C@sa: Intelligent Home Control and Simulation

Berardina De Carolis and Giovanni Cozzolongo

Abstract—In this paper, we present C@sa, a multiagent system aiming at modeling, controlling and simulating the behavior of an intelligent house. The developed system aims at providing to architects, designers and psychologists a simulation and control tool for understanding which is the impact of embedded and pervasive technology on people daily life. In this vision, the house is seen as an environment made up of independent and distributed devices, controlled by agents, interacting to support user’s goals and tasks.

Keywords—Ambient intelligence, agent-based systems, influence diagrams.

I. INTRODUCTION

Currently, houses are being networked, bringing the internet to the home and allowing for new services. In the future home environment, the user will be overwhelmed by a multitude of devices with complex capabilities, different access network interfaces and different multimedia and control services. However, introducing new visible technology does not always produce an improvement of the quality of interaction with technology that, especially in some countries, is still difficult to achieve [1]. Changing this trend and making home automation more accepted and spread through different user categories and type of services means creating environments in which technology is present but invisible to users. This goes in the direction of Weiser’s vision, in which the technology is going to be “invisible, everywhere that does not live on a personal device of any sort, but is in the woodwork everywhere” [2]. Then, smart homes should use AMIntelligence (AMI) solutions, putting together embedded technology and intelligence, in order to make inhabitants life easier.

In the AMI paradigm, people interact with a “real-digital” environment that is aware of their presence and of their interaction context. The environment answers in a proactive and adaptive manner to their needs, habits, emotional states, etc.. In this vision, people will be surrounded by intelligent and intuitive interfaces embedded into objects of daily use that will be able to recognize them and answer to their presence in a transparent way [3]. Then, an AMI environment is composed of independent and distributed devices (artefacts) interacting to support user-centered goals and tasks. The key characteristics of these intelligent artefacts are autonomy, distribution, adaptation, proactiveness, etc: therefore, in a way, they share the characteristics of agents [4,5].

Following this direction, we propose a MultiAgent System (MAS) which is aimed, on one side, at simulating control of an intelligent home from the functional viewpoint and, on the other side, at providing an interface layer for interacting with the house. In this paper, we discuss how an agent-based organization of the house control may help in achieving the goal of a National project1 to support architectural designers in testing the requirements of an intelligent house, in order to define guidelines for the integration of these technologies in tomorrow houses. In particular, the paper is structured as follows: in Section II we outline the architectural requirements of the system called C@sa (in Italian “casa” means “home”). In this Section we describe which is the role of each agent constituting the MAS and its organization and emphasize how the house behavior is decided. Section III illustrates the simulation and control 3D interface that allows to monitor the house behavior. Conclusions and future work directions are illustrated in the last Section.

II. C@SA REQUIREMENTS AND ARCHITECTURE

There are several projects concerning the realization of a Smart Home; for instance, Adaptive House [6] focuses on the development of a home that, observing the lifestyle and desires of the inhabitants, learns how to anticipate and accommodate their needs. In this system the control is handled by neural network reinforcement learning and prediction techniques. In the MavHome project [7] the house is seen as an intelligent agent that perceives its environment, through the use of sensors, and acts upon the environment through the use of actuators. The home has certain overall goals, such as minimizing the cost of maintaining the home and maximizing the comfort of its inhabitants. Another example of intelligent home that uses agent technology as a way to control the behavior of house appliances, from the resource consumption and coordination viewpoint, is the IHome Environment developed by [8]. IHome simulated environment is controlled by intelligent agents that are associated with particular appliances (i.e. WaterHeater, CoffeeMaker, Heater, A/C, etc.).

In developing our infrastructure, we were concerned about control, simulation and interaction with the home environment not only at a low abstraction level (single appliances behavior) but also at a higher level, closer to the user needs and goals. In our opinion, these ambient intelligence artefacts are likely to

Manuscript received November 5, 2004. This work is supported by the National project 1 PRIN 03 – “L’ambiente domestico informatizzato: progetto e verifica dell’integrazione di utente, tecnologia e prodotto” - Università di Bari and Siena, Politecnico di Milano.

B. De Carolis is with the Department of Computer Science, University of Bari, Bari 70125, Italy (e-mail: decarolis@di.uniba.it).

G. Cozzolongo is with the Department of Computer Science, University of Bari, Bari 70125, Italy (e-mail: cozzolongo@di.uniba.it).
be function-specific and will need to interact with numerous other ones present in the environment in order to achieve their goals and meet users expectations. Interactions can take place among artefacts and between artefacts and users, potentially requiring greater sophistication in interface issues and in user modeling. Achieving this aim requires establishment of a form of organization in which agents come together to form coherent groups able to satisfy user needs and desires.

Our research focuses primarily on the development of an infrastructure allowing the intelligent control of devices within a home. In particular, the system adapts the house behavior to inhabitant’s needs, adjusting the control of devices according to their “influence sphere”. An influence sphere, in our system, is defined in function of the type of service (comfort, wellness, saving, etc.) it provides to the house inhabitants and not in function of house zones (rooms) like in other systems [9]. In this first prototype of C@sa, we propose a hierarchical organization of different types of agents: operators, supervisors and interactors. The system has been implemented using JADE [10], a FIPA [11] complaint framework. The communication among the agents representing the house infrastructure is formalized using ACL (Agent Communication Language, [11]) messages which allows the information and knowledge exchange through a set of communicative acts. In particular, in order to use a more device neutral format, readable and easy to parse, we encoded the content of ACL messages in XML.

A. The OperatorAgent

An operator agent (Oi) controls and models the behavior of a simple artifact. It is defined by a set of attributes describing its features and by a set of behaviors describing the tasks that the user or another agent can perform on it. Each task is associated with a formal description that can be used with two aims: controlling the device and generating natural language explanation of its use [12]. In this way, if the user does not know how to use an appliance, he/she may ask explanations to the house that can employ the formal model as a knowledge base for generating help and user manuals [13].

Taking inspiration from the functional view of an agent presented in [14], the entire house can be seen as a macro-entity whose reasoning process is driven by sensing user actions and context parameters and whose output is manifested through some changes in the house appearance (controlled by some effectors). Then, an operator agent can be defined as belonging to one or both of the following classes: a) context sensor (CS) that measures the value of one or more device attributes (e.g. temperature, humidity, etc.), and b) effector (E) that affects directly the device state and/or other attributes (e.g.,heating at 26°, stereo playing a song, …).

B. The Supervisor Agent

Operator agents represent the entire home and are, in some way, related to each other (dependent, interacting, etc.). In particular, the state of a device may influence another device and therefore the house behavior.

For meeting users desires, artefacts, and therefore operator agents, need to interact with numerous other ones. Thus they need to be coordinated according to the recognized user needs.

This is the role of the Supervisor agent (S) which, according to the current context situation and to the presumed preferences and needs of the user in that context, reasons on how to coordinate the agents belonging to its influence sphere (Fig.1). Examples of influence sphere are the following: comfort, security, saving, wellness and entertainment. Therefore, we specialize the decisional behavior of each Supervisor agent according to the influence sphere it controls.

The Agent decisional behavior is determined by an influence diagram that models the relation between decisions (e.g. device actions), random uncertain quantities (e.g. user goals) and values (e.g. utility of the action).

Fig.2 shows the abstract schema that a generic supervisor agent uses for deciding the utility of an action on the user. In particular, in this influence diagram: i) the square box denotes the decision about performing an action at time ; ii) the round nodes are chance variables and, in this abstract model, they denote the house and user situation before and after the action execution: they describe the sensors situation and how this influence the context; obviously, since the house adapts its decision to the user in order to meet his/her requirements, the user situation at a given time is inferred accordingly; iii) the rhombus nodes represents the utility value for the user when is executed on the device .

Then, the semantics of this schema is: given a certain context configuration, defined by a set of sensors values, possible user goals and preferences in that situation are inferred at time , according to the probability functions defined in the model. The utility of performing is calculated in terms of cost of and of prevision of a change in the user situation after performing .

This approach provides a dynamic, uncertainty-based knowledge representation for modeling the inherent ambiguity in determining the likelihood of the agent to meet the user expectation performing some actions. This likelihood provides a decision-theoretic approach to change the state of the house for pursuing the goal of the Supervisor. The Supervisor agent
maintains a model of the user’s needs within a target influence
sphere.

Fig. 2. A General Decision Schema of a Supervisor Agent

An example of Supervisor is the Comfort agent, which
decides the appropriate atmosphere setting and controls the
behavior of the involved operator agents, according to some
contextual parameters (i.e. weather conditions, internal
temperature, etc.). For instance, the Comfort agent could set
the light intensity and colour, the internal temperature (by
activating the heating or the air condition), the music, the
smell (by an automated incense burner), the intrusiveness of
communication systems, etc.

To enable a Supervisor to reason about the trade-off of
different possible courses of action and to adapt behaviorally
to the changing environment, we implemented its decision
behavior as an instance of the abstract influence diagram
illustrated before. Since we had to include the Agent
Decisional Behaviour in the Supervisor Agent Class, that is
written in Java, we choose the Belief and Decision Network
Decisional Behaviour in the Supervisor Agent Class, that is
illustrated before. Since we had to include the Agent
Decisional Behaviour in the Supervisor Agent Class, that is
written in Java, we choose the Belief and Decision Network
java applet [15] as a tool for developing the influence
diagram. Fig. 3 shows a portion of the network representing
the reasoning behavior of the Comfort Supervisor. The
decision of opening windows is influenced by some contextual
parameters that can be derived by the context sensors (i.e. internal
temperature, humidity, user heart beat rate, and so on) and,
eventually, by some other more static parameters concerning
data about the user (i.e. age, environmental
profile.

Then, for instance, in case at time $t_{i}$ “the internal situation is
hot&wet”, the body parameters denotes a non comfortable
situation and consequently the personal wellness is
bad_tooHot/tooWet” then, the decision of opening the
windows (if they are closed) is not convenient if the external
situation would make the personal wellness at time $t_{i+1}$ worst.
In the considered example, the Comfort Supervisor will find
out an improvement of the personal wellness after opening the
windows.

This diagram aims at giving the idea of the general model
of the Comfort Influence Sphere. Then, employing a
structured view of an environment provides two major
advantages when attempting to control the home. First, control
can be achieved on any node of the network with a guarantee
that all causally dependent nodes will change accordingly.

For instance, the node representing the wellness level can be
forced into a specific state and all dependent nodes’ states
will subsequently be changed, if a state change is necessary.
Secondly, it can be used also for detecting problems with
sensors data (for instance, if the user feels bad because is hot
and the internal temperature is 10° C, then, probably there is a
problem with that sensor).

Once the Supervisor decides what to do, it
requires action execution to effectors involved in the
decision. This is done through an exchange of
ACL messages as shown in Fig. 4.

III. THE INTERACTOR AGENT

In our project we envisage two different interaction levels
directed to different categories of user: i) the environment
simulation and control interface to be used especially by
architectural designer for testing their hypothesis and ii) the
user interface level to be used by house inhabitants.

In the first case, the interface has to help the end user in
simulating context situations and in testing consequent house
reactions. In the second case, the house inhabitants should be
able to interact naturally with the house appliances or directly
(i.e. voice commands, tangible interfaces, touch screens, and
so on) or indirectly delegating tasks to an “house assistant”
(i.e. the virtual butler agent, a robot, etc.) or implicitly (i.e.
through sensors perception of relevant data).

In this first phase of the project, we are mainly concerned
with the implementation of the level of interaction aiming at
simulating and controlling what is happening in the house
given some context and user features.

The “Environment Simulation & Control” Interface has
been realized using 3D Graphics. In this first prototype, the
house zones and the objects within them have been realized in
3D Studio Max and then exported and transformed in VRML
(Virtual Reality Modeling Language, [16]). Fig. 5 shows a
portion of the 3DUI for interacting with the living room. In this selected view, the active entities controlled by operator agents are the light, the TV and the music stereo player, whose state is “on”.

In order to use the 3DUI for simulation and control purposes, it has been necessary to establish a connection with C@sa. This, at the moment, has been made through a protocol in which C@sa sends an ACL message whose content is the XML description of the situation in the selected house zone (i.e. the stereo is playing music), this will be received by a Java class able to parse it and to render, at the interface level, what is specified in the message.

Fig. 5. 3DUI showing the stereo playing music and the control menu.

On the other side, the user that is simulating or controlling the interaction with the house may want to: i) set some simulation conditions; ii) check the state of a particular device (exact parameters like the lux of the lamp or the movie the TV is playing); iii) change the state of a particular device in order to simulate what changes as a consequence of a user action.

A state change or the need to read some state attributes of an artifact is sent to the responsible operator agent through an ACL message. A change, obviously have an effect of the artifact is sent to the responsible operator agent through an influence sphere under discussion. In this case, actions in the decisional behaviour of the supervisor agent controlling the virtual world are collected by the usage model [17] that simulating the behaviour of an intelligent home. The idea at the moment, has been made through a protocol to our system.

In order to use the 3DUI for simulation and control purposes, it has been necessary to establish a connection with C@sa. This, at the moment, has been made through a protocol in which C@sa sends an ACL message whose content is the XML description of the situation in the selected house zone (i.e. the stereo is playing music), this will be received by a Java class able to parse it and to render, at the interface level, what is specified in the message.

ACKNOWLEDGMENT

We wish to thank those who cooperated in implementing the system described in this paper: Rosa Sarcina, Grazia Ricco, Anna Rowe and Vincenzo Silvestri. In particular, we thanks Fiorella de Rosis and Sebastiano Pizzutilo for their useful comments on the MAS organization. and Fabio Abbattista for his suggestions on the 3DUI.

REFERENCES