Memoirs of the Faculty of Engineering, Okayama University, Vol.27, No. 1, pp.33-44, November 1992

Process Fault Diagnosis using Neural Networks and Fault Tree Analysis Information

Yukiyasu SHIMADA[1], Kazuhiko SUZUKI[1] and Hayatoshi SAYAMA[1]

(Received September 30 , 1992)

SYNOPSIS

Neural nets have recently become the focus of much attention, largely because of their wide range of complex and nonlinear problems. This paper presents a new integrated approach using neural networks for diagnosing process failures. The fault propagation in process is modeled by causal relationships from the fault tree and its minimal cut sets. The measurement patterns required for training and testing the neural network were obtained from fault propagation model. The network is able to diagnose even in the presence of malfunction of certain sensors. We demonstrate via a nitric acid cooler process how the neural network can learn and successfully diagnose the faults.

1. INTRODUCTION

Neural nets or artificial neural networks have been rapidly used in date analysis, pattern recognition and decision making, and are expected to have a significant impact in many technological and business areas. The application of neural network techniques to process fault detection and diagnosis has received much attention in recent years. A number of approaches based on neural network have been proposed for fault diagnosis. In almost these methods, the measurement patterns required for training the neural network were obtained from simulation program that model the behavior of the plants. Fault tree analysis, which has been used as a method of system reliability/safety analysis, provides a systematic procedure for identifying failures within a system. This paper presents an approach for process fault diagnosis using neural networks and fault tree analysis

[1]Department of Mechanical Engineering

information. The fault propagation in process is modeled from causal relationships of the fault tree and its minimal cut sets. The neural network was trained by the use of patterns of malfunction obtained from fault propagation model. A nitric acid cooler process is used to demonstrate the effectiveness of the proposed method.

2. NEURAL NETWORKS

Rapid advances in neuroscience and in computer science are arousing renewed interests in neural networks as potentially new problem-solving architectures. Neural networks are dynamic systems composed of highly interconnected layers of simple neuron like processing elements[1~4]. These layers are categorized as input layers where patterns are presented to the network and as output layers which contain the response to a given input. Furthermore, they may also contain intermediate layers or hidden layers. Fig.1 presents an example of a typical neural network architecture. The circles denote neurons arranged in three layer neuron is connected to each input and output neuron. The arcs denote information flow channels between the neurons, called connections. The boxes are neurons that store inputs to the net. Fig.2 shows the j-th neuron used for the network.

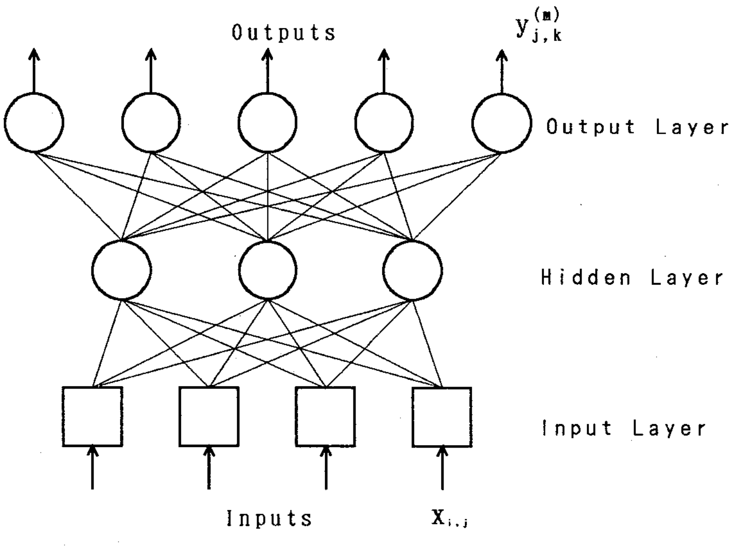

Fig.1 Three layer feedforward neural net.

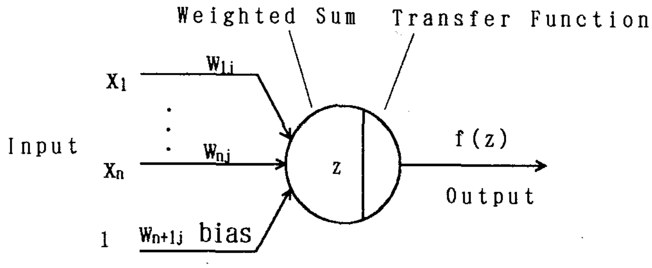

Fig.2 Input-output relation for j-th neuron.

Neural network operations consist of a learning or training phase, a recall phase, and a generalization phase. During the learning phase, the network is repeatedly presented with a set of input-output patterns. Learning is accomplished by a general rule which dynamically modifies the weights of all interconnections in an attempt to generate the desired output pattern for each presented input pattern. After the learning is complete, the network operates in a recall phase where it generates responses to input patterns used in training. It also operates in a generalization phase where it generates responses to similar or novel input patterns.

Neural network computations are collectively performed by the entire network with knowledge represented in the connection weights between processing elements. Consequently, the collective operations result in a high degree of parallel computation which enables the network to solve complex problems rapidly. In addition, the distributed representations lead to greater fault tolerance and to graceful degradation when problems are encountered beyond its range of experience. Furthermore, other beneficial attributes include the ability to adjust dynamically to environmental changes, to perform useful generalizations from specific examples, and to recognize invariances from complex high-dimensional data.

Thus, neural networks appear to be applicable to chemical process engineering problems requiring pattern recognition and pattern classification [5~8]. These areas include fault diagnosis and detection, process design, process control, and process simulation. Recently, researchers have begun to look at applications such as interpreting biosensor data[9], and for single fault diagnosis of three continuous-stirred-tank reactors connected in series[10].

3. FAULT PROPAGATION MODEL

3.1 Fault Tree for The Nitric Acid Cooler Process

Fault trees are widely used for the representation and acquisition of knowledge from the process operation and control system. A fault tree, graphic representation of the failure logic of a system, shows combinations and sequences of failures which can lead to a failure condition under consideration (the top event). Through the analysis of the fault tree, the cause of the top event can be determined as "minimal cut sets"(i.e., as sets of events that are top for the specific event to occur).

The process used in the study is the nitric acid cooler with temperature feedback and pump-shutdown feedforward loops illustrated in Fig.3[11]. The function of this process is to cool a nitric acid stream before reacting with Benzene to form Nitrobenzene. The undesired event for the system is high temperature in the nitric acid reactor feed since this could cause a reactor runaway. In this study, T4 high(high temperature in the effluent stream of nitric acid cooler) is selected as the top event.

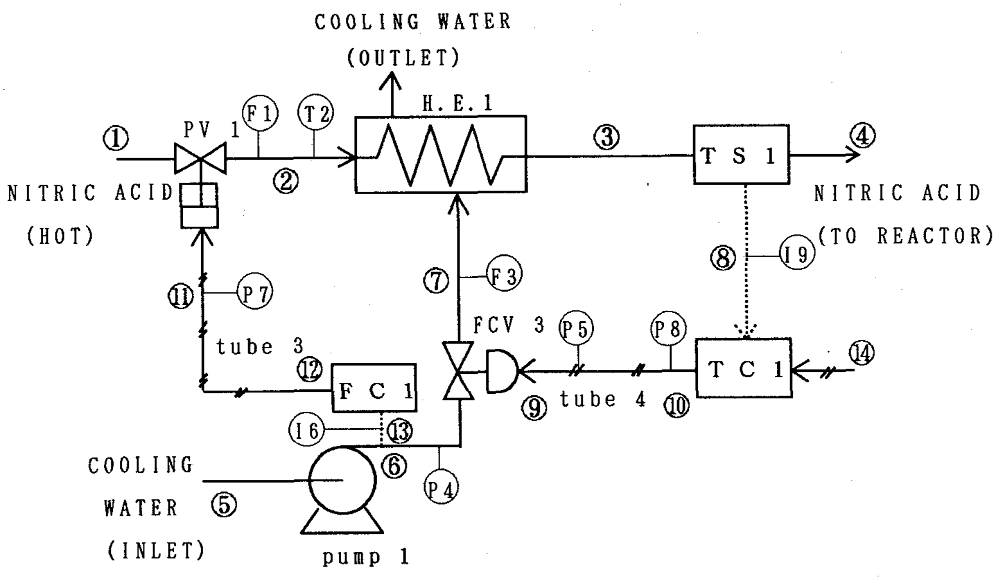

Fig.3 Nitric acid cooler process and sensor allocation for diagnosis.

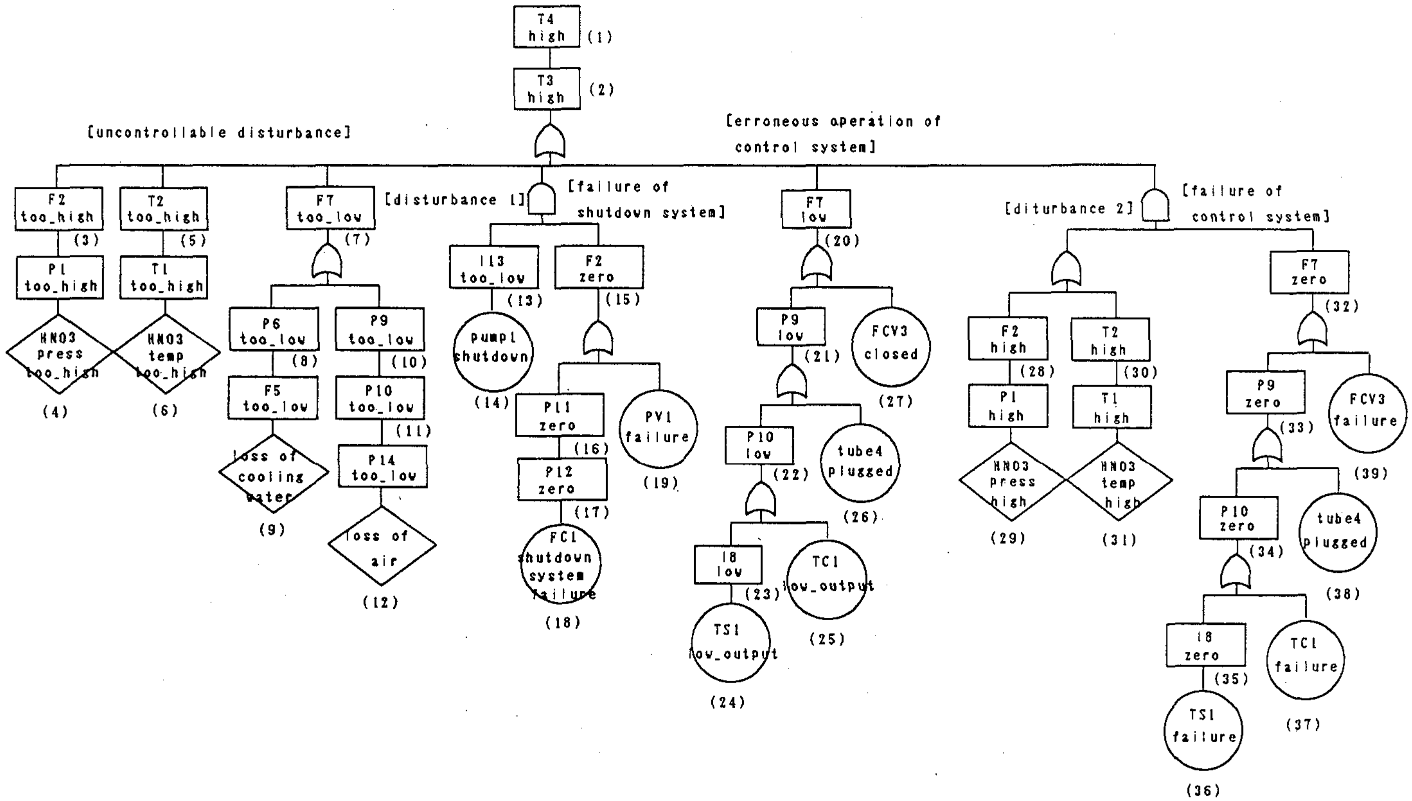

Fig.4 Fault tree of nitric acid cooler process for T4 high.

Fig.4 shows the fault tree of the process for the top event(T4 high). The fault tree consists of four sub trees, (1) uncontrollable disturbance, (2) disturbance and failure of shutdown system, (3) erroneous operation of control system, (4) disturbance and failure of control system.

A set of discrete values such as "too high, high, normal, low, too low, zero" are used to describe deviations in process variables. "Too high, too low" denotes a large deviation in the high, low direction respectively which can not be corrected by a control loop. "High, low" denotes a deviation which can be corrected by a control loop. "Zero" denotes no signal to cancel the effect of deviations in the process variables. And "normal" denotes normal state.

3.2 Sensor Allocation and Patterns of Abnormality in Process Variables[12]

Fault trees are used for the representation and acquisition of knowledge from the process operation and control system. The fault propagation in processes is modeled by causal relationships from the fault trees and their minimal cut sets. The method can determine the minimum number of sensors and the monitoring points to detect fault events in the processes. Knowledge engineering techniques are applied to the automated synthesis of fault tree, the modeling of fault propagation and the allocation of sensors. Table 1 shows eighteen minimal cut sets of the fault tree. The tree is transformed into CE-Table(Cut set Event Table) form illustrated in Table 2. CE-Table presents the tree events associated with minimal cut sets. The minimum number of sensors to detect fault events can be determined by using information from the fault propagation in processes(CE-Table). Fig.3 shows the allocation of nine sensors show in circles, such as flow sensor F1 for F2 too high and F2 zero and F2 high, temperature sensor T2 for T2 too high and T2 high, flow sensor F3 for F7 too low and F7 low and F7 zero, and so on.

Table 3 shows a list of the minimal cut sets and the associated sensor readings. The process has eighteen possible patterns of malfunctions, which can be diagnosed by the use of nine sensor readings. In table 3, +2 and -2 denote too high, too low (a large deviation), +1 and -1 denote high, low (a deviation), -(minus sign) denotes zero (no signal), and •(dot) denotes normal (normal state), respectively.

Table 1 Minimal cut sets of fault tree.

Minimal cut sets	Minimal cut sets classification
cutset(1,[4]). cutset(2,[6]). cutset(3,[9]). cutset(4,[12]).	(1) Uncontrollable disturbance
cutset(5,[14,18]). cutset(6,[14,19]).	(2) Pump off and failure of shutdown system
cutset(7,[24]). cutset(8,[25]). cutset(9,[26]). cutset(10,[27).	(3) Erroneous operation of control system
cutset(11,[29,36]). cutset(12,[29,37]). cutset(13,[29,38]). cutset(14,[29,39]). cutset(15,[31,36]). cutset(16,[31,37]). cutset(17,[31,38]). cutset(18,[31,39]).	(4) Disturbance and failure of control system

Table 2 Cut set Event table (CE-Table).

cut set No.	Number of event																																						
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39
1	*	*	*	*																																			
2	*	*			*	*																																	
3	*	*					*	*	*																														
4	*	*					*			*	*	*																											
5	*	*											*	*	*	*	*	*																					
6	*	*											*	*	*				*																				
7	*	*																		*	*	*	*	*															
8	*	*																		*	*	*			*														
9	*	*																		*	*					*													
10	*	*																		*							*												
11	*	*																										*	*			*	*	*	*	*			
12	*	*																										*	*			*	*	*			*		
13	*	*																										*	*			*	*					*	
14	*	*																										*	*			*							*
15	*	*																												*	*	*	*	*	*	*			
16	*	*																												*	*	*	*	*			*		
17	*	*																												*	*	*	*					*	
18	*	*																												*	*	*							*

Table 3 Pattern of abnormality used for training.

Minimal cut sets	Sensor readings F1	T2	F3	P4	P5	I6	P7	P8	I9
1 [4]	+2	•	+1	•	+1	-	-	+1	+1
2 [6]	•	+2	+1	•	+1	-	-	+1	+1
3 [9]	•	•	-2	-2	+1	-	-	+1	+1
4 [12]	•	•	-2	+1	-2	-	-	-2	+1
5 [14,18]	•	•	-2	-2	+1	-2	-	+1	+1
6 [14,19]	•	•	-2	-2	+1	-2	-2	+1	+1
7 [24]	•	•	-1	+1	-1	-	-	-1	-1
8 [25]	•	•	-1	+1	-1	-	-	-1	+1
9 [26]	•	•	-1	+1	-1	-	-	+1	+1
10 [27]	•	•	-1	+1	+1	-	-	+1	+1
11 [29,36]	+1	•	-	-	-	-	-	-	-
12 [29,37]	+1	•	-	-	-	-	-	-	+1
13 [29,38]	+1	•	-	-	-	-	-	+1	+1
14 [29,39]	+1	•	-	-	+1	-	-	+1	+1
15 [31,36]	•	+1	-	-	-	-	-	-	-
16 [31,37]	•	+1	-	-	-	-	-	-	+1
17 [31,38]	•	+1	-	-	-	-	-	+1	+1
18 [31,39]	•	+1	-	-	+1	-	-	+1	+1

4. NEURAL NETWORK USED FOR FAULT DIAGNOSIS

4.1 Networks Structure and Learning

Fig.5 shows a sample neural net with eighteen neurons in the hidden layer. The input nodes, which receive their inputs from sensors, are connected via a set of hidden nodes to the output nodes, which give diagnosis of fault. The neural net consists of nine input nodes corresponding to the nine sensor readings, and eighteen output nodes corresponding to the eighteen possible minimal cut sets listed in Table 1. The neural net was trained by the use of the eighteen measurement patterns in Table 3. A set of inputs to the neural net are sensor readings and the outputs of the network are the minimal cut sets. As shown in Table 4, input values of 0.9 for too high, 0.7 for high, 0.5 for normal, 0.3 for low, 0.1 for too low and 0.0 for zero are chosen for the sensor readings. Minimal cut set is defined to be equal 1.0 for the occurrence of the cut set and equal to 0.0 for the other. 50,000 iterations are required to get solutions by the use of Neural Works Professional II. Table 5 shows the calculated output of the neural net.

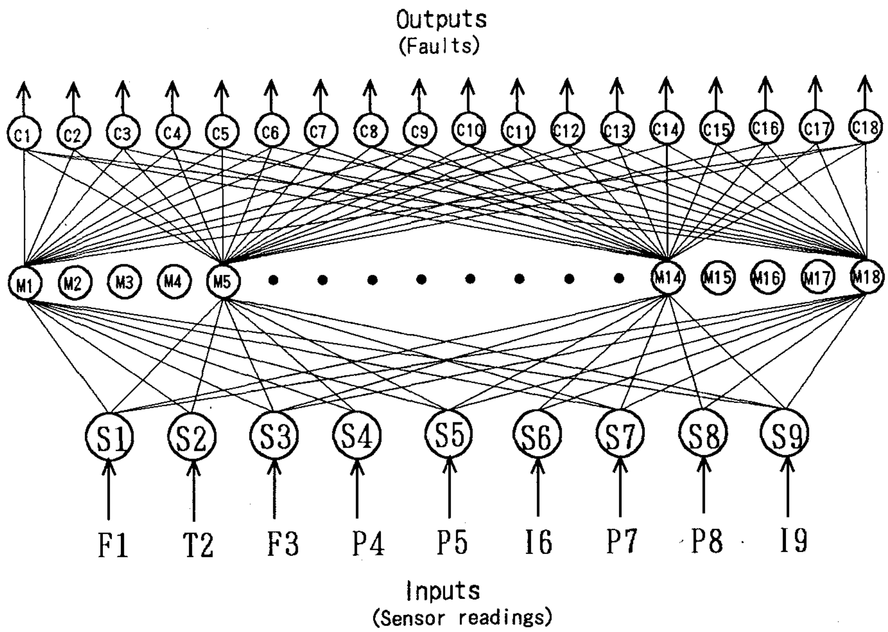

Fig.5 Neural network for case study.

Table 4 Input and output values used for training networks.

Sensor readings	Input values	Minimal cut set Ci	Desired output values
too high (+2)	0.9	Ci:occurrence	1
high (+1)	0.7		
normal (•)	0.5		
low (-1)	0.3	Ci:no occurrence	0
too low (-2)	0.1		
zero (-)	0.0		

4.2 Case Study

Table 6 shows the results of the tested cases. To test the ability of the network for fresh input patterns, sensor readings patterns not used in the training process were presented to the trained network. As an initial test of the recognition capability of the neural net, we used the pattern of inputs that represented a normally operating system as shown in case 1(Case 1-1: the process normally operated, Case 1-2 and Case 1-3: the control loop acts to cancel the effect of disturbance). In this case, all sensor used for diagnosis were assumed to function properly. As shown in Table 6, the output for each minimal cut set was all approximately zero and any minimal cut set would not occur.

As the second example, fault diagnosis in the presence of malfunction of sensors were investigated in Case 2. By a malfunction of sensors, we mean that particular sensor fails to detect the process value or control signal when the actual value of that parameter might be abnormal. The results in case of flow rate in the presence of malfunction of certain sensors to detect fault events, the neural net was able to predict correct minimal cut set in more than 50% of tested case. And in 70% of tested case, the neural net could identify process malfunction such as occurrence of uncontrollable disturbance, control loop failure and shutdown system failure.

5. CONCLUSION

In this paper, we have proposed an approach based on neural network to process fault diagnosis. As demonstrated, neural networks are able to acquire diagnostic knowledge from fault propagation model based on fault trees. Knowledge engineering techniques are applied to the automated synthesis of fault trees, the modeling of fault propagation. This research demonstrates the feasibility for future developments in the application of neural networks to chemical process fault detection and diagnosis. The knowledge-based system applied to the modeling of fault propagation can greatly simplify the knowledge acquisition process.

Table 5 Calculated outputs of the network.

	Input (Sensor readings)									Output (Faults)																	
	F1	T2	F3	P4	P5	I6	P7	P8	I9	C1	C2	C3	C4	C5	C6	C7	C8	C9	C10	C11	C12	C13	C14	C15	C16	C17	C18
1	0.9	0.5	0.7	0.5	0.7	0	0	0.7	0.7	0.975	0	0	0	0	0	0.001	0	0	0	0.004	0	0	0.002	0	0	0	0
2	0.5	0.9	0.7	0.5	0.7	0	0	0.7	0.7	0	0.971	0	0	0	0	0	0	0	0	0	0	0	0	0.004	0	0	0.003
3	0.5	0.5	0.1	0.1	0.7	0	0	0.7	0.7	0	0	0.955	0	0.046	0	0	0.002	0	0.007	0	0	0	0.017	0	0	0	0.017
4	0.5	0.5	0.1	0.7	0.1	0	0	0.1	0.7	0	0	0	0.983	0	0	0.019	0.001	0	0	0.001	0.006	0	0	0	0.001	0	0
5	0.5	0.5	0.1	0.1	0.7	0.1	0	0.7	0.7	0	0	0.024	0	0.886	0.056	0	0.001	0.003	0	0	0	0.001	0.001	0	0	0	0
6	0.5	0.5	0.1	0.1	0.7	0.1	0.1	0.7	0.7	0	0	0	0	0.045	0.958	0	0	0.001	0	0	0	0.001	0	0	0	0.004	0
7	0.5	0.5	0.3	0.7	0.3	0	0	0.3	0.3	0	0	0	0	0	0	0.996	0	0.003	0.002	0	0	0	0	0	0	0	0
8	0.5	0.5	0.3	0.7	0.3	0	0	0.3	0.7	0	0	0	0.013	0	0	0	0.974	0.014	0.002	0	0	0	0	0	0	0	0
9	0.5	0.5	0.3	0.7	0.3	0	0	0.7	0.7	0	0	0	0	0.017	0	0.013	0	0.976	0.012	0	0	0.007	0	0	0	0.001	0
10	0.5	0.5	0.3	0.7	0.7	0	0	0.7	0.7	0	0	0.012	0	0.001	0	0.006	0	0.008	0.980	0	0	0	0	0	0	0	0
11	0.7	0.5	0	0	0	0	0	0	0	0.004	0	0	0	0	0	0	0	0	0	0.981	0.003	0	0.003	0.013	0	0	0
12	0.7	0.5	0	0	0	0	0	0	0.7	0.001	0	0	0.004	0	0	0	0.001	0	0	0.009	0.981	0.009	0.006	0	0.010	0	0
13	0.7	0.5	0	0	0	0	0	0.7	0.7	0	0	0	0	0.001	0.005	0	0	0.003	0	0	0.006	0.977	0.006	0	0	0.010	0
14	0.7	0.5	0	0	0.7	0	0	0.7	0.7	0.014	0	0.013	0	0	0	0	0.002	0	0	0.014	0.011	0.012	0.986	0	0	0	0.001
15	0.5	0.7	0	0	0	0	0	0	0	0	0.003	0	0	0	0	0	0	0	0	0.012	0	0	0	0.981	0.005	0	0
16	0.5	0.7	0	0	0	0	0	0	0.7	0	0.001	0	0.002	0	0	0	0.001	0	0	0	0.011	0	0	0.011	0.981	0.008	0.005
17	0.5	0.7	0	0	0	0	0	0.7	0.7	0	0	0	0	0	0.019	0	0	0.003	0	0	0	0.015	0	0	0.006	0.981	0.003
18	0.5	0.7	0	0	0.7	0	0	0.7	0.7	0	0.015	0.012	0	0	0	0	0.002	0	0	0	0	0	0.001	0.005	0.011	0.012	0.989

Table 6 Case study results (no signal from F3).

No. of case	cut set with faulty signal	Input (Sensor readings)									Output (Faults)																		Predicted faults minimal cut sets
		F1	T2	F3	P4	P5	I6	P7	P8	I9	C1	C2	C3	C4	C5	C6	C7	C8	C9	C10	C11	C12	C13	C14	C15	C16	C17	C18	
1-1	normal state	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0	0	0	0	0	0.013	0	0	0	0	0	0	0	0	0	0	0	0	–
1-2		0.7	0.5	0.7	0.5	0.5	0	0	0.7	0.7	0.033	0	0	0	0	0	0.003	0	0.003	0	0	0	0	0	0	0	0	0	–
1-3		0.5	0.7	0.7	0.5	0.7	0	0	0.7	0.7	0	0.024	0	0	0	0	0.001	0	0	0.004	0	0	0	0	0	0	0	0	–
2-1	C1	0.9	0.5	0	0.5	0.7	0	0	0.7	0.7	0.286	0	0.006	0	0	0	0	0.001	0	0	0.032	0.004	0.013	0.654	0	0	0	0	C1 C14
2-2	C2	0.5	0.9	0	0.5	0.7	0	0	0.7	0.7	0	0.501	0.044	0	0	0	0	0.001	0	0.001	0	0	0	0	0.008	0.005	0.016	0.918	C2 C18
2-3	C3	0.5	0.5	0	0.1	0.7	0	0	0.7	0.7	0	0	0.809	0	0.019	0	0	0.004	0	0.010	0	0	0	0.045	0	0	0	0.046	C3
2-4	C4	0.5	0.5	0	0.7	0.1	0	0	0.1	0.7	0	0	0	0.988	0	0	0.002	0.001	0	0	0.001	0.028	0.001	0	0	0.005	0	0	C4
2-5	C5	0.5	0.5	0	0.1	0.7	0.1	0	0.7	0.7	0	0	0.008	0	0.824	0.027	0	0.002	0.004	0.001	0	0	0.002	0.002	0	0	0.001	0.001	C5
2-6	C6	0.5	0.5	0	0.1	0.7	0.1	0.1	0.7	0.7	0	0	0	0	0.026	0.890	0	0	0.002	0	0	0	0.003	0	0	0	0.012	0	C6
2-7	C7	0.5	0.5	0	0.7	0.3	0	0	0.3	0.3	0	0	0	0.037	0	0	0.511	0	0.001	0	0.005	0	0	0	0.003	0	0	0	C7
2-8	C8	0.5	0.5	0	0.7	0.3	0	0	0.3	0.7	0	0	0	0.608	0	0	0.003	0.002	0.010	0	0	0.001	0.003	0	0	0	0	0	C4
2-9	C9	0.5	-0.5	0	0.7	0.3	0	0	0.7	0.7	0	0	0	0.001	0.129	0.021	0	0.001	0.982	0.009	0	0	0.189	0	0	0	0.018	0	C9
2-10	C10	0.5	0.5	0	0.7	0.7	0	0	0.7	0.7	0	0	0.397	0	0.065	0	0	0.001	0.030	0.986	0	0	0	0	0	0	0	0	C10

Literature Cited

[1] Amari S.A.: A Mathematical Approach to Neural Systems, Systems Neuroscience J.Metzler,ed., 67 Academic Press, New York(1977).

[2] Feldman J.A. and D.H.Ballard: Connectionist Models and their Properties, Cognitive Sci., 6, (1982) p.205.

[3] Kohonen T.: Self-Organization and Associative Memory, Springer-Verlag, Berlin(1984).

[4] Rumelhart D.E., G.E.Hinton and J.L.McClelland: A General Framework for Parallel Distributed Processing, Parallel Distributed Processing: Explorations in the Microstructure of Cognition: I.Foundation, D.E.Rumelhart and J.L.McClelland, eds., MIT Press, Cambridge, MA(1986a).

[5] N.Bhat and T.J.McAvoy: Use of Neural nets for dynamic modeling and control of chemical process systems, Comput.Chem.Engng., 14-4/5, (1990) p.573.

[6] L.H.Unger, B.A.Powell and S.N.Kamens: Adaptive network methodology for fault diagnosis and process control, Comput.Chem.Engng., 14-4/5, (1990) p.561.

[7] V.Venkatasubramanian and K.Chan: A neural network methodology for process fault diagnosis, AIChem J., 35-12, (1989) p.1993.

[8] V.Venkatasubramanian, R.Vaidyanathan and Y.Yamamoto: Process fault detection and diagnosis using neural networks -1. Steady-state processes, Comput.Chem.Engng., 14-7, (1990) p.699.

[9] McAvoy T.J., N.S.Wang, S.Naida, N.Bhat, J.Gunter and M.Simmons: Interpreting Biosensor Data via Back-propagation, IJCNN Int. Joint Conf. on Neural Networks, 1, 227, Washington DC(June,1989).

[10] J.C.Hoskins and D.M.Himmelblau: Artificial neural network models of knowledge representation in chemical engineering, Comput.Chem.Engng., 12-9/10, (1988) p.881.

[11] S.A.Lapp and G.J.Powers: Computer-aided Synthesis of Fault-Trees, IEEE Trans. on Reliab., R-26-2, (1977) p.2.

[12] H.Sayama et al.: Study on Fault Diagnostic Method for Process Using Fault Trees (1st Report, Sensor Allocation for Fault Diagnosis), Nippon Setsubikanri Gakkaishi, 1-2, (1990) p.13. (2nd Report, Development of Fault diagnostic Expert System), Nippon Setsubikanri Gakkaishi, 2-1, (1990) p.11. (in Japanese).