Al-Nahrain University, College of Engineering Journal (NUCEJ) Vol.18 No.2, 2015 pp.202 - 207

Modeling of Continuous Stirred Tank Reactor based on Artificial Neural Network

Ahmed Sabah Al-Araji
Control and Systems Engineering Dept.
University of Technology

dr.ahmed_alaraji@uotechnology.edu.iq

Abstract

This paper presents the dynamic model identification algorithm of the continuous stirred tank reactor (CSTR) using a multi-layer perceptron (MLP) neural network topology. The neural network approach for (CSTR) dynamic modeling is trained by using a particle swarm optimization (PSO) technique as a simple and fast training unsupervised algorithm. Polywog wavelet activation function is used in the structure of MLP neural network. The identification algorithm given in this paper has been proved to be reasonable and precise via Matlab simulation results in terms of fast, stable and minimum number of fitness evaluation for the CSTR modeling.

Keywords: CSTR, MLP Neural Network, Particle Swarm Optimization.

Introduction

In control engineering, modeling and identification are important steps in the design of control, supervision and fault-detection system. Simulation modeling provides an effective and powerful approach for capturing and analyzing complex manufacturing systems based on computer generated data [1].

The problem of modeling of continuous stirred tank reactor (CSTR) is always attracting task for control system engineers because it's strong nonlinear behavior. Models can be used for simulations, analysis of the system's behavior, better understanding of the underlying mechanisms in the system, design of new processes and for controlling systems [2].

There are different modeling approaches for CSTR model as follows: Fuzzy clustering used for modeling the CSTR through a combination of local linear models as a means to capture global dynamic characteristics of complex CSTR system as explained in [2]. On-line identification is proposed in [3] that consisted of a modified growing and pruning algorithm for radial basis function (MGAP-RBF) neural network which used for affine modeling of nonlinear and time varying CSTR system. Also, in the real time as proposed in [4] a low cost embedded CSTR system based on an inexpensive microcontroller using the full neural networks for training and validation the system model using back propagation learning algorithm.

In addition to that, an on-line recursive least square identification method based on auto-regressive exogenous input (ARX) was used to have knowledge about dynamic behavior of CSTR system as explained in [5].

The CSTR model was identified with Hopfield network and it was relative degree, state variables and lie derivatives can be obtained from the identification network as a first order linear system as proposed in [6].

The main advantages of the presented approach are useful to build a precision nonlinear model from measured data by using a PSO as a simple steps proposed algorithm and fast the weights training of the neural network model to identify nonlinear systems accurately and to overcome the most important learning problems.

CSTR Mathematical Model

Consider standard two-state (CSTR) with an exothermic irreversible first-order reaction $A \rightarrow B$ take place, the heat of reaction is removed by a coolant medium that flows through a jacket around the reactor, as shown in figure (1) [4 and 5].

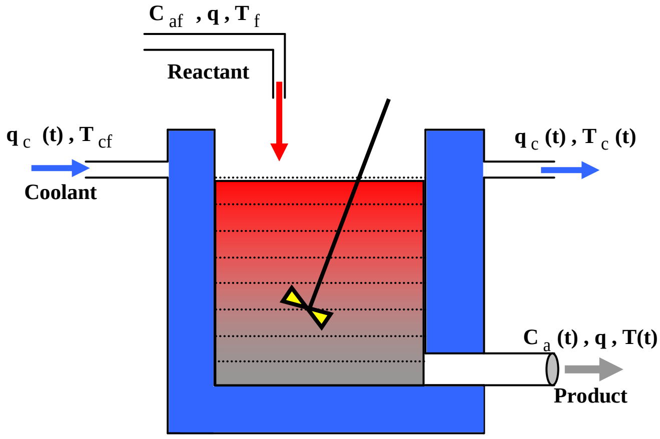

Figure 1: CSTR with cooling jacket [4,5].

The dynamics of system can be described by the following two nonlinear ordinary differential equations [7]:

$$\frac{\partial C_a(t)}{\partial t}=\frac{q}{Vol}(C_{af}-C_a(t))-K_o\times C_a(t)\times e^{\left(\frac{-E}{RT(t)}\right)}$$

$$\frac{\partial T(t)}{\partial t}=\frac{q}{Vol}(T_f-T(t))+\frac{(-\Delta H)\times K_o\times C_a(t)}{\rho\times C_\rho}\times e^{\left(\frac{-E}{RT(t)}\right)} \qquad \ldots.(1)$$

$$+\frac{\rho_c\times C_{\rho c}}{\rho_c\times C_{\rho c}\times Vol}\times q_c(t)\left(1-e^{\left(\frac{-h_a}{q_c(t)\times\rho_c\times C_{\rho c}}\right)}\right)\times(T_{cf}-T(t))$$

The nominal CSTR operating conditions can be shown in table (1).

Table 1: Nominal CSTR operating conditions [7]

Parameter	Description	Nominal Value
q	Process flow-rate	100 $lmin^{-1}$
$\mathbf{C_{af}}$	Inlet feed concentration	1 mol l^{-1}
$\mathbf{T_f}$	Feed temperature	350K
$\mathbf{T_{cf}}$	Inlet coolant temperature	350K
Vol	Reactor volume	100 l
$\mathbf{h_a}$	Heat transfer coefficient	$7*10^{10}$ cal $min^{-1}.K^{-1}$
$\mathbf{k_o}$	Reaction rate constant	$7.2*10^{10}$ Min^{-1}
E/R	Activation energy	$9.95*10^3$ K
ΔH	Heat of reaction	$2*10^5$ cal mol^{-1}
ρ, ρ_c	Liquid densities	1000 g l^{-1}
$\mathbf{C_p}$ $\mathbf{C_{pc}}$	Specific heats	1 cal g^{-1}. K^{-1}
$\mathbf{q_c}$	Coolant flow-rate	103.41 $l.min^{-1}$
T	Reactor temperature	440.2K
$\mathbf{C_a}$	Product concentration	$8.36*10^{-2}$ mol l^{-1}

Modeling Approach

To describe the dynamics model of (CSTR) by using multi-layered feedforward neural network (MLFNN), as shown in figure (2) which consist of the nodes of input layer, hidden layer and output layer as (4-7-1) respectively.

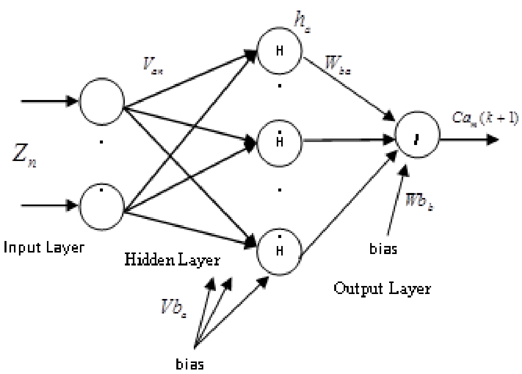

Figure 2: The MLFNN act as modeling [8].

The input and output units interact with the outside environment such as normalized and de-normalized data inputs and outputs respectively, while the hidden does not. The input units are only buffer units which pass the signals without changing them. The output unit is linear units [8]. The hidden units are non-linear Polywog wavelet activation functions [9], as shown in figure (3).

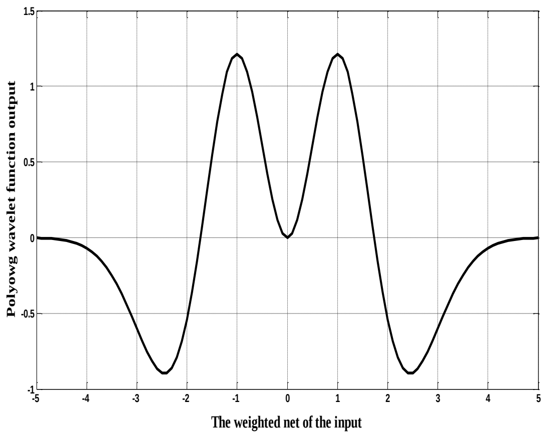

Figure 3: Polywog wavelet function

A multi-layer perceptron model is composed of many interconnected processing units called neurons or nodes, as shown in figure (2). The network notations are as follows:

V_{an} : Weight matrix of the hidden layers.

$\overline{Vb}_a$: Weight vector of the hidden layers.

W_{ba} : Weight matrix of the output layer.

$\overline{Wb}_b$: Weight vector of the output layer.

Where

n is equal to four.

a is equal to seven.

b is equal to one.

To explain these calculations, consider the general a'th neuron in the hidden layer shown in figure (2). The inputs to this neuron consist of an n– dimensional vector and (n'th is the number of the input nodes). Each of the inputs has a weight V associated with it. The first calculation within the neuron consists of calculating the weighted sum net_a of the inputs as [8 and 10]:

$$net_a=\sum_{a=1}^{nh}V_{an}\times\overline{Z_n}+bias\times\overline{Vb_a} \qquad \ldots..(2)$$

where

nh is number of the hidden nodes and $\overline{Z}$ is the input vector.

Next the output of the neuron h_a is calculated as the Polywog wavelet function of the net_a as:

$$h_a = H(net_a) \quad \text{….. (3)}$$

$$H(net_a) = (3(net_a)^2 - (net_a)^4)e^{-0.5(net_a)^2} \quad \text{….. (4)}$$

Once the outputs of the hidden layer are calculated, they are passed to the output layer. In the output layer, one linear neuron is used to calculate the weighted sum (neto) of its inputs.

$$neto_b = \sum_{b=1}^{nh} W_{ba} \times h_a + bias \times \overline{Wb}_b \quad \text{….. (5)}$$

where W_{ba} is the weight between the hidden neuron h_a and the output neuron. $\overline{Wb}$ is the weight vector for the output neuron. The one linear neuron, then, passes the sum ($neto_b$) through a linear function of slope 1 (another slope can be used to scale the output) as:

$$O_b = L(neto_b) \quad \text{….. (6)}$$

The output of the neural network is the modeling of the CSTR and can be defined as: $Ca_m(k+1)$.

PSO Learning Algorithm

Particle Swarm optimization (PSO) is a kind of unsupervised algorithm to search for the best solution by simulating the movement and flocking of birds. PSO algorithms use a population of individual (called particles) "flies" over the solution space in search for the optimal solution.

Each particle has its own position and velocity to move around the search space. The particles are evaluated using a fitness function to see how close they are to the optimal solution [11].

The previous best value is called as *pbest*. Thus, *pbest* is related only to a particular particle. It also has another value called *gbest*, which is the best value of all the particles *pbest* in the swarm.

The MLP neural network weight matrix is rewritten as an array to form a particle. Particles are then initialized randomly between ± 0.1 and updated afterwards according to equations (7 and 8):

$$\Delta w_{i,m}^{k+1} = \Delta w_{i,m}^{k} + c_1 r_1 (pbest_{i,m}^{k} - w_{i,m}^{k}) + c_2 r_2 (gbest_{m}^{k} - w_{i,m}^{k}) \quad \text{….. (7)}$$

$$w_{i,m}^{k+1} = w_{i,m}^{k} + \Delta w_{i,m}^{k+1} \quad \text{….. (8)}$$

$i = 1,2,3,.....pop$

$m = 1,2,3,.....D$

Where

pop is number of particles.

D is the dimension of particle.

$w_{i,m}^{k}$ is the weight of particle i at k iteration.

c_1 and c_2 are the acceleration constants with positive values equal to 1.25.

r_1 and r_2 are random numbers between 0 and 1.

$pbest_i$ is best previous weight of i[th] particle.

$gbest_m$ is best particle among all the particle in the population.

The number of dimension in particle swarm optimization neural network is referring to number of weight and bias of the MLPNN structure and can be described as equation (9):

Dimension = (input layer node × hidden layer node□) + (hidden layer node × output layer node) + hidden bias +output bias (9)

The mean square error function is chosen as criterion for estimating the model performance as the objective cost function as equation (10):

$$E = \frac{1}{pop}\sum_{j=1}^{pop}(Ca(k+1)^j - Ca_m(k+1)^j)^2 \quad (10)$$

The steps of PSO for learning MLP neural network can be described as follows:

- **Step1** Initial searching points w_1^0 and Δw_1^0 of each particle are usually generated randomly within the allowable range. Note that the dimension of search space consists of all the weights used in the MLP neural network, as shown in figure (2). The current searching point is set to *pbest* for each particle. The best-evaluated value of *pbest* is set to *gbest* and the particle number with the best value is stored.
- **Step2** The objective function value is calculated for each particle. If the value is better than the current *pbest* of the particle, the *pbest* value is replaced by the current value. If the best value of *pbest* is better than the current *gbest*, *gbest* is replaced by the best value and the particle number with the best value is stored.
- **Step3** The current searching point of each particle is update by using equations (7 and 8).
- **Step4** If the current iteration number reaches the predetermined maximum iteration number, then exit. Otherwise, go to step 2.

Results

The modeling and identifying are verified by means of computer simulation using Matlab m-file program.

The dynamic model of the CSTR described in section 2 is used. The objective is to model and identify the Ca(t) which can be done by introducing a coolant flow rate qc(t) as the manipulated variable, also the temperature can be varied too. To study the dynamic behavior of the CSTR model, the open loop output response of the CSTR for step changes in the coolant flow-rate are shown in figures (4-a & b) respectively.

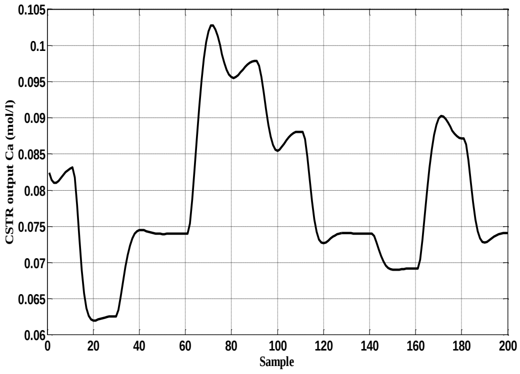

Figure (4-a): The CSTR open loop respose.

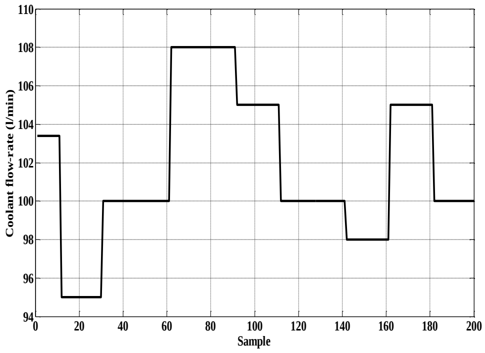

Figure (4-b): The coolant flow-rate step changes

As shown both the damping and the steady-state gain of the system varies considerably, depending on the set point, which gives an indication of the highly nonlinear dynamic behavior of the system.

The identification scheme of the nonlinear CSTR system is needed to input-output training data pattern to provide enough information about the dynamics CSTR model. This can be achieved by injecting a sufficiently rich input signal to excite all process modes of interest while ensuring that the training patterns adequately covers the specified operation region [12].

The training set is generated by solving the CSTR equation (1) for the input a pseudo random binary signal (PRBS) using the fourth order Range-Kutta method with sampling time of 0.1 minute.

It is very necessary to normalize the input signals of figure (5-a) and the desired output of figure (5-b) between (-1 to +1). The signals entered to or emitted from the network have been normalized to lie within (-1 to +1) in order to overcome numerical problems that is involved within real values.

Scaling functions have to be added at the neural network terminals to convert the scaled values to actual values and vice versa

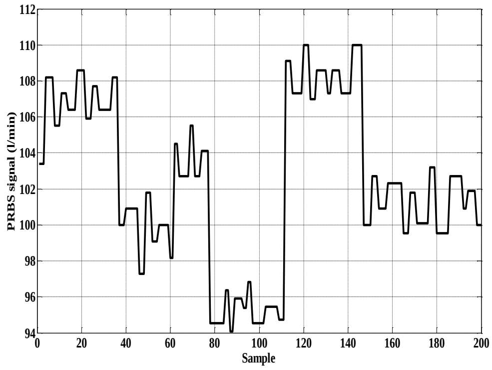

Figure (5-a): The PRBS input signal used to excite the system.

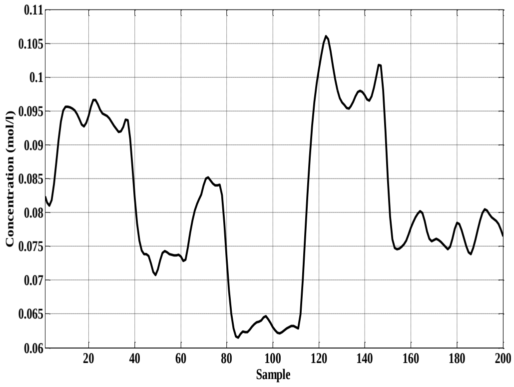

Figure (5-b): The CSTR open loop response to the PRBS input signal.

The proposed learning algorithm based particle swarm optimization is used with the MLPNN of the structure (4-7-1) as four nodes in the input layer, seven nodes in hidden layer and one node in output layers, as shown in figure (2).

A training set of 200 patterns has been used with the size of the particle dimension is equal to 43 by applying equation (9) and 20 particle numbers have been chosen.

Then using four steps of the proposed learning algorithm and after 100 iterations the mean square error has reached to less than 5×10^{-5} as shown in figure (6) with high speed of learning

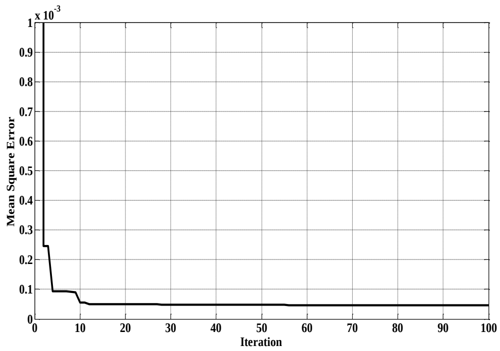

Figure 6: The objective cost function MSE.

Figure (7) shows the excellent learning for the CSTR neural model because it demonstrates the time response of the neural network model and the actual system output for the input learning set coolant flow-rate as well as reduces the output oscillation and minimizes the error between the actual output and neural network output.

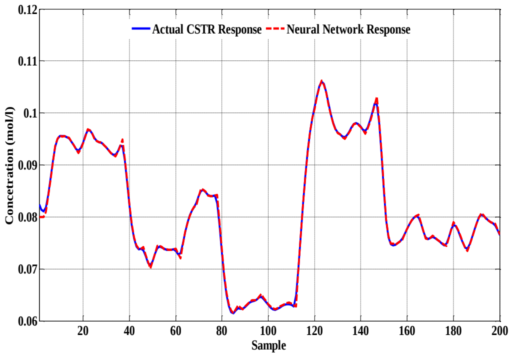

Figure 7: Neural network output and actual CSTR output for learning set

Figure (8) shows the verification of the CSTR neural network model which acts as a precious nonlinear dynamic behavior model, a coolant flow-rate testing set is used to feed the actual system and neural network model after that the responses of the actual system and neural model are obtained which showed high matching responses without any problems in the identification such as the over learning problem that its clear in [13].

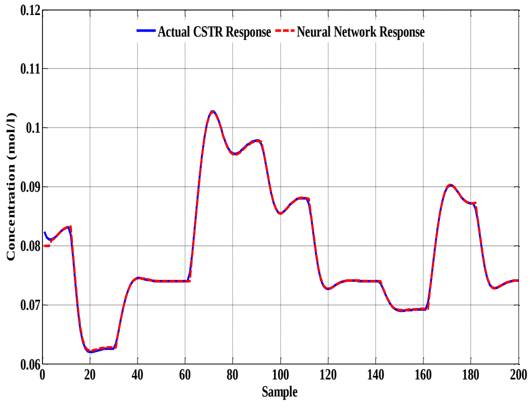

Figure 8: Neural network output and actual CSTR output for testing set.

Conclusions

This paper demonstrates the nonlinear CSTR neural model based on particle swarm optimization technique for learning the multi-layer perceptron neural network with Polywog wavelet activation function.

Simulation results via Matlab package demonstrate the neural network approach acting as a dynamic precious nonlinear behavior model for the dynamic continuous stirred tank reactor based on PSO was batter than the results in [13] in terms of the following:

- Increasing the speed of learning and minimizing the numbers of nodes in hidden layer.
- Reducing the output oscillation and minimizing the error between the actual output and neural network output.
- Overcoming the problem of the over learning in the identification system.

References

[1] X. Deng, "System Identification Based on Particle Swarm Optimization Algorithm". Proceedings of the 2009 International Conference on Computational Intelligence and Security, pp. 259-263, 2009.

[2] U.S. Banu, G. Uma, "Modelling of CSTR by Fuzzy Clustering". Proceedings of India International Conference on Power Electronics, pp. 371-376, 2006.

[3] A.S. Kamalabady, K. Salahshoor, "A new on-line identification approach for affine modelling of nonlinear processes using an adaptive neural network". Proceedings of the 16th Mediterranean Conference on Control and Automation, Congress Center, Ajaccio, France, June 25-27, pp. 735-740, 2008.

[4] L.S. Saoud, F. Rahmoune, V. Tourtchine, K. Baddari, "An Inexpensive embedded electronic continuous stirred tank reactor (CSTR) based on neural networks". Proceeding of the International Conference on Multimedia Technology (ICMT2011), pp. 6233-6237, 2011.

[5] M. Delbari, K. Salahshoor, B. Moshiri, "Adaptive generalized predictive control and model reference adaptive control for CSTR reactor". Proceedings of the International Conference on Intelligent Control and Information, Dalian, China, August 13-15, pp. 165-169, 2010.

[6] P. Guo, "Nonlinear predictive functional control based on hopfield network and its application in CSTR". Proceedings of the 5th International Conference on Machine Learning and Cybernetics, Dalian, August 13-16, pp. 3036-3039, 2006.

[7] E.P. Nahas, M.A. Henson, D.E. Seborg, "Nonlinear internal model control strategy for neural network models". Computers Chem. Eng., Vol. 16, no. 12, pp. 1039- 1057, 1992.

[8] J. M. Zurada, "Introduction to Artificial Neural Systems". (Jaico Publishing House, Pws Pub Co. 1992).

[9] E. Righeto, L-H M. Grassi, J.A. Pereira, "Onlinear plant identification by wavelet".

Proceeding of ABCM Symposium Series in Mechatronics, Vol. 1, pp. 392-398, 2004.

[10] S. Omatu, M. Khalid, R. Yusof, “Neuro - control and its applications”. London: Springer-Velag, 1995.

[11] A. Al-Araji, Designing a Nonlinear PID Neural Controller of Differential Braking System for Vehicle Model Based on Particle Swarm Optimization. Engineering & Technology Journal. Part (A), Vol. 32. No. 1, (2014), pp. 197-214.

[12] O. Nells, "Nonlinear system identification”. Springer – Verlag Berlin Heidelberg, 2001.

[13] A. Al-Araji, “Applying recurrent neural networks in modeling of continuous stirred tank reactor based on identification”. Proceeding of the Control, Computers and Mechatronics Engineering Conference ECCCM2, University of Technology, Baghdad-Iraq, February 25-27, pp. 106-112, 2014.

نموذج لخزان مفاعل مستمر الإثارة مبني على أساس الشبكة العصبية الذكية

أحمد صباح عبد الأمير الأعرجي
قسم هندسة السيطرة والنظم - الجامعة التكنولوجية

الخلاصة:

أن هذا البحث يقدم خوارزمية التعريف لنموذج ديناميكي لخزان مفاعل مستمر الإثارة (CSTR) باستخدام الشبكة العصبية متعددة الطبقات (MLPNN). لقد تعلمت الشبكة العصبية التي تمثل النموذج الديناميكي لخزان مفاعل مستمر الإثارة باستخدام تقنية حشد الجسيمات الامثلية لسهولة و سرعة هذه الخوارزمية للتعلم. وتم استخدام دالة التنشيط (Polywong Wavelet) في الشبكة العصبية.

نتائج المحاكات لهذه الخوارزمية التعريفية كانت معقولة و مضبوطة من خلال استخدام الحقيبة البرمجية ماتلاب من حيث سرعة واستقرارية مع أدنى عدد من الاستدعاء لداله التقييم لنموذج (CSTR).