12

ELSEVIER

Journal of Biotechnology 34 (1994) 109–118

journal of biotechnology

A knowledge based system using fuzzy inference for supervisory control of bioprocesses

Camilla von Numers [a,1], Mikio Nakajima [a], Terhi Siimes [b], Hajime Asama [a], Pekka Linko [b,*], Isao Endo [a]

[a] *Chemical Engineering Laboratory, The Institute of Physical and Chemical Research (RIKEN), Wako-shi, Saitama 351-01 Japan*
[b] *Laboratory of Biotechnology and Food Engineering, Helsinki University of Technology, FIN-02150 Espoo, Finland*

(Received 3 August 1993; accepted 12 January 1994)

Abstract

In this paper, a rule-based, real-time knowledge based system for bioprocess fault diagnosis and control is described. The system was designed to generate on-line advice for the operators and to supervise automatic control of bioprocesses, using biotechnical production of lactic acid as an example process. It consists of a real-time data acquisition and data processing system linked to a fuzzy expert system written in Smalltalk V/Mac. The expert knowledge was expressed in the form of a rule-based knowledge network, fuzzy membership functions and control strategies. The fuzzy expert system carries out on-line fault diagnosing on the basis of filtered specific rates calculated from process variable measurements, and provides suitable countermeasures to recover the process. Fault diagnosis was realized both by backward and forward chaining procedures. The system was constructed to allow three different control strategies (given here in Smalltalk syntax), change of *Setpoint*, *FuzzyAnswer* for each discovered fault, employing the fuzzy mean defuzzification method, and linguistic *Advice* to the operator. The system was successfully tested on-line with a laboratory scale process.

Key words: Fuzzy inference; Knowledge based system; Bioprocess control; Lactic acid cultivation

1. Introduction

In the field of bioprocess control, there are many kinds of objectives such as maintaining certain environmental conditions to optimize the process. There are methods which can serve these various objectives, but their applications to bioprocesses may cause some problems owing to considerable non-linearities, time varying parameters, and a number of disturbances in the process. Thus, in bioprocess control one of the major problems is real-time estimation of the system

* C[illegible]sponding author.

Ab[illegible]iations: A, B = fuzzy sets; NB = negative big; NM = negative medium; NS = negative small; P = product concentration (g l^{-1}); PB = positive big; PM = positive medium; PS = positive small; S = substrate concentration (g l^{-1}); U = universe of discourse; u_i = an element of U; X = biomass dry weight (g l^{-1}); ZE = zero; ν = specific substrate consumption rate (g s^{-1}); μ = specific growth rate (s^{-1}); $\mu_A(u_i)$ = membership function of fuzzy set A; π = specific product formation rate (g s^{-1}); σ = standard deviation; $\cup$ = union; $\cap$ = intersection. Text in italics refers to Smalltalk code.

[1] Visiting scientist from HUT at RIKEN. Present address: ValioData, Inc., PB 229 SF00101 Helsinki, Finland.

SSDI 0168-1656(94)00007-Y

state. This is often difficult owing to the lack of reliable sensors, system complexity, model uncertainties and parameter variations. Ideally, the state of the process should be determined by means of measurements, and the knowledge of the process behaviour expressed in terms of a model. Since bioprocesses often cannot be completely described by a single mathematical model, the experience gained by working with actual processes provides valuable information. However, human operators require time-consuming training. Further, reserve personnel is often needed. The operators vary in reliability, consistency and emotions when dealing with the problems involved. The use of human operators is often unsuitable under hazardous circumstances. In order to overcome all such problems in a bioprocess control, knowledge based systems have been introduced (Linko, 1988).

Knowledge based systems are constructed for emulating the reasoning process of a human operator. The knowledge obtained from experienced operators can be expressed as a set of rules or other form of heuristics. We have previously described a knowledge based fuzzy expert system developed on the basis of a shell called BIOTALK (Aarts et al., 1990). In this expert system shell, the knowledge base employs 'If, then' rules in the form of a knowledge network. According to Stephanopoulos (1987), rule-based expert systems are favoured by chemical and biochemical engineers to process verbally formulated knowledge. By collecting the knowledge from experts and cultivation processes into a knowledge network a number of example applications were developed for bioprocess fault diagnosis and control (Pokkinen et al., 1992; Siimes et al., 1992a,b). The goal of the present work was to incorporate online control ability to the expert system shell and to automate some of the supervisory tasks currently performed by expert operators according to the concept described by Endo et al. (1989).

2. System description

A 30-l jar bioreactor equipped with a sampling unit and an on-line laser turbidity electrode was employed. The on-line sampling unit was connected to an HPLC analyzer, and a sample of cell-free medium was injected every 20 min to an ion exchanger column in order to obtain actual values of product and substrate concentrations (Endo et al., 1985). The sterilizable on-line laser turbidity electrode was used to measure the optical density to represent cell mass concentration (Nagamune et al., 1985).

The bioreactor was monitored by a measurement and control system BIOACS (Bio Advanced Control System), installed in a Fujitsu A-240Σ workstation used as a process computer (Endo et al., 1989). The workstation performed real-time data acquisition by collecting data from the process controller and from the HPLC unit, and carried out real-time data processing by filtering the actual values of substrate (S), product (P) and biomass (X) concentrations, and by calculating the respective specific rates (ν), (π) and (μ) (Pokkinen et al., 1992). A conventional fourth-order delay filter with a time constant of approx. 26 min was used. The calculated specific rates along with the measured values of the process variables (temperature, pH, agitation rate, and substrate product and cell mass concentrations) were transferred to the expert system implemented in a Macintosh IIci computer. An inference engine, a database for standard variable time-courses, fuzzy sets for the process and state variables, and a knowledge network representing 'If, then' rules were incorporated into the Smalltalk/V Mac-based object oriented fuzzy expert system shell. The inference was based both on backward and forward chaining described in detail below.

The knowledge based system performed online, real-time diagnosing and control. This was made possible by multiplex communication between the system and the process computer through RS232C communication line as illustrated in Fig 1.

3. Knowledge expression

The system shell was constructed in the object oriented Smalltalk V/Mac programming environ-

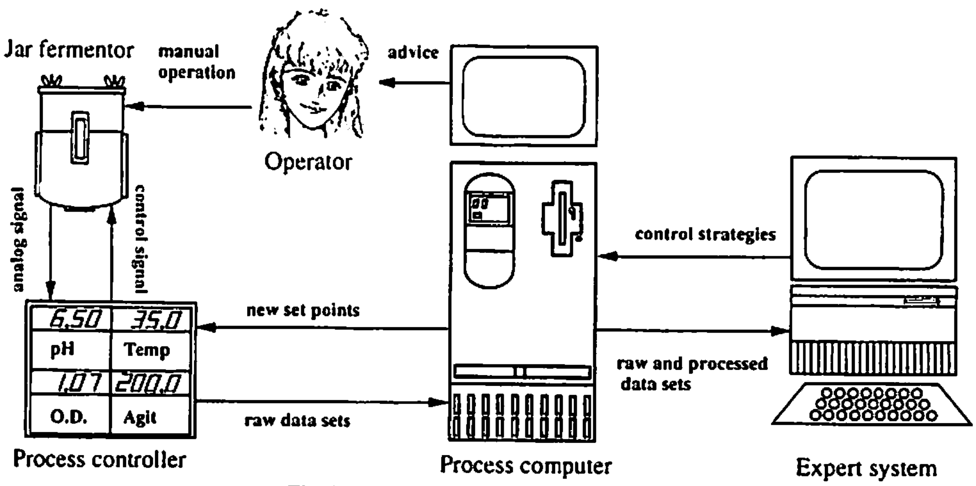

Fig. 1. A schematic diagram of the system.

ment. In the system, expert knowledge was expressed in a form of a rule-based frame network named knowledge network (Aarts et al., 1990). The network was constructed in a modular form in order to facilitate easy modifications and additions. An example of adding a new sub network to an existing network is shown in Fig. 2. Each frame consisted of a name representing a fact in the bioprocess, and of several slots indicating either stationary or dynamic values of knowledge. In the present work the knowledge network was constructed of nodes that have 'If, then' correlation. A number of different nodes such as a *StartNode, EndNode, OrNode, AndNode, ForwardOrNode, ForwardAndNode, ActionNode* and *ForwardActionNode* were defined in Smalltalk syntax as separate classes for various purposes. The knowledge network components are schematically shown in Fig. 3. The *StartNodes* represent the original causes for malfunctions, *EndNodes* the observable faults in the bioprocess, and *OrNode, AndNode, ForwardOrNode* and *ForwardAndNode* the fuzzy operations in question for backward and forward chaining procedures, respectively. *ActionNodes* and *ForwardActionNodes* represent the control actions in the case of backward chaining and forward chaining procedures to recover the process.

Fuzzy membership functions were integrated into the network and used to handle uncertainties both in the measured variables and in the causal relations between the nodes. For the *StartNodes* fuzzy sets were static, but for the *EndNodes* fuzzy sets could be dynamically changed with time. The fuzzy regions were defined on the basis of the standard deviations derived from experiments

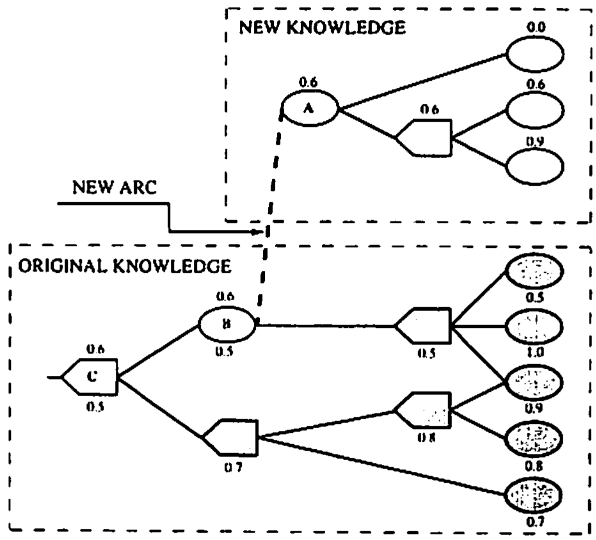

Fig. 2. Adding new knowledge to the network. *TruthValues* written below the nodes are calculated according to the original network and *TruthValues* on top of the nodes are calculated after adding new knowledge. The *TruthValues* of the shadowed nodes remain unchanged after the addition of the new subnetwork.

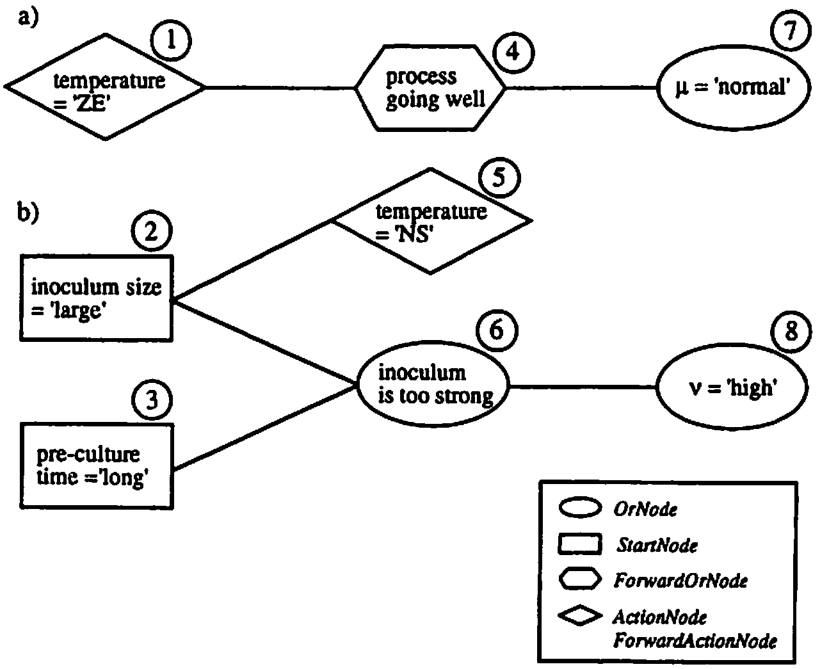

Fig. 3. Two example rule chains of the current knowledge network for lactic acid cultivation, representing the following rules. (a) Forward chaining: 7 → 4: If 'μ = 'normal'' then 'process is going well'; 4 → 1: If 'process is going well' then 'temperature = '*ZE*''. (b) Backward chaining: 8 → 6: If 'ν = 'high'' then 'inoculum is 'strong''; 6 → 2: If 'inoculum is 'strong'' then 'inoculum size is 'large''; 6 → 3: or 'preculture time is 'long''; 2 → 5: If 'inoculum size is 'large'' then 'temperature = '*NS*''.

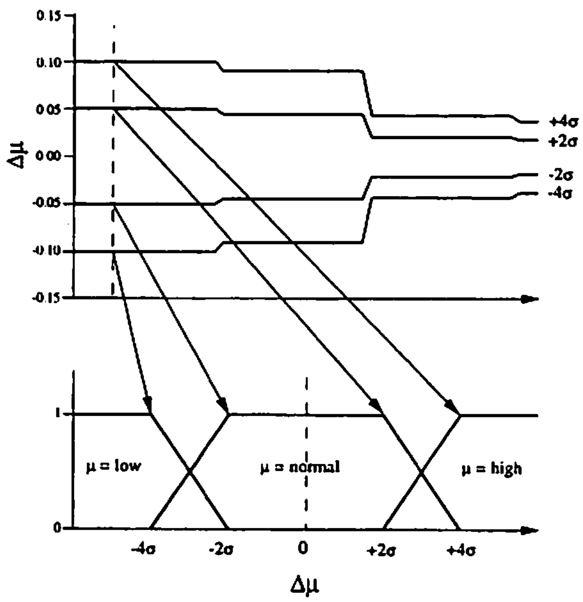

Fig. 4. Dynamically changing fuzzy membership functions for the change of state variable μ. σ = standard deviation of $\Delta\mu$.

carried out under standard conditions. The standard deviations of the state variables presented in the *EndNodes* were much larger in the beginning of the cultivation due to signal noise, and decreased gradually towards the end of the process. An example for a state variable is shown in Fig. 4, which clearly demonstrates how four different fuzzy intervals for each state variable were used depending on the process time involved.

4. Functional structure of the knowledge network

In order to realize the inferencing in the system, several slots and procedures were implemented to the nodes in the knowledge network. Slots could store either static or dynamic values, of which the latter were calculated by the procedures given. Since an object oriented programming environment was used, procedures are referred to here as messages and slots as instance variables. The procedures and slots employed are described in a greater detail below.

Slots named *InNodes* and *OutNodes* can be defined for each node in the knowledge network. The set of nodes connected to the right side of the certain node are defined as *InNodes* of that node and the nodes connected to the left side are called *OutNodes*. Together they indicate static knowledge of a causal relationship between two facts in a bioprocess. *InNodes* include the reasons for a fact in a node, and *OutNodes* show the results following from the reasons. A *TruthValue* is a slot, which indicates the degree of possibility of the fact as the basis of a single node. The *TruthValue* varied within a range from 0 to 1. In order to tie the knowledge of a certain bioprocess together a slot named *Model* was defined. This slot represents the model of the process under investigation. It includes a procedure to access to the process and state variables of the bioprocess in question, to the respective fuzzy membership functions, and to realize dynamic calculation of the *TruthValue* of the facts.

Each node in the network is given a preliminary *TruthValue* by an initializing procedure performed by the node itself as follows:

(a) *OrNode, ForwardOrNode*: The node initializes its *TruthValue* automatically to 0.

b) *AndNode, ForwardAndNode*: The node initializes its *TruthValue* to 1.

(c) *StartNode, EndNode*: The node calculates the *TruthValue* of its fact using a fuzzy membership function defined for the variable in question according to the slot *Model*.

Subsequently, the *TruthValue* of a node is recalculated from the actual *TruthValues* of the *InNodes* through backward chaining or from its *OutNodes* in forward chaining. The calculation depends on the node type in the following way (*TruthValues* μ_A, μ_B denote different values according to the nodes in question in each case):

(a) *OrNode, ForwardOrNode*: The *TruthValue* is set according to the maximum operation (Fuzzy-Or); a maximum of the *TruthValues* of the *InNodes* (*OutNodes* in case of *ForwardOrNodes*, respectively), ($\mu_A \cup \mu_B = \max[\mu_A, \mu_B]$).

(b) *AndNode, ForwardAndNode*: A minimum value (Fuzzy-And) of the *TruthValues* of its *InNodes* (*OutNodes* in case of *ForwardAndNodes*, respectively) is taken as a new *TruthValue*, ($\mu_A \cap \mu_B = \min[\mu_A, \mu_B]$).

The *TruthValue* calculations described above are independent of possible later network modifications or additions. Consequently, possible newly installed parts of the knowledge network can be considered separately in this respect. Fig. 2 illustrates how a newly installed part only affects the nodes B and C of the original network, which have a causal relationship with the new part of the network. The *TruthValues* of the rest of the nodes, marked with shadow in the original network do not change.

5. Backward and forward chaining procedures in fault diagnosis

In a typical deductive system, such as fault diagnosis, inference is done at query time using backward chaining. It is a method of reasoning by which goals are proven to be true by recursively proving that the sub-goals are true. In the present work, original causes of possible faults were selected as the goals in the fault diagnosis. Both backward and forward chaining procedures were needed in order to diagnose two different types of faults; those which can be realized by a simple measurement and those which are more abstract and cannot be measured. In the backward chaining procedure, each node in the network, starting from the *EndNodes*, sends a message backward to each of its *InNodes* and as a result the *TruthValue* from each of them is returned. After receiving all of the *TruthValues* from the appropriate *InNodes*, the node in turn will recalculate its own *TruthValue* and send it further to its *OutNodes* or, in case there are no more *OutNodes*, to the inference engine.

In the present work the forward chaining method was implemented to the fuzzy expert system shell in order to diagnose those characteristic phenomena that are not directly observable, such as microbial contamination during the process. In the present fault diagnosis system, the forward chaining is complementary to backward chaining. The inference begins with known facts and proceeds forward seeking to generate new facts by matching rules contained in the knowledge base (Russo and Peskin, 1987). Each node in the network sends the forward chaining message to each of its *Outnodes*, starting from the leftmost ones, and as a result the *TruthValue* from each of them is returned. After receiving all of the *TruthValues* from its *OutNodes*, the node in turn will recalculate its own *TruthValue* and send it further to its *InNodes* or, in case there are no more *InNodes*, to the inference engine.

6. Fault diagnosis

An illustrative example of fault diagnosing employing backward chaining is shown in Figs. 5 and 6. First, the system receives the measured or calculated values of the process variables. Then, the inference engine sends a message in a consecutive order to each one of the *EndNodes* to calculate their respective *TruthValues* on the basis of the measurement data. As a result each *EndNode* returns a *TruthValue* for its fact. If this *TruthValue* is higher than the pre-defined threshold, the *EndNode* is regarded as the observed fault or fact in the bioprocess.

In order to find possible reasons for this malfunction the inference engine sends a *backward chaining* message to the *EndNode* which further sends it to the *InNodes* in the network. During this procedure the nodes that are defined for forward chaining (*ForwardOrNodes, ForwardAndNodes*) will reply by similarly carrying out forward inference using the forward chaining procedure. When backward chaining reaches the *StartNodes*, which stand for the original causes for malfunctions, their *TruthValues* are calculated by evaluating the respective fuzzy operations defined in their facts.

In this example (Figs. 5 and 6), the *OrNodes* (π = 'low', conditions faulty) and *AndNode* (inoculum is weak) are initialized to 0.0 and 1.0, respectively, on the way to the *StartNodes*. The procedure is performed chain by chain, calculat-

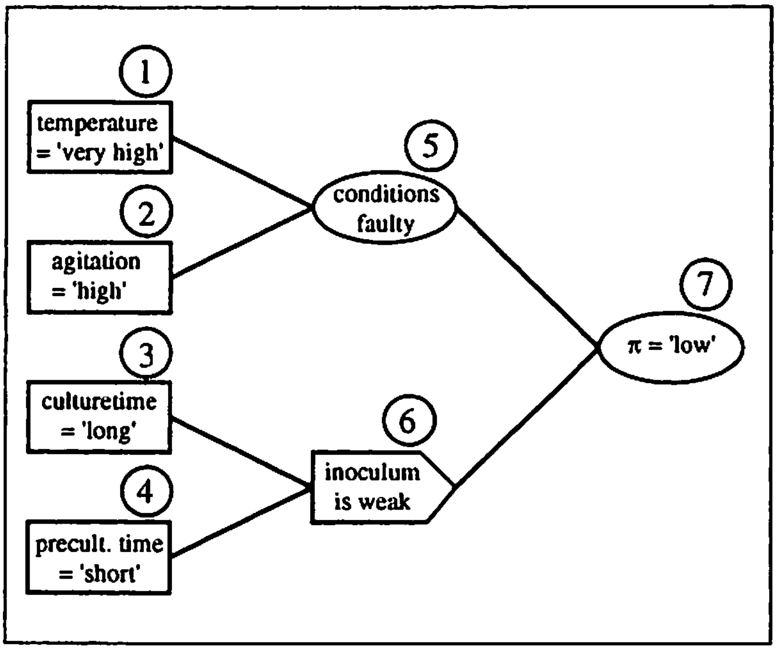

Fig. 5. A part of the knowledge network for lactic acid cultivation.

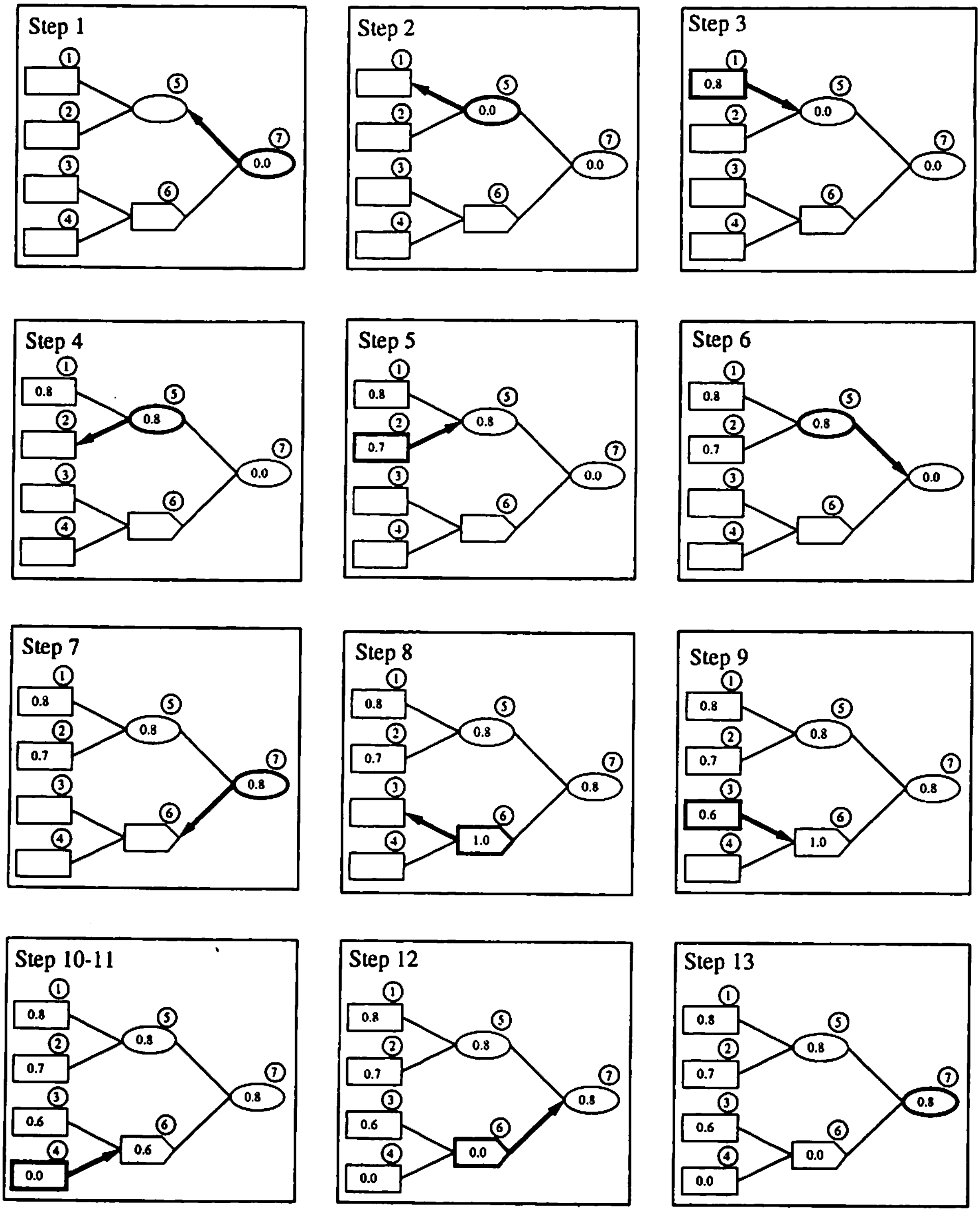

Fig. 6. An example of fault diagnosis in lactic acid cultivation.

ing the *TruthValue* for each node on the way back to the *EndNode* (π = 'low'). The *TruthValues* of the *StartNodes* 1 to 4 (temperature = 'very high', agitation = 'high', culture time = 'long', preculture time = 'short') were 0.8, 0.7, 0.6 and 0.0, respectively. After termination of the backward chaining procedure, every node has its own *TruthValue*. The *TruthValue* of one node can be regarded as a *TruthValue* of a certain part of the network. Consequently, for example in the step 13 the *OrNode* 5 (conditions faulty) has a *TruthValue* of 0.8 obtained as the maximum of the *TruthValues* of the *StartNodes* 1 (temperature = 'very high', *TruthValue* 0.8) and 2 (agitation = 'high', *TruthValue* 0.7).

After having set the *TruthValue* of every node by using the slots and procedures available, the inference engine starts to report all the chains in the network that have a *TruthValue* higher than a pre-defined threshold by displaying the faults and their reasons in a hierarchical order to the operator. As a result the appropriate *EndNode* (π = 'low') will be given a new *TruthValue* (0.8) calculated according to the Fuzzy-Or operation (the maximum of the *TruthValues* of the *InNodes*), representing the level of certainty at which the network could predict the original reason for the malfunction represented in the facts of the *EndNode*s at a given time. Fig. 7 gives an example display of the user interface at a given time during the diagnosing.

7. Control strategies

After the system has determined the cause of a process fault through inferencing as described above, it analyzes the appropriate control actions. All of the *StartNodes* which have a *TruthValue* greater than a pre-defined value are considered

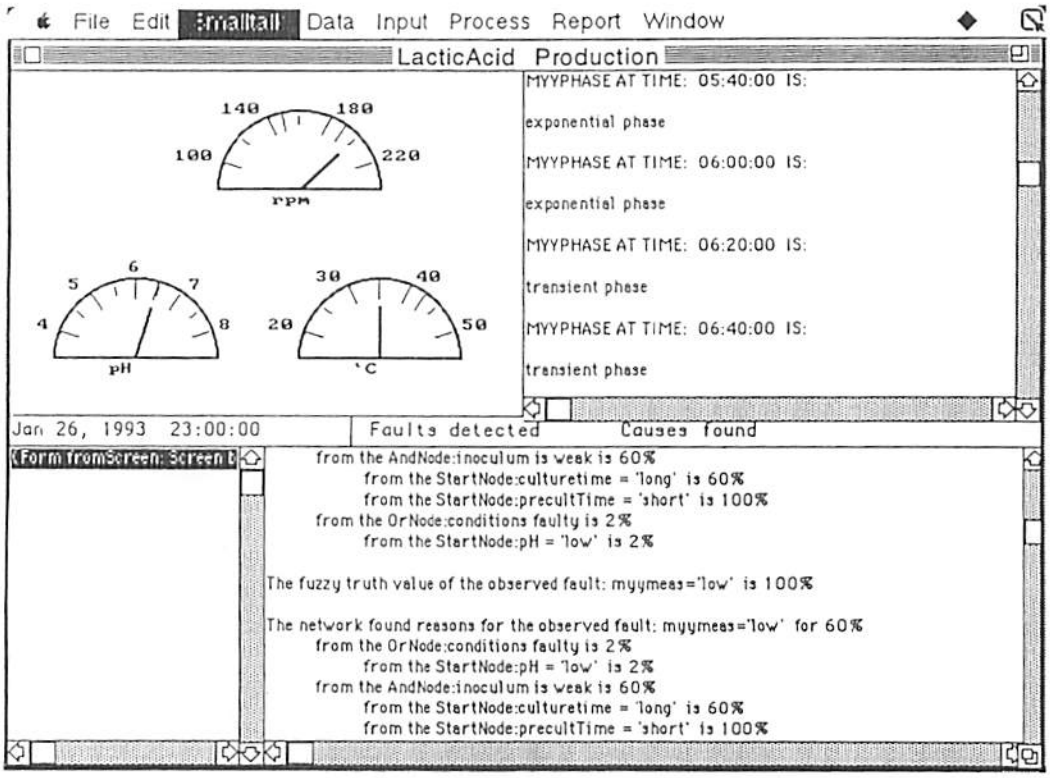

Fig. 7. User interface of the expert system during fault diagnosis of a lactic acid production process.

as possible reasons for the observed malfunction. With measurable faults each one of such nodes has its own control strategy represented in a form of a connected *ActionNode*, or similarly in the case of forward chaining with non-measurable faults a *ForwardActionNode* for sending a message on-line to the process computer for further actions. The control strategies were individually pre-determined for each one of such nodes on the basis of expert knowledge representing the key strategies to recover the process. The three basic types of control strategies used in the present work were, in Smalltalk syntax, *SetPoint*, *Advice* and *FuzzyAnswer*.

In the *SetPoint* control strategy an exact value combined with the name of the process variable was sent to the process computer in order to adjust the process variable to its pre-determined set point.

If the inference engine had reached the conclusion to recommend the control strategy called *Advice*, the process computer would draw the attention of the operator to this fact and give him advice for a manual operation.

The *FuzzyAnswer* control strategy consisted of three parts, a process variable, the respective pre-defined fuzzy membership function, and the *TruthValue* of the discovered original cause. For each discovered fault a *FuzzyAnswer* is created. To obtain crisp values when using the control strategy *FuzzyAnswer*, a defuzzifier based on the fuzzy mean (FM) method was employed according to Eq. 1 (Postlethwaite 1990):

$$A_D = \frac{\int_{s^-}^{s^+} u\mu_A(u)\,du}{\int_{s^-}^{s^+} \mu_A(u)\,du} \quad (1)$$

where A_D is the crisp value of the fuzzy set A, $\mu_A(u)$ is the fuzzy membership function of A, and $\mu_A(u) = 0$ for $u \leq s_-$ or $u \geq s_+$.

To further illustrate the function of *FuzzyAnswer* an example case is described below. The *EndNode* messages of 'ν = 'high'' and 'μ = 'normal'' were observed, and as a result of the fault diagnosis the *StartNode* 'inoculum size 'large'' with a *TruthValue* of 0.4 was found as the original cause of the malfunction of μ being 'high' (see Fig. 3). Further, the fact 'μ = 'normal'' was connected to the *ForwardOrNode* 'process going well' with a *TruthValue* of 0.6. After finding the two facts two *FuzzyAnswers* were activated as countermeasures, (a) 'Change temperature to NS' (*TruthValue* 0.4), and (b) 'Change temperature to ZE' (*TruthValue* 0.6). The latter control strategy is applied to reduce radical changes in the set points when the process is actually going well according to the measured value of μ. The real control value (change of setpoint) for the temperature was calculated on the basis of the *TruthValues* of these two activated *ForwardOrNodes*. The expert system calculates the crisp values for the control variables separately using FM method for defuzzification. In the fuzzy expert system shell the membership functions for the defuzzification of any chosen variable can be easily set through the user interface. The crisp values obtained are sent back on-line to the process computer together with the appropriate control strategies for further actions. The system performed satisfactorily when tested on-line with a laboratory scale lactic acid production process.

8. Conclusions

It is difficult to avoid problems related to malfunctions in a production process. However, if faults are diagnosed and corrected before becoming too serious, various losses in the bioindustry can be significally reduced. For this reason an on-line, real-time fuzzy knowledge based diagnosing and control system was developed. A fuzzy expert system was installed to realize the diagnosing and control faculties, and a real-time data acquisition and processing system was used to carry out on-line actions. Fuzzy membership functions of each process and state variable, casual relationships to malfunctions, and defuzzification rules were determined on the basis of historical data and experience from the actual production processes. The system could satisfactorily perform on-line, real-time diagnosing and control.

Acknowledgements

This study was supported by special co-ordination funds of the Science and Technology Agency of the Japanese Government. Grants for two of us (CvN, TS) from the Academy of Finland are gratefully acknowledged.

References

Aarts, R.J., Suviranta, A., Rauman-Aalto, P. and Linko, P. (1990) An expert system in enzyme production control. Food Biotechnol. 4, 301–315.

Endo, I., Asama, H. and Nagamune, T. (1989) A database system and an expert system for realizing factory automation in bioindustries. In: Fiechter, A., Okada, H. and Tanner, R.D. (Eds.), Bioproducts and Bioprocesses. Springer-Verlag, Berlin, pp. 337–346.

Endo, I., Nagamune, T. and Inoue, I. (1985) Autosampler. US Patent No. 4501161.

Endo, I. and Nagamune, T. (1987) A database system for fermentation processes. Bioprocess Eng. 2, 111–114.

Linko, P. (1988) Uncertainties, fuzzy reasoning and expert systems in bioengineering. Ann. NY Acad. Sci. 542, 83–101.

Nagamune, T., Inoue, I. and Takematsu, N. (1985) Instrument for measuring concentration of substrate in suspension. US Patent No. 4561779.

Pokkinen, M., Flores Bustamante, Z.R., Asama, H., Endo, I. and Linko, P. (1992) Diagnosing lactic acid fermentation based on specific rates of growth, substrate consumption and product formation. Bioprocess Eng. 7, 319–322.

Postlethwaite, B. (1990) Basic theory and algorithms for fuzzy sets and logic. In: McGhee J., Grimble, M.J. and Mowforth, P. (Eds.), Knowledge-based Systems for Industrial Control. Peregrinus, London. pp. 34–46.

Russo, M.F. and Peskin, R.L. (1987) Knowledge-based systems for the engineer. Chem. Eng. Progress 83, 38–43.

Siimes, T., Nakajima, M., Yada, H., Asama, H., Nagamune, T., Linko, P. and Endo, I. (1992a) Knowledge based diagnosis of inoculum properties and sterilization time in lactic acid fermentation. Biotechnol. Tech. 6, 385–390.

Siimes, T., Nakajima, M., Yada, H., Asama, H. Nagamune, T., Linko, P. and Endo, I. (1992b) Object-oriented fuzzy expert system for on-line diagnosing and control of bioprocesses. Appl. Microbiol. Biotechnol. 37, 756–761.

Stephanopoulos, G. (1987) The future of expert systems in chemical engineering. Chem. Eng. Progr. 83, 44–51.