

A Multi-Dimensional Model for Task Representation and Allocation in Intelligent Environments

Victor Zamudio, Vic Callaghan, and Jeannette Chin

University of Essex, Wivenhoe Park, Colchester CO43SQ, UK
{vmzamu, vic, jschin}@essex.ac.uk,
<http://ieeg.essex.ac.uk>

Abstract. In the future, homes will have numerous intelligent communicating devices, and the user would like to configure and coordinate their actions. Appliances and people in intelligent environments will have some degree of mobility. If the user wants to go from one place to another, using the same community, the agent should be able to generalize the service, trying to build an equivalent collection of coordinating services. This ‘work in progress’ paper addresses this issue by proposing a multi-dimensional model that allows visualisation of devices, temporal relationships, mutual interdependencies and the environment dynamics. The model both offers a simplified means of visualising the task space and the interdependencies together with a means of reasoning about algorithmic solutions to task processing. The work is aimed at supporting research into Pervasive Home Environment Networks (PHEN) which is funded by the UK’s Department of Trade and Industry Next Wave Technologies and Markets programme.

1 Introduction

Over the last decade, the rapid expansion of the INTERNET has opened new possibilities for communication with mobile phones and PDAs being used on an increasing scale. More recently the possibilities have extended to using the internet to control everyday living and working environments. A particularly popular and useful application of this work is, to assist elderly people [1], which require live longer in their homes. These kinds of system use a form of monitoring them, to help people with basic issues such as reminders, reports and control of devices. The overlap between pervasive (or ubiquitous) computing and intelligent agents has spawned the emerging area of Ambient Intelligence (AmI). This is a new multidisciplinary paradigm, which includes architecture, electronics, robotics, machine learning, etc [2], which has given rise to numerous new problems.

In this paper we propose a framework to solve problems related to task allocation in intelligent environments; particularly the formation of communities of communicating networked devices. We introduce a formalism about temporal communities, and discuss the advantages of this approach.



Fig. 1. The iDorm2 test bed

2 Related Work

2.1 Multi-Robot task allocation and cooperation

In robotics, although the problem of task allocation in multi-robots systems is well known, the efforts to formalize it are recent [3–5]. Cooperative robotics has become increasingly popular because it provides fault-tolerant and robust mechanisms to solve problems which a single robot would find difficult, if not impossible, to solve. In terms of fault tolerance, if one robot failed, other robots could continue with the task, albeit with a slightly degraded performance.

In Multi-Robots Task Allocation (MRTA) [3], a very frequent question is: which robot should execute which task? This simple question leads to more basic questions such as: what kind of task can the robots perform; what kind of task should they execute? These questions have been partially answered by other fields, such as operational research, combinatorial optimisation and set theory, amongst others.

Some of the core problems in MRTA relate to the heterogeneity of robots and tasks. In a multi-robot system, not all the robots are able to solve all the types of task that need to be accomplished. Gerkey and Mataric proposed a domain-independent taxonomy of MRTA problems [3] based on three axes: a) single-task robots (ST) vs. multi-task robots (MR), b) single-robot task (SR) vs. multi-robot task (MR), c) instantaneous assignment (IA) vs. time-extended assignment (TA). These three axes permit the description of a very wide spectrum of problems, abstracted in such a way as to aid the process of finding solutions.

Dudek et al [4] have proposed a taxonomy for robot collectives, using seven axes: collective size, communication range, communication topology, communication bandwidth, collective reconfigurability, processing ability and collective composition.

Chaimowicz et al [6] have proposed the use of a finite automaton approach, augmented with real-valued variables that changes with time. These hybrid automata can model continuous behaviour, communication and synchronization, and assume and exchange roles. The cooperation between several robots can be modelled by parallel automata.

2.2 Cooperative groups in intelligent environments

There has been some work done relating to community formation in AmI environments. The Smart-its project [7], have developed a system that allows sensors, computational capabilities and communication to be added to artefacts. By embedding such systems into artefacts, logical groups of coordinating devices can be formed. For example, by adding load sensors to the corners of a table, they have been able to detect and track items on a table. Vildjiounaite et al [8], have created temporal sets of clothes (all the parts of a business suit), sets of ingredients for cooking a cake, or sets of items needed for travelling (passports, flight tickets, etc). Duman et al [9] introduced a system to autonomously learn the cause-effect association between the action of an agent and the devices connected to it which is used to identify and remove redundant connections.

Shahi et al [10], have developed the concept of a Personal Operating Space (POS), which permits the user to control and interact with the environment, using a smart phone, and OTIS (Object Transfer in Smart Spaces)[11], which provides adequate support to transfer PC sessions within spaces. Task Computing [12] developed in the Fujitsu Laboratories, allow the user perform complex tasks and create new services in ubiquitous environments. This research will provide the task processing engine for POS.

Task Oriented Programming (TOP), developed by Chin [13, 14], permits the creation of new “virtual appliances” or “communities of coordinating devices”, by establishing logical connections between the sub-functions of appliances (audio, video, etc.). This paradigm called “The Decomposed Appliance Model” permits a logical link between pervasive entities, thereby enabling them to coordinate actions creating so-called “virtual appliances”. For example, the telephone, the light and the TV could form a new community, where the TV could react when the telephone rings. This work will provide the task mobility processing for TOP. Figure 2 shows the System being used on a tablet.



Fig. 2. TOP system on a tablet

2.3 Learning and Prediction in intelligent environments

Prediction in intelligent environments has been done in several ways, from fuzzy rules to markov chain approaches. Doctor et al [15] proposed a paradigm that uses fuzzy rules to learn “on-line” user behaviour. The system learns the behaviour of the user and finds relationships between this behaviour and the devices the user interacts with seeking to pre-emptively set the environment to match the users expected needs at any particular time. This work is based on the axiom which might be surmised as “the user rules ok” meaning that at any time the user may override the agent by simply setting the environment to the state he needs (which are, in turn, reflected into the rules learnt by the system). Markov Decision Process has been used to predict the user’s next task. Panduranga et al [16] introduced the Task-based Markov Model (TMM), clustering the user’s behaviours as high-level tasks, and using a hidden Markov model to predict the next user action. These clusters of actions (which are the hidden states of the Markov model) are formed according: i) the time difference between successive actions, ii) the difference in the device location, or iii) the number of actions in the group.

3 The Challenge

As we mentioned before, both appliances and people using the home of the future will have some degree of mobility. A person using a several services at a time should be able to move from place to place and use an equivalent set of services; the system should be able to generalize the service, trying to build an equivalent collection of coordinating services. If we have a new device in the environment, the system should be able to incorporate it to at least one community. In general, the environment would contain redundancy, in the sense that there would be more than one device that could supply a service. The following scenario is offered to clarify these concepts:

Part I

Peter, after a busy day, arrives home. He goes to the master room, with a headache, because of the heavy traffic. So, he turns on only the indirect light. Then he configures the main TFT monitor with a movie about the ocean, with slow and tranquil waves, while the sun sets. He decides to listen to some quiet and relaxing music through the local speakers, and selects Air on the G string, by Bach. Besides that, he closes the only window blind. This environment (in technical terms a task or virtual appliance) is then saved by the system as one of Peter’s personal preferences for future use (labelled by Peter as he as “Headache cure 1”)

Part II

In this environment he relaxes, and because he feels hungry he begins to look for an Italian food receipt. He chooses the lasagne, and goes to the kitchen. When he is there, the preferences he expressed in “Part I” are translated to this new environment and the two windows blinds close leaving only the indirect light over the table on. The music continues through the local speakers, and the familiar

picture on the TFT monitor in front of the microwave now shows the video with the sea, and the monitor of the i-fridge shows the receipt. Great!

Part III

The next week, his wife arrives home, really tired. She tells Peter she has a headache and asks him for a massage on her neck. Peter said: "Darling, this is better", and activates the "Headache cure 1" task which turns on the indirect light and selects Air on the G string; then, the main TFT monitor shows a sunset in the beach, and the only window begin to close.

In this scenario, some devices could be substituted (in the new environment) by a unique device: the speakers. Nevertheless, it is also possible that in the new environment that more than one device could perform the same (or equivalent) task. This is the case of the blinds, or the lights. The system should be able to choose which devices should used to compose the new (equivalent) community, according to their location, user desires (preferences), or performance. Some complex configurations, such as a traditional TV, which is a device that includes several sub-devices (monitor, speakers, tuner, switches etc), will need every sub-function to be discovered and an equivalent community constructed. There are several problems arising from the scenario:

- formatting temporal communities: some devices could be performing temporal tasks.
- learning of communities.
- reconfiguring of communities.
- dynamic environments: devices will come and go from the network (eg due to purchase of new devices, failure of old devices or nomadic use).
- mobility of the user: the user could be moving to new environments, and asking for communities previously configured in other environments. This could be seen as a particular case of a dynamic environment.
- complexity of the devices: some devices could be performing more than one task at a time and there may not be one to one matches in functional elements (eg one to many or vice-versa).

4 A Multi-Dimensional Model (MDM) of Pervasive Computing Space

We have developed a model of Pervasive Computing space that is formed using a 4D representation based on the following axis:

1. *Simple devices vs. complex devices.* A simple device can only perform one type of task, and can only perform one task at a time. Complex devices can perform several kinds of tasks at a time.
2. *Temporal tasks vs. non-temporal tasks.* A temporal task depends on time (eg are valid for a specific period). Non-temporal tasks do not depend on time.
3. *Coupled tasks vs. uncoupled tasks.* Coupled tasks have a mutual interdependency (ie are logically linked). Uncoupled task have no mutual dependency.

4. *Static vs. dynamic environment.* In a static environment, apart from system failure, devices do not move in time or space. In a dynamic environment devices come and go from the network.

In the next section we are going to formalize the problem, defining an allocation, a community, and an equivalent community. Then we will extend these communities in order to include time.

4.1 Formalising the MDM Model-Allocations and Communities

An *allocation* is a duple (d, T) where d is a device and T is a not empty set of tasks, i.e. $T = \{t_1, t_2, t_3, \dots, t_k\}$, with $k \geq 1$. If $k = 1$ we have a simple device, that is able to handle only one kind of task. This is the case of a speaker, or a microphone. If $k > 1$ then d is a complex device, which is composed by other sub-devices, i.e. can handle more than one task. This could be the case of a TV, composed by a device that can handle two different kinds of signals: audio and video. When the user configures a new set of virtual appliances, he defines a new *community*. A *community*, denoted by C , is a finite not empty collection of n allocations, i.e.

$$C = \{(d_1, T_1), (d_2, T_2), (d_3, T_3), \dots, (d_n, T_n)\} \quad (1)$$

If the user goes to a new environment, the agent should create an *equivalent community*, denoted by C_{eq} . In order to create this *equivalent community*, for each allocation $(d, T) \in C$ the agent should find an equivalent allocation (d_{eq}, T_{eq}) in the new environment. As we mentioned before, we have two cases: $k = 1$ and $k > 1$. i) If $k = 1$ then d is a simple device and $T = \{t_1\}$. The agent should find a new allocation $(d_{eq}, \{t_1\})$ such as the device d_{eq} is able to perform the only task t_1 . ii) If $k > 1$ then d is a complex device, and $T = \{t_1, t_2, t_3, \dots, t_k\}$. The agent should find, in the worst case, k allocations $(d_{eq}^1, \{t_1\}), (d_{eq}^2, \{t_2\}), (d_{eq}^3, \{t_3\}), \dots, (d_{eq}^k, \{t_k\})$, where every device d_{eq}^i is able to perform the task t_i , with $1 \leq i \leq k$.

4.2 Formalising the MDM Model-Temporal Communities

We could extend this framework in order to include time. A *temporal allocation* is a tuple (d, T, t_i, t_f) where d is a simple device, T is a (simple) task, t_i is the initial time and t_f is the final time. In other words, the device d will be performing the task T during $t_f - t_i$ units of time, beginning on the instant t_i . So, a *temporal community*, denoted by C_t is a non-empty set of temporal allocations:

$$C_t = \bigcup_{j=1}^k (d_j, T_j, t_{ji}, t_{jf}) \quad (2)$$

As we mentioned before, some devices (with their tasks) could be coupled, in the sense that there is a logical link or causal dependency between them. This could be expressed in the following way: Let $a = (d, T, t_i, t_f)$ and $a' = (d', T', t'_i, t'_f)$ two different temporal allocations with $t'_i > t_i$. We say that a causes a' or in other words that a' is a consequence of a if every time that a occurs it implies that a' will occur. We will denote this by $a \rightarrow a'$.

5 Preliminary Results

We conducted a number of trials using a total of 18 users, in our test-bed the iDorm2 (see Fig. 1), in which they completed various tasks such as configuring and using communities. In the first of these there was a fixed configuration of eight devices; occupancy of a bed and sofa, status of the window blind, bed light, desk light, heater, telephone and a media player software application. During this trial we collected binary information on the status of devices in the environment and used this to create the visualisation shown in Fig. 3. This provides a graphical representation of the temporal community, with eight devices showing binary status *on* and *off*, and its evolution over time. This graphical representation of

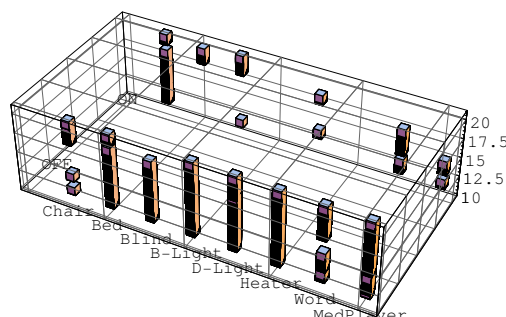


Fig. 3. Representation of a temporal community

a community summarizes all the information related to time, task and devices. The advantages of this approach are that it provides a way of simplifying the visualisation of complex tasks (depending on the users focus, any of the 3 planes can be used to view and reason about the tasks). Thus the user interface could switch between these various views of the task space. In addition, this model maps directly to the underlying formalism used by the machine processes.

In the second trial we used the TOP system [13] which allowed the users to form and operate their own communities from a selection of 5 networked devices; smart sofa, two table lights, media player and telephone. Using information collected from the TOP system, we were able to model the user formation of communities of devices, expressing their cause-effect relationship.

Figure 4 provides a visual representation of the communities created by one of the TOP users. In this case, the sub-set of devices involved are the sofa, the desk-light and the MediaPlayer. There are several cause-effects relationships (or, in terms of TOP, rules). The first relationship is: when the sofa is *off* (ie, when nobody is sat on it), the desk-light and the MediaPlayer are *off* as well. The second relationship is: when the sofa is *on* (ie, when somebody is sat on it) the desk light and the MediaPlayer should be *on*.

6 Discussion

Our Multi-Dimensional Model (MDM) is able to represent the user interaction with the environment (the iDorm2); in particular, we were able to represent in a graphical way temporal communities of devices with binary status (*on* and *off*). We are working to extend the model in order to include continues values. The TOP system used to collect information let the user include *if-then* rules for the devices. MDM include this cause-effect relationships as shown in Fig. 4. At the moment, our system is addressing only the case when one antecedent could cause several consequences (as shown in Fig. 4), although Chin's TOP tool is considerably more powerful as it allows the user to create multiple antecedent and consequences. In due course we hope to consider these more complex cases.

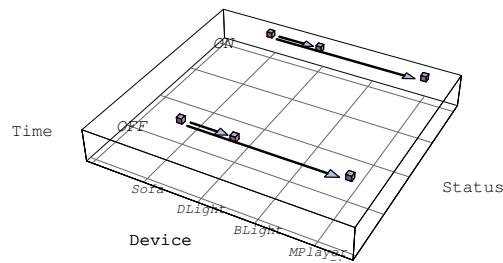


Fig. 4. A temporal community with causal relationships.

7 Summary and future work

This paper describes ongoing work into task processing. The main contribution of this work is the Multi-Dimensional Model (MDM) for solving the problem of configuration and automated retrieving of communities. In particular, we have identified four main axis that should be taken into account in order to solve the problem of reconfiguration of communities. This model allows the user to be presented with differing views of the task spaces, simplifying his visualisation and understanding of the space together with enabling machine processing such as pattern matching schemes to be employed. Part of our longer term aims are to use the visualisation and formalisms we have presented to allow systems such as TOP, to reason about resource constraints, functional conflicts and mobility of mapping in system of coordinating pervasive computing devices. This approach has the following advantages:

- It lets the user interact with the system, with intuitive (and easy to remember) information, such as device-task, device-time, or even just single information, such as device or task.
- The system is fault tolerant. If the user does not remember exactly the community, the system can deal with the differences or erroneous directions generated by the user.
- This representation simplifies the problems related to dealing with the complexity of the devices, temporality of the tasks, and dynamics of the environments.
- The visual representation of the cause-effect relationships between allocations provide rules of evolution of the communities.
- The formalisms open up a way to reason about multiple tasks and their interaction.

Currently we are working to include continuous states in the representation of the devices, and to include temporality in the description of the cause-effect relationships. At the moment we are only considering a *one-to-many* relationships for firing rules, but in the future we will include *many-to-one* relationships (several causes for one consequence). The work described in this paper is ongoing. It builds on existing work at Essex University, in particular it is aimed at providing task processing support needed to underpin paradigms such as Chin's Task-Oriented-Programming (TOP). At this stage we have established a representational and analytical model. Currently we are developing various task processing algorithms to use this model. For example, we are working to use pattern recognition techniques for retrieving the working communities. Also, we are investigating visual representations of the real time evolution of communities including cause-effect relationships. We look forward to reporting on these results at later conferences.

8 Acknowledgment

We are pleased to acknowledge the UK's DTI Next Wave Technologies and Markets Programme which has provided the underlying scientific challenge, access to state-of-the-art tools such as TOP and support for reporting this work. Victor Zamudio would also like to acknowledge the support of the Mexican National Council for Science and Technology (CONACYT). In addition, we would like to thank to Anuroop Shahi for his friendship and insight to the challenges of task programming that he provided. Finally we would like to thank Fernando Rivera-Illingworth for providing part of the experimental data set employed in this research.

References

1. Haigh, K. Z., Kiff, L. M., Myers, J., Guralnik, V., Geib, C. W., Phelps, J., and Wagner, T.: "The Independent LifeStyle AssistantTM(I.L.S.A.): AI Lessons

- Learned". In The Sixteenth Innovative Applications of Artificial Intelligence Conference (IAAI-04), July 25-29, 2004 San Jose, CA. Pages 852-857
2. Remagnino, P., Foresti, G. L.: Ambient Intelligence: A New Multidisciplinary Paradigm, Systems, Man and Cybernetics, Part A, IEEE Transactions on. Publication Date: Jan. 2005. Volume: 35, Issue: 1
 3. Gerkey, B. P., and Mataric, M. J.: A Formal Analysis and Taxonomy of Task Allocation in Multi-Robot Systems, International Journal of Robotics Research, 23(9): 934-954, September 2004
 4. Dudek, G., Jenkin, M., and Milius, E.: A Taxonomy for Multi-Agent Robotics. Robot Teams: From Diversity to Polymorphism, T. Balch and L. E. Parker (Eds.), 2002
 5. Gerkey, B. P., Mataric, M. J.: A Framework for Studying Multi-Robot Task Allocation. In Multi-Robot Systems: From Swarms to Intelligent Automata, Volume II. A.C. Schultz and other (eds.), pages 15-26, the Netherlands, 2003. Kluwer Academic Publishers
 6. Chaimowicz, L., Kumar, V. and Campos M. F. M.: "A Paradigm for Dynamic Coordination of Multiple Robots", Autonomous Robots 17(1): 7-21, July 2004
 7. Holmquist, L. E., Gellersen, H. W., Kortuem, G., Schmidt, A., Strohbach, M., Antifakos, S., Michahelles, F., Schiele, B., Beigl, M., Maze, R.: Building Intelligent Environments with Smart-Its. Computer Graphics and Applications, IEEE Volume 24, Issue 1, Jan-Feb 2004 Page(s):56 - 64
 8. Vildjounaite, E., Malm, E., Kaartinen, J., Alahuhta, P.: Networking of Smart Things in a Smart Home. UBIHCISYS 2003 Online Proceedings. UbiCom 2003, Workshop 7. 2003. <http://ubihcisyss.stanford.edu/online-proceedings/index.html>
 9. Duman, H., Hagaras, H., Callaghan, V.: A Soft-Computing based Approach to Intelligent Association in Agent-Based Ambient-Intelligence Environments. Published at 4th. International Conference on Recent Advances in Soft Computing 2002 RASC2002. Nottingham, U.K. December 2002.
 10. Shahi, A., Callaghan, V., Gardner, M.: Introducing Personal Operating Spaces for Ubiquitous Computing Environments. Pervasive Mobile Interaction Devices 2005 (PERMID 2005), hosted by 3rd International Conference on Pervasive Computing, Munich 8-13, May, 2005
 11. Shahi, A., Gardner, M., Callaghan, V.: Supporting Mobile Sessions Across Pervasive Smart Space Environments. The IEEE International Workshop on Intelligent Environments. University of Essex, 28th-29th June, 2005
 12. Masuoka, R., Labrou, Y., Song, Z.: Semantic Web and Ubiquitous Computing - Task Computing as an Example - AIS SIGSEMIS Bulletin 1(3) October 2004
 13. Chin, J., Callaghan, V., Hagaras, H., Colley, M., Clarke, G.: "End-User Programming in Pervasive Computing Environments", The 2005 International Conference on Pervasive Systems and Computing, Las Vegas, Nevada, USA, June 27-30, 2005
 14. Callaghan, V., Colley, M., Hagaras, H., Chin, J., Doctor, F., Clarke, G.: Programming iSpaces: A Tale of Two Paradigms, Chap 24 in book iSpaces published by Springer-Verlag, June 2005
 15. Doctor, F., Hagaras, H., Callaghan, V., Lopez, A.: An Adaptive Fuzzy Learning Mechanism for Intelligent Agents in Ubiquitous Computing Environments. Proceedings of the 2004 World Automation Conference, Seville, Spain
 16. Rao, S., Cook, D. J.: Predicting Inhabitant Actions Using Action and Task Models with Application to Smart Homes, International Journal of Artificial Intelligence Tools, 13(1), pages 81-100, 2004