IFAC
Publications
www.elsevier.com/locate/ifac

DYNAMIC NEURAL NETWORKS FOR ACTUATOR FAULT DIAGNOSIS: APPLICATION TO THE DAMADICS BENCHMARK PROBLEM

Krzysztof Patan *,[1] **Thomas Parisini** **

* *Institute of Control and Computation Engineering, University of Zielona Góra*
ul. Podgórna 50, 65-246 Zielona Góra, Poland,
K.Patan@issi.uz.zgora.pl
** *Dept. of Electrical, Electronic and Computer Engineering, DEEI-University of Trieste*
Via Valerio 10, 34127 Trieste, Italy,
parisini@univ.trieste.it

Abstract: The paper presents results achieved during realization of the international project DAMADICS (*Development and Application of Methods for Actuator Diagnosis in Industrial Control Systems*). The proposed fault detection and isolation system is designed using a bank of dynamic neural networks. Each network is trained using a stochastic approximation method, which can be viewed as a fast alternative to back-propagation based algorithm. Simulation results are carried out using the real process data recorded at the Lublin Sugar Factory, Poland.

Keywords: Actuators, benchmark examples, fault diagnosis, dynamic models, neural networks, stochastic approximation, performance indexes.

1. INTRODUCTION

Methods of FDI based on system identification and residual generation have been intensively studied for the last two decades. One of the most important classes of the FDI methods is neural modelling (Frank and Köppen-Seliger, 1997; Chen and Patton, 1999; Patton *et al.*, 2000). Artificial neural networks can be applied to nonlinear systems. They are useful when there are no mathematical models of the diagnosed system, hence, analytical models and parameter-identification algorithms cannot be applied. One of the most interesting solutions of the dynamic system identification problem is the application of neural networks composed of Dynamic Neuron Models (DNM) (Ayoubi, 1994; Patan and Parisini, 2002*b*). Such neuron models consists of an adder module, a linear dynamic system - Infinite Impulse Response (IIR) filter, and nonlinear activation module. Thus, the DNM activation depends on its actual inputs as well as inputs and outputs in proceeding time. The relatively complex DNM allows one to design a neural network of a feed-forward multi-layer structure (Patan and Parisini, 2002*a*). Derivation of the optimal neural network parameters, however, is not a trivial problem. This process seems to be an optimization problem, which is intrinsically related to a very rich topology. The effectiveness of the gradient based algorithms in many cases is very poor, because it usually finds one of the local minima.

[1] This work was supported by the EU FP 5 project DAMADICS

The paper proposes to use dynamic neural networks trained with stochastic approximation algorithm to design an actuator fault diagnosis system. The experiments are performed using real process data recorded at the Lublin Sugar Factory in Poland.

The paper is organized as follows. In the following Section 2 the problem of fault detection and isolation is formulated. The dynamic neuron model and considered network structure are introduced in Section 3. In Section 4, the stochastic approximation algorithm details and its adaptation to neural network training is explained. Section 5 describes the actuator to be diagnosed. Experimental results are presented in the next two sections.

2. FAULT DIAGNOSIS SCHEME

The technological process can be treated as a system which consists of some major types of subsystems, e.g. actuators, main structure (process) and sensors. The aim of a diagnostic system is to detect the faults in each subsystem and provide the information about their size and sources (Patton *et al.*, 2000). One of the most known fault detection and isolation methods is technique based on residual generation. The fault detection system based on residual generation can be designed using the network of dynamic neurons (Patan, 2000). In this scheme the residual generation and the residual evaluation operations are realized. First of all, the bank of neural models should be designed. Each neural model identifies one class of system behaviour. One model identifies system at its normal operation conditions and each another for suitable fault respectively. After that the residuals can be determined by comparison system output $y(k)$ and output of neural models $y_0(k), y_1(k), \dots, y_n(k)$, respectively. In this way the residual $\mathbf{r} = [r_0\ r_1 \dots r_n]$, which characterize each fault can be obtained. The residual $\mathbf{r}$ should be transformed by a classifier to determine location and occurrence of the faults.

3. DYNAMIC NEURAL NETWORK

Artificial neural networks can be viewed as an alternative solution to classical methods such as state observers or parameter estimation methods (Frank and Köppen-Seliger, 1997; Calado *et al.*, 2001). There are very useful especially when there is no mathematical model of the diagnosed system available. One of the possible solution, which provides dynamic behaviour of the neural model, is a network designed using dynamic neuron models. Such networks have architecture that is somewhere inbetween a feed-forward and globally recurrent architecture. Tsoi and Back (1994) called this class of neural networks the Local Recurrent Global Feed-forward (LRGF). There are some interesting motivations which cause that LRGF networks are very attractive (Campolucci *et al.*, 1999; Tsoi and Back, 1994). Among many, it is worth noting the following:

(1) Well-known neuron inter-connection topology,
(2) Small number of neurons required for a given problem,
(3) Stability of the network. The globally recurrent architectures have a lot of problems in settling to an equilibrium value,
(4) Simpler training than globally recurrent networks.

To design fault diagnosis scheme mentioned in the previous section, dynamic neural networks belonging to the LRGF class can be used. In such a network, dynamic characteristics are obtained using neuron models, which contain inner feedbacks. In this paper neuron models with infinite impulse response filter are considered (Ayoubi, 1994; Patan, 2000). In such models one can distinguish three main parts: a weight adder, a filter and an activation block. The behaviour of the dynamic neuron model is described by the following set of equations:

$$\begin{aligned} x(k) &= \sum_{p=1}^{P} w_p u_p(k) \\ \widetilde{y}(k) &= -\sum_{i=1}^{n} a_i \widetilde{y}(k-i) + \sum_{i=0}^{n} b_i x(k-i) \\ y(k) &= F(g\widetilde{y}(k) + c) \end{aligned} \tag{1}$$

where w_p, $p = 1, \dots, P$ denotes the input weights, $u_p(k)$, $p = 1, \dots, P$ are the neuron inputs, P is the number of inputs, $\widetilde{y}(k)$ is the filter output, a_i, $i = 1, \dots, n$ and b_i, $i = 0, \dots, n$ are the feedback and, feed-forward filter parameters, respectively, and n denotes filter order, $F(\cdot)$ is a non-linear activation function that produces the neuron output $y(k)$, and g and c are the slope parameter and threshold of the activation function, respectively. All unknown network parameters can be represented as a vector $\boldsymbol{\theta}$ which consist of matrices $\mathbf{w}, \mathbf{a}, \mathbf{b}, \mathbf{g}, \mathbf{c}$, where $\mathbf{w} = [w_{jp}^m]_{m=1,\dots,M;\, j=1,\dots,s_m;\, p=1,\dots,s_{m-1}}$ is the weights matrix, $\mathbf{g} = [g_j^m]_{m=1,\dots,M;\, j=1,\dots,s_m}$ is the slope parameters matrix, $\mathbf{c} = [c_j^m]_{m=1,\dots,M;\, j=1,\dots,s_m}$ is the bias parameters matrix and $\mathbf{a} = [a_{ij}^m]_{m=1,\dots,M;\, j=1,\dots,s_m;\, i=1,\dots,n}$ and $\mathbf{b} = [b_{ij}^m]_{m=1,\dots,M;\, j=1,\dots,s_m;\, i=0,\dots,n}$ are the filter parameters matrices. The adjustment of networks parameters can be carried out using stochastic approximation algorithm presented in the next section (Patan and Parisini, 2002*b*; Patan and Parisini, 2002*a*).

4. SIMULTANEOUS PERTURBATION STOCHASTIC APPROXIMATION

The considered class of algorithms is based on an approximation of the gradient of the loss function. The general form of Stochastic Approximation (SA) recursive procedure is as follows (Spall, 1992):

$$\hat{\boldsymbol{\theta}}_{k+1} = \hat{\boldsymbol{\theta}}_k - a_k \hat{\boldsymbol{g}}_k(\hat{\boldsymbol{\theta}}_k) \tag{2}$$

where $\hat{\boldsymbol{g}}_k(\hat{\boldsymbol{\theta}}_k)$ is the estimate of the gradient $\partial J/\partial \hat{\boldsymbol{\theta}}$ based on the measurements $y(\cdot)$ of the loss function. The essential part of this equation is the gradient approximation. The Simultaneous Perturbation Stochastic Approximation (SPSA) proposed in this paper as a leaning method, has all elements of $\hat{\boldsymbol{\theta}}$ randomly perturbed to obtain two measurements $y(\cdot)$, but each component $\hat{\boldsymbol{g}}_{ki}(\hat{\boldsymbol{\theta}}_k)$ is formed from a ratio involving the individual components in the perturbation vector and the difference in the two corresponding measurements. For two-sided simultaneous perturbation, estimation of gradient is obtained according to the formula (Spall, 1999):

$$\hat{g}_{ki}(\hat{\boldsymbol{\theta}}_k) = \frac{y(\hat{\boldsymbol{\theta}}_k + c_k\Delta_k) - y(\hat{\boldsymbol{\theta}}_k - c_k\Delta_k)}{2c_k\Delta_{ki}} \tag{3}$$

where the distribution of the user-specified p-dimensional random perturbation vector, $\Delta_k = (\Delta_{k1}, \Delta_{k2}, \ldots, \Delta_{kp})^T$ is independent and symmetrically distributed about 0 with finite inverse moments $E(|\Delta_{ki}|^{-1})$ for all k, i. One of the possible distributions that satisfies these conditions is the symmetric Bernoulli ± 1. Two commonly used distributions that not satisfy these conditions are the uniform and normal ones. The rich bibliography presents sufficient conditions for convergence of the SPSA ($\hat{\boldsymbol{\theta}}_k \to \boldsymbol{\theta}^*$ in the stochastic almost sure sense). However, the efficiency of the SPSA depends on the shape of the $J(\boldsymbol{\theta})$, the values of gain sequences $\{a_k\}$ and $\{c_k\}$ and distribution of the $\{\Delta_{ki}\}$. The choice of the gain sequences is critical to the performance of the algorithm. In the SPSA the gain sequences are calculated as follows (Spall, 1992):

$$a_k = \frac{a}{(A+k)^{\alpha}}, \quad c_k = \frac{c}{k^{\gamma}} \tag{4}$$

where a, c, A, α and γ are non-negative coefficients. To apply the SPSA to global optimisation, it is needed to use a stepwise (slowly decaying) sequence $\{c_k\}$ (Spall, 1999).

5. ACTUATOR FAULT DIAGNOSIS

In this section, the sugar evaporation station in the Lublin Sugar Factory (Poland) is presented (Patan and Korbicz, 2000). In a sugar factory, sucrose juice is extracted by diffusion. This juice is concentrated in a multiple-stage evaporator to produce a syrup. The liquor goes through a series of five stages of vapourisers, and in each passage its sucrose concentration increases. The first three sections are of Roberts type with a bottom heater-chamber, while the last two are of Wiegends type with a top heater-chamber. The sugar evaporation control should be performed in such a way that the energy used is minimised to achieve the required quality of the final product.

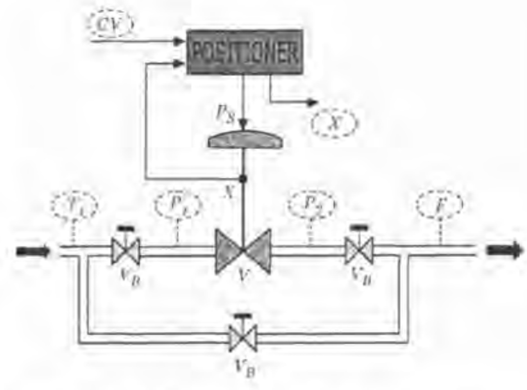

Fig. 1. Block scheme of the diagnosed actuator.

The actuator to be diagnosed is placed at the inlet to the first section of the evaporation station. The block scheme of this actuator is presented in Fig. 1. Considered actuator consists of three main parts: a control valve, a linear pneumatic servo-motor and a positioner (Bartyś and Kościelny, 2002). Description of the symbols and process variables is presented in Table 1. The control valve is typically used to allow, prevent and/or limit the flow of fluids. Changing the state of the control valve is accomplished by a servo-motor. The pneumatic servo-motor is a compressible fluid powered device in which the fluid acts upon the flexible diaphagm to provide linear motion of the servo-motor stem. The third part is the positioner applied to eliminate the control valve stem miss-positions developed by the external or internal sources such as frictions, pressure unbalance, hydrodynamic forces etc.

Taking into account structure of the actuator and expert knowledge about the technological process, one can define the following relations between variables:

Table 1. Description of the symbols

Symbol	Specification
V_B	hand driven bypass valve
V	control valve
P_1	pressure sensor (valve inlet)
P_2	pressure sensor (valve outlet)
F	process media flowmeter
X	piston rod displacement
P_s	pneumatic servo-motor supply
CV	control signal

- Servo-motor rod displacement

$$X = r_1(CV, P_1, P_2, T_1, X) \qquad (5)$$

- Flow through the actuator

$$F = r_2(X, P_1, P_2, T_1) \qquad (6)$$

where $r_1(\cdot)$ and $r_2(\cdot)$ are nonlinear functions. To identify (5) and (6) the dynamic neural network has been applied.

$$[X, F] = NN(P_1, P_2, T_1, CV) \qquad (7)$$

with four inputs and two outputs trained using the real process data recorded at sugar evaporator will be applied.

Furthermore, in collaboration with Sugar Factory there have been elaborated some faulty scenarios, which allow to evaluate and test the proposed fault detection and isolation system. These faulty data have been generated by manipulations on process variables within the control loop.

6. EXPERIMENTAL RESULTS

6.1 Modelling data

In order to carry out experiments, a real process data recorded during the sugar campaign 2001 in October and November was used. The input data was normalized to zero mean and unit standard deviation. In turn, the output data should be transformed taking into consideration the response range of the output neurons. For the hyperbolic tangent activation function, this range is [-1,1]. To perform such kind of transformation, the simple linear scalling can be used.

6.2 Model identification

The training process was carried out off-line using the SPSA algorithm for the dynamic network architecture. The neural faulty models were chosen using "test and trial" method. In the following Table 2 one can see the characteristic of the neural models.

6.3 Faulty scenarios

During experiments, the normal operation conditions f_0 and following faults have been considered:

f_1 – positioner supply pressure drop,
f_2 – unexpected pressure change across the valve,
f_3 – fully opened by-pass valve.

The first faulty scenario can be caused by many factors such as pressure supply station faults, an oversized system air consumption, air leading pipes breaks, etc. This is a rapidly developing fault. The physical interpretation of the second fault can be media pump station failures, an increased pipes resistance or external media leakages. The nature of this fault is rapidly developing as well. The last scenario can be caused by a valve corrosion or a seat sealing wear. This fault has an abrupt nature.

Table 2. Neural models for nominal conditions and examined faulty scenarios.

Fault	Structure	Filter order	Activation function
f_0	$N^2_{4,5,2}$	2	hyperbolic tangent
f_1	$N^2_{4,7,2}$	1	hyperbolic tangent
f_2	$N^2_{4,7,2}$	1	hyperbolic tangent
f_3	$N^2_{4,5,2}$	1	hyperbolic tangent

Fault f_1. The first fault has been simulated at 270 time step and has duration about 275 time steps. The neural network has been designed to identify actuator for fault f_1. During occurrence

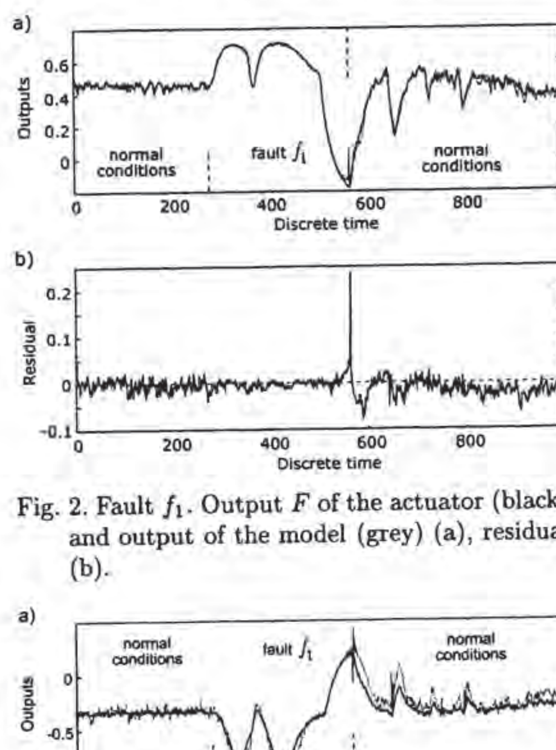

Fig. 2. Fault f_1. Output F of the actuator (black) and output of the model (grey) (a), residual (b).

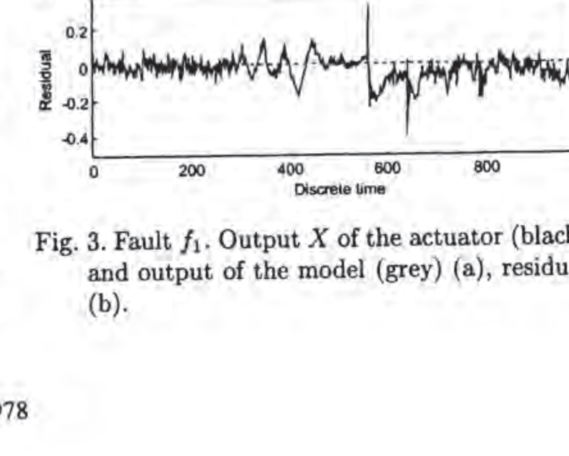

Fig. 3. Fault f_1. Output X of the actuator (black) and output of the model (grey) (a), residual (b).

of the f_1 residuals genereted by the model should be near zero, otherwise residuals should have a large values. As it can be seen in Figs. 2a and 3a, the model mimics the behaviour of the actuator very well in both faulty and normal operation conditions. In turn, in Fig. 2b and 3b one can see that the residuals are near zero regardless of the operation conditions. This means that this fault is undetectable by the neural model.

Fault f_2. The next faulty scenario has been simulated at 775 time step (pressure off) till 1395 time step (pressure on). Figure 4a presents the modelling of the flow through the actuator F, in the case of fault (775-1395 time steps), the output of the neural model (grey) follows the output of the actuators (black). The residual clearly shows that changes caused by fault are surely and immediately detected by the neural model (Fig. 4b).

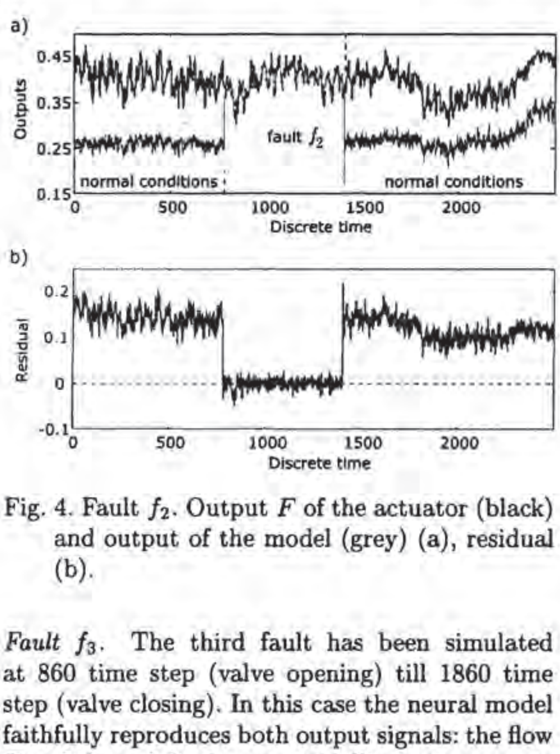

Fig. 4. Fault f_2. Output F of the actuator (black) and output of the model (grey) (a), residual (b).

Fault f_3. The third fault has been simulated at 860 time step (valve opening) till 1860 time step (valve closing). In this case the neural model faithfully reproduces both output signals: the flow F and the rod displacement X. Figure 5a presents the behaviour of the actuator (black) and neural model (grey) during normal operation conditions and during the fault (860–1860 time steps). The residual confirm that using neural model the fault f_3 can be detected and isolated distinctly (Fig. 5b).

7. PERFORMANCE INDEXES

In order to check the quality of proposed fault diagnosis system, a set of performance indexes was elaborated with the cooperation of project partners:

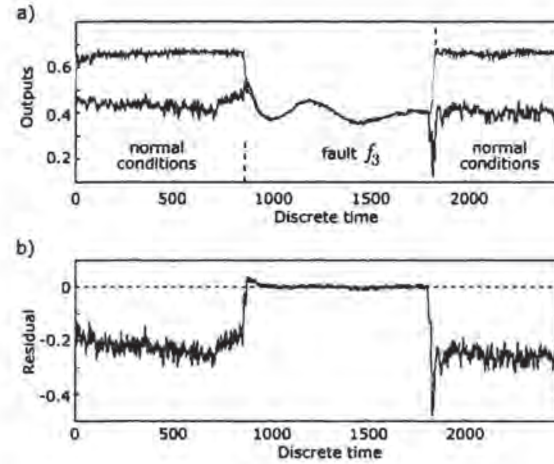

Fig. 5. Fault f_3. Output F of the actuator (black) and output of the model (grey) (a), residual (b).

- Detection time t_{dt}: period of time from the begin of fault start-up t_{from} to the moment of fault detection,
- False detection rate:

$$r_{fd} = \frac{\sum_i t^i_{fd}}{t_{from} - t_{on}} \tag{8}$$

  where t^i_{fd} is the period of ith false fault detection, t_{on} is the benchmark start up time,
- True detection rate:

$$r_{td} = \frac{\sum_i t^i_{td}}{t_{hor} - t_{from}} \tag{9}$$

  where t^i_{td} is the period of ith true fault detection, t_{hor} is the benchmark time horizon,
- Isolation time t_{it}: period of time from the begin of fault start-up t_{from} to the moment of fault isolation,
- False isolation rate:

$$r_{fi} = \frac{\sum_i t^i_{fi}}{t_{from} - t_{on}} \tag{10}$$

  where t^i_{fi} is the period of ith false fault isolation,
- True isolation rate:

$$r_{ti} = \frac{\sum_i t^i_{ti}}{t_{hor} - t_{from}} \tag{11}$$

  where t^i_{ti} is the period of ith true fault detection.

Decision making can be performed using simple threshold technique. In this work the fixed threshold with the value 0.05 is applied. The model designed for normal operation conditions is used to generate indexes concerning quality of fault detection and faults models are used to calculate indexes of the fault isolation quality. The results are shown in the Tables 3 and 4. Analyzing these results one can conclude that the faults f_1 and f_2 can be detected almost immediately. Somewhat worse are results for the fault f_3. Taking into

Table 3. Performance indexes derived for the variable F

Index	f_1	f_2	f_3
t_{dt}	7	2	75
r_{fd}	0.092	0,263	0,326
r_{td}	0,953	0,998	0,93
t_{it}	0	1	76
r_{fi}	0,978	0,002	0,008
r_{ti}	1	0,998	0,93

Table 4. Performance indexes derived for the variable X

Index	f_1	f_2	f_3
t_{dt}	13	3	46
r_{fd}	0,057	0,031	0,48
r_{td}	0,93	0,997	0,967
t_{it}	5	1	91
r_{fi}	0,432	0,296	0,007
r_{ti}	0,669	0,997	0,909

account the isolability, one can see that the fault f_1 is not isolable using the output F. The index r_{fi} is near one (in ideal case should be equal to 0). It means that the number of false alarms is very large. However, it is a chance to isolate this fault using output X. In this case the index $r_{fi} = 0.432$. Other two faults can be reliable isolated. It is necessary to notice, that results presented in Tables 3 and 4 strictly depend on a method of decision making. The other value of the fixed threshold can cause quite different results.

8. CONCLUSIONS

The experiments show that the considered dynamic neural networks can be easily and effectively applied to design model-based fault detection systems. The learning algorithm based on stochastic approximation makes it possible to perform very fast learning using noisy data. It is worth noting here, that it is impossible to model all potential system faults. The designer of the FDI systems can construct models based on available data. In many cases, only data for normal operation conditions is available and data for faulty scenarios have to be simulated. The paper presents also a set of performance indexes, which are very useful to check the quality of the FDI system.

REFERENCES

Ayoubi, M. (1994). Fault diagnosis with dynamic neural structure and application to a turbocharger. In: *Proc. Int. Symp. Fault Detection Supervision and Safety for Technical Processes, SAFEPROCESS'94, Espoo, Finland.* Vol. 2. pp. 618–623.

Bartyś, M. and J.M. Kościelny (2002). Application of fuzzy logic fault isolation methods for actuator diagnosis. In: *Proc. 15th IFAC Triennial World Congress, b'02. Barcelona, Spain, July 21–26.* Published on CD-ROM.

Calado, J.M.F., J. Korbicz, K. Patan, R.J. Patton and J.M.G. Sa da Costa (2001). Soft computing approaches to fault diagnosis for cynamic systems. *European Journal of Control* 7(2-3), 248–286.

Campolucci, P., A. Uncini, F. Piazza and B. D. Rao (1999). On-line learning algorithms for locally recurrent neural networks. *IEEE Transactions on Neural Network* **10**, 253–271.

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

Frank, P. M. and B. Köppen-Seliger (1997). New developments using AI in fault diagnosis. *Artificial Intelligence* **10**(1), 3–14.

Patan, K. (2000). Artificial Dynamic Neural Networks and Their Application in Modelling of Industrial Processes. Ph.D. dissertation.. Warsaw University of Technology, Faculty of Mechatronics. Warszawa.

Patan, K. and J. Korbicz (2000). Application of Dynamic Neural Networks in an Industrial Plant. In: *Proc. Int. Symp. Fault Detection Supervision and Safety for Technical Processes, SAFEPROCESS 2000, Budapest, Hungary.* pp. 186–191.

Patan, K. and T. Parisini (2002*a*). Stochastic approaches to dynamic neural network training. Actuator fault diagnosis study. In: *Proc. 15th IFAC Triennial World Congress, b'02. Barcelona, Spain, July 21–26.* Published on CD-ROM.

Patan, K. and T. Parisini (2002*b*). Stochastic learning methodsfor dynamic neural networks: simulated and real-data comparisons. In: *Proc. 2002 American Control Conference, ACC'02. Anchorage, Alaska, USA, May 8–10.* pp. 2577–2582.

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

Spall, J.C. (1992). Multivariate stochastic aproximation using a simultaneous perturbation gradient approximation. *IEEE Trans. Automatic Control* (37), 332–341.

Spall, J.C. (1999). Stochastic optimization, stochastic approximation and simulated annealing. In: *Encyclopedia of Electrical and Electronics Engineering* (J.G. Webster, (Ed.). John Wiley & Sons. New York.

Tsoi, A. Ch. and A. D. Back (1994). Locally recurrent globally feedforward networks: A critical review of architectures. *IEEE Transactions on Neural Network* **5**, 229–239.