Detection and Diagnosis of Incipient Faults in Heavy-Duty Diesel Engines

Ian Morgan, *Member, IEEE*, Honghai Liu, *Senior Member, IEEE*, Bernardo Tormos, and Antonio Sala, *Member, IEEE*

***Abstract*—This paper proposes a new methodology for detecting and diagnosing faults found in heavy-duty diesel engines based upon spectrometric analysis of lubrication samples and is compared against a conventional method, the redline limits, which is utilized in a number of major laboratories in the U.K. and across Europe. The proposed method applies computational power to a well-known maintenance technique and consists of an improved method of preprocessing to form a derivative tuple, which extracts further information from the measured elemental concentrations. To identify incipient faults, the distance in vector space is calculated using a Gaussian contour, generated from prior data, as the zero crossing, which enables novel samples to be classified as normal or abnormal. This information is utilized as the input to a probabilistic directed acyclic graph in the form of a belief network. This network provides a prognosis for the mechanism as well as suggesting possible actions that could be taken to rectify the diagnosed problem, supported with confidence probabilities. The proposed method is evaluated for both accuracy in detecting a fault as well as the duration of time that is provided before the event occurs, with significant improvements in both metrics demonstrated over the conventional method.**

***Index Terms*—Bayesian belief network, diagnosis, diesel engines, fault detection, Gaussian, incipient faults, one-class classification, spectrometry.**

I. Introduction

RESEARCH has suggested that the closer a computational model reflects the condition of the engine, the better the fault diagnosis and detection ability of the model [1]. Conversely, to form a detailed mathematical construct concerning a complex thermodynamic system can be inhibitively costly or, in some cases, impossible. As a result, it is necessary to find an appropriately accurate model which can also be applied to other systems without further significant work.

There are four primary methods of monitoring a mechanism's condition which are vibration [2], thermography, acoustic emission [3], and oil analysis [4]–[9]. It is preferred to utilize these methods in combination, an approach known as sensor fusion [10]; however, it is too expensive to be employed on large scale, hence, a method is selected based on the speed of degradation that can be expected.

The former three can traditionally be considered as real-time methods, which are more likely to be applied in gas turbines (where the onset of a fault can be relatively fast) [11], [12]. Oil analysis has also been achieved online, to the extent of robustly detecting debris size and ferromagnetic against nonferromagnetic particles in recent research [13], [14]. These approaches have also been successfully applied to diesel engines, as in both [2] and [15] and a comparative study in [3], where acoustic emission was concluded to be the optimal method to detect pitting in spur gears. On the other hand, both [7] and [16] conclude that the former three methods may not be so appropriate for the detection of problems in larger engines and, in fact, will not be able to provide a detailed diagnosis, as the information that such sensors provide is localized.

The latter method, oil analysis, is generally based on samples that are taken infrequently; on the other hand, the samples have more information about the condition of the engine. The process of "wear" cannot be observed directly; therefore, it is an assumption that can be made by identifying the constituents of the lubrication oil. Through this, the most active wear processes occurring in the engine can be deduced, as of course, there are many processes occurring at one given point in time. Normal wear in an engine differs throughout the engine's lifetime [8] and could be monitored using the methods in [13] and [14], although extra information from chemical analysis of the oil may be required for a more detailed diagnosis.

A recently replaced component, for example, will experience a period of higher wear known as *running in* until reaching a suitable wearing pattern; under higher load conditions, wear will also be higher, and wear will also increase when a component is in contact with another component. There are many factors which control the quantification of these reactions, such as the reliability of measurements, operating conditions (for example, temperature and humidity can also be a factor for long-distance shipping), as well as consumption and composition of the lubricating oil. In larger engines, it may also be necessary to adjust the measured concentrations according to some known, or learnt, method as in [8] where the known wear rate has been calculated from an extended statistical study on a subset of engines. Alternatively, such parameters can be learnt as in [17] and [18] which infer properties from a small number of measurements.

Manuscript received June 24, 2009; revised October 9, 2009 and October 29, 2009; accepted November 24, 2009. Date of publication December 31, 2009; date of current version September 10, 2010. The work of I. Morgan and H. Liu was supported by Engineering and Physical Sciences Research Council Grant CASE/CNA/05/67. The work of A. Sala was supported by Grants DPI2008-06731-C02-01 and PROMETEO/2008/088 from the Spanish and Valencian governments.

I. Morgan and H. Liu are with the Intelligent Systems and Robotics Group, University of Portsmouth, PO1 2UP Portsmouth, U.K. (e-mail: ian.morgan@port.ac.uk; honghai.liu@port.ac.uk).

B. Tormos is with the CMT Motores Termicos, Universidad Politecnica de Valencia, 46022 Valencia, Spain (e-mail: betormos@mot.upv.es).

A. Sala is with the Instituto de Automatica e Informatica Industrial, Universidad Politecnica de Valencia, 46022 Valencia, Spain (e-mail: asala@isa.upv.es; http://www.isa.upv.es/asala).

Digital Object Identifier 10.1109/TIE.2009.2038337

Utilizing a similar philosophy to [17], therefore, wear elements transported in the lubrication oil can also be used to indirectly observe the status of the engine. Not only can these be used to infer the presence of a fault but the rate of wear and type of element can also suggest both the location and extent of a fault.

As it is necessary to monitor changes to normal wearing patterns and to make inferences between the concentrations of different elements in the lubrication oil measured in parts per million (ppm), computational analysis is very appropriate for the task, although it has not yet been employed extensively upon the data obtained from elemental analysis. As noted in [6]:

> As users increase oil-analysis programs, increase the frequency of sampling, and add more sophisticated tests to the analyses, the need to manage, organize, alert, and diagnose automatically has never been greater. Thus, this justifies the interest in this type of automatic diagnosis tool.

Expert systems have been utilized in the majority of fault-diagnosis applications [6], [19], [20], which are often more trusted in industrial applications; however, the approach has been discounted in some domains due to the effort required to collect such a detailed knowledge base. As stated by [6], expert systems cannot respond "creatively" to unexpected circumstances, and it is difficult to determine when the input values go outside a predefined range; for example, a question could be posed as to whether it is preferable to receive a possibly incorrect answer or no answer at all [21]. Due to the explicit rule base, it has also been shown that such approaches become intractable for more complex interactions [22].

Although the expert system should not be discounted as it is invaluable to the diagnosis of faults, there has been some move toward developing more "intelligent" algorithms to perform a similar task, which generally take less effort to compile and, as a result, are more efficient for industrial purposes, which suggests the prevalence of two approaches.

The first are supervised approaches which typically use "targets" for a subset of possible faults. For example, [23] used a neural network to detect and diagnose a subset of faults seen in an interior-permanent-magnet (IPM) motor drive. The line currents of different faulted and normal conditions of the IPM motor were preprocessed using a wavelet transform, which were then trained on the generated subset of faults and validated using a traditional feedforward neural network. The antithesis of this is an "unsupervised" approach, which is more often used in research into heavy-duty engines, as such engines are too expensive to be actively damaged and, consequently, are not so easy to validate. Hence, an assumption must be made that the engine is running normally for the majority of time, an assumption that [24] utilizes where a statistical model is generated from the "normal" data obtained from trains running in real life, which can then be diagnosed using an expert system and which is a more practical approach for a large-scale intelligent analysis of diesel engines.

It was found in [6] that an expert system, coupled with elemental analysis, can provide more detailed feedback to the user than systems that purely use an intelligent classifier or extended statistical study [24], [25]. For example, [26] used a Bayesian belief network to identify and diagnose problems in a space-shuttle's main engine, where a complete set of data was available. The concept of a belief network has been used in a number of different situations, in research by [27] for an airplane engine and, in [21], a belief network was utilized for diagnosis of faults in a gas turbine. The typical methodology of fault diagnosis will be retained in this methodology, that of a preprocessing module, a detection module, and subsequently, a diagnosis module [28]. Consequently, the approach has been validated in a number of previous applications.

To summarize, little research has been specifically conducted on fault diagnosis and detection for diesel engines based on elemental analysis even though this approach demonstrates a good capability for providing a more complete understanding of the processes occurring in an engine, for comparatively little cost, when juxtaposed with real-time methods such as vibration analysis, thermography, or acoustic-emission analysis. Furthermore, elemental analysis enables a diagnosis even when the oil samples are taken from only one location. Due to the focus of intelligent algorithms upon classification, there are also now a number of ways that can be further explored for computational analysis of the signals provided by elemental analysis which would enable a significant added value for current oil-analysis systems which, despite being used in many industries, significantly in the train, automotive, and naval industries, is still not taken advantage of to its fullest potential.

This paper is organized as follows: The methodology and associated research is presented and summarized in Section II, complemented by a pseudocode to clarify the core modules. A number of evaluation metrics are presented in Section III, and, subsequently, the methodology is compared for sensitivity with redline limits, a commonly used industry standard. The results are summarized, and this paper is concluded in Section IV.

II. Fault Detection and Diagnosis Methodology

The aim of condition-based monitoring is to detect faults and provide a diagnosis, or prognosis, of potential problems, which can be systematically followed by an end user who may have little understanding of the different elements and what high or low concentrations may be indicative of. Therefore, it is necessary to both define a system that combines both detection and diagnosis and to demonstrate that this is an improvement over the typical industry standard of *redline limits* for detecting faults at an early stage (incipient faults) earlier than the redline limits, and that the diagnosis provided is accurate when compared with a ground truth. This section will introduce a methodology that has been developed specifically for utilization of data without targets, i.e., unsupervised, and consequently requires historical data from the same mechanism; however, it does not require detailed maintenance records which are typically difficult to obtain because such records are often paper based or commercially sensitive.

This section is organized as follows: Initially, redline limits are explained, and it is demonstrated why, as a method for detecting faults, these are not an optimal approach. Subsequently, the methodology is introduced, formed of three subsections:

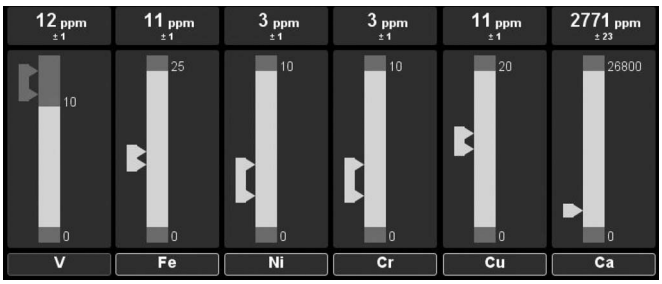

Fig. 1. Output from a redline classification. Here, V can be seen to be 2 ppm out of specification, so the whole sample is considered "abnormal" even though the remainder of the sample is considered within specification despite the confidence interval of ± 1 ppm. Taken from a fault-detection system in 2007, which was being used in a large marine-engine fault detection.

TABLE I
REDLINE LIMITS FOR A HEAVY-DUTY DIESEL ENGINE

Element	Al	Cr	Cu	Fe	Ni	Pb	Si
Concentration/ppm	15	20	40	125	50	25	30

preprocessing, detection, and diagnosis. The section is then concluded with a summary of the proposed methods and highlighted with a pseudocode.

A. Redline Limits

Redline limits are a set of maximum concentrations specified by either the engine or oil manufacturer concerning the maximum (and sometimes minimum) concentration in ppm that can be observed in oil samples, as can be seen in Fig. 1. The specifications used in this paper can be seen in Table I. These specifications are quite commonly used in industry where any diesel engine is being examined to analyze when an engine can be considered "out of specification" or running abnormally and can be thought of as the more traditional approach to fault detection. The limits used vary depending upon the engine type and the manufacturer, so they are unique to a subset of engine models.

The concept behind these rules can be reformulated as a question concerning the method of representation, reflecting both common sense and expert knowledge. For example, there is a significant amount of confidence placed upon the specification when a question such as the following is presented, which is gleaned from [6]

$$\text{IF } iron\ ppm > 125 \text{ THEN } engine\ overhaul$$

where an action is presented as *engine overhaul* which should be carried out if the condition "iron ppm" is greater than 125, referring back to Table I for the specification. This IF statement, in effect, states that at 125 ppm, the action *engine overhaul* should not be carried out; although if the iron content of the oil increases by 1 ppm, the action should be carried out. This statement is an accurate representation of one particular redline limit combined with expert knowledge (although as an exaggeration, an engine would unlikely be overhauled if it was over the specification). On the other hand, the 1 ppm increase would not reflect a large state change in the status of the engine, and yet, an action is triggered because of it. If questioning human experts upon the change, they themselves would unlikely state that such a small change would require any significant action, so it seems inappropriate to use a knowledge representation that fits the expert knowledge into such a rigid framework. On the other hand, more flexible representations are rarely requested in industrial applications as the company wants to ensure trust in the algorithm and consistency, and, above all, that they can easily read the rules and copy rules from existing algorithms or specifications; conversely, maintaining and building this type of fault tree may result in illegibility and duplication of rules. Despite the simplicity of the if–then rules, it forces the human expert to perform an extremely difficult reasoning process, explicit declaration of all faults satisfying a given set of conditions. Using top–down reasoning, the human expert is required to define poorly known relations explicitly and force a number of unknown relations into the same framework as well-known relationships. Therefore, not only does this type of representation cause problems for those experts who have to define them but also they are a poor representation of the engine.

There is a more effective approach to detecting and diagnosing problems with an engine, which is now demonstrated and evaluated. This is based upon extracting more information from the available data sets which takes the form of a preprocessing module in Section II-B followed by the interpretation of the data utilizing a classifier in Section II-C and finally, provide a suitable diagnosis in Section II-D, based upon the output from the classifier.

B. Preprocessing

Assuming that the elements are measured in ppm, there is much information present that needs to be extracted from a time series. Diagnosis and classification systems can be considered sensitive to outliers, so there is a requirement for the time series to be properly filtered before applying the data to further classification or diagnostic techniques [29], [30]. More notably, in aperiodic sampling systems whether online or offline, the frequency of samples should be normalized over a certain period. This is a more significant step in offline fault-diagnosis systems, as samples could be taken once per day, per month, or over even longer durations. For that reason, subsequent methods for processing the time series may assume a consistent duration between samples, which can be achieved using interpolation, whether linear or cubic. Here, linear is used as the cubic interpolation is seen to introduce anomalies that are not present in the original time series.

The majority of research that have used preprocessing systems, such as Pevzner *et al.* [19] or Macian *et al.* [6], [31], [32], used single values to represent the condition of an engine at a point in time. For example, [19] utilized a rate of change over two windows of time, one window covering a shorter duration and the second window a longer duration. These were then averaged to provide a scalar, to detect faults with a short and fast onset as well as those that may have a slower onset. The work in [6] normalized measured concentrations to bring the values into a smaller predefined range to allow fuzzy sets to be applied. Further work in [33] used ratios to analyze when one

element may be diverging more quickly than would normally be expected.

In the current work, it is proposed that a combination of these into a *descriptive tuple* is more appropriate, and, as a result, both the rate of change and normalized measured concentration form

$$x_t = \langle r_t^n, m_t^n \rangle \tag{1}$$

where n refers to each element, t is the current position in the time series, m is the measured concentration, and r is the calculated rate of change.

Furthermore, a heuristic is defined to only consider positive rates of change. Negative gradients are filtered out and replaced with an average value, moderately distorted by random noise. The heuristic, therefore, is more domain dependent and may not be implemented in other domains. In engine diagnosis, however, it is usual that a quicker than usual drop in elemental concentrations toward the average suggests that the engine is returning to a normal state as it may have been repaired.

This provides a 2-D vector, where one axis refers to the rate of change and the other, to the measured concentration. As such, a vector that converges toward one could be considered as abnormal and a vector that converges toward zero may be considered normal. On the other hand, it would be rare for there to be no fluctuation in the concentrations of different elements, and, as a result, it is more optimal to identify these distributions by utilizing a classifier that will take outliers into consideration.

C. Classification and Model Selection

The aim of classifying data points before applying the data into a diagnosis module is to perform a comparative statistical analysis across the available historical data for a similar engine model. In this sense, the novel data is compared with past data to enable subsequent processes to understand whether the novel data represents a sample that could be considered normal or abnormal, depending on previous experience of a component in a specific environment. This increases the flexibility of the thresholds utilized and also enables a learning characteristic where novel data are added to the distribution to shift the classification boundaries. Ideally, this will allow the method to improve over time.

To correctly characterize the distribution provided by the tuple defined for each measured elemental concentration, it is necessary to generate a model to correctly classify the majority of normal data at the same time as not overgeneralizing the model to incorrect cases. The most simple, and one that should be used initially, is the Gaussian model, which, as seen in Fig. 4, is usually appropriate for fault-diagnosis data. If utilizing the measure of, for example, the rate of change as one descriptive vector of the time series, it is fair to assume that the density can be modeled using a Gaussian. This is because a normal distribution will typically arise around zero, where there is no rate of change, primarily because there are few abnormal occurrences causing significant change in elemental concentrations in an engine, as can be seen in Fig. 4. As an example, the classifier uses the Mahalanobis distance estimator where the target class is defined as a Gaussian distribution. The classifier therefore is defined as in (2), where θ is a user-defined fraction of the training samples that should be included in the Gaussian and *normal* and *abnormal* represent qualitative labels applied to each of the two classes *target* and *outlier*, respectively, and μ refers to the mean and Σ the covariance, representing respective sample estimates. x therefore refers to the descriptive tuple.

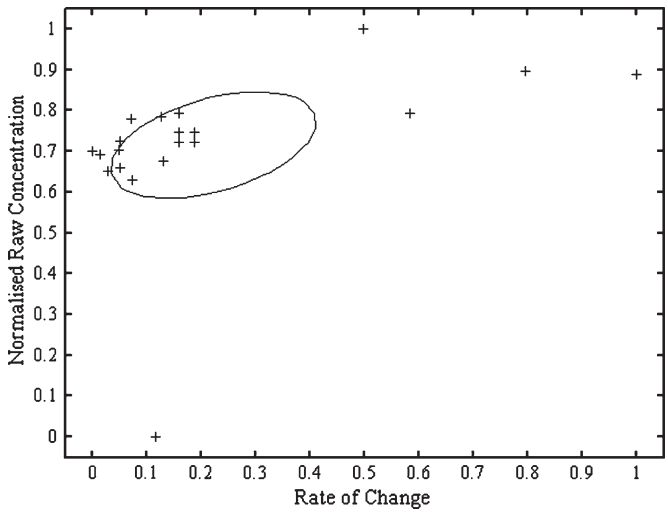

Fig. 2. Preprocessed silicon content measured from the sump of a heavy-duty diesel engine where those points inside the Gaussian are considered "normal" and those outside are "abnormal."

$$h(x) = \begin{cases} \text{normal}, & \text{if } f(x) \leq \theta \\ \text{abnormal}, & \text{if } f(x) > \theta. \end{cases} \tag{2}$$

An example of the Gaussian classifier is shown in Fig. 2. With such sparse data sets, the decision was made to use simple classifiers that will not overfit the data, as it cannot be guaranteed that the training samples taken accurately reflect normal running at other times. As can be seen, the Gaussian model is affected by outliers and so does not optimally characterize the normal data. For that reason, the computation of μ and Σ are altered. The outliers are reweighted according to their proximity to the mean where candidate outliers are down weighted so that a more optimal characterization is found, and the result can be seen in Fig. 3.

It was also decided to observe how a better fitted model would improve or reduce the accuracy of the model. Therefore, a mixture of Gaussians (MOG) was utilized which can be defined as follows. Gaussian mixture models are, as the title suggests, a mixture of a number of Gaussian distributions that are used to quantify the vector space. Such models can approximate a wide variety of probability-density functions and are consequently good solutions for situations where a singular normal distribution fails. While bearing this in mind, it is also common for the mixtures to be built out of one predefined basic distribution type, such as the Gaussian normal distribution. In multivariate analysis, the majority of inference problems are developed under the assumption of normal distribution. Therefore, if there is no specific knowledge regarding the probability density, only a general model can be used, and the Gaussian distribution is a reasonable choice.

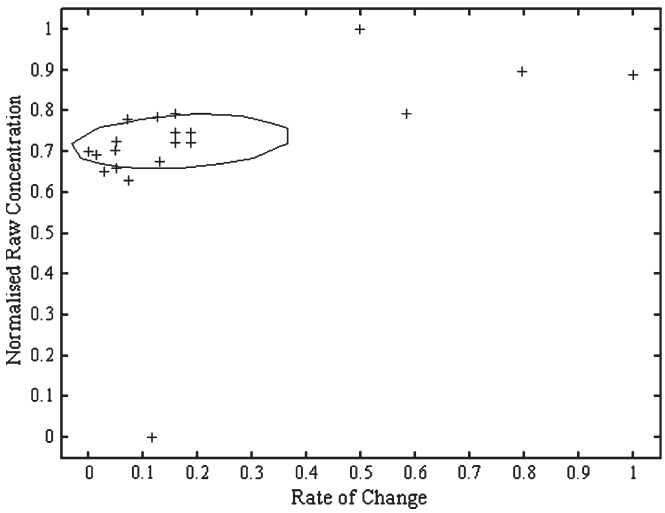

Fig. 3. Preprocessed silicon content measured from the sump of a heavy-duty diesel engine where the classifier used here is the robust Gaussian.

The algorithm works by first choosing the component (the Gaussian distribution) at random with probability $P(\omega_i)$. The algorithm subsequently samples a point in the vector space $N(\mu_i, \sigma^2 I)$. Supposing the variables $x_1, x_2, \ldots, x_N$ and $P(\omega_1), \ldots, P(\varnothing, \omega_K)$, σ, the likelihood of the sample $P(x|\omega_i, \mu_1, \mu_2, \ldots, \mu_K)$ can be obtained, while trying to maximize $P(x|\mu_1, \mu_2, \ldots, \mu_K)$ which refers to the probability of a single datum given the centers of the Gaussians μ_K and which can be defined as in

$$P(x|\mu_i) = \sum_i P(\omega_i) P(x|\omega_i, \mu_1, \mu_2, \ldots, \mu_K) \tag{3}$$

which can be expanded over all the data as in

$$P(\text{data}|\mu_i) = \prod_{i=1}^{N} \sum_i P(\omega_i) P(x|\omega_i, \mu_1, \mu_2, \ldots, \mu_K) \tag{4}$$

where the maximum likelihood should be maximized by calculating $\partial L / \partial \mu_i = 0$. This, however, is nontrivial, and as such, there are methods such as the expectation-maximization algorithm which can solve this.

Here, μ has two members or clusters to represent the two different distributions that could potentially be seen within an element (for both the x- and y-axis), a concept shown in Fig. 4, where both the rate of change and the normalized measured value are stored in a vector.

The output from both the Gaussian and MOG classifiers is binary, e.g., $h(x) \in \{0, 1\}$ which, despite adequately representing the membership of a point in time to a particular class, does not provide much information regarding the severity of a fault. To avoid the concept of thresholds, as used in redline limits, it is proposed that this should also be reflected in the classification process, where it would be naïve to present those points inside the model as 0% fault likelihood and those outside as having a 100% fault likelihood. Consequently, the classification model has been further developed to return information regarding distance from the model, as in (5), where the distance can be

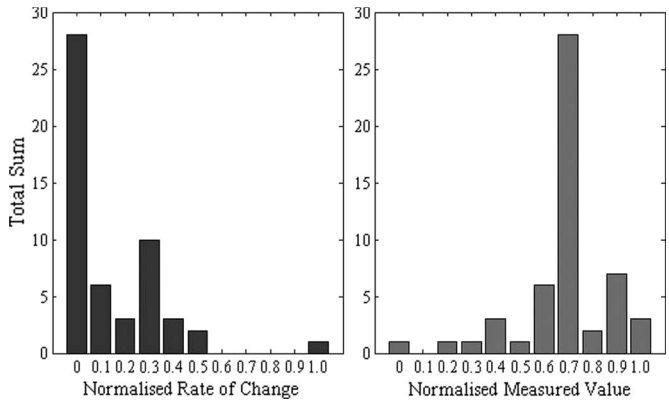

Fig. 4. Two distributions for the same element where the left bar chart shows the distribution of the rate of change over all elements, and the right chart shows the normalized measured value from the elemental concentrations.

thought of as the comparative extremum distance between the maximum and the minimum data, i.e.,

$$h(x) = \begin{cases} d(x) - \alpha, & \text{if } f(x) \le \theta \\ d(x) + \alpha, & \text{if } f(x) > \theta \end{cases}$$

$$d(x) = \text{abs}\left(\frac{e(x) / \max(\forall e(x))}{2} - \alpha \right) \tag{5}$$

where e is a distance function, such as Euclidean distance, and $\alpha = [0, 1]$ is a penalty term to determine the bias applied to values that are outside or inside the model, where the default is 0.2, and $P = [0, 1]$ is the severity of the data point x being a fault condition.

To conclude here, it can be seen that the robust Gaussian model reasonably characterizes the data accurately while not fitting it exactly. This is advantageous, as it cannot be considered that the data in the training set will exactly fit the current testing data or, in fact, will be similar to the future testing data. For that matter, it was seen that adding random noise to the training data would also improve the generated model. On the other hand, an additional noise added to the training data would have a more significant effect if the model used was an MOG, support vector machine, or a self-organizing map, as these classification techniques fit the input data much more closely. Consequently, the model that best separates the high density distribution from the lower density data is the most optimal model, assuming that future distributions also reflect this separation.

D. Accurate Diagnosis Under Uncertainty

Once there is an access to a set of floating-point distances[1], the severity of each individual element should contribute to a lesser or greater extent against the eventual diagnosis. It is the varying degrees with which each element reflects a particular diagnosis as suggested in [4] which is significant, and hence suggests a hierarchical graph where nodes represent the relationships between four different types of situation; the

[1]Where these values represent the distance the element at a given time t lies from the statistical expected norm.

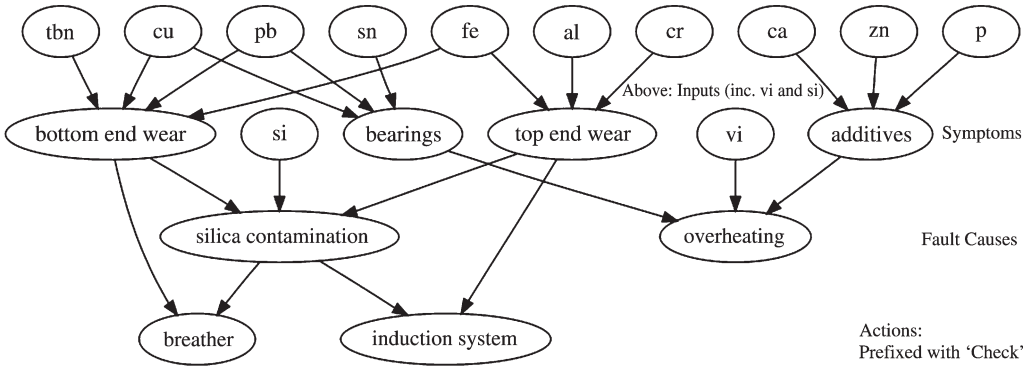

Fig. 5. Belief network defined for a subset of inputs, symptoms, fault conditions, and possible actions in a diesel engine, where the binary nodes are connected with probabilistic relationships determining the strength of association.

observations, the symptoms, the actions, and the tasks that should be implemented by the user. The observations consist of the output from the classifier, a single scalar representing the distance of the tuple from the statistical norm provided by historical data. As opposed to using a fixed derivative to represent the interactions between the elements, a more flexible method is to represent these interactions as a probabilistic relationship conjoining the different states, which could be altered depending upon the feedback from the user. That is to say, the accuracy of the diagnosis could be improved with exposure to new instances over time.

The so-called belief network, as shown in Fig. 5, is a graphical representation of an underlying probabilistic relationship. To most accurately reflect these circumstances, the selection of parameters, as mentioned in [21], is an important consideration which can be considered useful for reasoning under both ignorance and uncertainty. On the other hand, the main disadvantage of the belief network is the lack of conviction in results; for example, if a user is provided with many possible faults with similar percentages, which fault is the most likely, and is a maximum-likelihood solution the optimal method to select for the correct diagnosis.

The different layers of the network can be observed in the following.

1) **Input nodes** are shown in Fig. 5 as those nodes toward the top of the belief network, typically those with two or three letters, and represent the elemental inputs. These are the percentages obtained from the classifier.
2) **Output: Symptom nodes** are those nodes entitled with "symptoms." These are not the cause of the problem; they are the resulting outcome, such as "top-end wear," "bottom-end wear," "bearings" (referring to bearing wear), and "additives."
3) **Output: Fault nodes** are nodes entitled "silica contamination" and "overheating." These refer to the underlying problem.
4) **Output: Action nodes** are those nodes entitled with words toward the bottom of the figure, such as "breather" and "induction system." These are possible actions that can be taken in the case of a fault node being triggered. If these action nodes are triggered, textual descriptions can be associated with the node to suggest a probable next-step action, e.g., "Silica contamination is high (90%) as is top-end wear (80%); it is suggested that the next-step action should be to check the air induction system (70%)."

It is hoped that a more complete version of Fig. 5 would be of use for future work in developing algorithms that reflect expert knowledge.

E. Core Functions

Taking the previous literature into consideration, the methodology can be summarized here. An overview of the approach is presented in Algorithms 1, 2, 3, and 4. These can be thought of as functions or modules, where the algorithms presented in 2 and 3 are the core modules that are called. Initially, the function INIT-FAULT-PROG() is called, which generates a set of models G for each of the elements in the training matrix E. Subsequently, the function NEW-INSTANCE-FAULT-PROG() is called with a testing set of elements in the matrix E, although containing a subset of data unseen by the training algorithm. This uses the trained models from INIT-FAULT-PROG() to calculate distance scores between the model g_i and the datapoints in X, whereupon, depending on whether the value is inside or outside the generated model, where 0 refers to the contour, the value assigned to the belief network is penalized or given a bonus with the value of α.

Algorithm 1 Function CALL-FAULT-PROG

Ensure: E^0 the training data, for example, 40% of the entire data set.

Ensure: E^1 the testing data, for example, 60% of the data set.

1: $G \leftarrow$ INIT-FAULT-PROG(E^0)
2: $D \leftarrow$ NEW-INSTANCE-PROG(E^1, G)
3: **for all** t such that $1 \leq t \leq$ SIZE(D) **do**
4: **for all** l such that $1 \leq l \leq$ SIZE(D_t) **do**
5: **if** $D_{t,l} \neq NULL$ **then**
6: **print:** $D_{t,l}$
7: **end if**
8: **end for**
9: **end for**

Algorithm 2 Function INIT-FAULT-PROG
Ensure: $e_{t,i} \in E$
Ensure: $g_i \in G$
1: $X \leftarrow$ GENERATE-TUPLE (E)
2: **for all** i such that $1 \leq i \leq$ SIZE(ELEMENTS) **do**
3: $\mu \leftarrow \text{MEAN}(X_i), \Sigma \leftarrow \text{COV}(X_i)$
4: $g_i \leftarrow \text{GAUSSIAN}(X_i, \mu, \Sigma)$
5: **end for**
6: **return:** G

Algorithm 3 Function NEW-INSTANCE-FAULT-PROG
Ensure: $e_{t,i} \in E$
Ensure: $g_i \in G$
Ensure: $b_{1,i} \in Y$ the input nodes (BBN)
Ensure: $b_{l,i} \in Y$ the nodes (BBN)
Ensure: Y^F is the MAP fault node for a sample t
Ensure: $\alpha \in [0, 1]$
Ensure: $\theta \in [0, 1]$
Ensure: $d_{t,l} \in D$ a list of diagnoses
1: $X \leftarrow$ GENERATE-TUPLE(E)
2: **for all** t such that $1 \leq t \leq$ SIZE(E) **do**
3: **for all** i such that $1 \leq i \leq$ SIZE(ELEMENTS) **do**
4: $h(x) \leftarrow (\text{MAHAL}(g_i, X_{t,i})/\max(\text{MAHAL}(g_i, X_i))/2)$
5: **if** $h(x) > 0$ **then**
6: $b_{1,i} \leftarrow h(x) + \alpha$
7: **else**
8: $b_{1,i} \leftarrow h(x) - \alpha$
9: **end if**
10: **end for**
11: **for all** l such that $2 \leq l \leq$ SIZE(NODELAYERS) **do**
12: $Y^F \leftarrow \arg\max_{b_{l,k} \in Y} \prod_{k=1} P(b_{l,k} | Parents(b_{l,k}))$
13: **if** $Y^F > \theta$ **then**
14: $d_{t,l} \leftarrow Y^F$
15: **else**
16: $D_{t,l} \leftarrow NULL$
17: **end if**
18: **end for**
19: **end for**
20: **return:** D

Algorithm 4 Function GENERATE-TUPLE
Ensure: $e_{t,i} \in E$
Ensure: $X \leftarrow$ ZEROS(SIZE(E))
1: **for all** i such that $1 \leq i \leq$ SIZE(ELEMENTS) **do**
2: **for all** t such that $1 \leq t \leq$ SIZE(E) **do**
3: $m \leftarrow (e_{t,i} - \min_i(E_i) / \max_i(E_i) - \min_i(E_i))$
4: **end for**
5: **for all** t such that $2 \leq t \leq n$ **do**
6: $r \leftarrow e_{t,i} - e_{t-1,i}$
7: **end for**
8: $X_{t,i} = \langle m, r \rangle$
9: **end for**
10: **return:** X

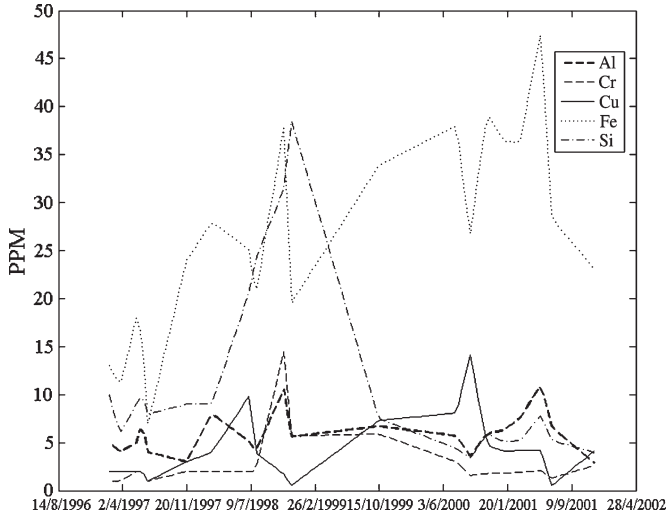

Fig. 6. Five primary elements used in the diagnosis of silica contamination for data set *engine1*.

Subsequently, these distances are assigned to the input nodes Y of the belief network. It is then possible to query the set of output nodes Y for joint probabilities given the value of the parents Y and from which the node with the highest maximum likelihood is used. If it is over a threshold θ, governing sensitivity, it is returned as a fault node; otherwise, it is ignored and assumed to be normal.

The anticipated usage could take the form shown in Algorithm 1 depicting the calling routine CALL-FAULT-PROG(), where E would take the form shown in

$$E = \begin{bmatrix} e_{1,1} & \cdots & e_{t,1} \\ \vdots & \ddots & \vdots \\ e_{1,i} & \cdots & e_{t,i} \end{bmatrix}. \tag{6}$$

III. CASE STUDY: DIAGNOSIS OF HEAVY-DUTY DIESEL ENGINES

The methodology described in Section II was then applied to three data sets from heavy-duty diesel engines of the same model, where *engine1* can be seen in Fig. 6, *engine2* in Fig. 7, and *engine3* in Fig. 8, where both the MOG and a Gaussian were used as classifiers from Section II-C and the output from the Gaussian classifier was used as an input into the belief network described in Section II-D.

There are two aims addressed in this section: the accuracy of the proposed algorithm when compared against the redline limits in Section III-A as well as the period of advance warning that can be given of a potential fault, which again forms a comparison between the proposed algorithm and redline limits, otherwise referred to as sensitivity in Section III-B.

The first aim is addressed using a receiver operating characteristic (ROC) curve as well as a diagnostic confusion matrix (DCM). The ROC curve identifies the accuracy of the proposed algorithm, where the true positive rate (TPR) is compared with the false positive rate (FPR) for the proposed algorithm and compared against the redline limits. The DCM then highlights where a true positive was found, and whether the true positive

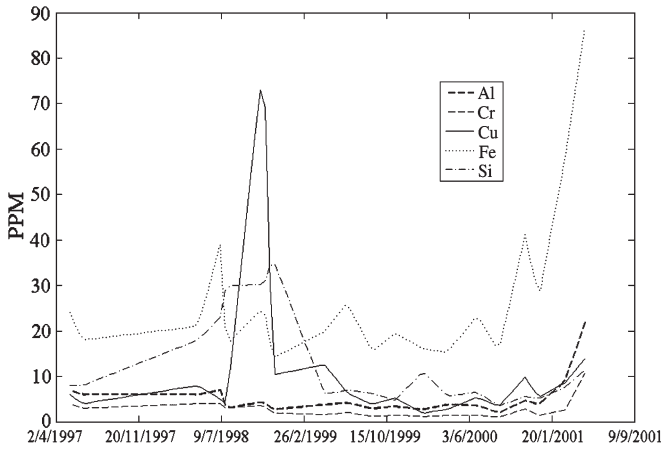

Fig. 7. Five primary elements used in the diagnosis of silica contamination for data set *engine2*.

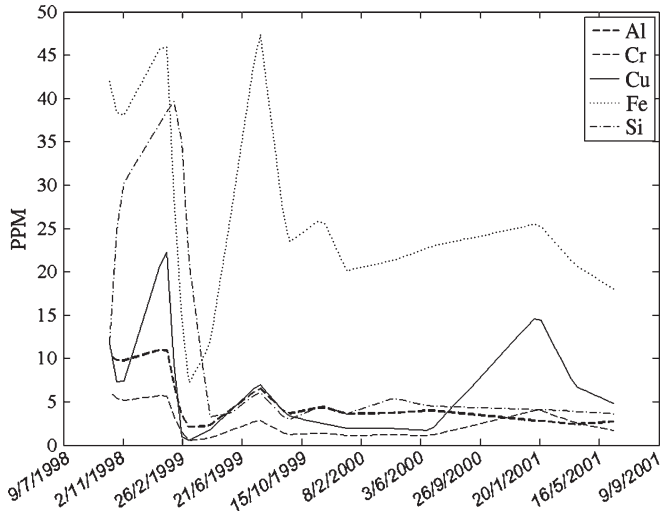

Fig. 8. Five primary elements used in the diagnosis of silica contamination for data set *engine3*.

correctly diagnosed the fault or whether an incorrect fault was identified.

Subsequently, in Section III-B, four significant peaks were located from the three time series, and the sensitivity of the proposed algorithm was compared against a complex derivative of the redline limits related to the known fault identified as being silicon contamination. Consequently, the two metrics that were addressed consisted of accuracy in Section III-A and sensitivity in Section III-B.

A. Accuracy

Accuracy is defined as a tradeoff between false negatives, false positives, true negatives, and true positives, which can be quantified in part by using an ROC curve which characterizes the better classification ability of both the proposed algorithm as well as the conventional redline limits. An ROC curve can be defined by FPR and TPR as x- and y-axis, respectively, which depicts relative tradeoffs between true positive (benefits) and false positives (costs) where a point in the top left corner of

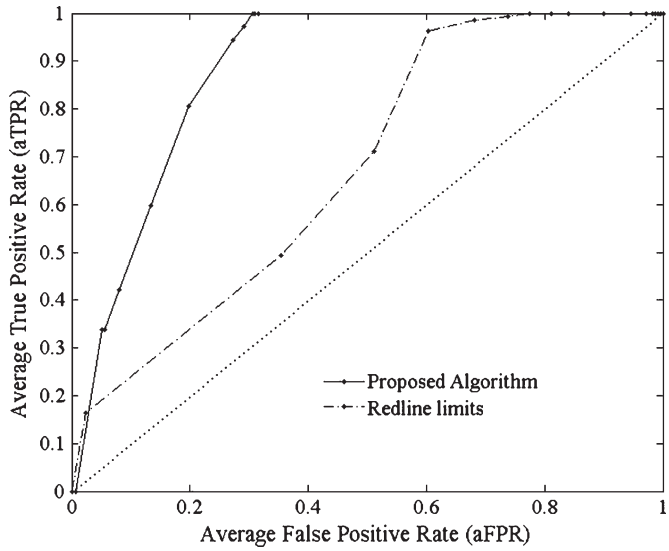

Fig. 9. ROC curve showing a comparison between the proposed algorithm and redline limits, where the diagonal dotted line represents "chance." A data point in the upper left-hand corner signifies a better tradeoff between true positives (benefits) and false positives (costs).

the space can be considered optimal, as more true positives are preferred.

The accuracy of the system was ascertained based upon a ground truth which was obtained by a manual diagnosis of the time series. For the proposed algorithm, the TPR and FPR were established against all possible faults and the accurate or inaccurate detection of these faults. Meanwhile, redline limits were established for a subset of elements related to silicon contamination, and in the event that the measured PPM exceeded the redline limit, a fault was flagged; consequently, the testing for the proposed algorithm was more rigorous than for the redline limits. This evaluation can be observed in Fig. 9, showing that the proposed algorithm is more optimal than the redline limits in detecting faults than the conventional method. Here, the mean value over the three data sets was taken.

To complement the ROC curve, the DCM defines the fault conditions that were correctly (or incorrectly) classified as shown in Fig. 10. The highest values (representing the highest frequency of classification for a particular fault) should be on the diagonal, where the x-axis represents the fault conditions identified in the ground truth and the y-axis represents the type of faults that the respective fault conditions were identified as. The same process was repeated for the data sets *engine2* and *engine3*. Results of these can be seen in Figs. 11 and 12. These figures utilized the Gaussian classifier and demonstrates very little inaccuracy in correctly identifying faults. Where faults were identified using the MOG classifier, these were all identified correctly.

B. Sensitivity

Sensitivity can be defined as the amount of prior warning that the algorithm gives for a verified fault condition. This will typically differ depending upon the type of system, for example, an algorithm based upon an online system (where samples are taken every second), the duration of the warning may be an

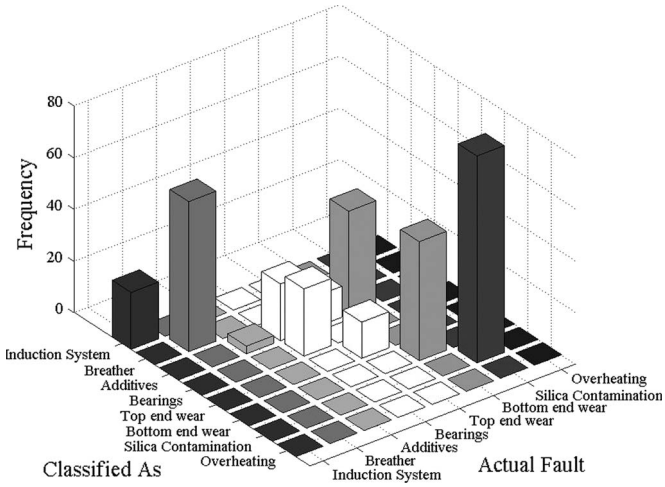

Fig. 10. DCM for fault-detection evaluation for *engine1* using a Gaussian classifier.

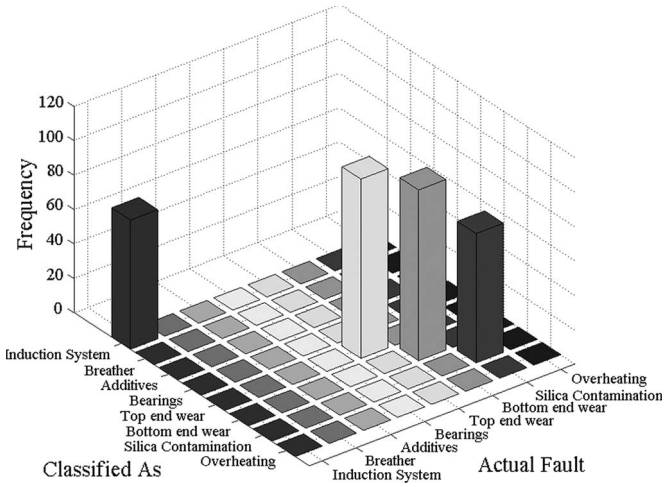

Fig. 11. DCM for fault-detection evaluation for *engine2* using a Gaussian classifier.

hour, or even just a few minutes or a few seconds. However, this duration of warning allows a prognosis to take place and any problems to be fixed before the fault results in a failure or downtime. Any period of time before a fault, therefore, is preferable. In offline systems, because of the fewer samples that are taken, failure can potentially be detected much further in advance with the caveat that imminent onset failures cannot be detected. Silica contamination would usually have a longer period of onset, where a gradual build up may not be noticed until too much wear has taken place; this will continue to build until the source of the contamination has been found. Therefore, the largest period of warning is preferable.

To clarify how these results were obtained, by selecting the data set *engine2* as in Fig. 7, the two most significant faults were identified on October 26, 1998 and again on April 23, 2001. Although there is no information to suggest that these points were at the highest severity of the fault (the fault may have occurred earlier, and the concentrations observed are byproducts or related to another fault), the sum of a complex derivative was taken where the complex derivative consisted of those elements related to silica contamination. Therefore, the

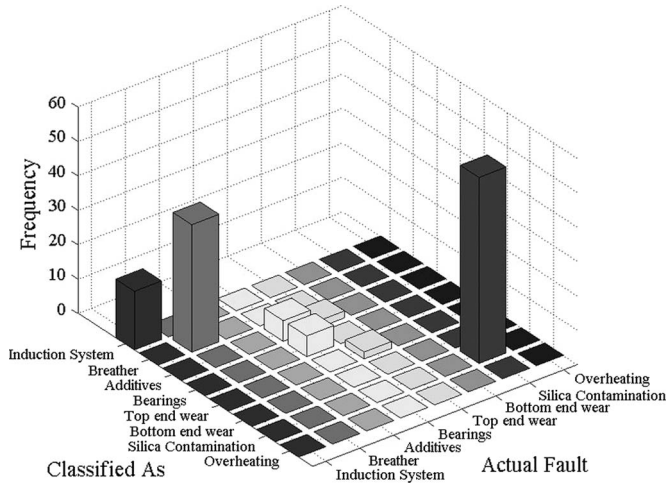

Fig. 12. DCM for fault-detection evaluation for *engine3* using a Gaussian classifier.

sum of these elements were highest at these two points. There were smaller peaks at other times in the data set; however, these did not result in a significant amount of wear material being measured, possibly because the problem had been fixed before it became too severe or that there were small changes in the engine's environment.

Therefore, the proposed method using both varieties of classification described in Section II-D and the same redline limits (forming a complex derivative related to silicon contamination) were queried to identify at what point prior to the fault were the sequential faults diagnosed.

That is to state that if there was a break in the chain of diagnosed faults, the faults before the break would not be considered related to the specific peak. The results can be seen in Fig. 13. The fault chains have been seen to start on February 2, 1998 and July 3, 2000 for peaks 2 and 3. The percentages have been calculated by observing the ideal start and the actual start of the fault chains in months for both systems and then taking these away from 100; therefore, the higher the values, the more sensitive the algorithm; in peak 2, the time between the start of the fault chain and the final peak was a period of nine months, and in peak 3, the duration was ten months. This process was repeated for the other two data sets, with one peak each for the remainder. As can be seen in Fig. 13, the MOG has a lower average than that of the redline limits, which can be explained by referring to Fig. 8. The peak's onset is at the beginning of the data set, and hence, the number of days' difference between the three algorithms is small, as the amount of advance warning that could be given is very little compared with the other peaks in the other data sets, where the peaks typically appear toward the end of the time series.

IV. Conclusion and Future Work

To summarize, this paper has identified a method of preprocessing the input data using a descriptive tuple in Section II-B to represent the salient aspects of the input data, which have subsequently been classified using a one-class classifier and a novel distance calculation to separate normal

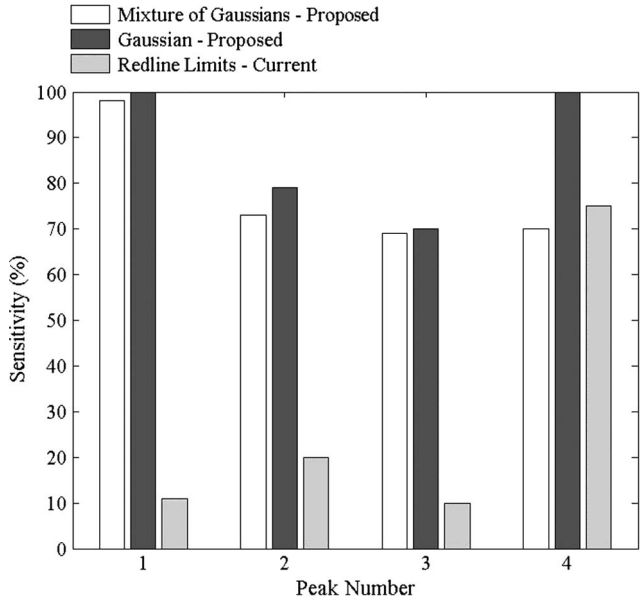

Fig. 13. Sensitivity of the framework applied to all the data sets in comparison with redline limits, expressed as a percentage. Peak 1 originates from *engine1*, peaks 2 and 3 originate from *engine2*, and peak 4 originates from *engine3*.

and abnormal data in Section II-C. This was designed to detect possible faults significantly earlier than the redline limits, as can be seen in Fig. 13, in considering the metric of "sensitivity" which is primarily because the rates of changes were characterized by historical data from the same engine. Furthermore, the method removes the concept of a given threshold, allowing potential problems to be watched as they gradually become more problematic. In most cases, it can be observed that the problem was detected as soon as the trend moved away from expected levels.

The output from the classification algorithm was diagnosed using a belief network with a specifically designed structure described in Section II-D. In Section III, a case study of three train drives was introduced to highlight the comparison between the redline limits and the proposed methods, assessed for both sensitivity in Section III-B and accuracy in Section III-A. The strengths of the proposed methods were seen in both identifying possible faults prior to redline limits, seen in Fig. 13, as well as in diagnosing faults more accurately, which can be seen in Fig. 9.

These results demonstrate a much higher level of accuracy and sensitivity than redline limits and perform well on correctly detecting and diagnosing the four large faults at an incipient level, in line with the ground truth and, as such, reflect the opinions of human experts who provided the ground truth.

There are a number of improvements that can be addressed in future work. The approach can be compared with other fault-diagnosis frameworks or more traditional artificial intelligence methods such as neural networks. Furthermore, it would be interesting to apply the method to other engines to observe the generalization ability of the model to classify faults. It is suspected that it would generalize well with other situations, ensuring that the expert knowledge is altered to the specific domain. Using the threshold θ is not appropriate, and as such, it may be possible to define a "No-Fault" node, where if this node was selected as the highest likelihood, a fault would not be reported, conforming to the principle of MAP.

To conclude, the proposed method is a modular approach to the concept of fault detection and diagnosis, incorporating a number of new developments from different domains which show promising results for further work and demonstrating significant improvements over the industry standard, the redline limits, which are commonly used in fault-detection algorithms. In some instances, a threshold is required by industry; however, this paper has demonstrated the lack of warning that such an approach provides. The amount of data available is sufficient to base a detection and diagnosis system and successfully improve upon older techniques and would only improve with application of larger data sets. It is hoped that the methods developed in this paper will assist in making the maintenance of complex mechanisms more efficient.

REFERENCES

[1] X. Wang, U. Kruger, G. Irwin, G. McCullough, and N. McDowell, "Nonlinear PCA with the local approach for diesel engine fault detection and diagnosis," *IEEE Trans. Control Syst. Technol.*, vol. 16, no. 1, pp. 122–129, Jan. 2008.

[2] C. Tan, P. Irving, and D. Mba, "A comparative experimental study on the diagnostic and prognostic capabilities of acoustics emission, vibration and spectrometric oil analysis for spur gears," *Mech. Syst. Signal Process.*, vol. 21, no. 1, pp. 208–233, Jan. 2007.

[3] R. Douglas, J. Steel, and R. Reuben, "A study of the tribological behaviour of piston ring/cylinder liner interaction in diesel engines using acoustic emission," *Tribol. Int.*, vol. 39, no. 12, pp. 1634–1642, 2006.

[4] I. Morgan and H. Liu, "Predicting future states with n-dimensional Markov chains for fault diagnosis," *IEEE Trans. Ind. Electron.*, vol. 56, no. 5, pp. 1774–1781, May 2009.

[5] B. Tormos, V. Macian, P. Olmeda, and L. Montoro, "Wear rate determination for IC engine condition monitoring: Results obtained in an urban transport fleet," *SAE Int.*, vol. 113, no. 3, pp. 1308–1314, 2004.

[6] V. Macian, B. Tormos, A. Sala, and J. Ramirez, "Fuzzy logic-based expert system for diesel engine oil analysis diagnosis," *Insight*, vol. 48, no. 2, pp. 1–8, Aug. 2006.

[7] Y. Liu, Z. Liu, Y. Xie, and Z. Yao, "Research on an on-line wear condition monitoring system for marine diesel engine," *Tribol. Int.*, vol. 33, no. 12, pp. 829–835, Dec. 2000.

[8] J. Dragsted and O. Bergeson, "Influence of low cylinder consumption on operating cost for 2-stroke engines," in *Proc. Int. Council Combust. Engines, CIMAC Congr.*, Kyoto, Japan, 2004, vol. 9.

[9] M. Carnero, "Selection of diagnostic techniques and instrumentation in a predictive maintenance program: A case study," *Decis. Support Syst.*, vol. 38, no. 4, pp. 539–555, Jan. 2005.

[10] K. Yu, "An intelligent fault diagnostic tool for marine diesel engine system by neural network with fuzzy modification," *Marine Res. J.*, vol. 10, pp. 91–102, 2001.

[11] V. Palade, R. Patton, F. Uppal, J. Quevedo, and S. Daley, "Fault diagnosis of an industrial gas turbine using neuro-fuzzy methods," in *Proc. 15th IFAC World Congr.*, 2002, pp. 2477–2482.

[12] J. Jung, J. Lee, and B. Kwon, "Online diagnosis of induction motors using MCSA," *IEEE Trans. Ind. Electron.*, vol. 53, no. 6, pp. 1842–1852, Dec. 2006.

[13] H. Hong and M. Liang, "A fractional calculus technique for on-line detection of oil debris," *Meas. Sci. Technol.*, vol. 19, no. 5, pp. 1–14, May 2008.

[14] X. Fan, M. Liang, and T. Yeap, "A joint time-invariant wavelet transform and kurtosis approach to the improvement of in-line oil debris sensor capability," *Smart Mater. Struct.*, vol. 18, no. 8, pp. 1–13, Aug. 2009.

[15] M. Chow, "Special section on motor fault detection and diagnosis," *IEEE Trans. Ind. Electron.*, vol. 47, no. 5, pp. 982–983, Oct. 2000.

[16] H. Zhang, Z. Li, and Z. Chen, "Application of grey modeling method to fitting and forecasting wear trend of marine diesel engines," *Tribol. Int.*, vol. 36, no. 10, pp. 753–756, Oct. 2003.

[17] Q. Butt and A. Bhatti, "Estimation of gasoline-engine parameters using higher order sliding mode," *IEEE Trans. Ind. Electron.*, vol. 55, no. 11, pp. 3891–3898, Nov. 2008.

[18] K. Veluvolu and Y. Soh, "Discrete-time sliding-mode state and unknown input estimations for nonlinear systems," *IEEE Trans. Ind. Electron.*, vol. 56, no. 9, pp. 3443–3452, Sep. 2009.
[19] L. Pevzner, G. Rozenburg, and V. Spirova, "Diagnostic interpretation of results from analyses of used motor oils," *Chem. Technol. Fuels Oils*, vol. 30, no. 5, pp. 235–239, May 1994.
[20] V. Macian, B. Tormos, P. Olmeda, and L. Montoro, "Analytical approach to wear rate determination for internal combustion engine condition monitoring based on oil analysis," *Tribol. Int.*, vol. 36, no. 10, pp. 771–776, Oct. 2003.
[21] T. Mast, A. Reed, S. Yurkovich, M. Ashby, and S. Adibhatla, "Bayesian belief networks for fault identification in aircraft gas turbine engines," in *Proc. Int. Conf. Control Appl.*, 1999, pp. 39–44.
[22] S. Russell and P. Norvig, *Artificial Intelligence: A Modern Approach*. Englewood Cliffs, NJ: Prentice-Hall, 2002.
[23] M. Khan and M. Rahman, "Development and implementation of a novel fault diagnostic and protection technique for IPM motor drives," *IEEE Trans. Ind. Electron.*, vol. 56, no. 1, pp. 85–92, Jan. 2009.
[24] I. Morgan, H. Liu, G. Turnbull, and D. Brown, "Predictive unsupervised organisation in marine engine fault detection," in *Proc. IEEE World Congr. Comput. Intell.*, 2008, pp. 249–256.
[25] S. Huang, K. Tan, and T. Lee, "Automated fault detection and diagnosis in mechanical systems," *IEEE Trans. Syst., Man, Cybern. C, Appl. Rev.*, vol. 37, no. 6, pp. 1360–1364, Nov. 2007.
[26] E. Liu and D. Zhang, "Diagnosis of component failures in the space shuttle main engines using Bayesian belief networks: A feasibility study," in *Proc. 14th IEEE Int. Conf. Tools Artif. Intell.*, 2002, pp. 181–188.
[27] F. Sahin, M. Yavuz, Z. Arnavut, and O. Uluyol, "Fault diagnosis for airplane engines using Bayesian networks and distributed particle swarm optimization," *Parallel Comput.*, vol. 33, no. 2, pp. 124–143, Mar. 2007.
[28] H. Mingjiang, W. Zhong, Y. Yinnan, and Q. Liqiao, "On-line sensor diagnosis of the diesel engine cold starting based on RBFNN," in *Proc. IEEE Circuits Syst. Int. Conf. Test. Diagnosis*, 2009, pp. 1–4.
[29] X. Li, K. Tso, X. Guan, and Q. Huang, "Improving automatic detection of defects in castings by applying wavelet technique," *IEEE Trans. Ind. Electron.*, vol. 53, no. 6, pp. 1927–1934, Dec. 2006.
[30] X. Li and X. Yao, "Multi-scale statistical process monitoring in machining," *IEEE Trans. Ind. Electron.*, vol. 52, no. 3, pp. 924–927, Jun. 2005.
[31] V. Macian, B. Tormos, and M. Lerma, "Evaluation of metallic elements in oil for an engine fault-diagnosis system," *Tribotest J.*, vol. 8, pp. 163–176, 2001.
[32] A. Kusiak and S. Shah, "Data-mining-based system for prediction of water chemistry faults," *IEEE Trans. Ind. Electron.*, vol. 53, no. 2, pp. 593–603, Apr. 2006.
[33] K. Qi, S. Zhang, and X. Zhang, "Element correlation analysis and its application on condition monitoring and fault diagnosis of diesel engines," in *Proc. CIMAC Congr.*, Hamburg, Germany, 2001, pp. 1165–1171.

Ian Morgan (S'07–M'09) received the B.Sc. degree in artificial intelligence and psychology from the University of Birmingham, Birmingham, U.K., in 2005 and the Ph.D. degree from the University of Portsmouth, Portsmouth, U.K., in 2009, where his Ph.D. candidacy was sponsored by the Engineering and Physical Sciences Research Council at the Institute of Industrial Research.

He is currently with the Intelligent Systems and Robotics Group, University of Portsmouth. His research interests are in fault detection and diagnosis, and he is also interested in artificial intelligence techniques to fully or partially automate such tasks.

Dr. Morgan is a member of the IEEE Computational Intelligence Society.

Honghai Liu (M'02–SM'06) received the Ph.D. degree in robotics from King's College, University of London, London, U.K., in 2003.

He previously held research appointments with the University of London and University of Aberdeen, U.K., and Project Leader appointments with the industrial control and system integration industries. He has been with the University of Portsmouth, Portsmouth, U.K., since September 2005. He has published over 100 research papers. His research interests are computational intelligence methods and applications with a focus on those approaches which could make contributions to the intelligent connection of perception to action.

Dr. Liu is the recipient of three Best Paper Awards.

Bernardo Tormos received the B.S. and Ph.D. degrees in mechanical engineering from Polytechnic University of Valencia, Valencia, Spain, in 1995 and 2002, respectively.

He has been with the Thermal Engines Department, Polytechnic University of Valencia, since 1999. Over the last ten years, he has participated in four projects funded by the European Commission, 12 competitive projects funded by national administrations, and more than 35 R&D projects funded by the industry. He has published 17 articles in international journals and more than 30 papers at scientific conferences. His research topics are focused in fuels and lubricants and mainly on alternative fuels for diesel engines (biodiesel) and engine-condition diagnosis based on lubrication-oil analysis.

Dr. Tormos is a member of several scientific commissions and has also been on the board of some national and international technical associations.

Antonio Sala (M'03) received the B.Eng. degree (honors) in combined engineering from Coventry University, Coventry, U.K., in 1990, the M.Sc. degree in electrical engineering, and the Ph.D. degree in control from Valencia Technical University (UPV), Valencia, Spain, in 1993 and 1998, respectively.

Since 1993, he has been with the Systems and Control Engineering Department, UPV, where he is currently a tenured full Professor, teaching in a wide range of subjects in areas such as linear systems theory and multivariable robust control and has supervised four Ph.D. theses. He has taken part in research and mobility projects funded by local industries, government, and the European community. He is the coauthor of 23 papers in middle- or top-impact journals, eight invited talks in Spanish and international conferences, and the book *Multivariable Feedback Control* (Springer, 2004), and coeditor of *Iterative Identification and Control* (Springer, 2002). His current research interests include fuzzy control and fault diagnosis, system identification, networked control systems, and process control applications.

Dr. Sala was awarded the Second Spanish National Graduation Prize in 1993.