Neural Models for an Intelligent Greenhouse – A CMAC Global Greenhouse Model

P. Eredics*, K. Gáti*, T.P. Dobrowiecki* and G. Horváth*

* Budapest University of Technology and Economics, Department of Measurement and Information Systems, Budapest, Hungary

eredics@mit.bme.hu, gatikr@mit.bme.hu, dobrowiecki@mit.bme.hu, horvath@mit.bme.hu

***Abstract*—The effectiveness of greenhouse control can be raised by the application of model based intelligent control. The large complexity of the thermal system in the greenhouse calls for a decomposed model. The key element of such model decomposition is the global greenhouse model creating predictions for all important parts of the greenhouse. A Cerebellar Model Articulation Controller (CMAC) model was built and trained to learn the thermal dynamics in the greenhouse and to provide forecasts for the future state of the house for 20 minutes ahead. The CMAC model was tested with 2 actuator configuration settings of the greenhouse, representing more than 57% of the operating time of the house.**

I. Introduction

Greenhouses are widely used artificial controlled environments for vegetable and ornamental plant production. The primary mechanism is the solar radiation passing through the transparent walls. It means a cheap and efficient way to keep a warmer temperature inside the greenhouse compared to the external environment.

As the solar radiation cyclically changes and is influenced by local weather conditions, actuators may be needed to keep internal conditions of the house under control. Windows and shading screens can help cooling down or keeping cool the house in the hot season. Controlled misting and irrigation serves the water needs of the plants. Several different heating appliances can be used to help keeping the house warm in the cold season or during longer periods without notable energy input from solar radiation.

Greenhouses come in various sizes and with different levels of automation. Large industrial vegetable growing greenhouses can span area of several thousand m^2, and may hold only the most important actuators to keep the inner conditions acceptable. Small scale recreational, scientific or special production greenhouses can be under 100 m^2, and may be very well equipped, to ensure the special needs of plants housed.

Greenhouses may be constructed for very different purposes, but the actuators are controlled in a uniformly similar way. Traditional greenhouse control solutions are available off the shelf and are widely used in production. Such traditional controllers refer to temperature levels to trigger control operations: windows open when the temperature increases over the level selected by the greenhouse operator. Windows close when temperature drops below the same level (usually with some hysteresis). Such temperature levels form the basis of all traditional control solutions available today on the market.

Traditional control solutions have the advantage of low computational complexity and easy to understand working logic. Unfortunately these methods suffer also from major drawbacks.

The first problem is the fact that controlling temperature levels have to be set manually by the greenhouse operator. Finding the optimal settings for each actuator is hard even for an expert operator, especially when the control is to be set up in a newly built greenhouse with less known micro climate and thermal behavior. What is more, these settings have to be tuned over and over again as the seasons or as the utilization (living charge) of the house changes.

The second problem is the reactive nature of the control. Actions to avoid adverse circumstances are taken only when the user selected temperature levels are exceeded. This means that the control is always a bit late and is never able to avoid suboptimal situations in advance. Nevertheless avoiding suboptimal situations before they occur would result in most cases in the lower operation costs for the owner.

The last problem of traditional control is the separate (and thus independent) control logic and settings for all actuators. E.g. the operation of the windows is not related to the movement of the shading screen or switching the heating. This lack of synchronization means no problem with carefully chosen settings, but may introduce serious oscillations when operating temperatures are set with less foresight, or in case of rare weather conditions.

The concept of intelligent greenhouse is aimed to eliminate the drawbacks of traditional solutions as detailed in the next section.

II. Intelligent Control

The intelligent greenhouse is a paradigm approaching the greenhouse control problem from the perspective of artificial intelligence. Here machine learning is used to create an adaptive solution to all problems outlined in the previous section.

To eliminate control levels set by the operator the intelligent greenhouse control introduces goals for the control. Goals can be formulated as target parameter zones for the physical state variables of the greenhouse, e.g. target temperature and humidity range suitable for the plants to grow. Multiple different goals can be specified and the conformity of the greenhouse operation to these goals can be defined as the performance of the control. This way the transition from goals to actuator commands is left to the control system and is done automatically without assuming that the operator must understand well the thermal dynamics of the greenhouse.

The reactive nature of traditional control can be overcome by predictive modeling [1]. Computing and maintaining updated the predictive models of the greenhouse makes it possible to forecast suboptimal situations within the greenhouse well before these situations evolve causing harm to the plants. This way the intelligent control will be able to take the necessary steps to keep the internal conditions acceptable in advance. This paper focuses on problems related to the greenhouse modeling, as detailed in the next section.

Having predictive models of the greenhouse makes it possible even to overtake the third limitation of traditional control, namely the missing synchronization between the actuators. AI planning can be used to generate a large number of potential plans common for all actuators. This way the synchronization is naturally ensured by the method applied to plan building. These potential plans can be evaluated by their conformance to the control goals, and the first step of the best plan is how the control starts in the greenhouse.

III. Greenhouse Modeling

The traditional way of modeling a thermal system (such as a greenhouse) is to build and identify a physical model based on the underlying physical heat transfer equations. This method is applicable in a research environment where the large number of measurements can be collected from the greenhouse but such an approach is unfeasible in an industrial environment. The active production going on in the greenhouse introduces constant changes in the thermal model because of the growing plants or the variable interior lay-out.

Because of these a more flexible modeling method, adaptive modeling with black-box neural models is required. To ensure realistic conditions for the modeling a 100 m^2 experimental greenhouse was equipped with a measurement and traditional control system to collect real world measurement in every 5 minutes [2]. The measurement system collected temperature data from all 18 desks holding the plants, and from several strategically selected points within the greenhouse. Local solar radiation measurements and on line regional forecasts were also recorded along with the temperature measurements.

This way a large number of data was collected to form a solid basis for the experiments with intelligent greenhouse control and with the greenhouse modeling. A notable advantage of the instrumentation is the high spatial resolution of the data collected making it possible to define detailed goals for the intelligent control. Sensors installed in the greenhouse were classified into zones, based on the thermally quasi-homogeneous regions of the house, see Fig. 1. Each zone holds one or more sensors collecting thermal data. In some selected zones (Zone-2 under the shading screen and Zone-4 outside the house) solar radiation sensors are also installed.

Unfortunately this high resolution makes it impossible to create a monolith model of the whole house and calls for model decomposition to lower the model complexity by utilizing the structure of the underlying physical system [3]. Fig. 2 shows the proposed decomposition into 6 modules.

Module-A is used to create accurate temperature forecasts for the micro climate in the close proximity of the greenhouse. This module is realized with time series mining methods, detailed in [4]. Module-B is used to repair missing or noisy forecasts recorded from the Internet. Different data cleaning methods are incorporated into this module to produce reliable forecasts even in permanent absence of the network link [5]. Module-C models a small but highly independent part of the thermal system, namely the heating pipe. Two neural networks are trained to predict the warming and heating pipe temperatures separately [6]. Module-D is the main focus of this paper. A CMAC-based model is detailed in the next section, responsible for the prediction of the future temperatures of all internal zones shown in Fig. 1. In the future Module-E and Module-F will be used to approximate each temperature on the desks, as desk temperatures are the most important parameters for the plants.

IV. The Global Greenhouse Model

The global greenhouse model is the key element of the decomposed model. This module transforms measurements from the actual state of the house into the predictions of future temperatures of the thermal zones. These predictions can be used later to generate predictions of high spatial resolution by extending the results to all desk sensors.

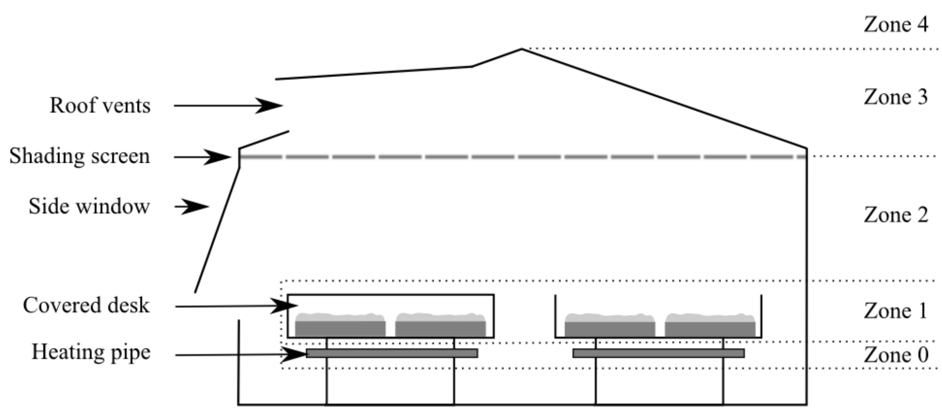

Figure 1. Thermal zones of the experimental greenhouse

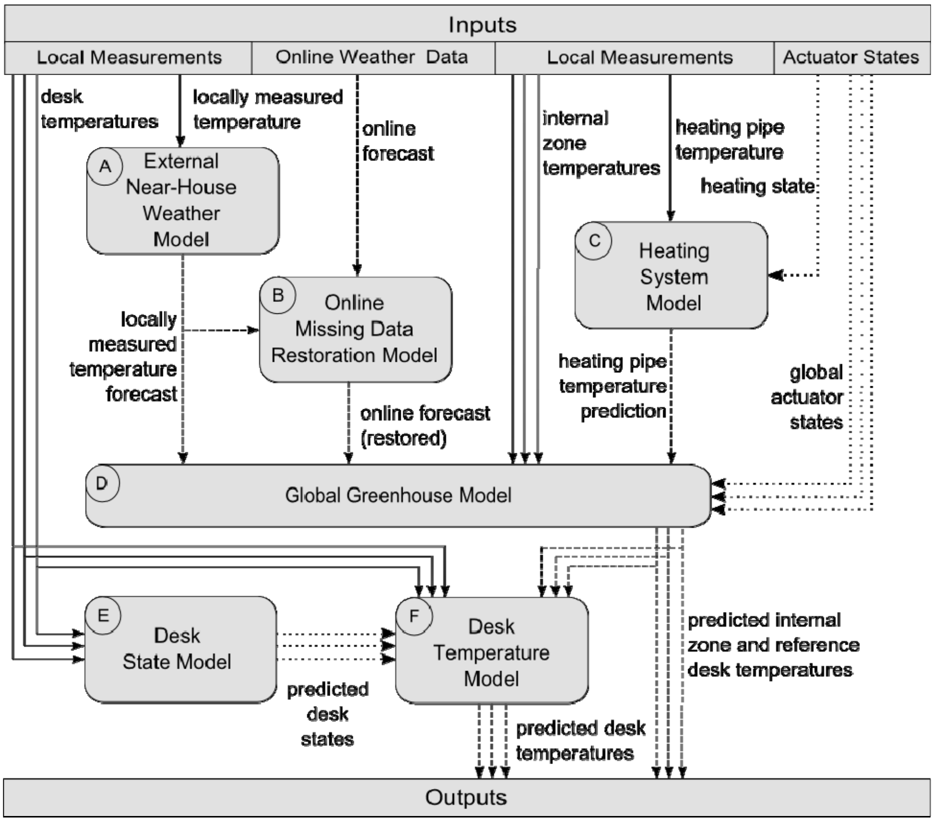

Figure 2. The decomposition of the greenhouse model into 6 modules related to the thermal substructures of the greenhouse (solid lines: measurements; dotted lines: states; dashed lines: predictions)

A. *Actuator configurations of the greenhouse*

Unfortunately the global greenhouse model is largely influenced by the configuration of the greenhouse actuators. For example the deactivation of the shading screen fuses Zone-2 and Zone-3 together resulting in a totally different thermal structure than earlier. Therefore the actuator configuration is very important in the predictive modeling. To free the model from learning such structural differences (and lowering this way the modeling complexity), separate models are trained for all important actuator configurations. The four most important actuators of the experimental greenhouse are collected in Table I. The total number of combinations of possible actuator states is in theory 36, but only 19 configurations are used in practice as all windows must be closed for energy conservation considerations while the heating is activated. In our experiments the two most common states of the greenhouse will be inspected. The common feature for both states is that all windows are closed and the heating is turned off. In State-03 the shading screen is set to 0% cover, while in State-15 the shading screen is set to 100% cover. So far about 330 thousand measurements where collected. State-03 represents 16% of all measurement time, while State-15 alone holds 41% percent of all records. This way these two selected states represent more than half of the operating time of the greenhouse.

TABLE I.
IMPORTANT ACTUATORS OF THE EXPERIMENTAL GREENHOUSE

Description	Possible states
Upper windows	0 = closed 1 = half-opened 2 = opened
Upper shading	0 = no cover 1 = 90% cover 2 = full cover
Side windows	0 = closed 1 = opened
Heating	0 = off 1 = on

B. *Description of the data*

Every record of the time series consist of 5 different measurements each with a 0.5°C granularity. There is one measurement from every Zone, from Zone-0 the temperature of the heating pipe, from Zone-1 the temperature of the covered desk, from Zone-2 the internal temperature under the shading screen, from Zone-3 the internal temperature over the shading screen and from Zone-4 we have the external temperature. This 5 data represent the thermal conditions of the green house at a given moment. This 5 value has to be predicted for the next 5-20 minutes.

We do not have other measurements because of the assumption that we have every required measurement for the forecasting. So the model to be built is a 5 input - 5 output MIMO model.

C. *Introducing CMAC*

The Cerebellar Model Articulation Controller is a neural network based on the simplified model of the cerebellum. It was originally proposed by Albus in the mid seventies [7]. The main features of the CMAC are the fast learning and the local approximation capability. These features made it a considerable choice in our experiments. Besides these attractive features, CMAC has a serious drawback: its memory complexity may be very large.

The CMAC has two layers [8]. The first layer maps from the input space to the feature space. This layer is fixed. The generated vector called association vector. The association vector is a binary sparse vector, and the value of 1 at a given index means, that the input is in the receptive field of the corresponding basis function. The receptive field is the area in the input space, where the basis function has a value of 1, otherwise this value will be 0. A special feature of CMAC is that the receptive field of these basis functions are finite. This cause a locally constant output, which is the reason why CMAC works with digitized inputs. The number of activated basis functions is the most important property of the network as this defines the first fixed layer. This is denoted with C, and this specifies the width of the basis function as well.

The second layer is a weighted sum of the association vector, which is the output of the network, see (1).

$$y = \mathbf{a}(\mathbf{x})^T \mathbf{w} = \sum_{i:a_i=1} w_i \qquad (1)$$

Here $\mathbf{a}(\mathbf{x})$ is the association vector corresponding to the input $\mathbf{x}$ and $\mathbf{w}$ is the weight vector. Because the association vector is binary, the product equals with the sum of weights corresponding to the activated basis functions, where the association vector has a value of 1.

The training of the CMAC may be done with the simple LMS rule, see (2).

$$\Delta w_i = \mu(y_d - y), i : a_i = 1 \qquad (2)$$

where the desired output is denoted with y_d, and μ is the learning rate.

D. Variants of the CMAC

There are two main types of extensions for the CMAC. One way is to increase the performance of the network. These extensions are the higher-order CMACs[9] and the weight-smoothing regularization[8]. The previous one proposes new basis functions instead of the binary functions [9], for example Gaussian or B-Spline functions. This means that the association vector is not binary anymore, however the finite receptive field of the basis functions remains.

The latter one gives additional constraints to the criteria function, and so a new training equation is obtained, which forces the network to learn a smooth function.

Due to the curse of dimensionality a classical CMAC cannot be implemented if a high dimensional problem has to be solved. In these cases special variants of the CMAC has to be used. Albus proposed hash-coding for this [7]. Some other types are the SOP-CMAC [11], Kernel CMAC [8][12], Fuzzy CMAC [13].

The SOP-CMAC stands for Sum-of-product CMAC. This is a special architecture where an, originally high dimensional, problem is divided into smaller dimensional parts. The output of the network is the sum of the submodules, and the output of a given submodule is the product of the CMAC networks of the submodule. These CMACs are small, so they can be implemented with

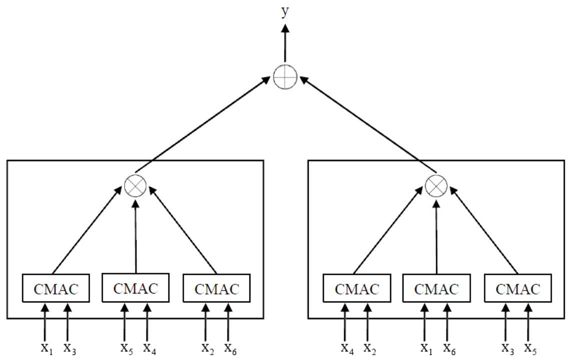

Figure 3. Schematic view of SOP-CMAC

classical CMACs. A schematic architecture can be seen on Fig. 3.

Kernel CMAC applies the kernel trick for the CMAC network [8][12]. The Kernel CMAC has the same learning capability as the classical CMAC however the complexity of Kernel CMAC is upper constrained by the number of training points. The output is calculated according to (3).

$$y = \sum_{i=1}^{P} \alpha_i \cdot K(\mathbf{x}, \mathbf{x}(k)) \qquad (3)$$

Note that this is also a weighted sum as in classical case, however the upper index of the sum is only *P*, where *P* is the number of training samples. It is not necessary to use all the training samples, but leaving some of them out from the weighted sum, will result in a different result as a classical CMAC.

Here α_i is the *i*-th element of the weight vector, and $K(\mathbf{x}, \mathbf{x}(k)) = \mathbf{a}(\mathbf{x})^T \mathbf{a}(\mathbf{x}(k))$ is the kernel function defined as the scalar product of two association vector.

The Fuzzy CMAC [13] is a very similar extension as the Kernel CMAC, however its theoretical and mathematical foundations are completely different. The only difference in the mathematical computation of the output between Fuzzy and Kernel CMAC is that Kernel CMAC uses a weighted sum while Fuzzy CMAC uses a weighted average as it can be seen in (4).

$$y = \frac{\sum_{j=1}^{K} w_j \cdot \phi_j(\mathbf{x})}{\sum_{j=1}^{K} \phi_j(\mathbf{x})} \qquad (4)$$

Here $\phi_j(\mathbf{x})$ is the value of the *j*-th basis function at the given input. From the point of view of a fuzzy system, we have rules, and we want to inference the output from the input with the known rules. In the case of a Fuzzy CMAC we may interpret the basis functions as rules. The placement of a given basis functions determines the value of the rule IF or condition part, and the value of the basis functions at the input is the firing strength of this particular rule. The calculation of the output, may be interpreted as a Center of Gravity (COG), defuzzyfication method, see (4).

One of the most significant problems of Fuzzy CMAC is the selection of the basis functions, rules. One way to handle this problem is to place a basis function on every training sample. This is not always a good choice, especially when large amount of training samples are available, because this will increase the time needed for the training, and increases the number of free parameters and so the chance of overfitting.

An additional constraint for the training sample selection is that we do not know the samples prior to the training so we cannot determine the number of the basis functions and their placement. A considerable option for the specified requirements is proposed in [13]. This selection method is based on the idea that we do not need a new basis function where we have already a lot. So after a given input, we have to check the maximal value of the basis functions, i.e. the maximal firing strength of the rules. If this value is below a prespecified threshold we place a new basis function centered at the given input. With this an easily tunable training selection method may be used for sorting the training samples. Based on the value of threshold we make a decision between complexity and performance, as the lower the threshold the less training samples we have, but the performance will be decreased as well. A good threshold value may be found with trial and error.

E. Test configurations with the CMAC

Due to the size of the input (5 dimensions and about ~130000 sample) the classical CMAC cannot be applied as it would require about $\sim 8 \cdot 10^{11}$ weights. From the previously described extensions we have chosen the Fuzzy CMAC.

As it was discussed earlier the CMAC works with digitized inputs. As it was mentioned earlier the recorded values of the greenhouse are at a 0.5°C resolution, so they can be easily transformed in a well suited form. The range of possible inputs were from 0 to 200 for each dimension. The selected basis functions were 4-th order B-Splines in all cases.

With the assumption we have, at a given time, every needed information to predict for the next measurement time, we did not need to use a recurrent neural network, a static CMAC was enough for our purposes. We wanted to predict results not just for the next 5 minutes but for the next 10, 15 and 20 minutes as well. So we used our prediction for the next 5 minutes and used it to predict for the next 10 minutes, and so on. Basically a long-term prediction method is applied, where the network uses its previous outputs as its input. This is a feedback, however this only appears in the recall phase, and the training is just a function approximation problem.

V. RESULTS

The CMAC were trained for 30 epochs, with a learning rate of 0.8. The 130000 samples were split. 80000 were used for training, and the remaining 50000 for testing. The result of the training can be seen, on Fig. 4. Here the threshold for the training point selection method was 0.4 (this value can be between 0 and 1.0). The low value of

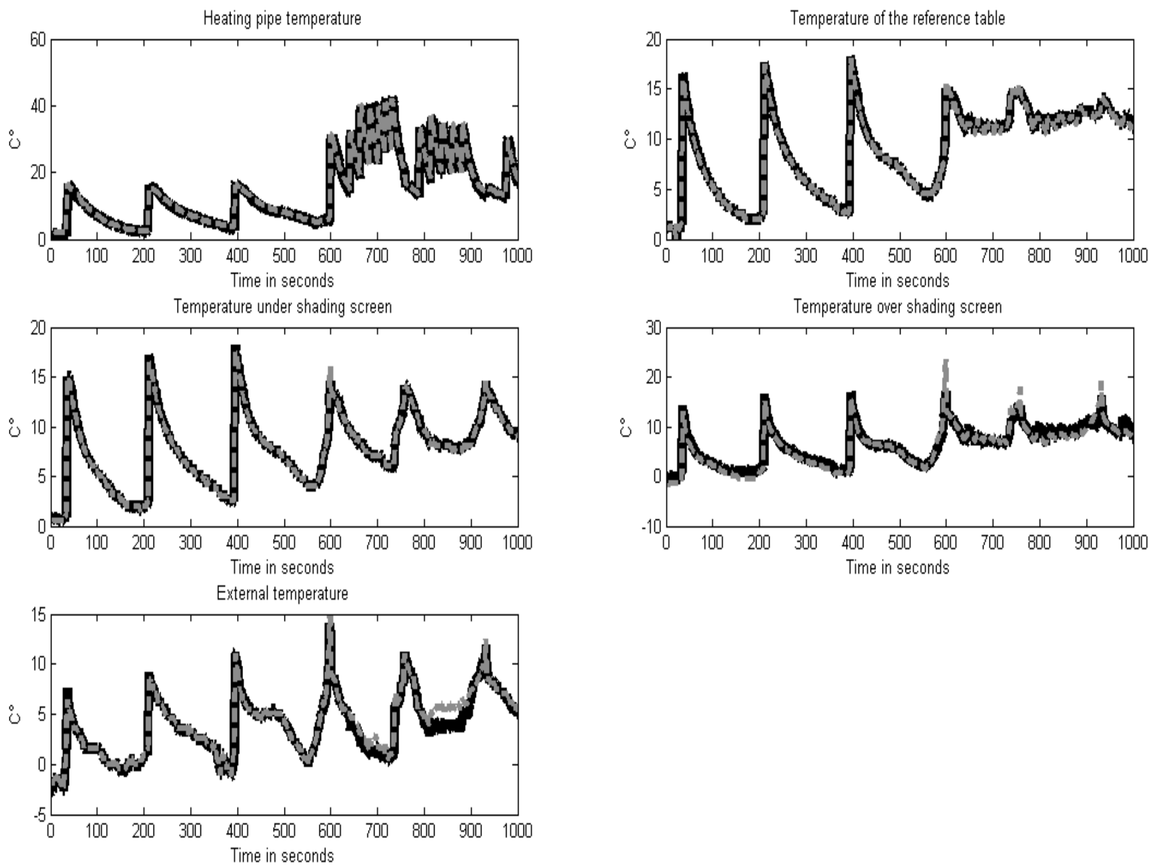

Figure 4. One step ahead prediction for the temperature signals of the greenhouse. The predicted temperature value is plotted with black, and the actual value is plotted with light gray.

TABLE II.
ERROR FOR THE TRAINING AND TEST SAMPLES OF THE ONE STEP AHEAD PREDICTION

	Training samples	**Test samples**	**Long term prediction**
Heating pipe temperature	0.428	2.48	2.638
Temperature of the reference table	0.104	0.788	0.804
Temperature under shading screen	0.104	0.578	0.695
Temperature over shading screen	0.505	1.608	1.788
External temperature	0.125	1.263	1.313

this threshold caused that from the 80000 training samples only 266 were selected as basis function.

The response for the test samples can be seen on Fig. 4. The errors are in Table II.

Setting the threshold to a higher value, thus using more basis function, did not resulted in better accuracy.

Testing the long term prediction for the next 20 minutes had about the same result as the test samples, see Table II.

VI. CONCULSION

Adaptive thermal modeling is the basis for the intelligent greenhouse control solutions to build on. The global greenhouse model is a key element of the decomposed model, as it is aimed to provide forecasts for all important zones of the greenhouse. These forecasts can be transformed later into high spatial resolution predictions for all sensors in the proximity of the plants.

The CMAC global greenhouse model was able to learn the thermal model of the house with high precision for all important zones for 1 step ahead predictions. These 1 step predictions could be fed back to the model, creating 4 step long forecasts, with very similar precision.

ACKNOWLEDGMENT

The authors gratefully acknowledge the support of the Hungarian Fund for Scientific Research (OTKA), Grant #73496.

This work is connected to the scientific program of the "Development of quality-oriented and cooperative R+D+I strategy and functional model at BME" project. This project is supported by the New Hungary Development Plan (Project ID: TÁMOP-4.2.1/B-09/1/KMR-2010-0002).

REFERENCES

[1] X. Blasco, M. Martineza, J.M. Herreroa, C. Ramosa, J. Sanchisa: Model-based predictive control of greenhouse climate for reducing energy and water consumption. Computers and Electronics in Agriculture, pp. 49–70, Elsevier Science Publishers B. V. Amsterdam, The Netherlands, 2007.

[2] P. Eredics: Measurement for Intelligent Control in Greenhouses, 7th International Conference on Measurement, pp 178-181, Smolenice Castle, Slovakia, 2009.

[3] P. Eredics, T.P. Dobrowiecki. Hybrid Knowledge Modeling for an Intelligent Greenhouse, 8th IEEE International Symposium on Intelligent Systems and Informatics, Subotica, Serbia, pp 459-463, 2010.

[4] P. Eredics: Short-Term External Air Temperature Prediction for Intelligent Greenhouse by Mining Climatic Time Series. In: 6th IEEE International Symposium on Intelligent Signal Processing, pp. 317–322, Budapest, Hungary, 2009.

[5] P. Eredics, T.P. Dobrowiecki. Data Cleaning for an Intelligent Greenhouse, 6th IEEE International Symposium on Applied Computational Intelligence and Informatics, Timisoara, Romania, pp 293-297, 2011.

[6] P. Eredics, T.P. Dobrowiecki. Neural Models for an Intelligent Greenhouse - The Heating, 11th IEEE International Symposium on Computational Intelligence and Informatics, Budapest, Hungary, pp 63-68, 2010.

[7] Albus, J.S. (1975) "A New Approach to Manipulator Control: The Cerebellar Model Articulation Controller (CMAC)", Transaction of the ASME, Sep. 1975. pp. 220-227.

[8] G. Horváth, K. Gáti, Kernel CMAC with Reduced Memory Complexity, Proceedings of the 19th International Conference on Artificial Neural Networks: Part I, September 14-17, 2009, Limassol, Cyprus

[9] S. H. Lane, D. A. Handelman, and J. J. 9and, "Theory and development of higher-order CMAC neural networks," IEEE Contr. Syst. Mag., pp. 23-30, Apr. 1992.

[10] T. Szabó and G. Horváth, "Improving the Generalization Capability of the Binary CMAC," Proc. Int. Joint Conf. on Neural Networks, IJCNN'2000. Como, Italy, vol. 3, pp. 85-90, 2000.

[11] C. K. Li and C. T. Chiang, "Neural Networks Composed of Single-variable CMACs", Proc. of the 2004 IEEE International Conference on Systems, Man and Cybernetics, The Hague, The Netherlands, pp. 3482-3487, 2004.

[12] G. Horváth and T. Szabó, Kernel CMAC with improved capability, IEEE Trans. Sys. Man Cybernet. B 37 (1): 124-138, 2007.

[13] Nie, J. and Linkens, D. A. 1994. FCMAC: a fuzzified cerebellar model articulation controller with self-organizing capacity. Automatica 30, 4 (Apr. 1994), 655-664.