Towards a Provenance-Preserving Trust Model in Agent Networks

Patricia Victor
Ghent University
Dept. of Applied Mathematics and CS
Krijgslaan 281 (S9), 9000 Gent, Belgium
Patricia.Victor@UGent.be

Martine De Cock
Ghent University
Dept. of Applied Mathematics and CS
Krijgslaan 281 (S9), 9000 Gent, Belgium
Martine.DeCock@UGent.be

Chris Cornelis
Ghent University
Dept. of Applied Mathematics and CS
Krijgslaan 281 (S9), 9000 Gent, Belgium
Chris.Cornelis@UGent.be

Paulo Pinheiro da Silva
The University of Texas at El Paso
Dept. of Computer Science
El Paso, TX 79968, USA
paulo@utep.edu

ABSTRACT

Social networks in which users or agents are connected to other agents and sources by trust relations are an important part of many web applications where information may come from multiple sources. Trust recommendations derived from these social networks are supposed to help agents develop their own opinions about how much they may trust other agents and sources. Despite the recent developments in the area, most of the trust models and metrics proposed so far tend to lose trust-related knowledge. We propose a new model in which trust values are derived from a bilattice that preserves valuable trust provenance information including partial trust, partial distrust, ignorance and inconsistency. We outline the problems that need to be addressed to construct a corresponding trust learning mechanism. We present initial results on the first learning step, namely trust propagation through trusted third parties (TTPs).

Categories and Subject Descriptors

H.3.3 [**Information Storage and Retrieval**]: Information Search and Retrieval—*Retrieval models*; I.2.4 [**Artificial Intelligence**]: Knowledge Representation Formalisms and Methods

General Terms

Algorithms, Human Factors

Keywords

Trust provenance, web of trust, distrust, bilattice, trust propagation

1. INTRODUCTION

As intelligent agents in the semantic web take over more and more human tasks, they require an automated way of trusting each other. One of the key problems in establishing this, is related to the dynamicity of trust: to grasp how trust emerges and vanishes. Once an understanding is reached, a new problem arises: how can the cyberinfrastructure be used to manage trust among users? To this aim, it is very important to find techniques that capture the human notions of trust as precisely as possible. Quoting [17]:

> If people can use their everyday trust building methods for the cyberinfrastructure and through it reach out to fellow human beings in far-away places, then that would be the dawn of the real Information Society for all.

In the near future, more and more applications and systems will need solid trust mechanisms. In fact, effective trust models already play an important role in many intelligent web applications, such as peer-to-peer (P2P) networks [13], recommender systems [14] and question answering systems [21]. All these applications use, in one way or another, a web of trust that allows agents to express trust in other agents. Using such a web of trust, an agent can develop an opinion about another, unknown agent.

Existing trust models can be classified in several ways, among which probabilistic vs. gradual approaches as well as representations of trust vs. representations of both trust and distrust. This classification is shown in Table 1, along with some representative references for each class.

Many models deal with trust in a binary way — an agent (or source) can either be trusted or not — and compute the probability or belief that the agent can be trusted [11, 12, 13, 21]. In such a setting, a higher trust score corresponds to a higher probability or belief that an agent can be trusted.

Apart from complete trust or no trust at all, however, in real life we also encounter partial trust. For instance, we of-

Table 1: Trust Models, State of the Art

	trust	**trust and distrust**
probabilistic	Kamvar et al. [13] Zaihrayeu et al. [21]	Jøsang et al. [11, 12]
gradual	Abdul-Rahman et al. [1] Almenárez et al. [2] Massa et al. [14]	De Cock et al. [6] Guha et al. [9]

WWW2006, May 22–26, 2006, Edinburgh, UK.
.

ten say "I trust this person very much", or "My trust in this person is rather low". More recent models like [1] take this into account: they make a distinction between "very trustworthy", "trustworthy", "untrustworthy" and "very untrustworthy". Other examples of a gradual approach can be found in [2, 7, 9, 14, 19]. In this case, a trust score is not a probability: a higher trust score corresponds to a higher trust. The ordering of the trust scores is very important, with "very reliable" representing a higher trust than "reliable", which in turn is higher than "rather unreliable". This approach leans itself better to the computation of trust scores when the outcome of an action can be positive to some extent, e.g., when provided information can be right or wrong to some degree, as opposed to being either right or wrong. It is this kind of application that we are keeping in mind throughout this paper.

Large agent networks without a central authority typically face ignorance as well as inconsistency problems. Indeed, it is likely that not all agents know each other, and different agents might provide contradictory information. Both ignorance and inconsistency can have an important impact on the trust score computation. Models that only take into account trust (e.g. [1, 13, 14, 16]), either with a probabilistic or a gradual interpretation, are not fully equipped to deal with trust issues in large networks where many agents do not know each other, because, as we explain in the next section, most of these models provide limited support for trust provenance.

Recent publications [10] show an emerging interest in modeling the notion of distrust, but models that take into account both trust and distrust are still scarce [6, 9, 12]. To the best of our knowledge, there is only one probabilistic approach considering trust and distrust simultaneously: in subjective logic (SL) [12] an opinion includes a belief b that an agent is to be trusted, a disbelief d corresponding to a belief that an agent is not to be trusted, and an uncertainty u. The uncertainty factor clearly indicates that there is room for ignorance in this model. However, the requirement that the belief b, the disbelief d and the uncertainty u should sum up to 1, rules out options for inconsistency although this might arise quite naturally in large networks with contradictory sources.

SL is an example of a probabilistic approach, whereas in this paper we will outline a trust model that uses a gradual approach, meaning that agents can be trusted to some degree. Furthermore, to preserve provenance information, our model deals with distrust in addition to trust. Consequently, we can represent partial trust and partial distrust. Our intended approach is situated in the bottom right corner of Table 1. As far as we know, besides our own earlier work [6], there is only one other existing model in this category: Guha et al. [9] use a couple (t, d) with a trust degree t and a distrust degree d, both in [0,1]. To obtain the final trust score, they subtract d from t. As we explain in the next section, potentially important information is lost when the trust and distrust scales are merged into one.

Our long term goal is to develop a model of trust that preserves trust provenance as much as possible. A previous model we introduced in [6], based on intuitionistic fuzzy set theory [4, 15], attempts this for partial trust, partial distrust and ignorance. In this paper, we will introduce an approach for preserving trust provenance about inconsistencies as well. Our model is based on a trust score space, consisting of the set $[0, 1]^2$ of trust scores equipped with a trust ordering, going from complete distrust to complete trust, as well as a knowledge ordering, going from a shortage of evidence (incomplete information) to an excess of evidence (in other words inconsistent information).

First of all, in Section 2, we point out the importance of a provenance-preserving trust model by means of some examples. In Section 3, we introduce the bilattice-based concept of a trust score space, i.e. a set of trust scores equipped with both a trust ordering and a knowledge ordering, and we provide a definition for a trust network. In developing a trust learning mechanism that is able to compute trust scores we will need to solve many challenging problems, such as how to propagate, aggregate, and update trust scores. In Section 4, we reflect upon our initial tinkering on candidate operators for trust score propagation through trusted third parties (TTPs). As these trust propagation operators are currently shaped according to our own intuitions, we will set up an experiment in the near future to gather the necessary data that provides insight in the propagation of trust scores through TTPs. We briefly comment on this in Section 5. Finally, subsequent problems that need to be addressed are sketched.

2. TRUST PROVENANCE

The main aim in using trust networks is to allow users or agents to form trust opinions on unknown agents or sources by asking for a trust recommendation from a TTP who, in turn, might consult its own TTP etc. This process is called trust propagation. In large networks, it often happens that an agent does not ask one TTP's opinion, but several. Combining trust information received from more than one TTP is called aggregation (see fig. 1). Existing trust network models usually apply suitable trust propagation and aggregation operators to compute a resulting trust value. In passing on this trust value to the inquiring agent, valuable information on how this value has been obtained is lost.

User opinions, however, may be affected by provenance information exposing how trust values have been computed. For example, a trust recommendation in a source from a fully informed TTP is quite different from a trust recommendation from a TTP who does not know the source too well but has no evidence to distrust it. Unfortunately, in current models, users cannot really exercise their right to interpret how trust is computed since most models do not preserve trust provenance.

Trust networks are typically challenged by two important problems influencing trust recommendations. Firstly, in large networks it is likely that many agents do not know each other, hence there is an abundance of ignorance. Secondly, because of the lack of a central authority, different agents might provide different and even contradictory information, hence inconsistency may occur. Below we illustrate how ignorance and inconsistency may affect trust recommendations.

EXAMPLE 1 (IGNORANCE). *Agent a needs to establish an opinion about agent c in order to complete an important bank transaction. Agent a may ask agent b for a recommendation of c because agent a does not know anything about c. Agent b, in this case, is a recommender that knows how to compute a trust value of c from a web of trust. Assume that b has evidence for both trusting and distrusting c. For in-*

stance, let us say that b trusts c 0.5 in the range [0,1] where 0 is full absence of trust and 1 is full presence of trust; and that b distrusts c 0.2 in the range [0,1] where 0 is full absence of distrust and 1 is full presence of distrust. Another way of saying this is that b trusts c at least to the extent 0.5, but also not more than 0.8. The length of the interval [0.5,0.8] indicates how much b lacks information about c.

In this scenario, by getting the trust value 0.5 from b, a is losing valuable information indicating that b has some evidence to distrust c too. A similar problem occurs using the approach of Guha et al. [9]. In this case, b will pass on a value of 0.5-0.2=0.3 to a. Again, a is losing valuable trust provenance information indicating, for example, how much b lacks information about c.

EXAMPLE 2 (IGNORANCE). *Agent a needs to establish an opinion about both agents c and d in order to find an efficient web service. To this end, agent a calls upon agent b for trust recommendations on agents c and d. Agent b completely distrusts agent c, hence agent b trusts agent c to degree* 0*. On the other hand agent b does not know agent d, hence agent b trusts agent d to degree* 0*. As a result, agent b returns the same trust recommendation to agent a for both agents c and d, namely 0, but the meaning of this value is clearly different in both cases. With agent c, the lack of trust is caused by a presence of distrust, while with agent d, the absence of trust is caused by a lack of knowledge. This provenance information is vital for agent a to make a well informed decision. For example, if agent a has a high trust in TTP b, agent a will not consider agent c anymore, but agent a might ask for other opinions on agent d.*

EXAMPLE 3 (CONTRADICTORY INFORMATION). *One of your friends tells you to trust a dentist, and another one of your friends tells you to distrust that same dentist. In this case, there are two TTPs, they are equally trusted, and they tell you the exact opposite thing. In other words, you have to deal with inconsistent information. What would be your aggregated trust score in the dentist? Models that work with only one scale can not represent this: taking e.g. 0.5 as trust score (i.e. the average) is not a solution, because then we can not differentiate from a situation in which both of your friends trust the dentist to the extent 0.5.*

Furthermore, what would you answer if someone asks you if the dentist can be trusted? A possible answer is: "I don't really know, because I have contradictory information about this dentist". Note that this is fundamentally different from "I don't know, because I have no information about him". In other words, a trust score of 0 is not a suitable option either, as it could imply both inconsistency and ignorance.

The examples above indicate the need for a model that preserves information on whether a "trust problem" is caused by presence of distrust or rather by lack of knowledge, as well as whether a "knowledge problem" is caused by having too little or rather too much, i.e. contradictory, information.

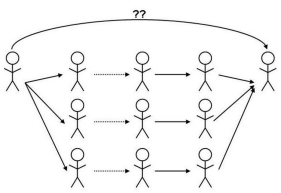

Figure 1: Trust propagation and aggregation

3. TRUST SCORE SPACE

We need a model that, on one hand, is able to represent the trust an agent may have in another agent in a given domain, and on the other hand, can evaluate the contribution of each aspect of trust to the overall trust score. As a result, such a model will be able to distinguish between different cases of trust provenance. To this end, we introduce a new structure, called trust score space $\mathcal{BL}^{\Box}$.

DEFINITION 1 (TRUST SCORE SPACE). *The trust score space*

$$\mathcal{BL}^{\Box} = ([0,1]^2, \leq_t, \leq_k, \neg)$$

consists of the set $[0,1]^2$ *of trust scores and two orderings defined by*

$$(x_1, x_2) \leq_t (y_1, y_2) \text{ iff } x_1 \leq y_1 \text{ and } x_2 \geq y_2$$

$$(x_1, x_2) \leq_k (y_1, y_2) \text{ iff } x_1 \leq y_1 \text{ and } x_2 \leq y_2$$

for all (x_1, x_2) *and* (y_1, y_2) *in* $[0,1]^2$*. Furthermore*

$$\neg(x_1, x_2) = (x_2, x_1).$$

The negation $\neg$ serves to impose a relationship between the lattices $([0,1]^2, \leq_t)$ and $([0,1]^2, \leq_k)$:

$$(x_1, x_2) \leq_t (y_1, y_2) \Rightarrow \neg(x_1, x_2) \geq_t \neg(y_1, y_2)$$

$$(x_1, x_2) \leq_k (y_1, y_2) \Rightarrow \neg(x_1, x_2) \leq_k \neg(y_1, y_2),$$

and $\neg\neg(x_1, x_2) = (x_1, x_2)$. In other words, $\neg$ is an involution that reverses the $\leq_t$-order and preserves the $\leq_k$-order. One can easily verify that the structure $\mathcal{BL}^{\Box}$ is a bilattice [3, 8].

Figure 2 shows the bilattice $\mathcal{BL}^{\Box}$, along with some examples of trust scores. The first lattice $([0,1]^2, \leq_t)$ orders the trust scores going from complete distrust $(0,1)$ to complete trust $(1,0)$. The other lattice $([0,1]^2, \leq_k)$ evaluates the amount of available trust evidence, going from a "shortage of evidence", $x_1 + x_2 < 1$ (incomplete information), to an "excess of evidence", namely $x_1 + x_2 > 1$ (inconsistent information). In the extreme cases, there is no information available $(0,0)$, or there is evidence that says that b is to be trusted fully as well as evidence that states that b is completely unreliable: $(1,1)$.

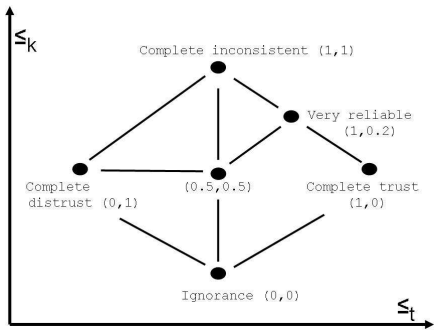

Figure 2: Trust score space $\mathcal{BL}^{\Box}$

The trust score space allows our model to preserve trust provenance by simultaneously representing partial trust, partial distrust, partial ignorance and partial inconsistency, and treating them as different, related concepts. Moreover, by using a bilattice model the aforementioned problems disappear:

1. By using trust scores we can now distinguish full distrust (0,1) from ignorance (0,0) and analogously, full trust (1,0) from inconsistency (1,1). This is an improvement of e.g. [1, 21].
2. We can deal with both incomplete information and inconsistency (improvement of [6]).
3. We do not lose important information (improvement of [9]), because, as will become clear in the next section, we keep the trust and distrust degree separated throughout the whole trust process (propagation and other operations).

The available trust information is modeled as a trust network that associates with each couple of agents a score drawn from the trust score space.

DEFINITION 2 (TRUST NETWORK). *A trust network is a couple (A, R) such that A is a set of agents and R is a $A \times A \to \mathcal{BL}^{\Box}$ mapping. For every a and b in A, we write*

$$R(a,b) = \big(R^+(a,b), R^-(a,b)\big)$$

- *$R(a,b)$ is called the trust score of a in b.*
- *$R^+(a,b)$ is called the trust degree of a in b.*
- *$R^-(a,b)$ is called the distrust degree of a in b.*

R should be thought of as a snapshot taken at a certain moment, since the trust learning mechanism involves recalculating trust scores, for instance through trust propagation as discussed next.

4. TRUST SCORE PROPAGATION

We often encounter situations in which we need trust information about an unknown person. For instance, if you are in search of a new dentist, you can ask your friends' opinion about dentist *Evans*. If they do not know *Evans* personally, they can ask a friend of theirs, and so on. In virtual trust networks, propagation operators are used to handle this problem. The simplest case (atomic propagation) can informally be described as (fig. 3): if the trust score of agent a in agent b is p, and the trust score of b in agent c is q, what information can be derived about the trust score of a in c? When propagating only trust, the most commonly used operator is multiplication. When taking into account also distrust, the picture gets more complicated, as the following example illustrates.

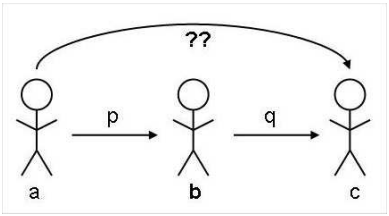

Figure 3: Atomic propagation

EXAMPLE 4. *Suppose agent a trusts agent b and agent b distrusts agent c. It is reasonable to assume that based on this, agent a will also distrust agent c, i.e. $R(a,c) = (0,1)$. Now, switch the couples. If a distrusts b and b trusts c, there are several options for the trust score of a in c: a possible reaction for a is to do the exact opposite of what b recommends, in other words to distrust c, $R(a,c) = (0,1)$. But another interpretation is to ignore everything b says, hence the result of the propagation is ignorance, $R(a,c) = (0,0)$.*

As this example indicates, there are likely multiple possible propagation operators for trust scores. We expect that the choice for a particular $\mathcal{BL}^{\Box} \times \mathcal{BL}^{\Box} \to \mathcal{BL}^{\Box}$ mapping to model the trust score propagation will depend on the application and the context but might also differ from person to person. Thus, the need for provenance-preserving trust models becomes more evident.

To study some possible propagation schemes, let us first consider the bivalent case, i.e. when trust and distrust degrees assume only the values 0 or 1. For agents a and b, we use $R^+(a,b)$, $R^-(a,b)$, and $\sim R^-(a,b)$ as shorthands for respectively $R^+(a,b) = 1$, $R^-(a,b) = 1$ and $R^-(a,b) = 0$. We consider the following three, different propagation schemes (a, b and c are agents):

1. $R^+(a,c) \equiv R^+(a,b) \wedge R^+(b,c)$
   $R^-(a,c) \equiv R^+(a,b) \wedge R^-(b,c)$
2. $R^+(a,c) \equiv R^+(a,b) \wedge R^+(b,c)$
   $R^-(a,c) \equiv \sim R^-(a,b) \wedge R^-(b,c)$
3. $R^+(a,c) \equiv (R^+(a,b) \wedge R^+(b,c)) \vee (R^-(a,b) \wedge R^-(b,c))$
   $R^-(a,c) \equiv (R^+(a,b) \wedge R^-(b,c)) \vee (R^-(a,b) \wedge R^+(b,c))$

In scheme (1) agent a only listens to whom he trusts, and ignores everyone else. Scheme (2) is similar but in addition agent a takes over distrust information from a not distrusted (hence possibly unknown) third party. Scheme (3) corresponds to an interpretation in which the enemy of an enemy is considered to be a friend, and the friend of an enemy is considered to be an enemy.

In our model, besides 0 and 1, we also allow partial trust and distrust. Hence we need suitable extensions of the logical operators that are used in (1), (2) and (3). For conjunction, disjunction and negation, we use respectively a t-norm T, a t-conorm S and a negator N. They represent large classes of logic connectives, from which specific operators, each with their own behaviour, can be chosen, according to the application or context.

T and S are increasing, commutative and associative $[0,1] \times [0,1] \to [0,1]$ mappings satisfying $T(x,1) = S(x,0) = x$ for all x in $[0,1]$. Examples of T are the minimum and the product, while S could be the maximum or the mapping S_P defined by $S_P(x,y) = x + y - x \cdot y$, for all x and y in $[0,1]$. N is a decreasing $[0,1] \to [0,1]$ mapping satisfying $N(0) = 1$ and $N(1) = 0$; the most commonly used one is $N_s(x) = 1 - x$.

Generalizing the logical operators in scheme (1), (2), and (3) accordingly, we obtain the propagation operators of Table 2. Each one can be used for modeling a specific behaviour. Starting from a trust score (t_1, d_1) of agent a in agent

Table 2: Propagation operators, using TTP b with $R(a,b) = (t_1, d_1)$ and $R(b,c) = (t_2, d_2)$

Notation	Trust score of a in c	Meaning
$\texttt{Prop}_1$	$(T(t_1,t_2), T(t_1,d_2))$	Skeptical, take no advice from enemies or unknown people.
$\texttt{Prop}_2$	$(T(t_1,t_2), T(N(d_1),d_2))$	Paranoid, distrust even unknown people's enemies.
$\texttt{Prop}_3$	$(S(T(t_1,t_2),T(d_1,d_2)), S(T(t_1,d_2),T(d_1,t_2)))$	Friend of your enemy is your enemy too.

b, and a trust score (t_2, d_2) of agent b in agent c, each propagation operator computes a trust score for agent a in agent c. Since the resulting value is again an element of the trust score space, trust provenance is preserved.

The remainder of this section is devoted to the investigation of some potentially useful properties of these propagation operators. In doing so, we keep the logical operators as generic as possible, in order to get a clear view on their general behaviour. First of all, if one of the arguments of a propagation operator can be replaced by a higher trust score w.r.t. to the knowledge ordering without decreasing the resulting trust score, we call the propagation operator knowledge monotonic.

DEFINITION 3 (KNOWLEDGE MONOTONICITY). *A propagation operator f on $\mathcal{BL}^{\Box}$ is said to be knowledge monotonic iff for all x, y, z, and u in $\mathcal{BL}^{\Box}$,*

$$x \leq_k y \text{ and } z \leq_k u \text{ implies } f(x,z) \leq_k f(y,u)$$

Knowledge monotonicity reflects that the better you know how well you should trust or distrust user b who is recommending user c, the better you know how well to trust or distrust user c. Although this behaviour seems natural, not all operators of Table 2 abide by it.

PROPOSITION 1. $\texttt{Prop}_1$ *and* $\texttt{Prop}_3$ *are knowledge monotonic.* $\texttt{Prop}_2$ *is not knowledge monotonic.*

Proof. The knowledge monotonicity of $\texttt{Prop}_1$ and $\texttt{Prop}_3$ follows from the monotonicity of T and S. To see that $\texttt{Prop}_2$ is not knowledge monotonic, consider

$$\texttt{Prop}_2((0.2,0.7),(0,1)) = (0,0.3)$$
$$\texttt{Prop}_2((0.2,0.8),(0,1)) = (0,0.2),$$

with N_s as negator. We have that $(0.2,0.7) \leq_k (0.2,0.8)$ and $(0,1) \leq_k (0,1)$ but $(0,0.3) \not\leq_k (0,0.2)$.

The intuitive explanation behind the non knowledge monotonic behaviour of $\texttt{Prop}_2$ is that, using this propagation operator, agent a takes over distrust from a stranger b, hence giving b the benefit of the doubt, but when a starts to distrust b (thus knowing b better), a will adopt b's opinion to a lesser extent (in other words: a derives less knowledge).

Knowledge montonicity is not only useful to provide more insight in the propagation operators but it can also be used to establish a lower or upper bound for the actual propagated trust score without immediate recalculation. This might be useful in a situtation where one of the agents has updated its trust score in another agent and there is not enough time to recalculate the whole propagation chain.

Besides atomic propagation, we need to be able to consider longer propagation chains, so TTPs can in turn consult their own TTPs and so on. $\texttt{Prop}_1$ turns out to be associative, which means that we can extend it for more scores without ambiguity.

PROPOSITION 2. *(Associativity:* $\texttt{Prop}_1$ *is associative, i.e. for all x, y, and z in $\mathcal{BL}^{\Box}$ it holds that:*

$$\texttt{Prop}_1(\texttt{Prop}_1(x,y),z) = \texttt{Prop}_1(x,\texttt{Prop}_1(y,z))$$

$\texttt{Prop}_2$ *and* $\texttt{Prop}_3$ *are not associative.*

Proof. The associativity of $\texttt{Prop}_1$ can be proved by taking into account the associativity of the t-norm. Examples can be constructed to show that the other two propagation operators are not associative. Take for example $N(x) = 1 - x$ and $T(x,y) = x \cdot y$, then

$$\texttt{Prop}_2((0.3,0.6),\texttt{Prop}_2((0.1,0.2),(0.8,0.1))) = (0.024,0.032)$$

while on the other hand

$$\texttt{Prop}_2(\texttt{Prop}_2((0.3,0.6),(0.1,0.2)),(0.8,0.1)) = (0.024,0.092)$$

With an associative propagation operator, the overall trust score computed from a longer propagation chain is independent of the choice of which two subsequent trust scores to combine first. When dealing with a non associative operator however, it should be specified which pieces of the propagation chain to calculate first.

Finally, it is interesting to note that in some cases the overall trust score in a longer propagation chain can be determined by looking at only one agent. For instance, if we use $\texttt{Prop}_1$ or $\texttt{Prop}_3$, and there occurs a missing link $(0,0)$ anywhere in the propagation chain, the result will contain no useful information (in other words, the final trust score is $(0,0)$). Hence as soon as one of the agents is ignorant, we can dismiss the entire chain. Notice that this also holds for $\texttt{Prop}_3$, despite the fact that it is not an associative operator. Using $\texttt{Prop}_1$, the same conclusion $(0,0)$ can be drawn if at any position in the chain, except the last one, there occurs complete distrust $(0,1)$.

5. CONCLUSIONS AND FUTURE WORK

We have introduced a new model that can simultaneously handle partial trust and distrust. We showed that our bilattice-based model alleviates some of the existing problems of trust models, more specifically concerning trust provenance. In addition, this new model can handle incomplete and excessive information, which occurs frequently in virtual communities, such as the WWW in general and trust networks in particular. Therefore, this new provenance-preserving trust model can lead to an improvement of many existing web applications, such as P2P networks, question answering systems and recommender systems.

A first step in our future research involves the further development and the choice of trust score propagation operators. Of course, the trust behaviour of users depends on the situation and the application, and is in most cases relative to

a goal or a task. A friend e.g. can be trusted for answering questions about movies, but not necessarily about doctors. Therefore, we are preparing some specific scenario's in which trust is needed to make a certain decision (e.g. which doctor to visit, which movie to see). According to these scenario's, we will prepare questionnaires, in which we aim to determine how propagation of trust scores takes place. Gathering such data, we hope to get a clear view on trust score propagation in real life, and how to model it in applications. We do not expect to find one particular propagation schema, but rather several, depending on a persons nature. When we obtain the results of the questionnaire, we will also be able to verify the three propagation operators we proposed in this paper. Furthermore, we would like to investigate the behaviour of the operators when using particular t-norms, t-conorms and negators, and examine whether it is possible to use other classes of operators that do not use t-(co)norms.

A second problem which needs to be addressed, is aggregation. In our domain of interest, namely a gradual approach to both trust and distrust, there are no aggregation operators yet. We will start by investigating whether it is possible to extend existing aggregation operators, like e.g. the ordered weighted averaging aggregation operator [20], fuzzy integrals [5, 18], etc., but we assume that not all the problems will be solved in this way, and that we will also need to introduce new specific aggregation operators.

Finally, trust and distrust are not static, they can change after a bad (or good) experience. Therefore, it is also necessary to search for appropriate updating techniques.

Our final goal is the creation of a framework that can represent partial trust, distrust, inconsistency and ignorance, that contains appropriate operators (propagation, aggregation, update) to work with those trust scores, and that can serve as a starting point to improve the quality of many web applications. In particular, as we are aware that trust is experienced in different ways, according to the application and context, we aim at a further development of our model for one specific application.

6. ACKNOWLEDGMENTS

Patricia Victor would like to thank the Institute for the Promotion of Innovation through Science and Technology in Flanders (IWT-Vlaanderen) for funding her research. Chris Cornelis would like to thank the Research Foundation-Flanders for funding his research.

7. REFERENCES

[1] A. Abdul-Rahman and S. Hailes. Supporting trust in virtual communities. In *Proceedings of the 33rd Hawaii International Conference on System Sciences*, pages 1769–1777, 2000.

[2] F. Almenárez, A. Marín, C. Campo, and C. García. Ptm: A pervasive trust management model for dynamic open environments. In *First Workshop on Pervasive Security, Privacy and Trust, PSPT2004 in conjuntion with Mobiquitous 2004*, 2004.

[3] O. Arieli, C. Cornelis, G. Deschrijver, and E. E. Kerre. Bilattice-based squares and triangles. *Lecture Notes in Computer Science*, 3571:563–574, 2005.

[4] K. Atanassov. Intuitionistic fuzzy sets. *Fuzzy Sets and Systems*, 20:87–96, 1986.

[5] G. Choquet. Theory of capacities. *Annales de l'Institut Fourier*, 5:131–295, 1953.

[6] M. De Cock and P. Pinheiro da Silva. A many-valued representation and propagation of trust and distrust. *Lecture Notes in Computer Science*, 3849:108–113, 2006.

[7] R. Falcone, G. Pezzulo, and C. Castelfranchi. A fuzzy approach to a belief-based trust computation. *Lecture Notes in Artificial Intelligence*, 2631:73–86, 2003.

[8] M. Ginsberg. Multi-valued logics: A uniform approach to reasoning in artificial intelligence. *Computer Intelligence*, 4:256–316, 1988.

[9] R. Guha, R. Kumar, P. Raghavan, and A. Tomkins. Propagation of trust and distrust. In *Proceedings of the 13th International World Wide Web Conference*, pages 403–412, 2004.

[10] P. Herrmann, V. Issarny, and S. Shiu (eds). *Lecture Notes in Computer Science*, volume 3477. 2005.

[11] A. Jøsang. A logic for uncertain probabilities. *International Journal of Uncertainty, Fuzziness and Knowledge-based Systems*, 9(3):279–311, 2001.

[12] A. Jøsang and S. Knapskog. A metric for trusted systems. In *Proc. 21st NIST-NCSC National Information Systems Security Conference*, pages 16–29, 1998.

[13] S. Kamvar, M. Schlosser, and H. Garcia-Molina. The eigentrust algorithm for reputation management in P2P networks. In *Proceedings of the 12th International World Wide Web Conference*, pages 640–651, 2003.

[14] P. Massa and P. Avesani. Trust-aware collaborative filtering for recommender systems. In *Proceedings of the Federated International Conference On The Move to Meaningful Internet: CoopIS, DOA, ODBASE*, pages 492–508, 2004.

[15] M. Nikolova, N. Nikolov, C. Cornelis, and G. Deschrijver. Survey of the research on intuitionistic fuzzy sets. *Advanced Studies in Contempory Mathematics*, 4(2):127–157, 2002.

[16] M. Richardson, R. Agrawal, and P. Domingos. Trust management for the semantic web. In *Proceedings of the Second International Semantic Web Conference*, pages 351–368, 2003.

[17] M. Riguidel and F. Martinelli (eds). Security, Dependability and Trust. *Thematic Group Report of the European Coordination Action Beyond the Horizon: Anticipating Future and Emerging Information Society Technologies*, http://www.beyond-the-horizon.net, 2006.

[18] M. Sugeno. *Theory of fuzzy integrals and its applications, PhD thesis*. 1974.

[19] W. Tang, Y. Ma, and Z. Chen. Managing trust in peer-to-peer networks. *Journal of Digital Information Management*, 3:58–63, 2005.

[20] R. Yager. On ordered weighted averaging aggregation operators in multicriteria decision making. *IEEE Transactions on Systems, Man, and Cybernetics*, 18:183–190, 1988.

[21] I. Zaihrayeu, P. Pinheiro da Silva, and D. McGuinness. IWTrust: Improving user trust in answers from the web. In *Proceedings of the Third International Conference On Trust Management*, pages 384–392, 2005.