Smart Homes

Diane J. Cook and Michael Youngblood
Department of Computer Science and Engineering
University of Texas at Arlington
{cook, youngbld}@cse.uta.edu

We live in an increasingly connected and automated society. Smart environments embody this trend by linking computers to everyday tasks and settings. Important features of such environments are that they possess a degree of autonomy, adapt themselves to changing conditions, and communicate with humans in a natural way. These systems can be found in offices, airports, hospitals, classrooms, or any other environment. This article discusses automation of our most personal environment: the home.

There are several characteristics that are commonly found in smart homes. This type of environment assumes controls and coordinates a network of sensors and devices, relieving the inhabitants of this burden. Interaction with smart homes is in a form that is comfortable to people: speech, gestures, and actions take the place of windows, icons, menus, and pointers.

We define a smart home as *one that is able to acquire and apply knowledge about its inhabitants and their surroundings in order to adapt to the inhabitants and meet the goals of comfort and efficiency*. Designing and implementing smart homes requires a unique breadth of knowledge not limited to a single discipline, but integrates aspects of machine learning, decision making, human-machine interfaces, wireless networking, mobile communications, databases, sensor networks, and pervasive computing. With these capabilities, the home can control many aspects of the environment such as climate, lighting, maintenance, and entertainment. Intelligent automation of these activities can reduce the amount of interaction required by inhabitants and reduce energy consumption and other potential operating costs. The same capabilities can be used to provide important features such as detection of unusual behaviors for health monitoring and home security.

Smart home operations can be characterized by the following scenario. At 6:45am, the home turns up the heat because it has learned that it needs 15 minutes to warm to the inhabitant's favorite waking temperature. The alarm sounds at 7:00, which signals the bedroom light to go on as well as the coffee maker in the kitchen. The inhabitant, Bob, steps into the bathroom and turns on the light. The home records this manual interaction, displays the morning news on the bathroom video screen, and turns on the shower. While Bob is shaving, the home senses that Bob is four pounds over his ideal weight and adjusts his suggested daily menu that will later be displayed in the kitchen. When Bob finishes grooming, the bathroom light turns off while the kitchen light and display turn on. During breakfast, Bob requests the janitor robot to clean the house. When Bob leaves for work, the home secures all doors behind him and starts the lawn sprinklers despite knowing the 30% predicted chance of rain. To reduce energy costs, the house turns down the heat until 15 minutes before Bob is due home. Because the refrigerator is low on milk and cheese, the home places a grocery order. When Bob arrives home, his grocery order has arrived, the house is back at Bob's desired temperature, and the hot tub is waiting for him.

Several smart home projects have been initiated in research labs. The Georgia Tech Aware Home (Kidd et. al 1999), the MIT Intelligent Room (Torrance 1995), and the Microsoft eHome focus on identifying user movements and activities with an impressive array of sensors. The MavHome Smart Home at the University of Texas at Arlington (Das et. al 2002) and the Neural Network House at the University of Colorado at Boulder (Mozer 1998) control aspects of the house such as lighting and temperature in response to inhabitant activities. The interest of industrial labs in intelligent environments is evidenced by the creation of Jini, Bluetooth, and SIP standards, and by supporting technologies such as Xerox PARC's Zombie Board, the Cisco Internet Home, and the Verizon Connected Family project. For an in-depth description of a smart home architecture, examine as a case study the MavHome smart home project.

Case Study: MavHome Smart Home

The MavHome smart home at the University of Texas at Arlington represents an environment that acts as an intelligent agent, perceiving the state of the home through sensors and acting upon the environment through device controllers. The agent's goal is to maximize comfort of its inhabitants while minimizing the cost of running the home. In order to achieve this goal, the house must be able to predict, reason about, and adapt to its inhabitants.

The desired smart home capabilities must be organized into a software architecture that seamlessly connects these components while allowing improvements to be made to any of the supporting technologies. Figure 1 shows the architecture of a MavHome agent. Technologies are separated into four cooperating layers. The *Decision* layer selects actions for the agent to execute. The *Information* layer collects information and generates inferences useful for decision making. The *Communication* layer routes information and requests between agents. The *Physical* layer contains the environment hardware including devices, transducers, and network equipment. The MavHome software components are connected using a distributed inter-process communication interface.

Because controlling an entire house is a very large and complex learning and reasoning problem, the problem is decomposed into reconfigurable subareas, or tasks. Thus the Physical layer for one agent may represent another agent somewhere in the hierarchy, which is capable of executing the task selected by the requesting agent.

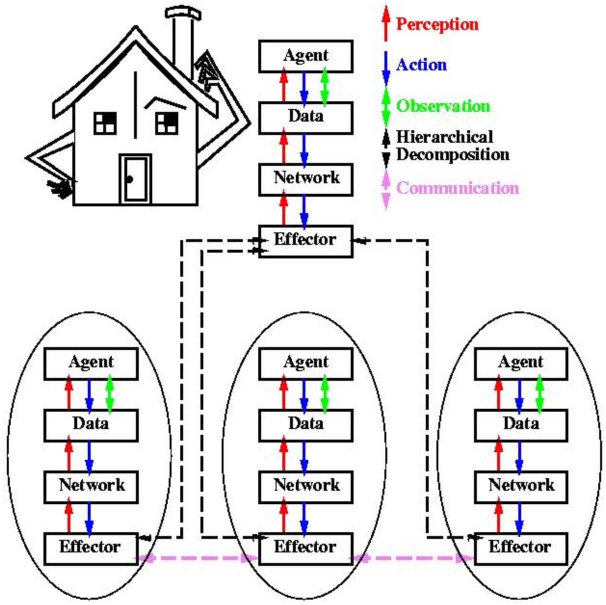

Figure 1. Mavhome agent architecture.

Perception is a bottom-up process. Sensors monitor the environment (e.g., lawn moisture level) and, if necessary, transmit the information to another agent through the Communication layer. The database records the information in the Information layer, updates its learned concepts and predictions accordingly, and alerts the Decision layer of the presence of new data. During action execution, information flows top down. The Decision layer selects an action (e.g., run the sprinklers) and relates the decision to the Information layer. After updating the database, the Communication layer routes the action to the appropriate effector to execute. If the effector is actually another agent, the agent receives the command through its effector as perceived information and must decide upon the best method of executing the desired action. Specialized interface agents allow interaction with users, robots, and external resources such as the Internet. Agents can communicate with each other using the hierarchical flow shown in Figure 1.

In order to maximize comfort, minimize cost, and adapt to inhabitants, a smart home must rely upon tools from artificial intelligence such as prediction and automated decision making.

Prediction is used to determine the inhabitant's next action as well as to predict behavior of devices in the home, such as how much time is needed to warm the house to a specified temperature, and how much energy will be utilized in doing this. Specifically, MavHome needs to predict the inhabitant's next action in order to automate selected repetitive tasks for the inhabitant. The home will need to make this prediction based solely on previously-seen inhabitant interaction with various devices and the current state of the inhabitant and the house. The number of prediction errors must be minimal, and the algorithms must be able to deliver predictions with minimal delays for computation. Prediction must then be handed over to a decision making algorithm that selects actions for the house to meet its goals.

Inhabitant Action Prediction. A smart home inhabitant typically interacts with various devices as part of his routine activities. These interactions may be considered as a sequence of events, with some inherent pattern of recurrence. This repeatability leads us to the conclusion that the sequence can be modeled as a stationary stochastic process. Inhabitant action prediction consists of first mining the data to identify sequences of actions that are regular and repeatable enough to generate predictions, and then using a sequence matching approach to predict the next action in one of these sequences.

To mine the data, a window can be moved in a single pass through the history of inhabitant actions, looking for sequences within the window that merit attention. Each sequence is evaluated using the Minimum Description Length principle, which favors sequences that minimize the description length of the sequence once it is compressed by replacing each instance of the discovered pattern with a pointer to the pattern definition. A regularity factor (daily, weekly, monthly) helps compress the data and thus increases the value of a pattern. Action sequences are first filtered by the mined sequences. If a sequence is considered significant by the mining algorithm, then predictions can be made for events within the sequence window. Using this algorithm as a filter for two alternative prediction algorithms, the resulting accuracy increases on average by 50%. This filter ensures that MavHome will not erroneously seek to automate anomalous and highly-variable activities.

The MavHome algorithm is based on the LZ78 text compression algorithm (Ziv and Lempel 1978). Well-investigated text compression methods have established that good compression algorithms are also good predictors. According to information theory, a predictor with an order (size of history used) that grows at a rate approximating the entropy rate of the source is an optimal predictor.

The prediction algorithm parses the input string (history of interactions) into substrings representing phrases. Because of the prefix property used by the algorithm, parsed substrings can be efficiently maintained in a trie along with frequency information. To perform prediction, the algorithm calculates the probability of each symbol (action) occurring in the parsed sequence, and predicts the action with the highest probability. To achieve optimal predictability, the predictor must use a mixture of all possible order models (phrase sizes) when determining the probability estimate. To accomplish this, techniques from the Prediction by Partial Match family of predictors are incorporated, that generate weighted Markov models of different orders. This blending strategy assigns greater weight to higher-order models, in keeping with the advisability of making the most informed decision.

In experiments run on sample smart home data, predictive accuracy of this approach converged on 100% for perfectly-repeatable data with no variation, and converged on 86% accuracy for data containing variations and anomalies.

Automated Decision Making. The goal of MavHome's decision making algorithm is to enable the home to automate basic functions in order to maximize the comfort of the inhabitants and minimize the cost of operating the home. For example, comfort can be measured by minimizing the number of manual interactions with the home, and operating cost can be measured as energy usage by the home.

Because the goal is a combination of these two factors, blind automation of all inhabitant actions is frequently not the desired solution. For example, an inhabitant might turn on the hallway light in the morning before opening the blinds in the living room. MavHome could, on the other hand, open the blinds in the living room before the inhabitant leaves the bedroom, thus alleviating the need for the hallway lights. Similarly, turning down the air conditioning after leaving the house and turning it back up before returning would be more energy efficient than turning the air conditioning to maximum after arriving home in order to cool it as quickly as possible.

To achieve its goal, MavHome uses reinforcement learning to acquire an optimal decision policy. In this framework, the agent learns autonomously from potentially-delayed rewards rather than from a teacher, reducing the requirement for the home's inhabitant to supervise or program the system. To learn a strategy, the agent explores the effects of its actions over time and uses this experience to form control policies which optimize the expected future reward.

MavHome learns a policy based on a state space, $S = \{s_i\}$, consisting of the states of the devices in the home, the predictions of the next event, and expected energy utilization over the next time unit. A reward function, r, takes into account the amount of required user interaction, the energy consumption of the house, and other parameters that quantify the performance of the home. This reward function can be tuned to the particular preferences of the inhabitants, thus providing a simple means to customize the home's performance. Q-learning is used (Watkins 1989) to approximate an optimal action strategy by estimating the predicted value, $Q(s_t, a_t)$, of executing action a_t in state s_t at time t. After each action, the utility is updated as $Q(s_t, a_t) \leftarrow \alpha[r_{t+1} + \gamma \max_{a \in A} Q(s_{t+1}, a) - Q(s_t, a_t)]$. After learning, the optimal action, a_t, can be determined as $a_t = \arg\max_{a \in A} Q(s_t, a)$.

MavHome Implementation. The MavHome smart home project is in place at the University of Texas at Arlington. Students register their presence in the environment using a fingerprint reader, and data is collected continuously based on their interactions with devices in the environment. Off-the-shelf X10 controllers automate most devices.

Using the ResiSim 3D simulator, a graphical model has been constructed of the intelligent environment. The model allows a visitor at a remote location to monitor or change the status of devices in MavHome, as shown in Figures 2 and 3. Images in the left column of Figure 2 show web cameras placed throughout the environment, and the simulator visualization is shown on the right. The "Information" window in the lower right indicates that devices have recently been manipulated, either manually or by MavHome. Figure 3 shows that the light in the entryway (upper left) is illuminated once Darin enters the environment and the lamp on Ryan's desk (lower left) turns on to assist him with work. The updated status of the lamp is shown by the yellow circle in the ResiSim model (right). The model will indicate the status of sensors as well – the orbs in Figure 4 indicate that there are two areas of activity captured by motion sensors.

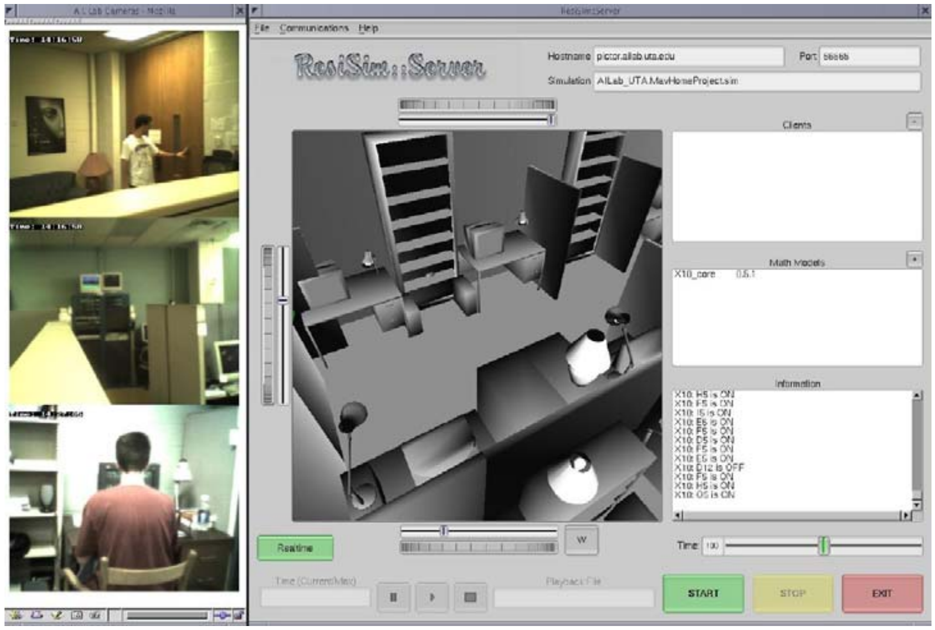

Figure 2. Web camera views of MavHome environment (left) and ResiSim visualization (right).

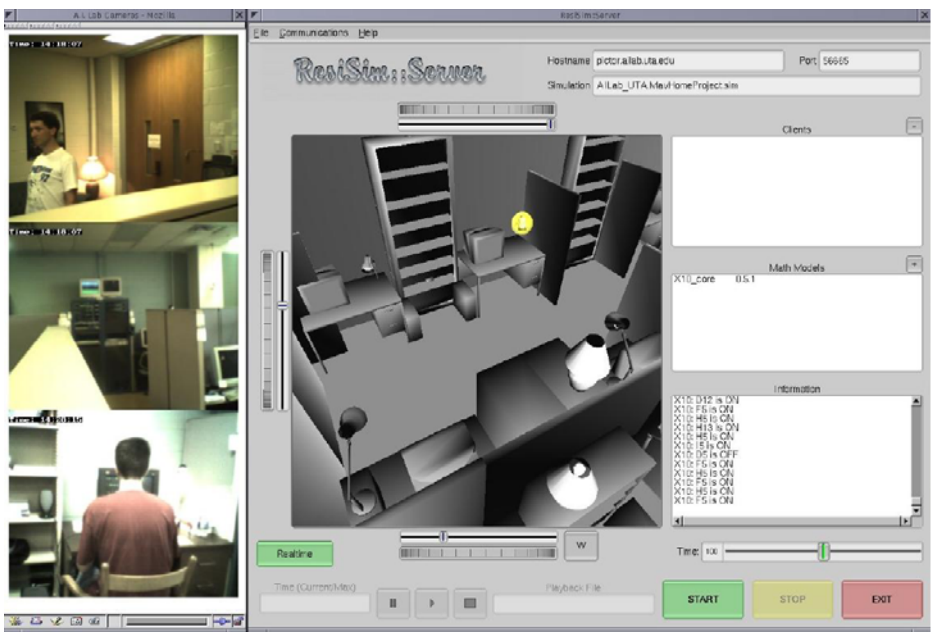

Figure 3. ResiSim update after desk lamp (lower left) is turned on.

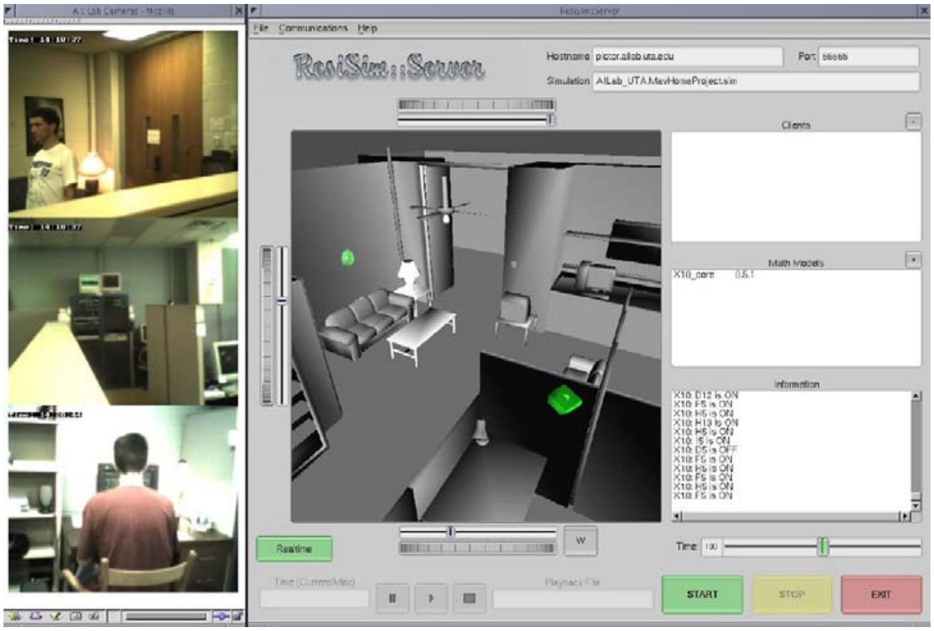

Figure 4. ResiSim indicates activated motion sensors with green orbs.

A live demonstration of MavHome was conducted in the fall of 2002. During the previous weeks, activity data was collected for one of the project participants ("MavHome Bob"). Actions included turning on lights en route to his desk in the morning, watching a live news feed on the computer, taking a coffee and TV break, and turning off devices on the way out at the end of the day. Despite the presence of approximately fifty people during the live demonstration (who were setting off motion sensors throughout the environment), MavHome correctly predicted and automated each activity. Figure 5 reflects the movements of MavHome Bob as he moves through the environment and lights are illuminated reflecting his typical activities.

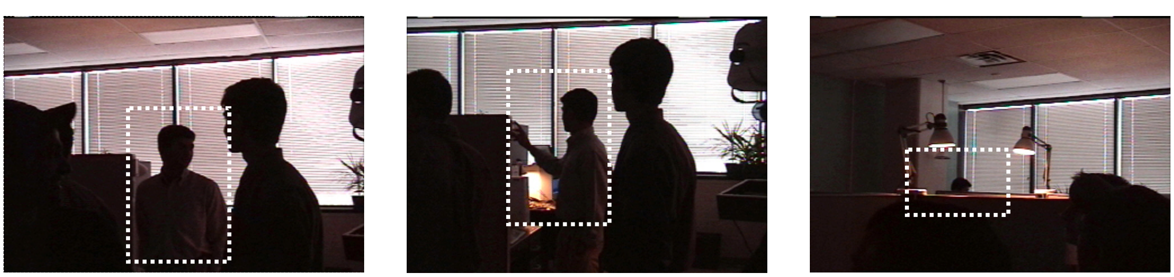

Figure 5. Bob's movements in MavHome. Bob's position is indicated by a dashed box.

Practical Considerations

So how easily can the features of a smart home be integrated into new or existing homes? The software described in the MavHome implementation consists of commercial X10 controllers, a computer, a variety of sensors, and a wireless network. A simple implementation can be integrated into an existing house for under a thousand dollars, in many cases. If robots or customized devices are introduce, the cost increases.

A computer interface to a smart home must be very simple. Manual control of devices can override home decisions, and alternative interfaces including voice control are offered. Other than starting or resetting the software, no interaction with the computer is required. In our experiments, the software adapted to user activities in a couple of weeks, but the training time will vary according to the complexity of user actions and the number of people in the home. Although minimal expertise is required, various types of interaction are possible depending on the needs of the user. The user can vary the certainly threshold at which activities are automated, although this is not necessary because manual resetting of actions selected by the house constitute negative reward and will eventually cause the house to not automate those particular commands. The user can also request that the home simply make suggestions for automation, and selection of rules for automation will be made on a case-by-case basis by the user.

Introducing intelligent control into a house can result in a number of privacy and safety issues. Safety constraints must be placed on each device to ensure that the house will not select an action which endangers inhabitants. The house may not be allowed, for example, to select a temperature setting below 50 degrees or above 90 degrees. The entire automation can be quickly disabled with one mouse click or voice command – each device can operate with or without computer control. The user also needs to specify the type of data that can be collected, and which data, if any, can be disseminated for learning across multiple households or cities.

Similarly, smart homes typically benefit from collecting information about the health, typical patterns, and other features of their inhabitants. This leads to a number of data privacy and security issues. Data should only be collected on features that are allowed by the inhabitants, and shared with other sites only as volunteered. New smart homes in neighboring locations could, for example, benefit from patterns learned in an older home, but care must be taken to share information in a way that does not violate the privacy of home inhabitants.

This article demonstrates the effectiveness of a smart home environment. These technologies will reduce the work to maintain a home, will lessen energy utilization, and provide special benefits for elderly and people with disabilities. In the future, these abilities can be generalized to other environments, including offices, hospitals, automobile, and airports.

Further Reading

Bobick, A., Intille, S., Davis, J., Baird, F., Pinhanez, C., Campbell, L., Ivanov, Y., Schutte, and Wilson, A. (2000). The KidsRoom: A perceptually-based interactive and immersive story environment. *Presence*, 8(4), 369-393.

Das, S., Cook, D. J., Bhattacharya, A., Heierman, III, E. O. and Lin, T.-Y. (2003). The role of prediction algorithms in the MavHome smart home architecture. *IEEE Wireless Communications*, 9(6), 77-84.

Demchak, G. (2000). Towards a post-industrial architecture: Design and construction of houses for the information age. Master’s thesis, Massachusetts Institute of Technology.

Fox, A., Johanson, B., Hanrahan, P., Winograd, T. (2000). Integrating information appliances into an interactive space. *IEEE Computer Graphics and Applications*, 20(3), 54-65.

Hedberg, S. After desktop computing: A progress report on smart environments research. *IEEE Intelligent Systems*, 15(5), 7-9.

Kidd, C., Orr, R. J., Abowd, G. D., Atkeson, D., Essa, I., MacIntyre, B., Mynatt, E., Starner, T. E., and Newstetters, W. (1999). The Aware Home: A living laboratory for ubiquitous computing. *Proceedings of the Second International Workshop on Cooperative Buildings*.

Lesser, V., Atighetchi, M., Benyo, B., Horling, B., Raja, A., Vincent, R., Wagner, T., Ping, X., Zhang, S. X. Q. (1999). The Intelligent Home testbed. *Proceedings of the Autonomy Control Software Workshop*.

Mozer, M. (1998). The Neural Network House: An environment that adapts to its inhabitants. *Proceedings of the AAAI Spring Symposium on Intelligent Environments*.

Torrance, M. C. (1995). Advances in human-computer interaction: The Intelligent Room. *Working Notes of the CHI 95 Research Symposium*.

Peace, S. and Holland, C. (eds.). 2001. *Inclusive Housing in an Ageing Society*. The Policy Press.

Russell, S. and Norvig, P. (2003). *Artificial intelligence: A modern approach*. Prentice Hall.

Watkins, C. J. C. H. (1989). Learning from delayed rewards. PhD thesis, Cambridge University.

Ziv, J. and Lempel, A. (1978). Compression of individual sequences via variable rate coding. *IEEE Transactions on Information Theory*, IT-24, 530-536.