We are IntechOpen, the world's leading publisher of Open Access books
Built by scientists, for scientists

6,900
Open access books available

185,000
International authors and editors

200M
Downloads

154
Countries delivered to

Our authors are among the

TOP 1%
most cited scientists

12.2%
Contributors from top 500 universities

CLARIVATE ANALYTICS
BOOK CITATION INDEX
INDEXED

WEB OF SCIENCE™

Selection of our books indexed in the Book Citation Index
in Web of Science™ Core Collection (BKCI)

Interested in publishing with us?
Contact book.department@intechopen.com

Numbers displayed above are based on latest data collected.
For more information visit www.intechopen.com

Centralized Distributed Parameter Bioprocess Identification and I-Term Control Using Recurrent Neural Network Model

Ieroham Baruch[1], Eloy Echeverria-Saldierna[1] and Rosalba Galvan-Guerra[2]
[1]CINVESTAV-IPN, Mexico City, Department of Automatic Control,
[2]ESIMEZ-IPN, Mexico City, Department of Control and Automation Engineering,
Mexico

1. Introduction

In the last two decades new identification and control tools, like Neural Networks (NN), have been used for biotechnological plants (Boskovic & Narendra, 1995). Among several possible network architectures the ones most widely used are the Feedforward NN (FFNN) and the Recurrent NN (RNN), (Haykin, 1999). The main NN property namely the ability to approximate complex non-linear relationships without prior knowledge of the model structure makes them a very attractive alternative to the classical modeling and control techniques. This property has been proved for both types of NNs by the universal approximation theorem (Haykin, 1999). The preference given to NN identification with respect to the classical methods of process identification is clearly demonstrated in the solution of the "bias-variance dilemma" (Haykin, 1999). The FFNN and the RNN have been applied for Distributed Parameter Systems (DPS) identification and control too. In (Deng & Li, 2003; Deng et al. 2005; Gonzalez et al, 1998), an intelligent modeling approach is proposed for Distributed Parameter Systems (DPS). In (Gonzalez et al, 1998), it is presented a new methodology for the identification of DPS, based on NN architectures, motivated by standard numerical discretization techniques used for the solution of Partial Differential Equations (PDE). In (Padhi et al, 2001), an attempt is made to use the philosophy of the NN adaptive-critic design to the optimal control of distributed parameter systems. In (Padhi & Balakrishnan, 2003) the concept of proper orthogonal decomposition is used for the model reduction of DPS to form a reduced order lumped parameter problem. In (Pietil & Koivo, 1996), measurement data of an industrial process are generated by solving the PDE numerically using the finite differences method. Both centralized and decentralized NN models are introduced and constructed based on this data. The multilayer feedforward NN realizing a NARMA model for systems identification has the inconvenience that it is sequential in nature and require input and feedback tap-delays for its realization. In (Baruch et al, 2002; Baruch et al, 2004; Baruch et al, 2005a; Baruch et al, 2005b; Baruch et al, 2007a; Baruch et al, 2007b; Baruch et al, 2008; Baruch & Mariaca-Gaspar, 2009; Baruch & Mariaca-Gaspar, 2010), a new completelly parallel canonical Recurrent Trainable NN (RTNN) architecture, and a dynamic BP learning algorithm has been applied for systems identification and control of nonlinear

plants with equal input/output dimensions, obtaining good results. The RTNN do not need the use of tap delays and has a minimum number of weights due to its Jordan canonical structure. In the present paper, this RTNN model will be used for identification, and direct control, of a digestion anaerobic DPS of wastewater treatment, (Aguilar-Garnica, 2006), modeled by PDE/ODE (Ordinary Differential Equations), and simplified using the Orthogonal Collocation Method (OCM) in four collocation points of the fixed bed and one more- for the recirculation tank. We needs to use this simplified ODE mathematical model as an input/output data generator for RTNN BP learning instead of the real plant. Furthermore the mathematical model description of the plant help us to understand the work and the meaning of all process variables of this complex biotechnological plant. Here the plant identification by means of RTNN BP learning will be changed by the RTNN Levenberg Marquardt (L-M) second order learning, (Baruch et al, 2009). This distributed nonlinear parameter plant, described by ODE, has excessive high-dimensional measurements which means that the plant output dimension is greater than the plant control input one (rectangular system), requiring to use learning by data fusion technique and special reference choice. Furthermore the used control laws are extended with an integral term, (Baruch et al.2005b; Baruch et al, 2007b), so to form an integral plus state control action, capable to speed up the reaction of the control system and to augment its resistance to noise.

2. Mathematical description of the anaerobic digestion bioprocess plant

The development of the anaerobic digestion process PDE model is based on the two-step (acidogenesis-methanization) mass-balance and bacterial kinetics involving the Monod equations of the specific growth rates (eq. 1-4). The model incorporates electrochemical equilibria in order to include the alkalinity which has to play a central role in the related monitoring and control strategy of a wastewater treatment plant. The dynamics of the species in the recirculation tank is described by ODE (eq. 7). The parameters of this model are obtained by parameter identification and validation (see Bernard et al., 2001). The biochemical nature of the processes of waste degradation is described in (Schoefs et al., 2003, and Bernard et al., 2001) and in cited bibliography. The physical meaning of all variables and constants (also its values), are summarized in Table 1. The complete analytical model of wastewater treatment anaerobic bioprocess (see Fig. 1), taken from (Schoefs et al. 2003 and Aguilar-Garnica et al., 2006), could be described by the following system of PDE:

$$\frac{\partial X_1}{\partial t} = \left(\mu_1 - \varepsilon D\right) X_1, \quad \mu_1 = \mu_{1\max} \frac{S_1}{K_{S1} X_1 + S_1}, \tag{1}$$

$$\frac{\partial X_2}{\partial t} = \left(\mu_2 - \varepsilon D\right) X_2, \quad \mu_2 = \mu_{2s} \frac{S_1}{K_{S2} X_2 + \dfrac{S_2^2}{K_{I2}}}, \tag{2}$$

$$\frac{\partial S_1}{\partial t} = \frac{E_z}{H^2} \frac{\partial^2 S_1}{\partial z^2} - D \frac{\partial S_1}{\partial t} - k_1 \mu_1 X_1, \tag{3}$$
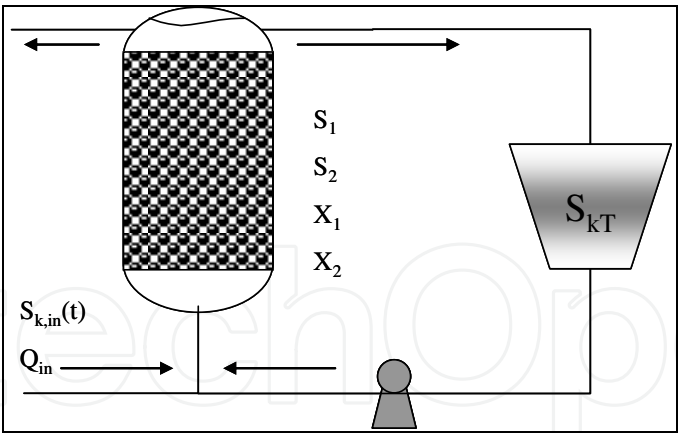
Fig. 1. Block-diagram of anaerobic digestion bioreactor

Variable	Units	Name	Value
z	z∈[0,1]	Space variable	
t	D	Time variable	
E_z	m²/d	Axial dispersion coefficient	1
D	1/d	Dilution rate	0.55
H	m	Fixed bed length	3.5
X_1	g/L	Concentration of acidogenic bacteria	
X_2	g/L	Concentration of methanogenic bacteria	
S_1	g/L	Chemical Oxygen Demand	
S_2	mmol/L	Volatile Fatty Acids	
ε		Bacteria fraction in the liquid phase	0.5
k_1	g/g	Yield coefficients	42.14
k_2	mmol/g	Yield coefficients	250
k_3	mmol/g	Yield coefficients	134
μ_1	1/d	Acidogenesis growth rate	
μ_2	1/d	Methanogenesis growth rate	
μ_{1max}	1/d	Maximum acidogenesis growth rate	1.2
μ_{2s}	1/d	Maximum methanogenesis growth rate	0.74
K_{1s}'	g/g	Kinetic parameter	50.5
K_{2s}'	mmol/g	Kinetic parameter	16.6
K_{I2}'	mmol/g	Kinetic parameter	256
Q_T	m³/d	Recycle flow rate	0.24
V_T	m³	Volume of the recirculation tank	0.2
S_{1T}	g/L	Concentration of Chemical Oxygen Demand in the recirculation tank	
S_{2T}	mmol/L	Concentration of Volatile Fatty Acids in the recirculation tank	
Q_{in}	m³/d	Inlet flow rate	0.31
V_B	m³	Volume of the fixed bed	1
V_{eff}	m³	Effective volume tank	0.95
$S_{1,in}$	g/L	Inlet substr. Concentration	
$S_{2,in}$	mmol/L	Inlet substr. Concentration	

Table 1. Summary of the variables in the plant model

$$\frac{\partial S_2}{\partial t} = \frac{E_z}{H^2}\frac{\partial^2 S_2}{\partial z^2} - D\frac{\partial S_2}{\partial t} - k_2\mu_1 X_1, \tag{4}$$

$$S_1(0,t) = \frac{S_{1,in}(t) + RS_{1T}}{R+1}, \quad S_2(0,t) = \frac{S_{2,in}(t) + RS_{2T}}{R+1}, \quad R = \frac{Q_T}{DV_{eff}}, \tag{5}$$

$$\frac{\partial S_1}{\partial z}(1,t) = 0, \quad \frac{\partial S_2}{\partial z}(1,t) = 0, \tag{6}$$

$$\frac{dS_{1T}}{dt} = \frac{Q_T}{V_T}\big(S_1(1,t) - S_{1T}\big), \quad \frac{dS_{2T}}{dt} = \frac{Q_T}{V_T}\big(S_2(1,t) - S_{2T}\big). \tag{7}$$

For practical purpose, the full PDE bioprocess model is reduced to an ordinary differential equations system using the OCM (Bialecki & Fairweather, 2001). The precision of the orthogonal collocation method of approximation of the PDE model depended on the number of measurement (collocation) points, but the approximation is always exact in that points. If the number of points is very high and the point positions are chosen inappropriately, the ODE model could loose identifiability.

Here, the ODE plant model is used as a plant data generator, illustrating the centralized neural identification and control of the DPS, so, the point number not need to be too high. Our reduced order model have only four points, (0.2H, 0.4 H, 0.6H, 0.8H), but it generated 18 measured variables as: X_1 (acidogenic bacteria), X_2 (methanogenic bacteria), S_1 (chemical oxygen demand) and S_2 (volatile fatty acids), and the following variables in the recirculation tank: S_{1T} (chemical oxygen demand) and S_{2T} (volatile fatty acids). So the plant input/output dimensions are M=2, L=18. The reference set points generated for all that variables keep the form but differ in amplification due to its position. The plant ODE system model, obtained by OCM is described by the following system of ODE:

$$\frac{dX_{1,i}}{dt} = \big(\mu_{1,i} - \varepsilon D\big)X_{1,i}, \quad \frac{dX_{2,i}}{dt} = \big(\mu_{2,i} - \varepsilon D\big)X_{2,i}, \tag{8}$$

$$\frac{dS_{1,i}}{dt} = \frac{E_z}{H^2}\sum_{j=1}^{N+2} B_{i,j}S_{1,j} - D\sum_{j=1}^{N+2} A_{i,j}S_{1,j} - k_1\mu_{1,i}X_{1,i}, \tag{9}$$

$$\frac{dS_{2,i}}{dt} = \frac{E_z}{H^2}\sum_{j=1}^{N+2} B_{i,j}S_{1,j} - D\sum_{j=1}^{N+2} A_{i,j}S_{2,j} - k_2\mu_{1,i}X_{2,i} - k_3\mu_{2,i}X_{2,i}, \tag{10}$$

$$\frac{dS_{1T}}{dt} = \frac{Q_T}{V_T}\big(S_{1,N+2} - S_{1T}\big), \quad \frac{dS_{2T}}{dt} = \frac{Q_T}{V_T}\big(S_{2,N+2} - S_{2T}\big), \tag{11}$$

$$S_{k,1} = \frac{1}{R+1}S_{k,in}(t) + \frac{R}{R+1}S_{kT}, \quad S_{k,N+2} = \frac{K_1}{R+1}S_{k,in}(t) + \frac{K_1 R}{R+1}S_{kT} + \sum_{i=1}^{N+1}K_i S_{k,i}, \tag{12}$$

$$K_1 = \frac{A_{N+2,1}}{A_{N+2,N+2}}, \quad K_i = \frac{A_{N+2,i}}{A_{N+2,N+2}}, \tag{13}$$

$$A = \Lambda \phi^{-1}, \quad \Lambda = \left[\omega_{m,l}\right] = (l-1)z_m^{l-2}, \tag{14}$$

$$B = \Gamma \phi^{-1}, \quad \Gamma = \left[\tau_{m,l}\right], \quad \tau_{m,l} = (l-1)(l-2)z_m^{l-3}, \quad \phi_{m,l} = z_m^{l-1}, \tag{15}$$

$$i = 2,\ldots N+2, \quad m,l = 1,\ldots N+2. \tag{16}$$

The reduced plant model (8)-(16), could be used as unknown plant model which generate input/output process data for centralized adaptive neural identification and control system design, based on the concepts, given in (Baruch et al., 2008; Baruch & Mariaca-Gaspar, 2009).

3. Description of the RTNN topology and learning

The more general BP rule for single neuron learning is the delta rule of Widrow and Hoff, given in Haykin, 1999. If we define the cost function of learning as:

$$\xi(k) = (1/2)e^2(k), e(k) = t(k) - y(k)$$

where $\xi(k)$ is the cost function, e(k) is the neuron output error, y(k) is the neuron output signal, t(k) is the neuron output target signal, w(k) is the neuron weight, x(k) is the neuron input, we could write the following delta rule of neuron learning as:

$$w(k+1) = w(k) + \Delta w(k), \Delta w(k) = \eta e(k)x(k)$$

We could generalise this delta rule and applied it for learning of multilayer feedforward, recurrent or mixed neural networks if we design and draw its topology in block form. Then using the diagrammatic method (see Wan & Beaufays, 1996) we could design the adjoint NN topology. The NN topology is used to execute the forward pass of the BP learning so to compute predictions of the input signals of NN weight blocks. The adjoint NN topology is used to execute the backward pass of the BP learning so to compute predictions of the error signals of the outputs of that NN weight blocks. So having both predictions we could execute the delta learning rule layer by layer and weight by weight. In the following part we will apply this simple methodology for the two layered RTNN topology and learning given in vector matrix form where the delta rule is generalized as vector product of the local error and local input RTNN vectors.

3.1 RTNN topology and recursive BP learning

The block-diagrams of the RTNN topology and its adjoint, obtained by means of the diagrammatic method of (Wan & Beaufays, 1996), are given on Fig. 2, and Fig. 3. Following Fig. 2, and Fig. 3, we could derive the dynamic BP algorithm of its learning based on the RTNN topology and its adjoined using the generalized delta rule, given above. The RTNN topology is described in vector-matrix form as:

$$X(k+1) = AX(k) + BU(k), B^T = |B_1 \quad B_2|, U^T = |U_1 \quad U_2| \tag{17}$$

$$V(k) = CZ(k), C = \begin{vmatrix} C_1 & C_0 \end{vmatrix}, Z^T = \begin{vmatrix} Z_1 & Z_2 \end{vmatrix}, Z_1(k) = G[X(k)], Y(k) = F[V(k)] \tag{18}$$

$$A = block - diag(A_i), |A_i| < 1 \tag{19}$$

The BP learning is described in the following general form:

$$W(k+1) = W(k) + \eta \Delta W(k) + \alpha \Delta W(k-1) \tag{20}$$

Using the adjoint RTNN we could derive the BP learning for the RTNN weights applying the generalized delta rule as vectorial products of input and error predictions, as:

$$\Delta C(k) = E_1(k)Z^T(k), \Delta B(k) = E_3(k)U^T(k), \Delta A(k) = E_3(k)X^T(k) \tag{21}$$

$$Vec(\Delta A(k)) = E_3(k) \otimes X(k) \tag{22}$$

Where the error predictions are obtained from the adjoint RTNN as follows:

$$E(k) = T(k) - Y(k), E_1(k) = F'[Y(k)]E(k), F'[Y(k)] = [1 - Y^2(k)] \tag{23}$$
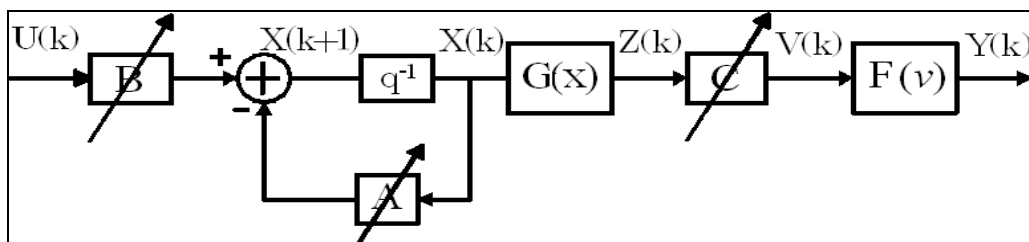
Fig. 2. Block diagram of the RTNN model
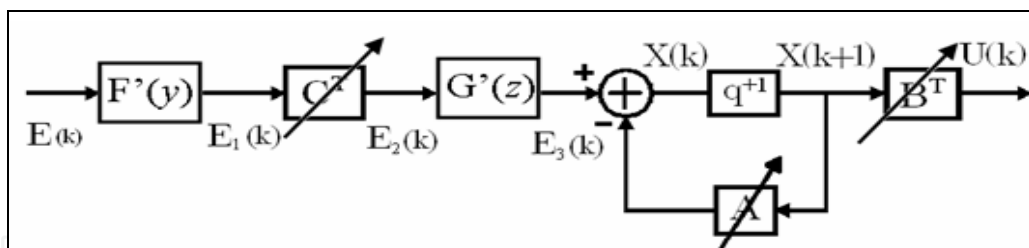
Fig. 3. Block diagram of the adjoint RTNN model

$$E_2(k) = C^T(k)E_1(k), E_3(k) = G'[Z(k)]E_2(k), G'[Z(k)] = [1 - Z^2(k)] \tag{24}$$

Here: X, Y, U are state, augmented output, and input vectors with dimensions N, (L+1), (M+1), respectively, where Z_1 and U_1 are the (Nx1) output and (Mx1) input of the hidden layer; the constant scalar threshold entries are $Z_2 = -1$, $U_2 = -1$, respectively; V is a (Lx1) pre-synaptic activity of the output layer; T is the (Lx1) plant output vector, considered as a RTNN reference; A is (NxN) block-diagonal weight matrix; B and C are [Nx(M+1)] and [Lx(N+1)]- augmented weight matrices; B_0 and C_0 are (Nx1) and (Lx1) threshold weights of the hidden and output layers; F[·], G[·] are vector-valued tanh(·)-activation functions with corresponding dimensions; F'[·], G'[·] are the derivatives of these tanh(·) functions, computed by (23), (24); W is a general weight, denoting each weight matrix (C, A, B) in the

RTNN model, to be updated; ΔW (ΔC, ΔA, ΔB), is the weight correction of W; η, α are learning rate parameters; ΔC is a weight correction of the learned matrix C; ΔB is a weight correction of the learned matrix B; ΔA is a weight correction of the learned matrix A; the diagonal of the matrix A is denoted by Vec(·) and equation (22) represents its learning as an element-by-element vector products; E, E_1, E_2, E_3, are error vectors with appropriate dimensions, predicted by the adjoint RTNN model, given on Fig.3. The stability of the RTNN model is assured by the activation functions (-1, 1) bounds and by the local stability weight bound condition, given by (19). Below it is given a theorem of RTNN stability which represented an extended version of Nava's theorem, (Baruch et al., 2008; Baruch & Mariaca-Gaspar, 2009; Baruch & Mariaca-Gaspar, 2010).

Theorem of stability of the BP RTNN. Let the RTNN with Jordan Canonical Structure is given by equations (17)-(19) (see Fig.2) and the nonlinear plant model, is as follows:

$$X_p(k+1) = G[X_p(k), U(k)], Y_p(k) = F[X_p(k)] \tag{25}$$

where: $\{Y_p(\cdot), X_p(\cdot), U(\cdot)\}$ are output, state and input variables with dimensions L, N_p, M, respectively; F(·), G(·) are vector valued nonlinear functions with respective dimensions. Under the assumption of RTNN identifiability made, the application of the BP learning algorithm for A(·), B(·), C(·), in general matricial form, described by equation (20)-(24), and the learning rates η (k), α (k) (here they are considered as time-dependent and normalized with respect to the error) are derived using the following Lyapunov function:

$$L(k) = L_1(k) + L_2(k), \tag{26}$$

Where: $L_1(k)$ and $L_2(k)$ are given by:

$$L_1(k) = \frac{1}{2} e^2(k),$$

$$L_2(k) = tr\left(\widetilde{W}_A(k) \widetilde{W}_A^T(k)\right) + tr\left(\widetilde{W}_B(k) \widetilde{W}_B^T(k)\right) + tr\left(\widetilde{W}_C(k) \widetilde{W}_C^T(k)\right);$$

Here: $\widetilde{W}_A(k) = \hat{A}(k) - A^*$, $\widetilde{W}_B(k) = \hat{B}(k) - B^*$, $\widetilde{W}_C(k) = \hat{C}(k) - C^*$, are vectors of the weight estimation error; (A^*, B^*, C^*), $(\hat{A}(k), \hat{B}(k), \hat{C}(k))$ denote the ideal neural weight and the estimate of the neural weight at the k-th step, respectively, for each case. Then the identification error is bounded, i.e.:

$$L(k+1) = L_1(k+1) + L_2(k+1) < 0,$$
$$\Delta L(k+1) = L(k+1) - L(k);$$

where the condition for $L_1(k+1) < 0$ is that:

$$\frac{\left(1 - \frac{1}{\sqrt{2}}\right)}{\psi_{max}} < \eta_{max} < \frac{\left(1 + \frac{1}{\sqrt{2}}\right)}{\psi_{max}};$$

and for $L_2(k+1)<0$ we have:

$$\Delta L_2\left(k+1\right) < -\eta_{\max}\left|e\left(k+1\right)\right|^2 \alpha_{\max}\left|e\left(k\right)\right|^2 + d\left(k+1\right).$$

Note that $\eta_{\max}$ changes adaptively during the RTNN learning and:

$$\eta_{\max} = \max_{i=1}^{3}\{\eta_i\};$$

where all: the unmodelled dynamics, the approximation errors and the perturbations, are represented by the d-term. The rate of convergence lemma used, is given below. The complete proof of that Theorem of stability is given in (Baruch et al., 2008).

Rate of convergence lemma (Baruch & Mariaca-Gaspar, 2009)**.** Let ΔL_k is defined. Then, applying the limit's definition, the identification error bound condition is obtained as:

$$\overline{\lim_{k\to\infty}}\frac{1}{k}\sum_{t=1}^{k}\left(\left|E(t)\right|^2 + \left|E(t-1)\right|^2\right) \le d.$$

Proof. Starting from the final result of the theorem of RTNN stability:

$$\Delta L\left(k\right) \le -\eta\left(k\right)\left|E\left(k\right)\right|^2 - \alpha\left(k\right)\left|E\left(k-1\right)\right|^2 + d$$

and iterating from k=0, we get:

$$L\left(k+1\right) - L\left(0\right) \le -\sum_{t=1}^{k}\left|E(t)\right|^2 - \sum_{t=1}^{k}\left|E(t-1)\right|^2 + dk,$$

$$\sum_{t=1}^{k}\left(\left|E(t)\right|^2 + \left|E(t-1)\right|^2\right) \le dk - L\left(k+1\right) + L\left(0\right) \le dk + L\left(0\right).$$

From here, we could see that d must be bounded by weight matrices and learning parameters, in order to obtain: $\Delta L\left(k\right) \in \mathbb{L}\left(\infty\right)$.

As a consequence: $A\left(k\right) \in \mathbb{L}\left(\infty\right), B\left(k\right) \in \mathbb{L}\left(\infty\right), C\left(k\right) \in \mathbb{L}\left(\infty\right)$

3.2 Recursive Levenberg-Marquardt RTNN learning

The general recursive L-M algorithm of learning, (Baruch & Mariaca-Gaspar, 2009; Baruch & Mariaca-Gaspar, 2010) is given by the following equations:

$$W\left(k+1\right) = W\left(k\right) + P\left(k\right)\nabla Y\left[W\left(k\right)\right]E\left[W\left(k\right)\right], \tag{27}$$

$$Y\left[W\left(k\right)\right] = g\left[W\left(k\right),U\left(k\right)\right], \tag{28}$$

$$E^2\big[W(k)\big] = \big\{Y_p(k) - g\big[W(k), U(k)\big]\big\}^2,$$ (29)

$$DY\big[W(k)\big] = \frac{\partial g\big[W(k), U(k)\big]}{\partial W}\bigg|_{W=W(k)};$$ (30)

Where: W is a general weight matrix (A, B, C) under modification; P is a symmetric matrix updated by (37); DY[·] is an Nw-dimensional gradient vector; Y is the RTNN output vector which depends of the updated weights and the input; E is an error vector; Yp is the plant output vector, which is in fact the target vector. Using the same RTNN adjoint block diagram (see Fig.3), it was possible to obtain the values of the gradients DY[·] for each updated weight, propagating the value D(k) = I through it. Applying equation (30) and using the RTNN adjoint (see Fig. 3) we could compute each weight matrix (A, B, C) in order to be updated. The corresponding gradient components are found as follows:

$$DY\big[C_{ij}(k)\big] = D_{1,i}(k)Z_j(k),$$ (31)

$$D_{1,i}(k) = F_j'\big[Y_i(k)\big],$$ (32)

$$DY\big[A_{ij}(k)\big] = D_{2,i}(k)X_j(k),$$ (33)

$$DY\big[B_{ij}(k)\big] = D_{2,i}(k)U_j(k),$$ (34)

$$D_{2,i}(k) = G_i'\big[Z_j(k)\big]C_i D_{1,i}(k).$$ (35)

Therefore the Jacobean matrix could be formed as:

$$DY\big[W(k)\big] = \Big[DY\big(C_{ij}(k)\big), DY\big(A_{ij}(k)\big), DY\big(B_{ij}(k)\big)\Big].$$ (36)

The P(k) matrix was computed recursively by the equation:

$$P(k) = \alpha^{-1}(k)\big\{P(k-1) - P(k-1)\Omega\big[W(k)\big]S^{-1}\big[W(k)\big]\Omega^T\big[W(k)\big]P(k-1)\big\};$$ (37)

where the S(·), and Ω(·) matrices were given as follows:

$$S\big[W(k)\big] = \alpha(k)\Lambda(k) + \Omega^T\big[W(k)\big]P(k-1)\Omega\big[W(k)\big],$$ (38)

$$\Omega^T\big[W(k)\big] = \begin{bmatrix} & & \nabla Y^T\big[W(k)\big] & & \\ 0 & \cdots & 1 & \cdots & 0 \end{bmatrix};$$

$$\Lambda(k)^{-1} = \begin{bmatrix} 1 & 0 \\ 0 & \rho \end{bmatrix}; \quad 10^{-4} \le \rho \le 10^{-6};$$ (39)

$$0.97 \le \alpha(k) \le 1; \quad 10^3 \le P(0) \le 10^6.$$

The matrix $\Omega(\cdot)$ had a dimension (Nwx2), whereas the second row had only one unity element (the others were zero). The position of that element was computed by:

$$i = k \bmod (Nw) + 1; \quad k > Nw \tag{40}$$

After this, the given up topology and learning are applied for an anaerobic wastewater distributed parameter centralized system identification and control.

4. Description of the direct centralized recurrent neural control with I-term

The block-diagram of the closed loop control system is given on Fig.4.
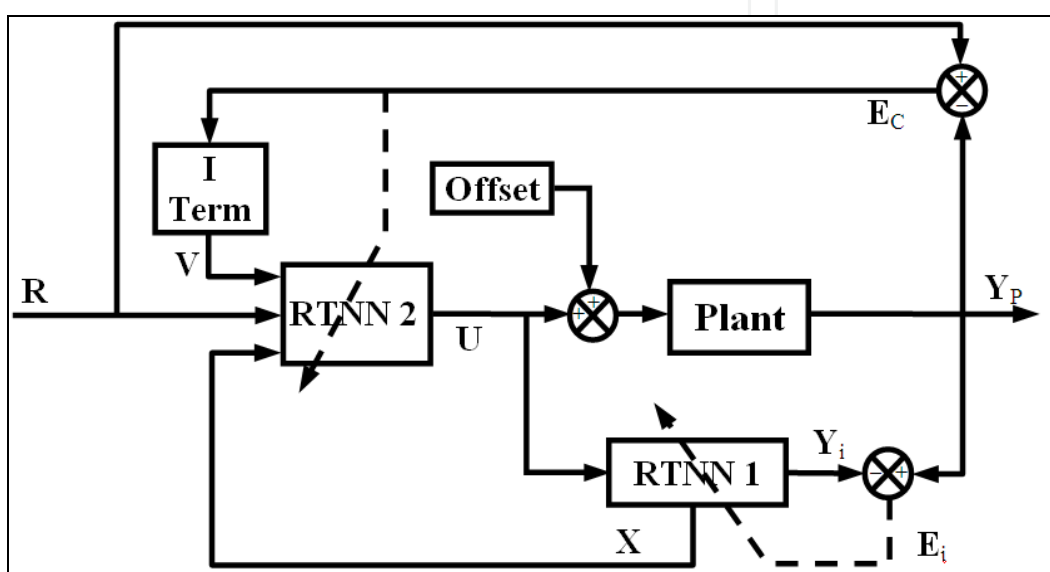
Fig. 4. Block diagram of the direct adaptive I-term control containing RTNN identifier and RTNN controller

It contained a recurrent neural identifier RTNN 1, and a RTNN-2 controller with entries – the reference signal R, the I-term signal V, and the state vector X estimated by the RTNN-1. The input of the plant is perturbed by a constant load perturbation Offset which took into account also the imperfect identification of the plant model. The RTNN-1, 2 topologies are given by (17)-(19), and the nonlinear plant model is given by equations (8)-(16). Let us to linearize the equations of the plant and the RTNN-2 controller and to introduce the equation of the I-term as:

$$V(k+1) = V(k) + T_0 E_c(k) \tag{41}$$

where the dimension of the I-term vector V(k) is equal of the dimension of the error vector, equal of the dimension L of the plant output Yp(k). Now we could write the following z-transfer functions with respect to V, X, R, corresponding to Fig.4:

$$W_p(z) = C_p(zI - A_p)^{-1} B_p \tag{42}$$

$$P_i(z) = (zI - A_i)^{-1} B_i \tag{43}$$

$$Q_1(z) = C_c(zI - A_c)^{-1}B_{cv} \tag{44}$$

$$Q_2(z) = C_c(zI - A_c)^{-1}B_{cx} \tag{45}$$

$$Q_3(z) = C_c(zI - A_c)^{-1}B_{cr} \tag{46}$$

$$I(z) = T_0(zI - I)^{-1} \tag{47}$$

The RTNN topology is controllable and observable, and the BP/L-M algorithms of learning are convergent, (see Baruch & Mariaca-Gaspar, 2009; Baruch & Mariaca-Gaspar, 2010). Then the identification and control errors tend to zero ($E_i(k) = Y_p(k) - Y(k) \to 0$ and $E_c(k) = R(k) - Y_p(k) \to 0$; $k \to \infty$). This means that each transfer function given by equations (42)-(46) is stable with minimum phase. The z-transfer functions (42)-(47) are connected by the next equation, derived following the block-diagram of the Fig. 4:

$$\{I + W_p(z)[I - Q_2(z)P_i(z)]^{-1}Q_1(z)I(z)\}Y_p(z) =$$
$$= W_p(z)[I - Q_2(z)P_i(z)]^{-1}[Q_1(z)I(z) + Q_3(z)]R(z) + W_p(z)Of(z) \tag{48}$$

Substituting (47) in (48), finally we obtained:

$$\{(z-1)I + W_p(z)[I - Q_2(z)P_i(z)]^{-1}Q_1(z)T_0\}Y_p(z) =$$
$$= W_p(z)[I - Q_2(z)P_i(z)]^{-1}[Q_1(z)T_0 + (z-1)Q_3(z)]R(z) + W_p(z)(z-1)Of(z) \tag{49}$$

The equation (49) showed that the closed-loop system is stable ($Y_p(k) \to R(k)$; $k \to \infty$) and that the I-term eliminated the constant perturbation $Of(z)$ because the last term tended to zero when z tended to 1.

The centralized DPS could be considered as a system with excessive measurements (L>M), where the Direct Adaptive Neural Control (DANC) performed a data fusion so to elaborate the control action. So we need to compute the plant input error for the learning of the RTNN-2 controller. An approximated way to obtain the input error from the output error is pre-multiplying it by the $(CB)^+$ using the estimated C,B-matrices as follows:

$$E_u(k) = (CB)^+E_c(k), (CB)^+ = [(CB)^T(CB)]^{-1}(CB)^T \tag{50}$$

5. Description of the indirect (sliding mode) centralized recurrent neural control with I-term

The block-diagram of the control system is given on Fig.5. It contained a recurrent neural identifier RTNN 1, and a Sliding Mode (SM) controller with entries – the reference signal R, the output error Ec, and the states X and parameters A, B, C, estimated by the neural identifier RTNN-1. The total control is a sum of the SM control and the I-term control, computed using (41).
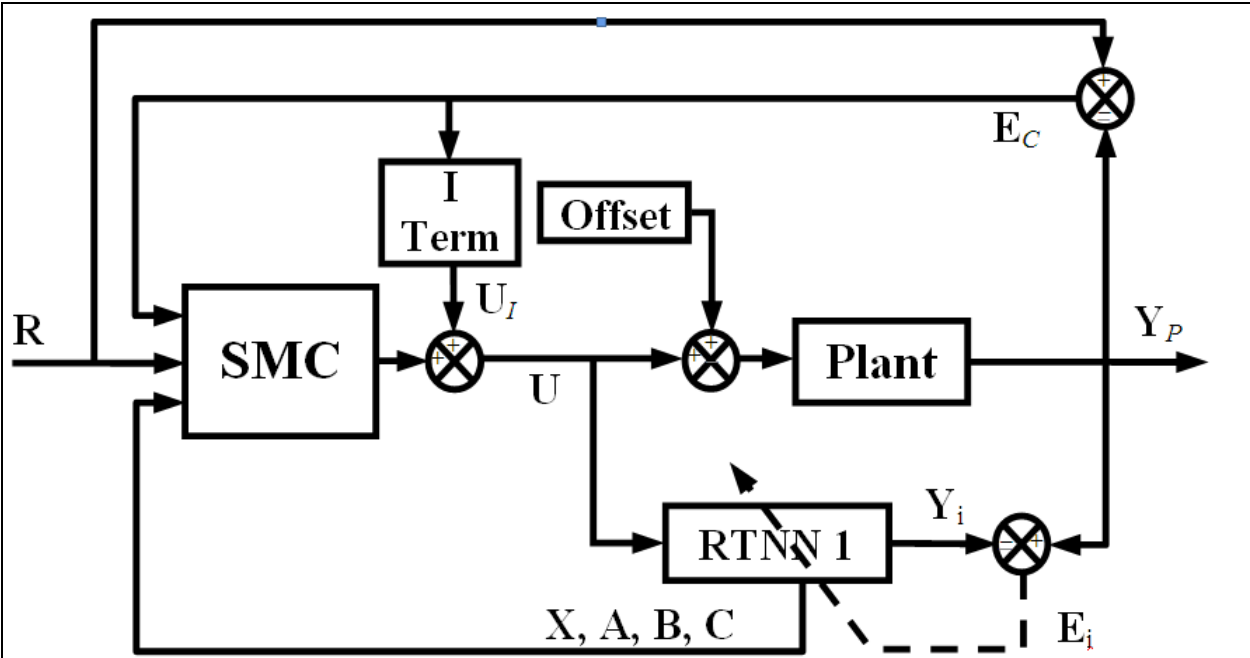
Fig. 5. Block diagram of the indirect adaptive SM control with I-term containing RTNN identifier and SM controller

The linearization of the activation functions of the local learned identification RTNN-1 model, which approximated the plant, leads to the following linear local plant model:

$$X(k+1) = AX(k) + BU(k), Y(k) = CX(k) \tag{51}$$

where L > M (rectangular system), is supposed. Let us define the following sliding surface with respect to the output tracking error:

$$S(k+1) = E(k+1) + \sum_{i=1}^{P} \gamma_i E(k - i + 1) \; ; \;\; |\gamma_i| < 1; \tag{52}$$

where: S(·) is the sliding surface error function; E(·) is the systems local output tracking error; γ_i are parameters of the local desired error function; P is the order of the error function. The additional inequality in (52) is a stability condition, required for the sliding surface error function. The local tracking error is defined as:

$$E(k) = R(k) - Y(k); \tag{53}$$

where R(k) is a L-dimensional local reference vector and Y(k) is an local output vector with the same dimension. The objective of the sliding mode control systems design is to find a control action which maintains the systems error on the sliding surface assuring that the output tracking error reached zero in P steps, where P<N, which is fulfilled if:

$$S(k+1) = 0. \tag{54}$$

As the local approximation plant model (51), is controllable, observable and stable, (Baruch et al., 2004; Baruch et al., 2008), the matrix A is block-diagonal, and L>M

(rectangular system is supposed), the matrix product (CB) is nonsingular with rank M, and the plant states X(k) are smooth non- increasing functions. Now, from (51)-(54), it is easy to obtain the equivalent control capable to lead the system to the sliding surface which yields:

$$U_{eq}(k) = (CB)^+ \left[-CAX(k) + R(k+1) + \sum_{i=1}^{P} \gamma_i E(k-i+1) \right] + Of, \qquad (55)$$

$$(CB)^+ = \left[(CB)^T (CB) \right]^{-1} (CB)^T. \qquad (56)$$

Here the added offset Of is a learnable M-dimensional constant vector which is learnt using a simple delta rule (see Haykin, 1999, for more details), where the error of the plant input is obtained backpropagating the output error through the adjoint RTNN model. An easy way for learning the offset is using the following delta rule where the input error is obtaned from the output error multiplying it by the same pseudoinverse matrix, as it is:

$$Of(k+1) = Of(k+1) = Of(k) + \eta (CB)^+ E(k). \qquad (57)$$

If we compare the I-term expression (41) with the Offset learning (57) we could see that they are equal which signifyed that the I-term generate a compensation offset capable to eliminate steady state errors caused by constant perturbations and discrepances in the reference tracking caused by non equal input/output variable dimensions (rectangular case systems). So introducing an I-term control it is not necessary to use an compensation offset in the SM control law (55).

The SMC avoiding chattering is taken using a saturation function inside a bounded control level Uo, taking into account plant uncertainties. So the SMC has the form:

$$U(k) = \begin{cases} U_{eq}(k), & \text{if } \left\| U_{eq}(k) \right\| < U_0 \\ \dfrac{-U_0 U_{eq}(k)}{\left\| U_{eq}(k) \right\|}, & \text{if } \left\| U_{eq}(k) \right\| \geq U_0. \end{cases} \qquad (58)$$

The proposed SMC cope with the characteristics of the wide class of plant model reduction neural control with reference model, and represents an indirect adaptive neural control, given by (Baruch et al., 2007a; Baruch et al., 2007b).

6. Description of the centralized optimal control with I-term

The block-diagram of the optimal control system is given on Fig.6. It contained a recurrent neural identifier RTNN 1, and an optimal controller with entries – the reference signal R, the output of the I-term block, and the states X and parameters A, B, C, estimated by the neural identifier RTNN-1. The optimal control algorithm with I-term could be obtained extending the linearized model (51) with the model of the I-term (41). The extended model is:
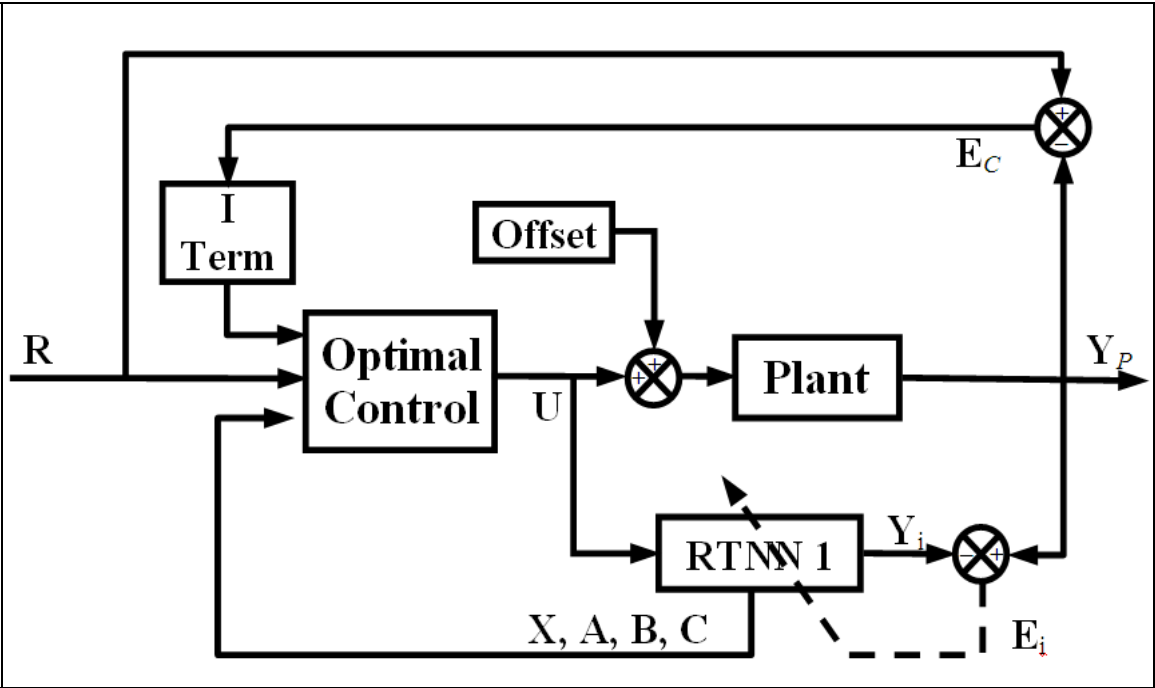
Fig. 6. Block diagram of the real-time optimal control with I-term containing RTNN identifier and optimal controller

$$X_e(k+1) = A_e X_e(k) + B_e U(k) \tag{59}$$

Where: $X_e = [X \mid V]^T$ is a state vector with dimension (L + N) and:

$$A_e = \begin{vmatrix} A & 0 \\ -(CB)^+ CA & I \end{vmatrix}, B_e = \begin{vmatrix} B \\ -I \end{vmatrix} \tag{60}$$

The optimal I-term control is given by:

$$U(k) = -[B_e^T P_e B_e + R]^{-1}[B_e^T P_e B_e]X_e(k) \tag{61}$$

Where the P_e is solution of the discrete Riccati equation:

$$P_e(k+1) = A_e^T [P_e(k) - P_e(k)B_e(B_e^T P_e(k)B_e + R)^{-1}B_e^T P_e(k)]A_e + Q \tag{62}$$

The given up optimal control is rather complicated and here it is used for purpose of comparison.

7. Simulation results

In this paragraph, graphical and numerical simulation results of system identification, direct, indirect (SM), and optimal control, with and without I-term, will be given. For lack of space we will give graphical results only for the X_1 variable. Furthermore the graphical results for the other variables possessed similar behavior. The identification results are obtained from an RTNN identifier by a BP or L-M learning. For sake of comparison we give results of systems identification using both algorithms of learning.

7.1 Simulation results of the system identification

The RTNN-1 performed real-time neural system identification (parameters and states estimation) of 18 output plant variables, which are: 4 variables for each collocation point z=0.2H, z=0.4H, z=0.6H, z=0.8H of the fixed bed as: X_1 (acidogenic bacteria), X_2 (methanogenic bacteria), S_1 (chemical oxygen demand) and S_2 (volatile fatty acids), and the following variables in the recirculation tank: S_{1T} (chemical oxygen demand) and S_{2T} (volatile fatty acids). For lake of space we shall show some graphical results (see Fig. 7-9) only for the X_1 variable. The topology of the RTNN-1 is (2, 20, 18), the activation functions are tanh(.) for both layers. The learning rate parameters for the L-M learning are as follows: the forgetting factor is $\alpha=1$, the regularization constant is $\rho=0.001$, and the initial value of the P matrix is an identity matrix with dimension 420x420. For the BP algorithm of learning the learning constants are chosen as $\alpha=0$, $\eta=0.4$. The simulation results of RTNN-1 system identification are obtained on-line during 400 days with a step of 0.5 day in four measurement points using BP and L-M learning. The identification inputs used are combination of three sinusoids as:

$$S_{1,in} = 0.5 + 0.02\sin\left(\frac{\pi}{100}t\right) + 0.1\sin\left(\frac{3\pi}{100}t\right) + 0.04\cos\left(\frac{\pi}{100}t\right) \tag{63}$$

$$S_{2,in} = 0.5 + 0.1\sin\left(\frac{\pi}{100}t\right) + 0.1\sin\left(\frac{5\pi}{100}t\right) + 0.1\cos\left(\frac{8\pi}{100}t\right) \tag{64}$$
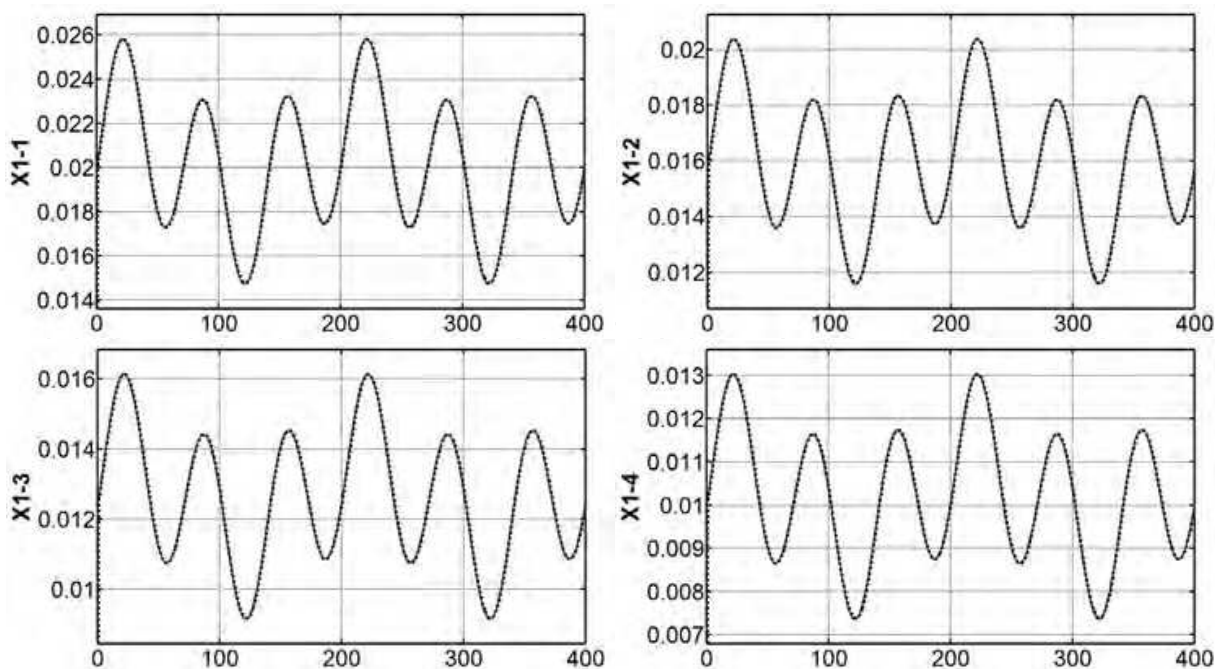
Fig. 7. Graphical simulation results of the neural identification of the plant output X_1 vs. RTNN output in four measurement points for the total time of L-M learning : a) z=0.2H, b) z=0.4H, c) z=0.6H, d) z=0.8H
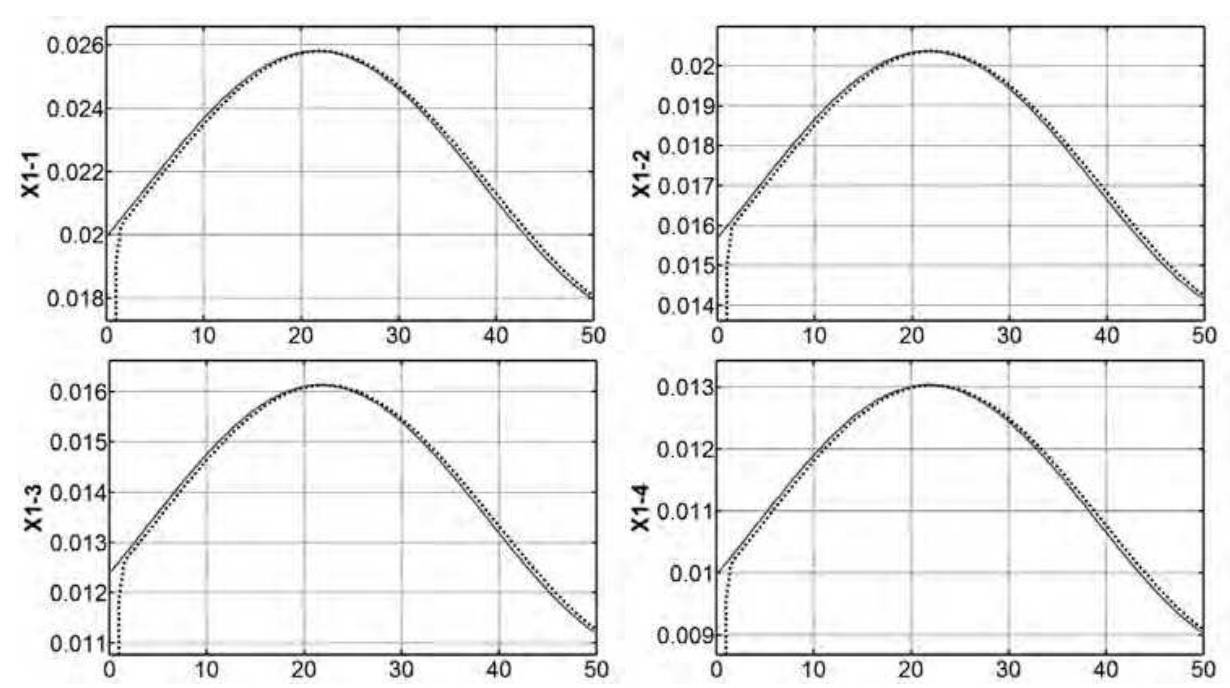
Fig. 8. Graphical simulation results of the neural identification of the plant output X_1 vs. RTNN output in four measurement points for the beginning of L-M learning : a) z=0.2H, b) z=0.4H, c) z=0.6H, d) z=0.8H

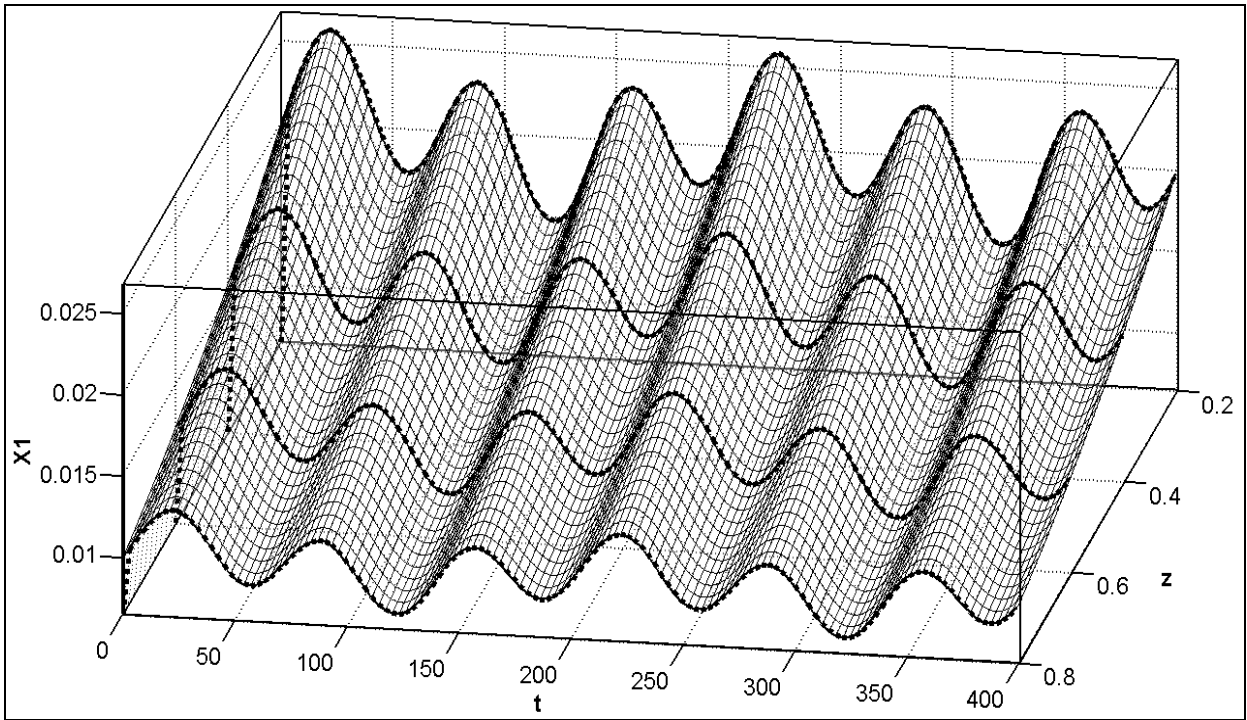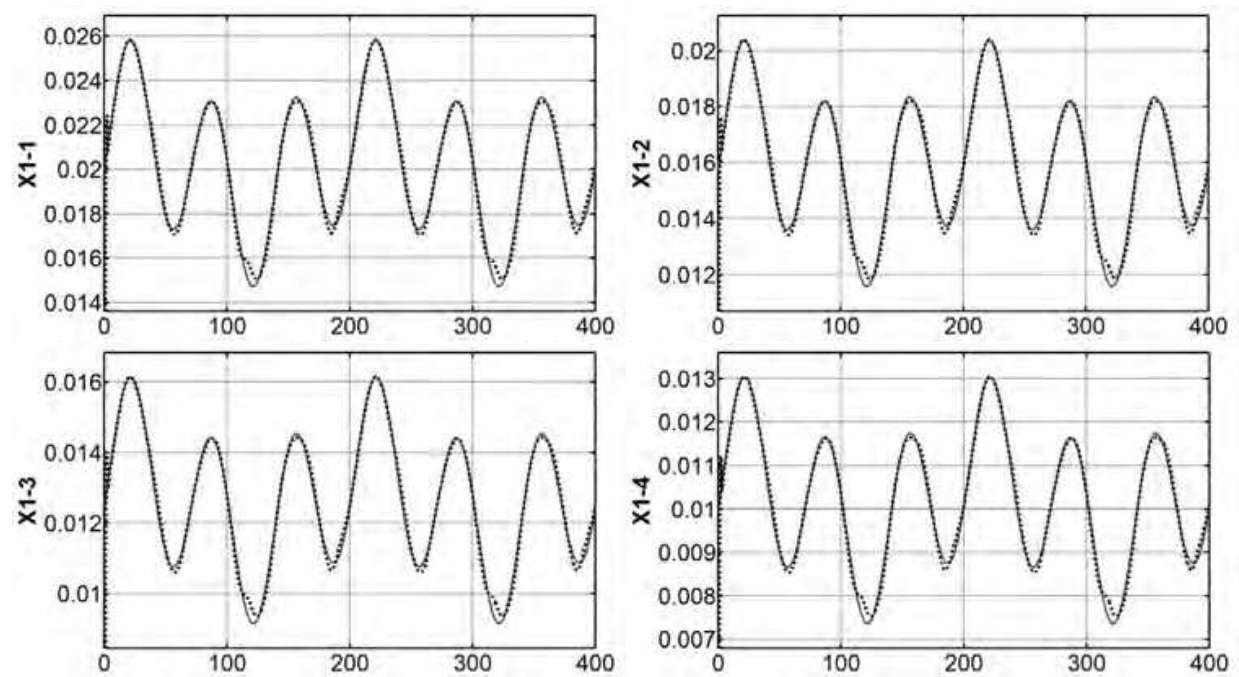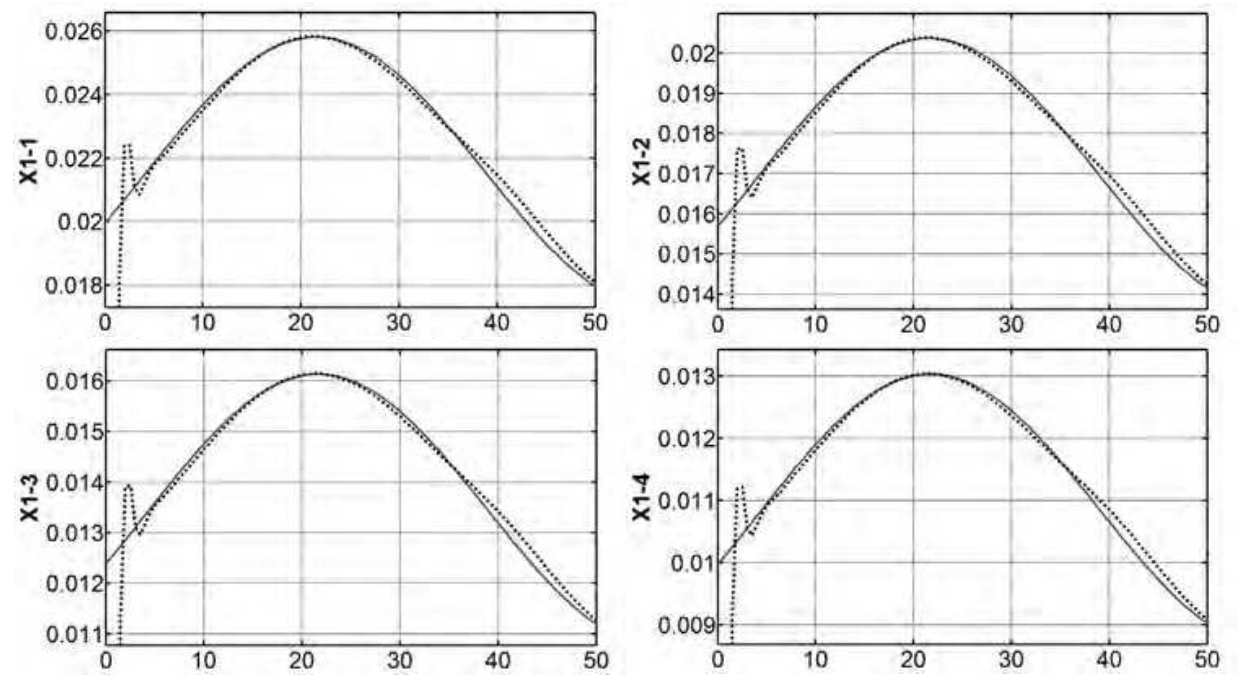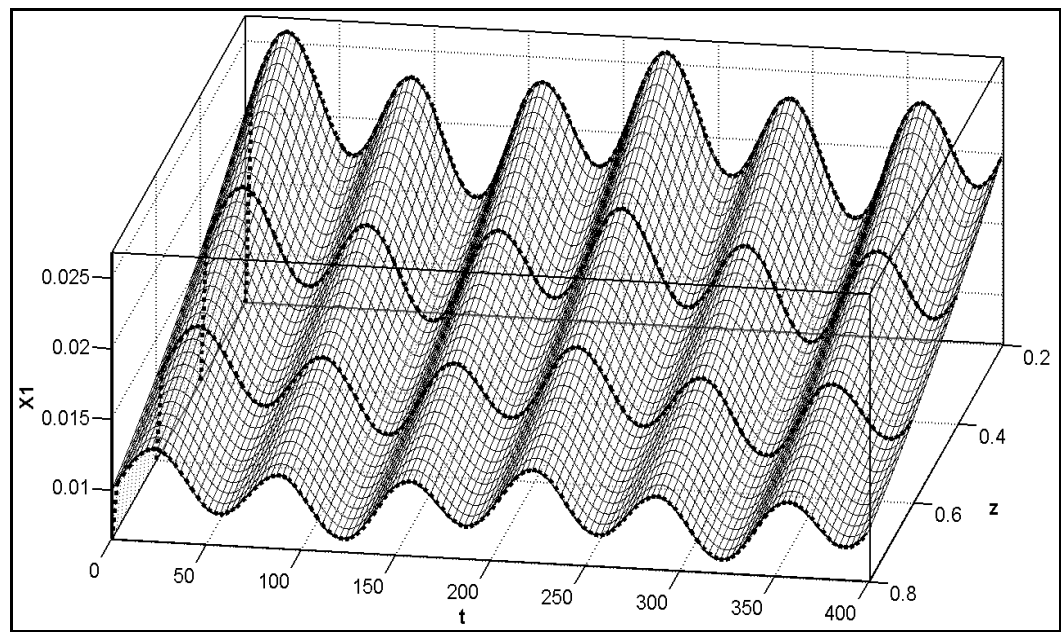Fig. 9. Three dimensional plot of the neural identification results of the plant output X_1 in four measurement points of L-M learning : z=0.2H, z=0.4H, z=0.6H, z=0.8H

Fig. 10. Graphical simulation results of the neural identification of the plant output X_1 vs. RTNN output in four measurement points for the total time of BP learning : a) z=0.2H, b) z=0.4H, c) z=0.6H, d) z=0.8H

Fig. 11. Graphical simulation results of the neural identification of the plant output X_1 vs. RTNN output in four measurement points for the beginning of BP learning: a) z=0.2H, b) z=0.4H, c) z=0.6H, d) z=0.8H

Fig. 12. Three dimensional plot of the neural identification results of the plant output X_1 in four measurement points of BP learning : z=0.2H, z=0.4H, z=0.6H, z=0.8H

Table 2 and Table 3 compared the final Means Squared Error (MSE%) results of the L-M and BP neural identification of plant variables for the fixed bed and the recirculation tank. Note that the form of the plant process variables in the different measurement points is equal but the amplitude is different depending on the point position.

Collocation point	X_1	X_2	S_1/S_{1T}	S_2/S_{2T}
z=0.2	5.0843E-7	1.8141E-6	1.3510E-4	2.5476E-4
z=0.4	3.1428E-7	1.3934E-6	8.3839E-5	1.8217E-4
z=0.6	1.9617E-7	9.6976E-7	5.2303E-5	1.2200E-4
z=0.8	1.2669E-7	6.6515E-7	3.3940E-5	8.1905E-5
Recirculation tank			2.6318E-5	6.3791E-5

Table 2. MSE of the centralized RTNN approximation of the bioprocess output variables in the collocation points, using the L-M RTNN learning

Collocation point	X_1	X_2	S_1/S_{1T}	S_2/S_{2T}
z=0.2	5.9981E-7	2.1006E-6	1.5901E-4	2.8282E-4
z=0.4	3.7111E-7	1.6192E-6	9.8240E-5	2.0506E-4
z=0.6	2.3145E-7	1.1308E-6	6.1119E-5	1.3908E-4
z=0.8	1.4997E-7	7.7771E-7	3.9595E-5	9.4061E-5
Recirculation tank			3.0694E-5	7.3404E-5

Table 3. MSE of the centralized RTNN approximation of the bioprocess output variables in the collocation points, using the BP RTNN learning

The graphical and numerical results of the centralized RTNN identification (see Fig. 7-12, and Tables 2, 3) showed a good RTNN convergence and precise plant output tracking (MSE is 2.5476E-4 for the L-M, and 2.8282E-4 for the BP RTNN learning in the worst case).

7.2 Simulation results of the centralized direct adaptive neural control with I-term using L-M learning

The real-time DANC (see Fig. 4) contained a neural identifier RTNN-1 and a neural controller RTNN-2 with topology (40, 10, 2). Both RTNNs-1, 2 are learnt by the L-M algorithm with parameters: RTNN-1 (α=1, ρ=0.0001, Po=10 I with dimension 420x420); RTNN-2 (α=1, ρ=0.01, Po=0.8 I with dimension 430x430). The simulation results of DANC are obtained on-line during 1000 days with a step of 0.1 day. The control signals are shown on Fig. 13. The Fig. 14-16 compared the plant output X_1 with the reference signal in different measurement points. The form of the set points (train of pulses with random amplitude) of the variable X_1 in the different measurement points is equal but the amplitude is different depending on the point position. This means that the plant has different signal amplification in each measurement point which needs to be taken in consideration.
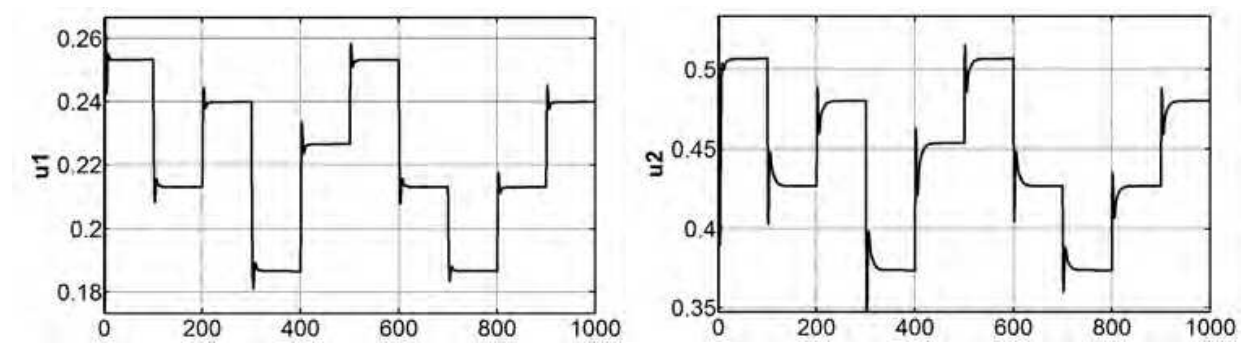
Fig. 13. Plant input control signals generated by I-term DANC: a) Sin1, and b) Sin2

Collocation point	X_1	X_2	S_1/S_{1T}	S_2/S_{2T}
z=0.2	2.2920E-8	1.3366E-7	5.9740E-6	1.7568E-5
z=0.4	1.4517E-8	8.0704E-8	3.4003E-6	9.3272E-6
z=0.6	8.5061E-9	4.3891E-8	1.9213E-6	4.9682E-6
z=0.8	4.4770E-9	2.1242E-8	1.2789E-6	3.1322E-6
Recirculation tank			1.0067E-6	2.2073E-6

Table 4. MSE of the centralized I-term DANC of the bioprocess output variables in the collocation points, using the L-M RTNN learning

The given on Fig. 14-16 graphical results of I-term DANC showed smooth exponential behavior. It could be seen also that the L-M learning converge fast and the I-term remove the constant noise Of, and the plant uncertainties. The obtained numerical results (see Table 4) of final MSE of L-M learning possessed small values (1.7568E-5 in the worse case).
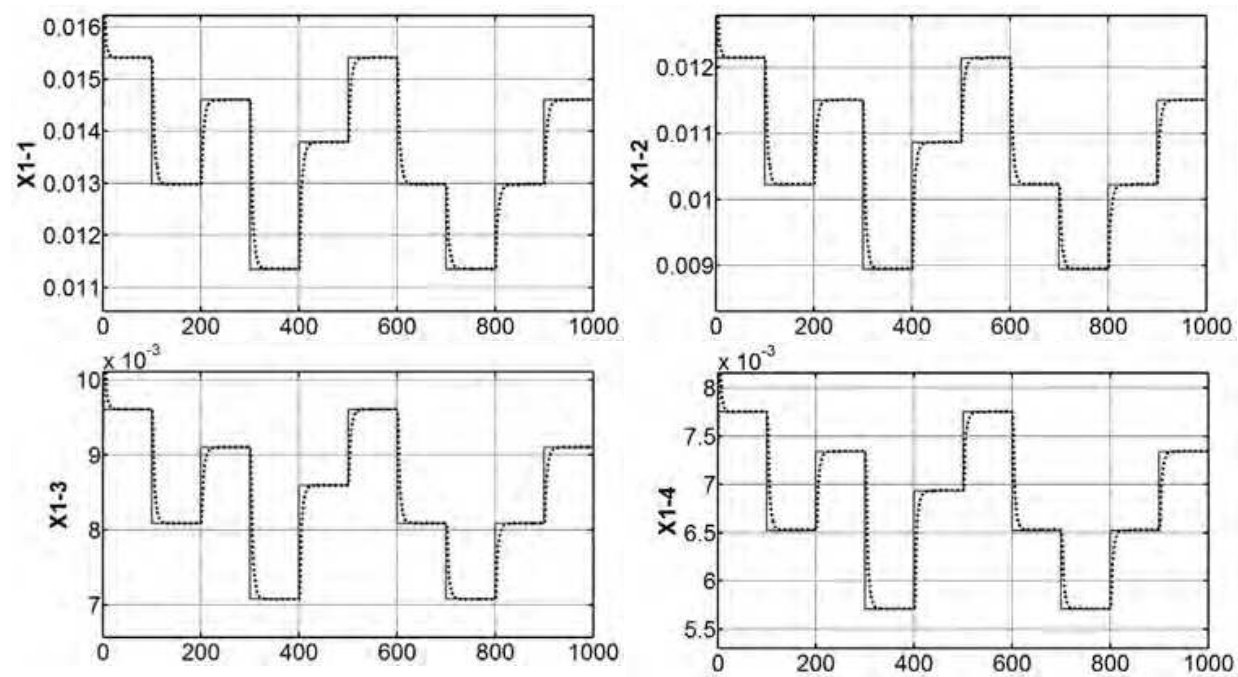
Fig. 14. Graphical simulation results of the I-term DANC of the plant output X_1 vs. system reference in 4 measurement points for the total time of L-M learning: a)z=0.2H, b) z=0.4H, c) z=0.6H, d)z=0.8H
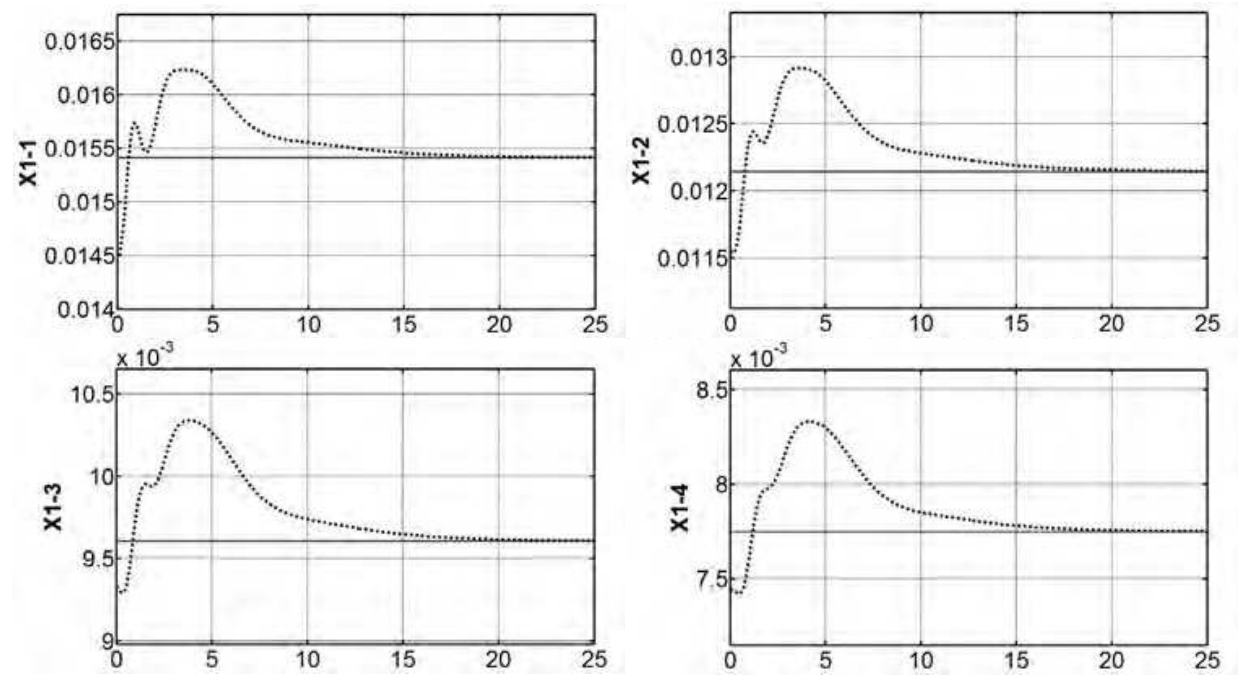

Fig. 15. Graphical simulation results of the I-term DANC of the output X_1 vs. system reference in four measurement points for the beginning of L-M learning: a) z=0.2H, b) z=0.4H, c) z=0.6H,d) z=0.8H
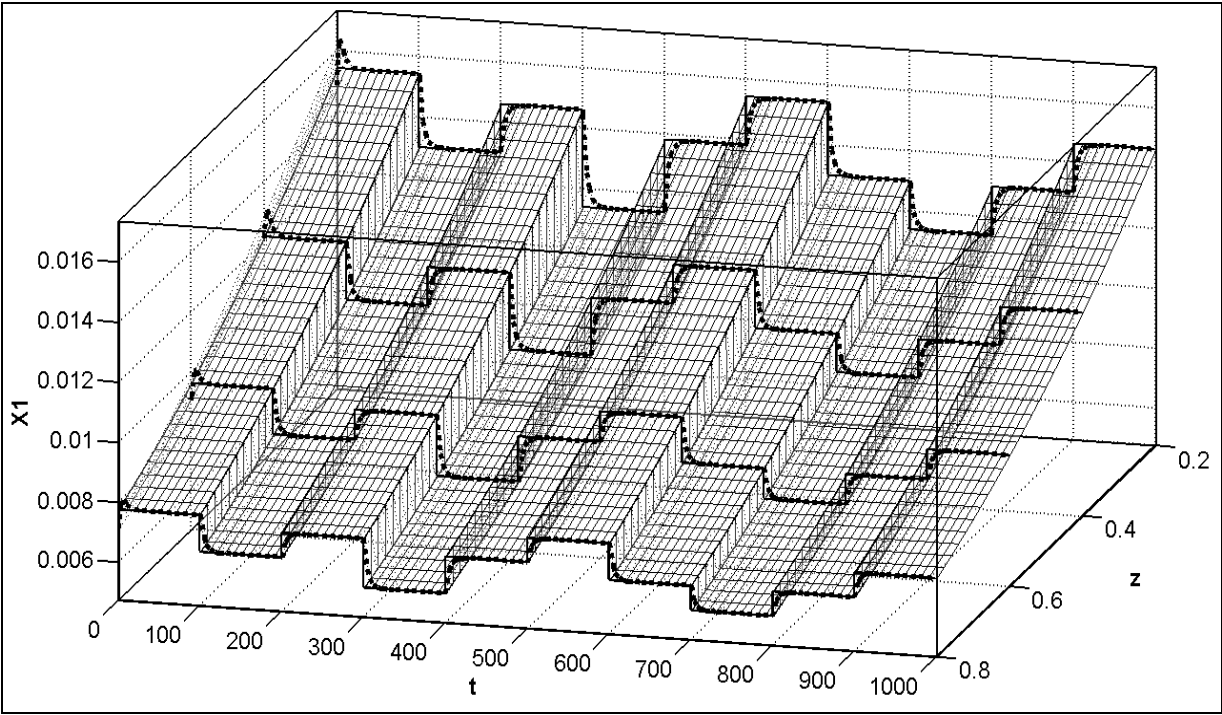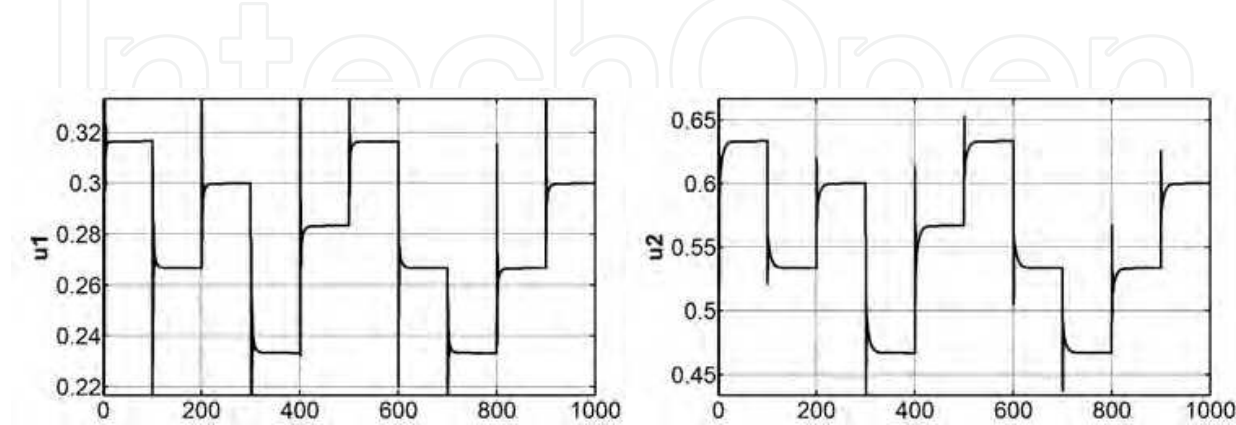
Fig. 16. Three dimensional plot of the I-term DANC of the plant output X_1 in four measurement points of L-M learning : z=0.2H, z=0.4H, z=0.6H, z=0.8H

7.3 Simulation results of the centralized indirect (SM) adaptive neural control with I-term using L-M learning

In this case the indirect adaptive I-term control is a sum of the I-term control signal and the SM control, computed using the state and parameter information issued from the RTNN-1 neural identifier. The control signals are shown on Fig. 17. The X_1 control simulation results are given on Fig. 18-20. The simulation results of SMC are obtained on-line during 1000 days with a step of 0.1 day. The given on Fig. 18-20 graphical results of I-term SMC demonstrated smooth behavior. It could be seen also that the L-M learning converge fast and the I-term remove the constant noise Of, and the plant uncertainties.

Fig. 17. Plant input control signals generated by the I-term centralized indirect SMC: a) Sin1, and b) Sin2
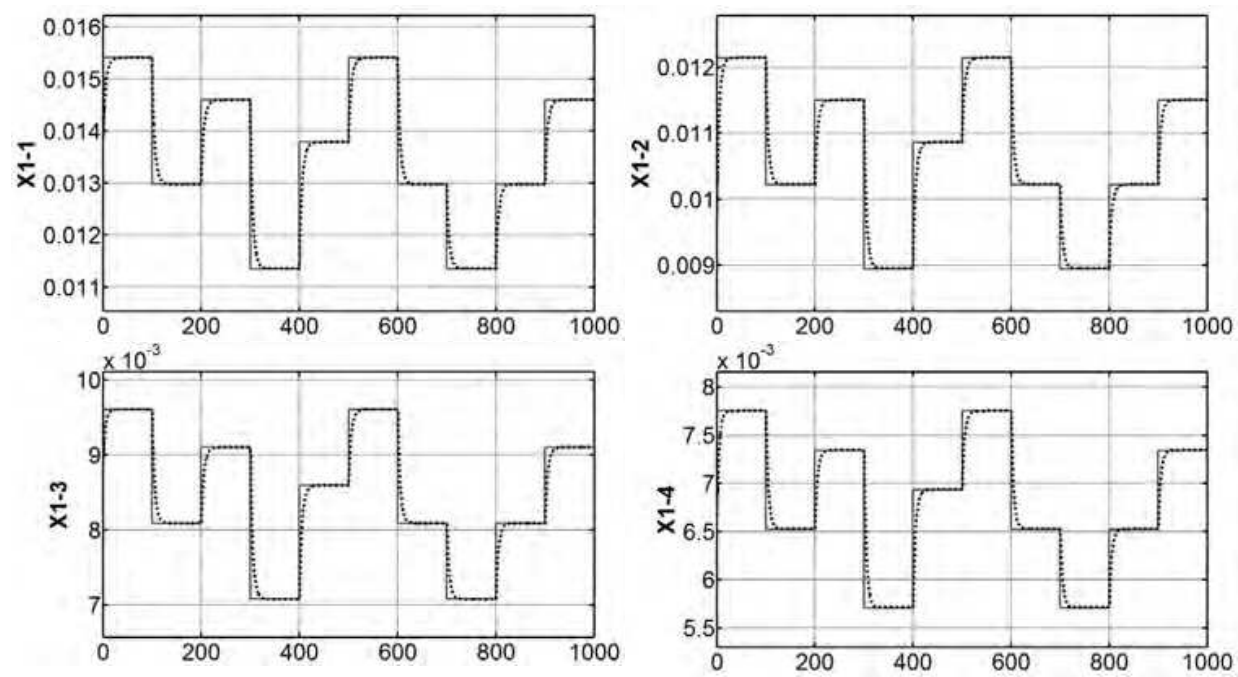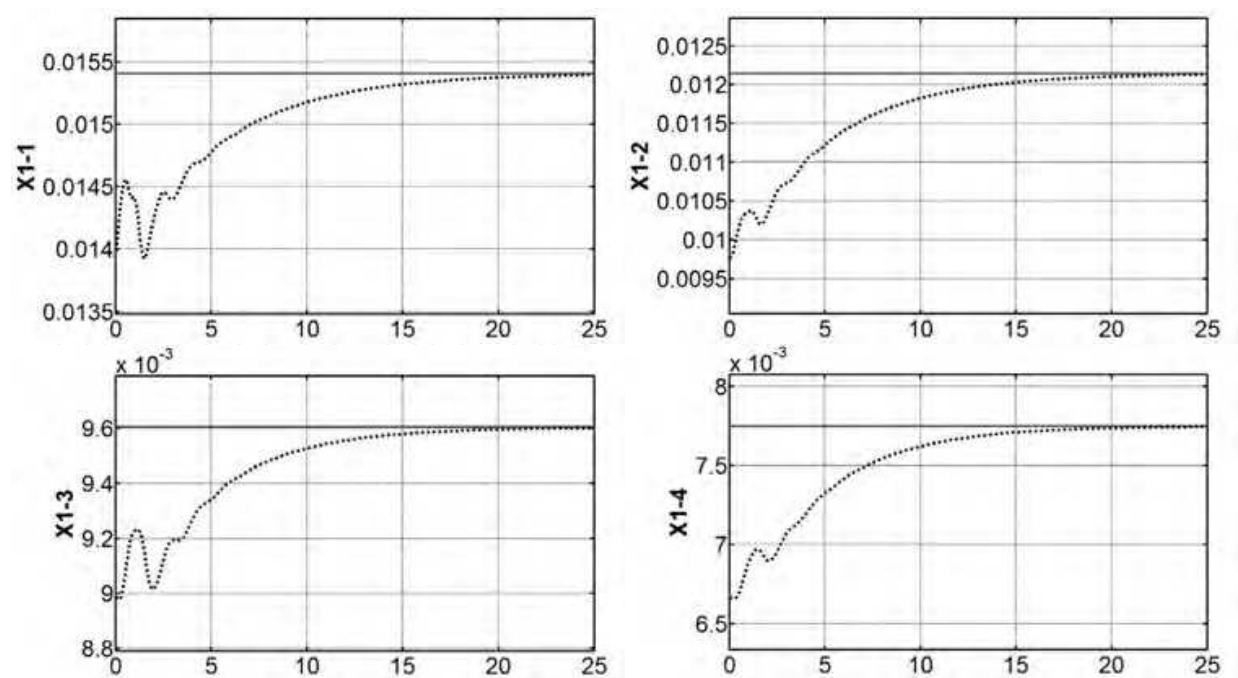
Fig. 18. Graphical simulation results of the I-term indirect SM control of the plant output X_1 vs. system reference in four measurement points for the total time of L-M learning: a) z=0.2H, b) z=0.4H, c) z=0.6H, d) z=0.8H
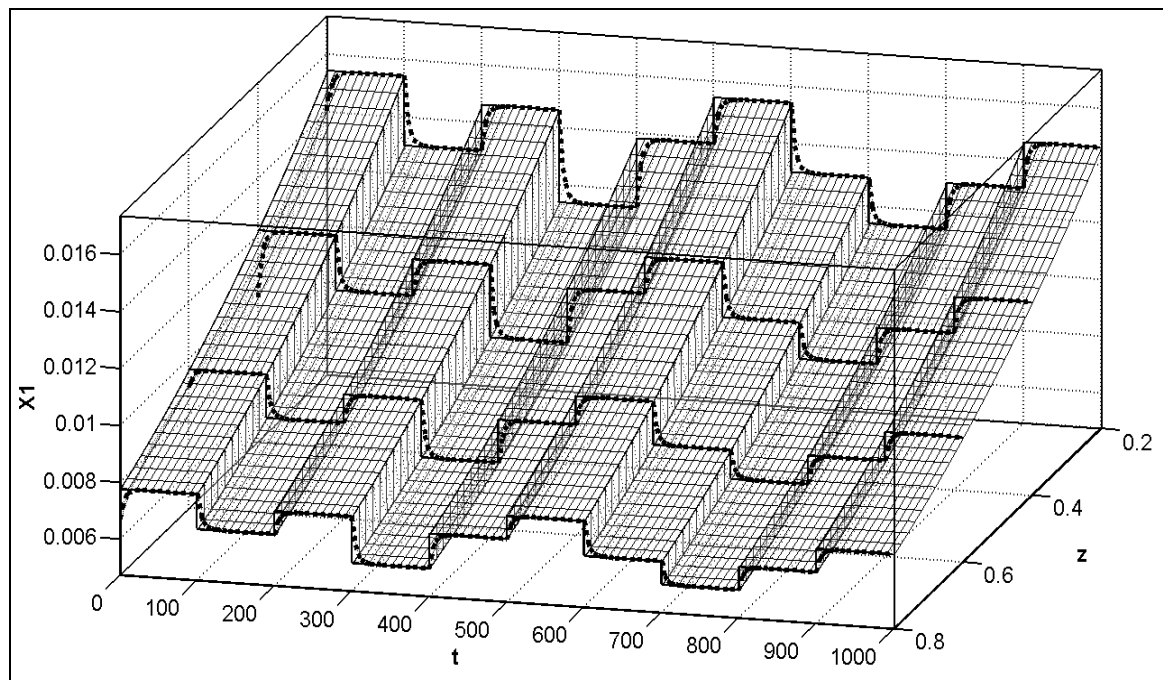

Fig. 19. Graphical simulation results of the I-term indirect SM control of the plant output X_1 vs. system reference in four measurement points for the beginning of L-M learning: a) z=0.2H, b) z=0.4H, c) z=0.6H, d) z=0.8H

Fig. 20. Three dimensional plot of the I-term indirect SM control results of the plant output X_1 in four measurement points of L-M learning : z=0.2H, z=0.4H, z=0.6H, z=0.8H

The Fig. 21 illustrated the behavior of the SMC system without I-term perturbed by a constant noise Of, causing a big error of reference tracking.
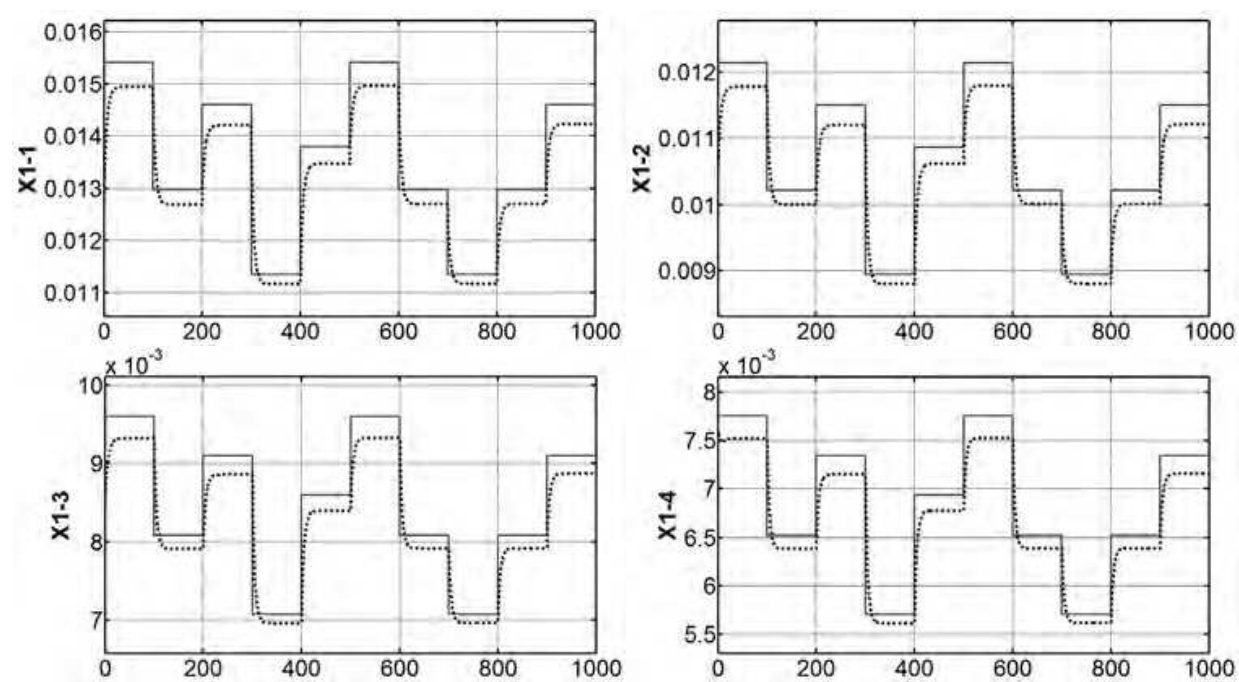

Fig. 21. Graphical simulation results of the indirect SM control without I-term of the plant output X_1 vs. system reference in four measurement points for the total time of L-M learning: a) z=0.2H, b) z=0.4H, c) z=0.6H, d) z=0.8H

The given on Fig. 18-20 graphical results of I-term SMC showed smooth exponential behavior, fast convergence and the removal of the constant noise terms. The Fig. 21 showed that the constant perturbation in the input of the plant caused a deviation of the plant output X_1 with respect of the set point R_1 and this occurred for all other plant output signals and measurement points. The final MSE for all point output variables are given in Table 5.

Collocation point	X_1	X_2	S_1/S_{1T}	S_2/S_{2T}
z=0.2	2.6969E-8	1.7122E-7	9.9526E-6	2.1347E-5
z=0.4	1.3226E-8	1.2511E-7	5.2323E-6	1.2903E-5
z=0.6	1.0873E-8	6.5339E-8	3.2234E-6	7.0511E-6
z=0.8	5.9589E-9	4.4750E-8	1.6759E-6	4.4548E-6
Recirculation tank			1.1842E-6	2.5147E-6

Table 5. MSE of the I-term indirect (SM) centralized control of the bioprocess plant variables in all measurement points

The final MSE given on Table 5 possessed small values (2.1345E-5 in the worse case).

7.4 Simulation results of the centralized I-term optimal control using neural identifier and L-M RTNN learning

The integral term extended the identified local linear plant model so it is part of the indirect optimal control algorithm. The generated by the optimal control plant input signals are given on Fig. 22. The Fig. 23-25 illustrated the X_1 I-term optimal control results. The MSE numerical results for all final process variable and measurement points control results, given on Table 6 possessed small values (1.4949E-5 in the worse case).
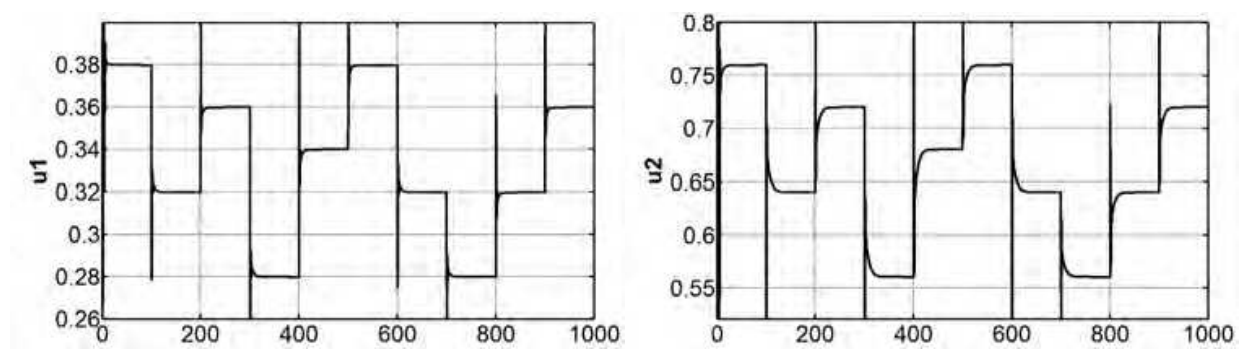

Fig. 22. Plant input control signals generated by the I-term centralized optimal control: a) Sin1, and b) Sin2

Collocation point	X_1	X_2	S_1/S_{1T}	S_2/S_{2T}
z=0.2	2.06772E-8	1.5262E-7	9.3626E-6	1.4949E-5
z=0.4	1.3819E-8	7.5575E-8	5.6917E-6	1.0197E-5
z=0.6	1.8115E-8	4.7505E-8	2.8872E-6	6.1763E-6
z=0.8	1.5273E-8	5.9744E-8	1.6295E-6	4.2868E-6
Recirculation tank			1.3042E-6	2.5136E-6

Table 6. MSE of the I-term opt. control of the bioprocess plant variables in all meas. points
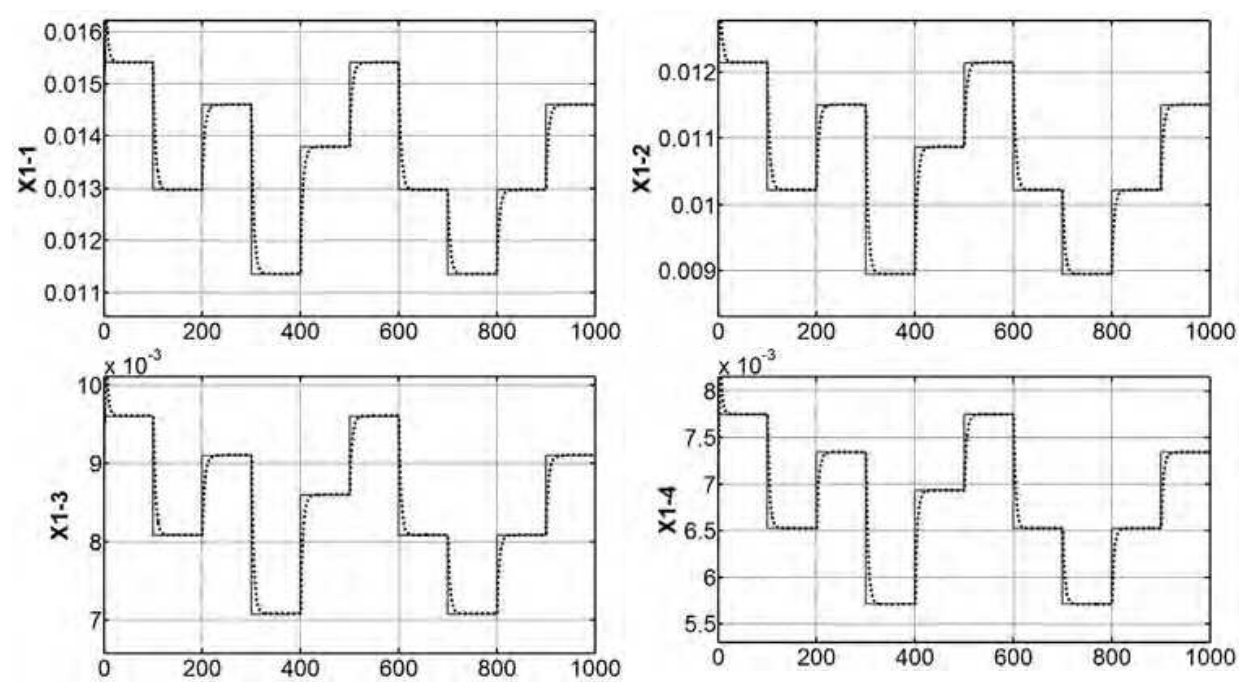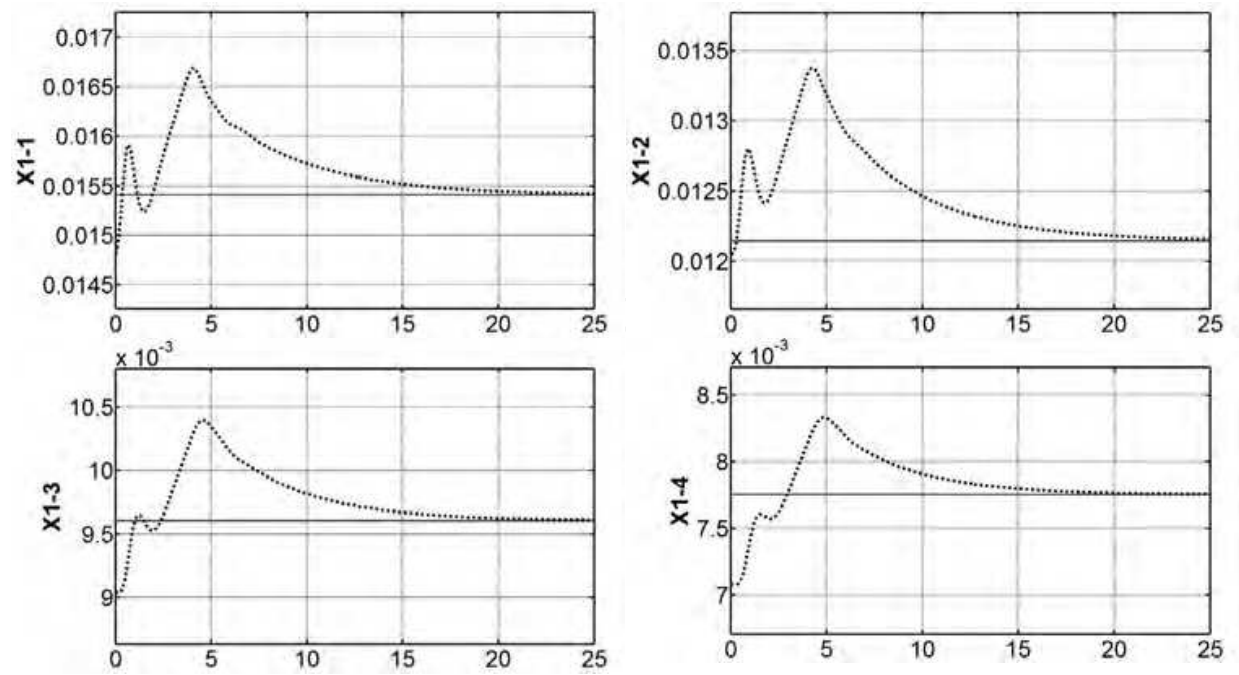
Fig. 23. Graphical simulation results of the I-term optimal control of the plant output X_1 vs. system reference in four measurement points for the total time of L-M learning: a) z=0.2H, b) z=0.4H, c) z=0.6H, d) z=0.8H

Fig. 24. Graphical simulation results of the I-term optimal control of the plant output X_1 vs. system reference in four measurement points for the beginning of L-M learning: a) z=0.2H, b) z=0.4H, c) z=0.6H, d) z=0.8H
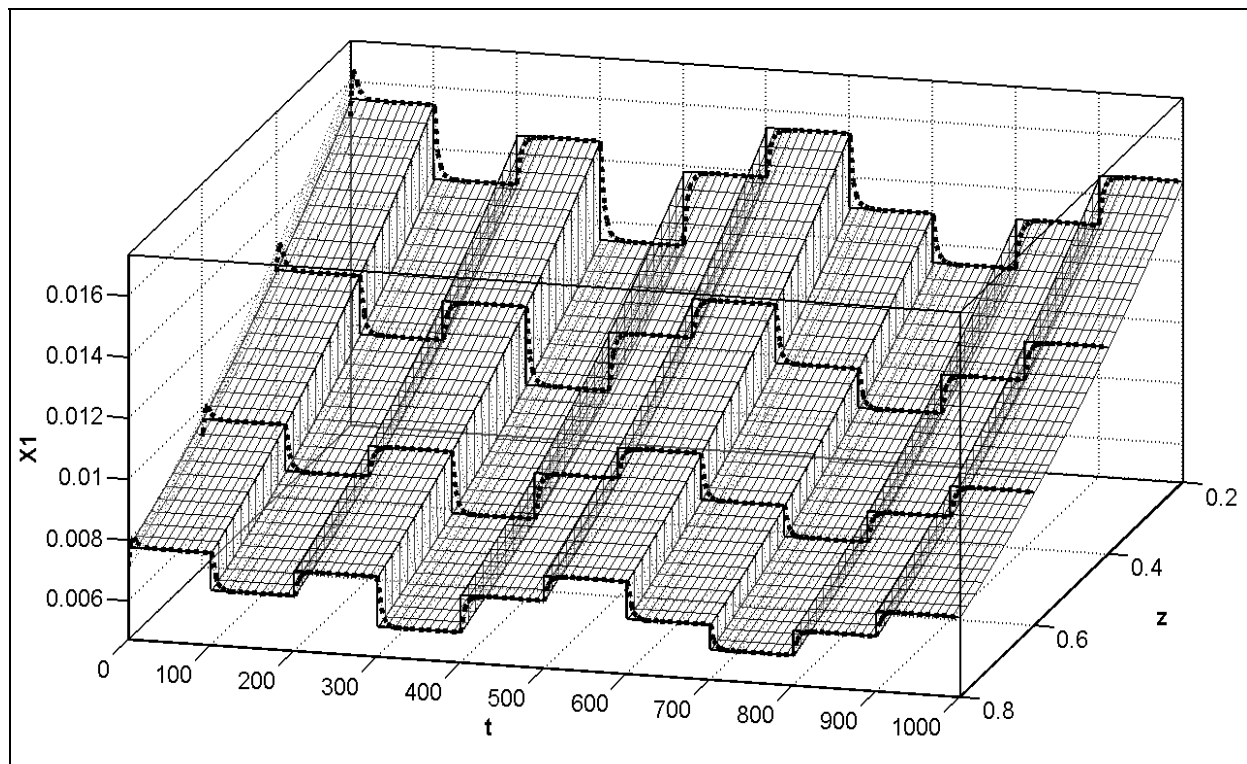
Fig. 25. Three dimensional plot of the I-term optimal control results of the plant output X_1 in four measurement points of L-M learning : z=0.2H, z=0.4H, z=0.6H, z=0.8H

The given on Fig. 23-25 graphical results of I-term optimal control showed smooth exponential behavior, fast convergence and the removal of the constant noise terms.

8. Conclusion

The paper proposed a new neural identification and control methodology for distributed parameter bioprocess plant. The simplification of the DPS given by PDEs is realized using the orthogonal collocation method in four collocation points, converting the PDE plant description in ODE one. The system is identified using RTNN model and BP and L-M learning, where a high precision of convergence is achieved (the final MSE% for both BP and L-M learning algorithms is of order of E-4 in the worse case). The comparative results showed a slight priority in precision and convergence of the L-M over the BP which could be seen in Figures 8, 11, and Tables 2, 3 (the worse case MSE for the L-M RTNN learning is 2.5476E-4 vs. BP RTNN learning which is 2.8282E-4). The obtained comparative simulation results of centralized adaptive direct, indirect SM and optimal control with I-term exhibited a good RTNN convergence and precise reference tracking. The MSE% of plant outputs tracking for the three considered methods of control is of order of E-5 in the worse case. The graphical simulation results showed that all control methods with I-term could compensate constant plant input noises and the I-term removal caused a system outputs deviation from the reference signals (see Fig. 21). The MSE study ordered the control methods used as: optimal, direct, and indirect, but the difference between them is little (see Tables 4.5.6 where worse case final MSE for DANC is 1.7568E-5; for SMC is 2.1347E-5; for the optimal control it is 1.4949E-5).

9. References

Aguilar-Garnica, F.; Alcaraz-Gonzalez, V. & Gonzalez-Alvarez, V. (2006). Interval observer design for an anaerobic digestion process described by a distributed parameter model, *Proceedings of the Second International Meeting on Environmental Biotechnology and Engineering: 2IMEBE*, pp. 1-16, ISBN 970-95106-0-6, Mexico City, Mexico, September 2006, CINVESTAV-IPN, Mexico

Baruch, I.S.; Flores, J.M.; Nava, F.; Ramirez, I.R. & Nenkova, B. (2002). An advanced neural network topology and learning, applied for identification and control of a D.C. motor. *Proceedings of the 1st Int. IEEE Symposium on Intelligent Systems,* Vol. 1, pp. 289-295, ISBN 0-7803-7601-3, Varna, Bulgaria, September 2002, IEEE Inc.,New York

Baruch, I.S.; Barrera-Cortes, J. & Hernandez, L.A. (2004). A fed-batch fermentation process identification and direct adaptive neural control with integral term. In: *MICAI 2004: Advances in Artificial Intelligence, LNAI 2972,* Monroy, R., Arroyo-Figueroa, G., Sucar, L.E., Sossa, H. (Eds.), page numbers (764-773), Springer-Verlag, ISBN 3-540-21459-3, Berlin Heidelberg New York

Baruch, I.S.; Georgieva P.; Barrera-Cortes, J. & Feyo de Azevedo, S. (2005a). 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, (O.Castillo, and P.Melin - guest editors),* Vol. 20, No 2, (February 2005) page numbers (173-194), ISSN 0884-8173

Baruch, I.S & Garrido, R. (2005b). 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, (O.Castillo, and P.Melin - guest editors),* Vol. 20, No 2, (February 2005) page numbers (213-224), ISSN 0884-8173

Baruch, I.S.; Hernandez, L.A.; & Barrera-Cortes, J. (2007a). Recurrent neural identification and sliding mode adaptive control of an aerobic fermentation plant. *Cientifica, ESIME-IPN,* Vol. 11, No 2, (March-April 2007), page numbers (55-62), ISSN 1665-0654

Baruch, I.S.; Hernandez, L.A.; Mariaca-Gaspar, C.R. & Nenkova, B. (2007b). An adaptive sliding mode control with I-term using recurrent neural identifier. *Cybernetics and Information Technologies, BAS, Sofia, Bulgaria,* Vol. 7, No 1, (January 2007) page numbers 21-32, ISSN 1311-9702

Baruch, I. S., Mariaca-Gaspar, C.R., and Barrera-Cortes, J. (2008). Recurrent neural network identification and adaptive neural control of hydrocarbon biodegradation processes. In: *Recurrent Neural Networks,* Hu, Xiaolin, Balasubramaniam, P. (Eds.), Chapter 4, page numbers (61-88), I-Tech Education and Publishing KG, ISBN 978-953-7619-08-4, Vienna, Austria

Baruch, I.S. & Mariaca-Gaspar, C.R. (2009). A Levenberg-Marquardt learning algorithm applied for recurrent neural identification and control of a wastewater treatment bioprocess. *International Journal of Intelligent Systems,* Vol. 24, No 11, (November 2009) page numbers (1094-1114), ISSN 0884-8173

Baruch, I.S. & Mariaca-Gaspar, C.R. (2010). Recurrent neural identification and control of a continuous bioprocess via first and second order learning, *Journal of Automation, Mobile Robotics and Intelligent Systems,* Vol. 4, No 4, (December 2010) page numbers (37-52), ISSN 1897-8645 (print), ISSN 2080-2145 (on-line)

Bernard, O.; Hadj-Sadok, Z.; Dochain, D.; Genovesi, A. & Steyer, J.P. (2001). Dynamical model development and parameter identification for an anaerobic wastewater treatment process. *Biotechnology and Bioengineering*, Vol. 75, No 4, (June 2001) page numbers (424–438), ISNN 1097-0290

Bialecki, B. & Fairweather G. (2001), Orthogonal spline collocation methods for partial differential equations, *Journal of Computational and Applied Mathematics*, Vol. 128, No 1-2, (March 2001) page numbers (55-82), ISSN 0377-0427

Boskovic, J.D. & Narendra, K. S. (1995). Comparison of linear, nonlinear and neural-network-based adaptive controllers for a class of fed-batch fermentation processes. *Automatica*, Vol. 31, No 6, (June1995) page numbers (817-840), ISSN 0005-1098

Bulsari, A. & Palosaari, S. (1993). Application of neural networks for system identification of an adsorption column. *Neural Computing and Applications*, Vol. 1, No 2, (June 1993) page numbers (160-165), ISSN 0941-0643

Deng H. & Li H.X. (2003). Hybrid intelligence based modeling for nonlinear distributed parameter process with applications to the curing process. *IEEE Transactions on Systems, Man and Cybernetics*, Vol. 4, No. 4, (October 2003) page numbers (3506 - 3511), ISSN 1062-922X

Deng H., Li, H.X. & Chen G. (2005). Spectral-approximation-based intelligent modeling for distributed thermal processes. *IEEE Transactions on Control Systems Technology*, Vol. 13, No. 5, (September 2005) page numbers (686-700), ISSN 1063-6536

Gonzalez-Garcia, R; Rico-Martinez, R. & Kevrekidis, I. (1998). Identification of distributed parameter systems: a neural net based approach. *Computers and Chemical Engineering*, Vol. 22, No 1, (March 2003) page numbers (965-968), ISSN 0098-1354

Haykin, S. (1999). *Neural Networks, a Comprehensive Foundation*, Second Edition, Section 2.13, page numbers (84-89); Section 4.13, page numbers (208-213), Prentice-Hall, ISBN 0-13-273350-1, Upper Saddle River, New Jersey 07458

Padhi, R.; Balakrishnan, S. & Randolph, T. (2001). Adaptive critic based optimal neuro-control synthesis for distributed parameter systems, *Automatica*, Vol. 37, No 8, (August 2001) page numbers (1223-1234), ISSN 0005-1098

Padhi, R. & Balakrishnan, S. (2003). Proper orthogonal decomposition based optimal neuro-control synthesis of a chemical reactor process using approximate dynamic programming. *Neural Networks*, Vol. 16, No 5-6, (June 2003) page numbers (719 - 728), ISSN 0893-6080

Pietil, S. & Koivo, H.N. (1996). Centralized and decentralized neural network models for distributed parameter systems. *Proceedings of CESA'96 IMACS Multiconference on Computational Engineering in Systems Applications*, page numbers (1043-1048), ISBN 2-9502908-9-2, Lille, France, July 1996, Gerf EC Lille, Villeneuve d'Ascq, FRANCE

Ray, W.H. (1989). *Advanced Process Control*, page numbers (133-242), Butterworths Publ., ISBN 0-409-90231-4, Boston, London, Singapore, Sydney, Toronto, Wellington

Schoefs, O.; Dochain, D.; Fibrianto, H. & Steyer, J.P. (2004). Modelling and identification of a partial differential equation model for an anaerobic wastewater treatment process. *Proceedings of 10th World Congress on Anaerobic Digestion*, page numbers (343 - 347), ISBN 1-843-39550-9, Montreal, Canada

Wan, E. & Beaufays, F. (1996). Diagrammatic method for deriving and relating temporal neural networks algorithms. *Neural Computations*, Vol. 8, No.1, (January 1996) page numbers (182–201), ISSN 0899-7667
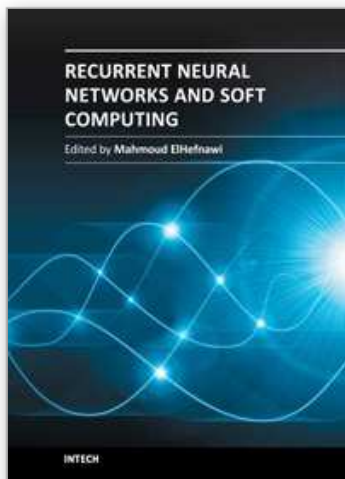
Recurrent Neural Networks and Soft Computing

Edited by Dr. Mahmoud ElHefnawi

How to reference

In order to correctly reference this scholarly work, feel free to copy and paste the following:

Ieroham Baruch, Eloy Echeverria-Saldierna and Rosalba Galvan-Guerra (2012). Centralized Distributed Parameter Bioprocess Identification and I-Term Control Using Recurrent Neural Network Model, Recurrent Neural Networks and Soft Computing, Dr. Mahmoud ElHefnawi (Ed.), ISBN: 978-953-51-0409-4, InTech, Available from: http://www.intechopen.com/books/recurrent-neural-networks-and-soft-computing/centralized-distributed-parameter-bioprocess-identification-and-i-term-control-using-recurrent-neura

INTECH

open science | open minds