INTELLIGENT COMPUTER CONTROL OF AIR CONDITIONING SYSTEMS BASED ON GENETIC ALGORITHM AND CLASSIFIER SYSTEM

H.N. Lam
Department of Mechanical Engineering
University of Hong Kong
Pokfulam Road, Hong Kong

ABSTRACT

Conventional control schernes for air condítioning systems are in lack of the capabilities to adapt to a changing enviroranent and to optimize against given criteria. In this paper, a methodology is presented, which employs a classifier system with genetic algorithm to enable an air-conditioning controller to learn from its own experience the best control strategy against a given performance evaluation scherne. Solar insolation and outdoor air temperature are chosen to be the input environmental variables of the classifier system which utilizes a simple genetic algorithm to formulate the optimal control rule based on Fanger's thermal comfort index of predicted mean vote (PMV). A split-unit air conditioning system with a variable-speed compressor is used for experimental testing of the genetics-based rule learning system. Preliminary results obtained are encouraging.

INTRODUCTION

Throughout the world, air conditioning systems for buildings are a major consumer of electrical energy. To control such systems efficiently and effectively in the presence of dynamic interactions and random disturbances so as to conserve energy while maintaining the desired thermal comfort level requires more than a conventional control methodology (Lam 1993). By using a classifier system with genetic algorithm, a model-free learning and control strategy can be implemented which is capable of formulating, and continuously updating the optimal control rule against given performance criteria (Goldberg 1983). Compared to conventional control schemes, therefore, this new control strategy has obvious advantages in terms of self-learning and adaptive capabilities when operating under a varying environment. Classifier systems are genetics-based machine learning systems which learn rules to guide their performance in an arbitrary environment. Genetic algorithms are search procedures based on the law of natural selection which emulate the adaptation of a species to its environment according to genetic rules. A classifier system incorporating a genetic algorithm for rule optimization is proposed for the computer control of a split-unit air-conditioning system for machine learning.

CLASSIFIER SYSTEM AND GENETIC ALGORITHM

A classifier system is a general-purpose machine learning system which learns string rules incrementally to guide its performance in some specified environment (Holland 1986, Goldberg 1983, Davis 1987). A classifier system comprises the following components:

(1) Performance Component
(2) Reinforcement Component
(3) Discovery Component

In the Performance Component, there are string rules called classifiers which have the following form:

classifier : <condition> : <action>

Both the condition and action parts are finite-length strings over the three-letter alphabet {1, 0, #}. In the condition part, the # symbol plays the role of a "don't care" element in the sense that it matches either a 0 or a 1. In the action part, the same symbol plays the role of a "pass through" element in the sense that whenever the # occurs in the action part, the corresponding bit in a message satisfying the condition is passed through into the outgoing message. A classifier is activated when the condition part is matched by a message in the message list which contains all the messages posted by the detectors and effectors. Incoming information from the environment to the classifier system is processed through the detectors where it is decoded into a finite-length message. Likewise, an action taken by the classifier system is triggered through the effectors.

In the Reinforcement Component, the probability of a classifier with matched conditions becoming active is governed by the size of its bid which depends on three parameters: strength, specificity and bid coefficient. Strength is a classifier's net worth. Specificity is a measure of the generality of the classifier's conditions. Specificity of a classifier is indicated by its bid ratio:

$$\text{Bid Ratio} = (2^{\text{no. of bit}} - 2^{\text{no. of \#}} \quad (2^{\text{no. of bit}})$$

Bid coefficient is a small constant, usually about 0.1, which determines the proportion of a classifier's strength that will go to a bid.

The bid of a classifier at time-step t is given by the equation:

$$\text{Bid}(t) = \text{Bid Coefficient} \times \text{Strength}(t) \times \text{Bid Ratio}$$

Holland's bucket brigade algorithm is used in the determination of a classifier's strength at each time-step. When a classifier is activated, its strength is reduced by an amount equal to its bid which in turn is distributed and shared by those classifiers which posted messages matching the bidding classifier's condition. If a payoff is received from the environment, the strengths of the activated classifiers will be increased proportionately to reflect a total adjustment equal to the payoff. Over time, the bucket brigade algorithm reallocates strength from classifiers receiving environmental payoffs to those classifiers which act as stage-setters and indirectly lead to payoff.

In the Discovery Component, genetic algorithm is employed to inject new, possibly better classifiers into the system. The invocation of genetic algorithm learning is governed by the GA period, T_{ga} which specifies the number of time steps between GA calls. A proportion of the population of classifiers is replaced at a given genetic algorithm invocation. Three genetic operators are involved: Reproduction, Crossover and Mutation. Reproduction is a process in which individual classifiers are copied according to their fitness measured by the strength value. Classifiers with a higher fitness value have higher probability of having one or more offspring in the next generation. In Crossover, newly reproduced classifiers are first mated at random and then a random number of individual bits in a mated pair of classifiers are swapped to yield a pair of new classifiers. Mutation is the occasional random alteration of the value of a string position within a classifier. Since classifiers are defined on the ternary alphabet {1, 0, #}, the changed character is selected with a probability of 0.5 from the two remaining characters.

For a detailed explanation of the principles of operation of classifier system and genetic algorithm, the interested reader is referred to Goldberg's work (Goldberg 1989).

THERMAL COMFORT CRITERION

According to the theory of Fanger (Fanger 1972), the conditions for optimal thermal comfort for a person in a given environment are expressed by the Fanger's comfort equation. For any given type of clothing worn and activity engaged in, it is possible to use the comfort equation to determine all reasonable combinations of air temperature, air humidity, mean radiant temperature and relative air velocity which will create optimal thermal comfort for persons under steady-state conditions. Based on the comfort equation, Fanger developed a thermal sensation index called Predicted Mean Vote (PMV) which can be used to predict the thermal sensation of a person for any given combination of the six parameters mentioned above when the comfort equation is not satisfied. There are, however, criticisms on the validity and suitability of Fanger's comfort equations for field applications as they have only been validated under steady-state laboratory conditions.

In this study, the PMV value was used as a comfort variable for the control of the air-conditioning system to achieve specified comfort levels.

IMPLEMENTATION OF MACHINE LEARNING FOR COMPUTER CONTROL OF AN AIR CONDITIONING SYSTEM

In this study, a split-unit air-conditioning system which served the Building Services Laboratory in the University of Hong Kong was chosen. A frequency inverter was installed to modulate the compressor speed for control of the cooling capacity of the air-conditioner. A 486-microcomputer fitted with an analog-to-digital and digital-to-analog card was employed for data acquisition, execution of the machine learning program and control of the frequency inverter. Three environmental variables were monitored using two temperature sensors and one pyranometer: outdoor air temperature, solar radiation and indoor air temperature. The measured outdoor air temperature and solar radiation were fed to the detectors of the classifier system where they were encoded into a 6-bit string and a 7-bit string respectively, which were then concatenated into a 13-bit message to represent the condition part of the classifier. The indoor air temperature was used in the formulation of a payoff function to constitute the reinforcement component of the classifier system to reward a winning classifier for good performance. The output from the classifier system through the effectors was in the form of a 13-bit action message used to control the frequency inverter output within the range from 30 to 50 Hz. The lower limit of the power supply frequency was chosen to ensure that the compressor would not operate at too low a speed which could cause insufficient circulation of lubricating oil in the refrigerant. The machine learning program was developed using the C language under a commercially available integrated development environment which supports compilation, debugging, data acquisition and user interface functions. The following important parameters were defined for the machine learning system, the values of which can be set interactively by the program:

. Population Size

This is the number of classifiers in the classifier system. The default value is 300 and the maximum value is 500.

2. Initial Strength

It defines the strength of all the classifiers at the start of the program. The default value is 200.

3. Bid Coefficient

This coefficient is used in the bidding equation to determine the value of a bid. The default value is 0.1.

4. Time-Step

This is the time period between successive scanning of input parameters. The default value is 5 minutes.

5. Payoff

Payoff =50 + 50 [(Tolerance - Deviation)/Tolerance]
if |Deviation| < Tolerance
= 0
if |Deviation|>Tolerance

where

Tolerance = specified tolerance of PMV value
Deviation = deviation from the set PMV value

6. PMV Tolerance

This parameter is used to define when and how a payoff is to be given to the winning classifier.

7. Probability of Crossover

The default value is 0.95

8. Probability of Mutation

The default value is 0.

9. GA Period

This is the number of time steps between successive invocations of genetic algorithm. The default value is 50.

The program allows the values of the following environmental parameters to be specified for a particular situation prior to program execution (default values are as shown in brackets):

1. Activity Level (100 kcal hr^{-1} m^{-2})

This defines the type of activity for the persons in the conditioned space.

2. Mechanical Efficiency (5%)

This refers to the mechanical efficiency of a human body.

3. Clothing Factor (1.15)

This defines the ratio of the area of clothing to that of the nude body.

4. Clothing Insulation (1 Clo)

This defines the thermal resistance of clothing worn.

5. Relative Air Velocity (0.62 ms^{-1})

This is the air velocity in the conditioned space.

6. Relative Humidity (40%)

This is the relative humidity of the air in the conditioned space.

The operation cycle of the on-line computer control system for machine learning follows the sequence as described below:

(1) At the start of the nth time-step, the outdoor air temperature and solar radiation are measured and input to the classifier system.

(2) The required control voltage for the frequency inverter to be applied at the next time-step is determined by the classifier system.

(3) At the start of the (n+1)th time-step, the appropriate control voltage is applied to the

frequency inverter throughout the time-step to run the compressor at the required speed.

(4) At the start of the (n+2)th time-step, the indoor air temperature is measured and used to calculate the payoff to be awarded to the winning classifier.

The operation cycle is then repeated.

EXPERIMENTAL RESULTS

The air-conditioning system was operated under the real-time control of the computer-based classifier system for machine learning. At the chosen time step of 5 minutes, the control algorithm worked satisfactorily, yielding a calculated output value for the frequency inverter input at every time step.

Two tests were carried out in which the GA Period was set at 20 and 50 in order to assess the effect of the frequency of invocation of genetic algorithm on the performance of the classifier system. For each GA Period, the graphs of indoor air temperature, PMV value and operating frequency for the compressor were plotted against time-step as shown in Figures 1 to 6. Since the weather during the testing period was still cold, the desired indoor air temperature was set at 18.8°C as determined by the comfort equations. It was found that the GA Period had substantial effect on the proper operation of the machine learning system. With a GA Period of 50, it can be observed that the classifier system operated satisfactorily and had learnt to control the conditioned space at acceptable temperatures. In comparison, the results obtained by using a GA Period of 20 were inferior. This showed the dependence of the performance of the classifier system on how frequently the genetic algorithm was invoked; too high an invocation rate would tend to weaken the learning capability of the classifier system.

The selection proportion by which a proportion of the population of classifiers is replaced at an invocation of genetic algorithm was varied over the range 0.5 to 1.0 in order to assess the sensitivity of the control methodology to population overlap. The results showed that the effects of selection proportion on control performance were insignificant although in principle it is generally not desirable to adopt the nonoverlapping population approach (selection proportion = 1) for machine learning applications.

CONCLUSIONS

In this research study, it was found that classifier system with genetic algorithm was applicable to on-line computer control of air-conditioning systems and implementation of a self-learning control system. The machine learning methodology was implemented on a microcomputer to successfully control a split-unit air-conditioning system without the use of any system model. Initial tests yielded promising results indicating the applicability of classifier system and genetic algorithm to intelligent real-time control of air-conditioning systems.

DISCUSSIONS AND SUGGESTIONS FOR FURTHER RESEARCH

The research findings provide some encouragement for further investigations into many other aspects relating to the use of classifier system with genetic algorithm for on-line control of air-conditioning systems. In order that the performance of the control scheme can be studied in greater details under a repeatable and adjustable indoor and outdoor environmental conditions, a building emulator can be used to simulate the building and air conditioning system in real time to replace their physical counterparts. Work is currently under way to develop such a building emulator using the HVACSIM$^+$ simulation package and the GPIB interface for data communication with the classifier control program running in a separate microcomputer.

The present work deals only with the application of the machine learning methodology to low-level control loop of the air-conditioning system directly. It is envisaged that the methodology is equally powerful and suitable for application to distributed hierarchical control systems in which the supervisory role will be taken up by the machine learning system. No account of the effects of usage pattern of the conditioned space on the required control action has been taken in the present study. Lighting level and occupancy are important factors to be considered by the machine learning system to cater for heat loads other than those due to the external environment. More work also needs to be done in the area of fine-tuning of parameters for classifier system and genetic algorithm to obtain improved performance.

ACKNOWLEDGEMENTS

The author would like to express his appreciation to the University of Hong Kong for the award of a grant in partial support of this research. Thanks are also due to Mr. H.Y. Leung, who was a final-year student in the Department of Mechanical Engineering in 1993/94, for his many contributions to the successful completion of this research work.

REFERENCES

Davis, L. (Ed.) 1987. *Genetic Algorithms and Simulated Annealing*. Pitman, Morgan Kaufmann.

Fanger, P.O. 1972. *Thermal Comfort Analysis and Applications in Environmental Engineering.* McGraw-Hill.

Goldberg, D.E. 1983. "Computer-Aided Gas Pipeline Operation Using Genetic Algorithms and Rule Learning." Doctoral Thesis, University of Michigan.

Goldberg, D.E. 1989. *Genetic Algorithms in Search, Optimization, and Machine Learning.* Addison-Wesley.

Holland, J.H. et al 1986. *Induction: Processes of Inference, Learning, and Discovery*. MIT Press.

Lam, H.N. 1993. "Stochastic Modelling and Genetic Algorithm-Based Optimal Control of Air Conditioning Systems.", *Proceedings of the Building Simulation '93 Conference*, the Third International Conference of the International Building Performance Simulation Association, Adelaide, Australia, 16-18 August 1993, 435-441.

FIGURES

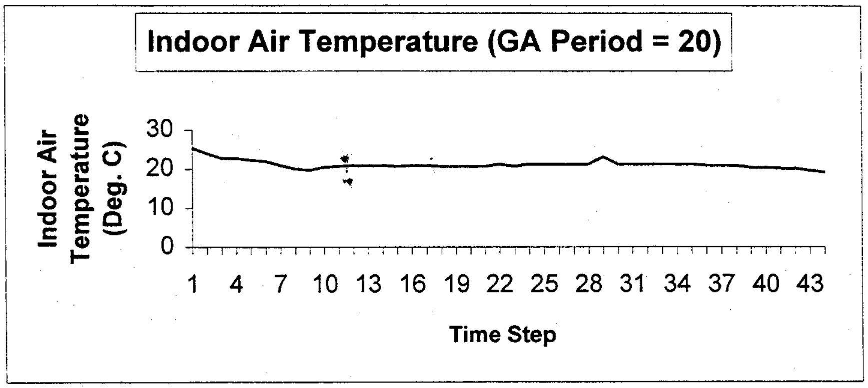

Figure 1: Graph of Indoor Air Temperature Vs Time Step (GA Period = 20)

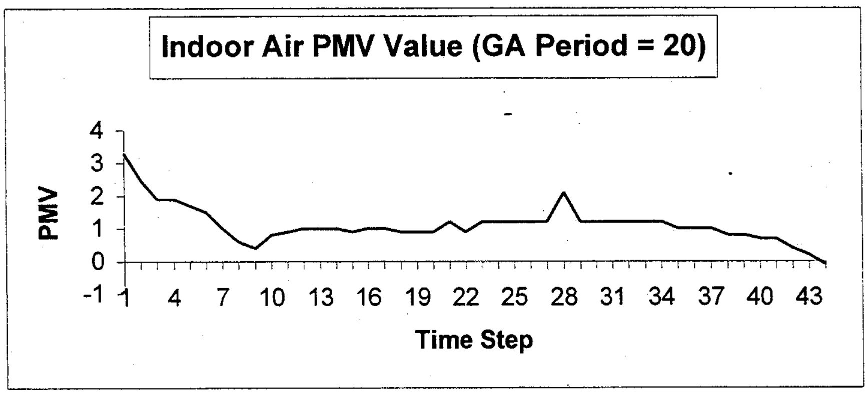

Figure 2: Graph of Indoor Air PMV Value Vs Time Step (GA Period = 20)

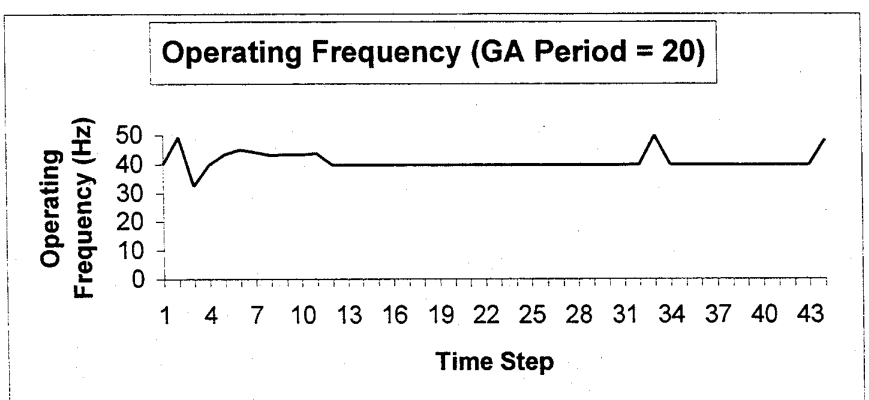

Figure 3: Graph of Compressor Operating Frequency Vs Time Step (GA Period = 20)

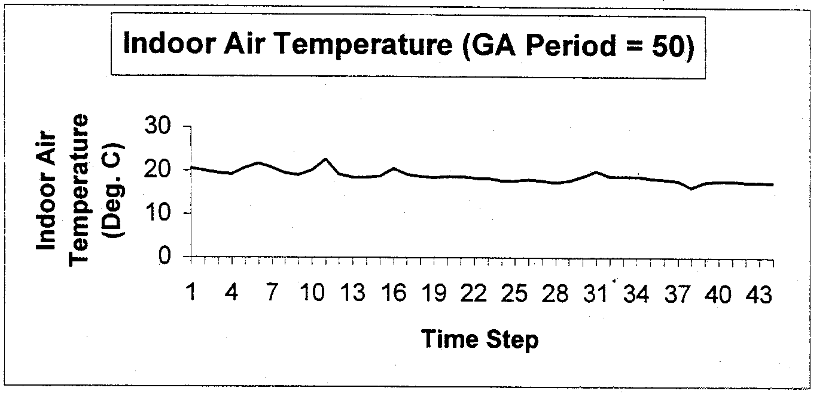

Figure 4: Graph of Indoor Air Temperature Vs Time Step (GA Period = 50)

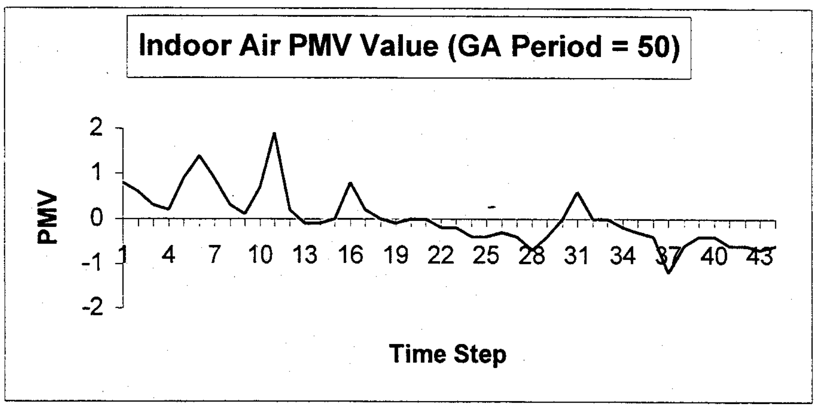

Figure 5: Graph of Indoor Air PMV Value Vs Time Step (GA Period = 50)

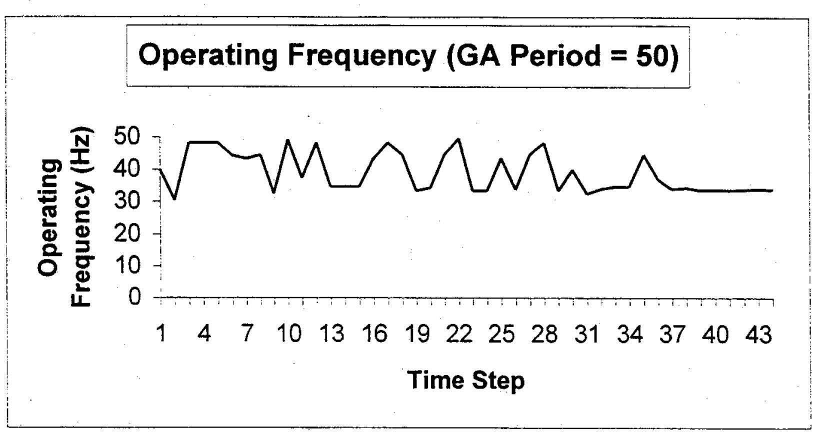

Figure 6: Graph of Compressor Operating Frequency Vs Time Step (GA Period = 50)