Automation and Flexibility of Analog Systems Diagnostics

Piotr Bilski , Jacek Wojciechowski
Institute of Radioelectronics, Warsaw University of Technology
ul. Nowowiejska 15/19 Warsaw, Poland
{ pbilski, jwojc}@ire.pw.edu.pl

Abstract - **The paper provides general insight into the analog systems diagnostics architecture. Automation of the wavelet transform de-noising procedure is presented. Decision trees based fuzzy logic decision making module for single and multiple fault detection of the 4th order servomechanism is described. The factor for assessment of the diagnostic quality considering limited size of the learning data structure is proposed.**

Index Terms – fault detection, analog systems, diagnostics, machine learning, artificial intelligence.

I. INTRODUCTION

Modern Fault Detection and Isolation (FDI) techniques focus on the artificial intelligence methods. Neural networks or genetic algorithms [1] are successfully used to diagnose faults in various technical domains. Automation and versatility of the methods is strongly preferred. In [2] the architecture addressing these issues was proposed. Working in noisy conditions, automatic fault detection and self-design of the diagnostic module are advantages of the architecture (Fig. 1). It divides the diagnostic operation into smaller, basic stages (labelled by letters A-E). Therefore it can be used for variety of objects, requiring only learning data from simulation of their models. Using the latter is popular and efficient method of systems' analysis [3]. The architecture minimizes required knowledge about the objects' work regime. Novelty of the approach is the ability to treat different analyzed objects in the same way, opposed to the traditional methods with limited applications. In our technique SUTs are processed in the form of the set of the numbers – measurement information. Decision making module works on the data extracted from system under test (SUT) responses, i.e. coordinates of certain response points (called stamps), determining SUT element values change. Method's usefulness was confirmed on SUTs from electronic and mechatronic domains [2]. Aiming at the increase of the architecture quality requires insight into particular stages. The most important are de-noising (B) and decision making (D), having the greatest influence on the diagnostics quality. Wavelet transform was used to clear the noisy signal, while fuzzy logic has been utilized to provide decision module [2]. Both methods can be parameterized and automated. Fuzzy logic proved its usefulness in uncertainty conditions, when multiple faults occur, but is not self-designable. Versatile architecture requires incorporating existing and new concepts into a generic scheme. It enables the designer to choose particular algorithms and deploy them freely. Architecture flexibility allows for different methods application, but requires efficient assessment methods for them. The approach goes beyond the traditional understanding of the diagnostics and significant increase of possible applications is expected.

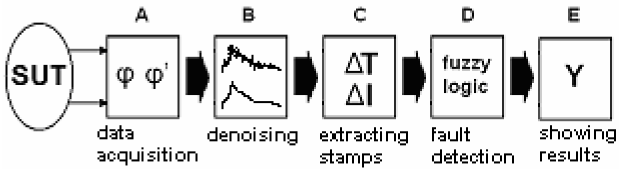

Fig. 1. Generic diagnostic architecture

Possible applications of the architecture require detailed examination of its stages and comparing various diagnostic methods. The most important stages (B,D) are considered in the paper. It has three goals. Firstly, the architecture versatility is confirmed by fault analysis of the not tested yet 4th order servomechanism. Secondly, automated de-noising and multiple fault detection of the SUT are considered. Finally, a method for the diagnostics quality assessment is introduced. It helps to design optimal learning data set. In section II, the testing example – 4th order servomechanism is presented. Section III covers the issue of de-noising operation automation. Section IV focuses on the diagnostic method versatility, while section V addresses the diagnostic quality assessment. All presented algorithms were implemented in Matlab 6.1 environment.

II. SYSTEM UNDER TEST

The generic architecture has been so far tested on SUTs, belonging to different technical domains [2,4], e.g. electrical machine or electronic circuit. Another popular field of diagnostic usage is control domain. Practical applications reveal problems of systems with moving parts (friction) [3] or robotic components [5]. They must be analyzed to detect and predict [3] faults. Versatility of the proposed architecture will be confirmed by additional tests on the 4th order servomechanism [6]. It was examined before, using sensitivity analysis [7], therefore comparison between this method and our approach can be performed. It will be also used to illustrate concepts presented in the paper.

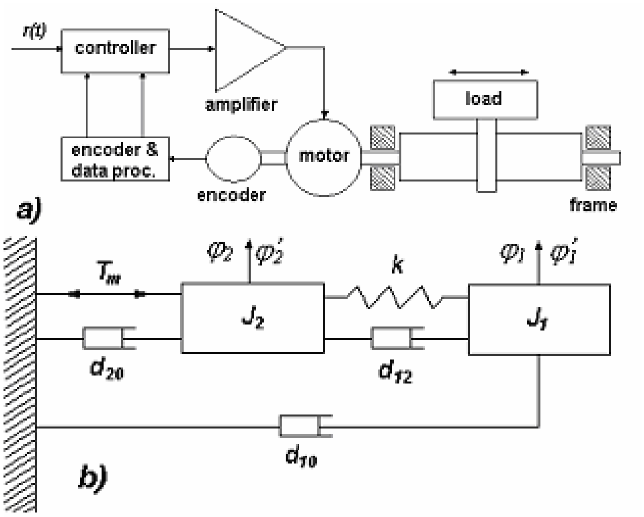

Fig. 2. General layout (a) and mechanical part (b) of the servomechanism

The control system (Fig. 2) contains feedback loop, repeating the output signal on the input. Compensation of elements' changes is in this way assured, making diagnostics difficult.

The SUT controls move of the load. Rotational movement of the motor is translated into the translational movement of the load. The feedback loop ensures keeping the position, velocity and acceleration of motor and load within desired margins. Processed loop data is returned to the controller through the encoder.

The state space equations explaining work regime of the mechanical part (Fig. 2b) are as follows (measured responses are φ_1, φ'_1, φ_2, φ'_2):

$$\begin{aligned}\varphi_1'' &= -\frac{k}{J_1}\cdot\varphi_1 - \frac{d_{10}+d_{12}}{J_1}\cdot\varphi_1' + \frac{k}{J_1}\cdot\varphi_2 + \frac{d_{12}}{J_1}\cdot\varphi_2',\\ \varphi_2'' &= \frac{k}{J_2}\cdot\varphi_1 + \frac{d_{12}}{J_2}\cdot\varphi_1' - \frac{k}{J_2}\cdot\varphi_2 - \frac{d_{12}+d_{20}}{J_2}\cdot\varphi_2' + \frac{C_s}{J_2}\cdot r(t).\end{aligned} \quad (1)$$

where d_{12}, d_{10}, d_{20} are dampings in mechanical part, φ_1, φ'_1, φ''_1 are position [rad], velocity [rad/s] and acceleration [rad/s^2] of rotor motor, φ_2, φ'_2, φ''_2 are position, velocity and acceleration of the load, C_s is servo stiffness, k is transmission stiffness, D_s is servo damping, J_1, J_2 are inertia and rotor loads, T_m is motor torque and $r(t)$ is the input signal. Nominal values of diagnosed parameters are $C_s = 3{,}432$ *Nm/rad*, $D_s = 4.695e\text{-}2$ *Nms/rad*, $d_{12} = 4{,}943e\text{-}5$ *Nms/rad*, $k = 6$ *Nm/rad*, $J_1 = 4{,}489e\text{-}4$ *kgm*^{2}, $d_{10} = 5{,}5e\text{-}4$ *Nms/rad*. Other parameters are kept at their nominal values: $d_{20} = 2e\text{-}3$ *Nms/rad*, $J_2 = 5e\text{-}4$ *kgm*^{2}. Example of the SUT response is in Fig. 3. Stamps are indicated by the crosses on the response pattern. Large number of stamps is recorded for all the responses to deliver data for design of the decision making module.

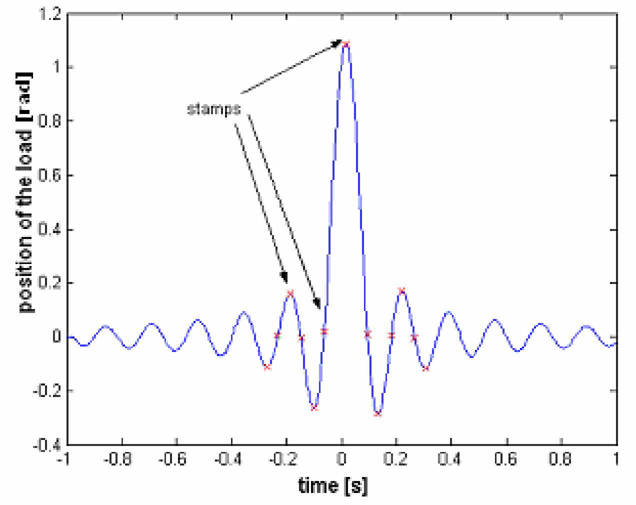

Fig. 3. Response of the servomechanism for the Sa(αt) excitation.

III. AUTOMATION OF THE DE-NOISING PROCEDURE

This section discusses automation of the denoising module. Wavelet transform (WT) was successfully used to clear noisy signals in tested SUTs [2]. It represents signal's characteristics at multiple frequencies (time-scale domain) with high accuracy. The WT de-noising (used for clearing both one [8] and two dimensional [8,9] data arrays) is an iterative process based on the interaction between the signal and two functions: wavelet and scaling (high- and lowpass filters respectively). This rules out wavelets lacking the latter (e.g. Mexican Hat). De-noising separates low and high frequency signal components (called approximations *A(t)* and details *D(t)*, respectively). WT analyses low and high frequency signals [10]:

$$\begin{aligned} A_n(t) &= \sum_{m=-\infty}^{\infty} c_{n,m}\, 2^{-\frac{n}{2}}\phi(t/2^n - m),\\ D_k(t) &= \sum_{m=-\infty}^{\infty} d_{k,m}\, 2^{-\frac{k}{2}}\psi(t/2^k - m),\\ c_{m,n} &= \int_{-\infty}^{+\infty} f(t)\cdot 2^{-\frac{n}{2}}\phi(t/2^n - m)\,dt,\\ d_{k,m} &= \int_{-\infty}^{+\infty} f(t)\cdot\psi(t-m)\,dt. \end{aligned} \quad (2)$$

where ϕ is a wavelet function, ψ is a scaling function. The original signal *f(t)* is reconstructed from (2):

$$f(t) \approx A_n(t) + \sum_{k=1}^{n} D_k(t). \quad (3)$$

We use (2) and (3) to decompose original, noisy signal into not important details and approximation, bearing crucial information about the servomechanism's state. The main automation issue is the wavelet type selection for noise filtering and setting number of iterations for the operation. The former is obtained using similarity between the reference and de-noised signals. For every wavelet candidate denoising operation on the signal is performed. Then, the measure of the resemblance, based on the correlation between the reference and de-noised signal is determined. Firstly, for the reference signal (obtained from the nominal model) autocorrelation function is calculated. Then, for the de-noised signal, crosscorrelation with the reference signal is determined. The minimum absolute value of the difference between summed samples of auto- and crosscorrelation functions is the wavelet selecting criterion.

The number of de-noising iterations is set by using the entropy criterion to decide when the signal contains no more relevant noisy coefficients. In every iteration the threshold entropy (with the threshold ε) for details is calculated:

$$E(s)=\sum_{i=1}^{n}\eta(s_i),\quad \eta(s_i)=\begin{cases}1 & if \quad |s_i|>\varepsilon\\ 0 & if \quad |s_i|\le\varepsilon\end{cases} \tag{4}$$

where s_i is a sample of D_k. Searching for the threshold to determine proper decomposition level usually is based on the power of the noise analysis [8, 11]. In our approach the threshold ε is also determined using the noise power estimation. We use signal-to-noise ratio (SNR) to find how much signal is affected by the noise. Firstly, dynamic range of the analyzed response is calculated. It is used to determine parts of the signal the most vulnerable to the noise. They are signal samples differing no more than five percent from the minimum signal value. For them, average SNR is calculated to find the threshold:

$$\varepsilon = 0.35\cdot\frac{1}{e^{(SNR/20)}} \tag{5}$$

Example

Application of the formula (5) to the vulnerable parts of the noisy response of the servomechanism (similar to Fig. 3) resulted in average SNR=11,41dB. Then, threshold ε = 0,6431 rad. Using this, the de-noising procedure stopped at the fourth iteration. When ε was too large, for example 2 rad, algorithm stopped after two iterations, leaving too many noisy coefficients. For all the experiments conducted, the number of iterations was never greater than 5 (depending on the noise level).

Proper selection of ε makes possible accurate extraction of stamps. Too small value of ε results in too many iterations misshaping the signal, which leads to erroneous diagnosis. Too large threshold value leaves multiple noisy coefficients intact, making stamps extraction difficult.

IV. FAULT DETECTION AND LOCATION

Decision module is designed to work autonomously. The automation refers not only to the fault detection and location, but also to self-adjustment, based on the set of learning examples (acquired from tested SUT model). Examples provide signal responses, from which stamps are extracted. Matrix ***A*** containing experiments' responses is created, with stamps from single experiment in every row. Every column contains particular stamp's values for different experiments. Information about the fault source in the experiment is stored in the last column of ***A***. It is a code of two digits and a sign, informing about the fault source (number of the faulty element) and measure of its deviation from nominal value. The latter discretizes continuous space of the elements' values. It is represented by the number from the set {-2, -1, 0, 1, 2}. In the simplest form of the decision making procedure only three elements, i.e. {-1, 0, 1}, suffice. However, additional states (referring to larger deviations) often increase diagnostic efficiency, as in this case. For example, code "-21" states that "element number 2 has too small value", while "11" means that "element 1 has too large value". Code "0" is for fault-free state. Although data in ***A*** contain only experiments for single faults, the architecture is expected to detect multiple faults also. Two simultaneous codes would be then indicated. Expert's knowledge required for the fuzzy logic design is obtained from ***A*** using decision trees (similarly to [12] Recursive algorithm [2] of the tree generation extracts the important data from the matrix *A*. The tree structure contains two kinds of the vertices. The former are the nodes holding the tests (the threshold value compared to the value of one of the example's stamps). The latter are the leaves (terminal nodes), storing the fault code. Based on that data and decision tree structure, components of the fuzzy logic are created: input and output membership functions (MFs) and rules. The input MF sets are created based on the tree nodes. The tests considering the same stamp are extracted to design one input MF set. Trapezoidal functions are used here. The output MF sets are created based on the fault codes. Number of these MFs in a set depends on the number of the codes concerning particular element. Triangular functions are used here. The path leading from the highest tree node (root) to the leaf is converted into the fuzzy logic rule. In our approach it contains numbers of the MFs, which must have membership degree greater than zero to make the rule active.

Selection of the fuzzy logic enables us to consider elements' tolerances and uncertainties related to the de-noising operation and decision making. They are addressed by the input MFs in the rule detecting "fault-free" state. Their flat fragments determine the tolerance regions. Coordinates of these functions are calculated based on the values of the stamp in the node threshold [4]. Multiple experiments for the "fault-free" state in ***A*** allow determination of the "safe area" in the stamps' variable space. Additionally, function's transitional regions (slopes) help to avoid false alarm. Example of the input MF set is in Fig. 4. The MF determining the tolerance region is grayed. Both transitional regions and flat fragment coordinates are parametrized, so they can be adjusted for the particular SUT.

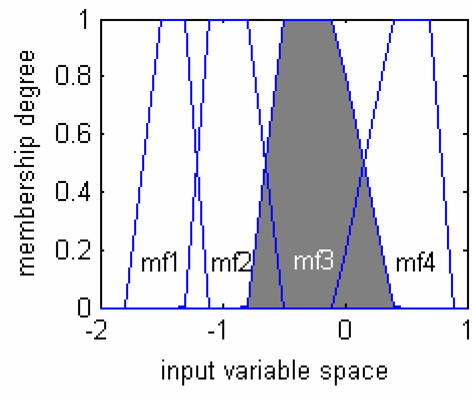

Fig. 4. Exemplary input MF set.

Although the decision tree structure imposes only single decision (fault code), fuzzy logic allows multiple detections. Every set of the output MFs (responsible for the information about the particular element's state) is processed independently on the others. Result of the analysis is the set of the numbers – every one related to certain fault source. Example of the output MF set is in Fig. 5. Every function in a set is related to the fault code (in Fig. 5 there are codes for the second element). Information about the element's state is the number within [-2,6 2,6] range (function coordinates are calculated as in [4]), where values close to zero indicate the element is in the nominal state (function "0" is active).

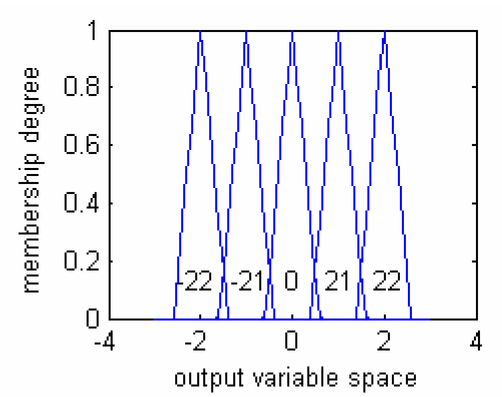

Fig. 5. Exemplary output MF set.

Important modification driven by the multiple fault detection is introduced, i.e. fuzzy logic based on the multiple decision trees [4]. Fuzzy logic based on the single decision tree binds rules and firing more than one of them is difficult (though possible [4]). Multiple trees allow firing many rules at the same time, making multiple faults detection feasible. The idea is to create one decision tree for every fault source and then bind them into one fuzzy logic structure. Two solutions are possible. The first one divides learning experiments matrix into submatrices, one for every fault source. For every submatrix the decision tree is designed and subsequently fuzzy logic module created. This approach gives acceptable results for the SUTs, where multiple faults are easily distinguishable in responses (as for the DC-motor [4]). When responses are smooth and stamps for simultaneous faults are hard to determine, the second approach gives better results. This approach was applied for the servomechanism example. Multiple decision trees are created based on the whole matrix. For every fault source codes in the last column are altered. All faults other than currently selected get common code to distinguish them from the element, for which the tree is generated. This way the decision tree sorts examples with only few codes: these referring to the current fault source and one for the rest of the elements. The procedure is repeated for every fault source. Design of the MFs is performed the same way as explained above. This time stamps' values and fault codes are extracted from all of the trees to obtain one fuzzy logic module.

Example: This example shows transformation of the learning matrix ***A*** of the servomechanism from Fig. 2 to design multiple decision trees. For the clarity only fragment of the matrix is considered. Assume the last column of ***A*** has the form presented below. The trees created for every element are based on the modified columns (all but current element get '99' code). For the exemplary 1st tree we have three faults classified: 11, 12 and 99. This way six decision trees (one for every element) were created for the servomechanism.

Original codes	Codes for the 1st tree	Codes for the 2nd tree	Codes for the 3rd tree
11	11	99	99
12	12	99	99
-21	99	-21	99
22	99	22	99
-32	99	99	-32
-31	99	99	-31

Servomechanism is difficult to diagnose because of the feedback loop which suppresses changes in the object responses caused by the elements values deviations. This makes parametric faults detection impossible. Through experiments we found that the optimal diagnostic results require excitation signal with uniform spectrum in the whole servomechanism's operating frequency interval. Among unit step, sinusoid, polynomial signals and $Sa(\alpha t)$ function, only the latter fulfills the condition (in [7] sensitivity analysis also confirmed that only this function useful). The spectrum band depends on the α parameter. The highest operating frequency of the SUT is 8 Hz. Above this constraint the SUT is unable to repeat input signal on the output. To make diagnostics possible, we decided to excite the SUT with *Sa* function of 15 Hz frequency. The best results were obtained by the combination of the fast excitation and opening feedback loop. SUT becomes then an integrator with impulse response as in Fig. 6. For the fault simulation, the elements' values were changed within range ensuring stability of the SUT.

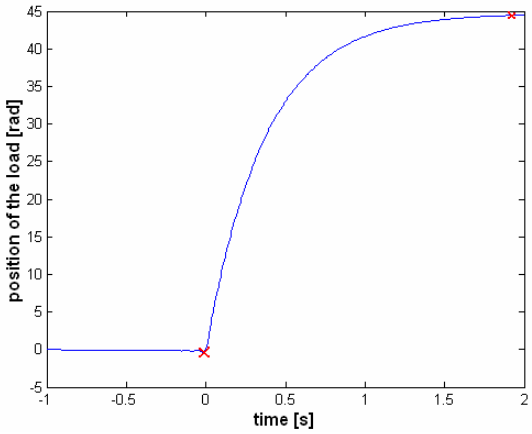

Fig. 6. Impulse response of the servomechanism after opening feedback loop.

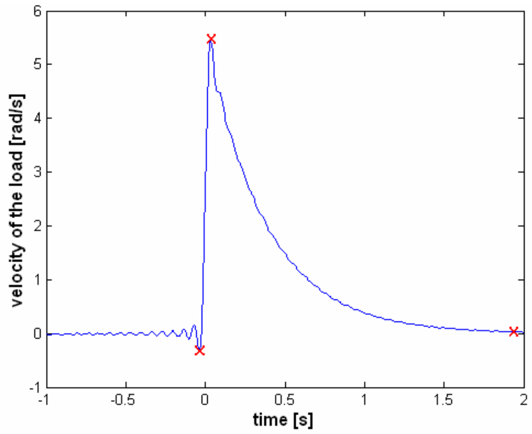

Fig. 7. Response of φ'_1 and φ'_1 in the open loop configuration

To examine the method two matrices were prepared: learning matrix for design of the decision making module and testing matrix to determine diagnostic quality. Both were of the same size (42 rows and 79 columns for the closed loop and 42 rows with 21 columns for the open loop servomechanism), containing identical number of examples for every faulty element. The diagnostic quality was determined by the fraction between the correct detections and all analyzed experiments. SUT was analyzed after closing and opening feedback loop. The results for the optimal diagnostic configuration (open loop and fast excitation) are in Table I (single faults) and Table II (multiple faults). For the closed loop servomechanism as stamps (total seventy eight) we selected maximum values of φ_1, φ_1', φ_2, φ_2' and instances of their zero crossings (Fig. 3). The stamps for the open loop servomechanism (total twenty) are in Fig. 6 and Fig. 7. Results are output MFs values multiplied by 100 for a better readability (results are between –200 and 200 for too small and too large elements' values). Bold fonts indicate correct diagnostic outcomes, while italics are for errors.

TABLE I
RESULTS OF SERVOMECHANISM SINGLE FAULT DETECTION

Faults	**Output membership functions responses**					
	C_s	**D_s**	**d_{12}**	**K**	**J_1**	**d_{10}**
Cs=5e-1	**-1,81E+02**	2,36E-15	2,36E-15	-1,16E-17	2,36E-15	2,36E-15
Cs=2	**-4,64E+01**	1,96E-15	1,96E-15	-9,11E-16	1,96E-15	*7,66E+01*
All OK	-1,25E-03	2,50E-15	1,92E+00	3,11E-15	2,49E-15	2,49E-15
Ds=8e-3	-2,90E-15	**-1,95E+02**	2,36E-15	-1,16E-17	2,36E-15	2,35E-15
Ds=7e-2	-2,90E-15	**1,00E+02**	2,36E-15	-1,16E-17	2,36E-15	2,36E-15
d12=2e-5	-2,20E-03	2,45E-15	*2,35E+00*	2,23E-01	2,45E-15	2,45E-15
k=3,9	-2,90E-15	2,36E-15	2,35E-15	**-8,07E+01**	2,35E-15	2,35E-15
k=6,9	-2,90E-15	2,36E-15	2,32E-15	**8,06E+01**	2,35E-15	2,35E-15
J1=5e-5	-2,90E-15	2,36E-15	2,36E-15	-1,16E-17	**-1,81E+02**	2,36E-15
J1=3,6e-4	-2,90E-15	*-1,81E+2*	2,36E-15	-1,15E-17	2,36E-15	2,36E-15
J1=7,1e-4	-2,90E-15	2,36E-15	2,12E-15	-1,16E-17	**1,00E+02**	2,36E-15
d10=1e-4	-2,90E-15	2,36E-15	2,36E-15	-1,16E-17	2,11E-15	**-1,81E+02**
d10=3e-4	-2,90E-15	2,36E-15	2,37E-15	-1,16E-17	2,36E-15	**-1,00E+02**

TABLE II
RESULTS OF SERVOMECHANISM MULTIPLE FAULT DETECTION.

Faults	**Output membership functions responses**					
	C_s	**D_s**	**d_{12}**	**k**	**J_1**	**d_{10}**
Cs = 1.5 Ds = 2e-2	**-6.23e+1**	**-6.15e+1**	-2.89e-15	-3.46	2.35e-15	2.35e-15
Cs = 4 d12 = 7e-5	**2.73e+1**	2.36e-15	**-3.94e+1**	2.60e-1	*-6.07e+1*	2.35e-15
Cs = 6 J1 = 9e-5	*-2.89e-15*	2.36e-15	1.17e-12	*2.70e+1*	**-6.30e+1**	-4.27e-15
Ds = 7e-2 d10 = 1e-4	*2.75e+1*	2.36e-15	2.35e-15	***-2.70e+1***	1.67e-13	**-6.31e+1**
Cs = 4.5 k = 9	**2.74e+1**	2.36e-15	6.47e-15	**-2.70e+1**	2.37e-15	*-6.30e+1*
d12 = 4e-5 J1 = 5e-5	-2.89e-15	-5.84	*-3.19e-15*	*2.70e+1*	**-6.30e+1**	-5.89e-15

The results for the SUT tested using integral sensitivity analysis [7] varied between 70 and 85 percent, depending on the particular element, except d_{12}, which formed ambiguity group with C_s. Results obtained by our architecture for the SUT with closed loop were comparable, giving 75 percent quality. Opening loop increased diagnostic efficiency, giving 90 percent of the correct decisions. It also confirmed difficulty in detecting changes of d_{12} element. It has large tolerance - over 50 percent [6], so its changes are hardly detectable in the signal responses. Results for the architecture are better than in the compared method. The denoising method applied by us is also more universal (it tackles wide classes of noise, where integral sensitivity has problems, for example, with the gaussian noise). For the multiple fault detection output MFs are more active with one or two outputs pointing at the fault source (values above 25). Most faults are detected correctly, in some cases one masks another. The method versatility issue is addressed.

V. DIAGNOSTIC QUALITY ASSESSMENT

Flexibility of the architecture allows for different methods and algorithm to be used for the particular stages (Fig. 1). Introduction of efficient method assessing the diagnostic quality is required to compare methods. The learning and testing data sets are matrices of equal size. Every row of such a matrix contains measurement data (stamps) for one experiment. Because SUT model simulation is costly, size of the matrix must be minimum and provide acceptable diagnostics quality. A coefficient considering size of the set is needed. There are coefficients proposed and applied to industrial processes assessment [13] (another option, especially for frequency measurements, is the usage of the probability distribution function [14]). The simplest one is the ratio of correct fault detections with respect to all detections. We modify it by adding exponential factor:

$$Q = \frac{|R_c|}{|R|} \cdot 2 \cdot \exp(-|R|) \cdot 100\%, \tag{6}$$

where $|R|$ is the number of the fault detections (experiments) and $|R_c|$ is the number of the correct detections. Due to the exponential factor the smaller data sets for learning are preferred. The optimal experiments' number must be determined for every SUT and for the servomechanism it was 7.

Another problem is the choice of the experiments for learning and testing data sets. They must reflect SUT's work regime and be representative sample of the common faults. Completeness of the learning data influences generalization ability of the decision trees and quality of the generated module. Applying different testing matrices to the SUT diagnostics helps to determine optimal sets of experiments. This should be investigated further.

VI. CONCLUSIONS

The paper addressed traits and problems of the automated diagnostic architecture automation, versatility and quality assessment. The approach proposed for the diagnostics utilizes existing artificial intelligence and machine learning methods and algorithms which can be automated. Versatility of the method was confirmed by numerous SUT analyses. Examination of the servomechanism gave satisfying results and proved usefulness of the proposed architecture. It is generic enough to fit characteristic traits of different SUTs. Particular methods, i.e. WT and decision tree-based fuzzy logic are flexible enough to de-noise signals and detect faults in different objects, including multiple fault location. Disadvantage of the wavelet selection and fuzzy logic is low speed, crucial for on-line applications. Therefore simple modules are preferred. Other methods for fuzzy logic design (such as rough sets) should be examined. Quality assessment proposed for the task is based on existing coefficients, but considers specific aspects of our approach. It prefers small learning set giving the best diagnostic results.

ACKNOWLEDGMENT

This work was supported by the National Reseach Committee of Poland, grant No. 3 T10C 043 28.

REFERENCES

[1] A. Bernieri, G. Betta, and C. Liguori, "On-Line Fault Detection and Diagnosis Obtained by Implementing Neural Algorithms on a Digital Signal Processor," *IEEE Trans. Instr & Meas*, Vol. 45, No. 5, 1996, pp. 894-899.

[2] P. Bilski, and J. Wojciechowski, and W. Brygilewicz, "Diagnostics of Mechatronic Systems Using Fuzzy Logic and Wavelet Transform," *Proc. ECCTD'03 Conf.*, Aug. 2003, Kraków, Poland, pp. III-377-380.

[3] E. G. Papadopoulos, G. C. Chasparis, "Analysis and Model-based Control of Servomechanisms with Friction," *Proc. IROC'02 Conf., Sept. 30 - Oct. 5, 2002, vol. 3, pp. 2109-2114.*

[4] P. Bilski, and J. Wojciechowski, "Automated Diagnostic System in Analog Testing," *Proc. ICSES'04 Conf.*, Sept., 2004 Poznań, Poland, pp. 385-388.

[5] Y. S. Kim, H. S. Kim, W. H. Kwon, "A Design of a Network-based Robot Control System," *Proc 3rd ACC Conf., July, 2000, Shanghai, pp. 2229-2234.*

[6] H. H. B. Roelofs, C.J. Heuvelman, "Optimization of Design Tolerances of Servomechanisms," *Annals of the CIRP*, Vol. 41/1/1992, pp. 467-471.

[7] W. Brygilewicz, ""Fault Diagnosis of Dynamic Analog System in the Presence of Noise," Ph. D. Dissertation, Warsaw, 1997, pp. 92-96.

[8] S. Sardy, P. Tseng, A. Bruce, "Robust Wavelet Denoising," IEEE Trans. Signal Processing, Vol. 49, No. 6, June 2001, pp. 1146-1152.

[9] A. M. Wink, J. B. T. M. Roerdink, "Enchancing Functional Neuroimages: Wavelet Denoising as an Alternative to Gaussian Smoothing," *Proc. ICCVG'02 Conf., Sept. 2002, Zakopane, Poland, pp. 787-792.*

[10] A.N. Akansu, M.V. Tazebay, D. Sarnoff, M.J. Medley, and O.K. Das, "Wavelet and Subband Transforms: Fundamentals and Communication Applications," *IEEE Communications Magazine*, Dec. 1997, pp. 104-115.

[11] J. Cesar, S. O. Lyrio, "Wavelet Denoising of Gravity Gradiometry Data," *SEG International Exposition and 71st Annual Meeting, Sept., 2001, San Antonio, USA.*

[12] J. R. Quinlan, "Generating Production Rules from Decision Trees," *Proc IJCAI Conf.*, Aug. 1987, Milan, Italy, pp. 304-307.

[13] J. M. Kościelny, "Diagnostics of Automated Industrial Processes," Akademicka Oficyna Wydawnicza EXIT, Warsaw, 2001, pp. 372-381 (in Polish).

[14] L. M. Gelman, V. Giurgiutiu, I. V. Petrunin, "Crack Diagnosis via Fourier Transform: Real and Imaginary Components vs. Power Spectral Density," *Proc. 3rd IWSHM Workshop, Sept., 2001, Stanford, USA, pp. 1382-1390.*