Multi-agent Software Control System with Hybrid Intelligence for Ubiquitous Intelligent Environments

Kevin I-Kai Wang, Waleed H. Abdulla, and Zoran Salcic

Department of Electrical and Computer Engineering, University of Auckland. Private Bag 92019, Auckland, New Zealand
{kevin.wang,w.abdulla,z.salcic}@auckland.ac.nz

Abstract. In this paper, a novel ubiquitous intelligent environment platform and its multi-agent control system are presented. The platform named Distributed Embedded Intelligence Room (DEIR) has been constructed with embedded sensors, actuators and computing devices. All the devices are interconnected using five different physical networks. This platform aims to facilitate realistic data collection and online system performance evaluation. The multi-agent control system incorporating two machine learning algorithms, fuzzy inference and decision tree, has been designed to conform to DEIR architecture. Devices to be controlled are classified based on their possible output states and modelled separately by fuzzy inference agents and decision tree agents in the system. The multi-agent control system with hybrid intelligence shows 11% improvements on overall control accuracy and 84% improvements on learning time compared to its predecessor control system. The vast improvement on computational time shows suitability of the approach towards real-time, embedded applications.

1 Introduction

With the improvement of embedded and communication technologies, the possibility of ubiquitous intelligent environments starts to emerge. In order to achieve such an environment, vast numbers of devices, which are interconnected with each other, are embedded into ordinary living or working spaces. In general, intelligent environments are developed and are expected to be able to adapt, to predict and to have high level of autonomy [1]. Different levels of automation and intelligence can be realised through different hardware architectures, control paradigms and application services to assist human activities. Ultimately, ubiquitous intelligent environments should be aware of the environment context, be able to model and adapt to user's behaviour and respond on user's behalf [2].

There have been a number of academic projects mainly on modelling, controlling and service-providing aspects of ubiquitous intelligent environments. The AIRE (Agent-based Intelligent Reactive Environments) of MIT CSAIL (Computer Science and Artificial Intelligence Laboratory) has been designed to carry a number of ubiquitous computing projects, such as the multi-agent framework - Metaglue [3], resource manager [4], and the recent Hyperglue [5]. The University of Essex uses embedded agents with Fuzzy Inference System (FIS) to model occupant's behaviours

J. Indulska et al. (Eds.): UIC 2007, LNCS 4611, pp. 1046–1055, 2007.

[6]. In the University of Colorado, Artificial Neural Networks (ANNs) are used to control the lighting and heating services of residential environments involving different types of living spaces such dining room, living room and bathroom [7]. Industrial research efforts such as Microsoft Smart House [8] and Amigo project [9] are more concentrated on providing integrated working behaviours, device automation, and integrated network infrastructure.

In this paper, a novel intelligent control system using Multi-Agent System (MAS) approach with hybrid machine learning algorithms is presented. User control behaviours on different physical devices within the environment are modelled with two types of algorithms, namely FIS and decision tree. FIS and decision tree have been applied separately or jointly in other research applications such as activity classification [10] and medical decision making [11]. In our project, FIS and decision tree have been built into agents and incorporated in the multi-agent control system. In order to perform realistic data collection and online system performance evaluation, a ubiquitous intelligent environment platform named DEIR (Distributed Embedded Intelligence Room) has been constructed.

The rest of the paper is organised as follows. Section 2 introduces DEIR platform and its system architecture. Section 3 explains the hybrid modelling techniques. Section 4 evaluates the system performance by comparing learning time and control accuracy with the previous version of multi-agent control system. Section 5 provides future directions of the project and section 6 concludes the paper.

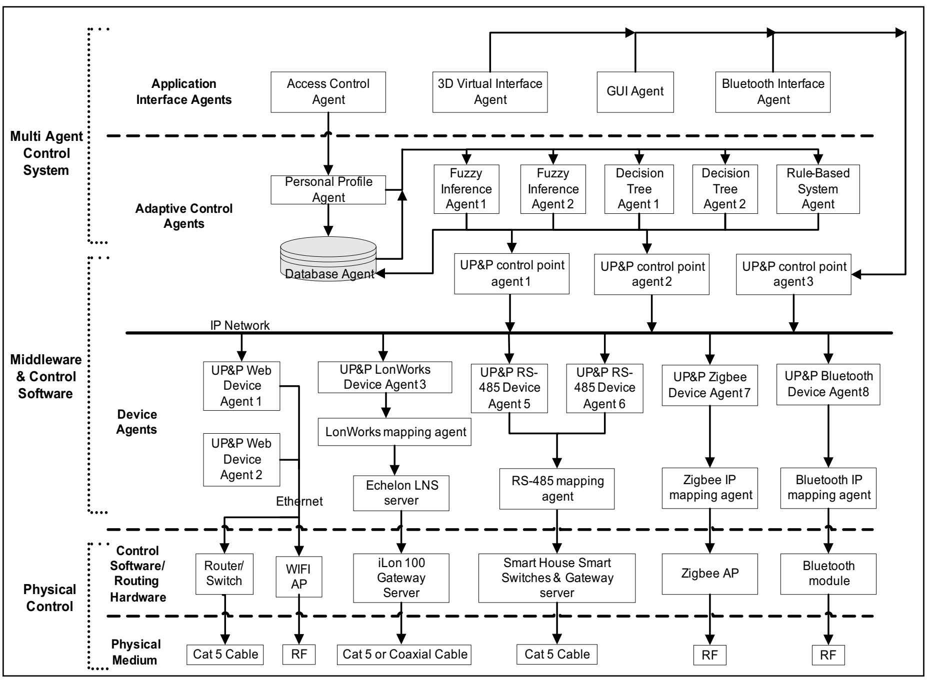


Fig. 1. DEIR System Architecture

2 Distributed Embedded Intelligence Room (DEIR)

To realise a ubiquitous intelligent environment, DEIR has been constructed with a number of sensors, actuators and computing devices [12, 13]. Embedded sensors such as light intensity, temperature, pressure and motion sensors are spread within the environment to monitor occupant's activities and environmental contexts. Embedded actuators are implemented to automate windows, curtains, dimmable lights and other appliances. The overall DEIR system architecture can be considered in 3 parts: the physical layer, middleware layer and multi-agent control layer, as depicted in Fig. 1. The physical layer consists of various physical communication media and routing devices that physically link all the embedded devices. The middleware layer contains middleware agents and control software for certain commercialised network. The middleware is introduced to integrate hybrid device networks and to provide unique portal between high level MAS and low level device networks. The multi-agent control layer includes high level modelling agents and user interface agents which model user activities for proactive device automation and provide integrated control interfaces.

2.1 Device Networks and Middleware

In the current DEIR system architecture, five different device networks, RS-485, LonWorks network [14], Ethernet/WIFI, Bluetooth and Zigbee have been considered and implemented. RS-485 network links most of the actuators for windows, curtains and dimmable lights. RS-485 devices are managed by a smart switch device that resembles traditional switch interface. The smart switch contains a M16C micro-controller that enables RS-485 communication and certain low level automation. LonWorks network connects part of the embedded sensors and the connected devices are managed by an iLon100 gateway server. Ethernet/WIFI and Bluetooth form the backbone communication for all the embedded and mobile devices. Zigbee is employed as the communication protocol for some embedded sensors, such as pressure sensors, and commercial office appliances.

Hybrid device networks improve the system flexibility in terms of incorporating different ranges of hardware devices. However, it adds a burden to the high level multi-agent control system to communicate with different devices. Furthermore, it is even more difficult, in terms of the system architecture, to replace an existing network or to introduce new network technologies. In order to integrate hybrid device networks and provide common portal to the high level control, a middleware level is introduced. In DEIR, Universal Plug and Play (UPnP) is used as the middleware. In general, UPnP is an IP-based device discovery protocol which enables automatic device discovery, configuration and management [12, 13]. Hence, each individual device network is mapped onto IP network and controlled as one integrated network. To enable UPnP communication, two UPnP components, UPnP device agents and UPnP control point agents, need to be implemented. The UPnP device agent is responsible for communicating with the physical devices. It is essentially the software replica of the underlying hardware. UPnP control point agents communicate with the device agent and pass necessary information to the high level modelling and/or interface agents.

2.2 Multi-agent Control System

In DEIR, a well known agent platform, JADE (Java Agent DEvelopment Framework), has been employed as the underlying framework of the high level multi-agent control system [13]. Since both JADE and UPnP are provided in Java, middleware components can be easily integrated as low level agents in the control system. Apart from the system management agents provided by JADE, high level MAS mainly consists of three types of agents: Utility agents, Modelling agents and Interface agents. Utility agents handle things such as access control, user personal profile management and user activities data deposition. Modelling agents involve three types of intelligent agents, namely fuzzy inference agents, decision tree agents and rule-based system agents. The first two agents are data-driven and will perform a series of learning steps to reach the final model of user behaviours. In comparison, the rule-based system agent contains directly hard-coded IF-THEN rules to realise reactive responses of the system. These agents provide the system the ability to model user activities and provide automatic controls to the environment. The modelling techniques will be explained in detail in the next section. Interface agents communicate with user control interfaces such as mobile interface and GUI interface.

3 Hybrid Learning

According to our past research outcomes, it has been found that different devices exhibit different control behaviours [12, 13]. That implies single modelling technique may only be suitable for some control behaviours but not for all of them. The multi-agent architecture provides our control system the ability to apply different machine learning techniques to model different device control behaviours. In the current multi-agent control system, two types of machine learning algorithms, Fuzzy Inference System (FIS) and decision tree, are used. Both algorithms have high tolerance to uncertainties, which are more suitable for modelling human behaviours. The algorithms are incorporated as agents in the control system. The fuzzy inference and decision tree agents acquire necessary user activities data through database agent and perform required data processing and learning steps. Once the learning steps are completed, both techniques will present the modelling results in terms of IF-THEN rules, which coincide with the rule-based system agent and allow a more coherent automatic control rule base. The developed control system is called Multi-Agent Hybrid Intelligence control System (MAHIS).

3.1 Device Grouping and Classification

In our experiments, ten output devices including four dimmable lights, curtain, heater, non-dimmable bed light, non-dimmable desk light, MS Word, MS Media Player are to be controlled by the system. Seven input sensor devices including internal light intensity, external light intensity, internal temperature, external temperature, chair pressure, bed pressure and time of the day are recorded by the system as context information when output devices change status. However, it is important to note that not all the input devices provide useful context information towards controlling the output device. Therefore, modelling agents only acquire data of relevant input devices. Currently, the relevant input devices for each output device are defined by the designer. Fig. 2 shows each output device with its relevant input devices.

In order to apply different modelling techniques to different output devices, output devices need to be classified in certain ways. In our current experiments, simple classification has been made based on the possible number of output states. That means, whether the device has two (binary) or more than two output states. The classification is also shown in Fig. 2. The reasons of using such simple classification method are as follows:

1. Binary output devices are generally easier to be classified by decision tree than devices with multiple output states or even continuous valued output states.
2. The number of relevant input devices is small and hence the learning of decision tree will be efficient. The resultant tree structure is also not complicated.
3. Binary devices tend to contain fewer uncertainties in their control behaviours, possibly due to the limited number of output states. Based on the past experience, fuzzy inference does not work well with modelling behaviours with low uncertainties [12]. Nevertheless, fuzzy inference has been proven to work well with continuous valued output devices with high uncertainties [12].

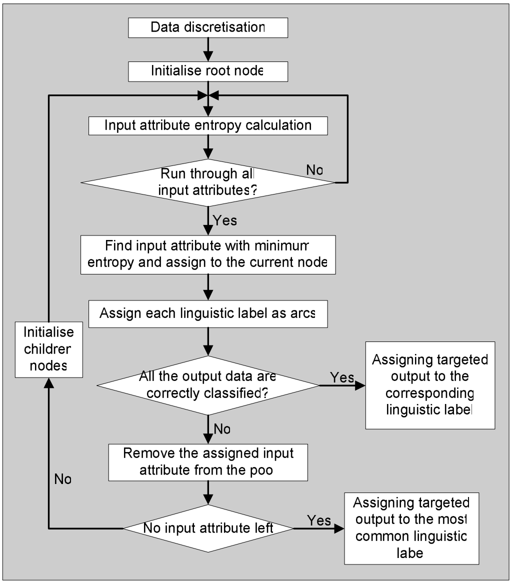
Output devices	Dimmable lights	Bed light (non-dimmable)	Desk light (non-dimmable)	Curtain	Heater	MS Word	MS Media Player
No. of possible states	>2	2	2	2	2	2	2
Modelling techniques	Fuzzy Inference	Decision Tree	Decision Tree	Decision Tree	Decision Tree	DecisionTree	Decision Tree
No. of relevant input device	5	5	2	5	2	2	3
Relevant input device	Int. Light Ext. Light Chair Press. Bed Press. Time	Int. Light Ext. Light Int. Temp Ext. Temp Bed Press.	Ext. Light Chair Press.	Int. Light Ext. Light Chair Press. Bed Press. Time	Int. Light Time	Ext. Light Chair Press.	Int. Light Ext. Light Chair Press.

Fig. 2. MAHIS device grouping and classification

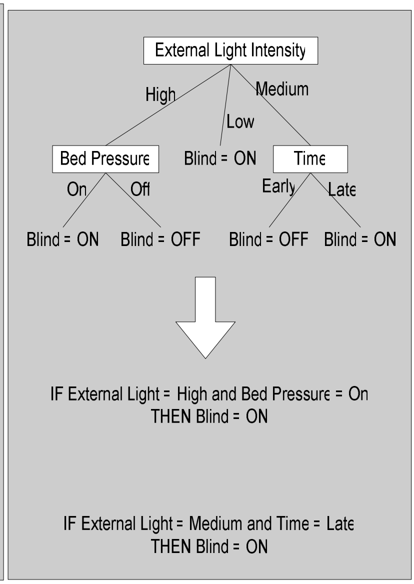
3.2 Decision Tree Learning

The decision tree learning algorithm implemented in the current decision tree agents is the traditional ID3 algorithm [15]. The learning process of ID3 is shown in Fig. 3(a). It should be noted in Fig. 3(a) that before feeding data into the ID3 algorithm, certain pre-processing is required. Due to the fact that decision tree classification does not handle continuous valued data well, continuous valued input device data needs to be discretised. For simplicity and efficiency, the dynamic range of a particular input device data is divided into arbitrary numbers of even divisions and each division is assigned with a specific linguistic label that denotes the value range.

ID3 algorithm construct the decision tree based on the information gain provided by a specific input attribute towards classifying the target output attribute. To evaluate the most suitable input attribute for a particular node of the tree, the information gain



(a)



(b)

Fig. 3. (a) Decision tree learning (b) IF-THEN rule conversion

of each available input attribute is calculated according to Eq.(1). The selected attribute is one that generates the maximum information gain at that particular node:

$$Info.Gain(S, A) \equiv Entropy(S) - \sum_{v \in Values(A)} \frac{|S_v|}{|S|} Entropy(S_v) \tag{1}$$

where S is the total population, A is the input attribute, Sv is a subset of population when A has value v.

However, refer to Eq.(1), Entropy(S) is common to all the input attributes and hence can be ignored in the actual implementation to improve algorithm efficiency. The algorithm terminates when all the output data is correctly classified or each path has reached its leaf node, which means it used all the available input attributes.

In order to use the resultant decision tree in the control system, the decision tree is converted into IF-THEN rules. Each path from the root node to leaf node is a unique "multiple inputs, single output" IF-THEN rule as shown in Fig. 3(b).

3.3 Fuzzy Inference Learning

FIS is the first machine learning technique introduced into DEIR's multi-agent control system, named Multi-Agent Fuzzy Inference System (MAFIS) [12]. Due to its tolerance to uncertainty, it has been recognised as one of the best modelling techniques for human behaviours. The learning algorithm used in current fuzzy inference agents and MAFIS is adopted from the Adaptive Online Fuzzy Inference

System (AOFIS) developed at the University of Essex [6]. The first step is to convert multi-dimensional dataset into single dimensional fuzzy sets using double clustering algorithm [16]. The second step is to quantify each fuzzy set using Gaussian membership function. Finally, the fuzzy rules are extracted using the extended Mendel Wang's method [17]. The resultant fuzzy rules will be in format of "multiple inputs, multiple outputs" IF-THEN rules, which are more efficient to model output devices of similar type altogether and to achieve a more compact rule base.

4 Performance Evaluation

To evaluate the system performance, MAHIS is compared with its predecessor, MAFIS, in terms of control accuracy and learning time. The dataset used for this evaluation contains seven input devices and ten output devices as described in section 3.1. This particular dataset contains 408 data instances collected over three consecutive days monitoring real user activities. The dataset is randomised and split into six training sets and six testing sets with 272 and 136 data instances, respectively. The control accuracy is measured based on Scaled Root Mean Square Error (SRMSE) which scales the Root Mean Square Error (RMSE) with the output device data dynamic range [13]. SRMSE ensures fair comparison of control accuracy among different types of output devices. In MAFIS, similar types of output devices are grouped together with their relevant input devices and each device group is modelled by an independent fuzzy inference agent [12]. The same fuzzy inference technique as in MAFIS is implemented in the current control system, MAHIS, but incorporated with the decision tree algorithm.

Fig. 4 (a) and (b) show the control accuracy and learning time results of MAFIS for each device group with different numbers of fuzzy sets [12]. The optimum control accuracy and corresponding learning time are highlighted. The overall system control accuracy and average learning time are calculated by taking averages of the optimum control accuracies and corresponding learning time of each device group. Fig. 5 shows the control accuracy and learning time of each device in MAHIS according to the group division shown in Fig. 2. The number of features shown in Fig. 4 and Fig. 5 indicates the number of relevant input and output attributes used in each modelling process. Fig. 5 shows clearly that the decision tree algorithm generally requires fewer features in the modelling process to achieve comparable control accuracy. The learning process of decision tree is also much faster than fuzzy inference technique.

Fig. 6 shows the performance comparison of MAHIS and MAFIS. The simulation results of MAHIS are further processed by taking averages to fit into the device grouping of MAFIS. For instances, the control accuracy of group 3 is the average control accuracy of curtain and heater shown in Fig. 5. The overall control accuracy and average learning time calculated in Fig. 4 and Fig. 5 are used to estimate the performance improvements of MAHIS. As shown in Fig. 6, MAHIS has achieved 11% improvements on overall control accuracy and 84% improvements on average learning time. The substantial improvement on learning time indicates that MAHIS is far more suitable for real-time, embedded applications such as ubiquitous intelligent environments.

MAFIS Scaled Root Mean Squared Error (SRMSE)				
No. of Fuzzy Sets	Group 1 (9 features)	Group 2 (9 features)	Group 3 (6 features)	Group 4 (7 features)
2	0.6726	0.7963	0.6608	0.5415
3	0.2102	0.2406	0.3157	0.1390
4	0.1798	0.2057	0.3127	0.1094
5	0.1389	0.1775	0.2687	0.0819
6	0.1204	0.1705	0.2199	0.0853
7	0.0979	0.1739	0.2220	0.1047
8	0.0931	0.1911	0.1536	0.0735
9	0.0893	0.1496	0.1249	0.0665
10	0.0974	0.1354	0.1004	0.0652
11	0.0835	0.1290	0.0972	0.0697
12	0.0865	0.1335	0.1330	0.0735
13	0.0716	0.1126	0.1566	0.0912
14	0.0731	0.1140	0.1404	0.0866
15	0.0742	0.0976	0.0735	0.0735
16	0.0807	0.1080	0.0891	0.0547
17	0.0792	0.1154	0.0976	0.0773
18	0.0780	0.1258	0.0857	0.0676
19	0.0900	0.1080	0.0819	0.0971
20	0.0783	0.1349	0.0949	0.0488
Overall accuracy	(0.0716+0.0976+0.0735+0.0488)/4 = 0.0729			

(a)

MAFIS learning time in seconds				
No. of Fuzzy Sets	Group 1 (9 features)	Group 2 (9 features)	Group 3 (6 features)	Group 4 (7 features)
2	21.46	19.3	10.61	17.14
3	15.80	21.2	15.64	15.92
4	15.80	18.6	14.67	16.95
5	18.71	22.0	12.29	14.49
6	22.40	17.5	14.53	16.06
7	17.20	16.8	13.43	14.63
8	18.10	18.8	15.69	15.31
9	17.18	15.3	12.39	12.08
10	24.43	17.0	13.46	12.73
11	18.39	17.3	13.96	15.90
12	16.75	20.0	12.10	14.52
13	17.32	25.0	13.79	14.18
14	18.95	23.0	15.86	13.54
15	15.85	16.2	12.76	15.52
16	16.18	20.3	15.62	15.55
17	15.48	15.6	14.93	13.18
18	17.25	16.3	13.85	20.27
19	15.45	18.4	13.19	18.71
20	16.94	23.6	14.40	19.07
Average learning time	(17.32+16.2+12.76+19.07)/4 = 16.33			

(b)

Fig. 4. (a) MAFIS control accuracy (b) MAFIS learning time

Output device	Modelling techniques	No. of features	Control accuracy (SRMSE)	Learning time (secs)
Dimmable Light 1-4	Fuzzy Inference	9	0.0716	17.32
Curtain	Decision tree	6	0.0286	0.09
Heater	Decision tree	3	0.0948	0.186
Bed Light	Decision tree	6	0.032	0.09
Desk Light	Decision tree	3	0.0572	0.0186
MS Word	Decision tree	3	0.0948	0.0186
MS Media Player	Decision tree	4	0.0756	0.096
Overall control accuracy	(0.0716+0.0286+0.0948+0.032+0.0572+0.0948+0.0756)/7 = 0.0649			
Average learning time	(17.32+0.09+0.0186+0.09+0.0186+0.0186+0.096)/7 = 2.546			

Fig. 5. MAHIS control accuracy and learning time

Device Groups	Devices	Control accuracy		Learning time	
		MAHIS	MAFIS	MAHIS	MAFIS
Group 1	Dimmable Light 1-4	0.0716	0.0716	17.32	17.32
Group 2	MS Word MS Media Player	0.0852	0.0976	0.141	16.2
Group 3	Curtain Heater	0.0617	0.0735	0.138	12.76
Group 4	Bed Light Desk Light	0.0446	0.0488	0.138	19.07
Overall control accuracy & average learning time		0.0649	0.0729	2.55	16.33
% of improvement		11%		84%	

Fig. 6. Performance comparison of MAHIS and MAFIS

In previous experiments, the adopted fuzzy inference technique AOFIS has been proved to generate comparable or better control accuracy compared with the other techniques including Genetic Programming (GP), Adaptive-Network-Based Fuzzy Inference System (ANFIS) and Multi-Layer Perceptron (MLP) [6]. Later, MAFIS has been evaluated to outperform AOFIS by applying the technique in the multi-agent architecture [12, 13]. Accordingly, MAHIS shows better control accuracy and faster learning than MAFIS should also perform better than those offline control systems.

5 Future Works

Based on the current findings, some aspects of the project can be further investigated.. In MAHIS, data division and linguistic mapping for decision tree learning use even scales. Different data division method involving statistically sound distribution such as Gaussian distribution should be discovered. Algorithms for automatic device classification and grouping should also be implemented. Online update function which adapts the IF-THEN rule base should be incorporated into the system to better suit the user from time to time and hence achieve a real-time unsupervised control system.

6 Conclusions

In this paper, we presented a novel ubiquitous intelligent environment platform, DEIR, and its multi-agent control system, MAHIS. DEIR contains a number of embedded sensors, actuators and computing devices. Contemporary network technologies such as WIFI, Zigbee and Bluetooth are employed as device networks to interconnect all the devices. UPnP is incorporated as middleware which integrates various device networks and provides a unique control interface to the high level multi-agent system. In order to model human control behaviours over different devices, two types of machine learning techniques, namely fuzzy inference and decision tree, are used in MAHIS. The modelling results are presented as IF-THEN rules for the system to apply automatic controls according to the environmental context. The system performance has been evaluated in terms of control accuracy and learning time. MAHIS achieves 11% improvements on control accuracy and 84% improvements on learning time. The results show that MAHIS outperforms many contemporary offline control systems including MAFIS, AOFIS, ANFIS, GP and MLP and is more suitable for real-time, embedded applications due to high computation efficiency.

Acknowledgement

The authors would like to thank Dr. Faiyaz Doctor, Prof. Victor Callaghan and Prof. Hani Hagras for their kind contribution of providing the dataset for comparative analysis and various helps regarding to the use of their AOFIS learning technique.

References

1. Augusto, J.C., Nugent, C.D. (eds.): Designing Smart Homes. LNCS (LNAI), vol. 4008. Springer, Heidelberg (2006)
2. Ducatel, K., Bogdanowicz, M., Scapolo, F., Burgelman, J.-C.: Scenarios for Ambient Intelligence in 2010. Information Soc. Technol., Advisory Group (ISTAG), Inst. Prospective Technol. Studies (IPTS), Seville (2001)
3. Philips, B.A.: Metaglue: A Programming Language for Multi-Agent Systems. In: M.Eng. thesis, MIT, Cambridge, MA, USA (1999)

4. Gajos, K.: A Knowledge-Based Resource Management System For The Intelligent Room. In: M.Eng. thesis, MIT, Cambridge, MA, USA (2000)
5. Peters, L.S.: Hyperglue: An Infrastructure for Human-Centred Computing in Distributed, Pervasive, Intelligent Environments. In: PhD thesis, MIT, Cambridge, MA, USA (2006)
6. Doctor, F., Hagras, H., Callaghan, V.: A Fuzzy Embedded Agent-Based Approach for Realizing Ambient Intelligence in Intelligent Inhabited Environment. IEEE Trans. Sys. Man Cybern. 35(1), 55–65 (2005)
7. Mozer, M.: The neural network house: An environment that adapts to its inhabitants. In: Proc. Amer. Assoc. Artif. Intell. Spring Symp. Intell. Environ., pp. 110–114 (1998)
8. Brumitt, B., Cadiz, J.J.: Let There Be Light! Comparing Interfaces for Homes of the Future. Microsoft Research, Redmond, WA 98052, MSR-TR-2000-92 (2000)
9. Ambient intelligence for the networked home environment (Amigo) (January 2007), http://www.hitech-projects.com/euprojects/amigo/index.htm
10. Parkka, J., Ermes, M., Korpipaa, P., Mantyjarvi, J., Peltola, J., Korhonen, I.: Activity Classification Using Realistic Data From Wearable Sensors. IEEE Trans. Info. Tech. in Biomedicine 10(1), 119–128 (2006)
11. Papageorgiou, E., Stylios, C., Groumpos, P.: A Combined Fuzzy Cognitive Map and Decision Trees Model for Medical Decision Making. In: Proc. of the 28th IEEE EMBS Annual Int. Conf, pp. 6117–6120. IEEE Computer Society Press, Los Alamitos (2006)
12. Wang, K.I., Abdulla, W.H., Salcic, Z.: Distributed Embedded Intelligence Room with Multi-agent Cooperative Learning. In: Ma, J., Jin, H., Yang, L.T., Tsai, J.J.-P. (eds.) UIC 2006. LNCS, vol. 4159, pp. 147–156. Springer, Heidelberg (2006)
13. Wang, K.I., Abdulla, W.H., Salcic, Z.: Multi-agent fuzzy inference control system for intelligent environments using JADE. In: Proc. of 2nd IET Int. Conf. on Intel. Environ., pp. 285–294 (2006)
14. Echelon Corporation: LonWorks Overview (February 2006) http://www.echelon.com/solutions/overview/default.htm
15. Mitchell, T.M.: Machine Learning. McGraw-Hill, New York (1997)
16. Castellano, G., Fanelli, A.M., Mencar, C.: Generation of interpretable fuzzy granules by a double clustering technique. Arch. Contr. Sci. 12(4), 397–410 (2002)
17. Wang, L.X.: The MW method completed: A flexible system approach to data minig. IEEE Trans. Fuzzy Syst. 11(6), 678–782 (2003)