Fault Diagnosis using a Combined Model and Data Based Approach: Application to a Water Cooling Machine

Sergio Galve[1], Xavier Vilajosana[1] and Vicenç Puig[2]

Abstract— **In this paper, the problem of fault diagnosis of an industrial water cooling system is addressed using a combined data-driven and model based approach. Using the energy balance equations, the design of the fault diagnosis system is based on structural analysis. As result of this analysis, a set of Minimally Structurally Overdetermine Sets (MSOs) are obtained presenting the desired fault detectability and isolability properties. Since the mathematical expressions of such MSOs are very complex and highly non-linear, and there are an important number of parameters that should be estimated from data, a system identification approach based on machine learning techniques is used. Not only the nominal model but also the error model is estimated. Finally, the proposed approach is tested with the data obtained from a water cooling machine.**

I. Introduction

Maintenance is a fundamental part of industry. Nowadays, in the context of Industry 4.0 revolution, there is an industrial growing interest in moving from corrective to preventive maintenance. All the predictive strategies can be seen as an extension of the fault diagnosis towards prognosis systems [1]. In a sector where a fault leading to stop the production can have a critical economic impact in a company, being able to anticipate the detection of the fault is an important advantage. This problem has been worked for a long time as an extension of the fault detection [2] considering incipient faults. Nowadays, different fault detection and isolation approaches exist and are well established. However, fault diagnosis still is a very active field of research because of the introduction of new ways of gathering and processing data. Fault diagnosis problem has been widely studied from both the automatic control and artificial intelligence communities [3] using model based approaches that require a detailed knowledge of the system dynamics including the modelling uncertainty. By means of analytical redundancies obtained from the system, it is possible to generate a set of fault indicators by comparing the model predictions with the measured values (residuals). When the model does not match the current measures, a fault is indicated. But, to avoid false alarms is necessary to set boundaries to the modelling error and include them in the fault diagnosis process leading to the robust approaches. Although the acceptance and success of model based methods, in industrial applications there is an important effort to build the system model and fit it to the real system that in many cases can not be afforded. To model a certain machine, it is necessary to know the physical equations with its parameters. Moreover, in many cases, the development of such machines is continuously evolving appearing new series or changes in components. As a result, not only obtaining the model becomes a highly expensive process for complex machines, but also it needs to take into account every change performed. In data-driven approaches for fault detection, the objective is usually to find statistically the features that identify the faults, distinguishing therefore each strange behaviour [4]. In the same way, these methods can model the whole system and detect the deviations, failing to identify and isolate the error that is occurring without a fault database. Here lays the problem, obtaining a database of faults that is big enough and that can truly represent them is unfeasible for complex devices. The defects of both methods suppose a stopper for industries with limited resources and where their products are complex devices (many components and many types of fault). What is also relevant is that both, problems and advantages of these approaches, are complementary. This paper seeks to develop a combined method (like the work early proposed by [5]–[7]) that could use the tools from each approach (Machine Learning (ML) and Control Theory) to solve the practical problems of implementation presented above. The result is a methodology to merge both approaches, using data analysis as a tool to guide the model selection through Structural Analysis (SA). To test it, a practical scenario is presented where the data from a refrigeration system is analysed and a fault detection system is developed.

The paper is structured in the following way: Section 2 presents the methodology and its motivation. Section 3 describes the case study. Finally, Section 4 presents the results obtained and Section 5 the conclusions and future work.

II. Methodology: Motivation and Overview

The objective of the proposed methodology is to offer a feasible solution from the practical point of view. Therefore, the constrains imposed by productions environments take a great part in shaping the methodology. The considered application scenario would imply a complex device where to apply the method, which we define through the following considerations:

[1]S. Galve and X. Vilajosana are with Wireless Networks Research Lab, Universitat Oberta de Catalunya, 08860 Castelldefels (Barcelona), Spain (sergio.galve,xavi.vilajosana)@uoc.edu

[2]V. Puig is with the Institut de Robòtica i Informàtica Industrial UPC-CSIC, Barcelona, Spain vicenc.puig@upc.edu

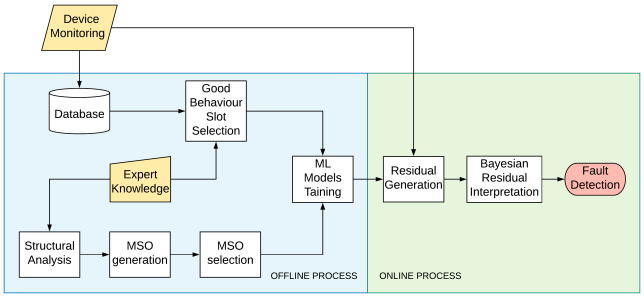

Fig. 1. Diagram describing the sequence of steps that compose the model-data combined methodology.

- Unknown Control Scheme: control schemes in industrial applications can be developed by third parties and it might be harder to access to their description.
- Several parts: it is frequent to find that industrial devices are equipped with many different parts from different providers.
- Complex dynamics: most of the industrial equipment can be modelled using classical mechanics or thermodynamics that lead to complex models involving many coefficients (of performance, drag, charge loss, etc.) that sould be indentified from real data.
- High rotation of components: considering that these complex devices usually have high costs, the personalization level demanded by customers sets scalability as a key requirement of any fault detection solution.

With these limiting factors, the method will have as objective to provide scalability, fault detection system performance (low false alarm rate while fast responses), simple interaction with expert knowledge and fast adaptation to changes. These requirements are presented as a guide to develop the methodology to be implemented as part of a production scheme.

In Fig. 1, the sequence of stages that composes the methodology is presented. The process is divided in two parts, the initial work to set up the fault detection system structure (offline) and the interpretation of the online measures to carry out the diagnosis.

A. *Offline construction*

Initially the objective is to select sets of measured variables from where models can be built. Each variable will carry information about the system, that in normal conditions should fit with the physics modelling this scenario. If an estimator can link one of them to the others (given that there is information enough in the input), the predicted value could be compared to the measurement of that variable.

The method presented needs to provide information about the detected fault, so the expert service team could interpret it and act in consequence. One way to extract more information will be to group the variables according to that expert knowledge, then build the models that can make estimations.

1) Variable Selection: The first step in the offline work would be to define those groups of variables. In any machine composed by different components interconnected it would be possible to identify and link the inputs and outputs with, for example, energy balance equations. The expert knowledge of the engineering team could define these relations without knowing the exact parameters or relations. Since each of these equations is defined by the expert knowledge, the faults to be detected can also be associated with these equations.

Once it is possible to obtain that set of approximated equations, SA is applied using the tools described by [8]. The variables are related to each other through equations, these type of relation can be seen as a bipartite graph (equations and variables). The tools of the SA allow to define the biggest overdetermined set of equations, from where subsets of Minimally Structurally Overdetermined sets (MSO) can be obtained.

To find the most suitable set, an optimization can be proposed by selecting in first instance a criteria that can evaluate the better candidates. In this case the metric to compare two MSOs will be the faults they include and the measured variables they would use. The first step is necessary to evaluate which faults could identify a potential MSO set. On the other hand, the second step is necessary to evaluate that the best suitable variables are grouped together. For the fault consideration, a huge penalty can be set in the minimization process if it does not reach the theoretical detectability according to [8]. On the other side, for the variables it would be necessary to establish a criteria that could identify which variables work better together. In this paper a method is proposed considering that the ”goodness” of a pair of variables is weighted by two indicators:

- Model Performance: how accurately can a variable be predicted when the other is used in the model input.
- Model Correlation: how disperse is the sensitivity to all inputs when this variable is used.

While the first is more general, the second is necessary given the SA approach. The statistical correlation among variables lead to models that are highly dependent on a single variable. To build a weight matrix that can account for how interesting is the set of variables in each MSO of a potential set the following steps have to be followed:

1) For each variable measured m_k find all the variables it can be linked through the different feasible MSOs obtained from the SA, $\vec{v}_k$.
2) For each m_k train a model that aims to predict it using all the variables in $\vec{v}_k$. In this paper, it is used ElasticNet with crossvalidation to select regularization parameters ([9]).
3) Taking the models trained in the step before, select the variables with bigger weights so that $\sum_{i=1}^{Q} w_i > \gamma \sum_{j=1}^{N} w_j$ where γ is the ratio of the total weight (sum of the weights for all the N variables used) that the subset of size Q must account for. Train a new model

for each variable v_q of the subset generated where m_k is predicted by the variables in $\vec{v}_k$ except from v_q.

4) Repeat the step before p times, generating smaller subsets each time based on the results of the previously trained models.
5) For each of the trained models compute the R^2 metric using a test set and $D(\vec{w_h})$, where w are the weights of model h. The metric function $D(\vec{w_h})$ is defined as

$$D(\vec{w}) = \sum_{i=1}^{N-1} \left(\frac{\mid w_i \mid - \mid w_{i+1} \mid}{\sum_{j=1}^{N} w_j} \right)^2 \quad (1)$$

subject to: $\forall i \in [0, N-1] \mid w_i > w_{i+1}$

and evaluates how uneven is the weight distribution in the model.

6) Since the MSO exploration is driven by a heuristic minimization, for each of the trained models combine both metrics resulting from the previous step according to

$$S(f) = (1 - R^2(f)) + \alpha D(w_f) \quad (2)$$

The combined metric S takes as input the trained models f and the parameter α tunes which element is more important: the weight distribution or the model accuracy.

7) To build the weight matrix ϕ each element $\phi_{i,j}$ is the mean value of all $S(f)$ where f uses variable i as a target and j as one of the input variables. The diagonal values are equal to 0 as well as those variables that were not used in the MSOs.

This proposal explores which combinations of variables would lead to better results and can be exploited in different ways. An alternative to the last step would be to compute an index of which variables tend to give better results when used as targets. This could lead to an array ϵ, where each element ϵ_i is the mean value of all $S(f)$ where f uses variable i as a target. Using this matrix ϕ, the minimization process will lead to a set of MSOs $F = [f_0, \ldots, f_p]$ that has the best combined score in $S(f)$ defined in (2). In addition, the fault signature of this set has to reach the theoretical detectability. A way to ease the search is to use heuristic search such as Genetic Algorithms [10] that are fast to implement and allow to include additional constraints.

2) Residual Generation: Given the sets of variables measured in each MSO, the objective is to build a model that can identify the deviations on the model caused by faults. This requires two parts that will model the relation: the estimator that can predict a variable from the others and the uncertainty of the deviation obtained. Mathematically it would be expressed as:

$$r(k, \theta) = y(k) - y_m(k, \theta) \quad (3)$$

where $r(k, \theta)$ is the residual obtained from the measurement $y(k)$ and the model estimation $y_m(k, \theta)$ at discrete time k with the trained parameters θ. This residual will show the divergence of the system current behaviour with respect to the "good behaviour model" trained. To evaluate this divergence it is necessary to establish a criteria, in this case it is assumed that the probability density function (PDF) of the error can be obtained.

In first instance, the PDF can be studied with the prior observation of a certain systems state. With this probability distribution, confidence boundaries are defined so that there are upper and lower thresholds that help to select the more relevant samples

$$r(k, \theta) \in [\underline{r}, \overline{r}] \quad (4)$$

where $\underline{r}$ and $\overline{r}$ are respectively the upper and lower bounds of the residual.

The selection of the variables to be predicted from the others is not a trivial problem. However, this analysis is not feasible under the stationary assumptions and the equilibrium equations defined. As an alternative, the method proposed for MSO selection also provide a vector ϵ that shows which variables perform better (lower weights), allowing to choose the variable that perform best when predicted among the ones available in the MSO.

The relations that the variables present in the equations could be inferred through the processing of many samples (under good behaviour condition). In this paper, the case study will use a Multi Layer Neural Network [11] to link the variables in each MSO. The network would solve at the same time the issue of unknown parameters and the link between variables, considering that the non-linearities can be learned by the network if its big enough and has enough data [12].

When it is possible to study the distribution of enough residuals, the probability density function (PDF) of the results can be inferred as well. Obtaining this distribution would allow to establish where are the upper and lower limits that a certain degree of confidence would give us.

The model precision vary with the system state, it would provide more sensitivity to allow the uncertainty threshold depend on the state of the system at each point. Kernel Density Estimation (KDE, developed from the work of [13]) is the technique proposed in this paper to define a PDF with respect to more than one variable.

The more dimensions considered, the fewer density of samples (lower accuracy) and better data generation processes are required. To deal with that problems, Principal Component Analysis (PCA) can be used over contour condition variables to extract one or more variables that characterizes the system state. Combined with the KDE at each sample, we could obtain the PDF of the errors at a given state, adapting the confidence and therefore augmenting detection accuracy.

However, the KDE method is based on density of samples and will approximate the distribution. Since false detections must be avoided as much as possible, the bias in the sample

density is too great. The KDE tuning parameter provide a well estimated distribution when combined the PCA components. But, to relax the limits and ensure that all good behaviours are accounted, a Gaussian filter is proposed to relax the obtained distribution.

Since the method will provide a discrete distribution, applying a Gaussian filter on the resulting distribution will generate an error distribution with smaller concentrations. To tune this filter, σ can be chosen so that all variables of the training set for KDE are included in the generated bounds. To avoid huge sensitivity loss, the training set can be filter applying an outlayer removal step before determining σ for each MSO.

B. Online Detection

Taking the trained pair of variable and PDF estimators, new samples can be fed to extract the residual value and the confidence bounds for such error. These results indicate if at a certain point the system behaves different from what is expected (residual out of confidence bounds, activated). But, when there are thousands of samples to evaluate, statistical anomalies are expected without being necessarily triggered by a fault.

The position of the error with respect to the distribution given by the KDE at that instant k can be translated into two variables:

- Activations: This is a Boolean indicator regarding if the error is outside of the boundaries defined. These boundaries are the confidence intervals, adjusted according to the application needs.
- Confidences: A value to quantify how wide is the deviation between predicted and measured value. To determine it, the probability of the error can be compared with the best possible scenario (or a range of values if some uncertainty can be taken from the sensors) according to the probability distribution. As a result it would be a value that ranges from 0 (error is the most expected) to 1 (the error is far from the expected result)

The online detection of faults must include an interpretation of the results to make the fault detection robust to false alarms. If the training has been successful, the consecutive occurrence of anomalies will indicate a fault. The solution proposed is an adaptation of the one developed by [14] where a Bayesian interpretation of consecutive detections is built.

This method computes the posterior probability of each fault given the fault indicators obtained from the trained models. In this computation, it is necessary to use the prior fault probability, which is updated with the result obtained at each time. The recursive application must converge indicating that a fault is far more probable that the others, or that maybe the probability is not big enough to consider it as a fault.

Complementary, to avoid that separated events are mixed together, at each sample time is verified if all the residuals are within the confidence boundaries using the updated probabilities. With a properly tuned forget factor γ, the fault prior probabilities are taken back to its originally defined values according to

$$P_{fi}(k) = \frac{(1-\gamma)P_{fi}(k) + \gamma P_{fi}(0)}{2} \tag{5}$$

As a result the probabilities of the faults are updated only for potential fault cases leading to an increasing probability, i.e., for the faults whose marking in the FSM are matching with the detected anomalies (activation of MSO, error beyond threshold). This adapted version of [14] apart from considering previous results through the priors, the activations are treated as a damped signal.

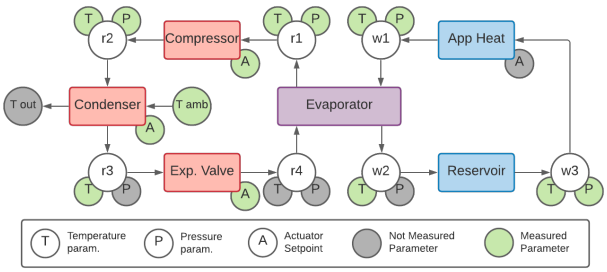

Fig. 2. Diagram describing the refrigeration process.

III. Case Study: Water Cooling Machine

A. System Description

The thermodynamic application presented is a case of a Carnot Cycle for refrigeration (see Fig. 2). The water cooling system is composed by two closed circuits, one for the water and other for the refrigerant. The system is driven by four actuators that control the mass flows of water (pump), refrigerant (compressor and Electronic Expansion Valve (EEV)) and air (fan).These flows lead heat exchanges in three points where air takes the heat from the refrigerant (condenser), the refrigerant from the water (evaporator) and the water from the application to cool down. The heat extraction is the machine purpose, facilitating water at a given temperature (set point). In such a system, the relevant variables to measure are the power consumed by the actuators (except for the EEV where the percentage of time open is taken), the temperature, the pressure and the mass flow. In the proposed sensing set, there are 12 variables and the 3 control output values. The instruments used are: one mass flow meter for measuring the water flow and the rest are 4 pressure sensors (sensitivity of $\pm 1.2\%$) and 7 temperature sensors (sensitivity of $\pm 0.5^oC$).

1) Thermodynamic model: To model this system, the equations derived from the first law of thermodynamics have been used to describe equilibrium relationships. This assumption can be extrapolated to a wide range of applications where this energy transformations apply, while being flexible enough to fit other specific constraints.

From the theoretical point of view, it could be accepted if the input heat source is seen as a hot reservoir and the

ambient air as a cold reservoir. If the system dynamics can reach the equilibrium fast enough when changes are applied, the thermodynamic equilibrium assumption would be coherent.

2) Faults considered: For some of the equations, it is possible to associate a fault that would represent the failure of the behaviour described in the corresponding equation. The fault cases considered are divided in four types:

- Leaks: Both circuits can suffer a leak and since it is a closed loop, the leak effects would feel like a global mass loss. But, the presence of a water tank make the water leak not detectable without high accuracy pressure sensors or a level sensor.
- Components: Each of the components will have an additive fault that represents the inefficiency of the component's fault. Therefore every time one of the components misbehave it could be detected.
- Obstructions: The obstructions can be detected when unexpected pressure drops appear in a section of the circuit.
- Sensing: Finally, the faults in the sensors will be included in the equations that link some variables with the sensed value, as an additive deviation that might affect the perceived value.

B. Results

1) Residual Generation: For the generation of residuals, the first step is to obtain the MSO sets and identify the variables measured in each group. Starting with the relations given in the described system, a set of elementary relation between variables can be obtained. With this set all the combinations of redundant groups of equations can be obtained. But from the original equations only the ones belonging to the overdetermined set are relevant.

From the overdetermined set. all the combinations are obtained for MSO groups of equations, obtaining 525 sets. There are 21 faults to identify, from which one is not detectable and five are not isolable (groups of 3 and 2).

The obtained result is a set of 6 MSOs capable of identifying the detectable faults and non isolable groups. These sets can be linked to the variables that they involved, the equations and ultimately the faults. In this way, a detectable and isolable fault (or non isolable group) would be defined by a specific combination of MSOs being deviated beyond the thresholds toguether.

To reach this set, the weight matrix ϕ in Fig. 3 used was calculated from the training set and guiding the MSO selection. This selection is guided then by the sets that will perform better together and will aim for the variable that might lead to evenly distributed weights.

2) Models Training: The selected NN will have $36 * z$ neurons (where z is the input size) with Rectified Linear Unit (ReLU) activation functions. These neurons are distributed in 6 layers that also use L2 regularization and, for the first two layers, Dropout (with a 10% rate). For the training the optimizer used is Adam [15] and mean square error as the loss metric.

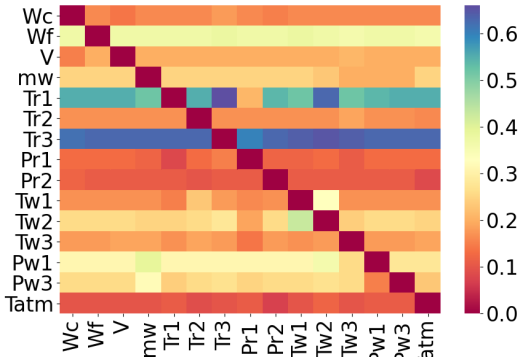

Fig. 3. Weight matrix ϕ used to evaluate how good variable i is when in the input set is variable j.

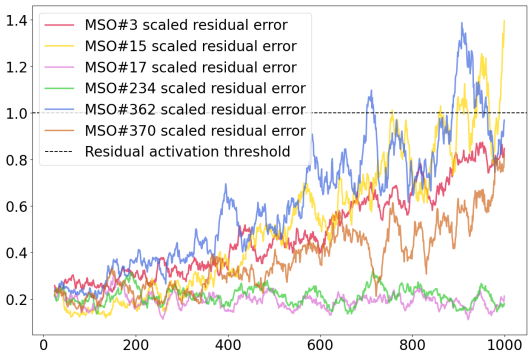

Fig. 4. This figure shows the models trained for each of the four MSOs, where only MSOs 17 and 234 are not sensitive to the fault. Each model uses the same test dataset with an increasing disturbance affecting only to the evaporator outlet temperature in the water circuit. The plot displays the relative distance of the residuals to the error boundaries (where values above 1 would be activations). The boundaries are obtained with a 98% confidence boundaries obtained from the described approach in Section II-A.2.

A training set for the NN is obtained with 24747 samples, subsampled to 4601 for the KDE training and a validation set with 1000 samples. The data considered has been filtered to include only data from when the actuators are activated in equilibrium states. For other machine models, it might be necessary to consider the division between different working state, obtaining one trained solution for each.

3) Fault detection test: To test a simple fault case, the same sample is provided to the model with an incremental noise disturbance in the pressure sensor after the compressor. This deviation takes an incremental value from 0 to a 50% of the variable mean, and a fiftieth of it for the standard deviation of the noise. The residuals obtained are shown in Fig. 4 where the disturbed variable is the evaporator outlet temperature in the water circuit.

This fault should trigger the MSOs 3, 15, 362 and 370 to be correctly identified (its signature). All these models are disturbed from the nominal example, progressively showing more activations (out of bounds errors).

However, this also highlights the problem of the unaccounted sensitivity of each model. Some of the models are less influenced by that variable than others and that makes that some models are presenting activations way sooner than others. Again, this is an issue that must shift the fault identification as new evidence is provided.

In Fig. 5, the online detection solution displays how the fault identification changes as the deviation introduced grows. The identification proposed converges at each time to the most probable fault given the observed activations (and confidences) and the previous historic values. When the last models start presenting deviations big enough to identify activations the target fault is detected. This solution could be tuned in many different ways according to the application specifications.

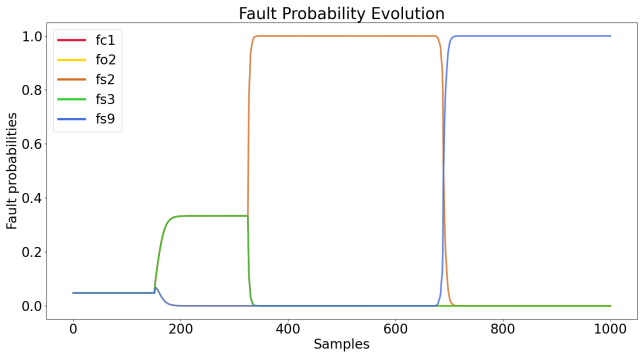

Fig. 5. Probabilities of the 5 faults that showed bigger probabilities along the fault example: fo3 (obstruction of refrigerant in condenser), fs1 (sensor of the mass water flow), fs2 (sensor of T_{r1}), fs9 (sensor of T_{w2}) and fs10 (sensor of T_{w3}). This interpretation is obtained from the results of models in Fig. 4 and shows how the model reaches convergence towards the fs9.

IV. Conclusion

The present paper aimed to introduce a methodology that focused on the scalability of a fault detection system, proving that it could be possible to meet the industrial requirements. The solution presented could be executed automatically once the expert knowledge (structural matrix) and the telemetry data of correct behaviour is fed. Without further guidance, it is necessary to establish a criteria that can lead the search of the best possible combinations of variables provided by the SA. The proposed weight construction is an easy way to tune the trade off between model performance and sensitivity. The trained MSOs for the case of the refrigeration equipment show how these limitations of the trade-off affect the modelling. Through the learned PDF and estimator, it is possible to provide a residual and establish a confidence level of the deviation observed. However, it has been shown that the monitored data will present anomalies from the model point of view, that are not faults. To increase the sensibility implies to have a higher chance of false detections. These scenarios exemplify the importance of the interpretation module. It must be able to extract a dynamic analysis from isolated samples, in a way that the target experts can benefit from this information. The posterior evaluation of these residuals allow to provide a probability for each fault and to update the prior in an iterative way. The convergence of these results is more easily tuned than the data-driven models and can be done in accordance to the application needs. As future work many options appear to complete this method and fully implement it in a production environment. Among the many options the authors propose alternative estimators (as e.g. ANFIS, Bayesian Networks), feature generation (autoencoders instead of PCA), sensor evaluation for new measurements, an interpretation scheme that considers multiple faults simultaneously or the model sensitivity use for fault identification.

Acknowledgements

This work has been funded by the Catalan Agency for Management of University and Research Grants (AGAUR) ACCIO RIS3CAT UTILITIES 4.0 P4 ACTIV 4.0 and Grant 001-P-001643 Agrupació Looming Factory.

References

[1] H. L. R. Mobley, R. K. and D. J. Wikoff, *Maintenance Engineering Handbook*. New York: New York : McGraw-Hill, 2008.

[2] R. N. V. W. Haasl, D. and F. Goldberg, *Fault Tree Handbook*. U.S. Nuclear Regulatory Commission, 1981.

[3] B. A. P. B. Escobet, T. and V. Puig, *Fault Diagnosis of Dynamic Systems*. Springer, 2019.

[4] Y. D. Zhang, W. and H. Wang, "Data-driven methods for predictive maintenance of industrial equipment: A survey," *IEEE Systems Journal*, vol. 13, no. 3, pp. 2213–2227, 2019.

[5] J. de Kleer and J. C. Williams, "Diagnosing multiple faults," *Artificial Intelligence*, vol. 32, no. 1, pp. 97–130, 1987.

[6] B. Pulido, J. M. Zamarreño, A. Merino, and A. Bregon, "State space neural networks and model-decomposition methods for fault diagnosis of complex industrial systems," *Engineering Applications of Artificial Intelligence*, vol. 79, pp. 67–86, 2019. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0952197618302665

[7] D. Jung, "Residual generation using physically-based grey-box recurrent neural networks for engine fault diagnosis," 2020.

[8] K. M. Frisk, E. and D. Jung, "A toolbox for analysis and design of model based diagnosis systems for large scale models," *IFAC-PapersOnLine*, vol. 50, no. 11, pp. 3287–3293, 2017.

[9] J. Friedman, T. Hastie, and R. Tibshirani, "Regularization paths for generalized linear models via coordinate descent," 2009.

[10] D. Whitley, "A genetic algorithm tutorial," *Statistics and Computing*, vol. 4, no. 2, pp. 65–85, 1994.

[11] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, "Parallel distributed processing: Explorations in the microstructure of cognition, vol. 1," D. E. Rumelhart, J. L. McClelland, and C. PDP Research Group, Eds. Cambridge, MA, USA: MIT Press, 1986, ch. Learning Internal Representations by Error Propagation, pp. 318–362.

[12] B. Y. Goodfellow, I. and A. Courville, *Deep Learning*. MIT Press, 2016, http://www.deeplearningbook.org.

[13] M. Rosenblatt, "Remarks on some nonparametric estimates of a density function," *The Annals of Mathematical Statistics*, vol. 27, no. 3, p. 832–837, 1956.

[14] B. J. T.-S. S. Fernandez-Canti, R. M. and V. Puig, "Fault detection and isolation for a wind turbine benchmark using a mixed bayesian/set-membership approach," *Annual Reviews in Control*, vol. 40, pp. 59–69, 2015.

[15] M. Abadi, "TensorFlow: Large-scale machine learning on heterogeneous systems," 2015, software available from tensorflow.org. [Online]. Available: https://www.tensorflow.org/