

Trust in Context

Abhaya C. Nayak

Department of Computing
Macquarie University
Sydney, Australia 2109
Abhaya.Nayak@mq.edu.au

Abstract. This paper examines how Spohn's *Ordinal Conditional Functions*, which are used to model belief dynamics, when appropriately adapted and interpreted, can fruitfully model the dynamics of trust. In this framework, trust is defined in terms of lack of trust (trust deficit) which is taken to be a primitive notion. It explores the notion of context sensitive trust, i.e., how in different domains of human interaction, one agent might trust another agent in varying degrees, and suggests the dynamics of such trust.

One can judge a man by the company he keeps.
Euripides (480 – 406 A.D.)

1 Introduction

The concept of trust is playing an increasingly important role in different fields related to information technology [13], such as multi agent systems [21], communication networks [15], and social networks [9]. Given the similarity between *trust* and *belief*, as betrayed by the words used in many languages to denote them, such as in Hebrew (*emun/emunah*), Latin (*credo*) and Sanskrit (*bishwas*), one would expect that the approaches adopted to model beliefs can be easily adapted to represent trust and reason with it. In fact, some existing approaches have already done it. In some accounts, this acknowledgement is only implicit. For instance, Yu and Singh [23] employ a Dempster-Shafer *belief functions* [17] with respect to every agent *a* telling to what extent *a* is to be trusted, and to what extent distrusted, and use Dempster Rules of combination to process recommendations about an agent. Similarly Colin Tan's approach to trust computation [20] also uses belief functions. In some other approaches, such as Demolombe's account of graded trust [4], the connection between belief and trust is made quite explicit.

Both belief and trust are mental attitudes; nonetheless there is a crucial difference. Belief is a propositional attitude, but trust is not. In the sentence *John believes X*, the place holder *X* is more likely than not to be a proposition, as in, "John believes *that dragons breath fire*". On the other hand, in the sentence *John trusts Y*, the place holder *Y* is more likely than not to be a person (or agent), as in "John trusts Betty but not Adam" which says something akin to: *John trusts (can count on) what Betty will do, but not on what Adam will*. Hence, while as an attitude, belief is propositional, trust is *actional*. Thus a thorough investigation into the foundation of trust must take into account not only the agents, but also their ability to perform different actions, their

intention behind performing (or not performing) some action, as well as the relevant *speech acts* [14]. Literature in the area [3, 8, 18] indicates support for such a view point.

In an earlier work [11] I proposed an alternative framework for reasoning about trust, inspired by Wolfgang Spohn's account of *ordinal conditional functions* [19] that is used to model belief dynamics. In this work, invoking political realism, I took *distrust* or *trust deficit* to be a more basic notion, and defined trust via distrust, and proposed how the statics and dynamics of trust can be represented. The notion of trust in this account is also *graded trust* as in Demolombe's account [4]; however it is also *unqualified trust*. It is as if, if I were to trust John more than I trust Jack, I must trust John more in every context. It is possible that John is better at writing programs and Jack at proof reading; so depending upon the context I might trust Jack more than John, or the other way around. Hence, the notion of trust must not only be graded, but also be contextualised. The basic sentences in the language of trust would have the form somewhat along the line: (*agent*) *A trusts (agent) B in the context (of carrying out the task) C to the degree D*. Ungraded trust as well as unqualified (bare) trust should be reducible to such fine-grained trust. The aim of this paper is to develop the semantic framework for such an account. For the sake of simplicity, the framework does not take into account the intentions and plans of agents, and the proposal is primarily semantic. Since the framework for trust being developed is very new, not much can be assumed as background knowledge. Hence I will need to introduce a relatively large number of concepts.

In the section *Bare Trust* we summarise and expand upon the account of the statics of trust developed in [11]. This accounts for graded, but unqualified trust. In the next section, *Trust Contextualized*, we examine a qualified notion of trust. The account of how the trustworthiness of different agents are updated is developed in the section *Trust Dynamics*. In the final section we conclude with a brief outline of some related issues that we will take up in our future work.

2 Bare Trust

In a Hobbesian state of nature, there is “continuall feare, and danger of violent death; And the life of man, solitary, poore, nasty, brutish, and short” [7]. If we assume that trust and cooperation must emerge out of human behaviour in such a state of nature [2], then it is not trust but rather its deficit, *distrust* that should be taken to be a more basic concept, and trust be defined in terms of it. However, trust is more than just the absence of distrust. We define trust in terms of suspicion in the following manner:

Definition 1. *An agent a trusts b iff a distrusts (suspicious of) the “enemies” of b.*

This definition of trust via suspicion or distrust raises two immediate questions.

1. *Why should an agent a's trust in b depend on a's attitude towards the enemies of b at all? After all it is conceivable that a distrusts (or, for that matter trusts) both b and the enemies of b!*¹ In response, we note that there is no mathematical compulsion to define a's trust in b in terms of a's attitude towards the enemies of b. However, as the Euripidean saying (quoted on the title page) implies, a person's

¹ I am thankful to Professor David Makinson for raising this issue in a private communication.

character may be judged based on the character of their friends. By the same token, a person's character may also be judged based on the character of their enemies. As we would see later, we are advocating that a 's assessment of both the enemies as well as friends of b should be taken into account in order to judge a 's trust in b . The scene of a (dis-)trusting both b and the enemies of b is a special case; and in our model it will correspond to the situation where a suspends making any "trust judgment" on b .

2. *What will happen if the agent b does not have any enemy at all? Would it then mean that a cannot trust b ?* This is a special case, and it is not really that important how we deal with it. We can avoid the problem by assuming that every agent has at least one enemy. It is a fair assumption since (a) in general, by eliminating enemies, an agent creates more enemies, and (b) success of an agent creates envy and enmity.

Accordingly, we make the following two assumptions, while Definition 2 provides an operational (but very informal) definition of suspicion.

Assumption 1. *Every agent a has at least one enemy b .*

Assumption 2. *No agent is paranoid; i.e., for every agent a there is an agent b (different from a) such that a is not suspicious of b .*

Definition 2. *An agent a is suspicious of b just in case a expects to be surprised or disappointed by b 's behaviour at some point.*

Agent a is suspicious of b just in case a expects that b may disappoint a at any moment. Thus a 's trust in b is reducible to b 's potential for disappointing/surprising a . In this context Shackle's concept of *degree of potential surprise* [16] that is similar to Wolfgang Spohn's notion of *ordinal conditional functions* [19] is very pertinent, considering its role in the development of the *logic of belief change* [1, 5, 12]. The framework developed below is inspired by Spohn's account of belief change.

We will assume a pre-theoretic notion of *agent*. Let $\mathcal{A}^+ = \{a_0, a_1, \dots, a_n\}$ be the set of all agents. We use a with possible decoration to denote individual agents. Presumably, members of $\mathcal{A}^+$ observe each other's behaviour, manipulate them, trust them, distrust them, and what have you. We are not going to deal with most of those issues. We will, instead, assume a contextually fixed, prominent member, say a_0 , and develop an account of how the trust of a_0 in *other members* of $\mathcal{A}^+$, namely $\mathcal{A} = \{a_1, \dots, a_n\}$, is defined and maintained. So although a_0 is a member of the agent-system, and is subject to all the systemic constraints that other members of $\mathcal{A}$ are subject to, and members of $\mathcal{A}$ trust/distrust a_0 just as a_0 does to them, we will abstract away most of those things. In particular, we will assume that a_0 has the ability to distinguish itself from the rest of $\mathcal{A}^+$, and primarily deals with the members of $\mathcal{A}$ in its "model of the world".

Suspicion Directed Toward an Individual

We assume that a_0 has views as to which members of $\mathcal{A}$ are friends (resp. foes) of which members. Hence, a_0 assigns to each member a_i of $\mathcal{A}$ a set of members of $\mathcal{A}$ that it considers are friends of a_i , and another set of members that it considers are foes of a_i . Hence we assume two unary functions: $friends_{a_0}(\cdot)$ and $foes_{a_0}(\cdot)$ as defined below. Since the agent a_0 is contextually fixed, we will drop the subscripts for readability.

Definition 3

1. The function $friends : \mathcal{A} \rightarrow 2^{\mathcal{A}}$ maps an agent $a \in \mathcal{A}$ to the set of agents $friends(a)$ that are considered (by a_0) to be a 's friends.
2. The function $foes : \mathcal{A} \rightarrow 2^{\mathcal{A}}$ maps an agent $a \in \mathcal{A}$ to the set of agents $foes(a)$ that are considered (by a_0) to be a 's enemies.
3. The sets $friends(a)$ and $foes(a)$ are mutually disjoint for every agent $a \in \mathcal{A}$.

We leave open many issues including where a_0 obtains these two functions from, and whether, the *Friend* and *Foe* relations corresponding to the $friends(\cdot)$ and $foes(\cdot)$ functions are reflexive, symmetric or transitive. We can graphically illustrate who is whose friend (or enemy) by drawing different types of arrows. In Fig. 1 below, we take $\mathcal{A}$ to be composed of agents $a_1 \dots a_{10}$. We identify the sets $friends(a_1) = \{a_4, a_9\}$ and $foes(a_1) = \{a_7, a_{10}\}$ by drawing appropriate solid and broken arrows respectively. The friends and enemies of other agents are not being shown to avoid clutter.

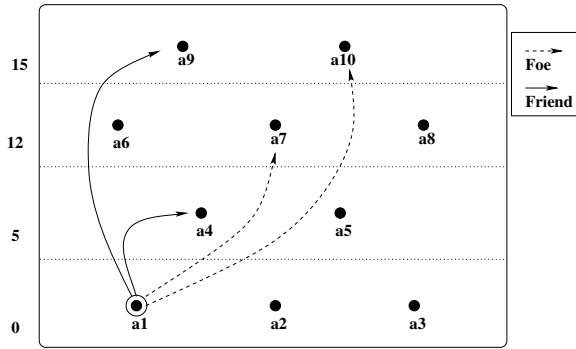


Fig. 1. The trustworthiness of agent a_1 is 12 since its suspicion index is 0 and the least degree of suspicion directed toward its foes is 12

Apart from keeping information as to who is whose friend or enemy, agent a_0 also has its opinion on who is likely to disappoint it how much. We do this by assigning another function $susp_{a_0}(\cdot)$ to it, which, for each agent a in $\mathcal{A}$, gives a non-negative integer indicating a 's degree of suspicion (trust deficit, or potential disappointment). As before, we will drop the subscript a_0 for readability.

Definition 4. The suspicion function $susp : \mathcal{A} \rightarrow \mathcal{N}$ maps an agent $a \in \mathcal{A}$ to a non-negative integer $susp(a)$ which indicates how disappointing a could be.

If $susp(a)$ is 0, it indicates that the agent a_0 is not suspicious of a at all. On the other hand, an inequality such as $susp(a_i) < susp(a_j)$ would indicate that the agent a_0 is more suspicious of a_j than of a_i . For all practical purpose, these values are ordinal numbers, and we use them by and large for comparison. It immediately follows from Assumption 2 that agent a_0 is not paranoid if and only if $susp(a) = 0$ for some agent $a \in \mathcal{A}$. Accordingly, in Fig. 1, there are some agents (namely, a_1, a_2 , and a_3) that are assigned the degree of potential suspicion 0. In fact, Assumption 2 leads to a more general constraint on the functions $susp(\cdot)$ as follows:

Observation 1. *In presence of Assumption 2, given an arbitrary agent a_i , there exists some agent $a_j \in \mathcal{A}^+ \setminus \{a_i\}$ such that $\text{susp}_{a_i}(a_j) = 0$.*

This observation states that since the agents we are dealing with are not paranoid, each of them places the minimum possible degree of distrust, 0, on at least one agent.

Suspicion Directed Toward a Group

Suspicion is often directed toward a group of individuals. For instance, during an election, one might have strong suspicion against a particular interest groups. In principle, any subset $\mathcal{A}' \subseteq \mathcal{A}$ could constitute an interest group; and hence we should be able to discuss the degree to which a_0 suspects $\mathcal{A}'$. In general, a group of individuals will have an underlying power structure which will play a crucial role in determining an individual's attitude towards it. If the group leader is not trustworthy, individuals will have little trust in the group itself. However, we do not assume any power structure for the groups. In absence of such a structure it is not quite clear how suspicion directed towards a group can be reduced to suspicion directed towards its members. Analogous to the *weakest link principle*, we postulate that *a group is only as bad as its least bad member*, implying that no better person could have survived as a member of the group.² Accordingly we generalize the definition of the function $\text{susp}(\cdot)$ as follows:

Definition 5. $\text{susp}(\mathcal{A}') = \min_{a \in \mathcal{A}'} \text{susp}(a)$ for any group of agents $\mathcal{A}' \subseteq \mathcal{A}$.

Note that this general definition of $\text{susp}(\cdot)$ gracefully deals with the special case when $\mathcal{A}'$ is a singleton set in that $\text{susp}(\{a\}) = \text{susp}(a)$ for any $a \in \mathcal{A}$. Going back to Fig. 1, of particular interest would be the calculation of the degree to which agent a_0 would be suspicious of the friends of a_1 and of the enemies of a_1 , that is, calculation of the values of $\text{susp}(\text{friends}(a_1))$ and of $\text{susp}(\text{foes}(a_1))$. Note that in this figure the set $\mathcal{A}$ is partitioned into equivalence classes modulo the susp -rank of its members, and the ranks (0, 5, 12 and 15) of different agents are indicated on the left. It is easily verified that these two values are respectively 5 and 12.

From Suspicion to Trust

As noted earlier as a rationale behind Definition 1, trust is more than simply the lack of suspicion or distrust. In order that agent a_0 positively trust some agent a , it is a necessary condition that a_0 bears no suspicion toward a ; but that does not constitute a sufficient condition. Although I have no reason to be suspicious of a perfect stranger, I have no reason to trust them either. We can now offer a more formal definition of trust. As before, the reference to the agent a_0 is being dropped for readability.

Definition 6

1. *An agent a is trusted (by a_0) if and only if both $\text{susp}(a) = 0$ and $\text{susp}(\text{foes}(a)) > 0$ for all $a \in \mathcal{A}$.*
2. *Given an agent $a \in \mathcal{A}$ trusted by a_0 , the degree of trust that a_0 places on a is given by $\text{trust}(a) = \text{susp}(\text{foes}(a))$.*

² Indeed, as an image building exercise, often “bad” groups enrol “good” individuals, intending to mislead others. This deceit works precisely because there is a presumption that the group could not be worse than its least bad member.

Thus, going back to Fig. 1, the degree of trust that a_0 places on the agent a_1 is 12 since $susp(a_1) = 0$ and $susp(foes(a_1)) = susp(\{a_7, a_{10}\}) = 12$. On the other hand, since $susp(a_5) = 5$, it follows that a_0 does not trust a_5 at all. In fact, a_0 distrusts (is suspicious of) a_5 to the degree 5.

Now we are in a position to define *trust in a group*. Trusting a set of agents $\mathcal{A}'$ entails trusting every member of that group. The degree of that trust is the minimal trust earned by members of that group in the relevant manner.³

Definition 7

1. Given any $\mathcal{A}' \subseteq \mathcal{A}$, agent a_0 trusts $\mathcal{A}'$ if and only if a_0 trusts every agent $a \in \mathcal{A}'$.
2. Given that agent a_0 trusts some $\mathcal{A}' \subseteq \mathcal{A}$, the degree of trust that a_0 places on $\mathcal{A}'$ is given by $trust(\mathcal{A}') = \min_{a \in \mathcal{A}'} trust(a)$.

Referring back to Fig. 1, we easily verify that $trust(\{a_1, a_5, a_6\})$ is undefined since the agents a_5 and a_6 are not trusted. On the other hand, $trust(\{a_1, a_2, a_3\})$ cannot be evaluated at this point since the values of $foes(a_2)$ and $foes(a_3)$ are not available.

3 Trust Contextualised

The account we provided of trust above pretends that trust is not context-sensitive. However, trust is an actional attitude – it is sensitive to the action at hand, and perhaps to other contextual factors. We will assume that there are certain things called *contexts*, and actions are part of the relevant contexts. We will not investigate into the inner mechanism of contexts, but assume that a context encodes such information as the relevant action, and the agent's plan, ability, and intention. Our purpose here is simply to generalize the account of bare trust presented earlier in order that contexts can be incorporated into the accounts of trust. Accordingly, we assume that there is a fixed set of contexts, $\mathcal{C}$, and we denote the member contexts by c with possible decorations.

From the Section *Bare Trust* it is clear that the two central notions used here are the functions $friends(\cdot)$ vs. $foes(\cdot)$ on the one hand and the function $susp(\cdot)$ on the other. The concept of trust is defined in terms of them. We will need to consider which of these primary notions need to be parametrized by a context argument.

Let us first examine if the functions $friends(\cdot)$ and $foes(\cdot)$ should be contextualised. We note that the words “friend” and “foe” as used here are convenient labels standing for similar clusters of properties as the concepts *friend* and *enemy*. In a certain sense, one might take the concept *friends* to denote a cohort group; thus, for instance, the set of agents that a hangs around with in a fishing trip could be very different from the set of agents it accompanies in a shopping expedition. We can similarly argue that one's set of *enemies* can change from context to context. Note that letting these functions accept the context as a parameter will not force us to change the output of the functions as the context changes. Accordingly we introduce two corresponding functions as follows:

³ It is not always correct. I might trust a company because its CEO is my trusted friend, even if I don't trust many employees of this company. But such power relations cannot be captured in our framework without adding further structure to it.

Definition 8

1. The function $\text{friends}_{\text{con}} : \mathcal{A} \times \mathcal{C} \longrightarrow 2^{\mathcal{A}}$ maps an agent $a \in \mathcal{A}$ in a context $c \in \mathcal{C}$ to the set $\text{friends}_{\text{con}}(a, c)$ considered by a_0 to be a 's friends in the context c .
2. The function $\text{foes}_{\text{con}} : \mathcal{A} \times \mathcal{C} \longrightarrow 2^{\mathcal{A}}$ maps an agent $a \in \mathcal{A}$ in a context $c \in \mathcal{C}$ to the set $\text{foes}_{\text{con}}(a, c)$ considered by a_0 to be a 's enemies in the context c .

It is very natural to assume that distrust is context sensitive. Accordingly:

Definition 9. The function $\text{susp}_{\text{con}} : \mathcal{A} \times \mathcal{C} \longrightarrow \mathcal{N}$ maps $a \in \mathcal{A}$ in $c \in \mathcal{C}$ to a non-negative integer $\text{susp}_{\text{con}}(a, c)$ telling how disappointing a could be to a_0 in that context.

Definition 10. $\text{susp}_{\text{con}}(\mathcal{A}', c) = \min_{a \in \mathcal{A}'} \text{susp}_{\text{con}}(a, c)$ for any group of agents $\mathcal{A}' \subseteq \mathcal{A}$ and context $c \in \mathcal{C}$.

We again refer to the figure Fig. 1 which can be taken to illustrate the the notion of suspicion or distrust in a given context c . Finally, we contextualize the notions of trust (directed towards an individual as well as at a group) in the obvious fashion.

Definition 11

1. An agent a is trusted (by a_0) in context c if and only if both $\text{susp}_{\text{con}}(a, c) = 0$ and $\text{susp}_{\text{con}}(\text{foes}_{\text{con}}(a, c)) > 0$ for all $a \in \mathcal{A}$.
2. Given an agent $a \in \mathcal{A}$ trusted by a_0 , the degree of trust that a_0 places on a in c is given by $\text{trust}_{\text{con}}(a, c) = \text{susp}_{\text{con}}(\text{foes}_{\text{con}}(a, c))$.

Definition 12

1. Given any $\mathcal{A}' \subseteq \mathcal{A}$ and context $c \in \mathcal{C}$, agent a_0 trusts $\mathcal{A}'$ in context c if and only if a_0 trusts every agent $a \in \mathcal{A}'$ in context c .
2. Given that agent a_0 trusts some $\mathcal{A}' \subseteq \mathcal{A}$ in context c , the degree of trust that a_0 places on $\mathcal{A}'$ in c is given by $\text{trust}_{\text{con}}(\mathcal{A}', c) = \min_{a \in \mathcal{A}'} \text{trust}_{\text{con}}(a, c)$.

Thus we see that the concept of bare trust is naturally generalised to a more notion of trust that is context sensitive, and hence is a more appropriate model of trust.

4 The Dynamics of Trust

In the section *Bare Trust* we outlined the basic framework to describe a *trust state* that is in equilibrium. In the subsequent section *Trust Contextualised* we generalised this framework to reflect that trust is a context-sensitive notion. Apart from context, trust also changes over time subject to an agent making new observations, receiving new information, and even receiving recommendations from others. For instance, if agent a_0 were to receive a recommendation from a reputable source that a_5 is very trustworthy, a_0 's evaluation of the trustworthiness of a_5 will change. How this change is effected – the *dynamics of trust* – is the topic of this section. Without any loss of generality, we take it to be the study of how the function $\text{susp}(\cdot)$ [resp. $\text{susp}_{\text{con}}(\cdot)$] changes in response to relevant triggers. We will restrict the discussion of trust dynamics to how $\text{susp}(\cdot)$ responds to received recommendations of the appropriate sort.

Let us first note that *who* makes the recommendation makes a difference. If the recommendation comes from someone that a_0 considers to be completely unreliable, a_0 is likely to ignore the recommendation. On the other hand, if the recommendation comes from someone whose judgment a_0 trusts, then it will take up the recommendation seriously and “update” its trust values in different agents. There are other relevant factors as well. If the recommendation regarding a_j comes from a_i , and a_0 “knows” that a_j is a friend of a_i , then chances are a_0 would take the recommendation not as seriously as it normally would; it might in fact “discount” the recommendation to some extent. On the other hand, if a_j is believed to be an enemy of a_i , then it would have the opposite effect. The framework developed here can deal with these variations in a judicious manner. In this paper we will make many simplifying assumptions. In particular, we will assume that the agent a_0 considers the recommendation received to be practically infallible, the recommending agent has no (relevant) conflict of interest, and means what she says – in other words, no special *speech acts* [14] and *conversational implicature* [6] are being employed. We will first recapitulate the account of (bare) trust dynamics proposed in [11], and then outline a more general dynamics for contextual trust.

4.1 Bare Trust Dynamics

In this section we discuss how bare trust, represented in terms of the un-contextualised functions $susp(\cdot)$, $foes(\dots)$, etc., gets updated in light of received recommendations. Let us assume that agent a_0 has received a recommendation from an infallible source to trust agent a_5 to degree 3. This recommendation has the general form *new_trust(agent a , value x)*. In response, agent a_0 will update its function $susp(\cdot)$ to, say, $susp'(\cdot) = susp(\cdot \mid \langle a, x \rangle)$ such that when the trust value of agent a is computed modulo $susp'(\cdot)$, we will get the result x . Borrowing terminology from probability theory, we will term $susp(\cdot)$ to be the *prior suspicion function* of the agent a_0 , and $susp'(\cdot) = susp(\cdot \mid \langle a, x \rangle)$ to be its *posterior suspicion function*. When no confusion is imminent, we will represent the posterior $susp'(\cdot) = susp(\cdot \mid \langle a, x \rangle)$ simply as $susp_{\langle a, x \rangle}(\cdot)$.

Definition 13

1. A trust recommendation is denoted by a pair $\langle a, x \rangle$ where $a \in \mathcal{A}$ is an agent, and $x \in \mathcal{N}$ is the recommended trust-value of a .
2. $susp(\cdot \mid \langle a, x \rangle)$ represents the “posterior suspicion function”, also denoted $susp_{\langle a, x \rangle}(\cdot)$, after the prior $susp(\cdot)$ has been updated in light of the trust recommendation $\langle a, x \rangle$.

Since the recommendation is accepted as is, the posterior suspicion function $susp_{\langle a, x \rangle}(\cdot)$ should be so constructed that the trust value for the agent a generated from it as per Definition 6 should be x . This naturally leads to the following two constraints:

Assumption 3

1. $susp_{\langle a, x \rangle}(a) = 0$
2. $susp_{\langle a, x \rangle}(foes(a)) = x$

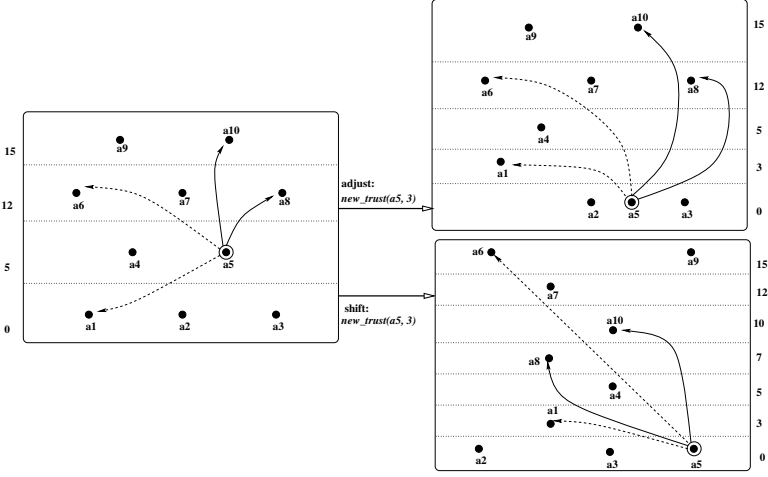


Fig. 2. Agent a_5 who was distrusted to degree 5, is now trusted to degree 3

Let us consider the prior suspicion function $susp(\cdot)$ represented by the left rectangle in Fig. 2. The agent a_0 distrusts a_5 to degree 5; the friends of a_5 are two, namely a_8 and a_{10} ; and a_5 also has two enemies: a_1 and a_6 . Suppose also that the agent a_0 receives a recommendation $\langle a_5, 3 \rangle$ that it must accept at face value. Given Assumption 3, it then follows that $susp_{\langle a_5, 3 \rangle}(a_5) = 0$, and $susp_{\langle a_5, 3 \rangle}(\{a_1, a_6\}) = 3$. This raises three questions.

1. Clearly the posterior suspicion function $susp'(\cdot) = susp_{\langle a_5, 3 \rangle}(\cdot)$ will assign a_5 the value 0. What would it do to the friends of a_5 ? For instance, should $susp(a_8)$ be left untouched in this updating process? If not, what would be the correct value?
2. Since $susp_{\langle a_5, 3 \rangle}(\{a_1, a_6\}) = 3$, the minimum of the values assigned by $susp_{\langle a_5, 3 \rangle}(\cdot)$ to a_1 and a_6 is 3. But which of them should receive this minimum value?
3. As in the case (1) above, given that in this trust updating process, the $susp(\cdot)$ value of a_1 is changed from 0 to 3, should that of a_6 be left unchanged?

Each of these issues admits multiple, even bizarre solutions such as:

Proposal 1. For all $a' \in \mathcal{A}$,

$$susp_{\langle a_5, 3 \rangle}(a') = \begin{cases} 0 & \text{if either } a' = a_5 \text{ or } a' \in \text{friends}(a_5) \\ 3 & \text{otherwise} \end{cases}$$

In order to block such solutions, we propose *trust adjustment* and *trust shift*, two proposals that are inspired by work in iterated belief revision [10, 22].

Trust Adjustment. This approach strictly follows the *principle of minimal repair*. We may assume that agent a_0 's function $susp(\cdot)$ summarizes a lot of valuable, historical information, that should not be lost without very good reason. However, Proposal 1 will obliterate such distinction, even when the trust recommendation, $\langle a_5, 3 \rangle$ has nothing

to do, directly or indirectly, either with a_2 or with a_9 . It is guilty of *over repair*. We should not rectify the $susp(\cdot)$ function more than necessary. The changes that are *really necessary* are explicitly identified in Assumption 3. If we follow this recipe, then, in the context of Fig.2, the following modification in $susp(\cdot)$ would appear to be appropriate:

Proposal 2. For all $a' \in \mathcal{A}$,

$$susp_{\langle a_5, 3 \rangle}(a') = \begin{cases} 0 & \text{if } a' = a_5 \\ 3 & \text{if } a' \text{ is } susp\text{-minimalin } foes(a_5) \\ susp(a') & \text{otherwise.} \end{cases}$$

According to this proposal, the $susp$ -value of a_5 will be reduced to 0, and that of a_1 will be increased to 3. The $susp$ -value of every other agent in $\mathcal{A}$ will be left as is, ensuring that the a_5 will be trusted to degree 3 with minimal repair to the function $susp(\cdot)$. However, in general this approach will not work. In particular, suppose that the agent a_5 had one more enemy, say a_4 with $susp(a_4) = 2$. In such a case, after $susp(\cdot)$ was updated to $susp_{\langle a_5, 3 \rangle}(\cdot)$ in response to the received trust-recommendation $\langle a_5, 3 \rangle$ following the procedure specified in Proposal 2, the agent will be trusted to the degree 2 instead of the desired degree 3. Hence a slight correction in Proposal 2 is due:

Definition 14 (Trust Adjustment). Let $a, a' \in \mathcal{A}$ and $x \in \mathcal{N}$. The updated suspicion function is given by:

$$susp_{\langle a, x \rangle}^{adjust}(a') = \begin{cases} 0 & \text{if } a = a' \\ x & \text{if both } a' \in foes(a), \text{ and } susp(a') < x \\ susp(a') & \text{otherwise.} \end{cases}$$

The figure Fig. 2 (top-right) graphically illustrates how the $susp$ -values of different agents get adjusted in this process.

Trust Shift. Although *trust adjustment* satisfies both Assumption 3, and the principle of minimal repair, the way it distorts the relationship between different friends (or enemies) of the agent a is debatable. Consider again Fig. 2. All the three friends, a_5, a_8 and a_{10} were distrusted to various degrees, ranging from 5 to 15. A recommendation was received that the agent a_5 is fairly trust worthy, and should be trusted to degree 3, and a_0 accordingly trusted a_5 . It is only rational to expect that agent a_0 would now distrust a_8 and a_{10} less than it did before. But that does not happen. Similarly, this recommendation “taints” a_5 ’s enemy a_1 , who was not distrusted before. However, there is no corresponding loss of trust in a_6 , the other enemy of a_5 . It will be more appropriate to shift *en bloc* the suspicion values of a_5 ’s friends (or enemies), up or down as the case may be, by the same magnitude. This is captured as follows:

Proposal 3. For all $a' \in \mathcal{A}$,

$$susp_{\langle a_5, 3 \rangle}(a') = \begin{cases} 0 & \text{if } a' = a_5 \\ \max(0, (susp(a') - 5)) & \text{if } a' \in friends(a_5) \\ susp(a') + 3 & \text{if } a' \in foes(a_5) \\ susp(a') & \text{otherwise.} \end{cases}$$

The result of such uniform shift is illustrated in the figure Fig.2 (bottom-right). It is easily verified that the relative *trust deficit* between different friends (or enemies) of a_5 is not affected in this process. Definition 15 generalises this proposal.

Definition 15 (Trust Shift). Let $a, a' \in \mathcal{A}$, $x \in \mathcal{N}$, and y denote $\text{susp}(\text{foes}(a))$. Then, the new (updated) suspicion function is given by:

$$\text{susp}_{(a,x)}^{\text{shift}}(a') = \begin{cases} 0 & \text{if } a = a' \\ \max(0, (\text{susp}(a') - \text{susp}(a))) & \text{if } a' \in \text{friends}(a) \\ \text{susp}(a') - y + x & \text{if both } a' \in \text{foes}(a) \text{ and } x \neq y \\ \text{susp}(a') & \text{otherwise.} \end{cases}$$

The first clause in this definition ensures that a is not distrusted. This reduces the *susp*-value of a by $\text{susp}(a)$; and that is the magnitude by which we would like to reduce the *susp*-value of all its friends. This is done in the second clause, with the proviso that the revised *susp*-value has a floor value of 0. The more interesting case is captured by the third clause. Note that y , defined as $\text{susp}(\text{foes}(a))$, is really, as per Definition 5, the minimal *susp*-value associated with any enemy of a . Since a' is an enemy of a , it follows that y will never exceed $\text{susp}(a')$, and hence $\text{susp}(a') - y + x$ will always be non-negative, and the *susp*-value of all enemies of a will be uniformly shifted by the same magnitude $|x - y|$, as desired. The last clause effectly says that agents that are neither friends nor enemies of a are not affected.

4.2 Dynamics of Contextual Trust

Since trust is an actional attitude, one's trust towards a person could fluctuate depending on the action being considered to be carried out. While in the section *Bare Trust* we ignored this aspect of trust, in the section *Trust Contextualised* we sensitized our notion of trust by introducing *context* as an extra parameter to the relevant functions, $\text{friends}_{\text{con}}$, foes_{con} and susp_{con} being the primary ones. In the current section, *The Dynamics of Trust*, so far we have looked at the dynamics of bare trust. We now consider the dynamics of trust that is sensitive to context.

We assume that the current trust state of the agent a_0 is given by a function $\text{susp}_{\text{con}}(\cdot)$ that accepts two parameters, namely, an agent a and a context c , and returns a suspicion value. Since the recommendation of the form $\text{new_trust}(\text{agent } a, \text{value } x)$ does not tell in which context the recommendation is being made, the only reasonable thing the agent a_0 can do with it is to discard it altogether, or to revise all the trust values it has for a . The first option wastes a recommendation, the second results in serious loss of information. Hence it is appropriate to assume that that the recommendation will have the general form $\text{new_trust}(\text{agent } a, \text{value } x, \text{context } c)$. The intended reading is that agent a_0 's trust in a should be left unchanged in all contexts except c where the new trust value should be x . Accordingly, we generalise Definitions 14 and 15.

Definition 16 (Contextual Trust Adjustment). Let $a, a' \in \mathcal{A}$, $x \in \mathcal{N}$, and $c \in \mathcal{C}$. Let the (contextualized) trust state of an agent a_0 be given by the suspicion function $\text{susp}_{\text{con}}(\cdot)$. Then, the revised suspicion function is given by:

$$susp_{con(a,x,c)}^{adjust}(a') = \begin{cases} susp_{con}(a',c') & \text{if } c \neq c' \\ \mathbf{ELSE} \text{ (i.e. if } c = c') & \\ 0 & \text{if } a = a' \\ x & \text{if both } a' \in foes_{con}(a,c), \\ & \text{and } susp_{con}(a',c) < x \\ susp_{con}(a',c) & \text{otherwise.} \end{cases}$$

It is easily verified that the Definition 16 leaves the trust state unchanged when the input context is different from the context in which the agent a is being “assessed”, and in the relevant context c it changes as does bare trust adjustment defined by Definition 14.

Definition 17 (Contextual Trust Shift). Let $a, a' \in \mathcal{A}$, $x \in \mathcal{N}$, and $c \in \mathcal{C}$. Let the (contextualized) trust state of an agent a_0 be given by the suspicion function $susp_{con}(\cdot)$. Also, let y denote $susp_{con}(foes_{con}(a))$. Then, the revised suspicion function is given by:

$$susp_{con(a,x,c)}^{shift}(a') = \begin{cases} susp_{con}(a',c') & \text{if } c \neq c' \\ \mathbf{ELSE} \text{ (i.e. if } c = c') & \\ 0 & \text{if } a = a' \\ \max(0, (susp_{con}(a',c) - \\ \quad susp_{con}(a,c))) & \text{if } a' \in friends_{con}(a,c) \\ susp_{con}(a',c) - y + x & \text{if both } a' \in foes_{con}(a,c) \text{ and } x \neq y \\ susp_{con}(a',c) & \text{otherwise.} \end{cases}$$

As in the case of the Definition 16, the trust state is left unchanged when the input context is different from the context in which the agent a is being “assessed”, otherwise it behaves like bare trust shift defined by Definition 15.

We end this section making an important but obvious point, that the definitions of trust adjustment and trust shift, be it for the bare version (Definitions 14 and 15) or the contextualised version (Definitions 16 and 17), indeed define how the appropriate *susp* functions change, and the change in the corresponding *trust* functions, defined in terms of the *susp* functions, is made is only implicitly defined.

5 Discussion and Future Work

In this paper we developed a simple framework for representing the statics of trust – both bare and contextualized – and provided two reasonable mechanisms for managing its dynamics. There are many issues that we will address in future work:

1. We intend to re-interpret less informative recommendations such as “trust a ” as: $\langle a, susp(foes(a)) \rangle$ (or as $\langle a, susp_{con}(foes(a)), c \rangle$), and investigate its consequences.
2. This framework can be enhanced to deal with recommendations about a group, or originating from a group.
3. The current trust-value (*reputation*) of the recommender is not taken into account while processing a reputation. This can be used to further fine-tune this framework.

4. Apart from the dynamics of the suspicion functions, we also need to investigate the dynamics of friendship and enmity (the functions *friends* and *foes*)
5. The account we provided is semantic. We need to devise an appropriate language of trust, and syntactically characterise the trust shift and trust adjustment operations.
6. Finally, it will be important to examine how it fares against existing relevant approaches such as [8], and how it can be practically validated.

References

1. Alchourrón, C.E., Gärdenfors, P., Makinson, D.: On the logic of theory change: Partial meet contraction and revision functions. *Journal of Symbolic Logic* 50, 510–530 (1985)
2. Axelrod, R.M.: *Evolution of Cooperation*. Basic Books (1984)
3. Castelfranchi, C., Tan, Y.H. (eds.): *Trust and Deception in Virtual Societies*. Springer (2001)
4. Demolombe, R.: Graded trust. In: *AAMAS Trust*, pp. 1–12 (2009)
5. Gärdenfors, P.: *Knowledge in Flux: Modeling the Dynamics of Epistemic States*. Bradford Books, MIT Press, Cambridge, Massachusetts (1988)
6. Grice, H.P.: Utterer's meaning and intentions. *Philosophical Review* 78, 147–177 (1969)
7. Hobbes, T.: *Leviathan or the Matter, Forme, & Power of a Common-Wealth Ecclesiastical and Civill* (Gutenberg EBook #3207). The Green Dragon in St. Paul's Churchyard (1651)
8. Jones, A.J.I.: On the concept of trust. *Decision Support Systems* 33(3), 225–232 (2002)
9. Jøsang, A., Hayward, R., Pope, S.: Trust network analysis with subjective logic. In: *Proc. of the 29th Australasian Comp. Sc. Conf., ACSC 2006*, vol. 48, pp. 85–94 (2006)
10. Nayak, A.C.: Iterated belief change based on epistemic entrenchment. *Erkenntnis* 41, 353–390 (1994)
11. Nayak, A.C.: The deficit and dynamics of trust. In: *EUC*, pp. 517–522. IEEE (2010)
12. Nayak, A.C., Pagnucco, M., Peppas, P.: Dynamic belief revision operators. *Artif. Intell.* 146(2), 193–228 (2003)
13. Sabater, J., Sierra, C.: Review on computational trust and reputation models. *Artif. Intell. Rev.* 24(1), 33–60 (2005)
14. Searle, J.: *Speech Acts: An Essay in the Philosophy of Language*. Cambridge University Press (1969)
15. Sen, J.: A Distributed Trust and Reputation Framework for Mobile Ad Hoc Networks. In: Meghanathan, N., Boumerdassi, S., Chaki, N., Nagamalai, D. (eds.) *CNSA 2010. CCIS*, vol. 89, pp. 538–547. Springer, Heidelberg (2010)
16. Shackle, G.L.S.: The logic of surprise. *Econometrica New Series* 149(78), 112–117 (1953)
17. Shafer, G.: *A Mathematical Theory of Evidence*. Princeton University Press (1976)
18. Singh, M.P.: Trust as dependence: a logical approach. In: Sonenberg, L., Stone, P., Tumer, K., Yolum, P. (eds.) *AAMAS*, pp. 863–870. IFAAMAS (2011)
19. Spohn, W.: Ordinal conditional functions: A dynamic theory of epistemic states. In: Harper, W.L., Skryms, B. (eds.) *Causation in Decision, Belief Change, and Statistics, II*, pp. 105–134. Kluwer Academic Publishers (1988)
20. Tan, C.K.Y.: A belief augmented frame computational trust model. In: Russell, I., Markov, Z. (eds.) *FLAIRS Conference*, pp. 647–652. AAAI Press (2005)
21. Wang, Y., Singh, M.P.: Formal trust model for multiagent systems. In: Veloso, M.M. (ed.) *IJCAI*, pp. 1551–1556 (2007)
22. Williams, M.A.: Transmutations of knowledge systems. In: Doyle, J., et al. (eds.) *KR*, pp. 619–629. Morgan Kaufmann (1994)
23. Yu, B., Singh, M.P.: Distributed reputation management for electronic commerce. *Computational Intelligence* 18(4), 535–549 (2002)