Agent UNO: Winner in the 2nd Spanish ART competition

Víctor Muñoz, Javier Murillo

Universitat de Girona
Avda. Lluís Santaló, s/n
Girona, 17071
{vmunozs,jmurillo}@eia.udg.edu

Abstract

In multi-agent systems where agents compete among themselves, trust is an important topic to have in mind. The ART Testbed Competition has been created with the objective of evaluating objectively different trust strategies that agents can use in this kind of environments. In this paper we present the winning strategy in the 2nd Spanish ART competition together with an analysis of the factors that have contributed to this success. We also present the results obtained using the same strategy in the 2nd International ART competition.

Keywords: ART Testbed, trust, reputation, agent, UNO.

1 Introduction

In shared and competitive environments, agents interact with each other in order to achieve their goals. This interaction allows them to obtain better results than they would get isolatedly. However, since agents are not interested in a global outcome but only on their own, maybe some of this interaction is done with the intention of disserving other agents. In such situations agents may need to use a trust and reputation mechanism, providing an uncertainty model that would allow them to discern other agents' behaviors and intentions, by means of whom the agent would be able to select when and which agents to trust.

In recent years there has been a growing interest un trust mechanisms for multi-agent systems [11] and a large number of models and strategies have been proposed to deal with this [14, 3, 9, 16]. Unfortunately, models have been usually tested in dissimilar problems. In consequence, with the goal of providing a "*common platform on which researchers can compare their technologies against objective metrics*" [5] the Agent Reputation and Trust (ART) Testbed Competition was created in 2006, at both national and international level [1]. This competition serves also as an impulse to promote research in this field and to design innovative strategies that are also applicable to real-world situations.

The ART Testbed is a game where the participating agents act as painting appraisers with varying levels of expertise in different artistic eras [7, 6, 5, 8]. There is a fixed total number of clients who request appraisals for their paintings from different eras. Appraisers receive more clients, and thus more profit, for giving more accurate appraisals. It may be the case that, for some received requests, an appraising agent does not have sufficient expertise to complete an acceptable appraisal; in such a case it can be improved by

ISSN: 1137-3601.

requesting opinions from other appraiser agents, against a payment. However, it is not necessary for the requested agents to provide good information. They can decide, for each agent's request, whether to lie or to tell the truth. In fact, as a result of the competition among agents, it is quite likely that agents will lie frequently. Thus, trust and reputation become important factors for an agent to obtain better results [15, 10, 4, 17]. For more information about the ART Testbed simulator and the rules of the International ART competition see [1]. Figure 1 shows the main interface of the ART Testbed simulator used for the competitions.

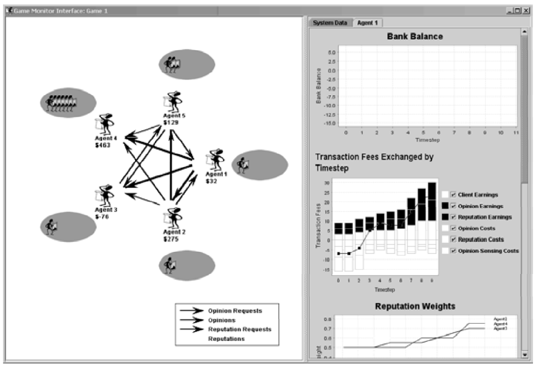

Figure 1. ART Testbed simulator

In this paper we describe the strategy used by agent UNO, which allowed it to win the 2nd Spanish ART competition held in Valencia in March 2007. The paper is structured as follows. The following section describes the general procedure of the agent UNO. The next two sections show two fundamental components in its strategy, specifically the question and answer procedures. Then we present the results obtained by agent UNO in both national and International ART competitions of 2007. We finally derive some conclusions and outline future improvements.

2 General considerations

For the design of the agent UNO two parts of its behavior have been studied independently, concretely the *asking* and *answering* procedures. In the former the agent decides, for the paintings that it has received from its clients, which agents to request information from and which weight to give to their answers (how much to trust them). On the other hand in the answering procedure the agent receives a set of questions coming from other appraisal agents and determines how much money to spend on each appraisal (including which agents to lie to).

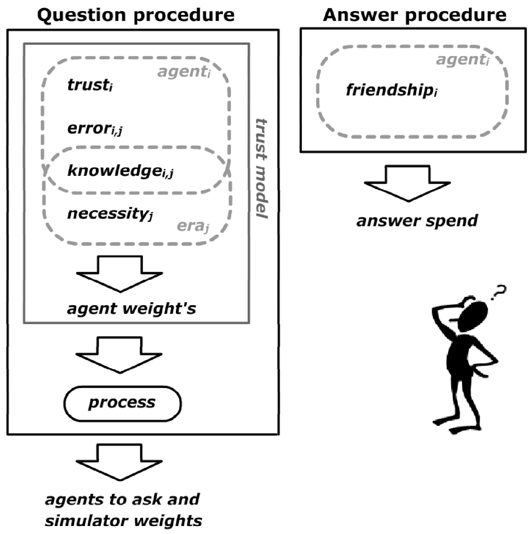

Figure 2. Schema of agent UNO

Figure 2 shows a general operation schema of the agent UNO. The question procedure (at the left side) is composed by four main components: trust, error, knowledge and necessity. These components compute a weight for each agent, used to determine which agents to ask and how much to trust each one (the translation from the trust model to the simulator weights is what is called "process" in the figure). Hence all these elements together compose the trust model of the agent UNO.

At the right side of the figure the answer procedure shows that UNO uses an element called *friendship* to determine how much to spend for the appraisals received from other agents. Both procedures make use of the knowledge that the agent has about the other participants. The information that UNO stores about other agents is:

- Error committed by agents on their appraisals.
- Whether the agent answers or not to the questions UNO asks him.
- The total number of questions the agent has asked UNO.

We could also consider the certainty value that agents inform before performing an appraisal indicating their expertise on the respective era. However we have decided not to consider it; we believe it is more robust to work directly with the real error committed by each agent. Moreover, considering that there are few agents participating in a game, the savings obtained by not asking an agent after considering its certainty level, compared with always asking without taking it into account is not much significant.

Another feature that we have decided not to use is the reputation. The reasons are the same as for the certainty, in addition to that it is possible, for the agent asked about the reputation of another agent, to have insufficient knowledge of him. Furthermore, it is possible for an agent to behave opposedly for different agents. Therefore we decided not to use reputation, basing our strategy uniquely on the knowledge that we learn directly from other agents. This feature gains special meaning when there are few agents participating in a game. Anyway in the 2nd International ART competition we introduced a variant of agent UNO (called Marmota) to use reputation, however the results were not better (see the results section for more information).

3 Questions Procedure

In order to know which agents to ask to, agent UNO calculates a weight $P_{i,j}$ representing how much to trust other agents' appraisals (including the agent UNO itself), for each agent and each era (given that agents have different expertise values for each era). This value is later used to inform the simulator about the weights that will be assigned to each agent to make the final appraisal. The weight of an agent i in the era j is calculated using the following formula:

$$P_{i,j} = (1 - e_{i,j}) \cdot t_i \cdot k_{i,j} \cdot ne_j \qquad (1)$$

All the elements in this formula ($e_{i,j}$, t_i, $k_{i,j}$ and ne_j) are explained below:

- **Error**. The value $e_{i,j}$ stands for the error committed by the agent i on the era j. It is measured as an average of the errors produced in all the appraisals done by agent i for the paintings concerning the era j.

$$e_{i,j} = \sum_{p \in P_{i,j}} \frac{error_{i,p}}{|P_{i,j}|} \qquad (2)$$

where $P_{i,j}$ is the set of all painting appraisals by agent i on era j, $|P_{i,j}|$ is the number of elements of this set and $error_{i,p}$ is the relative error committed by agent i on the appraisal of the single painting p, calculated as follows:

$$error_{i,p} = \frac{rValue_p - aValue_{i,p}}{rValue_p} \qquad (3)$$

where $rValue_p$ is the real value of painting p (this value is known after each round of appraisals in the game) and $aValue_{i,p}$ is the appraised value for the painting p by agent i.

- **Trust**. The value t_i reflects the certainty that agent UNO has about agent i to tell him the truth. It is obtained computing the percentage of *lies* committed by this agent in relation to the total number of appraisals made.

$$t_i = 1 - \frac{lies_i}{totalAppraisals_i} \qquad (4)$$

We consider that an agent has lied intentionally when the error committed for a single appraisal $error_{i,p}$ is higher than a certain threshold.

We have calculated this threshold looking at the formula used by the simulator to compute the standard deviation of the error committed by an agent to an appraisal. According to [5] the formula is $s = (s^* + \frac{\alpha}{C_g}) \cdot t$, where s is the standard deviation of the error, s^* is the expertise that the agent has on the era related to the painting, α is a constant value, C_g is the money spent by the agent to perform the appraisal and t is the real value of that painting. If we assume $s^* = 1$ (i.e. the agent has the worst possible expertise value on that era), and set $\alpha = 0.5$ (its default value), we obtain that with $C_g = 3$ the standard deviation is $s = 1.66$. Looking at table of the normal distribution with this standard deviation we see that the area falling out of this value (that is, the probability of making an error greater than 3) is 0.0094. Hence, given that it is highly improbable (0.94%) for an agent to produce such an error still assuming the agent to be completely inexpert on that era, we consider that an error greater than 3 certainly indicates an intentional lie.

After that, the trust value t_i is modified to avoid trusting agents that do not trust agent UNO. We have called this *bilateral trust*. The idea is that if an agent is never making questions to UNO, it is probably because he does not trust in agent UNO, thus this agent is more likely to lie UNO, therefore we should set our trust in him to 0. Obviously, the trust that UNO has in himself is always 1 (UNO never lies to himself).

- **Knowledge**. The value $k_{i,j}$ represents the degree of information (between 0 and 1) that we have about the agent i on the era j. We begin defining a value that is the minimum number of questions that an agent has to have answered to UNO (on each era), to consider that we completely know him. This value is called the minimum knowledge ($minKnow$). Then $k_{i,j}$ is computed as the percentage of answered questions in relation to this minimum knowledge.

$$k_{i,j} = \min\left(\frac{totalAQ_{i,j}}{minKnow}, 1\right) \quad (5)$$

where $totalAQ_{i,j}$ is the total number of questions answered by agent i on era j. If an agent has done more than $minKnow$ questions our knowledge on him is also 1, therefore a maximum of 1 is applied. Like in trust, the knowledge of ourselves is always 1.

- **Necessity**. The necessity n_j stands for the urgency that the agent UNO has in an era to ask others for help. As more expert the agent is in the era, the lower is its necessity. Necessity values are predefined for each era j and for each possible expertise value of an agent following this formula:

$$n_j = (expertise_{UNO,j})^{\alpha} \quad (6)$$

Necessity affects the final weights, assigning more importance to the agent UNO (the necessity is inverted $1 - n_j$ when the agent considered is UNO itself) in the eras where it is more expert (it has low expertise values). Although other agents may be still more expert on some eras, they can also lie sporadically, thus we consider that it is better to have a controlled small error than an uncontrolled very-small error. Of course, in the eras where the agent is not expert, as its necessity is higher, the weights assigned to other agents are significant.

The necessity values can be adjusted varying the parameter α (currently set to 1) in the following way: if $\alpha > 1$ then all the necessities turn lower (thus we only request help to other agents in eras where we are completely inexperts), otherwise if $\alpha < 1$ (and greater than 0) then all the necessities turn higher (we request help for all the eras except the ones in which we are particularly experts).

Once all the data of the trust model has been collected, the final simulator weights are calculated. This process is divided into 6 steps, as it can be seen in the example of the figure 3.

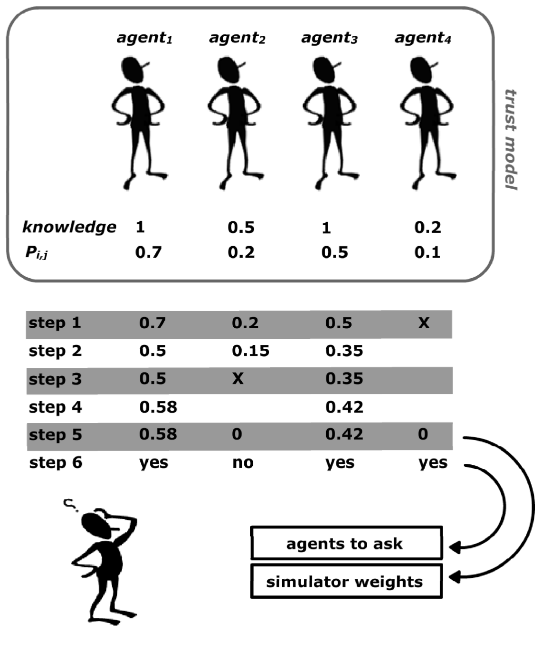

Figure 3. Calculation of weights (example).

- **Step 1.** This step reduces the total number of agents to ask for each appraisal to a maximum of n. This limit is calculated based on the percentage of earnings (money won with our appraisals) that we want to spend asking for other agents' appraisals, according to this formula:

$$n = apCost \cdot \frac{percentageWaste}{opCost} \quad (7)$$

where *apCost* is the cost of an appraisal, *percentageWaste* is the percentage of earnings that UNO wants to spend in making questions to other agents and *opCost* is the cost of a single opinion. After that, the top n weights of each era are kept, while the others are set to 0.

- **Step 2.** The second step consists in a normalization of the remaining weights to achieve that the sum of them is exactly one.
- **Step 3.** It may be the case that some of the final selected weights are so close to zero that would barely affect on the final appraisal (nonetheless we should pay the same money for asking them). In order to avoid these kind of questions we apply two thresholds, one absolute (before the normalization) and another relative (after the normalization), that set to zero those insignificant values.
- **Step 4.** Then, as some weights may have been set to zero, another normalization is performed.
- **Step 5.** The weights higher than 0 represent the agents that we have decided to ask for help, in the next step some more agents may be also added.
- **Step 6.** In this step the agent checks how many agents are going to be asked, and in case that the agent is going to ask less questions than the previously calculated limit n (in step 1), the remaining questions are randomly made to the agents that we know less about, in order to increment our knowledge of them. Of course, if we already have sufficient knowledge (*minKnow*) of every agent, these additional questions are not made.

4 Answers Procedure

To decide the amount of money that agent UNO will spend on generating opinions for other agents, we first calculate the degree of *friendship* with each agent. The agent will then spend more money with the "friendliest" agents. The friendship with an agent i (f_i) is calculated with the following formula:

$$f_i = \sum_{j=1}^{eras} \max(error_{UNO,j} - error_{i,j}, 0) \qquad (8)$$

This calculation results in an "interested" friendship, since it actually measures the amount of information other agents do have and UNO does not. Consequently, agent UNO spends more money on the agents from whom it needs more their information (i.e. they are expert in the eras where UNO is not), with the aim of obtaining a mutual collaboration with these agents in a way that they contribute providing their valued knowledge.

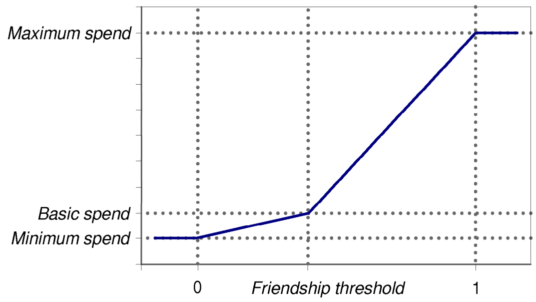

Figure 4. Friendship function

To calculate how much money to spend on each agent, a function is defined as shown in figure 4. The friendship value of an agent is a value between 0 and 1, with an intermediate value named *friendship threshold* which indicates the value at which an agent becomes a friend. The function presents two parts, the left part is for non-friend agents and the right for friend agents. Then, a maximum, minimum and basic spends are defined in such a way that the agent UNO does not spend more than a basic amount of money with non-friend agents, but spends up to a maximum quantity for friend agents.

This mechanism favours the interaction among agents needing mutual help, increasing both their benefits, although on the other hand the non-friend agents are somehow discriminated.

5 Results

During the development of agent UNO, the agents that participated in the 1st International ART competition of 2006 were used for testing. This competition was celebrated in Hakodate (Japan) and the agent IAM [17] resulted the winner. In figure 5 we show the results of our agent UNO competing amongst the five finalist agents on that competition; the graph shows the average and the standard deviation of the earnings of each agent after repeating the game 50 times. Agent UNO is the one having the higher average of money, and its standard deviation is very similar to the second agent (IAM).

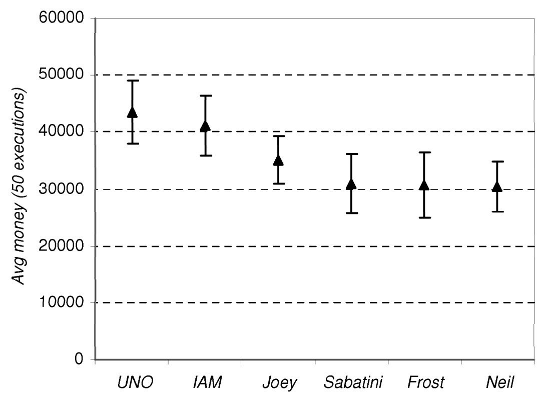

Figure 5. Agent UNO versus the finalist agents in the 1st International ART competition.

Agent UNO has participated in the 2nd Spanish ART competition celebrated in Valencia [2] obtaining the first position. In this competition 13 agents were inscribed. The competition consisted in a serie of 6 games with 5 different agents each. Each of the games was repeated 3 times. The results are shown in figure 6, where the Y axis represents the total earnings obtained for each agent in all the games where it participated. In this competition 3 variants of the agent UNO where presented (UNO_TDI, UNO_VTR and UNO_F40) obtaining the 3 first positions.

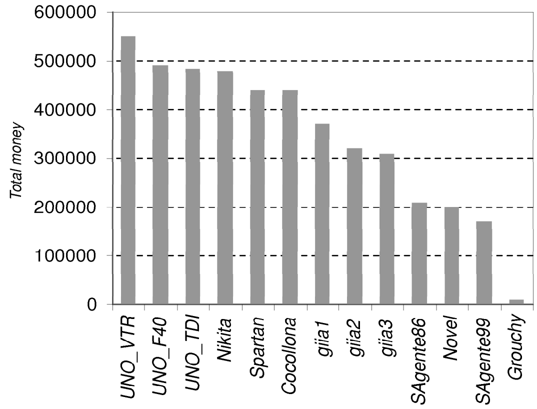

Figure 6. Final classification in the 2nd Spanish ART competition

The three variants are the same agent described in this paper with slight differences. The difference between UNO_TDI and UNO_VTR is the initial values of knowledge (UNO_TDI has an initial value of 0 while UNO_VTR has 0.5). Also the minimum knowledge (*minKnow*) is lower (therefore requiring fewer questions to detect liars) in the UNO_VTR than UNO_TDI. Finally UNO_F40 is the same as UNO_VTR but presents a new feature turning the agent into an absolute liar from a given time step until the end of the game. Seeing the results we can oberve that the agent UNO_VTR (not liar) obtains better results than his liar version (UNO_F40).

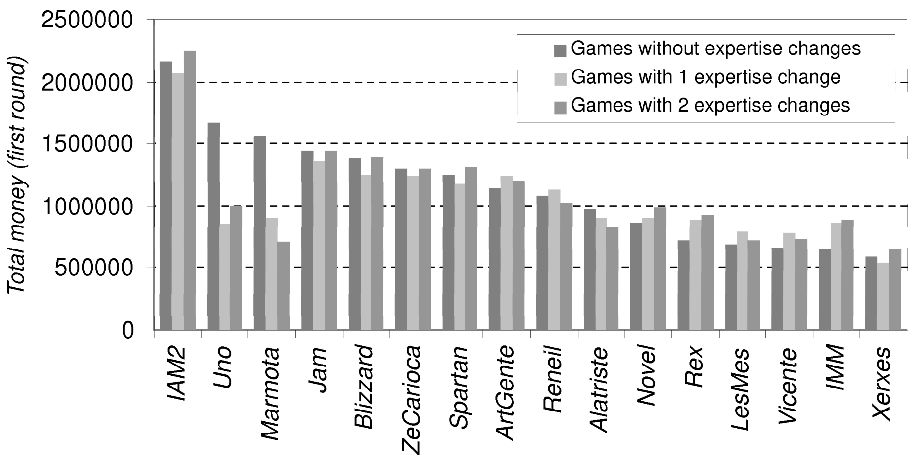

Figure 7. Classification at the first round of the 2nd International ART competition

Our agent has also participated in the 2nd International ART competition of 2007 held in Honolulu (Hawaii, USA), which consisted in a first round and the final round. In the first round 8 games were played with 8 agents each. Games were played in 3 variants. In the first variant, expertise values of the agents were maintained during all the game, in the second variant there was one single change in the expertise values of all the agents at some point of the game, and the third variant had two changes.

Agent UNO was not prepared to work with expertise changes during the game and this was a handicap that caused its elimination on the first round. However, the two versions of the agent presented (UNO and Marmota) obtained the seventh and ninth position respectively (of a total of 18) which is not bad considering that it was at a disadvantage. Furthermore, if we analyze the 3 variants individually, as it is shown in figure 7, our agents finalized second and third in the games without expertise changes.

6 Conclusions

In this paper we have presented the strategy used in agent UNO int the 2nd Spanish ART competition and 2nd International ART competition both celebrated in 2007. The agent was the winner of the national competition and finalized seventh in the international. The results are encouraging given that, even though the final position in the international competition was not so good, if we consider only the games without expertise changes, UNO was the second agent obtaining more profit.

Viewing the results, we can draw out some conclusions that are in fact applicable to any multiagent system where trust and reputation play relevant roles.

In general, lying is not a good strategy, since other agents generally detect the liar and immediately stop trusting him. Moreover the agents begin also to lie to the liar, so he ends up forced to trust only in himself, and it is clear that the results using only one's own expertise are worse than getting help from other (collaborating) agents.

It is usually better to know about other agents by your own experiences uniquely, instead of asking other agents the opinion they have on them, foremost when the number of participating agents in a game is not very large. This conclusion is supported by three reasons: firstly, the agents maybe do not know anything about the agent we ask for; secondly, that agent may behave differently for each agent so the received reputation opinion may be not valid for us; and third, the agent can decide to lie us about that agent, for example saying that the agent is a complete liar when he is not, in order to avoid other agents to get also beneficiated from him. On the other hand, in games with a large number of agents participating, this feature would have more sense since it would require two much time for the agent to know of everyone based only on its own experience.

Starting with a low trust on other agents, and increasing it as the agent gathers more knowledge about them, as well as not to trust agents that do not trust agent UNO, induces robustness in the agent specially in games where there are liar agents participating.

7 Future work

Future versions of the ART Testbed will include much more agents on each game, the games will be longer and will incorporate various kinds of changes apart from the changes in expertise, as for example the inclusion of new agents in running games, changes in the behavior of some agents, etc. In order to prepare the agent for these changes, the following points will be considered on new versions of agent UNO:

- The first point to improve the agent UNO for future competitions is to provide the agent with a mechanism to be aware of the possible expertise changes occurring during the game. There are two possibilities for designing this mechanism.
  - The first alternative is the elimination of all the information stored about the agents when an expertise change occurs, followed by immediately starting collecting new data.
  - The second alternative is the incorporation of limited memory to the agent. A memory of tm timesteps would cause the entire trust model to take into account only the information

of the last tm timesteps. This limited memory would also help the agent to adapt to any other kind of changes.

- The second point is the utilization of reputation. In future games it will be imperative to use reputation since there will be a large number of participating agents, therefore the agent will only ask to a small subset of them because it will be too expensive to ask all of them. The selected agents to ask will be the ones with the highest reputation. Reputation can be used also to determine the simulator weights.
- Another point is to take into account the certainty values. These values can let the agent learn the behavior of other agents by determining the relation between the given certainty and the quality of the opinion.
- We think that in future versions of the ART Testbed simulator the use of physical memory must be controlled and limited, furthermore seeing that it is planned to run games with a lot of agents and timesteps, therefore the agents will not be able to store all the information of the game. The limited memory (described before) deleting old unnecessary data can be used to achieve this.
- Finally, we also plan to use the knowledge about trust mechanisms acquired during the development of the agent UNO by applying it to real-world problems as for example in the Waste Water Treatment Plant domain [12, 13]. In this domain there are a number of industries discharging waste to the sewage and the treatment plant processes it to generate a clean waterstream that can be put back into the river. Here industries inform to the WWTP about his discharges and the plant uses this information to coordinate their discharges in order to do not overflow its capacity. Sometimes the industries lie to the WWTP (about the discharges they are going to perform), and also sometimes they disobey the decisions of the plant; therefore a trust mechanism would help the plant to generate more robust plans.

Acknowledgements

We would like to thank Beatriz López and Dídac Busquets for their help and support during the development of the Agent UNO.

References

[1] Agent Trust and Reputation Testbed. http://www.art-testbed.net.

[2] Spanish ART Testbed competition 2007. http://megatron.iiia.csic.es/spanishART/.

[3] Rajdeep Dash, Sarvapali Ramchurn, and Nicholas Jennings. Trust-based mechanism design. In *Proceedings of the Third International Joint Conference on Autonomous Agents and MultiAgent Systems*, pages 748–755. ACM, 2004.

[4] Karen Fullam and Suzanne Barber. Learning trust strategies in reputation exchange networks. *The Fifth International Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS-2006)*, 2006.

[5] Karen Fullam, Tomas Klos, Guillaume Muller, Jordi Sabater, A. Schlosser, Z. Topol, Suzanne Barber, J. Rosenschein, Laurent Vercouter, and M. Voss. A specification of the agent reputation and trust (art) testbed: Experimentation and competition for trust in agent societies. *Proceedings of the 4th International Join Conference on Autonomous Agents and Multiagent systems, pp. 512-518*, 2005.

[6] Karen Fullam, Tomas Klos, Guillaume Muller, Jordi Sabater, Z. Topol, J. Rosenschein, and Laurent Vercouter. The agent reputation and trust (art) testbed architecture. *Frontiers in Artificial Intelligence and Applications. IOS Press, Vol 131, pp 389-396 (Proceedings CCIA-05)*, 2005.

[7] Karen Fullam, Tomas Klos, Guillaume Muller, Jordi Sabater, and Laurent Vercouter. The agent reputation and trust (art) testbed. *Lecture Notes in Computer Science, Proc. iTrust06*, 2006.

[8] Karen Fullam, Jordi Sabater, and Suzanne Barber. Toward a testbed for trust and reputation models. *Trusting Agents for Trusting Electronic Societies, Springer, pp. 95-109*, 2005.

[9] T. Dong Huynh, Nicholas R. Jennings, and Nigel R. Shadbolt. Developing an integrated trust and reputation model for open multi-agent systems. In *AAMAS-04 Workshop on Trust in Agent Societies.*

[10] O. Kafali and P. Yolum. Trust strategies for art testbed. *The Workshop on Trust in Agent Societies at The Fifth International Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS-2006)*, 2006.

[11] Ana Mas. *Agentes Software y Sistemas Multi-Agente*. Prentice-Hall, 2005.

[12] Víctor Munoz, Javier Murillo, Dídac Busquets, and Beatriz López. Improving water quality by coordinating industries schedules and treatment plants. In Mathijs Michiel de Weerdt, editor, *Proceedings of the Workshop on Coordinating Agents' Plans and Schedules (CAPS)*, pages 1–8. IFAAMAS, may 2007.

[13] Javier Murillo, Víctor Muñoz, Beatriz López, and Dídac Busquets. Dynamic configurable auctions for coordinating industrial waste discharges. In Paolo Petta, Jörg P. Müller, Matthias Klusch, and Michael P. Georgeff, editors, *MATES*, volume 4687 of *Lecture Notes in Computer Science*, pages 109–120. Springer, september 2007.

[14] Sarvapali D. Ramchurn, Dong Hunyh, and Nicholas R. Jennings. Trust in multi-agent systems. *Knowledge Engineering Review*, 2004.

[15] S. Sen, I. Goswami, and S. Airiau. Expertise and trust-based formation of effective coalitions: An evaluation of the art testbed. *The Workshop on Trust in Agent Societies at The Fifth International Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS-2006)*, 2006.

[16] R. Stranders. Argumentation based decision making for trust in multi-agent systems. *Master's Thesis, Delft University of Technology*, 2006.

[17] W. T. L. Teacy, T. D. Huynh, R. K. Dash, N. R. Jennings, M. Luck, and J. Patel. The art of iam: The winning strategy for the 2006 competition. In *The 10th International Workshop on Trust in Agent Societies*, pages 102–111, 2007.