A Framework for Developing Multi-Agent Systems in Ambient Intelligence Scenarios

(Demonstration)

Pablo Campillo-Sanchez
Universidad Complutense de Madrid
pabcampi@ucm.es

Jorge J. Gomez-Sanz
Universidad Complutense de Madrid
jjgomez@ucm.es

ABSTRACT

Research in Ambient Intelligence usually cites Multi-Agent Systems as control solution. This demo introduces a development kit that combines Android, JADE-LEAP, and game engines to create a simulation environment where different systems can be developed. The demoed software will enable agent researchers to look for ways of controlling the elements of an Android based Ambient Intelligence system and deploying the solution to real devices. The demoed scenario assumes an Android based remote control, an Android Smart TV, and an Android Wearable. The user in the Ambient is a Parkinson's Patient whose tremors make hard to handle the remote. Through the detection of the condition of the patient, the devices can be rearranged in the best interest of the patient.

Categories and Subject Descriptors

I.2.11 [**Distributed Artificial Intelligence**]: Multiagent systems; I.6.7 [**Simulation Support Systems**]: Environments

Keywords

AmI, MAS, simulation, Android, JADE

1. INTRODUCTION

Ambient Intelligence (AmI) is a multidisciplinary area where Multi-Agent Systems (MAS) are frequently cited as an enabling technology. Nevertheless, a MAS researcher willing to contribute to AmI may find difficulties because of the infrastructure needed even for simple experiments. The problem is not unique to MAS researchers and it may be stopping others from making relevant contributions. To facilitate general experimentation in AmI and, in particular, with MAS in AmI, a framework is introduced that combines JADE based agents together with a 3D simulation. As a difference to other works, this aims for a software-in-the-loop solution. The MAS deployment working in the simulation is the same deployed in the real devices in a real environment. Since testing in real environments is expensive a computer simulation ought to make AmI research more affordable.

Appears in: *Proceedings of the 14th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2015), Bordini, Elkind, Weiss, Yolum (eds.), May 4–8, 2015, Istanbul, Turkey.*

The framework is part of an effort for facilitating the development of AmI systems for people with Parkinson's Disease (PD) in the SociAAL project (`http://grasia.fdi.ucm.es/sociaal`). A PD patient has tremors whose intensity depends greatly on how much medication they receive and the development stage of the PD itself. There is no a-priori periodicity of the symptoms, but caregivers have a major concern for them. As an example, a person with PD may fall off from a chair while sitting down due to these tremors. Specially designed AmI systems may aid PD patients and contribute to maintaining their autonomy.

The chosen scenario for this paper is an intelligent remote control development. In this scenario, the remote control has to collaborate with other devices in order to adapt to the PD patient's status. Tremors can make the operation of the remote control troublesome. They can make difficult grabbing the remote control itself; they can cause the patient to press a different button; press many at the same time; or just drop the remote control. More unlikely, but a remote may also be the cause of a patient fall because carrying weights may make them fall forward without noticing.

In the scenario, a MAS based control for an AmI environment is essayed. There are devices with basic capabilities and user interfaces that simulated humans can operate. Depending on the patient status, the MAS ought to respond to user actions differently, for instance, making buttons bigger in a remote control if the patient is having a tremors episode. Knowing the patient status may be possible if interaction with other devices, such as smartwatch, updated others about the existence of such tremors. If the AmI environment is an open one, different MAS may be chosen to use the same devices or to explore different control alternatives. The contribution suggests to use normative systems as means for ensure basic concerns, like not using camera sensors in certain situations, are not violated. The scenario is developed with Physical Human Activity Tester (PHAT) [2]. Videos and demos of the system can be seen at `http://grasia.fsi.ucm.es/sociaal/demos`.

The description of the simulation framework is given in section 2 together with an account of the relevance of MAS in this framework. The case study is briefly introduced in section 3. Related work is cited in section 4, and conclusions are left to the last section, section 5.

2. ABOUT PHAT AND MAS

PHAT permits to create 3D scenarios where interaction between simulated users and AmI devices happens. Devices in the framework are represented in the 3D scenario and connected to Android Virtual Machines, which host their actual control software. In this 3D scenario, a virtual actor reproduces the daily living activities of a PD patient, which includes the interaction with the devices.

The working hypothesis of the PHAT framework is that control software running inside the Android Virtual Machines will be unable to distinguish between the real world and the simulation. This hypothesis holds during the development because the input and output of the control software is limited by the Android Virtual Machines, which ensure a high compatibility, though not perfect, between the emulated device and a real one. As long as the Android sensing/acting capabilities are used, the control software within will not distinguish between reality and simulation. The use of Android limits the scope of the kind of AmI that can be done, though. Not all elements in an AmI system are complex enough to deserve hosting an Android OS instance to control them.

Android is only one element of the framework. Other elements are jME3&JBullet, a game engine for rendering scenarios and a physics engine to handle physical interaction among objects, respectively; and JADE-LEAP, for running the agents. The scenario setup is declared in a specification built with a modeling language called SociAALML. This modeling language was developed using INGENME `http://ingenme.sf.net`. The specification includes, but it is not limited to, the elements in the environment, the devices in the environment, if those devices are attached or not to the patient, and so on. An important part of the specification is a declaration of how a disease affects the activities of the virtual actor.

A MAS researcher will experiment with different AmI configurations and scenarios and will evaluate how a MAS is performing. MAS in PHAT are being studied as technology to deal with the coordination among devices and in cases where the control of a single device is not sufficiently defined and/or results of the combination of the skills of multiple agents. In this case, it implies a greater difficulty to harmonize the activity oriented architectures demanded by Android and the behavior-based ones demanded by JADE. A basic decoupling was applied to separate both the Android application and its corresponding agents, if more than one was needed.

3. THE CASE STUDY

The case study refers to the agent control in the SmartTV and in the remote control. Deployed devices are a smartwatch, a smart remote control, and a smart tv. The PD patient comes from another room, takes the remote control, and starts watching the TV. In this apparently simple scenario, several incidents may happen: the patient may drop the remote control, the patient may fall from the sofa or while walking, and the patient may have trembling issues while using the remote.

The challenges the agents have to deal with are: identifying when the patient is having tremors affecting the current activities, predict which action is going to be executed, and then, monitoring when this activity is actually requiring assistance. There are different MAS solutions that could be applied for this case. Our current experiments refer to the regulation of agent populations in this context in order to prevent undesirable behaviors. A normative system can evaluate agent actions over the devices and decide if they are rewarded or punished. APIs in the devices define available actions, making easier the monitoring problem. Also, the API enables the reuse of the scenario on behalf other researchers to create different solutions.

4. RELATED WORK

The idea of PHAT simulator is close to the initial agent testbeds used in AI research [3]. Issues common to initial works, such as addressing the simultaneity of actions over the environment, are dealt already through advances in game developments. Robocup [1] and Robocup-rescue [4] are another two initiatives in line with the proposal of this demo which are still active. The difference is the focus on the devices themselves and the delegation of event processing and scenario representation to the game and physics engine. Also, the use of virtualization technologies to isolate the control software, MAS in this case, from the simulation so that it can be reused in the real world. There are other AmI simulators in the literature, but they lack the software-in-the-loop feature [5].

5. CONCLUSIONS

The framework means new opportunities for AmI research by reducing the cost of experimentation with convincing, yet simulated, scenarios involving simulated users and simulated devices. The incorporation of agent technology to these simulations will allow MAS researchers to bring their knowledge to the AmI in a more consisting way. As an example, the paper briefly introduces a case study where a PD patient wants to use a remote control and find difficulties in doing so. The software is available from `http://grasia.fdi.ucm.es/sociaal`. It is distributed under GPLv3 license and includes a tutorial.

Acknowledgements

We acknowledge support from the project "SOCIAL AMBIENT ASSISTING LIVING - METHODS (SociAAL)", supported by Spanish Ministry for Economy and Competitiveness, with grant TIN2011-28335-C02-01 and by the Programa de Creación y Consolidación de Grupos de Investigación UCM-Banco Santander (GR3/14) for the group number 921354 (GRASIA group).

REFERENCES

[1] H. L. Akin, N. Ito, A. Jacoff, A. Kleiner, J. Pellenz, and A. Visser. Robocup rescue robot and simulation leagues. *AI Magazine*, 34(1):78, 2012.

[2] P. Campillo-Sanchez and J. J. Gomez-Sanz. Agent based simulation for creating ambient assisted living solutions. In *PAAMS*, pages 319–322, 2014.

[3] S. Hanks, M. E. Pollack, and P. R. Cohen. Benchmarks, test beds, controlled experimentation, and the design of agent architectures. *AI magazine*, 14(4):17, 1993.

[4] H. Kitano and S. Tadokoro. Robocup rescue: A grand challenge for multiagent and intelligent systems. *AI magazine*, 22(1):39, 2001.

[5] L. Tang, Z. Yu, X. Zhou, H. Wang, and C. Becker. Supporting rapid design and evaluation of pervasive applications: challenges and solutions. *Personal and ubiquitous computing*, 15(3):253–269, 2011.