Modeling Occupancy Behavior for Energy Efficiency and Occupants Comfort Management in Intelligent Buildings

Tina Yu
Department of Computer Science
Memorial University of Newfoundland
St Johns, NL A1B 3X5 Canada
Email: gwoingyu@gmail.com

***Abstract*—We applied genetic programming algorithm to learn the behavior of an occupant in single person office based on motion sensor data. The learned rules predict the presence and absence of the occupant with 80%–83% accuracy on testing data from 5 different offices. The rules indicate that the following variables may influence occupancy behavior: 1) the day of week; 2) the time of day; 3) the length of time the occupant spent in the previous state; 4) the length of time the occupant spent in the state prior to the previous state; 5) the length of time the occupant has been in the office since the first arrival of the day. We evaluate the rules with various statistics, which confirm some of the previous findings by other researchers. We also provide new insights about occupancy behavior of these offices that have not been reported previously.**

***Keywords*-buildings occupancy model; energy efficiency; buildings simulation; genetic programming; sensor data.**

I. Introduction

While investment in newer energy resources, such as wind and solar power, or conventional ones, such as coal, gas, oil or nuclear power, are important for protecting the environment and improving the economy, it has been identified that investment in energy efficiency benefits several times more than any other kind of energy investment [3]. Energy efficiency means delivering an improved level of service – or at least service at the same level – with less energy consumption. Buildings, which are responsible for at least 40% of energy use in many countries, are becoming a worldwide focus of energy efficiency improvement.

The estimation of a building's energy consumption is normally carried out by building simulation tools, such as Energy+ and eQuest. These tools are helpful in designing energy saving in both equipments, such as lighting, heating, ventilation and air conditioning (L-HVAC) systems, and in building levels. One crucial information for the estimation of the energy demand of a building is its occupancy. When a building is occupied, the individuals modify the indoor environment, as human beings emit heat and pollutants. They also interact with the building, such as adjusting lighting and air conditioning systems, to enhance their comfort. They also operate electronic appliances, such as computers, which consume energy and produce heat. A model that predicts the presence or absence of occupants in an building helps the simulation tool to produce more accurate energy consumption estimation.

Occupancy models are also important in running energy efficient buildings. For example, in an office building, the simplest occupancy model is to assume that all offices are occupied during working hours and not being used at night and weekend. Obviously, the L-HVAC system controlled under this model is not optimal as it wastes energy when the offices are not occupied and sacrifices the comfort of the occupants who work after hour or weekend. Over the past decade, much research has devoted to developing intelligent buildings by applying machine learning techniques on various kinds of sensor data to model occupancy behavior with the goal of reducing energy consumption while maintaining maximum comfort for the occupants (see Section II).

This work has a similar goal but differs from other works in that we apply Genetic Programming (GP) [4] to learn occupancy behavioral rules using low-cost motion sensor data. While motion sensors can provide information about the presence and absence of the occupant in an office, they do not have predictive power. By contrast, our occupancy rules can anticipate occupancy based on historical information, hence can be used to optimize building systems, such as HVAC, by keeping it off until just before arrival and then pre-condition as necessary, thus saving energy. Moreover, rules can be analyzed and interpreted to obtain knowledge, unlike previous work of Markov chain models[5], which are black-box and can not be interpreted easily. To convince a domain expert (building engineers in this case) to adopt a solution, interpretability of the solution is very important.

The learned rules give prediction accuracy of 80%–83% on testing data from 5 different offices. Moreover, the rules reveal that the following variables may influence occupancy behavior: 1) the day of week; 2) the time of day; 3) the length of time the occupant spent in the previous state; 4) the length of time the occupant spent in the state prior to the previous state; 5) the length of time the occupant has been in the office since the first arrival of the day. Although the study is on single occupancy office and the testing data is only 4 weeks long, these results encourage us to extend the work to multiple occupancy offices with a longer period of testing data.

The rest of the paper is organized as follows. Section II summarizes works related to this research. In Section III, the motion sensor data used in this study are described. Section IV presents the GP system we used to learn the occupancy behavior rules. The results are reported in Section V. We evaluate the model performance based on different statistics in Section VI. Finally, Section VII concludes the paper and outlines our future work.

II. Related Work

Based on their analysis of the motion (infrared) sensor data collected from 35 single occupancy offices, Wang and colleagues [6] hypothesized that the occupancy and vacancy intervals in a single person office (the length of time the office is occupied and unoccupied) have an exponential distribution. They then devised a statistical model to simulate the occupancy behavior in a single person office. Their simulation results supported the case of vacancy intervals, but not the case of occupancy intervals.

That work also made two important observations: 1) the distribution of occupancy intervals is time varying; 2) different offices have different hourly occupancy rates. Based on these information, Page and colleagues proposed a generalized Markov chain model to simulate the occupant's presence in an office [5].

In their approach, each occupant in a single person office had a different profile describing his/her probabilities of presence in the office at different time of day. The profile was based on the motion sensor data collected from that office. The profile was then used to construct a Markov chain model, which simulated the presence/absence of an occupant in the office at each time step (15 minutes in their case). They reported that the simulated results matched well to the original sensor data used to develop the profile, in terms of the probability of presence of the occupant. However, they did not provide prediction accuracy of the model. Instead, they evaluated their model based on the following statistics:

- *first arrival and last departure times*: the information is important for modeling activities that an occupant performs when first arrive to an office (e.g. window opening) and before leaving the office for the day;
- *cumulated presence per day*: the information is used to estimate energy consumed by the office appliances;
- *periods of intermediate presence and absence*: these are the occupancy and vacancy intervals studied in [6];
- *number of changes of the state of presence during the same day*;

For the first arrival and the last departure times, the authors noted that they were particular to the occupant and that these times could depend on the day of week. In terms of the periods of intermediate presence and absence, their model simulation results confirmed that reported in [6]: the occupancy interval did not follow an exponential distribution and that each occupant had his/her own behavior.

We will use the same sensor data to conduct our study (see Section III) and provide similar statistics to evaluate our models in Section VI.

In contrast to the two previous works, Dong and Andrews [1] modeled occupancy behavioral patterns using a wider range of sensor data: acoustics, lighting, motion, CO_2, temperature and relative humidity. They first applied an unsupervised learning algorithm to discover event patterns from these sensor data. They then used the patterns to generate a Markov model that described an occupant's behavior transition over a time period. They tested their model performance on a HVAC system by running the Energy+ simulation tool. The results suggested that their model had potential energy saving of 30% compared with other basic HVAC energy saving control strategies.

Dong and colleagues [2] also tackled a more challenging task of measuring the number of occupants present in two multi-occupancy rooms. Among a very rich set of available sensor data (CO_2, CO, total volatile organic compounds, small particulate, acoustics, lighting, motion, temperature and humidity etc.), their feature selection algorithm picked 6 of them for modeling; some of them were higher-order of CO_2 and acoustics that they designed by hand, e.g. CO_2_diff and acoustics_FD2. The selected sensor data (2-4 weeks) were then used to train Support Vector Machines (SVM), Neural Network (NN) and Hidden Markov Models (HMM) to predict the number of occupants in these two rooms. They also used cameras to capture occupants' movements from one room to another, which they called ground truth data. When evaluated on a testing data set (24-hour) against the ground truth data, the SVM, NN and HMM models gave similar average prediction accuracy about 73%. The authors noted that the quality of acoustics data could be effected by activities in nearby area. Also, the moving average CO_2 information had caused the model to delay in detecting the office has become empty.

III. The Motion Sensor Data

We used the same data used in [5] to conduct our study. These data were collected from 5 singly occupied offices (A to E). The sensor only recognized two states of presence: occupied and vacant. When the state changed, the time and date were recorded. In other words, the data are a time-series of alternating 1 (indicating the state changed from vacant to occupied) and 0 (indicating the state changed from occupied to vacant) with their associated time stamps. Also, the 0-type data that was less than 2-minute long were removed as they corresponded to the case where a sensor stopped sensing the presence of the occupant because (s)he was too still without any movement.

Among the provided data (1 year), we selected those from January 7 to March 1, 2002 for training and from March

4 to March 28, 2002 for testing. Moreover, weekend data were removed since the occupancy pattern during weekend is different from that during weekday. The final training set contains 40 days (8 weeks) of sensor data while the final testing set contains 19 days (4 weeks) of sensor data. The number of data points (the number of occupied/vacant changes) for training and testing are given in Table I.

Table I
THE NUMBER OF DATA POINTS USED FOR TRAINING AND TESTING.

Office	A	B	C	D	E
Training	1672	1366	2008	2238	1768
Testing	666	630	1142	1056	848

IV. METHODOLOGY

GP is a machine learning paradigm that is commonly used for supervised learning. In our case, we used GP to learn the occupancy behavior, which was represented as rules, describing the conditions when the occupant was absent (0 as the target label) and when the occupant was present (1 as the target label) in the office. Based on the findings by other researchers (see Section II), we designed the following variables for GP to learn the occupancy behavioral rules:

- `day`: day of week (valued between 1 and 5)
- `hour`: hour of day (valued between 0 and 23)
- `minute`: minute of hour (valued between 0 and 59)
- `noChange`: the length of time (in minute) the occupant spent in the previous state (`t-1`)
- `sameLast`: the length of time (in minute) the occupant spent in the state prior to the previous state (`t-2`)
- `accIn`: the length of time (in minute) the occupant has been in the office since the first arrival of the day.

`day`, `hour` and `minute` were based on [6] and [5] who reported that occupancy behavior varied depending on the day and time. `noChange` was motivated by the Markov chain model of [5], which predicted the current state based on previous state. The other two variables were our inventions based on common sense. It seemed that the length of time one last spent in the same state (e.g. the last break I was away from my office) may influence the length of time one may spend in the current state (e.g. the current break I will take away from my office). Also, some people might take more breaks away from their office when they are tired at the end of the day while others might like to take care of jobs requiring them being away from their offices when they first arrive. We therefore included `sameLast` and `accIn` information for GP to learn occupancy rules.

Three kinds of random constants were provided: `randomDay` (valued between 1 and 5), `randomHour` (valued between 0 and 23) and `randomMinute` (valued between 0 and 59).

We also provided operators (see Table II) for GP to combine the above variables and constants to create rules. Since the variables have different types (e.g. `hour` type is different from `day` type), it is important to specify the types of variables that these operators can be applied. For example, $<$ can be applied to compare two variables of the same type, e.g. `hour` with `hour`, `minute` with `minute` and so on. The a is a type variable which can be substituted with any concrete type, such as `hour` and `minute`.

Table II
THE NON-TERMINALS FOR GP TO CONSTRUCT BEHAVIORAL RULES.

Name	Type	Name	Type
and	$bool \rightarrow bool \rightarrow bool$	+	$a \rightarrow a \rightarrow a$
or	$bool \rightarrow bool \rightarrow bool$	-	$a \rightarrow a \rightarrow a$
not	$bool \rightarrow bool$	$\times$	$a \rightarrow a \rightarrow a$
if-then-else	$bool \rightarrow a \rightarrow a \rightarrow a$	==	$a \rightarrow a \rightarrow bool$
$>$	$a \rightarrow a \rightarrow bool$	$<$	$a \rightarrow a \rightarrow bool$

We used a GP system that supports type checking to learn the occupancy rules. For more details about the type checking mechanism and the GP system, please refer to [7].

A. Experimental Setup

As each office had its own occupancy behavior, we trained one rule for each office using the sensor data collected from that office. We first made single GP run on each of the training set using a range of parameter values: population size – 10, 50, 100; crossover rate – 0, 0.2, 0.5, 0.8, 1; mutation rate – 0, 0.2, 0.5, 0.8, 1; number of generation – 300, 600, 3000. The parameter values that produced the best run were used to conduct subsequent runs. In this case, the same set of parameter values (see Table III) produced the best result for all 5 training sets.

Table III
GP PARAMETERS VALUES TO LEARN OCCUPANCY RULES.

Parameter	Value	Parameter	Value
population size	10	number of run	40
number of generation	3,000	elitism	1
maximum rule length	200	mutation rate	1

The fitness function is the prediction accuracy:

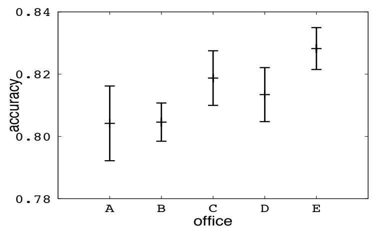
$$f = \frac{\sum_{i=1}^{N} hit_i}{N} \qquad hit = 1 \text{ when } predicted = target.$$

where N is the number of data in the training set. When predicted equals to target, it means that at the time when the change from occupied to vacant or from vacant to occupied took place, the rule correctly predicted the change.

V. RESULTS AND INTERPRETATION

Table IV (left) gives the prediction accuracies of the 5 occupancy rules for the 5 offices (A-E) on the training data. They are averaged over 40 runs with error bars indicating standard deviations. For each of the 5 offices, we selected the rule that gave the best training accuracy and applied it to the testing data. The results are given on the right of Table IV. As shown, these 5 best rules give similar prediction accuracies on training and on testing data (between 80% and 83%), indicating that they are robust, not over-fitting. Meanwhile, the prediction accuracies of these 5 rules are close to each other, indicating GP is a suitable algorithm for learning occupancy rules from motion sensor data.

Table IV
RULE ACCURACY ON TRAINING DATA (LEFT) & TESTING DATA (RIGHT).



Office	Accuracy
A	0.831832
B	0.8
C	0.797723
D	0.830492
E	0.838443

A. Occupancy Rules Analysis

We also analyzed the best 5 occupancy rules for each of the 5 offices. Due to space constraint, we only list the variables appeared in the 5 rules in Table V without giving the entire rules. Interested readers can refer to the website: http://sites.google.com/site/occupancyrules/ for details.

Table V
VARIABLES APPEARED IN THE 5 BEST OCCUPANCY RULES.

Variable	A	B	C	D	E
day	×				
hour		×	×		
minute					×
noChange	×	×	×	×	×
sameLast	×	×	×	×	×
accIn	×	×	×	×	×

For office A, the rule involves 4 variables: `day`, `noChange`, `sameLast` and `accIn`. In particular, the rule shows that the individual has a different behavior pattern on Friday from that on the rest of the week. For office B, instead of `day`, the rule uses `hour` to predict occupancy behavior. In particular, between 10:00 and 11:00 am, the individual behaves differently from that during the rest of the day. Also, if the individual's first arrival is later than 13:00, he/she has a different behavior pattern from his/her normal pattern. The rule for office C contains the same variables as that of office B. However, the rule indicates that before 10am the individual has a different behavior pattern from that of the rest of the day. The rule for office D shows that time and day do not influence the occupancy pattern. The rule for office E indicates that the occupancy pattern at the first 2 minutes of the hour is different from that of the rest of the hour. These results agree with previous work [6] that different office (individual) has different occupancy behavior pattern and the day of week [5] and time of day [6] may influence an occupant's behavior.

VI. ANALYSIS AND DISCUSSION

Figures 1 to 5 give the probabilities of presence at different time of day based on the 4 weeks of testing data. We first partitioned the time of day into 2-minute intervals. At each interval, we calculated the percentage of days among the 4 weeks that the office was occupied, according to the predicted and the sensor data. This gives the probability of presence of the office at that time interval. As shown, the predicted data follow the trends in the sensor data well, except the final departure time, which we will give more discussion in the subsection. One observation is that while all office occupants took 10am, lunch and 4pm breaks, the occupant in office C didn't. The individual only took the 10am break and worked through the rest of the day.

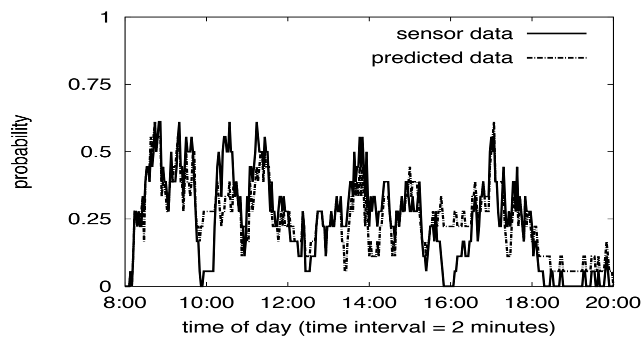


Figure 1. Office A: probability of presence.

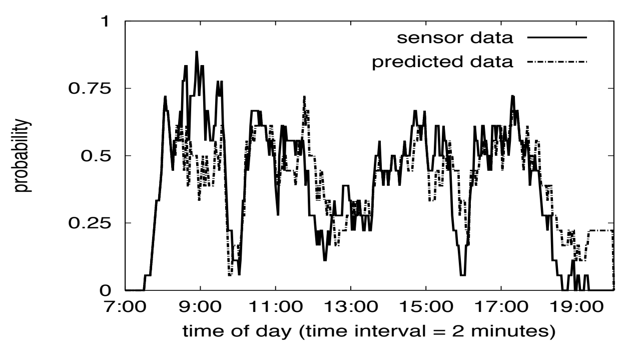


Figure 2. Office B: probability of presence.

A. First Arrival and Final Departure Times Analysis

Figures 6 to 10 give the probabilities of the first arrival and last departure at different time of the day. They show that all rules predicted the first arrival data correctly. This is understandable as the first arrival data have some distinguished features: the length of time that the occupant has been in the office (`accIn`) is 0 and the length of time the occupant was in the previous state (`noChange`) is a very

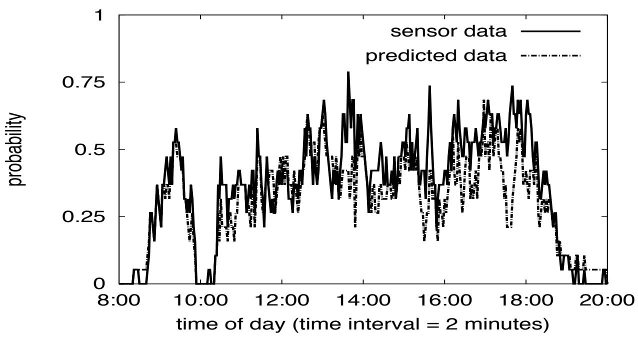


Figure 3. Office C: probability of presence.

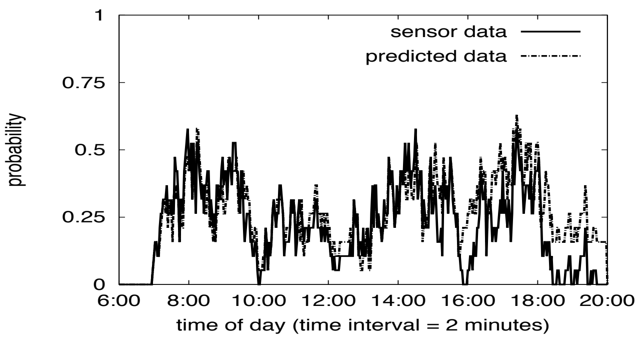


Figure 4. Office D: probability of presence.

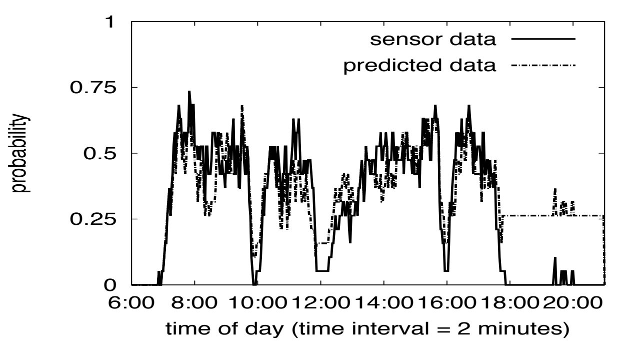


Figure 5. Office E: probability of presence.

large number. It is therefore not difficult for GP to detect the pattern and to learn the rules that made correct predictions.

The final departure data, however, do not have any unique pattern for GP to learn. Consequently, the rules failed to predict them correctly in some occasions. This is particularly evident in office B (4 errors), D (3 errors) and E (5 errors). When a rule fails to predict the final departure data correctly (predicted as 1 instead of 0), it implies that the occupant stayed in the office over-night. This is a serious problem as the building's system operations that are normally performed after the occupant has made the final departure will not be executed. We need to resolve this energy wasting issue in our future work. One possible solution is to change the office to a new state called "departed" after working hour. If the office is not empty, the occupant would trigger the sensor and the state will return to "occupied". This checking should be performed iteratively after working hour, so that the final departure of the occupant will not be missed.

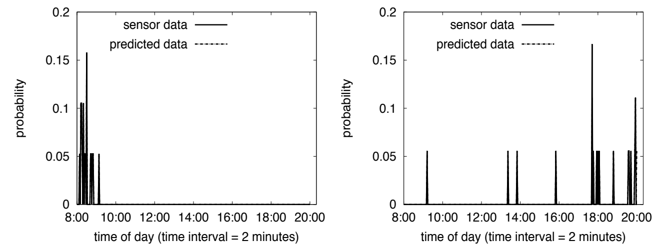


Figure 6. Office A: first arrival time (left) & last departure time (right).

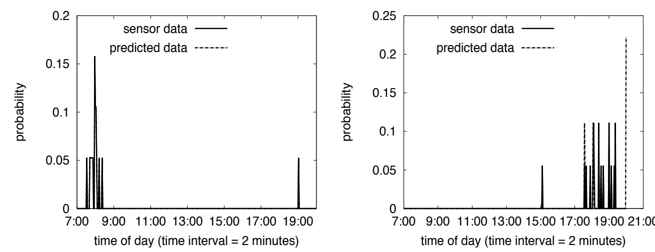


Figure 7. Office B: first arrival time (left) & last departure time (right).

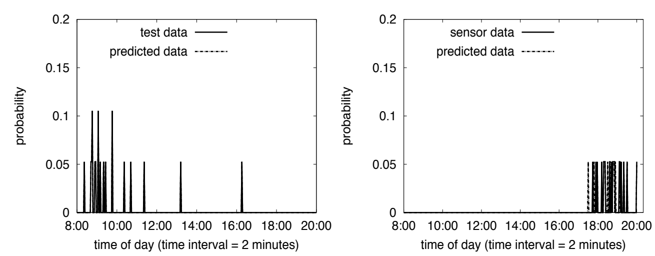


Figure 8. Office C: first arrival time (left) & last departure time (right).

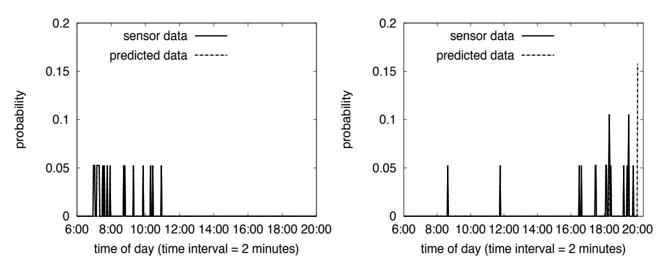


Figure 9. Office D: first arrival time (left) & last departure time (right).

B. Occupancy and Vacancy Intervals Distributions Analysis

We also analyzed the distribution of occupancy and vacancy intervals for the 5 offices and found that they have a similar pattern: both occupancy and vacancy intervals are exponentially distributed. In particular, there are many more

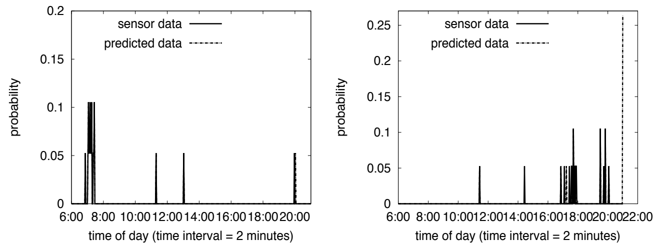


Figure 10. Office E: first arrival time (left) & last departure time (right).

short presence and absence intervals ($<$ 30 minutes) than longer ones. Due to space constraint, we only show the distributions for office A in Figure 11.

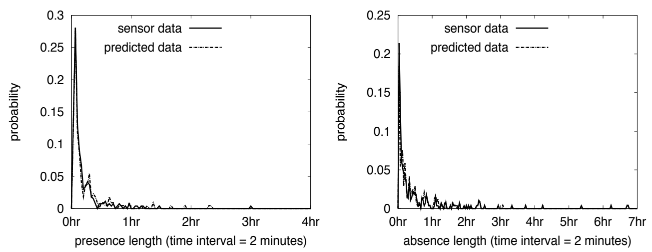


Figure 11. Office A: distribution of presence (left) & absence (right).

This result is different from that reported in [6] and [5]. We investigated what has caused the difference and found that the occupants in these offices made many more state changes each day (see Table VI) from that reported in [6] and [5]. In [6], they reported that the number of departure and arrival each occupant made was mostly five to six times a day, while in [5], the average number of state changes per day was around 10. Although we used the same data as that used in [5], we evaluated occupancy and vacancy intervals on 4-week of data only, while [5] used the data of the entire year. The high volume of state changes during the 4-week of testing period might caused by special events which required the occupants in these 5 offices to made more short in-and-out visits of their offices than they normally did.

Table VI
AVERAGE NUMBER OF STATE CHANGES PER DAY.

Office	A	B	C	D	E
Changes/day	18	17	30	27	26

VII. CONCLUSION

We have demonstrated the application of GP to learn occupancy behavioral rules that predict the presence and absence of an occupant in a single-person office. The rules give prediction accuracies of 80%–83% on testing data from 5 different offices. These occupancy rules can be used by a building simulation tool to estimate the energy demand of a building in many areas: L-HVAC systems, office appliances and the window open activities which modify the indoor ventilation. With the ability to estimate energy consumption more accurately, we are more likely to design energy saving for a building. These rules can also be used to operate a building by integrating it to the control of building systems, such as L-HVAC. With the ability to predict a building's occupancy more accurately, we are more likely to maintain an energy efficient building that also provide maximum comfort to its occupants.

Before expanding the work to learn occupancy behavior patterns year round, we will first verify the usefulness of the rules by applying them to control a HVAC system and running buildings simulation tools (e.g. Energy+) to estimate its energy saving capability.

We will also explore other type of data, not limited to sensors, that can assist in modeling occupancy behavior in more complicated situations, such as multi-occupancy offices and conference rooms. We will also investigate other learning/modeling approaches, e.g. co-learning, co-training, co-evolution of rules, to advance our understanding of building occupancy, which has been shown to be non-random. The ultimate goal of our research is to reduce energy consumption of a building while maintaining maximum comfort for its occupants.

ACKNOWLEDGMENT

I would like to thank Dr. Darren Robinson for providing me the motion sensor data to conduct this research.

REFERENCES

[1] B. Dong and B. Andrews. Sensor-based occupancy behavioral pattern recognition for energy and comfort management in intelligent buildings In *Proceedings of the 11th International IBPSA Conference*, pages 1444–1451, 2009.

[2] B. Dong, B. Andrews, K. P. Lam, M. Hoynck, R. Zhang, Y-S Chiou and D. Benitez An information technology enabled sustainability test-bed (ITEST) for occupancy detection through an environmental sensing network. *Energy and Buildings* volume 42, pages 1038-1046, Elsevier, 2010.

[3] D. Goldstein, *Invisible Energy: Strategies to Rescue the Economy and Save the Planet*, Bay Tree Publishing, 2010.

[4] J. Koza, *Genetic Programming: On the Programming of Computers by Means of Natural Selection*, MIT Press, 1992.

[5] J. Page, D, Robinson, N. Morel and J.-L. Scartezzini A generalised stochastic model for the simulation of occupant presence *Energy and Buildings*, volume 40, pages 83-98, Elsevier, 2008.

[6] D. Wang, C. C. Federspiel and F. Rubinstein, Modeling occupancy in single person offices. *Energy and Buildings*, volume 37, pages 121-126, Elsevier, 2005.

[7] T. Yu and C. Clack. PolyGP: A Polymorphic Genetic Programming System in Haskell. In *Genetic Programming 1998: Proceedings of the Third Annual Conference*, pages 416–421. Morgan Kaufmann, 1998.