Available online at www.sciencedirect.com
ScienceDirect
Electronic Notes in Theoretical Computer Science
ELSEVIER
Electronic Notes in Theoretical Computer Science 244 (2009) 3–12
www.elsevier.com/locate/entcs

An Evolutionary Trust and Distrust Model

Isaac Agudo, Carmen Fernández-Gago and Javier Lopez [1,2]

Department of Computer Science
University of Malaga
Malaga, Spain

Abstract

In this paper we propose a trust model, where besides considering trust and distrust, we also consider another parameter that measures the reliability on the stability of trust or distrust. The inclusion of this new parameter will allow us to use trust in a more accurate way. We consider trust is not static but dynamic and trust values can change along time. Thus, we will also take time into account, using it as a parameter of our model. There is very little work done about the inclusion of time as an influence on trust.
We will show the applicability of our model in the scenario of the process of reviewing papers for a conference. Sometimes for these kind of processes the Chair of the conference should first find the suitable reviewers. He can make this selection by using our model. Once the reviewers are selected they send out their reviews to the Chair who can also use our model in order to make the final decision about acceptance of papers.

Keywords: Trust, distrust, reliability, time.

1 Introduction

Since their origins trust management systems [4] have been used in order to assist entities that have to interact with others in a system. It has been a very important tool for the decision-making process. Sometimes, the information available about the other entities is not enough for establishing a secure exchange of information, but still the interaction must take place. Trust management systems try to supply this lack of information. In the last years, due to the growth of electronic communications and transactions, reputation systems [1] have been developed to aid trust management systems for assisting the trust decision process.

Interactions among entities in a system are not static but they might happen in different moments in time. Most of the existing trust management or reputation

[1] This work has been partially funded by the Spanish Ministry of Science and Education through the research project ARES (CONSOLIDER CSD2007-00004) and by the European Commission through the research project GREDIA (FP6 34363 - Grid enabled access to rich media content).
[2] Email: {agudo,mcgago,jlm}@lcc.uma.es

doi:10.1016/j.entcs.2009.07.034

systems do not consider or take into account how time influences the trust or reputation outcomes [7,11]. However, some authors have realised that time can influence trust. Thus, in [6] the authors mentioned that trust is a very dynamic phenomenon evolving in time and having a history. In [12] a dynamic trust model for mobile ad-hoc networks is introduced. Their proposal is used to add a measure of trust to the routing process. In this work, the authors highlight the importance of taking time into account, updating trust values as new evidences arrive. One of the problems of this approach is that it is really dependent on the scenario and, mainly because of this, they avoid discussing about transitiveness of trust. Another trust model that takes into account past trust history of users is [3]. Herrmann [8] also considers the influence of time on trust and proposes to use cTLA (compositional Temporal Logic of Actions [9]) as a method for modelling and verifying trust mechanisms.

One of the approaches that resembles more to ours is that of Mezzeti [13]. He proposes a trust model that takes time into account as one of the parameters to consider. He also gives more relevance to the freshness of the trust values since he considers obsolete information is not that accurate to describe recent behaviours. He proposes a formula in order to update trust degrees as time passes by.

We also believe this change in time should be reflected in the way a final trust value is obtained and that recent behaviours are more relevant for the final measure of trust. Thus, if the time unit is months, the weight given to an interaction happened a year ago should be smaller than the weight given to an interaction happened in the last month. We consider then a three-dimensional model based on trust values, reliability values on the behaviour of the participating entities and time. Thus, our model considers trust and reliability values, and the influence of time in order to derive a decision trust value.

Trust is a concept that can be related also to delegation [2] in the sense that when delegation takes places, an implicit trust relationship is established. According to this idea we will present an application scenario of our model where the Chair of a conference delegates the review process to the Program Committee members.

The paper is organized as follows. Section 2 introduces the trust model that we propose in order to use past trust history of users and the influence of time. Section 3 shows some application scenarios for our model and Section 4 concludes the paper and outlines the future work.

2 The Trust Model

2.1 The Concepts for Our Model

In this section we will present our model of trust. First, we will introduce some basic concepts.

We mean by *trust* the level of confidence that a user s (trustor) places on another user t (trustee) of a system referring to its honest behaviour. Analogously, we mean by *distrust* the level of confidence that a user will behave dishonestly. We mean by *reliability* the level of confidence that the trust or distrust levels will stay stable along the future.

Definition 2.1 A trust statement is a tuple

$$(Trustor, Trustee, TrustValue, ReliabilityValue, TimeStamp)$$

in $U \times U \times TD \times RD \times Time$, where U is the set of all users in the community; $TD = [-1, 1]$; $RD = [0, 1]$ and $Time$ is the domain of the time measurement.

Trust and reliability values could for example be obtained by means of a reputation system or observed information gathered previously. Thus, if we consider that *reputation* is what is said or believed about a person or things character or standing (Concise Oxford Dictionary) it is not strange that reliability values could be obtained this way.

If we think of the feedback system used in eBay and how it is used to build a reputation system [5,14] we can picture how the reliability and trust values are related. An easy example of how the eBay reputation system works and why reliability is not taken into account at all is the following.

Let us consider two eBay sellers, each of them with a 100% positive feedback. The difference between them is that the first one has only carried out 10 transactions whereas the second one has carried out 10.000 transactions. The trust component of the statement tuple made by a potential buyer will be the same for both sellers, but the reliability component will be higher for the second one, as there are more evidences that support the second seller.

Now let us suppose that the transactions of the second seller were all carried out five years ago, but the transactions of the first one were all carried out in the last month. The situation will change dramatically and the first seller will obtain then a higher reliability value.

The reliability value will be a function of many parameters, including freshness of the evidences used to compute the trust value and the quantity and/or quality of the evidences.

Concerning the values in TD, negative values will mean distrust. For simplicity we will omit the trustor and trustee in trust statements in the following when they are not relevant, or can be deduced by the context. Thus, trust statements will now be of the form (t, r, tm) where $t \in TD$, $r \in RD$ and $tm \in Time$.

Note that a tuple of the form $(0, r, tm)$ does not mean anything as a 0 value for TD means no information at all about the user.

If we fix a time instant, tuples of our model could be graphically viewed in the axes shown in Figure 1.

Regarding storage, our model could be considered distributed in the sense that statements are stored locally by each user. We will see later that trust statements can be exported and imported in such a way that at the end, all the computation is done in the user side taking into account all the imported information. Regarding trust scope, our system is clearly local in the sense that the trust and the reliability values are computed taking into account personal bias. A classification of trust models attending to the way trust statements are stored (centralized or distributed), and to the scope of trust statements (local or global) can be found in [15].

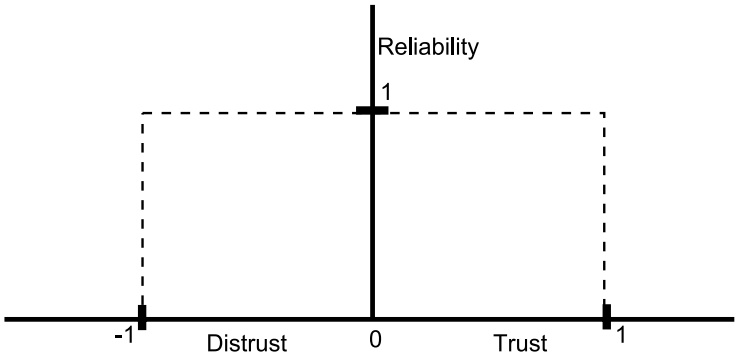

Fig. 1. Trust and reliability values

2.2 Trust Transitiveness

In our model, each user releases trust statements and uses them in order to make a trust decision about another user. Sometimes users may be interested in the statements that other users have already delivered. In these cases, it may be useful to export or import trust statements in such a way that re-using them could be possible.

Exporting a trust statement is straightforward and can be done by simply publishing it in a web page for example, together with some authentication token (a digital signature will work). The users importing such a trust statement can be assured of the precedence and authenticity of it. Of course, this information can be made available only to a group of people by encrypting the trust statements.

Importing a trust statement involves an adaptation mechanism, because of the local scope of trust statements. A requirement for importing a trust statement is having a trust statement about the trustor of it stored in the local database of trust statements. Statements are updated in such a way that the influence of all the trust and reliability values are reflected in the resulting trust statement. This resembles the way a trust network is built, such as in the case of PGP [16]. Thus, if an entity A produces a trust statement (t, r, tm) about entity B, and entity B produces a trust statement (t', r', tm') about C, then A could infer and store a new statement over C, (t'', r'', tm''), where t'', r'' and tm'' are obtained in the following way.

Definition 2.2 We define the transitivity operation for two trust statements with the same timestamp as

- $t'' = t \times t' = max(0, t) \cdot t'$.
- $r'' = min(r, r')$
- $tm'' = tm = tm'$.

In case we want to combine trust statements with different timestamps, we have to update all of them to the date of the newest one following the procedure introduced in Section 2.3

Let us note that the operation defined by the symbol $\times$ in Definition 2.2 does not correspond to the usual product define over $\mathbb{R}$. Let us also note that this product

is not commutative.

If we used the usual product over $\mathbb{R}$, multiplying two values of distrust (two negative real numbers) will give us as a result a positive real number. This in our model does not make sense as from two values of distrust is not possible to obtain a positive value of trust. Instead we set this value of trust to 0. The rationale behind this is that once there is a value of distrust we will not use it as referral for the statements to follow.

Also, from a value of distrust and a value of trust (a negative and a positive value respectively) we do not necessarily derive a value of distrust. In this case, the resulting trust value is 0 for the same reason as above. However, from a positive (trust) value and a negative one (distrust) we will obtain a value of distrust. We can summarize all of this as follows:

- $+ \times + = +$
- $+ \times - = -$
- $- \times + = 0$
- $- \times - = 0$

2.3 The Influence of Time

As we mentioned at the beginning of Section 2, the parameter of time will influence the reliability values. Thus, we consider that if an information was given to us a year ago it will be less reliable that a value that have just been collected.

In order to reflect this in our model we will define the following function.

Definition 2.3 A Time Influence function, f, is defined over the domain $Time$, with values in [0,1], as $f(x) = d^{-x}$, where d $(1 \leq d)$ is the time degradation parameter and x is a variable representing the time passed between the current state and the instant we want to measure. f verifies the following property

(i) $f(x)f(y) = f(x + y)$

Note that the greatest the time degradation parameter (d) is, the greatest the influence of time is, because

$$f(x + \Delta) = f(x)f(\Delta) = \frac{f(x)}{d^{\Delta}}$$

where Δ denotes the time increment. A trust statement can be updated by changing the timestamp of the trust statement to the current time instant, however this will affect the value of the reliability component. As we can see in Figure 2, when past trust statements are re-evaluated at the present time or at any future time, the reliability of the statements decreases.

Even though more complex approaches could have been taken, the simplest way to reflect reliability degradation caused by the past of time is by multiplying the initial reliability value by the result of the time influence function in order to obtain the current reliability value. Let us suppose a trust statement (t_0, r_0, tm_0) that we are interested in updating to time tm_1, where $tm_1 > tm_0$. Then, the updated trust

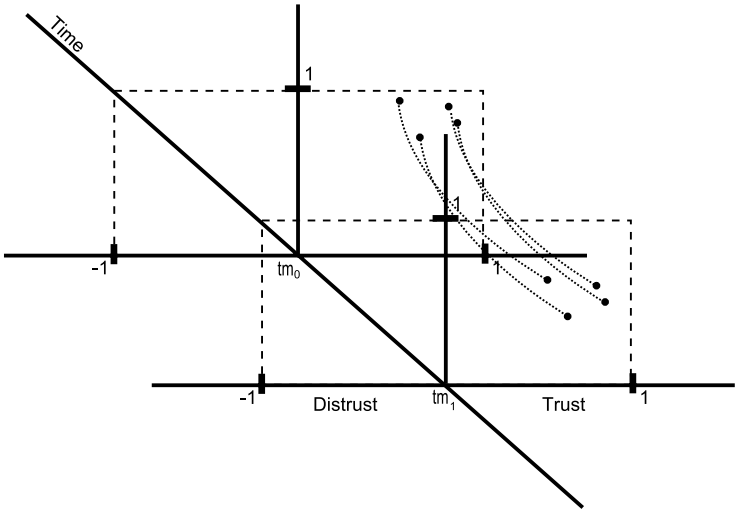

Fig. 2. Time Evolution

statement will be $(t_1 = t_0, r_1 = r_0 \cdot f(tm_1 - tm_0), tm_1)$.

Thanks to the properties of the time influence function, updating a trust statement from its initial time or from a previous updating will give us the same result. That is, if we want to update again the previous trust statement to a time tm_2, where $tm_2 > tm_1 > tm_0$, the updated trust statement will be,

$$(t_2 = t_1, r_2 = r_0 \cdot f(tm_2 - tm_0), tm_2)$$

if we consider as the base trust statement the initial one, but also the following updated trust statement

$$(t_2 = t_1, (r_1 \cdot f(tm_2 - tm_1), tm_2)$$

as the two results after updating are the same, once a trust statement is updated, the base statement can be safely removed.

2.4 Trust Consensus

In previous sections (see Section 2.1) we described how trust statements from other users could be imported. However, for some other cases there are several trust statements for the same trustee and we might be interested in reaching a consensus or deriving a global *trust decision* about them. When importing many trust statements from different users it is very likely that we end up with different, even contradictory, statements about the same trustees. Thus, reaching a consensus becomes very important. Next we will describe how we can reach that consensus.

Let us assume that a given user, the Trustor, owns several trust statements where the Trustee is the same. Thus, if we omit the static parameters of these statements, we obtain a set of tuples (t_i, r_i, tm_i) that encode the trust information of each statement.

The trust consensus is not an internal operator on the set of trust statements, like the transitivity operator defined in Section 2.2. It is instead a real number in

the interval $[-1, 1]$ that gives us an idea of whether the trustee is really to be trusted or not.

If we only had a trust statement, the trust consensus could be seen as a trust evaluation in such a way that given a trust statement it returns a real number to aid us making decisions at a given instant of time. The input parameters of the trust consensus function, or simply trust evaluation, are then a set of trust tuples and time instants.

Definition 2.4 Let G be the a finite set of n trust statements

$$\{(t_i, r_i, tm_i)\}_{i=1}^{n}$$

Then the *trust evaluation* of the set G at time $CurrentTime$ is defined as the value t_D obtained as:

$$t_D(G, CurrentTime) = \sum_{i=1}^{n} \frac{t_i r_i f(CurrentTime - tm_i)}{n} \tag{1}$$

We can also define the trust evaluation of a trustee, which could change if computed by different trustors, as the trust evaluation of the set of tuples corresponding to the trust statements that a trustor possesses referring the given trustee.

3 Application Scenario: The Reviewing Process for a Conference

Reviews Process

We consider a scenario for the reviewing papers process of a conference. In this scenario the Chair of the conference entrusts members of the program committee with the review of the papers and the recommendation whether they should or not be accepted for presentation at the conference. Usually, the Chair of the conference trusts the judgement of the reviewers assuming once they have become a program committee member of a conference they are reliable. In the following we will see how our model can help the Chair make his decision.

Choosing the reviewers

Let us assume the scenario above where the Chair has to first elaborate the list of the Program Committee (PC in the following) members. If the Chair has direct information about some reviewers for instance, because he knows them personally or has worked with them, he will build his own trust statements for these reviewers, (t_i, r_i, tm_i). The value assigned for t_i will be high as we are assuming in this first case the Chair knows the potential reviewers. r_i will depend on the topic of the paper and how familiar the PC member is with it.

It might happen that the Chair does not know enough experts and has to ask other PC members (or known and trusted person) for a recommendation. This scenario is depicted in Figure 3.

Then, the Chair can build his trust statement about the recommended PC member by the procedure presented in Definition 2.2. If the trust and reliability values

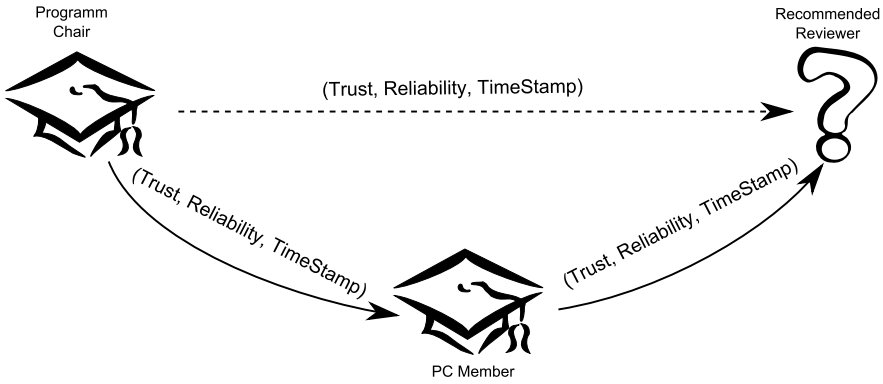

Fig. 3. Recommendation Scenario

are beyond a threshold that the Chair considers appropriate, the recommended person will be invited to become a PC member.

If finally the recommended person is invited to become a PC member, he or she will have to carry out the reviews of the assigned papers. Once the revision procedure for the conference has finished the Chair can issue a new trust statement about the new PC member. The reliability value can be influenced by factors such as the quality of the reviews and comments, whether his work was in accordance to the majority of the other reviewers, etc. This new statement happens in time t_1. Thus, if in the future the same Chair has to decide whether to interact or not with the same PC member, he can use all this information to compute a trust decision value by using Equation 1. Of course, besides these two moments the Chair can gathered information from other people who may have interacted with the PC member in order to compute this value.

Choosing the accepted papers

This stage is out of the scope of the trust model that we have proposed. However, our model can also help the process of accepting papers for the conference. We can take advantage of the formulae introduced in Definition 2.2 in order to compute an acceptance value for each reviewed paper. This acceptance value is computed as a function of the recommendation made by the reviewers about the papers and the trust statements that the Chair has issued about the PC member.

Note that both statements, the recommendation and the trust statement, happen at the same moment in time. For this reason we have omitted the time parameter as it will be the current moment Now. The two following cases apply:

- The Chair issues a trust statement about a reviewer. In the case this trust statement is from the past, the Chair has to update it by using the Time Influence function (See Definition 2.3). Then, the time parameter can be omitted from the tuple and the resulting new tuple is (t, r).
- Reviewers have also issued a recommendation about the papers. This recommendation consists of an acceptance value, for instance, in the set $[-1, 1]$, where -1 will mean strong rejection and 1 will mean strong acceptance. Together with

this value, the reviewer should also establish a confidence level about his level of expertise on the topic of the paper. This value could range in the set $[0, 1]$, where 1 denotes the maximum level of confidence. This will result in a tuple $(Evaluation, c)$, where c is the confidence of the reviewer on the topic of the paper.

In real conference management applications for the review process, these variables are considered but with discrete values ranging in different intervals. Depending on the application we should scale them in order to match our definition domain.

Despite the two tuples described above having mismatched semantic types it makes sense to combine them as they are in the same domain $[-1, 1] \times [0, 1]$, when considered as numerical values.

Then we can define the acceptance rate of the paper as

$$\sum_{i=1}^{n} \frac{(t_i \times e_i) r_i c_i}{n} \tag{2}$$

Where $\{(e_i, c_i\}_{i=1}^{n}$ is the set of evaluations issued by the reviewers and $\{(t_i, r_i\}_{i=1}^{n}$ is the set of updated trust statements over the reviewers. The operator $\times$, used in Equation 2, is the one defined in Definition 2.2.

When the number of papers with a positive acceptance rate is too high, only the papers with the best acceptance rate will be accepted.

4 Conclusions and Future Work

In this paper we present a trust model based on trust (distrust for negative values) and reliability. As a novelty, we have considered an extra parameter for our model. This parameter is *time*. In particular, we consider the time passed between the initial moment when there are transactions recorded and the current moment. This is precisely one of the features that distinguishes our model from others, for instance the trust models based on subjective logics introduced by Jøsang [10], where his parameter, certainty is semantically related to our reliability parameter. However, time degradation is not considered in the proposal developed by Jøsang.

As an example of application of our model, we propose the reviewing process of papers for a conference. We have shown how our model can be used for assisting the Chair of the conference in the process of choosing reviewers and how to use this information in order to select the accepted papers. We have shown that the influence of time can be used for refining the way the process of accepting papers for a conference takes place.

Our intention is to check the proposed model in real world applications for managing conferences. Also, we intend to apply this model to other scenarios where time is an issue or influences the development of the scenario.

References

[1] Abdul-Rahman, A. and S. Hailes, *Supporting Trust in Virtual Communities*, in: *Proceedings of the 33rd Hawaii International Conference on System Sciences*, 2000.

[2] Agudo, I., J. Lopez and J. A. Montenegro, "Delegation Service: a Step beyond Authorization," Idea Group, 2006 pp. 149–168.

[3] Almenarez, F., A. Marin, D. Dyaz and J. Sanchez, *Developing a Model for Trust Management in Pervasive Devices*, in: *PERCOMW '06: Proceedings of the 4th annual IEEE international conference on Pervasive Computing and Communications Workshops* (2006), p. 267.

[4] Blaze, M., J. Feigenbaum and J. Lacy, *Decentralized Trust Management*, in: *IEEE Symposium on Security and Privacy*, 1996.

[5] Cabral, L. and A. Hortacsu, *The Dynamics of Seller reputation: Theory and Evidence from eBay*, Working Paper 10363, National Bureau of Economic Research (2004).
URL http://www.nber.org/papers/w10363

[6] Falcone, R. and C. Castelfranchi, *The Socio-Cognitive Dynamics of Trust*, in: *4th Workshop on Agents-Trust in Cyber-Societies*, Lectures Notes in Computer Science **2246** (2001), pp. 55–72.

[7] Grandison, T. and M. Sloman, *A Survey of Trust in Internet Applications*, IEEE Communications Surveys. (2000).

[8] Herrmann, P., *Temporal Logic-Based Specification and Verification of Trust Models*, in: K. Stølen, W. H. Winsborough, F. Martinelli and F. Massacci, editors, *Trust Management, 4th International Conference, iTrust 2006*, Lecture Notes in Computer Science **3986**, Pisa, Italy, 2006, pp. 105–119.

[9] Herrmann, P. and H. Krumm, *A Framework for Modeling Transfer Protocols*, Computer Networks **34** (2000), pp. 317–337.

[10] Jøsang, A., *A Logic for uncertain Probabilities*, International Journal of Uncertainty Fuzziness and Knowledge-Based Systems. **9(3)** (2001), pp. 279–311.

[11] Jøsang, A., R. Hayward and S. Pope, *Trust Network Analysis with Subjective Logic*, in: *ACSC '06: Proceedings of the 29th Australasian Computer Science Conference* (2006), pp. 85–94.

[12] Liu, Z., A. W. Joy and R. A. Thompson, *A Dynamic Trust Model for Mobile Ad-Hoc Networks*, in: *FTDCS '04: Proceedings of the 10th IEEE International Workshop on Future Trends of Distributed Computing Systems* (2004), pp. 80–85.

[13] Mezzetti, N., *A Socially Inspired Reputation Model*, in: S. K. Katsikas et al., editor, *1st European Workshop on Public Key Infrastructure, EuroPKI 2004*, Lectures Notes in Computer Science **3093** (2004), pp. 191–204.

[14] Resnick, Paul, Zeckhauser, Richard, Swanson, John, Lockwood and Kate, *The Value of Reputation on eBay: A Controlled Experiment*, Experimental Economics **9** (2006), pp. 79–101.
URL http://dx.doi.org/10.1007/s10683-006-4309-2

[15] Ziegler, C. N. and G. Lausen, *Spreading Activation Models for Trust Propagation*, in: *IEEE International Conference on e-Technology, e-Commerce, and e-Service (EEE'04)*, Taipei, 2004.

[16] Zimmermann, P., "The Official PGP User's Guide," MIT Press, Boston, MA (1995).