Journal of Universal Computer Science, vol. 12, no. 1 (2006), 15-29
submitted: 6/5/05, accepted: 13/12/05, appeared: 28/1/06 © *J.UCS*

Health Monitoring and Assistance to Support Aging in Place

Diane J. Cook
(The University of Texas at Arlington, USA
cook@cse.uta.edu)

Abstract: To many people, home is a sanctuary. For those people who need special medical care, they may need to be pulled out of their home to meet their medical needs. As the population ages, the percentage of people in this group is increasing and the effects are expensive as well as unsatisfying. We hypothesize that many people with disabilities can lead independent lives in their own homes with the aid of at-home automated assistance and health monitoring. In order to accomplish this, robust methods must be developed to collect relevant data and process it dynamically and adaptively to detect and/or predict threatening long-term trends or immediate crises. The main objective of this paper is to investigate techniques for using agent-based smart home technologies to provide this at-home health monitoring and assistance. To this end, we have developed novel inhabitant modeling and automation algorithms that provide remote health monitoring for caregivers. Specifically, we address the following technological challenges: 1) identifying lifestyle trends, 2) detecting anomalies in current data, and 3) designing a reminder assistance system. Our solution approaches are being tested in simulation and with volunteers at the UTA's MavHome site, an agent-based smart home project.
Key Words: multiagent systems, artificial intelligence, smart environments
Category: I.2.6, I.2.11

1 Introduction and Motivation

Since the beginning, people have lived in places that provide shelter and basic comfort and support, but as society and technology advance there is a growing interest in improving the intelligence of the environments in which we live and work. The MavHome (**M**anaging an **a**daptive **v**ersatile **Home**) project is focused on providing such environments [Das et al., 2002]. We take the viewpoint of treating an environment as an intelligent agent, which perceives the state of the environment using sensors and acts upon the environment using device controllers in a way that can optimize a number of different goals including maximizing comfort of the inhabitants, minimizing the consumption of resources, and maintaining safety of the environment and its inhabitants. In this paper we discuss methods by which we can adapt a smart home environment such as MavHome to perform health monitoring and assistance for persons with disabilities and for aging adults.

As Lanspery and Hyde [Lanspery et al., 1997] state, "For most of us, the word 'home' evokes powerful emotions [and is] a refuge". They note that older adults and people with disabilities want to remain in their homes even when their conditions worsen and the home cannot sustain their safety. In a national survey, researchers found that 71% of the respondents felt strongly that they wanted to remain in their current residence as long as possible, and another 12% were somewhat likely to remain there [AARP, 2000]. Nearly 1/4 of the respondents expected that they or a member of their household would have problems getting around their house in the next five years. Of these respondents, 86% stated that they had made at least one modification to their home to make it easier to live there, and nearly 70% believe that the modifications will allow them to live in the current homes longer than would have otherwise been possible. A separate study supported these results and found that the most common modifications were an easy-to-use climate control system and a personal alert system.

Zola [Zola, 1997] maintains that the problems of aging and disability are converging. Improvements in medical care are resulting in increased survival into old age, thus problems of mobility, vision, hearing, and cognitive impairments will increase [Pynoos, 2002, Parr and Russell, 1997]. As the baby boomers enter old age, this trend will be magnified. By 2040, 23% will fall into the 65+ category [Lanspery et al., 1997]. An AARP report [AARP, 2000, AARP, 2003] strongly encourages increased funding for home modifications that can keep older adults with disabilities independent in their own homes.

While use of technology can be expensive, it may be more cost effective than the alternative [Grayons, 1997]. Nursing home care is generally paid either out-of-pocket or by Medicaid. Typical nursing home costs are about $40,000 a year, and the $197 billion of free care offered by family members comes at the sacrifice of independence and job opportunities by the family caregivers.

In this paper, our goal is to assist the elderly and individuals with disabilities by providing home capabilities that will monitor health trends and assist in the inhabitant's day to day activities in their own homes. The result will save money for the individuals, their families, and the state.

2 Overview of the MavHome Smart Home

We define an intelligent environment 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 [Cook and Das, 2004]. These capabilities rely upon effective prediction, decision making, robotics, wireless and sensor networking, mobile computing, databases, and multimedia technologies. With these capabilities, the home can adaptively control many aspects of the environment such as climate, water, lighting, maintenance, and multimedia entertainment. Intelligent automation of these activities can reduce the amount of interaction required by inhabitants, reduce energy consumption and other potential wastages, and provide a mechanism for ensuring the health and safety of the environment occupants [Das and Cook, 2004b].

As the need for automating these personal environments grows, so does the number of researchers investigating this topic. Some design interactive conference rooms, offices, kiosks, and furniture with seamless integration between heterogeneous devices and multiple user applications in order to facilitate collaborate work environments [AIRE Group, 2004, Fox et al., 2000, Romn et al., 2002, Streitz et al., 1999]. Abowd and Mynatt's work [Abowd and Mynatt, 2005] focuses on ease of interaction with a smart space, and work such as the Gator Tech Smart House [Helal et al., 2005] focuses on application of smart environments to elder care.

Mozer's Adaptive Home [Mozer, 2005] uses neural network and reinforcement learning to control lighting, HVAC, and water temperature to reduce operating cost. In contrast, the approach taken by the iDorm project [Hagras et al., 2004] is to use a fuzzy expert system to learn rules that replicate inhabitant interactions with devices, but will not find an alternative control strategy that improves upon manual control for considerations such as energy expenditure.

These projects have laid a foundation for our work. However, unlike related projects, we learn a decision policy to control an environment in a way that optimizes a variety of possible criteria, including minimizing manual interactions, improving operating efficiency, and ensuring inhabitant health and safety. We also ensure that our software need not be redesigned as new devices are registered, new spaces are tested, or new inhabitants move into the environment. To accomplish this goal, our intelligent environment must harness the features of multiple heterogeneous learning algorithms in order to identify repeatable behaviors, predict inhabitant activity, and learn a control strategy for a large, complex environment.

The MavHome architecture shown in Figure 1 consists of cooperating layers [Cook and Das, 2004, Das and Cook, 2005]. Perception is a bottom-up process. Sensors monitor the environment using physical components (e.g., sensors) and make information available through the interface layers. The database stores this information while other information components process the raw information into more useful knowledge (e.g., patterns, predictions). New information is presented to the decision making applications (top layer) upon request or by prior arrangement. Action execution flows top-down. The decision action is communicated to the services layer which records the action and communicates it to the physical components. The physical layer performs the action using powerline control, and other automated hardware, thus changing the state of the world and triggering a new perception.

All of the MavHome components are implemented and are being tested in two physical environments, the MavLab workplace environment and an on-campus

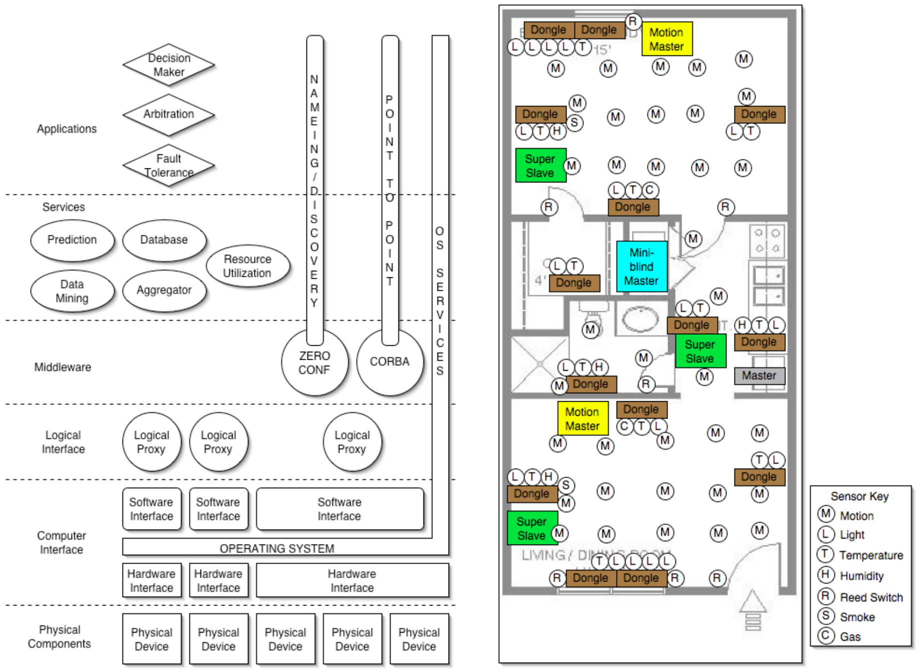

Figure 1: MavHome architecture (left) and MavPad sensor layout (right).

apartment. Powerline control automates all lights and appliances, as well as HVAC, fans, and miniblinds. Perception of light, humidity, temperature, smoke, gas, motion, and switch settings is performed through a sensor network developed in-house. Inhabitant localization is performed using passive infrared sensors yielding a detection rate of 95% accuracy [Youngblood et al., 2005a].

Communication between high-level components is performed using CORBA, and each component registers its presence using zero configuration (ZeroConf) technologies. Implemented services include a PostgreSQL database that stores sensor readings, prediction components, data mining components, and logical proxy aggregators. Resource utilization services monitor current utility consumption rates and provide usage estimates and consumption queries.

MavHome is designed to optimize a number of alternative functions, but for this evaluation we focus on minimization of manual interactions with devices. The MavHome components are fully implemented and are automating the environments shown in Figure 2 [Youngblood et al., 2005b]. The MavLab environment contains work areas, cubicles, a break area, a lounge, and a conference room. MavLab is automated using 54 X-10 controllers and the current state is determined using light, temperature, humidity, motion, and door/seat

Figure 2: The MavLab (left) and MavPad (right) environments.

status sensors. The MavPad is an on-campus apartment hosting a full-time student occupant. MavPad is automated using 25 controllers and provides sensing for light, temperature, humidity, leak detection, vent position, smoke detection, CO detection, motion, and door/window/seat status sensors. Figure 1 shows the MavPad sensor layout.

3 Core Technologies

To automate our smart environment, we collect observations of manual inhabitant activities and interactions with the environment. We then mine sequential patterns from this data using a sequence mining algorithm. Next, we predict the inhabitant's upcoming actions using observed historical data. Finally, a hierarchical Markov model is created using low-level state information and high-level sequential patterns, and is used to learn an action policy for the environment. Figure 3 shows how these components work together to improve the overall performance of the smart environment. Here we describe the learning algorithms that play a role in this approach.

3.1 Mining Sequential Patterns Using ED

In order to minimize resource usage, maximize comfort, and adapt to inhabitants, we rely upon machine learning techniques for automated discovery, prediction, and decision making. 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. Agrawal and Srikant [Agrawal and Srikant, 1995] pioneered work in mining sequential patterns from time-ordered transactions, and our work is loosely modeled on this approach.

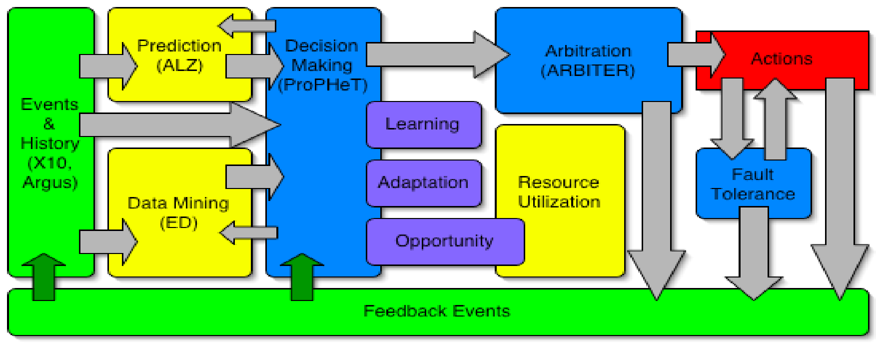

Figure 3: Integration of AI techniques into MavHome architecture.

Typically, each inhabitant-home interaction event is characterized as a triple consisting of the device manipulated, the resulting change that occurred in that device, and the time of interaction. We move a window in a single pass through the history of events or inhabitant actions, looking for episodes (sequences) within the window that merit attention. Candidate episodes are collected within the window together with frequency information for each candidate. Candidate episodes are evaluated and the episodes with values above a minimum acceptable compression amount are reported. The window size can be selected automatically using the size that achieves the best compression performance over a sample of the input data.

When evaluating candidate episodes, the Episode Discovery (ED) algorithm [Heierman and Cook, 2003] looks for patterns that minimize the description length of the input stream, O, using the Minimum Description Length (MDL) principle [Rissanen, 1989]. The MDL principle targets patterns that can be used to minimize the description length of a database by replacing each instance of the pattern with a pointer to the pattern definition.

Our MDL-based evaluation measure thus identifies patterns that balance frequency and length. Periodicity (daily, every other day, weekly occurrence) of episodes is detected using autocorrelation and included in the episode description. If the instances of a pattern are highly periodic (occur at predictable intervals), the exact timings do not need to be encoded (just the pattern definition with periodicity information) and the resulting pattern yields even greater compression. Although event sequences with minor deviations from the pattern definition can be included as pattern instances, the deviations need to be encoded and the result thus increases the overall description length. ED reports the patterns and encodings that yield the greatest MDL value.

Deviations from the pattern definition in terms of missing events, extra events, or changes in the regularity of the occurrence add to the description length because extra bits must be used to encode the change, thus lowering the value of the pattern. The larger the potential amount of description length compression a pattern provides, the more representative the pattern is of the history as a whole, and thus the potential impact that results from automating the pattern is greater.

In this way, ED identifies patterns of events that can be used to better understand the nature of inhabitant activity in the environment. Once the data is compressed using discovered results, ED can be run again to find an abstraction hierarchy of patterns within the event data. As the following sections show, the results can also be used to enhance performance of predictors and decision makers that automate the environment.

3.2 Predicting Activities Using ALZ

To predict inhabitant activities, we borrow ideas from text compression, in this case the LZ78 compression algorithm [Ziv and Lempel, 1978]. By predicting inhabitant actions, the home can automate or improve upon anticipated events that inhabitants would normally perform in the home. Well-investigated text compression methods have established that good compression algorithms also make 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. Other approaches to prediction or inferring activities often use a fixed context size to build the model or focus on one attribute such as motion [Cielniak et al., 2003, Philipose et al., 2004].

LZ78 incrementally processes an input string of characters, which in our case is a string representing the history of device interactions, and stores them in a trie. The algorithm parses the string $x_1, x_2, \ldots, x_i$ into substrings $w_1, w_2, w_{c(i)}$ such that for all $j > 0$, the prefix of the substring w_j is equal to some w_i for $1 < i < j$. Thus when parsing the sequence of symbols *aaababbbbbaabccddcbaaaa*, the substring *a* is created, followed by *aa*, *b*, *ab*, *bb*, *bba*, and so forth.

Our Active LeZi (ALZ) algorithm enhances the LZ78 algorithm by recapturing information lost across phrase boundaries. Frequency of symbols is stored along with phrase information in a trie, and information from multiple context sizes are combined to provide the probability for each potential symbol, or inhabitant action, as being the next one to occur. In effect, ALZ gradually changes the order of the corresponding model that is used to predict the next symbol in the sequence. As a result, we gain a better convergence rate to optimal predictability as well as achieve greater predictive accuracy. Figure 4 shows the trie formed by the Active-LeZi parsing of the input sequence *aaababbbbbaabccddcbaaaa*.

To perform prediction, ALZ calculates the probability of each symbol (inhabitant action) occurring in the parsed sequence, and predicts the action with

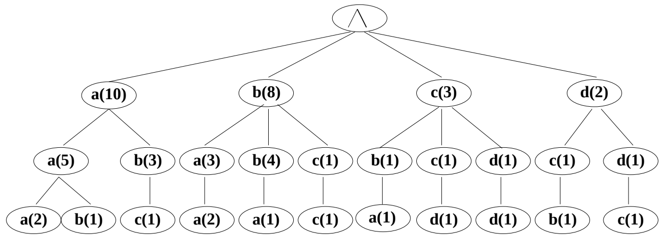

Figure 4: Trie formed by ALZ parsing.

the highest probability. To achieve optimal predictability, we use a mixture of all possible higher-order models (phrase sizes) when determining the probability estimate. Specifically, we incorporate the Prediction by Partial Match strategy of *exclusion* [Bell et al., 1990] to gather information from all available context sizes in assigning the next symbol its probability value.

We initially evaluated the ability of ALZ to perform inhabitant action prediction on synthetic data based on six embedded tasks with 20% noise. In this case the predictive accuracy converges to 86%. Real data collected based on six students in the MavLab for one month was much more chaotic, and on this data ALZ reached a predictive performance of 30% (although it outperformed other methods). However, when we combine ALZ and ED by only performing predictions when the current activity is part of a sequential pattern identified by ED, ALZ performance increases by 14% [Gopalratnam and Cook, 2004, Gopalratnam and Cook, 2005].

3.3 Decision Making Using ProPHeT

In our final learning step, we employ reinforcement learning to generate an automation strategy for the intelligent environment. To apply reinforcement learning, the underlying system (i.e., the house and its inhabitants) could be modeled as a Markov Decision Process (MDP). This can be described by a four-tuple $< S, A, Pr, R >$, where S is a set of system states, A is the set of available actions, and $R : S \rightarrow R$ is the reward that the learning agent receives for being in a given state. The behavior of the MDP is described by the transition function, $Pr : S \times A \times S \rightarrow [0, 1]$, representing the probability with which action a_t executed in state s_t leads to state s_{t+1}.

With the increasing complexity of tasks being addressed, recent work in decision making under uncertainty has popularized the use of Partially Observable Markov Decision Processes (POMDPs). Recently, there have been many published hierarchical extensions that allow for the partitioning of large domains into a tree of manageable POMDPs [Pineau et al., 2001, Theocharous et al., 2001].

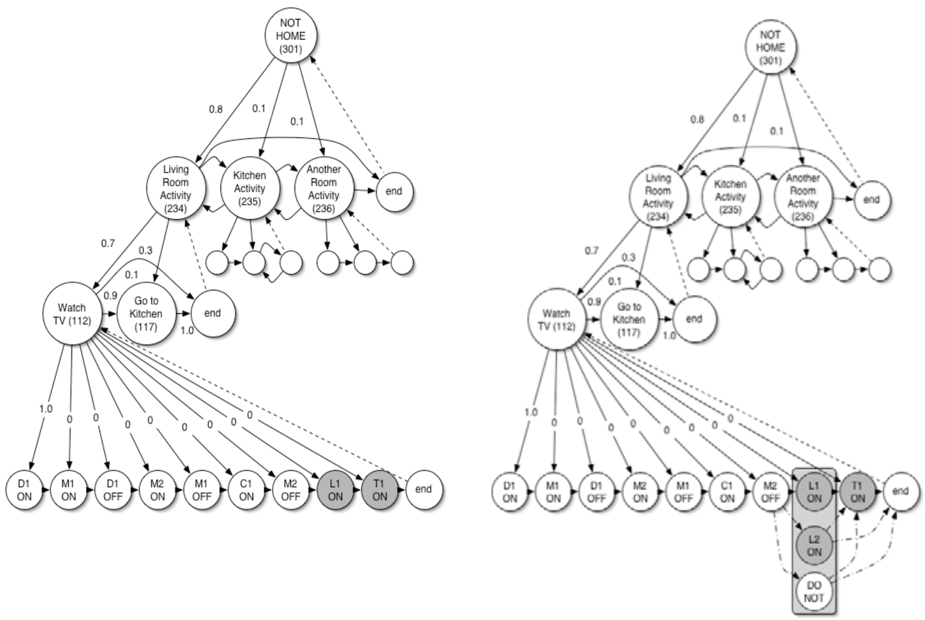

Figure 5: Hierarchical model constructed from static (left) and dynamic (right) smart home data.

Research has shown that strategies for new tasks can be learned faster if policies for subtasks are already available [Precup and Sutton, 1997]. Although a Hierarchical POMDP (HPOMDP) is appropriate for an intelligent environment domain, current approaches generally require *a priori* construction of the hierarchical model. Unlike other approaches to creating a hierarchical model, our decision learner, ProPHeT, actually automates model creation by using the ED-mined sequences to represent the nodes in the higher levels of the model hierarchy.

The lowest-level nodes in our model represent a single event observed by ED. Next, ED is run multiple iterations on this data until no more patterns can be identified, and the corresponding abstract patterns comprise the higher-level nodes in the Markov model. The higher-level *task* nodes point to the first event node for each permutation of the sequence that is found in the environment history. Vertical transition values are labeled with the fraction of occurrences for the corresponding pattern permutation, and horizontal transitions are seeded using the relative frequency of transitions from one event to the next in the observed history. As a result, the n-tier hierarchical model is thus learned from collected data. An example hierarchical model constructed from MavHome test data is shown on the left in Figure 5.

Given the current event state and recent history, ED supplies membership probabilities of the state in each of the identified patterns. Using this information along with the ALZ-predicted next action, ProPHeT maintains a belief state and selects the highest-utility action.

To learn an automation strategy, the agent explores the effects of its decisions over time and uses this experience within a temporal-difference reinforcement learning framework [Sutton and Barto, 1998] to form control policies which optimize the expected future reward. The current version of MavHome receives negative reinforcement (observes a negative reward) when the inhabitant immediately reverses an automation decision (e.g., turns the light back off) or an automation decision contradicts Arbiter-supplied safety and comfort constraints.

Before an action is executed it is checked against the policies in the policy engine, Arbiter. These policies contain designed safety and security knowledge and inhabitant standing rules. Through the policy engine the system is prevented from engaging in erroneous actions that may perform actions such as turning the heater to 120^{o}F or from violating the inhabitant's stated wishes (e.g., a standing rule to never turn off the inhabitant's night light).

4 Initial Case Study

As an illustration of the above techniques, we have evaluated a week in an inhabitant's life with the goal of reducing the manual interactions in the MavLab. The data was generated from a virtual inhabitant based on captured data from the MavLab and was restricted to just motion and lighting interactions which account for an average of 1400 events per day.

ALZ processed the data and converged to 99.99% accuracy after 10 iterations through the training data. When automation decisions were made using ALZ alone, interactions were reduced by 9.7% on average. Next, ED processed the data and found three episodes to use as abstract nodes in the HPOMDP. Living room patterns consisted of lab entry and exit patterns with light interactions, and the office also reflected entry and exit patterns. The other patterns occurred over the remaining 8 areas and usually involved light interactions at desks and some equipment upkeep activity patterns. The hierarchical Markov model with no abstract nodes reduced interactions by 38.3%, and the combined-learning system (ProPHeT bootstrapped using ED and ALZ) was able to reduce interactions by 76%, as shown in Figure 6 (left).

Experimentation in the MavPad using real inhabitant data has yielded similar results. In this case, ALZ alone reduced interactions from 18 to 17 events, the HPOMDP with no abstract nodes reduced interactions by 33.3% to 12 events, while the bootstrapped HPOMDP reduced interactions by 72.2% to 5 events. These results are graphed in Figure 6 (right).

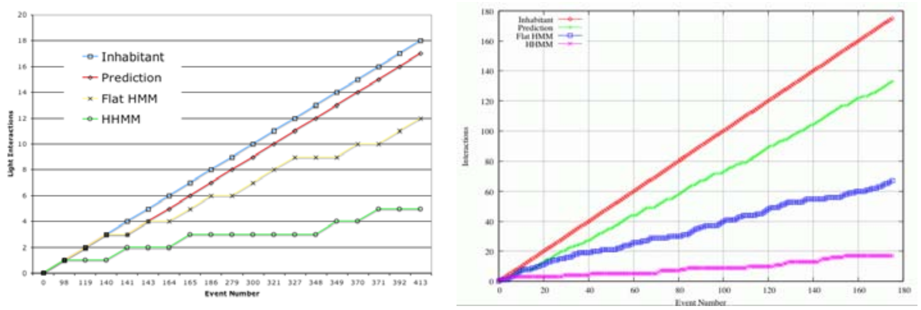

Figure 6: Interaction reduction.

5 Using a Smart Home to Assist Elderly and Disabled

The data mining, prediction, and multiagent technologies available in MavHome can be employed to provide health care assistance in living environments. Specifically, models can be constructed of inhabitant activities and used to learn activity trends, detect anomalies, intelligently predict possible problems and make health care decisions, and provide automation assistance for inhabitants with special needs.

A variety of approaches have been investigated in recent years to automate caregiver services. Many of the efforts offer supporting technologies in specialized areas, such as using computer vision techniques to track inhabitants through the environment and specialized sensors to detect falls or other crises. Some special-purpose prediction algorithms have been implemented using factors such as measurement of stand-sit and sit-stand transitions and medical history [Cameron et al., 1997, Najafi et al., 2002, Najafi et al., 2003], but are limited in terms of what they predict and how they use the results. Remote monitoring systems have been designed with the common motivation that learning and predicting inhabitant activities is key for health monitoring, but very little work has combined the remote monitoring capabilities with prediction for the purpose of health monitoring. Some work has also progressed toward using typical behavior patterns to provide reminders, particularly useful for the elderly and patients suffering from various types of dementia [Kautz et al., 2002, Pollack et al., 2003].

Our smart environment can identify patterns indicating or predicting a change in health status and can provide inhabitants with needed automation assistance. Collected data includes movement patterns of the individual, periodic vital signs (blood pressure, pulse, body temperature), water and device usage, use of food items in the kitchen, exercise regimen, medicine intake (prescribed and actual), and sleep patterns [Das and Cook, 2004a, Das and Cook, 2004b]. Given this data, models can be constructed of inhabitant activities and use to learn lifestyle trends, detect anomalies, and provide reminder and automation assistance.

5.1 Capability 1: Identify lifestyle trends

Our ED algorithm is designed to process data as it arrives. Because of this feature, trends in the data including increasing / decreasing pattern frequency, introduction of patterns, and change in pattern details can be automatically detected [Heierman, 2004]. When changing patterns include health-specific events (vital signs, medication intake, or events targeted by the caregiver), a report will be given to the inhabitant and caregiver of these trends.

5.2 Capability 2: Detect anomalies in current data

The ED data mining algorithm and ALZ predictor can work together to detect anomalies in event data. ED identifies the most significant and frequent patterns of inhabitant behavior, as well as the likelihood that the current state is a member of one of these patterns. Whenever the current state falls within one of these patterns, ALZ can determine the probability distribution of next events. As a result, when the next event has a low probability of occurrence, or when the expected next event does not occur at the expected time, the result is considered an anomaly.

When an anomaly occurs, the home will first try to contact the inhabitant (through the interactive display for a lesser critical anomaly, or through the sound system for a more critical anomaly). If the inhabitant does not respond and the criticality of the anomaly is high, the caregiver will be notified.

5.3 Capability 3: Design reminder assistance system

Reminders can be triggered by two situations. First, if the inhabitant queries the home for his next routine activity, the activity with the highest probability will be given based on the ALZ prediction. Second, if a critical anomaly is detected, the environment will initiate contact with the inhabitant and remind him of the next typical activity. Such a reminder service will be particularly beneficial for individuals suffering from dementia.

As described in the initial MavHome design, automation assistance is always available for inhabitants, which is beneficial if some activities are difficult to perform. A useful feature of the architecture is that safety constraints are embedded in the Arbiter rule engine. If the inhabitant or the environment is about to conflict with these constraints, a preventative action is taken and the inhabitant notified. This can prevent accidents such as forgetting to turn off the water in the bathtub or leaving the house with doors unlocked.

6 Conclusion

The MavHome software architecture has successfully monitored and provided automation assistance for volunteers living in the MavPad site. We are currently collecting health-specific data in the MavHome sites and will be testing in the living environments of recruited residents at the C.C. Young Retirement Community in Dallas, Texas.

Acknowledgements

This work is supported by US National Science Foundation under ITR grant IIS–0121297.

References

[AARP, 2000] AARP (2000). Fixing to stay: A national survey of housing and home modification issues.

[AARP, 2003] AARP (2003). These four walls... Americans 45+ talk about home and community.

[Abowd and Mynatt, 2005] Abowd, G. D. and Mynatt, E. D. (2005). Designing for the human experience in smart environments. In Cook, D. J. and Das, S. K., editors, *Smart Environments: Technology, Protocols, and Applications*, pages 153–174. Wiley.

[Agrawal and Srikant, 1995] Agrawal, R. and Srikant, R. (1995). Mining Sequential Patterns. In *Proceedings of the 11th International Conference on Data Engineering*, pages 3–14.

[AIRE Group, 2004] AIRE Group (2004). MIT Project AIRE – About Us. http://www.ai.mit.edu/projects/aire.

[Bell et al., 1990] Bell, T. C., Cleary, J. G., and Witten, I. H. (1990). *Text compression*. Prentice Hall.

[Cameron et al., 1997] Cameron, K., Hughes, K., and Doughty, K. (1997). Reducing fall incidence in community elders by telecare using predictive systems. In *Proceedings of the International IEEE-EMBS Conference*, pages 1036–1039.

[Cielniak et al., 2003] Cielniak, G., Bennewitz, M., and Burgard, W. (2003). Where is ...? learning and utilizing motion patterns of persons with mobile robots. In *Proceedings of the Eighteenth International Joint Conference on Artificial Intelligence*, pages 909–914.

[Cook and Das, 2004] Cook, D. J. and Das, S. K. (2004). Mavhome: Work in progress. *IEEE Pervasive Computing*.

[Das and Cook, 2004a] Das, S. K. and Cook, D. J. (2004a). Health monitoring in an agent-based smart home. In *Proceedings of the International Conference on Smart Homes and Health Telematics (ICOST)*.

[Das and Cook, 2004b] Das, S. K. and Cook, D. J. (2004b). Health monitoring in an agent-based smart home by activity predition. In Zhang, D. and Mokhari, M., editors, *Toward a Human-Friendly Assistive Environment*, pages 3–14. IOS Press.

[Das and Cook, 2005] Das, S. K. and Cook, D. J. (2005). Smart home environments: A paradigm based on learning and prediction. In *Wireless Mobile and Sensor Networks*. Wiley.

[Das et al., 2002] Das, S. K., Cook, D. J., Bhattacharya, A., Heierman, E. O., and Lin, T.-Y. (2002). The role of prediction algorithms in the mavhome smart home architecture. *IEEE Wireless Communications*, 9(6):77–84.

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

[Gopalratnam and Cook, 2004] Gopalratnam, K. and Cook, D. J. (2004). Active LeZi: an incremental parsing algorithm for sequential prediction. *International Journal of Artificial Intelligence Tools*, 14(1-2):917–930.

[Gopalratnam and Cook, 2005] Gopalratnam, K. and Cook, D. J. (2005). Online sequential prediction via incremental parsing: The Active LeZi algorithm. *IEEE Intelligent Systems*.

[Grayons, 1997] Grayons, P. J. (1997). Technology and home adaptation. In Lanspery, S. and Hyde, J., editors, *Staying Put: Adapting the Places Instead of the People*, pages 55–74. Baywood Publishing Company.

[Hagras et al., 2004] Hagras, H., Callaghan, V., Colley, M., Clarke, G., Pounds-Cornish, A., and Duman, H. (2004). Creating an ambient-intelligence environment using embedded agents. *IEEE Intelligent Systems*, 19(6).

[Heierman, 2004] Heierman, E. O. (2004). *Using information-theoretic principles to discover interesting episodes in a time-ordered sequence*. PhD thesis, The University of Texas at Arlington.

[Heierman and Cook, 2003] Heierman, E. O. and Cook, D. J. (2003). Improving home automation by discovering regularly occurring device usage patterns. In *Proceedings of the International Conference on Data Mining*.

[Helal et al., 2005] Helal, A., Mann, W., El-Zabadani, H., King, J., Kaddoura, Y., and Jansen, E. (2005). The gator tech smart house: A programmable pervasive space. *IEEE Computer*, 38(3):50–60.

[Kautz et al., 2002] Kautz, H., Arnstein, L., Borriello, G., Etzioni, O., and Fox, D. (2002). An overview of the assisted cognition project. In *Proceedings of the AAAI workshop on automation as caregiver*.

[Lanspery et al., 1997] Lanspery, S., Jr, J. J. C., Miller, J. R., and Hyde, J. (1997). Introduction: Staying put. In Lanspery, S. and Hyde, J., editors, *Staying Put: Adapting the Places Instead of the People*, pages 1–22. Baywood Publishing Company.

[Mozer, 2005] Mozer, M. C. (2005). Lessons from an adaptive home. In Cook, D. J. and Das, S. K., editors, *Smart Environments: Technology, Protocols, and Applications*, pages 273–298. Wiley.

[Najafi et al., 2002] Najafi, B., Aminian, K., Loew, F., Blanc, Y., and Robert, P. (2002). Measurement of stand-sit and sit-stand transitions using a miniature gyroscope and its application in fall risk evaluation in the elderly. *IEEE Transactions on Biomedical Engineering*, 49(8):843–851.

[Najafi et al., 2003] Najafi, B., Aminian, K., Paraschiv-Ionescu, A., Loew, F., Bula, C., and Robert, P. (2003). Ambulatory system for human motion analysis using a kinematic sensor: Monitoring of daily physical activity in the elderly. *IEEE Transactions on Biomedical Engineering*, 50(6):711–723.

[Parr and Russell, 1997] Parr, R. and Russell, S. (1997). Reinforcement learning with hierarchies of machines. In *Advances in Neural Information Processing Systems 10*, Denver, CO. MIT Press.

[Philipose et al., 2004] Philipose, M., Fishkin, K., Perkowitz, M., Patterson, D., Fox, D., Kautz, H., and Hahnel, D. (2004). Inferring activities from interactions with objects. *Pervasive Computing*, 3(4):50–56.

[Pineau et al., 2001] Pineau, J., Roy, N., and Thrun, S. (2001). A Hierarchical Approach to POMDP Planning and Execution. Workshop on Hierarchy and Memory in Reinforcement Learning (ICML).

[Pollack et al., 2003] Pollack, M. E., Brown, L., Colbry, D., McCarthy, C. E., Orosz, C., Peintner, B., Ramakrishnan, S., and Tsamardinos, I. (2003). Autoreminder: An intelligent cognitive orthotic system for people with memory impairment. *Robotics and Autonomous Systems*, 44:273–282.

[Precup and Sutton, 1997] Precup, D. and Sutton, R. S. (1997). Multi-time models for temporally abstract planning. *Advances in Neural Information Processing Systems*, 10:1050–1056.

[Pynoos, 2002] Pynoos, J. (2002). Neglected areas in gerontology: Housing adaptation. In *Panel presentation at the Annual Scientific Meeting of the Gerontological Society of America*.

[Rissanen, 1989] Rissanen, J. (1989). *Stochastic Complexity in Statistical inquiry*. World Scientific Publishing Company.

[Romn et al., 2002] Romn, M., Hess, C. K., Cerqueira, R., Ranganathan, A., Campbell, R. H., and Nahrstedt, K. (2002). Gaia: A middleware infrastructure to enable active spaces. *IEEE Pervasive Computing*, pages 74–83.

[Streitz et al., 1999] Streitz, N. A., Geiler, J., Holmer, T., Konomi, S., Mller-Tomfelde, C., Reischl, W., Rexroth, P., Seitz, P., and Steinmetz, R. (1999). i-LAND: an interactive landscape for creativity and innovation. In *CHI*.

[Sutton and Barto, 1998] Sutton, R. S. and Barto, A. G. (1998). *Reinforcement Learning: An Introduction*. MIT Press, Cambridge, MA.

[Theocharous et al., 2001] Theocharous, G., Rohanimanesh, K., and Mahadevan, S. (2001). Learning Hierarchical Partially Observable Markov Decision Processes for Robot Navigation. IEEE Conference on Robotics and Automation.

[Youngblood et al., 2005a] Youngblood, G. M., Holder, L. B., and Cook, D. J. (2005a). A learning architecture for automating the intelligent environment. In *to appear in Proceedings of the Conference on Innovative Applications of Artificial Intelligence*.

[Youngblood et al., 2005b] Youngblood, G. M., Holder, L. B., and Cook, D. J. (2005b). Managing adaptive versatile environments. In *Proceedings of the International Conference on Pervasive Computing*.

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

[Zola, 1997] Zola, I. K. (1997). Living at home: The convergence of aging and disability. In Lanspery, S. and Hyde, J., editors, *Staying Put: Adapting the Places Instead of the People*, pages 25–40. Baywood Publishing Company.