A GMDH NEURAL NETWORK BASED APPROACH TO PASSIVE ROBUST FAULT DETECTION USING A CONSTRAINTS SATISFACTION BACKWARD TEST

V. Puig[a], M. Mrugalski [b], A. Ingimundarson[a], J. Quevedo[a], M. Witczak [b], J. Korbicz [b]

[a]*Automatic Control Department (ESAII) - Campus de Terrassa*
Universidad Politécnica de Cataluña (UPC)
Rambla Sant Nebridi, 10. 08222 Terrassa (Spain)
vicenc.puig@upc.es
[b]*Institute of Control and Computation Engineering*
University of Zielona Góra
M.Mrugalski @issi.uz.zgora.pl

Abstract: This paper focus on the problem of passive robust fault detection using non-linear models that include parameter uncertainty. The non-linear model considered here is described by a Group Method of Data Handling Neural Network (GMDHNN). The problem of passive robust fault detection using models including parameter uncertainty has been mainly addressed checking if the measured behaviour is inside the region of possible behaviours following what will be called in the following a forward test. In this paper, a backward test based on checking if there exists a parameter in the uncertain parameter set that is consistent with the measured behaviour is introduced. This test is implemented using interval constraint satisfaction algorithms which can perform efficiently in deciding if the measured state is consistent with the GMDHNN model and its associated uncertainty. Finally, this approach is tested on the servoactuator proposed as a FDI benchmark in the European Project DAMADICS.

Keywords: Fault Detection, Fault Diagnosis, Robustness, Adaptive Threshold, Neural Network.

1. INTRODUCTION

Model-based fault detection is based on the use of mathematical models of the monitored system. The better the model used to represent the dynamic behaviour of the system, the better will be the chance of improving the reliability and performance in detecting faults. However, modelling errors and disturbances in complex engineering systems are inevitable, and hence there is a need to develop robust fault detection algorithms. The ***robustness*** of a fault detection system means that it must be only sensitive to faults, even in the presence of model-reality differences (Chen, 1999). One of the approaches to robustness, known as ***passive***, enhances the robustness of the fault detection system at the decision-making stage, mainly using an adaptive threshold. The passive approach based on adaptive thresholds is based not in avoiding the effect of uncertainty in the residual through perfect decoupling, but in propagating the uncertainty to the residual, and then bounding the residual uncertainty. Of course this approach has the drawback that faults that produce a residual deviation smaller than the residual uncertainty due to parameter uncertainty will be missed.

This paper focus on the problem of passive robust fault detection using Artificial Neural Networks (ANNs) (Korbicz et al., 2004). The attractiveness of the ANNs in the robust passive fault detection schemes follows from the fact that they are useful when there are no phenomenological models available, i.e. the models, which are built with the physical consideration of the underlying system of interest. Such a situation causes that the models, which merely approximate the observed behaviour, should be employed. Unfortunately, there are no efficient algorithms for selecting structures of classical ANNs and hence many experiments should be carried out to obtain an appropriate configuration of the neural model. The appropriate quality of the model settles about the reliability and performance of the fault detection system. To tackle this problem the GMDH approach can be employed (Ivakhnenko and

Mueller, 1995). The concept of the synthesis of the GMDHNN is based on the iterative processing of a defined sequence of operations leading to the evolution of the resulting structure, which generates the best possible approximation of the real system output. Furthermore, the GMDH approach make possible to determine not only the structure and parameters of the neural network but also associated parameter's uncertainty (Mrugalski, 2004; Witczak and Mrugalski, 2003). The problem of passive robust fault detection using models including parameter uncertainty has been mainly addressed checking if the measured behaviour is inside the region of possible behaviours following what will be called in the following a ***forward test*** (Armengol et al., 2001; Travé-Massuyès and Milne, 1997; Puig et al., 2002; Patton and Korbicz, 1999; Escobet et al., 2001; Ploix et al., 2000; Witczak et al., 2005). In this paper, a ***backward test*** based on checking if there exists a parameter in the uncertain parameter set that is consistent with the measured behaviour is introduced. This test is implemented using interval constraint satisfaction algorithms, which can perform efficiently in deciding if the measurements are consistent with the GMDHNN model and associated uncertainty. Finally, this approach is tested on several real fault scenarios of a servo-actuator proposed as a FDI benchmark in the European Project DAMADICS.

2. ROBUST MODEL BASED FAULT DETECTION USING BEA GMDHNN

2.1 Model based fault detection principle

Model-based fault detection is based on the generation of a discrepancy between the measured and estimated process behaviours using a model. This discrepancy is known as a ***residual***. A ***residual generator*** can be constructed by

$$r(k) = y(k) - \hat{y}(k), \tag{1}$$

where: $y(k)$ and $\hat{y}(k)$ are the measured and estimated outputs using a model, respectively. The residual signal should be normally close to zero in the fault free mode, otherwise it should be distinguishably different from zero when a fault occurs. The residual should ideally carry only fault information.

2.2 Robustness issues

However, the presence of disturbances, noise and modelling errors causes the residuals to become nonzero and interfering with the detection of faults. Therefore, the fault detection procedure has to be ***robust*** in the face of these undesired effects. Robustness can be achieved in the residual generation (***active robustness***) or in the decision making stage (***passive robustness***) (Chen and Patton, 1999). The passive approach, when considering the uncertainty described by a vector $\boldsymbol{\theta}$ of uncertain parameters of dimension n_p with their values bounded by a compact set $\boldsymbol{\Theta}$, is based not in avoiding the effect of uncertainty in the residual, but in propagating the effect of uncertainty to the estimated output such that

$$y(k) \in \left[\underline{\hat{y}}(k), \overline{\hat{y}}(k)\right], \tag{2}$$

or equivalently to the residual:

$$\begin{aligned} r(k) &= y(k) - \hat{y}_c(k) \\ &\in \left[-\Delta\hat{y}(k), \Delta\hat{y}(k)\right] = \left[\underline{r}(k), \overline{r}(k)\right] \end{aligned}, \tag{3}$$

where: $\hat{y}_c(k) = \frac{1}{2}(\underline{\hat{y}}(k) + \overline{\hat{y}}(k))$ is the predicted output interval centre and $\Delta\hat{y}(k) = \frac{1}{2}(\overline{\hat{y}}(k) - \underline{\hat{y}}(k))$ the radius. Then, the detection test can be translated to check if either condition (2) or (3) are satisfied. Otherwise, the fault should be indicated. This approach will be known in the following as the ***forward detection test***.

2.3 BEA GMDHNN

In case of using a GMDHNN, $\hat{y}(k)$ can be written in the following form

$$\hat{y}(k,\boldsymbol{\theta}) = g(\boldsymbol{\theta}_1^{(1)}, \cdots, \boldsymbol{\theta}_{n_1}^{(1)}, \cdots \boldsymbol{\theta}_1^{(L)}, \cdots, \boldsymbol{\theta}_{n_L}^{(L)}), \tag{4}$$

where $g(\cdot)$ stands for the neural network structure obtained using the approach proposed in (Mrugalski, 2004; Witczak et al., 2005), L is the number of layers of GMDHNN model and n_l is the number of neurons in the l-th layer. Each neuron has the following structure

$$\hat{y}_n^{(l)}(k, \boldsymbol{\theta}_n^{(l)}) = \xi\left(\left(\boldsymbol{\varphi}_n^l(k)\right)^T \boldsymbol{\theta}_n^{(l)}\right), \tag{5}$$

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

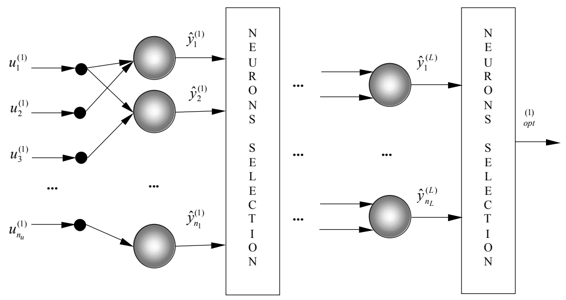

Fig. 1 A GMDH neural network

The structure and parameters of GMDHNN can be determined as well as the modelling uncertainty using algorithms proposed in (Mrugalski, 2004; Witczak et al., 2005). Parameters and its associated uncertainty will define a set of admissible parameter values (*feasible parameter set*) that should provide a confidence region for the estimation of model output such that all available measured output in non-faulty situation are included, i.e., $y(k) \in \left[\underline{\hat{y}}(k), \bar{\hat{y}}(k)\right]$. Parameter uncertainty is determined using the Bounding Error Approach (BEA) to parameter estimation (Milanese et al., 1996). This is way such neural network will be called in the following as BEA GMDH. The bounding-error approach allows estimating parameter uncertainty without any assumption on statistical properties of the noise. The only assumption is that its value is bounded $\varepsilon(k) \in \left[\underline{\varepsilon}(k), \bar{\varepsilon}(k)\right]$. These bounds are known *a priori* but can also be estimated (Witczak et al., 2005). The feasible parameter set is defined as

$$\begin{aligned}\boldsymbol{\Theta} = \{&\boldsymbol{\theta} \in \Re^p \mid y(k) - \bar{\varepsilon}(k) \le \boldsymbol{\varphi}^T(k)\boldsymbol{\theta} \le y(k) - \underline{\varepsilon}(k), \\ &k = 1, \cdots, n_T\}\end{aligned} \quad (6)$$

where n_T is the number of input-output measurements. It can be obtained as a region of the parameter space determined by n_T pairs of hyperplanes

$$\boldsymbol{\Theta} = \bigcap_{k=1}^{n_T} \mathcal{S}(k), \quad (7)$$

where each pair defines the parameter strip (Fig. 2)

$$\begin{aligned}\mathcal{S}(k) = \{&\boldsymbol{\theta} \in \Re^p \mid \\ &y(k) - \bar{\varepsilon}(k) \le \boldsymbol{\varphi}^T(k)\boldsymbol{\theta} \le y(k) - \underline{\varepsilon}(k)\}\end{aligned}, \quad (8)$$

Any parameter vector contained in $\boldsymbol{\Theta}$ is a valid estimate of $\boldsymbol{\theta}$.

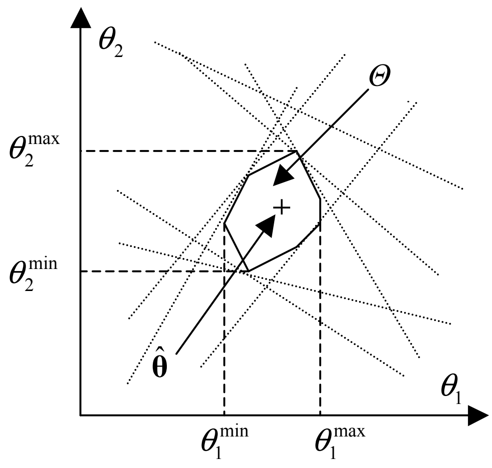

Fig. 2 Feasible parameter set

3. FORWARD AND BACKWARD FAULT DETECTION TESTS USING BEA GMDHNN

3.1 Forward detection test

Let $\mathcal{V}$ be the set of all vertices $\boldsymbol{\theta}^i$, $i=1,\ldots,n_v$ describing the feasible parameter set $\boldsymbol{\Theta}$. Considering that there is no error in the regressor, the interval for the estimated output for a given neuron when its activation function is linear since the uncertainty in the estimated output can be determined by

$$\boldsymbol{\varphi}^T(k)\underline{\boldsymbol{\theta}}(k) + \underline{\varepsilon}(k) \le y(k) \le \boldsymbol{\varphi}^T(k)\bar{\boldsymbol{\theta}}(k) + \bar{\varepsilon}(k), \quad (9)$$

where:

$$\underline{\boldsymbol{\theta}}(k) = arg \min_{\theta \in \mathcal{V}} \varphi^T \boldsymbol{\theta}, \qquad \bar{\boldsymbol{\theta}}(k) = arg \max_{\theta \in \mathcal{V}} \varphi^T \boldsymbol{\theta}.$$

In case of errors in the regressors and non-linear activation functions in the neurons these expressions can be adapted as in (Mrugalski, 2004; Witczak and Mrugalski, 2003). The drawback of this approach is that it loses the dependence between parameters and the outputs while computing the envelope. Only maximum and minimum values of the parameters are used for the envelope. This leads to tests that is specially sensitive to faults close to the extreme points of the parameter intervals (corners) while less sensitive to faults where some parameters are close to the center of the its interval and where many more parameter combinations can explain the data.

3.2 Backward detection test

The backward detection test applied to the GMDHNN consists on checking if there exist some $\boldsymbol{\theta} \in \boldsymbol{\Theta}$ such that:

$$\exists \boldsymbol{\theta} \in \boldsymbol{\Theta} \mid y(k) - \bar{\varepsilon}(k) \le \boldsymbol{\varphi}^T(k)\boldsymbol{\theta} \le y(k) - \underline{\varepsilon}(k). \quad (10)$$

In case that does not exist any parameter $\boldsymbol{\theta} \in \boldsymbol{\Theta}$ such that (10) is satisfied, a discrepancy between the measured output and the model is detected and a fault should be indicated. In fact this test can be viewed as a kind of parameter identification, since the set of parameters that are consistent with the actual set of N test measurements are

$$\boldsymbol{\Theta}_N = \{\boldsymbol{\theta} \in \Re^p \mid y(k) - \bar{\varepsilon}(k) \le \boldsymbol{\varphi}^T(k)\boldsymbol{\theta} \le y(k) - \underline{\varepsilon}(k), \; k = 1, \cdots, N\} \quad (11)$$

Then, equivalently, the backward fault detection test consists in checking if

$$\boldsymbol{\Theta}_N \cap \boldsymbol{\Theta} \neq \varnothing . \quad (12)$$

The computation of the set of parameters consistent with a given set of N measurements

$$\boldsymbol{\mathcal{Y}} = [y(1)] \times \cdots \times [y(N)], \quad (13)$$

with $[y(k)] = [y(k) - \bar{\varepsilon}(k), y(k) - \underline{\varepsilon}(k)]$ can be computed using the same bounded error parameter estimation algorithms used in computing the feasible parameter set (Milanese et al., 1996) or by computing the ***inverse image*** of the interval function g introduced in (4) describing the neural net model:

$$\boldsymbol{\Theta}_N = g^{-1}(\boldsymbol{Y}), \quad (14)$$

Jaulin (2001) has proposed an algorithm called SIVIA that computes the inverse image of an interval function using subpavings. This algorithm make use that the point test

$$t(\boldsymbol{x}) = (\boldsymbol{x} \in f^{-1}(\boldsymbol{\mathcal{Y}})), \quad (15)$$

associated to the inverse image can be easily evaluated computing $f(\boldsymbol{x})$ and checking is belongs to $\boldsymbol{\mathcal{Y}}$. However, when the dimension of the set to characterize is of high dimension the computational complexity explodes since SIVIA uses bisection in all directions.

Moreover, after computing (14), either using bounded error parameter estimation algorithms either using the inverse image of an interval function using SIVIA, the intersection presented in (12) should be computed, being not an easy task in general (Ploix et al., 2000). Then, the backward test presented in Section *3.2* will be implemented using (10) using constraints satisfaction algorithms since the use of contractors save a lot of computation because bisections are only used when required.

4. IMPLEMENTING THE BACKWARD TEST USING CONSTRAINTS SATISFACTION

4.1 Constraint satisfaction problem

An ***interval constraint satisfaction problem*** (ICSP) can be formulated as a 3-tuple $\mathcal{H} = (\mathcal{V}, \mathcal{D}, \mathcal{C})$, where $\mathcal{V} = \{v_1, \cdots, v_n\}$ is a finite set of variables, $\mathcal{D} = \{[v_1], \cdots, [v_n]\}$ is the set of their domains represented by closed real intervals and $\mathcal{C} = \{c_1, \cdots, c_n\}$ is a finite set of constraints relating variables of $\mathcal{V}$ (Waltz, 1975; Hyvönen, 1992) A ***point solution*** of $\mathcal{H}$ is a n-tuple $(\tilde{v}_1, \cdots, \tilde{v}_n) \in \mathcal{V}$ such that all constraints $\mathcal{C}$ are satisfied. The set of all point solutions of $\mathcal{H}$ is denoted by $\mathcal{S}(\mathcal{H})$. This set is called the ***global solution set***. The variable $v_i \in \mathcal{V}_i$ is ***consistent*** in $\mathcal{H}$ if and only if:

$$\forall v_i \in \mathcal{V}_i \; \exists (\tilde{v}_1 \in [v_1], \cdots, \tilde{v}_i \in [v_i], \cdots, \tilde{v}_n \in [v_n]) \mid (\tilde{v}_1, \cdots, \tilde{v}_n) \in \mathcal{S}(\mathcal{H}) \quad (16)$$

4.2 Backward test as an ICSP

The fault detection test in (10) can be formulated as an ICSP in the following way:

$$\begin{aligned} & y(1) \in [y(1)] \\ & \cdots \\ & y(k) \in [y(k)] \\ & \boldsymbol{\theta} \in \boldsymbol{\Theta} \\ & y(1) = \boldsymbol{\varphi}^T(1)\boldsymbol{\theta} \\ & \cdots \\ & y(k) = \boldsymbol{\varphi}^T(k)\boldsymbol{\theta} \end{aligned} \quad . \quad (17)$$

Then, using an ICSP solver as, for example, *Interval Peeler* developed by the group of Jaulin using the principles described in (Jaulin, 2001) (see http://www.istia.univ-angers.fr/~baguenar/), this problem can be solved. In case that no solution is found, a fault should be indicated since there is no parameter $\boldsymbol{\theta} \in \boldsymbol{\Theta}$ such that (10) is satisfied. Since in the case of the GMDHNN, the feasible parameter set is not described by parameter vector but instead by a polygon, being $\mathcal{V}$ be the set of all vertices $\boldsymbol{\theta}^i$, $i=1,\ldots,n_v$ (Fig. 2), additional restrictions should be introduced in (17) to reflect it. Each pair of adjacent vertices $\boldsymbol{\theta}^i, \boldsymbol{\theta}^{i+1}$ introduce a linear restriction of the following type:

$$f(\boldsymbol{\theta}, \boldsymbol{\theta}^i, \boldsymbol{\theta}^{i+1}) = a_1(\boldsymbol{\theta}^i, \boldsymbol{\theta}^{i+1})\theta_1 + \cdots + a_{n_p}(\boldsymbol{\theta}^i, \boldsymbol{\theta}^{i+1})\theta_{n_p} - b(\boldsymbol{\theta}^i, \boldsymbol{\theta}^{i+1}) \le 0 , \quad (17)$$

with $\boldsymbol{\theta} \in hull(\boldsymbol{\Theta}) = [\underline{\boldsymbol{\theta}}, \bar{\boldsymbol{\theta}}]$, i.e., the minimum interval box containing the feasible parameter set $\boldsymbol{\Theta}$. In this case $hull(\boldsymbol{\Theta})$ can be easily computed

$$\underline{\theta}_i = \min_{\theta \in \mathcal{V}} \theta_i , \quad i = 1, \cdots, n_p$$

$$\bar{\theta}_i = \max_{\theta \in \mathcal{V}} \theta_i , \quad i = 1, \cdots, n_p .$$

5. APPLICATION TO THE DAMADICS BENCHMARK

The application example proposed to show the effectiveness of the backward fault detection approach using a model based on GMDHNNs is based on the FDI DAMADICS Benchmark. The real data used for system identification and fault detection were collected on 17th November 2001. A detailed description regarding the data and the artificially introduced faults can be found on DAMADICS website (DAMADICS, 2004). Based on the actuator benchmark definition a GMDH model g describing the juice flow F is designed:

$$F = g(X, P_1, P_2, T_1), \tag{18}$$

where X is the servomotor rod displacement, P_1 is the pressure before the valve, P_2 is the pressure after the valve and T_1 is the juice temperature before the valve. Using the training algorithm proposed by (Witczak et al., 2005) the final structure of the GMDHNN is presented in Fig. 3

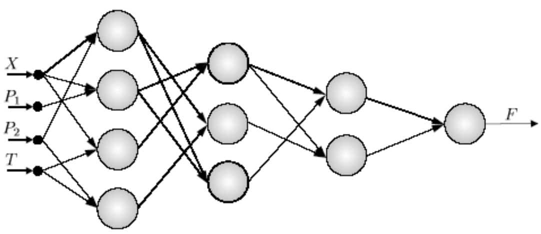

Fig. 3 Final structure of $F = g(\cdot)$

Forward and backward tests are applied to the last neuron using (9) and (10), respectively. In this case there are two uncertain parameters (θ_1 and θ_2), each one associated to a one of the last neuron inputs.

5.1 Fault scenario f_{17}

The first fault scenario considered consists of an unexpected pressure drop across the valve. Using the forward test, the fault is detected at sample k=38 (Fig. 4) since at this time the measured output leaves the prediction interval provided by the GMDHNN. On the other hand, in Fig. 5 and 6 the result of the backward test is presented. In this case, the fault is detected at the same sample k=38 (Fig. 6), since when considering the corresponding measured output, the set of parameters consistent with this measurement (in solid line) do not intersect with the feasible parameter set obtained in the training phase (vertices shown with crosses) (Fig. 5). The set of parameters consistent with output envelopes (in dotted line) also do not contain the parameters consistent with the measurements what explain why measurements go out the envelope. In Fig. 5, the result of the backward test is presented when sample k=37 is considered. In this case, set of parameters consistent with measurements intersect with the feasible parameter set determined in the training phase, so the fault is not detected. The set of parameters consistent with envelope intersects with the feasible parameter set what explains why at that time the envelope is not violated.

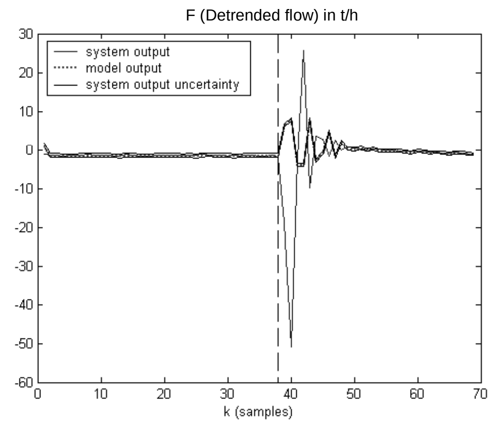

Fig. 4 Forward detection test for fault f_{17}

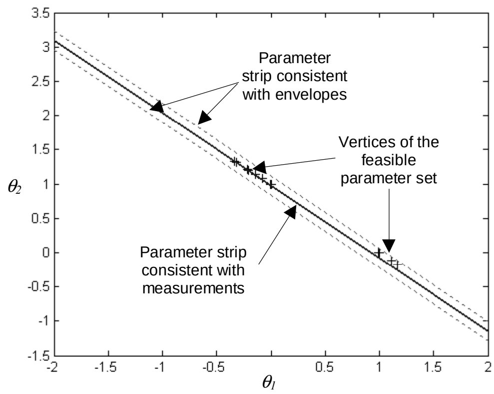

Fig. 5 Backward detection test for fault f_{17} using data at time k=37

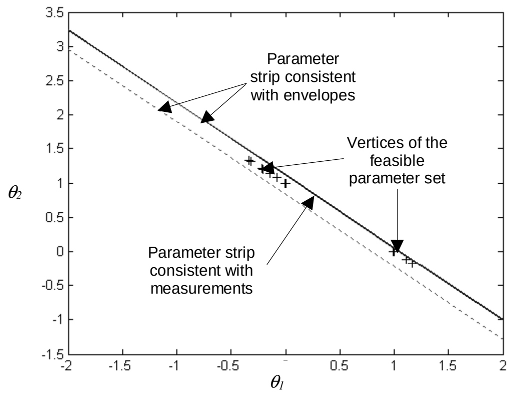

Fig. 6 Backward detection test for fault f_{17} using data at time k=38

5.2 Fault scenario f_{19}

The second fault scenario considered consists of a flow rate sensor fault. In Fig. 7, the fault detection result is presented when the forward test is used. In this case fault is detected at time k=32. On the other hand, Fig. 8 presents the fault detection result when the backward test is applied. In this case using a measurement at time k=28 the fault is detected. This can be seen in this figure since the set of parameters consistent with measurement do not intersect with the feasible parameter set described by its vertices represented using crosses. However, the set of parameters consistent with envelope intersect with this set being this reason why the fault is not detected using the forward test.

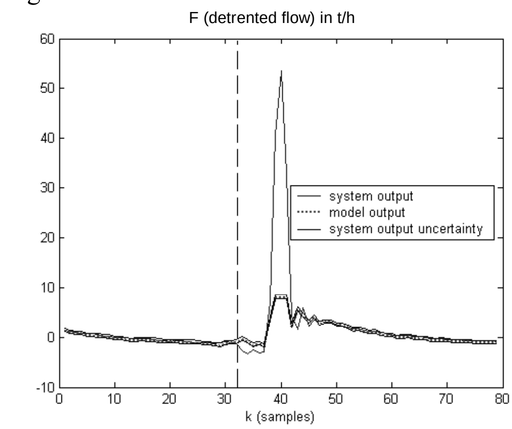

Fig. 7 Forward detection test for fault f_{19}

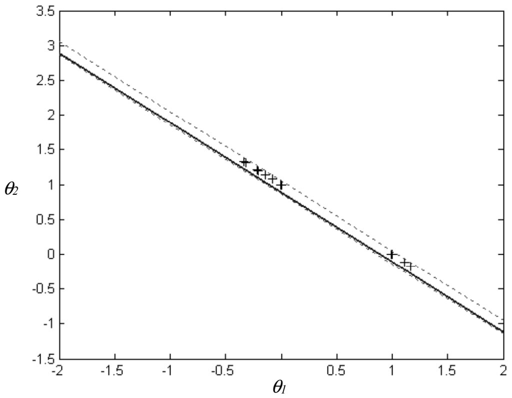

Fig. 8 Backward detection test for fault f_{19} using data at time k=28

6. CONCLUSIONS

In this paper, a GMDHNN based approach to passive robust fault detection is presented using a constraints satisfaction backward test. In the passive robustness FDI literature, the forward fault detection test based on propagating the parameter uncertainty to the residual or predicted output has been the predominant. Here, a backward fault detection test based on checking if there is a value inside the uncertainty model that can explain the measured output is introduced. The backward test has been implemented using constraints satisfaction tools. The usefulness of this test has been shown when compared to the classical forward test in several real fault scenarios proposed in the DAMADICS FDI benchmark.

ACKNOWLEDGMENTS

The authors wish to thank the support received by the Research Comission of the Generalitat of Catalunya (ref. 2001SGR00236) and by Spanish CICYT (ref. DPI2002-02147). The authors also want to acknowledge funding support under the EC RTN contract (RTN-1999-00392) DAMADICS. Thanks are expressed to the management and staff of the Lublin sugar factory, Cukrownia Lublin SA, Poland for their collaboration and provision of manpower and access to their sugar plant.

REFERENCES

Armengol, J., Vehi, J., Travé-Massuyès, L. and M.A. Sainz (2000). *Interval Model-Based Fault Detection using Multiple Sliding Windows*. IFAC SAFEPROCESS'00, Budapest, Hungary, pp.168-173.

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

DAMADICS (2004). Website RTN DAMADICS: *Development and Application of Methods for Actuator Diagnosis in Industrial Control Systems*. http://diag.mchtr.pw.edu.pl/damadics.

Escobet, T., Travé, L., Tornil, S. and J. Quevedo (2001). *Fault Detection of a Gas Turbine Fuel Actuator Based on Qualitative Causal Models*. ECC'01. Portugal.

Hyvönen, E. (1992). Constraint reasoning based on interval arithmetic: The tolerance approach. *Artificial Intelligence*, 58, pp.71-112.

Ivakhnenko, A.G. and J.A. Mueller (1995). Self-organizing of nets of active neurons. *System Analysis Modelling Simulation*, 20, pp.93-106.

Jaulin, L., Kieffer, M., Didrit, O. and E. Walter (2001). *Applied Interval Analysis, with Examples in Parameter and State Estimation, Robust Control and Robotics*. Springer-Verlag. London.

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

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

Mrugalski, M. (2004). *Neural Network Based Modelling of Non-linear Systems in Fault Detection Schemes*. Zielona Góra, *Faculty of Electrical Engineering, Computer Science and Telecommunications, University of Zielona Góra*, doctoral dissertation. (In Polish)

Ploix, S., Adrot, O. and J. Ragot (2000). *Bounding Approaches to the Diagnosis of Uncertain Static Systems*. IFAC SAFEPROCESS'00, Budapest, Hungary, pp.149-154.

Puig, V., Quevedo, J., Escobet, T., S. De las Heras (2002). *Robust Fault Detection Approaches using Interval Models*. IFAC World Congress (b'02). Barcelona. Spain, CD-ROM.

Patton, R.J. and J. Korbicz (Eds.) (1999). Advances in computational intelligence for fault diagnosis systems. *Applied Mathematical and Computer Science*, special issue, **9**(3), pp.468-735.

Travé-Massuyès, L. and R. Milne (1997). TIGER: Gas turbine condition monitoring using qualitative model based diagnosis. *IEEE Expert Intelligent Systems and Applications*, May-June.

Waltz, D. (1975). *Understanding Line Drawings of Scenes with Shadows*. P.H. Winston Ed. *The Psychology of Computer Vision*. McGraw-Hill, New York, pp.19-91.

Witczak, M., Korbicz, J., Mrugalski, M. and R.J. Patton (2005). A GMDH neural network based approach to robust fault detection and its application to solve the DAMADICS benchmark problem. *Control Engineering Practice* (accepted).

Witczak, M. and M. Mrugalski (2003). *Designing GMDH Neural Networks with the Bounded-Error Approach: Application to Robust Fault Detection.* Proc. 9th IEEE Inter. Conf. Methods and Models in Automation and Robotics, MMAR 2003, Międzyzdroje, Poland, pp.1377-1382.