An Integral Plus States Adaptive Neural Control of Aerobic Continuous Stirred Tank Reactor

Ieroham S. Baruch, Petia Georgieva, and Josefina Barrera-Cortes

Abstract— **A direct adaptive neural network control system with and without integral action term is designed for the general class of continuous biological fermentation processes. The control system consists of a neural identifier and a neural controller, based on the recurrent trainable neural network model. The main objective is to keep the glucose concentration, which is considered as external substrate, close to a constant set-point reference using the dilution rate as manipulating function. It is illustrated by simulations that both adaptive neural control schemes (with and without integral-term) work successfully and exhibit good convergence. However, the control system with integral action is able to compensate a constant offset while the scheme without integration term failed. Results are presented which show a favorable behavior of the neural controller in comparison with existing solutions.**

***Index Terms*—Backpropagation learning, direct adaptive neural control with integral term, recurrent neural networks, stirred tank bioreactor.**

I. Introduction

The recent advances in understanding of the working principles of artificial neural networks (ANN) and the rapid growth of available computational resources led to the development of a wide number of ANN-based modelling, identification, prediction and control applications, [1], [2], [3], [4], especially in the field of chemical engineering and biotechnology, [5]-[22]. The ability of ANN to approximate complex non-linear relationships without prior knowledge of the model structure makes them a very attractive alternative to classical modelling and control techniques. Many of the applications currently reported are based on the classical Nonlinear Autoregressive Moving Average (NARMA) model, where a Feedforward Neural Network (FFNN) is implemented, [2], [5], [7], [9], [18], [22], [24], [25]. However, the FFNN has in general a static structure, therefore it is adequate to approximate mainly static (nonlinear) relationships and their real-time applications for dynamical systems require the introduction of external time-delayed feedbacks. The application of the FFNN for modeling, identification and control of nonlinear dynamic plants caused some problems which could be summarized as follows: 1. The dynamic systems modeling usually is based on the NARMA model which need some information of input/output model orders, and input and output tap-delays ought to be used, [23]; 2. The FFNN application for multi-input multi-output systems identification needs some relative order structural information; 3. The ANN model structure ought to correspond to the structure of the identified plant where four different input/output plant models are used, [25]; 4. The lack of universality in ANN architectures caused some difficulties in its learning and a Backpropagation through time learning algorithm needs to be used, [23]; 5. Most of NARMA-based ANN models are sequential in nature and introduced a relative plant-dependent time-delay; 6. Most of the ANN-based models are nonparametric ones, [25], and so, not applicable for an indirect adaptive control systems design; 7. All this ANN-based models does not perform state and parameter estimation in the same time, [2]; 8. All this models are appropriate only for identification of nonlinear plants with smooth, single, odd, nonsingular nonlinearities.

Recurrent Neural Networks (RNN) possesses its internal time-delayed feedbacks, so they are promising alternative for system identification and control, particularly when the task is to model dynamical systems [2], [26], [27]. Their main advantage is the reduced complexity of the network structure. However, the analysis of state of the art in the area of classical RNN-based modeling and control has also shown some of their inherent limitations as follows: 1. The RNN input vector consists of a number of past system inputs and outputs and there is no a systematic way to define the optimal number of past values [27] and usually, the method of trials and errors is performed; 2. The RNN model is naturally formulated as a discrete model with fixed sampling period, therefore, if the sampling period is changed, the network has to be trained again; 3. It is assumed that the plant order is known, which represents a quite strong modeling assumption in general, [27]. Driven by these limitations, a new Recurrent Trainable Neural Network (RTNN) topology and the respective Backpropagation (BP) type learning algorithm in vector-matrix form was derived, [28], [29], [30], and its convergence was studied in [31].

Modeling and control of fermentation processes has been always a challenging problem in bioengineering due to the

Manuscript received February, 2005. This work is partially supported by the project number 42035-Z/1.06.2003 of SEP-CONACYT.

Ieroham Solomon Baruch is with the Department of Automatic Control, CINVESTAV-IPN, A.P. 14-740, Col. Zacatenco, 07360 Mexico D.F., Mexico (phone: (+52) (55) 5061-3800/ext. 42-29; fax:(+52) (55) 5061-3812; e-mail: baruch@ctrl.cinvestav.mx).

Petia Georgieva is with the Department of Electronics and Telecommunications, University of Aveiro, 3810-193 Aveiro, Portugal (e-mail: petia@det.ua.pt).

Josefina Barrera-Cortes is with the Department of Biotecnology and Bioengineering, CINVESTAV-IPN, 07360 Mexico D.F., Mexico (e-mail: jbarrera@cinvestav.mx).

presence of strong time-varying process characteristics and lack of sufficient process reproducibility, [32]. Adaptive control solutions based on ANNs seem to gain more interest in fermentation industry mainly for controlling of *metabolite production* (a *metabolite* is a product of the microbial activity during the fermentation). In [33] a comparative study of linear, nonlinear and NN-based adaptive controllers for a class of fed-batch baker's yeast fermentation processes are reported. A FFNN, first introduced in [25], makes part of the identification and control structure, proposed, or as part of a hybrid control systems [5], [10], [15], [16], [17]. The FFNN (multilayer perceptrons [5], [7], [9], [18], [22], [25]; radial basis function ANNs [10]; dynamic wavelet ANNs [6]; neuro-fuzzy models [14]) ensures a good approximation of the nonlinear plant dynamics, assuming that the plant order is known, but the price to be paid is a high complexity of the static ANNs. All these ANNs are incorporated in a model reference adaptive control schemes [25], nonlinear predictive control schemes [7], [9], [13], inverse plant model control schemes (usually as a hybrid control systems, in combination with classical PID [10], Implicit Control [5], Sliding Mode Control [15], or Optimal Control Facilities [12], [13]). The Neural PID Control algorithm trained with a Gradient PID learning algorithm [11] raised a great popularity in the control of fermentation plants. The main difficulty here is the lost of stability for high order plants. In this case for the general class of continuous fermentation processes known also as aerobic continuous stirred tank reactors (CSTR) - an Integral- Plus –State (IPS) RNN Control could be applied [29].

The aim of this paper is to propose a new improved hybrid RNN control with I-term, where the I-term noise reduction control part is independent with respect to the RNN proportional plus state (P-S) dynamic compensation control part, which simplified the control structure and improved the reference tracking.

The paper is organized as follows: part two described the RNN topology and learning, and discussed its stability; part three gives block diagrams of the derived P and Integral, Proportional-Plus-States neural control schemes and studied its stability; part 4 described the baker's yeast fermentation aerobic process model; part 5 gives the simulation results obtained; part 6 represents the conclusions.

II. Topology and Learning of the Recurrent Trainable Neural Network

A. RTNN Topology

The Recurrent Trainable Neural Network (RTNN) model and the respective learning algorithm of dynamic *Backpropagation* type are introduced in [28]-[31]. The given there Jordan canonical topology is completely parallel with minimum number of parameters. The general RTNN architecture has the following mathematical description:

$$X(k+1) = AX(k) + BU(k) \tag{1}$$

$$Z(k) = \Gamma[X(k)] \tag{2}$$

$$Y(k) = \Phi[CZ(k)] \tag{3}$$

$$A = block - diag(a_{ii}); |a_{ii}| < 1 \tag{4}$$

Where: *X(k)* is a n - state vector; *U(k)* is a m -input vector; *Y(k)* is a l- output vector; *Z(k)* is an auxiliar vector variable with dimension n. Equations (1), (2) define the hidden layer of the RTNN and by equation (3) the feedforward output layer, is represented. The (nxn) state weight matrix *A* has a block-diagonal structure, with (a_{ii}) being the i-th diagonal block with (1x1) or (2x2) dimension. Variables *B* and *C* are weight (nxm) input and (lxn) output matrices respectively with structure, corresponding to the structure of *A*, and the integer k is the sampling time instant. Equation (4) is the local stability condition imposed on all blocks of *A*. The constraint (4) and the vector-valued sigmoid or hyperbolic tangent-activation functions Γ [.], Φ[.] ensure the stability of the overall RTNN model (see [28]-[31] for more details).

B. RTNN Learning

The learning of the network weights is based on the dynamic *Backpropagation* algorithm. Standard BP is a gradient descent algorithm in which the network weights are moved in the steepest descent direction i.e. along the negative of the gradient of the performance index. This is the direction in which the performance index is decreasing most rapidly. The term *Backpropagation* refers to the manner in which the gradient is computed. One k-iteration of the most general BP updating rule [28]-[31], is described by:

$$W_{ij}(k+1) = W_{ij}(k) + \eta \Delta W_{ij}(k) + \alpha \Delta W_{ij}(k-1) \tag{5}$$

Where: W_{ij} is a general notation, denoting each weight matrix element (C_{ij}, A_{ij}, B_{ij}) subject to adaptation in the RTNN model; ΔW_{ij}, (respectively ΔC_{ij}, ΔA_{ij}, ΔB_{ij}), is the weight correction of W_{ij}, while η and α are learning rate parameters. The weight matrix corrections of the network model (ΔC_{ij}, ΔA_{ij} and ΔB_{ij}) are defined by the following update equations:

$$\Delta C_{ij}(k) = [T_i(k) - Y_i(k)]\Phi'[Y_i(k)]Z_j(k) \tag{6}$$

$$G = C_i(k)[T_i(k) - Y_i(k)]\Gamma'[Z_i(k)] \tag{7}$$

$$\Delta A_{ij}(k) = GX_j(k-1) \tag{8}$$

$$\Delta B_{ij}(k) = GU_j(k) \tag{9}$$

Where: T_j is the j-th element of the target vector (the actual process output); Y_j is the j-th element of the RTNN output; X_i is an i-th element of the state vector; Z_i is the i-th element of the output vector of the hidden layer; Γ', Φ' are first-order derivative of the activation functions; *G* is an auxiliary variable. A vector-matricial version of this BP-learning algorithm is derived in [31]. Stability proof of the RTNN and convergence of the BP learning algorithm are given also in [31], where the proved Theorem of stability is defined as follows:

C. Theorem of Stability

Let the RTNN with Jordan Canonical Structure, is given by equations (1), (2), (3) and the nonlinear plant model is supposed to be:

$$X_p(k+1) = f[X(k), U(k)] \tag{10}$$

$$Y_p(k) = h[X(k), U(k)] \tag{11}$$

Where $X_p(k)$, $U(k)$ and $Y_p(k)$ are plant state, input and output vector variables with dimensions n, m, l, respectively; $h(.)$, $f(.)$ are smooth bounded nonlinear functions. Under the assumptions made, the application of the BP learning algorithm for $A(k)$, $B(k)$, $C(k)$, in general matricial form, described by (5), and the learning rates $\eta(k)$, $\alpha(k)$, normalized with respect to the error, and derived using the following Lyapunov function:

$$L(k) = \| A(k) \| + \| B(k) \| + \| C(k) \| \tag{12}$$

Then the identification error is bounded, and:

$$\Delta L(k) \le -\eta(k) \, | E(k) |^2 - \alpha(k) \, | E(k-1) |^2 + d \tag{13}$$
$$E(k) = Y_p(k) - Y(k)$$

Where $\Delta L(k) = L(k) - L(k-1)$, the learning parameters $\eta(k)$, $\alpha(k)$ are normalized and depends on the error structure with the following bounds: $0 \le \|\eta(k)\| \le 1$, $0 \le \| \alpha(k)\| \le 1$, and d is bounded error perturbation term.

The described above RTNN is applied for identification and direct adaptive control of a baker's yeast fermentation process.

III. Direct Adaptive Neural Control with Integral Term

Block-diagrams of the proposed direct adaptive control system without and with integral (I)-term are given in Fig.1 and Fig.2, respectively. The control scheme, given in Fig. 2 consists of two RTNNs and a discrete-time integrator. The first RTNN is a neural identifier, which supplies the estimated state vector as an entry to the second RTNN, being the neural controller. An additional entry to the neural controller is the control error, i. e. $E_c(k) = R(k) - Y_p(k)$, where $R(k)$ is the reference vector variable with dimension l. The control action includes as an additional part the integral of the control error. Both networks are trained by the BP algorithm summarised by equation (5).

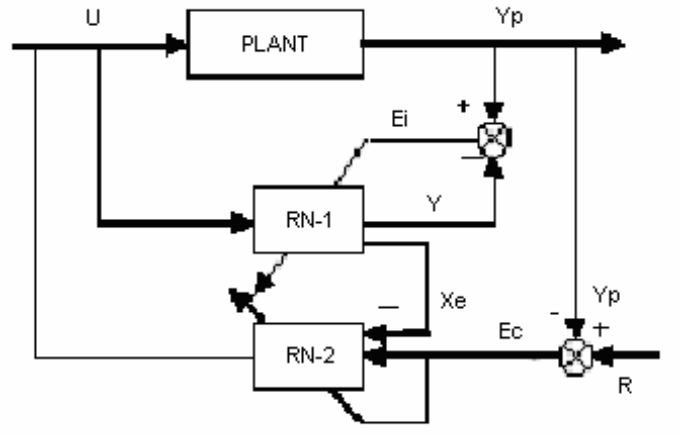

Fig. 1. Block-diagram of the direct adaptive neural control without I-term.

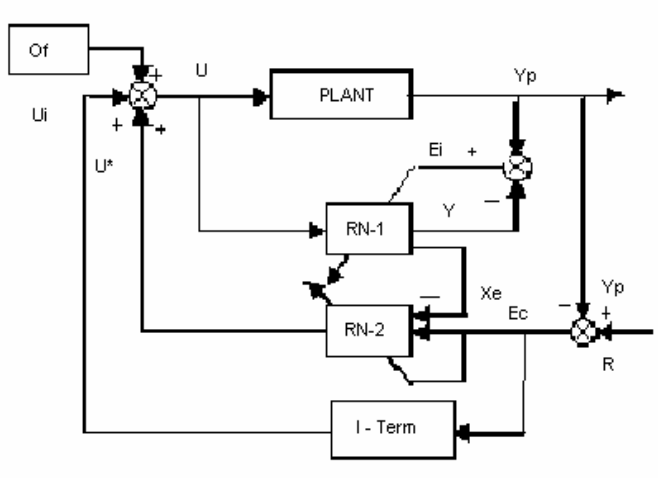

Fig. 2. Block-diagram of the direct adaptive neural control with I-term.

For the neural identifier (RN-1), the identification error $E_i(k) = Y_p(k) - Y(k)$ is used as the training signal and for the neural controller the control error $E_c(k)$ is the training signal. The linear approximation of the neural identifier model RN-1 (see equations (1)-(3)) is given by the following equations:

$$X_e(k+1) = AX_e(k) + BU(k) \tag{14}$$
$$Y(k) = CX_e(k) \tag{15}$$

Where: the state weight matrix A cope with the constraint (4). Similarly, the linear form of the neural controller model RN-2 is:

$$U^*(k) = C^* X^*(k) \tag{16}$$
$$X^*(k+1) = A^* X^*(k) - B_1 X_e(k) + B_2 E_c(k) \tag{17}$$
$$E_c(k) = R(k) - Y_p(k) \tag{18}$$

Where: $X^*(k)$ is n_c-dimensional state vector of the neural controller; $U^*(k)$ is an input vector with dimension m; $Ec(k)$ is a control error with dimension l; C^*, A^*, B_1, B_2 are weight matrices with compatible dimensions, where the state weight matrix A^* cope with the constraint (4). The reference signal $R(k)$ is a train of pulses and the control objective is that the control error tends to zero when k tends to infinity. The control action is sum of two components:

$$U(k) = U^i(k) + U^*(k) \tag{19}$$
$$U^i(k+1) = U^i(k) + T_0 K_i E_c(k) \tag{20}$$

Where: $U^i(k)$ is the output of the integrator with dimension l (it is supposed that l = m); T_0 is the period of discretization; K_i is an integrator (lxl) gain matrix. Applying the z-transformation we could obtain the following expressions and z-transfer functions:

$$U^i(z) = (z-1)^{-1} T_0 K_i E_c(z) \tag{21}$$
$$Q_1(z) = C^*(zI - A^*)^{-1} B_1 \tag{22}$$
$$Q_2(z) = C^*(zI - A^*)^{-1} B_2 \tag{23}$$

$$P(z) = (zI - A)^{-1} B; X_e(z) = P(z)U(z) \tag{24}$$

$$W_p(z) = C_p(zI - A_p)^{-1} B_p; \quad Y_p(z) = W_p(z)[U(z) + Of(z)] \tag{25}$$

Where: *Of(.)* is an offset variable with dimension m, which represents the imperfections of the plant. Equation (21) defines the I-term dynamics, equation (24) describes the hidden layer dynamics of the neural identifier RN-1, derived from equations (14) and (15), and equations (25) represents the plant dynamics, derived in linear state-space form. The z-transformation of the linearized description of RN-2 (equations (16), (17)), using also (22) and (23), yields:

$$U^*(z) = -Q_1(z)X_e(z) + Q_2(z)E_c(z) \tag{26}$$

Substituting $X_e(z)$ from (24) in (26), and together with (21), in the z-transformed equation (19), the following control signal expression is derived:

$$U(z) = [I + Q_1(z)P(z)]^{-1}[(z-1)^{-1}T_0 K_i + Q_2(z)]E_c(z) \tag{27}$$

Using the error $E_c(z)$ from (18), the control $U(z)$ from (27) and the plant model (25), we finally get an expression for the closed loop dynamics of the control system with proportional, integral, and state feedbacks. The z-operational form of the closed loop dynamics is as follows:

$$\{(z-1)I + W_p(z)[I + Q_1(z)P(z)]^{-1}[T_0 K_i + (z-1)Q_2(z)]\}Y_p(z) = W_p(z)[I + Q_1(z)P(z)]^{-1}. [T_0 K_i + (z-1)Q_2(z)]R(z) + (z-1)W_p(z)Of(z) \tag{28}$$

Equation (28) describes the dynamics of the closed-loop system with I-term neural control. Note that if (25) represents a stable and minimum phase plant and the neural networks RN-1, RN-2 are convergent, i.e. transfer functions (22), (23) and (24) are also stable and minimum phase, then the closed-loop system (28) will be stable. Since the term (z-1) is equivalent to a discrete-time derivative, then a constant offset *Of(k)* can be compensated by the I-term and the control error tends to zero $(E_c(k) \to 0)$ when $k \to \infty$.

IV. Mathematical Model of the Baker's Yeast Fermentation Aerobic Process

In this section, the model of the baker's yeast fermentation process is considered as a special case of the general model of biochemical aerobic processes introduced in [16]. The kinetic model for cellular productivity of a continuous culture of *Saccharomyces cerevisiae*, more commonly known as baker's yeast is studied and exemplified by numerous authors, referenced in [34]. The dynamical model is obtained from a mass balance of the components, and it is assumed that the reactor is well mixed, the yield coefficients are constant, and the dynamics of the gas phase can be neglected. The yeast fermentation goes through three pathways: sugar oxidation, ethanol oxidation and sugar fermentation with ethanol as an end product. Following the detailed description of the model in [34], the continuous mode operation of this process can be described in the form:

$$\begin{bmatrix} \dot{S} \\ \dot{X} \\ \dot{C} \\ \dot{E} \end{bmatrix} = \begin{bmatrix} -c_{11} & 0 & -c_{13} \\ c_{21} & c_{22} & c_{23} \\ c_{31} & c_{32} & c_{33} \\ 0 & -c_{42} & c_{43} \end{bmatrix} \begin{bmatrix} \mu_1(S,O) \\ \mu_2(S,O,E) \\ \mu_3(S,O) \end{bmatrix} X - D \begin{pmatrix} S \\ X \\ C \\ E \end{pmatrix} - \begin{pmatrix} 0 \\ 0 \\ k_{CO_2} C \\ 0 \end{pmatrix} + \begin{pmatrix} DS^{in} \\ 0 \\ 0 \\ 0 \end{pmatrix} \tag{29}$$

$$\dot{O}(t) = \begin{bmatrix} -c_{O1} & -c_{O2} & 0 \end{bmatrix} \begin{bmatrix} \mu_1(S,O) \\ \mu_2(S,O,E) \\ \mu_3(S,O) \end{bmatrix} O - DO + k_{La}(O^* - O) \tag{30}$$

Where the state variables are:

S(t)	Substrate concentration (glucose) in the reactor;
x(t)	Biomass concentration (yeast) in the reactor;
C(t)	Concentration of the dissolved CO_2 in the reactor;
E(t)	Ethanol concentration in the reactor;
O(t)	Dissolved oxygen concentration in the reactor.

And further variables and constants are:

D(t)	Dilution rate considered as input;
$c_{ij} > 0$	Stoichiometric (or yield) coefficients corresponding to the production of one unit of biomass (i.e. yeast) in each reactor;
S^{in}	Glucose concentration in the feed;
O^*	Equilibrium concentration of dissolved oxygen;
k_{La}	Oxygen mass transfer constant;
$k_{CO_2} C(t)$	Gaseous CO_2 outflow rate proportional to *C(t)*.

The main objective is to keep the glucose concentration, which is considered as external substrate, close to the reference values using the dilution rate D(t) as manipulating function. For technical reasons, the input must be bounded. The model is based on a limited oxidation capacity, which is a function of the oxygen concentration in the liquid phase. If the oxidation capacity is sufficiently high to oxidize all glucose consumed, then no ethanol is produced. If in this situation the ethanol is present in the medium as well, then co-consumption of ethanol is possible. If not, then all glucose can be oxidized and the surplus glucose will be consumed according to the reductive metabolism, resulting in ethanol formation. The process of yeast growth on glucose with ethanol production is described by three metabolic reactions. All constants involved are positive. The specific growth rate, corresponding to the reaction - *respiratory growth on glucose,* is:

$$\mu_1(S,O)=\begin{cases} c_{11}^{-1}\dfrac{q_{s,\max}S}{S+Ks}\cdot\dfrac{O}{O+Kc} \;\; if \;\; \dfrac{q_{s,\max}S}{S+Ks}<\dfrac{q_{c,\max}}{a} \\ c_{11}^{-1}\dfrac{q_{c,\max}S}{a}\cdot\dfrac{O}{O+Kc} \;\; if \;\; \dfrac{q_{s,\max}S}{S+Ks}>\dfrac{q_{c,\max}}{a} \end{cases} \quad (31)$$

Where:

$q_{S,\max}$	Maximal specific uptake rate of glucosa;
$q_{C,\max}$	Maximal specific uptake rate of oxygen;
Ks	Saturation parameters for glucose uptake;
Kc	Saturation parameters for oxygen uptake;
$a = c_{01}\, c_{11}^{-1}$	Stoichiometric coefficient of oxygen.

If the oxidation capacity is sufficiently high to oxidize both ethanol and glucose, then their co-consumption is possible. In this case *respiratory growth on ethanol* reaction is considered with the respective specific growth rate:

$$\mu_2(S,O,E)=\frac{\mu_{e,\max}E}{Ke+E}\cdot\frac{Ki}{S+Ki}\cdot\frac{O}{O+\beta o} \quad (32)$$

Where:

$\mu_{e,\max}$	Maximal specific ethanol growth rate;
Ki	Inhibition parameter (free glucose inhibits ethanol uptake);
Ke	Saturation parameter for growth on ethanol;
βo	Saturation parameter for the free respiratory capacity available.

Finally the specific growth rate of the reaction *fermentative growth on glucose* is:

$$\mu_3(S,O)=\begin{cases} c_{13}^{-1}\dfrac{q_{s,\max}S}{S+Ks}\cdot\dfrac{Kc}{O+Kc}, & if \;\; \dfrac{q_{s,\max}S}{S+Ks}<\dfrac{q_{c,\max}}{a} \\ c_{13}^{-1}\left[\dfrac{q_{s,\max}S}{S+Ks}-\dfrac{q_{c,\max}}{a}\cdot\dfrac{O}{O+Kc}\right], & if \;\; \dfrac{q_{s,\max}S}{S+Ks}>\dfrac{q_{c,\max}}{a} \end{cases} \quad (33)$$

Since the growth capacity of microorganisms population is strongly limited, the specific growth rates are bounded. The upper bounds are:

$$\mu_1(S,O)\le\bar{\mu}_1:=c_{11}^{-1}\frac{q_{c,\max}}{a} \quad (34)$$

$$\mu_2(S,E,O)\le\bar{\mu}_2:=\mu_{e,\max} \quad (35)$$

$$\mu_3(S,O)\le\bar{\mu}_3:=c_{13}^{-1}\frac{q_{c,\max}}{a} \quad (36)$$

Usually, the exact values of these parameters are not available but the range of their variations is well known, [34]. Therefore the maximal growth capacity of the yeast population in each reaction is known. Furthermore, the upper bound of the biomass concentration *x(t)* is usually known in applications.

From the above findings, the model of the baker's yeast fermentation process is a special case of the general model of biochemical aerobic processes, and it will be used further.

V. Simulation Results

In this section we simulate the application of the adaptive neural control illustrated by Fig.2 to the baker's yeast fermentation process. The process model is given by equations (29)-(36). The following kinetic data, taken from [34], are used:

$q_{s,\max}=3.5\ \left[g_{gluc}g_{biomass}^{-1}h^{-1}\right]$

$q_{c,\max}=0.256\ \left[g_{O_2}g_{biomass}^{-1}h^{-1}\right],\ \mu_{e,\max}=0.17\left[h^{-1}\right]$

$Ks=0.2\ \left[g_{gluc}/l\right],\quad Kc=0.0001\ \left[g_{O_2}/l\right]$

$Ke=0.1\ \left[g/l\right],\quad Ki=0.1\ \left[g/l\right],\quad a=0.4142\ \left[g_{O_2}/g_{gluc}\right]$

$\beta o=0.003\ \left[g/l\right],\quad k_{La}=100h^{-1},\quad O^*=0.007\ \left[g/l\right]$

$S^{in}=10\ \left[g/l\right],\ K_{CO_2}=0,$

Where: g_{gluc}, g_{O_2} and $g_{biomass}$ denote gram glucose, gram oxygen and gram biomass respectively. The stoichiometric matrix K=[c_{ij}] in (29) and the vector $k_o=\left[-c_{O1}\quad -c_{O2}\quad 0\right]$ in (30) are:

$$K=\begin{bmatrix} -2.04 & 0 & -20 \\ 1 & 1 & 1 \\ 1.23 & 0.9 & 9.09 \\ 0 & -1.39 & 10 \end{bmatrix},\ k_o=\left[-0.83\quad -1.56\quad 0\right],$$

$D^*=0.2,\ D_{\max}=0.385.$

The initial values of the state variables are:

$S(0)=0.95;\qquad O(0)=0.066;\qquad X(0)=0.1;$

$C(0)=0.000325;\qquad E(0)=0.0001;$

The control objective is to regulate the glucose concentration *S(t)* to follow a constant reference concentration $S_{ref}=0.05$.

The control design can be divided in two stages –system identification (provided by RN-1) and control action (provided by RN-2), see Fig.1 and Fig.2. Both networks (RN-1, RN-2) are trained by the same BP learning rule. The topology and learning parameters of the neural identifier RN-1 are: (1, 5, 1), $\eta=0.1$, $\alpha=0.01$. Since the output of RN-1 *Y(k)* is a normalized signal, the plant output $Y_p(k)$ has to be normalized in the same range, to get compatible terms in the identification error, $E_i(k)=Y_p(k)-Y(k)$, required for the adaptation of the neural weights. The neural controller RN-2 has the topology (6, 5, 5, 1), which means that it has a second hidden layer with 5 neurons. The learning parameters for RN-2 are: $\eta=0.25$, $\alpha=0.01$. The period of discretization is kept in all simulations as $T_0=0.01$. The identification results are given in the next part.

A. System Identification

We choose a closed-loop strategy for initial system identification. A control input *U(.)* is generated by the λ-tracking control, [34], and it is supplied simultaneously as the plant input and as the input of RN-1. The λ-tracking control is an adaptive control strategy thoroughly discussed in [34]. It is a proportional output feedback of the error:

$$E(t) = Y_p(t) - R(t); U(t) = sat_{[0,U\max]}[K(t)E(t)]$$
$$K(t) = \delta \begin{cases} (|E(t)| - \lambda)r, if\ |E(t)| > \lambda \\ 0, if\ |E(t)| \leq \lambda \end{cases} \quad (37)$$

Where: $R(0) = 0.05$, $U_{max} = 0.0385$, $\lambda = 0.0025$, $\delta = 45$ and $r = 1$ are design parameters that influence the transient behavior of the closed-loop system crucially. Once the initial RN-1 is successfully trained, the control source is changed and now it is provided by the neural controller RN-2. The simulation results of system identification are show in Fig.3 together with the results of Direct Adaptive Control (DANC) without I-term.

B. Direct Adaptive Neural Proportional Control System Free of Perturbations

The graphical results are summarized in Fig.3a-h and illustrate the ideal (free of noise and offset) case. The simulations are performed over a period of 24 hours, which captures a typical period of the process behavior. Therefore Fig.3a and Fig.3b illustrate the initial phase of 5 h. and the remaining 5-24 h. respectively. Note that the control *Mean Squared Error* (MSE) at the end of the simulation run (24 h.) reached the value of 1.446% (Fig.3f) and the plant identification MSE reached 1.06%. Comparison of the output of the plant and the output of the neural identifier RN-1, the instantaneous control and identification errors, the control signal, and the systems states, issued by the neural identifier RN-1 and used for control, are depicted in Fig.3 c-h, respectively.

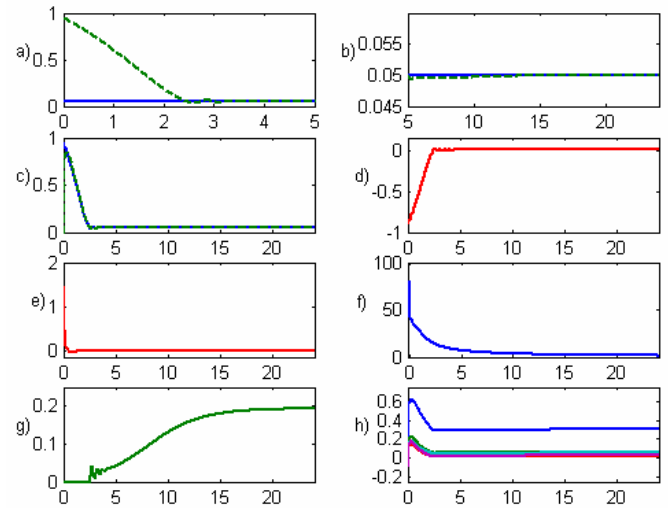

Fig. 3. Graphical results of the Proportional DANC. a) Comparison of the plant output and the set point signal in (0-5 hrs.); b) the same as a) in (5-25 hrs.); c) Comparison of the plant output and the output of the RN-1; d) Instantaneous error of control; e) Instantaneous error of identification; f) MSE of control; g) Control signal; h) System states.

The next simulations comprise the realistic case when noise is corrupting the plant output and an input offset is detected. A 10% white Gaussian noise, commonly used to simulate noisy measurement, is added to the output data and an offset equal of *Of(.)=0.02* is added to the plant input.

C. Lambda (λ)-Tracking Control System

For the purpose of comparison, plant operation with the λ-tracking control (see equations (37)) was equally simulated and the graphical results are depicted in the Fig. 4 and Fig. 5. Note that the introduced offset causes a displacement of the plant output (Fig. 4d) and a substantial increment of the control MSE (Fig.5b), which reached the value of 2.358% at the end of the simulation run (24 h.). The results obtained with DANC without I-term (Fig.6 and Fig.7) show the same tendency, though the performance is slightly better, i.e. the MSE at about 24 h. goes to 2.354% (Fig.7d). Nevertheless, it has to be recognized that the λ-tracking controller would manage to keep the tracking if the tolerated error is increased, which is one of the design parameters.

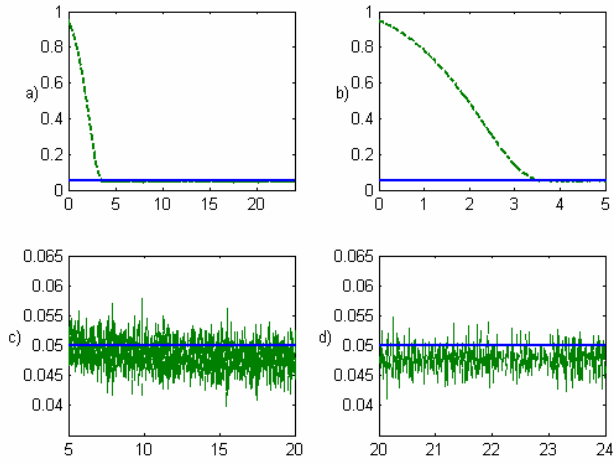

Fig. 4. Graphical results of the λ-tracking control with 10% white measurement noise and an offset in the plant input; a) Comparison of the plant output and the set point signal in (0-25 hrs.); b) the same as a) in (0-5 hrs.); c) the same as a) in (5-20 hrs.); d) the same in (20-24 hrs.)

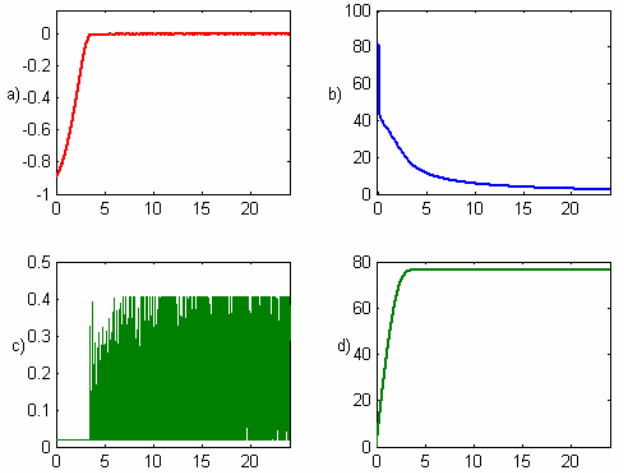

Fig. 5. Graphical results of the λ-tracking control with 10% white noise and offset. a) Instantaneous error of control; b) MSE of control; c) Control signal; d) Adaptive gain.

D. *Direct Adaptive Neural Proportional Control System Perturbed by Measurement Noise and Input Offset*

The results of DANC without I-term perturbed by measurement noise and offset in the plant input are shown on Fig. 6, and Fig.7. The results are better then the previous case but the perturbation effect is still substantial.

E. *Direct Adaptive Neural Control System with I-term, Perturbed by Measurement Noise and Input Offset*

The graphical results are summarized in Fig.8, Fig.9 and Fig.10. The simulations are performed assuming that the I-term gain (see equation (20)) is set to *Ki=0.09*. DANC with I-term exhibits better performance with respect to the previous control structures discussed

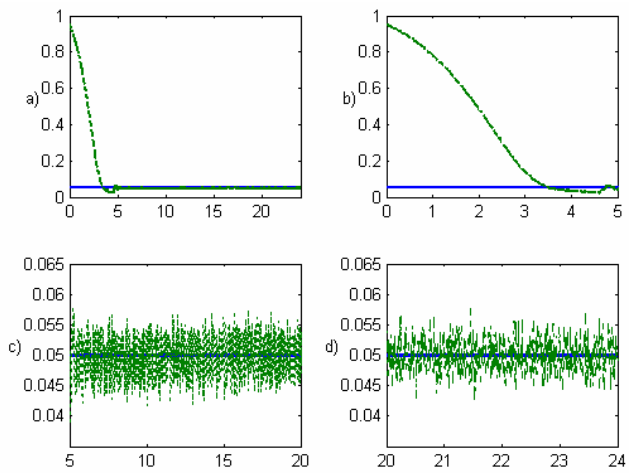

Fig. 6. Graphical results of the Proportional DANC with 10% white measurement noise and an offset in the plant input. a) Comparison of the plant output and the set point signal in the time interval 0-25 hrs.; b) the same as a) in the time interval 0-5 hrs.; c) the same as a) in the time interval 5-20 hrs.; d) the same as a) in the time interval 20-24 hrs.

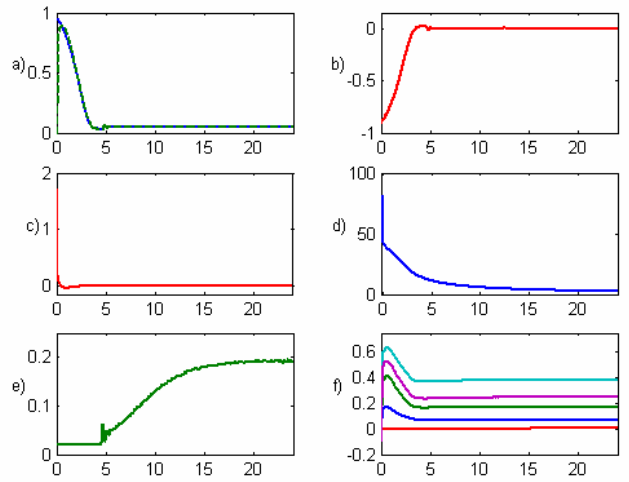

Fig. 7. Graphical results of the Proportional DANC with 10% white measurement noise and an offset in the plant input. a) Comparison of the plant output and the output of the identification RN-1; b) Instantaneous error of control; c) Instantaneous error of identification; d) MSE of control; e) Control signal; f) Five system states, issued from the identification RN-1, applied for control.

Note that the control MSE at the end of the simulation run (24 h.) reached the value of 1.525% (Fig.9d), which is quite close to the ideal case (Fig.3f). The other variables – biomass, ethanol, dissolved oxygen and carbon dioxide, tend to their steady states within 17, 30, 15, 20 h. respectively (see Fig.10). These results ensure that the DANC with I-term is able to compensate constant offsets, it can reduce substantially the noise in the control system and thus, a good set-point tracking is achieved.

VI. Conclusion

In this work we reported the design of a direct adaptive neural control system with and without I-term and its application to the class of continuous biochemical aerobic processes represented by a mathematical model of the baker's yeast fermentation.

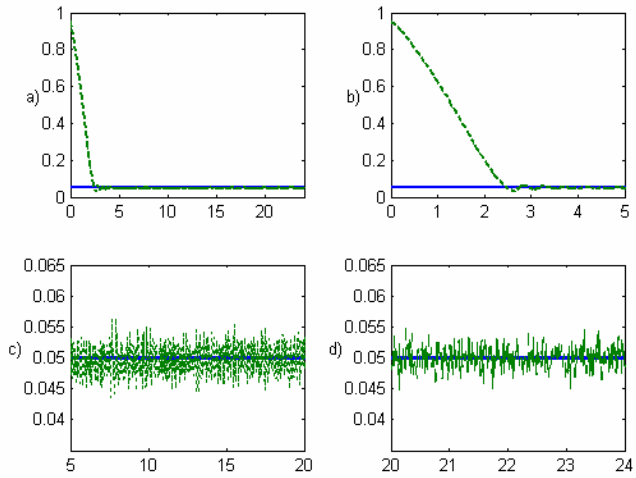

Fig. 8. Graphical results of I-term DANC with 10% white measurement noise and an offset in the plant input. a) Comparison of the plant output and the set point signal in the time interval 0-25 hrs.; b) the same as a) in the time interval 0-5 hrs.; c) the same as a) in the time interval 5-20 hrs.; d) the same as a) in the time interval 20-24 hrs.

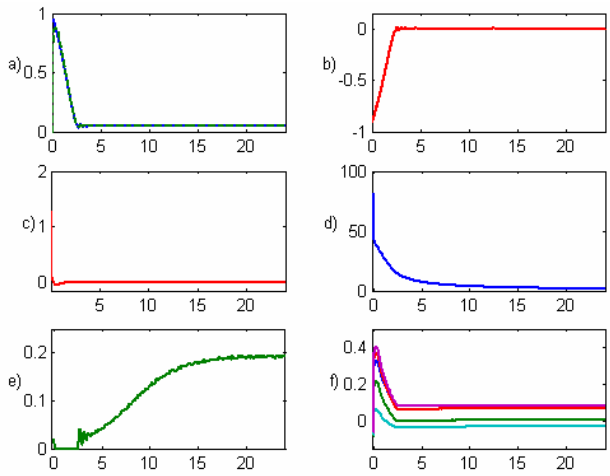

Fig. 9. Graphical results of I-term DANC with 10% white measurement noise and an offset in the plant input. a) Comparison of the plant output and the output of the identification RN-1; b) Instantaneous error of control; c) Instantaneous error of identification; d) MSE of control; e) Control signal; f) Five system states of RN-1.

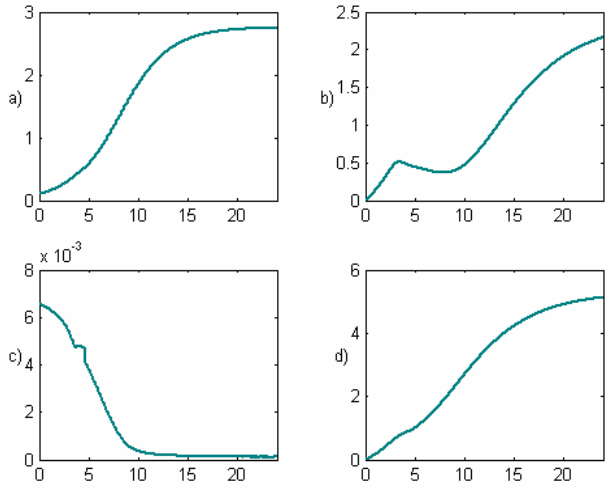

Fig. 10. Graphical results of the I-term DANC with 10% measurement noise and an offset in the plant input. a) Biomass produced; b) Ethanol; c) Dissolved oxygen; d) Carbon dioxide [g/l].

The system contains a neural identifier and a neural controller, based on the Recurrent Trainable Neural Network model. The performance of the two Direct Adaptive Neural Control schemes (with and without I-term) was evaluated on its own and relatively to the observed behavior of λ-tracking control. All control schemes studied work successfully for the ideal (free of noise and offset) case. However, the DANC with an integral term proved to exhibit favorable behavior in the presence of considerable noise and constant offset.

REFERENCES

[1] W. T. Miller III, R. S. Sutton, and P. J. Werbos, *Neural Networks for Control*. London: MIT Press, 1992.

[2] K. J. Hunt, D. Sbarbaro, R. Zbikowski, and P. J. Gawthrop, "Neural Networks for Control Systems - A Survey," *Automatica*, vol. 28, pp. 1083-1112, 1992.

[3] J.B.D. Cabrera, and K.S. Narendra, "Issues in the Application of Neural Networks for Tracking Based on Inverse Control", *IEEE Transactions on Automatic Control, Special Issue on Neural Networks for Control, Identification, and Decision Making,* vol. 44, No 11, 2007-2027, 1999.

[4] Ch. Lingji, and K.S. Narendra, "Nonlinear Adaptive Control Using Neural Networks and Multiple Models", *Automatica, Special Issue on Neural Network Feedback Control,* vol. 37, No 8, 1245-1255, 2001.

[5] S.S. Ge, J. Zhang, and T.H. Lee, "Direct MNN Control of Continuous Stirred Tank Reactor Based on Input-Output Model, "*Proc. of the 41-st SICE Annual Conference, SICE 5-7 Aug. 2002,* vol. 5, pp. 2770-2775, 2002.

[6] Y. Tan, X. Dang, F. Liang, and Ch. Yi Su, "Dynamic Wavelet Neural Network for Nonlinear Dynamic System Identification," *Proc. of the 2000 IEEE Int. Conf. on Control Applications, 25-27 Sept. 2000,* pp. 214-219, 2000.

[7] O. de Jesus, A. Pukrittayakamee, and M.T. Hagan, "A Comparison of Neural Network Control Algorithms," *Proc. of the Int. Joint Conference on Neural Networks, IJCNN'01, 15-19 July 2001*, vol. 1, pp. 521-526, 2001.

[8] Ch. Zhang, and H. Hu, "An Evolved Recurrent Neural Network and Its Application in the State Estimation of the CSTR System," *Proc. of the IEEE Int. Conf. on Systems, Man and Cybernetics, 10-12 Oct. 2005*, vol. 3, pp. 2139-2143, 2005.

[9] S. Piche, B. Sayyar-Rodsari, D. Johnson, and M. Gerules, "Nonlinear Model Predictive Control Using Neural Networks," *IEEE Control Systems Magazine*, vol. 20, issue 3, pp. 53-62, June 2000.

[10] H.J. Chen, and R. Chen, "Adaptive Control of Non-Affine Nonlinear Systems Using Radial Basis Function Neural Network," *Proc. of the IEEE Int. Conf. on Networking, Sensing and Control*, vol. 2, pp. 1195-1200, 2004.

[11] Y. Tan, X. Dang, and A. van Cauwenberghe, "Generalized Nonlinear PID Controller Based on Neural Networks," *Proc. of the Int Conf. on Information, Decision and Control, IDC'99, 8-10 Febr. 1999*, pp. 519-524, 1999.

[12] Y. Dongyong, J. Jingping, and Y. Yuzo, "Distal Supervised Control and Its Application to CSTR Systems," *Proc of the 39-th SICE Annual Conf. Intern. Session Papers, 26-28 July 2000*, pp. 209-214, 2000.

[13] J.J. Govindhasamy, S.F. McLoone, and G.W. Irwin, "Srecond-Order Training of adaptive Critics for Online Process Control," *IEEE Trans. On Systems, Man and Cybernetics, Part B*, vol. 35, issue 2, pp. 381-385, Apr. 2005.

[14] Q. Wu, Y.J. Wang, Q.M. Zhu, and K. Warwick, "Neurofuzzy Model Based Iinf.Predictive Control of nonlinear CSTR System," *Proc. of the IEEE Int. Conf. on Control Applications, 18-20 Sept. 2002, Glasgow, Scotland, U.K.,*, vol. 1, pp. 59-64, 2002.

[15] M.A. Hussain, and P.Y. Ho, "Adaptive Sliding Mode Control with Neural Network Based Hybrid Models," *Journal of Process Control*, vol. 14, No 2, pp. 157-176, March 2004.

[16] J. Madar, J. Abonyi, and F. Szeifert, "Feedback Linearizing Control Using Hybrid Neural Networks Identified by Sensitivity Approach," *Engineering Applications of Artificial Intelligence*, vol. 18, issue 3, pp. 343-351, Apr. 2005.

[17] C.W. Ng, and M.A. Hussain, "Hybrid Neural Network-Prior Knowledge Model in Temperature Control of a Semi-Batch Polymerization Process," *Chemical Engineering and Processing*, vol. 43, issue 4, pp. 559-570, Apr. 2004.

[18] T.K. Chang, D.L. Yu, and D.W. Yu, "Neural Network Model Adaptation and its Application to Process Control," *Advanced Engineering Informatics*, vol. 18, issue 1, pp. 1-8, Jan. 2004.

[19] P. Holubar, L. Zani, M. Hager, W. Froschl, Z. Radak, and R. Braun," *Water Research*, vol. 36, issue 10, pp. 2582-2588, May 2002.

[20] T.D. Knapp, H.M. Budman, and G. Broderick, "Adaptive Control of a CSTR with a Neural Network Model," Journal of Process Control, vol. 11, issue 1, pp. 53-68, Febr. 2001.

[21] C. Kambhampati, J.D. Mason, and K. Warwick, "A Stable One-Step-Ahead Predictive Control of Non-Linear Systems," *Automatica*, vol. 36, issue 4, pp. 485-495, Apr. 2000.

[22] Y. Tan, and A. van Cauwenberghe, "Nonlinear One-Step-Ahead Control Using Neural Networks: Control Strategy and Stability Design," *Automatica*, vol. 32, issue 12, pp. 1701-1706, Dec. 1996.

[23] S. Haykin, *Neural Networks, A Comprehensive Foundations*, 2-nd Edition, Englewood Cliffs, NJ: Prentice Hall, 1999.

[24] S. Chen, and S. A. Billings, " Neural Networks for Nonlinear Dynamics System Modeling and Identification," *International Journal of Control,* vol. 56, pp. 263-289, 1992.

[25] K. S. Narendra, and K. Parthasarathy, "Identification and Control of Dynamic Systems using Neural Networks," *IEEE Transactions on Neural Networks,* vol. 1, No 1, pp. 4-27, 1990.

[26] L. Jin, and M. Gupta, "Stable Dynamic Backpropagation Learning in Recurrent Neural Networks," *IEEE Transactions on Neural Networks*, vol. 10, pp. 1321-1334, 1999.

[27] P. Frasconi, M. Gori, and G. Soda, "Local Feedback Multilayered Networks," *Neural Computation*, vol. 4, pp. 120-130, 1992.

[28] I.S. Baruch, P. Georgieva, J. Barrera-Cortes, and S.F. de Azevedo, “Adaptive Recurrent Neural Network Control of Biological Wastewater Treatment,” *International Journal of Intelligent Systems, Special Issue on Soft Computing for Modelling, Simulation, and Control of Nonlinear Dynamical Systems (Guest Eds.: O. Castillo, and P. Melin)*, vol. 20, No 2, pp. 173-193, Febr. 2005.

[29] I.S. Baruch, and R. Garrido, “A Direct Adaptive Neural Control Scheme with Integral Terms,” *International Journal of Intelligent Systems, Special Issue on Soft Computing for Modelling, Simulation, and Control of Nonlinear Dynamical Systems (Guest Eds.: O. Castillo, and P. Melin)*, vol. 20, No 2, pp. 213-224, Febr. 2005.

[30] I. Baruch, J. Barrera, J. Perez, and L. A. Hernandez, "An Adaptive Neural Control of Fed-Batch Fermentation Processes", *Proc. of the 2003 Int. IEEE Conf. on Control Applications,* Istanbul, Turkey, paper CF-001898, June 23-25, 2002.

[31] F. Nava, I. Baruch, A.Pozniak and B. Nenkova, “Stability Proofs of Advanced Recurrent Neural Network Topology and Learning,” *Comptes Rendus (Proceedings of the Bulgarian Academy of Sciences),* ISSN 0861-1459, vol. 57, No 1, pp. 27-32, 2004.

[32] P. Georgieva, and M. Ignatova, “Implementation of Robust Control Theory to a Continuous Stirred Tank Bioreactor”, *Bioprocess Engineering*, vol. 22, pp. 563-568, 2000.

[33] J. D. Boskovic, and K. S. Narendra, “Comparison of Linear, Nonlinear and Neural-Network-Based Adaptive Controllers for a Class of Fed-Batch Fermentation Processes,” *Automatica*, vol. 31, No 6, pp. 817-840, 1995.

[34] P. Georgieva, A. Ilchmann, and M.F. Weiring, ”Modeling and Adaptive Control of Aerobic Continuous Stirred Tank Reactors”, *European Journal of Control*, vol. 1, pp.1-16, 2001.

Ieroham S. Baruch is born in Sofia, Bulgaria. Hi obtained his PhD in Technical Sciences in the Higher Mechanical and Electrical Engineering Institute, Sofia, Bulgaria, 1974. From 1967 to 1997 he worked as a Research Associate and Associate Professor in the Institute of Engineering Cybernetics and Robotics, Bulgarian Academy of Sciences, Sofia, Bulgaria. In 1998 he joined the Department of Automatic Control, CINVESTAV-IPN, Mexico City, Mexico, where he is a Professor of Automatic Control till now. Hi is author and co-author of more that 100 scientific papers. His short biography is published in the 23rd Edition of Marquis Who’sWho in the World, 2006. His research interests include identification and control of mechanical and biotechnological plants using neural networks and fuzzy-neural multi-models.

Petia Georgieva is currently a Professor at the Department of Electronics and Telecommunications, IEETA, University of Aveiro, Portugal. She obtained MS and PhD degrees in Control Engineering from the Technical University of Sofia, Bulgaria. Her interests are in the areas of modeling, optimization and process control.

Josefina Barrera-Cortes is currently a Professor in the Department of Biotechnology and Bioengineering at CINVESTAV-IPN, México City, Mexico, from 1996. She obtained a MS degree in Chemical Engineering at the UNAM, Mexico in 1987 and a PhD degree in Engineering Sciences (Processes Engineering) from the University of North Paris, France in 1996. Her interests are in the areas of modeling and control of Biological processes.