Classifier Fusion in the Accelerometer Sensor Diagnostics

Piotr Bilski[1,2]

[1] *Institute of Radioelectronics, Warsaw University of Technology, ul. Nowowiejska 15/19, 00-665 Warsaw, Poland, pbilski@ire.pw.edu.pl*
[2] *Faculty of Applied Informatics and Mathematics, Warsaw University of Life Sciences, Nowoursynowska 159, 02-776 Warsaw, Poland, piotr_bilski@sggw.pl*

Abstract- The paper presents the application of multiple classifiers to increase the accuracy of the fault detection module. The structure of the module is presented with the applied voting mechanisms. The classifiers implemented to the fault detection operation are briefly described. Issues of the practical implementation are considered. The scheme is used in the diagnostics of the analog part of the MEMS accelerometer. The paper is concluded with the possible prospects for the proposed scheme.
Keywords - analog systems, accelerometer sensor, fault classification, classifier fusion.

I. Introduction

Contemporary diagnostic systems often rely on the Artificial Intelligence (AI) methods, which provide fast and reliable apparatus for the fault detection and location based on the analysis of previously collected data. Their advantages are automated operation and the ability to extract knowledge from data sets, often difficult to follow for the human operator. Disadvantages include sensitivity of the classification accuracy to the quality of the learning data. Applied approaches include rule-based methods, Artificial Neural Networks (ANN), statistical or dictionary algorithms. Each method stores knowledge in the unique form, therefore its generalization ability is different than for other approaches. Based on the comparative analysis [1] it is apparent that each classification method produces errors for different cases in the same System Under Test (SUT). Therefore combining multiple algorithms into one module might improve the diagnostic efficiency measured by the percentage of correctly identified faults.
The paper presents the application of combined AI-based classifiers to the diagnostics of analog system. In section II the AI application to data-driven fault detection is explained. Section III introduces the fusion (ensemble) architecture. Section IV describes the analysed accelerometer, while in section V experimental results are presented. Section VI contains conclusions and future prospects.

II. Data driven approaches

The presented architecture belongs to the data-driven approaches using the supervised learning paradigm. Here the applied methods exploit training and testing data sets ($\boldsymbol{L}$ and $\boldsymbol{T}$, respectively), obtained through the simulation of the SUT model. Each set consists of labelled examples (with the category c_i for the i-th example determining the actual SUT state identification), representing values of symptoms s_{ij} acquired form the SUT responses after exciting them with the selected input signal (1). In the presented research, the category is determined by the fault code, combining the number of the faulty parameter and its deviation from the desired (nominal) state. The integer value “11” indicates that the first parameter has value “larger than nominal”, while “-21” is for the second parameter with the value “smaller than nominal”. The fault-free state is indicated by the code “0”.

$$\boldsymbol{L} = \boldsymbol{T} = \begin{bmatrix} s_{11} & \cdots & s_{1m} & c_1 \\ \vdots & \ddots & \vdots & \vdots \\ s_{n1} & \cdots & s_{nm} & c_n \end{bmatrix} \quad (1)$$

This way the SUT work regime is represented by the set of examples, from which knowledge is extracted during the training procedure. The training consists in extracting knowledge the resulting in the smallest sample error

(2) on the training set (where *d* is the decision of the module). The testing set is used to check the classification quality. Although different cross-validation techniques can be used to partition data [2], the most popular is the usage of two sets of two similar (or identical) sizes and describing the same types of faults.

$$e(\boldsymbol{L}) = \frac{|d_i = c_i|}{|\boldsymbol{L}|} \tag{2}$$

The selection of simulations and symptoms for analysis is a separate problem, requiring from the designer a deeper knowledge about the system's work regime. It requires deeper interest in the future.

III. Fusion (ensemble) architecture

The application of a single classification module to the fault detection and location is well established in the diagnostic domain. The scheme consists in training the method using the learning data set and using this knowledge $\boldsymbol{K}$ to work with the actual SUT. The classifier fusion is the implementation of multiple modules to work in parallel and make decision about the state of the same object simultaneously. This potentially increases the diagnostic accuracy. Outputs of each classifier are aggregated to select one valid diagnostic decision. The training is performed in the off-line mode, therefore it can be done sequentially (using slower computer systems). The training can be implemented in two ways. The first one requires all methods to be trained on the same data. The knowledge for the *j-th* module is extracted and stored in memory. Alternatively, different modules are trained using different sets (for instance, consisting of different symptoms, or coming from various domains). In the presented research only the first scheme was used, requiring simpler data acquisition process.
The trained modules are used on-line, during the SUT operation. Because of the Real-Time mode requirements, they should work in parallel, which requires the multi-processing unit, such as the multi-core processor. The final decision d_o is made using the selected strategy. Among multiple approaches [3] two were used in the presented research:

- the selection of the most frequent decision (unweighted majority voting [4]):

$$d_o = \max|d = d_j| \tag{3}$$

  In this scheme each classifier is treated equally, therefore the decision depends on the greatest number of responses pointing at the same fault.
- the weighted classification, where the impact of each classifier on the final decision depends on its sample error. The sum of all errors is the denominator for the fraction defining the importance of the particular module. The voting mechanism is similar to (3), but considering weights (4).

$$d_o = \max|d_j \cdot w_j : d_j = d| \tag{4}$$

  where:

$$w_j = \frac{e_j(\boldsymbol{L})}{\sum_{j=1}^{J} e_j(\boldsymbol{L})} \tag{5}$$

The scheme form Fig. 1 is the implementation of the analytical redundancy, as the alternative to the hardware redundancy. It is assumed that the combined decision from multiple modules is more reliable than from only one. Also, in the uncertainty conditions the combined knowledge may be more efficient. The number of classifiers in the diagnostic module and selection of the particular algorithms to the task is the important parameter. In the unweighted voting (3) the odd number of modules should be used (assuming the majority of them will produce the proper decision). In the scheme from Fig. 1 three modules are the smallest group. If the even number of diagnostic modules is used (two, four, etc.), it is possible to have equal number of different outcomes, which requires the additional analysis. In the weighted voting the number of classifiers is not limited in any way.
The selection of classification modules can be done in two ways. Firstly, multiple instances of the same classification method are used, but with different parameters. For example, multiple SVM classifiers with various widths of the Gaussian kernel [6] can be used. In this solution all instances of knowledge $\boldsymbol{K}_1, \ldots, \boldsymbol{K}_l$ extracted from the set $\boldsymbol{L}$ have the same form.

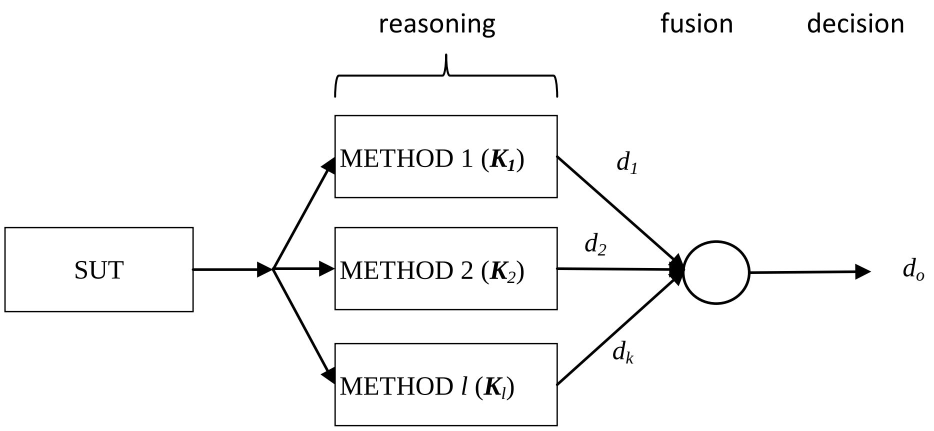

Fig. 1. Decision making module for the multiple classification methods [5].

The second approach, used in this research, is the selection of different classifiers. If the diversity of knowledge representation is important, the most different approaches are selected. For instance, RS, ANN and NBC are three algorithms storing knowledge in different forms. They must be examined to determine their behaviour with training and testing sets. Even methods storing knowledge in the similar way (such as fuzzy logic or rough sets), process the same examples differently, producing separate diagnostic decisions. Also, comparison between them shows which SUT's states are simpler or more difficult to diagnose.
Example of the classifier fusion applications is in [6], where a set of SVM classifiers detects faults in the motor bearing in the presence of noise. In [7] three classifiers are applied to analyse the state of the aircraft engine. The compared approaches included ANN, SVM and decision trees. The fusion often proves more accurate than separate approaches. In [1] FL and RS were compared to find their advantages and disadvantages for the analysis of electronic filter and the DC motor.

IV. System Under Test

The example of the accelerometer model from Analog Devices (www.analog.com) is in Fig. 2. Its task is to measure changes in the velocity (up to *2g*) along *x* and *y* axes. The device is positioned over the silicon basis using springs. The acceleration is measured indirectly by checking the value of the differential capacitor C_{DC} with one plate constant and the second one attached to the moving the plate of the sensor. The demodulation of the obtained square wave allows for determining the direction of the acceleration. The signal is then filtered to increase the reading accuracy. The acceleration is considered as the duty cycle of the square wave (for the nominal, zero acceleration, it is 50%).

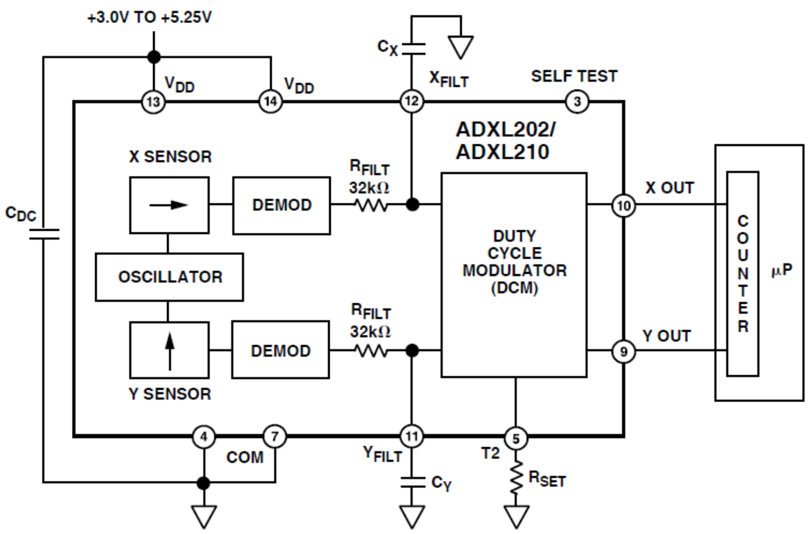

Fig. 2. Scheme of the accelerometer

Four parameters of the sensors were diagnosed: *x* and *y* axis beam coefficients (*a* and *b*, respectively), amplifier gain *K [V/g]* and demodulating filter effects coefficient *e*. Their nominal values were: *a=8.374e-10*, *b=5.788e-6*, *K=12.54 [V/g]* and *e=2.27e-5*. The remaining parameters were constant during the experiments: self-test gain $s_g=0.8g$, sensitivity *S=1 V/g*, the filter 3db-band frequency $b_f=50$ *Hz* and z-axis beam coefficients *w=8.25e-10* and *g=2.872e-5*.
Testing of the sensor requires exciting it with the sinusoidal signal and measuring its responses. The linearity of the sensor's response to the measured quantity is the most important feature - nonlinearities should be eliminated. Two responses were considered during simulations: *x*- and *y*- axis output voltages. From them the coordinates of the maximum, the minimum voltage with time instants, mean value, amplitude (dynamic range), cycle RMS (the sum of all squared samples), the sum of positive samples and the sum of negative samples were collected (Fig. 3). The main problem with this system is low sensitivity of its parameters, allowing for detecting only their relatively large changes.

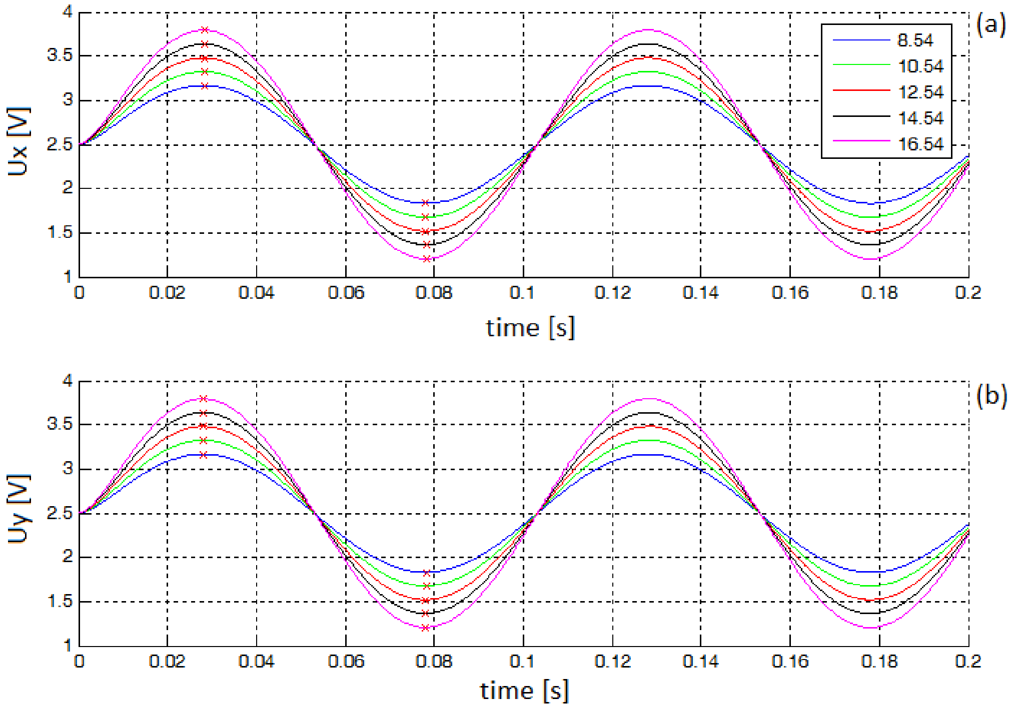

Fig. 3. The output x-axis (a) and the y-axis (b) voltages of the accelerometer for different values of the amplifier gain with stamps extracted.

V. Experimental results

The accelerometer was tested by three classifiers working in parallel. The aim of the experiment was to check if the combined classification supported by voting of all modules is more efficient than the single diagnostic algorithm. The decision tree (DT), the Naïve Bayes Classifier (NBC) and the neural network (MLP) were used. All these approaches have low computational complexity during the decision making (even for large number of symptoms), which makes their implementation in the on-line module justified. The selected methods belong to three different AI categories, facilitating comparison of knowledge forms stored by each approach. DT and ANN work with continuous data, while the NBC is the statistical algorithm, requiring discrete symptoms' values. Therefore training and testing sets were first discretized, using the Equal Frequency Binning (EFB) [8]. Both voting schemes presented in section III were implemented. For the second one the sample error was measured (see Table 1).
Data sets prepared for the experiment consisted of twenty four examples, describing the sensor behavior after changes in four parameters. To determine the system's state, twenty symptoms were collected for both voltage waveforms (Fig. 3). The original information about the faulty parameter and its actual value was exchanged with fault codes. As the range of changes for each parameter was different, the normalization of parameter values was first used (6). This allowed for the uniform calculation of codes based on the relative parameter changes. Eleven categories were obtained this way. The diagnosed parameters were indexed by the subsequent numbers: *x*- and *y*-axis beam coefficients *a* and *b* (1 and 2, respectively), gain *K* (3) and demodulating filter effect coefficient *e* (4).

$$s_{ij} = \frac{s_{ij}}{\max\left(\left|\boldsymbol{s}_j\right|\right)} \tag{6}$$

All classifiers were trained on the same data, although NBC required discretization, while for MLP the "One vs All" coding was used to represent categories, leading to eleven neurons in the output layer. The network with one hidden layer was designed with changing number of units to check the influence of the MLP structure on the diagnostic efficiency. Remaining methods were not parameterized. First tests on training sets (to ensure they acquired knowledge correctly) were executed. Because MLP has the lowest sample error, its responses will have the greatest influence on the diagnostic outcome.

Table 1. Correct classifications on L and weights for the classifiers in the ensemble.

Classifier	DT	MLP	NBC
1-e(L)	54,16 %	58,33 %	29,16 %
w	0.382	0.411	0.206

The experiments with the testing set ***T*** allowed for drawing the following conclusions:

- Approaches working with the continuous data are better suited for the classification of faults in the analog system. Although the analyzed sets were difficult, MLP and DT were able to correctly detect over half of them. The efficiency of the Bayesian method was around thirty percent. This is caused by the problems with discretization. Their symptoms often belong to different intervals, although the fault code is identical. Even the best EFB discretization (with two or three intervals for every stamp) leads to poor results on relatively small sets. As the diagnostic efficiency is the lowest of all used classifiers, larger data sets would be required.
- The presented classifier fusion allowed for improving the diagnostic quality compared to the single modules. As shows Table 1, the best outcome of the single algorithms is below sixty percent. Thanks to the majority voting the overall classification outcome was about sixty five percent. The increase in the diagnostic efficiency was possible, because at least two methods identified different faults correctly.
- The DT was the fastest approach, both in the learning and the testing stage. Its structure consisted of 39 nodes, including 18 leaves. Some examples belonging to the same category were represented by separate leaves. The tests in the tree were selected to maximize the number of analyzed symptoms, to avoid multiple tests dividing the range of the single parameter into too many intervals. The tree structure was easily readable, allowing for monitoring the decision making process.
- The MLP training the longest one, especially as it requires repeating the procedure for changing network structure. Its efficiency is comparable to the DT, although ANNs are better prepared for the analysis of noisy data. The optimal number of neurons was between 15 and 19, depending on the initial weight generation.

Table 2. Comparative results of the diagnostic decisions made by the selected classifiers.

Row No.	Actual fault code	DT	MLP	NBC	Majority voting	Weighted voting
1	0	0	0	0	0	0 (1)
2	11	11	21	11	11	11 (0.588)
3	22	-31	22	X	?	22 (0.411)
4	31	31	31	31	31	31 (1)
5	-32	0	11	-31	?	11 (0.411)
6	12	-32	11	12	?	11 (0.411)
7	41	41	41	41	41	41 (1)
8	42	42	42	21	42	42 (0.793)

The examples of the grouped classification for both voting schemes are in Tab. 2. The "X" value in the NBC column means the module was unable to make decision (because of the unique combination of interval numbers in the testing example). In the majority voting three situations are considered:

- All classifiers point at the same fault code (for example, categories "0", "31" and "41"). This is the simplest case, when the decision is made unanimously.

- One of approaches is wrong, but remaining two are correct (for instance, “11” vs “21 in the row No. 2). Using the majority vote rule, the fusion module makes the decision.
- Only one method is correct, while two others identify other faults (such as “-32”, “11” and “12”), or no classifier made the correct decision. This is the most difficult situation (indicated by ”?” in Tab. 1), which can’t be resolved by simple voting. In such case the support for each module is calculated. Alternatively, the random selection was made, giving similar results. If two classifiers identify the same parameter but with different intensity, one of codes proposed by them should be the diagnostic decision.

After introducing the second voting scheme, the change is visible in rows No. 3, 5 and 6 (the value in brackets is the weight of the winning decision). In the first case introduction of the weights allowed for increasing the diagnostic accuracy. In two remaining cases the decision of the ensemble was still incorrect, because each classifier pointed at the wrong fault and the most important method (MLP) pointed at the wrong fault. In row No. 2 the MLP was wrong, but remaining methods pointed at the correct fault (and their combined weight is greater than that of MLP). Also, the weighted voting was never worse than its majority counterpart.

VI. Conclusions

The presented experiment shows significant differences between various approaches used for the same purpose. To increase the diagnostic efficiency, subsequent faults should be described by the greater number of examples. Also, other classification approaches (such as SVM or rules-induction algorithms) should be tested. Adding more methods to the ensemble may lead to the further increase in the diagnostic efficiency, although the analysis of computational abilities of the hardware must be performed to check if they are enough to run them in the reasonable time.

The fusion of classifiers is an efficient tool in the difficult diagnostic cases, where the relation between the symptoms and the SUT state are not easy to establish. Because each algorithm stores knowledge in different form, there is the possibility to correctly identify faults just by combining multiple approaches, or by considering their performance on the training set. The future research should focus on selecting new classifiers and testing their different configurations.

References

[1] P. Bilski, “An unsupervised learning method for comparing the quality of the soft computing algorithms in analog systems diagnostics,” Przegląd Elektrotechniczny, No. 11a/2010, pp. 242-247.
[2] R. Kohavi, "A study of cross-validation and bootstrap for accuracy estimation and model selection". Proc. the Fourteenth International Joint Conference on Artificial Intelligence, San Mateo, CA, USA, pp. 1137–1143, 1995.
[3] L. Kuncheva, "Combining Pattern Classifiers. Methods and Algorithms," Wiley and Sons, New Jersey, 2004.
[4] F. Moreno-Seco, J. M. Iñesta, P. J. Ponce De León, and L. Mico, "Comparison of classifier fusion methods for classification in pattern recognition tasks," Structural, Syntactic, and Statistical Pattern Recognition, Lecture Notes in Computer Science, Vol. 4109, pp. 705-713, 2006.
[5] P. Bilski,”Artificial intelligence methods in the diagnostics of analog systems,” Warsaw University of Technology, Warsaw, 2013.
[6] L. Batista, B. Badri, R. Sabourin, and M. Thomas, "A classifier fusion system for bearing fault diagnosis," Expert Systems with Applications, Vol. 40, pp. 6788–6797, 2013.
[7] W Yan, C.J. Li, and K.F. Goebel, "A Multiple Classifier System for Aircraft Engine Fault Diagnosis," Proceedings of the 60th Meeting of the Society For Machinery Failure Prevention Technology, (MFPT), pp. 271-279, 2006.
[8] A. Øhrn and J. Komorowski, "ROSETTA: A Rough Set Toolkit for Analysis of Data," Proc. Third International Joint Conference on Information Sciences, Fifth International Workshop on Rough Sets and Soft Computing (RSSC'97), Durham, NC, USA, March 1-5, Vol. 3, pp. 403-407, 1997.