Int. J. of Sustainable Water & Environmental Systems
Volume 8, No. 1 (2016) 27-31

Smart Home System for Energy Saving using Genetic-Fuzzy-Neural Networks Approach

Anwar Jarndal*

Electrical and Computer Engineering Department, University of Sharjah, Sharjah, 27272, UAE

Abstract

Home energy saving is very important to realize sustainable improvement. This can be achieved by designing a smart home system that provides a productive and cost-effective environment through optimization of different factors that will be explained in this paper. In this paper, an adaptive smart home system for optimal utilization of power will be designed. The system is based on genetic-fuzzy-neural networks technique, which can capture a human behavior patterns and use it to predict the user's mood. This technique will improve the intelligence of the smart home control to minimize the power losses.

Keywords: *Smart Home, Neural Networks; Genetic Optimization, Fuzzy Logic, Power Saving*

I. Introduction

Smart home is the term commonly used to define a residence that integrates technology and services through home networking to enhance power efficiency and improve the quality of living. This system is generally sensor based system aims to reduce the power losses to limited level.[1-5]. The smart home depends on a computer that controls a building and all its various components like elevators ,telecommunication systems ,security systems and even the consumption of energy level .Therefore, this system detects through its intelligent sensors the activity of the components connected to it to decide the necessary level of applied control to achieve the desired comfort level. In the last few years, artificial neural networks (ANN) has been proved as powerful alternative for smart home system that it is not relying on human experience [6-7]. Neural networks are essentially non-linear circuits that have a demonstrated capability to do non-linear curve fitting [8]. The output of an artificial neural network is some linear or nonlinear mathematical function of its inputs. The inputs may be the outputs of other network elements as well as actual network inputs. In practice network elements are arranged in a relatively small number of connected layers of elements between network inputs and output [8]. The input data will be used to “learn” the model and finding the optimal weights for best fitting for the outputs. After that the model can be used for predicting the output for any other input. The prediction capability of the ANN model can be improved by using a combination of neural networks and fuzzy logic [9]. These two techniques are complementary in the design of intelligent systems and each one has merits and demerits. Neural networks are essentially low-level computational structures and algorithms that offer good performance in dealing with sensory data. On the other hand, fuzzy logic techniques often deal with issues, such as reasoning, on a higher level than neural networks. However, since fuzzy systems do not have much learning capability, it is difficult for a human operator to tune the fuzzy rules and membership functions from the training data set. Also, because the internal layers of neural networks are always opaque to the user, the mapping rules in the network are not visible and are difficult to understand. Furthermore, the convergence of learning is usually slow and may not guaranteed due to local minima problem of the implemented conventional local optimization techniques. Thus, a promising approach for getting the benefits of both the fuzzy systems and neural networks is to merge them in to an integrated system. This collaboration possesses the advantage of both neural networks and fuzzy systems. The convergence of learning can be improved by using a global optimization technique such as genetic algorithm [10].

[*] Corresponding author
E-mail: ajarndal@sharjah.ac.ae

DOI: 10.5383/swes.8.01.006

This paper is aimed to develop ANN model in combination with fuzzy logic system for smart home system. The ANN model accuracy will be improved by adopting a genetic algorithm based optimization procedure for finding the optimal weights of the model.

II. Smart Home System

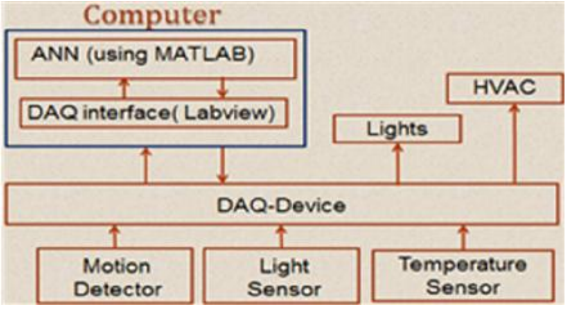

Fig. 1. Block Diagram of the Smart Home System

The block diagram of the proposed smart home system is shown in Fig. 1. The system consists of a group of sensors that monitor the home environment and these sensors are interfaced with a computer through a Data Acquisition Device (DAQ). The input data are conditioned (amplified and filtered) by DAQ that is connected to computer and interfaced by LabView software. The LabView program collects current data and submit it to Matlab code, which classifies the data using fuzzy logic unit. The current classified data in addition to the previous history of the controlled device will be used to construct the ANN model. The ANN model is then predicts the next setting of the controlled device, which is connectedto DAQ, as it is illustrated in Fig. 2 for air conditioner AC as a controlled device.

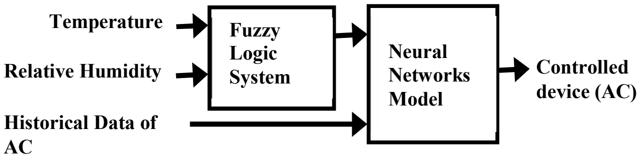

Fig. 2. Block Diagram of Air Condition Fuzzy-ANN Controlling Unit

In this control unit, two climate factors of temperature and humidity are considered. The data of these two inputs are fed to the fuzzy logic system, which outputs the controlled device proportional to these two parameters. However the current device setting is not just depending on the weather but also on the previous setting values. For that reason, the output of fuzzy logic system is fed to the neural networks model for training and comparing a set of collected data (past settings) to adjust the device according to the user's desire. Fig. 3 presents Matlab implementation of the Fuzzy logic model with two data inputs (temperature and humidity) and one control output (device setting).

A. *Fuzzy Logic System*

The main processes of the fuzzy inference system include: fuzzification, knowledge base, decision making and defuzzification. In the first stage of fuzzification the temperature and device setting have been classified into low, medium, high and very high while humidity was categorized into low, medium and high.

The corresponding ranges of temperature are 25-36 °C, 35-41 °C, 40-46 °C and 45-50 °C, respectively. The load is represented by the ranges of 0-4,3.5-7, 6.5-8.5 and 8-10, correspondingly. The load ranges has been defined based on its values for the first day. The values of relative humidity are classified into 60-71, 70-90 and 89-100. As it is illustrated in Fig. 4, triangular membership functions have been used to represent these ranges in Matlab. The rules of mapping between input and output classifications (ranges) are listed in Table I. The estimated values of the device setting (the fuzzy logic output) are represented by the surface in Fig. 5.

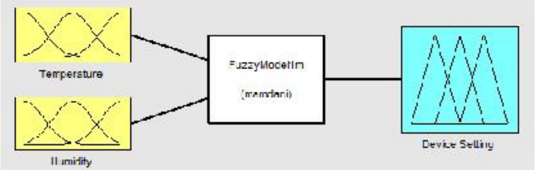

Fig. 3. Fuzyy Logic Model (Matlab Implementation)

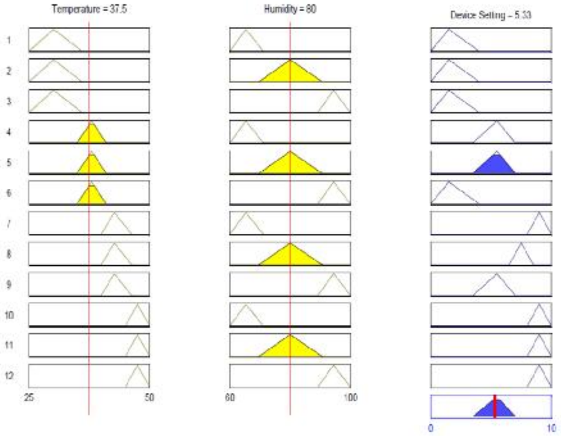

Fig. 4. Inputs And Output Membership Functions

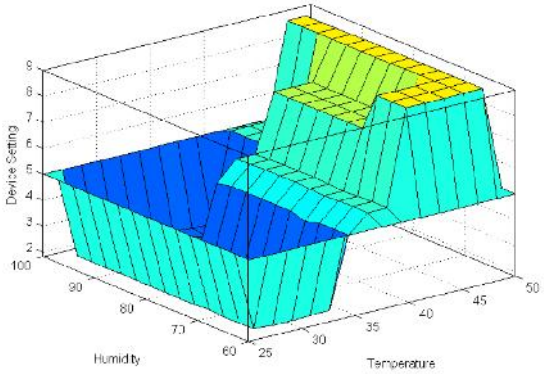

Fig. 5. Variation Of The Fuzzy Logic System Output (Device Setting) With Respect To Temperature And Humidity Inputs

Table 1 Base Rules

Temperature Humidity	Low	Medium	High	Very high
Low	Low level	Medium level	High level	Very high level
Medium	Low level	Medium level	High level	Very high level
High	Low level	Low level	Medium level	Very high level

B. Neural Networks Model

The single hidden layer ANN model shown in Fig. 6 is used to characterize the fuzzy logic output and the past values of the device control signal. This model will "learn" the relation between these two inputs and forecast the future value of the device setting. According to this model, the output can be expressed as

$$Y=\sum_{i=1}^{3} W_i \tanh\left(W_{1i}X_1+W_{2i}X_2+W_{3i}\right) \quad (1)$$

where Y represents forecasted load, X_2 represents the historical load and X_1 is the output of the fuzzy inference system. W_{1i}, W_{2i} and W_{3i} are the input weights and W_i is the output weight. An optimization technique is used in training the neural network model and finding the optimal values for the input and output weights. Here, a genetic algorithm based optimization procedure has been developed.

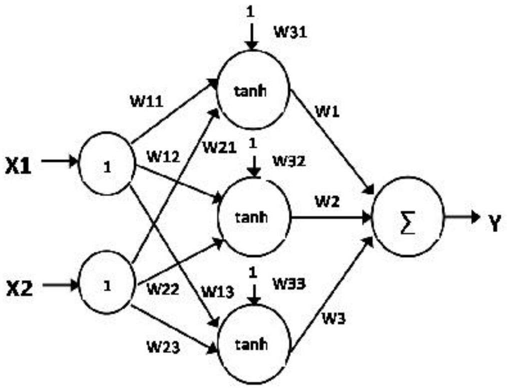

Fig. 6. Single Hidden Layer ANN

The steps of the implemented genetic optimization is presented in Fig. 7 and it can be summarized as follows:

1. Randomly, generation of initial population of individuals. Each individual consists of 12 variables (9 input weights and 3 output weights). The generated values of these weights are within -1 and 1. The current of generation no.1 will be the parents of the next generation individuals and the optimization will continue over N_{max} generation.
2. Calculating the corresponding error between the actual and simulated values of the output for each individual as follows:

- Computing Y in (1) using the values of the input and output weight in the individual over the entire values of the $X_1$and X_2.
- Determining the total error between the simulated Y and the corresponding actual one as follows:

$$Error=\frac{1}{M}\sum_{m=1}^{M}\left(Y_{act}-Y_{sim}\right)^{2} \qquad (2)$$

Where M is the total number of the actual data, Y_{act} is the actual data and Y_{sim} is the corresponding simulated one.

3. Ranking the individuals of the selected population and their errors to reject some of the maximum error individuals in the population.
4. Recombining the selected individuals to perform crossover reproduction by using double-point crossover routine. The individuals are ordered such that individuals in odd numbered positions are crossed with the individuals in the adjacent even numbered position.
5. Mutating (the values of each individual are altered randomly) the reproduced offspring from the crossover process using low probability mutation technique.
6. Repeating step no. 2 to calculate the error of each reproduced individual
7. The next step is reinsertion. Reinsertion replaces the most error individuals in the old population (parents) with individuals in the new reproduced population (offspring).
8. The generational counter is incremented, and the steps from 3 to 7 are repeated until generation no. = N_{max}.
9. When the number of generational counter is equal to N_{max}or the minimum error is smaller than a fixed threshold value ε_s, the algorithm reaches the last generation and stops.
10. The minimum error individual will be chosen and the values of its variables will be considered as optimal values for the network model weights.

III. System Implementation

This procedure has been applied to actual data of temperature, humidity and device setting of 4 days. The temperature and humidity values are fed to the fuzzy inference system, which produces an estimated setting. This value in addition to the historical device setting are then input to the ANN model. The modeling procedure of ANN is started by generating a uniformly distributed random initial population of 10000 individuals. Each individual consists of 12 variables (input and output weights). The maximum number of generations is set to 100 and ε_s is defined to be equal to 0.001. Fig. 8 shows variations of the minimum error of individuals over subsequence generations. As it can be observed from the curves, the error is asymptotically approaches values very close to the minimum possible error and finally the procedure is stopped after small number of generations when ε_s is reached. The simulation time, using 2.27 GHz and 3 GB RAM computer, is 26.8 seconds.

This accordingly demonstrates the very good convergence capability of the developed procedure and its robustness of exploring the search space and capturing the region of the global minimum. Table II lists the optimized weights of the ANN model.

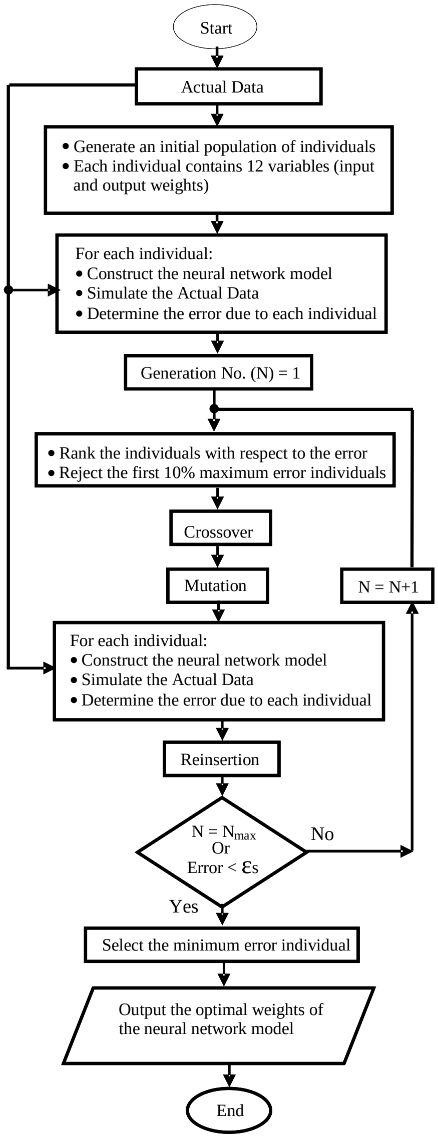
Start
Actual Data
• Generate an initial population of individuals
• Each individual contains 12 variables (input and output weights)
For each individual:
• Construct the neural network model
• Simulate the Actual Data
• Determine the error due to each individual
Generation No. (N) = 1
• Rank the individuals with respect to the error
• Reject the first 10% maximum error individuals
Crossover
Mutation
N = N+1
For each individual:
• Construct the neural network model
• Simulate the Actual Data
• Determine the error due to each individual
Reinsertion
N = N_max Or Error < Ɛs
No
Yes
Select the minimum error individual
Output the optimal weights of the neural network model
End

Fig. 7. Flowchart of the Neural Network Weights Optimization Using Genetic Algorithm

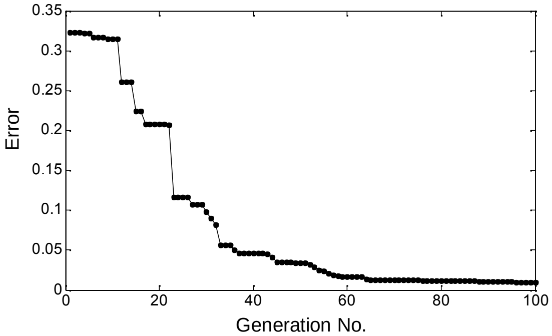

Fig. 8 Variation of Average and Minimum Errors During the Optimization (Training) of the Neural Network Model

Table II
Optimized Weights of the Neural Network Model

Input Weights			Output Weights
w_{11}= 0.0381	w_{12}= 0.985	w_{13}= 0.059	w_1= 0.99
w_{21}= 0.001	w_{22}= 0.205	w_{23}= -0.006	w_2= -0.047
w_{31}= -0.97	w_{32}= 0.33	w_{33}= -93	w_3= 0.391

The whole model is implemented in Matlab and interfaced with LabView and Data acquisition device NI-DAQ. The LabView acquires updated measured data for temperature, humidity and AC operating level from the connected sensors with the DAQ. These data are stored in excel file and exported to the Matlab code of the genetic-fuzzy-neural networks model. Fig. 9 shows LabvView schematic for the whole implemented system.

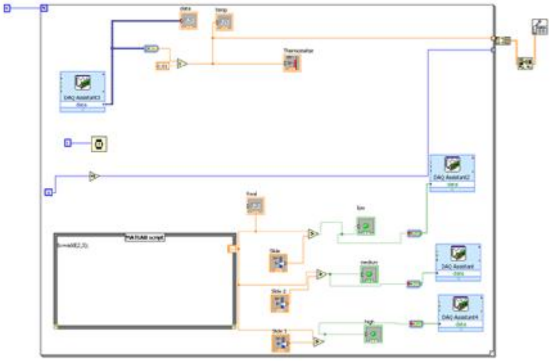

Fig. 9 Labview **Implementation for AC Smart Control Sysyem**

Fig. 10 shows compression between the actual and simulated device setting over the 3days in Dubai on from 26 to 28 of July, 2015. As it can be seen the model can efficiently capture the nonlinear variation of the AC operating level with climate factors and also the linear trend with time. The model accuracy can be further improved by increasing the order of the model using multilayer topology.

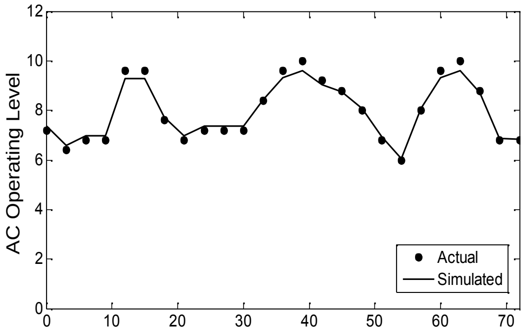

Fig. 10. Comparison between Actual and Simulated AC Operating Levels over 3 Days (26-28 July 2015 in Dubai)

To evaluate the forecasting capability of the developed model, the AC operating levels have been forecasted for the next forth day. The results are compared with actual data. As it can be observed in Fig. 11 the model can accurately predict the level of operation that meet the user's mode.

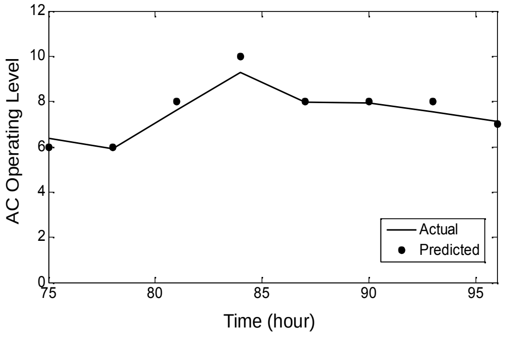

Fig. 11. Comparison between actual and forecasted AC operating levels for the fourth day (29 July, 2015)

IV. Conclusion

In this paper a genetic-fuzzy-neural networks approach for home smart control system is presented. A fuzzy Logic system is used to obtain the variation of the device setting with temperature and humidity. The output of this system is fed to neural networks model, which derives a correction factor based on historical data of the considered AC device and predicts the next level of operation. It has been observed that using genetic optimization improves the learning process of the model.

The model was validated by actual data. Even though the simplicity of the implemented model topology, it can efficiently forecast the future device setting. The model accuracy can be improved by increasing the model order and the training data.

Acknowledgement

The author gratefully acknowledges the support from the Research Institute of Science and Engineering (Mixed Analogue-Digital Smart Electronic Circuits & Systems Research Group), University of Sharjah, Sharjah, UAE.

References

[1] Albert Ting-pat So, WaiLok Chan, Intelligent Building System. Springer publisher, New York, 1999.

[2] Wang Shengwei, Intelligent Buildings and Building Automation. Taylor & Francis publisher, New York, 2010.

[3] Jian Li, Jae Yoon Chung, Jin Xiao, "On Design And Implementation of a Home Energy Management System," International Symposium on Wireless and Pervasive Computing (ISWPC), Hong Kong,February, pp. 1-6.

[4] J. Shah, L. Pathrabe and B. Patel, “Wireless smart power saving system for home automation,” International Conference on Emerging Technology Trends in Electronics, Communication and Networking (ET2ECN), Surat, Gujarat, India,DEcember 2012, pp. 1-6.

[5] Basil Hamed, “Design & Implementation of Smart House Control Using LabVIEW," International Journal of Soft Computing and Engineering (IJSCE) ISSN: 2231-2307, Volume-1, Issue-6, January 2012.

[6] Amit Badlani, Surekha Bhanot, “Smart Home System Design based on Artificial Neural Networks,” Proceedings of the World Congress on Engineering and Computer Science 2011 Vol IWCECS 2011, October 19-21, 2011, San Francisco, USA.

[7] J. G. Carbonell and J. Siekmann, “Designing Smart Homes: The Role of Artificial Intelligence,” Lecture Notes in Artificial Intelligence (Subseries of Lecture Notes in Computer Science), Springer, 2006.

[8] Kevin Gurney, An introduction to neural networks, UCL Press, London, 1997.

[9] Bart Kosko.Neural Networks and Fuzzy Systems: A Dynamical Systems Approach to Machine Intelligence. Prentice Hall, 1991.

S. Rajashekaran and G.A. Vijayalksmi. Neural Networks, Fuzzy Logicand Genetic Algorithms: Synthesis and Applications.PHI Learning Pvt. Ltd., 2003