Available online at www.sciencedirect.com
ScienceDirect
Simulation Modelling Practice and Theory 15 (2007) 1016–1028
SIMULATION MODELLING PRACTICE AND THEORY
www.elsevier.com/locate/simpat

A greenhouse control with feed-forward and recurrent neural networks

Fathi Fourati, Mohamed Chtourou *

Intelligent Control, Optimization and Design of Complex Systems (ICOS) ENIS, B.P. W, 3038 Sfax, Tunisia

Received 8 July 2006; received in revised form 27 November 2006; accepted 4 June 2007
Available online 13 June 2007

Abstract

Greenhouses are classified as complex systems, so it is difficult to implement classical control methods for this kind of process. In our case we have chosen neural network techniques to drive the internal climate of a greenhouse. An Elman neural network has been used to emulate the direct dynamics of the greenhouse. Based on this model, a multilayer feed-forward neural network has been trained to learn the inverse dynamics of the process to be controlled. The inverse neural network has been placed in cascade with the neural model in order to drive the system outputs to desired values. Simulation results will be given to prove the performance of neural networks in control of the greenhouse.

Keywords: Greenhouse; Neural networks; Neural model; Neural controller

1. Introduction

Greenhouses are considered as complex processes. In fact, they are non linear, multi-input multi-output (MIMO) systems, they present time-varying behaviors and they are subject to pertinent disturbances depending generally on meteorological conditions [1–5]. All these make difficult to describe a greenhouse with analytic models and to control them with classical controllers. In [2] the author has presented a model of a greenhouse using the energy balance. The proposed model is then used to try simulation on the greenhouse climate (temperature and hygrometry) with optimal control in a part of day. In [3] the author has proposed a greenhouse model including the crop transpiration. Then he showed a comparison between optimal and predictive control on the considered greenhouse in a part of day. In [5] the authors have described the application of model predictive control (MPC) for temperature regulation in agricultural processes (a greenhouse). The MPC algorithm used here takes in account the constraints in both manipulated and controlled variables using an on-line linearisation with a very low computational burden. This MPC scheme is compared with an adaptive PID controller. In [17] the authors have proposed an application of fuzzy logic to identify and control of

* Corresponding author. Tel.: +216 74 274 088; fax: +216 74 275 595.
E-mail addresses: fethi.fourati@ipeis.mu.tn (F. Fourati), mohamed.chtourou@enis.rnu.tn (M. Chtourou).

1569-190X/$ - see front matter
doi:10.1016/j.simpat.2007.06.001

multi-dimensional systems. They describe a method to reduce the complexity of a fuzzy controller and they show an application on a real system (a greenhouse). In our case we opt to the use of neural networks to model and to control a greenhouse. A recurrent neural network based on an Elman structure [6,7] is trained to emulate the direct dynamics [7–9] of the greenhouse and used as a greenhouse model and a multilayer feed-forward neural network [10,11] trained to emulate the inverse dynamics of the considered greenhouse is applied as a controller [12–15] to provide the control inputs to the greenhouse.

This paper is organized as follows: in Section 2, we describe the considered greenhouse. In Section 3 we present the architecture of the used Elman neural network to emulate the direct dynamics of the greenhouse and the results of the modeling step. In Section 4, we show, the training structure of feed-forward neural network to emulate the inverse dynamics of the greenhouse then the recall structure used for control following with simulation results and comments. Finally, a conclusion and prospects are given in Section 5.

2. Greenhouse description

The considered greenhouse is a classical one with glasses armatures and defined by, a surface with 40 m^2 and a volume with 120 m^3. It is equipped with sensors allowing measurements of the internal and external climates. The internal climate defined by the internal temperature and the internal hygrometry constitute the greenhouse outputs. In order to control the internal climate the greenhouse is equipped with a set of actuators composed with a heater functioning in the on/off mode with 5 kw power, a sliding shutter with an opening between 0° and 35°, a sprayer and a curtain with a length varying between 0 and 3 m. The external climate composed with the external temperature, the external hygrometry, the global radiant and the wind speed act directly on the functional state of the greenhouse. The external climate parameters are considered as non controllable inputs or disturbances.

The whole functioning system is equivalent to a multi-variable and a non-linear. It can be summarized in the functional bloc diagram given in Fig. 1. Where Te is the external temperature, He is the external humidity, Rg is the global radiant, Vv is the wind speed, Ch is the heating input, Ov is the sliding shutter, Br is the sprayer, Om is the curtain, Ti is the internal temperature and Hi is the internal humidity.

The inside temperature can be defined by [1–3]:

$$Ti(k+1) = \boldsymbol{F}[Ti(k), Hi(k), Te(k), He(k), Rg(k), Vv(k), Ch(k), Ov(k), Br(k), Om(k)]. \tag{1}$$

The inside humidity can be defined by [1–3]:

$$Hi(k+1) = \boldsymbol{G}[Hi(k), Ti(k), Te(k), He(k), Rg(k), Vv(k), Ch(k), Ov(k), Br(k), Om(k)], \tag{2}$$

where F and G are unknown non-linear functions [1–3].

For the considered greenhouse, measurements have been taken through one day. The sampling time has been chosen equal to one minute. This has permit to get a database file with 1440 rows, each row is composed with eight columns which represent, the external temperature $Te(k)$, the internal temperature $Ti(k)$, the external hygrometry $He(k)$, the internal hygrometry $Hi(k)$, the wind speed $Vv(k)$, the global radiant $Rg(k)$, the hea-

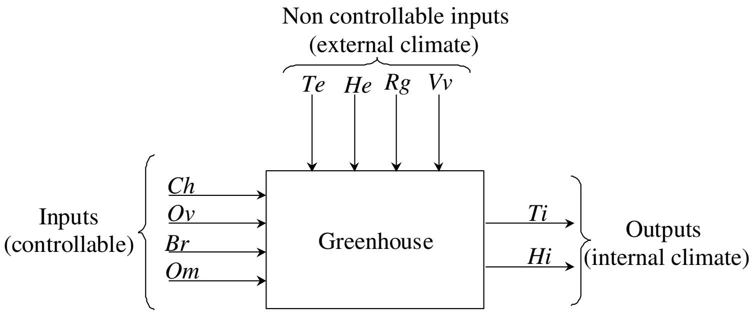

Fig. 1. Greenhouse functional bloc diagram.

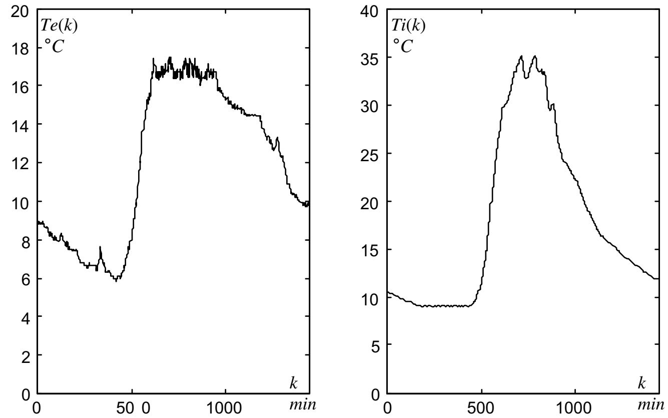

Fig. 2. Evolution of the external and internal temperature.

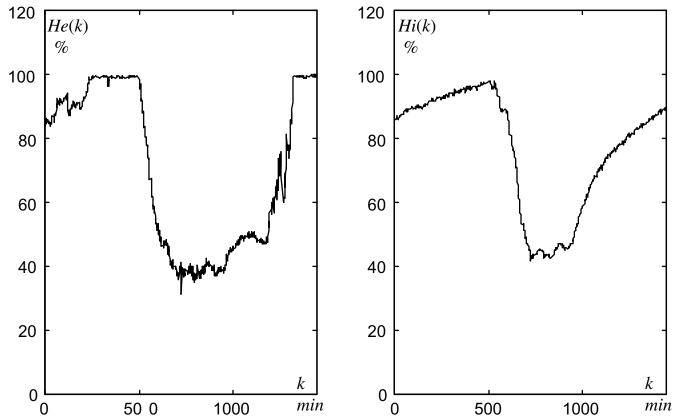

Fig. 3. Evolution of the external and internal hygrometry.

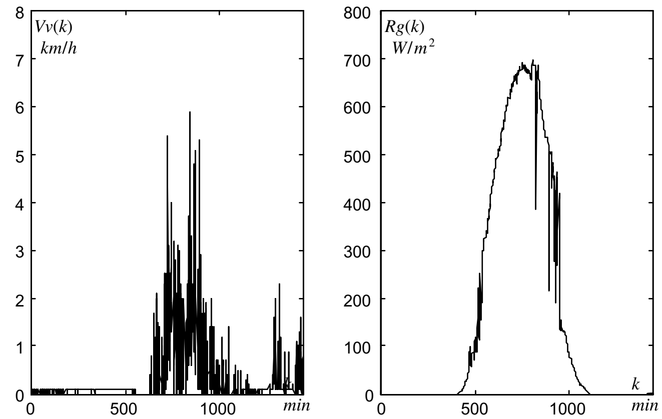

Fig. 4. Evolution of the wind speed and the global radiant.

ter $Ch(k)$, the sprayer $Br(k)$, the sliding shutter $Ov(k)$ and the shade $Om(k)$. Figs. 2–6 show the evolution of these parameters respectively in the considered day. The measurements started at 0 h (0 min) and achieved at 23 h and 59 min (1440 min).

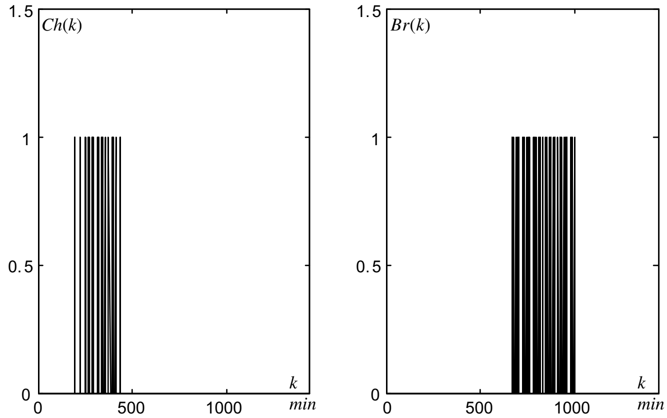

Fig. 5. Evolution of the actuators heater and sprayer.

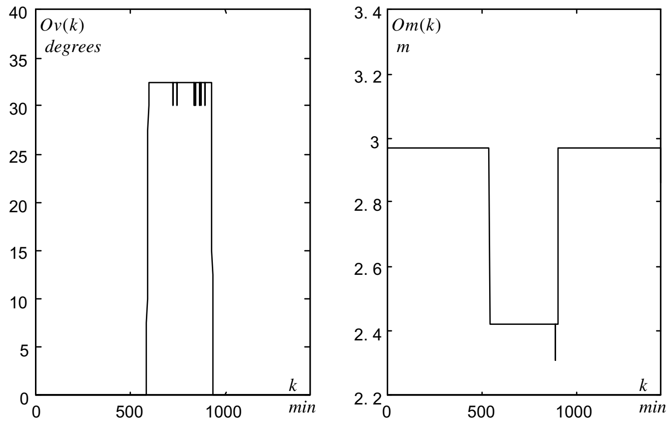

Fig. 6. Evolution of the actuators sliding shutter and curtain.

3. Greenhouse neural modeling

Greenhouses are classified as complex processes. So it is very difficult to obtain kinetic models that represent the whole dynamics of the system. For this we have resort to advanced techniques to resolve such problem. In our case we have chosen a resolution with neural network and precisely an Elman structure [6,7].

3.1. Elman neural network

To emulate the direct dynamics of the greenhouse we have chosen an Elman structure which is a recurrent neural network. In contrast with a feed-forward neural network the recurrent structure presents faster computation due to the smaller number of units in the input layer and a recall structure similar to the training structure [7].

An Elman network structure is composed with an input layer, a hidden layer, a context layer and an output layer. The input and output units interact with the outside environment, while the hidden and context units do not. Fig. 7 shows the Elman neural structure used to model the greenhouse. Where, $U(k)$ is the input vector, $Y^m(k)$ is the output vector, W^{xu} is the weights vector between the hidden and the input layers, W^{yx} is the weights vector between the output and the hidden layers and W^{xc} is the weights vector between the hidden and the context layers. The activation function of the hidden units is the sigmoïdal one [7].

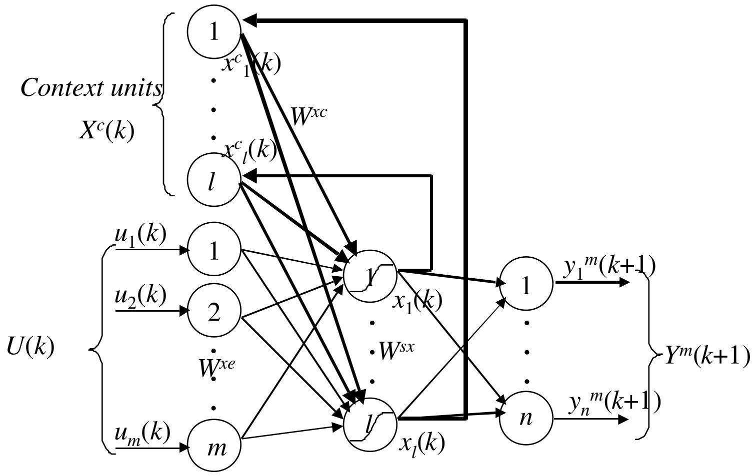

Fig. 7. Elman neural network structure.

3.2. Training structure

The database file of the greenhouse has been divided into two parts alternatively. Each part constitutes a database file with 720 rows, so the effectively sample time now is 2 min. The two files have been used receptively, for training and to validate the neural model.

Fig. 8 shows the procedure of training the considered neural network to emulate the direct dynamics of the greenhouse [7,12,13].

In our case, $U(k) = [Ov(k), Ch(k), Br(k), Om(k), Te(k), He(k), Rg(k), Vv(k)]^T$, $Y^p(k) = [Ti(k), Hi(k)]$, each layer, hidden and context is constituted with four units. The used algorithm to adjust the weights vectors is the backpropagation algorithm minimizing a square error criterion $J_1(k)$ (3) between effective greenhouse outputs and neural network outputs at time k [7,13].

$$J_1(k) = \frac{1}{2}\sum_{i=1}^{n}[y_i^p(k) - y_i^m(k)]^2 = \frac{1}{2}\sum_{i=1}^{n}[\varepsilon_i(k)]^2 \tag{3}$$

Here, $n = 2$

$J_1(k)$ is minimized with a gradient method. The weights vectors are adjusted as following:

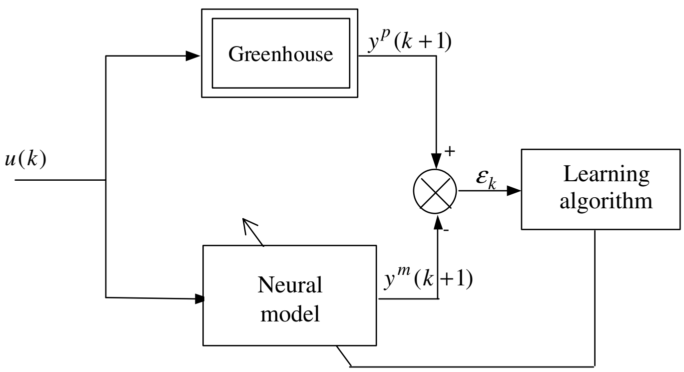

Fig. 8. Learning of direct greenhouse dynamics.

$$W^{yx}(t) = W^{yx}(t-1) - \eta \frac{\partial J_1(k)}{\partial W^{yx}}, \tag{4}$$

$$W^{xu}(t) = W^{xu}(t-1) - \eta \frac{\partial J_1(k)}{\partial W^{xu}}, \tag{5}$$

$$W^{xc}(t) = W^{xc}(t-1) - \eta \frac{\partial J_1(k)}{\partial W^{xc}}, \tag{6}$$

where η is the learning rate, such that: $\eta < 1$ and t is the iteration number.

3.3. Recall structure

After the learning step, the neural model will imitate the greenhouse behavior, it will be exploited to achieve a feed-back control loop, it provides greenhouse outputs from a given input vector. Figs. 9 and 10 show the real greenhouse outputs ($Ti(k)$ and $Hi(k)$) with continued lines and neural model outputs ($Tin(k)$ and $Hin(k)$) with dashed lines.

In order to evaluate the modelling step, we define the average statistic prediction error and the prediction variance, respectively:

$$\bar{m}_{\varepsilon_i} = \frac{\sum_{k=1}^{720} \varepsilon_i(k)}{720}, \tag{7}$$

where $i \in \{1,2\}$

$$\sigma^2_{\varepsilon_i} = \frac{\sum_{k=1}^{720} [\varepsilon_i(k) - \bar{m}_{\varepsilon_i}]^2}{720}. \tag{8}$$

In the case of temperature,

$$\bar{m}_{\varepsilon_1} = -0.09443907999, \tag{9}$$

$$\sigma^2_{\varepsilon_1} = 0.71584255096. \tag{10}$$

In the case of hygrometry,

$$\bar{m}_{\varepsilon_2} = -0.37995060776, \tag{11}$$

$$\sigma^2_{\varepsilon_2} = 4.0926788077. \tag{12}$$

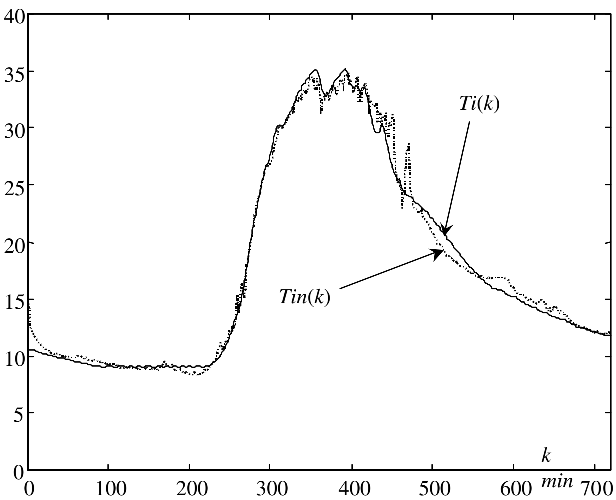

Fig. 9. Real greenhouse temperature and first output of the neural model.

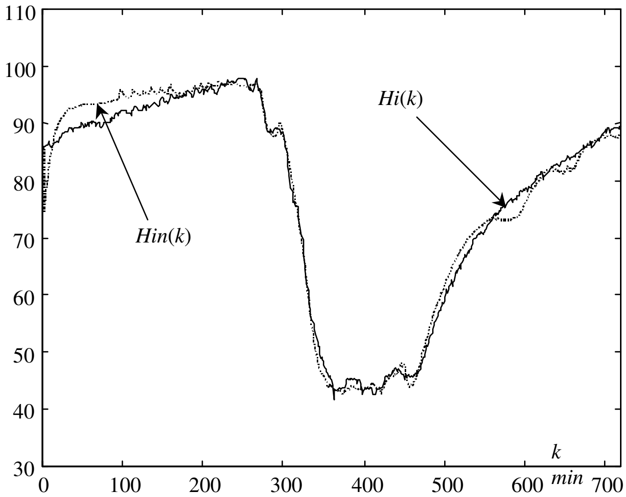

Fig. 10. Real greenhouse hygrometry and second output of the neural model.

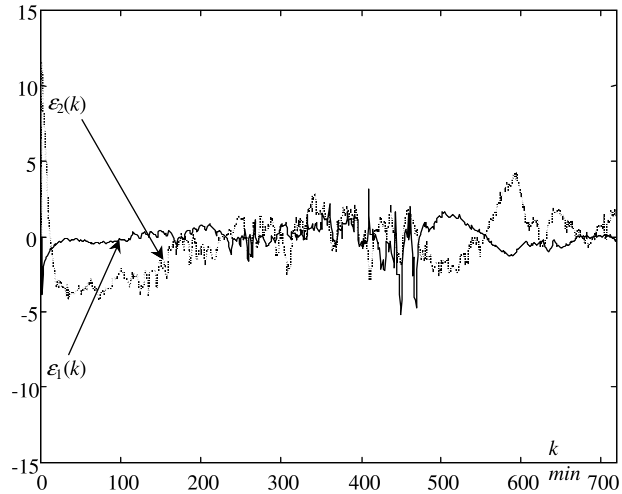

Fig. 11. Prediction errors between real and neural model outputs.

Figs. 11 and 12 show the evolution of the prediction error and the prediction variance through time, respectively.

The model has been tested with the second database file of the greenhouse. Figs. 13 and 14 illustrate the outputs of greenhouse and model.

The above results confirm that adopted neural Elman model of the greenhouse has well emulate the direct dynamics of the greenhouse. It can be used efficiently in a control task.

4. Greenhouse neural control

Now the real greenhouse is replaced by the described Elman neural network model above. To control the greenhouse we need a controller able to take with the complexity of the system. The multilayer feed-forward neural network with an input layer, an output layer and one hidden layer can be used as solution to control such process [7,12,13].

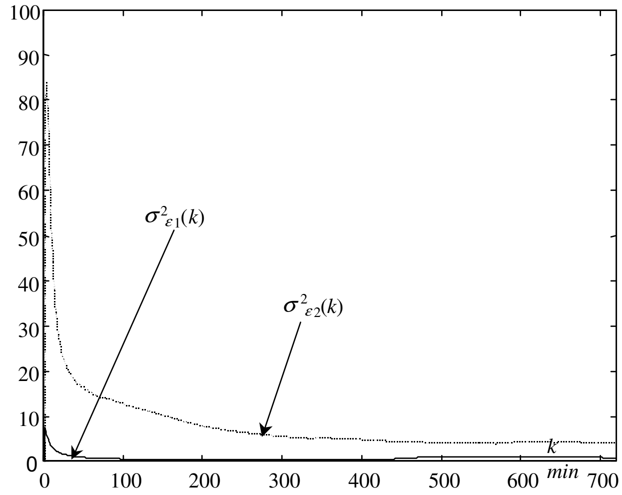

Fig. 12. Evolution of the temperature and the hygrometry prediction variances.

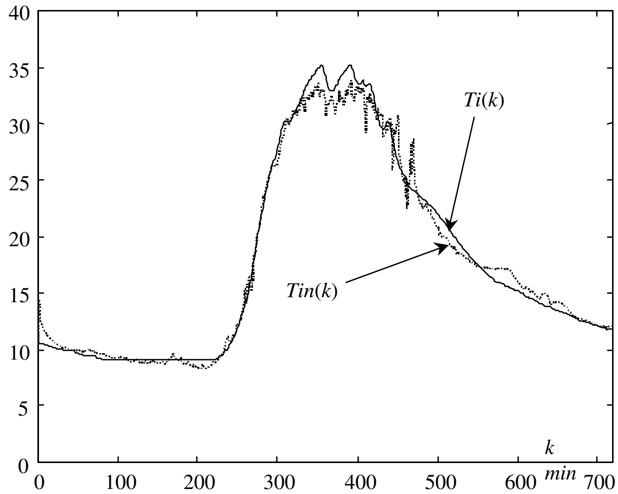

Fig. 13. Real greenhouse temperature and first output of the neural model (generalization test).

4.1. Neural controller structure

Three controllers have played important roles in research on neural control. They are: Albus's cerebellar model articulation (CMAC), Kawato et al.'s hierarchical neural network controller and Psaltis et al.'s multilayered neural network controller [7,12,13]. The last one offers important architecture for control and it is essentially a feed-forward neural network. Fig. 15 shows the architecture of a multilayer feed-forward neural network.

The above neural network is composed with an input layer, one hidden layer, and an output layer. The activation function of the hidden and the output units is the sigmoidal one. Here, the connection weight between a hidden unit j and an input unit i is $w_{j,i}$ and the connection weight between an output unit o and a hidden unit j is $w_{o,j}$.

This kind of neural network is an universal nonlinear function approximator [12,14].

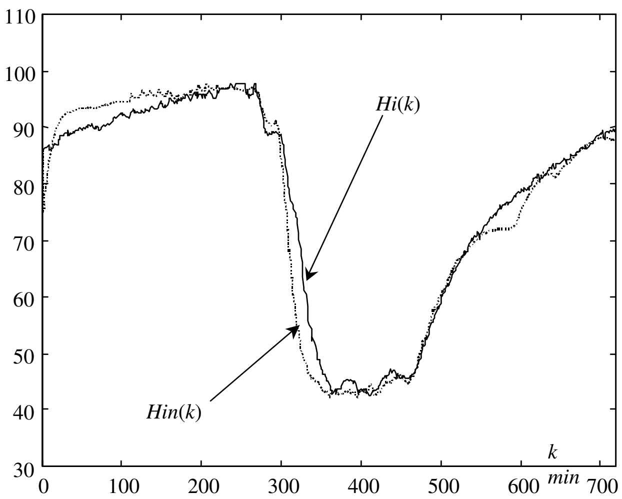

Fig. 14. Real greenhouse hygrometry and second output neural model (generalization test).

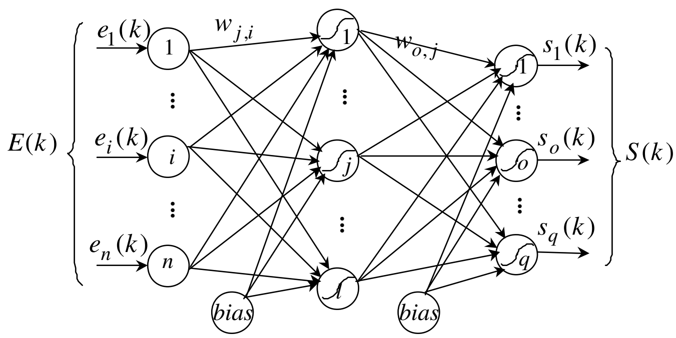

Fig. 15. Architecture of a multilayer feed-forward neural network.

4.2. Training structure

Here the training structure is similar to an off line learning for emulating the inverse dynamics of the plant [7,13]. Fig. 16 shows the training method of the neural controller.

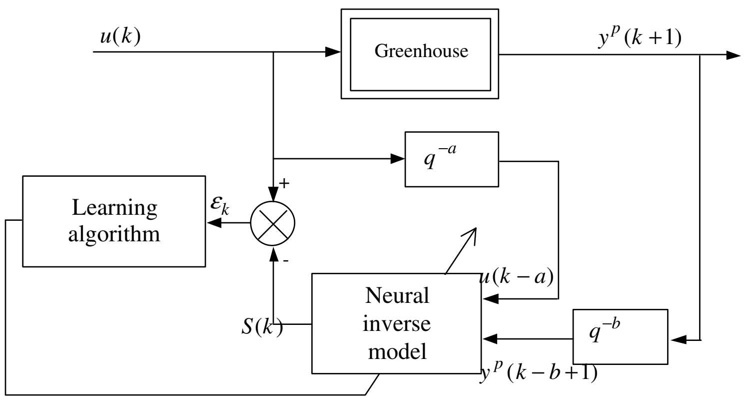

Fig. 16. Training structure of the controller.

The parameters a and b are chosen according to the order of the considered system (greenhouse in our case).

In this architecture, the neural network is trained to minimize the error between the greenhouse input $u(k)$ and the network output $s(k)$.

Here, the input vector of the controller is $E(k) = [Ti(k+1), Hi(k+1), Ti(k), Hi(k), Te(k), He(k), Rg(k), Vv(k)]^{\mathrm{T}}$, the output vector (control actions) is $S(k) = U(k) = [Ov(k), Ch(k), Br(k), Om(k)]^{\mathrm{T}}$.

The used algorithm to adjust the connections weight is the backpropagation algorithm minimizing a square error criterion $J_2(k)$ (13) between neural outputs and effective inputs of the greenhouse at time k.

$$J_2(k) = \frac{1}{2}\sum_{i=1}^{m}[s_i(k) - u_i(k)]^2 \tag{13}$$

Here, $m = 4$

The connections weight $w_{j,i}$ et $w_{o,j}$ are adjusted respectively by Eqs. (14) and (15):

$$w_{j,i}(t) = w_{j,i}(t-1) - \eta\frac{\partial J_2(k)}{\partial w_{j,i}}, \tag{14}$$

$$w_{o,j}(t) = w_{o,j}(t-1) - \eta\frac{\partial J_2(k)}{\partial w_{o,j}}, \tag{15}$$

where η is the learning rate, such that: $\eta < 1$ and t is the iteration number.

The bloc diagram of greenhouse neural controller is shown in Fig. 17.

After the learning step the same neural network will be used to generate control signals, defining a neural controller based on the inverse model of the plant or a feedback sate control [7,12,14,16].

4.3. Control structure

After training, the neural controller is able to provide an appropriate $u(k)$ to the greenhouse if a desired output y^d is defined. The considered neural network is placed in cascade with the greenhouse and as shown in Fig. 18, the whole system constitutes a feedback control with a nonlinear controller [7,12,13,15,16].

In our case, the desired output is $Y^d(k) = [Ti^d(k), Hi^d(k)]^{\mathrm{T}} = [11, 70]^{\mathrm{T}}$, $a = 0$ and $b = 1$.

A temperature of $Ti^d = Ti^d(k) = 11$ °C and an hygrometry $Hi^d = Hi^d(k) = 70\%$ are the references recommended from agriculturists.

The above control strategy has been applied to the greenhouse represented with the Elman neural network model. Fig. 19 shows the outputs of the greenhouse after a control phase.

Comparing to the open loop control (see Figs. 9 and 10) the error between references and greenhouse outputs has been reduced. It is small during the night but it is larger during the day. This is can be explained by the limit of the actuators power, in fact the energy provided is not sufficient to drive the internal climate to the desired one.

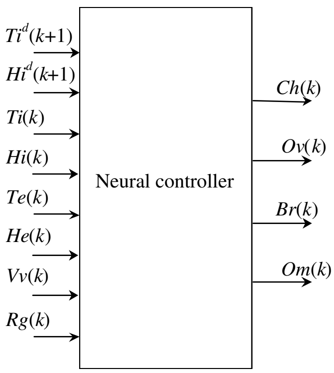

Fig. 17. Controller bloc diagram.

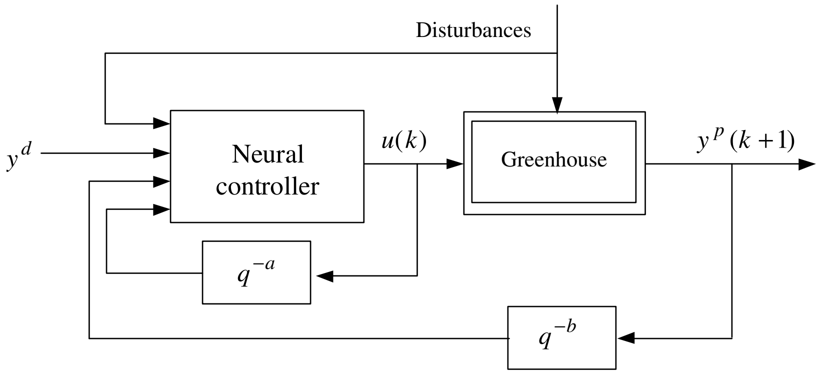

Fig. 18. Neural control strategy.

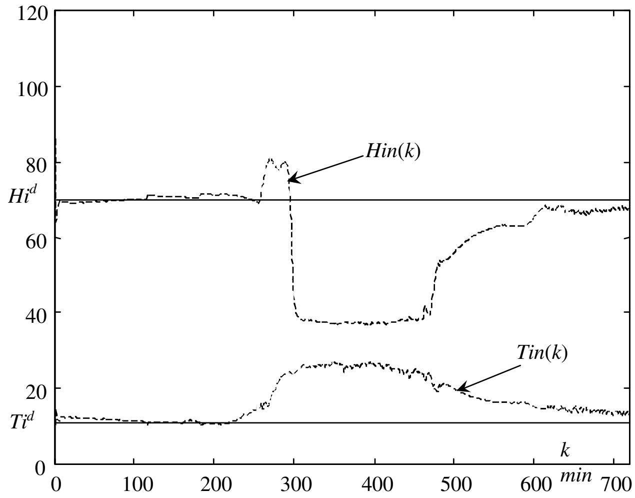

Fig. 19. Desired and greenhouse outputs.

In order to evaluate the control step, we define the following error criterion:

$$Ec = \frac{1}{720}\sum_{k=1}^{720}[(Ti^d(k) - Ti(k))^2 + (Hi^d(k) - Hi(k))^2] \tag{16}$$

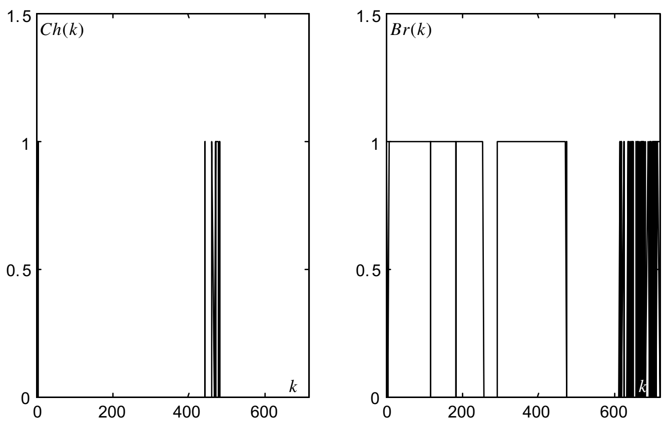

Fig. 20. Evolution of the heater and sprayer actuators.

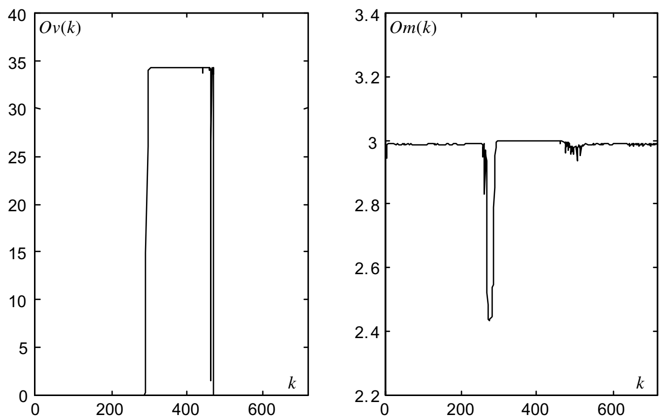

Fig. 21. Evolution of the sliding shutter and curtain actuators.

In the case of the open loop control, $Ec = 533.31$ and with the neural control strategy $Ec = 344.12$.
The error Ec has been reduced when we have applied the neural control strategy.
The evolutions of the actuators during the neural control step are given in Figs. 20 and 21.

5. Conclusion and prospects

In this paper we have used an Elman neural network to emulate the direct dynamics of a greenhouse. The obtained model has been used next in closed loop control using a multilayer feed-forward neural network. This last is trained to emulate the inverse dynamics of the greenhouse and then used as a nonlinear controller with feedback state to provide the control actions for the process. The simulation results show that neural networks strategies give good performances when controlling complex process such greenhouses. The control results can be more improved if the considered greenhouse is equipped with powerful actuators and an adaptive neural controller or a multiple neural control strategy are adopted.

References

[1] V.C. Gaudin, Simulation et commande auto-adaptative d'une serre agricole. Ph.D. Thesis, University of Nantes, 1981.
[2] L. Oueslati, Commande multivariable d'une serre agricole par minimisation d'un critère quadratique. Ph.D. Thesis, University of Toulon, Toulon, 1990.
[3] M. Souissi, Modélisation et commande du climat d'une serre agricole. Ph.D. Thesis, University of Tunis, Tunis, 2002.
[4] F. Fourati, Contribution à la commande neuronale de systèmes dynamiques complexes: Application à une serre agricole. Ph.D. Thesis, ENIS Sfax-Tunisia, 2005.
[5] M.Y. El Ghoumari, H.-J. Tantau, J. Serrano, Non-linear constrained MPC: real-time implementation of greenhouse air temperature control, Computers and Electronics in Agriculture 49 (2005) 345–356.
[6] J.L. Elman, Finding Structure in Time Cognitive Science 14 (1990) 179–211.
[7] D.T. Pham, X. Liu, Neural Networks for Identification, Prediction and Control, Springer, London, 1995.
[8] K.S. Narendra, K. Parthasarathy, Identification and control of dynamical systems using neural networks, IEEE Transactions on Neural Networks 1 (1990) 4–27.
[9] G. Cybenko, Approximation by superpositions of a sigmoïdal function, Mathematics of Control Signal and Systems 2 (1989) 303–314.
[10] J.A. Freeman, D.M. Skapura, Neural Networks: Algorithms, Applications, and Programming Techniques, Addison Wesley, New York, 1991.
[11] G. Dreyfus, J.-M. Martinez, M. Samuelides, M.B. Gordon, S. Thiria, L. Hérault, Réseaux de Neurones Méthodologies et Applications, Editions Eyrolles, Paris, 2004.
[12] D. Psaltis, A. Sideris, A.A. Yamamura, A multilayer neural network controller, IEEE Control Systems Magazine 8 (1988) 17–21.
[13] K.J. Hunt, D. Sbarbaro, R. Zbikowski, P.J. Gawthrop, Neural networks for control systems – a survey, Automatica 28 (1992) 1083–1112.
[14] W. Li, J.J.E. Slotine, Neural network control of unknown nonlinear systems, in: American Control Conference, vol. 2, Pittsburgh, 1989, pp. 1136–1141.

[15] A.U. Levin, K.S. Narendra, Control of nonlinear dynamical systems using neural networks: controllability and stabilization, IEEE Transactions on Neural Networks 4 (1993) 192–206.
[16] J.B.D. Cabera, K.S. Narendra, Issues in the application of neural networks for tracking based on inverse control, IEEE Transactions on Automatic Control 44 (1999) 2007–2027.
[17] F. Lafont, J.F. Balmat, Fuzzy logic to the identification and the command of the multidimensional systems, International Journal of Computational Cognition 2 (2004) 21–47.