Certified Trust Model

[1]Vanderson Botêlho, [1]Fabríco Enembreck, [1]Bráulio C. Ávila, [2]Hilton de Azevedo, [1]Edson E. Scalabrin

[1]PUCPR, Pontifical Catholic University of Paraná

PPGIA, Graduate Program on Applied Computer Science

R. Imaculada Conceição, 1155

Curitiba PR Brazil

{vanderson, fabricio, avila, scalabrin}@ppgia.pucpr.br

[2]UTFPR, Federal Technological University of Paraná

PPGTE, Graduate Program on Technology

Av. 7 de Setembro, 3165

Curitiba, PR, Brazil

hilton@utfpr.edu.br

Abstract This paper presents a certified confidence model which aims to ensure credibility for information exchanged among agents which inhabit an open environment. Generally speaking, the proposed environment shows a supplier agent *b* which delivers service for a customer agent *a*. The agent *a* returns to *b* a cryptographed evaluation *r* on the service delivered. The agent *b* will employ *R* as testimonial when requested to perform the same task for a distinct customer agent. Our hypotheses are: (i) control over testimonials can be distributed as they are locally stored by the assessed agents, i.e., each assessed agent is the owner of its testimonials; and (ii) testimonials, provided by supplier agents on their services, can be considered reliable since they are encapsulated with public key cryptography. This approach reduces the limitations of confidence models based, respectively, on the experience resulted from direct interaction between agents (*direct confidence*) and on the indirect experience obtained from reports of witnesses (*propagated confidence*). Direct confidence is a poor-quality measure for a customer agent *a* hardly has enough opportunities to interact with a supplier agent *b* so as to grow a useful knowledge base. Propagated confidence depends on the willingness of witnesses to share their experiences. The empiric model was tested in a multiagent system applied to the stock market, where supplier agents provide recommendations for buying or selling assets and customer agents then choose suppliers based on their reputations. Results demonstrate that the confidence model proposed enables the agents to more efficiently choose partners.

1 Introduction

Nowadays, distributed and flexible approach seems to be a good to complex applications that deal with huge amount of data and services, the main reason being the system necessity of dynamic adaptation to structure and environment changes. Thus, multiagent systems are good candidates for building distributed heterogeneous flexible open architectures that shall ensure a great amount of services in a collective work context with no *a priori* structure. Nevertheless, even if data and control distribution may bring reliability when considering service availability, the lack of a information centralizer adds weakness to the trust relations the clients of a service or product and its service providers or hosts. For Huynh *et al* [2], the basic question is: in an open system how can an agent trust in a stranger?

The way to get the value that represents the level of trust depends on the system architecture, particularly, in how it allows getting and giving feedbacks. For instance, in an eBay like system [6], where transactions are made by people, the delays may attain days. On another hand, in P2P systems [7], where transactions may be concluded in some milliseconds. In this case, scalability becomes a crucial factor, requesting a distributed trusting model to bring more reliability when comparing with centralized models.

Studies were made in order to reduce the interaction risk between agents in open systems. Castelfranchi and Falcone [10] consider the trust relation inside a MAS as a mental state that is essential to allow delegation mechanisms between agents. Other works [11], [12] sustain that trust may be useful for reducing the risk related to interactions among agents. Mui et al. [13] consider trust as a multidisciplinary subject, representing it by the use of ontologies. They divide trust definitions as direct and indirect. REGRET [8] combines the models of direct and propagated trust and defines three agent interaction dimensions (i.e. individual, social and ontological). In the Individual Dimension, trust is obtained by direct interaction. In the Social Dimension, trust is obtained by indirect interaction (i.e. testimonies). In the Ontological Dimension, trust is obtained by the combination of both. Huynh et al. [2] propose a trust model based on *certified reputation*, which combines both direct trust and indirect trust by testimonies.

The trust model based in *certified reputation* [2], where testimonies are store locally by the agents that were assessed, has two advantages: (i) The assessment agent shares its experience only once and, (ii) in order to obtain a trust information it's necessary only two agents. Nevertheless, the set of trust assessments about a service provider agent may be changed by that agent in order to better notify its reputation, i.e. a service provider agent may inform, when asked, only its positive assessments, omitting its bad ones. This arbitrary selection adds distortions when computing the trust and decreases the efficiency of agents when choosing partners. Our proposal is to enhance the model of certified reputation by the use of assessment that has signatures made by asymmetric keys [14], doing so, the assessment content can not be read by the assessed agent.

Section 2 presents our model of certified trust. Section 3 describes a test scenario where, in a multiagent system, provider agents give recommendations about buying and selling assets for client agents. The client agents have the interest in select the best provider agents. Section 4 illustrates how the experiment has being conducted and discusses the results till now.

2 Certified Trust Model

Basically, a trust model takes into account an agent *a* that quantifies the trust it has regarding an agent *b* [2]. For example, agent *a* is the *evaluator* and the agent *b is the target*. A rating is calculated based on the past experiences regarding the quality of a service made by an agent to the other. Every *rating* is represented by a *tuple* $r=(a,b,i,v,c)$, where *a* and *b* are agents that participate in a interaction *i* and *v* is the assessment made by *a* over *b* about a given term *c*. Every assessment is stored locally by the service agent that was evaluated. So, when asked by a client agent, it can inform about the assessment results it had before. Term *c* brings to the trust model the capability assess every agent in different contexts. For instance, every evaluation is given for a specific time. The notation of the trust from *a* on *b*, about the term *c* is $T(a,b,c)$. Quantifying trust requires a set of relevant assessments. This set is notated as $R(a,b,c)$ and is the basis for the certified trust model we propose.

2.1 Model Definition

The certified model follows a typical scenario. An agent *b* provides a service to a client agent *a*. The client agent *a* returns agent *b* an assessment *r* about the service performed. Agent *b* stores *r* locally and will use *r* as a testimony if it is inspected by another client agent to realize the same kind of service. Target agent can not modify assessment contents or inform only about a minimal set of its better assessments. The reason is that assessments are signed by their evaluators. Only evaluators can know about the assessment contents regarding a target agent.

When an interaction *i* ends up, the target agent *b* asks the client agent *a* to assess its performance *v* about a given term *c*. This ends in a rating $r=(a,b,i,v,c)$. Agent *b* stores the rating inside its local repository. When an client agent *a* informs its interest on a term *c*, from a provider *b*, *b* answers informing its more relevant ratings *R*. This approach reduces the problem when an evaluator agent refuses to share its experiences. Another advantage is that the request is made one time and only two agents are concerned by the procedure. The calculus is made by weighed mean (Equation 1) of all ratings returned by the target agent. *Ratings* have a coefficient that decreases as the rating gets older. The calculus of a *rating r* in function of time is named $\omega(r_i)$, with $(\omega(r_i) \geq 0)$. The calculus of trust is defined by:

$$\mathrm{T}(a,b,c) = \frac{\sum_{r_i \in R(a,b,c)} \omega(r_i) \cdot v_i}{\sum_{r_i \in Rc(a,b,c)} \omega(r_i)} \quad (\mathbf{1})$$

Rating coefficients have their values decreased (Equation 2) by a time dependent exponential law; this makes old ratings few significant or irrelevant. This is important because it allows a client agent detect changes in quality of services provided by a provider more quickly, because the recent ratings have more relevance than the other ratings:

$$\omega \mathrm{Re}(r_i) = e^{-\frac{\Delta t(r_i)}{\lambda}} \quad (\mathbf{2})$$

Where $\omega\mathrm{Re}(r_i)$ is the value of coefficient r_i related to the time variation $\Delta t(r_i)$, that means the time elapsed between the time at the moment of the request and the moment the rating was created. Finally, λ is the factor that determines the coefficient decreasing speed related to time.

We point that there is no guarantee that the agents are honest on their assessments or that their capabilities to assess service agents are inaccurate or imprecise. Our trust certified model reduces this problem introducing in the process the *credibility* of the evaluator agent as another element in order to determine the relevance of a specific rating inside a trust calculus. This process determines how an evaluator is reliable and can be calculated when customers evaluate their personal interactions in order to compare with the ratings received. The *credibility* of an assessment agent *w* is calculated by another assessment agent *a* and, is named $T\mathit{RCr}(a,w) \in [-1,+1]$, where *RCr*. A rating weight is related to the time $\omega\mathrm{Re}(r_i)$ and the credibility $\omega\mathit{RCr}(r_i)$:

$$\omega c(r_i) = \omega \mathrm{Re}(r_i) \cdot \omega\mathit{RCr}(r_i) \quad (\mathbf{3})$$

When $\omega\mathit{RCr}(r_i)$ is negative, the assessment agent has no credibility and its rating is adjusted to zero:

$$\omega\mathit{RCr}(r_i) = \begin{cases} 0 \quad se \quad \mathit{TRCr}(a,w) \le 0 \\ \mathit{TRCr}(a,w) \quad se \quad \mathit{TRCr}(a,w) > 0 \end{cases} \quad (\mathbf{4})$$

The scenario considers a minimum of three agents: *a, b* and *w*. Considering that *a* assess *b* and *b* stores locally its rating, given by $r_a=(a,b,i_a,c,v_a)$. When agent *a* receives a *rating* of another evaluator agent, at this case agent *w*, *a* calculates w credibility by comparing the performance of agent *b* (i.e. v_a) with the evaluation made by *w* over *b*. The *rating* of *w* related to *b* is given by $r_w=(a,b,i_w,c,v_w)$. The credibility of agent *w* is obtained by the difference between both values (v_a,v_w). It is expressed by $v_{k,}$ according to equation 5.

$$v_k = \begin{cases} 1 - |v_w - v_a| & se \quad |v_w - v_a| < \iota \\ -1 & se \quad |v_w - v_a| > \iota \end{cases} \quad \text{where: } (0 \le \iota \le 2) \tag{5}$$

So, v_k receives a positive value if the difference between v_w and v_a stays below the limit ι, otherwise, the credibility is negative, i.e. the evaluator agent can not be trusted.

The honesty of provider agents when they envoy their ratings is granted by a digital signature based on asymmetric keys. The signature is composed by both, private and public keys. With this method, a System Administrator agent creates a code key for every kind of service *c*. This key is sent to all agents, it is a public key. Then the System Administrator agent creates a second key that is used only for decoding. This key is sent only for the evaluators agents.

Every time a evaluator/client agent sends a rating to a provider agent, the public key for service *c* is used to encrypt the value of *v*. As *v* can be decrypted only with the private key that belongs to the client agents, no provider agent can know about the value of *v* related to the rating *r*. This avoids that a provider agent selects a subset of relevant ratings *R*. In our experiments we used the Pretty Good Privacy (PGP) algorithm [16] to encrypt and decrypt the values of the ratings.

3 Experiment

We defined four behavior groups for the provider agents: *good providers* (which use a analysis method with gives high level of success to their recommendations), *bad providers* (with low level of success in their recommendations), *ordinary providers* (with a level of recommendations success around the average, i.e. a mobile average) and, *malicious providers* (this provider agents used the same method used by the third group but they order their ratings with the purpose of sending only the better ones and make difficult the differentiation between good and bad service provider agents).

Client agents are organized in four groups: *No_Trust* (the ones that do not have any trust model); *Direct_Trust*, (the ones that implement a direct trust model); *Cr_Trust*, (the ones that implement certified trust model based on certified reputation and; *Cryp_Trust*, (the ones that implement the certified trust model). Client agents interact with different kind of service provider agents and, according to the trust model they have, they select the service provider agent that seems to maximize their interests.

Every client agent starts consulting several service provider agents with whom it performs as many buying/selling orders of actions as necessary. The evaluation of the trust model is made by measuring the performance of every service agent portfolio. Every agent receives the same amount of money to invest. At the end of every working day, the percentage of every service agent portfolio is observed

growing. The agents acted over historical real data of Bovespa stock market [15]. To this experiment we considered only one kind of action quoted at Bovespa from January/2nd/2006 to December/18th/2007, totalizing 473 working days. Data regarding 2006 were used for training. At the end of 2006, the portfolios were restarted. Nevertheless, the agents kept the experience acquired during 2006 year. Then, during the year of 2007, every client agent (investor) evaluated the performance of its service provider agents (market expert) by using its trust model.

Every experiment was started by the creation of client and service agents. Service agents had only one strategy to perform financial analysis. Client agents had only one trust model. Client agent's utility gain, named UG, represents the utility gain of the trust model. At the end of every working day, the function of utility of client agents was added according to agent's trust model. The average of those values represented the utility gain of the trust model.

Four scenarios were set in order to evaluate the behavior of the trust model. At the Scenario I has service agents that are honest, despite they select their *ratings*, that selection do not disturbed their real performance because all service provider agents use the same technique of analysis during the all scenario (Table 1 presents the variables used).

Table 1. System variables for a honest environment.

Simulation variable	Symbol	Value
Number of simulation rounds	N	473
Total number of provider agents:	N_P	500
Good providers	N_{PG}	166
Ordinary providers	N_{OP}	168
Bad providers	N_{PB}	166
Malicious providers	N_{MI}	0
Number of consumers in each group	N_C	500

At the scenario II, service provider agents have different performance because their techniques of analysis change during the scenario. By doing so, a service provider agent that was using a very good analysis technique may have its performance decreased because it starts using a worse one, and vice-versa. Parameters defined at Table 2 ensure that scenarios I and II have service provider agents with rational behavior and constant performance due to the absence of agents from the *Malicious service provider* agent group. Here, all service provider agents used a same financial analytical technique.

Table 2. Parameters of the model.

Parameters	Symbol	Value
Speed Factor	λ	$-\frac{5}{\ln(0.5)}$
Maximum number of better ratings	NR	10
Credibility limit	ι	0.5

Figure 1 shows that at Scenario I all agents that use a trust model obtained similar results (+100%). This happened because service agents had completely predictable behaviors. In another hand, agent without trust model had their performance compromised (-23%). Figure 2 shows (Scenario II) that the *cryp_trust* model has the best performance in most of the time. Significant variations happen when a good service provider agent starts to have a bad performance (due to changes in the financial analytical technique). The reason is because the service provider agent sends good ratings, related to a recent past. This deceives the *cr_trust* model and decreases its performance.

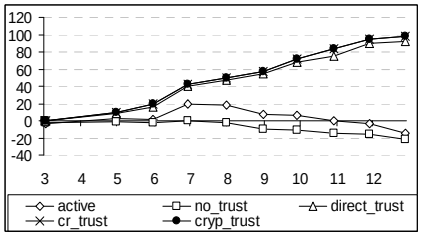

Fig. 1. Honest context without changes.

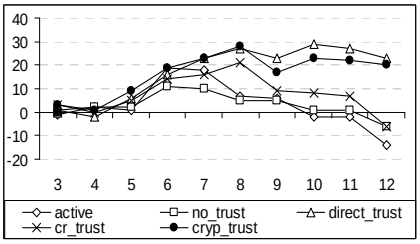

Fig. 2. Honest context with changes.

In scenarios III and IV, we introduce malicious service providers that even select and send their best ratings in order to influence calculus of trust made by the client agents. Similarly to scenarios I and II, at Scenario III service providers agents have a constant performance. At Scenario IV, they have variations in their performance. Table 3 shows the configuration used.

Figure 3 shows a simulation where service provider agents do not make changes in their financial analytical techniques, thus keeping their performance constant. The *cr_trust* model had the worst result (-58%). The reason is that the client agent is deceived by the malicious service agents that send ratings arbitrary selected. On the other hand, the *cryp_trust* model with crypted rating avoid malicious service agent to select their best ratings. As consequence, the *cryp_trust* model performance remains similar to the scenario where there are no malicious agents.

Table 3. System variables for dishonest context.

Simulation variable	Symbol	Value
Number of simulation rounds	N	473
Total number of provider agents:	N_P	500
Good providers	N_{PG}	100
Ordinary providers	N_{OP}	100
Bad providers	N_{PB}	100
Malicious providers	N_{MI}	100
Number of consumers in each group	N_C	500

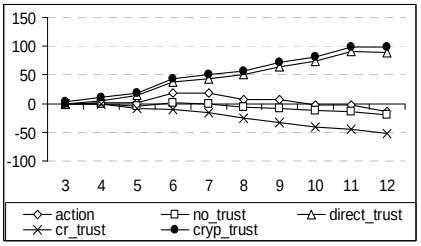

Fig. 3. Dishonest context without changes.

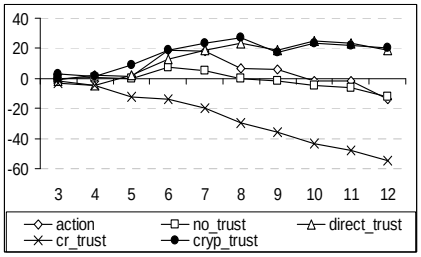

Fig. 4. Dishonest context with changes.

Scenario IV has the worst situation: the existence of malicious service provider agents and the variation of their performance due to changes, in runtime, in their financial strategies. Figure 4 shows that the *cr_trust* model presents much lower performance if compared to the others (-55%), even presenting a drop of performance when compared with scenario III the *cryp_trust* model keeps a positive performance (+20%). The great difference between both models is due to: the existence of malicious service providers and the changes of financial strategies in runtime.

5 Conclusion

We presented a certified trust model applied to multiagent systems (*cryp_trust model)*. The model enhances the concept of reputation from the certified reputation model. Both approaches, certified trust and certified reputation, use the assessment of service providers agents made by client agents. The assessments act as testimonies about their performance. The certified trust model allows increasing the system reability against malicious service provider agents that could try to manipulate the information concerning their performance in order to have some benefits. The results show that our certified trust model is more efficient specially in malicious scenarios. A key point is the use of asymmetric signing keys in order to protect and keep the ratings distributed.

Future works shall focus on online detection of service provider agents and the treatment of malicious agents. Here a hypothesis to improve the *cryp_trust model* may be the use of strategies for tendency change detection.

References

1. Wooldridge, M. and Jennings, N. R.: Pitfalls of agent-oriented development. In: Proceedings 2nd International Conf. on Autonomous Agents, Minnesota, United States (1998)

2. Huynh, T. D., Jennings, N. R., and Shadbolt, N. R: Certified reputation: how an agent can trust a stranger. In: Proceedings 5th international Joint Conference on Autonomous Agents and Multiagent Systems, Hakodate, Japan (2006)
3. Jennings, N. R., Huynh, D., Shadbold, N. R.: Developing an integrated trust and reputation model for open multi-agent systems. In: Proceedings 7th International Workshop on Trust in Agent Societies, New York, United States (2004)
4. Teacy, W. T., Patel, J., Jennings, N. R., and Luck, M: Coping with Inaccurate Reputation Sources: Experimental Analysis of a Probabilistic Trust Model. In: Proceedings 4th Inter. Joint Conf, on Autonomous Agents and Multiagent Systems. The Netherlands (2005)
5. Nguyen, G. H., Chatalic, P., and Rousset, M. C.: A probabilistic trust model for semantic peer to peer systems. In: Proceedings International Workshop on Data Management in Peer-To-Peer Systems, Nantes, France (2008)
6. Ebay Inc. http://www.ebay.com.
7. Aberer, K. and Despotovic, Z.: Managing trust in a peer-2-peer information system. In: Proceedings 10th international Conference on information and Knowledge Management, Atlanta, Georgia, United States (2001)
8. Sabater, J. and Sierra, C.: REGRET: reputation in gregarious societies. In: Proceedings 5th international Conference on Autonomous Agents, Montreal, Quebec, Canada (2001)
9. Ramchurn, S. D., Huynh, D., and Jennings, N. R.: Trust in multi-agent systems. Knowl. Eng. Rev. 19, 1 (Mar. 2004)
10. Castelfranchi, C. and Falcone, R.: Principles of Trust for MAS: Cognitive Anatomy, Social Importance, and Quantification. In: Proceedings 3rd international Conference on Multi Agent Systems. ICMAS, Washington, United States (1998).
11. Griffiths, N.: Task delegation using experience-based multi-dimensional trust. In: Proceedings 4th international Joint Conference on Autonomous Agents and Multiagent Systems, Netherlands (2005)
12. Fullam, K. K., Klos, T. B., Muller, G., Sabater, J., Schlosser, A., Topol, Z., Barber, K. S., Rosenschein, J. S., Vercouter, L., and Voss, M.: A specification of the Agent Reputation and Trust (ART) testbed: experimentation and competition for trust in agent societies. In: Proceedings 4th international Joint Conference on Autonomous Agents and Multiagent Systems, Netherlands, (2005)
13. Mui, L., Mohtashemi, M., and Halberstadt, A.: Notions of reputation in multi-agents systems: a review. In: Proceedings 1st International Joint Conference on Autonomous Agents and Multiagent Systems, Bologna, Italy (2002)
14. Rivest, R. L., Shamir, A., and Adleman, L.: A method for obtaining digital signatures and public-key cryptosystems. Commun. ACM 26, 1 (Jan. 1983)
15. Bovespa: Bolsa de Valores de São Paulo, http://www.bovespa.com.br
16. Branagan, J. Ippolito, K. Musgrave, and Waggenspack W.: Pretty good privacy. In: ACM SIGGRAPH 96 Visual Proceedings: the Art and interdisciplinary Programs of SIGGRAPH '96, New Orleans, United States, (1996)