A Simple Logical Approach to Reasoning with and about Trust

Simon Parsons and **Elizabeth Sklar**
Department of Computer & Information Science,
Brooklyn College,
City University of New York,
2900 Bedford Avenue,
Brooklyn, NY 11210 USA
{sklar,parsons}@sci.brooklyn.cuny.edu

Peter McBurney
Department of Computer Science,
University of Liverpool,
Ashton Building,Ashton Street,
Liverpool, L69 3BX,
United Kingdom
mcburney@liverpool.ac.uk

Abstract

Trust is an approach to managing the uncertainty about autonomous entities and the information they store, and so can play an important role in any decentralized system. As a result, trust has been widely studied in multiagent systems and related fields such as the semantic web. Here we introduce a simple approach to reasoning about trust with logic.

Introduction

Trust is an approach to managing the uncertainty about autonomous entities and the information they deal with. As a result, trust can play an important role in any decentralized system. As computer systems have become increasingly distributed, and control in those systems has become more decentralized, trust has become steadily more important within Computer Science (Artz & Gil 2007; Grandison & Sloman 2000). Given the role that provenance plays in trust (Geerts, Kementsiedtsidis, & Milano 2006; Golbeck 2006), we suggested (Parsons, McBurney, & Sklar 2010) that *argumentation* — which tracks the origin of data used in reasoning — might prove useful in reasoning about trust. Here we start to back up that claim, discussing how the usual approach to dealing with trust information can be captured in logic and how it can be integrated with argumentation-based reasoning about beliefs.

Trust

We are interested in a finite set of agents Ags and how these agents trust one another. Following the usual presentation (for example (Wang & Singh 2006)), we start with a *trust relation*:

$$\tau \subseteq Ags \times Ags$$

which identifies which agents trust one another. If $\tau(Ag_i, Ag_j)$, where $Ag_i, Ag_j \in Ags$, then Ag_i trusts Ag_j. This is not a symmetric relation, so it is not necessarily the case that $\tau(Ag_i, Ag_j) \Rightarrow \tau(Ag_j, Ag_i)$.

It is natural to represent this trust relation as a directed graph, and we define a *trust network* to be a graph comprising, respectively, a set of nodes and a set of edges:

$$\mathcal{T} = \langle Ags, \{\tau\}\rangle$$

where Ags is a set of agents and $\{\tau\}$ is the set of pairwise trust relations over Ags so that if $\tau(Ag_i, Ag_j)$ is in $\{\tau\}$ then $\{Ag_i, Ag_j\}$ is a directed arc from Ag_i to Ag_j in $\mathcal{T}$ indicating that Ag_i trusts Ag_j.

In this graph, the set of agents is the set of vertices, and the trust relations define the arcs. We are typically interested in *minimal* trust networks, which are connected — these thus capture the relationship between a set of agents all of whom, in one way or another are connected by a "web of trust". A directed path between agents in the trust network implies that one agent indirectly trusts another. For example if:

$$\langle Ag_1, Ag_2, \ldots Ag_n\rangle$$

is a path from agent Ag_1 to Ag_n, then we have:

$$\tau(Ag_1, Ag_2), \tau(Ag_2, Ag_3), \ldots, \tau(Ag_{n-1}, Ag_n)$$

and the path gives us a means to compute the trust that Ag_1 has in Ag_n. Below we will make use of the function $length(\cdot)$ which returns the number of agents in a path: $length(\langle Ag_1, Ag_2, \ldots Ag_n\rangle)$ is n.

The usual assumption in the literature is that we can place some measure on the trust that one agent has in another, so we have:

$$tr : Ags \times Ags \mapsto \Re$$

where tr gives a suitable trust value. In this paper, we take this value to be between 0, indicating no trust, and 1, indicating the greatest possible degree of trust. We assume that tr and τ are mutually consistent, so that:

$$\begin{aligned} tr(Ag_i, Ag_j) \neq 0 \quad &\Leftrightarrow \quad (Ag_i, Ag_j) \in \tau \\ tr(Ag_i, Ag_j) = 0 \quad &\Leftrightarrow \quad (Ag_i, Ag_j) \notin \tau \end{aligned}$$

Now, this just deals with the direct trust relations encoded in τ. It is usual in work on trust to consider performing inference about trust by assuming that trust relations are transitive. This is easily captured in the notion of a trust network.

The notion of trust embodied here is exactly Jøsang's "indirect trust" or "derived trust" (Jøsang, Keser, & Dimitrakos 2005) and the process of inference is what (Guha *et al.* 2004) calls "direct propagation". If we have a function tr, then we can compute:

$$\begin{aligned} &tr(Ag_i, Ag_j) = \\ &\quad tr(Ag_i, Ag_{i+1}) \otimes^{tr} \\ &\quad tr(Ag_{i+1}, Ag_{i+2}) \otimes^{tr} \ldots \otimes^{tr} tr(Ag_{j-1}, Ag_j) \quad (1) \end{aligned}$$

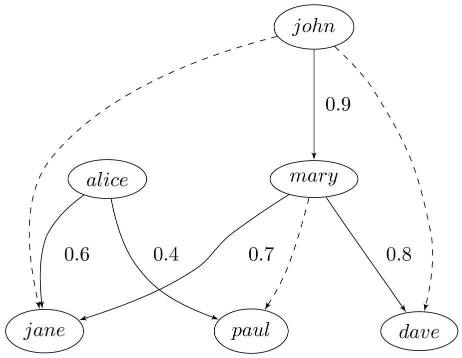

Figure 1: An example trust graph. The solid lines represent trust relations, and the dashed lines represent derived trust. The link between $john$ and $jane$ and the link between $john$ and $dave$ are the result of direct propagation, the link between $mary$ and $paul$ is the result of co-citation.

for some operation $\otimes^{tr}$. Here we follow (Wang & Singh 2006) in using the symbol $\otimes$, to stand for this generic operation. The superscript distinguishes this from a similar operation $\otimes^{bel}$ on belief values which we will meet below. Sometimes it is the case that there are two or more paths through the trust network between Ag_i and Ag_j indicating that Ag_i has several opinions about the trustworthiness of Ag_j. If these two paths are

$$\langle Ag_i, Ag'_{i+1}, \dots Ag_j \rangle \quad \text{and} \quad \langle Ag_i, Ag''_{i+1}, \dots Ag_j \rangle$$

and

$$\begin{aligned} tr(Ag_i, Ag_j)' &= \\ &tr(Ag_i, Ag'_{i+1}) \otimes^{tr} \dots \otimes^{tr} tr(Ag'_{j-1}, Ag_j) \\ tr(Ag_i, Ag_j)'' &= \\ &tr(Ag_i, Ag''_{i+1}) \otimes^{tr} \dots \otimes^{tr} tr(Ag''_{j-1}, Ag_j) \end{aligned}$$

then the overall degree of trust that Ag_i has in Ag_j is:

$$tr(Ag_i, Ag_j) = tr(Ag_i, Ag_j)' \oplus^{tr} tr(Ag_i, Ag_j)'' \tag{2}$$

Again we use the standard notation $\oplus$ for a function that combines trust measures along two paths (Wang & Singh 2006). Clearly we can extend this to handle the combination of more than two paths. We include this description of combining trust paths for completeness, and will not discuss it further in this paper, though we will consider it in future work.

As an example of a trust graph, consider Figure 1 which shows the trust relationship between $john$, $mary$, $alice$, $jane$, $paul$ and $dave$. This is adapted from the example in (Katz & Golbeck 2006) by normalizing the values to lie between 0 and 1 and adding $paul$. The solid lines are direct trust relationships and the dotted lines are indirect links derived from the direct links. Thus, for example, $john$ trusts $jane$ and $dave$ because he trusts $mary$ and $mary$ trusts $jane$ and $dave$.

Reasoning about trust

The standard approach in the literature on trust is to base the computation on the the trust graph (see (Wang & Singh 2006)). Here we take a different tack, describing a simple scheme for encoding this kind of computation in logic.

We will assume that every agent i has some collection of information about the world, which we will call Δ_i, that is expressed in logic. Δ_i is made up of a number of partitions, one of which, Δ_i^{tr}, holds information about the degree of trust i has in other agents it knows. For example, the agent $john$ from the above example might have the following collection of information:

$$\Delta^{tr}_{john} \quad \begin{array}{l} (t1 : trusts(john, mary) : 0.9) \\ (t2 : trusts(mary, jane) : 0.7) \\ (t3 : trusts(mary, dave) : 0.8) \\ (t4 : trusts(alice, jane) : 0.6) \\ (t5 : trusts(alice, paul) : 0.4) \end{array}$$

Each element of Δ^{tr}_{john} has the form:

$$(\langle index \rangle : \langle data \rangle : \langle value \rangle)$$

The first is a means of referring to the element, the second is a formula, the third is the degree of trust between the individuals mentioned in the $trust$ relation.

From Δ^{tr}_{john} we can then construct arguments mirroring the trust propagation discussed above. For example, using the first two rules from Figure 2 we can construct the argument:

$$\begin{aligned} &\Delta^{tr}_{john} \vdash_{tr} \\ &\qquad (trusts(john, jane) : \{t1, t2\} : \{Ax, Ax, dp\} : \tilde{t}) \end{aligned}$$

All arguments in our approach take the form:

$$(\langle conclusion \rangle : \langle grounds \rangle : \langle rules \rangle : \langle value \rangle)$$

where $conclusion$ is inferred from the $grounds$ using the rules of inference $rules$ and with degree $value$. In this case the argument says $john$ trusts $jane$ with degree $\tilde{t}$ (which is just $0.9 \otimes^{tr} 0.7$), through two applications of the rule Ax and one application of the rule dp to the two facts indexed by $t1$ and $t2$[1].

The rule Ax says that if some agent i has a triple:

$$t1 : trusts(john, mary) : 0.9$$

in its Δ_i^{tr} then it can construct an argument for $trusts(john, mary)$ where the grounds are $t1$, the degree of trust is 0.9, and which records that the Ax rule was used in its derivation.

The rule dp captures direct propagation of trust values. It says that if we can show that $trusts(x, y)$ holds with degree $\tilde{d}$ and we can show that $trusts(y, z)$ holds with degree $\tilde{e}$, then we are allowed to conclude $trusts(x, z)$ with a degree $\tilde{d} \otimes^{tr} \tilde{e}$, and that the conclusion is based on the union of the information that supported the premises, and is computed using all the rules used by both the premises.

[1]There are good reasons for using the formulae themselves in the grounds and factoring the whole proof into the set of rules, but here, for clarity, we use the relevant indices.

$$\text{Ax}\frac{(n : trusts(x, y) : \tilde{d}) \in \Delta_i^{tr}}{\Delta_i^{tr} \vdash_{tr} (trusts(x, y) : \{n\} : \{Ax\} : \tilde{d})}$$

$$dp\frac{\Delta_i^{tr} \vdash_{tr} (trusts(x, y) : G : R : \tilde{d}) \text{ and } \Delta_i^{tr} \vdash_{tr} (trusts(y, z) : H : S : \tilde{e})}{\Delta_i^{tr} \vdash_{tr} (trusts(x, z) : G \cup H : R \cup S \cup \{dp\} : \tilde{d} \otimes^{tr} \tilde{e})}$$

$$cc\frac{\Delta_i^{tr} \vdash_{tr} (trusts(x, y) : G : R : \tilde{d}) \text{ and } \Delta_i^{tr} \vdash_{tr} (trusts(x, z) : H : S : \tilde{e}) \text{ and } \Delta_i^{tr} \vdash_{tr} (trusts(w, z) : K : T : \tilde{f})}{\Delta_i^{tr} \vdash_{tr} (trusts(w, y) : G \cup H \cup K : R \cup S \cup T \cup \{cc\} : \tilde{d} \otimes^{tr} \tilde{e} \otimes^{tr} \tilde{f})}$$

Figure 2: Part of the tr consequence relation

Why is this interesting? After all, it does no more than trace paths through the trust graph.

Well, one of the strengths of argumentation, and the reason we are interested in using argumentation to handle trust, is that we want to record, in the form of the argument for some proposition, the *reasons* that it should be believed. Since information on the source of some piece of data, and the trust that an agent has in the source, is relevant, then it should be recorded in the argument, and this is easier to achieve if we encode data about who trusts whom in logic.

One of the nice things that this approach allows us to do is to track the application of the rules for propagating trust. When we just use direct propagation, this is not terribly interesting (though it does allow us to distinguish between the bits of information used in the formation of arguments, which may be a criterion for preferring one over another (Loui 1987)), but it becomes more obviously useful when we start to allow other rules for propagating trust. For example, (Guha *et al.* 2004) suggest a rule they call *co-citation*, which they describe as:

> For example, suppose i_1 trusts j_1 and j_2 and i_2 trusts j_2. Under co-citation, we would conclude that i_2 should also trust j_1.

In our example (see Figure 1), therefore, co-citation suggests that since $alice$ trusts $jane$ and $paul$, and $mary$ trusts $jane$, then $mary$ should trust $paul$. (Guha *et al.* 2004) also tells us how trust values should be combined in this case — $mary$'s trust in $paul$ is just the combination of trust values along the path from $mary$ to $jane$ to $alice$ to $paul$.

This form of reasoning is captured by the rule cc in Figure 2, and the rule also takes care of the necessary book keeping of grounds, proof rules and trust values. Combining the application of cc with dp as before allows the construction of the argument that $john$ trusts $paul$:

$$\Delta_{john}^{tr} \vdash_{tr} (trusts(john, paul) : \{t1, t2, t4, t5\} : rules : \tilde{r})$$

where $rules$ is:

$$\{Ax, Ax, Ax, Ax, dp, cc\}$$

and $\tilde{r}$ is $0.9 \otimes^{tr} 0.7 \otimes^{tr} 0.6 \otimes^{tr} 0.4$.

Now, when we have several rules for propagating trust, keeping track of which rule has been used in which derivation is appealing, especially since one might want to distinguish between arguments that use different rules of inference. For example, one might prefer arguments, no matter the trust value, which only make use of direct propagation over those that make use of co-citation.

Reasoning with trust

What we have presented so far explains how agent Ag_i can reason about the trustworthiness of its acquaintances. The reason for doing this is so i can use this information to decide how to use information that it gets from those acquaintances. To formalize the way in which i does this, we will assume that, in addition to Δ_i^{tr} Ag_i has a set of beliefs about the world Δ_i^{bel} (which we assume come with some measure of belief), and some information Δ_i^j provided by each of its acquaintances Ag_j, and that:

$$\Delta_i = \Delta_i^{tr} \cup \Delta_i^{bel} \cup \bigcup_j \Delta_i^j$$

All of this information can then be used, along with the consequence relation from Figure 3 to construct arguments that combine trust and beliefs.

The proof rules in Figure 3 are based on those we introduced in (McBurney & Parsons 2000). The rule Ax, as in the previous set of proof rules, bootstraps an argument from a single item of information, while the rules $\wedge$-I and $\rightarrow$-E are typical natural deduction rules — the rules for introducing a conjunction and eliminating implication — augmented with the combination of degrees of belief, and the collection of information on which data and proof rules have been used. (The full consequence relation would need an introduction rule and elimination rule for every connective in the language, and the definition of these is easy enough — we omit them here in the interests of space.)

The new rule in Figure 3 is the rule named Trust. This says that if it is possible to construct an argument for θ from some Δ_j^i with degree of belief $\tilde{d}$, and i trusts j with degree of trust $\tilde{e}$, then i has an argument for θ. The grounds of this argument combine all the data that was used from Δ_j^i, and all the information about trust used to determine that i trusts j, and the set of rules in the argument record all the inferences needed to build this combined argument. Finally, the belief i has in the argument is the belief in θ as it was derived from Δ_j^i combined with the trust i has in j. We carry out this last combination by first turning the trust value into a belief using some suitable function $ttb(\cdot)$.

In other words, this rule sanctions the use of information from an agent's acquaintances, provided that the degree of

$$\text{Ax}\,\frac{(n : \theta : \tilde{d}) \in \Delta_i^{bel}}{\Delta_i \vdash_{bel} (\theta : G : \{Ax\} : \tilde{d})}$$

$$\text{Trust}\,\frac{\Delta_i^{tr} \vdash_{tr} (trusts(i,j) : G : R : \tilde{d}) \text{ and } \Delta_i^{j} \vdash_{bel} (\theta : H : S : \tilde{e})}{\Delta_i \vdash_{bel} (\theta : G \cup H : R \cup S \cup \{Trust\} : ttb(\tilde{d}) \otimes^{bel} \tilde{e})}$$

$$\wedge\text{-I}\,\frac{\Delta_i \vdash_{bel} (\theta : G : R : \tilde{e}) \text{ and } \Delta_i \vdash_{bel} (\phi : H : S : \tilde{d})}{\Delta_i \vdash_{bel} (\theta \wedge \phi : G \cup H : R \cup S \cup \{\wedge\text{-I}\} : \tilde{d} \otimes^{bel} \tilde{e})}$$

$$\to\text{-E}\,\frac{\Delta_i \vdash_{bel} (\theta : G : R : \tilde{d}) \text{ and } \Delta_i \vdash_{bel} (\theta \to \phi : H : S : \tilde{e})}{\Delta_i \vdash_{bel} (\phi : G \cup H : R \cup S \cup \{\to\text{-E}\}) : \tilde{d} \otimes^{bel} \tilde{e})}$$

Figure 3: Part of the bel consequence relation

belief in that piece of information is modified by the agent's trust in that acquaintance.

Summary

In this paper we have outlined work on reasoning about trust using a form of logic-based argumentation which, as the paper demonstrated, can be integrated with a system of argumentation that uses the conclusions about trust. A notable feature of the system for reasoning about trust is its flexibility — new approaches to propagating trust can easily be added (or, indeed, removed) by altering the proof rules that are used in propagation.

Clearly the systems we have described are work in progress. Neither of the formal systems is complete as presented — both are missing much of the proof mechanism and a proper description of the syntax at the very least — and neither is rigorously evaluated. Our aim was simply to illustrate the basic ideas captured in the systems, and to illustrate the possibilities that they offer. Our future work will, in due course, fill in the details that are missing here. However, we believe that the work we have presented here has value in describing an area of research that we think is interesting and identifying some new approaches to handling it.

Acknowledgement

Research was sponsored by the Army Research Laboratory and was accomplished under Cooperative Agreement Number W911NF-09-2-0053. The views and conclusions contained in this document are those of the authors and should not be interpreted as representing the official policies, either expressed or implied, of the Army Research Laboratory or the U.S. Government. The U.S. Government is authorized to reproduce and distribute reprints for Government purposes notwithstanding any copyright notation here on.

References

Artz, D., and Gil, Y. 2007. A survey of trust in computer science and the semantic web. *Journal of Web Semantics* 5(2):58–71.

Geerts, F.; Kementsiedtsidis, A.; and Milano, D. 2006. iMondrian: A visual tool to annotate and query scientific databases. In *10th International Conference on Extending Database Technology*, volume 3896 of *Lecture Notes in Computer Scienc*. Berlin: Springer-Verlag.

Golbeck, J. 2006. Combining provenance with trust in social networks for semantic web content filtering. In *Proceedings of the International Provenance and Annotation Workshop*.

Grandison, T., and Sloman, M. 2000. A survey of trust in internet applications. *IEEE Communications Surveys and Tutorials* 4(4):2–16.

Guha, R.; Kumar, R.; Raghavan, P.; and Tomkins, A. 2004. Propagation of trust and distrust. In *Proceedings of the 13th International Conference on the World Wide Web*.

Jøsang, A.; Keser, C.; and Dimitrakos, T. 2005. Can we manage trust? In *Proceedings of the 3rd International Conference on Trust Management*.

Katz, Y., and Golbeck, J. 2006. Social network-based trust in prioritzed default logic. In *Proceedings of the 21st National Conference on Artificial Intelligence*.

Loui, R. P. 1987. Defeat among arguments: a system of defeasible inference. *Computational Intelligence* 3(3):100–106.

McBurney, P., and Parsons, S. 2000. Tenacious tortoises: A formalism for argument over rules of inference. In *Proceedings of the ECAI Workshop on Computational Dialectics*.

Parsons, S.; McBurney, P.; and Sklar, E. 2010. Reasoning about trust using argumentation: A position paper. In *Proceedings of the Workshop on Argumentation in Multiagent Systems*.

Wang, Y., and Singh, M. P. 2006. Trust representation and aggregation in a distributed agent system. In *Proceedings of the 21st National Conference on Artificial Intelligence*.