Confidence estimation of GMDH neural networks and its application in fault detection systems

Józef Korbicz\textsuperscript{a} and Marcin Mrugalski\textsuperscript{a,*}

\textit{Institute of Control and Computation Engineering, University of Zielona Góra, ul. Podgórna 50, 65–246 Zielona Góra, Poland}

\textit{(Received 29 April 2006; final version received 5 December 2007)}

This article deals with the problem of determination of the model uncertainty during the system identification via application of the self-organising group method of data handling (GMDH) neural network. In particular, the contribution of the neural network structure errors and the parameter estimates inaccuracy to the model uncertainty were presented. Knowing these sources and applying the Outer Bounding Ellipsoid (OBE) algorithm it was possible to calculate the uncertainty of the parameters and the model output. The mathematical description of the model uncertainty enabled designing the robust fault detection system, whose effectiveness was verified by the DAMADICS benchmark.

\textbf{Keywords:} system identification; model uncertainty; neural networks; bounded-error modelling; non-linear systems; fault diagnosis; application

1. Introduction

The most crucial challenge which should be undertaken by production engineers who supervise modern production systems is to design a mathematical model describing the system behaviour. The scope of applications of a mathematical models in the contemporary industrial systems is extremely broad and includes the design of the systems (Herskovits et al. 2005), the control (Delaleu et al. 2001, Etien et al. 2002, Narendra and Lewis 2001, Wang and Hill 2006, Van den Boom et al. 2005) and the system diagnosis (Calado et al. 2001, Isermann 2005, Korbicz et al. 2004, Mrugalski and Korbicz 2005a, Patton and Chen 2000, Patton et al. 2005, Scola et al. 2003). The models are usually created based on the physical laws which describe the system behaviour. Unfortunately, in the case of the most contemporary industrial systems these laws are too complex or unknown. As a consequence, the phenomenological models are not available. In order to solve this problem, the system identification can be applied (Patton and Korbicz 1999, Soderstrom and Stoica 1989). One of the most desirable features of the model obtained during the system identification is small modelling uncertainty, which is defined as a mismatch between the model and the system being considered (Blanke et al. 2003). The system identification is a two-stage process, which relies on a model structure selection and parameter estimation. It is crucial to use such an identification technique, which enables reduction of the contribution of the structure errors and the parameter estimates inaccuracy to the model uncertainty.

One of the most popular non-linear system identification approaches is based on the application of the Artificial Neural Networks (ANNs) (Bouthiba 2004, Gupta et al. 2003, Nelles 2001, Vieiraa et al. 2004). The ANNs can be most adequately characterised as computational models with particular properties such as the ability to learn, adapt and parallel data processing. Other advantages of the ANNs over conventional identification methods include simplicity of implementation, generalisation abilities and good approximation properties. Unfortunately, the successful application of the ANNs in the system identification task depends on a proper selection of the neural network architecture. In the case of the classical ANNs such as Multi-Layer Perceptron (MLP), the problem reduces to the

*Corresponding author. Email: M.Mrugalski@issi.uz.zgora.pl
selection of the number of layers and the number of neurons in a particular layer. If the obtained network does not satisfy pre-specified requirements, then a new network structure is selected and the parameter estimation is repeated once again. Determination of the appropriate structure and parameters of the model in the presented way is a complex task. Furthermore, an arbitrary selection of the ANN structure can be a source of the model uncertainty. Thus, it seems desirable to have a tool which can be employed to the automatic selection of the ANN structure, based only on the measured data. To overcome this problem a group method of data handling (GMDH) neural networks (Farlow 1984, Ivakhnenko and Mueller 1995, Mrugalski 2004) have been proposed. The synthesis process of GMDH neural network is based on the iterative processing of a sequence of operations. This process leads to the evolution of the resulting model structure in such a way so as to obtain the best quality approximation of the identified system. Thus, the task of designing a neural network is defined in order to obtain a model with small uncertainty.

Apart from the contribution to the model uncertainty of the errors caused by the inappropriate structure selection, the parameter estimates inaccuracy also influences the model quality. This problem was widely presented in Witczak et al. (2006), where the potential sources of the model uncertainty following from the application of improper parameters estimation algorithm were described. In particular, it was shown that the usual parameters estimation algorithms work based on the incorrect assumptions concerning the properties of the noises and disturbances which affect on the data used during system identification. In order to overcome this problem, the Bounded Error Approach (BEA) (Milanese et al. 1996, Walter and Pronzato 1997) was applied. This approach is based on more realistic assumption that the noises and disturbances lie between given prior bounds. Unfortunately, in spite of many advantages this algorithm also has significant memory and time expenditure. It follows from the fact that the BEA estimates the model parameters and their uncertainty in the form of the admissible parameter set. Based on this result it is possible to define the model uncertainty in the form of the system output uncertainty interval, which can be applied in the numerous industrial applications, e.g. robust fault detection.

This article is structured in the following manner: in the second section, the concept of the robust model-based fault detection is shortly described. In the third section, the idea of the GMDH network synthesis is widely presented. The fourth section presents the sources of the GMDH neural model uncertainty. In particular, the contribution of the structure errors to the model uncertainty is described in detail. The fifth section introduces the OBE algorithm which is compared with the classical LMS algorithm in the seventh section. The next two sections present the confidence estimation and the synthesis of the GMDH neural network with application of the OBE technique. Then a study regarding the application of the developed approach to the robust fault detection is presented. Finally, the last section is devoted to conclusions.

2. Robust model-based fault detection

The most known structure of the fault detection system is based on the model of the diagnosed system. Model-based fault diagnosis can be defined as the detection, isolation and identification of faults in the system based on a comparison of the system available measurements, with information represented by the system mathematical model (Figure 1) (Chen and Patton 1999, Korbicz et al. 2004). The model of the diagnosed system is created before its application in the fault diagnosis system. The comparison of the system $y(k)$ and model response $\hat{y}(k)$ lead to the generation of the residual $\epsilon(k):

$$
(k) = y(k) - \hat{y}(k),
$$

Figure 1. Scheme of the residual-based fault detection system.
which is a source of the information about faults for the further processing. In the model-based fault detection approach, it is assumed that the residual $e(k)$ should normally be close to zero in the fault-free case, and it should be distinguishably different from zero in the case of a fault. In other words, the residual should ideally carry only an information regarding a fault. Under such an assumption, the faults are detected by setting of a fixed threshold on the residual signal (Figure 2). In this case, the fault can be detected when the absolute value of the residuum $|e(k)|$ will be larger than the arbitrary assumed threshold value $\delta_e$:

$$|e(k)| \leq \delta_e.$$  \hspace{1cm} (2)

The fundamental difficulty with this kind of symptom evaluation is that measurement of the system output $y(k)$ is usually corrupted by noise and disturbances $e^m(k) \leq e(k) \leq e^M(k)$, where $e^m(k) \leq 0$ and $e^M(k) \geq 0$. Another difficulty follows from the fact that the model obtained during the system identification is usually uncertain (Mrugalski 2004, Witczak et al. 2006). The model uncertainty can appear during model structure selection and also parameters estimation. In practice, due to modelling uncertainty and measurement noise, it is necessary to assign wider thresholds in order to avoid false alarms. It usually implies a reduction of the fault detection sensitivity.

To tackle this problem, the adaptive time-variant threshold that is adapted according to the system behaviour can be applied. Indeed, knowing the model structure and possessing the knowledge regarding its uncertainty it is possible to design a robust fault detection scheme. The idea of the proposed approach is illustrated in Figure 3.

The proposed technique relies on the calculation of the model output uncertainty interval based on the estimated parameters whose values are known at some confidence level.

$$\hat{y}^m(k) \leq \hat{y}(k) \leq \hat{y}^M(k).$$  \hspace{1cm} (3)

Additionally, as the measurement of the controlled system response $y(k)$ is corrupted by the noise, it is necessary to add the boundary values of the output error $e^m(k)$ and $e^M(k)$ to the model output uncertainty interval. Defined in this way system output interval (Figure 4) should contain the real system response in the fault free mode. An occurrence of the fault is signaled when system output $y(k)$ crosses the system output uncertainty interval:

$$\hat{y}^m(k) + e^m(k) \leq y(k) \leq \hat{y}^M(k) + e^M(k).$$  \hspace{1cm} (4)

The effectiveness of the suggested method of robust fault detection requires determination of a mathematical description of model uncertainty and knowing maximal and minimal values of disturbances $e$. For this reason in the next part of this article the sources of model uncertainty will be presented. Moreover, the method of their calculation in the form of the model output uncertainty interval will be shown. The exact values of the disturbances $e$ can be calculated following the method described in Witczak et al. (2006). However, it should be pointed out that during application of the fault detection system, the disturbances cannot be larger than those which were taken into account in designing the fault detection system.

![Figure 2. Residual-based fault detection.](image2)

![Figure 3. Robust fault detection with the adaptive time-variant threshold.](image3)

![Figure 4. Application of system output uncertainty interval to robust fault detection.](image4)
Not following this rule may cause false alarms in fault detection. The suggested method of detection is robust to model uncertainty and disturbances; however, it is crucial to use the representative data sets to system identification procedure. It prevents inappropriate working of fault detection system in case of using new data sets. The problem of the selection of the identification data can be solved by the application of the experiment design techniques (Atkinson and Donev 1992).

3. Synthesis of the GMDH neural network

The idea of the GMDH approach relies on replacing of the complex neural model by the set of the hierarchically connected partial models. The model is obtained as a result of the neural network structure synthesis with the application of the GMDH algorithm (Farlow 1984, Ivakhnenko and Mueller 1992, Pham and Xing 1995). The synthesis process consists of the partial model structure selection and parameter estimation. The parameters of each partial model (a neuron) are estimated separately. In the next step of the process synthesis, the partial models are evaluated, selected and included to the newly created neurons layers (Figure 5). During the network synthesis new layers are added to the network. The process of the network synthesis leads to the evolution of the resulting model structure to obtain the best quality approximation of the real system output signals. The process is completed when the optimal degree of network complexity is achieved. An outline of the GMDH algorithm can be as follows (Mrugalski 2004):

**Step 1:** Based on the available inputs \( u(k) \in \mathbb{R}^n \), the GMDH network grows its first layer of the neurons. It is assumed that all the possible couples of inputs from signals \( u_1^{(0)}(k), \ldots, u_n^{(0)}(k) \), belong to the training data set \( T \), constitute the stimulation which results in the formation of the neurons outputs \( y_n^{(0)}(k) \):

\[
y_n^{(0)}(k) = f(u) = f(u_1^{(0)}(k) \ldots, u_n^{(0)}(k)),
\]

where \( l \) is the layer number of the GMDH network and \( n \) is the neuron number in the \( l \)-th layer. The number of neurons in the first layer of the GMDH network depends on the number of external inputs \( n \).

In order to estimate the unknown parameters \( \hat{\theta} \), the techniques for the parameter estimation of linear-in-parameter models can be used, e.g. Least Mean Square method (LMS). After the estimation, the parameters are ‘frozen’ during the further network synthesis.

**Step 2:** Using a validation data set \( V \), not employed during the parameter estimation phase, calculate a processing error of the each partial model in the current \( l \)-th network layer. The processing error is calculated with the application of the evaluation criterion such as: Final Prediction Error (FPE), Akaike Information Criterion (AIC) or F-test. Based on the defined evaluation criterion it is possible to select the best-fitted neurons in the layer. The selection methods in the GMDH neural networks play a role of a mechanism of the structural optimisation at the stage of construing a new layer of neurons. During the selection, neurons which have too large a value of the evaluation criterion \( Q(y_n^{(0)}) \) are rejected. A few methods of performing the selection procedure can be applied (Mueller and Lemke 2000). One of the most frequently used is the constant population method. It is based on a selection of \( g \) neurons, for which the evaluation criterion \( Q(y_n^{(0)}) \) reaches the least values. The constant \( g \) is chosen in an empirical way. The most important advantage of this method is the simplicity of implementation. Unfortunately, constant population method has very restrictive structure evolution possibilities. One way out of this problem is an application

![Figure 5. Synthesis of the GMDH neural network.](image-url)
of the optimal population method. This approach is based on rejecting the neurons whose value of the evaluation criterion is larger than an arbitrarily determined threshold \( \varepsilon_\theta \). The threshold is usually selected for each layer in an empirical way depending on the considered task. Difficulty with the selection of the threshold means that the optimal population method should not be applied too often and there is a need to develop a new more efficient selection method. After the selection procedure, the outputs of the selected neurons become the inputs to other neurons in the next layer:

\[
\begin{align*}
  u^{(l+1)}_1 &= y^{(l)}_1, \\
  u^{(l+1)}_2 &= y^{(l)}_2, \\
  &\vdots \\
  u^{(l+1)}_{n_l} &= y^{(l)}_{n_l}.
\end{align*}
\]

(6)

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 the 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 procedure of removing unnecessary neurons is the last stage of the synthesis of the GMDH neural network.

It should be strongly underlined that all the hard computations regarding the design of the GMDH neural network are performed off-line, and hence the problem concerning time consuming calculations during the GMDH network synthesis is not of paramount importance. Then a suitably designed neural network is used for on-line fault detection according to the approach presented in Section 2. During the fault detection the structure of the GMDH network does not change.

**4. Sources of the GMDH model uncertainty**

The application of the GMDH approach to the model structure selection can improve the quality of the model but it cannot eliminate the model uncertainty at all. For this reason it is necessary to analyse each stage of the neural model synthesis process in order to identify each source of the uncertainty. This knowledge makes it possible to estimate and reduce the model uncertainty.

**4.1. Selection of the improper partial model structure**

The GMDH approach allows much freedom in defining the partial model structure. The original GMDH algorithm developed by Ivakhnenko is based on the partial model which describes the relation between the input \( u(k) \in \mathbb{R}^{n_u} \) and the output \( y(k) \) of the system in the form of the linear or second-order polynomial functions (Ivakhnenko and Mueller 1995). In general, in the case of the identification of the static non-linear systems, the partial model can be described as follows:

\[
\hat{y}^{(l)}_n(k) = \xi\left((r^{(l)}_n(k))^T p^{(l)}_n\right),
\]

(7)

where \( \hat{y}^{(l)}_n(k) \) stands for the neuron output \( l \) is the layer number, \( n \) is the neuron number in the \( l \)-th layer), corresponding to the \( k \)-th measurement of the system input, whilst \( \xi(\cdot) \) denotes a non-linear invertible activation function, i.e. there exists \( \xi^{-1}(\cdot) \). Moreover, \( r^{(l)}_n(k) = f(u^{(l)}_n(k), u^{(l)}_n(k)) \), \( i, j = 1, \ldots, n_u \) and \( p^{(l)}_n \in \mathbb{R}^{n_p} \) are the regressor and the parameter vectors, respectively, and \( f(\cdot) \) is an arbitrary bivariate vector function, e.g. \( f(x) = [x_1^2, x_2, x_1 x_2, x_1, x_2, 1]^T \) that corresponds to the bivariate polynomial of the second degree. The feature of the above partial model is that the techniques for the parameter estimation of linear-in-parameter models can be used. Indeed, since \( \xi(\cdot) \) is invertible, the neuron described by (7) can relatively easily be transformed into a linear-in-parameter one.

The above described partial model can be used for the identification of the non-linear static systems. Unfortunately, most of the industrial systems are dynamic in their nature (Patton et al. 2000). The application of the static neural network will result in a large model uncertainty. Thus, during the system identification it seems desirable to employ the models, which can represent the dynamic of the system. In the case of the classical neural network, for example Multi-Layer Perceptron (MLP), the problem of modelling of the dynamics is solved by introduction of additional input signals. The input vector for this kind of networks consists of suitably delayed input and output signals:

\[
y(k) = f(u(k), u(k-1), \ldots, u(k-n_u), y(k-1), \ldots, y(k-n_y)),
\]

(8)

where \( n_u \) and \( n_y \) represent the number of delays. Unfortunately, the described approach cannot be applied into the GMDH neural network easily, because the GMDH network is constructed through gradual connection of the partial models. The introduction of the global output feedback lines complicates the synthesis of the network. On the other hand, the behaviour of each partial model should reflect the behaviour of the identified system. It follows from
the rule of the GMDH algorithm that the parameters of each partial model are estimated in such a way that their output signals are the best approximation of the real system output. In this situation, the partial model should have an ability to represent the dynamics. One way out of this problem is to use dynamic neurons. Due to the introduction of different local feedbacks to the classical neuron model, it is possible to achieve a few types of dynamic neurons, e.g. with local activation feedback (Fasconi et al. 1992), local synapse feedback (Back and Tsoi 1991) and output feedback (Gori et al. 1989). The main advantage of networks constructed with application of the dynamic neurons is that their stability can be proven relatively easily. As a matter of fact, the stability of the network only depends on the stability of neurons. The feed-forward structure of such networks seems to make the training process easier. On the other hand, introduction of dynamic neurons increases the parameter space significantly. This drawback together with the non-linear and multi-modal properties of an identification index implies that parameter estimation becomes relatively complex.

In order to overcome this drawback, it is possible to use another kind of dynamic neuron model (Mrugalski et al. 2003a). Dynamics in this neuron is realised by introduction of a linear dynamic system – an Infinite Impulse Response (IIR) filter. In this way, each neuron in the network reproduces the output an Infinite Impulse Response (IIR) filter. In this way, the filter output is used as the input for the activation module

or equivalently,

or equivalently,

where \( \mathbf{r}_n^0(k) = [-y_n^0(k-1), \ldots, -y_n^0(k-n_a), \mathbf{u}_n^0(k), \mathbf{u}_n^0(k-1), \ldots, \mathbf{u}_n^0(k-n_b)] \) and \( \mathbf{p}_n^0 = [a_1, \ldots, a_{n_a}, v_0, v_1, \ldots, v_{n_b}] \) are the regressor and the filter parameters, respectively. The filter output is used as the input for the activation module

The application of the dynamic neurons during the GMDH network synthesis can improve the model quality. In order to additional reduction of the uncertainty of the dynamic neural model it is necessary to assume the appropriate order of the IIR filter. This problem can be solved by the application of the Lipschitz index approach based on so-called Lipschitz quotients (Nelles 2001).

4.2. Influence of the evaluation criteria on the partial models selection

In order to perform the model construction procedure, it is necessary to define the evaluation criterion of the partial models. Mueller and Lemke (2000) present a comprehensive table of the most common evaluation criteria used in the parametric GMDH algorithm. The most frequent are applied Akaike Information Criterion (AIC) and Final Prediction Error (FPE). These criteria are based on the statistics taking into consideration the complexity of partial models. The optimal structure of the partial model is obtained when the statistics has the minimal value. In the case of AIC criterion statistics has the following general form:

or equivalently,

where \( J_{n_T}(N_{\text{arch}}) \) represents the goal function for the model architecture \( N_{\text{arch}} \), e.g. \( J_{n_T}(N_{\text{arch}}) = (1/n_T) \sum_{t=1}^{n_T} (\gamma(k) - \mathbf{j}_n^0(k))^2 \), and \( \gamma(n_T, n_p) \) is the function of the number of the data samples \( n_T \) and the number of partial model parameters \( n_p \). The appropriate selection of (12) ensures its increasing along with increasing of the number of parameters and converge to zero along with increasing of the data samples set. The selection of the function characterised by the above mentioned properties ensures an elimination of the over-parameterised partial models. In the case of the AIC criterion, the factor \( \gamma(n_T, n_p) \) is equal \( 2n_p \) which leads to the following final form of the criterion:

\[
Q_{\text{AIC}}(\mathbf{j}_n^0) = n_T \log J_{n_T}(N_{\text{arch}}) + 2n_p.
\]
In the case of the FPE criterion, the statistics reflect an expected variance of prediction error during if prediction new observations based on the model obtained for the identification data set:

\[ Q_{\text{FPE}}(\hat{\rho}_n^{(0)}) = \mathcal{E}(\rho_n^2, N_{\text{arch}}), \quad (14) \]

where \( \tau \) denote the prediction period. In (Soderstrom and Stoica 1989) it was theoretically and practically proved that for \( n_p < \frac{2}{C_3} \) and finally:

\[ Q_{\text{FPE}}(\hat{\rho}_n^{(0)}) \approx \Lambda(n_\text{p}/n_\tau), \quad (15) \]

where an asymptotic unbiased estimate of the \( \Lambda \) is:

\[ \hat{\Lambda} = J_{n_\tau}(N_{\text{arch}}) \left( 1 - \frac{n_p}{n_\tau} \right). \quad (16) \]

As a result of substituting (16) into (15) the final form of the FPE criterion is obtained:

\[ Q_{\text{FPE}}(\hat{\rho}_n^{(0)}) = J_{n_\tau}(N_{\text{arch}}) \left[ 1 + \frac{n_p}{n_\tau} \right] \frac{1}{1 - \frac{n_p}{n_\tau}}. \quad (17) \]

In the case of AIC criterion, it is possible to select a better partial model based on the inequality defined with the statistics (12):

\[ n_\tau \log J_{n_\tau}(N_{\text{arch},1}) + \gamma(n_\tau, n_p, 1) \leq n_\tau \log J_{n_\tau}(N_{\text{arch},2}) + \gamma(n_\tau, n_p, 2), \quad (18) \]

which after simple transformation has the form:

\[ J_{n_\tau}(N_{\text{arch},1}) \leq J_{n_\tau}(N_{\text{arch},2}) \exp \left[ \frac{(\gamma(n_\tau, n_p, 2) - \gamma(n_\tau, n_p, 1))}{n_\tau} \right]. \quad (19) \]

and finally:

\[ \chi_\nu^2(n_p, 2 - n_p, 1) = n_\tau \left( \exp \left[ \frac{(\gamma(n_\tau, n_p, 2) - \gamma(n_\tau, n_p, 1))}{n_\tau} \right] - 1 \right). \quad (20) \]

Based on (20) the AIC criterion can be perceived as the F-test Soderstrom and Stoica (1989) with in advance defined confidence level. The same disadvantage occurs in the case of the FPE criterion. In Soderstrom and Stoica (1989) it was theoretically and practically proved that for \( n_p, 2 - n_p, 1 = 1 \) degree of freedom, the confidence level is 0.157. This result means that the probability of selection over-parameterised structure \( N_{\text{arch},2} \) via AIC or FPE criteria is 15.7%. If the number of partial models in the GMDH neural network is higher then probability of selection over-parameterised neurons is not acceptable. Another reason opposite to the application of the AIC and FPE criteria is the fact that the probability of selection over-parameterised partial neurons does not decrease along with \( n_\tau \to \infty \). Furthermore, the AIC and FPE criteria were designed for comparison of the hierarchical partial models \( N_{\text{arch},1} \subset N_{\text{arch},2} \). In the case of the GMDH neural network this assumption is not fulfilled (Figure 7).

4.3. Contribution of the parameter estimates inaccuracy to the model uncertainty

In the previous subsections, the contributions of the neural network structure errors to the model uncertainty were presented. Apart from the model structure selection stage, inaccuracy of the parameter estimates also contribute to the model uncertainty. Most of them result from the application of the LMS algorithm to the parameters estimation. Indeed, while applying this approach to the parameter estimation of partial models (7) a set of restrictive assumptions has to be satisfied. One of them concerns the properties of the noise \( \varepsilon_n \) which affect on the system output \( y(k) \). In order to obtain the unbiased and minimum variance parameter estimate for (7) it has to be assumed:

\[ \mathcal{E}[\varepsilon_n^{(0)}] = 0, \quad (21) \]

and

\[ \text{cov}[\varepsilon_n^{(0)}] = (\sigma_n^{(0)})^2 I. \quad (22) \]

The assumption (21) means that there are no deterministic disturbances, which unfortunately are usually caused by the structural errors. However, the condition (22) means that the model uncertainty is described in a purely stochastic way. The assumptions

![Figure 7. The problem of unhierarchy of the partial models in the first layer of the GMDH neural network.](image-url)
(21) and (22) are not usually fulfilled in practice, which cause increasing of the model uncertainty. The above presented assumption concerning the properties of the noise is one of the potential sources of the model uncertainty. More of them are widely described in the papers of Mrugalski and Korbicz (2005b) and Witczak et al. (2006), which focus on the application of the bounded error approach to the fault detection of the industrial systems. Among the most important sources of the model uncertainty caused by application of the LMS can be mentioned the parameter estimation of nonlinear-in-parameter partial models and the problem of the biased parameters estimates obtained for the partial models in the next layers of the GMDH neural network.

5. Parameters estimation via OBE algorithm

The classical ANNs parameter estimation problem is formulated as the parameter estimation of nonlinear-in-parameter models. Among the existing algorithms, a few groups can be distinguished: gradient-based algorithms, evolutionary algorithms, and stochastic algorithms (Gupta et al. 2003, Nelles 2001). Undoubtedly, the gradient-based algorithms form the most popular group of them. Their properties can be summarised as follows: low-computational burden, easy implementation and fast performance. On the other hand, the training of the ANN is usually an optimisation problem of a multimodal cost function. This means that the gradient-based algorithms usually find one of the unsatisfactory local minima. To overcome this problem, it seems desirable to use either stochastic or evolutionary algorithms (Obuchowicz 2003, Wouwer et al. 2003). It is well-known that such algorithms possess global convergence properties. Unfortunately, the number of parameters of an ANN is rather large and it leads to an increase in the computational burden, which makes the parameter estimation process extremely time consuming.

The application of the GMDH approach during the neural network synthesis allows us to apply the parameter estimation of linear-in-parameters models algorithms, e.g. least square method (Farlow 1984, Ivakhnenko and Mueller 1995). It follows from the facts that the parameters of the each partial models are estimated separately and the neuron’s activation function $\xi(\cdot)$ fulfills the following conditions:

1. $\xi(\cdot)$ is continuous and bounded, i.e.

$$\forall x \in \mathbb{R} : a < \xi(x) < b.$$  \hspace{2cm} (23)

2. $\xi(\cdot)$ is monotonically increasing, i.e.

$$\forall x, y \in \mathbb{R} : x \leq y \iff \xi(x) \leq \xi(y).$$  \hspace{2cm} (24)

3. $\xi(\cdot)$ is invertible, i.e. there exists $\xi^{-1}(\cdot)$.

The advantage of this approach is the simple computation algorithm that gives good results even for small sets of measuring data. Unfortunately, the usual statistical parameter estimation framework assumes that the data are corrupted by the errors, which can be modelled as realisations of independent random variables with a known or parameterised distribution. A more realistic approach is to assume that the errors lie between given prior bounds. It leads directly to the bounded error set estimation class of algorithms, and one of them called the OBE algorithm (Walter and Pronzato 1997, Milanese et al. 1996) can be employed to solve the parameter estimation problem.

The OBE algorithm requires the system output to be described in the form:

$$y^{(0)}_n(k) = (\hat{p}^{(0)}_n(k))^T \hat{P}^{(0)}_n + \hat{e}^{(0)}_n(k).$$  \hspace{2cm} (25)

Moreover, it is assumed that the output error can be defined as:

$$\epsilon(k)^{(0)}_n = y^{(0)}_n(k) - \hat{y}^{(0)}_n(k),$$  \hspace{2cm} (26)

where $\hat{y}^{(0)}_n(k)$ is the $k$-th scalar measurement of the system output, and $\hat{y}^{(0)}_n(k)$ is the corresponding neuron output. The problem is to estimate the parameter vector $\hat{p}^{(0)}_n$, i.e. to obtain $\hat{p}^{(0)}_n$, as well as an associated parameter uncertainty in the form of the admissible parameter space $E$. In order to simplify the notation the index $0$ is omitted. The set of admissible parameter values allows us to calculate the confidence region of the model output which satisfies:

$$\hat{y}^{min}(k) \leq y(k) \leq \hat{y}^{max}(k),$$  \hspace{2cm} (27)

where $\hat{y}^{min}(k)$ and $\hat{y}^{max}(k)$ are the minimum and maximum admissible values of the model output. As has been already mentioned, it is possible to assume that $\epsilon(k)$ lies between given prior bounds. In this case, the output error is assumed to be bounded as follows:

$$\epsilon^{min}(k) \leq \epsilon(k) \leq \epsilon^{max}(k),$$  \hspace{2cm} (28)

where the bounds $\epsilon^{min}(k)$ and $\epsilon^{max}(k)$ ($\epsilon^{min}(k) \neq \epsilon^{max}(k)$) are known a priori. An example can be provided by the data collected with an analogue-to-digital converter or for measurements performed with a sensor of a given type. Based on the measurements $(\mathbf{r}(k), y(k))$, $k = 1, \ldots, n_T$ and the error bounds (28) a finite number of linear inequalities is defined. Each inequality
associated with the \( k \)-th measurement can be put in the following standard form:

\[
-1 \leq y(k) - \hat{y}(k) \leq 1,
\]

where

\[
y(k) = \frac{2y(k) - \varepsilon M(k) - \varepsilon m(k)}{\varepsilon M(k) - \varepsilon m(k)},
\]

\[
\hat{y}(k) = \frac{2}{\varepsilon M(k) - \varepsilon m(k)}\hat{y}(k).
\]

Inequalities (29)–(31) define two parallel hyperplanes for each \( k \)-th measurement:

\[
H^+ = \{ p \in \mathbb{R}^n : y(k) - r^T(k-1)p = 1 \},
\]

and

\[
H^- = \{ p \in \mathbb{R}^n : y(k) - r^T(k-1)p = -1 \},
\]

bounding a strip \( S(k) \) containing set of \( p \) values which satisfy the constrains with \( y(k) \):

\[
S(k) = \{ p \in \mathbb{R}^n : -1 \leq y(k) - \hat{y}(k) \leq 1 \}.
\]

By the intersection of the strips \( S(k) \) for the \( k = 1, \ldots, n_T \) measurements the parameters feasible region \( E \) is obtained and its center is chosen as the parameter estimate. Unfortunately, the polytopic region \( E \) becomes very complicated when the number of measurements and parameters is significant what cost that its determination is time-consuming. A easier solution relies on the approximation of the convex polytopes \( S(k) \) by simpler ellipsoids. In a recursive OBE algorithm which is based on this idea, the measurements are taken into account one after the other to construct a succession of ellipsoids containing all values of \( p \) consistent with all previous measurements. After the first \( k \) observations the set of feasible parameters is characterised by the ellipsoid:

\[
E(\hat{p}(k), P(k)) = \{ p \in \mathbb{R}^n : (p - \hat{p}(k))^T P^{-1}(k)(p - \hat{p}(k)) \leq 1 \},
\]

where \( \hat{p}(k) \) is the center of the ellipsoid constituting \( k \)-th parameter estimate, and \( P(k) \) is a positive-definite matrix which specifies its size and orientation. By means of an intersection of the strip (34) and the ellipsoid (35), a region of possible parameter estimates is obtained. This region is outershadowed by the new \( E(k+1) \) ellipsoid. The OBE algorithm provides rules for computing \( \hat{p}(k) \) and \( P(k) \) in such a way that the volume of \( E(\hat{p}(k+1), P(k+1)) \) is minimised (cf. Figure 8). The center of the last \( n_T \)-th ellipsoid constitutes the resulting parameter estimate while the ellipsoid itself represents the feasible parameter set. However, any parameter vector \( p \) contained in \( E(n_T) \) is a valid estimate of \( p \). A detailed structure of the OBE recursive algorithm is as follows (Walter and Pronzato 1997):

**Input:** \( \hat{p}(0) \) – the initial estimate, \( c \) – a sufficiently large positive real number.

**Output:** \( \hat{p} \) – the parameter estimate, \( P \) – the matrix which specifies a size and orientation of the ellipsoid \( E \).

1. Set initial ellipsoid

\[
P(0) = cI.
\]

2. While \( r(k) = 0 \), set

\[
E(\hat{p}(k), P(k)) = E(\hat{p}(k-1), P(k-1)), \quad k \rightarrow k+1.
\]

3. Calculate model output

\[
\hat{y}(k) = r^T(k)\hat{p}(k),
\]

and normalise \( \hat{y}(k) \) according to (30) and (31).

4. Set

\[
v = y(k) - \hat{y}(k), \quad g = r^T(k)P(k-1)r(k).
\]

5. Calculate indicators \( a_+ \) and \( a_- \) allowing for heuristic modification of the OBE algorithm (Walter and Pronzato 1997).

\[
a_+ = \max \left( \frac{v - 1}{\sqrt{g}}, -1 \right), \quad a_- = \max \left( -\frac{v + 1}{\sqrt{g}}, -1 \right).
\]

6. As \( a_+ \) and \( a_- \) were modified, \( v \) and \( g \) have to be updated:

\[
v = \frac{a_- - a_+}{a_- + a_+}, \quad g = \left( \frac{2}{a_- + a_+} \right)^2.
\]
(7) If \( a_- > 1 \) or \( a_+ > 1 \), then terminate the algorithm as condition (28) is contradictory.

(8) If \( a_+a_- > 1/2 \), then set \( k \leftarrow k + 1 \), and go to step 3.

(9) Calculate the variable \( \lambda \) which is obtained from the minimisation of a measure that reflects the geometrical size of the ellipsoid \( E(k) \):

\[
\lambda = \frac{-\alpha_2 + \sqrt{\alpha_2^2 - 4\alpha_1\alpha_3}}{2\alpha_1},
\]

where:

\[
\alpha_1 = g^2,
\alpha_2 = g^2\left[\frac{1}{2}(a_+^2 + a_-^2) + 2a_-a_+ - 1 + \frac{2\nu^2}{g}\right],
\alpha_3 = g^2\left(\frac{a_+ + a_-}{2}\right)(2a_-a_+ - 1).
\]

(10) Calculate coefficient \( c_1(\lambda) \) allowing us to obtain matrix \( P(k) \) specifying the subsequent ellipsoid \( E(\hat{p}(k), P(k)) \):

\[
c_1(\lambda) = 1 + \lambda - \frac{\lambda^2}{1 + \lambda g}.
\]

(11) Update parameter estimate \( \hat{p} \) and matrix \( P \):

\[
\hat{p}(k) = \hat{p}(k - 1) + \lambda \nu \bar{P}(k)p(k),
\]

\[
P(k) = c_1(\lambda) \bar{P}(k),
\]

where

\[
\bar{P}(k) = \left[ I - \frac{P(k - 1)p(k)p^T(k)}{\lambda^{-1} + r^T(k)P(k - 1)p(k)} \right] p(k - 1).
\]

(12) If \( k = n_l \), then STOP else \( k \leftarrow k + 1 \), and go to step 2.

6. Estimation of the GMDH neural model uncertainty

The methodology described in Section 5 makes it possible to obtain the parameter estimate \( \hat{p} \) and the associated admissible parameter set \( E \). But from the point of view of practical applications it is more important to obtain the model output uncertainty, i.e. the interval in which the ‘true’ model output \( \hat{y}(k) \) can be found. The range of the confidence interval of the partial model output depends on the size and the orientation of the ellipsoid which defines the admissible parameter set \( E \) (cf. Figure 9). Taking the minimal and maximal values of the admissible parameter set \( E \) into consideration it is possible to determine the minimal and maximal values of the model output uncertainty interval for each partial model of the GMDH neural network:

\[
r^T(k)\hat{p} - \sqrt{r^T(k)Pr(k)} \leq r^T(k)p \leq r^T(k)\hat{p} + \sqrt{r^T(k)Pr(k)}.
\]

It should be pointed out that the values of the interval (36) change along with the changes of the regressor values in the time \( k \). The Figure 10 presents the model output uncertainty interval and the ‘true’ model output for the error-free regressor case.

The partial models in the \( l \)-th (\( l > 1 \)) layer of the GMDH neural network are based on outputs incoming from the \((l - 1)\)-th layer. Since (36) describes the model output uncertainty interval in the \((l - 1)\)-th layer, parameters of the partial models in the next layers have to be obtained with an approach that solves the problem of an uncertain regressor. Let us denote an unknown ‘true’ value of the regressor \( r_i^*(k) \) by a difference between a known (measured) value of the regressor \( r(k) \) and the error in the regressor \( e(k) \):

\[
r_i^*(k) = r(k) - e(k),
\]

\[
\begin{align*}
\mathbf{r}^T(k)p &= \mathbf{r}^T(k)\hat{p} + \sqrt{\mathbf{r}^T(k)\mathbf{P}r(k)} \\
\mathbf{r}^T(k)p &= \mathbf{r}^T(k)\hat{p} - \sqrt{\mathbf{r}^T(k)\mathbf{P}r(k)}
\end{align*}
\]

Figure 9. Relation between the size of the ellipsoid and the model output uncertainty.

Figure 10. Model output uncertainty interval for the error-free regressor.
where the regressor error $e(k)$ is bounded as follows:
\[-\epsilon_i \leq e_i(k) \leq \epsilon_i, \quad i = 1, \ldots, n_p.\]  
(38)

Substituting (37) into (36) it can be shown that the partial models output uncertainty interval have the following form:
\[
\hat{y}^m(k) \leq r^T(k)p \leq \hat{y}^M(k),
\]
(39)

where
\[
\hat{y}^m(k) = r^T_n(k)\hat{p} + e^T(k)\hat{p} - \sqrt{(r_n(k) + e(k))^TP(r_n(k) + e(k))},
\]
(40)
\[
\hat{y}^M(k) = r^T_n(k)\hat{p} + e^T(k)\hat{p} + \sqrt{(r_n(k) + e(k))^TP(r_n(k) + e(k))}.
\]
(41)

In order to obtain the final form of the expression (39) it is necessary to take into consideration the bounds of the regressor error (38) in the expressions (40) and (41):
\[
\hat{y}^m(k) = r^T_n(k)\hat{p} + \sum_{i=1}^{n_p} \text{sgn} \left( \hat{\beta}_i \right) \hat{\beta}_i \epsilon_i - \sqrt{r^T_n(k)P r_n(k)},
\]
(42)
\[
\hat{y}^M(k) = r^T_n(k)\hat{p} + \sum_{i=1}^{n_p} \text{sgn} \left( \hat{\beta}_i \right) \hat{\beta}_i \epsilon_i + \sqrt{r^T_n(k)P r_n(k)},
\]
(43)

where
\[
r_n, i(k) = r_n(k) + \text{sgn}(r_n, i(k))\epsilon_i.
\]
(44)

7. Comparison of the OBE and LMS algorithms

The aim of the present section is to demonstrate superiority of the OBE algorithm over the LMS which is the most often used method in synthesis of the GMDH network. Both approaches will be used to calculate the parameters estimates and the neuron output uncertainty interval. In order to obtain this goal, the following static system was chosen to the modelling:
\[
y(k) = p_1 \sin(u_1^2(k)) + p_2 u_2^2(k) + \epsilon(k),
\]
(45)

where the nominal values of parameters are $p = [0.2, 0.15]^T$, the input data $u(k)$ and the noise $\epsilon(k)$, $k = 1, \ldots, n_T$ are generated according to the uniform distribution, i.e. $u(k) \in \mathcal{U}(0,2)$ and $\epsilon(k) \in \mathcal{U}(-0.05, 0.1)$. Note that the noise does not satisfy (21). The problem is to obtain the parameter estimate $\hat{p}$ and the corresponding neuron uncertainty using the set of input-output measurements $(\{u(k), y(k)\})^n_{k=1}$. In the case of the application of the OBE algorithm, the parameters estimate amounted $\hat{p} = [0.1953, 0.1546]$ and they were calculated according to rules presented in the Section 5. As far as the LMS algorithm is concerned the parameters estimate were obtained following equation:
\[
\hat{p} = \left(R^T R\right)^{-1} R^T y,
\]
(46)

and its results were $\hat{p} = [0.2281, 0.1763]$. The results show that the parameters estimates obtained with the application of the OBE are similar to the nominal parameters $p = [0.2, 0.15]^T$, opposite to parameters estimates calculated with the LMS. It follows from the fact that the condition (21) concerning noise is not fulfilled. In order to calculate the neuron output uncertainty interval with the OBE algorithm, the equation presented in (39) was applied. In a similar manner this interval can be calculated with the LMS approach. A detailed description of determination of model uncertainty in the form of interval was presented in Mrugalski and Korbicz (2005b). According to this method the $(1-\alpha)$ confidence interval for the neuron output (7) has the following form:
\[
\hat{y}(k) - t_{a, n_T-n_p-1} \sqrt{\hat{\sigma}^2 r^T(k)(R^T R)^{-1} r(k)} < r^T(k)p < \hat{y}(k) + t_{a, n_T-n_p-1} \sqrt{\hat{\sigma}^2 r^T(k)(R^T R)^{-1} r(k)},
\]
(47)

where: $t_{a, n_T-n_p-1}$ represents $(1-\alpha)$-th order quantel of a random variable which has a $T$-Student distribution with $(n_T - n_p - 1)$ degrees of freedom, and $\hat{\sigma}^2$ represents a variance of the random variable defined as a difference of the system output and its estimate. For both methods the intervals are calculated for the following common validation signal:
\[
u_1(k) = -0.1 \sin(0.02\pi(k)) + 0.4 \quad \text{for} \quad k = 1, \ldots, 100,
\]
\[
u_2(k) = 0.5 \sin(2\pi(k/50)) + 0.1 \sin(\pi(k/5)) + 1.2 \quad \text{for} \quad k = 1, \ldots, 100.
\]

Figure 11 shows the confidence interval of the model output obtained with the application of the OBE and the LMS. The former was based on the expression (36) and the latter on (47). The results obtained with the LMS indicate that the neuron output uncertainty interval does not contain the system output calculated based on the nominal parameters $p$. Therefore, only the application of the OBE allows us to obtain unbiased parameters estimates and neuron uncertainty.

8. Synthesis of the GMDH neural model via the OBE algorithm

The GMDH neural network is gradually constructed by the connection of the partial models according to
the procedure described in Section 3. In the beginning it is necessary to adapt the OBE algorithm to the parameter estimation of the partial models with the non-linear activation function \( \xi(\cdot) \). In order to avoid the noise additivity problem described in Section 4.3 with the application of the OBE algorithm, it is necessary to transform the following relation:

\[
\varepsilon^m(k) \leq y(k) - \xi \left( \left( \rho_n^0(k) \right)^T \psi_n^0 \right) \leq \varepsilon^M(k) \tag{48}
\]

using \( \xi^{-1}(\cdot) \), and hence:

\[
\xi^{-1}(y(k) - \varepsilon^M(k)) \leq \left( \rho_n^0(k) \right)^T \psi_n^0 \leq \xi^{-1}(y(k) - \varepsilon^m(k)). \tag{49}
\]

The transformation (49) is appropriate if the conditions (23) and (24) concerning the properties of the non-linear activation function \( \xi(\cdot) \) are fulfilled. For the parameters and the confidence estimation of the partial models in the first layer of the GMDH neural network, the OBE algorithm not taking into account the error in the regressor can be applied. The boundary values of the neurons output uncertainty interval are obtained based on inequality (39). The outputs of the selected partial models become the inputs to other partial models in the next layer. Because of the partial model output is known with some confidence defined by the modelling uncertainty. In this way, the partial models with small values of classical evaluation criteria are rejected based on chosen selection methods. Unfortunately, as has been already mentioned in Section 4.3, the application of the classical evaluation criteria like the AIC and FPE (Ivakhnenko and Mueller 1995, Mueller and Lemke 2000) during the network synthesis may lead to the selection of an inappropriate structure of the GMDH neural network. This follows from the fact that the above criteria does not take into account the modelling uncertainty. In this way, the partial models with small values of classical evaluation criteria \( Q_V \) but with large uncertainty can be obtained. In order to overcome this difficulty, the following evaluation criterion can be used:

\[
Q_V = \frac{1}{n_V} \sum_{k=1}^{n_V} \left| \left( \hat{y}^M(k) + \varepsilon^M(k) \right) - \left( \check{y}^m(k) + \varepsilon^m(k) \right) \right|,
\]

where \( n_V \) is the number of input–output measurements for the validation data set, \( \hat{y}^M(k) \) and \( \check{y}^m(k) \) are calculated with (36) for each neuron in the first layer of the GMDH network and with (39) for the neurons in the subsequent layers. Besides, the definition of the evaluation criterion is necessary to apply the appropriate selection method, which ensures the proper

![Figure 11. Confidence intervals of the neuron output.](image-url)

![Figure 12. Accumulation of model output uncertainty in the form of the interval (dotted lines), model response (continuous line).](image-url)
network structure. The selection methods described in Section 3 have several disadvantages which can lead to large uncertainty of the neural model. In order to overcome these difficulties, the method based on the soft selection approach can be applied. An outline of the soft selection method (Mrugalski et al. 2003b) is as follows:

**Input:** The set of all \( n_s \) neurons in the \( l \)-th layer, \( n_j \) – the number of opponent neurons, \( n_w \) – the number of winnings required for the \( n \)-th neuron selection.

**Output:** The set of neurons after selection.

1. Calculate the evaluation criterion \( Q(y_n^{(j)}) \) for \( n = 1, \ldots, n_s \) neurons
2. Conduct series of \( n_j \) competitions between each \( n \)-th neuron in the layer and \( n_j \) randomly selected neurons (the so-called opponent) from the same layer. The \( n \)-th neuron is so-called winner neuron when:
   \[
   Q(y_n^{(j)}) \leq Q(y_j^{(j)}), \quad j = 1, \ldots, n_j
   \]
   where \( y_n^{(j)} \) denotes a signal generated by the opponent neuron
3. Select the neurons for the \((l + 1)\)-th layer with the number of winnings bigger then \( n_w \) (the remaining neurons are removed)

The property of the soft selection follows from the specific series of competitions. It may happen that the potentially unfitted neuron will be selected. Everything depends on its score in the series of competition. In this way, distinct from other selection methods, it is possible to use potentially unfitted neurons which in the next layers may improve the quality of the model. Moreover, if the neural network is not fitted perfectly to the identification data set, it is possible to achieve a network which possesses better generalisation abilities. One of the most important parameters which should be chosen in the selection process is the number of \( n_j \) opponents. The bigger value of \( n_j \) makes that the probability of the selection of a neuron with small quality index is lower. In this way, in extreme situations when \( n_j \gg n_w \) the soft selection method will behave as the constant population method which is based on the selection only of the best fitted neurons. Some experimental results performed on a number of selected examples indicate that soft selection method makes it possible to obtain a more flexible network structure. Another advantage, compared to the optimal population method, is that an arbitrary selection of the threshold is avoided. Instead of this we have to select a number of winnings \( n_w \). It is of course a less sophisticated task.

9. Robust fault detection for the intelligent actuator

In Section 2, the model-based fault detection scheme was presented. Moreover, it was shown that the model uncertainty and the disturbances are the elementary factors influencing on the reliability and performance of fault diagnosis system. Since the model uncertainty as well as disturbances are inevitable in industrial systems, a significant pressure exists to create the robust fault diagnosis systems. To tackle this problem, the proposed GMDH neural model can be applied. Knowing the model structure and possessing the knowledge regarding its uncertainty it is possible to design a robust fault detection scheme. The model output uncertainty interval, calculated with the application of the GMDH model for the same input signals, should contain the real system response in the fault free mode. The range of model output confidence interval depends on the GMDH model uncertainty. As the measurements of the system response are corrupted by the noise and it is not possible to obtain their exact value, it is necessary to modify the model output uncertainty interval. This modification relies on adding the boundary values of the output error (28) to the model output uncertainty interval (42) and (43).

\[
\begin{align*}
\tilde{r}_n^T(k) \hat{p} & - \sum_{i=1}^{n_p} \text{sgn}(\hat{p}_i) \hat{p}_i \epsilon_i - \sqrt{\tilde{r}_n^T(k) \hat{p}_n \epsilon(k)} - \epsilon(k) \leq y(k) \\
& \leq \tilde{r}_n^T(k) \hat{p} + \sum_{i=1}^{n_p} \text{sgn}(\hat{p}_i) \hat{p}_i \epsilon_i + \sqrt{\tilde{r}_n^T(k) \hat{p}_n \epsilon(k) + \epsilon(k)}.
\end{align*}
\]

The newly defined interval (Figure 13) is called the system output uncertainty interval and it is calculated for the partial model in the last GMDH neural network layer, which generates the model output. An occurrence of the fault is signalled when the system output signal crosses the system output uncertainty interval.

9.1. DAMADICS benchmark – the intelligent actuator

In order to show the effectiveness of the GMDH model-based fault detection system, the RTN DAMADICS benchmark was employed. The Research Training Network on Development and Application of Methods for Actuator Diagnosis in Industrial Control Systems (DAMADICS) was focused on the diagnosis of valve plant actuators and looks towards real implementation methods for new actuator systems. In particular, the RTN DAMADICS focused on the diagnosis of a 5-stage evaporation and boiler station process of the Lublin Sugar Factory S.A. in Poland (Bartys et al. 2006). Three actuators have been chosen for research purposes. Two actuators are
connected with the evaporation station. The first one is situated on the inflow of thin juice into the evaporation station, and the second one is situated on the outlet of thick juice from the fifth section of the evaporation section. The third actuator connected with the fourth boiler house is situated on the water inflow into boiler drum. The element selected for modelling and fault detection, the third actuator, is a final control element, which interacts with the controlled process. The input of the actuator is the output of the process controller (flow or level controller) and the actuator modifies the position of the valve allowing a direct effect on the primary variable in order to follow the flow or level set-point. Each actuator is equipped with similar measuring devices shown in Figure 14, where $V_1$, $V_2$ and $V_3$ denotes the bypass valves, $ACQ$ and $CPU$ are the data acquisition and the positioner central processing units. $E/P$ and $FT$ are the electro-pneumatic and valve flow transducers. Finally, $DT$ and $PT$ represent the displacement and the pressure. On the ground of the process analysis and taking into account the expert process knowledge, the following model of the juice flow at the outlet of the valve $F = r_F(X, P_1, P_2, T_1)$, and the servomotor rod displacement $X = r_X(C_v, P_1, P_2, T_1)$ were considered, where $r_F(\cdot)$ and $r_X(\cdot)$ denote the modelled relationships, $C_v$ is the control valve, $P_1$ and $P_2$ are the pressures at the inlet and the outlet of the valve, respectively, and $T_1$ represents the juice temperature at the inlet of the valve.

The data used for the system identification and the fault detection were generated with the application of the DAMADICS MATLAB SIMULINK actuator model (Bartyś et al. 2006). This tool makes it possible to generate data for 19 different faults, which are described in Table 1. In the benchmark scenario, the abrupt A and incipient I faults are considered.

![Figure 13. Fault detection via system output uncertainty interval.](image)

![Figure 14. Diagram of the actuator.](image)
Furthermore, the abrupt faults can be regarded as small S, medium M and big B according to the benchmark description. The synthesis process of the GMDH neural network proceed according to the steps described in Section 3. As was mentioned in Section 4.1, the choice of the proper partial model structure is an important problem in the GMDH approach. In this work, the dynamic neurons with the hyperbolic tangent activation functions were employed. Dynamics in partial models was realised by introduction of the IIR filter. The order of the IIR filters in each layer of the network was obtained with the application of the Lipschitz index approach based on so-called Lipschitz quotients. The selection of best performing neurons in each layer of the GMDH network in terms of their processing accuracy was realised with the application of the soft selection method which is based on the evaluation criterion (50). The values of this criterion were calculated separately for each neuron in the GMDH network, whereas \( \tilde{y}^M(k) \) and \( \tilde{y}^I(k) \) in (50) were obtained with (36) for the neurons in the first layer of the GMDH network and with (39) for the subsequent ones.

Table 1 presents the results for the subsequent layers, i.e. these values were obtained for the best performing partial models in a particular layer. Additionally, for the sake of comparison, the results based on the AIC evaluation criterion (12) are presented as well. The results presented in Table 1 show that the gradual decrease of value of the evaluation criteria occurs when a new layer of the GMDH network is introduced. It follows from the increasing of the model complexity as well as its modelling abilities. However, when the model is too complex the quality index \( Q_V \) increases. This situation occurs when the 5th layer of the network is added. It means that the neural model corresponding to \( F=r_F(\cdot) \) and \( X=r_X(\cdot) \) should have only four layers. From Table 1, it can be also seen that the application of the AIC evaluation criterion gives similar results for \( F=r_F(\cdot) \), i.e. the same number of layers was selected, whilst for \( X=r_X(\cdot) \) it leads to the selection of too simple a structure, i.e. a network with only two layers is selected. It implies that the quality index \( Q_V \) and \( Q_{AIC} \) makes it possible to obtain a model with smaller uncertainty.

After the synthesis of the \( F=r_F(\cdot) \) and \( X=r_X(\cdot) \) GMDH models, it is possible to employ them for robust fault detection. This task can be realised with the application of the system output interval obtained according to (51). Figures 15–18 present the system responses and the corresponding system output uncertainty intervals for the faulty data.

Table 2 shows the results of fault detection of all the considered faults. The notation given in Table 2 can be explained as follows: ND means that it is impossible to detect a given fault, \( D_F \) or \( D_X \) means that it is possible to detect a fault with \( r_F(\cdot) \) or \( r_X(\cdot) \), respectively, while \( D_{FX} \) means that a given fault can be detected with both \( r_F(\cdot) \) or \( r_X(\cdot) \). From the results presented in Table 2 it can be seen that it is impossible to detect faults \( f_9, f_{10} \) and \( f_{14} \). Moreover, some small and medium faults cannot be detected, i.e. \( f_9 \) and \( f_{12} \). This

<table>
<thead>
<tr>
<th>( f )</th>
<th>Faults description</th>
<th>S</th>
<th>M</th>
<th>B</th>
<th>I</th>
</tr>
</thead>
<tbody>
<tr>
<td>( f_1 )</td>
<td>Valve clogging</td>
<td>ND</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>( f_2 )</td>
<td>Valve plug or valve seat sedimentation</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>( f_3 )</td>
<td>Valve plug or valve seat erosion</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>( f_4 )</td>
<td>Increased of valve or busing friction</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>( f_5 )</td>
<td>External leakage</td>
<td>ND</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>( f_6 )</td>
<td>Internal leakage (valve tightness)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>( f_7 )</td>
<td>Medium evaporation or critical flow</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>( f_8 )</td>
<td>Twisted servomotor’s piston rod</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>( f_9 )</td>
<td>Servomotors housing or terminals tightness</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>( f_{10} )</td>
<td>Servomotor’s diaphragm perforation</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>( f_{11} )</td>
<td>Servomotor’s spring fault</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>( f_{12} )</td>
<td>Electro-pneumatic transducer fault</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>( f_{13} )</td>
<td>Rod displacement sensor fault</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>( f_{14} )</td>
<td>Pressure sensor fault</td>
<td>ND</td>
<td>ND</td>
<td>ND</td>
<td></td>
</tr>
<tr>
<td>( f_{15} )</td>
<td>Positioner feedback fault</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>( f_{16} )</td>
<td>Positioner supply pressure drop</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>( f_{17} )</td>
<td>Unexpected pressure change across the valve</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>( f_{18} )</td>
<td>Fully or partly opened bypass valves</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>( f_{19} )</td>
<td>Flow rate sensor fault</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Layer</th>
<th>( r_F(\cdot) )</th>
<th>( r_F(\cdot) )</th>
<th>( r_X(\cdot) )</th>
<th>( r_X(\cdot) )</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1.1034</td>
<td>0.5998</td>
<td>0.3198</td>
<td>0.1878</td>
</tr>
<tr>
<td>2</td>
<td>1.0633</td>
<td>0.5711</td>
<td>0.2931</td>
<td>0.1756</td>
</tr>
<tr>
<td>3</td>
<td>1.0206</td>
<td>0.5525</td>
<td>0.2895</td>
<td>0.1771</td>
</tr>
<tr>
<td>4</td>
<td>0.9434</td>
<td>0.5138</td>
<td>0.2811</td>
<td>0.1749</td>
</tr>
<tr>
<td>5</td>
<td>1.9938</td>
<td>0.5528</td>
<td>0.2972</td>
<td>0.1719</td>
</tr>
</tbody>
</table>
situation can be explained by the fact that the effect of these faults is at the same level as the effect of noise.

10. Conclusion

This article proposes a complete design procedure of the neural model. The presented approach makes it possible to solve several problems which appear during the system identification via neural networks. One of the most important problems, which relays on the neural model structure selection, was solved by the application of the GMDH algorithm. The main advantage of the presented approach, compared to the classical MLP, is that GMDH networks have a structure that grows to fit the particular tasks being considered. It enables them to perform better than networks with fixed structures. The proper partial models structure selection and the application of the proper evaluation criteria and the selection methods can improve the neural model quality. Apart from the contribution of the structure errors to the model uncertainty the parameter estimates inaccuracy also influences the model quality. For this reason, in this article it was shown how to estimate the parameters and corresponding uncertainty of an individual partial model. The proposed solution is based on the OBE algorithm which is superior to the celebrated LMS in many practical applications. The methodology developed for the parameter and uncertainty estimation of the partial models makes it possible to formulate an algorithm that allows obtaining a neural network with relatively small modelling uncertainty.

In the illustrative part of this article, an example concerning the practical application of the developed approach was presented. The calculation of the model
uncertainty in the form of the system output uncertainty interval allowed us to perform robust fault detection of the industrial systems. In particular, the proposed approach was tested on the DAMADICS benchmark problem. The obtained results show that almost all faults can be detected except for a few incipient or small ones.

Notes on contributors

Josef Korbicz has been a full-rank professor of automatic control at the University of Zielona Góra, Poland, since 1994. He currently heads the Institute of Control and Computation Engineering (ICCE). Born in Poland on 19 March 1951, Josef Korbicz received the M.Sc., PhD. and D.Sc. (doctor habilitatis) degrees in automatic control from the Kiev University of Technology, Ukraine in 1975, 1980 and 1986, respectively. He obtained his professorial title from the Institute of System Research of the Polish Academy of Sciences, Warsaw, in 1993. In 1991 he was with the University of Colorado, USA, as an IREX research fellow, and then in 1994 with the Universities of Duisburg and Wuppertal, Germany, as a DAAD research fellow. In 1991 he founded the International Journal of Applied Mathematics and Computer Science (AMCS) and up to now he has been the Editor-in-Chief. Moreover, together with prof. J.M. Kosičyn, he founded (1996) the Polish SAFEPROCESS conferences, the so-called DPS. His current research interests include computational intelligence, fault detection and isolation (FDI) and control theory. The primary aim of his research group is to contribute towards the diagnosis of dynamical systems. His research projects in this field have been sponsored by the State Committee for Scientific Research in Poland and since 1997 by the European Commission: INCO-Copernicus (1997–1999) and DAMADICS, (2000–2004). Josef Korbicz has published more than 326 technical papers, 48 of them in international journals. He is a co-author of 8 monographs and text books and a co-editor of 4 books. His one of the last book (co-editor) is entitled Fault Diagnosis. Models, Artificial Intelligence, Applications, Springer-Verlag (2004). Professor Korbicz is a senior member of IEEE, and member of IFAC TC on SAFEPROCESS. He was the IPC chairman of the IFAC Symposium on SAFEPROCESS held in Beijing, China (August 2006). Moreover he is the member of the Polish Academy of Sciences since 2007.

Marcin Mrugalski was born in Poland on 22 November 1975. He received the M.Sc. and PhD. degrees in electrical engineering from the University of Zielona Góra, Poland in 1999 and 2004, respectively. In 2004, he was with the University of Hull, UK, as a Research Fellow. He has been an Assistant Professor of automatic control and robotics at the Institute of Control and Computation Engineering, University of Zielona Góra, since 2004. His current research interests include neural networks, computational intelligence and fault diagnosis. He has taken part in the realization of a number of research projects sponsored by the Ministry of Scientific Research and Information Technology in Poland and since 1999 by the European Commission: INCO-Copernicus on Integration of quantitative and qualitative fault diagnosis methods within the framework of industrial application (1997–1999), the 5th FP EU RTN on Development and Application of Methods for Actuator Diagnosis in Industrial Control Systems (DAMADICS) (2000–2004). He has published more than 35 papers in international journals, book chapters and conference proceedings.

References


