In Journal of Uncertain Systems, Special Issue on "Uncertainty-based Technologies for Ambient Intelligence Systems", Vol. 2, Nr. 2, World Academic Press, pp.133-143, May 2008.

Intelligent Association Exploration and Exploitation of Fuzzy Agents in Ambient Intelligent Environments

Hakan Duman, Hani Hagras, Vic Callaghan

Department of Computer Science
University of Essex
Wivenhoe Park
Colchester, CO43SQ
United Kingdom

Abstract

This paper presents a novel fuzzy-based intelligent architecture that aims to find relevant and important associations between embedded-agent based services that form Ambient Intelligent Environments (AIEs). The embedded agents are used in two ways; first they monitor the inhabitants of the AIE, learning their behaviours in an online, non-intrusive and life-long fashion with the aim of pre-emptively setting the environment to the users preferred state. Secondly, they evaluate the relevance and significance of the associations to various services with the aim of eliminating redundant associations in order to minimize the agent computational latency within the AIE. The embedded agents employ fuzzy-logic due to its robustness to the uncertainties, noise and imprecision encountered in AIEs. We describe unique real world experiments that were conducted in the Essex intelligent Dormitory (iDorm) to evaluate and validate the significance of the proposed architecture and methods.

Keywords Intelligent Agents, Ambient Intelligent Environments, Fuzzy Control.

1. Introduction

The notion of Ambient Intelligence (AmI) has initially arisen through the efforts of the European Commission in identifying challenges for European research and development (Hagras, 2002; Luck, 2003; Doctor, 2005). The main aim of AmI is to deliver digital services and applications for the occupants of every inhabitable environment and to support their every day activities in a non-intrusive manner (Luck, 2003; Hagras, 2004; Callaghan, 2004). AmI is a multi-disciplinary research field combining many areas including ubiquitous computing, pervasive communications and intelligent (multi-) agent systems. The ubiquitous computing and pervasive communication technologies provide a scalable distribution and seamless integration of devices and their services into a heterogeneous environment. I*ntelligent agents* equip these devices with reasoning and learning capabilities. In general, the AmI vision describes an environment of potentially hundreds or thousands of embedded and mobile devices interacting to support particularized user goals and activities. However the multitude of interconnected devices and services can result in major *agent computational latency* (accumulating from the processing and communication overheads) as well as creating inherent complexities in programming and configuring the Ambient Intelligent Environments (AIEs). Hence, a major challenge in AmI, besides monitoring and learning the habits of the occupants, involves finding the *most relevant* associations between devices and services that are suitable for the environment and the user's specific needs while eliminating unnecessary and redundant associations (thus dissolving unneeded communication links). This will enable the realisation of AIEs with large number of devices as well as improving the efficiency and reliability of the overall network in an AIE.

To date, different approaches have been proposed to address the relevancy of devices for a given problem and domain. One of the best-known approaches is the semantic web (Hendler, 2001; Luck, 2003) where devices and their services are tagged with attributes and semantic descriptions so that they can exhibit the ability to autonomously search through the space for similar devices and services and form associations with them. Based on a given ontology the devices emerge to semantically-driven functional clusters and provide a reliable device as well as service discovery and aggregation. The ANS (McCann, 2004) features, for instance, a middleware for the integration of autonomic systems in ubiquitous computing environments. The ANS tool is somewhat similar to (Chin, 2005) as it uses the OWL ontology to find replacement devices e.g. in the case of a failure, but it also facilitates an adaptive agent-like component (node), which can learn the user's preferences. This ability is also used to make a decision over the importance of the replacement devices. The ANS rewards those nodes moves along with the user's preferences. The component based software agents come with predefined sets of abilities that pertain to that agent and its role with the system. The disadvantage of the ANS is that it requires updating its' ontology descriptions every time a new type of a device needs to be integrated into the system, otherwise the service of that device would not be available for the system. Each component is described in terms of what it requires and what it provides at data structure level, although they become adaptive at a later stage. In addition, ANS requires a continuous exchange of a large amount of messages among the agents of the environment (McCann, 2004), which are also processed at the agent level. A significant body of work is also emerging within the multi-agent system community which studies the integration of agent technologies to intelligently learn and exploit relevancy between associations (McCann, 2004; Dulay, 2005). The learning employed in (McCann, 2004; Dulay, 2005) was accomplished by task-specific predefined policies which enable the agents to specifically associate with available devices in the search space and allow some degree of adaptation e.g. to have policies for new devices joining the domain or a existing ones leaving, which is commonly the case for mobile devices. However, it cannot always be assumed that the devices and services present in an AIE are semantically described or predefined with operational policies. Hence, to allow the full realization of AIEs, the devices need to learn their associations and adapt their policies or rules during operation to changes and failures occurring within the environment. In addition to the above mentioned application-level approaches, there is an increasing popularity among the networking community which aim to employ intelligent techniques to equip nodes and peers with autonomic and self-* properties (Dulay, 2005, Duman, 2007a). In Anthill (Babaoglu, 2003), peers construct or remove associations to other 'relevant' peers (called neighbours) according to instantaneous information discovered by a special kind of ant agent during a search for resources. The overall aim is to be more resilient and dynamic to changes where prior knowledge is not assumed. This is a very important requirement for all dynamic systems that need to operate life-long and in an online fashion.

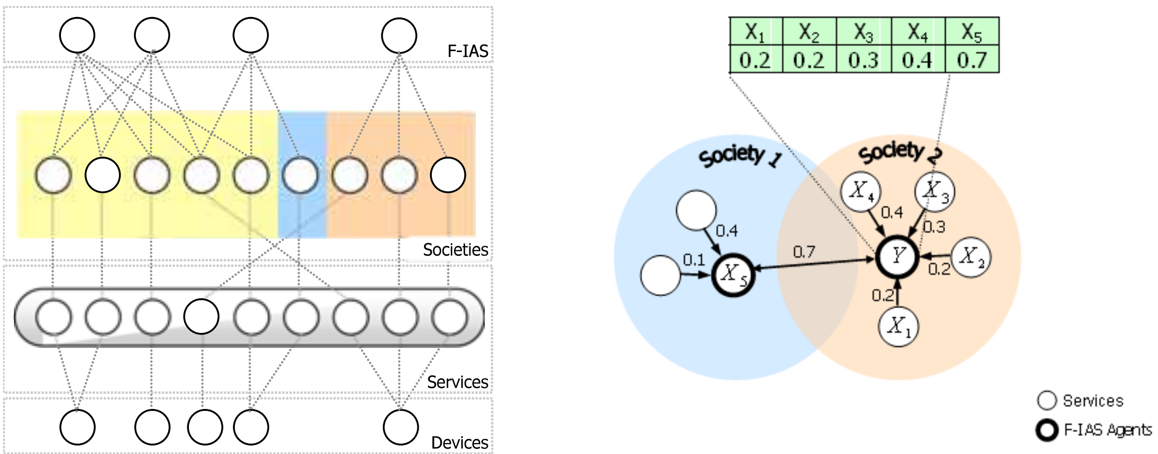
In this paper we present a novel fuzzy based intelligent method that is based on a function/semantic-free exploration algorithm to find and learn the most relevant associations between various devices and services operating in an AIE. The proposed approach employs fuzzy-based embedded agents due to their robustness to the uncertainties, noise and imprecision attributed to real world systems as in AIEs. The embedded agents seek to reduce the number of associations to other devices to minimize both the communication and computational processing latencies generated by the redundantly interconnected devices while satisfying the AIE and the user's personalized needs. The proposed approach for the Fuzzy Intelligent Association System (F-IAS) based embedded agents employs two processes. The first process uses a one-pass unsupervised online life-long learning technique to generate a fuzzy model of the user's particularized behaviour and needs. The second process performs an online intelligent association exploration based on modified hebbian-learning to calculate the association weights (relevancy) between services and the F-IAS agents. The proposed approach has been tested and verified in the Essex intelligent Dormitory (iDorm) which is a unique test bed for AIEs research.

This paper is arranged into 5 sections. In section 2, we introduce the Intelligent Association System (IAS) framework and architecture. Section 3 presents and covers the Fuzzy-IAS agents capable of monitoring and learning the occupant's behaviours and preferences. This section describes also the use of a

Mamdani-type fuzzy controller which can handle uncertainties and noise in the system. Section 4 presents the experimentations within the iDorm, a unique test bed for AIE research, to validate the significance of the proposed architecture and methods and discusses the obtained results. Finally, section 5 concludes with a summary of this paper and presents the future work.

2. The Intelligent Association System Framework

The Intelligent Association System (IAS) defines the architectural framework for the intelligent association exploration and exploitation method as depicted in Figure 1a (Duman, 2007a). The IAS framework resides on top of a physical network which combines all devices and their services to a decentralized service-oriented overlay network architecture. With this, the IAS becomes truly open, dynamic and resilient to changes in the system as well as fault-tolerant as it eliminates the phenomena of single-point-of-failures of centralized systems. The services provided by devices may be combined into societies according to their relevancy and are associated with Fuzzy-IAS agents (F-IAS) which integrate intelligent association learning and exploration capabilities. It should be noted that F-IAS agents can be separate entity of the system (e.g. in form of software agents) however here these agents are embedded into actuation devices (e.g. lamp, heater, TV etc) thus becoming *intelligent embedded agents*. In contrast, the services are pure information providers where they can transmit sensory states, etc. The F-IAS agents aim to learn their particularized behaviour by monitoring the user interacting with them in a non-intrusive and life-long manner. In addition, these agents simultaneously conduct a search through the IAS overlay network to find the most relevant services required for their tasks (Duman, 2007a, Duman, 2007b) which may lead to a more efficient operation of the agents after reducing agent computational latencies by eliminating redundant associations to services.



a) b)

Figure 1. a) The Intelligent Association System Framework b) The structural concept of weighted fuzzy cognitive maps for F-IAS agents.

3. The Fuzzy-IAS Agents

3.1. Fuzzy Cognitive Maps

The F-IAS agents use the structural notions and descriptions of Fuzzy Cognitive Maps (FCMs) (Kosko, 1992) allowing causal evaluation among the associated services. Before explaining how FCMs are employed by F-IAS agents the following briefly explains what Fuzzy Cognitive Maps are.

A Fuzzy Cognitive Map is a combination of fuzzy logic and neural networks; it combines the heuristics and common sense rules of fuzzy logic with the learning capabilities of neural networks. They were first introduced in (Kosko, 1992), where they used a fuzzy reasoning to enhance the cognitive maps that had been previously used in the field of socio-economic and political sciences to analyse social decision making problems. Kosko considered fuzzy values in the variables of cognitive maps and utilized them in order to represent causal reasoning. FCMs have been applied for many applications in different scientific fields (Duman, 2007a). A sample case (also facilitating intelligent agents) on how FCMs are employed to obtain answers to *what-if* conditions can be found in (Kosko, 1992).

The most significant characteristics of FCMs that comply with the F-IAS agents are:

- The FCMs structure allows the F-IAS agents to dynamically reorganize themselves, e.g. new services can be integrated or existing ones removed during operation in an ad-hoc fashion.
- The F-IAS agents or services can form the concepts/nodes of FCMs and the connections of the FCMs can illustrate the association and causality.
- The association among the F-IAS agents and services is indicated with a weight value, which illustrates the strength of the causation and as a result represent the importance of the association.
- An unsupervised mechanism (mostly Hebbian learning (Kosko, 1992)) for FCMs gradually learns the association strengths after correlating the changes of two associated entities.
- The overall system is event-driven and decentralized.

The above characteristics show that FCMs in general are well suited for determining the relevant associations among F-IAS agents and services. In addition, FCMs comply with the high-level concept and notions of the IAS architecture as depicted in Figure 1a. F-IAS agents and services are associated with each other which usually result in shaping societies. The dynamic nature of FCMs allows the societies to become an ad-hoc and distributed environment, where new devices can join or existing ones leave in real time and without human intervention. Figure 1b illustrates how FCMs are employed for the IAS framework based on two societies. Note that the services and F-IAS agents of each society are fully interconnected with each other at the physical layer of a network. However since the IAS framework resides at a higher level the services can demonstrate different associations between the F-IAS agents forming various structures of societies. Furthermore the F-IAS agents personify *Em*bedded Am*bassador* (Embassador) Agent capabilities (Duman, 2007b) which can form a bridge between the societies so that information from a service based in society 1 can be transmitted via the F-IAS agent to society 2. The Embassadors are agents with higher computational and networking capabilities and as such become the mediators of a multi-society based messaging infrastructure. In other words, the Embassadors (besides operating as intelligent agents that seek to optimise the number of their associations) are also expected to be the message brokers between multiple societies in order to avoid local communication overloading of the societies. The Embassador has filtering and routing mechanisms installed which only forward events coming from external societies to the agents of its own society, if and only if they are requested, required and assumed to be useful. In addition to the filtering and forwarding capabilities, Embassadors are capable of generating new communication channels on-the-fly and informing the corresponding agent to subscribe to the newly created channel so that a “private” communication channel can be established. The main purpose of this is to decrease the number of messages multicast to all agents of a certain society. More information on Embassador Agents can be obtained in (Duman, 2007a).

3.2. The F-IAS Agent Controller

The F-IAS agents *perceive* the environment through the information provided by the associated services as an effect of the user’s intervention with them. Moreover, they affect the environment through their integrated actuators based on their learnt fuzzy logic controller which approximates and models the particularised preferences of the user. Here, the use of fuzzy logic as a controller is of particular importance since it provides robustness to uncertainties, noise and imprecision attributed to real world systems as in AIEs.

It is assumed that each F-IAS agent has a *N:1* relationship meaning that *N* possible services can be associated to *1* F-IAS agent’s output. It should be noted that this assumption can easily be extended to a relationship associating multiple inputs to multiple outputs (Zadeh, 1965; Zadeh, 1996) but for the sake of simplicity to describe the proposed approaches, it is considered that the F-IAS agent has only to control one output actuator. In addition, it should be emphasized that an F-IAS agent output can also be used as a service for another F-IAS agent, however in this paper it is assumed that sensory services are the sole inputs for the F-IAS agents.

For an AIE, after collecting the data set of K input-output data pairs each vector datum $(\vec{x}^k, y^k)$ can be expressed as $(x_1^k, x_2^k, \ldots, x_N^k; y^k)$, where $k = 1,2,\ldots,K$. The fuzzy system rule base comprises of a set of L IF-THEN fuzzy rules where the i^{th} rule has the following form:

$$R^i : \text{IF } x_1 \; is \; A_1^i \text{ AND } x_2 \; is \; A_2^i \text{ AND ... AND } x_N \; is \; A_N^i \text{ THEN } y \; is \; B^{i*} \tag{1}$$

where N is number of the services of the F-IAS Agent. The variable y represents the output of the F-IAS Agent and is represented by a gaussian fuzzy set B^{i*}. The F-IAS Agent controller uses singleton fuzzification, *max-product* inference method and height defuzzification (Zadeh, 1996), so the crisp output of this controller can be written as follows (Kosko, 1992; Duman, 2007a):

$$y = \frac{\sum_{i=1}^{L} w_i \overline{B}^{i*}}{\sum_{i=1}^{L} w_i} \tag{2}$$

where $\overline{B}^{i*}$ is the centre of the output fuzzy set of the i^{th} rule and w_i is the rule firing strength which is equal to the product of the membership values of the inputs for each rule. The lifelong learning and adaptation capabilities of the F-IAS agent for AIEs *requires* to have an effective, fast and reliable learning method that can generate new rules as well as adapting, changing and removing the existing rules. The rule induction method of the F-IAS which operates in an online and lifelong learning mode is described next

3.3. The Fuzzy Rule Extraction Algorithm

The rule extraction algorithm adapted by the F-IAS agents is based on the enhanced version of the Wang-Mendel (WM) method, using a one-pass technique to extract fuzzy rules from a sampled data set (Wang, 2003; Doctor, 2005; Duman, 2007a; Duman, 2007b). This algorithm enables every F-IAS agent within the IAS framework to learn the model and its behaviour through interacting with the user. It is a simple, reliable and fast data mining approach to extract fuzzy rules based on collected raw data. The procedure involves the following steps and is used to obtain the initial model of the system which at the same time characterises the user's behaviour (Wang, 2003):

- *Step I*: Establish associations to selected services. These services may have been selected by the user, randomly or intelligently (Duman, 2007a).
- *Step II:* Monitor the user interaction with the associated services and in the event of a change, the information is saved in a local storage.
- *Step III:* Once enough data (based on events) have been collected, assign for each input of the Fuzzy Logic Controller (FLC) a set of fuzzy membership functions. A double-clustering approach combining fuzzy c-means and hierarchical clustering is applied for obtaining these fuzzy membership functions. More information regarding the clustering algorithm can be obtained from (Doctor, 2005).
- *Step IV:* Expert rules are allowed but not necessary and may be combined with rules induced from the collected data set. This can be in form of accompanied rule (or fuzzy policies) provided by the manufacturer or freely added by the user (e.g. safety rules should not be overridden).
- *Step V:* Start reading events from the collected data set and for each data pair (x^k, y^k), compute the membership values $\mu_{A_j^q}(x_j^k)$ for each fuzzy set $q = 1,\ldots,V$, and input $j = 1,\ldots,N$, find $q \in \{1,\ldots,V\}$, such that $\mu_{A_j^q}(x_j^k)$ is maximum. The following is the rule generated by (x^k, y^k).

$$\text{IF } x_1^k \textit{ is } A_1^q \text{ AND } \ldots \text{ AND } x_N^k \textit{ is } A_N^q \text{ THEN } y \textit{ is } y^k \tag{3}$$

- *Step VI:* Repeat *Step V* for all *k* from *1,...,K* to obtain *K* data generated fuzzy rules in the form of equation 3. Divide the resulting rules into groups (*conflicting rule groups*) sharing the same IF part (antecedents) and having different consequents. Combine the group *l* with K_l rules into a single rule in the form of equation 1 where B^{i^*} is a Gaussian fuzzy set. The antecedents and the consequent of the obtained rule becomes the following form

$$\text{IF } x_1^k \textit{ is } A_1^q \text{ AND } \ldots \text{ AND } x_N^k \textit{ is } A_N^q \text{ THEN } y^{k(av^l,\sigma^l)} \tag{4}$$

with the *consequents' average* av^l and *variance* σ^l computed as follows:

$$av^l = \frac{\sum_{k=1}^{K_l} y_k^l w_k^l}{\sum_{k=1}^{K_l} w_k^l} \tag{5}$$

$$\sigma^l = \frac{\sum_{k=1}^{K_l} \left| y_k^l - av^l \right| w_k^l}{\sum_{k=1}^{K_l} w_k^l} \tag{6}$$

where w_k^l is the rule weight of each conflicting rules within group *l* and is computed as

$$w_k^l = \prod_{j=1}^{N} \mu_{A_j^q}(x_j^k) \tag{7}$$

- *Step VII:* Repeat this combination for all conflicting groups *l* to obtain the final rule set which contains *L* rules in the form of equation 1 and store it in F-IAS Agent's *Rule Base*.

The above procedure is employed online and in a life-long learning mode which allows the rule base to be adaptive so that new rules may be inserted or existing rules modified or deleted. In addition through the use of fuzzy rules the overall system guarantees immunity to noise and uncertainties associated in sensory information provided in AIEs.

3.4. The Offline Intelligent Association Analysis

After obtaining an initial model based on a set of associated services, the F-IAS agents conduct an offline analysis of the causal significance of the individual services and rank them according to their effectiveness. This process can be regarded as an intermediate step before the F-IAS agents transform to operate online and aim to explore and exploit associations to services.

The basic idea of the offline intelligent association analysis is to analyse the *cause-effect relationship* of the associated services to the F-IAS Agent. The *Causal Significance* module within the F-IAS agents simulates the *what-if* conditions to evaluate the possible cause-effect relationship of associations by computing and analysing the effectiveness of each service in respect to the F-IAS agent (Kosko, 1992; Duman, 2007a). The evaluation is based on a simple idea: the most effective and relevant associations do the best job of predicting the output for the F-IAS Agent.

The steps involved to analyse the causal significance is described by the following steps (Lin, 1998):

- *Step I:* For each F-IAS agent's service x_j, compute the fuzzy membership function value of *K* data points $(x_j^k, y^k),\ j = 1,...,N$, in each x_j^k - *y* space $k = 1,...,K$ which is in the form of a Gaussian fuzzy set as follows:

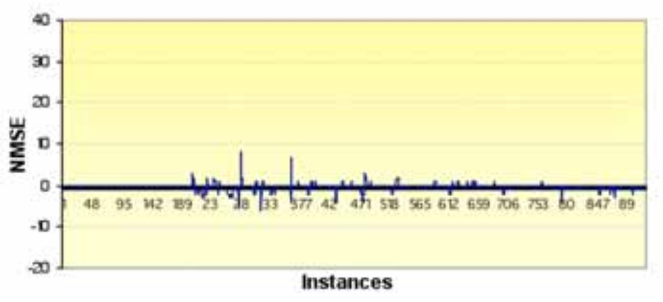
$$\mu_{A_j^i}^k(x_j^k) = \exp\left(-\left(\frac{x_j^k - c_{A_j^i}}{\sigma_{A_j^i}}\right)^2\right), \qquad k = 1,2,3,...,K \tag{8}$$

where A_j^i is the linguistic label of the fuzzy set, $c_{A_j^i}$ is the centre and $\sigma_{A_j^i}$ is the variance of the Gaussian membership function.

- *Step II:* Defuzzify these K data points (collected during the fuzzy rule extraction process – see section 3.3) by applying the rule base to the F-IAS Agent to produce an output $\tilde{y}^k_{FCAC}$ for each service x_j (Lin, 1998) using

$$\tilde{y}^k_{FCAC}(x_j^k) = \frac{\sum_{i=1}^{L} \mu^k_{A_j^i}(x_j^k)\overline{B}^{i*}}{\sum_{i=1}^{L} \mu^k_{A_j^i}(x_j^k)} \tag{9}$$

If an association to service x_j is more important than an association to a service x_m, then the approximation $\tilde{y}^k_{FCAC}(x_j^k)$ will be closer to y^k than $\tilde{y}^k_{FCAC}(x_m^k)$. Figure 2 shows the plotted $\tilde{y}^k_{FCAC}(x_j^k)$ for the associated services ILL, ITEMP in respect to the F-IAS agent DIM2. It can be observed that the ILL results in less prediction errors in comparison to the other service which means that ILL is of more importance for the F-IAS agent DIM2 than ITEMP.



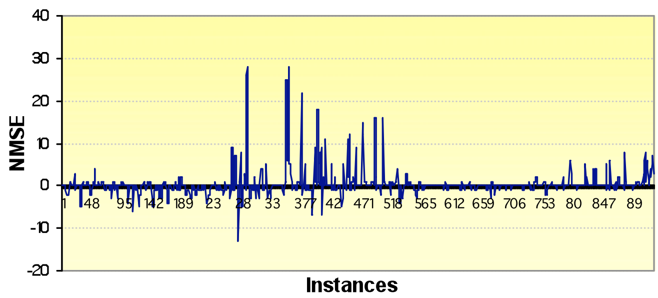


Figure 2: Offline Intelligent Association Analysis for F-IAS (DIM2)

The offline intelligent association analysis can be used to automatically and quickly extract significant information about the causal significance of associations to services. According to this information the associations can be evaluated and ranked. However, the elimination of the least causally significant associations can only be realised if, and only if, this wouldn't cause a major decrease of the F-IAS agent's accuracy and performance beyond a given threshold (e.g. here 10%).

3.5 Online Adaptive Intelligent Association Exploration and Exploitation

In contrast to the previous section, the online adaptive intelligent association exploration and exploitation evaluates the relevance of the associated services in a real-time and online fashion. With this, the F-IAS agents become more resilient and dynamic to changes and do not require collecting data sets to perform a history-based analysis. Rather, the assessment of the association importance takes place instantaneously which is more suitable to operate in real-time, online and lifelong manner. The following describes how the online intelligent association exploration and exploitation algorithm is utilized.

In the intelligent association exploration system, the weights are continuously computed and updated in the occurrence of an event. The weights increase when the state of both, the F-IAS agent and the corresponding services simultaneously change. In contrast, the weights decrease over time when the state change of a service cannot be correlated. The algorithm of the intelligent association exploration of an F-IAS is described as follows:

- *Step I:* Initiate a $FCM(Y)$ for the F-IAS Agent and establish associations to N services ($X_j \rightarrow Y$), *j=1,…,N*. The associations during the rule extraction process are used although different services may be selected.

- *Step II:* Set the association weights $\alpha_{X_j \to Y}$ of $FCM(Y)$ to zero, for all associated services so that the association matrix of F-IAS becomes $\Lambda_{X_j \to Y} = [X_1 = 0, X_2 = 0, X_3 = 0 \ldots X_N = 0]$.
- *Step III:* Set the learning rate δ = 0.1 and initialize the pre-associative flag to zero (ξ_{X_j} = 0) and post-associative flag to zero (ξ_Y = 0) for the agent pair $X_j \to Y$.
- *Step VI:* In the occasion of an event (service state change) do the following:
  a. For each event, update the pre-associative flag of the corresponding service ξ_{X_j} to 1.
  b. Calculate the resulting output of the event x_j^k by applying equation 2 of the F-IAS Agent.
  c. Update ξ_Y to 1 *only* if the F-IAS Agent has adjusted its output state due to event x_j^k.
  d. Calculate the new association weights $\alpha_{X_j \to Y}$ for each F-IAS Agent/Service pair $X_j \to Y$ at time *k* by applying the following equation.
  $$(\alpha_{X_j \to Y})^k = (\alpha_{X_j \to Y})^{k-1}(1-\tau)\delta\xi_{X_j}\xi_Y \tag{10}$$
  where $(\alpha_{X_j \to Y})^{k-1}$ is the association weight before applying equation 6 and τ is the decay value which is set to 0.01.
  e. Reset ξ_{X_j} and ξ_Y to 0.
  f. Repeat *Step IV* continuously until a given time *k*. As our system is operating in a life long learning mode, it will keep receiving new events and updating the associations.

The reason for adding a decay value τ is to prevent the association values calculation to increase endlessly. An association which was important for the F-IAS at the beginning might become redundant over time and without a decrease in the association weights this would never be noticeable. The decay value 0.01 has been derived empirically. Another major issue that needs to be addressed is the frequency of use of the services. It is obvious that the use of different services differs according to their functionality and purpose. Additionally different services provide different information. For example a chair pressure sensor only sends out an event if someone sits on it or stands up, whereby a temperature sensor constantly measures the temperature of the environment and regularly sends the updates. By using the algorithm above it is clear that the fewer events a service transmits, the less likely it will be regarded as a strong association. To overcome this situation, the following procedure aims to normalize the association weights according to their frequency of use, so that a better judgment of the importance between all associated agents can be performed.

- *Step I*: While running the above intelligent association weight exploration, count the total number of events $Count\left(x_j^k\right)$ for each service X_j.
- *Step II*: For each X_j, apply the following equation to obtain the normalization constant value.
$$\gamma_{X_j} = \left(1 - \left(\frac{Count\left(x_j^k\right)}{K}\right)\right) \tag{11}$$
- *Step III*: Assign γ_{X_j} to the following equation which forms a sigmoid function with a normalized constant value for each service.
$$f_{sig}(\alpha_{X_j \to Y}) = \frac{1}{1 + e^{-\alpha_{X_j \to Y} \times \gamma_{X_j}}} \tag{12}$$
- *Step IV*: The above equation generates normalized association weights so that an equal and fair judgment on the importance of association between the services and the F-IAS can be performed.
- *Step V*: Insert the obtained $\alpha_{X_j \to Y}$ into the association matrix Λ of the F-IAS Agent

If $f(\alpha_{X_j \to Y}) \geq \Theta$ (where Θ is a predefined threshold e.g. 0.15) then the association is of great importance to the F-IAS agent. On the other hand if the association weight of an F-IAS agent is smaller than the threshold Θ than this service may be considered as irrelevant or redundant and would become a candidate to be removed. During the life-time of the F-IAS agents, the intelligent association exploration mechanism constantly seeks to reduce irrelevant associations to services and simultaneously evaluates new and potentially more relevant and significant services that will maintain the fuzzy model quality while decreasing the overall agents communication and computational processing loads.

4. Experimental Results

4.1. The intelligent Dormitory (iDorm)

We have conducted several unique experiments within the iDorm at the University of Essex (see Figure 3), which is a real world Ambient Intelligent Environment fitted with a plethora of embedded sensors and actuators and different network platforms such as LonTalk, Tini 1-wire, IP and X10 all glued together by UPnP to form a heterogeneous network. Any F-IAS Agent can request and sent commands to other agents within the iDorm using UPnP without requiring knowledge about the underlying protocol translation. Further details on the interoperability and UPnP interactions between agents can be found at (Duman, 2007a).

The iDorm is ubiquitous because the user is surrounded by a multitude of interconnected devices and transparent since the devices are seamlessly integrated into the environment. The current iDorm is equipped with the following sensors: Internal Light Level (ILL), External Light Level (ELL), Internal Temperature (ITEMP), External Temperature (ETEMP), Chair Pressure (CHAIR), BED Pressure (BED), and a Clock (HOUR). The actuators and thus the F-IAS agents consist of the Desk Lamp (DESKLAMP), Bed Lamp (BEDLAMP), 4 Dimmable Ceiling Lamps (DIM1-4), Heater (HEATER) and a Cooler (COOLER). It should be noted that during the experiments, all the sensors were run on separate and independent hardware processing units whilst the UPnP stacks and intelligent agent mechanisms were run on a single PC as software multi-processes. The hardware processors used in this experiments were small (20MHz, 0.5MB RAM processors) that could not support complex agents and so we utilized the PC as a proxy for these agents (a common technique in distributed embedded-architectures). This approach provides a more flexible experimental structure.

Figure 3. The intelligent Dormitory (iDorm)

4.2. Evaluation

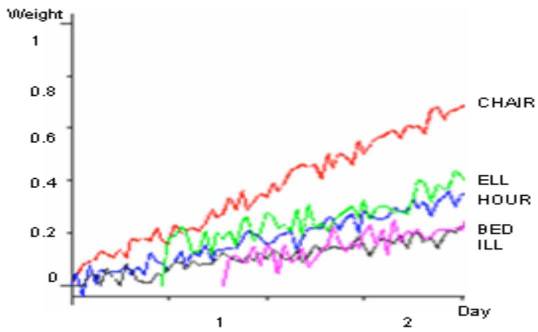
Several experiments on the intelligent association exploration mechanism were conducted within the iDorm with different users. Due to the limited space, this paper will present only a small subset of these experiments as described below. A detailed description of the experimental setup and other results can be acquired from (Duman, 2007a; Duman, 2007b). A user was asked to stay in the iDorm for 5 consecutive days. During this period the user interacted with the devices and performed his everyday normal activities, such as studying, watching DVDs and sleeping. During, the first 3 days the F-IAS agents (embedded in the actuating devices) monitored the user and collected the data based on events from the associated services (sensors). For this experiment, the F-IAS agents selected associations to all available services with the aim to eliminate the redundant services that are not required for the user's tasks.

For simplicity, the DESKLAMP F-IAS agent (hereafter only F-IAS agent) is used to explain the results. At the end of the third day the F-IAS agent extracted a total of 297 rules from 400 collected data sets. This formed the initial fuzzy rule base which resulted in a fuzzy model that approximated the user behaviour with a Normalized Mean Squared Error (NMSE) of 0.0108.

Figure 4. The F-IAS Agent with all associations.

The processing time obtained for each F-IAS agent to go through a single control cycle for the 400 data sets was in average 4797ms (Figure 4). An agent computational latency criterion (ACLC) was introduced to measure the F-IAS agent's additional computational processing load based on communicating UPnP sensors and F-IAS agents. More specifically, the ACLC calculates the period between an UPnP event initiation and acknowledgement delivery. Initially, the ACLC for this experiment was 3220ms. After the F-IAS agent has generated the initial model and extracted the required rules for its controller, evaluated and ranked the services based on their effectiveness and then it changed its operation to the online actuation mode and started to perform the intelligent association exploration. In this mode, the F-IAS agent continued with adaptation of rules and calculating the relevance of association whenever a state change of the services (events) took place. At every simultaneous change of the F-IAS agent and the services, the new association weight $\alpha_{X_j \to Y}$ was calculated. The calculated association weights of the F-IAS agent and different services during the last two days are depicted in Figure 5a. It becomes clear that the F-IAS agent in this experiment is strongly associated with the CHAIR. The resultant association weights for all services with the F-IAS agent are as follows:

$$\Lambda_{X_j \to DESKLAMP} = [ILL = 0.15, ELL = 0.31, ETEMP = 0.04, ITEMP = 0.03, CHAIR = 0.63, BED = 0.26, HOUR = 0.34].$$



Λ	X1	X2	X3	X4	X5	X6	X7
DIM1	0.24	0.25	x	x	0.35	0.21	0.28
DIM2	0.27	0.34	x	x	0.23	0.27	0.32
DIM3	0.16	x	x	x	0.23	x	x
DIM4	0.42	0.38	x	x	0.32	0.28	0.25
BEDL	0.23	x	x	x	0.16	0.56	0.34

Services: X1 = ILL, X2 = ELL, X3 = ETEMP, X4 = ITEMP, X5 = CHAI
X6 = BED, X7 = HOUR

a) b)

Figure 5. a) The online adaptive intelligent association exploration and exploitation process; b) F-IAS agent's association matrix

After applying the threshold Θ (0.15), (which can differ for each F-IAS agent depending on the limitations of resources in processing, network connections and memory) the most relevant associations were selected and irrelevant ones removed. With this, the F-IAS agent in this experiment defines its own society of services which included ILL, ELL, CHAIR, BED and HOUR. After eliminating ETEMP and ITEMP the number of rules reduced to 139, the NMSE increased slightly to 0.0518 (thus the model quality was not so much affected), the processing time for one control cycle reduced to 1203ms (Figure 6) and

In Journal of Uncertain Systems, Special Issue on "Uncertainty-based Technologies for Ambient Intelligence Systems", Vol. 2, Nr. 2, World Academic Press, pp.133-143, May 2008.

finally the ACLC decreased to 924ms due to less message transmission in the overlay network. This demonstrates the efficiency of the F-IAS agent where it reduces the processing and communication latencies while maintaining the quality of the learnt model. The experiments were also conducted in a life learning modes in which the system managed to find the most relevant associations in various scenarios when services break down or new services are introduced. Figure 5b shows the association matrix for all remaining F-IAS agents that were used in this experiment, where x indicates a deleted association.

Figure 6. The F-IAS agent after the intelligent association learning process.

5. Conclusions

This paper presented a novel intelligent association exploration and exploitation method used by the F-IAS agents to learn the causal strength and thus the importance of associated services operating in an AIE environment. The F-IAS Agents employ a special type of FCMs which use a Hebbian learning style procedure to calculate the association weights of their interconnections. The weights are continuously computed and updated after the occurrence of an event. The weights increase when the state of both, the F-IAS agents and the services simultaneously change. In contrast, the association weight decreases over time when the state change of the agents cannot be correlated.

FCMs have been used because of their similarity to the presented architectural framework of the IAS. The FCMs in general represent the society for the individual F-IAS agent and comply with the notions and paradigms for agent societies. Furthermore, FCMs provide a good framework for adaptation and dynamism of the system. New agents can be included and existing ones removed in an ad-hoc fashion and during real-time operation. Through FCMs, the F-IAS Agents adopt these features and facilitate a dynamic and adaptive structure.

Several sets of online experiments based on the iDorm were conducted to validate the significance of the proposed intelligent association mechanism for F-IAS agents. The experiments described in this paper involved multiple F-IAS agents associated to all services of the iDorm. The aim of this experiment was to illustrate weight calculation and the elimination of associations based on these weights. A user stayed for 3 consecutive days in the iDorm and interacted with the agents. The F-IAS Agents monitored the changes the user had caused and adapted to the new condition. After 3 days the F-IAS Agents successfully reduced their associations, e.g. from 7 to 2 associations in the case of the DIM3 F-IAS Agent while the system's prediction accuracy NMSE dropped at an acceptable level (e.g. from 0.0541 to 0.1339 for the DIM3 Agent). At the same time the number of fuzzy rules reduced from 297 to 24 for the DIM3 F-IAS Agent. The 91% reduction in the rule base resulted in less need for memory and faster processing for the F-IAS Agent.

As the experiments demonstrated the reduction of irrelevant and insignificant input agents is an important step in increasing efficiency and reducing the processing latency of the F-IAS agents. However, an arguable disadvantage of these eliminations is the relearning of rules aspect that the F-IAS Agents have to perform, since after removing an input agent, important rules may disappear and the F-IAS Agents depend on the user to relearn these rules. For instance, after removing the ITEMP input agent, the user intervened 11 times to re-teach the DESKLAMP F-IAS Agent previous conditions. The tradeoff between human intervention and accuracy and efficiency of an F-IAS Agents is considered to be an important issue and requires more attention, especially in AIEs where in the future hundreds or even thousands of agents may be interconnected with each other. The evaluation needs to be based on the available capabilities and resources of the agents. The vision of realizing ubiquitous AIEs assumes that most embedded agents will have limited computational and networking resources and where the networks do not overload due to the large amount of agents. Thus, the agents need to include the ability to make good judgements on the priorities of their functionalities. E.g. an agent frequently used by the user may try to minimize the forgetting factor of its rules. In contrast, agents which don't rely on external sources to relearn the rules may focus on optimizing themselves at the processing and networking level. For the iDorm experiments described in this paper, the re-learning aspects after applying the intelligent association calculation and removing associations were of a less burden for the users as the number of agents was limited and the adaptation process acceptable.

For our future work, we are investigating the trial of the proposed system in a truly distributed AIE with a richer set of sensors, actuators and F-IAS agents based on multiple overlapping societies, e.g. in the form of multiple rooms (like the newly established Essex intelligent apartment, namely iDorm-2). We are also aiming to explore more the tradeoff between human intervention and accuracy and efficiency of F-IAS agents as well providing the agents with the ability to make good judgment on the priorities of their functionalities.

References

[1] G. Acampora, V. Loia, Fuzzy Control Interoperability and Scalability for Adaptive Domotic Framework, IEEE Trans. On Industrial Informatics, vol. 1, 2005, 97-111.

[2] O. Babaoglu et al., Anthill: A Framework for the Development of Agent-based Peer-to-Peer Systems, 22nd International Conference on Distributed Computing Systems, 2003.

[3] V. Callaghan et al., Inhabited Intelligent Environments, BT Technology Journal, vol. 22, issue 3, 2004, 233-247.

[4] J. Chin et al., End-User Programming in Pervasive Computing Environments, Pervasive Systems and Computing, PSC-05, 2005.

[5] F. Doctor et al., A Fuzzy Embedded Agent-based Approach for Realizing Ambient Intelligence in Intelligent Inhabited Environments, IEEE Transactions on Systems, Man, and Cybernetics – Part A: Systems and Humans, vol. 35, issue 1, 2005, 55-65.

[6] N. Dulay et al., AMUSE: Autonomic Management of Ubiquitous e-Health Systems, Proc. UK e-Science All Hands Meeting, Nottingham, 2005.

[7] H. Duman et al., Intelligent Association Selection of Embedded Agents in Intelligent Inhabited Environments, Journal of Pervasive and Mobile Computing, vol. 3, issue 2, 2007a, 117-157.

[8] H. Duman et al., A Fuzzy-based Architecture for Learning Relevant Embedded Agents Association in Ambient Intelligent Environments, Fuzz-IEEE, 2007b.

[9] H. Hagras et al., Online Learning and Adaptation for Intelligent Embedded Agents Operating in Domestic Environments, Fusion of Soft Computing and Hard Computing for Autonomous Robotic Systems, Studies in Fuzziness and Soft Computing Series, Physica-Verlag, 2002.

[10] H. Hagras et al., Creating an Ambient Intelligence Environment Using Embedded Agents, IEEE Intelligent Systems Magazine, vol. 19, issue 6, 2004, 12-19.

[11] J. Hendler, Agents and the Semantic Web, IEEE Intelligent Systems, vol. 16, issue 2, 2001, 30-37.

[12] B. Kosko, Neural Network and Fuzzy Systems, ISBN 0-13-611435-0, Prentice-Hall, 1992.

[13] Y. Lin et al., Nonlinear systems input structure identification: Two stage fuzzy curves and surfaces, IEEE Trans. On Systems, Man, and Cybernetics – Part, vol. 28, no 5, 1998.

[14] M. Luck et al., Agent Technology: Enabling Next Generation Computing, AgentLink II, 2003.

[15] J. McCann et al., Building Ambient Intelligence into a Ubiquitous Computing Management System, Proc. International Symposium of Santa Caterina on Challenges in the Internet and Interdisciplinary Research (SSCCII-2004), Amalfi, Italy, 2004.

[16] L-X. Wang, The WM method completed: A flexible fuzzy system approach to data mining, IEEE Trans. on Fuzzy Systems, vol. 11, issue 6, 2003, 768-782.

[17] L.A. Zadeh, Fuzzy sets, Information & Control, 8, 1965, 338-353.

[18] L.A. Zadeh, Fuzzy logic = Computing with words, IEEE Trans. on Fuzzy Systems, 4, 1996, 103-111.

