Unwanted Periodic Behaviour in Pervasive Computing Environments

Victor Zamudio and Vic Callaghan
Department of Computer Science, University of Essex, United Kingdom
vmzamu@essex.ac.uk;vic@essex.ac.uk

***Abstract*—This paper addresses a fundamental problem related to the interaction of autonomous agents in pervasive and intelligent environments. Such autonomous agent could be static, nomadic or highly mobile, and in general, be programmed (rule based) to produce required behaviours by different users. In addition, the communication between these agents may include delays, because of the network, or because their own speed of processing the information. These two characteristics – the rules of behaviour and the temporal delays - could lead the system to display some unwanted periodic behaviour. In this paper we describe our work in progress which includes a framework to study this problem, and a set of initial guidelines to detect this behaviour. We conclude by describing the future direction of our work.**

***Index Terms*—Autonomous Agents, Pervasive Computing, Periodic Behaviour, Instability.**

I. Introduction

The area of pervasive computing is growing rapidly as new technologies such as Internet, smart phones, PDA's become available at ever cheaper prices. A key difference to pervasive computing and earlier generations is that devices are networked together, and the there can be an interdependence in the actions of all the devices on the network [1]. Thus devices, such as lights, heathers, mp3 players, TVs, etc. could be programmed to react (or perform a task) according to some rules of interaction, based on the behaviour of other devices [2]. Programming might be by autonomous agents or users using tools such as Pervasive Interactive Programming (PiP) [2] which provides a highly intuitive way of interaction between the devices and the program. Because devices could be programmed by more than one user (or programmer), the rules of behaviour for the devices could become very complicated, inconsistent, or generate some oscillating loops. This complex behaviour then is caused by there being several users, different rules, nomadic devices, and temporal delays. These delays could be caused by different factors, such as network delays, different speeds of processing etc. and could result in some devices receiving old information, and some queues of information could be formed. This phenomena is being observed increasingly in pervasive computing systems as the architectures moves from centralized to distributed control.

In this paper we are presenting a model for autonomous pervasive computing devices with delays. We show some examples of periodic behaviour, where the agents are subjected to some delays, and we present a theorem for two autonomous devices with different delays. Even though the first motivation of this work is related to the area of pervasive computing and intelligent environments, this result is general for autonomous devices, including, for example, agents or social networks with delays.

II. Related work

Asynchronous systems have some problems related to delays, such as hazards, races, and metastability. These hazards can lead the systems to display unwanted behaviour; in particular, dynamic hazards are related to different trajectories (with different delay times) for a single variable in a circuit [3] [4].

Software agents may be involved in loops with other agents, for example in email mailing list, where users have configured auto-replays that answer each other [5].

In Home Automation, the problem of Service Interaction occurs when two or more services show unexpected behaviour, caused by interference between them. Although there are several strategies to avoid this problem [6] [7] they don't detect interaction with loops.

III. The Problem

The problem that we are focused on concerns the behaviour of autonomous interacting devices (eg rule based agents). Such devices could interact with each other according to certain rules provided by, in general, several users. Besides that, there could be some delays in the propagation of the information between devices, because of the different speed of processing information in each device, or because of delays in the network. In situations where the state of one device is dependent on that of a second device, and that second devices state is dependent on the first, there is the potential for oscillation. If oscillation is not wanted, then this is a problem that needs to be identified and eliminated, which is the

Manuscript received April 2, 2006. Victor Zamudio would like to acknowledge the support of the Mexican National Council for Science and Technology (CONACyT).
V. Zamudio is with the Department of Computer Science, University of Essex, Wivenhoe Park, Colchester CO4 3SQ, United Kingdom (e-mail: vmzamu@essex.ac.uk).
V. Callaghan is with the Department of Computer Science, University of Essex, Wivenhoe Park, Colchester CO4 3SQ, United Kingdom (e-mail: vic@essex.ac.uk).

purpose of this work. In order to study this problem, we introduce some definitions. In this description we will use the term agents for autonomous interacting devices.

Let's suppose we have n autonomous devices agents $A_1, A_2, \ldots A_n$. Each agent A_i has a state $s_i \in \{0,1\}$, where 0 means that the agent is *off*, and 1 means that the agent is *on*. The state of the system is $S = (s_1, s_2, \ldots, s_n)$. Each agent A_i has two rules:

i) If ϕ_i then $s_i = 1$

ii) If ψ_i then $s_i = 0$

where

$$\phi_i : S_n \rightarrow \{0,1\}$$
$$\psi_i : S_n \rightarrow \{0,1\}$$

are boolean functions that depend on the states of the agents.

Let's suppose we have a minimal expression for the boolean functions ϕ_i and ψ_i of the agent A_i . If the functions ϕ_i and ψ_i depends on the state of k agents $A_j, A_{j+1}, \ldots A_{j+k-1}$, then there should be a direct link from these k agents to agent A_i (see Fig. 1). Each link between A_l with $j \leq l \leq j + k - 1$ and A_i has a delay $w_{li} \in Z^+$, and in general $w_{li} \neq w_{il}$. It means that if the state of the agent A_l is updated, A_i is going to evaluate their two rules after w_{li} units of time, ie, asynchronously. Let $U \subseteq S$ be a subset of S . Because of the dynamics of the system, the system will produce a sequence of states $U_1, U_2, \ldots U_p$. If this sequence of states is periodic then the subsystem U is said to be ***periodic***.

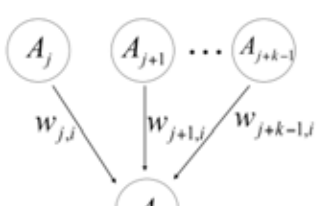

Fig. 1. The rules of the agent A_i depends on the states of *k* agents. For each dependency there is a delay *w*.

The fundamental questions are: what characteristics should the functions ϕ_i and ψ_i have in order to cause this periodic behaviour? What is the role of the delays? How can we prevent and correct this situation? The following theorem will helps us to understand this periodic behaviour.

Theorem 1: Let A_1 and A_2 be two agents as defined before, with rules defined by the boolean functions $\phi_1 = s_2$, $\psi_1 = \overline{s_2}$, $\phi_2 = s_1$, $\psi_2 = \overline{s_1}$, and delays $w_{12} = n$ and $w_{21} = m$, with $n, m \geq 2$ (see Fig. 2). Originally the state of the system is $S = (0,0)$. If at $t = 0$ the state is $S = (1,0)$ then the system is periodic, with period $T = n + m$.

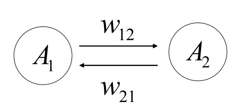

Fig. 2. The agents A_1 and A_2 have a mutual dependency, and different delays $w_{12} = n$ and $w_{21} = m$.

Proof: At $t = 0$ the system is in state $S = (1,0)$, and because of the delays, A_1 and A_2 should process, according to their rules, the first element of the string $b_1 = 0_1 0_2 \ldots 0_m$ and $b_2 = 0_1 0_2 \ldots 0_{n-1} 1_n$ respectively. With this, A_2 will be processing the state $s_1 = 1$ after n units of time. At $t = 1$ the system will be in state $S = (0,0)$, and $s_1 = 0$ and $s_2 = 0$ will be added at the ends of the strings b_2 and b_1 respectively (new information should be processed after a delay, according to each agent), ie. $b_1 = 0_2 \ldots 0_m 0_{m+1}$ and $b_2 = 0_2 \ldots 0_{n-1} 1_n 0_{n+1}$. Let's suppose, without any lose of generality that $n < m$. Because the next states to be processed by the agents are all 0's, all the following states will be $S = (0,0)$. At $t = n - 1$ A_1 will be processing the first element of $b_1 = 0_n 0_{n+1} \ldots 0_m \ldots 0_{m+n-1}$ and A_2 the first element of $b_2 = 1_n 0_{n+1} \ldots 0_m \ldots 0_{2n-1}$ and then at $t = n$ the system will be in state $S=(0,1)$, with $b_1 = 0_{n+1} \ldots 0_m \ldots 0_{m+n-1} 1_{m+n}$ and $b_2 = 0_{n+1} \ldots 0_m \ldots 0_{2n-1} 0_{2n}$. Because the next states to be processed by the agents are all 0's, all the following states will be $S=(0,0)$. At $t = m+n-1$ the state of the system will be $S=(0,0)$ with $b_1 = 1_{m+n} 0_{m+n+1} \ldots 0_{2m+n-1}$ and $b_2 = 0_{m+n} 0_{m+n+1} \ldots 0_{m+2n-1}$. Therefore, at $t = m+n$ the system will be in state $S=(1,0)$ and $b_1 = 0_{m+n+1} \ldots 0_{2m+n-1} 0_{2m+n}$ and $b_2 = 0_{m+n+1} \ldots 0_{m+2n-1} 1_{m+2n}$ which is the same situation as at $t=0$. All the process will continue exactly in the same way, and therefore the system is periodic. □

Figure 3 illustrates the evolution of the system. It can be seen that if n or m m are 1, the evolution of the system would lose a chain of states $(0,0)$. If we have $n=m=1$, the system will be oscillating between the states $(1,0)$ and $(0,1)$.

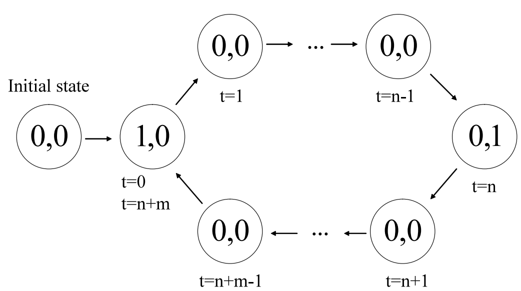

Fig. 3. Evolution of the periodic system. In each node, the first element is the state of the agent A_1 and the second element is the state of agent A_2. The systems has a period of $T = n+m.$

IV. Computer Simulations

The following computer simulations confirm Theorem 1. In *Fig. 4* we have the result of the evolution of the system for the case $n=3, m=2$. The computational simulations show a period of $T=5$, as predicted by Theorem 1.

As we mentioned before, if n or m are 1, the system will lose a sequence of $(0,0)$ states. In Fig. 5, $n=4, m=1$, and the evolution of the system will go directly from (0,1) to (1,0).

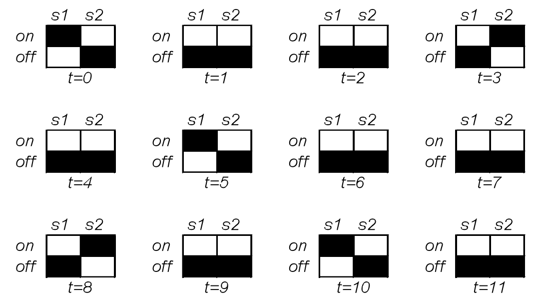

Fig. 4. Evolution of the periodic system when $n=3$ and $m=2$. The system has a period of $T=5$.

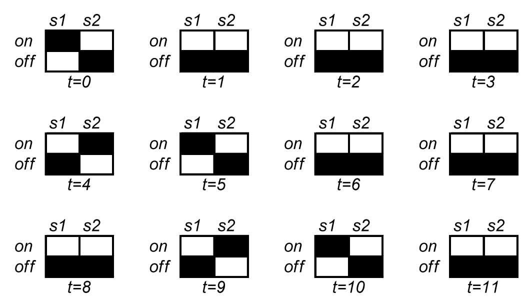

Fig. 5. Evolution of the periodic system when $n=4$ and $m=1$. The system has a period of $T=5$. After visiting state $(0,1)$, the system will go directly to state $(1,0)$.

V. Conclusions and Future Work

In this paper we have addressed the problem of instability that occurs in pervasive computing systems composed of multiple distributed interacting pervasive computing devices (rule based agents). Earlier systems, such as smart homes, were based mainly on centralized control servers where such problems do not exist. The move to distributed models has exposed this issue which is rooted in interacting rules and delays, We have discussed how this issue relates to other engineering domains such as asynchronous logic, and distributed computing. We have developed a theorem for a simple two device situation, which is general sable to any number of sequential devices in a loop, and shown some computer simulations. This is part of on-going work and our immediate plans are to expand upon this work to address cases of more agents, and to develop our formalism to modify the system rules or topology to eliminate instability.

References

[1] V. Callaghan, M. Colley, H. Hagras, J. Chin, F. Doctor, G. Clark. Programming iSpaces: A Tale of Two Paradigms, Chapter 24 in book iSpaces. Springer Verlag, June 2005.

[2] J. Chin, V. Callaghan, G. Clarke. "An End-User Programming Paradigm for Pervasive Computing Applications", International Conference on Pervasive Services 26-29 June 2006, Lyon, France.
[3] D. Lewin, *Logical Design of Switching Circuits*. 2nd Ed. Nelson. 1980.
[4] S. H. Hunger. Hazards, critical races, and metastability. Computers, IEEE Transaction on, Vol. 44, Issue 6, June 1995. Pages: 754-768.
[5] M. Mowbray, M. Williamson. Resilience for Autonomous Agents. Technical Report HPL-2003-210. Internet Systems and Storage Laboratory, HP Laboratories Bristol. October 17th 2003.
[6] M. Wilson, E. Magill, A Model for Service Interaction Avoidance in Home Networks, in Proceedings of the 5th Annual PostGraduate Symposium on the Convergence of Telecommunications, Networking and Broadcasting. Liverpool, June 2005.
[7] M. Wilson, E. Magill, M. Colberg. An Online Approach for the Service Interaction Problem in Home Automation. Consumer Communications and Networking Conference 2005 CCNC Second IEEE 3-6, Jan. 2005, 251-256.

