

Autonomic Pervasive Computing: A Smart Mall Scenario Using Promise Theory

Mark Burgess and Siri Fagernes

Oslo University College, Norway
mark@iu.hio.no

Abstract. We present a simplified example of a pervasive computing scenario using promises to model the interaction policies between the agents. We examine how the autonomic nodes stabilize into a robust functional system in spite of their autonomous decision making. We use promises both as a means of modelling a potential specification and as a complementary eye glass for interpreting and understanding emergent behaviour. The analysis of promises reveals ‘faults’ in the policies, which prevent the collaborative functioning of the system as a whole. Successful interactions are the result of a bargaining process. The method of eigenvector centrality is used to locate the most important and vulnerable agents to the functioning of the system.

1 Introduction

Pervasive or ubiquitous computing considers the technologies and problems associated with having a high density of computing devices, owned by potentially many different users with different goals and interests[1–4]. Managing the complexity of an uncertain environment is a challenge and there is a lack of modelling techniques to address these uncertainties[5, 6].

Promise theory is a theory of autonomous components joined together in a logical network[5, 7]. It assumes a service view of interactions, in which each component node offers to play a part in a collaboration, by promising to constrain its behaviour according to a standardized set of issues. Since it deals with both autonomy and constraint, promise theory is of particular interest in connection with pervasive computing management[8, 9].

Promise theory builds distributed systems from autonomous components; in doing so, it forces one to confront the uncertainty that is inherent in distributed management, and is especially suited to the ad hoc network scenario. The concept of *voluntary cooperation* is central, since by their very nature autonomous agents are not externally controllable: they must be induced to cooperate by incentive alone.

Promise theory also provides a natural organizing principle by which to view the resulting logical network, as a typed theory of graphs. We consider the approach to be of great utility for understanding modern, distributed peer systems in potentially ad hoc networks.

In previous work[5, 7, 10], we have laid out the rationale behind promise theory from a theoretical viewpoint. In this paper, we seek a test model of sufficient size that we can see the usefulness of the promise approach, but which is also of sufficient limitation to be accessible to a reader without undue difficulty. We wish to show here how promise theory identifies the natural structure in a network without any prejudice of centralization, based only on a discovery of the autonomous promises made. If centralization of management is natural, this fact should emerge from a modelling framework without being assumed at the start.

We begin by describing a constructed scenario (a shopping mall that is ‘complicated enough’, but not too complicated), from the viewpoint of a privileged observer with prior knowledge: seeing the mall only as a collection of promises between the various agents. But, what if we did not have such an inside view of the roles of the agents, and could only collect the list of promises made by some means of discovery. Would that be sufficient to understand the policies and hence the behaviours in the distributed system?

To investigate this, we then proceed to imagine that we know only the promises themselves and attempt to derive and understand this snapshot of the mall purely from a discovery of the promises. Although it is not clear that anyone in the mall would have such an overview, this gives us a mode of simulation for understanding the relationships between autonomous individuals.

2 A Smart Mall

We consider the future shopping mall as our test scenario. A public meeting place, like a shopping mall full of peers interacting freely by a mixture of private and public communication channels, is a potentially difficult scenario to understand, as it is formed of an ad hoc mixture of fixed and transient infrastructure. Although we pick a specific scenario, our computations are not really tied to this: any scenario in which a service-oriented interaction pattern prevails can be analyzed using our method. However, we think that the pervasive shopping mall is naturally more ad hoc than a “designed” system, and hence we use this as a plausible example in which the analysis crosses the boundary from management by design, to merely coping.

In the mall, a multitude of mobile devices roams in a changing environment, with many conflicting interests. The shops and the mall itself are part of a fixed infra-structure backbone network, while customers, laden with personal mobile devices move around, interacting freely with a variety of shops and retailers over short-distance links (e.g. Bluetooth), making and breaking connections and allegiances. Some conversations between parties (either verbal or electronic) are authoritative, while others are informal: there is thus a divergence of trust between the individual players.

The mall consists of a number of types of individual: a ‘mall management’, an Internet Service Provider (ISP), a number of shops, and a number of customers. These are all network capable entities. To offer their customers a greater level of service, the mall has an agreement with an Internet service provider (ISP), which

provides a wireless network throughout the entire mall. This can be accessed by any customer carrying an appropriate cellular device. Customers and shops and communicate *ad hoc* by short-range private channels; external communication must be routed via the ISP.

Inside the mall, we find different logical *regions* with different network coverage.

- *A global network.* This is a wireless network operated by the mall. It network can be used by customers, by registering for a username and password. Through this network the customers can obtain a variety of services, for instance download information about bus schedules, order bus tickets, download a map of the mall with information on the different shops and so on, or communicate with the outside world. To pay for their services, the customers can either use a credit card, or earn virtual credits which can be used as payment within the mall.
- *Local networks.* Each shop has its own wireless network for customer benefit, which is open and offers a limited number of services. Customers can download available information about current special offers, and regular prices on goods which they are interested in. This local network is only available to customers within the physical wireless range of the store.
- *Ad hoc networks.* Customers have the ability to communicate directly, using a low power network such as BlueTooth for instance, as long as they are within a certain range of each other. This way, customers can send and receive messages without loading the mall network or paying them for this service.

The networks listed above can be utilised for several benefits for each actor in this scenario. Particularly, we can group the network traffic in two main groups:

1. *Information.* The customers might find it useful to be able to download information about bus schedules, maps of the shops, product information and so on. The providers of this information, i.e. the shops and the mall itself, this is a basic service to the customers, and has only a goodwill value.
2. *Commercial messages.* Shops can have special deals, to attract more customers, and hence increase their turnover. For the customers, this kind of information might be a source of annoyance, as their mobile devices is flooded with unwanted data (spam).

3 The promises and the autonomous agents

The autonomous agents in this scenario are the set of individual entities that comprise the mall environment and who have an interest in one another's behaviour. Each of these agents is represented by a *node* in a *promise graph*, and there is a labelled, directed edge (arrow) between node *i* and node *j* for each promise made by node *i* to node *j*. They are categorized as follows:

- The ISP (node 1)
- The mall (node 2)
- The shops (nodes 3,4,5)
- The customers (nodes 6-15)

A promise is a labelled edge in a graph, of the form

$$a \xrightarrow{\tau, \chi} b \quad (1)$$

where a, b are agents or nodes in the graph, and τ is a *type* of promise. χ represents the constraint that node a promises to uphold with regard to b ; b is informed of this, but has no other influence over a . See ref. [5] for an introduction. The promise bodies τ, χ represent behavioural constraints at the outset, and can later be interpreted in terms of a *value* or *currency* that can be exchanged between the actors. We shall not explore the details of the promise body in this paper, in order to keep matters simple.

We can now represent the promises between the different agents. We label the promise types $\tau \in \{q, d, f, lf, r, i\}$. The roles of these promises is as follows:

- q : This promise asserts a certain Quality of Service (QoS) in network performance, and is used
 - From ISP to the mall
 - From the mall to the customers
- d : A promise to download messages from another agent. Customers agree to download a certain amount of advertising from the shops and from the mall, in return for some kind of reward. Customers can also spread the messages to other customers in return for reward from the mall on production of a transfer receipt.
- r : A promise to reward forwarding of messages to other customers. The mall and the individual shops promise to reward customers with points or credits for forwarding commercial messages to other customers by ad hoc means (this could occur while they are outside the mall, e.g. in their home neighbourhood).
- i : A promise to offer relevant information from the mall and shops to the customers in the environment.
- f : This is a promise from the customers to the mall/the shops regarding forwarding downloaded messages to other customers. It could involve the transmission of forwarding receipts, for instance.
- lf : This promise is made between the customers. It acts as a courtesy treaty limiting the amount of messages offered for forwarding to relevant, official messages, plus some personal advertising. Without such a treaty, the amount of personal junk messages might overload them and they could stop choosing the exchange messages altogether. Note, that due to the autonomy of the nodes, no customer can force messages onto another. The exchange must be by mutual consent[11].

Fig. 2 shows the full set of promises made by the agents in the mall, for completeness. Their types and the engineering dimensions used in their constraints are summarized in table 1.

Promise body	Units	Category
$q, q \geq q_n$	Mb/sec	Service
$r, r = R$	Credits per message	Service
$i, i \in \{T, F\}$	True/False	Service
$d, d \in \{T, F\}$	Messages per hour	Use-message
$f, f \geq f_n$	Messages per hour	Service
$lf, lf \leq l_n$	Messages per hour	Use-value

Table 1. A summary of promise types.

We illustrate the promises, as graphs, in figs. 3-7. Even with the small system of fifteen nodes, some of these graphs are relatively busy; however, they help to show the mutual relationships in the distributed system. We note that fig. 4 covers both d and f promise types, since the arrows are the same in each case. (This makes sense, since the promises to download and forward messages to other customers are both contracts made with the shops and the mall. They have formally different types, with different constraints but we can save ourselves an illustration by noting that the topology is the same.)

In a similar vein, it is worth noting that customers do not promise anyone to download advertising from one another. If they do this, they do so at their own option, as a matter of *voluntary cooperation*.

4 Graphical analysis

We have constructed the typed promise graph based on this deliberate scenario, but we can now analyze it, as a list of discovered promises, without prejudice of its original intent.

There are two reasons why we might want to do this: i) verification, i.e. for understanding the planning of promises required to bind a group to a distributed objective; or ii) forensic reconstruction of the scenario being played out. The list of promises we have discovered might or might not be complete, or even consistent, in relation to the objective observer's impression of the network's goals. That remains for us to determine.

Moreover, in a more extensive scenario, we might simply discover this list of promises without any particular knowledge of the real world origin. What we refer to as a 'shop' might bring with it certain associations, such as a building with a facade and shop employees; however, a shop might simply be a server process, making and performing all the same promises as a physical shop from

Node 1 (the ISP)	$\left. \begin{array}{c} n_1 \xrightarrow{q, q \geq q_1} n_2 \\ n_2 \xrightarrow{r, r=R_1} n_6 \\ n_2 \xrightarrow{r, r=R_2} n_7 \\ n_2 \xrightarrow{r, r=R_3} n_8 \\ n_2 \xrightarrow{r, r=R_4} n_9 \\ n_2 \xrightarrow{r, r=R_5} n_{10} \\ n_2 \xrightarrow{r, r=R_6} n_{11} \\ n_2 \xrightarrow{r, r=R_7} n_{12} \\ n_2 \xrightarrow{r, r=R_8} n_{13} \\ n_2 \xrightarrow{r, r=R_9} n_{14} \\ n_2 \xrightarrow{r, r=R_{10}} n_{15} \end{array} \right\}$
Node 2 (the Mall)	$\left. \begin{array}{c} n_2 \xrightarrow{q, q \geq q_2} n_3 \\ n_2 \xrightarrow{q, q \geq q_3} n_4 \\ n_2 \xrightarrow{q, q \geq q_4} n_5 \end{array} \right\}, \left. \begin{array}{c} n_2 \xrightarrow{i, i \in \{T, F\}} n_6 \\ n_2 \xrightarrow{i, i \in \{T, F\}} n_7 \\ n_2 \xrightarrow{i, i \in \{T, F\}} n_8 \\ n_2 \xrightarrow{i, i \in \{T, F\}} n_9 \\ n_2 \xrightarrow{i, i \in \{T, F\}} n_{10} \\ n_2 \xrightarrow{i, i \in \{T, F\}} n_{11} \\ n_2 \xrightarrow{i, i \in \{T, F\}} n_{12} \\ n_2 \xrightarrow{i, i \in \{T, F\}} n_{13} \\ n_2 \xrightarrow{i, i \in \{T, F\}} n_{14} \\ n_2 \xrightarrow{i, i \in \{T, F\}} n_{15} \end{array} \right\}$
Node 3 (Shop1)	$\left. \begin{array}{c} n_3 \xrightarrow{q, q \geq q_5} n_6 \\ n_3 \xrightarrow{q, q \geq q_6} n_7 \\ n_3 \xrightarrow{q, q \geq q_7} n_8 \end{array} \right\}, \left. \begin{array}{c} n_3 \xrightarrow{r, r=R_{11}} n_6 \\ n_3 \xrightarrow{r, r=R_{12}} n_7 \\ n_3 \xrightarrow{r, r=R_{13}} n_8 \end{array} \right\}, \left. \begin{array}{c} n_3 \xrightarrow{i, i \in \{T, F\}} n_6 \\ n_3 \xrightarrow{i, i \in \{T, F\}} n_7 \\ n_3 \xrightarrow{i, i \in \{T, F\}} n_8 \end{array} \right\}$
Node 4 (Shop2)	$\left. \begin{array}{c} n_3 \xrightarrow{q, q \geq q_8} n_{15} \\ n_3 \xrightarrow{r, r=R_{14}} n_{15} \\ n_3 \xrightarrow{i, i \in \{T, F\}} n_{15} \end{array} \right\}$
Node 5 (Shop3)	$\left. \begin{array}{c} n_5 \xrightarrow{q, q \geq q_9} n_9 \\ n_5 \xrightarrow{q, q \geq q_{10}} n_{10} \\ n_5 \xrightarrow{q, q \geq q_{11}} n_{11} \end{array} \right\}, \left. \begin{array}{c} n_5 \xrightarrow{r, r=R_{15}} n_9 \\ n_5 \xrightarrow{r, r=R_{16}} n_{10} \\ n_5 \xrightarrow{r, r=R_{17}} n_{11} \end{array} \right\}, \left. \begin{array}{c} n_5 \xrightarrow{i, i \in \{T, F\}} n_9 \\ n_5 \xrightarrow{i, i \in \{T, F\}} n_{10} \\ n_5 \xrightarrow{i, i \in \{T, F\}} n_{11} \end{array} \right\}$
Node 6 (Customer)	$\left. \begin{array}{c} n_6 \xrightarrow{d, d \in \{T, F\}} n_2 \\ n_6 \xrightarrow{d, d \in \{T, F\}} n_3 \\ n_6 \xrightarrow{d, d \in \{T, F\}} n_4 \\ n_6 \xrightarrow{d, d \in \{T, F\}} n_5 \end{array} \right\}, \left. \begin{array}{c} n_6 \xrightarrow{f, f \geq f_1} n_2 \\ n_6 \xrightarrow{f, f \geq f_2} n_3 \\ n_6 \xrightarrow{f, f \geq f_3} n_4 \\ n_6 \xrightarrow{f, f \geq f_4} n_5 \end{array} \right\}, \left. \begin{array}{c} n_6 \xrightarrow{lf, lf \leq l_1} n_7 \\ n_6 \xrightarrow{lf, lf \leq l_1} n_8 \end{array} \right\}$
Node 7 (Customer)	$\left. \begin{array}{c} n_7 \xrightarrow{d, d \in \{T, F\}} n_2 \\ n_7 \xrightarrow{d, d \in \{T, F\}} n_3 \\ n_7 \xrightarrow{d, d \in \{T, F\}} n_4 \\ n_7 \xrightarrow{d, d \in \{T, F\}} n_5 \end{array} \right\}, \left. \begin{array}{c} n_7 \xrightarrow{f, f \geq f_5} n_2 \\ n_7 \xrightarrow{f, f \geq f_6} n_3 \\ n_7 \xrightarrow{f, f \geq f_7} n_4 \\ n_7 \xrightarrow{f, f \geq f_8} n_5 \end{array} \right\}, \left. \begin{array}{c} n_7 \xrightarrow{lf, lf \leq l_3} n_6 \\ n_7 \xrightarrow{lf, lf \leq l_4} n_8 \end{array} \right\}$
Node 8 (Customer)	$\left. \begin{array}{c} n_8 \xrightarrow{d, d \in \{T, F\}} n_2 \\ n_8 \xrightarrow{d, d \in \{T, F\}} n_3 \\ n_8 \xrightarrow{d, d \in \{T, F\}} n_4 \\ n_8 \xrightarrow{d, d \in \{T, F\}} n_5 \end{array} \right\}, \left. \begin{array}{c} n_8 \xrightarrow{f, f \geq f_9} n_2 \\ n_8 \xrightarrow{f, f \geq f_{10}} n_3 \\ n_8 \xrightarrow{f, f \geq f_{11}} n_4 \\ n_8 \xrightarrow{f, f \geq f_{12}} n_5 \end{array} \right\}, \left. \begin{array}{c} n_8 \xrightarrow{lf, lf \leq l_5} n_6 \\ n_8 \xrightarrow{lf, lf \leq l_6} n_7 \end{array} \right\}$
Node 9 (Customer)	$\left. \begin{array}{c} n_9 \xrightarrow{d, d \in \{T, F\}} n_2 \\ n_9 \xrightarrow{d, d \in \{T, F\}} n_3 \\ n_9 \xrightarrow{d, d \in \{T, F\}} n_4 \\ n_9 \xrightarrow{d, d \in \{T, F\}} n_5 \end{array} \right\}, \left. \begin{array}{c} n_9 \xrightarrow{f, f \geq f_{13}} n_2 \\ n_9 \xrightarrow{f, f \geq f_{14}} n_3 \\ n_9 \xrightarrow{f, f \geq f_{15}} n_4 \\ n_9 \xrightarrow{f, f \geq f_{16}} n_5 \end{array} \right\}, \left. \begin{array}{c} n_9 \xrightarrow{lf, lf \leq l_7} n_{10} \\ n_9 \xrightarrow{lf, lf \leq l_8} n_{11} \end{array} \right\}$
Node 10 (Customer)	$\left. \begin{array}{c} n_{10} \xrightarrow{d, d \in \{T, F\}} n_2 \\ n_{10} \xrightarrow{d, d \in \{T, F\}} n_3 \\ n_{10} \xrightarrow{d, d \in \{T, F\}} n_4 \\ n_{10} \xrightarrow{d, d \in \{T, F\}} n_5 \end{array} \right\}, \left. \begin{array}{c} n_{10} \xrightarrow{f, f \geq f_{17}} n_2 \\ n_{10} \xrightarrow{f, f \geq f_{18}} n_3 \\ n_{10} \xrightarrow{f, f \geq f_{19}} n_4 \\ n_{10} \xrightarrow{f, f \geq f_{20}} n_5 \end{array} \right\}, \left. \begin{array}{c} n_{10} \xrightarrow{lf, lf \leq l_9} n_9 \\ n_{10} \xrightarrow{lf, lf \leq l_{10}} n_{11} \end{array} \right\}$
Node 11 (Customer)	$\left. \begin{array}{c} n_{11} \xrightarrow{d, d \in \{T, F\}} n_2 \\ n_{11} \xrightarrow{d, d \in \{T, F\}} n_3 \\ n_{11} \xrightarrow{d, d \in \{T, F\}} n_4 \\ n_{11} \xrightarrow{d, d \in \{T, F\}} n_5 \end{array} \right\}, \left. \begin{array}{c} n_{11} \xrightarrow{f, f \geq f_{21}} n_2 \\ n_{11} \xrightarrow{f, f \geq f_{22}} n_3 \\ n_{11} \xrightarrow{f, f \geq f_{23}} n_4 \\ n_{11} \xrightarrow{f, f \geq f_{24}} n_5 \end{array} \right\}, \left. \begin{array}{c} n_{11} \xrightarrow{lf, lf \leq l_{11}} n_9 \\ n_{11} \xrightarrow{lf, lf \leq l_{12}} n_{10} \end{array} \right\}$

Fig. 1. The list of promises grouped according to sender and type (completed over).

Node 12 (Customer)	$\left. \begin{array}{l} n_{12} \xrightarrow{d, d \in \{T, F\}} n_2 \\ n_{12} \xrightarrow{d, d \in \{T, F\}} n_3 \\ n_{12} \xrightarrow{d, d \in \{T, F\}} n_4 \\ n_{12} \xrightarrow{d, d \in \{T, F\}} n_5 \end{array} \right\}, \left. \begin{array}{l} n_{12} \xrightarrow{f, f \geq f_{25}} n_2 \\ n_{12} \xrightarrow{f, f \geq f_{26}} n_3 \\ n_{12} \xrightarrow{f, f \geq f_{27}} n_4 \\ n_{12} \xrightarrow{f, f \geq f_{28}} n_5 \end{array} \right\}, \left. \begin{array}{l} n_{12} \xrightarrow{lf, lf \leq l_{11}} n_{13} \\ n_{12} \xrightarrow{lf, lf \leq l_{12}} n_{14} \end{array} \right\}$
Node 13 (Customer)	$\left. \begin{array}{l} n_{13} \xrightarrow{d, d \in \{T, F\}} n_2 \\ n_{13} \xrightarrow{d, d \in \{T, F\}} n_3 \\ n_{13} \xrightarrow{d, d \in \{T, F\}} n_4 \\ n_{13} \xrightarrow{d, d \in \{T, F\}} n_5 \end{array} \right\}, \left. \begin{array}{l} n_{13} \xrightarrow{f, f \geq f_{29}} n_2 \\ n_{13} \xrightarrow{f, f \geq f_{30}} n_3 \\ n_{13} \xrightarrow{f, f \geq f_{31}} n_4 \\ n_{13} \xrightarrow{f, f \geq f_{32}} n_5 \end{array} \right\}, \left. \begin{array}{l} n_{13} \xrightarrow{lf, lf \leq l_{15}} n_{12} \\ n_{13} \xrightarrow{lf, lf \leq l_{16}} n_{14} \end{array} \right\}$
Node 14 (Customer)	$\left. \begin{array}{l} n_{14} \xrightarrow{d, d \in \{T, F\}} n_2 \\ n_{14} \xrightarrow{d, d \in \{T, F\}} n_3 \\ n_{14} \xrightarrow{d, d \in \{T, F\}} n_4 \\ n_{14} \xrightarrow{d, d \in \{T, F\}} n_5 \end{array} \right\}, \left. \begin{array}{l} n_{14} \xrightarrow{f, f \geq f_{33}} n_2 \\ n_{14} \xrightarrow{f, f \geq f_{34}} n_3 \\ n_{14} \xrightarrow{f, f \geq f_{35}} n_4 \\ n_{14} \xrightarrow{f, f \geq f_{36}} n_5 \end{array} \right\}, \left. \begin{array}{l} n_{14} \xrightarrow{lf, lf \leq l_{17}} n_{12} \\ n_{14} \xrightarrow{lf, lf \leq l_{18}} n_{13} \end{array} \right\}$
Node 15 (Customer)	$\left. \begin{array}{l} n_{15} \xrightarrow{d, d \in \{T, F\}} n_2 \\ n_{15} \xrightarrow{d, d \in \{T, F\}} n_3 \\ n_{15} \xrightarrow{d, d \in \{T, F\}} n_4 \\ n_{15} \xrightarrow{d, d \in \{T, F\}} n_5 \end{array} \right\}, \left. \begin{array}{l} n_{15} \xrightarrow{f, f \geq f_{37}} n_2 \\ n_{15} \xrightarrow{f, f \geq f_{38}} n_3 \\ n_{15} \xrightarrow{f, f \geq f_{39}} n_4 \\ n_{15} \xrightarrow{f, f \geq f_{40}} n_5 \end{array} \right\}$

Fig. 2. The list of promises grouped according to sender and type (see previous page).

a customer's personal device. Our analysis will avoid such prejudice and group entities according to their function. We might then use the following analysis as a forensic method of reconstructing its function.

We emphasize yet again that a band of autonomous agents has no guaranteed purpose or guaranteeable objective, in the traditional sense: it can only have an emergent purpose, if the agents themselves choose to collaborate to a sufficient and appropriate degree. Hence, our question becomes: does this scenario behave like a shopping mall with the economic model suggested above?

4.1 Roles

We begin by identifying the roles in the graph. There are two approaches here. The first is based in ref. [7] and the latter on ref. [12], which we shall address later. From ref. [7], we take a node that is pointed to by many nodes of a type τ to be a τ -role. Identifying all such structures in fig. 2, or the illustrations, we find that:

- Nodes 2,3,4,5 play d -roles, i.e. they are recipients of multiple promises and form an implicit group of those nodes which make such promises to them.
- Promises d, f to node 2,3,4,5 all come from group $\mathcal{G}_a = \{6, 7, 8, 9, 10, 11, 12, 13, 14, 15\}$. Hence these are all *groups/roles by association*. They play a specific role in the promise graph and hence they have a noteworthy meaning.
- Node 2 makes a q -promise to appointed group $\mathcal{G}_b = \{3, 4, 5\}$, hence these form a q -role. Similarly node 1 makes the same promise to node 2. Other q -roles include $\mathcal{G}_c = \{9, 10, 11\}$ $\mathcal{G}_d = \{6, 7, 8\}$ $\mathcal{G}_e = \{15\}$. Thus nodes 1,2,3,4,5 have roles in common by appointing other roles.

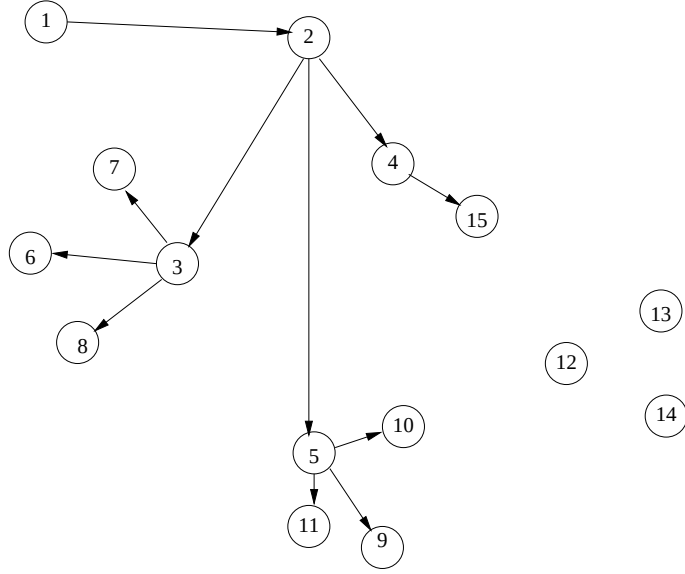


Fig. 3. The q -promise graph without labels.

- 6,7,8,9,10,11,12,13,14,15 are d,f recipients.
- with regard to lf -promises, the following complete subgraphs are self-appointed groups: $\mathcal{G}_f = \{6, 7, 8\}$, $\mathcal{G}_g = \{9, 10, 11\}$, $\mathcal{G}_h = \{12, 13, 14\}$.

We see the following association:

- 1,2,3,4,5 are all q -service providers. However these have the dependency structure $(1, (2, (3, 4, 5)))$.

The roles are thus summarized in fig. 9. We derive this from a random list of promises, in no particular order and extract the schematic behaviour accordingly. It is now easy to identify these groups with the ISP, mall, shops and customers from the original scenario.

4.2 Games

If we define a game, as in ref. [10], to be a back-to-back bilateral promise, then we obtain the games in fig. 8. This undirected graph yields its own structure, from ref. [12], and thus we identify topological roles: $\mathcal{G}_i = \{3, 6, 7, 8\}$, $\mathcal{G}_j = \{5, 9, 10, 11\}$, $\mathcal{G}_k = \{12, 13, 14\}$ and $\mathcal{G}_l = \{4, 15\}$.

From fig. 8 we see that, of all the nodes in the graph, only node 1 does not participate in a bilateral promise. This violates the principal of remuneration for a stable relationship[10], and is therefore a problem to be rectified (see below).

Finally, we note that there are no ‘cooperative groups’ in this graph, in the sense of ref. [7]. That is, there are no groups who promise to cooperate in

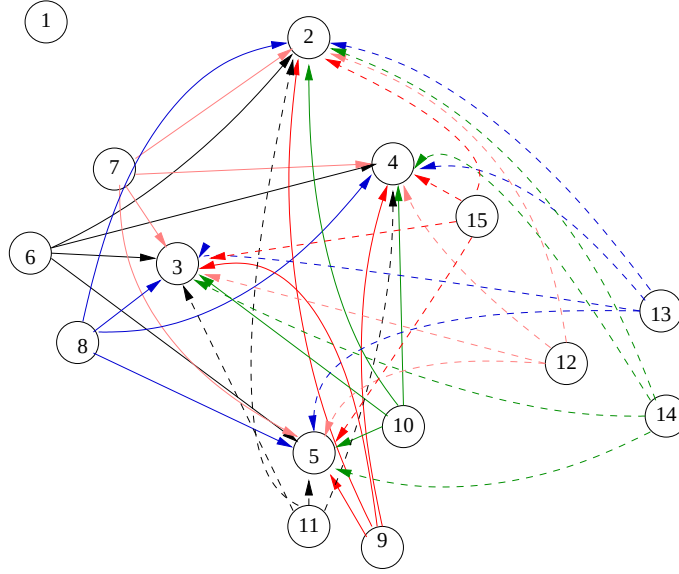


Fig. 4. The d/f -promise graph without labels. The graphs for d and f are the same.

their behaviour so as to behave as a seamless, indistinguishable group, working together. This is a natural feature of the present model, where the focus is on individual autonomy rather than in cooperative collaboration by design.

5 Stability and games

Any working system requires a sufficiently stable core to allow it to have predictable consequences[13]. In a collection of autonomous individuals, predictability is related both to environmental uncertainties (in a traditional reliability sense) as well as to the likelihood that nodes will cooperate. Cooperation is, in turn, related to what value autonomous agents derive from the relationships with their neighbours. Stability in such a relationship is known to require mutual benefit. This tells us immediately that sources and sinks in the type-free or common-currency promise graph are potentially unstable areas. Node 1 is clearly in such a position, in our model as described.

The notion of a bargaining game has been used as a reasonable means of prediction of such cooperation[14, 15, 10, 16]. Having identified the games, in terms of back-to-back promises pairs, as defined in ref. [10] (see fig. 11). One identifies the stability of such games by relating them to an equivalent cooperative dilemma[15] (see fig. 10), in which agents decide whether it is worth their while to cooperate with their agreement, or whether they rather ought to ‘defect’ from their promises.. Such a game is characterized by four values, for which one stability requires $T > R > P > S$. If the each agent’s perceived value for cooperation

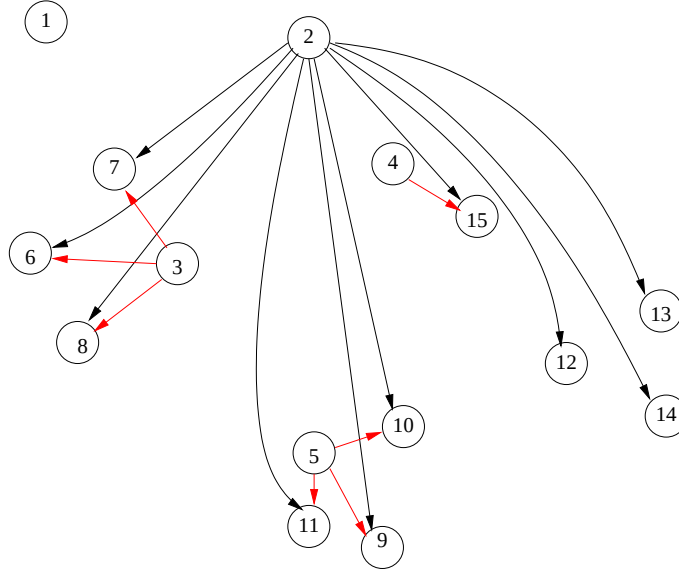


Fig. 5. The r -promise graph without labels.

and non-cooperation (defection) satisfies this inequality, there is a likelihood for stability in the promise relationship, else it is unlikely that the promises will be kept.

It is worth noting that, of all the nodes, the ISP node 1 does not participate either implicitly or explicitly in any game. It is purely ‘exploited’ in the sense of ref. [10]. This is a weakness in the current model. What is missing here is the remuneration node 1 receives for its own promises. In a real scenario, this would doubtless be ‘real’ money, and we henceforth add such a promise to the model.

Having established the model as a collection of bargaining games, the next step is to analyse the typical interaction policies between the agents, to understand whether the games lead to lasting relationships, or whether they are likely to fail; i.e. one can use a basic two-player game model to estimate the likely stability of the interaction between the agents who have games between them.

To cover all incidents, we will need a separate game matrix for each interaction, i.e. we need to define what kind of outcome each agent will receive in the different possible scenarios. For simplicity, we consider only the different *types* of game, parameterized in a general way.

5.1 ISP and mall

The relationship between the ISP and the (centralised) mall is rather uncomplicated, the main service present is of course the Internet (and phone) connection that the mall depends on in order to provide the necessary service for its cus-

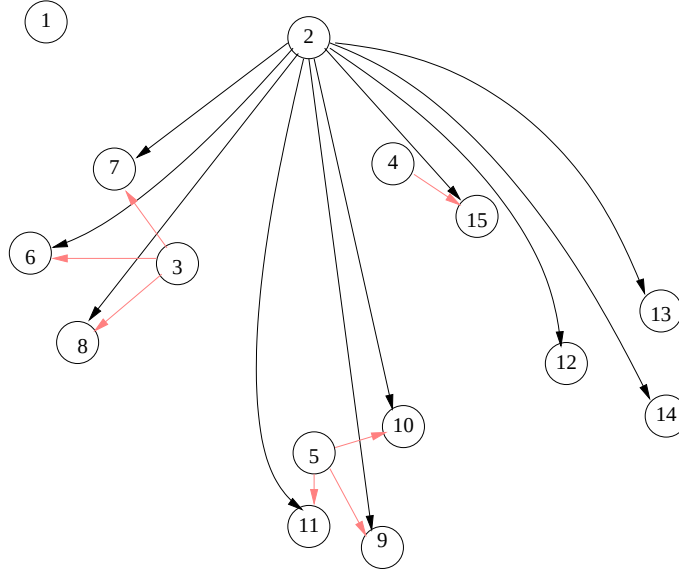


Fig. 6. The i -promise graph without labels.

ISP, Mall	Mall cooperate	Mall defect
ISP cooperate	$P_s - C_s, S - P_s$	$0 - C_s, S$
ISP defect	$P_s, C_r - P_s$	$0, C_r$

Table 2. Payoff matrix for the ISP-Mall game

tomers. The only responsibility the mall has to the ISP is to actually *pay* for the services provided by the ISP. The payoff matrix for this game is given in 2.

In our example we have considered only *two* parameters in the payoff-model:

P_s :	The price or fee the mall has to pay for service.
C_s :	The cost for the ISP as a consequence of providing the service.
S :	The agreed level of service. This parameter is a numerical unit, associated with the <i>value</i> of the service.
C_r :	The cost incurred by not having a sufficient Internet service/connection.

The bargaining interpretation of moves is:

1. *Both cooperate.* The ISP delivers the agreed level of service, while the mall pays the expected fee.

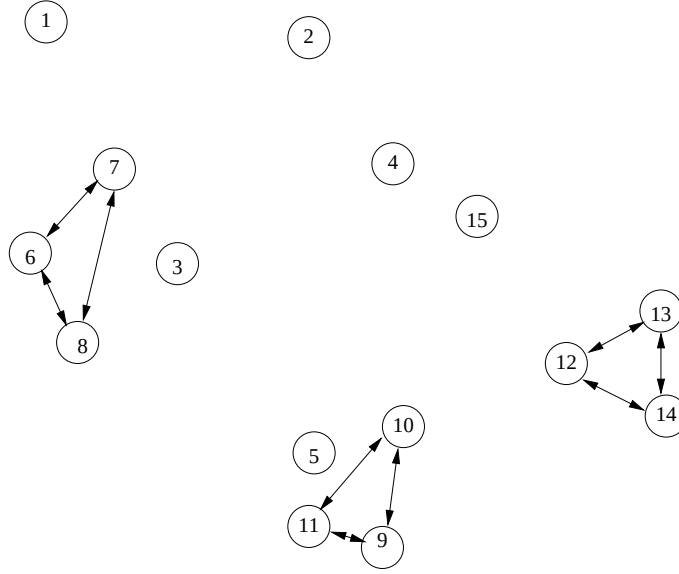


Fig. 7. The lf -promise graph without labels.

2. *The mall cooperates while the ISP defects.* This means the mall pays the agreed price while the ISP fail to deliver the agreed service. For simplicity, we assume that the value of the service is 0.
3. *The ISP cooperates while the Mall defects.* This means that the ISP delivers the agreed service while the mall fails to pay. Again we assume the value of the payment is 0.
4. *Both defect.* The mall does not receive any service, and the ISP does not receive any payment.

5.2 Customer and mall

The mall wishes to reach each customer with certain information, for instance by sending out messages to all customers that are *online* on the malls network. The customers however, are often suffering from limited battery power and memory space, and hence they not very interested in being flooded by unwanted information.

Customers want to be able to receive the information they need *on demand*, in addition to having a subsidised Internet and phone connection. Providing the connection/access (through an ISP) is a pure cost to the mall, but might also increase the customers usage of their networks, and hence increased the probability that they will receive and read the messages distributed by the mall. In addition to being a significant cost, giving users access to the Internet also impose the risk of the users actually taking advantage of the mall network by

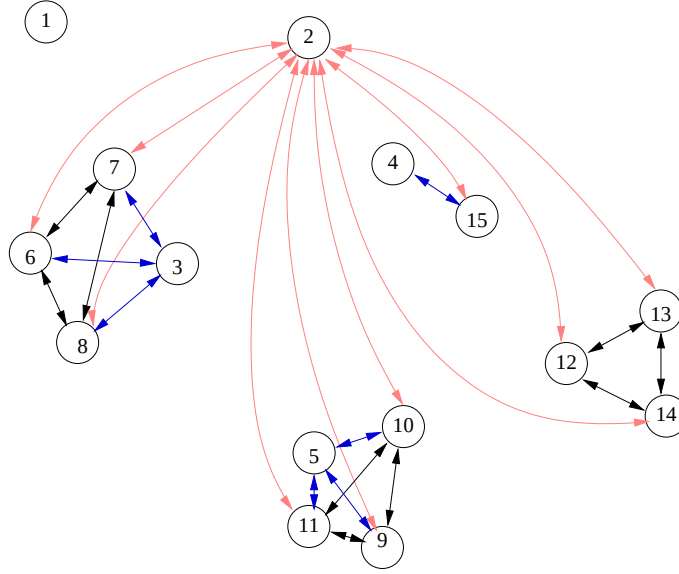


Fig. 8. The graph of various bargaining games within the network..

consuming too much. These scenarios include aspects that are not necessarily related, and hence we have chosen to represent them by two different game matrices.

The first case is solved by the mall promising the customer to keep the flow of messages within a defined limit, while the customer promises to download the messages as they arrive. The game matrix for this case is given in 3. The parameters used in this model are

$R_{\max}$:	The maximum limit for the number of messages per hour sent to a customer(mobile host)
R_{∞} :	Rate of messages per hour sent to a customer(mobile host) that exceeds the maximum rate $R_{\max}$
V_{msg} :	The estimated value of sending <i>one</i> message to a customer. This value is a result of the saved costs from not needing to use billboards etc for diverse information.
C_{msg} :	The associated cost (for the mall) of sending one message.
I :	The associated value of receiving a message (information value).
C_r :	The cost (in energy) associated with receiving a message

The agents' move combinations are:

CC The Mall is able to reach the customer, and the customer is not bothered too much by overflow of messages. The mall has the gain $(V_{msg} - C_{msg})R_{\max}$,

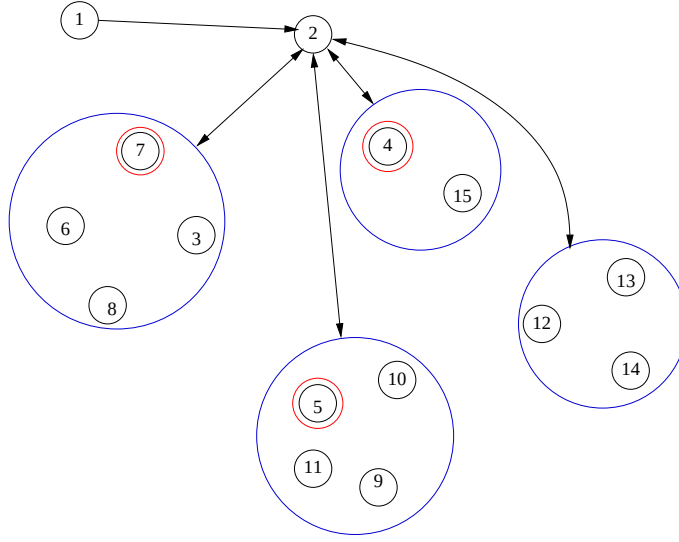


Fig. 9. A simplified schematic graph based on the derived roles.

which is the net value of sending a message times the number of messages sent per defined time interval. The customer, on the other hand, receives the value $(I - C_r)R_{\max}$.

- CD** *The customer cooperates.* The customer stays online, while the mall exploits this to send a larger amount of messages than the customer appreciates. The mall has the gain $(V_{msg} - C_{msg})R_{\infty}$, while the customer receives $(I - C_r)R_{\infty}$.
- DC** *The mall cooperates.* The mall keeps its message rate below a sensible level, but the customer still does not bother to stay online. The mall has a net loss equal to the cost of sending the messages, while the customer has a gain of zero.

	C ₁	D ₁
C ₂	(R,R)	(S,T)
D ₂	(T,S)	(P,P)

Fig. 10. The basic payoff matrix structure for a dilemma game.

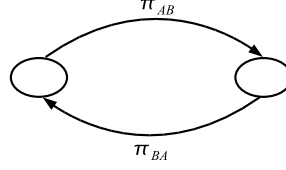


Fig. 11. Node A has made a promise (π_{AB}) to node B , and node B has made a promise (π_{BA}) to node A .

Customer, Mall	Mall cooperate	Mall defect
Customer cooperate	$(I - C_r)R_{\max}, (V_{msg} - C_{msg})R_{\max}$	$(I - C_r)R_{\infty}, (V_{msg} - C_{msg})R_{\infty}$
Customer defect	$0, -C_{msg}R_{\max}$	$0, -C_{msg}R_{\infty}$

Table 3. Payoff matrix for the Customer-Mall game: sending/receiving messages from mall

DD Both defect. The mall spends a lot of resources on sending a large volume of messages, which are not received as the customer is not online. Payoff values are similar to the previous example.

In the second game, the mall promises the customer an Internet connection with a certain bandwidth, while the customer promises to not over-consume bandwidth.

$U_{\max}$: The maximum amount of bandwidth a customer should consume (on average over some time interval).
BW_{min} : The minimum average bandwidth provided by the mall (via the ISP).
C_{bw} : The cost of providing the agreed service/bandwidth.
r_+ : The (positive) reputation/popularity the mall receives from providing a good service.
r_- : The (negative) reputation the mall loses from providing a poor service.
V_{low} : The value of receiving lower bandwidth.
V_{bw} : The value of consuming the maximum bandwidth allocated for one customer.
V_{∞} : The value of consuming more than the maximum allocated bandwidth.

We assume $V_{low} < V_{bw} < V_{\infty}$. The interpretation of table 4 is;

CC Both cooperate. The Mall provides the customer a reasonable bandwidth, while the customer does not consume more than his fair share. The mall

Customer, Mall	Mall cooperate	Mall defect
Customer cooperate	$V_{bw}, r_+ - C_{bw}$	V_{low}, r_+
Customer defect	$V_{\infty}, r_- - C_{bw}$	V_{low}, r_-

Table 4. Payoff matrix for the Customer-Mall game: network usage

gains reputation amongst its customers, as the available bandwidth is not affected too much by the usage from the customer, but still has the cost of providing the service. The net gain for the mall is represented by $r_+ - C_{bw}$, while the customer has the net gain of V_{bw} .

- CD** *The customer cooperates.* The customer uses the network with caution, while the mall fails to deliver the agreed bandwidth. The mall avoids the cost of providing the full service, and since the customer avoids over-consumption, the average performance is not as affected, and the reputation of the malls service is not affected. The net gain for the mall is hence r_+ . The customer on the other hand receives the reduced value V_{low} .
- DC** *The mall cooperates.* The mall provides the agreed bandwidth, which the customer exploits by over-consuming, to the obvious disadvantage of its peer customers and other users of the network. Implicitly this is a big disadvantage to the mall. The result payoff for the mall is hence $r_- - C_{bw}$, while the customer receives the V_{∞} .
- DD** *Both defect.* The mall fails to deliver the appropriate service, while the customer consumes as much as possible. The overall reputation is low, and the customer does not receive maximum bandwidth because of lower total available capacity. Payoff for the mall is hence r_- while the customer receives V_{low} .

5.3 Customer-customer

The only interaction between customers modelled in our example, is the exchange of promotional messages originally sent by the mall/shops. The parameters are

R_f : The reward (points) received from the mall for forwarding a message to a <i>peer</i> customer.
E_{send} : The energy consumed by sending a message.
E_{rec} : The energy consumed by receiving a message.
N_{max} : The maximum number of messages a node should forward to another during some time interval.
N_{∞} : Notation for a number of messages forwarded that satisfies $N_{max} < N_{\infty}$.

These parameters are used when computing the different payoff values in the game matrix in table 4. The reasoning behind these values is:

- CC** *Both cooperate.* This means they both keep the flow of forwarded messages to each other below the agreed limit (N_{max}). Both customers receive $(R_f - E_s - E_r)N_{max}$

Customer 1, Customer 2	Customer 2 cooperate
Customer 1 cooperate	$(R_f - E_s - E_r)N_{\max}, (R_f - E_s - E_r)N_{\max}$
Customer 1 defect	$(R_f - E_s)N_{\infty} - E_r N_{\max}, (R_f - E_s)N_{\max} - E_r N_{\infty}$
Customer 1, Customer 2	Customer 2 defect
Customer 1 cooperate	$(R_f - E_s)N_{\max} - E_r N_{\infty}, (R_f - E_s)N_{\infty} - E_r N_{\max}$
Customer 1 defect	0, 0

Table 5. Payoff matrix for the Customer-Customer game: forwarding messages from mall

- CD** *One cooperates.* The defector wins more credits, while the cooperator suffers from an overload of messages. The cooperating customer receives the payoff $(R - E_s)N_{\max} - E_r N_{\infty}$, while the defecting one receives $(R - E_s)N_{\infty} - E_r N_{\max}$.
- DD** *Both defect.* Both offer more than the agreed number of messages and receive a higher number of credits, but in the long run suffer from battery and bandwidth problems. They both should gain $(R_f - E_s - E_r)N_{\infty}$, but as the demand for energy is too high, they both receive 0.

6 Policy, strategy and uncertainty

The foregoing game models describe the outcomes of single ‘encounters’ between different kind of agents. To predict the outcome of models over time, we need to analyse the games over multiple interactions. The common way to do this, is to use the *iterated* dilemma models, where the outcome of each round is recorded, and is taken into consideration when the players meet again for a new round. This requires the recent game history of both players to be available to their opponents.

From the previous paragraphs, it should be obvious that these kinds of scenarios constitute a huge potential complexity in terms of the many individual games to be played. Without some simplifying device, it could become intractable to obtain a realistic picture of all the issues. Fortunately, one may be inspired by the principle that management is about the approximate constraint of a system, not about completely predictable control. The latter would be a hopeless expectation in a collaboration based on voluntary cooperation.

Voluntary cooperation theory tells us that ‘rational’ agents, in some sense, behave so as to maximize their particular payoff. Since this involves an element of trust of those one interacts with, each round of interaction involves the choice of whether to cooperate or whether to break that trust, for each of the autonomous agents. How they choose to play is their own concern. We consider only a few well-known policies (strategies) for interaction[15, 14]:

1. *Always cooperate:* The agent keeps its promises agent all the time, independently on the other’s behaviour.

2. *Tit for tat*: The agent always does what its ‘opponent’ did on the previous encounter. The agent never defects unless the other agent does.
3. *Tit for two tats*: Analogous to Tit for tat, but an agent does not defect until the opponent has defected *twice* in a row. This strategy is hence known for being more *forgiving* than Tit for tat.
4. *Occasional defections*: The agent cooperates per default, but tries to get away with occasional defections, to improve overall score.
5. *Random*: The agent has no general rule, it cooperates or defects at random.
6. *Always defect*: The agent never keeps its promises to any other agent.

To follow the Pervasive Computing Environment as the conditions change, we can perform a simulation of node activity. This is done by assigning each node a certain strategy, from the above, and then letting them interact with one another over some time. Each pair of agents involved in an interaction plays one round of a game and receives payoff according to the game matrix.

The cumulative result of many rounds of interaction results in one kind of common currency graph for the model. We defined this graph in [7], as a snapshot of the promise graph, where each promise has been reduced to a single, dimensionless real number. There might be alternative approaches to estimating the reliability of the agents, with respect to keeping their promises. Traditional reliability theory could play a role here, depending on the nature of the environment[17].

The common currency graph is represented by its adjacency matrix, which is a linear combination of the adjacency matrices of the separate promise types in the scenario. There is no unique matrix that fits this description, so we shall deal only with an example. The size of the adjacency matrix is equal to square of the number of agents participating in the scenario, or the number of nodes in the graph who have made any promises to anyone. For our scenario, this means that the size of the matrices is 15×15 (see fig. 12).

Each element $a_{i,j}$ is a measure of the reliability and activity level of agent i as experienced by agent j . For each promise π_k i has given to agent j , j monitors with which probability this promise is kept. This results in a number p_k for each promise π_k , where $p_k \in [0, 1]$. For each pair of agents (i, j) these numbers are summed up and represented in the adjacency matrix.

Large numbers in the matrix then indicates two different aspects of agent behaviour:

1. The promising agent has a high level of reliability, as there is a high probability of kept promises.
2. The promising agent has made a large number of promises.

The weakness in the method lies in the fact that one can not separate between a low number of agreements made by an agent, and whether the agent has failed to keep its promises.

One can establish any number of common currency graphs[7] in order to explore different properties of the distributed system. As a straightforward example, we estimate the sum of the individual promise probability graphs, i.e. the

reliability of keeping each of the promise types (see fig 12). In section 8 we shall use this to compute the importance of the nodes to the distributed system.

$$A_{\text{Common-currency}} = \begin{pmatrix} 0.0 & 0.9 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 \\ 0.0 & 0.0 & 1.7 & 1.7 & 1.7 & 1.7 & 1.7 & 1.7 & 1.7 & 1.7 & 1.7 & 1.7 & 1.7 & 1.7 & 1.7 & 1.7 \\ 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 2.5 & 2.5 & 2.5 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 \\ 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 2.6 \\ 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 2.4 & 2.4 & 2.4 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 \\ 0.0 & 1.4 & 1.4 & 1.4 & 1.4 & 0.0 & 0.7 & 0.7 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 \\ 0.0 & 1.8 & 1.8 & 1.8 & 1.8 & 0.9 & 0.0 & 0.9 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 \\ 0.0 & 1.8 & 1.8 & 1.8 & 1.8 & 0.9 & 0.9 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 \\ 0.0 & 1.4 & 1.4 & 1.4 & 1.4 & 0.0 & 0.0 & 0.0 & 0.0 & 0.7 & 0.7 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 \\ 0.0 & 1.0 & 1.0 & 1.0 & 1.0 & 0.0 & 0.0 & 0.0 & 0.5 & 0.0 & 0.5 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 \\ 0.0 & 0.4 & 0.4 & 0.4 & 0.4 & 0.0 & 0.0 & 0.0 & 0.2 & 0.2 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 \\ 0.0 & 1.6 & 1.6 & 1.6 & 1.6 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.8 & 0.8 & 0.0 & 0.0 \\ 0.0 & 1.8 & 1.8 & 1.8 & 1.8 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.9 & 0.0 & 0.9 & 0.0 & 0.0 \\ 0.0 & 1.6 & 1.6 & 1.6 & 1.6 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.8 & 0.8 & 0.0 & 0.0 & 0.0 \\ 0.0 & 1.4 & 1.4 & 1.4 & 1.4 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 \end{pmatrix}$$

Fig. 12. The sum matrix of probabilities for promises of all types, estimated over time.

6.1 Example reliability

Given the game models, or expected decisions, and the chosen strategy by each agent, we can predict an agent's moves. By letting the agents interact several times over some time interval, we can compute the level of cooperation among the agents on average for that specific time interval. An example of this was presented in ref. [10].

Definition 1 (Cooperation probability). *The cooperation probability of an agent j relative to agent i is noted by $p_c^i(j)$, and is computed according to the formula*

$$p_c^i(j) = \frac{C}{I}$$

where C is the number of times agent j cooperated in interactions with agent i , and I is the total number of interactions between agents i and j .

Consider two of the agents in the Smart Mall, for instance node 6 and node 7, noted by agent A and agent B in this example, which are both customers. These two are in the same wireless range, and are able to forward messages to

each other. They both make a promise to each other about keeping the number of forwarded messages below *ten* per hour, which on average implies maximum *one* per five minute interval.

An interaction between these two agents consists of a message being forwarded from one to the other, for instance from *A* to *B*. As long as the total number of messages from *A* to *B* is equal to or less than ten during one hour, agent *B* record this as cooperative behaviour. As soon as message number *eleven* arrives at *B* within one hour, the action is being identified as ‘defectious’ behaviour, or that the promise π_{AB} has been broken.

For each of the agents to be able to evaluate the behaviour of its opponent, they both keep a record of the previous interactions between them, or more precisely, the actions of the opponent agent. From this record, the *cooperation probability* of each of the agent can be calculated.

For the entire graph representing all the agents in the smart mall example, the same mechanisms mentioned in the previous example must be taken into consideration for *each pair* of interacting agents. The cooperation probability is computed for each promise, and added as a scalar *weight* to the edge each promise graph. We then create the new graph of weighted edges, where each weight now represents the level of cooperative behaviour for each promise. This graph can be applied in several different ways.

7 Predictable Dilemmas

To be able to predict the outcome of the games, we are particularly interested in whether the payoff matrices of our example scenario satisfy the constraints of a Dilemma game model, i.e. that we should have:

1. $T > R > P > S$
2. $R > \frac{T+S}{2}$

The promises in the game, seem reasonable, as do the costs involved in making them—but is this seeming reasonableness sufficient to ensure a functional network?

We will now go through each of our game matrices, and for each one we will discuss for what kind of parameters these constraints will or will not satisfy the constraints.

7.1 Table II : ISP-Mall game

We see that constraint 1 reduces to

$$P_s > P_s - C_s > 0 > -C_s \quad (2)$$

which is trivial iff $C_s > 0$, and hence the cost of providing a service should ideally be non-zero and positive. Constraint 2 simply reduces to

$$P_s > C_s \quad (3)$$

which means that the price demanded by the ISP should be larger than the cost it incurs in providing the service.

For the Mall, constraint 1 hence implies

$$S > S - P_s > C_r > C_r - P_s \quad (4)$$

which, given that $P_s > 0$, implies that $S - P_s > C_r$, which also is the result constraint after applying constraint 2.

This implies that the value of the agreed service minus the price paid for the service, i.e. the net value for the mall, after buying the service, should be larger than the costs associated with not having the service. This will always hold when the net gain is greater than or equal to zero and makes intuitive sense.

7.2 Table III: Customer - Mall I

The constraints here do not hold for either of the parties in this case. The implications of this is that we do not have a Dilemma game for the customer-mall interaction, and hence the known stabilities of dilemma games do not apply here, without modifications to the policy.

7.3 Table IV: Customer - Mall II

Again, constraint 1 is not fulfilled, since

$$T > R > P = S \quad (5)$$

i.e. the requirement $P > S$ is not fulfilled. For the mall, we need

$$r_+ > r_+ - C_{bw} > r_- > r_- - C_{bw}, \quad (6)$$

which is fulfilled as long as $C_{bw} > 0$, and $r_+ - C_{bw} > r_-$. This implies that the value of having a good reputation, minus the cost associated with gaining it, must be worth more to the mall than the loss of reputation from not providing an adequate service. Since r_- is *negative*, this means that the constraint allows C_{bw} to be up to $|r_-|$ units bigger than r_+ , which means the cost of maintaining a reputation is bigger than the associated value of the reputation.

For constraint 2 to be satisfied, we need

$$r_+ - C_{bw} > \frac{r_+ + r_- - C_{bw}}{2}, \quad (7)$$

which holds iff $r_+ - r_- > C_{bw}$, implying that maintaining a good reputation needs to be a cheaper proposition than the cost to the agent to turn a bad reputation into a good one. This seems like a reasonable assumption.

7.4 Table V: Customer - Customer

Consider constraint 1: $T > R$, we need $R_f - E_s > 0$, which is equivalent of $R > E_s$, which means that the value of the credits earned by sending a message must be larger than the value of the energy wasted by sending a message. Simply put, a customer must have a net gain from forwarding a message to another customer.

Further, for $R > P$, it is necessary that

$$(R_f - E_s - E_r)N_{\max} > 0 \quad (8)$$

which again implies that $R > E_s + E_r$. We notice that this constraint is stronger than the previous one. It translates as: the credits earned from forwarding messages must be of greater value than the sum of the energy consumed by sending a message and the energy consumed by receiving a message.

Finally, to check if $P > S$, we need to establish whether

$$(R_f - E_s)N_{\max} - E_r N_{\infty} < 0. \quad (9)$$

Constraint 2 requires that

$$(R_f - E_s - E_r)N_{\max} > \frac{(R_f - E_s - R_r)(N_{\max} + N_{\infty})}{2}. \quad (10)$$

For instance, if we assume $N_{\infty} = 2N_{\max}$, the requirement $R > \frac{T+S}{2}$ then reduces to

$$R_f > \frac{3}{2}R_f, \quad (11)$$

which holds only if R_f is negative, defeating its purpose. This means that the sum of the energy consumed sending and receiving a message is bigger than the credits earned forwarding a message. This requirement fails to hold if constraint 1 is to hold, so this case is clearly *not* a Prisoner's Dilemma for this value of N_{∞} .

The conclusion is that the promise values in this model are not sufficiently well motivated to make voluntary cooperation a likelihood. A policy maker, who hopes to have this mall environment succeed would be advised to arrange for the conditions of the promises to be more conducive to cooperative behaviour by altering either policy or pricing. This can only be achieved by altering the basic valuations or attitudes specified.

8 Importance Ranking

We return, finally to the extraction of management overviews that might be useful to a planner, or to an agent entering into a new environment.

The importance ranking of nodes using the spectral properties of a network is a method that derives from social network theory and is growing in

importance[18–20, 12, 21]; see ref. [13] for an introduction. This model has been applied to ad hoc networking in ref. [22] In ref. [7] it was proposed to use this method to establish the average understanding of critical node structure.

The adjacency matrix of the common currency graph is a directed graph of probabilities for cooperation, as determined either by observational experience or game theoretical prediction. Each arrow in this graph is a probable promise, or a willingness to do something for a neighbour in the pervasive distributed system. The spectral properties of this graph can tell us about the global properties and its structure. In particular, the method of eigenvector ranking allows us to rank the agents in terms of the currency of their cooperation, with regard to every other node in the network. By doing this, we obtain a fair picture of the contributions each agent makes to the collective activity. The functionality of the distributed system is already implicitly defined by the topology of the promise *types*; we are now looking at the reliability of the system.

Since the promise graph is directed, the common currency adjacency matrix is also directed and hence $A \neq A^T$. We compute the principal eigenvector of both these to obtain the following interpretations (see fig 6).

- $\text{eig}(A)$ makes the highest ranking node equal to the most willing to cooperate (the most subservient or easily influenced node). This information is of great value to anyone wishing to exploit agents.
- $\text{eig}(A)$ makes the highest ranking node that which is most able to attract promises (the strongest recipient of services from around the network).

The eigenvalues and components of the principal eigenvector sum up recursively the contributions from the entire graph, i.e. they do more than simply count promise reliabilities between neighbours: they give a distributed notion of reliability and hence stability. The similarity of concept with refs. [23, 24] is noteworthy. Eigenvector ranking of the common currency graph leads to the results in table 6.

One sees from the graph that the most ‘powerful’ or coercive agents in this system are the shops. They even outrank the mall. Indeed the service provider (node 1) is a non-entity in this regard, since it is largely outside the area of activity.

The mall node is also of central importance in binding the other nodes together. If one removes the mall node 2, the graph splits into two natural regions, in the sense of ref. [12], which are dominated by two of the shop nodes 4 and 5 (see fig. 15). These are then bridged together in a virtual mall by the interacting customers. This is an interesting result, because it tells us that the establishment of a centralized mall authority is not necessary for the functioning of the mall. While it might have acted as a seed for the resulting structure, business can proceed without it when customers are enabled with personalized services.

We see the power of the present method, in being able to identify these functional structures regardless of one’s initial prejudices about the functions of the agents. The roles they assume occur through their promises to others in the collective. A mall is simply a community structure. It does not necessarily require a named object to control it.

Rank position	Agent numbers		
	A (Senders)	A^T (Receivers)	$A + A^T$
1	2	3	2
2	13	4	3
3	7	5	5
4	8	2	7
5	12	6	8
6	14	7	4
7	3	8	6
8	6	11	13
9	9	10	9
10	15	9	12
11	10	15	14
12	5	12	15
13	1	14	10
14	11	13	11
15	4	1	1

Table 6. Ranking of the nodes from top to bottom. The topmost nodes are most powerful (A^T), or most subservient (A). Rank values for nodes $\{7,8\}$ and $\{12,14\}$ are identical in each case, indicating a symmetry. Also, the rank positions for $\{3,4,5\}$ are identical for A^T .

Customers are ‘powerful’ if they are associated with powerful shops or conglomerates of customers, i.e. they inherit promises from those they are promised by, and hence accumulate the certainty that bolsters their own standing in the network.

9 Discussion and conclusions

We have explored a pervasive computing scenario, in which a central feature is the autonomy of the devices within a region of interaction. We have demonstrated how promise theory can be utilised as a tool for predicting the coalitions and structures in a pervasive computing scenario, with or without prior knowledge. Contradictory policies are easily found using the notion of a broken promise, and potential flaws in the scenario are apparent as sources and sinks in the directed graph.

As a tool for modelling, the programme we propose is summarized as follows:

- Identify *autonomous agents*, and determine their policies and *strategies* for interaction.
- Identify the *promises* between the interacting agents.
- Define interaction pattern (interaction frequency) of the agents.
- Derive the labelled *adjacency matrices* for each *type* of promise.
- For each interacting pair of agents compute estimated payoff for each agent in order to estimate the likely strategy and hence cooperativeness of the agent.

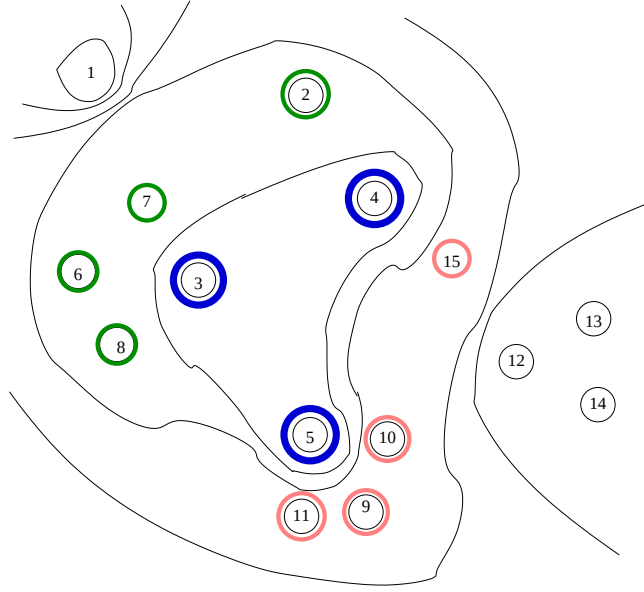


Fig. 13. A rank contour for the model, showing the influentialness or ‘power’ the agents have to attract promises (strong receivers), $\text{eig}(A^T)$.

- Update the adjacency matrices according to the payoff matrices and the interaction pattern.

There are many outstanding issues to be considered, both in promise theory and in its application to pervasive computing.

- How do promises arise to begin with?
- What consequences arise from the specifics of the individual promises (here we have only focused on the differing types of promise).
- How does one develop the payoff function, in respect to an environment?
- Does the number of nodes itself influence the payoff values?
- How do we model interaction/encounter between agents that have made several promises to each other?
- Network usage: could be customer-customer game?

We have made almost no reference to ‘cooperative’ promises $C(\pi)$ in this paper[5]. Such promises are instrumental in securing reliability and consistency in a distributed environment. They are therefore most important in more traditional scenarios of ‘managed networks’.

The value of the present method to network forensics can also be explored in more depth. What can we learn from structure itself? In the view of refs. [12, 25], structure and function are often closely mapped and a full understanding of

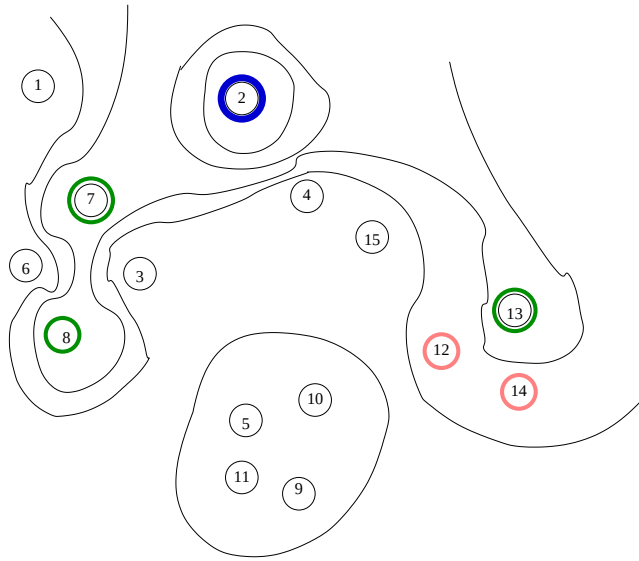


Fig. 14. A rank contour of the ‘willingness’ of the nodes to offer and keep promises (strong senders), $\text{eig}(A)$.

the structure allows one to infer much about the function. This point of view is important in a situation of widespread autonomy, as management takes on the role of a police force watchful of infractions rather than pulling the strings of the devices.

This work is supported by the EC IST-EMANICS Network of Excellence (#26854)

References

1. J. Undercoffer, F. Perich, A. Cedilnik, L. Kagal, and A. Joshi. A secure infrastructure for service discovery and access in pervasive computing. *Mobile Networks and Applications*, 8(2):113–125, 2003.
2. L. Kagal, T. Finin, and Joshi Anupam. A policy language for a pervasive computing environment. In *IEEE 4th International Workshop on Policies for Distributed Systems and Networks, 2003. Proceedings. POLICY 2003*, pages 63–74, 2003.
3. K. Henriksen, J. Indulska, and A. Rakotonirainy. Modeling context information in pervasive computer systems. In *Lecture Notes in Computer Science*, volume 2414, pages 167–180, 2002.
4. I W Marshall and L Sacks. Autonomous Management for Pervasive Computing. In *Proceedings ICCS*, Nashua, unknown 2002.
5. Mark Burgess. An approach to understanding policy based on autonomy and voluntary cooperation. In *IFIP/IEEE 16th international workshop on distributed systems operations and management (DSOM)*, in *LNCS 3775*, pages 97–108, 2005.

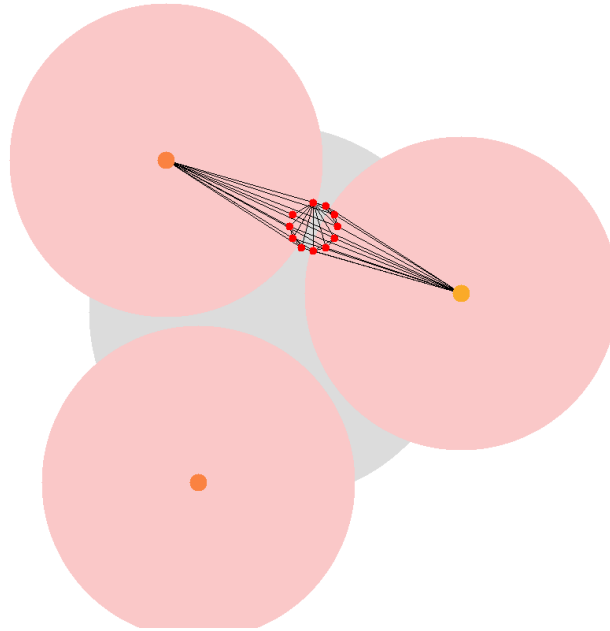


Fig. 15. Eliminating the mall node 2 isolates the ISP node and forms two network regions dominated by two of the shops and bridged by the intermediary customers.

6. R. Badonnel, R. State, and O. Festor. Probabilistic management of ad-hoc networks. In *Proceedings of 10th IFIP/IEEE Network Operations and Management Symposium NOMS 2006*.
7. M. Burgess and S. Fagernes. Pervasive computing management: A model of network policy with local autonomy. *IEEE Transactions on Software Engineering*, page (submitted).
8. F. Mattern and M. Naghshineh, editors. *Pervasive Computing*, volume 2414 of *Lecture Notes in Computer Science*. Springer, 2002.
9. H.W. Gellersen, R. Want, and A. Schmidt, editors. *Pervasive Computing*, volume 3468 of *Lecture Notes in Computer Science*. Springer, 2005.
10. M. Burgess and S. Fagernes. Voluntary economic cooperation in policy based management. *IEEE Transactions on Software Engineering*, page (submitted).
11. M. Burgess and K. Begnum. Voluntary cooperation in a pervasive computing environment. *Proceedings of the Nineteenth Systems Administration Conference (LISA XIX) (USENIX Association: Berkeley, CA)*, page 143, 2005.
12. G. Canright and K. Engø-Monsen. A natural definition of clusters and roles in undirected graphs. *Science of Computer Programming*, 53:195, 2004.
13. M. Burgess. *Analytical Network and System Administration — Managing Human-Computer Systems*. J. Wiley & Sons, Chichester, 2004.
14. R. Axelrod. *The Evolution of Co-operation*. Penguin Books, 1990 (1984).
15. R. Axelrod. *The Complexity of Cooperation: Agent-based Models of Competition and Collaboration*. Princeton Studies in Complexity, Princeton, 1997.

16. M. Wooldridge. *An Introduction to MultiAgent Systems*. Wiley, Chichester, 2002.
17. A. Høyland and M. Rausand. *System Reliability Theory: Models and Statistical Methods*. J. Wiley & Sons, New York, 1994.
18. L. Page, S. Brin, R. Motwani, and T. Winograd. The pagerank citation ranking: Bringing order to the web. *Technical report, Stanford University, Stanford, CA*, 1998.
19. Lawrence Page, Sergey Brin, Rajeev Motwani, and Terry Winograd. The pagerank citation ranking: Bringing order to the web. Technical report, Stanford Digital Library Technologies Project, 1998.
20. C. Ding, X. He, P. Husbands, H. Zha, and H. Simon. PageRank, HITS and a unified framework for link analysis. Technical Report 49372, LBNL, 2002.
21. M. Burgess, G. Canright, and K. Engø. Importance-ranking functions from the eigenvectors of directed graphs. *Journal of the ACM (Submitted)*, 2004.
22. R. Badonnel, R. State, and O. Festor. Management of mobile ad-hoc networks: evaluating the network behavior. In *Proceedings of the 9th IFIP/IEEE International Symposium on Integrated Network Management*, pages 17–30. IEEE Press, 2005.
23. J.M. Hendrickx et al. Rigidity and persistence of three and higher dimensional forms. In *Proceedings of the MARS 2005 Workshop on Multi-Agent Robotic Systems*, page 39, 2005.
24. J.M. Hendrickx et al. Structural persistence of three dimensional autonomous formations. In *Proceedings of the MARS 2005 Workshop on Multi-Agent Robotic Systems*, page 47, 2005.
25. M. Burgess, G. Canright, and K. Engø. A graph theoretical model of computer security: from file access to social engineering. *International Journal of Information Security*, 3:70–85, 2004.