IMPLEMENTATION OF IDENTIFICATION ALGORITHM USING RECURRENT RBF NEURAL NETWORKS FOR THE INSIDE TEMPERATURE IN A GREENHOUSE

L. Ezzine[a], M. Guerbaoui[a] ,Y. El Afou[a], A. Ed-dahhak[a], A. Lachhab[a], L. Belkoura[b], B. Bouchikhi[a],

[a]Sensor, Electronic and Instrumentation Team, Faculty of Science
BP: 11201, Zitoune, Meknes, Morocco
[b]LAGIS UMR CNRS, 8146 USTL – Bâtiment P2
59655, Villeneuve d'ascq Cedex, France

[a]latifae@yahoo.com, [a]guerbaoui@gmail.com, [a]elafou.youssef@gmail.com,
[a]a.eddahhak@yahoo.fr,[a]abd_lachhab@yahoo.fr, [b]lotfi.belkoura@univ-lille1.fr, [a] benachir.bouchikhi@gmail.com,

ABSTRACT
The goal of this paper is to carry out a statistical study whose objective is the identification of time series of the greenhouse climatic parameters, in order to optimize the expenditure in cost and time of the culture under greenhouse. In this study, we showed that the inside temperature is the most influential parameters on the greenhouse. However, the automatic climate control requires the development of appropriate control laws that are based on models representing linear and nonlinear system. We are therefore forced to make a study of the system to generate a model that faithfully reproduces the operating parameters of greenhouse climate. In order to achieve the maximum benefit it is important to exploit the available data and an obvious choice here are the machine learning methods such as artificial neural networks. The use of recurrent Radial Basis Function (RBF) models is justified by employing a nonlinear greenhouse system, and hence to give the possibility to identify and to control in the real time the inside temperature in the greenhouse, taking into accounts other climatic parameters within and outside the greenhouse. A comparison of the measured and simulated data proved that the found model can envisage correctly the inside greenhouse temperature.

Keywords: RBF networks, Recurrent RBF Networks, Nonlinear systems, Climatic Parameters, Greenhouse.

1. INTRODUCTION

The main aim of this paper is to provide a methodology of implementation of Radial Basis Function Networks (RBF) for identification of the inside temperature model in a greenhouse. The conventional techniques for this identification are based upon polynomial regression or auto-correlation-based statistical methods such as ARIMA (Ezzine, Lachhab, Eddahhak, and Bouchikhi 2007; Ezzine, Lachhab, Eddahhak, and Bouchikhi 2008; Lachhab, Ezzine, Eddahhak, Didi, Salinas Vazquez, García Lagos, Atencia, Joya, and Bouchikhi 2008). However, the functional relationship between the different climatic parameters is complex and cannot always be captured by these traditional modelling techniques. Often this limits the practical usefulness as it is the case, the use of simplified traditional techniques based on conceptual models, such as regression techniques, may hardly be justified. In order to achieve the maximum benefit, it is important to exploit the available data and an obvious choice here are the machine learning methods such as artificial neural networks.

In the present work, a special type of neural networks that employs RBF network in the hidden layer has been utilized (Dreyfus, Martinez, Samuelides, Gordon, Badran, Thiria, and Herault 2002; Haykin 1994). Recurrent Radial Basis Function (RBF) Networks architecture is used to learn temporal sequences. It is based on the advantages of Radial Basis Function Networks in terms of training process time (Catfolis 1993; Chen, Giles, Sun, Chen, Lee, and Goudreau 1993). The recurrent or dynamic aspect is obtained by cascading looped neurons in the first layer. This layer represents the dynamic memory of the recurrent RBF network that permits to learn temporal data. The proposed network combines the easy use of the RBF network with the dynamic performance of the Locally Recurrent Globally Feed forward network.

In the present model, the variables eventually used included the inside temperature, the outside temperature, the inside humidity, the outside humidity, the CO_2 and the solar radiation, which produced the smallest error in the validation data. At a first stage, all climatic parameters are used, but we noted that the concentration of CO_2 and the solar radiation have little influence compared to other variables (Ezzine, Lachhab, Ed-dahhak, and Bouchikhi 2008). Thus these two parameters are eliminated at the second stage.

The rest of the paper is organised in three sections: a brief survey of system is presented in the second section. The third section describes the Radial Basis Function (RBF) Networks, their training process algorithms and the architecture of recurrent RBF Networks for the time series prediction. Finally, the models and the results obtained for the temperature and the humidity under greenhouse are presented.

2. MATERIAL AND EXPERIMENTAL DEVICE

The greenhouse environment is automated with several actuators and sensors that are connected to an acquisition and control system based on a personal computer (Figure 1). Sensor devices are the basics of climate control because they provide necessary data for optimisation. Productivity, repeatability, output signal and usability by growers are the required characteristics of sensors (Bouchikhi, Eddahhak, El Harzli, and El Bari 2004). The developed system allows the acquisition of measurements of internal and external temperature and hygrometry, and global radiation.

It consists of a heating system, moistening ducts and a static ventilation to control the internal climate. In order to improve the climate and water conduct of a culture under greenhouse, a tool of supervision is associated to the actuator system. After the phase of installation and standardization of sensors inside and outside the greenhouse, a climatic data base of the greenhouse can be generated (Eddahhak, Lachhab, Ezzine, and Bouchikhi 2005; Eddahhak, Lachhab, Ezzine, and Bouchikhi 2007; Lachhab, Eddahhak, Ezzine, and Bouchikhi 2005; Lachhab, Eddahhak, Didi, Ezzine, Salinas, García-Lagos, Atencia, Joya, and Bouchikhi 2007).

The controller is designed to insure:

- The measurement, via some sensors, which monitor climate parameters, especially temperature (LM35), humidity (HIH 4000-001), carbon dioxide (FIGARO AM4 module), lighting (Photo Resistor) and water availability in a desired range;
- The processing of the data and storage of the results;
- The operating of the climatic control devices, usually installed at a modern greenhouse.

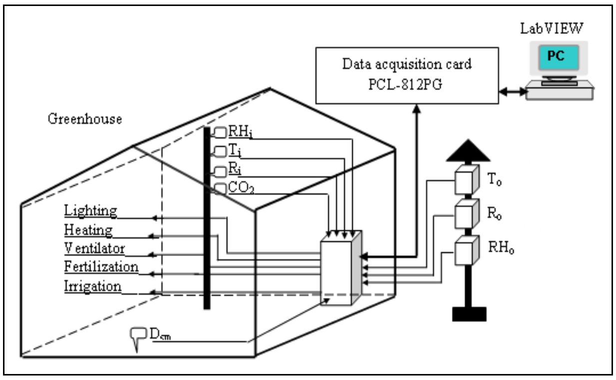

Figure 1: Synoptic diagram of the experimental device

A supervision tool optimises the commands sent to actuators, in order to optimize the climate under the greenhouse. We opted for a graphic program, carried out using LabVIEW (Figure 2), regrouping the following functionalities: acquirement of sensor data outputs, display and treatment of information in real time, commands to actuators, and, in short, creation of a historic based upon a picture database. The driving software permanently compares the magnitudes measured with the reference range in order to start or stop the appropriate actuators. The user has a possibility to fix the threshold of the inside temperature that start up the audible and visual alarms (Eddahhak, Lachhab, Ezzine, and Bouchikhi 2007).

Data were collected in a greenhouse located at the Faculty of Sciences, Meknes, Morocco, between 11 and 18 December 2007. The system is based around a Personal Computer (PC). Sensors and conditioning modules permit to measure the different climatic parameters inside and outside the greenhouse.

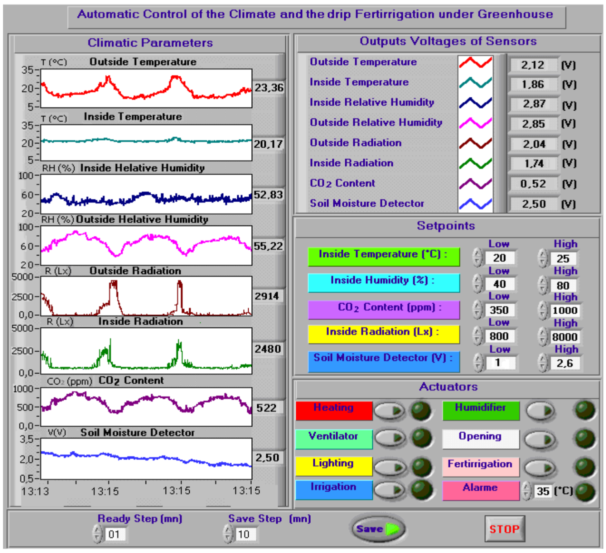

Figure 2: Graphical User Interface based on LabVIEW

3. METHODOLOGY

3.1. Model description

The typical structure of an RBFN consists of three layers, namely the input, hidden and output layers. Each layer consists of a number of neurons (nodes). The nodes in the input layer are used only to pass the input data to hidden layer. No calculations are performed in the input layer nodes, and the connections between the input and the output layer are not weighed (Dreyfus, Martinez, Samuelides, Gordon, Badran, Thiria, and Herault 2002; Haykin 1994).

The hidden layer contains k nodes, which apply a nonlinear transformation on the input variables. More specifically, each node j = 1, 2, …, k has a center value cj, where cj is a vector whose dimension is equal to the number of inputs to the node. For each new input vector x = [x1, x2, …, xN], the Euclidean norm of difference between the input vector and the node center is calculated as follows:

$$v_j = \left\| c_j - x \right\| = \sqrt{\sum_{i=1}^{N} \left(x_i - c_{j,i} \right)^2} \qquad (1)$$

the output of the hidden layer nodes is determined by a nonlinear function, called the ‘activation function’ of the network. A typical selection for the activation function is the Gaussian function:

$$f(\nu) = e^{\left(-\frac{\nu^2}{2\sigma^2}\right)} \tag{2}$$

where the width σ^2 is a variance. Thus, the output of the hidden layer node j is:

$$z_j = f(\nu_j) \tag{3}$$

A set of synaptic weights w_j, j = 1, 2, …, k is applied to the connections between the hidden and the output layer. The nodes in the output layer serve only as summation units, which produce the final output of the network. Considering a one-output network (Figure 3), the overall output will be:

$$\hat{y} = \sum_{j=1}^{k} w_j z_j \tag{4}$$

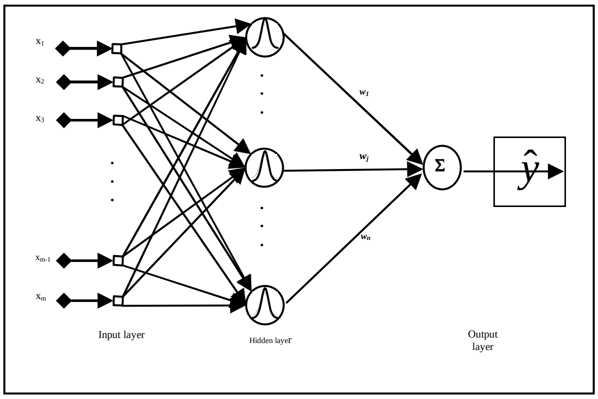

Figure 3: Typical structure of an RBF Networks

3.2. Model training and validation

3.2.1. Training of RBF networks

From a design point of view, the training of RBF Networks is to find the number of hidden layer neurons N and the appropriate parameter set {yk, σ, wk} to map a given input vector to a desired output scalar efficiently with good accuracy and generalization. Many different approaches have been proposed in the literature over recent years for selecting these free parameters and optimizing the complexity of RBF networks. Normally, after the network structure parameter N is determined, the RBF networks are trained by using a two-phase approach, where the centres and width are computed first, and the output weights are calculated in the second phase.

In the phase of selecting the locations of centres, three main strategies have been put forward. The first one is to randomly select a set of samples from training set and the positions of the centres are set according to these samples. This approach can only produce reasonable results when the training data are distributed in a representative manner. The second approach is to perform a pre-clustering on the training set, and the centres of the clusters are used as the centres of the RBF network. Since this clustering is performed without the knowledge of the weights of the output nodes, it is likely that the selection of the centres is sub-optimal with respect to the accuracy of the final result. The selection of the initial values of the centres is also a key problem. The third strategy is to use a gradient descent algorithm to determine the centres. Convergence to a global minimum cannot be guaranteed since the problem is nonlinear with respect to the centres. Therefore, all these approaches have various shortcomings in selecting appropriate centres. Since the practical signals are inevitably disturbed by stochastic noise, training data cannot always represent all samples even if they are acquired from a wide range of amplitude and frequency. Therefore, the pre-clustering is necessary either for the training data, or for the simplification of the networks (Dreyfus, Martinez, Samuelides, Gordon, Badran, Thiria and Herault 2002; Haykin 1994).

When the centres are selected, a uniform width can be heuristically determined by the equation in the following section $\sigma = d/\sqrt{M}$, where d : the maximum distance between the chosen centres and M: is number of centres.

Once the centres and width are fixed, the weights can be known very efficiently, since the computation is reduced to a linear or generalized linear model. There are also some approaches for output layer weights training. One approach, which is called generalized regression neural network (GRNN), assigns the target values as the output layer weights and the output of the network is divided by the sum of the output of a hidden layer neuron. GRNN is effective when a large amount of training data is used and no new training data is far from them. The other approach is to use an iterative training technique such as gradient descent algorithm. In fact, from Equation (4), when the centers and width are determined, weights can be trained by solving the system of linear equations directly. After the final step of calculating the output layer weights is finished, all parameters of the RBF network have been determined.

3.2.2. Evolving RBF networks

The algorithm of gradient descent have been used to train RBF networks, which are called evolving RBF networks. In fact, we specially present the algorithm of training RBF networks.

However, in the learning procedure of an RBF neural network, the determination of the hidden centres and the widths is of particular importance to the improvement of the performance of networks, and the proposed algorithm consists of Gradient descent (Du, Lam, and Zhang 2006; Fahlman and Lebiere 1991; Fahlman 1991; Lee and Sankar 2007, Venkateswarlu and Venkat 2005).

This algorithm is used for both simple models. As for recurring patterns using the same instructions that algorithm, except that it adds the recurrence for the variable in question (the temperature and humidity).

However, for the standardization of data, an uncertainty of 10% was adopted, but the results found were not interesting. In a second time we took incertitude 1%. Standardization of the equation is given as follows:

$$0,99\times\left(\frac{x_i-\min\limits_{1\le i\le N}(x_i)}{\max\limits_{1\le i\le N}(x_i)-\min\limits_{1\le i\le N}(x_i)}\right)+0,01 \qquad (5)$$

As there are no methods that provide a way to correct the centres used, it adopts an estimate with the arithmetic average for all variables, then the weighting coefficients with "random" is that minimize total error of the model. The width of the Gaussian is defined by the maximum distance between the centres divided by the square root of the number of centres. In step (6) for the algorithm, a gradient descent is used to correcting the weight of which to minimize the error.

The proposed Recurrent RBF neural network considers the time as an internal representation (Du, Lam, and Zhang 2006, Lee and Sankar 2007; Venkateswarlu and Venkat 2005). The dynamic aspect is obtained by the use of an additional self-connection on the input neurons. The recurrent RBF network can thus take into account a certain past of the input signal.

Algorithm

Step learning

1. Normalization of the database and Choice of centres
2. Set the width of the Gaussian
   The estimator adopted is:

$$\sigma=\frac{d}{\sqrt{M}}$$

With **M** is the number of centres

$$d=\max_{1\le i,j\le M}\left\|c_i-c_j\right\|$$

3. Initializing weight
4. Creating the matrix of distances.

$$\left[f(\|t_i-c_1\|)\ \ \ldots\ \ f(\|t_i-c_M\|)\right]\begin{bmatrix}w_1\\ \ldots\\ w_M\end{bmatrix}=output$$

5. Calculate error.

Calculate the error measure E during the learning

6. Correcting weight used the method of gradient descent

$$W_{i+1}=W_i-\alpha\times E_i\times\frac{d}{t_i}\left[f\left(\|t_i-c_1\|\right)\cdots f\left(\|t_i-c_M\|\right)\right]$$

With $\boldsymbol{E_i}$ is the error at time i

7. Return to step **(4)** until to finish learning data.

Step test

In this step, the same treatment for learning data is adopted and the calculation of the output used the following equation:

$$\left[f(\|t_i-c_1\|)\ \ \ldots\ \ f(\|t_i-c_M\|)\right]\begin{bmatrix}w_1\\ \ldots\\ w_M\end{bmatrix}=outputtes$$

Calculate the error measure for every example.

4. RESULTS AND DISCUSSIONS

Our work above shows that the inside temperature is the important parameter in the greenhouse and they influence most the culture under greenhouse (Ezzine, Lachhab, Eddahhak and Bouchikhi 2008). So we have chosen these parameters in order to identify the climatic evolution under greenhouse using the RBF networks. In this section we present the results obtained for a model of temperature.

It is important to avoid extrapolations, especially with non-linear, over-parameterized models such as neural networks. Reducing the number of inputs to the bare minimum is: therefore, advisable. This can be done by removing less-relevant and well-correlated inputs; in our case we have eliminated the lighting and CO_2 concentration. When the least important variables cannot be readily identified, statistical reduction (compression) methods, such as regression linear model or bottleneck neural networks may be used to reduce the effective system's dimension. These methods can be applied to the inputs as well as to the state variables.

The model uses a simple RBF neural network consists of four inputs and one output. And in the recurrent against the model consists of five inputs and one output.

From the whole database, we have reserved data from different, but not too far, days to be used as an independent verification set. This prevents over-training and may help to identify outlying records. This technique is superior to, e.g., splitting the weather-dependent data set into odd and even records (or groups of records), since then the two resulting sub-sets are likely to be statistically similar, although they do not share even a single record.

In this section, we will present model temperature. In a first step we use a simple model using the RBF network.

Figure 4 and 5 give the curves of adjustment of the inside temperature according to the found model in a step learning and test. We notice that the desired (measured temperature) and the simulated temperature's evolve in the same direction and the recorded error is understood about between -0,5 °C and 0,5 °C.

The search results are not satisfying the error estimate. To address this problem we introduced the dynamic factor using the RBF recurrent network. The graphic representation for the inside temperature is a single day. This is done in order to have a clearer view on the networks used.

The table 1, gives results of the minimal and maximal of the errors, the average errors and the standard deviations.

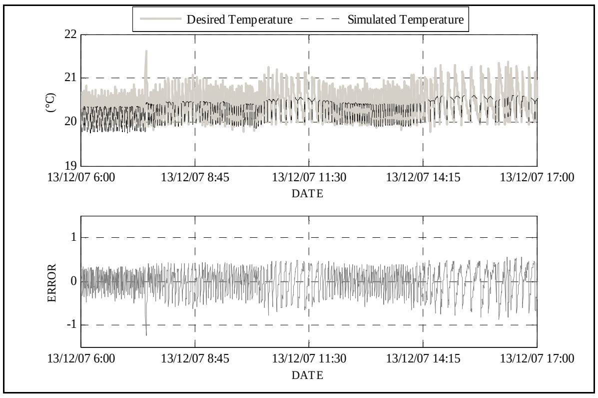

Figure 4: Learning of Simple Model of the temperature

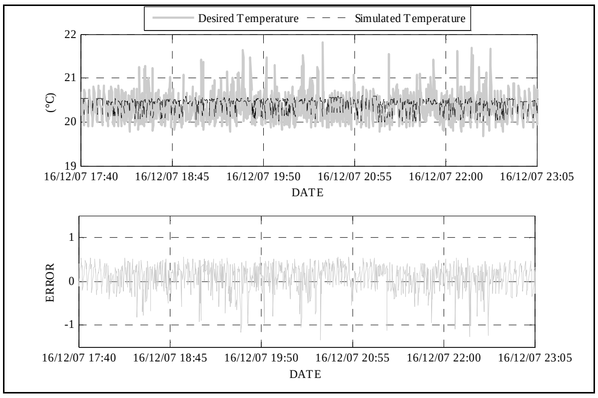

Figure 5: Test of Simple Model of the temperature.

In the test phase, error keeps almost the same pace as that found in the phase of training. Hence our model is validating. Despite this improvement in non-linear modelling of the inside temperature in greenhouse, relative to that linear; the error is always important in the sense dispersion (important variance) (table 1).

Table 1: Results of the RBF and recurrent RBF models for the temperature under greenhouse

		The Simple Model of the temperature (°C)	The Recurrent Model of the temperature (°C)
Learning	Mean error Max error Min error Std error variance	0,23640 1,47651 0,00001 0,29832 0,08900	0,23644 0,56085 1,44364E-05 0,06403 0,00410
Test	Mean error Max error Min error Std error variance	0,28652 1,45759 0,00000 0,31937 0,10200	0,34538 0,40031 0,00015 0,05831 0,00340

So in order to improve the prediction we can go from a simple model to the recurrent model. With the training and testing data established, an evolving recurrent RBF network is used to create a mapping model that simulated the evolution of the temperature and it show the relationship between the inside humidity, outside humidity, outside temperature and command.

Table 1 shows the results of the network training for the temperature under Greenhouse. The recurrent RBF Neuronal Networks gives better results than the simple model RBF with percentage 70,98 %.This model does not take account of the interaction between the inside temperature and its past which gives an error a bit important. So, to remedy this problem, the recurrent RBF networks model is used.

In Figure 6 and 7, the identification of the temperature, where the learning phase data are taken from a period of three days, whereas a one day period is extracted for the test phase. The results illustrated in the figure 6 show the evolution of the experimentally observed temperature inside the greenhouse and the one determined after training the network.

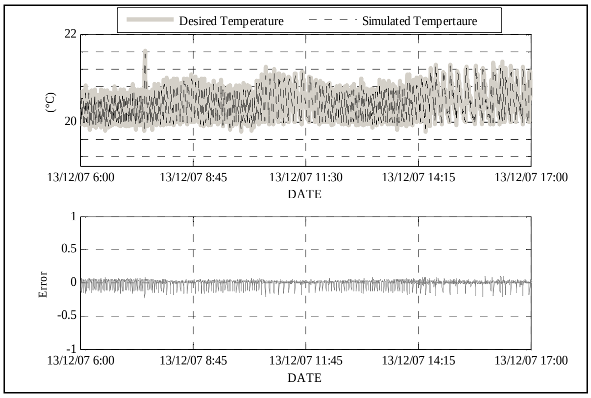

Figure 6: Learning of the Recurrent Model of the temperature.

The sequences are memorized by the Gaussian nodes and the temporal distances between the temperature at time t and their observation at time t-1 by the looped neurons. The recurrent RBF network has the advantage of the local generalization of the Gaussian node while the RBF networks are easier to use, but they have a significant error in comparison with that of recurrent RBF networks.

The test phase is to validate the model developed in phase learning using a new database. The results of this phase are shown in figure 7. A fully connected recurrent RBF network with one hidden layer is selected to be trained using the training data for the forward model. The model is determined to have four input neurons and one output neuron by trial and error. The four inputs to the model include current outside humidity Hot, outside temperature Tot, inside humidity Hit, the command and past inside temperature Tit-1, where t denotes the time variable. The output is the predicted inside temperature. This can be explained that to predict the inside temperature at time t, the inputs of the recurrent RBF network are taken t and the target temperature at time t-1. In order to compare the effectiveness of this approach in training and testing the recurrent RBF network, the Standard Square Error (SSE) between the true outputs and the network predictions are calculated for this approach are shown in Table 1.

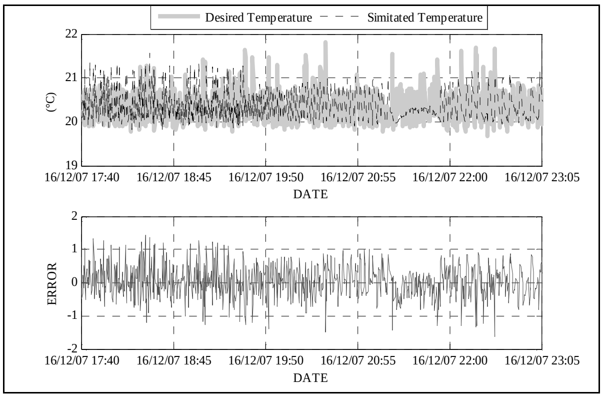

Figure 7: Test of the Recurrent Model of the temperature.

We can notice that the maximum error committed during the Learning is 0,4003 °C and the average error in the east of learning 0,23644 °C. The RRBF can so detect some degradation and gives some solutions to diagnose new failures.

Based on these results, the error is acceptable given the field of variation in temperature, and we can say that the error is like a white noise, namely its average tends to 0. Then we compare the results achieved by the simple model and the recurrent model, a clear improvement of the error is observed.

5. CONCLUSION

This short review covered a few potentially useful neural network applications for greenhouse environmental control. It should be emphasized that the neural network greenhouse modelling refers only to existing structures. These models cannot be used to design new greenhouses, since they lack explicit expressions for the various components and transfer coefficients. Changes in equipment will also require model modification. However, this model could later be fine-tuned to local conditions and requirements, based on data collected on location.

The neural network model can also be useful as controllers, since they may be taught various control rules. Two examples are the simulating of a model-based optimal (feed-forward) controller and a human optimizer (expert grower), who uses some feedback information from the state of the crop.

A comparison of the measured and simulated recursive model proves that the found model can envisage correctly well the inside temperature and humidity in the greenhouse.

ACKNOWLEDGEMENTS

This works was supported by “Volubilis Programm” N° MA/09/211.

REFERENCES

Bouchikhi, B., Eddahhak A., El Harzli M., and El Bari N., 2004. Les capteurs et leur rôle dans la mesure des paramètres climatiques pour une gestion rationnelle de l’eau en agriculture. Journées Internationales des Sciences et Technologies (JICT‘2007), Dec. 2004, Cadiz-Tanger.

Catfolis T., 1993. A method for Improving the Real-Time Recurrent Learning Algorithm. Neural Networks, 6(6): p. 807-821,

Chen D., Giles C. L., Sun G. Z., Chen H. H., Lee Y. C. and Goudreau M. W., 1993. Constructive Learning of Recurrent Neural Networks. Proc. IEEE ICNN, vol. III: p. 1196-1201, San Francisco,

Dreyfus G., Martinez J.-M., Samuelides M., Gordon M. B., Badran F., Thiria S. and Herault L.. 2002. Reseaux de neurones: Methodologie et applications. Ed. Eyrolles, Paris, ISBN: 2-212-11019-7.

Du H., Lam J. and Zhang N., 2006. Modelling of a magneto-rheological damper by evolving radial basis function networks. Engineering Applications of Artificial Intelligence, 19 p. 869–881.

Eddahhak A., Lachhab A., Ezzine L. and Bouchikhi B., 2005. Performance Evaluation of Low-Cost sensors and Instrumentation Conditioning Modules for the Control and Measuring Greenhouse Climate. International Conference on Modeling and Simulation, General Applications and Models in Engineering Science, 22-24 November 2005, Marrakech, Morocco.

Eddahhak A., Lachhab A., Ezzine L. and Bouchikhi B., 2007. Performance Evaluation of a Developing Greenhouse Climate Control with a Computer System. AMSE journal, Association for the Advancement of Modelling and Simulation Techniques in Enterprises, C, 68 (1) p. 53-64.

Ezzine L., Lachhab A., Eddahhak A., and Bouchikhi B., 2007. Statistical modelling of the inside temperature in greenhouse. Information and Communication Technologies International Symposium (ICTIS’07), p 540-543, 3-5 April, Fes, Morocco.

Ezzine L., Lachhab A., Eddahhak A.and Bouchikhi B., 2008. Identification of a Model of Regression of the Climatic Parameters under Greenhouse. Physical & Chemical news, Phys. Chem. News 39, p 43-51.

Fahlman S. E., Lebiere C., 1991. The Cascade-Correlation Learning Architecture. Rapport Technique, CMU-CS-90-100, août 1991.

Fahlman S. E., 1991. The Recurrent Cascade-Correlation Architecture. Rapport Technique, CMU-CS-91-100, mai 1991.

Haykin S., 1994. Neural Networks: A comprehensive foundation. Macmillan College Publishing Company, Inc., United States of America.

Lachhab A., Eddahhak A., Ezzine L. and Bouchikhi B., 2005. Strategy of Developing a Greenhouse Climate Control With a Computer System. International Conference on Modeling and Simulation , General Applications and Models in Engineering

Science,22-24 November 2005, Marrakech, Morocco.

Lachhab A., Eddahhak A., Didi S., Ezzine L., Salinas J.R., García-Lagos F., Atencia M., Joya G. and B. Bouchikhi, 2007. Mise au point d’un systeme de controle et de supervision du climat et de la fertirrigation goutte-a-goutte sous serre. Journées Internationales des Sciences et Technologies (JICT‘2007).

Lachhab A., Ezzine L., Eddahhak A., Didi S., Salinas Vazquez J. R., García Lagos F., Atencia M., Joya Caparros G. and Bouchikhi B., 2008. Identification of greenhouse dynamic temperature model using statistical modeling. AMSE Journal, Association for the Advancement of Modelling and Simulation Techniques in Enterprises. 29 (2) p 38-48.

Lee J., Sankar R., 2007. Theoretical derivation of minimum mean square error of RBF based equalizer. Signal Processing, 87 p. 1613–1625.

Venkateswarlu Ch., Venkat Rao K., 2005. Dynamic recurrent radial basis function network model predictive control of unstable nonlinear processes. Chemical Engineering Science, 60, p. 6718 – 6732.