Proceedings of the 17th World Congress
The International Federation of Automatic Control
Seoul, Korea, July 6-11, 2008

On fault detection under soft computing model uncertainty

Józef Korbicz * **Marcin Witczak** *

* *University of Zielona Góra, Poland, ul. Podgórna 50, Zielona Góra (e-mail: {j.korbicz, m.witczak}@issi.uz.zgora.pl)*

Abstract: The paper deals with the problems of robust fault detection using soft computing techniques, in particular neural networks (Group Method of Data Handling, GMDH), multi-layer perceptron), and neuro-fuzzy networks (Takagi-Sugeno model). The model based approach to Fault Detection and Isolation (FDI) is considered. The main objective is to show how to employ the bounded-error approach to determine the uncertainty of the neural and fuzzy models. It is shown that, based on soft computing models uncertainty defined as a confidence range for the model output, adaptive thresholds can be defined. Finally, the presented approaches are tested on a servoactuator being an FDI benchmark in the DAMADICS project. *Copyright©2007 IFAC*

Keywords: fault detection, robustness, adaptive threshold, neural networks, neuro-fuzzy networks

1. INTRODUCTION

Fault diagnosis becomes an important issue in modern control systems due to the increasing complexity of such systems. Early diagnosis of faults that might occur in the supervised process renders it possible to perform preventing actions. Moreover, it allows avoiding heavy economic losses involved in stopped production, the replacement of elements, units and parts, etc. Among many known FDI methods and approaches (Chen and Patton [1999], Gertler [1998], Iserman [2006], Korbicz et al. [2004], Korbicz [2006], Patton et al. [2000]), the most efficient fault-diagnostic strategy is the so-called model based approach. Model based fault detection is on the use of mathematical or artificial intelligence models (Patton et al. [2005], Korbicz and Cempel [1993], Witczak [2007]) or a combination of both. The main difficulty with applying mathematical models (Hui and Zak [2005]) is the fact that imprecise models are generally available independently of identification methods and techniques applied (Bubnicki [2004], Nelles [2001], Walter and Pronzato [1996]).

As model uncertainty and disturbances are usually difficult to eliminate, there is a need for developing robust fault detection algorithms. The robustness of a fault detection system means that it must be only sensitive to faults, even in the presence of model-reality differences (Chen and Patton [1999], Korbicz et al. [2004], Witczak et al. [2006], Witczak [2007]). One of the approaches to the robustness problem, known as the passive one, enhances the robustness of the fault detection system at the decision-making stage, mainly using an adaptive threshold (Emami-Naeini et al. [1988], Frank [2002]). Contrary to active approaches (Chen and Patton [1999]), the adaptive threshold based passive approach does not rely on eliminating the effect of model uncertainty in the residual. Indeed, adaptive threshold based techniques rely on propagating model uncertainty to the residual, and then bounding the resulting residual uncertainty.

The main objective of this paper is to present recent developments in robust model based fault detection using soft computing methods and techniques. In order to overcome this problem, it is necessary to describe model uncertainty of artificial intelligence models. To solve this problem, the so-called Bounded Error Approach (BEA) and its extension – the Outer Bounding Ellipsoid (OBE) algorithms (Milanese et al. [1996], Walter and Pronzato [1996]) – are applied. The paper is organized as follows: Section 2 describes the concept of robust model based fault detection with model uncertainty. Section 3 outlines the idea of robust fault detection with the Takagi-Sugeno fuzzy model, GMDH and MLP neural networks designed with the bounded-error parameter estimation technique. In all cases the algorithms for computing the adaptive threshold are presented. The final section presents a comprehensive study regarding the application of the approaches considered to the DAMADICS benchmark problem.

2. UNCERTAINTY OF SOFT COMPUTING MODELS

A common disadvantage of analytical approaches to FDI is the fact that a precise mathematical model of the diagnosed system is required. An alternative solution can be obtained with soft computing techniques (Calado et al. [2001], Patton and Korbicz [1999], Rutkowski [2004], Rutkowska and Zadeh [2000]). Bellow we focus on the problem of designing selected soft computing models.

2.1 Uncertainty of the Takagi-Sugeno fuzzy model

The structure of the Takagi-Sugeno system could be presented in the form of a layered topology similar to the neural network (Brown and Harris [1994]). However, knowl-

* The work was financed as a research project with the science funds for years 2007-2010

edge coded in this structure could be viewed in the form of fuzzy rules (Babuska [1998], Piegat [2001]):

$$R_i: \ IF \ \boldsymbol{x} \ is \ \boldsymbol{A}_i \ THEN \ \tilde{y}_i = \boldsymbol{z}_i^T \boldsymbol{\theta}_i, \tag{1}$$

where

- $\boldsymbol{x}$ is the vector of global network inputs
- $\boldsymbol{A}_i$ is the multivariate fuzzy set
- $\tilde{y}_i$ is the scalar output of the rule
- $\boldsymbol{z}_i$ is the vector of local linear system inputs
- $\boldsymbol{\theta}_i$ is the vector of local linear system parameters, and i is the index of the rule.

Fuzzy sets usually have Gaussian membership functions. The global output of the neuro-fuzzy network is a composition of the responses of all rules:

$$\tilde{y} = \frac{\sum_{i=1}^{n} \mu_k \tilde{y}_i}{\sum_{i=1}^{n} \mu_i}, \tag{2}$$

where $\tilde{y}$ is the global output of the network, μ_i is the membership degree achieved for i-th rule, $\tilde{y}_i$ is the output of the i-th rule (local linear system), n is the number of rules. It is worth noticing that the number of rules determines the number of local linear models, which are responsible for piecewise local linear approximation of the non-linear system. The dynamical T-S network could be done by introducing into the input vector $\boldsymbol{z}_i$ the delayed inputs $\boldsymbol{u}_i$ of the local model and the delayed output of the local output $\tilde{y}_i$, i.e., $\boldsymbol{z}_i = [u_i(k), u_i(k-1), \ldots, u_i(k-n_a), \tilde{y}_i(k-1), \tilde{y}_i(k-2), \ldots, \tilde{y}_i(k-n_b)]$. To settle the problem of parameter and model uncertainty estimation, Kowal [2005] proposed to use the BEA approach. The choice of such a strategy is not accidental and it is clearly justified by a number of theoretical and practical reasons described in Kowal and Korbicz [2007].

Let us consider the following Takagi-Sugeno fuzzy model:

$$\tilde{y}(k) = \sum_{i=1}^{n} \phi_i(k) \tilde{y}_i(k), \tag{3}$$

where $\tilde{y}_i(k)$ is the output of the i-th rule and

$$\phi_i(k) = \frac{\mu_i(k)}{\sum_{j=1}^{n} \mu_j(k)}. \tag{4}$$

The model described by (3) could be viewed as a system linear in parameters:

$$\tilde{y} = \boldsymbol{x}^T(k)\boldsymbol{\theta}, \tag{5}$$

where

$$\boldsymbol{x}(k) = \begin{bmatrix} \phi_1(k)\boldsymbol{z}_1(k) \\ \phi_2(k)\boldsymbol{z}_2(k) \\ \vdots \\ \phi_n(k)\boldsymbol{z}_n(k) \end{bmatrix}, \quad \boldsymbol{\theta} = \begin{bmatrix} \boldsymbol{\theta}_1 \\ \boldsymbol{\theta}_2 \\ \vdots \\ \boldsymbol{\theta}_n \end{bmatrix},$$

if the parameters of fuzzy sets are treated like constant values. Let us define the output error $\varepsilon(k)$:

$$\varepsilon(k) = y(k) - \boldsymbol{x}^T(k)\boldsymbol{\theta}, \tag{6}$$

where $y(k)$ is the output of the system. In the case of the BEA, it is assumed that errors lie between a priori given bounds (Milanese et al. [1996]):

$$\varepsilon^m(k) \leq \varepsilon(k) \leq \varepsilon^M(k). \tag{7}$$

Let us assume that

$$\varepsilon^M(k) = \varepsilon, \ \varepsilon^m(k) = -\varepsilon. \tag{8}$$

Thus the feasible set of parameters for N data points is given by the following expression

$$\begin{aligned} \boldsymbol{\Theta} = \big\{ &\boldsymbol{\theta} \in \mathbb{R}^{n_\theta} \mid y(k) + \varepsilon \leq \boldsymbol{x}^T(k)\,\boldsymbol{\theta} \leq y(k) - \varepsilon; \\ &k = 1, \ldots, N \big\}. \end{aligned} \tag{9}$$

Then the confidence interval for the system output is described by the inequality (Kowal and Korbicz [2007]):

$$\boldsymbol{x}^T(k)\,\boldsymbol{\theta}^m(k) + \varepsilon \leq y(k) \leq \boldsymbol{x}^T(k)\,\boldsymbol{\theta}^M(k) + \varepsilon, \tag{10}$$

where

$$\boldsymbol{\theta}^M(k) = \arg\max_{\boldsymbol{\theta} \in \boldsymbol{\Theta}} \boldsymbol{x}^T(k)\,\boldsymbol{\theta}, \tag{11}$$

$$\boldsymbol{\theta}^m(k) = \arg\min_{\boldsymbol{\theta} \in \boldsymbol{\Theta}} \boldsymbol{x}^T(k)\,\boldsymbol{\theta}. \tag{12}$$

This algorithm requires to determine the set of all vertices $\mathbb{W}$ of the convex polyhedron $\boldsymbol{\Theta}$. The minimum and maximum values for the parameters considered are determined using linear programming techniques (Milanese et al. [1996]). The obtained confidence interval can be directly used to calculate an adaptive threshold for the residual $r(k) = y(k) - \tilde{y}(k)$. Finally, the adaptive threshold is described by the following inequality:

$$\begin{aligned} &\boldsymbol{x}^T(k)\boldsymbol{\theta}^m(k) + \varepsilon^m(k) - y(k) \leq r(k) \\ &\leq \boldsymbol{x}^T(k)\boldsymbol{\theta}^M(k) + \varepsilon^M(k) - y(k). \end{aligned} \tag{13}$$

The presented approach does not take into account the fact that not only the output variable $y(k)$ is uncertain but also all input variables $\boldsymbol{x}(k)$ can be uncertain. The problem of computing the feasible set of parameters when some or all explanatory variables, as well as the output, are uncertain is usually called the Error-In-Variables (EIV) problem. The study of this problem can be found in Milanese et al. [1996]. Moreover, in Kowal and Korbicz [2007], the EIV parameter-bounding algorithm is adapted for use with the Takagi-Sugeno fuzzy model in order to compute the adaptive threshold.

2.2 Dynamic GMDH networks and their uncertainty

The idea of the GMDH (Ivakhnenko [1971]) is based on replacing the complex model of the process with partial models (neurons) by using the rules of variable selection. As usual, partial models have a small number of inputs $u_i(k)$, $i = 1, 2, \ldots, m$, and are implemented by GMDH neurons. The synthesis process of the GMDH network (Farlow [1984], Pham and Xing [1995]) is based on iterative processing of a sequence of operations. This process leads to the evolution of the resulting model structure in such a way as to obtain an approximation of the optimal degree of model complexity. In a general form, the GMDH neural network output can be written as

$$\tilde{y}(k, \boldsymbol{\theta}) = g\left(\boldsymbol{\theta}_1^{(1)}, \ldots, \boldsymbol{\theta}_{n_1}^{(1)}, \ldots, \boldsymbol{\theta}_1^{(L)}, \ldots, \boldsymbol{\theta}_{n_L}^{(L)}\right), \tag{14}$$

where $g(\cdot)$ stands for the neural network structure (Witczak et al. [2006]), L is the number of layers of the GMDH model and n_l is the number of neurons in the l-th layer. Each GMDH neuron has the following structure:

$$\tilde{y}_n^{(l)}(k) = \xi\left(\left(\boldsymbol{z}_n^{(l)}(k)\right)^T \boldsymbol{\theta}_n^{(l)}\right), \tag{15}$$

where $\tilde{y}_n^{(l)}(k)$ stands for the neuron output ($l = 1, \ldots, L$ is the layer number, $n = 1, \ldots, n_l$ is the neuron number in the l-th layer) corresponding to the k-th input signal $\boldsymbol{u}(k)$ of the system, $\xi(\cdot)$ denotes a non-linear invertible activation function, $\boldsymbol{z}_n^{(l)}(k) = f([u_i^{(l)}(k), u_j^{(l)}(k)]^T)$, $i, j = 1, \ldots, n_u$ (n_u being the number of inputs in the i-th layer) are the regressor vectors with $f(\cdot)$ being an arbitrary bivariate vector function, and $\boldsymbol{\theta}_n^{(l)}$ are the parameter vectors.

An outline of the GMDH algorithm can be as follows (Mrugalski [2004], Pham and Xing [1995]):

Step 1: Determine all neurons in the first layer (estimate their parameter vectors $\boldsymbol{\theta}_n^{(l)}$ with the training data set $\mathcal{T}$) whose inputs consist of all possible couples of input variables, i.e., $(n_u - 1)n_u/2$ couples (neurons).

Step 2: Using a validation data set $\mathcal{V}$, not employed during the parameter estimation phase, select several neurons which are best fitted in terms of the chosen criterion.

Step 3: If the termination condition is fulfilled (the network fits the data with desired accuracy or the introduction of new neurons did not induce a significant increase in approximation abilities of the neural network), then *Stop*, otherwise use the outputs of the best-fitted neurons (selected in *Step 2*) to form the input vector for the next layer, and then go to *Step 1*.

To obtain the final structure of the network, all unnecessary neurons are removed, leaving only those which are relevant to the computation of the model output. The feature of the above algorithm is that the techniques for parameter estimation of linear-in-parameter models can be used during the realisation of Step 1. Indeed, since $\xi(\cdot)$ is invertible, the neuron (15) can relatively easily be transformed into a linear-in-parameter one.

Similarly as in Section 2.1, while applying the least-square method to parameter estimation of neurons, a set of restrictive assumptions has to be satisfied. The first, and the most controversial, assumption is that the structure of the neuron is the same as that of the system (no structural errors). In the case of the GMDH neural network, this condition is extremely difficult to satisfy. Indeed, neurons are created based on two input variables selected from $\boldsymbol{u}$ and hence it is impossible to eliminate the structural error. Another assumption concerns transformation with $\xi^{-1}(\cdot)$. A more realistic approach is to assume that errors lie between a priori given bounds.

Let us consider the following system:

$$y(k) = \left(\boldsymbol{z}_n^{(l)}(k)\right)^T \boldsymbol{\theta}_n^{(l)} + \varepsilon_n^{(l)}(k). \tag{16}$$

The problem is to obtain the parameter estimate vector $\hat{\boldsymbol{\theta}}_n^{(l)}(k)$, as well as the associated parameter uncertainty required to design a robust fault detection system. In order to simplify the notation, the index $_n^{(l)}$ is omitted in further descriptions.

As proposed in (Mrugalski et al. [2005], Witczak et al. [2006], Korbicz and Mrugalski [2008]), the structure, parameters and modelling uncertainty of the GMDH net can be determined with the bounded-error approach, presented in the previous subsection. As a consequence, any parameter vector contained in $\boldsymbol{\Theta}$ is a valid estimate of $\boldsymbol{\theta}$. In practice, the centre (in some geometrical sense) of $\boldsymbol{\Theta}$ is chosen as the parameter estimate $\hat{\boldsymbol{\theta}}$, e.g.,

$$\hat{\theta}_i = \frac{\theta_i^m + \theta_i^M}{2}, \quad i = 1, \ldots, n_\theta, \tag{17}$$

where

$$\theta_i^m = \arg\min_{\theta \in \boldsymbol{\Theta}} \theta_i, \quad i = 1, \ldots, n_\theta, \tag{18}$$

$$\theta_i^M = \arg\max_{\theta \in \boldsymbol{\Theta}} \theta_i, \quad i = 1, \ldots, n_\theta. \tag{19}$$

Using the methodology described above it is possible to obtain the parameter estimate $\hat{\boldsymbol{\theta}}$ and the associated feasible parameter set $\boldsymbol{\Theta}$. However, from the practical point of view, it is more convenient to obtain model output uncertainty, i.e., the interval in which the "true" model output $\tilde{y}(k)$ can be found. This kind of knowledge makes it possible to obtain an adaptive threshold, and hence to develop a fault diagnosis scheme that is robust to model uncertainty.

If there is no error in the regressor, then the problem of determining system output uncertainty can be solved as follows:

$$\boldsymbol{z}^T(k)\boldsymbol{\theta}^m(k) \leq \boldsymbol{z}^T(k)\boldsymbol{\theta} \leq \boldsymbol{z}^T(k)\boldsymbol{\theta}^M(k), \tag{20}$$

where

$$\boldsymbol{\theta}^m(k) = \arg\min_{\boldsymbol{\theta} \in \boldsymbol{\Theta}} \ \boldsymbol{z}^T(k)\boldsymbol{\theta}, \tag{21}$$

$$\boldsymbol{\theta}^M(k) = \arg\max_{\boldsymbol{\theta} \in \boldsymbol{\Theta}} \boldsymbol{z}^T(k)\boldsymbol{\theta}. \tag{22}$$

The computation of (21) and (22) is realised by multiplying parameter vectors corresponding to all vertices belonging to $\boldsymbol{\Theta}$ by $\boldsymbol{z}^T(k)$.

The above technique can easily be adapted for parameter and uncertainty estimation of the non-linear neuron model (15) (Witczak [2007]). Moreover, the presented technique can also be adapted to the so-called error-in-variable case, i.e. when the regressor is not precisely known but there exist its lower and upper bounds.

In a similar way as was done for the T-S fuzzy model (1), the confidence interval (20) can be directly used to calculate the adaptive threshold as follows:

$$\begin{aligned} &\boldsymbol{z}^T(k)\boldsymbol{\theta}^m(k) + \varepsilon^m(k) - y(k) \\ &\qquad \leq r(k) \leq \boldsymbol{z}^T(k)\boldsymbol{\theta}^M(k) + \varepsilon^M(k) - y(k). \end{aligned} \tag{23}$$

For the situation when an error in the regressor of the GMDH model is taken into account, the adaptive threshold is considered by Witczak [2007].

Finally, it should be pointed out that the main argument behind using a GMDH net is the fact that linear parameter estimation based techniques can be used for its design purposes. This means that the resulting parameter confidence set (or the feasible parameter set) is precisely described. This is in contrast with other approaches, e.g., multilayer perceptrons (Witczak [2006]), where a linearization technique is employed.

2.3 Multi-layer perceptron and its uncertainty

To simplify the study of MLP model uncertainty, the two-layer network is considered. The hidden layer includes neurons with non-linear activation function $\xi(\cdot)$; however, in the output layer one neuron with the linear activation function was employed. The network is described by

$$\tilde{y}(k) = \sum_{i=1}^{n_h} \theta_{0,i}\xi\left(\sum_{j=1}^{n_u} \theta_{i,j}u_j(k)\right), \tag{24}$$

where $\boldsymbol{u} \in \mathbb{R}^{n_u}$ represents a vector of the network inputs, n_h is the number of the neurons in the hidden layer.

The parameters vector $\boldsymbol{\theta} = [\boldsymbol{\theta}_l^T, \boldsymbol{\theta}_n^T]^T \in \mathbb{R}^{n_h(n_u+1)}$ consists of the parameters of the linear neuron $\boldsymbol{\theta}_l^T = [\theta_{0,1}, \ldots, \theta_{0,n_h}]^T$, and the parameter vectors of the non-linear neurons from the hidden layer $\boldsymbol{\theta}_n^T$, where $\boldsymbol{\theta}_n^T = [\boldsymbol{\theta}_{1,n}^T, \ldots, \boldsymbol{\theta}_{n_h,n}^T]^T$, $\boldsymbol{\theta}_{i,n} = [\theta_{i,1}, \ldots, \theta_{i,n_u}]^T$. The network model (24) can be described in a more condensed form:

$$\tilde{y}(k) = \boldsymbol{z}^T(k)\boldsymbol{\theta}_l, \tag{25}$$

where the regressor of the output neuron is given by:

$$\tilde{\boldsymbol{z}}(k) = \left[\xi(\boldsymbol{u}^T(k)\boldsymbol{\theta}_{1,n}), \ldots, \xi(\boldsymbol{u}^T(k)\boldsymbol{\theta}_{n_h,n})\right]. \tag{26}$$

Let us assume that the system output is described in the following form:

$$y(k) = \tilde{y}(k) + \varepsilon(k). \tag{27}$$

The polytopic region $\boldsymbol{\Theta}$ becomes very complicated when the number of measurements and parameters is large, which means that its determination is extremely complex and time-consuming. Since the number of neurons in the hidden layer n_h of the MLP considered is usually large, then the number of parameters $n_\theta = n_h(n_u + 1)$ is large as well, and hence the above approach cannot be directly applied to parameter estimation of (24) (Witczak [2006]). An easier solution relies on approximating the convex polytope $\hat{\boldsymbol{\Theta}}$ by an ellipsoid. In a recursive OBE algorithm (Milanese et al. [1996], Walter and Pronzato [1996]), the measurements are taken into account one after the other to construct a succession of ellipsoids containing all values of $\hat{\boldsymbol{\theta}}$ consistent with all previous measurements. The OBE algorithm provides rules for computing $\boldsymbol{\theta}(k)$ and $\boldsymbol{P}(k)$ in such a way that the volume of $\boldsymbol{\Theta}(\hat{\boldsymbol{\theta}}(k+1)$, where $\boldsymbol{P}(k+1)$) denotes a positive-definite matrix which specifies the size and orientation of the ellipsoid.

Taking into account process linearization (Mrugalski et al. [2007]), the non-linear model (24) can be rewritten as

$$y(k) = \tilde{y}_l(k) + o(\boldsymbol{\theta}, \hat{\boldsymbol{\theta}}), \tag{28}$$

where $o(\boldsymbol{\theta}, \hat{\boldsymbol{\theta}})$ stands for the higher-order terms of the Taylor series expansion, and $\tilde{y}_l(k)$ is the output of the linearized model.

Moreover, based on (28), the system output can be written as

$$y(k) = \tilde{y}_l(k) + \varepsilon(k), \tag{29}$$

where $\varepsilon(k)$ is the output error.

Applying the OBE algorithm, it is possible to obtain the bounds of the linear model output uncertainty interval (Mrugalski et al. [2007]):

$$\tilde{y}_l^m < \tilde{y}_l < \tilde{y}_l^M(k), \tag{30}$$

where

$$\tilde{y}_l^m(k) = \nabla\tilde{y}^T\hat{\boldsymbol{\theta}}_l - \sqrt{\nabla\tilde{y}^T\boldsymbol{P}\nabla\tilde{y}}, \tag{31}$$

$$\tilde{y}_l^M(k) = \nabla\tilde{y}^T\hat{\boldsymbol{\theta}}_l + \sqrt{\nabla\tilde{y}^T\boldsymbol{P}\nabla\tilde{y}}. \tag{32}$$

In order to obtain the output uncertainty interval of the whole model, it is necessary to obtain the bounds of $o(\boldsymbol{\theta}, \hat{\boldsymbol{\theta}})$ based on the expression (32):

$$\begin{aligned} o(\boldsymbol{\theta}, \hat{\boldsymbol{\theta}}) &= y(k) - y_l(k) \\ &= \tilde{\boldsymbol{z}}_0(k)\boldsymbol{\theta}_l - \tilde{\boldsymbol{z}}(k)\boldsymbol{\theta}_l - \nabla\tilde{y}_n^T|_{\boldsymbol{\theta}=\hat{\boldsymbol{\theta}}(k-1)}(\boldsymbol{\theta}_n - \hat{\boldsymbol{\theta}}_n(k)) \\ &= (\tilde{\boldsymbol{z}}_0(k) - \tilde{\boldsymbol{z}}(k))\boldsymbol{\theta}_l - \nabla\tilde{y}_n^T|_{\boldsymbol{\theta}=\hat{\boldsymbol{\theta}}(k-1)}(\boldsymbol{\theta}_n - \hat{\boldsymbol{\theta}}_n(k)). \end{aligned} \tag{33}$$

After some computations it is shown (Mrugalski et al. [2007]) that the linearisation error $o(\boldsymbol{\theta}, \hat{\boldsymbol{\theta}})$ depends on the parameter error $\boldsymbol{e}(k) = \boldsymbol{\theta}(k) - \hat{\boldsymbol{\theta}}(k)$. Moreover, each element $e_i(k)$ is overbounded by the square roots of the diagonal elements matrix $\boldsymbol{P}$, which defines the size and orientation of the ellipsoid:

$$-\sqrt{P_{i,i}} \le e_i(k) \le \sqrt{P_{i,i}}, \tag{34}$$

where $i = 1, \ldots, n_h(1 + n_u)$. Thus, depending on the changes of the values $e_i(k)$, also the value $o(\boldsymbol{\theta}, \hat{\boldsymbol{\theta}})$ will be changing in the limited interval:

$$o(\boldsymbol{\theta}, \hat{\boldsymbol{\theta}})^m \le o(\boldsymbol{\theta}, \hat{\boldsymbol{\theta}}) \le o(\boldsymbol{\theta}, \hat{\boldsymbol{\theta}})^M. \tag{35}$$

Based on the expressions (35), (36) and (39), it is possible to obtain the neural model output uncertainty interval:

$$\begin{aligned} \tilde{\boldsymbol{z}}^T\hat{\boldsymbol{\theta}}_l - \sqrt{\nabla\tilde{y}^T\boldsymbol{P}\nabla\tilde{y}} + o(\boldsymbol{\theta}, \hat{\boldsymbol{\theta}})^m &\le \tilde{y}(k) \\ \le \tilde{\boldsymbol{z}}^T\hat{\boldsymbol{\theta}}_l + \sqrt{\nabla\tilde{y}^T\boldsymbol{P}\nabla\tilde{y}} + o(\boldsymbol{\theta}, \hat{\boldsymbol{\theta}})^M&. \end{aligned} \tag{36}$$

The work Mrugalski et al. [2007] provides a detailed description regarding the determination of $o(\boldsymbol{\theta}, \hat{\boldsymbol{\theta}})^m$ and $o(\boldsymbol{\theta}, \hat{\boldsymbol{\theta}})^M$. As a consequence, the adaptive threshold can be reliably determined.

3. EXPERIMENTAL RESULTS

The example being considered in this section is concerned with the so-called DAMADICS benchmark (Patton et al. [2006]). The benchmark is oriented towards fault diagnosis of a valve actuator being a part of the evaporation station of the Lublin Sugar Factory in Poland (Fig. 1).

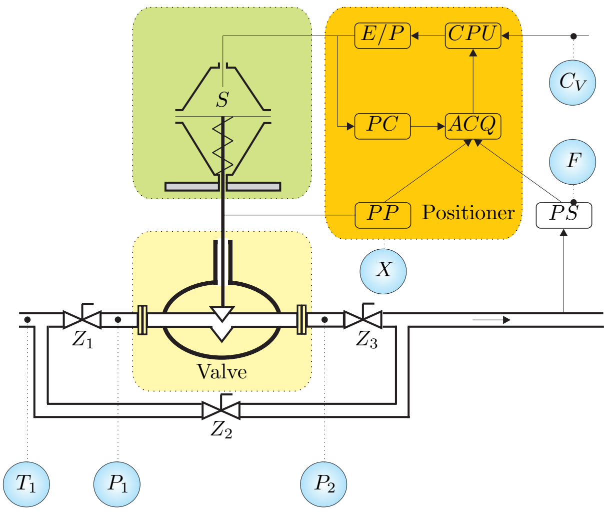

Fig. 1. Scheme of the actuator

The following notations are used: V_1, V_2 and V_3 are cut-off valves, ACQ is a data acquisition unit, CPU is a positioner central processing unit, E/P is an electro-pneumatic transducer, and DT, PT and FT denote displacement, pressure and volume flow transducers, respectively. For remote on-line diagnostics, the following measured variables are accessible: the flow rate of juice after the control valve (F), the actuator's rod displacement (X), the input set-point (C_V), juice temperature at the input of the control valve (T_1), and juice pressures at the input and outlet of the control valve, respectively (P_1 and P_2). The benchmark specifies a set of 19 faults $f_1, \ldots, f_{19}$ whose description can be found in (DAMADICS [2004]).

Based on the actuator benchmark definition (DAMADICS [2004]), two structural models can be defined: $F = f_F(X, P_1, P_2, T_1)$, and $X = f_X(C_V, P_1, P_2, T_1)$, where $f_F(\cdot)$ and $f_X(\cdot)$ denote unknown non-linear functions of the flow rate and displacement, respectively. Using these functional relations, GMDH neural dynamic models were developed. Figure 2 presents system responses and the corresponding system output uncertainty intervals for the faulty data.

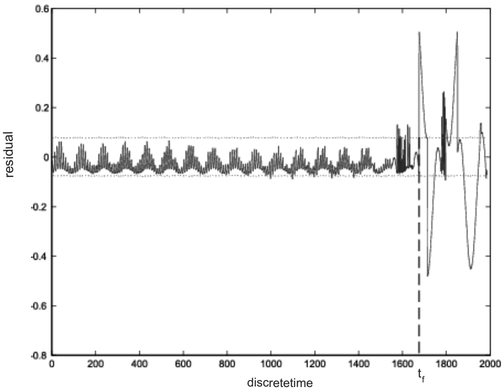

Fig. 2. System response and the system output uncertainty interval for the fault f_4

Now, the MLP model of the juice flow at the outlet of the valve is considered. The selection of the proper structure of the MLP model relies on the gradually increasing number of the neurons in the hidden layer from 1 to 20. For each network architecture, the set of the initial parameters was obtained with the application of the global optimization algorithm called Adaptive Random Search (ARS) (Walter and Pronzato [1996]), and then the approach described in the preceding part of this paper was applied. Figure 3 presents system responses and the corresponding system output uncertainty intervals for the faulty data.

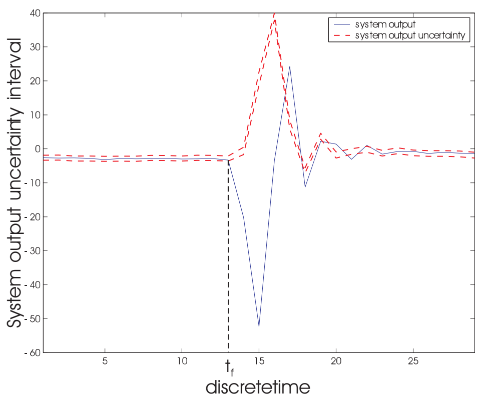

Fig. 3. System response and the system output uncertainty interval for the fault f_{17}

Finally, the Takagi-Sugeno network based model was employed according to the approach described in the preceding part of this paper. Figure 4 presents system responses and the corresponding system output uncertainty intervals for the faulty data.

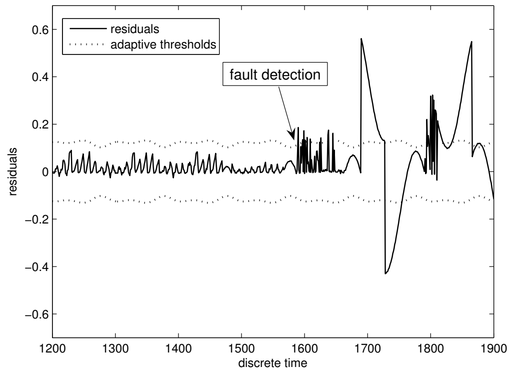

Fig. 4. System response and the system output uncertainty interval for the fault f_4

4. CONCLUDING REMARKS

The main purpose of this paper was to present an overview regarding robust model based fault detection systems applying soft computing models. Special attention was paid to the uncertainty of such models and their usefulness in fault diagnosis. In particular, uncertainties of MLP and GMDH neural networks as well as the Takagi-Sugeno fuzzy model were considered. The presented approaches are based on the bounded-error approach and its extension to the outer bounding ellipsoid. It was shown that the defined confidence interval for the system output of the GMDH, MLP and Takagi-Sugeno networks can be used to develop an adaptive threshold that permits robust fault detection. In the last part, an experimental study performed with the DAMADICS benchmark problem showed the effectiveness of such robust fault detection based on the uncertainty of soft computing models.

REFERENCES

R. Babuska. *Fuzzy Modelling in Control.* Kluwer Academic Publishers, 1998.

M. Brown and C. J. Harris. *Neuro-Fuzzy Adaptive Modelling and Control.* Prentice Hall., 1994.

Z. Bubnicki. *Analysis and Decision Making in Uncertain Systems.* London: Springer-Verlag, 2004.

J. M. F. Calado, J. Korbicz, K. Patan, R. J. Patton, and J. M. G. S da Costa. *Soft computing approaches to fault diagnosis for dynamic systems.*, volume 7, pages 248–286. 2001.

J. Chen and R. J. Patton. *Robust Model Based Fault Diagnosis for Dynamic Systems.* Kluwer Academic Publishers, London, 1999.

DAMADICS. Website of the research training network on Development and application of methods for actuator diagnosis in industrial control systems http://diag.mchtr.pw.edu.pl/damadics, 2004.

A. Emami-Naeini, M.M. Akhter, and S.M. Rock. Effect of model uncertainty on failure detection: the threshold selector. *IEEE Trans. Automatic Control*, AC-33:1106–1115, 1988.

S. J. Farlow. *Self-Organizing Methods in Modeling GMDH Type Algorithms.* Marcel Dekker, New York, 1984.

P. M. Frank. Handling modelling uncertainty in fault detection and isolation systems. In *Proc. 9th Int. Conf. IPMU*, pages 1729–1749, Annecy, France, 2002.

J. Gertler. *Fault Detection and Diagnosis in Engineering Systems.* Marcel Dekker, New York, 1998.

S. Hui and S.H. Zak. Observer design for systems with unknown input. *International Journal of Applied Mathematics and Computer Science*, 15(4):431–446, 2005.

R. Iserman. *Fault diagnosis systems. An introduction from fault detection to fault tolerance.* Springer-Verlag, New York, 2006.

A. G. Ivakhnenko. Polynominal theory of complex systems. SMC-1(4):44–58, 1971.

J. Korbicz. Fault detection using analytical and soft computing methods. *Bulletin of the Polish Academy of Sciences: Technical Sciences*, 54(1):75–88, 2006.

J. Korbicz and C. Cempel. Analytical and knowledge-based redundancy in fault detection and diagnosis. 3 (3), 1993.

J. Korbicz and M. Mrugalski. Confidence estimation of GMDH neural networks and its application in fault detection systems. *International Journal of Systems Science, (in print)*, 2008.

J. Korbicz, J. Kościelny, Z. Kowalczuk, and W. Cholewa (Eds.). *Fault diagnosis. Models, Artificial Intelligence, Applications.* Springer-Verlag, Berlin, 2004.

M. Kowal. *Optimization of neuro-fuzzy structures in technical diagnostics.* Lecture Notes in Control and Computer Science, Vol. 9. University of Zielona Góra Press, Zielona Góra, 2005.

M. Kowal and J. Korbicz. Fault detection under fuzzy model uncertainty. *Int. J. Automation and Computing*, 4(2):117–124, 2007.

M. Milanese, J. Norton, H. Piet-Lahanier, and E. Walter. *Bounding Approaches to System Identification.* Plenum Press, New York, 1996.

M. Mrugalski. *Neural Network Based Modelling of Non-linear Systems in Fault Detection Schemes.* PhD thesis, Faculty of Electrical Engineering, Computer Science and Telecommunications, University of Zielona Góra, Zielona Góra, 2004.

M. Mrugalski, J. Korbicz, and R.J. Patton. Robust fault detection via gmdh neural networks. In *16th IFAC World Congress*, Prague, Czech Republic, 2005.

M. Mrugalski, M. Witczak, and J. Korbicz. Confidence estimation of the multi-layer perceptron and its application in fault detection systems. *Engng. Appl. of Artificial Intelligence, (in print)*, 2007.

O. Nelles. *Non-linear Systems Identification. From Classical Approaches to Neural Networks and Fuzzy Models.* Springer, Berlin, 2001.

R. J. Patton and J. Korbicz, editors. *International Journal of Applied Mathematics and Computer Science : Special Issue : Advances in Computational Intelligence for Fault Diagnosis Systems, ISSN: 0867-857X*, volume 9(no 3). Technical University Press, Zielona Góra, 1999.

R. J. Patton, P. M. Frank, and R. N. Clark. *Issues of Fault Diagnosis for Dynamic Systems.* Springer-Verlag, Berlin, 2000.

R. J. Patton, J. Korbicz, M. Witczak, and F. J. Uppal. Combined computational intelligence and analytical methods in fault diagnosis. In *Intelligent Control Systems using Computational Intelligence Techniques, (Ruano, A. E. (Ed.))*, pages 349–385. The IEE Press, London, 2005.

R.J. Patton, J. Korbicz, and S. Lesecq. A benchmark study of fault diagnosis for an industrial actuator. *Special issue of Control Engineering Practice*, 14(6):575–717, 2006.

D. T. Pham and L. Xing. *Neural Networks for Identification, Prediction and Control.* Springer-Verlag, London, 1995.

A. Piegat. *Fuzzy Modelling and Control (Studies in Fuzziness and Soft Computing).* Berlin: Springer-Verlag, 2001.

D. Rutkowska and L. Zadeh. Neuro-fuzzy and soft computing. *Special issue of Int. J. Appl. Math. and Comput. Sci.*, 10(4), 2000.

L. Rutkowski. *New Soft Computing Techniques for System Modelling, Patern Classification and Image Processing.* Springer, Berlin, 2004.

E. Walter and L. Pronzato. *Identification of Parametric Models from Experimental Data.* Springer, London, 1996.

M. Witczak. Toward the training of feed-forward neural networks with the D-optimum input sequence. *IEEE Transactions on Neural Networks*, 17(2):357–373, 2006.

M. Witczak. *Modelling and Estimation Strategies for Fault Diagnosis of Non-linear Systems.* Springer-Verlag, Berlin, 2007.

M. Witczak, J. Korbicz, M. Mrugalski, and R. J. Patton. A GMDH neural network-based approach to robust fault diagnosis: application to the DAMADICS benchmark problem. *Control Engineering Practice*, 14(6):671–683, 2006.