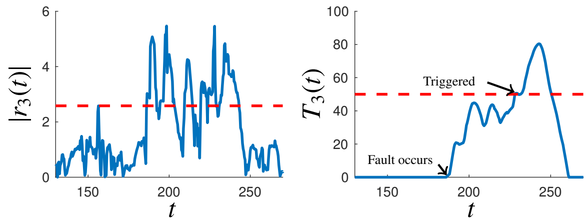

Figure 8: The left figure shows the absolute value of the residual. The right figure shows the CUSUM test which is triggered when the residual exceeds the threshold.

Table 2: Computed diagnosis candidates and their ranking after each iteration. Faults that are not minimal diagnoses in each iteration are marked with -.

Iteration	Injected fault	f_{pim}	f_{pic}	f_{Tic}	f_x
1	f_{pim}	**0**	-	-	1
2	f_{pic}	0	**0**	-	1
3	f_{pim}	**0.99**	0	-	0.01
4	f_{Tic}	0	-	**0**	1
5	f_{pic}	0	**0.61**	-	0.39
6	f_{Tic}	0	-	**0.91**	0.09

mental SVDD classifiers to rank the computed diagnosis candidates. It is shown that unknown faults are detected and the true diagnosis candidate is identified (has the highest rank) even though there are multiple minimal diagnosis candidates. The second analysis is a Monte Carlo study to show that the fault mode classification performance of the incremental SVDD improves as more data are collected. Finally, the third analysis illustrates how multiple-fault classification is performed using only single-fault training data. If multiple data sets are evaluated in sequence, it is assumed that the true diagnosis candidate is identified by an expert, after each fault scenario, and the logged data is used to update the corresponding SVDD classifier modeling that fault mode.

6.2.1. Fault isolation using incremental SVDD

To better visualize the advantage of the proposed hybrid fault isolation approach, two of the residual generators sensitive to three of the four faults in Table 1, $\{r_3, r_4\}$, are plotted against each other in Fig. 7. The two residuals also illustrates the case where all single-faults are not isolable (f_{pic} and f_{Tic} are not isolable from f_{pim}).

Initially, there are no trained SVDD classifiers. A sequence of different faulty data is evaluated, see Table 2, and the SVDD classifiers for each fault mode are updated as new data are collected. The decision boundary for each classifier is shown in Fig. 7 as well.

As test quantity, a CUmulative SUM (CUSUM) test is tuned for each residual using nominal data to reduce the risk of false alarms [32],

$$T_k(t) = \max\left(T_k(t-1) + |r_k(t)| - J_k, 0\right) \tag{6}$$

An example of the residual $r_k(t)$ and the test quantity $T_k(t)$ is shown in Fig. 8. The thresholds J_k for each of the six residual are shown as dashed lines in Fig. 5 and Fig. 6, respectively. The CUSUM tests are also used to estimate the starting time when a fault occurs by storing the last instance of time t when the test quantity $T(t)$ was equal to zero as illustrated in Fig. 8. When a fault is detected, the estimated time of the fault occurrance is used to determine the interval of the data set to rank the difference minimal diagnosis candidates.

In each iteration of the analysis, the minimal diagnosis candidates are computed based on the residuals that have triggered. Then, each minimal diagnosis candidate is ranked using the corresponding SVDD classifier, if available. Finally, after the true diagnosis candidate has been identified, the faulty data are used to update the corresponding SVDD classifier.

A summary of the fault isolation performance in each iteration is tabulated in Table 2. Only single-fault minimal diagnosis candidates are presented in the table. All faults that are minimal diagnosis candidates, including the unknown fault case, are ranked in the interval $[0, 1]$ representing the percentage of samples classified to belong to that fault mode. Single-faults that are not feasible diagnosis candidates, i.e. have been rejected, are marked with an (-). The true diagnosis candidate in each iteration is also highlighted. Note that in iteration 1, the double-faults $\{f_{pic}, f_{Tic}\}$ is also a minimal diagnosis. However, this is only the case in iteration 1 and therefore the mode is not included in the table.

In iterations 1, 2, and 4, i.e. the first time each fault occurs, respectively, their corresponding minimal diagnoses are ranked zero since there is no SVDD classifier trained for the fault modes. Thus, in these cases the unknown fault case is ranked highest. This is expected, as the faults have not been observed before. When the same type of fault occurs for the second time, the true diagnosis candidate is ranked highest showing that the SVDD classifiers improve on the fault isolation accuracy. Note that in iteration 3, when f_{pim} occurs for the second time, the fault magnitude is smaller than the first time, causing only r_4 to trigger, as shown in Fig. 7. The residual value exceeds the threshold but the CUSUM test for r_3 has not deviated enough to trigger. Therefore, both f_{pim} and f_{pic} are minimal diagnoses in iteration 3, compared to only f_{pim} in iteration 1. However, the correct diagnosis candidate received the highest rank which means that the true fault is identified even though fewer residuals have triggered.

6.2.2. *Robustness analysis using Monte Carlo simulation*

A Monte Carlo study is performed to evaluate how classification accuracy of the SVDD classifiers improves as more data are collected from different faults. Here, the computation of diagnosis candidates using consistency-based diagnosis is omitted and all fault modes are ranked in all scenarios. The evaluation is performed such that the SVDD classifiers are initialized without any training data. Then, the faults are selected one at a time in a repeated random order. Then for each fault, a data set including one realization of the fault is selected randomly for evaluation. In this analysis, all single-fault modes and the unknown fault case are ranked during each iteration. Before data from all four fault modes has been evaluated, only the faults from available SVDD classifiers are evaluated.

A summary showing the improvement in fault classification accuracy over 100 Monte Carlo simulations is shown in Fig. 9. Each row represents the true fault while each column represents each of the four fault classifiers and the unknown fault case. Each subfigure shows the distribution of the fault ranking as a function of the number of observed fault scenarios as box plots, where '+' represents outliers. The results show that fault isolation improves after each iteration. This is visible as the ranking distribution of the true fault increases and

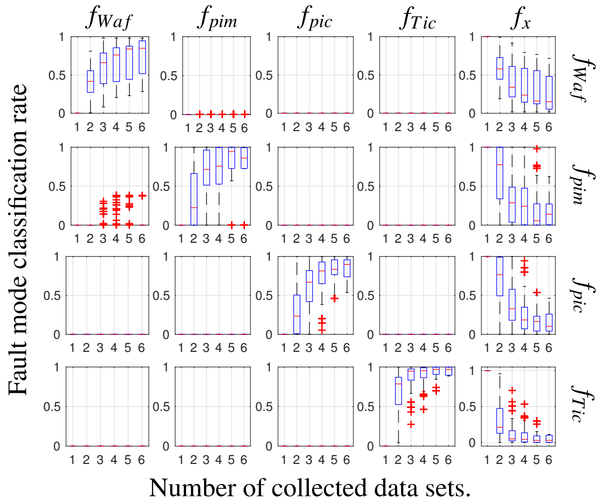

Figure 9: Monte Carlo analysis showing that fault isolation accuracy improves as more data from different fault modes is collected. There are six fault scenarios from each fault mode that are evaluated in permuted order in each simulation. The updated SVDD classifiers improve ranking of the true diagnosis candidate and decrease the rank of the unknown fault case.

the unknown fault case decreases with each iteration, as more data is collected. In some cases when f_{Pim} is the true fault, the fault f_{Waf} also has positive ranking. However, the true diagnosis candidate is still ranked higher on average. This Monte Carlo study shows the robustness of the proposed method and that fault isolation performance improves as more faulty data is collected.

6.2.3. *Classifying and ranking multiple-faults*

To illustrate the multiple-fault classification approach described in Section 5.3 two double-fault data sets are generated for evaluation, $\{f_{\text{Waf}}, f_{\text{pim}}\}$ and $\{f_{\text{pim}}, f_{\text{pic}}\}$, where the two faults are occurring simultaneously in each data set.

The multi-variate residual data from each single-fault and the two double-fault cases are visualized using a data-driven algorithm called t-Student Stochastic Nearest Embedding (t-SNE) [33], see Figure 10. The multiple-faults clearly deviate from single-fault data. This means that unless there are training data from the multiple faults, a diagnosis candidate will be ranked zero since there is no classifier for that mode.

In addition to the set of SVDD classifiers for each of the four single-faults used in the previous analyses, two multiple-fault SVDD classifiers (5) for each of the two new modes are trained using single-fault data as discussed in Section 5.3. To classify the double-faults $\{f_{\text{Waf}}, f_{\text{pim}}\}$, two residual subsets are selected where each

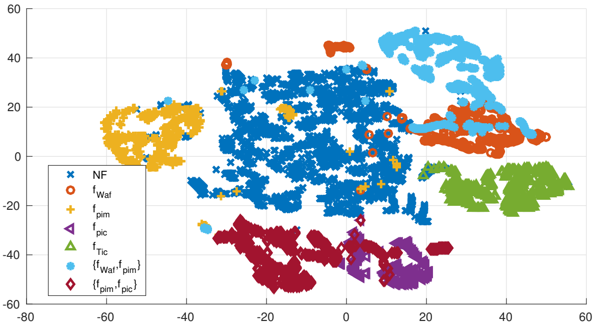

Figure 10: Visualization of residual data using t-SNE from all single-faults and the two double-fault scenarios.

of the two faults are decoupled, respectively. The residuals $R_{f_{\text{pim}}} = \{r_2, r_5, r_6\}$ are all insensitive to f_{pim} while the residuals $R_{f_{\text{Waf}}} = \{r_3, r_4, r_6\}$ are insensitive to f_{Waf}, see Table 1. Ranking of the multiple-faults $\{f_{\text{Waf}}, f_{\text{pim}}\}$ is performed given (5) using

$$\text{rank}(\{f_{\text{Waf}}, f_{\text{pim}}\}) = \frac{1}{N}\sum_{k=1}^{N}\left(C_{f_{\text{Waf}}}^{R_{f_{\text{pim}}}}(\bar{r}_k) C_{f_{\text{pim}}}^{R_{f_{\text{Waf}}}}(\bar{r}_k)\right) \quad (7)$$

The t-SNE plots are shown in Fig. 11 for each residual subset and it is visible that the double-fault data $\{f_{\text{Waf}}, f_{\text{pim}}\}$ are projected onto the corresponding single-fault data since data samples are overlapping in each figure. Also, note that the decoupled fault is projected to nominal data as expected. Thus, the double-fault case can be identified by counting samples belonging both single-fault modes for each corresponding residual subset.

The results when evaluating the identification of each double-fault scenario are summarized in Table 3. The true multiple-fault mode is correctly ranked highest in each case. The results illustrate that (5) can help to identify double-faults, using the fault sensitivity information of the residual set, even when only single-fault training data are available.

6.3. Discussion

The first analysis in Section 6.2.1 illustrates the advantage of the proposed hybrid diagnosis system compared to more conventional model-based diagnosis system design where the diagnosis candidate can be identified even though all residuals have not triggered as expected as illustrated in this analysis. Initially, before any training data has been collected, the hybrid diagnosis system will give equal results as a conventional model-based diagnosis system since all diagnosis candidates will be ranked equally likely. Isolation performance will improve over time as more training data are

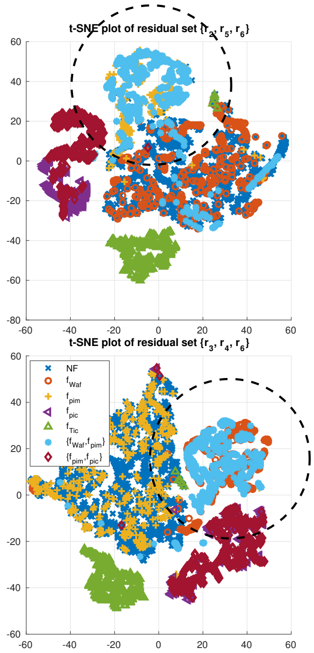

Figure 11: Visualization using t-SNE of subset of residuals where each fault f_{Waf} and f_{pim} is decoupled, respectively. It is visible that double-fault data overlaps with respective single-fault case and the decoupled fault data overlaps with nominal data.

collected from different faults as shown in the analysis in Section 6.2.2.

A complicating factor of data-driven fault isolation is that training data are needed from all fault modes to be classified. By utilizing model information, as shown in Section 6.2.3, it is possible to decouple the effects of different faults in the classifiers which makes it possible to classify multiple-fault modes without having multiple-fault training data. Another advantage of including physical-based models is that is possible to identify likely locations of unknown faults. Data-driven methods can detect and automatically cluster data, see

Table 3: Ranking of different fault modes when evaluating data with multiple-faults. The true multiple-fault mode is identified in both test cases.

	Injected faults	
Fault mode	$\{f_{Waf}, f_{pim}\}$	$\{f_{pim}, f_{pic}\}$
f_{Waf}	0	0
f_{pim}	0	0
f_{pic}	0	0.01
f_{Tic}	0	0
$\{f_{Waf}, f_{pim}\}$	**0.63**	0
$\{f_{pim}, f_{pic}\}$	0	**0.55**
f_x	0.37	0.43

for example [6] but fault localization of unknown faults is complicated without help from an expert. However, model development can be time consuming and model accuracy is important for model-based fault detection. For the proposed hybrid diagnosis system, it is important that the model is accurate enough to be able to decouple faults when designing residual generators to compute diagnosis candidates. For technical systems where such models are available, the proposed hybrid diagnosis system design can be used.

7. Conclusions

When introducing new industrial systems, the limited amount of training data restricts the application of machine learning methods for fault diagnosis. Physical models and model-based diagnosis can solve this problem and, as new data is collected, can be used to incrementally improve classification performance over time. The combined model-based and data-driven diagnosis system design has shown to improve fault isolation accuracy without increasing the risk of falsely rejecting the true diagnosis candidate. It is also possible to correctly classify multiple-faults, even when training data only contains single-fault scenarios. However, it is assumed that the true fault mode is verified by an expert before being used to update the data-driven classifiers to assure that data is correctly labeled. The proposed diagnosis system design can be implemented either online as a whole, or as one part run on-line in the system and the other part as a cloud-based system for data logging and analysis. This is useful in industrial applications, such as troubleshooting or maintenance planning, where fleet operational data is available.

7.1. Future works

For future works, it is relevant to investigate how to select suitable residual generators for fault detection and isolation when all types of faults are not known. The proposed fault isolation method assumes there is an expert able to correctly label faulty data before training each classifier. There is always a risk that the expert misclassifies the data which will have negative impact on classification performance. Thus, it is relevant to evaluate different types of semi-supervised learning algorithms to generate the one-class classifiers that can handle both non-labelled and mislabelled data.

Acknowledgement

The research has been partially funded by Volvo Car Corporation in Gothenburg, Sweden.

References

[1] A. Pernestål, M. Nyberg, H. Warnquist, Modeling and inference for troubleshooting with interventions applied to a heavy truck auxiliary braking system, Engineering applications of artificial intelligence 25 (4) (2012) 705–719.

[2] V. Venkatasubramanian, R. Rengaswamy, K. Yin, S. Kavuri, A review of process fault detection and diagnosis: Part i: Quantitative model-based methods, Computers & chemical engineering 27 (3) (2003) 293–311.

[3] S. Yin, S. Ding, X. Xie, H. Luo, A review on basic data-driven approaches for industrial process monitoring, IEEE Transactions on Industrial Electronics 61 (11) (2014) 6418–6428.

[4] A. Theissler, Detecting known and unknown faults in automotive systems using ensemble-based anomaly detection, Knowledge-Based Systems 123 (2017) 163–173.

[5] C. Sankavaram, A. Kodali, K. Pattipati, S. Singh, Incremental classifiers for data-driven fault diagnosis applied to automotive systems, IEEE Access 3 (2015) 407–419.

[6] L. Dong, L. Shulin, H. Zhang, A method of anomaly detection and fault diagnosis with online adaptive learning under small training samples, Pattern Recognition 64 (2017) 374–385.

[7] M.-O. Cordier, P. Dague, F. Levy, J. Montmain, M. Staroswiecki, L. Trave-Massuyes, Conflicts versus analytical redundancy relations: a comparative analysis of the model based diagnosis approach from the artificial intelligence and automatic control perspectives, IEEE Transactions on System, Man, and Cybernetics, Part B: Cybernetics 34 (5) (2004) 2163–2177.

[8] X. Pucel, W. Mayer, M. Stumptner, Diagnosability analysis without fault models, in: 20th International Workshop on Principles of Diagnosis, 2009, pp. 67–74.

[9] K. Tidriri, N. Chatti, S. Verron, T. Tiplica, Bridging data-driven and model-based approaches for process fault diagnosis and health monitoring: A review of researches and future challenges, Annual Reviews in Control 42 (2016) 63–81.

[10] D. Jung, K. Ng, E. Frisk, M. Krysander, A combined diagnosis system design using model-based and data-driven methods, in: IEEE 3rd Conference on Control and Fault-Tolerant Systems, 2016, pp. 177–182.

[11] V. Venkatasubramanian, R. Rengaswamy, S. Kavuri, K. Yin, A review of process fault detection and diagnosis: Part iii: Process history based methods, Computers & chemical engineering 27 (3) (2003) 327–346.

[12] S. Ding, P. Zhang, T. Jeinsch, E. Ding, P. Engel, W. Gui, A survey of the application of basic data-driven and model-based methods in process monitoring and fault diagnosis, in: IFAC World Congress, 2011, pp. 12380–12388.

[13] K. Tidriri, N. Chatti, S. Verron, T. Tiplica, Generic framework for hybrid fault diagnosis and health monitoring of the tennessee eastman process., in: 17th International Conference on Control, Automation and Systems (ICCAS), 2017, pp. 155–160.

[14] I. Loboda, S. Yepifanov, A mixed data-driven and model based fault classification for gas turbine diagnosis, in: ASME Turbo Expo: Power for Land, Sea, and Air, American Society of Mechanical Engineers, 2010, pp. 257–265.

[15] J. Luo, M. Namburu, K. Pattipati, L. Qiao, S. Chigusa, Integrated model-based and data-driven diagnosis of automotive antilock braking systems, IEEE Transactions on Systems, Man and Cybernetics, Part A: Systems and Humans 40 (2) (2010) 321–336.

[16] N. Shashoa, G. Kvaščev, A. Marjanović, Ž. Djurović, Sensor fault detection and isolation in a thermal power plant steam separator, Control Engineering Practice 21 (7) (2013) 908–916.

[17] D. Jung, C. Sundström, A combined data-driven and model-

based residual selection algorithm for fault detection and isolation, IEEE Transactions on Control Systems Technology PP (99) (2017) 1–15.

[18] Y. Cheng, R. Wang, M. Xu, A combined model-based and intelligent method for small fault detection and isolation of actuators, IEEE Transactions on Industrial Electronics 63 (4) (2016) 2403–2413.

[19] D. Jung, H. Khorasgani, E. Frisk, M. Krysander, G. Biswas, Analysis of fault isolation assumptions when comparing model-based design approaches of diagnosis systems, IFAC-PapersOnLine 48 (21) (2015) 1289–1296.

[20] C. Chen, M. Pecht, Prognostics of lithium-ion batteries using model-based and data-driven methods, in: IEEE Conference on Prognostics and System Health Management, 2012.

[21] C. Sankavaram, B. Pattipati, A. Kodali, K. Pattipati, M. Azam, S. Kumar, M. Pecht, Model-based and data-driven prognosis of automotive and electronic systems, in: IEEE International Conference on Automation Science and Engineering, 2009, pp. 96–101.

[22] C. Svärd, M. Nyberg, E. Frisk, Realizability constrained selection of residual generators for fault diagnosis with an automotive engine application, IEEE Transactions on Systems, Man, and Cybernetics: Systems 43 (6) (2013) 1354–1369.

[23] R. Reiter, A theory of diagnosis from first principles, Artificial intelligence 32 (1) (1987) 57–95.

[24] T. Hastie, R. Tibshirani, J. Friedman, J. Franklin, The elements of statistical learning: data mining, inference and prediction, The Mathematical Intelligencer 27 (2) (2005) 83–85.

[25] V. Chandola, A. Banerjee, V. Kumar, Anomaly detection: A survey, ACM computing surveys (CSUR) 41 (3) (2009) 15.

[26] B. Schölkopf, R. Williamson, A. Smola, J. Shawe-Taylor, J. Platt, et al., Support vector method for novelty detection., in: NIPS, Vol. 12, Citeseer, 1999, pp. 582–588.

[27] D. Tax, R. Duin, Support vector data description, Machine learning 54 (1) (2004) 45–66.

[28] D. Tax, Ddtools, the data description toolbox for matlab, version 2.1.2 (June 2015).

[29] L. Eriksson, Modeling and control of turbocharged si and di engines, OGST-Revue de l'IFP 62 (4) (2007) 523–538.

[30] E. Frisk, M. Krysander, D. Jung, A toolbox for analysis and design of model based diagnosis systems for large scale models, in: IFAC World Congress, Toulouse, France, 2017.

[31] L. Eriksson, S. Frei, C. Onder, L. Guzzella, Control and optimization of turbo charged spark ignited engines, in: IFAC World Congress, 2002.

[32] M. Basseville, I. Nikiforov, et al., Detection of abrupt changes: theory and application, Vol. 104, Prentice Hall Englewood Cliffs, 1993.

[33] L. Van Der Maaten, Accelerating t-sne using tree-based algorithms., Journal of machine learning research 15 (1) (2014) 3221–3245.