Preprints of the 2012 20th Mediterranean Conference on Control & Automation (MED), Barcelona, Spain, July 3-6, 2012

Software sensors for biomass concentration in a SSC process using Artificial Neural Networks and Support Vector Machine

Gonzalo Acuña, *Senior Member, IEEE*, Cristián Ramirez and Millaray Curilem

Abstract— **In this work NARX-ANN, NARMAX-ANN and NARX-SVM models are compared when acting as software sensors of a relevant state variable for a Solid-substrate cultivation (SSC) process. Results show that NARX-SVM outperforms the other models with an Index of Agreement close to 1.0 even under very noisy conditions thus confirming the claimed superiority of SVM over other black-box techniques for approximating non-linear functions. NARMAX-ANN outperforms NARX-ANN because of its better predictive capabilities.**

I. Introduction

THE lack of appropriate sensors greatly affects the capability of developing good control and optimization procedures in some non-linear industrial processes like submerged or solid state fermentations [1].

Solid-substrate cultivation processes offer a number of advantages over submerged fermentations including higher productivity, lower operating costs and less downstream processing. Unfortunately it is very difficult to have on-line measurements of some of the most relevant state variables of SSC processes -like biomass concentration- because of the lack of sensors in a pilot scale or in a industrial scale level.

Software sensors emerged as reliable and useful techniques to perform on-line observation of difficult-to-measure state variables. These sensors relate the relevant state variables with the measured outputs normally using static or dynamic models of the process.

Unfortunately, the complexity of the SSC process (distributed, non-linear and time-variant) makes the development of suitable dynamic models based on laws very difficult. A successful alternative approach to address this problem is to design data-driven models. In that sense, in the field of linear systems there is a vast literature originated in the pioneering work of Box and Jenkins [2]. They proposed the development of models that consider that the evolution of a phenomenon can be explained by its previous behavior and by the effect of exogenous variables when they exist.

In case of non-linear phenomena, nonlinear autoregressive models, NAR, or nonlinear autoregressive with exogenous variables (NARX) are proposed. The predictive power of these models can be increased when previous errors are incorporated as regressors. This results in the so called nonlinear autoregressive moving average model NARMA or NARMAX when also exogenous variables are included. These models may be better predictors because they use information about past errors in order to improve the prediction. The price to pay is that they are more difficult to identify.

Extensive use of Artificial Neural Networks (ANN) in the field of system identification, predictive control, design of observers and predictors can be found in the literature [3, 4]. Despite the good results achieved with ANN some difficulties remain in its design, such as choosing the number of neurons in the hidden layers, the problem of overfitting, the existence of local minima in the objective function and the low capacity of generalization, among others.

Support Vector Machine (SVM) are considered as very efficient tools for classification and regression. They have many advantages such as good generalization ability and an optimization process based on a convex function with no local minima [5]. In the case of dynamical systems, almost all works that use SVM focus on NARX type models [6].

In this work we develop software sensors for the biomass concentration of a SSC process based on dynamic NARX and NARMAX models. These models are elaborated using ANN and SVM. Their robustness is tested under different levels of noise.

II. Model Identification

A. NARX

A non-linear regressive model with exogenous input (NARX) is the extension of an ARX model and is given by equation (1) [7]:

$$y_t = f(y_{t-1}, \ \ldots \ y_{t-n}, u_{t-1}, \ldots u_{t-m}) + e_t \qquad (1)$$

Manuscript received January 29, 2012. This work was supported in part by the Chilean Government under Grant 1090316, the DICYT-USACH Grant 061219AL and the Dirección de Investigación, UFRO.

G. Acuña is with the Departamento de Ingeniería Informática, Universidad de Santiago de Chile, USACH, Av. Ecuador 3659, Santiago 833011, Chile (phone: 56-2-7180920; fax: 56-2-7763511; e-mail: gonzalo.acuna@usach.cl).

C. Ramirez is with the Departamento de Ingeniería Informática, Universidad de Santiago de Chile, USACH, Av. Ecuador 3659, Santiago 833011, Chile (e-mail: cristian.ramirez@gestran.cl).

M. Curilem is with the Departamento de Ingeniería Eléctrica, Universidad de la Frontera, Temuco, Chile (e-mail: millaray@ufro.cl)

where e_t, is the prediction error at time t and is modeled as a Gaussian white noise zero mean process with variance σ. It represents the model uncertainty and the noise of the experimental data. The predictor associated with this type of models is given by equation (2) and is outlined in Figure 1, where $\hat{y}$ is the prediction of the autoregressive variable y from previous experimental data of itself and of the exogenous variable u in times $t-1$, $t-2$, ... considering the nonlinear function ψ [7].

$$\hat{y}_t = \psi(y_{t-1}, \ \ldots \ y_{t-n}, u_{t-1}, \ldots u_{t-m}) \tag{2}$$

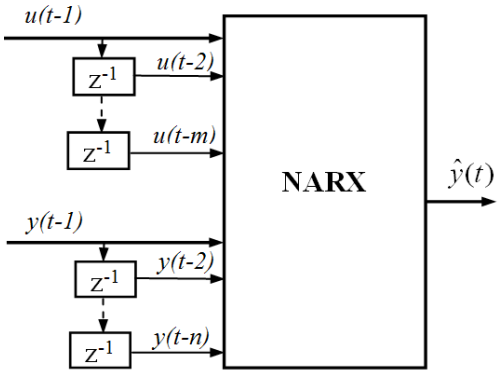

Figure 1. Associated predictor for NARX-type models. u is the exogenous input and y the autoregressive variable. $\hat{y}$ corresponds to the prediction of the autoregressive variable.

From Figure 1 it can be seen that for identification of such models the simple series-parallel identification method can be used. Indeed, it is enough to provide the following to the chosen approximator (ANN or SVM) of the prediction function ψ.

As input:

- experimental data for the autoregressive variables from *t*-1 to *t*-n (n has to do with the order of the system model).
- experimental data of the exogenous variables from *t*-1 to *t*-m.

As output:

- experimental data of autoregressive variables at a later time, *t*.

In the case of ANN the training algorithm consists of the well known backpropagation. In the case of SVM the quadratic programming (QP) algorithm is used. To adjust the output to the prediction function, it is necessary to provide adequate input and output to the optimization methods [5]. This process also requires adjusting some parameters that will be discussed in section IV-C and IV-D.

A. *NARMAX*

A non-linear auto regressive moving average model with exogenous input (NARMAX) is the extension of an ARMAX model and is given by equation (3) [7].

$$y_t = f(y_{t-1}, \ \ldots \ y_{t-n}, u_{t-1}, \ldots u_{t-m}, e_{t-1} \ldots e_{t-p}) + e_t \tag{3}$$

where e_t correspond to the random variable described above for Eq. (1). This term correspond to the prediction error from t-1 to t-p (e.g. $e_{t-1} = y_{t-1} - \hat{y}_{t-1}$).

The predictor associated with this kind of model is given by Equation 4 [7] and is outlined in Figure 2:

$$\hat{y}_t = \psi(y_{t-1}, \ \ldots \ y_{t-n}, u_{t-1}, \ldots u_{t-m}, e_{t-1} \ldots e_{t-p}) \tag{4}$$

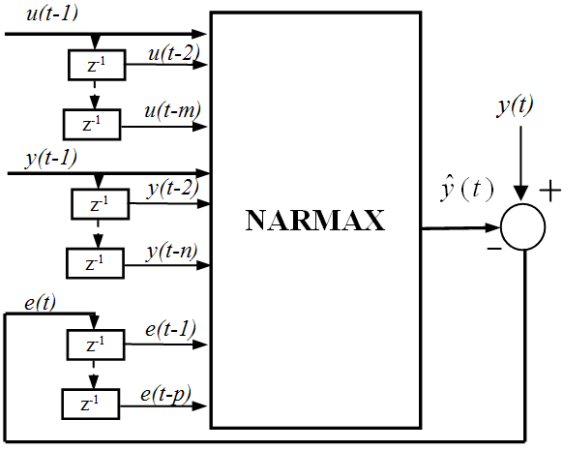

Figure 2. Associated predictor for NARMAX-type models. u is the exogenous input, y the autoregressive variable and e the predictive error. $\hat{y}$ corresponds to the prediction of the autoregressive variable.

To identify this kind of model it is necessary to have the values of previous prediction errors. The predictor has then to be used during training in order to obtain those previous prediction errors. This makes the identification, also known as parallel method of system identification, much more complex to perform than the series-parallel identification previously presented for NARX-type models.

In the case of ANN the required training algorithm consists now of the backpropagation-through-time method [8].

III. SSC PROCESS

The bioprocess considered is the production of gibberelic acid (a vegetal growth hormone) by the filamentous fungi *Gibberella fujikuroi* growing in a solid state batch culture (SSC) at a laboratory level. A simplified model describing the evolution of the main variables is reported in [9]. This phenomenological model based on material balance laws considers 7 state variables: living Biomass (X), measured or total Biomass (X_m), urea (U), intermediate nitrogen (N_I), soluble starch (S), produced CO_2 and consumed O_2. Dead biomass is not considered as a nutrient. Only the last two variables can be easily measured on-line but we will also consider X_m as an on-line measured variable. The model equations are the following:

$$\frac{dX_m}{dt} = \mu \cdot X \tag{5}$$

$$\frac{dX}{dt} = \mu \cdot X - k_d \cdot X \tag{6}$$

$$\frac{dU}{dt} = -k \tag{7}$$

$$\frac{dN_I}{dt} = \begin{cases} 0,47 \cdot k - \mu \cdot \left(\frac{X}{Y_{X/N_I}} \right), si\ U \geq 0 \\ -\mu \cdot \left(\frac{X}{Y_{X/N_I}} \right), U(t) = 0, si\ U < 0 \end{cases} \tag{8}$$

$$\frac{dS}{dt} = -\frac{\mu \cdot X}{Y_{X/S}} - m_s \cdot X \tag{9}$$

$$\frac{dCO_2}{dt} = \mu \cdot \frac{X}{Y_{X/CO_2}} + m_{CO_2} \cdot X \tag{10}$$

$$\frac{dO_2}{dt} = \mu \cdot \frac{X}{Y_{X/O_2}} + m_{O_2} \cdot X \tag{11}$$

μ, corresponds to the specific growth rate and its intermediate nitrogen dependence is modeled by a Monod law.

$$\mu = \mu_m \cdot \frac{N_I}{k_n + N_I} \tag{12}$$

The other model parameters were identified on the basis of some specific experiments. Their values are included in Table I for controlled temperature and water activity conditions (T=25ºC, Aw=0,992) [9].

One of the main problems is to have on-line measurements of living biomass. This is a very hard to measure variable and it requires off-line laboratory processing. Hence, it is useful to construct a software sensor for this specific variable in order to support the control and optimization of the process. The software sensor stated as a dynamic model allows the prediction of living biomass from its estimated previous values and measured past values of X_m, CO_2 and O_2. Only an initial condition of living biomass is required.

IV. DATA PROCESSING AND TRAINING

A. Data processing

300 data points obtained from simulation of the complete model (eqs. (5)-(12)) were used for training purposes while 300 data points were left for validation. All data were perturbed adding a 5% amplitude Gaussian noise in order to have more realistic conditions.

B. Performance Indices

Two indices were used to quantify the prediction performance of the models. The index of agreement (IA) (Eq. 13)

$$IA = 1 - \frac{\sum_{i=1}^{n} \left(y_i - \hat{y}_i \right)^2}{\sum_{i=1}^{n} \left(\left| y_i' \right| + \left| \hat{y}_i' \right| \right)^2} \qquad \begin{array}{l} with \\ y_i' = y_i - y_m \\ \hat{y}_i' = \hat{y}_i - y_m \end{array} \tag{13}$$

y_m is the mean of the observed value.

y_i is the observed value.

$\hat{y}$ is the simulated value.

and the symmetric mean absolute percentage error (SMAPE) (Eq. 14) [10].

$$SMAPE = \frac{1}{n} \sum_{i=1}^{n} \frac{\left| \hat{y}_i - y_i \right|}{\left(\left| \hat{y}_i \right| + \left| y_i \right| \right) / 2} * 100 \tag{14}$$

$\hat{y}$ is the simulated value.

y_i is the observed value.

C. ANN training

The NNSYSID Toolbox for Matlab [11] was used for training NARX and NARMAX models using the multilayer perceptron (MLP) ANN architecture. As mentioned before, the following variables were considered as exogenous inputs: measured biomass (u1); produced CO_2 (u2); consumed O_2 (u3). Living biomass (y) is the regressive variable. Data were normalized in the range (0 1) and 150 different trainings beginning with random weights were performed in order to avoid local minima. The training process was based on IA maximization for multiple-step-ahead (called model predictive output, MPO) predictions on the validation set. The final architectures for ANN NARX and NARMAX models are summarized in Table II.

To determine the amount of autoregressors a Lipschitz function is applied [11] founding that 2 delays are needed for the output variable y (living biomass (X)). Only one delay

was considered for the exogenous variables and also one delay was chosen for the error for the NARMAX model. Several configurations were tested to adjust the number of neurons in the hidden layer of ANN.

D. SVM training

TABLE II
ANN ARCHITECTURES FOR NARX AND NARMAX MODELS

Model	Number of Neurons (input-hidden-output)	Input Variables	Output Variable
NARX	5-3-1	yt-1, yt-2, u1t-1, u2t-1, u3t-1	yt
NARMAX	6-3-1	yt-1, yt-2, u1t-1, u2t-1, u3t-1, et-1	yt

The library of tools developed at INSA de Rouen, France [12] was used to implement the SVM models. As mentioned above, the same sets of training, validation and testing were used for both ANN and SVM models. As for ANN, the training process was based on IA maximization for MPO predictions on the validation set. The inputs and the output of the SVM model were the same than the NARX model shown in Table II. A Gaussian kernel was used to process the non-linearity of data.

Although the QP optimization algorithm adjusts the weights of the model, some parameters have to be manually adjusted. Here, model training required to adjust C, a parameter that controls the trade-off between the complexity of the model and the experimental error, sigma that regulates the bending of the Gaussian kernel function and ε, the radius of the regression tube enveloping the experimental data [5]. In addition, λ, a parameter of the QP method was also tuned. For NARX, the parameter C was increased by powers of 2 [13]. The exponents took values from -5 to 15 with step 1 and a more detailed search was carried out near the best values of C decreasing the step down to 0.1. The same method was used to find the values of sigma and λ, ranging from -5 to 5 in the first case and from -32 to -7 in the second case. In the first case, the step was decreased the same way than C, from 1 to 0.1, while for λ from 4 to 0.25. Parameter ε varied between 0 and 0.2 with an initial step of 0.05 and finishing with 0.025.

V. RESULTS

The three different developed software sensors (NARX-ANN, NARMAX-ANN, NARX-SVM) were tested when used as MPO predictors of living biomass under three different levels of noise in the data (5%, 10% and 20% amplitude). The MPO predictor for NARX models is described in Eq. 15.

$$\hat{y}_{MPO}(k) = f[\hat{y}(k-1), \hat{y}(k-2), u_1(k-1),.., u_3(k-1)] \quad (15)$$

While the MPO predictor for the NARMAX model is described in Eq. 16:

$$\hat{y}_{MPO}(k) = f[\hat{y}(k-1), \hat{y}(k-2), u_1(k-1),.., u_3(k-1), 0] \quad (16)$$

For MPO prediction the error variable is fixed to 0 because it is impossible to *a priori* know future values of the error [14].

In Table III the quality indices for the different software sensors acting as MPO predictors of living biomass are resumed. The IA should be near 1 to demonstrate similarity between the wave forms and the SMAPE should be low. It can be observed that noise levels influence the performance of the ANN models, however they have limited impact in the SVM model that presents the best indices.

TABLE III
QUALITY INDICES FOR MPO PREDICTIONS OF LIVING BIOMASS (X)

Software Sensor	Noise Level (%)	IA	SMAPE
NARX-ANN	5	0.96	39.56
	10	0.87	87.28
	20	0.75	123.62
NARMAX-ANN	5	0.98	41.20
	10	0.93	52.98
	20	0.81	112.95
NARX-SVM	5	1.00	38.59
	10	0.99	36.57
	20	0.99	36.79

Figures 3 to 5 show the evolution of living biomass as computed by integrating eq. (6) adding different levels of noise and the predictions performed by the three software sensors.

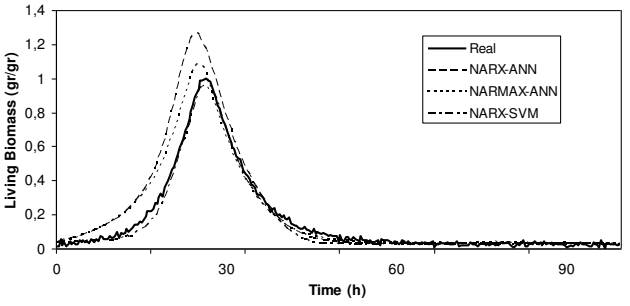

Figure 3. MPO predictions of Living Biomass for a 5% amplitude noise using the 3 different developed software sensors. Solid line corresponds to "real" data obtained by integrating eq. (6) adding the corresponding level of noise.

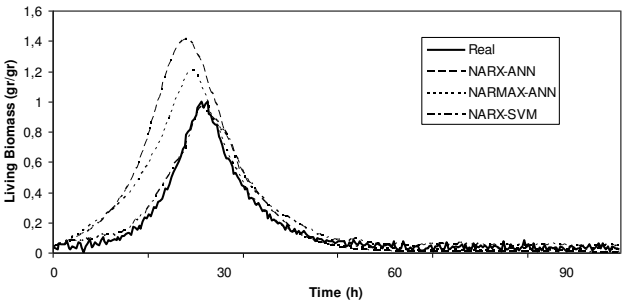

Figure 4. MPO predictions of Living Biomass for a 10 % amplitude noise using the 3 different developed software sensors. Solid line corresponds to "real" data obtained by integrating eq (6) adding the corresponding level of noise.

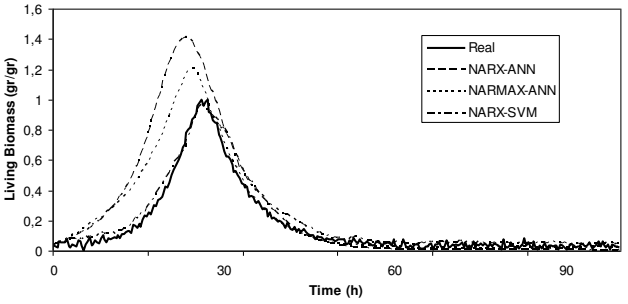

Figure 5. MPO predictions of Living Biomass for a 20 % amplitude noise using the 3 different developed software sensors. Solid line corrsponds to "real" data obtained by integrating eq. (6) adding the corresponding level of noise.

VI. CONCLUSIONS

From the different quality indices and Figures 3 to 5 it can be seen that NARMAX-ANN outperforms NARX-ANN. This is an expected result because of the better predictive capabilities of NARMAX models which include past errors as input information.

SVR-NARX appears as a very good and robust predictive model. Indeed, even with a high level of noise (20%) the Index of Agreement, IA, is very close to 1.0 and SMAPE is quite low (36.79). This is even valuable because a NARX model is trained only to predict one step ahead (OSA) and in this case all models were tested for MPO predictions.

The above mentioned results are coherent with many authors that claim that SVM are better approximators that equivalent black-box models as ANN.

REFERENCES

[1] Shukla, R., Chand, S., and Sristava A.K., 2005, Improvement of gibberellic acid production using a model based fed-batch cultivation of *Gibberella fujikuroi*, Process Biochemistry, 40, pp. 2045-2050.

[2] Box, G. and Jenkins, G., 1970, Time series analysis: Forecasting and control, San Francisco: Holden-Day.

[3] Chai, M-I, S., Song, S., and Li, N-n, 2005, A review of some main improved models for neural network forecasting in time series. In: Proceedings of the IEEE Intelligent Vehicles Symposium, Issue 6-8, pp. 866--868

[4] Woolley, J., Agarwal, P., Baker, J., 2010, Modeling and prediction of chaotic systems with artificial neural networks. Internacional Journal for Numerical Methods in Fluids, 63, pp. 989-1004.

[5] Schölkopf, B., Smola, A., Williamson, R.C., Bartlett, P.L., 2000, New support vector algorithms, Neural Computation, 12, pp. 1083--121

[6] Sapankevych, N and Sankar, R, 2009, Time series prediction using support vector machines: a survey, .IEEE Computational Intelligence Magazine, May 2009, pp. 24-38.

[7] Ljung, L., 1987, System Identification: Theory for the user. Prentice-Hall, Englewood Cliffs, NJ.

[8] Werbos P.J., 1990, Backpropagation Through Time: What It Does and How to do it. Proc. IEEE, Vol 78(10), pp 1150-1560.

[9] Gelmi, C., Perez-Correa, R. and Agosin, E., 2002, Modelling Gibberella fujikuroi growth and GA3 production on solid-state fermentation, Process Biochemistry, 37(9), pp.1033-1040.

[10] Garcia-Pedrero, A., and Gomez-Gil, P.: Time Series Forecasting using Recurrent Neural Networks and Wavelet Reconstructed Signals

[11] Norgaard, M., 2003, The NNSYSID Toolbox, for use with Matlab, http://www.iau.dtu.dk/research/control/nnsysid.html

[12] Canu, S., Grandvalet, Y., Guigue, V., Rakotomamonjy, A, 2005, SVM and Kernel Methods Matlab Toolbox. Perception Systèmes et Information, INSA de Rouen, Rouen, France.

[13] Frohlich, H., Zell, A., 2005, Efficient parameter selection for support vector machines in classification and regression via model-based global optimization. In: Proceedings of the IEEE International Joint Conference on Neural Networks, IJCNN 2005. pp. 1431-1436

[14] Ping-Hei Chen c, Ya-Wei Lee a, Tien-Li Chang,.: Predicting thermal instability in a closed loop pulsating heat pipe system. Applied Thermal Engineering (2008)