In the 6th International Conference on Intelligent Environments (IE'10), Kuala Lumpur, Malaysia. 19-21st of July 2010

# c-INPRES: Coupling Analysis Towards Locking Optimization in Ambient Intelligence

Victor Zamudio, Rosario Baltazar and
Miguel Angel Casillas
Division of Research and Postgraduate Studies
Leon Institute of Technology
Leon Guanajuato, Mexico
vic.zamudio@ieee.org,
charobalmx@yahoo.com.mx,
mcasillasa@prodigy.net.mx

Vic Callaghan
School of Computer Science and Electronic Engineering
University of Essex
Wivenhoe Park, United Kingdom
vic@essex.ac.uk

***Abstract*—Ambient Intelligence, and in general, any autonomous rule based system has been found to suffer from cyclic instability. This behaviour is characterized by unwanted oscillations, due to interacting rules within networks of pervasive computing devices. The binary behaviour of each agent is defined via a set of boolean rules, and the behaviour of the system as a whole is given by the ensemble of rules defined over the set of agents. From complex theory it has been found that the problem of cyclic instability cannot be solved analytically; however, it is possible to prevent it. In this paper we present a novel solution based on locking, to prevent cyclic instability. This strategy makes use of the topological properties of the digraph associated called Interaction Network (IN), and the local rules of the interacting agents. The concept of strong and weak coupling is introduced. Using the strong and weak concepts, a strategy c-INPRES that minimizes the number of agents locked is presented. Preliminary and encouraging results are shown.**

***Keywords-cyclic instability, combinatorial optimizacion, complex systems***

## I. INTRODUCTION

The possibility of configuring digital services in the smart home opens up a new and fascinating world. With this, it is possible to customize a set of rule-based agents embedded in the environment, offering personalized services to the end user. However, as a counterpart, and due to the set of rules configuring the space, it is possible to have configurations that would lead the system to cyclic instability, ie, a subset of the agents or devices behaving erratically, and alternating their states between on and off's periodically.

Formally, these instabilities can be represented by

$$S(t) = S(t + np \pm \delta) \tag{1}$$

where $S(t)$ is the state of the binary system at time $t$, $p$ is the period of the oscillations and $n \in Z^+$. The variable $\delta$ denotes network delays, latency, the different processing speeds of the devices etc.

The set of agents with their rules, initial conditions, user perturbations, and the evolution of its state through time can be seen as a complex system [17]. From complex systems theory it has been found that it is not possible to predict if a set of rules would lead a system to cyclic behaviour [14]. However, it is possible to prevent unwanted cyclic behaviour by making changes to the system based on analysing the structure's topological properties, together with the local rules allocated to the agents.

Failure in complex devices can be difficult to predict. This is the case for example of modern cars, where devices can have hundreds of inputs. With this, unexpected outputs can arise under strange and bizarre circumstances, and even very difficult to replicate, cars being one of the most complex real-time software systems [1].

In the AmI paradigm, the environment is aware of the presence of people, letting them interact with a digital world. The environment, populated with intelligent artifacts, will behave proactively to the needs, habits, emotional states and inputs from the user in general [4][11][12]. Multiagent systems have been useful for decentralized network security, where self-organized multi-agent swarms are evolved, in order to find the optimal rules [7]. There have been other attempts to define agents with certain functionalities in ubiquitous computing systems. For example, Bigraphical Reactive Systems (BRSs) are a way to represent hierarchical agents, buildings, computers, rooms, together with their reaction rules, that define how part of the system may change [8].

## II. THEORETICAL FRAMEWORK

### A. *Decision-taking in Ambient Intelligence*

In recent years, the importance of modelling relationships and, in particular, relationships of dependencies in pervasive computing has grown. A significant reason for this growth is that, without this information, it has been shown that decisions made by context-aware applications can be inappropriate or even lead the system to become unstable [6] [9]. Also, complex networks have emerged as a valuable tool in many areas, from biology, economy, internet, social networks, etc. [13].

It is possible to configure the set of rules allocated to the agents either automatically or manually [2][3][5]. In both cases, the main goal is to satisfy the user's desires. The set of rules are given in the form of logic gates or boolean rules, whose variables are the states of other agents and also input conditions given by the users.

An agent $A_k$ is an autonomous device consisting of a triplet $[s_k, r_k, w_k]$ where $k$ is the agent number for $k = 1,2,3,\ldots,n$, with $n$ being the total agents number and:

$s_k$: is the binary state of the *k*-agent defined over $\{0,1\}$
$w_k$: is the importance or weight over $\{Low, Medium, High\}$
$r_k$: is the set of boolean rules of the *k*-agent $\{\varphi_k, \psi_k\}$ defined as

$$\text{If } \varphi_k \text{ then } s_k = 1 \tag{2}$$

$$\text{If } \psi_k \text{ then } s_k = 0 \tag{3}$$

with

$$\varphi_k, \psi_k : S \to \{0,1\} \tag{4}$$

If we have $n$ autonomous devices $A_1, A_2, \ldots, A_n$ the state of the system is $S = (s_1, s_2, \ldots, s_n)$.

The rules defined in (2) and (3) are consistent in the sense that $\varphi_k = \psi_k^{-1}$. With this, the case of contradictory rules (e.g. one device ending up with two different states simultaneously) is avoided.

The set of rules defined over the agents can be used to build a network capturing the functional dependencies between the agents, as will be shown in the next section.

The factor of importance $w$ correspond to the inherent weight of the agent, taking into account the following aspects:

a) Inherent importance: Different devices can have different importance according to the services or functionality provided. For example, an alarm should have greater importance than a lamp or a microwave.
b) User´s preferences: different users could have different preferences. For example, for a teenager an iPod could have greater value than a microwave.

In spite of the importance of the weight of the agents, for the purpose of this work we are considering that all the agents have the same weight.

As it can be seen, this model is very similar to a state machine, in particular, Boolean networks [17]. However, in the case of Boolean networks the rules are homogeneous, and the connections are symmetric and time-independent.

In this paper we are not taking into account the role that time delays could play. However, it has been shown that in cellular automata with delays, the dynamics becomes more disordered and the information processing capacities are preserved and extended [10].

## B. *Interaction Networks*

Interaction Network (IN) is a digraph $(V, E)$ in which the vertex $v_k \in V$ is a pervasive intelligent device $A_k$ and $(v_i, v_j) \in E$ if the Boolean functions $\varphi_j$ or $\psi_j$ of the pervasive intelligent device $A_j$ depends on the state $s_i$ of the device $A_i$. An example of an Interaction Network can be seen of Fig. 1. Interaction Networks are able to represent the topological properties of the system. In particular, the presence of feedback or loops in the system is a necessary condition for the instabilities to emerge.

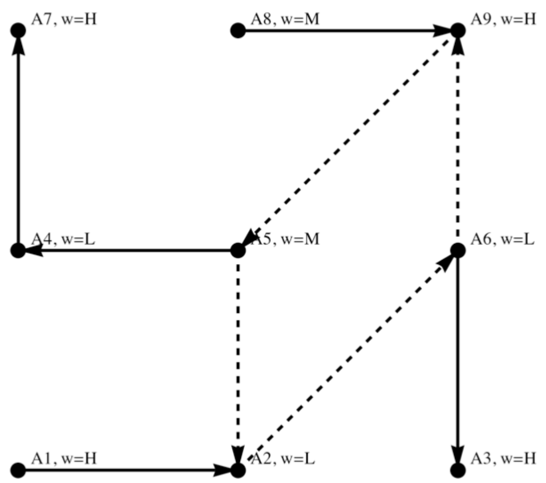



Figure 1. An Interaction Network showing a loop in dashed lines.

Based on these topological properties on the digraph, different strategies can emerge. In particular, the strategy based on locking a set of agents with less connectivity has been proven to be effective, [15][16]. However, in the case of complex topologies and in particular with coupled loops i.e., with common vertex between loops, this strategy (Instability Prevention System-INPRES) tends to overlock the system, as for each loop or feedback circuit found in the IN, there is a locked agent.

## III. C-INPRES

As it was mentioned earlier, INPRES provides a solution to the problem of cyclic instability. However, the number of locked agents is, in the case of coupled cycles, not optimum. The algorithm c-INPRES, presented in this paper, is a refinement that aims to minimize the number of agents locked.

In order to avoid the instabilities we must find the set of agents that when locked, stabilizes the system and minimizes the cost W. If unitary weights for the agents are considered, the number of agents is minimized, and in general the optimum is calculated using

$$\min\{W = \sum_{A_i \in \Delta} w_i\} \qquad (5)$$

where $\Delta = \{A_i\}$ is the set of agents that stabilize the system, and $w_i$ is the weight of agent $A_i$. In figure 2 we illustrate a high-level algorithm c-INPRES.

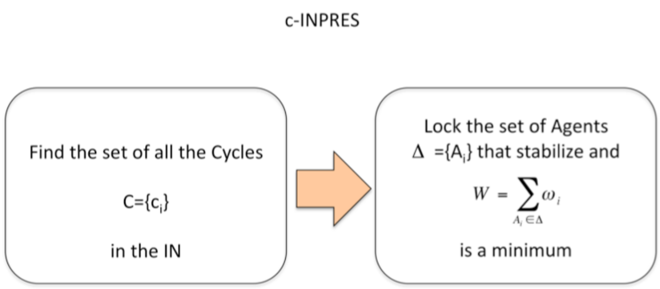



Figure 2. High level algorithm c-INPRES

As we can see in Figure 2, from all the sets of agents that could stabilize the system, we select those with the lowest W. In this paper we focus on minimizing the number of agents locked, in order to have a less-disabled system. With this intention, the concepts of weak and strong coupling are introduced in the next section.

### A. *Coupled Oscillators and Local Rules*

As we mentioned earlier, the presence of loops in an Interaction Network is a necessary condition for instabilities to emerge. However complex coupling in an IN can lead to unnecessary locking. Based on the local rules, the number of locked agents can be reduced, according to eq. 5.

Lets consider an environment with two cycles $C_1$ and $C_2$. There are two possibilities:

i. $C_1 \cap C_2 = \varnothing$: the two cycles are uncoupled

ii. $C_1 \cap C_2 \neq \varnothing$: the cycles share at least one node

For simplicity, lets suppose that in case ii) the two cycles share only one node, i.e., the two cycles are coupled. If that is the case, a closed trail will emerge. That closed trail will include all the nodes in $C_1$ and $C_2$. An example can be seen in Figure 3.

Two potential oscillators (i.e. two sets of nodes with feedback) coupled in one point are *weakly coupled* if the coupling node was assigned an OR rule; on the contrary, if the coupling node was assigned an AND rule, they would be *strongly coupled*. This is illustrated in Figure 3.

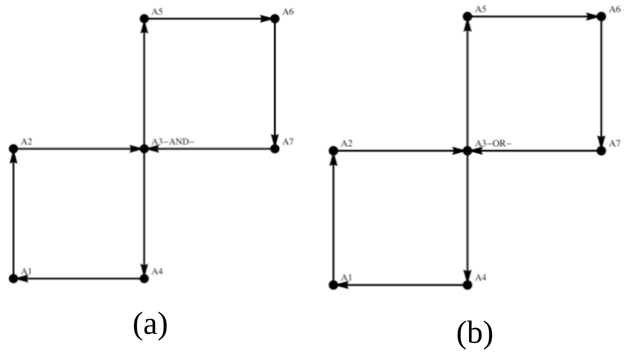



Figure 3. Interaction Networks with the same topology, but different rules in the shared node: a) strong coupling b) weak coupling.

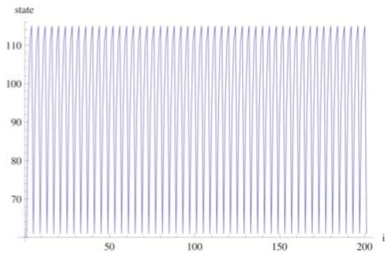
In Fig. 3 two cycles can be seen: 1-2-3-4-1 and 3-5-6-7-3. Additionally, the closed trail: 1-2-3-5-6-7-3-4-1 emerges.

Weak coupling lets either of the closed paths oscillate, or not, independently of the other (e.g. one might be oscillating, whilst the other is not). On the other hand, strong coupling implies that if any of the subsystem is oscillating, the other will also be oscillating. Using the result of simulation, this is illustrated in more detail in the following section.

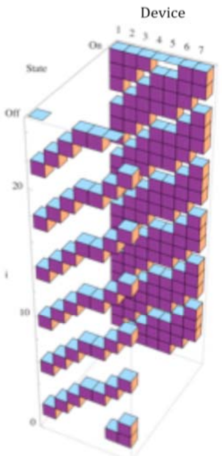
## IV. EXPERIMENTAL RESULTS

### A. *Weak coupling*

Let's consider a system with 7 nodes, with rules of behaviour given by the vector {1,0,0,0,1,0,1}, where a 1 is interpreted as an AND rule, and a 0 is interpreted as an OR rule. The topology given by {{1,2}, {2,3}, {3,4}, {3,5}, {4,1}, {5,6}, {6,7}, {7,3}} can be seen on Fig 3. The system has 2 simple cycles: {1,2,3,4,1} and {3,5,6,7,3} sharing node 3, which has been assigned an OR gate. In addition to these two simple cycles, there is a closed trail {1,2,3,5,6,7,3,4,1}. With the initial condition {0, 1, 1, 1, 1, 0, 0}, all the nodes oscillate, as is illustrated in Figure 4a. Additionally, in Figure 4b we are using the Multi-Dimensional Model (MDM) of Pervasive Computing Space to visualize the task or state of every single agent in the pervasive space, and its dynamics through time [17].



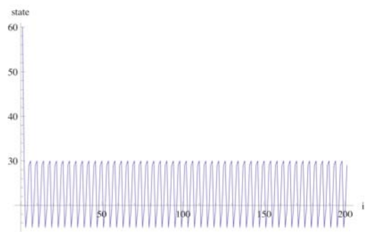

(a)



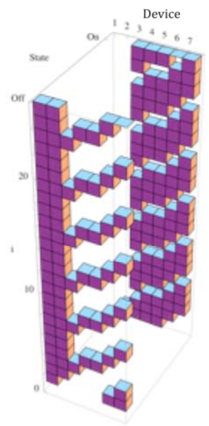

(b)

Figure 4. Weak coupling: without locking, all the devices are oscillating. In a) the state is the decimal representation of the binary state of the system. In b) the MDM [17] shows the time-device-state evolution of the system.

Locking device 1 stabilizes device 1 and 2; however, the nodes in cycle 3-5-6-7-3, together with node 4, are still oscillating, as is shown in Fig. 5.




(a)



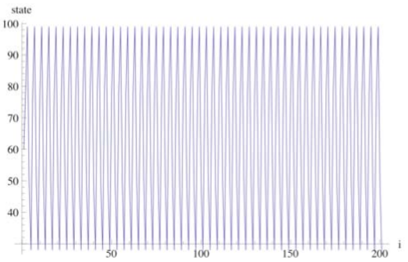

(b)

Figure 5. Weak coupling: after locking device 1, the system is still oscillating

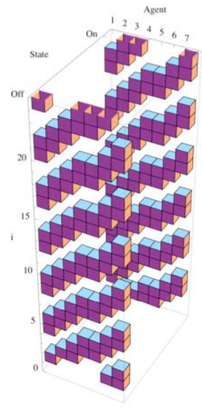
Thus, weak coupling allows oscillators to behave independently; one cycle can be oscillating disregarding the dynamics of the other. In the next section an example of strong coupling is presented.

### B. *Strong coupling*

A strong coupled system can be obtained by reassigning node 3, in the previous example, to an AND rule. Using the same initial conditions, the two subsystems oscillate, as shown in Figure 6.



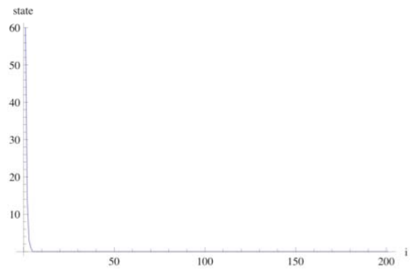

(a)



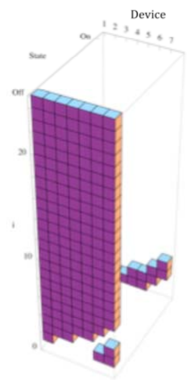

(b)

Figure 6. Strong coupling: the two subsystems are oscillating

Using the same principle as in the previous example, node 1 was locked. Under this condition, the system stabilized, as is shown in Figure 6.




(a)




(b)

Figure 7. Strong coupling: when node 1 is locked, oscillations are prevented.

## V. DISCUSSION AND FUTURE WORK

In this work we have shown the significance of coupling of the loops in an Interaction Network. In practical terms this means that the nature of cyclic instability depends both on the agent interconnection topology and rules within each agent. If the two sub-loops are strongly coupled, it is possible to lock only one node in order to prevent instabilities or oscillations in the whole system. With this, the system as a whole shows better performance in terms of the user's initial configuration, as the locking prevents instability being propagated throughout the network of devices.

By analyzing the local rules (detecting strongly coupled systems), together with the connectivity properties of the nodes in the interaction network, it is possible to lock fewer devices (thereby reducing the extent of the disabling effects of locking). Also, as this introduces more locking options, it may be possible to choose which devices to lock based on aspects such as convenience (in terms of the connectivity) or inherent importance (in the case of alarms or similar devices). Clearly, this process introduces additional computational overheads, in terms of calculations, but results in a less disabled system.

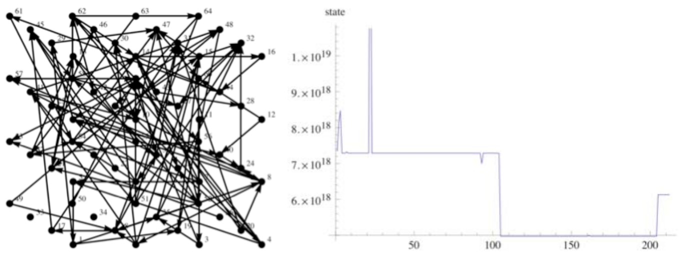
In this paper we have considered a constant weight associated to the pervasive devices. However, in general terms this is not completely true, as a device could have higher importance or priority during certain periods of time, therefore eq. 5 can be rewritten as

$$W = \sum_{j \in D} w_j(t) \tag{6}$$

Additionally, a more realistic scenario would involve dynamic and time-dependant rules:

$$\varphi, \psi : S \times t \rightarrow \{0,1\} \tag{7}$$

Apart from the inconvenience of disabling a system with the locking, there is a natural tendency of a system to suffer from self-locking: as the density of a system grows (the average number of cycles per node), the overlapping of cycles increases, and nodes will have in general more constraints on their behaviour and, at some point, some nodes will not be able to change their state. A system under those conditions is said to be self-locked. This can be illustrated in fig. 8, where the number of cycles were 586, with a density $\rho = 586/64 = 9.15625$



(a) (b)

Figure 8. An example of self-locked system.

It has been found experimentally that as the density approaches the maximum density, the system ceases to exhibit cyclic behaviour [17]. This is due to the presence of multiple constraints on the system (a type of destructive interference) However, such *self-locked systems* have little usability, as they cannot propagate information. It should be noted that the identification of the density threshold point $\Theta$ from where the system is not usable due to the self locking, is still an open problem for further research. Figure 9 summarises this.

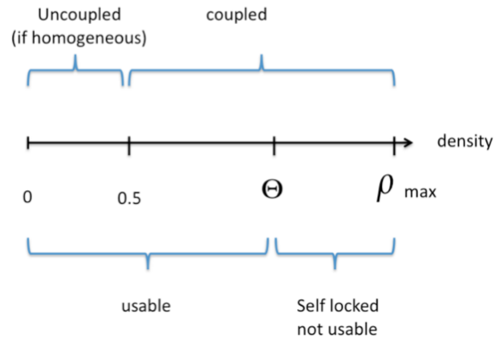



Figure 9. Density and usability of a system.

As it can be seen, there are several collateral conditions to be taken into account: size of the system, connectivity and density, and local rules. As the number of agents and their connections grows, so does the functionality (and usefulness) of the system increase, but so does coupling between loops. It is in this zone where our strategies and algorithms are useful. However, as the number agents and their connections continue to grow the coupling will exceed a threshold point $\Theta$, the system will self-lock, and hence become unusable.

Additionally, this problem can be modelled by a combinatorial optimization strategy, in order to minimize the oscillations using a metaheuristic algorithm.

While more research is needed to fully understand the behaviour of more complex topologies and devices with different levels of importance, our initial theoretical work and practical experimentation provide encouraging evidence that such unwanted cyclic instabilities can be understood, managed and even eliminated. We look forward to reporting on our progresses in these areas in future papers.

## REFERENCES


[1] Are Cars Too Complicated?, http://spectrum.ieee.org/podcast/green-tech/advanced-cars/are-cars-too-complicated

[2] Callaghan V, Colley M, Hagras H, Chin J, Doctor F and Clarke G, Programming iSpaces: a tale of two paradigms, in Steventon, A., Wright, S. (Eds.), Intelligent Spaces: The Application of Pervasive ICT part of the series Computer Communications and Networks, Springer, Heidelberg, Germany. pp. 162. 2005.

[3] Chin JS, Callaghan V and Clarke G, An End-User Programming Paradigm for Pervasive Computing Applications, The IEEE International Conference on Pervasive Services, Lyon, France, June 26-29, 2006 Page(s):325 – 328.

[4] De Carolis, B. and Cozzolongo, G. (2004) 'C@sa: intelligent home control and simulation', *Internat. J. Comput. Intelligence*, Vol. 1, No. 1, pp. 1-12.

[5] Hagras H, Callaghan V, Colley M, Clarke G, Pounds-Cornish A and Duman H, Creating an ambient-intelligence environment using embedded agents. IEEE on Intelligent Systems, Volume 19, Issue 6, Nov-Dec 2004 Page(s):12 – 20.

[6] Henricksen K, Indulska J, Rakotonirainy A, Modeling context information in pervasive computing systems, Proceedings of the First International Conference on Pervasive Computing, Zurich, Switzerland, August 26-28 2002, pg 167. Springer (2002).

[7] Holloway, Eric M., Lamont, Gary B., Self organized multi-agent entangled hierarchies for network security, Proceedings of the 11th Annual Conference Companion on Genetic and Evolutionary Computation Conference: Late Breaking Papers. Montreal, Quebec, Canada. Workshop Session: Evoutionary computation and multi-agent systems and simulations. Page: 2589-2596, 2009.

[8] Milner R., The Space and Motion of Communicating Agents. Cambridge University Press, 1st edition, 2009.

[9] Razzaque MA, Dobson S and Nixon P, Categorisation and Modelling of quality in context information, in Proceedings of the IJCAI 2005 Workshop on AI and Autonomic communications. Roy Sterrit, Simon Dobson and Mikhail Smirnov (ed). 2005.

[10] Rohlf, T, Jost, J., A new class of cellular automata: How spatio-temporal delays affect dynamics and improve computation. European Conference on Complex Systems. University of Warwick, United Kingdom. September 21-25 2009.

[11] Shehata M., Eberlein A., Fapojuwo A., Mohamed A. (2007) Managing Policy Interactions in KNX-based Smart Homes, 1st IEEE International Workshop on Development and Application of Knowledge-Based Software Tool (KASET), Proceedings of the 31st Annual IEEE International Computer Software and Applications Conference (COMPSAC 2007), July 23-27, 2007, Beijing, China

[12] Shehata, M., Eberlein, A., and Fapojuwo, A. 2007. Using semi-formal methods for detecting interactions among smart homes policies. *Sci. Comput. Program*. 67, 2-3 (Jul. 2007), 125-161. DOI= http://dx.doi.org/10.1016/j.scico.2006.11.002

[13] Strogatz SH, Exploring Complex Networks, Nature 410, 268-276. 2002.

[14] Weisbuch G, *Complex Systems*. Lecture Notes Volume II. Santa Fe Institute Studies in the Sciences of Complexity. 1991.

[15] Zamudio V and Callaghan V. Facilitating the Ambient Intelligence Vision: a Theorem, Representation and Solution for Instability in Rule-Based Multi-Agent Systems. Special Section on Agent Based System Challenges for Ubiquitous and Pervasive Computing. International Transactions on Systems Science and Applications. Vol. 4, No. 2, May 2008. pp. 108-121. Guest Editor: . J. Gaber

[16] Zamudio V and Callaghan V. Understanding and Avoiding Interaction Based Instability in Pervasive Computing Environments. International Journal of Pervasive Computing and Communications, Vol. 5 Issue 2, 2009. Page: 163-186. Guest Editors: Evi Syukur and Javier Garcia-Villalba.

[17] Zamudio V, PhD Dissertation. University of Essex. 2009.