Effective Use of Organisational Abstractions for Confidence Models*

Ramón Hermoso, Holger Billhardt, Roberto Centeno, and Sascha Ossowski

Artificial Intelligence Group - DATCCCIA
University Rey Juan Carlos, Madrid (Spain)
{ramon.hermoso, holger.billhardt, roberto.centeno, sascha.ossowski}@urjc.es

Abstract. Trust and reputation mechanisms are commonly used to infer expectations of future behaviour from past interactions. They are of particular relevance when agents have to choose appropriate counterparts for their interactions as it may also happen within virtual organisations. However, when agents join an organisation information about past interactions is usually not available. The use of organisational structures can tackle this problem and can improve the efficiency of trust and reputation mechanisms by endowing agents with some extra information to choose the best agents to interact with. In this context, we present how certain structural properties of virtual organisations can be used to build an efficient trust model in a local way. Furthermore, we introduce a testbed tool (TOAST) that allows to analyse different trust and reputation models in situations where agents act within virtual organisations. We experimentally evaluate our approach and show its validity.

1 INTRODUCTION

The concept of organisation is of significant importance to MultiAgent Systems (MAS). Particularly, organisational structures are often used in research about Agent-oriented Software Engineering [26]. In fact, organisational concepts are commonly used as abstract pieces that help designers to build more complex models in MAS design processes [14].

Organisational abstractions can be used to impose some structure on a society of agents and can endow MAS with certain behaviours. Agents joining an organisation play specific *roles* in different *interactions* and they are supposed to act conforming to the prescriptions of these concepts. Furthermore, these prescriptions may be complemented by a more general set of *norms* [23] and some kind of mechanisms that make it difficult for agents to transgress norms (e.g. by providing specific "governor" agents [4], by integrating "filtering" mechanisms [13], or by using protocols of sequential actions). We will call MAS with such organisational structures as *Virtual Organisations* (VOs) [18].

VOs can be considered as limiting the freedom of choice of agents because they regulate the interactions within a MAS. However, especially within low

* The present work has been funded by the Spanish Ministry of Education and Science under project TIC2003-08763-C02-02.

regulated organisations, agents will still have to tackle the problem of choosing appropriate counterparts for their interactions according to their own beliefs and goals. Within this scenario, trust and reputation mechanisms can be integrated into VOs providing support to agents' decision-making processes.

Much work has been done in the field of trust and reputation systems that provide agents with some hints about the future behaviour of their acquaintances based on past experience [9, 2, 17]. Nevertheless, most of the research is based on the importance of distributed trust and on the exchange of information among agents (e.g. reputation values about third parties) in poorly structured systems.

Trust and reputation systems are not only useful for VOs, but VOs also for trust and reputation mechanisms. The structure provided by a VO can be used to construct more effective trust mechanisms. In particular, the structural elements defined in a VO (e.g., roles and interactions) provide a certain notion of similarity which allows agents to infer the expected behaviour of acquaintances within totally new situations by analysing their past behaviour within *similar* situations. This property is especially useful in situations where agents can not count on their own past experiences, e.g., when they just have joined an organisation, or within very volatile environments.

In this paper we continue our previous work [7] on trust and reputation mechanisms for VOs. We present some experiments that show how the use of organisational abstractions can effectively improve trust and reputation mechanisms. The experiments have been carried out using a testbed called *TOAST* (Trust Organisational Agent System Testbed) and which we have developed for testing trust and reputation models. In Section 2 we briefly summarise our model and show how it can guide an agent's decision-making within a VO. Section 3 introduces the testbed. In Section 4 we present experimental results comparing different models. We summarise related work in Section 5, and we present some conclusions and future lines of work in Section 6.

2 CONFIDENCE AND TRUST FOR ORGANISATIONAL STRUCTURES

In this section we summarise our previous work [7] on a trust model for VOs. We first show how basic trust mechanisms can be integrated in VOs. Afterwards, we explain how an agent can use knowledge about the organisational structure to infer confidence in an issue if no previous experience is available.

2.1 Basic Local-Based Trust Model for Virtual Organisations

As outlined in [7], it is natural to consider that agents participating in a VO play some roles in different interactions. In addition, we assume that the agents know the organisational structure, e.g., they know the existing roles and interaction types, the roles that participate in each interaction type, as well as the roles other agents are playing within the organisation.

Similarly to other approaches [10, 24, 16, 15], we build our trust model on the idea of *confidence* and *reputation*. Both are ratings agents use in order to evaluate the trustworthiness of other agents in a particular issue (e.g., playing a particular role in a particular interaction). *Confidence* is a local measure that is only based on an agent's own past experiences, while *reputation* is an aggregated value an agent gathers by asking its acquaintances about their opinion regarding the trustworthiness of another agent. Thus, reputation can be considered as an external or *social* measure. We define *trust* as a rating resulting from combining *confidence* and *reputation* values.

A typical scenario for the use of a trust model is the following. An agent A wants to evaluate the trustworthiness of some other agent B – playing the role R – in the interaction I. This trustworthiness is denoted as $t_{A\rightarrow\langle B,R,I\rangle}$, with $t_{A\rightarrow\langle B,R,I\rangle} \in [0..1]$, and it measures the trust of A in B (playing role R) being a "good" counterpart in the interaction I. When evaluating the trustworthiness of a potential counterpart[1], an agent can combine its local information (confidence) with the information obtained from other agents regarding the same counterpart (reputation).

Confidence, $c_{A\rightarrow\langle B,R,I\rangle}$, is collected from A's past interactions with agent B playing role R and performing interactions of type I. We call LIT – *Local Interaction Table* – the agent's data structure dedicated to store confidence values for past interactions with any counterpart the agent has interacted with. Each entry corresponds to an *issue*: an *agent* playing a specific *role* in a particular *interaction*. LIT_A denotes agent A's LIT. An example is shown in Table 1.

Table 1. An agent's local interaction table (LIT_A)

$\langle X,Y,Z\rangle$	$c_{A\rightarrow\langle X,Y,Z\rangle}$	$r_{A\rightarrow\langle X,Y,Z\rangle}$
$\langle a_9, r_2, i_3\rangle$	0.2	0.75
$\langle a_2, r_7, i_1\rangle$	0.7	0.3
$\vdots$	$\vdots$	$\vdots$
$\langle a_9, r_2, i_5\rangle$	0.3	0.5

Each entry in a LIT consists of: i) the Agent/Role/Interaction identifier $\langle X,Y,Z\rangle$, ii) the confidence value for the issue ($c_{A\rightarrow\langle X,Y,Z\rangle}$), and iii) a reliability value ($r_{A\rightarrow\langle X,Y,Z\rangle}$). The confidence value is obtained from some function that evaluates past experiences on the same issue. We suppose $c_{A\rightarrow\langle X,Y,Z\rangle} \in [0..1]$ and higher values to represent higher confidence.

Each direct experience of an agent regarding an issue $\langle X,Y,Z\rangle$ changes its confidence value $c_{A\rightarrow\langle X,Y,Z\rangle}$. In this sense, we suppose that the agents have some kind of mechanism to evaluate the behaviour of other agents they interact with. Let $g_{\langle X,Y,Z\rangle} \in [0..1]$ denote the evaluation value an agent A calculates for a particular experience with the agent X playing role Y in the interaction of type

[1] By potential counterpart we mean an agent which is a candidate to interact with.

Z. In our work, we use the following equation to update confidence:

$$c_{A\rightarrow\langle X,Y,Z\rangle} = \epsilon \cdot c'_{A\rightarrow\langle X,Y,Z\rangle} + (1-\epsilon) \cdot g_{\langle X,Y,Z\rangle}, \tag{1}$$

where $c'_{A\rightarrow\langle X,Y,Z\rangle}$ is the confidence value in A's LIT before the interaction is performed and $\epsilon \in [0..1]$ is a parameter specifying the importance given to A's past confidence value. In general, the aggregated confidence value from past experiences will be more relevant than the evaluations of the most recent interactions.

Reliability ($r_{A\rightarrow\langle X,Y,Z\rangle}$) measures how certain an agent is about its own confidence in an issue. We suppose $r_{A\rightarrow\langle X,Y,Z\rangle} \in [0..1]$. Furthermore, we assume that $r_{A\rightarrow\langle X,Y,Z\rangle} = 0$ for any tuple $\langle X, Y, Z\rangle$ not belonging to LIT_A. We calculate reliability by using the approach proposed by Huynh, Jennings and Shadbolt [9, 10]. This approach takes into account the number of interactions a confidence value is based on and the variability of the individual values across past experiences.

An agent may build trust directly from its confidence value or it may combine confidence with reputation. Reputation will be particularly useful when an agent has no experience on an issue or if the reliability value for the confidence is not high enough. Although we will not deal with it in this paper, social reputation may be obtained by asking other agents about their opinion on an issue. Agents that have been requested for their opinion will return the corresponding confidence and reliability ratings from their LIT. The requester might then be able to build trust by calculating a weighted mean over its own confidence value and the confidence values received from others, as it is represented in equation (2):

$$t_{A\rightarrow\langle B,R,I\rangle} = \begin{cases} c_{A\rightarrow\langle B,R,I\rangle}, & if \quad r_{A\rightarrow\langle B,R,I\rangle} > \theta \\ \dfrac{\sum\limits_{X\in AA\cup\{A\}} c_{X\rightarrow\langle B,R,I\rangle}\cdot w_{X\rightarrow\langle B,R,I\rangle}}{\sum\limits_{X\in AA\cup\{A\}} w_{X\rightarrow\langle B,R,I\rangle}} & otherwise \end{cases} \tag{2}$$

$\theta \in [0..1]$ is a threshold on the reliability of confidence. If the reliability is above θ then an agents own confidence in an issue is used as the trust value. Otherwise trust is build by combining confidence and reputation. AA is a set of acquaintance an agent asks about their opinion regarding the issue $\langle B, R, I\rangle$. Within a VO, the structural abstractions may provide hints for the proper selection of such a set of acquaintance. For instance, in some scenarios it may be useful to ask other agents that play the same role as A, since they may have similar interests and goals.

The weights $w_{X\rightarrow\langle B,R,I\rangle}$ given to the gathered confidence values is composed of the corresponding reliability value and a constant factor α that specifies the importance given to A's own confidence in the issue, as it is shown in the following equation:

$$w_{X\rightarrow\langle B,R,I\rangle} = \begin{cases} r_{X\rightarrow\langle B,R,I\rangle} \cdot \alpha, & if \quad X = A \\ r_{X\rightarrow\langle B,R,I\rangle} \cdot (1-\alpha), & otherwise \end{cases} \tag{3}$$

2.2 Confidence Inference using Organisational Structure Similarities

In this section we propose a local way for building trust on an issue when no past interactions have been performed and without relying upon social reputation. In [7] we proposed to use the agent/role confidence $c_{A\rightarrow\langle B,R,\_\rangle}$ (or the agent confidence $c_{A\rightarrow\langle B,\_,\_\rangle}$) as an estimation for $c_{A\rightarrow\langle B,R,I\rangle}$ if agent A has no reliable experience about issue $\langle B,R,I\rangle$. This approach relies on the hypothesis that, in general, agents behave in a similar way in all interactions related to the same role. We argue that, exploiting this idea, the more similar I' and I are, the more similar the values $c_{A\rightarrow\langle B,R,I'\rangle}$ and $c_{A\rightarrow\langle B,R,I\rangle}$ will be. The same applies to roles. Using this assumption, confidence ratings accumulated for similar agent/role/interaction tuples may provide evidence for the trustworthiness of the issue $\rightarrow\langle B,R,I\rangle$. Based on this idea, we propose to build trust by taking into account all the past experiences an agent has, focusing on their degree of similarity with the issue $\langle B,R,I\rangle$. In particular, we calculate trust as a weighted mean over all the confidence values an agent has accumulated in its LIT. This is shown in the following equation:

$$t_{A\rightarrow\langle B,R,I\rangle} = \frac{\sum\limits_{\langle X,Y,Z\rangle\in LIT_A} c_{A\rightarrow\langle X,Y,Z\rangle}\cdot w_{A\rightarrow\langle X,Y,Z\rangle}}{\sum\limits_{\langle X,Y,Z\rangle\in LIT_A} w_{A\rightarrow\langle X,Y,Z\rangle}} \tag{4}$$

$w_{A\rightarrow\langle X,Y,Z\rangle}$ is the weight given to agent A's confidence on issue $\langle X,Y,Z\rangle$. The weights combine the confidence reliability with the similarity of the issue $\langle X,Y,Z\rangle$ to the target issue $\langle B,R,I\rangle$ in the following way:

$$w_{X\rightarrow\langle X,Y,Z\rangle} = r_{A\rightarrow\langle X,Y,Z\rangle}\cdot sim(\langle X,Y,Z\rangle,\langle B,R,I\rangle) \tag{5}$$

The similarity function $sim(\langle X,Y,Z\rangle,\langle B,R,I\rangle)$ is computed as the weighted sum of the similarities of the individual elements (agent, role and interaction) as it is shown in the following equation:

$$sim(\langle X,Y,Z\rangle,\langle B,R,I\rangle) = \begin{cases} \beta\cdot sim_R(R,Y)+\gamma\cdot sim_I(I,Z),\ if\quad B=X \\ 0,\ otherwise \end{cases} \tag{6}$$

where $sim_R(R,Y)$, $sim_I(I,Z)\in[0..1]$ measure the similarity between roles and interactions, respectively, and β and γ, with $\beta+\gamma=1$, are parameters specifying the sensibility regarding the individual similarities.

We suppose that organisational models include taxonomies of roles and/or interactions from which role and interaction similarity measures can be derived. In this case, $sim_R(R,R')$ and $sim_I(I,I')$ can be implemented by *closeness functions* that estimate the similarity between two concepts on the basis of their closeness in the concept hierarchy.

Equation (4) can be used as an alternative way to build trust. Especially if an agent has no reliable experience about a particular agent/role/interaction issue, this model can be used to estimate trust without the necessity to rely on the opinions of other agents. Thus, the proposed model makes agents less dependent on others, which is an important issue, in particular in VOs that do not provide mechanisms to keep its members from cheating.

3 Trust Organisational Agent System Testbed (TOAST)

In this section we present TOAST, the tool we have developed in order to evaluate trust models by showing the influence of these models on the evolution of the overall utility of an agent or a society of agents. This testbed simulates a virtual organisation where agents have to interact with others in order to to achieve their goals. TOAST is based on the following simplifications:

- TOAST does not consider the problem of finding appropriate interactions that help to achieve an agent's goals, nor it considers the task of locating possible candidates that can act as counterparts in the interactions an agent wants to perform. We suppose that both these problems are resolved, e.g., the corresponding information is fully available to each agent within the organisation. The only problem the testbed actually addresses is the selection of appropriate counterparts out of a set of possible candidates.
- All interactions are binary, e.g., exactly two agents are required in order to perform an interaction.
- Agents are always willing to participate in an interaction if they have been chosen by other agents to do so. That means we do not consider the problem of the selected agent to decide whether or not it is useful for itself to participate in the interaction.
- Interactions in TOAST are not actually carried out; they are simulated. In practice, this means that agents participating in an interaction do not actually evaluate the behaviour of the others in that interaction. Instead, the agents receive these evaluation values directly from the testbed. In order to generate these values, the system uses the notion of capability. Capability here indicates the "goodness" of agent's A behaviour in playing role R in an interaction of type I. Capability is represented through a normal probability distribution with constant mean and variance and is assigned to each tuple $\langle A, R, I\rangle$. The evaluation value an agent receives after performing an interaction is drawn from the corresponding capability distribution of the counterpart agent. The use of this schema implies the following simplifying assumptions:
  - An agent's capability playing a specific role within a particular interaction is stable with some variations. That is, each time an agent plays the same role in the same type of interaction it will behave in a similar way. The assumption of stability is actually the basis for any trust and reputation model. Nevertheless, in real cases it is likely that an agent's behaviour changes over time.
  - Agents don't cheat. That is, agents participating in an interaction do the best they can; they do always behave in the prescribed way (corresponding to their capabilities).

3.1 Setting up a VO in TOAST

The testbed allows the user to create virtual organisations of agents. The elements that compose a virtual organisation and the relationships among them are represented in a class diagram in Figure 1.

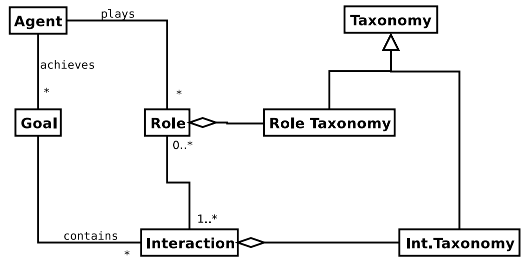

Fig. 1. Organisational elements in TOAST.

– *Agents* are the entities that participate in a VO. They can play different *roles* depending on their aptitudes. Within the VO, agents aim to achieve their goals. TOAST does not fix the type of agents. That is, users can implement different types of agents (agents with different behaviours) in order to make the VO more heterogeneous.
– *Roles* describe the functionality of the organisation. They will be played in specific interactions. The roles that are defined within an organisation have to be specified in a *role types taxonomy*. This taxonomy also specifies a conceptual hierarchy for the defined roles.
– *Interactions* are the actions that the VO allows to perform between agents. They are defined in an *interaction types taxonomy*, which also defines the roles that are involved in each interaction as well as the hierarchical relationships between interactions.
– *Goals* have to be achieved by agents. An agent may have several, different goals. A goal will be achieved by means of performing a specific interaction. The relation between goals and the interactions that help to achieve them has to be specified in a XML file which is loaded into the testbed. The following is an example of a fragment of such a file:

```
<Goal Name="Select Lecturer">
    <Interaction Name="Teach">
        <Role Name="Academic Staff">
        </Role>
        <Role Name="Student">
        </Role>
    </Interaction>
</Goal>
<Goal Name="Select Assignment Partner">
    <Interaction Name="Do Assignments">
        <Role Name="Student">
        </Role>
        <Role Name="Student">
        </Role>
    </Interaction>
</Goal>
. . .
```

After defining a virtual organisation, agents have to be added to the organisation. In particular, the user has to select the number of agents of each type that will be added to the organisation as it is shown in Figure 2. In our experiments

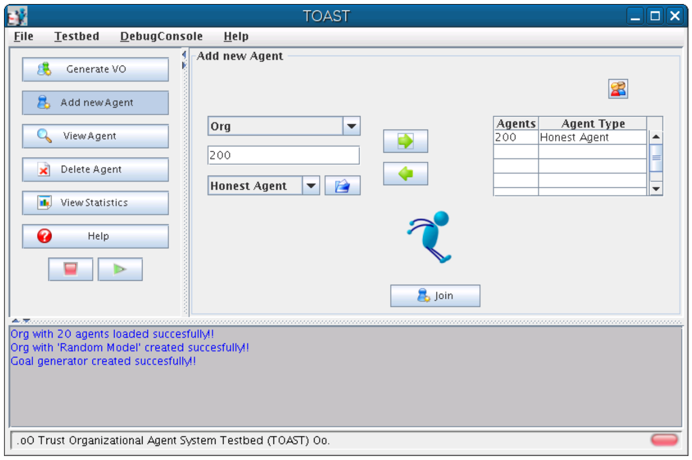

Fig. 2. Add Agents panel in TOAST's GUI

we have only used agents of one type (with the same behaviour).

After agents have been added to the VO and according to the organisation specification described above, roles have to be assigned to agents and capability values for Agent/Role/Interaction tuples have to be generated. This is done automatically as follows:

– ***Agent/Role assignment***. Roles are assigned randomly to agents. An agent may play different roles. For each role, the user can choose the percentage of agents to which the role will be assigned. This selection should be done with care. If some roles are played by only very few agents, then the scenario may not be appropriate for evaluating trust mechanisms because of the lack of sufficient candidates out of which an agent can choose its counterparts in its interactions.
– ***Agent/Role/Interaction capability values***. As we have mentioned before, in order to evaluate the behaviour of an agent in a particular situation, the system uses a normal probability distribution that model the capability of that agent in this situation. Such distributions are assigned to each tuple $\langle A, R, I \rangle$ at startup by selecting the mean and standard deviation randomly such that $\mu \in [0..1]$ and $\sigma \in [0..0.5]$. The capabilities are correlated according to the type of interaction, the type of role and the agent which is performing the action; that is, similar normal probability distributions will be assigned to the same agent playing similar roles within similar interactions[2].

[2] Similar distributions means similar mean values.

3.2 Running the Testbed

As we have mentioned before, agents will deal with the problem of selecting appropriate counterparts to interact with in order to achieve their goals. The basic process that is repeated when the testbed is executed is summarised in Figure 3.

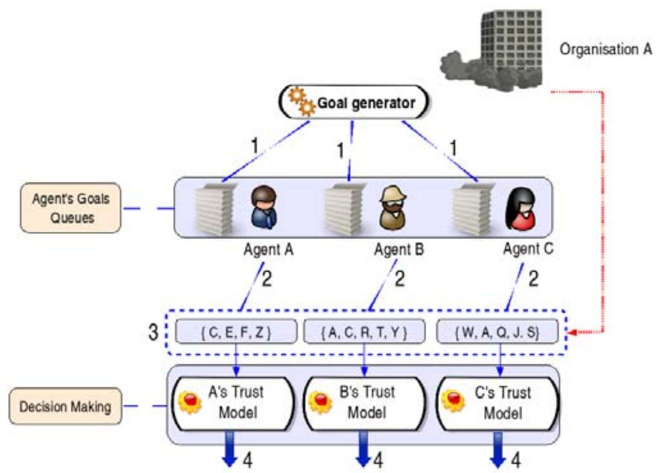

Fig. 3. Interaction simulation in TOAST

The process is the following:

1. A goal is generated for an agent. Each goal has an associated interaction that the agent must perform in order to achieve that goal. Each agent has a goal queue where goals are stored until they can be processed.
2. Once the agent has identified the interaction that eventually helps to achieve the goal, a list with potential agents that are possible counterparts in the required interaction is obtained from the organisation. As we mentioned before, in order to simplify the experiments all interaction types are binary, e.g., require exactly two agents playing particular roles.
3. The agent uses its trust model in order to select the agent that is expected to behave best in the required interaction and playing the specified role.
4. The interaction is simulated. As we have mentioned before, this step consists of sending to each agent the evaluation values ($g(A, R, I)$) for the other agent that participated in the interaction. These values are generated from the corresponding capability distributions. Finally, each agent uses the received evaluation values to update the confidence and reliability values in its LIT as described in section 2.1.

4 EXPERIMENTAL RESULTS

We have used TOAST to experimentally evaluate our confidence inference approach. In this section, first we describe the scenario that we have chosen to evaluate our assumptions. Then, we describe the different trust models that we have tested and, finally, we show the obtained results.

4.1 The University Scenario

As a test scenario we use a School of Computer Science organisation, whose members play roles out of the taxonomy shown in Figure 4. Furthermore, the social functionalities provided by the organisation are summarised in the interaction taxonomy illustrated in Figure 5.

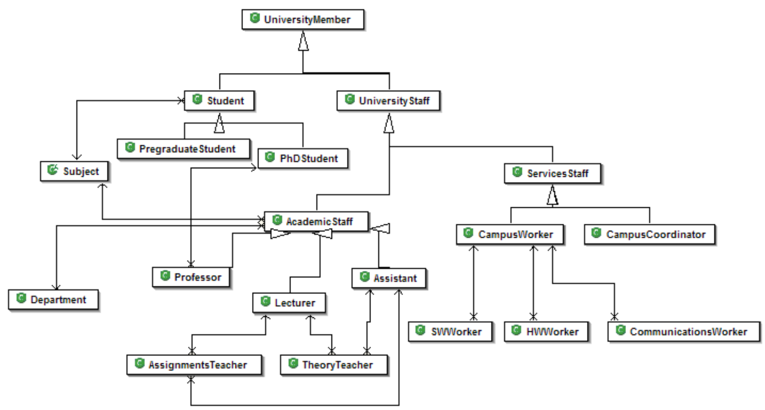

Fig. 4. Fragment of role taxonomy provided by University organisation

In this scenario, a typical situation could be, for example, when an agent playing the role of a student needs to find a partner for some kind of assignment in a specified subject. The student will use its LIT to select the best partner for the assignment (e.g., another student) according to his/her own experiences about past interactions with other students.

4.2 Different Trust Models

We tested and compared three different *local* trust models in the sense that they do not use social reputation to compute trust and are based only on different confidence evaluation approaches. Thus, each model defines a different way how an agent uses its own past experience in order to select the best counterparts for its interactions out of the set of possible candidates. The models are the following:

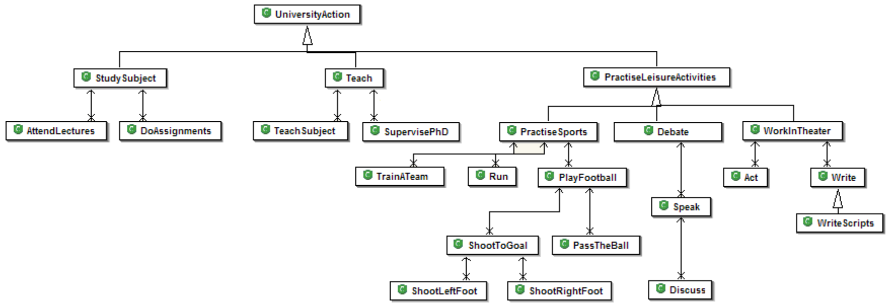

Fig. 5. Fragment of interaction taxonomy provided by University organisation

- *Random Model*: in this model agents choose the counterparts for their interactions randomly among the potential agents provided by the organisation. Thus, the selection does not take into account any experience on past interactions.
- *Basic Model*: in this model, agents evaluate the expected behaviour of the potential candidates for an issue by using the corresponding confidence value stored in their LITs. If no entry exists about an issue, e.g., no previous experiences are available, the counterpart is selected randomly.
- *Inference Model*: this model implements our local trust model, as described in Section 2.2 using equation (4). Agents using this model will use the following simple formula to calculate the similarity between roles and interactions, respectively:

$$sim_R(x,y) = sim_I(x,y) = 1 - \frac{h}{h_{MAX}} \tag{7}$$

  where x, y are either roles or interactions, h is the number of hops between x and y in the corresponding taxonomy, and h_{MAX} is the longest possible path between any pair of elements in the hierarchy tree. Equation (7) is a very simple formula to measure the similarity between concepts in a taxonomy. Other functions have been described in [11, 6].

4.3 Results

We tested and compared the performance of the *random*, *basic* and *inference model* in the university scenario and using exactly the same conditions. In particular we used an organisation with 20 agents and the goal generator generated randomly 40000 goals for those 20 agents. The generated goals, the agent/role assignment and the agent/role/interaction capability generation was exactly the same for each model. In the *inference model* we used the similarity weights $\beta = 0.8$ and $\gamma = 0.2$. Furthermore, we repeated each experiment five times using a different random seed. The presented results are the averages of the five runs.

Figure 6 shows the evolution of the overall system utility over the number of interactions that have taken place. The overall system utility is calculated as the average of the utilities of all individual agents. As utility values we use the evaluation values (about its counterparts) an agent receives after performing interactions. As it can be observed, in general, the utility improves with the number of interactions if a trust model is used to select the best counterpart out of a set of possible agents for a requested interaction. Both trust models are clearly better than a random agent selection. Moreover, it can be observed that the utility improves as the agents gain more experience, that is, as more interactions take place. The *inference model* obtains better outcomes as compared to the *basic model*, since agents' utility curves grow faster. Hence, with the *inference model* agents are able to find faster "good" counterparts to interact with. This confirms the hypothesis that the *inference model* improves agents' decision making process when agents have none or only very few experiences from past interactions. In this case, the *inference model* makes use of past experiences about similar roles and interactions in order to assess an unknown $\langle X, Y, Z \rangle$ tuple.

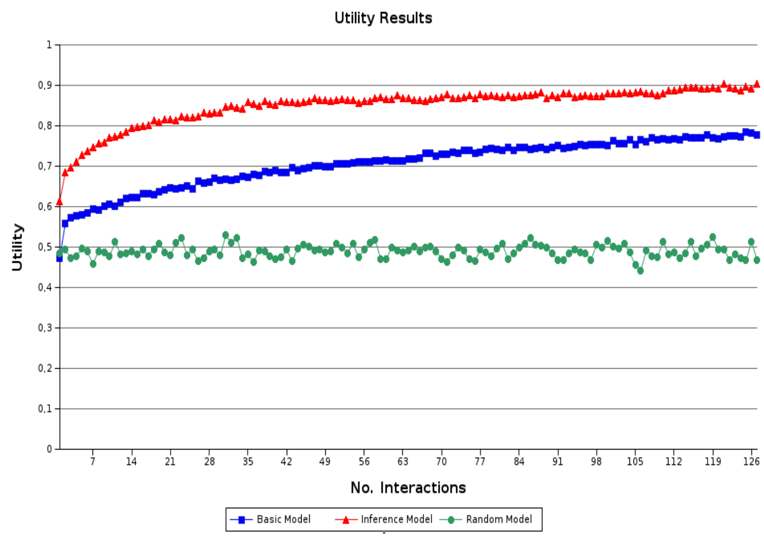

Fig. 6. Overall system utility.

This evolution is similar if we consider individual agents. Figure 7 gives an example of how the utility curve of an individual agent evolves over time. Also here, the *inference model* curve grows faster that the others, because the agent is able to find good counterparts faster.

5 DISCUSSION

A wide range of organisational (meta-) models aimed at describing basic organisational concepts and their interrelation in the context of MAS has been

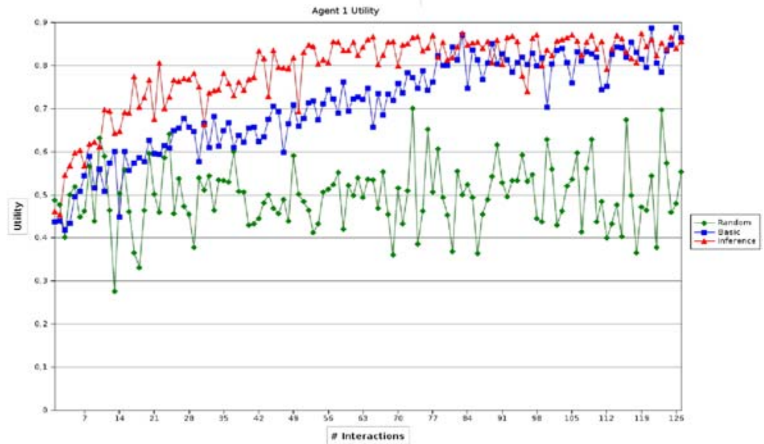

Fig. 7. Example of the evolution of an agent's individual utility.

developed in last years [5, 8, 20, 12]. It is commonly accepted that the notion of *role* is central for putting agents and organisational models together. Roles are sometimes defined by the *actions* they can perform, but usually they are characterised by the types of *social interactions* to which they contribute. The latter term does not primarily refer to the interaction protocols that agents engage in, but rather to the social functionality that such interactions shall achieve. In this sense, we assume that VOs define roles and specify the interactions (functionalities) in which each role can participate.

Several meta-models allow for *specialisation* relations among essential organisational concepts. In the organisational model underlying the FIPA-ACL, for instance, *information exchange* interactions are a special kind of *request* interaction, where the requested action is a communicative action of type *inform* (e.g., [21]). In much the same way, the *informer* role involved in this interaction can be conceived as a specialisation of the *requester* role. In summary, organisational models often contain *taxonomies* of concept types, e.g., for roles or for interactions. Such taxonomies can be provided to the agents participating in an organisation – for instance, as an organisational service – and can be used to define the similarities among roles and interactions as it is described in this paper.

Trust and reputation mechanisms have been widely studied, recently above all in *peer-to-peer* systems in general (e.g. [25, 2]), and in MAS in particular (e.g. [9, 24, 17]). In contrast to other approaches to trust systems (most of them based on reputation distribution – reputation values exchange about third parties), we have presented a way of evaluating trust at a local level that focuses on the experience of agents obtained in past interactions. The FIRE model proposed by Huynh, Jennings and Shadbold [9] is also related to *interaction trust* and *role-based trust*. As in our approach, the former is built from direct experience of an agent, while the latter is the rating that results from role-based relationships between agents. However, the FIRE model does not consider inference on VO structures.

Sensoy and Yolum [19] deal with the problem of distributed service selection in an e-commerce setting where consumers are allowed to capture their experiences with the service providers. This approach is similar to the witness reputation approach presented in [9] in the sense that agents can locate others by making use of other agents' past experience. Nevertheless, their approach does not consider inference on organisational structures, because it uses ontologies to match between required and provided services as rules and, not in order to better approximate expected agents' behaviours in a local way.

The model proposed by Sabater and Sierra [17] also exploits ontologies to make up trust values (*ontological dimension of reputation*). Nevertheless, it does not consider organisations as a whole issue, and thus it does not take into account organisational structures. Teacy et al. [22] deals with a similar approach, where trust is obtained from using probability theory (*beta distributions* taking into account agent's own past experience when it exists, and information gathered from third parties (with reputation techniques) otherwise. Although the approach considers agents living within VOs, it does not deal with VO's internal structures – as our approach does. While our approach tackles the problem of improving agent skills to decide appropriate counterparts based only on local experiences, they mainly focus on assessing reputation source accuracy.

Abdul-Rahman and Hailes [1] propose a trust model for virtual communities but use qualitative ratings for estimating trust. They focus on evaluating trust from past experiences and reputation coming from *recommender agents* without considering explicitly VO structures.

6 CONCLUSION

In this paper we have presented results of our work, aimed at integrating trust mechanisms into virtual organisations. We have tackled the problem of locally calculating trust, that is, finding "good" counterparts, even if only very few previous experiences are available and without the need of using reputation information obtained from external sources. The proposed model takes into account key concepts of organisational models, such as *roles* and *interactions*. It has confidence inference capabilities exploiting *taxonomies* of concept types provided by VOs. We have tested our model, confirming that the use of organisational structures makes agents' decision-making easier and more efficient, in particular when agents join an organisation and, thus, can not count on their own previous experiences. Furthermore, we have presented TOAST, the testbed that we have developed to test our assumptions.

In future work, we plan to extend our model with social reputation capabilities, and also study the problem of dishonest and non-cooperative agents. Furthermore, we will focus on developing an extension of TOAST that allow for evaluating trust models in non-stable and high-scalable environments, where agents join and leave organisations very often. We are also focusing on finding more accurate similarity functions to apply to organisation taxonomies.

References

1. Alfarez Abdul-Rahman and Stephen Hailes. Supporting trust in virtual communities. In *HICSS*, 2000.
2. Karl Aberer and Zoran Despotovic. Managing trust in a peer-2-peer information system. In *CIKM '01: Proceedings of the tenth international Conference on Information and Knowledge Management*, pages 310–317, New York, NY, USA, 2001. ACM Press.
3. M. Esteva, J. A. Rodriguez, C. Sierra, P. Garcia, and J. L. Arcos. On the formal specifications of electronic institutions. In F. Dignum and C. Sierra, editors, *Agent-mediated Electronic Commerce (The European AgentLink Perspective)*, volume 1191 of *LNAI*, pages 126–147, Berlin, 2001. Springer.
4. Marc Esteva, Bruno Rosell, Juan A. Rodríguez-Aguilar, and Josep Ll. Arcos. AMELI: An agent-based middleware for electronic institutions. In *Proceedings of the Third International Joint Conference on Autonomous Agents and Multiagent Systems*, volume 1, pages 236–243, 2004.
5. J. Ferber and O. Gutknecht. A meta-model for the analysis of organizations in multi-agent systems. In Y. Demazeau, editor, *Proceedings of the Third International Conference on Multi-Agent Systems (ICMAS'98)*, pages 128–135, Paris, France, July 1998. IEEE Press.
6. Prasanna Ganesan, Hector Garcia-Molina, and Jennifer Widom. Exploiting hierarchical domain structure to compute similarity. *ACM Trans. Inf. Syst.*, 21(1):64–93, 2003.
7. Ramón Hermoso, Holger Billhardt, and Sascha Ossowski. Integrating trust in virtual organisations. In *AAMAS-06 Workshop on Coordination, Organization, Institutions and Norms in agent systems (COIN)*, pages 121–133, 2006.
8. J. Fred Hübner, J. Simão Sichman, and O. Boissier. Using the moise+ for a cooperative framework of mas reorganisation. In *Advances in Artificial Intelligence - SBIA 2004*, pages 506–515, 2004.
9. T. Dong Huynh, Nicholas R. Jennings, and Nigel R. Shadbolt. Developing an integrated trust and reputation model for open multi-agent systems. In Rino Falcone, Suzanne Barber, Jordi Sabater, and Munindar Singh, editors, *AAMAS-04 Workshop on Trust in Agent Societies*, 2004.
10. T. Dong Huynh, Nicholas R. Jennings, and Nigel R. Shadbolt. FIRE: An integrated trust and reputation model for open multi-agent systems. In *Proceedings of the 16th European Conference on Artificial Intelligence (ECAI)*, 2004.
11. Yuhua Li, Zuhair A. Bandar, and David McLean. An approach for measuring semantic similarity between words using multiple information sources. *IEEE Transactions on Knowledge and Data Engineering*, 15(4):871–882, 2003.
12. James Odell, Marian H. Nodine, and Renato Levy. A metamodel for agents, roles, and groups. In James Odell, Paolo Giorgini, and Jörg P. Müller, editors, *AOSE*, volume 3382 of *Lecture Notes in Computer Science*, pages 78–92. Springer, 2004.
13. A. Omicini, S. Ossowski, and A. Ricci. Coordination infrastructures in the engineering of multiagent systems. In Federico Bergenti, Marie-Pierre Gleizes, and Franco Zambonelli, editors, *Methodologies and Software Engineering for Agent Systems: The Agent-Oriented Software Engineering Handbook*, chapter 14, pages 273–296. Kluwer Academic Publishers, June 2004.

14. Andrea Omicini and Sascha Ossowski. Objective versus subjective coordination in the engineering of agent systems. In Matthias Klusch, Sonia Bergamaschi, Peter Edwards, and Paolo Petta, editors, *Intelligent Information Agents: An AgentLink Perspective*, volume 2586 of *LNAI: State-of-the-Art Survey*, pages 179–202. Springer-Verlag, March 2003.
15. Sarvapali D. Ramchurn, Carles Sierra, Lluís Godó, and Nicholas R. Jennings. A computational trust model for multi-agent interactions based on confidence and reputation. In *Proceedings of 6th International Workshop of Deception, Fraud and Trust in Agent Societies*, pages 69–75, 2003.
16. Jordi Sabater and Carles Sierra. REGRET: a reputation model for gregarious societies. In Jörg P. Müller, Elisabeth Andre, Sandip Sen, and Claude Frasson, editors, *Proceedings of the Fifth International Conference on Autonomous Agents*, pages 194–195, Montreal, Canada, 2001. ACM Press.
17. Jordi Sabater and Carles Sierra. Reputation and social network analysis in multi-agent systems. In *AAMAS '02: Proceedings of the first international joint conference on Autonomous agents and multiagent systems*, pages 475–482, New York, NY, USA, 2002. ACM Press.
18. M. Schumacher and S. Ossowski. The governing environment. In Weyns, Parunak, and Michel, editors, *Environments for Multiagent Systems II*, volume 3830, pages 88–104. Springer-Verlag, 2006.
19. Murat Sensoy and Pinar Yolum. A context-aware approach for service selection using ontologies. In *Proceedings of the Fifth International Joint Conference on Autonomous Agents and Multiagent Systems, AAMAS 2006*, pages 931–938, 2006.
20. Juan M. Serrano, S. Ossowski, and A. Fernández. The pragmatics of software agents - analysis and design of agent communication languages. *Intelligent Information Agents - An AgentLink Perspective (Klusch, Bergamaschi, Edwards & Petta, ed.), Lecture Notes in Computer Science*, 2586:234–274, 2003.
21. Juan M. Serrano and Sascha Ossowski. On the impact of agent communication languages on the implementation of agent systems. *Cooperative Information Agents VIII (Klusch, Ossowski & Kashyap, ed.)*, 2004.
22. W. T. Luke Teacy, Jigar Patel, Nicholas R. Jennings, and Michael Luck. Travos: Trust and reputation in the context of inaccurate information sources. *Autonomous Agents and Multi-Agent Systems*, 12(2):183–198, 2006.
23. Javier Vázquez-Salceda, Virginia Dignum, and Frank Dignum. Organizing multiagent systems. *Autonomous Agents and Multi-Agent Systems*, 11(3):307–360, 2005.
24. Bin Yu and Munindar P. Singh. An evidential model of distributed reputation management. In *AAMAS '02: Proceedings of the first international joint conference on Autonomous agents and multiagent systems*, pages 294–301, New York, NY, USA, 2002. ACM Press.
25. Bin Yu, Munindar P. Singh, and Katia Sycara. Developing trust in large-scale peer-to-peer systems. In *Proceedings of First IEEE Symposium on Multi-Agent Security and Survivability*, pages 1–10, 2004.
26. Franco Zambonelli, Nicholas R. Jennings, and Michael Wooldridge. Organizational abstractions for the analysis and design of multi-agent systems. In Paolo Ciancarini and Michael J. Wooldridge, editors, *AOSE*, volume 1957 of *LNCS*, pages 235–252. Springer, 2000.