Sequential Adaptive Networks for Feed Forward Control of Fed-batch Bioprocesses

RAJIB NAYAK, JAMES GOMES
Department of Biochemical Engineering and Biotechnology
Indian Institute of Technology, Delhi
Hauz Khas, New Delhi, 110016
INDIA

Abstracts: - The development of Sequential Adaptive Networks (SAN) for on-line control of bioprocesses is presented. SAN architecture is an assembly of networks arranged chronologically with each sub-network assigned to a specific sampling interval of the process. The sub-network can have any desired architecture. For the fed-batch yeast fermentation process, each sub-network is a three-layered feed-forward network with outer layer intra connections from the Dissolved Oxygen (DO) node to the other nodes. The information of the metabolic state of the process obtained from DO measurement is used to update the SAN, enabling an on-line adaptation to changing process dynamics. Two different feed-forward controllers, a mechanistic and a heuristic, are used in combination with SAN for on-line control of the fermentation process. SAN exhibits accurate prediction of unmeasured states, and a robust and stable tracking of the control trajectory. A comparative study of its performance is presented.

Key-Words: - Sequential Adaptive Networks, Neural Networks, Hybrid Modeling, On-line Adaptation, Fed-batch Bioprocess, Feed forward Controller, Yeast cultivation.

1 Introduction

Control strategies for bioprocess should be stable, robust and capable of precisely tracking the control objective because biological products must conform to strict regulations laid down by various governing agencies. The more recent and innovative products have extremely high value and every effort needs to be made to maximize economic benefits. It is already well known that biological processes exhibits strong nonlinear properties and must be treated with special care. To address issues arising because of the inherent nonlinear nature of bioprocesses, many methods have been used including nonlinear system theory [1, 2] and neural networks [3, 4]. Often neural networks methods are preferred over other methods since a meaningful mathematical representation of bioprocesses becomes unnecessarily complex. Another attractive feature of neural networks is that, it can be used in various ways for control implementation either independently or in combination with mechanistic process model.

In this work, we present the detailed development of SAN and its performance analysis in hybrid feed-forward control systems. A fed-batch yeast fermentation process is used as model system for on-line control implementation. Dissolved Oxygen measurements at every sampling instant are used to adapt SAN to changing physiological states of the process. A comparison of the performance of the heuristic feed-forward control and mechanistic feed-forward control based on SAN is presented.

2 SAN Architecture and Derivation of Adaptation Algorithm

The physiological state of the cell population and the process conditions in fermentation varies at different phases of cell growth; therefore, it is not possible for single neural network to store information about all physiological states of process. Also, these networks are more effective in predicting data similar to the more recent training data and less effective in predicting past trends. The SAN architecture presented here, addresses this problem. Sub-networks assigned to each sampling interval enable a distribution of memory for various physiological states occurring over these smaller time frames. Each of these networks in turn are adapted to changing process conditions based on on-line measurements that guide the entire ensemble along the evolution of the process. The connections between the layers of different nodes are of the feed forward type. The process variables s, x, c_l and E_t (glucose, biomass, DO and ethanol concentrations respectively) and the volume of the reactor, V are inputs to each of the sub-networks.

The training for SAN has been described with the help of Fig. 1a. Here, $\Delta t = t_n - t_{n-1}$ for any instant n. The network is trained such that it is able to predict the output variables at time $t+\Delta t$, if input variables are supplied at time t. Here Δt is the sampling time interval for which network has been trained.

Each sub-network considered has the same architecture as is shown schematically in Fig. 1b. The mathematical formulation is described by the following equations.

Considering the hidden layer input (H_j *in*) and output (H_j *out*), we obtain the relations

$$H_j in = \sum_i U_{ij} I_i + \xi_j \tag{1}$$

$$H_j out = \phi(H_j in) \tag{2}$$

Considering the output layer input (O_k *in*) and output (O_k *out*), we obtain the relations

$$O_k in = \sum_j V_{jk} H_j out + W_k O_{DO} out(n-1) + \xi_k \tag{3}$$

$$O_k out = \psi(O_k in) \tag{4}$$

Error with respect to the desired values is given by

$$e_k = d_k - \mathrm{O}_k out \tag{5}$$

and the total mean square error (mth dataset, for the particular sampling interval)

$$E_m = \frac{1}{2} \sum_k e_k{}^2 \tag{6}$$

Total error of m datasets is therefore

$$E = \sum_m E_m \tag{7}$$

The training of the network is done using gradient descent using the delta rule back propagation algorithm [5]. The weights are updated as follows

$$\Delta V_{jk}(n) = -\eta \frac{\partial E_m}{\partial V_{jk}} + \alpha\left(\Delta V_{jk}(n-1)\right) \tag{8}$$

$$V_{jk}(n) = V_{jk}(n-1) + \Delta V_{jk}(n) \tag{9}$$

$$\Delta U_{ij}(n) = -\eta \frac{\partial E_m}{\partial U_{ij}} + \alpha\left(\Delta U_{ij}(n-1)\right) \tag{10}$$

$$U_{ij}(n) = U_{ij}(n-1) + \Delta U_{ij}(n) \tag{11}$$

Where, U_{ij}, V_{jk} and W_k are the weights between input and hidden layer, hidden and output layer and intra-connections in output layer respectively. d_k is the desired output. I_i is the input to the networks. Ψ, ϕ are the transfer functions and ξ is the biases. α is the momentum coefficient and η is the learning rate.

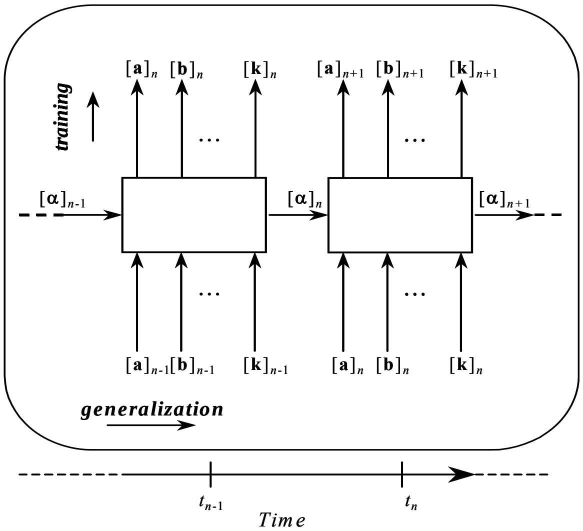
$[\mathbf{a}]_n$ $[\mathbf{b}]_n$ $[\mathbf{k}]_n$
$[\mathbf{a}]_{n+1}$ $[\mathbf{b}]_{n+1}$ $[\mathbf{k}]_{n+1}$
training
$[\alpha]_{n-1}$
$[\alpha]_n$
$[\alpha]_{n+1}$
$[\mathbf{a}]_{n-1}[\mathbf{b}]_{n-1}$ $[\mathbf{k}]_{n-1}$
$[\mathbf{a}]_n$ $[\mathbf{b}]_n$ $[\mathbf{k}]_n$
generalization
t_{n-1}
t_n
Time

Fig.1a Architecture of SAN

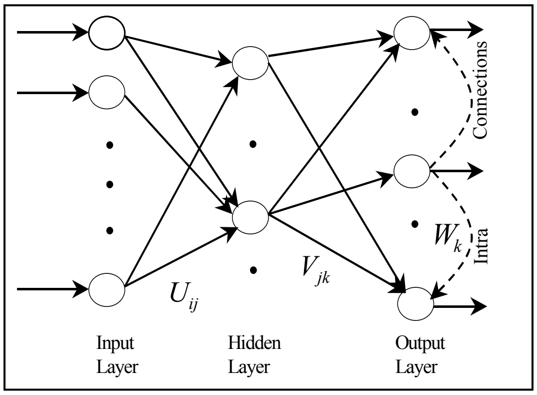

Fig.1b Schematic diagram of feed-forward neural network for each sub-network of SAN

3 Fed-Batch Model for Simulation of Yeast Cultivation

In the absence of large experimental data sets required for training the SAN a process model was used to generate the data. In this study the Sonnleitner & Kappeli model for yeast fermentation was used [6]. The model proposed by Sonnleitner is mechanistic and is based on the fact that glucose degradation proceeds via two metabolic pathways - oxidative and reductive. Under aerobic conditions ethanol is formed as the end product of reductive energy metabolism. Ethanol can only be used oxidatively because of the involvement of a limited respiratory capacity. This model has been modified by adding the dilution terms for fed-batch yeast fermentation to enable the simulation of various process conditions needed for the present study. The equations for fed-batch model are

$$\frac{dx}{dt} = Y^{oxid}_{bio/\mathrm{glu}} \frac{q_{o_2 glu\max}}{a} \left(\frac{C_L}{C_L + k_0}\right) x + \tag{12}$$

$$Y^{red}_{bio/glu}(q_s - q^{oxid}_{O_2})x + Y_{bio/Ethanol} q_{ethanol} x + D(t)(x_0 - x)$$

$$\frac{ds}{dt} = -q_{s,\max} \frac{s}{s + k_s} x + D(t)(s_{in} - s) \tag{13}$$

$$\frac{dEt}{dt} = Y^{red}_{ethanol/glu} q^{red}_s x - \frac{q_{O_2 ethanol}}{k} x + D(t)(Et_{in} - Et) \tag{14}$$

$$\frac{dc_l}{dt} = k_L a(c^* - c_L) - \tag{15}$$

$$q_{O_2,glu,\max} \frac{c_L}{c_L + K_o} x - q_{O_2,ethanol,\max} \frac{c_L}{c_L + K_o} x$$

$$\frac{dV}{dt} = F(t) \tag{16}$$

Where *x*, *s*, c_l, E_t are the biomass, substrate, DO and ethanol concentration (g l^{-1}) respectively. *D* is the dilution rate (h^{-1}), *F* is the substrate feed flow rate (l h^{-1}), *t* is time (h) and *V* is the volume of the reactor (l). The values of the parameters are directly taken from Sonnleitner & Kappeli model [6].

4 Control strategies to implement SAN for yeast cultivation process

The main objective of this study is to implement the proposed SAN for maximization of biomass production by keeping ethanol concentration at a predefined minimum value after batch phase is over. For this purpose two control strategies are proposed. In the first strategy, SAN is used in combination with a feed-forward neuro controller derived from the Sonnleitner & Kappeli model and in the second strategy, SAN is used to tune the parameters of a heuristic controller. In both the cases SAN is used to predict the unmeasured state variables to calculate the manipulated state variable, namely, the substrate feed flow rate.

Control implementation begins with training of SAN with the process data simulated from the Sonnleitner & Kappeli model. The trained SAN has reasonable prediction capacity that needs to be adapted to changing process condition to meet performance criteria in control implementations. The weights of the individual sub-networks of the SAN are updated at every sampling instant by minimizing the difference between the predicted and measured values. This is carried out for each sub-network sequentially.

4.1 Control strategy I

Here we propose a feed forward control law that maximizes yeast production without producing ethanol. The control objective is to keep the residual ethanol concentration below a threshold value so that all the substrate can be used for biomass production. This is achieved if the rate of formation of ethanol is maintained at zero i.e., $dE_t/dt = 0$ during the fed-batch phase. Now from Eq. 14 and 12 a feed forward control law can be derived with the assumption that if E_t is a small quantity then $(DE_t) \approx 0$. Thus, either

$$F = V\left[\left(\frac{ds/dt + \{8f(c_l)\}}{(a - 1.87)f(c_l) + 1.87} x\right) / (s_{in} - s)\right] \tag{17}$$

when $q_{O_2 ethanol}{}^{\max} = (8 - q_{O_2 glu\max})$

or $F = V\left[\left(\frac{ds/dt + q_{ethanol}}{1.87\{1 - f(c_l)\}} x\right) / (s_{in} - s)\right]$ (18)

when $q_{O_2\,ethanol}{}^{\max} = (q_{ethanol} \times k)$

SAN predicts the unmeasured states required by the feed forward control law for calculating the input feed flow rates.

4.2 Control strategy II

To achieve the maximum yeast production by keeping ethanol concentration at a predefined value, Ringbom *et al*[7] proposed that the substrate feed rate was set to follow a profile proportional to the quantity of biomass present in the system.

$$F(t) = \frac{kVx}{s_{in}} \tag{19}$$

Where *k* is the highest possible substrate feed maintained in the beginning of the run, so that no ethanol is formed. The value of the *k* has to be adjusted on-line in order to maximize the yeast growth without ethanol fermentation. The value of parameter k can be calculated with the help of predicted values of unmeasured state variables by SAN and also tuned in such a way that if the ethanol concentration increased beyond a set value (E_{tmin}) then the value of k was reduced by factor of 0.8. Thus the variation in k follows

$$k = \begin{cases} k & E_t < E_{t\min} \\ 0.8k & E_t \geq E_{t\min} \end{cases} \qquad (20)$$

The flow diagram of both control strategies is shown in Fig. 2.

5 Results and Discussion

5.1 Training and Generalization Performance of SAN

The data was generated at a constant sampling interval 15 min for 10 h with batch time 5 h and the batch volume was 6 l. Training and validation data sets with three varying initial conditions, initial substrate concentration, substrate concentration in feed and minimum ethanol concentration (a predefined set value of ethanol concentration, E_{tmin}) with respect to time are shown in Fig 3. The data sets were chosen in such a way that it can cover a large region of metabolic space of the process while remaining within the physical constraints such as the maximum working reactor volume. The values of initial ethanol concentration, initial DO %, air flow rate, initial biomass concentration and volume of the reactor were 0.6 gl^{-1}, 100, 4 lm^{-1}, 0.7 gl^{-1} and 12 l respectively. The values of these initial conditions were kept constants throughout the simulated data sets. Three data sets from the each corner of the cube data sets (total twenty four) were taken for training and four data sets from the centre of the cube are taken for validation.

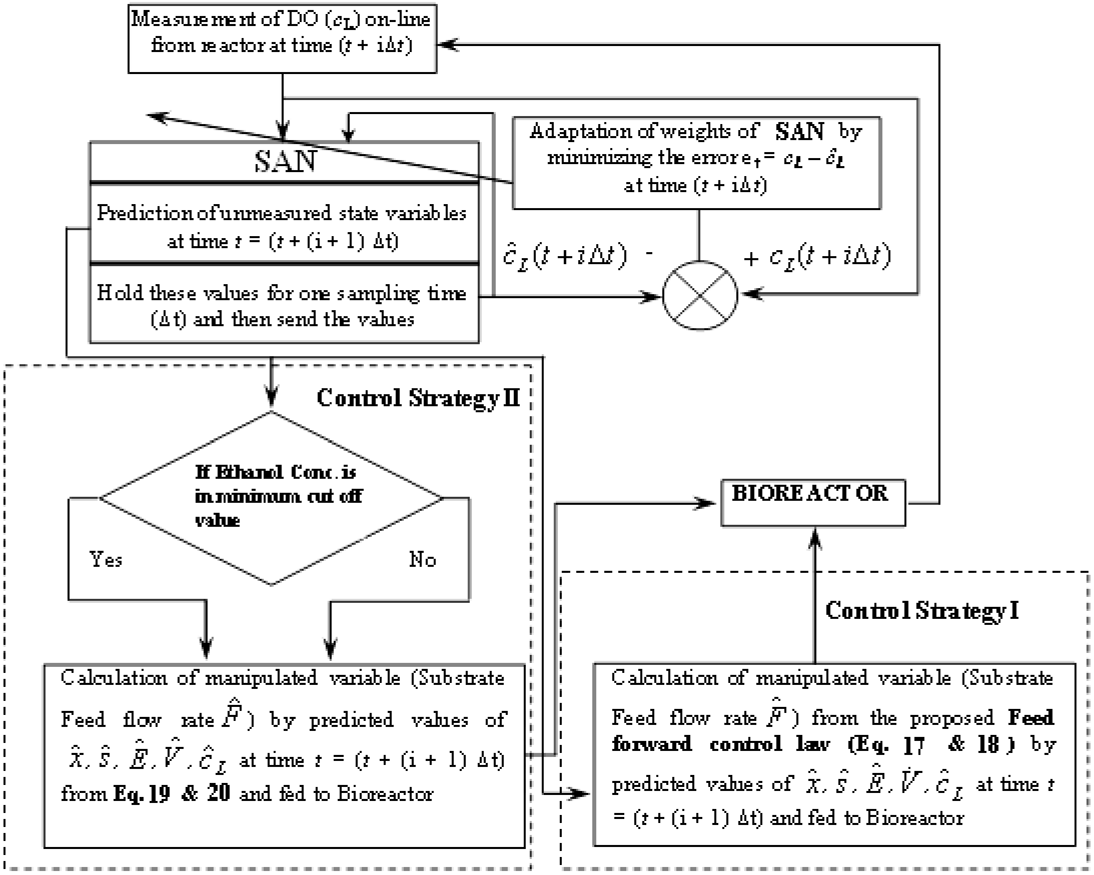
Measurement of DO (c_L) on-line from reactor at time (t + iΔt)
SAN
Prediction of unmeasured state variables at time t = (t + (i + 1) Δt)
Hold these values for one sampling time (Δt) and then send the values
Adaptation of weights of SAN by minimizing the error $e_t = c_L - \hat{c}_L$ at time (t + iΔt)
$\hat{c}_L(t+i\Delta t)$ -
+ $c_L(t+i\Delta t)$
Control Strategy II
If Ethanol Conc. is in minimum cut off value
Yes
No
Calculation of manipulated variable (Substrate Feed flow rate $\hat{F}$) by predicted values of $\hat{x}, \hat{s}, \hat{E}, \hat{V}, \hat{c}_L$ at time t = (t + (i + 1) Δt) from Eq. 19 & 20 and fed to Bioreactor
BIOREACTOR
Control Strategy I
Calculation of manipulated variable (Substrate Feed flow rate $\hat{F}$) from the proposed Feed forward control law (Eq. 17 & 18) by predicted values of $\hat{x}, \hat{s}, \hat{E}, \hat{V}, \hat{c}_L$ at time t = (t + (i + 1) Δt) and fed to Bioreactor

Fig. 2 Flow diagram of SAN application in control strategies for Fed-batch fermentation

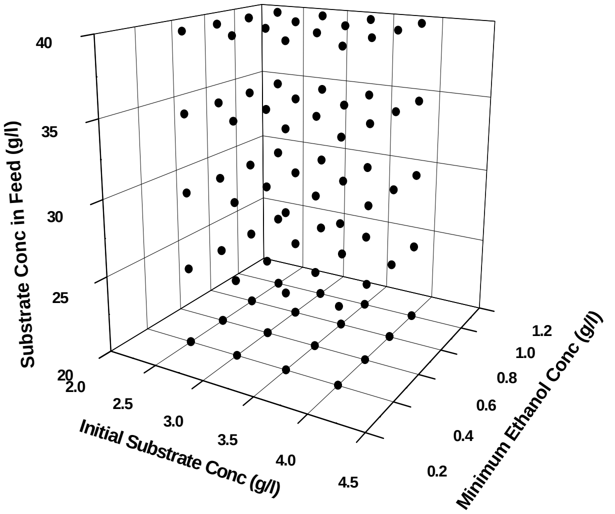

Fig. 3 Various combinations of initial conditions to generate data sets

Initial weights are assigned randomly between -0.1 to +0.1. The number of hidden nodes was optimized with respect to recall error and number of iterations to converge the networks. Seven hidden nodes were found to be optimum. The learning rate coefficient of 0.05 and momentum coefficient of 0.95 were used. The network was trained for an average threshold mean square error per data point of 10^{-5}. The average number of iterations per feed forward neural network unit of the network was 2500. The recall profiles matched the training dataset profiles with recall errors of the order of 10^{-5}. For validation of SAN outside the training domain, recall profiles matched the simulation profiles with generalization errors (average mean square error per data point) of the order of 10^{-4} (see Fig. 4 and Fig. 5).

5.2 Performance of implemented SAN for control strategies

Data set with initial substrate concentration 3.5 gl^{-1}, substrate concentration in feed 30 gl^{-1}, and minimum cut off ethanol concentration 0.8 gl^{-1} was taken for control implementation of SAN as well as to test the generalization performance of SAN outside the training domain, while other conditions were kept constant. 5% of random measurement noise was added to the DO data to test the robustness of the SAN in prediction of unmeasured state variables. The value of *k* for the control strategy II came to be 0.19 from the simulation. A sampling interval of 15 minutes is used to allow for measurement delay.

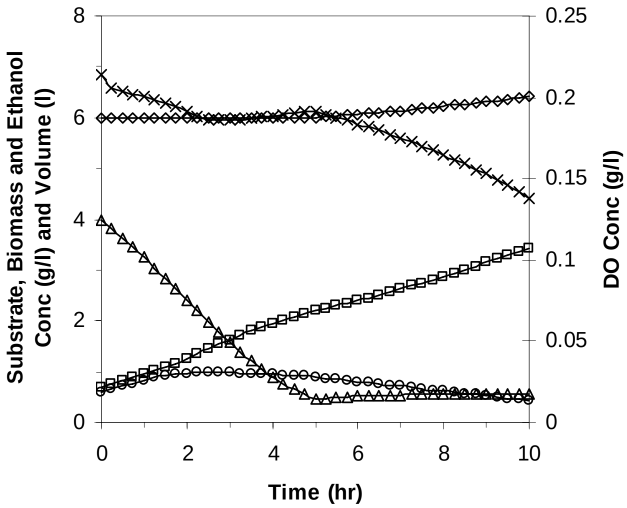

Fig. 4 Recall profiles of SAN for training data set simulated (—— from model, ○ Ethanol Conc., □ Biomass Conc., Δ Substrate Conc., ◊ Volume of reactor and × DO Conc.)

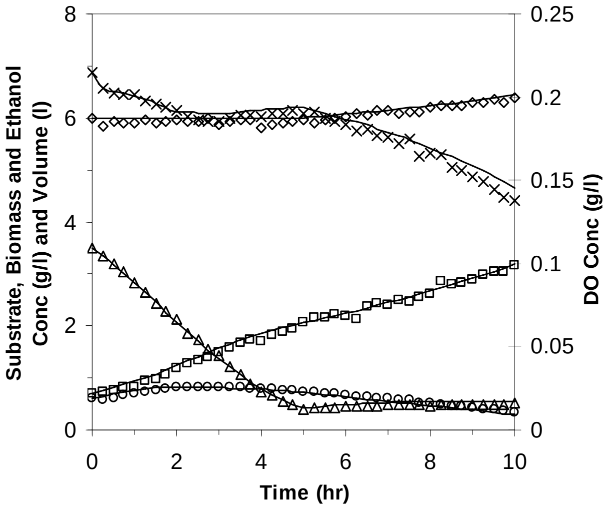

Fig. 5 Generalization profiles of SAN for training data set simulated (—— from model, ○ Ethanol Conc., □ Biomass Conc. Δ Substrate Conc. ◊ Volume of reactor and × DO Conc.)

From the simulation results we had got generalization errors (average mean square error per data point) of the order of 10^{-4} for without noise data and 10^{-3} for noise data. In case of data with added noise we observed that average mean square error per data point for biomass concentration, ethanol and feed flow rate were 10^{-3}, 10^{-4}, 10^{-3} respectively for control strategy I and II. The computation time for prediction and adaptation of the each sub-network of SAN is less then 5 minutes depend upon the nature of the data and tolerated error required for adaptation. The results are shown in Fig. 6 and 7. The profile of the manipulated variable, substrate feed flow rate (*F*) predicted by the SAN was closely matched the profile calculated from simulated data. From the simulation results it can also be concluded that in control strategy I is superior than the control strategy II because the amount of biomass produced is higher than that of control strategy II starting with the same initial conditions. These results also showed that a well trained SAN has enough potential to predict the one sampling time step ahead state variables and can estimate the control input for the process.

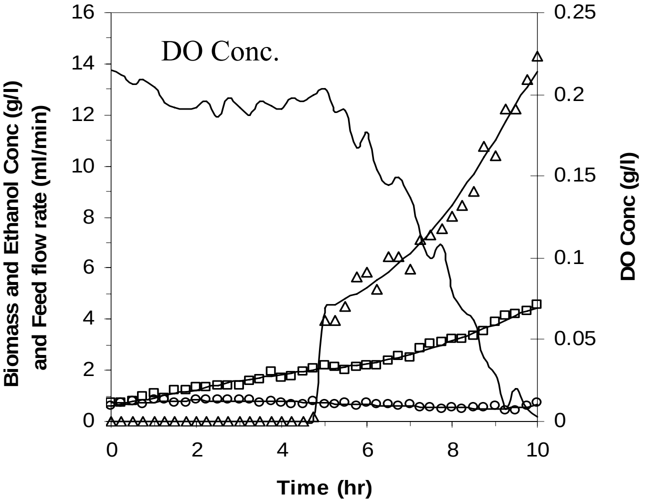

Fig. 6 Implementation of Control Strategy I simulated (—— from model, ○ Ethanol Conc., □ Biomass Conc., Δ Feed flow rate)

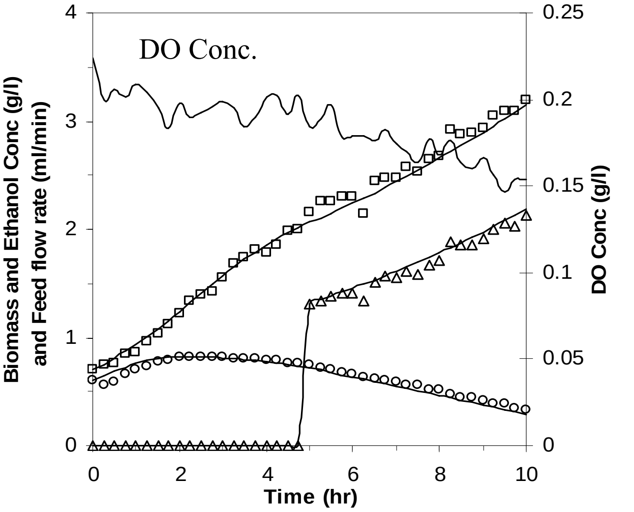

Fig. 7 Implementation of Control Strategy II simulated (—— from model, ○ Ethanol Conc., □ Biomass Conc., Δ Feed flow rate)

6 Conclusions

In this present study, we have developed a hybrid neural controller using SAN and evaluated its performance for controlling the yeast cultivation. Results show that it can use on-line with the process for various control strategies as it has capabilities of adapting with the changing dynamics of the process caused by inherent nonlinearity of the system as well as by the external disturbance. It adapts with the system dynamics by updating its weights from on-line measured state variable contains information about the system. The off-line computation time for training is around 3-7 minutes with convergence error 10^{-5} (Average mean square error per data point) whereas adaptation and generalization time is 2-5 minutes with convergence error 10^{-3} to 10^{-4}. In on-line implementation the trained SAN requires only a few seconds to adapt to the new conditions therefore SAN can be implemented irrespective of control strategy.

References:

[1] Gomes, J.; Menawat, A. S., Precious control of dissolved oxygen in bioreactors: a model-based geometric algorithm, *Chem. Engg. Sci.*, Vol.55, No.1, 2000, pp.67 – 78.

[2] Smets, I. Y. M.; Versyck, K. J. E.; Van Impe, F. M., Optimal control theory: A generic tool for identification and control of (bio-) chemical reactors, *Ann. Rev. Cont.*, Vol.26, 2002, pp.57 – 73.

[3] Oliveira, R., Combining first principle modeling and artificial neural networks: a general framework, *Comput. Chem. Eng.*, Vol.28, 2004, pp.755-766.

[4] Gadkar, K.; Mehra, S.; and Gomes, J., On-line Adaptation of Neural Networks for Bioprocess Control, *Comp. and Chem. Eng.*, Vol.29, 2005, pp.1047 – 1057.

[5] Baughman, D.R.; Liu Y.A., *Neural Networks in Bioprocessing and Chemical Engineering*, Academic press, 1995.

[6] Sonnleitner, B.; Kappeli, O., Growth of *Saccharomyces cerevisiae* is controlled by its limited respiratory capacity: formulation and verification of a hypothesis, *Biotechnol. Bioeng.*, Vol.**8**, 1986, pp. 927-937.

[7] Ringbom, K.; Rothberg, A.; Saxen, B., Model-based automation of baker's yeast production, *J. Biotechnol.*, Vol.51, 1996, pp. 73-82.