

FAULT DIAGNOSIS OF NON-LINEAR DYNAMICAL SYSTEMS USING ANALYTICAL AND SOFT COMPUTING METHODS

Józef Korbicz

Abstract:

The paper deals with the problems of robust fault detection using analytical methods (observers and unknown input observers) and soft computing techniques (neural networks, neuro-fuzzy networks and genetic programming). The model-based approach to Fault Detection and Isolation (FDI) is considered. In particular, observers for non-linear Lipschitz systems and extended unknown input observers are discussed. In the case of soft computing techniques, the main objective is to show how to employ the bounded-error approach to determine the uncertainty of the GMDH and neuro-fuzzy networks. It is shown that based on soft computing models uncertainty defined as a confidence range for the model output, adaptive thresholds can be defined. The final part of the paper presents two illustrative examples that confirm the effectiveness of the unknown input observers and the neuro-fuzzy networks approaches.

Keywords: *Fault detection, unknown input observer, dynamical neural networks, neuro-fuzzy systems, evolutionary algorithms*

1. Introduction

A continuous increase in the complexity, efficiency, and reliability of modern industrial systems necessitates a continuous development in the control and fault diagnosis theory and practice (Blanke *et al.*, 2003; Chen and Patton, 1999; Isermann, 2006; Korbicz *et al.*, 2004; Kościelny, 2001; Rodrigues *et al.*, 2007). These requirements extend beyond the normally accepted safety-critical systems of nuclear reactors, chemical plants or aircrafts, to new systems such as autonomous vehicles or fast rail systems. An early detection and maintenance of faults can help avoid system shutdown, breakdowns and even catastrophes involving human fatalities and material damage.

The core of the fault detection and isolation system is the so-called model-based approach. In the general case, this concept can be implemented using various kinds of models: analytical, knowledge-based and data-based ones (Köppen-Seliger and Frank, 1999), which are used to model a diagnosed system working in normal-operation or faulty conditions. Conventional model-based fault detection techniques make use of analytical or quantitative models (Korbicz *et al.*, 2004), mostly in the framework of observers or Kalman filters (Chen and Patton, 1999; Witczak, 2007; Zolghadri *et al.*, 1996). The dynamic behaviour of the system is described by differential equations or transfer functions together with the respective parameter values.

Unfortunately, the analytical model-based approach is usually restricted to simpler systems described by linear models. When there are no mathematical models of the diagnosed system or the complexity of the dynamic model increases and the task of modeling is hard, an analytical model cannot be applied in the fault diagnosis system nor give satisfactory results. Currently many efforts are made to use knowledge-based or qualitative or data-based models (Patton and Korbicz, 1999; Korbicz, 2006a). They represent system behaviour in terms of heuristic or qualitative knowledge (Frank, 1990; Korbicz, 2004). The relationship between inputs and outputs may be described by a rule base or by a set of parameters that have to be determined during an identification stage based on the learning data set. In this case data-based models, such as neural networks (Korbicz *et al.*, 1999), fuzzy sets (Frank and Köppen-Seliger, 1997; Kowal, 2005), the evolutionary algorithms (Chen and Patton, 1999; Obuchowicz, 2003; Witczak *et al.*, 2002) or their combination (Patton *et al.*, 2005; Korbicz *et al.*, 2001), can be considered.

Irrespective of the modelling method used (analytical or soft computing), there is always the problem of model uncertainty, i.e., the model-reality mismatch. Thus, the better the model used to represent system behaviour, the better the chance of improving the reliability and performance in diagnosing faults. Indeed, disturbances as well as model uncertainty are inevitable in industrial systems (Patton *et al.*, 2006), and hence there exists a pressure creating the need for robustness in fault diagnosis systems. This robustness requirement is usually achieved at the fault detection stage, i.e., the problem is to develop residual generators which should be insensitive (as far as possible) to model uncertainty and real disturbances acting on a system while remaining sensitive to faults. In one way or another, all the above-mentioned approaches can realise this requirement for linear systems.

Taking into account the above conditions, a large amount of knowledge on designing robust fault diagnosis systems has been accumulated through the literature since the beginning of the 1980s. For a comprehensive survey regarding such techniques, the reader is referred to the excellent monographs (Chen and Patton, 1999; Gertler, 1998; Korbicz *et al.*, 2004; 2002; Patton *et al.*, 2000). The most common approach to robust fault diagnosis is to use robust observers. This is mainly because of the fact that the theory of robust observers is relatively well developed in the control engineering literature.

The main objective of this survey paper is to present recent developments in modern fault diagnosis with non-linear observers, neural networks, neuro-fuzzy networks

and genetic programming. In particular, the paper is organised as follows: Section 2 outlines the problem of observer-based robust fault diagnosis and presents two different observer structures that can be employed for non-linear systems. The genetic programming approach in modelling of diagnosed systems in Section 3 is presented. Section 4 presents alternative neural network-based approaches that can be used to settle the fault diagnosis problem when the mathematical state-space model is not available. The neuro-fuzzy structure optimization problem in Section 5 is considered. The subsequent Section 6 presents application of observer approach for fault detection of an induction motor and neuro-fuzzy approach in the diagnosis of an intelligent actuator.

2. Analytical models in fault detection systems

The analytical redundancy of measurement line exists when an additional value of process variable is obtained using mathematical model that connects the calculated variables with other measured signals. Analytical redundancy is used for fault detection where the analytical model of diagnosed system is most important part. In general analytical models applied to fault detection can be divided in following groups (Chen and Patton, 1999; Gertler, 1998; Korbicz *et al.*, 2004):

physical models (equations of movement, balance equations, etc.),

input-output type linear models (continuous or discrete transmittances),

state linear and non-linear equations.

Among these models a special role belongs to state space equations that are applied for designing of state observers (Korbicz and Bidyuk, 1993).

The basic idea underlying observer-based (or filter-based, in the stochastic case) approaches to fault detection is to obtain the estimates of certain measured and/or unmeasured signals. Then, in the most usual case, the estimates of the measured signals are compared with their originals, i.e., the difference between the original signal and its estimate is used to form a residual signal $r_k = y_k - \hat{y}_k$. To tackle this problem, many different observers (or filters) can be employed, e.g., Luenberger observers, Kalman filters, etc.

The admiration for observer-based fault detection schemes is caused by the still increasing popularity of state-space models as well as the wide usage of observers in modern control theory and applications. Due to such conditions, the theory of observers (or filters) seems to be well developed (especially for linear systems). This has made a good background for the development of observer-based FDI schemes.

Irrespective of the linear or non-linear FDI technique being employed (Witczak, 2003), FDI performance will be usually impaired by the lack of robustness to model uncertainty. As can be observed in the literature (Chen and Patton, 1999; Gertler, 1998; Korbicz *et al.*, 2004; Patton *et al.*, 2000), the most common approach to robust fault diagnosis is to use robust observers. In particular, the so-called unknown input model uncertainty is mostly preferred. The observer resulting from such an approach is

called the Unknown Input Observer (UIO). Although the origins of UIOs can be traced back to the early 1970s (cf. the seminal work of Wang *et al.* (1975)), the problem of designing such observers is still of paramount importance both from the theoretical and practical viewpoints (Hui and Žak, 2005). The main objective of the subsequent part of this section is to present two unknown input observer design strategies that can be employed for the Lipschitz (Korbicz *et al.*, 2007; Witczak and Korbicz, 2006; Witczak *et al.*, 2006b) and a general class of non-linear systems (Witczak *et al.*, 2002; 2006), respectively.

2.1. Observers for non-linear Lipschitz systems

Let us consider Lipschitz systems that can be described as follows:

$$\mathbf{x}_{k+1} = \mathbf{A}\mathbf{x}_k + \mathbf{B}\mathbf{u}_k + \mathbf{h}(\mathbf{y}_k, \mathbf{u}_k) + \mathbf{g}(\mathbf{x}_k, \mathbf{u}_k), \quad (1)$$

$$\mathbf{y}_{k+1} = \mathbf{C}\mathbf{x}_{k+1}, \quad (2)$$

where $\mathbf{x}_k \in \mathbb{R}^n$ stands for the state vector, $\mathbf{y}_k \in \mathbb{R}^m$ is the output, $\mathbf{u}_k \in \mathbb{R}$ is the input, and $\mathbf{g}(\bullet)$ and $\mathbf{h}(\bullet)$ are non-linear functions. Additionally, $\mathbf{g}(\bullet)$ satisfies

$$\|\mathbf{g}(\mathbf{x}_1, \mathbf{u}) - \mathbf{g}(\mathbf{x}_2, \mathbf{u})\|_2 \leq \gamma \|\mathbf{x}_1 - \mathbf{x}_2\|_2, \quad \forall \mathbf{x}_1, \mathbf{x}_2, \mathbf{u}, \quad (3)$$

and $\gamma > 0$ stands for the Lipschitz constant.

Let us consider an observer for the system (1)-(2) described by the following equation:

$$\hat{\mathbf{x}}_{k+1} = \mathbf{A}\hat{\mathbf{x}}_k + \mathbf{B}\mathbf{u}_k + \mathbf{h}(\mathbf{y}_k, \mathbf{u}_k) + \mathbf{g}(\hat{\mathbf{x}}_k, \mathbf{u}_k) + \mathbf{K}(\mathbf{y}_k - \mathbf{C}\hat{\mathbf{x}}_k), \quad (4)$$

where $\hat{\mathbf{x}}_k$ denotes the state estimate and \mathbf{K} stands for the gain matrix.

The subsequent part of this section shows theorem that presents convergence conditions of (4). Following Thau (1973), let us assume that the pair (\mathbf{A}, \mathbf{C}) is observable. Let $\mathbf{P} = \mathbf{P}^T$, $\mathbf{P} > \mathbf{0}$ be a solution of the following Lyapunov equation:

$$\mathbf{Q} = \mathbf{P} - \mathbf{A}_0^T \mathbf{P} \mathbf{A}_0, \quad \mathbf{A}_0 = \mathbf{A} - \mathbf{K} \mathbf{C}, \quad (5)$$

where \mathbf{A}_0 is a stable matrix, i.e., $\rho(\mathbf{A}_0) < 1$, and $\mathbf{Q} = \mathbf{Q}^T$, $\mathbf{Q} > \mathbf{0}$. Moreover, let $\underline{\sigma}(\cdot)$ and $\bar{\sigma}(\cdot)$ stand for the minimum and maximum singular values of its argument, respectively.

Theorem 1. (Witczak and Korbicz, 2006) *Let us consider the observer (4) for the systems described by (1)-(2). If the Lipschitz constant γ (cf. (3)) satisfies*

$$\gamma < \sqrt{\frac{\underline{\sigma}(\mathbf{Q} - \mathbf{A}_0^T \mathbf{P} \mathbf{A}_0)}{\bar{\sigma}(\mathbf{P}) + 1}}, \quad \mathbf{Q} - \mathbf{A}_0^T \mathbf{P} \mathbf{A}_0 > \mathbf{0}, \quad (6)$$

then the observer (4) is asymptotically convergent.

Unfortunately, the condition (6) may merely serve as a method of checking the convergence, but the gain matrix \mathbf{K} has to be determined beforehand. This means that the design procedure boils down to selecting various gain

matrices \mathbf{K} , solving the Lyapunov equation (5), and then checking the convergence condition (6). There is no doubt that this is an ineffective and inconvenient approach.

To tackle such a challenging problem, an effective design procedure was proposed in (Witczak and Korbicz, 2006), which can be written as follows:

Step 1: Obtain γ for (1)-(2).

Step 2: Solve a set of linear matrix inequalities:

$$\begin{bmatrix} \beta \mathbf{I} & \mathbf{P} \\ \mathbf{P} & \beta \mathbf{I} \end{bmatrix} \succ 0, \quad \beta > 0, \quad \mathbf{P} \succ 0, \quad (7)$$

$$\begin{bmatrix} \mathbf{X} & \mathbf{A}^T \mathbf{P} - \mathbf{C}^T \mathbf{L}^T \\ \mathbf{P} \mathbf{A} - \mathbf{L} \mathbf{C} & \mathbf{I} \end{bmatrix} \succ 0, \quad (8)$$

$$\begin{bmatrix} \mathbf{P} - \gamma^2(\beta + 1)\mathbf{I} - \mathbf{X} & \mathbf{A}^T \mathbf{P} - \mathbf{C}^T \mathbf{L}^T \\ \mathbf{P} \mathbf{A} - \mathbf{L} \mathbf{C} & \mathbf{P} \end{bmatrix} \succ 0. \quad (9)$$

Step 3: Obtain the gain matrix $\mathbf{K} = \mathbf{P}^{-1}\mathbf{L}$.

In the next part of this section a straightforward approach for extending the techniques proposed in the preceding sections to discrete-time Lipschitz systems with unknown inputs is presented. The system can be described as follows

$$\mathbf{x}_{k+1} = \mathbf{A}\mathbf{x}_k + \mathbf{B}\mathbf{u}_k + \mathbf{h}(\mathbf{y}_k, \mathbf{u}_k) + \mathbf{g}(\mathbf{x}_k, \mathbf{u}_k) + \mathbf{E}\mathbf{d}_k, \quad (10)$$

$$\mathbf{y}_{k+1} = \mathbf{C}\mathbf{x}_{k+1}, \quad (11)$$

where $\mathbf{d}_k \in \mathbb{R}^q$ is the unknown input, and \mathbf{E} is a known unknown input distribution matrix. In order to use the techniques described above for state estimation of the system (10)(11), it is necessary to introduce some modifications concerning the unknown input. Let us assume that

$$\text{rank}(\mathbf{C}\mathbf{E}) = \text{rank}(\mathbf{E}) = q, \quad (12)$$

(see (Chen and Patton, 1999, p. 72, Lemma 3.1). If the condition (12) is satisfied, then it is possible to calculate $\mathbf{H} = (\mathbf{C}\mathbf{E})^+ = [(\mathbf{C}\mathbf{E})^T \mathbf{C}\mathbf{E}]^{-1} (\mathbf{C}\mathbf{E})^T$, where $(\cdot)^+$ stands for the pseudo-inverse of its argument. By multiplying (11) by \mathbf{H} and then inserting (10), it can be shown that

$$\mathbf{x}_{k+1} = \bar{\mathbf{A}}\mathbf{x}_k + \bar{\mathbf{B}}\mathbf{u}_k + \bar{\mathbf{h}}(\mathbf{u}_k, \mathbf{y}_k) + \bar{\mathbf{g}}(\mathbf{x}_k, \mathbf{u}_k) + \bar{\mathbf{E}}\mathbf{y}_{k+1}, \quad (13)$$

where

$$\begin{aligned} \bar{\mathbf{A}} &= \bar{\mathbf{C}}\mathbf{A}, & \bar{\mathbf{B}} &= \bar{\mathbf{C}}\mathbf{B}, & \bar{\mathbf{g}}(\cdot) &= \bar{\mathbf{C}}\mathbf{g}(\cdot), \\ \bar{\mathbf{h}}(\cdot) &= \bar{\mathbf{C}}\mathbf{h}(\cdot), & \bar{\mathbf{C}} &= \mathbf{I} - \mathbf{E}\mathbf{H}\mathbf{C}, & \bar{\mathbf{E}} &= \mathbf{E}\mathbf{H}. \end{aligned}$$

Thus, the unknown input observer for (10)-(11) is given as follows:

$$\hat{\mathbf{x}}_{k+1} = \bar{\mathbf{A}}\hat{\mathbf{x}}_k + \bar{\mathbf{B}}\mathbf{u}_k + \bar{\mathbf{h}}(\mathbf{u}_k, \mathbf{y}_k) + \bar{\mathbf{g}}(\hat{\mathbf{x}}_k, \mathbf{u}_k) + \bar{\mathbf{E}}\mathbf{y}_{k+1} + \mathbf{K}(\mathbf{y}_k - \mathbf{C}\hat{\mathbf{x}}_k). \quad (14)$$

A simple comparison of (1) and (13) leads to the conclusion that the observer (14) can be designed with the

above-mentioned three-step procedure, taking into account the fact that:

$$\|\bar{\mathbf{g}}(\mathbf{x}_1, \mathbf{u}) - \bar{\mathbf{g}}(\mathbf{x}_2, \mathbf{u})\|_2 \leq \bar{\gamma}\|\mathbf{x}_1 - \mathbf{x}_2\|_2, \quad \forall \mathbf{x}_1, \mathbf{x}_2, \mathbf{u}, \quad (15)$$

and assuming that the pair $(\bar{\mathbf{A}}, \bar{\mathbf{C}})$ is observable.

2.2. Extended unknown input observers and genetic programming

Let us consider a non-linear discrete-time system described by

$$\mathbf{x}_{k+1} = \mathbf{g}(\mathbf{x}_k) + \mathbf{h}(\mathbf{u}_k) + \mathbf{E}_k \mathbf{d}_k, \quad (16)$$

$$\mathbf{y}_{k+1} = \mathbf{C}_{k+1} \mathbf{x}_{k+1}, \quad (17)$$

where \mathbf{d}_k is the unknown input, and \mathbf{E}_k is the known matrix. Using the similar approach as in Section 2.1, it can be shown (Witczak *et al.*, 2006) that the structure of the so-called Extended Unknown Input Observer (EUIO) is

$$\hat{\mathbf{x}}_{k+1} = \hat{\mathbf{x}}_{k+1/k} + \mathbf{K}_{k+1}(\mathbf{y}_{k+1} - \mathbf{C}_{k+1}\hat{\mathbf{x}}_{k+1/k}), \quad (18)$$

where

$$\hat{\mathbf{x}}_{k+1/k} = \bar{\mathbf{g}}(\hat{\mathbf{x}}_k) + \bar{\mathbf{h}}(\mathbf{u}_k) + \bar{\mathbf{E}}_k \mathbf{y}_{k+1}. \quad (19)$$

As a consequence, the algorithm used for state estimation of (16)(17) can be given as follows:

$$\hat{\mathbf{x}}_{k+1/k} = \bar{\mathbf{g}}(\hat{\mathbf{x}}_k) + \bar{\mathbf{h}}(\mathbf{u}_k) + \bar{\mathbf{E}}_k \mathbf{y}_{k+1}, \quad (20)$$

$$\mathbf{P}_{k+1/k} = \bar{\mathbf{A}}_k \mathbf{P}_k \bar{\mathbf{A}}_k^T + \mathbf{Q}_k, \quad (21)$$

$$\mathbf{K}_{k+1} = \mathbf{P}_{k+1/k} \mathbf{C}_{k+1}^T (\mathbf{C}_{k+1} \mathbf{P}_{k+1/k} \mathbf{C}_{k+1}^T + \mathbf{R}_{k+1})^{-1}, \quad (22)$$

$$\hat{\mathbf{x}}_{k+1} = \hat{\mathbf{x}}_{k+1/k} + \mathbf{K}_{k+1}(\mathbf{y}_{k+1} - \mathbf{C}_{k+1}\hat{\mathbf{x}}_{k+1/k}), \quad (23)$$

$$\mathbf{P}_{k+1} = [\mathbf{I} - \mathbf{K}_{k+1} \mathbf{C}_{k+1}] \mathbf{P}_{k+1/k}, \quad (24)$$

where

$$\bar{\mathbf{A}}_k = \left. \frac{\partial \bar{\mathbf{g}}(\mathbf{x}_k)}{\partial \mathbf{x}_k} \right|_{\mathbf{x}_k = \hat{\mathbf{x}}_k} = \bar{\mathbf{G}}_k \left. \frac{\partial \mathbf{g}(\mathbf{x}_k)}{\partial \mathbf{x}_k} \right|_{\mathbf{x}_k = \hat{\mathbf{x}}_k} = \bar{\mathbf{G}}_k \mathbf{A}_k. \quad (25)$$

Employing the Lyapunov approach to convergence analysis of the EUIO it can be proved that the domain of attraction significantly depends on the covariance matrices \mathbf{Q}_{k-1} and \mathbf{R}_{k-1} of the process \mathbf{w}_k and measurement \mathbf{v}_k noise, respectively. Unfortunately, an analytical derivation of the \mathbf{Q}_{k-1} and \mathbf{R}_{k-1} matrices seems to be an extremely difficult problem. However, it is possible to set the above matrices as $\mathbf{Q}_{k-1} = \beta_1 \mathbf{I}$, $\mathbf{R}_{k-1} = \beta_2 \mathbf{I}$, with β_1 and β_2 large enough. On the other hand, it is well-known that the convergence rate of such an Extended Kalman Filter (EKF)-like approach can be increased by an appropriate selection of the covariance matrices \mathbf{Q}_{k-1} and \mathbf{R}_{k-1} , i.e., the more accurate (near "true" values) the covariance matrices, the better the convergence rate. This means that in the deterministic case ($\mathbf{w}_k = 0$ and $\mathbf{v}_k = 0$) both matrices

should be zero. Unfortunately, such an approach usually leads to the divergence of the observer as well as to other computational problems. To tackle these difficulties, a compromise between the convergence and the convergence rate should be established.

Using (23), the state estimation error (for fault-free mode) can be given as

$$e_{k+1} = \mathbf{x}_{k+1} - \hat{\mathbf{x}}_{k+1} = [\mathbf{I} - \mathbf{K}_{k+1}\mathbf{C}_{k+1}] e_{k+1/k}, \quad (26)$$

and

$$e_{k+1/k} = \mathbf{x}_{k+1} - \hat{\mathbf{x}}_{k+1/k} = \bar{\mathbf{g}}(\mathbf{x}_k) - \bar{\mathbf{g}}(\hat{\mathbf{x}}_k) = \alpha_k \bar{\mathbf{A}}_k e_k, \quad (27)$$

where $\alpha_k = \text{diag}(\alpha_{1,k}, \dots, \alpha_{n,k})$ is an unknown diagonal matrix. Thus, using (27), the equation (26) becomes

$$e_{k+1} = [\mathbf{I} - \mathbf{K}_{k+1}\mathbf{C}_{k+1}] \alpha_k \bar{\mathbf{A}}_k e_k. \quad (28)$$

It is clear from (27) that α_k represents the linearisation error. First, let us define

$$\bar{\alpha}_k = \max_{j=1, \dots, n} |\alpha_{j,k}|, \quad \underline{\alpha}_k = \min_{j=1, \dots, n} |\alpha_{j,k}|. \quad (29)$$

Theorem 2. (Witczak *et al.*, 2006) *If*

$$\bar{\alpha}_k \leq \left(\frac{\underline{\alpha} (\bar{\mathbf{A}}_k)^2 \underline{\alpha} (\mathbf{C}_{k+1})^2 \underline{\alpha} (\bar{\mathbf{A}}_k \mathbf{P}_k \bar{\mathbf{A}}_k^T + \mathbf{Q}_k)}{\bar{\sigma} (\mathbf{C}_{k+1} \mathbf{P}_{k+1/k} \mathbf{C}_{k+1}^T + \mathbf{R}_{k+1})} + \frac{(1-\zeta) \underline{\alpha} (\bar{\mathbf{A}}_k \mathbf{P}_k \bar{\mathbf{A}}_k^T + \mathbf{Q}_k)}{\bar{\sigma} (\bar{\mathbf{A}}_k)^2 \bar{\sigma} (\mathbf{P}_k)} \right)^{\frac{1}{2}}, \quad (30)$$

where $0 < \zeta < 1$, then the proposed extended unknown input observer is locally asymptotically convergent.

It is clear from (30) that the bound of $\bar{\alpha}_k$ can be maximised by suitable settings of the instrumental matrices \mathbf{Q}_k and \mathbf{R}_k . This can be realised as follows (Witczak *et al.*, 2006):

$$\mathbf{Q}_k = (\gamma \varepsilon_k^T \varepsilon_k + \delta_1) \mathbf{I}, \quad \varepsilon_k = \mathbf{y}_k - \mathbf{C}_k \hat{\mathbf{x}}_k, \quad (31)$$

$$\mathbf{R}_{k+1} = \delta_2 \mathbf{I}, \quad (32)$$

with $\gamma > 0$ and $\delta_1 > 0, \delta_2 > 0$ are large and small enough, respectively. On the other hand the instrumental matrices can be set as follows:

$$\mathbf{Q}_{k-1} = q^2(\varepsilon_{k-1}) \mathbf{I}, \quad (33)$$

$$\mathbf{R}_k = r^2(\varepsilon_k) \mathbf{I}, \quad (34)$$

where $q(\varepsilon_{k-1})$ and $r(\varepsilon_k)$ are nonlinear functions of the output error ε_k (the squares are used to ensure the positive definiteness of \mathbf{Q}_{k-1} and \mathbf{R}_k). Thus, the problem reduces to identifying the above functions. To tackle this problem, Genetic Programming (GP) (Obuchowicz, 2003; Koza, 1992) can be employed. The unknown functions $q(\varepsilon_{k-1})$ and $r(\varepsilon_k)$ can be expressed as a tree. Thus, in the case of $f q(\cdot)$ and $r(\cdot)$, the terminal sets are $\mathbf{T} = \{\varepsilon_{k-1}\}$ and $\mathbf{T} = \{\varepsilon_k\}$, respectively. In both cases, the function set can be defined

as $\mathbf{F} = \{+, *, /, \xi_1(\cdot), \dots, \xi_t(\cdot)\}$, where $\xi_t(\cdot)$ is a nonlinear univariate function and, consequently the number of populations is $n_p = 2$. Since the terminal and function sets are given, the GP approach can be easily adapted for the identification purpose of $q(\cdot)$ and $r(\cdot)$. First, let us define the performance index including a necessary ingredient of the \mathbf{Q}_{k-1} and \mathbf{R}_k selection process.

Since the instrumental matrices should be chosen in order to maximize the convergence rate, we have

$$(\mathbf{Q}_{k-1}, \mathbf{R}_k) = \arg \max_{q(\varepsilon_{k-1}), r(\varepsilon_k)} J_{obs,1}(q(\varepsilon_{k-1}), r(\varepsilon_k)). \quad (35)$$

where $J_{obs,1}(q(\varepsilon_{k-1}), r(\varepsilon_k)) = \sum_{k=0}^{n_t-1} \text{trace } \mathbf{P}_k$.

On the other hand, owing to FDI requirements, it is clear that the output error should be closed to zero in the fault free mode. In this case, one can define another performance index:

$$(\mathbf{Q}_{k-1}, \mathbf{R}_k) = \arg \min_{q(\varepsilon_{k-1}), r(\varepsilon_k)} J_{obs,2}(q(\varepsilon_{k-1}), r(\varepsilon_k)), \quad (36)$$

where

$$J_{obs,2}(q(\varepsilon_{k-1}), r(\varepsilon_k)) = \sum_{k=0}^{n_t-1} \varepsilon_k^T \varepsilon_k. \quad (37)$$

Therefore, in order to couple (35) and (36), the following identification criterion is employed:

$$(\mathbf{Q}_{k-1}, \mathbf{R}_k) = \arg \min_{q(\varepsilon_{k-1}), r(\varepsilon_k)} \frac{J_{obs,1}(q(\varepsilon_{k-1}), r(\varepsilon_k))}{J_{obs,2}(q(\varepsilon_{k-1}), r(\varepsilon_k))} \quad (38)$$

Since the identification criterion is established, it is straightforward to use the GP algorithm.

The numerical example considered here is a fifth-order two-phase nonlinear model of an induction motor (Boutayeb and Aubry, 1999). Moreover, the following three cases concerning the selection of \mathbf{Q}_{k-1} and \mathbf{R}_k were considered:

Case 1: Classical approach (constant values), i.e., $\mathbf{Q}_{k-1} = 0.1, \mathbf{R}_k = 0.1$.

Case 2: Selection supported by an analytical consideration:

$$\mathbf{Q}_{k-1} = 10^3 \varepsilon_{k-1}^T \varepsilon_{k-1} \mathbf{I} + 0.01 \mathbf{I}, \quad \mathbf{R}_k = 0.01 \mathbf{I}.$$

Case 3: GP-based approach. In order to obtain the matrices \mathbf{Q}_{k-1} and \mathbf{R}_k using the GP-based approach (Case 3) (Witczak and Korbicz, 2004), a set of $n_t = 300$ input-output measurements was generated. The simulation results (for all the cases) are shown in Fig. 1 (Witczak *et al.*, 2002). It can be seen, that the proposed approach is superior to the classical technique of selecting the instrumental matrices \mathbf{Q}_{k-1} and \mathbf{R}_k .

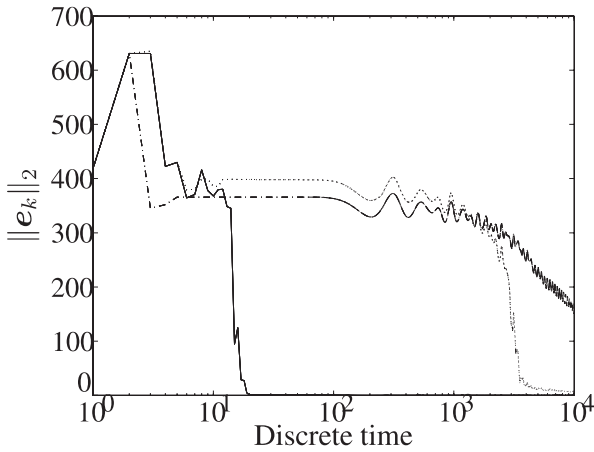


Fig. 1. State estimation error norm $\|e_k\|_2$ for Case 1 (dash-dotted line), Case 2 (dotted line) and Case 3 (solid line)

3. Genetic programming in modelling

Although there are many techniques for constructing non-analytical models, in one way or another, they finally boil down to several global optimization problems, like searching for an optimal model structure, the allocation of model parameters etc. They are nonlinear, multi-modal, usually multi-objective, so that conventional “local” optimization methods are insufficient to solve them. In recent years, direct search techniques, which are problem-independent, have been widely used in optimization. Unlike calculus-based methods (gradient descent, etc.), direct search algorithms do not require the use of derivatives. Gradient-descent methods work well when the objective surface is relatively smooth, with few local minima. However, real-world data are often multimodal and contaminated by noise which can further distort the objective surface.

Evolutionary Algorithms (EAs) are a broad class of stochastic optimization algorithms inspired by some biological processes which allow populations of organisms to adapt to their surrounding environment (Goldberg, 1989; Michalewicz, 1996; Obuchowicz, 2003). Genetic Programming (GP) (Koza, 1992; Obuchowicz, 2003) is an extension of EAs. The main difference between these two approaches is that in GP the evolving individuals are parse trees rather than fixed-length binary strings (Kowalczyk and Białaszewski, 2006).

3.1. Input-output representation of the system via GP

The set of possible candidate models from which the system model will be obtained constitutes an important preliminary task in any system identification procedure (Nelles, 2001; Walter and Pronzato, 1997; Janczak, 2005). Knowing that the diagnosed system exhibits nonlinear characteristics, a choice of the nonlinear model set must be made. In this section, an NARX (Nonlinear AutoRegressive with exogenous variable) model was selected as the foundation for identification methodology. Let a Multi-Input and Multi-Output (MIMO) NARX model has the following form:

$$\hat{y}_{i,k} = g_i(\hat{y}_{1,k-1}, \dots, \hat{y}_{1,k-n_{1,y}}, \dots, \hat{y}_{m,k-1}, \dots, \hat{y}_{m,k-n_{m,y}},$$

$$u_{1,k-1}, \dots, u_{1,k-n_{1,u}}, \dots, u_{r,k-1}, \dots, u_{r,k-n_{r,u}}, \theta_i), \quad (39)$$

$$i = 1, \dots, m.$$

Thus the system output is given by

$$y_k = \hat{y}_k + \varepsilon_k, \quad (40)$$

where ε_k consists of a structural deterministic error, caused by the model-reality mismatch, and the measurement noise \mathbf{v}_k . The problem is to determine an unknown function $g(\cdot) = (g_1, \dots, g_m)$ and to estimate the corresponding parameters vector $\theta = (\theta_1, \dots, \theta_m)$.

One possible solution to this problem is the genetic programming approach. A tree is the main ingredient underlying the GP algorithm. In order to adapt GP to system identification it is necessary to represent the model (39) as a tree, or a set of trees. Indeed, as is shown in Fig. 2, the Multi-Input and Single-Output (MISO) NARX model can be easily put in the form of a tree, and hence to build the MIMO model (39) it is necessary to use m trees. In such a tree, two sets can be distinguished, namely, the terminal T and function F sets. The language of the trees in GP is formed by the user-defined function F set and the terminal T set, which form the nodes of the trees.

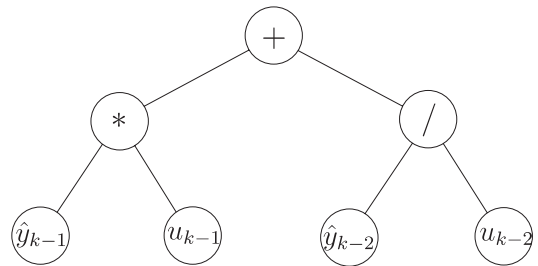
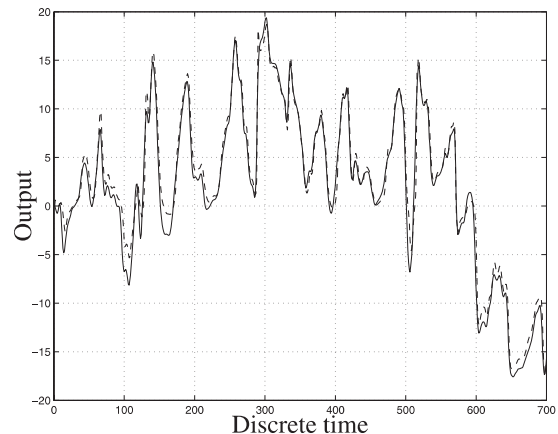


Fig. 2. Exemplary GP tree representing the model $\hat{y}_k = \hat{y}_{k-1}u_{k-1} + \hat{y}_{k-2}/u_{k-2}$

The functions should be chosen so that they are *a priori* useful in solving the problem, i.e., any knowledge concerning the system under consideration should be included in the function set. This function set is very important and should be universal enough to be capable of representing a wide range of nonlinear systems. The terminals are usually variables or constants. In (Esparcia-Alcazar, 1998), a tree representation is extended by the so-called *node gains*. A node gain is a numerical parameter associated with the node, which multiplies its output value.



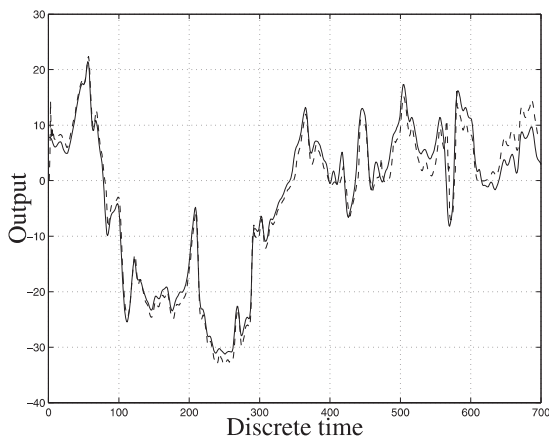


Fig. 3. System (solid line) and model (dashed line) output for the identification (left) and validation (right) data sets

One of the best known criteria which can be employed to select the model structure and to estimate its parameters is the Akaike Information Criterion (AIC) (Walter and Pronzato, 1997), where the following quality index is minimized:

$$J_{AIC}(M_i) = \frac{1}{2}J(M_i(\hat{\theta}^i)) + \frac{1}{n_T} \dim \theta^i, \quad (41)$$

where

$$J(M_i(\theta^i)) = \ln \det \sum_{k=1}^{n_T} \varepsilon_k \varepsilon_k^T, \quad (42)$$

and $\hat{\theta}^i = \arg \min_{\theta^i} J(M_i(\theta^i))$ are the obtained using the identification data set of n_T pairs of input/output measurements. The GP algorithm was successfully applied to identify the input-output model of the evaporation station at the Lublin Sugar Factor S.A. (Poland) (DAMADICS, 2002). Figure 3 illustrates the obtained results (Witczak *et al.*, 2002).

3.2. Choice of the gain matrix for the robust nonlinear observer

The solution of the diagnosed system modeling presented in the previous subsection possesses a disadvantage. Usually, the parameters of the obtained GP model have no physical interpretations. Here a nonlinear state observer designing methodology based on the classical approach and a GP technique and proposed by Witczak and co-workers (2002) is presented. Consider the nonlinear discrete system

$$\begin{aligned} \mathbf{x}_{k+1} &= \mathbf{f}(\mathbf{x}_k, \mathbf{u}_k, \mathbf{w}_k), \\ \mathbf{y}_k &= \mathbf{h}(\mathbf{x}_k, \mathbf{v}_k), \end{aligned} \quad (43)$$

where \mathbf{u}_k is the input, \mathbf{x}_k is the output, \mathbf{y}_k is the state, \mathbf{v}_k and \mathbf{w}_k represent the process and measurement noise, and $\mathbf{h}(\cdot)$, $\mathbf{f}(\cdot)$ are nonlinear functions.

The problem is to estimate the state \mathbf{x}_k of the system (43), where a set of measured inputs and outputs and the model of the system are given. The classical methods using

different kinds of approximation are often applied (Korbicz and Bidyuk, 1993), and can be given as follows:

$$\hat{\mathbf{x}}_k = \hat{\mathbf{x}}_k^- + \mathbf{K}_k \varepsilon_k^-, \quad (44)$$

$$\mathbf{y}_k = \mathbf{h}(\mathbf{x}_k, \mathbf{v}_k), \quad (45)$$

where ε_k^- denotes the *a priori* output error, $\hat{\mathbf{x}}_k^-$ is an *a priori* state estimate, $\hat{\mathbf{x}}_k$ is a state estimate and \mathbf{K}_k is the gain matrix.

The gain matrix \mathbf{K}_k of the observer (44)-(45) can be searched for by various methods (e.g., the Kalman filter, the Luenberger observer, etc.) which, in a large majority, consist of constant elements and are not robust to model uncertainties. In (Witczak *et al.*, 1999), the gain matrix is composed of certain functions, i.e., each entry of the gain matrix is a function which depends on the *a priori* output error and the system input. Therefore, it can be written as follows:

$$\hat{\mathbf{x}}_k = \hat{\mathbf{x}}_k^- + \mathbf{K}(\varepsilon_k^-, \mathbf{u}_k) \varepsilon_k^-. \quad (46)$$

Thus the main goal is to obtain an appropriate form of $\mathbf{K}(\varepsilon_k^-, \mathbf{u}_k)$ based on a set of measured outputs and inputs and the mathematical model of the system. Even if the mathematical model is uncertain and/or the initial state is far from its expected value, it seems possible to obtain $\mathbf{K}(\varepsilon_k^-, \mathbf{u}_k)$ to ensure the best fitness to the real system. For that purpose, a GP technique is exploited, where the gain matrix is obtained off-line from a randomly created population by means of an evolutionary process.

3.3. GP approach to the state-space representation of the system

Let us consider the following class of nonlinear discrete-time systems:

$$\mathbf{x}_{k+1} = \mathbf{g}(\mathbf{x}_k, \mathbf{u}_k) + \mathbf{w}_k, \quad (47)$$

$$\mathbf{y}_{k+1} = \mathbf{C} \mathbf{x}_{k+1} + \mathbf{v}_k. \quad (48)$$

Assume that the function $\mathbf{g}(\cdot)$ has the form

$$\mathbf{g}(\mathbf{x}_k, \mathbf{u}_k) = \mathbf{A}(\mathbf{x}_k) \mathbf{x}_k + \mathbf{h}(\mathbf{u}_k). \quad (49)$$

Thus, the state-space model of the system (47) can be expressed as

$$\hat{\mathbf{x}}_{k+1} = \mathbf{A}(\hat{\mathbf{x}}_k) \hat{\mathbf{x}}_k + \mathbf{h}(\mathbf{u}_k), \quad (50)$$

$$\mathbf{y}_{k+1} = \mathbf{C} \mathbf{x}_{k+1} + \mathbf{v}_k. \quad (51)$$

Without loss of generality, it is possible to assume that

$$\mathbf{A}(\hat{\mathbf{x}}_k) = \text{diag}(a_{i,i}(\hat{\mathbf{x}}_k) \mid i = 1, 2, \dots, n). \quad (52)$$

The problem reduces to identifying the nonlinear functions $a_{i,i}(\hat{\mathbf{x}}_k)$, $h_i(\mathbf{u}_k)$ ($i = 1, \dots, n$), and the matrix \mathbf{C} . Assuming $\max_{i=1, \dots, n} |a_{i,i}(\hat{\mathbf{x}}_k)| < 1$, it can be shown (Witczak *et al.*, 2002) that the model (50) is globally asymptotically

stable. This implies that $a_{i,i}(\hat{x}_k)$ should have the following structure:

$$a_{i,i}(\hat{x}_k) = \tanh(s_{i,i}(\hat{x}_k)), \quad i = 1, \dots, n, \quad (53)$$

where $\tanh(\cdot)$ is a hyperbolic tangent function, and $s_{i,i}(\hat{x}_k)$ is a function to be determined.

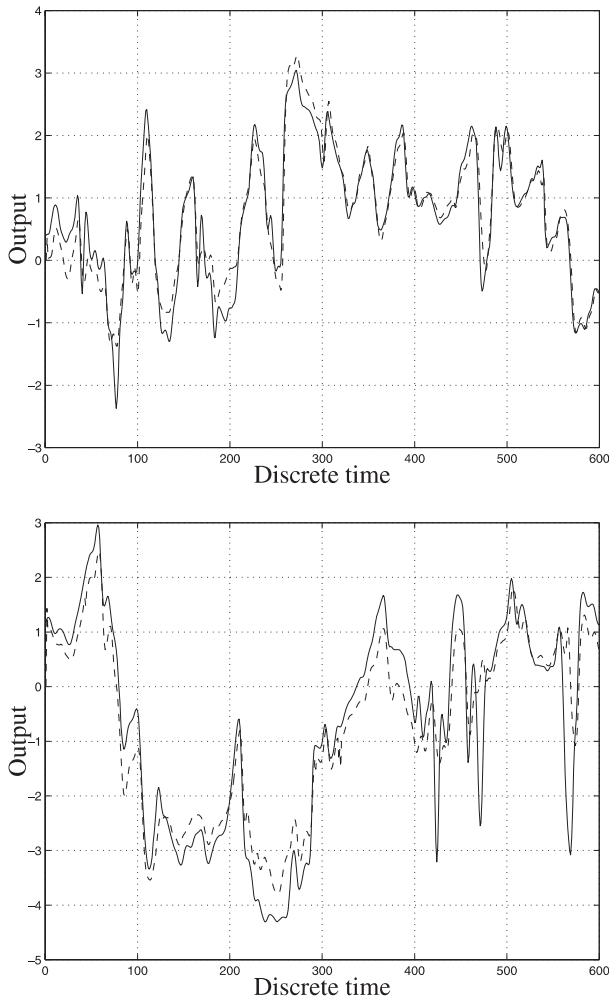


Fig. 4. System (solid line) and model (dashed line) output for the identification (left) and validation (right) data sets

In order to identify $s_{i,i}(\hat{x}_k), h_i(u_k)$ ($i = 1, \dots, n$) and the matrix C , the GP algorithm is applied. The fitness function is defined by (41)-(42).

The GP algorithm was successfully applied to build a model of the apparatus at the Lublin Sugar Factor S.A. (Poland) (DAMADICS, 2002). Figure 4 illustrates obtained results (Witczak *et al.*, 2002).

4. Neural networks in fault detection

For the model-based approach (Korbicz *et al.*, 2004; 2002; Patton *et al.*, 2000), the neural network replaces the analytical model that describes the process under the normal operating conditions (Frank and Köppen-Seliger, 1997; Korbicz, 2006; Patan *et al.*, 2005; Patton *et al.*, 2005). First, the network has to be trained to settle this task. Learning data can be collected directly from the process, if possible, or from a simulation model that should be as realistic as possible. The latter possibility is of special

interest for data acquisition in different faulty situations. This is especially important for the task of testing the residual generator because such data are not generally available from the real process. The training process can be carried out off-line or on-line (it depends on the availability of data) (Gupta *et al.*, 2003; Osowski, 2006; Tadeusiewicz, 1993).

The possibility to train a network on-line is very attractive, especially in the case of adapting a neural model to mutable environment or time-varying systems. After finishing the training, a neural network is ready for on-line residual generation. In order to be able to capture the dynamic behaviour of the system, the neural network should have dynamic properties (Gupta *et al.*, 2003; Norgard *et al.*, 2000; Patan, 2007), e.g., it should be a recurrent network.

Residual evaluation is a decision-making process that transforms quantitative knowledge into qualitative Yes or No statements. It can also be perceived as a classification problem. The task is to match each pattern of the symptom vector with one of the pre-assigned classes of faults and the fault-free case. This process may be highly facilitated with intelligent decision making. To perform residual evaluation, neural networks can be applied, e.g., feed-forward networks or self-organizing maps (Haykin, 1999; Korbicz *et al.*, 1994).

4.1. Robust model-based approach

As was mentioned in Section 2, when non-linear state space models are available, fault diagnosis can be realised by using the concept of an unknown input observer. Unfortunately, when the direction of faults is similar to that of an unknown input, then the unknown input decoupling procedure may considerably impair fault sensitivity. If the above-mentioned approach fails, then describing model uncertainty in a different way seems to be a good remedy. One of the possible approaches is to use statistical techniques (Atkinson and Donev, 1992; Walter and Pronzato, 1997) (for an example regarding different approaches, the reader is referred to (Delebecque *et al.*, 2003)) to obtain parameter uncertainty of the model and, consequently, model output uncertainty. Such parameter uncertainty is defined as the parameter confidence region containing a set of admissible parameters that are consistent with the measured data. Thus it is evident that parameter uncertainty depends on measurement uncertainty, i.e., noise, disturbances, etc.

The knowledge about parameter uncertainty makes it possible to design the so-called adaptive threshold (Frank *et al.*, 1999). The adaptive threshold, contrary to the fixed one, bounds the residual at a level that is dependent on model uncertainty, and hence it provides a more reliable fault detection.

Contrary to the typical industrial applications of neural networks that are presented in the literature (Karpenko *et al.*, 2003; Korbicz *et al.*, 2004; Mrugalski and Korbicz, 2006). Witczak *et al.* (2006a) defined the task of designing a neural network in such a way as to obtain a model with a possibly small uncertainty. Indeed, the approaches presented in the literature try to obtain a model that is best suited to a particular data set. This may result in a model

with a relatively large uncertainty. A degraded performance of fault diagnosis constitutes a direct consequence of using such models.

To tackle this challenging problem for non-linear dynamic systems, the GMDH (Group Method of Data Handling) approach (Ivakhnenko and Mueller, 1995; Korbicz and Mrugalski, 2007) can be effectively adapted (Witczak *et al.*, 2006a). A complete design procedure concerning the application of GMDH neural networks to robust fault detection is proposed. Starting from a set of input-output measurements of the system, it is shown how to estimate the parameters and the corresponding uncertainty of a neuron using the so-called bounded-error approach (Milanese *et al.*, 1996; Walter and Pronzato, 1997). As a result, they obtained a tool that is able to generate an adaptive threshold. The methodology developed for parameter and uncertainty estimation of a neuron makes it possible to formulate an algorithm that allows obtaining a neural network with a relatively small modelling uncertainty. All the hard computations regarding the design of the GMDH neural network are performed off-line, and hence the problem regarding the time-consuming calculations is not of paramount importance.

As has been mentioned, the reliability of such fault diagnosis schemes is strongly dependent on model uncertainty, i.e., the mismatch between a neural network and the system being considered. Thus, it is natural to minimise model uncertainty as far as possible. This can be realised with the application of Optimum Experimental Design (OED) theory (Atkinson and Donev, 1992; Uciński, 2005; Walter and Pronzato, 1997). Recently, Witczak (2006b) developed a D-optimum experimental design strategy that can be used for training single-output neural networks.

Robust GMDH neural networks. A successful application of the ANNs to the system identification and fault diagnosis tasks (Korbicz *et al.*, 2001) 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 selection of the number of layers and the number of neurons in a particular layer. If the obtained network does not satisfy prespecified requirements, then a new network structure is selected and parameter estimation is repeated once again. The 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 model uncertainty. Thus, it seems desirable to have a tool which can be employed for automatic selection of the ANN structure, based only on the measured data. To overcome this problem, GMDH neural networks (Ivakhnenko and Mueller, 1995; Mrugalski, 2004; Mrugalski and Korbicz, 2005) have been proposed. The synthesis process of the GMDH model 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 the best quality approximation of the identified system. Thus, the task of designing a neural network is defined in such a way so as to obtain a model with a small uncertainty.

The idea of the GMDH approach relies on replacing the complex neural model by the set of hierarchically connected neurons. The behaviour of each neuron should reflect the behaviour of the system being considered. It follows from the rule of the GMDH algorithm that the parameters of each neuron are estimated in such a way that their output signals are the best approximation of the real system output. In this situation, the neuron should have the ability to represent the dynamics. One way out of this problem is to use dynamic neurons (Korbicz and Kuś, 1999; Patan and Parisini, 2005). Dynamics in these neurons are realised by introducing a linear dynamic system an IIR filter. The process of GMDH network synthesis leads to the evolution of the resulting model structure in such a way as to obtain the best quality approximation of the real system (Mrugalski, 2004; Witczak *et al.*, 2006a).

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

Confidence estimation of GMDH neural networks.

Even though the application of the GMDH approach to model structure selection can improve the quality of the model, the resulting structure is not the same as that of the system. It can be shown (Mrugalski, 2004) that the application of the classical evaluation criteria such as the Akaike Information Criterion (AIC) and the Final Prediction Error (FPE) (Ivakhnenko and Mueller, 1995; Mueller and Lemke, 2000) can lead to the selection of inappropriate neurons and, consequently, to unnecessary structural errors.

Apart from the model structure selection stage, inaccuracy in parameter estimates also contributes to modelling uncertainty. Indeed, while applying the least-square method to parameter estimation of neurons, a set of restrictive assumptions has to be satisfied (Witczak *et al.*, 2006a). An effective remedy to such a challenging problem is to use the so-called Bounded Error Approach (BEA) (Milanese *et al.*, 1996; Witczak *et al.*, 2006a). Let us consider the following system:

$$y_k = \mathbf{r}_k^T \boldsymbol{\theta} + \varepsilon_k. \quad (54)$$

where \mathbf{r}_k stands the regressor vector, $\boldsymbol{\theta} \in \mathbb{R}^{n_\theta}$ denotes the parameter vector, and ε_k represents the difference between the original system and the model.

The problem is to obtain the parameter estimate vector $\hat{\boldsymbol{\theta}}$, as well as the associated parameter uncertainty required to design robust fault detection system. The knowledge regarding the set of admissible parameter values allows obtaining the confidence region of the model output which satisfies

$$\tilde{y}_k^m \leq y_k \leq \tilde{y}_k^M, \quad (55)$$

where \tilde{y}_k^m and \tilde{y}_k^M are the minimum and maximum admissible values of the model output that are consistent with the input-output measurements of the system.

It is assumed that ε_k consists of a structural deterministic error caused by the model-reality mismatch, and the stochastic error caused by the measurement noise is bounded as follows:

$$\varepsilon_k^m \leq \varepsilon_k \leq \varepsilon_k^M, \quad (56)$$

where the bounds ε_k and ε_k^M ($\varepsilon_k^m \neq \varepsilon_k^M$) can be estimated (Witczak *et al.*, 2006a).

The idea underlying the bounded-error approach is to obtain a feasible parameter set \mathbb{P} (Milanese *et al.*, 1996) that is consistent with the input-output measurements used for parameter estimation. The resulting \mathbb{P} is described by a polytope defined by a set of vertices \mathbb{V} . Thus, the problem of determining the model output uncertainty can be solved as follows:

$$\mathbf{r}_k^T \boldsymbol{\theta}_k^m \leq \mathbf{r}_k^T \boldsymbol{\theta} \leq \mathbf{r}_k^T \boldsymbol{\theta}_k^M, \quad (57)$$

where

$$\boldsymbol{\theta}_k^m = \arg \min_{\boldsymbol{\theta} \in \mathbb{V}} \mathbf{r}_k^T \boldsymbol{\theta}, \quad \boldsymbol{\theta}_k^M = \arg \max_{\boldsymbol{\theta} \in \mathbb{V}} \mathbf{r}_k^T \boldsymbol{\theta}. \quad (58)$$

As has been mentioned, the neurons in the l -th ($l > 1$) layer are fed with the outputs of the neurons from the $(l - 1)$ -th layer. In order to modify the above-presented approach for the uncertain regressor case, let us denote an unknown "true" value of the regressor $\mathbf{r}_{n,k}$ by a difference between the measured value of the regressor \mathbf{r}_k and the error in the regressor e_k :

$$\mathbf{r}_{n,k} = \mathbf{r}_k - e_k, \quad (59)$$

where it is assumed that the error e_k is bounded as

$$e_{i,k}^m \leq e_{i,k} \leq e_{i,k}^M, \quad i = 1, \dots, n_p. \quad (60)$$

Using (54) and substituting (59) into (60), one can define the space containing the parameter estimates:

$$\varepsilon_k^m - e_k^T \boldsymbol{\theta} \leq y_k - \mathbf{r}_k^T \boldsymbol{\theta} \leq \varepsilon_k^M - e_k^T \boldsymbol{\theta}, \quad (61)$$

which makes it possible to adapt the above-described technique to the error-in-regressor case (Witczak *et al.*, 2006a).

The proposed modification of the BEA makes it possible to estimate the parameter vectors of the neurons from the l -th, $l > 1$ layers. Finally, it can be shown that the model output uncertainty has the following form:

$$\tilde{y}_k^m \leq \mathbf{r}_n^T \boldsymbol{\theta} \leq \tilde{y}_k^M. \quad (62)$$

In order to adapt the presented approach to parameter estimation of non-linear neurons with an activation function $\xi(\cdot)$, it is necessary to transform the relation

$$\varepsilon_k^m \leq y_k - \xi \mathbf{r}_k^T \boldsymbol{\theta} \leq \varepsilon_k^M, \quad (63)$$

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

$$\xi^{-1}(y_k - \varepsilon_k^M) \leq \mathbf{r}_k^T \boldsymbol{\theta} \leq \xi^{-1}(y_k - \varepsilon_k^m). \quad (64)$$

Knowing the model structure and possessing the knowledge regarding its uncertainty, it is possible to design a robust fault detection scheme with an adaptive threshold (Fig. 5).

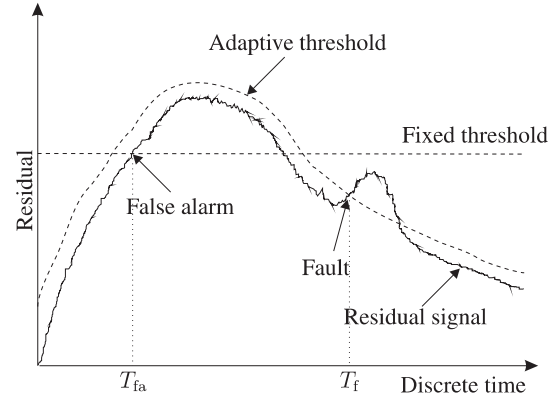


Fig. 5. Illustration of the concept of the adaptive threshold

The model output uncertainty interval, calculated with the application of the GMDH model, should contain the real system response in the fault-free mode. Therefore, the system output should satisfy

$$\tilde{y}_k^m + \varepsilon_k^m \leq y_k \leq \tilde{y}_k^M + \varepsilon_k^M. \quad (65)$$

This means that robust fault detection boils down to checking if the output of the system satisfies (65). Thus, when (65) is violated, then a fault symptom occurs.

5. Neuro-fuzzy networks in fault detection

The procedure of Neuro-Fuzzy (NF) network design consists of the structure selection stage and the parameter estimation stage (Korbicz and Kowal, 2001; Rutkowska, 2002; Rutkowski, 2005; Piegat, 2003). The pessimistic scenario assumes the construction of the neuro-fuzzy network only on the basis of the available measurements. The main problem is to obtain the required accuracy and transparency of the rule base in such a situation. A lot of different methods have already been developed both for structure selection and parameter estimation of the neuro-fuzzy network, but there is a demand for better, more effective algorithms, and active research is still conducted in this area.

Tagagi-Sugeno neuro-fuzzy networks can be viewed as multi-model systems which consist of some rules, and each rule defines a single model as the consequent of the rule (Babuška, 1998; Kowal and Korbicz, 2002a; 2002b; 2003; Uppal *et al.*, 2006). The global neuro-fuzzy system is a set of N_r partial models, where N_r determines the number of fuzzy rules. The output of the global system is calculated as a mixture of partial model outputs. The rule fulfillment is determined by fuzzy sets. In order to ensure the desired accuracy of the neuro-fuzzy system, the membership functions of fuzzy sets must be placed properly in the input space, the number of rules must be appropriate and the parameters of partial models must be chosen to minimize the defined error.

Two main strategies for placing fuzzy sets in the input space can be distinguished: the first one proposes to minimize the output error of the global model (Leith and Leithead, 1999), and the other one is based on partial models that model the local behavior of the system (Abonyi *et al.*, 2002). A typical property of the first approach is to arrange fuzzy sets in the input space in such a way that all partial models are active in the whole domain of input variables. In this case, the accuracy of the global model is guaranteed by the proper mixture of partial model outputs. The alternative approach does not examine the global accuracy of the model but concentrates on partial models, which should tune in to the local behavior of the system. The problem of rule base declaration reduces to the determination of the number of rules required for a precise description of the problem to be solved.

The simplest method used to determine the number of rules is based on generating a uniformly distributed grid of rules in the input space. The usage of such an approach is limited only to simple systems with a small number of inputs. The approach does not work well for more complicated systems because it generates a combinatorial explosion of rules, which make this method useless. Fuzzy clustering algorithms are another technique which is often used for fuzzy rule generation (Babuška, 1998; Chen *et al.*, 1998; Kowal *et al.*, 2002; Mendes *et al.*, 2002). The idea of this approach is to find natural groups of data in order to apply to each group one fuzzy rule. It seems to be natural to use fuzzy clustering algorithms in the case of neuro-fuzzy networks. The task of fuzzy clustering is usually reduced to finding the local minimum of the nonlinear cost function, defined by the following expression:

$$J(\mathbf{X}; \mathbf{U}, \mathbf{V}) = \sum_{i=1}^c \sum_{k=1}^N \mu_{ik}^m D_{ik}^2, \quad (66)$$

where the matrix \mathbf{U} contains the membership degrees of data points from the matrix to the defined clusters \mathbf{X} , $\mathbf{V} = [v_1, v_2, \dots, v_c]$, $v_i \in R_n$ is a matrix which defines the centers of the clusters, D_{ik} is a metric used to determine the distance between the data points and the cluster centers:

$$D_{ik}^2 = \|\mathbf{x}_k - \mathbf{v}_i\|^2 = (\mathbf{x}_k - \mathbf{v}_i)^T \mathbf{A} (\mathbf{x}_k - \mathbf{v}_i), \quad (67)$$

and the parameter m takes values from 1 to ∞ and determines the degree of fuzziness of the clusters. The cost function (66) can be viewed as a total variance of the data \mathbf{x}_k with respect to the cluster centers \mathbf{v}_i . The matrix \mathbf{A} which occurs in the expression (67) is used to tune the shape and orientation of the clusters in the space.

The fuzzy clustering algorithm which uses such a norm to calculate the distance between data points and cluster centers is called *Fuzzy C-Mean* (FCM). However, the number of the found clusters strongly depends on the values of coefficients, which must be defined by the designer at the beginning of the procedure, so the application of the algorithm is difficult. Two clustering algorithms were applied to build the model of the valve which is a part of the industrial installation of the Lublin Sugar Factory (DAMADICS, 2002). The learning procedure of the Takagi-

Sugeno neuro-fuzzy network was divided into two phases. In the first step, clustering methods were used to optimize the network structure and prepare the initial values of the parameters. In the second step, the gradient descent method was used to tune all parameters. Two clustering algorithms were used in the first step: the mountain method and the fuzzy C-mean algorithm (Kowal *et al.*, 2002; Mendes *et al.*, 2002). Sample results are shown in Fig. 6.

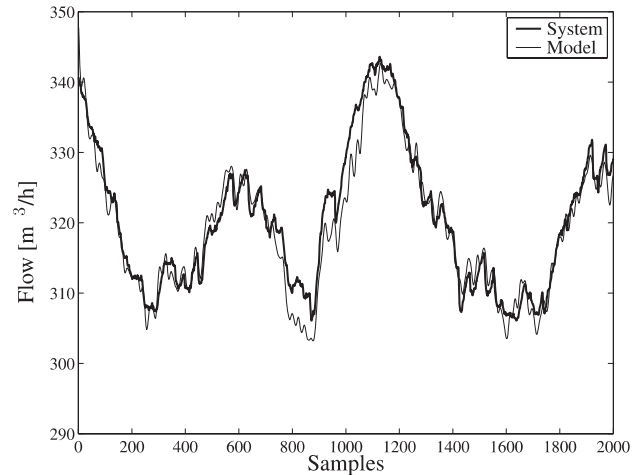


Fig. 6. Performance of the TSK neuro-fuzzy model for the valve V_{51} ($SSE/N \cdot \sigma$ of samples = 6.774)

5.1. Robust neuro-fuzzy networks

The application of neuro-fuzzy networks in diagnostic areas (Calado *et al.*, 2006) creates a demand for suitable design procedures which would take into account the specificity of the fault diagnosis task. An important problem from the diagnostic point of view is residual confidence interval minimization because it makes it possible to detect a fault appropriately early. It has to be stressed that the value of the confidence interval for residuals depends directly on the uncertainty of the model which is used to generate the residuals. If the confidence interval is not consistent with model uncertainty, the fault detection system can trigger off a lot of false alarms. It is obvious in such a situation that model uncertainty has to be considered in fault detection threshold calculations (Chen and Patton, 1999; Mrugalski, 2003; Witczak, 2003). It is also important to minimize model uncertainty in order to obtain a reliable fault detection system that would be able to detect a fault fast and at an early stage, so special procedures for neuro-fuzzy model design must be developed.

To overcome the problem, an alternative approach in the form of the Bounded Error Approach (BEA) method can be applied to tune the parameters of the Takagi-Sugeno neuro-fuzzy network and to calculate the admissible set of parameters and the confidence interval for the network output. The method requires only the information about the range of the disturbances which corrupt measurements. The application of the BEA algorithm for computing the confidence interval of the Takagi-Sugeno fuzzy model output requires to establish some assumptions in order to view the model in the form of an LP system (Kowal, 2005; Kowal and Korbicz, 2005a). The main assumption based on

the fact that the parameters of the membership functions of the fuzzy sets are known. Appropriate selection of the values of these parameters has an essential influence on the uncertainty of the whole fuzzy model. Wrong values of these parameters can significantly increase model uncertainty, thus the model can be unsuitable for diagnostic tasks.

In the proposed approach the clustering algorithm is used to determine the ellipsoid clusters in the input-output space in order to generate for each found cluster one local linear submodel and to determine the parameters of the fuzzy partitions by cluster projection (Babuška, 1998). Another approach is based on the detection of approximately linear dependencies in the data space using a modified BEA (Kowal, 2005; Kowal and Korbicz, 2005a; 2005b). The algorithm consists in the generation of a single rule for each found linear dependency and allows the parameters of fuzzy partitions.

In order to present the BEA approach for estimating the parameters of the determining dynamic Takagi-Sugeno (T-S) network, let us consider the following T-S neuro-fuzzy model:

$$y_k = \sum_{i=1}^n \phi_{i,k} y_{i,k}, \tag{68}$$

where $y_{i,k}$ is the output of the i -th rule and

$$\phi_{i,k} = \frac{\mu_{i,k}}{\sum_{j=1}^n \mu_{j,k}}. \tag{69}$$

The model described by the equation (68) can be viewed in the form of an LP system:

$$y = \mathbf{x}_k^T \boldsymbol{\theta}, \tag{70}$$

$$\mathbf{x}_k = \begin{bmatrix} \phi_{1,k} \mathbf{z}_{1,k} \\ \phi_{2,k} \mathbf{z}_{2,k} \\ \vdots \\ \phi_{n,k} \mathbf{z}_{n,k} \end{bmatrix}, \quad \boldsymbol{\theta} = \begin{bmatrix} \theta_1 \\ \theta_2 \\ \vdots \\ \theta_n \end{bmatrix}.$$

if the parameters of the fuzzy sets are treated like constant values. Here \mathbf{z}_i^k denotes the input vector containing the delayed inputs u_i^k of the local models and the delayed output y_i^k of the local models, i.e.:

$$\mathbf{z}_i^k = [u_i^k, u_i^{k-1}, \dots, u_i^{k-n_a}, y_i^{k-1}, y_i^{k-2}, y_i^{k-n_b}].$$

The output error is given by the following formulae:

$$\varepsilon_k = y'_k - \mathbf{x}_k^T \boldsymbol{\theta}, \tag{71}$$

where e_k is the error and y'_k is the output of the system. The error is bounded by means of the following inequalities:

$$\varepsilon_k^{\min} \leq \varepsilon_k \leq \varepsilon_k^{\max}. \tag{72}$$

thus the admissible set of parameters for N data points is given by the following expression:

$$\mathbf{P} = \{ \boldsymbol{\theta} \in \mathbb{R}^n \mid y'_k - \varepsilon^{\max} \leq \mathbf{x}_k^T \boldsymbol{\theta} \leq y'_k - \varepsilon^{\min}, \quad k = 1, \dots, N \}. \tag{73}$$

Each point inside the set \mathbf{P} defines the vector of model parameters and all sets of parameters determine the group of models consistent with the measurements and bounds. This means that, instead of one model, a set of models with different parameters is given and the output signal is represented in the form of an interval which contains all possible model responses. Real applications usually require a single output value, thus one set of parameters must be chosen. The most common approach chooses the geometrical center of the area \mathbf{P} as the set of parameters that is used to calculate the output of the model. This sample procedure is shown in Fig. 7. If the maximum and minimum values of the parameters are known,

$$\theta_i^{\min} = \arg \min_{\boldsymbol{\theta} \in \mathbf{P}} \theta_i, \tag{74}$$

$$\theta_i^{\max} = \arg \max_{\boldsymbol{\theta} \in \mathbf{P}} \theta_i, \tag{75}$$

the estimates of the parameters can be computed using the following formula:

$$\theta_i = \frac{\theta_i^{\min} + \theta_i^{\max}}{2}, \quad i = 1, \dots, N. \tag{76}$$

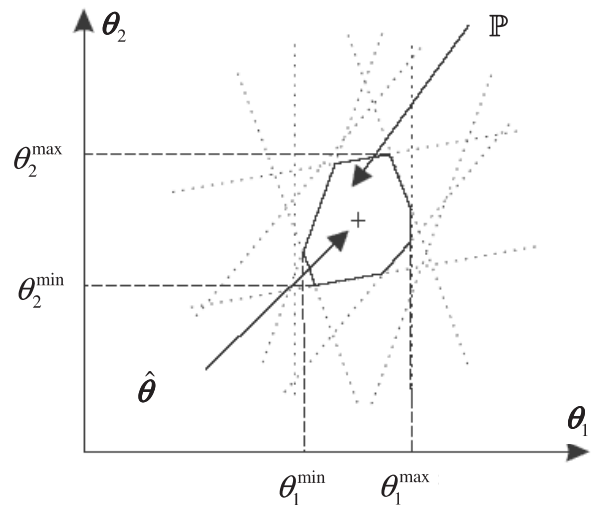


Fig. 7. Sample set of parameters \mathbf{P}

The minimum and maximum values for the following parameters are determined using the linear programming technique (Milanese *et al.*, 1996). The feasible set of parameters is used also to compute the confidence interval for the output of the system:

$$\mathbf{x}_k^T \boldsymbol{\theta}_k^{\min} + \varepsilon^{\min} \leq y'_k \leq \mathbf{x}_k^T \boldsymbol{\theta}_k^{\max} + \varepsilon^{\max}, \tag{77}$$

where

$$\boldsymbol{\theta}_k^{\max} = \arg \max_{\boldsymbol{\theta} \in \mathbf{W}} \mathbf{x}_k^T \boldsymbol{\theta}, \tag{78}$$

$$\boldsymbol{\theta}_k^{\min} = \arg \min_{\boldsymbol{\theta} \in \mathbf{W}} \mathbf{x}_k^T \boldsymbol{\theta}. \tag{79}$$

The confidence interval can be directly applied to calculate the adaptive threshold for the residual signal:

$$r_k = y'_k - y_k. \quad (80)$$

Finally, the adaptive threshold is described by the following inequalities:

$$\mathbf{x}_k^T \boldsymbol{\theta}_k^{\min} + \varepsilon_k^{\min} - y_k \leq e_{r,k} \leq \mathbf{x}_k^T \boldsymbol{\theta}_k^{\max} + \varepsilon_k^{\max} - y_k. \quad (81)$$

Unfortunately, the computations required to determine all vertices W of the convex polyhedron P are so time and memory consuming that it is hard to employ the classical BEA algorithm for complicated models. In this case the methods that approximate the actual set P by the area which has a simplified shape should be employed (Milanese *et al.*, 1996). One of the proposed solution is the Outer Bounding Ellipsoid method which has been applied to fault detection in a DC engine (Kowal, 2005; Kowal and Korbicz, 2006).

6. Illustrative examples

The main objective of this section is to presents two examples which illustrate the effectiveness of the approaches described in the preceding sections. In particular, the first example is devoted to robust fault diagnosis of an induction motor with the extended unknown input observer. The second one concerns the fault detection of an intelligent actuator with the Takagi-Sugeno network.

6.1. Observer-based fault detection of an induction motor

The numerical example considered here is a fifth-order two-phase non-linear model of an induction motor, which has already been the subject of a large number of various control design applications (Boutayeb and Aubry, 1999). The complete discrete-time model in a stator-fixed (a,b) reference frame is

$$x_{1,k+1} = x_{1,k} + h \left(-\gamma x_{1k} + \frac{K}{T_r} x_{3k} + K p x_{5k} x_{4k} + \frac{1}{\sigma L_s} u_{1k} \right), \quad (82)$$

$$x_{2,k+1} = x_{2,k} + h \left(-\gamma x_{2k} - K p x_{5k} x_{3k} + \frac{K}{T_r} x_{4k} + \frac{1}{\sigma L_s} u_{2k} \right), \quad (83)$$

$$x_{3,k+1} = x_{3,k} + h \left(\frac{M}{T_r} x_{1k} - \frac{1}{T_r} x_{3k} - p x_{5k} x_{4k} \right), \quad (84)$$

$$x_{4,k+1} = x_{4,k} + h \left(\frac{M}{T_r} x_{2k} + p x_{5k} x_{3k} - \frac{1}{T_r} x_{4k} \right), \quad (85)$$

$$x_{5,k+1} = x_{5,k} + h \left(\frac{pM}{JL_r} (x_{3k} x_{2k} - x_{4k} x_{1k}) - \frac{T_L}{J} \right), \quad (86)$$

$$y_{1,k+1} = x_{1,k+1}, \quad y_{2,k+1} = x_{2,k+1}, \quad (87)$$

where $\mathbf{x}_k = [x_{1,k}, \dots, x_{n,k}]^T = [\dot{i}_{sak}, \dot{i}_{sbk}, \psi_{rak}, \psi_{rbk}, \omega_k]^T$ represents the currents, the rotor fluxes, and the angular speed, respectively, while $\mathbf{u}_k = [u_{sak}, u_{sbk}]^T$ is the stator voltage control vector, p is the number of the pairs of poles, and T_L is the load torque. The rotor time constant T_r and the remaining parameters are defined as

$$T_r = \frac{L_r}{R_r}, \quad \sigma = 1 - \frac{M^2}{L_s L_r}, \quad K = \frac{M}{\sigma L_s L_r}, \quad \gamma = \frac{R_s}{\sigma L_s} + \frac{R_r M^2}{\sigma L_s L_r^2}, \quad (88)$$

where R_s, R_r and L_s, L_r are stator and rotor per phase resistances and inductances, respectively, and J is the rotor moment inertia.

The numerical values of the above parameters are as follows: $R_s = 0.18 \Omega$, $R_r = 0.15 \Omega$, $M = 0.068 \text{ H}$, $L_s = 0.0699 \text{ H}$, $L_r = 0.0699 \text{ H}$, $J = 0.0586 \text{ kgm}^2$, $T_L = 10 \text{ Nm}$, $p = 1$, a n d $h = 0.1 \text{ ms}$. The input signals are

$$u_{1,k} = 350 \cos(0.03k), \quad u_{2,k} = 300 \sin(0.03k). \quad (89)$$

The initial conditions for the system and the observer are $\mathbf{x}_k = \mathbf{0}$ and $\hat{\mathbf{x}}_k = [200, 200, 50, 50, 300]^T$, and $\mathbf{P}_0 = 10^3 \mathbf{I}$,

$$\mathbf{Q}_{k-1} = 10^{10} \boldsymbol{\varepsilon}_{k-1}^T \boldsymbol{\varepsilon}_{k-1} \mathbf{I} + 0.001 \mathbf{I}, \quad \mathbf{R}_k = 0.01 \mathbf{I}. \quad (90)$$

Let us assume that the unknown input distribution matrix is

$$\mathbf{E} = [1.2, 0.2, 2.4, 1, -1.6]^T, \quad (91)$$

and the corresponding unknown input is simulated by

$$d_k = 3.0 \sin(0.5\pi k) \cos(0.03\pi k). \quad (92)$$

Thus, the system (16)(17) is described using (82)-(87) and 88. The following fault scenarios were considered (Witczak, 2007):

Case 1: Abrupt fault of the $y_{1,k}$ sensor:

$$f_{1,k} = \begin{cases} 0, & 500 < k < 140, \\ -0.1 y_{1,k}, & \text{otherwise,} \end{cases} \quad (93)$$

and $f_{2,k} = 0$.

Case 2: Abrupt fault of the $u_{1,k}$ actuator:

$$f_{1,k} = \begin{cases} 0, & 500 < k < 140, \\ -0.1 y_{1,k}, & \text{otherwise,} \end{cases} \quad (94)$$

and $f_{2,k} = 0$.

Thus, the system is now described by

$$\mathbf{x}_{k+1} = \mathbf{g}(\mathbf{x}_k) + \mathbf{h}(\mathbf{u}_k) + \mathbf{E}_k d_k + \mathbf{L}_{1,k} \mathbf{f}_k, \quad (95)$$

$$\mathbf{y}_{k+1} = \mathbf{C}_{k+1} \mathbf{x}_{k+1} + \mathbf{L}_{2,k+1} \mathbf{f}_{k+1}, \quad (96)$$

with (82)-(87), 88, $\mathbf{f}_k = [f_{1,k}, f_{2,k}]^T$, and

$$\mathbf{L}_{1,k} = \begin{bmatrix} \frac{1}{\sigma L_s} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}^T, \quad \mathbf{L}_{2,k} = \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix}. \quad (96)$$

From Figs. 8 and 9, it can be observed that the residual signal is sensitive to the faults under consideration, which confirms its reliability and abilities of unknown input decoupling.

This, together with unknown input decoupling, implies that the process of fault detection becomes a relatively easy task.

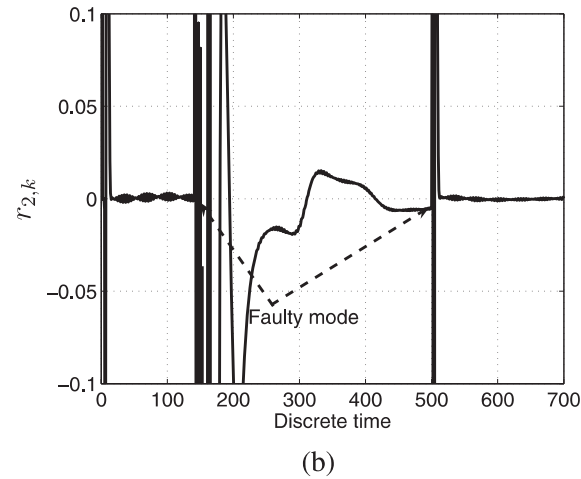
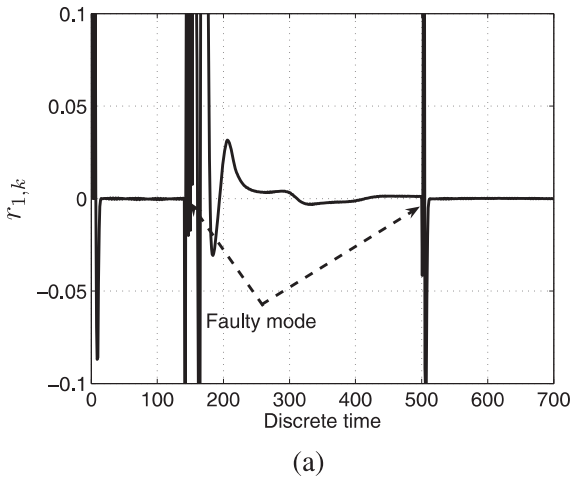


Fig. 8. Residuals for a sensor fault

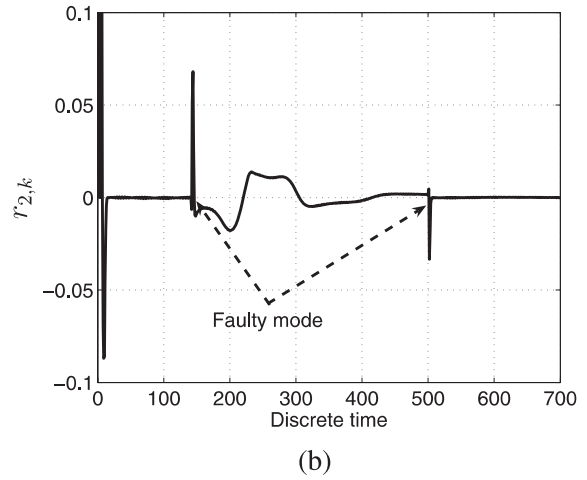
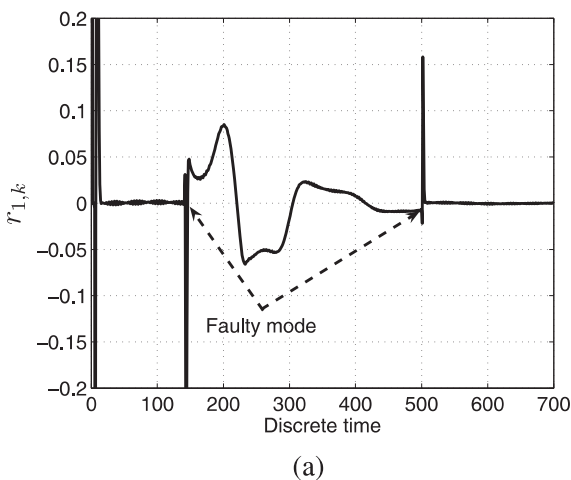


Fig. 9. Residuals for an actuator fault

6.2. Neuro-fuzzy-based fault detection of an intelligent actuator

The scheme of the actuator with an intelligent positioner is given in Fig. 10. Such actuator has been investigated by international research group during the realization of the so-called DAMADICS (2002) project. In the Fig. 10 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 denoted is placement, 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 (CV), 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).

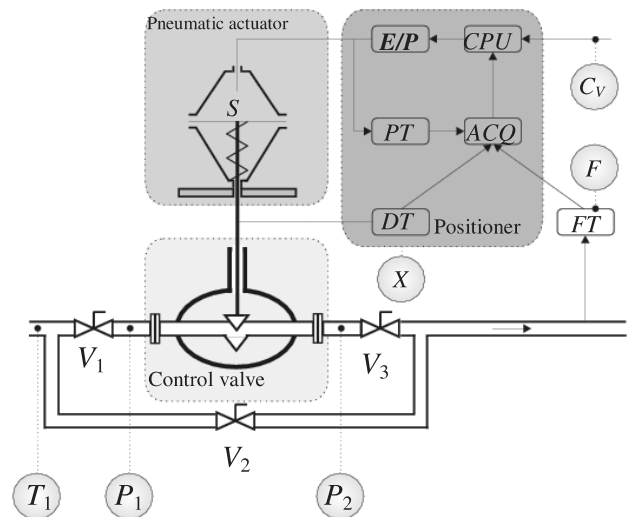


Fig. 10. Scheme of the intelligent actuator

Applying the method for structure generation of NF models (Kowal, 2005; Kowal and Korbicz, 2005b) and the results presented in Sub-section 5.1, two NF models can be defined. The obtained structures are described in Table 1.

Table 1. Neuro-fuzzy models

quantity	f_F	f_X
global inputs	X	C_V
local inputs	X, P_1, P_2, T_1	C_V, P_1, P_2, T_1
no. of rules	7	3

The parameters of fuzzy sets were estimated from the results obtained during structure generation and the parameters of the consequents were estimated using the OBE algorithm.

The first step of the experimental study was to present the modelling abilities of the obtained NF models and, additionally, their system output uncertainty. Figure 11 presents the modelling abilities of the obtained model along with corresponding system output bounds. At the time $T_f = 250$ the big fault (the valve was blocked) occurred.

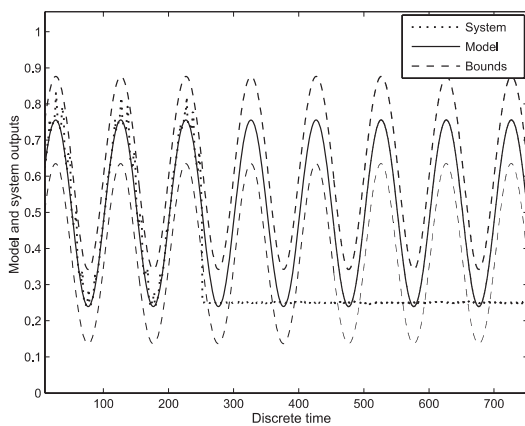


Fig. 11. Model and system output with system output bounds (big fault)

From Fig. 12, which shows the residual and its bounds given by the adaptive threshold, follows that this fault is detected very fast, with a small delay, approximately 5 units.

The developed fault detection scheme with NF models using the available data containing 44 faulty scenarios generated by the actuar simulator (DAMADICS, 2002) was tested as well.

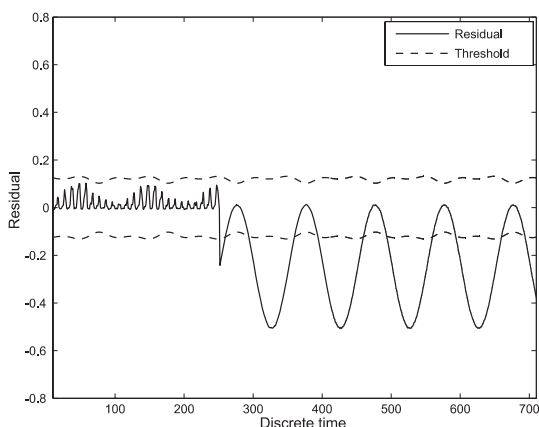


Fig. 12. Residual and adaptive threshold (big fault)

The faults were divided into two main groups: abrupt and incipient faults. Then abrupt faults were divided into three groups: small, medium and big faults.

The fault detection results obtained for all scenarios are shown in Table 2, where the following notations are introduced: Y indicates the fault detected using the designed NF models, N indicates the fault that was not detected by the designed NF models.

From Table 2 it follows that most faults can be detected, however, there are a few faults that cannot. The reason for such a situation was that the system output bounds obtained by the OBE algorithm were too large and hence sensitivity to faults was not high enough. This means that it is necessary to employ a more accurate technique than the OBE algorithm.

7. Conclusion

From the point of view of engineering, it is clear that providing fast and reliable fault detection and isolation is an integral part of control design, particularly as far as the control of complex industrial systems is considered. Unfortunately, most of such systems exhibit non-linear behaviour, which makes it impossible to use the well-developed techniques for linear systems. If it is assumed that the system is linear, which is not true in general, and even if robust techniques for linear systems are used (e.g., unknown input observers), it is clear that such an approximation may lead to unreliable fault detection and, consequently, an early indication of faults which are developing is rather impossible. Such a situation increases the probability of the occurrence of faults, which can be extremely serious in terms of economic losses, environmental impact, or even human mortality. Indeed, robust techniques are able to tolerate a certain degree of model uncertainty. In other words, they are not robust to everything, i.e., are robust to an arbitrary degree of model uncertainty. This real world development pressure creates the need for new techniques which are able to tackle fault diagnosis of non-linear systems. In spite of the fact that the problem has been attacked from various angles by many authors and a number of relevant results have already been reported in the literature, there is no general framework which can be simply and conveniently applied to maintain fault diagnosis for non-linear systems.

Taking into account the above discussion, the main objective of this paper was to consider a robust model-based fault detection system applying analytical and soft computing models. Special attention was paid to the uncertainty of such models and their usefulness in fault diagnosis. In particular, uncertainties of GMDH neural networks and Takagi-Sugeno NF networks were considered. The proposed approach was based on the bounded-error approach, which is superior to the celebrated least-square method in many practical applications. It was shown that the defined confidence interval for the system output of the GMDH 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 non-linear model of an induction motor and the DAMADICS benchmark problem showed the effectiveness of such robust fault detection based on the

extended unknown input observer and the uncertainty of neuro-fuzzy models.

No.	Description	S	M	B	I
	<i>Control valve faults</i>				
f_1	Valve clogging	Y	Y	Y	
f_2	Sedimentation			Y	Y
f_3	Seat erosion				Y
f_4	Bushing frictions				Y
f_5	External leakage				N
f_6	Internal leakage				Y
f_7	Medium evaporation	Y	Y	Y	
	<i>Servo-motor faults</i>				
f_8	Twisted piston rod	N	N	Y	
f_9	Housing				N
f_{10}	Diaphragm perforation	Y	Y	Y	
f_{11}	Spring fault			Y	Y
	<i>Positioner faults</i>				
f_{12}	E/P transducer fault	N	N	N	
f_{13}	Rod displ. sensor fault	Y	Y	Y	Y
f_{14}	Pressure sensor fault	N	N	N	
f_{15}	Feedback fault			Y	
	<i>External faults</i>				
f_{16}	Pressure drop	Y	Y	Y	
f_{17}	Unexpected pressure change			Y	Y
f_{18}	Opened bypass valves	Y	Y	Y	Y
f_{19}	Flow rate sensor fault F	Y	Y	Y	

Table 2. Fault detection results (S-small, M-medium, B-big, I-incipient)

AUTHOR

Józef Korbicz – Deputy Rector for Scientific Research and International Cooperation and professor at the Institute of Control and Computation Engineering, University of Zielona Góra, Poland, e-mail: J.Korbicz@issi.uz.zgora.pl.

References

- [1] Abonyi J., Babuška R. and Szeifert F., "Modified Gath-Geva fuzzy clustering for identification of Takagi-Sugeno fuzzy models", *IEEE Trans. Systems, Man, and Cybernetics*, 2002, Part B, vol. 32, no. 5, pp. 612-621.
- [2] Atkinson A.C. and Donev A.N., *Optimum Experimental Designs*, New York: Oxford University Press, 1992.
- [3] Babuška R., *Fuzzy Modeling for Control*, London: Kluwer Academic Publisher, 1998.
- [4] Blanke M., Kinnaert M., Lunze J. and Staroswiecki M., *Diagnosis and Fault-Tolerant Control*, New York: Springer-Verlag, 2003.
- [5] Boutayeb M. and Aubry D., "A strong tracking extended Kalman observer for nonlinear discrete-time systems", *IEEE Trans. Automatic Control*, 1999, vol. 44, no. 8, pp. 1550-1556.
- [6] Calado J.M.F., Sa da Costa J.M.G., Bartys M. and Korbicz J., "FDI approach to the DAMADICS benchmark problem based on qualitative reasoning coupled with fuzzy neural networks", *Control Engineering Practice*, 2006, vol. 14, no. 6, pp. 685-698.
- [7] Chen J. and Patton R.J., *Robust Model Based Fault Diagnosis for Dynamic Systems*, London: Kluwer Academic Publishers, 1999.
- [8] Chen J-Q., Xi Y-G. and Zhang Z-J., "A clustering algorithm for fuzzy model identification", *Fuzzy Sets and Systems*, 1998, vol. 98, pp. 319-329.
- [9] DAMADICS (2002): *Website of the RTN DAMADICS: Development and Application of Methods for Actuator Diagnosis in Industrial Control Systems*. At: <http://diag.mchtr.pw.edu.pl/damadics>
- [10] Delebecque F., Nikoukah R. and Rubio Scola H., "Test signal design for failure detection: A linear programming approach" - *Int. J. Appl. Math. and Comput. Sci.*, 2003, vol. 13, no. 4, pp. 515-526.
- [11] Esparcia-Alcazar A.I. (1998): *Genetic Programming for Adaptive Digital Signal Processing*, Ph.D. dissertation, Glasgow University.
- [12] Frank P.M., (1990): "Fault diagnosis in dynamic systems using analytical and knowledge-based redundancy - a survey and some new results", *Automatica*, 1990, vol. 26, pp. 459-474.
- [13] Frank P.M. and Köppen-Seliger B., "New developments using AI in fault diagnosis", *Artificial Intelligence*, 1997, vol. 10, no. 1, pp. 314.
- [14] Frank P.M., Schreier G. and Garcia E.A., *Nonlinear observers for fault detection and isolation*, New Directions in Nonlinear Observer Design (Nijmeijer H. and T. Fossen, Eds.), Berlin: Springer-Verlag, 1999.
- [15] Gertler J., *Fault Detection and Diagnosis in Engineering Systems*, New York: Marcel Dekker, 1998.
- [16] Goldberg D.E., *Genetic Algorithms in Search, Optimization and Machine Learning*, Massachusetts: Addison-Wesley, 1989.
- [17] Gupta M.M., Jin L. and Homma N., *Static and Dynamic Neural Networks*. From Fundamentals to Advanced Theory, New Jersey: Wiley, 2003.
- [18] Haykin S., *Neural Networks*. A Comprehensive Foundation, New Jersey: Prentice-Hall, 1999.
- [19] Hui S. and Zak S.H., "Observer design for systems with unknown input", *Int. J. Appl. Math. and Comput. Sci.*, 2005, vol. 15, no. 4, pp. 431-446.
- [20] Isermann R., *Fault Diagnosis Systems. An Introduction from Fault Detection to Fault Tolerance*, New York: Springer-Verlag, 2006.
- [21] Ivakhnenko A.G. and Mueller J.A., "Self-organizing of nets

- of active neurons", *System Analysis Modelling Simulation*, 1995, vol. 20, pp. 93-106.
- [22] Janczak A., "Identification of Nonlinear Systems Using Neural Networks and Polynomial Models. A Block-oriented Approach", *Lecture Notes in Control and Information Sciences*, 2005, vol. 310, Berlin: Springer-Verlag.
- [23] Karpenko M., Sepehri N. and Scuse D., "Diagnosis of process valve actuator faults using multilayer neural network", *Control Engineering Practice*, 2003, vol. 11, pp. 1289-1299.
- [24] Köppen-Seliger B. and Frank P.M., "Fuzzy logic and neural networks in fault detection", In: *Fusion of Neural Networks, Fuzzy Sets, and Genetic Algorithms* (L. Jain and N. Martin, Eds.), New York: CRC Press, 1999, pp. 169-209.
- [25] Korbicz J., "Fault detection using analytical and soft computing methods", *Bulletin of the Polish Academy of Sciences: Technical Sciences*, 2006, vol. 54, no. 1, pp. 75-88.
- [26] Korbicz J. (Ed.), "Soft Computing in Control and Fault Diagnosis", *Int. J. Appl. Math. and Comput. Sci.*, 2006a, vol. 16, no. 1, pp. 7-114.
- [27] Korbicz J., "Computer designing of diagnostic systems", In: *Engineering of Machine Diagnostics* (Zółtowski B. and Cz. Cempel, Eds.), Radom: Instytut Technologii Eksploatacji (Institute for Sustainable Technologies in Radom, Poland), 2004, pp. 453-474, (in Polish).
- [28] Korbicz J. and Bidyuk P.I., *State and Parameter Estimation. Digital and Optimal Filtering*, Applications, Technical University of Zielona Góra Press, 1993.
- [29] Korbicz J. and Kus J. (1999): "Dynamic GMDH neural networks and their application in fault detection systems", *Proc. European Control Conference, ECC*, Karlsruhe, Germany, 31 August-3 Sept., CD ROM.
- [30] Korbicz J. and Mrugalski M. (2007): "Confidence estimation of GMDH neural networks and its application in fault detection systems", *Int. J. System Science*, (accepted).
- [31] Korbicz J. and Kowal M., "Neuro-Fuzzy Systems in Processes Diagnostics", In: *Fuzzy Sets and their Applications* (Chojcan J. and Leski J., Eds.), Gliwice, Poland: Wydawnictwo Politechniki Slaskiej (Silesian University of Technology Press), 2001, pp. 333-379, (in Polish).
- [32] Korbicz J., Koscielny J.M., Kowalczyk Z. and Cholewa W. (Eds.), *Fault Diagnosis. Models*, Artificial Intelligence, Applications, Berlin: Springer-Verlag, 2004.
- [33] Korbicz J., Koscielny J.M., Kowalczyk Z. and Cholewa W. (Eds.), *Processes Diagnosis. Models, Methods of Artificial Intelligence, Applications*, Warsaw: Wydawnictwo Naukowo Techniczne, WNT, 2002, (in Polish).
- [34] Korbicz J., Obuchowicz A. and Ucinski D., *Artificial Neural Networks. Foundations and Applications*, Warsaw: Akademicka Oficyna Wydawnicza EXIT, 1994, (in Polish).
- [35] Korbicz J., Patan K. and Obuchowicz A., "Neural network fault detection system for dynamic processes", *Bulletin of the Polish Academy of Sciences. Technical Sciences*, 2001, vol. 49, no. 2, pp. 301-321.
- [36] Korbicz J., Patan K. and Obuchowicz A., "Dynamic neural networks for process modelling in fault detection and isolation systems", *Int. J. Appl. Math. and Comput. Sci.*, 1999, vol. 9, no. 3, pp. 510-546.
- [37] Korbicz J., Witczak M. and Puig V., "LMI-based strategies for designing observers and unknown input observers for non-linear discrete-time systems", *Bulletin of the Polish Academy of Sciences, Technical Sciences*, 2007, vol. 55, no. 1, pp. 31-42.
- [38] Kościelny J.M., *Diagnostics of Automatic Industrial Processes*. Warsaw: Akademicka Oficyna Wydawnicza EXIT, 2001, (in Polish).
- [39] Kowal M., "Optimization of Neuro-fuzzy Structures in Technical Diagnostics", *Lecture Notes in Control and Computer Science*, 2005, vol. 9, University of Zielona Góra Press.
- [40] Kowal M. and Korbicz J., "Robust fault detection using neuro-fuzzy models", *Przegląd Elektrotechniczny*, 2006, no. 1, pp. 32-36.
- [41] Kowal M. and Korbicz J., "Robust fault detection under fuzzy-neural model uncertainty", *Pomiary, Automatyka, Kontrola*, 2005a, no. 9, pp. 93-95, (in Polish).
- [42] Kowal M. and Korbicz J., "Neuro-fuzzy structures in FDI system", *Proc. 16-th IFAC World Congress, 2005b*, Prague, Czech Republic, CD-ROM.
- [43] Kowal M. and Korbicz J., "Self-organizing Takagi-Sugeno fuzzy model in a fault detection system", *Proc. 6-th Nat. Conf. Diagnostics of Industrial Processes 2003, DPP'03*, Władysławowo, Poland, pp. 253-258, (in Polish).
- [44] Kowal M. and Korbicz J., "Fault detection using neuro-fuzzy networks", *Proc. 14-th Polish Conf. Automatic Control, KKA*, Zielona Góra, Poland, 2002a, pp. 595-600, (in Polish).
- [45] Kowal M. and Korbicz J., "Fault detection using Takagi-Sugeno neuro-fuzzy networks and ARX models", *Proc. Nat. Conf. Fuzzy Systems*, 2002b, Cracow, Poland, pp. 198-205.
- [46] Kowal M., Korbicz J., Mendes M.J.G.C. and Calado J.M.F., "Fault detection using neuro-fuzzy networks", *Systems Science*, 2002, vol. 28, no. 1, pp. 45-57.
- [47] Kowalczyk Z. and Białaszewski (2006): "Niche mechanisms in evolutionary computation", *Int. J. Appl. Math. and Comput. Sci.*, vol. 16, no. 1, pp. 59-84.
- [48] Koza J.R., *Genetic Programming: On the Programming of Computers by Means of Natural Selection*, Cambridge: The MIT Press, 1992.
- [49] Leith D.J. and Leithead W.E., "Analytic framework for blended multiple model systems using local models", *Int. J. Control*, 1999, vol. 72, no. 7/8, pp. 605-619.
- [50] Mendes M.J.G.C., Kowal M., Korbicz J. and Sa da Costa J.M.G., "Neuro-fuzzy structures in FDI system", *Proc. 15-th IFAC Triennial World Congress*, 2002, Barcelona, Spain, CD-ROM.
- [51] Michalewicz Z., *Genetic Algorithms + Data Structures = Evolution Programs*, Berlin: Springer-Verlag, 1996.
- [52] Milanese M., Norton J., Piet-Lahanier H. and Walter E., *Bounding Approaches to System Identification*, New York: Plenum Press, 1996.
- [53] Mrugalski M., *Neural Network Based Modelling of Non-linear Systems in Fault Detection Schemes*, Ph.D. thesis, 2004, Faculty of Electrical Engineering, Computer Science and Telecommunications, University of Zielona Góra, (in Polish).
- [54] Mrugalski M., *Neural Network Based Modelling of Nonlinear Systems in Fault Detection Schemes*, Ph.D. thesis, 2003, University of Zielona Góra, Faculty of Electrical Engineering, Computer Science and Telecommunications, (in Polish).
- [55] Mrugalski M. and Korbicz J., "Application of the MLP neural network to the robust fault detection", *Proc. 6-th IFAC Symp. Fault Detection Supervision and Safety of Technical Processes, SAFEPROCESS*, 2006, Beijing, China, CD-ROM.
- [56] Mrugalski M. and Korbicz J. (2005): *Robust fault detection*

- via GMDH neural networks, *Proc. 16-th IFAC World Congress*, Prague, Czech Republic, CD-ROM.
- [57] Mueller J.E. and Lemke F., *Self-organising Data Mining*, Hamburg: Libri, 2000.
- [58] Nelles O., *Nonlinear System Identification. From Classical Approaches to Neural Networks and Fuzzy Models*, Berlin: Springer-Verlag, 2001.
- [59] Norgard M., Ravn O., Poulsen N. and Hansen L., *Networks for Modelling and Control of Dynamic Systems*, London: Springer-Verlag, 2000.
- [60] Obuchowicz A., *Evolutionary Algorithms for Global Optimization and Dynamic System Diagnosis*, Zielona Góra: Lubuskie Towarzystwo Naukowe, 2003.
- [61] Osowski S., *Artificial Neural Networks for Information Processing*, Warsaw: Oficyna Wydawnicza Politechniki Warszawskiej (Warsaw University of Technology Press), 2006, (in Polish).
- [62] Patan K., "Stability analysis and the stabilization of a class of discrete-time dynamic neural networks", *IEEE Trans. Neural Networks*, 2007, vol. 18, (accepted).
- [63] Patan K., Korbicz J. and Prętki P., "Global stability conditions of locally recurrent neural networks", *Lecture Notes in Computer Science*, 2005, vol. 3697, pp. 191-196.
- [64] Patan K. and Parisini T., "Identification of neural dynamic models for fault detection and isolation: The case of a real sugar evaporation process", *J. Process Control*, 2005, vol. 15, pp. 67-79.
- [65] Patton R.J. and Korbicz J. (Eds.), "Advances in Computational Intelligence for Fault Diagnosis Systems", *Int. J. Appl. Math. and Comput. Sci.*, Special Issue, 1999, vol. 9, no. 3, pp. 471-735.
- [66] Patton R.J., Frank P.M. and Clark R.N., *Issues of Fault Diagnosis for Dynamic Systems*, Berlin: Springer-Verlag, 2000.
- [67] Patton R.J., Korbicz J. and Lesecq S. (Eds.), "A Benchmark Study of Fault Diagnosis for an Industrial Actuator", *Control Engineering Practice*, 2006, vol. 14, no. 6, pp. 575-717.
- [68] Patton R.J., Korbicz J., Witczak M. and Uppal F., "Combined computational intelligence and analytical methods in fault diagnosis", In: *Intelligent Control Systems using Computational Intelligence Techniques* (Rauno A.E. (Ed.)), London: The IEE Press, 2005, pp. 349-392.
- [69] Piegat A. (2003): *Modelling and Fuzzy Systems*, Warsaw: Akademicka Oficyna Wydawnicza EXIT, (in Polish).
- [70] Rodrigues M., Theilliol D., Aberkane S. and Sauter D., "Fault tolerant control design for polytopic LPV systems", *Int. J. Appl. Math. and Comput. Sci.*, 2007, vol. 17, no. 1, pp. 27-37.
- [71] Rutkowska D., *Neuro-Fuzzy Architectures and Hybrid Learning*, Heidelberg, New York: Physica-Verlag, Springer-Verlag Company, 2002.
- [72] Rutkowski L., *Methods and Techniques of Artificial Intelligence*, Warsaw: Wydawnictwo Naukowe PWN, 2005, (in Polish).
- [73] Tadeusiewicz R., *Neural Networks*, Warsaw: Akademicka Oficyna Wydawnicza, 1993, (in Polish).
- [74] Thau F.E., "Observing the state of non-linear dynamic systems", *Int. J. Control*, 1973, vol. 17, no. 3, pp. 471-479.
- [75] Uciński D., *Optimal Measurement Methods for Distributed Parameter System Identification*, Boca Raton: CRC Press, 2005.
- [76] Uppal F., Patton R.J. and Witczak M., "A neuro-fuzzy multiple-model observer approach to robust fault diagnosis based on the DAMADICS benchmark problem", *Control Engineering Practice*, 2006, vol. 14, no. 6, pp. 699-717.
- [77] Walter E. and Pronzato L., *Identification of Parametric Models from Experimental Data*, London: Springer, 1997.
- [78] Wang S.H., Davison E.J. and Dorato P., "Observing the states of systems with unmeasurable disturbances", *IEEE Trans. Automatic Control*, 1975, vol. 20, no. 5, pp. 716-717.
- [79] Witczak M., "Modelling and Estimation Strategies for Fault Diagnosis of Non-linear Systems", *Lecture Notes in Control and Information Sciences*, 2007, no. 354, Berlin, Heidelberg: Springer.
- [80] Witczak M., "Advances in model-based fault diagnosis with evolutionary algorithms and neural networks", *Int. J. Appl. Math. and Comput. Sci.*, 2006a, vol. 16, no. 1, pp. 85-99.
- [81] Witczak M., "Toward the training of feed-forward neural networks with the D-optimum input sequence", *IEEE Trans. Neural Networks*, 2006b, vol. 17, no. 2, pp. 357-373.
- [82] Witczak M., "Identification and Fault Detection of Non-linear Dynamic Systems", Serie: *Lecture Notes in Control and Computer Science*, Technical University of Zielona Góra Press, 2003, vol. 1.
- [83] Witczak M. and Korbicz J., "Design of observers for Lipschitz non-linear discrete-time systems", *Proc. 14-th IFAC Symp. System Identification, SYSID*, Newcastle, Australia, 2006, pp. 985-990, CD-ROM.
- [84] Witczak M. and Korbicz J., "Observers and genetic programming in the identification and fault diagnosis of non-linear dynamic systems", In: *Fault Diagnosis. Models, Artificial Intelligence, Applications* (Korbicz J., Kościelny J.M., Kowalczyk Z. and Cholewa W., (Eds.)), Berlin: Springer-Verlag, 2004, pp. 457-509.
- [85] Witczak M., Korbicz J., Mrugalski M. and Patton R.J., "A GMDH neural network-based approach to robust fault diagnosis: application to the DAMADICS benchmark problem", *Control Engineering Practice*, 2006a, vol. 14, no. 6, pp. 671-683.
- [86] Witczak M., Korbicz J. and Puig V., "An LMI approach to designing observers and unknown input observers for nonlinear systems", *Proc. 6-th IFAC Symp. Fault Detection Supervision and Safety of Technical Processes, SAFEPROCESS*, 2006b, Beijing, China, CD-ROM.
- [87] Witczak M., Obuchowicz A. and Korbicz J., "Genetic programming based approaches to identification and fault diagnosis of non-linear dynamic systems", *Int. J. Control*, 2002, vol. 75, no. 13, pp. 1012-1031.
- [88] Witczak M., Obuchowicz A. and Korbicz J., "Design of non-linear state observers using genetic programming", *Proc. 3-rd Nat. Conf. Evolutionary Algorithms and Global Optimization*, 1999, Potok Złoty, Poland, pp. 345-352.
- [89] Witczak M., Prętki P., Korbicz J. and Puig V., "Design of an extended unknown input observer", *Proc. Int. Workshop Advanced Control Diagnosis, ACD*, 2006, Nancy, France, CD-ROM.
- [90] Zolghadri A., Henry D. and Monson M., "Design of non-linear observers for fault diagnosis. A case study", *Control Engineering Practice*, 1996, vol. 4, no. 11, pp. 1535-1544.