Journal of Process Control 23 (2013) 1562–1566

Contents lists available at ScienceDirect

Journal of Process Control

journal homepage: www.elsevier.com/locate/jprocont

Short communication

Integrated OBF-NN models with enhanced extrapolation capability for nonlinear systems

H. Zabiri*, M. Ramasamy, Lemma D. Tufa, A. Maulud

Chemical Engineering Department, Universiti Teknologi PETRONAS, Bandar Seri Iskandar, Tronoh, 31750 Perak, Malaysia

ARTICLE INFO

Article history:
Received 2 January 2013
Accepted 20 June 2013
Available online 18 July 2013

Keywords:
Nonlinear systems
Residuals
Parallel integration
Orthonormal basis filters
Neural networks
Range extrapolation

ABSTRACT

This paper proposes a nonlinear system identification using parallel linear-plus-neural network models that provide more accurate predictions on the process behavior even on extrapolated regions. For this purpose, a residuals-based identification algorithm using parallel integration of linear orthonormal basis filters (OBF) and neural networks model is developed and analyzed under range extrapolations. Results on the van de Vusse reactor case study show enhanced extrapolation capability when compared to the conventional neural network (NN) and the series Wiener-NN models.

1. Introduction

In nonlinear system identification using black box models such as neural networks (NN), one possible approach is to use a parallel combination of linear-plus-NN models. Sjoberg et al. [1] suggested via their personal communication with McAvoy that the residuals from a linear model may be used to develop an NN model to pick up the nonlinearities in nonlinear system identification. This parallel combination through the usage of residuals is very attractive in two ways: viz. (1) a nonlinear model that is not properly developed performs worse than a linear one, hence by having a linear model developed in the first step ensures that reasonable models are obtained [1], and (2) applying the NN on the residuals (inputs and residuals as network input and output) ensures that the overall nonlinear model performs at least as good as or better than the linear model [1]. The usage of residuals provides another interesting perspective. In [2], it is stated that there are cases where residuals are not due to randomness and may actually inherit the characteristics of the original system. Under such circumstances, a high correlation in the residuals is usually observable, indicating the failure of the model to capture the full characteristics of the underlying system.

Even though the idea is not new, no guidelines for the selection of linear structures that will be effective for practical identification nor any analysis on the extrapolation capability of such combination are reported in literature. Residuals analysis has been explored as part of an identification method that involved embedding theorem and hybrid Elman-NARX network for predicting chaotic time series data [2].

Based on these gaps, this paper proposes a nonlinear system identification using parallel linear-plus-NN models that provide more accurate predictions on the process behavior even on extrapolated regions. In this work, the justification on how the proposed parallel linear-plus-NN models help in improving the range extrapolation capability of conventional NN is presented. The selection of effective linear structures that is useful for practical identification is also elaborated. A case study is used to illustrate the identification and the extrapolation capabilities of the proposed structure. The full extent of the extrapolation capability of such models is extensively analyzed under range extrapolations, which are typically encountered in practice in process industries.

2. Residuals-based parallel OBF-NN model

2.1. Model structure

Consider a general nonlinear output error (NOE) model structure expressed as

$$y(k) = f(u(k-1),\ \dots, u(k-m), \hat{y}(k-1), \dots, \hat{y}(k-m)) + e(k) \qquad (1)$$

* Corresponding author. Tel.: +60 5 3687625; fax: +60 5 3656176.
E-mail addresses: haslindazabiri@petronas.com.my, zhaslinda@yahoo.com (H. Zabiri).

0959-1524/$ – see front matter © 2013 Elsevier Ltd. All rights reserved.
http://dx.doi.org/10.1016/j.jprocont.2013.06.005

where $e(k)$ refers to the system white noise. A general linear model structure, on the other hand, may be represented as

$$y(k) = G(q)u(k) + e(k) \tag{2}$$

Without loss of generality (1) and (2) can be combined to get

$$y(k) = G(q)u(k) + f(u(k-1),\ \ldots,u(k-m),\\ \hat{y}_r(k-1),\ \ldots,\hat{y}_r(k-m)) + e(k) \tag{3}$$

where $\hat{y}_r$ refers to the predicted residuals of the linear model, i.e. $\hat{y}_r = y_{\text{measured}} - \hat{y}_{\text{linear}}$. Equation (3) represents a parallel structure in which a linear model is combined with a non-linear model represented by $f(\cdot)$.

In this paper, orthonormal basis filters (OBF) (e.g. Laguerre, Kautz filters) is selected to represent the linear model. OBF models have recently found widespread applications in linear system identification [3]. OBF models have several characteristics that make them very promising for control relevant system identification. Their parameters can be easily estimated using linear least square method. They are consistent in their parameters for most practical open-loop identification problems and time delays can be easily estimated and incorporated into the model.

It is known that OBF based model has qualities that are very attractive to capture linear dynamics [3]. One of the major reasons for this is the fact that the OBF model allows one or more poles of the system to be introduced in the model which renders OBF model its gray nature. The OBF model is expressed as

$$y(k) = \left(\sum_{j=1}^{N} c_j L_j(q)\right) u(k) + e(k) \tag{4}$$

where N is the number of orthonormal basis filters, c_j are the optimal OBF model parameters, $L_j(q)$ are the orthonormal basis filters, q is the forward shift operator, $u(k)$ is the input to the system, and $e(k)$ is the system white noise. Note that in (4), the poles of the system are part of the orthonormal filters construction, and can be seen as the incorporation of prior knowledge of the system dynamics into the model [3].

For the nonlinear model structure, we propose that the nonlinearity in a system be effectively captured by the multi-layer perceptron (MLP) neural networks, yielding the proposed parallel OBF-NN model. MLP network is chosen due to its simpler structure and a fewer parameters in comparison to recurrent neural networks (RNN). Though not thoroughly covered here, MLP and RNN are just two out of many neural networks. However, not all are equally suitable for modeling and control of dynamic systems, and the most commonly adopted network for these purposes is the MLP [4].

Hence, for a SISO system with a MLP neural network with one hidden layer in parallel with a linear OBF model of (4), the one-step ahead prediction from (3) becomes

$$\hat{y}(k+1) = \left(\sum_{j=1}^{N} c_j L_j(q)\right) u(k) + \beta\left[b^2 + \sum_{i=1}^{K} w_i^2 \varphi(b_i^1 + w_{i,1}^1 x(k))\right] \tag{5}$$

where the nonlinear neural network function approximation is trained with regression vectors consisting of previous plant inputs and previous residuals of the linear model, $x(k) = [u(k-1),\ \ldots,u(k-m), \hat{y}_r(k-1),\ \ldots,\hat{y}_r(k-m)]$. Also $\varphi, \beta: R \to R$ are the nonlinear activation functions (e.g. hyperbolic tangent etc.), b are the biases, K is the number of hidden neurons, and the weights of the network are denoted by $w_{i,j}^1$, $i = 1,\ \ldots,K$ (with ith neuron and jth input, in this case $j = 1$) for the first layer, and w_i^2, $i = 1,\ \ldots,K$ for the second layer.

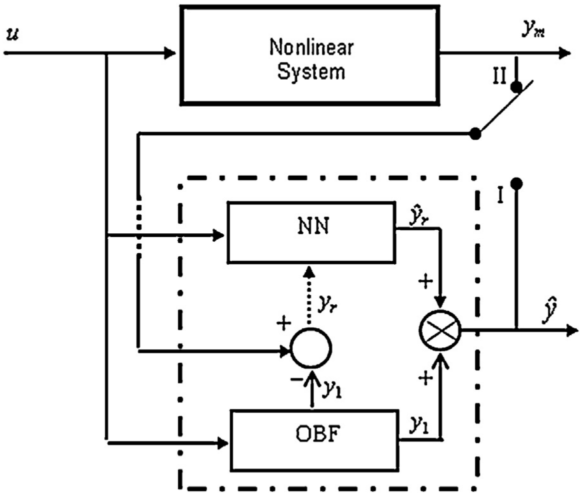

Fig. 1. The proposed sequential identification of residuals-based parallel OBF-NN models (I: simulation configuration, II: prediction configuration).

2.2. Parameter estimation

The sequential identification structure proposed for the residuals-based parallel OBF-NN models is illustrated in Fig. 1. The linear OBF model is identified first, and the nonlinear NN model is then trained with the predicted residuals. The *pseudo*-independent nature of this parallel structure allows both the models to capture the essential characteristics of the underlying process separately and hence more accurately.

Given a set of nonlinear data to be identified $[u(k), y_m(k)]$, the algorithm can be described as follows:

1. Develop a parsimonious OBF model using methods described by [3] to get y_1.
2. Calculate the predicted residuals using $\hat{y}_r = y_m - y_1$.
3. Develop the NN model using standard BP algorithm with $x(k) = [u(k-1),\ldots,u(k-m), \hat{y}_r(k-1),\ldots,\hat{y}_r(k-m)]$ as inputs and $\hat{y}_r(k)$ as outputs of the NN model.

2.3. Extrapolation using residuals based parallel OBF-NN model

Consider a single hidden layer MLP NN, with hyperbolic tangent activation functions in the hidden layer, and a linear output layer with coefficient of 1. The general equation for the configuration with one input, one output and two hidden nodes can be written as

$$\hat{y}(k+1) = w_1^2 \tan\ h(b_1^1 + w_{1,1}^1 x(k)) + w_2^2 \tan\ h(b_2^1 + w_{2,1}^1 x(k)) + b^2 \tag{6}$$

Equation (6) refers to the MLP NN model output. For OBF-NN model, the corresponding equation can be derived from (5) assuming that the residuals neural network configuration of OBF-NN has the same configurations as (6).

$$\hat{y}(k+1)\\ = y_{\text{OBF}} + \left[\underbrace{\tilde{w}_1^2 \tan\ h(\tilde{b}_1^1 + \tilde{w}_{1,1}^1 \tilde{x}(k)) + \tilde{w}_2^2 \tan\ h(\tilde{b}_2^1 + \tilde{w}_{2,1}^1 \tilde{x}(k)) + \tilde{b}^2}_{\text{residuals-NN}}\right] \tag{7}$$

where ∼ sign refers to the corresponding weights, biases and input regression vectors for the residuals network.

From (6) and (7), it is obvious that the output of the hidden nodes in both networks is governed by the hyperbolic tangent function. Similarly, if another activation function is used (i.e. logistic function), the output from the hidden layer's nodes would be governed by that function. If these hidden nodes outputs were to be plotted for the networks in (6) and (7), they both would resemble a hyperbolic tangent curve within the saturation points of [-1,1]. During training, the network translates and scales these hidden layer curves within the hard limits of [-1,1] until the best fit for the training range is obtained. This is done by adjusting the weights and biases. The network weights determine the activation function's slopes, and the biases determine the activation function's position [4]. Hence one would expect that both networks (or any networks in general) would not be able to handle satisfactorily data that are beyond the original range of that used during training.

However, what greatly differentiates between the two network models is the nature of the data that the networks work upon. In the conventional MLP NN given by equation (6), the network deals with actual values of the process variables. In contrast, the residuals network in (7) handles only residuals values of the process variables. The linear OBF model in (7) acts as an excellent base for the residuals network, provided that the parameters are chosen adequately. Linear model usually captures the average dynamics of the nonlinear process, and more importantly, its extrapolation behavior is linear which is preferable for dynamic systems. The linear OBF model ensures that the inputs to the residuals network are always not too far off from the original training range.

Therefore, by combining a linear, gray box model (OBF) and a nonlinear model (NN), the dynamics of a nonlinear system can be effectively captured with a good potential for extrapolation. In the next section, a van de Vusse reactor case study is presented to illustrate the efficacy of the proposed method in range extrapolation analysis.

3. Case study

In this section, the identification and *range* extrapolation capabilities of the proposed model structure are compared with that of pure linear OBF, conventional MLP network as well as with series Wiener-MLP structure [5] using the nonlinear van de Vusse reactor which is frequently used as a benchmark test problem for various identification and control strategies [6].

The linear component is identified using Laguerre filters as described in [3]. The input data to the NN, i.e., $x(k) = [u(k-1), \ldots, u(k-m), \hat{y}_r(k-1), \ldots, \hat{y}_r(k-m)]$, is segregated as training and validation sets (75% for training, and 25% for validation) as is normally done with any NN modeling. Standard approach in developing any NN model is by using the trial-and-error method [4] in determining the number of layers, the number of neurons and the transfer functions in each layer. In this work, a single hidden layer MLP network is used since it is the most common configuration adopted. The transfer functions are fixed with hyperbolic transfer functions in the hidden layer for all models. The number of hidden layer neurons is allowed to vary from 4 to 30, and the one that gives the lowest error is selected. Root mean square error (RMSE) is used as the convergence criteria.

The performance of the proposed model is compared with conventional MLP NN and the series Wiener-MLP developed by Saha

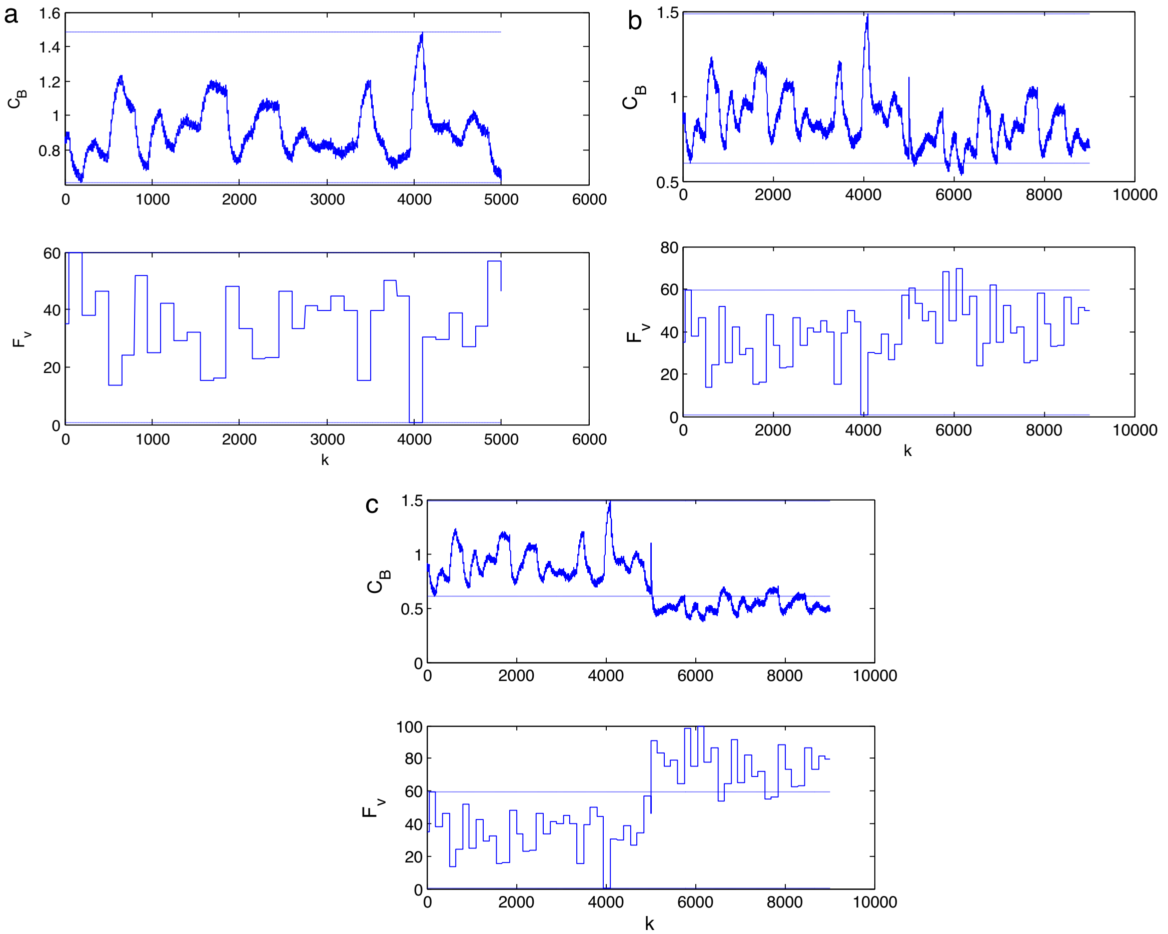

Fig. 2. Input–output data set for the van de Vusse reactor.

et al. [5]. In [5] specifically, the separate blocks of the Wiener models are represented by Laguerre filters and MLP NN in series, respectively, and the models are solved sequentially. The Laguerre parameter 'p' is chosen based on the step response of the process at its nominal operating point, as 'p' dictates the dominant pole(s) of the process. After fixing a proper value of 'p' the network is then trained. Inputs to the NN consists of outputs of the Laguerre filters.

In the van de Vusse reactor [6], reactant A is to be converted to the desired product B, but the product B degrades to by-product C. In addition to this consecutive reaction, a high-order parallel reaction occurs by which the reactant A is converted to by-product D.

$$A\xrightarrow{k_1}B\xrightarrow{k_2}C$$

$$2A\xrightarrow{k_3}D$$

The mathematical model of this reactor is described by the following set of ordinary differential equations (ODE):

$$\frac{dc_A}{dt}=\frac{q_r}{V_r}(c_{A0}-c_A)-k_1c_A-k_2c_A^2$$

$$\frac{dc_B}{dt}=-\frac{q_r}{V_r}c_B+k_1c_A-k_2c_B$$

$$\frac{dT_r}{dt}=\frac{q_r}{V_r}(T_{ro}-T_r)-\frac{\Delta H_r}{\rho_r c_{pr}}+\frac{A_rU}{V_r\rho_r c_{p_r}}(T_c-T_r)$$

$$\frac{dT_c}{dt}=\frac{1}{m_c c_{pc}}(Q_c+A_rU(T_r-T_c))$$

The net heat of reaction (ΔH_r) for the above reactions is expressed as:

$$\Delta H_r=\Delta h_1k_1c_A+\Delta h_2k_2c_B+\Delta h_3k_3c_A^2$$

where Δh_i refers to heat of reactions. Nonlinearity can be found in reaction rates (k_j) which are described via the Arrhenius expression:

$$k_j(T_r)=k_{0,j}\exp\left(\frac{-E_j}{RT_r}\right),\quad \text{for } j=1,\ 2,\ 3$$

where k_{0j} represents the pre-exponential factors and E_j are activation energies. Fixed parameters of the system are taken from [6].

Nonlinear system identification is carried out for the SISO system by considering the dynamic characteristics from the changes in the space velocity, $F_v = q_r/V_r$ (h^{-1}), and the product outlet concentration, C_B (kmol/m^3). The training set of the input–output data for the range extrapolation study are as shown in Fig. 2(a). The simulated data is generated by introducing a 'Random number' input signal to the system with mean and variance of 20 and 200, respectively.

The OBF model for the OBF-NN is developed with 6 Laguerre filters and one pole at 0.9048. The estimated OBF parameters are

$$c_{\text{OBF-NN}}=[-8.5179\times10^{-4}-7.6897\times10^{-4}-2.8837\times10^{-4}$$
$$-0.0010\,1.7712\times10^{-4}-0.0014]$$

The NN model for the residuals consists of 5 hidden neurons. The identified Wiener-MLP model has 9 hidden neurons in its NN subsystem, and the OBF model has one pole at 0.9801 with 6 Laguerre filters. The estimated OBF parameters are

$$c_{\text{Wiener-MLP}}=[-0.0013-1.7215\times10^{-5}\,5.2602\times10^{-5}\,8.5982$$
$$\times10^{-6}\,5.5387\times10^{-5}-3.3944\times10^{-6}]$$

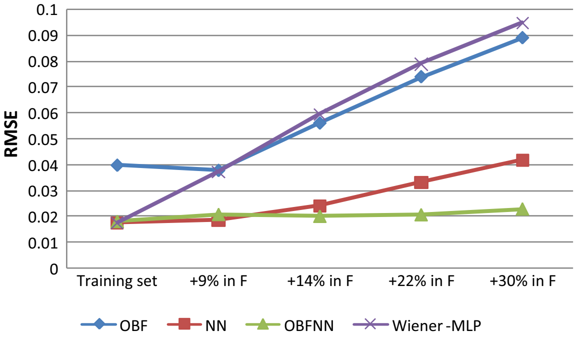

Fig. 3. Prediction errors for the training and extrapolation data sets under medium nonlinearity conditions.

The identified conventional MLP (NN) model has 19 neurons in its hidden layer. Fig. 2(b) shows the extrapolation data set up to an average of 9% increase in F_v beyond the original training range, and Fig. 2(c) shows the extrapolated data set up to an average of 30% increase in F_v.

The corresponding prediction errors are represented in Fig. 3. As can be observed, the RMSE errors for the training set are similar for all nonlinear models, except for the linear OBF model. However,

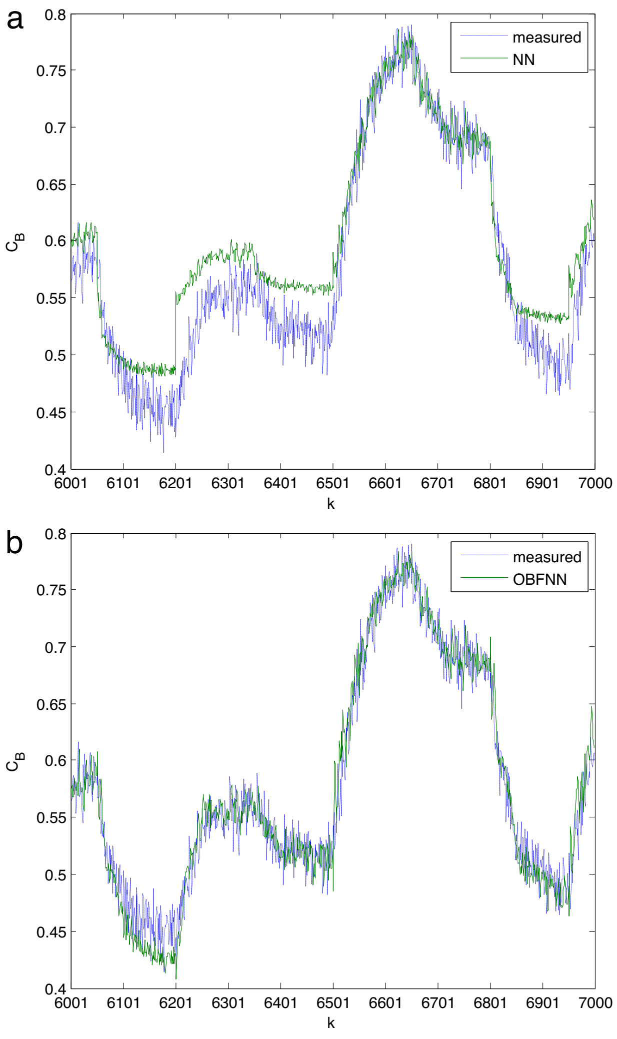

Fig. 4. Range extrapolation analysis: performance comparison.

when subjected to data that slowly drifts away from the original training set, it is observed that the proposed parallel OBF-NN model has superior extrapolation performance in comparison to the series Wiener-MLP model, as well as against conventional MLP model. Conventional MLP model failed to properly estimate the values outside the original training range. Series Wiener-MLP structure on the other hand, relies on pure linear behavior when subjected to extrapolation. The corresponding extrapolation errors for this model are almost equivalent to that of pure linear OBF model, as observed from Fig. 3.

Fig. 4 shows the measured and predicted values of the product outlet concentration, C_B (kmol/m^3) for an average increase of 22% in the F_v (h^{-1}) beyond the original training range. Excellent prediction performance is visible for the proposed OBF-NN model.

Comparing the series and parallel structures used in this paper, the linear OBF model in both cases may give, in general, the same predicted output provided that the number of Laguerre filters used is the same. However, since the identification is done sequentially in [5], whatever error exists during the calculation of the Laguerre filters output in the series structure is carried forward to the NN predictions. Hence, the purely linear extrapolation behavior observed in Fig. 3. In contrast, the parallel structure depicted in Fig. 1 takes into account both the linear and NN models predictions *pseudo*-independently, and this enhanced the overall extrapolation behavior.

4. Conclusion

In this paper, a residuals-based algorithm using parallel integration of linear orthonormal basis filters (OBF) and a multilayer perceptrons (MLP) neural networks model is developed and analyzed under range extrapolations. Results on the nonlinear van de Vusse reactor case study show the enhanced range extrapolation capability of the proposed structure when compared to the conventional MLP NN and the series Wiener-MLP models.

Acknowledgement

The authors would like to thank Universiti Teknologi PETRONAS for the funding and the support provided for this work.

References

[1] J. Sjoberg, Q. Zhang, L. Ljung, A. Benveniste, B. Deylon, P. Glorennec, H. Hjalmarsson, A. Juditsky, Nonlinear black-box modeling in system identification: a unified overview, Automatica 31 (1995) 1691–1724.

[2] M. Ardalani-Farsa, S. Zolfaghari, Chaotic time series prediction with residual analysis method using hybrid Elman-NARX neural networks, Neurocomputing (2010), http://dx.doi.org/10.1016/j.neucom.2010.06.004.

[3] Lemma D. Tufa, M. Ramasamy, M. Shuhaimi, Improved method for development of parsimonious orthonormal basis filter models, Journal of Process Control 21 (2011) 36–45.

[4] M. Norgaard, O. Ravn, N.K. Poulsen, L.K. Hansen, Neural Networks for Modeling and Control of Dynamic Systems: A Practitioner's Handbook, Springer-Verlag London Ltd., 2000.

[5] P. Saha, S.H. Krishnan, V.S.R. Rao, S.C. Patwardhan, Modeling and predictive control of MIMO nonlinear systems using Wiener–Laguerre models, Chemical Engineering Communication 191 (2004) 1083–1119.

[6] J. Vojtesek, P. Dostal, R. Haber, Simulation and control of a continuous stirred tank reactor, in: Proc. of the Sixth Portuguese Conference on Automatic Control CONTROLO, 2004, pp. 315–320, Faro Portugal.