AAMAS 2009

Workshop on
TRUST IN AGENT SOCIETIES

Seventh International Conference on
Autonomous Agents & Multi-Agent Systems

May 11, 2009
Budapest, Hungary

Organizers:

Rino Falcone, Suzanne Barber, Jordi Sabater-Mir, Munindar Singh

DESCRIPTION OF THE WORKSHOP

The aim of the workshop is to bring together researchers who can contribute to a better understanding of trust and reputation in agent societies. Most agent models assume trustworthy communication to exist between agents. However, this ideal situation is seldom met in reality. In the human societies, many techniques (e.g. contracts, signatures, long-term personal relationships, reputation) have been evolved over time to detect and prevent deception and fraud in communication, exchanges and relations, and hence to assure trust between agents. Artificial societies will need analogous techniques.

Trust is more than secure communication, e.g., via public key cryptography techniques. For example, the reliability of information about the status of your trade partner has little to do with secure communication. With the growing impact of electronic societies, trust and privacy become more and more important.

Trust is important in applications such as human-computer interaction to model the relationship between users and their personal assistants. Different kinds of trust are needed: trust in the environment and in the infrastructure (the socio-technical system) including trust in your personal agent and in other mediating agents; trust in the potential partners; trust in the warrantors and authorities (if any). Another growing trend is the use of reputation mechanisms, and in particular the interesting link between trust and reputation. Many computational and theoretical models and approaches to reputation have been developed in the last few years.

Trust appears to be foundational for the notion of "agency" and for its defining relation of acting "on behalf of". It is also critical for modeling and supporting groups and teams, organizations, co-ordination, negotiation, with the related trade-off between individual utility and collective interest; or in modeling distributed knowledge and its circulation. In several cases the electronic medium seems to weaken the usual bonds in social control: and the disposition to cheat grows stronger. In experiments of cooperation supported by computers it has been found that people are more leaning to defeat than in face-to-face interaction, and a preliminary direct acquaintance reduces this effect. So, computer technology can even break trust relationships already held in human organizations and relations, and favor additional problems of deception and trust.

We encourage an interdisciplinary focus of the workshop -although focused on virtual environments and artificial agents- as well as presentations of a wide range of models of deception, fraud, reputation and trust building.

Just to mention some examples: AI models, BDI models, cognitive models, game theory, and organizational science theories. Suggested topics include, but are not restricted to, the following. Here "mechanisms" include considerations of architecture, design, and protocols.

- *Models of trust and of its functions*
- *Models of deception and fraud; approaches for detection and prevention*
- *Models and mechanisms of reputation*
- *Role of control and guaranties mechanisms*
- *Models and mechanisms for privacy and access control*
- *Theoretical aspects, e.g., autonomy, delegation, ownership*
- *Integration of conventional and agent-based mechanisms*
- *Policies, interoperability, protocols, ontologies, and standards*
- *Scalability and distribution across multiple domains or within the global domain*
- *Test-beds and frameworks for computational trust and reputation models*
- *Legal aspects*
- *Application studies (e.g., e-commerce, e-health, e-government)*

WORKSHOP ORGANIZERS

Rino **Falcone** - ISTC-CNR - Italy, (contact person);
Suzanne **Barber** - The University of Texas - USA;
Jordi **Sabater-Mir** - IIIA-CSIC - Spain;
Munindar **Singh** - North Carolina State University – USA

PROGRAM COMMITTEE

Suzanne **Barber** - Computer Science, The University of Texas, USA
Cristiano **Castelfranchi** - Cognitive Science, ISTC National Research Council, Italy
Robert **Demolombe** - Institut de Recherche en Informatique de Toulouse, IRIT, France
Torsten **Eymann** - Department of Information Systems, University of Bayreuth
Rino **Falcone** - Cognitive Science, ISTC National Research Council Italy
Wander **Jager** - Economics, University of Groeningen, The Netherlands
Andrew **Jones** - Department of Computer Science, King's College London, U.K.
Catholijn **Jonker** - Computer Science, Vrije Universiteit Amsterdam, The Netherlands
Churn-Jung **Liau** – Institute of Information Science, Academia Sinica, Taiwan
Stephane **Lo Presti** - Computer Science, University of Southampton, U.K.
Brendan **Neville** - Imperial College, London, U.K.
Mario **Paolucci** - Cognitive Science, ISTC National Research Council, Italy
Jordi **Sabater-Mir** - Computer Science, IIIA-CSIC, Spain
Sandip **Sen** - Computer Science, University of Tulsa, USA
Onn **Shehory** - IBM Haifa Research Lab, Israel
Munindar **Singh** - Computer Science - North Carolina State University, USA
Chris **Snijders** - Sociology, Utrecht University,The Netherlands
Leon **Van der Torre** - Faculty of Sciences, Technology and Communication, University of Luxembourg

TABLE OF CONTENTS:

Graded Trust

Robert Demolombe

Institut de Recherche en Informatique de Toulouse
France
[*] robert.demolombe@orange.fr

Abstract. After a brief analysis of several trust definitions a common pattern is exhibited which takes the form of a truster's belief about the regularity of some trustee's property. That leads to a definition of graded trust in terms of two independent components: graded belief and graded regularities, where grades take qualitative levels. This idea is formalized in the framework of classical modal logics. After an informal discussion of the axiom schemas and inference rules of the selected logic, a formal definition of its proof theory and of its model theory are given. Finally, the main features of this approach are compared with other proposals for the formalization of qualitative graded beliefs, in particular the Spohn's approach.

1 Introduction

There are many definitions of trust, nevertheless most of them agree on the fact that trust is essentially a mental attitude of an agent, the truster, with regard to another agent, the trustee. This attitude involves truster's beliefs, and also, in some definitions, other features like truster's goal.

This concept of trust has been formalized in a quantitative framework, like probabilities [12], or in a qualitative framework, like formal logic [3, 8, 10, 6]. In formal logic most of the proposals deal with non graded frameworks. That is, either the truster trusts, or does not trust, the trustee.

In this paper a new framework for qualitative graded trust is proposed in modal logic. Two guidelines have been followed in designing this logic: the first one is to be compatible with most of the definitions of the concept of trust, thought there is no consensus in this area, and the second one is to be compatible, as far as possible with a quantitative formalization, in particular with probability theory.

In the following sections we start with a brief survey of some of the most well known definitions of trust and a common core is exhibited. In section 3 it is shown that the notion of graded trust involves two components. This requires two independent grades that are called ”graded beliefs” and ”graded regularities”. The following section 4 is devoted to the analysis and definition of an appropriate modal logic; the first sub section is about the proof theory and the second one is about the model theory. In the last section our proposal is compared to related works, and some conclusions are presented.

[*] This work has been partially supported by the French Agence Nationale de la Recherche under contract ANR-06-SETI-006.

2 About trust

In [10] (see also [11]) A.J.I. Jones surveys several definitions of trust. He points out that for some authors, like M. Bacharach, and D. Gambetta [1], trust is a truster's expectation of an action to be performed by the trustee. More specifically for T. Rea [16] this expectation is about the trustee's competence and his fulfilment of all fiduciary obligations. In [9] K. Giffin, add to trust definition the fact that the truster's expectation is related to some truster's objective. Then, after a deep analysis of concrete scenarios Jones concludes that the minimal constituents of the core of trust definition should be defined in terms of truster's beliefs about some regularity and conformity properties satisfied by the trustee, and that truster's goal may also be present in the truster's attitude, but that this is not necessarily the case.

C. Castelfranchi and R. Falcone in [3] offer a different analysis, based on cognitive science, where they argue that the truster's goal is a constitutive part of trust definition, and they integrate other features in their definition, in particular truster's dependence about the trustee.

In [6] (see also [5]) R. Demolombe adopting a simpler trust definition has presented a classification of the different kinds of properties the truster may ascribe to the trustee. This classification is briefly recalled below in order to show that they all share a common formal pattern which can also be found in most of the other definitions, even if these definitions cannot be reduced to these patterns. The classification is defined in terms of epistemic, dynamic and deontic properties. Some examples are presented below.

Sincerity. Agent i trusts agents j about his sincerity about p iff i believes that IF j informs i about p ($Inf_{j,i}p$), THEN j believes p ($Bel_j p$).
In formal terms: $Bel_i(Inf_{j,i}p \Rightarrow Bel_j p)$.

Competence. Agent i trusts agents j about his competence about p iff i believes that IF j believes p ($Bel_j p$), THEN p holds.
In formal terms: $Bel_i(Bel_j p \Rightarrow p)$.

Vigilance. Agent i trusts agents j about his vigilance about p iff i believes that IF p holds, THEN j believes p ($Bel_j p$).
In formal terms: $Bel_i(p \Rightarrow Bel_j p)$.

Cooperativity. Agent i trusts agents j about his cooperativity about p iff i believes that IF j believes p ($Bel_j p$), THEN j informs i about p ($Inf_{j,i}p$).
In formal terms: $Bel_i(Bel_j p \Rightarrow Inf_{j,i}p)$.

Ability. Agent i trusts agents j about his ability to bring it about that p iff i believes that IF j has attempted to bring it about that p ($H_j p$), THEN p holds.
In formal terms: $Bel_i(H_j p \Rightarrow p)$.

Obedience. Agent i trusts agents j about his obedience about the obligation to bring it about that p iff i believes that IF it is obligatory that j brings it about that p ($ObgE_j p$), THEN j brings it about that p ($E_j p$).
In formal terms: $Bel_i(ObgE_j p \Rightarrow E_j p)$.

Honesty. Agent i trusts agents j about his honesty with respect to the permission to bring it about that p iff i believes that IF j brings it about that p ($E_j p$), THEN it is permitted that j brings it about that p ($PermE_j p$).
In formal terms: $Bel_i(E_j p \Rightarrow PermE_j p)$.

In these definitions, formulas of the form: $\phi_j \Rightarrow \psi_j$ can be read: ϕ_j entails ψ_j.

More recently, E. Lorini and R. Demolombe [15] have formalized in modal logic trust definitions which are very close to those proposed in [3]. For instance, they have defined trust in positive action as follows: *i trusts j to do α with regard to his goal that ϕ if and only if i wants ϕ to be true and i believes that:*

1. *j, by doing α, will ensure that ϕ, and*
2. *j has the capacity to do α, and*
3. *j intends to do α*

It can be easily shown that conditions 1 and 2 have a conditional form. For instance, the condition 1 can be rephrased as: ”i believes that if j performs the action α, then ϕ holds”.

At the end of this analysis, our conclusion is that in almost all the trust definitions we find patterns of the form:

$$Bel_i(\phi_j \Rightarrow \psi_j)$$

where $\phi_j \Rightarrow \psi_j$ represents some j’s property that i ascribes to j.

3 Graded trust

In most of realistic situations it is an over simplification to say that a truster i either trusts, or does not trust, a trustee j. Rather, in informal terms, we say, for instance, that i **has a limited trust in** j, or i**’s trust in** j **is high**. Then, we are faced to this **question:** ”*what is the meaning of such sentences?*.

A **first answer** to the question, when trust is represented by a formula of the form $Bel_i(\phi_j \Rightarrow \psi_j)$, is that i is **uncertain** to be in a world where the set of ϕ_j worlds (the set of worlds where ϕ_j is true) is included into the set of ψ_j worlds (the set of worlds where ψ_j is true). For example, i may be uncertain about the fact j is sincere about p, that is, about the fact that in every circumstances where j informs i about p it is the case that j believes p.

Here, graded trust can be defined by the strength level of i’s belief about j’s sincerity. Notice that this ”uncertainty” level refer to the validity of i’s beliefs, not to the completeness of i’s beliefs. In more formal terms graded trust can be represented by the formula: $Bel_i^g(\phi_j \Rightarrow \psi_j)$, and this formula can be read: *the strength level of i’s belief about the fact that $\phi_j \Rightarrow \psi_j$ is true is g*, where Bel_i^g is used to denote a ”*graded belief*”.

A **second answer** may be that i believes that the set of ϕ_j worlds is ”**partially included**” into the set of ψ_j worlds. In that case the fact that i’s trust in j’s sincerity is high can be interpreted as the fact that i believes that in almost all circumstances if j informs i about p, then j believes p.

The ”inclusion level” of the set of ϕ_j worlds into the set of ψ_j worlds is called the ”*regularity level*” of j’s attitude. This level is formally represented by the formula: $\phi_j \Rightarrow^h \psi_j$, and we also say that it represents a graded regularity. In that case graded trusts are represented by formulas of the form: $Bel_i(\phi \Rightarrow^h \psi)$.

Our proposal in this paper is that graded trust refers to **both** answers and that they should be represented by formulas of the form:

$$Bel_i^g(\phi \Rightarrow^h \psi)$$

whose intended meaning is that the strength level of i's belief about the fact that ϕ entails ψ with a regularity level h is g.

4 A modal logic for graded beliefs

We have defined a formal logic for reasoning about graded trust in order to be able to derive the consequences of a set of assumptions that are supposed to represent a particular situation in a given application. This part of the logic is defined by its proof theory. It is complemented by its model theory whose objective is to formalize the meaning of the concepts, and their correspondent modalities. Roughly speaking, the model theory defines the meaning of the fact that a formula of this logic is true in a particular situation which is represented by a formal model.

4.1 Proof theory

The proof theory is presented progressively in order to explain the meaning and the justification of the inference rules and axiom schemas that have been chosen.

First, it is assumed that levels are represented by a finite non empty set of qualitative grades G, and that there is a total order on G represented by the relation: $\leq$. The highest level is denoted by max, and the lowest level is denoted by min.

To represent beliefs we have two modalities. The first one represents beliefs to which an agent has not assigned a strength level, for example, because he has not enough information to fix the grade of this belief. They are called ”standard beliefs”. The second one represents graded beliefs. Their notations and intuitive meanings are:

$Bel_i(\phi)$: i believes that ϕ is true.

$Bel_i^g(\phi)$: the strength level of i's belief about the fact that ϕ is true is (exactly) g.

The logical connective $\Rightarrow^h$ is a conditional in Chellas's sense [4]. As mentioned before it is used to represent graded regularities, and its intuitive meaning is:

$\phi \Rightarrow^h \psi$: ϕ entails ψ at the level h.

Semi formal analysis

For the modality Bel_i we have adopted as usual a KD logic (see [4]).

For the modality Bel_i^g we have the following inference rules and axiom schemas.

(U0) In $Bel_i^g(\phi)$, ϕ can be substituted by any logically equivalent formulas.

(U1) If ψ is a logical consequence of ϕ (i.e. $\vdash \phi \rightarrow \psi$), then if i has ascribed a strength level to his belief about ϕ and to his belief about ψ, then the level of ψ cannot be lower than the level of ϕ.

Notice that this rule does not impose that if i has ascribed a level to ϕ, he has necessarily also ascribed a level to ψ. The reason why we have this cautious rule is that it may be that ψ may contain sub formulas which are not relevant to ϕ. For example, it may be that the meaning of ϕ is that j is sincere, and the meaning of ψ is that j is sincere or k is honest. In that example ψ is a logical consequence of ϕ. However, if i ignores who is k, i may have no opinion about the fact k is honest. Then, i cannot ascribe a level to ψ, although he knows that if he had to fix a level for ψ, it should be greater or equal to the level of ϕ.

It is also worth noting that we do not have the axiom schema: $Bel_i^g(\phi) \rightarrow Bel_i^g(\phi \vee \psi)$, because in most cases, if the level of $\phi \vee \psi$ is fixed, it is greater than g. This observation shows that Bel_i^g is not a normal modality, it is a classical modality.

(U2) If the levels of beliefs of the formulas ϕ_1 and ϕ_2 are fixed, then the level of their disjunction is the maximum of these two levels.

If the levels in graded beliefs would be interpreted in terms of probabilities, we would have that the probability of the disjunction is greater or equal to the maximum of the two probabilities. Then, the choice of (U2) is not compatible with probabilities, but it is as close as possible to probabilities when we are dealing with qualitative levels.

(U3) If the levels of beliefs of the formulas ϕ_1 and ϕ_2 are fixed, then the level of their conjunction is the minimum of these two levels.

The justification of (U3) is similar as the justification of (U2).

(U4) The strength level of i's belief is unique for a given sentence.

(U5) Graded beliefs are consistent with standard beliefs. Notice that $Bel_i^{g_1}\phi$ and $Bel_i^{g_2}\neg\phi$ are consistent in a similar way as $g_1 = Pr(\phi)$ and $g_2 = Pr(\neg\phi)$ are consistent, provided $g_1 + g_2 = 1$.

(U6) If ϕ represents the formula which is believed at the minimum level and ψ is a standard belief, then ϕ implies ψ. [1]

The intuitive idea is that there is no proposition that is believed at a lower level than the proposition that characterizes **all** i's standard beliefs. In some sense this proposition is the most specific one which is believed (in a standard sense) by i.

(U7) If ϕ represents the formula which is believed at the maximum level and ψ is a standard belief, then ψ implies ϕ.

The intuitive idea is that there is no proposition that is believed at a greater level than the level of tautologies.

(U8) If ϕ is believed at the level g, then i believes that ϕ is believed at the level g.

This positive introspection axiom schema means that no level is ascribed by i to his evaluation of the level of a belief. If such a level would be ascribed, then one could ask the question: *what is i's evaluation of this "second" order level?*, and we would be leaded to an infinite number of introspection levels, which is far to be intuitive.

Notice also that there is no justification to ascribe the maximum level to introspection beliefs because the maximum level is restricted to tautologies, while graded beliefs are contingent propositions.

(U9) If ϕ is not believed at the level g, then i believes that ϕ is not believed at the level g.

The justification of (U9) is similar as the justification of (U8).

For the conditional connective $\Rightarrow^h$ we have the following inference rules and axiom schemas.

(R0) In $\phi \Rightarrow^h \psi$, ϕ and ψ can be substituted by logically equivalent formulas.

[1] We would like to thanks the anonymous referee who pointed out an error in the preliminary version of axioms schemas (U6),(U7) and (R2).

(R1) If ϕ entails ψ at the level h, then if ϕ holds, ψ holds "at the level" h.

This axiom schema can be easily understood if we think to its possible interpretation in terms of conditional probabilities. If $\phi \Rightarrow^h \psi$ is interpreted as: $h = Pr(\psi|\phi)$, if we have: $1 = Pr(\phi)$, we can infer that: $h = Pr(\psi)$, which can be rephrased as: $h = Pr(\psi|\top)$, and this formula can be seen as the interpretation of $\top \Rightarrow^h \psi$. That is why, in the following, ψ^h is used as a notation for: $\top \Rightarrow^h \psi$.

If we have $\phi_1 \Rightarrow^{h_1} \psi$ and $\phi_2 \Rightarrow^{h_2} \psi$ and $h_1 \neq h_2$, from ϕ_1 and ϕ_2 we can infer ψ^{h_1} and ψ^{h_2}, which contradicts the further unicity schema (R3). We have the same kind of contradiction with conditional probabilities if we have $h_1 = Pr(\psi|\phi_1)$ and $h_2 = Pr(\psi|\phi_2)$, and ϕ_1 and ϕ_2 are both true; because we get $h_1 = Pr(\psi)$ and $h_2 = Pr(\psi)$.

(R2) There exists a function F such that if $n = F(h_1, k_1, h_2, k_2)$, then, if ϕ entails ψ at the level h_1, ψ entails θ at the level k_1, ϕ entails $\neg\psi$ at the level h_2 and $\neg\psi$ entails θ at the level k_2, then ϕ entails θ at the level n.

The reason why we have this axiom schema is that, in general, from: $((\phi \Rightarrow^{h_1} \psi) \wedge (\psi \Rightarrow^{k_1} \theta)$ we cannot infer what is the value of n such that: $(\phi \Rightarrow^n \theta)$, because there may be ϕ worlds that are θ worlds, and which are not ψ worlds. Notice that axiom schema (R2) is perfectly compatible with conditional probabilities if we accept some uniform distribution assumptions. In that case the form of F is: $n = (h_1 \times k_1) + (h_2 \times k_2)$.

(R3) The regularity level of ϕ entails ψ is unique.

Finally, it is worth noting that we do not have an axiom schema that allows us to infer from: $(\phi \Rightarrow^{h_1} \psi) \wedge (\phi \Rightarrow^{h_2} \theta)$, the value of h_3 such that: $(\phi \Rightarrow^{h_3} \psi \wedge \theta)$. The reason is that, for a given level of h_1 and h_2, the set of ψ worlds and the set of θ worlds may be either disjoint or one may be included into the other one.

Formal definition

The proof theory is formally defined as follows.

The syntax of the language is defined as usual for a multimodal propositional logic (see [4]).

In addition to the inference rules and axiom schemas of classical propositional logic we have the following inference rules and axiom schemas.

Notations.

$Forall(g, cond)F(g) \overset{\text{def}}{=} \bigwedge_{g\in G, cond(g)} F(g)$

$Exist(g, cond)F(g) \overset{\text{def}}{=} \bigvee_{g\in G, cond(g) F(g)}$

$\psi^h \overset{\text{def}}{=} \top \Rightarrow^h \psi$

$Bel_i(\phi)$ obeys a KD system.

(U0) If $\vdash \phi \leftrightarrow \psi$ then $\vdash Bel_i^g(\phi) \leftrightarrow Bel_i^g(\psi)$

(U1) If $\vdash \phi \rightarrow \psi$ then $\vdash Bel_i^g(\phi) \rightarrow \neg Exist(g', g' < g)Bel_i^{g'}\psi$

(U2) If $g_3 = Max\{g_1, g_2\}$ then $\vdash Bel_i^{g_1}(\phi_1) \wedge Bel_i^{g_2}(\phi_2) \rightarrow Bel_i^{g_3}(\phi_1 \vee \phi_2)$

(U3) If $g_3 = Min\{g_1, g_2\}$ then $\vdash Bel_i^{g_1}(\phi_1) \wedge Bel_i^{g_2}(\phi_2) \rightarrow Bel_i^{g_3}(\phi_1 \wedge \phi_2)$

(U4) $\vdash Forall(g_1, g_2, g_1 \neq g_2)\, \neg(Bel_i^{g_1}(\phi) \wedge Bel_i^{g_2}(\phi))$

(U5) $\vdash Bel_i^g\phi \rightarrow \neg Bel_i \neg\phi$

(U6) $\vdash (Bel_i^{min}\phi \wedge Bel_i\psi) \rightarrow (\phi \rightarrow \psi)$

(U7) $\vdash (Bel_i^{max}\phi \wedge Bel_i\psi) \rightarrow (\psi \rightarrow \phi)$

(U8) $\vdash Bel_i^g(\phi) \rightarrow Bel_i Bel_i^g(\phi)$
(U9) $\vdash \neg Bel_i^g(\phi) \rightarrow Bel_i \neg Bel_i^g(\phi)$
(R0) If $\vdash \phi \leftrightarrow \phi'$ and $\vdash \psi \leftrightarrow \psi'$ then $\vdash (\phi \Rightarrow^h \psi) \rightarrow (\phi' \Rightarrow^h \psi')$
(R1) $\vdash (\phi \Rightarrow^h \psi) \rightarrow (\phi \rightarrow \psi^h)$
(R2) There exists a function F such that if $n = F(h_1, k_1, h_2, k_2)$, then
$\vdash ((\phi \Rightarrow^{h_1} \psi) \wedge (\psi \Rightarrow^{k_1} \theta) \wedge (\phi \Rightarrow^{h_2} \neg\psi) \wedge (\neg\psi \Rightarrow^{k_2} \theta)) \rightarrow (\phi \Rightarrow^n \theta)$
(R3) $\vdash Forall(h_1, h_2, h_1 \neq h_2)\ \neg((\phi \Rightarrow^{h_1} \psi) \wedge (\phi \Rightarrow^{h_2} \psi))$

4.2 Model theory

The model theory gives a formal semantics to the concepts of standard beliefs, graded beliefs and graded regularities. Models are a particular sort of minimal conditional model as defined by Chellas (see [4], section 10.1). They are defined as a tuple M such that:

$$M =< W, \{B_i\}, \{B_i^g\}, \{R^h\}, v >$$

In M, W is a set of possible words, $\{B_i\}$ is a set of functions: $B_i : W \rightarrow 2^W$, which assign to each world a set of worlds, $\{B_i^g\}$ is a set of functions: $B_i^g : W \rightarrow 2^{2^W}$, which assign to each world a set of sets of worlds, $\{R^h\}$ is a set of functions: R^g : $W, 2^W \rightarrow 2^{2^W}$, which assign to each pair formed with a world and a set of worlds, a set of sets of worlds, and v is a function: $v : ATOM \rightarrow 2^W$, which assigns to each atomic formula a set of worlds.

In this kind of models a proposition and the set of worlds where this proposition is true are identified.

The intuitive meaning of these functions can be seen through formal examples.

If $B_i(w) = X$, X is the set of worlds consistent with **ALL** the propositions believed by i in w. This set of worlds is usually characterized by an accessibility relation in models of normal modal logics.

If $B_i^g(w) = \{X_1, X_2\}$, the set of propositions believed by i in w at the level g is represented by the set of sets of worlds: X_1 and X_2. That means that the set of propositions believed at the level g is represented by these two sets.

If $R^h(w, X) = \{X_2, X_4\}$, in w the set of propositions entailed at the level h by the proposition represented by X is represented by the set of sets of worlds X_2 and X_4. These two sets can also be interpreted as two propositions.

Satisfiability conditions

For reasons that are explained in the comments about the satisfiability conditions for graded regularities, the truth value of a formula is defined in the context of a set of worlds X.

$M, X, w \models \phi$ can be read: ϕ is true in the world w, in the context X and in the model M.

This notion of truth relativized to a context is related to the standard notion of truth by the following condition.

$M, w \models \phi$ iff $M, W, w \models \phi$.

Let us assume that X is a subset of W.

Notation: $|\phi|_X \overset{\text{def}}{=} \{w_1 \ : \ w_1 \in X \ and \ M, X, w_1 \models \phi\}$.

$M, X, w \models atom$ iff $w \in v(atom)$ and $atom$ is an atomic formula.
$M, X, w \models \neg\phi$ iff $M, X, w \not\models \phi$.
$M, X, w \models \phi \vee \psi$ iff $M, X, w \models \phi$ or $M, X, w \models \psi$.
$M, X, w \models Bel_i\phi$ iff $\exists Y(Y = B_i(w)\ and\ \forall w'(w' \in Y\ \Rightarrow M, Y, w' \models \phi))$.
$M, X, w \models Bel_i^g\phi$ iff $\exists Y(Y = B_i(w)\ and\ |\phi|_Y \in B_i^g(w))$.
$M, X, w \models \phi \Rightarrow^h \psi$ iff $|\psi|_X \in R^h(w, |\phi|_X)$.

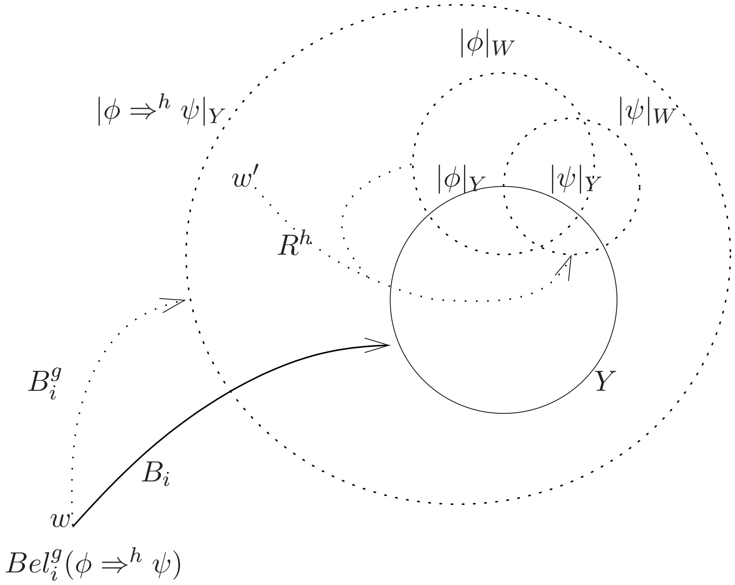


Fig. 1. Evaluation of a graded belief about a graded regularities.

Example. $M, X, w \models Bel_i^g(\phi \Rightarrow^h \psi)$ iff $\exists Y(Y = B_i(w)\ and\ |\phi \Rightarrow^h \psi|_Y \in B_i^g(w))$. From $|\phi|_X$ definition we have: $|\phi \Rightarrow^h \psi|_Y = \{w' : w' \in Y\ and\ M, Y, w' \models \phi \Rightarrow^h \psi\}$. From the satisfiabilities conditions we have: $M, Y, w' \models \phi \Rightarrow^h \psi$ iff $|\psi|_Y \in R^h(w', |\phi|_Y)$.

This example shows why formulas are evaluated with respect to a given context. Indeed, to evaluate to what extend i believes that the set of ϕ worlds is included into the set of ψ worlds, we have to restrict ϕ extension and ψ extension to the set of worlds which are consistent with all i's beliefs in w, i.e. to the set of worlds Y, which is $B_i(w)$ (see figure 1). That is the reason why formulas are evaluated with respect to a given context. If a formula is not in the scope of some agent's beliefs, then the context is not restricted, i.e. the context is W.

Notice that, since graded beliefs are standard beliefs, the set of worlds $|\phi \Rightarrow^h \psi|_Y$ contains $B_i(w)$.

5 Related works

In [17] W. Spohn as defined a framework to represent graded beliefs in order to give a more satisfying account of rational epistemic changes. His final goal is to raise deterministic epistemology to a level as satisfying as probabilistic conditionalization in the field of non deterministic changes.

The framework, like in this paper, is defined by a set of possible worlds, and propositions are also identified to sets of worlds. A set of beliefs is represented by a set of worlds, called the "net content", the set of worlds which is included into all the believed propositions (in our framework this set of worlds is $B_i(w)$). The first idea is to represent belief changes with simple conditional functions (SCF) which collect all the possible changes of the net content of epistemic states brought about by all possible information. Then, a SCF g is a function from 2^W to 2^W. Spohn shows that the information represented by the SCFs can be represented by a well ordered partition (WOP) of W, where a WOP is a partition such that ordinals 0,...,n are assigned to each member of the partition. These ordinals are intended to represent the strength of **disbelief** of propositions represented by each partition. These members are denoted by E_0,...,E_n. The partition E_0 is the least disbelieved proposition and it represents the net content of an epistemic state.

According to these definitions a WOP represents a SCF iff for all non empty set of worlds A we have $g(A) = E_\beta \cap A$, where E_β is the least disbelieved partition that intersects A. On the basis of this correspondence between WOPs and SCFs it is shown that no SCF can appropriately represents epistemic changes, in the sense that it is possible to get the same epistemic change after getting and removing an information A, and that getting information A and then B leads to the same epistemic state as getting B and then A.

That is the reason why Spohn introduces the ordinal conditional functions (OCF). An OCF k is defined on a complete field of propositions and assigns to each non empty proposition an ordinal such that 0 is not obtained from an empty set, and the ordinal $k(w)$ is the same for all the worlds w in the same atomic proposition. Then, for any non empty proposition A, the ordinal $k(A)$ characterizes the least disbelieved world in A, i.e. $k(A) = min\{k(w) \ : \ w \in A\}$. Notice that $k(A) = 0$ means that A is not believed to be false, and we may have both $k(A) = 0$ and $k(\neg A) = 0$.

Finally, to have the properties of reversibility and commutativity of epistemic changes a complementary parameter α is added to complement the values of k. The grade α characterizes the strength of $\neg A$. It is used to increment the value of $k(\neg A)$ after getting the information A. We have no room here to explain in detail how this parameter is defined and how it is used.

There are some commonalities with graded beliefs that have been presented here in the sense that qualitative grades are assigned to beliefs. For example, $Bel_i^{min} A$ and $k(A) = 0$ have similar meanings. We also have $k(A \cup B) = min\{k(A), k(B)\}$ which is very to close to our axiom schema (U2). However, there are significant differences. The first one is that in our framework there is no need to assign a grade to all the propositions. The second one is that the meaning of the grades are different: in our framework they represent the strength of beliefs, while for Spohn they represent the strength of disbeliefs. Is there a one to one correspondence between each of them? The answer is far to be obvious. The third one is that to define the OCFs we have to assign grades to all the worlds. We think that in a non trivial application domain where a world is defined by tens of atomic propositions, it is quite difficult to consider each world and to evaluate the appropriate grade for this world. May be a trick could be to ”cluster” sets of worlds, and to assign to them the same grade, but that is exactly what we do if these sets are seen as propositions. The difference being that we do not request a ”complete” assignment. Finally, in Spohn proposal there is no proof theory.

Several authors have taken inspiration in Spohn's proposal in the perspective of modeling belief changes (see, for example, C. Boutilier [2]). In [14] N. Laverny and J. Lang have explicitly integrated these ideas in a modal logical framework. They define a modality $B^i\phi$ whose intuitive meaning is that ”the agent believes ϕ with strength i” [2] and whose satisfiability condition for a given OCF k is: $k \models B^i\phi$ iff $i \leq k(\neg\phi)$.

N. Laverny in [13] has defined a normal modal logic for graded beliefs where modalities $B^i\phi$ obey a $KD45$ system called $KD45_G$. The positive introspection (negative introspection is similar) axiom schema takes the form: $B^j\phi \rightarrow B^iB^j\phi$ which, in our view is questionable, as mentioned in section 4.1. The author shows how these modalities can be ”translated” into the OCF framework. For a given OCF k we have: $k, s \models B^i\phi$ iff $\forall s'(k(s') < i \Rightarrow k, s' \models \phi)$.

In [15] E. Lorini and R. Demolombe have defined a normal modal logic for graded beliefs where $Bel^{\geq x}\phi$ can be read ”agent i believes ϕ at least with strength x”. These modalities are interpreted by binary relations P_i^x, and $P_i^x(w)$ denotes the set of worlds accessible from the world w. These relations are structured by the constraint: if $y < x$ then $P_i^y(w) \subseteq P_i^x(w)$, which can be seen as a structure of spheres. From these modalities are defined modalities $Bel^x\phi$ whose meaning is that agent i believes ϕ at strength x. The satisfiability conditions for these modalities can be expressed as: $M, w \models Bel^x\phi$ iff $P_i^x(w) \subseteq |\phi|_W$ and $P_i^{suc(x)}(w) \not\subseteq |\phi|_W$. The correspondence with Spohn's OCF is defined by a translation of the set of spheres into an EOP. The significant point of this works is that these graded beliefs are integrated into a logical framework that defines different kinds of trust.

R. Demolombe and C. J. Liau in [7] have defined graded beliefs and graded trust in order to propose a solution to belief revision. They define modalities $B_i^\alpha\phi$ whose meaning is that agent i believes ϕ at the level α. These modalities are normal modalities. They also define classical modalities of the form $TV_{i,j}^\alpha\phi$ and $TC_{i,j}^\alpha\phi$, whose meaning are that agent i trusts agent j at the level α for being a valid (respectively complete) information source for ϕ. Here, a valid information source is an information source who is both sincere and competent, and a complete information source is defined in a dual way. The meaning of these graded trust definitions can be well understood with the axiom schemas: $TV_{i,j}^\alpha\phi \rightarrow (K_iInf_{j,i}\phi \rightarrow B_i^\alpha\phi)$, and $TC_{i,j}^\alpha\phi \rightarrow (K_i\neg Inf_{j,i}\phi \rightarrow B_i^\alpha\neg\phi)$.

We have seen that most of the works dealing with graded beliefs have been done to formalize belief change. The few ones which consider graded beliefs for modeling graded trust identify the level of beliefs and the level of trust. The most significant difference with what we have proposed is the fact that we have considered that two

[2] In fact the meaning of this modality is that the agent believes ϕ with strength at least equal to i.

independent grades are involved in trust definition. The following example is intended to show why we need graded trust and graded regularities.

Let us assume that agent i has a low belief strength g about the fact that j is very competent with regard to p, because on the basis of 5 observations where j has believed p, in 4 situations it was the case that p was true. The strength of i's belief is low because the number of observations is quite limited. If after a greater number of observations, say 5 additional observations, it is confirmed that j is very competent, the i's belief strength g will be greater while the grade h of j's competence remain the same. However, if for these 5 new observations it happens that j's belief was wrong in 3 cases, i will believe that j has a moderate competence, that is that h is lower. Even if the grades g and h are not necessarily assigned on the basis of observations, from this simple example we can understand why we need two different grades to evaluate i's trust about j's competence.

6 Conclusion

A logical framework has been defined to represent graded trust in terms of two independent components: graded beliefs and graded regularities. We do not pretend that trust can be reduced to a set of components of this kind, but we have shown that they are included in most of the trust definitions.

A class of non normal modalities formalize graded beliefs, while standard beliefs obey a normal modal system. Graded regularities are also represented by non normal modalities. The model theory for these operators is defined in the framework of minimal conditional models ”a la Chellas”. It is possible not to ascribe a grade to all the standard beliefs. For example, in the financial domain, it may be that i believes that the trader j is competent, but i does not have enough background in the domain to assign a grade to j's competence.

We have accepted a framework which has limited logical properties but offers more flexibility for further specializations. For example, it is not imposed to assign a grade to every believed proposition, as it is the case in Spohn's framework. If for some specific reason one would like to impose such constraint, it would not be a great difficulty to add correspondent constraint in the logic.

In the future we want to analyze to what extend the framework could be adapted to a quantitative analysis in terms of probabilities. A possible direction to be investigated would be to interpret graded beliefs as subjective probabilities, and graded regularities as objective conditional probabilities. For example, sentences of the form: $Bel_i^g(\phi \Rightarrow^h \psi)$, could be interpreted as: $Bel_i Pr_{sub}(Pr_{obj}(\psi|\phi) = h) = g$.

Another direction for future works is to go deeper in the analysis of mathematical properties of the logical framework. In particular the constraints to be imposed to the models in order to validate the axiom schemas must be analyzed carefully. For example, (U2) is valid if we impose the constraint: (CU2) If $X_1 \in B_i^{g_1}(w)$, $X_2 \in B_i^{g_2}(w)$ and $g_3 = Max\{g_1, g_2\}$, then $X_1 \cup X_2 \in B_i^{g_3}(w)$, and (U5) is valid if we impose the constraint: (CU5) If $X \in B_i^g(w)$, then $B_i(w) \cap X \neq \emptyset$.

Finally, we believe that the constraint to have a total order on the set of grades could be easily relaxed if that is required in a particular domain.

References

1. M. Bacharach and D. Gambetta. Trust as type detection. In C. Castelfranchi and Y-H. Tan, editors, *Trust and Deception in Virtual Societies*. Kluwer Academic Publisher, 2001.
2. C. Boutilier. *Conditional Logics for Default Reasoning and Belief Revision*. PhD thesis, University of Toronto, 1992.
3. C. Castelfranchi and R. Falcone. Social trust: a cognitive approach. In C. Castelfranchi and Y-H. Tan, editors, *Trust and Deception in Virtual Societies*. Kluwer Academic Publisher, 2001.
4. B. F. Chellas. *Modal Logic: An introduction*. Cambridge University Press, 1988.
5. R. Demolombe. To trust information sources: a proposal for a modal logical framework. In C. Castelfranchi and Y-H. Tan, editors, *Trust and Deception in Virtual Societies*. Kluwer Academic Publisher, 2001.
6. R. Demolombe. Reasoning about trust: a formal logical framework. In C. Jensen, S. Poslad, and T. Dimitrakos, editors, *Trust management: Second International Conference iTrust (LNCS 2995)*. Springer Verlag, 2004.
7. R. Demolombe and C-J. Liau. A logic of graded trust and belief fusion. In C. Castelfranci and R. Falcone, editors, *Proc. of 4th Workshop on Deception, Fraud and Trust*, 2001.
8. R. Falcone and C. Castelfranchi. Trust dynamics: How trust is influenced by direct experiences and by trust itself. In *Proceedings of the 3rd International Conference on Autonomous Agents and Multi-Agent Systems (AAMAS-04)*, pages 740–747. New York, ACM, 2004.
9. K. Giffin. The contribution of studies of source credibility to a theory of interpersonal trust in the communication process. *Psychological Bulletin*, 62:104–120, 1967.
10. A.J.I. Jones. On the concept of trust. *Decision Support Systems*, 33, 2002.
11. A.J.I. Jones and B.S. Firozabadi. On the characterisation of a trusting agent. Aspects of a formal approach. In C. Castelfranchi and Y-H. Tan, editors, *Trust and Deception in Virtual Societies*. Kluwer Academic Publisher, 2001.
12. R. Kohlas, J. Jonczy, and R. Haenni. A trust evaluation method based on logic and probability theory. In Y. Karabulut, J. Mitchell, P. Herrmann, and C. D. Jensen, editors, *IFIPTM'08, 2nd Joint iTrust and PST Conferences on Privacy Trust Management and Security*, volume II of *Trust Management*, pages 17–32, Trondheim, Norway.
13. N. Laverny. *Raisonnement sur les actions et les observations, et programmes à base de croyances graduelles*. PhD thesis, Univeristé Paul Sabatier, Toulouse, 2006.
14. N. Laverny and J. Lang. From knowledge-based programs to graded belief-based programs part ii: Off-line reasoning. In *Proceedings of the International Joint Conference on Artificial Intelligence*, 2005.
15. E. Lorini and R. Demolombe. From binary trust to graded trust in information sources: a logical perspective. In R. Falcone, S. Barber, J. Sabater-Mir, and M. Singh, editors, *Proceedings of the Workshop Trust in Agent Societies*. Springer, To appear.
16. T. Rea. Engendring trust in electronic environments - roles of a trusted third party. In C. Castelfranchi and Y-H. Tan, editors, *Trust and Deception in Virtual Societies*. Kluwer Academic Publisher, 2001.
17. W. Spohn. Ordinal conditional functions: A dynamic theory of epistemic states. In W. L. Harper and B. Skyrms, editors, *Causation in Decision, Belief Change and Statistics*, pages 105–134. Springer, 1988.

From Dependence Networks to Trust Networks

Rino Falcone and Cristiano Castelfranchi

National Research Council– Institute of Cognitive Sciences and Technologies
Via San Martino della Battaglia, 44 00185 - Roma, Italy
{rino.falcone, cristiano.castelfranchi}@istc.cnr.it

Abstract. Dependence networks among agents (describing how each agent can be linked with each other able to satisfy any need/goal/desire) are really important for evaluating future collaborations among them. In fact, the Dependence Networks alone are not sufficient for a real allocation of tasks among the agents. For this allocation, it is also necessary that each agent could satisfy his own expectations about the trustworthiness of the other agents with respect to the specific tasks. We present in this paper a cognitive theory of trust as a capital, which is, in our view, a good starting point to include this concept in the issue of negotiation power. That is to say that if somebody is (potentially) strongly useful to other agents (in the sense that are declared a set of its skills), but it is not trusted, its negotiation power does not improve. Our claim is to underline how (for a set of agents linked to each other) the competitive advantage is not simply of being part of a network, but more precisely of being trusted in that network.

Keywords: Trust, Dependence Networks, Relational Capital.

1 Introduction

In almost the present approaches to the trust study the focus of the analysis is on the trustier and on the ways for evaluating the trustworthiness of other possible trustees. But trust can be viewed at the same time as an instrument both for an agent selecting the right partners in order to achieve its own goals (the trustier's point of view), and for an agent of being selected from other potential partners (the point of view of the trustee) in order to establish with them a cooperation/collaboration and to take advantage from the credit of the accumulated trust.
In this paper we will analyze trust as the agents' *relational capital*. Starting from the classical dependence network (in which needs, goals, abilities and resources are distributed among the agents) with potential partners, we introduce the analysis of what it means for an agent to be trusted and how this condition could be strategically used from him for achieving his own goals, that is, why it represents a form of power. We address this point, analyzing what it means that trust represents a strategic resource for agents that are trusted, proposing a model of 'trust as a capital' for individuals and suggesting the implication for strategic action that can be performed.

Our thesis is that to be trusted:

i) increases the chance to be requested or accepted as a partner for exchange or cooperation;

ii) improves the 'price', the contract that the trustee can obtain.

The need of this new point of view derives directly from the fact that in human societies as well as in multi-agent systems it is strategically important not only to know who is trusted by whom and how much, but also to understand how being trusted can be used by several potential trustiers.

It has been already shown that using different levels of trust represents an advantage in performing some task such as allocating task or choosing between partners. Therefore, having "trust" as a cognitive parameter in agents' decision making can lead to better (more efficient, faster etc.) solutions than proceeding driven by other kind of calculation such as probabilistic or statistical one.

In order to improve this approach and to better understand dynamics of social networks, now we propose a study of what happens on the other side of the two-way trust relationship, focusing on the trustee, in particular on a cognitive trustee. Our aim is an analytical study of what it means to be trusted. The idea of taking the other point of view is particularly important if we consider the judge amount of studies in social science that connect trust with social capital related issues. This work develops and refines the thesis claimed in our previous work (1).
Our claims are:

- to be trusted usually is an advantage for the trustee (agent *Y*); more precisely received trust is a capital that can be invested, and that requires decision and costs to be cumulated;

- it is possible to measure this capital, which is relational, that is depends on a position in a network of relationships;

- trust has different sources: from personal experience that the other agents have had with *Y*; from circulating reputation of *Y*; from *Y's* belongingness to certain groups or categories; from the signs and the impressions that *Y* is able to produce;

- the value of this capital is context dependent (and market dependent) and dynamic;

- received trust strongly affects the 'negotiation power' of *Y* that cannot simply be derived from the "dependence bilateral relationships".

Although there is a big interest in literature about 'social capital' and its powerful effects on the wellbeing of both societies and individuals (7, 8), often it is not clear enough what is it the object under analysis, even if some attempts in this direction were been made (2,3). To overcome this gap, we propose a study that first attempts to understand what trust is as capital of individuals. How is it possible to say that "trust" is a capital? How is this capital built, managed and saved? Then we aim to analytically study the cognitive dynamics of this object, with a particular focus on how they depend on beliefs and goals.

2 Being Trusted

The theory of trust and the theory of dependence are not independent from each other. Not only because – as we modelled (4, 5, 21), before deciding to actively trust somebody, to rely on him (*Y*), one (*X*) has to be dependent on *Y*: *X* needs an action or a resource of *Y* (at least *X* has to believe so). But also because *objective* dependence relationships (10, 11, 12) that are the basis of adaptive social interactions, are not enough for predicting them. *Subjective* dependence is needed (that is, the dependence relationships that the agents know or at least believe), but is not sufficient; it is also necessary to add two relevant beliefs:
(i) the belief of being dependent, of needing the other;
(ii) the belief of the trustworthiness of the other, of the possibility of counting upon him.
If *X* would not feel dependent on *Y*, she could not rely on him.

2.1 Objective and Subjective Dependence

The theory of dependence includes in fact two types of dependences:
(1) the *objective dependence*, which says who needs whom for what in a given society (although perhaps ignoring this). This dependence has already the power of establishing certain asymmetric relationships in a potential market, and it determines the actual success or failure of the reliance and transaction (see Figure 1).

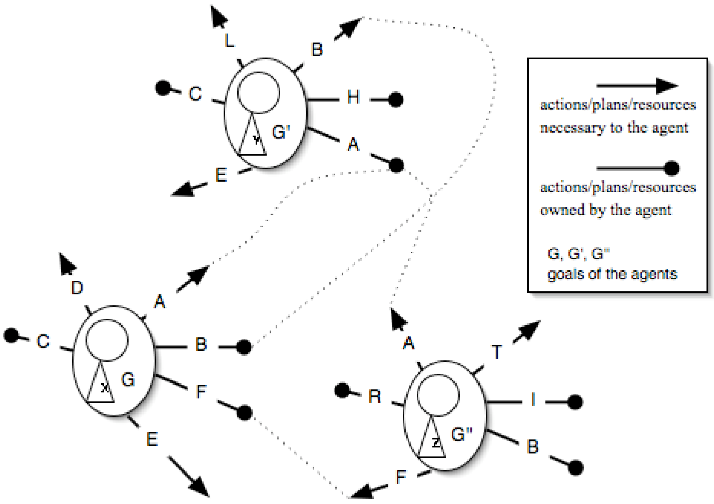

L
B
C
H
G'
Y
A
E
D
A
C
B
G
X
F
E
A
T
R
I
G''
Z
B
F
actions/plans/resources necessary to the agent
actions/plans/resources owned by the agent
G, G', G''
goals of the agents

Figure 1: objective dependence network

(2) the *subjective (believed) dependence*, which says who is believed to be needed by who. This dependence is what determines relationships in a real market and settles on

the negotiation power; but it might be illusory and wrong, and one might rely upon unable agents, while even being autonomously able to do as needed. In Figures 2 is shown the dependence relationships as believed by X: it is different from the objective dependence showed in Figure 1.

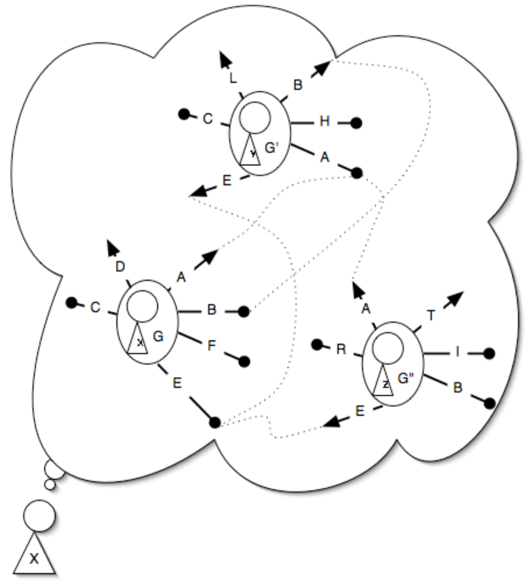


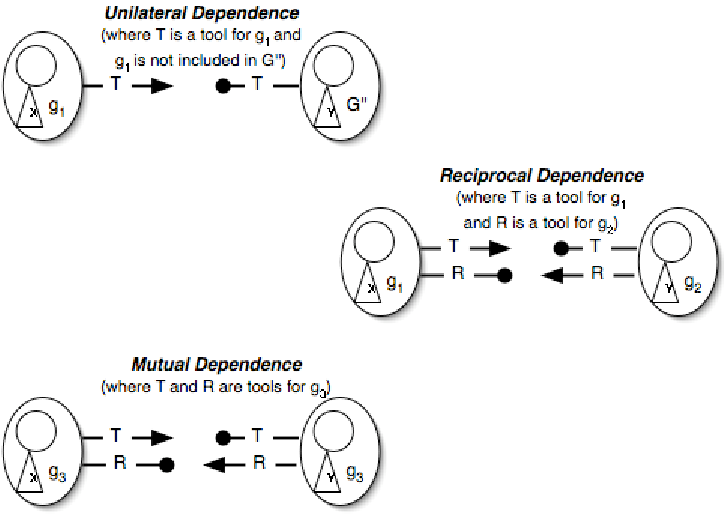
Figure 2: subjective dependence network (believed by *X*)

More Formally, let $Agt=\{Ag_1,..,Ag_n\}$ a set of *agents*; we can associate to each agent $Ag_i \in Agt$:
- a set of *goals* $G_i=\{g_{i_1},..,g_{i_q}\}$;
- a set of *actions* $Az_i=\{\alpha_{i_1},.., \alpha_{i_z}\}$; these are the elementary actions that Ag_i is able to perform;
- a set of plans $\Pi =\{p_{i_1},..,p_{i_s}\}$; the Ag_i's plan library: the set of rules/prescriptions for aggregating the actions; and
- a set of *resources* $R_i=\{r_{i_1},..,r_{i_m}\}$.

The achievement/maintenance of each goal needs of actions/plans/resources.
Then, we can define the *dependence relationship* between two agents (Ag_j and Ag_i) with respect a goal g_{jk}, as *Obj-Dependence* (Ag_j, Ag_i, g_{jk}) and say that:
An agent Ag_j *has an Objective Dependence Relationship with agent* Ag_i *with respect to a goal* g_{jk} *if for achieving* g_{jk} *are necessary actions, plans and/or resources that are owned by* Ag_i *and not owned by* Ag_j.
More in general, Ag_j *has an Objective Dependence Relationship with* Ag_i *if for achieving at least one of its goals* $g_{jk} \in G_j$, *are necessary actions, plans and/or*

resources that are owned by Ag_i and not owned by Ag_j (or, that is the same, they are owned by Ag_j but not usable by it for several reasons).

As in (10) we can introduce the *unilateral*, *reciprocal*, *mutual* and *indirect* dependence (see Figure 3). In very short and simplified terms, we can say that the difference between reciprocal and mutual is that the first is on different goals while the second is on the same goal.



Unilateral Dependence
(where T is a tool for g1 and g1 is not included in G")
T
T
x
g1
y
G"
Reciprocal Dependence
(where T is a tool for g1 and R is a tool for g2)
T
R
T
R
x
g1
y
g2
Mutual Dependence
(where T and R are tools for g3)
T
R
T
R
x
g3
y
g3

Figure 3

The thing really operative in the resulting interactions among the agents is due to their beliefs about the reciprocal dependences rather than the objective dependences.
We call *Subj-Dependence(Ag_j, Ag_i, g_{jk})* for representing the Ag_j's point of view with respect its dependence relationships with Ag_i about its k-th goal g_{jk}. Analogously, we call *Obj-Dependence(Ag_j, Ag_i, g_{jk})* for representing the objective dependence relationship of Ag_j with Ag_i about its k-th goal g_{jk}.
We define *Dependence-Network(Agt,t)* the set of dependence relationships (both subjective and objective) among the agents included in *Agt* set at the time *t*:
Dependence-Network(Agt,t) = Obj-Dependence(Ag_j, Ag_i, g_{jk}) ∪ Subj-Dependence(Ag_j, Ag_i, g_{jk}) with Ag_j ,$Ag_j \in Agt$.

2.2 Power of Negotiation in the Dependence Networks

Given a *Dependence-Network(Agt,t)*, we define
Objective Potential for Negotiation of $Ag_j \in Agt$ about an its own goal g_{jk} -and call it *OPN(Ag_j, g_{jk})*- the following function:

$$OPN(Ag_j, g_{jk}) = f(\sum_{i=1}^{l} \frac{1}{1+p_{ki}}) = \sum_{i=1}^{l} \frac{1}{1+p_{ki}}$$

Where:
- f is in general a function that preserves monotonicity (we will omit this kind of functions in the next formulas);
- l represents the number of agents in the set *Agt* that have a objective dependence relation with Ag_j with respect to g_{jk};
- p_{ki} is the number of agents in *Agt* that are objectively requiring the same actions/plans/resources (as useful for g_{jk}) to Ag_i on which is based the dependence relation between Ag_j and Ag_i and that in consequence are competitors with Ag_j on that actions/plans/resources in a not compatible way (Ag_i is not able to satisfy at the same time all the agents: there is a saturation effect). See Figure 4 for an example.

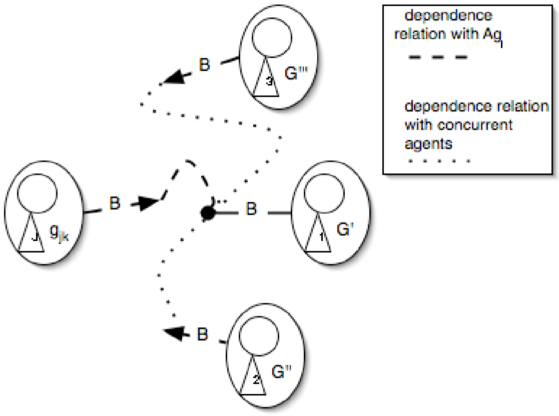

dependence relation with Ag$_i$
- - -
dependence relation with concurrent agents
· · · · ·
B
G'''
3
B
J
g$_{jk}$
B
G'
1
B
G''
2

Figure 4

So, if there are no competitors with Ag_j ($p_{ki}=0$ for each $i \in \{1,..., l\}$) we have:

$$OPN(Ag_j, g_{jk}) = f(\sum_{i=1}^{l} \frac{1}{1+p_{ki}}) = l$$

In general, we can represent the objective dependence of Ag_j as shown in Figure 5: *set1* represents the set of agents who depend from Ag_j for something (actions, plans, resources), *set2* represents the set of agents from which Ag_j depends for achieving an own specific goal g_{jk}. The intersection between *set1* and *set2* (part *set3*) is the set of agents with whom Ag_j could potentially negotiate for achieving g_{jk}. The greater the overlap the greater the *negotiation power* of Ag_j in that context.

However, the negotiation power of Ag_j also depends on the possible alternatives that its potential partners have: the few alternatives to Ag_j they have, the greater its negotiation power (see Figure 4).

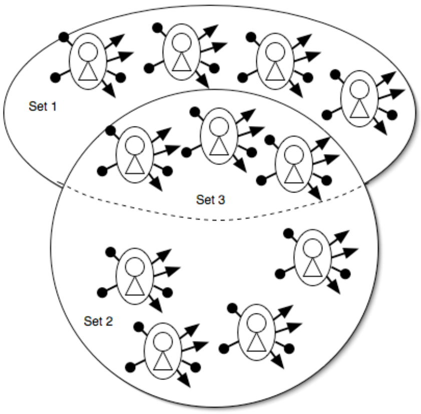


Figure 5

We can define the *Subjective Potential for Negotiation* of $Ag_j \in Agt$ about an its own goal g_{jk} -and call it *SPN(*Ag_j*,* g_{jk}*)*- the following function:

$$SPN(Ag_j, g_{jk}) = \sum_{i=1}^{l^{Bj}} \frac{1}{1 + p_{ki}^{\ Bj}}$$

where the apex *Bj* means "believed by Ag_j"; in fact in this new formula Ag_j both believes the number of potential collaborative agents (*l*) and the number of competitors (p_{ki}) for each of them.

It is clear how, on the basis of these parameters (l^{Bj} and $p_{ki}{}^{Bj}$), the negotiation power of Ag_j is determined. And, at the same time, will be strongly influenced his own decisions.

2.3 Trust Role in Dependence Networks

We are interested now to introduce into the dependence network also the trust relationships. In fact, the dependence network alone is not sufficient for a real allocation of tasks among the agents. It is true that Ag_i should be able and willing to realize the action α_k: But how? And, it will be sufficient given Ag_i's expectations? Would be it more or less trustworthy than Ag_z? For answering to these questions the agents in the dependence network have to establish among them also the reciprocal trust about the different tasks they can allocate to each other.

Indeed, *although it is important to consider dependence relationship between agents in a society, there will be not exchange in the market if there is not trust to enforce these connections*. Considering the analogy with the Figure 4, we will have now a representation as given in Figure 6 (where *Set 4* includes the set of agents that Ag_j considers trustworthy for achieving g_{jk}).

We have now a new subset (darked agents in Figure 7) containing the potential agents for negotiation. Introducing in the *Subjective Potential for Negotiation* (of $Ag_j \in Agt$

about an its own goal g_{jk}) also the basic beliefs about trust (we introduce the superscript index *T* for differentiate from the *SPN* without trust), we have:

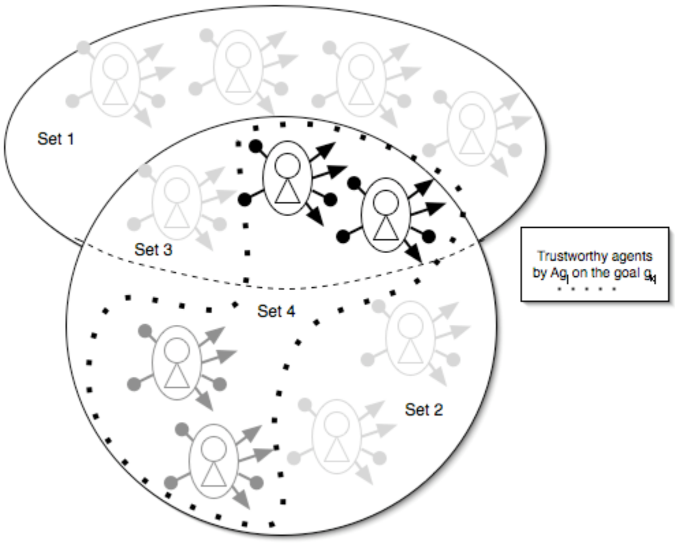


Figure 6

$$SPN^T(Ag_j, g_{jk}) = \sum_{i=1}^{l^{Bj}} \frac{DoA_{ik}^{\ Bj} * DoW_{ik}^{\ Bj}}{1 + p_{ki}^{\ Bj}}) \qquad 1 \geq DoA, DoW \geq 0$$

where $DoA_{ik}^{\ Bj}$ and $DoW_{ik}^{\ Bj}$ are, respectively, the degree of ability and willingness (with respect the goal g_{jk}) of the agent Ag_i as believed by Ag_j.

Analogously, but less relevant in this case, we can introduce the *Objective Potential for Negotiation* (of $Ag_j \in Agt$ about an its own goal g_{jk}), we have:

$$OPN^T(Ag_j, g_{jk}) = \sum_{i=1}^{l} \frac{DoA_{ik} * DoW_{ik}}{1 + p_{ki}}$$

When a cognitive agent trusts another cognitive agent, we talk about social trust.

We consider that the set of actions, plans and resources owned/available by an agent that can be useful for achieving a set of tasks ($\tau_1, \ldots, \tau_r$).

We take now the point of view of the trustee in the dependence network: so we present a cognitive theory of trust as a capital. That is to say that if somebody is (potentially) strongly useful to other agents, but it is not trusted, its negotiation power is not good.

As showed in (3, 9) for delegating a task we have to introduce a *Degree of Trust* of the Agent Ag_j on the agent Ag_i about the task τ_k ($DoT(Ag_j\ Ag_i\ \tau_k)$):

$$DoT(Ag_j,Ag_i,\tau_k)^{Bj} = DoA_{ik}{}^{Bj} * DoW_{ik}{}^{Bj}$$

Where the apex *Bj* means "as believed by Ag_j". At the same way we can also define the *self-trust* of the agent Ag_i about the task τ_k:

$$ST(Ag_i,\tau_k) = DoA_{ik}{}^{Bi} * DoW_{ik}{}^{Bi}$$

We call the *Objective Trust Capital* of $Ag_i \in Agt$ about a potential delegable task τ_k:

$$OTC(Ag_i,\tau_k) = \sum_{j=1}^{l} DoA_{ik}{}^{Bj} * DoW_{ik}{}^{Bj} = \sum_{j=1}^{l} DoT(Ag_j,Ag_i,\tau_k)^{Bj}$$

Where *l* is the number of agents (included in the dependence network) needed for the task τ_k. Note that we are calling as objective trust capital the sum of the trustworthiness that the other agents in the DN attribute to Ag_i rather than the capital Ag_i could deserve on the basis of his own objective relationships: in other words, in it is included the partial (subjective) point of views of the other agents.

We call the *Subjective Trust Capital* of $Ag_i \in Agt$ about a potential delegable task τ_k the function:

$$STC(Ag_i,\tau_k) = \sum_{j=1}^{l^{Bi}} DoA_{ik}{}^{BiBj} * DoW_{ik}{}^{BiBj} = \sum_{j=1}^{l^{Bi}} DoT(Ag_j,Ag_i,\tau_k)^{BiBj}$$

Where the apex *BiBj* means "as Ag_i believes is believed by Ag_j". Subjectivity means that both the network dependence and the believed abilities and willingness are believed by (the point of view of) the agent Ag_i.

Starting from the Trust Capital we would like evaluate its usable part. In this sense, we introduce the *Subjective Usable Trust Capital* of $Ag_i \in Agt$ about a potential delegable task τ_k as:

$$SUTC(Ag_i,\tau_k) = \sum_{j=1}^{l^{Bi}} \frac{DoT(Ag_j,Ag_i,\tau_k)^{BiBj}}{1 + p_{kj}{}^{Bi}}$$

where $p_{kj}{}^{Bi}$ is (following the Ag_i's belief about the beliefs of Ag_j) the number of other agents in the dependence network that can realize and achieve the same task to whom Ag_j can delegate the task τ_k (see Figure 8). We say that there are two *comparable trust values* when the difference between them is in a range under a given threshold that could be considered meaningless with respect to the achievement of the task. In Figure 8, Ag_1 and Ag_2 strengthen the trust capital of Ag_i (they are competitors with Ag_j about the task τ); while Ag_3, Ag_4 and Ag_5 weaken the trust capital of Ag_i because they are competitors with Ag_i in offering (at the same trustworthy value) the task τ.

As showed in Figure 8 it is possible that Ag_i believes about potential competitors with him (jeopardizing his trust capital) but they are not really competitors because there are no links with his potential clients/delegating (see Ag_3, Ag_4 and Ag_5 that are not linked with Ag_1 and Ag_2 but only with Ag_j).

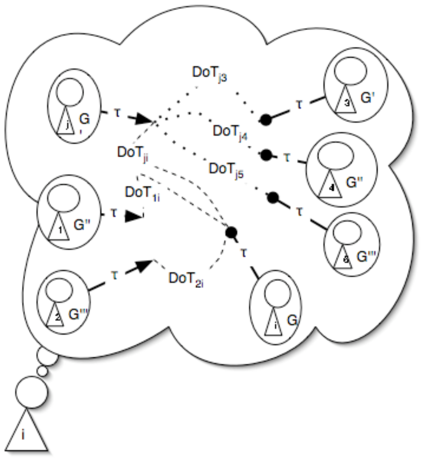


Figure 8

Of course, we can analogously introduce the *Objective Usable Trust Capital* of $Ag_i \in Agt$ about a potential delegable task τ_k as:

$$OUTC(Ag_i,\tau_k) = \sum_{j=1}^{l} \frac{DoT(Ag_j,Ag_i,\tau_k)}{1+p_{kj}}$$

In this paragraph we have introduced in the dependence network (that establishes, objectively or subjectively, how each agent can potentially depend from other agents for solving its own tasks) the trust relationships (that introduce an additional dimension, again assessable both objectively and subjectively, in a potential partner selection for achieving tasks). In general, we can say that the introduction of trust relationships reduces the set of potential partners for each agent and for each task, with respect to the situation with the dependence relationships alone: $OPN > OPN^T$, and $SPN > SPN^T$.

3 Dynamics of Relational Capital

What has not been considered enough in organization theory is the fact that the *relational capital* is peculiar in its being crucially based on beliefs: again, what makes relationships become a capital is not simply the structure of the networks (who “sees” whom and how clearly) but the levels of trust which characterize the links in the networks (who trusts whom and how much). Since trust is based on beliefs – including, as we said, also the believed dependence (who needs whom) – it should be

clear that relational capital is a form of capital, which can be manipulated by manipulating beliefs.
Thanks to a structural theory of what kind of beliefs are involved it is possible not only to answer some very important questions about agents' power in network but also to understand the dynamical aspects of relational capital. In addition, it is possible to study what a difference between trustee's beliefs and others' expectations on him implies in terms of both reactive and strategic actions performed by the trustee itself.

3.1 Changing Trust Capital

For what concerns the dynamic aspects of this kind of capital, it is possible to make hypotheses on how it can increase or how it can be wasted, depending on how each of basic beliefs involved in trust could be manipulated.

In general, starting from the analysis of the previous paragraph, we can see how matching the different terms we have different interesting situations.

First of all, even if $OTC(Ag_i,\tau_k)$ is a relevant factor for the agent Ag_i (it shows in absolute terms how is recognized the trustworthiness of Ag_i), in fact the really important thing for an agent cumulating trust capital is $OUTC(Ag_i,\tau_k)$ that indicates not only the trustworthiness cumulated on the dependent agents, but also the number of possible other competitors on that offered task.

Again more interesting is to consider the $SUTC(Ag_i,\tau_k)$ factor (in which a relevant role is played by the beliefs of the involved trustee) and its relationships with $OUTC(Ag_i,\tau_k)$, $SPN^T(Ag_j,g_{jk})$, and $OPN^T(Ag_j,g_{jk})$ factors. As we have seen in the previous paragraph, these factors are constituted by the beliefs of the trustee or the trustier, so can be interesting to analyze the different situations matching them and evaluating the consequences of their coherence or incoherence.

A general rule (that could be easily translated in an algorithm) regards the fact that the trust capital of an agent (say Ag_i) increases when:

i) decreases the number of other agents (competitors) in the DN offering the solution to the given task (or classes of tasks); and/or

ii) increases the number of agents (delegators/clients) in the DN requiring the solution to the given task (or classes of tasks).

Following this analysis, the trustee should work for decreasing the competitors (for example, disconnecting the links in the network, reducing the reputation of them, and so on) and/or he should work for increasing the delegators (for example, connecting new of them, changing the needs of the connected ones, and so on).

Let us consider what kind of strategies can be performed to enforce the other's dependence beliefs and his beliefs about *agent's competence*. If Ag_i is the potential trustee (the collector of the trust capital) and Ag_j is the potential trustier we can say:

i) Ag_i can make Ag_j dependent on him by making Ag_j lacking some resource or skill (or at least inducing Ag_j to *believe* so). He has to work on $SPN^T(Ag_j,g_{jk})$.

ii) Ag_i can make the Ag_j dependent on him by activating or inducing in her a given goal (need, desire) on which Ag_j is not autonomous (13) but is dependent from Ag_i (or in any case she believes so). In this case he has to find the way for including in G_j an additional g_{jk} such that Ag_j is dependent from Ag_i for that goal (and she believes that).

iii) Since dependence beliefs are strictly related with the possibility of the others (for example Ag_j) to see the agent (for example Ag_i) in the network and to know her ability in performing useful tasks, the goal of the agent who wants to improve his own relational capital will be to *signaling* his presence and his skills (14, 15, 16). While for showing his presence he might have to shift his position (either physically or figuratively like, for instance, changing his field), to communicate his skills he might have to hold and show something that can be used as a signal (such as certificate, social status, proved experience, and so on). It is important to underline that using these signals often implies the participation of a third subject in the process of building trust as a capital: a third part which must be trusted (4).

Obviously also Ag_i's *previous performances* are 'signals' of trustworthiness. And this information is also provided by the circulating *reputation* of Ag_i (17, 18).

iv) Alternatively, Ag_i could work for reducing the believed (by Ag_j) value of ability of each of the possible competitors of Ag_i (in number of p_{kj}) on that specific task τ_k: he has again to work $SPN^T(Ag_j,g_{jk})$.

Let us now consider how *willingness beliefs* can be manipulated. In order to do so, consider the particular strategy performed to gain the other's good attitude through gifts (19). It is true that the expected reaction will be of reciprocation, but this is not enough. While giving a gift Ag_i knows that the Ag_j will be more inclined to reciprocate, but Ag_i also knows that his action can be interpreted as a sign of the good willingness he has: since he has given something without being asked, Ag_j is driven to believe that Ag_i will not cheat on her. Then, the real strategy can be played on trust, sometimes totally and sometimes only partially – this will basically depend on specific roles of agents involved.

Again in formal terms, we can say that Ag_i has to work for increasing his DoW_i as believed by Ag_j ($Bel_j(DoA_i)$).

Alternatively, it could work for reducing the believed (by Ag_j) value of willingness of each of the possible competitors of Ag_i (in number of p_{kj}) on that specific task τ_k.

An important consideration we have to do is that a dependence network is mainly based on the set of actions, plans and resources owned by the agents and necessary for achieving the agents' goals (we considered a set of tasks each agent is able to achieve). The interesting thing is that the dependence network is modified by the dynamics of the agents' goals: from their variations (as they evolve in time), from the emergency of new ones, from the disappearance of old ones, from the increasing request of a subset of them, and so on (6, 20). On this basis, changing the role of each agent in the dependence network, changes in fact the trust capital of the involved agents.

4 Conclusions

Individual trust capital (relational capital) and collective trust capital not only should be disentangled, but their relations are quite complicated and even conflicting. In fact, since the individual is in competition with the other individuals, he has a better position when trust is not uniformly distributed (everybody trusts everybody), but when he enjoys some form of concentration of trust (an oligopoly position in the trust network); while the collective social capital could do better with a generalized trust among the members of the collectivity.

References

[1] Castelfranchi C., Falcone R., Marzo F., (2006), *Being Trusted in a Social Network: Trust as Relational Capital*, in Ketil Stølen, William H. Winsborough, Fabio Martinelli, Fabio Massacci (Eds.): Trust Management, 4th International Conference, iTrust 2006, Pisa, Italy, May 16-19, 2006, Proceedings. Lecture Notes in Computer Science 3986 Springer 2006, ISBN 3-540-34295-8.

[2] Dasgupta P. Trust as a Commodity, in Trust: Making and Breaking Cooperative Relations, chapter 4, pages 49–72. Gambetta(ed), Blackwell, Department of Sociology, University of Oxford, 1998.

[3] Sen S., Banerjee D., Monopolising Markets by Exploiting Trust, Proceedings of the AAMAS 2006. Pages 1249-1256.

[4] Castelfranchi C., Falcone R., Principles of trust for MAS: cognitive anatomy, social importance, and quantification, *Proceedings of the International Conference of Multi-Agent Systems (ICMAS'98)*, pp. 72-79, Paris, July, 1998.

[5] Falcone R., Castelfranchi C., (2001). Social Trust: A Cognitive Approach, in *Trust and Deception in Virtual Societies* by Castelfranchi C. and Yao-Hua Tan (eds), Kluwer Academic Publishers, pp. 55-90.

[6] Falcone R., Castelfranchi C. (2001), The socio-cognitive dynamics of trust: does trust create trust? In *Trust in Cyber-societies: Integrating the Human and Artificial Perspectives* R. Falcone, M. Singh, and Y. Tan (Eds.), LNAI 2246 Springer. pp. 55-72.

[7] Bourdieu, P. 1983: Forms of capital. In: Richards, J. C. ed. Handbook of theory and research for the sociology of education, New York, Greenwood Press.

[8] Coleman, J. C. 1988: Social capital in the creation of human capital. American Journal of Sociology 94: S95-S120.

[9] Castelfranchi, C., Falcone, R., (1998) Towards a Theory of Delegation for Agent-based Systems, *Robotics and Autonomous Systems*, Special issue on Multi-Agent Rationality, Elsevier Editor, Vol 24, Nos 3-4, , pp.141-157.

[10] Castelfranchi C., and Conte R., The Dynamics of Dependence Networks and Power Relations in Open Multi-Agent Systems. In Proc. COOP'96 – Second International Conference on the Design of Cooperative Systems, Juan-les-Pins, France, June, 12-14. INRIA Sophia-Antipolis, 1996. P.125-137).

[11] Sichman, J, R. Conte, C. Castelfranchi, Y. Demazeau. A social reasoning mechanism based on dependence networks. In *Proceedings of the 11th ECAI*, 1994.

[12] Conte, R. e Castelfranchi, C. (1996) Simulating multi-agent interdependencies. A two-way approach to the micro-macro link. In U. Mueller & K. Troitzsch (eds) *Microsimulation and the social science*. Berlin, Springer Verlag, Lecture Notes in Economics.

[13] Castelfranchi, C. Falcone R. (2003), From Automaticity to Autonomy: The Frontier of Artificial Agents, in Hexmoor H, Castelfranchi, C., and Falcone R. (Eds), Agent Autonomy, Kluwer Publisher, pp.103-136.

[14] Schelling, T., *The Strategy of Conflict*. Cambridge, Harvard University Press, 1960.

[15] Spece, M. 1973 Job market signaling. *Quarterly Journal of Economics, 87*, 296-332.

[16] R. Bliege Bird & E. Alden Smith "Signaling Theory, Strategic Interaction, and Symbolic Capital", *Current Antropology*, vol. 46, n.2. April 2005.

[17] R. Conte and M. Paolucci, Reputation in Artificial Societies. Social Beliefs for Social Order. Kluwer 2002.

[18] A. Jøsang and R. Ismail. *The Beta Reputation System*. In the proceedings of the 15th Bled Conference on Electronic Commerce, Bled, Slovenia, 17-19 June 2002.

[19] Cialdini, R. B. 1990: Influence et manipulation, Paris, First.

[20] Pollack, M., Plans as complex mental attitudes in Cohen, P.R., Morgan, J. and Pollack, M.E. (eds), *Intentions in Communication*, MIT press, USA, pp. 77-103, 1990.

[21] Castelfranchi, C. Falcone R., Trust Theory: Structures, Processes and Dynamics" John Wiley and Sons 2009 (in press).

Composing Trust: An Effective Trust Model for Multiagent Teamwork*

Feyza Merve Hafızoğlu[1] and Pınar Yolum[2]

[1] Department of Systems & Control Engineering
[2] Department of Computer Engineering
Boğaziçi University, TR-34342, Bebek, İstanbul, Turkey
{feyza.isik,pinar.yolum}@boun.edu.tr

Abstract. Composed services consist of interacting services. Generally each service in a composed service is brought out by a different service provider. The quality of the composed service depends not only on the individual capabilities of the providers but also on how well they work together. Existing trust models are geared towards identifying single services rather composed services. However, in many settings it is important to find a group of service providers that can be trusted for a composed service. To address this, we propose a trust model that captures how trustworthy a group of service providers is for a particular composed service. The approach is based on capturing relations between services. Our proposed approach is tested on an adaptation of ART Testbed. We compare our proposed model with an existing approach in the literature and show that capturing relations between services pays off in finding useful groups of service providers.

1 Introduction

In dynamic open systems, many agents interact with each other to achieve their goals. In such environments, a self-interested agent selects the most trusted and suitable partners to interact with from a pool of agents whose behaviors are not known. Ideally, an agent should interact with the agent who most probably fulfills the expectations of the requester agent. Trust model consists of opinions of an agent about other agents; it's formed by using its own experience with the related agent and other agents' opinions about the related agent. Each agent builds its own trust model and uses this model to decide on whom to trust.

Whereas there are various approaches for finding individual trustworthy services, most real life needs are satisfied by composite services, rather than single services. Such composite services are realized by groups of service providers. A service in a composite service cannot be performed without considering other

* This research has been partially supported by Boğaziçi University Research Fund under grant BAP09A106P and the Scientific and Technological Research Council of Turkey by a CAREER Award under grant 105E073. We thank the anonymous reviewers for helpful comments.

services, because the services are dependent on each other. Consider a house owner that wants her house to be repaired in a short time. The repairing is a composite service that needs to be carried out by an electrician, a painter, and a plumber. These service providers have to work at the same time in the house due to the limited time of the housekeeper. It is obvious that the manner in which one provider works will affect the service of another provider in the group. Hence, the house owner needs to find a group of service providers that she can trust for the composed service, since even if the service providers work well individually, they may not have the same performance when they work together.

When an agent needs a team of agents rather than a single agent to fulfill its request, the agent will consider the trust to the team instead of the trust to each individual agent in this team. In this case, the agent needs a trust model to evaluate the trustworthiness of possible teams. Whereas a vast literature exists for modeling trustworthiness of individual service providers, there is not much work done in modeling groups of providers for a given composed service.

Developing a group model for trust requires the following questions to be answered. When a group of agents attempt to carry out a composite service but is unsuccessful, how can the blame be distributed among the participants? How can an unsuccessful group of agents be modified so that they become successful? Can addition of agents to a successful group be risky? We study these problems by developing a group model of trust.

Our model is based on the idea of the service graphs [**?**] to build composed services. Service graphs are used to represent the relationship between services, which have one or more common subtask. Intuitively, service graphs are helpful in reasoning about services that are related to each other. If a service provider is a good candidate in a service, it might be a good candidate in a related service as well. Further, to capture the trustworthiness of a group of agents for a particular composed service, a group trust model is introduced. The group trust model measures how a group of agents would be useful in performing a particular composed service. This group trust model is updated as the same group is used for the same composed service.

The remainder of this paper is organized as follows. First, our proposed group trust model and service graph model are presented in detail. Then, we give a brief information of ART Testbed and explain the adaptation of ART to handle composed services. Next, we give a step by step description of the strategies. Next, experimental results are provided. Last, we compare our approach to existing work in the literature.

2 Group Trust

When a group of providers are sought for a composed service, considering the providers' individual behavior is not enough because in carrying out the composed service providers will be participating in teamwork. Teamwork trust problem emerges with the following issue: the behavior of the agent in teamwork environment may differ from its behavior in single service environment. In teamwork, the behavior of the agent depends on the teamwork, other agents in the group, and so on.

One can naively think that whenever a group of agents are required for teamwork, for each subtask, we may select the most trusted agent for the corresponding service in the environment. But there is no guarantee that an agent has the same performance when it is taking place in teamwork and when it is acting individually.

Being in a collaboration may have a positive or negative influence on the performance of agents. For example an agent who is a successful painter works very well individually. However, it has a worse performance when it participates teamwork as a painter. As a conclusion, the idea behind the teamwork is totally different from a single service and it is more complicated in the sense of both representation and its reasoning. Hence, considering only the participant's individual trustworthiness is not going to be enough to understand the trustworthiness of a team.

Possible tendencies of agents who participate teamwork can be listed as the following:

- *Ideal Behavior:* The agent performs well both individually and in teamwork.
- *Group Antipathy:* The agent may dislike being in a team. Thus, whenever the agent participates in teamwork, its performance will be low.
- *Group Motivation:* The agent performs well in the teamwork even if it does not perform well individually, i.e. other agents may help the agent. Being in a group has a positive influence on the agent's behavior.
- *Colleague Effect:* The agent's behavior changes based on the other agents in the team. The agent may perform better with some agents but not with others.
- *Teamwork Effect:* The agent may have a bad performance due to the task characteristics. For example, a painter may work well with another plumber but not so well with an electrician.
- *Familiarity Effect:* The agents in the group improve their performance as the number of cooperations increase.

2.1 Representation

There are two important representations that we use to enable group trust. First representation is used to keep the instances of the teamwork, in other words the previous teamwork experiences of the agent are stored. The latter one is a service graph and used to represent the relation between the services of teamwork.

In group trust model, each agent classifies its experiences with respect to the requested composed service and the agents those participate in carrying it out. Each experience instance in composed trust model consists of a subtask list and the corresponding agents who are assigned these subtasks: that is, a list of agent-subtask pairs.

A group trust model has the information about the agents that exist in the group and the subtasks which agents in the group are assigned requested about.

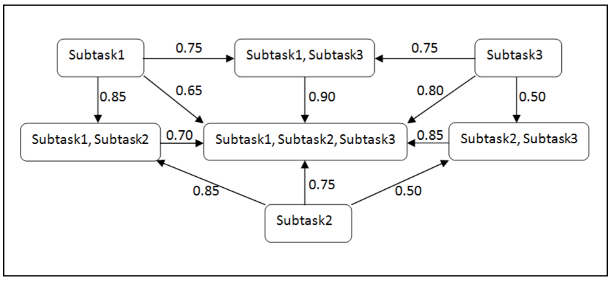


Fig. 1. Service Graph for Services including Subtask1, Subtask2, and Subtask3

Each model has an expertise weight, which is updated after each experience of the group for the same composed service, and the number of interactions which determines the accuracy of the expertise weight. As the number of interactions increase, the expertise weight would be more accurate as mentioned in several models in the literature. The expertise weight of a group trust model instance has a default value of 0.5 when it's created, and it's updated based on the overall performance of the group by using geometric update method.

2.2 Service Graphs

In order to characterize the tasks in teamwork, a graph-based representation of services [?] is used. A service graph is a weighted, directed graph including nodes for service and the edges for transitions between services. The weights on the edges show how likely providers that are successful in a source node are likely to be successful in the target node. Each composite service is a node in the graph. By using this relationship between different composite services, the agent composes a new group of agents for a given service.

In the service graph, only the nodes or services those have at least one common subtask are connected to each other. Otherwise, there is no relationship between two nodes. There is a weight related to each edge in the graph. Default weight, which has a value of 0.5, is assigned to each edge when it is created. The weights capture the likelihood of a group in the source node to be successful in the target node.

When it's the first time the agent is assigned a service which consists of Subtasks 1, 2, and 3, agent adds the directed edges to the service graph for this service, which becomes the destination node and the other services with combinations of the subtask set of this service become source nodes. In this case, assigned service becomes target node, and the service that contains Subtask1, the service that contains Subtask2, the service that contains Subtask3, the service that contains Subtasks 1 and 2, the service that contains Subtasks 1 and 3, and the service that contains Subtasks 2 and 3 are the source nodes. Service graph including these services and weighted edges is shown in Figure **??**.

Using a service graph enables to look from the aspect of agent teamwork. As we asserted our motivation before, the agent may not behave the same way when it is acting independently or when it is taking place in a team. The most important information for an agent, who is assigned a service including more than one subtask, is the edge weights of the graph. Each edge of the graph has an information of weight and number of usages. These edges of the target service are used to establish a group when this target service is assigned to the agent. The agent uses a set of edges, where the union of subtasks of these edges' source services is equal to the subtasks of the target service. Using edges means that the agent selects more useful edges with higher weights and composes groups which are experienced before for the source services of the selected edges to form a group for the currently assigned target service. The weights of edges which are used to build a composed service are updated based on the performance of the actual group whenever it's experienced, and the number of usages is increased by one for each selected edge.

Example 1 Let's say that an agent is assigned a composite service which consists of Subtasks 1, 2, and 3 and decides to use service graph to build a team. Related (services) nodes for this assigned service are all the subgroups of this composite service. The agent can follow the edges to obtain a group for the assigned service. Specifically, it selects a set of useful edges from the edges of the service graph, considering the weights on the edges. Average weights of three possible compositions are calculated by using the edge weights in Figure **??** as the following: (1) the composition of the service that contains Subtasks 1 and 2, and the service that contains Subtask3 is (0.70 * 2 + 0.80) / 3 = 0.73, (2) the composition of the service that contains Subtasks 1 and 3, and the service that contains Subtask2 is (0.90 * 2 + 0.75) / 3 = 0.85, (3) the composition of the service that contains Subtasks 2 and 3, and the service that contains Subtask1 is (0.85 * 2 + 0.65) / 3 = 0.78. Most probably, the agent would obtain a better service by composing the service that contains Subtasks 1 and 3, and the service that contains Subtask2 because this composition has the highest average weight which reflects the relationship between this composition and the required service. Actually, the agent composes the groups of agents experienced before for these selected services.

Example 2 The above is an example for composing nodes of the service graph to obtain requested composite service. Another alternative is separating nodes to obtain the requested service. Let's say an agent is assigned a service that contains Subtasks 1, 2, and 3. The agent may use previous experiences that belong to the service that contains Subtasks 1, 2, 3, and 4. This is done by removing the agent from the group that successfully performs the service with Subtasks 1, 2, 3, and 4. This is an example for the *separation of nodes.* In this study, we only use *composing nodes* method yet.

If the agent decides to use a service graph, it should carry out a two step procedure to determine the group of service providers. The agent begins with finding the most appropriate edges for the assigned composite service by the aid of composing nodes technique. The union of the subtasks of the selected nodes should be equal to the subtasks of the service. Actually, edge is the transition between the different service types and the current service. Second step is the finding of suitable group instance for the selected nodes. If the selected services (nodes) are the service that contains Subtasks 1 and 3, and the service that contains Subtasks 6 and 8, then the agent remembers its experiences for these two services. When it finds successful experiences, it assigns subtasks of these services to corresponding agents with respect to previous experience.

Note that the service graph does not give any information about the expertise of individual agents or a group of agents. It just captures how likely groups that perform certain services are likely to perform other services.

2.3 Non-cooperative Behavior

In this paper, we consider agents whose behavior may change based on the particular composed service that it's taking part of. That is, even if an agent performs well independently, in certain types of composed services, its performance may be bad. Each agent has a finite list of composed services in which it is going to be non-cooperative. This is called the *noncooperativeness list* and may be different for each agent. *Noncooperativeness level* shows the extent of cooperation. If the noncooperativeness level of the agent is 0, then the agent cooperates with all assigned subtasks. If it is 1, then the agent never cooperates in any of the possible collaborations.

3 Experimental Setting

Proposed solution to teamwork trust problem is implemented within the Agent Reputation and Trust (ART) Testbed [?] which is a popular simulation platform to compare different trust strategies in the context of single tasks.

3.1 The ART Testbed

The ART Testbed domain consists of agents that appraise paintings. Each painting belongs to an era, and agents have varying expertise in these eras. An appraiser's expertise is its ability to generate an opinion about the value of a painting. At each timestep, agents are assigned a number of paintings to appraise. All agents are assigned the same number of paintings at the beginning of the game. Agents are paid a fee for each appraised painting. The goal of agents is to maximize their *bank balance* by minimizing appraisal error. As the accuracy of the appraisals of the assigned paintings increases, agents have more clients and consequently earn more money.

If an agent has low expertise value in an era in which assigned painting belongs to, the agent asks opinions of other agents to come up with more accurate appraisals. Intuitively, the agent should query the agents who have higher expertise values of corresponding era to increase the profit. However, the expertise values of other agents are not directly known by the agent. So each agent tries to model and learn the behavior of other agents for existing eras by using its past experiences with other agents and reputation information of an agent, which is asked from other agents. Agents may ask certainty of agents, which is an information about the expertise of an agent about a particular era to decide from which agents to ask opinion. Agents are also paid a fixed fee for each opinion and reputation they provide, so agents may also increase their profit by selling opinions and reputation information to other agents. During the game, the correctness of the replies are not guaranteed, and the strategies of other agents are not known due to the heterogeneity of agents. In fact, it is quite likely that agents provide incorrect information in order to decrease the requester's client base in such a competition environment.

ART simulator is responsible for assigning paintings to the appraisal agents for evaluations, receiving the agents' answers about the painting, calculating the true value of the paintings, and informing the agents about this information. The agents, then, can calculate their errors and act accordingly.

3.2 Frost

The basic trust model includes the answers of the question: how can I trust agent x for era y? We use this basic model to support single tasks. If there are n agents and m eras in the environment, there exist $n*m$ model instances for each agent-era pair. A model instance contains the information of the past interactions with the agent for a particular era such as the expertise weight, number of interactions, and so on. At the initialization of the agent, a default model which has an expertise weight of value 0.5 is created for each agent-era pair. When an agent is assigned a painting belongs to one era and requests the opinion of an agent, the corresponding model instance for this agent-era pair is updated. The expertise weight is increased or decreased based on the appraisal error of the painting and the number of interactions is increased by one.

Geometric update is used for expertise weights as shown in Figure **??**:

```
$error = |appraisedValue - trueValue| / trueValue$
if $(1.0 - error) < expertise$ then
    $expertise = penalty * (1 - error) + (1 - penalty) * expertise$
else
    $expertise = reward * (1 - error) + (1 - reward) * expertise$
end if
```

Fig. 2. Updating the Expertise Values

Reward and penalty weights can be monitored to obtain the most suitable strategy. In this study, we prefer to use high penalty weight and low reward weight. If an agent obtains a better result than its current expertise value which is defined in the basic trust model, then we increase its expertise weight by a small amount. However, if it performs worse, we decrease its expertise by a higher amount to penalize this agent.

3.3 Modified ART

ART is originally designed and developed for comparing and evaluating different single dimensional trust models, since agents are expected to provide a single service, i.e., evaluating a single painting that belongs to a single era. However, to evaluate trust models for composed services, the environment needs to be modified so that an agent will be requested to evaluate a composed service. To facilitate this, we first modified ART Testbed simulator to provide teamwork environment and then developed an agent that can participate in this new platform.

Simulator Side Modification: The fundamental task in ART domain is appraising the value of a painting, which belongs to exactly one era. This can be viewed as a single service. However, a composed service consists of several services that act in combination and each service is fulfilled by a single provider. In order to achieve this, we extend the framework so that a composed service is represented as a painting that belongs to one or more eras. The true cost of a painting is determined at the creation time of the painting. Each era of the painting has an effect on the painting's cost. The effect of each era is represented with normalized weights whose sum is equal to 1.0. In order to evaluate a painting, opinions related to all the eras to which the painting belongs need to be collected. That is, the agent asks the opinion of exactly one agent for each era of the assigned painting, and then agents, whose opinion are requested for any era of the painting, become a group and offer a composed service. Each agent in the group appraises a value for the corresponding era part of the painting.

In the modified ART, the creation process of a painting has three fundamental steps: determining the number of eras, determining the eras, and determining the weights of eras, respectively. Our assumption is that a painting may belong to at least one and at most four eras (out of 10 eras) and the actual number is picked randomly. Finally, the weights of the eras, whose sum is 1.0, are generated.

In ART domain, each composed service (i.e., painting) is characterized by the weights of its subtasks. Hence, the weights of corresponding eras for two different paintings, which belong to the same combination of eras, should be similar to each other. According to this characterization, the first time a combination of eras is created, the weights for this combination are determined randomly and registered in the *weights table*, which stores the weights of eras for each combination has been so far. After registration, whenever a painting belonging to an existing combination of eras is created, the weights of eras for this combination are taken from the *weights table* and slightly perturbed (between 0 and 0.05). This perturbation is decided randomly for the weight of each era. Note that these weights are only known by the simulator. That is, agents are not aware of the weights of the eras for a painting.

These weights have important roles in calculating true values of paintings, appraised values, and thus error rates in the following ways. After determining the weights of a painting at the creation time, the simulator generates a true value for each era part of the painting, and the overall true value of the painting becomes the weighted sum of these partial true values, by using the weights of eras. Remember that each agent in the group appraises a value for the corresponding era part of the painting. Overall appraised value is the weighted sum of the appraised values of agents in the group. Appraisal error of each agent in the group is calculated for the corresponding era by using the appraised value of this era part of the painting and the true value of this era part of the painting. Overall appraisal error is again the weighted sum of these partial appraisal errors.

Example 3 A painting is created and randomly decided to have three eras: Era1, Era3 and Era7. Let's say this is the first time a painting, which belongs to Era1, Era3 and Era7, is created. In this case, normalized weights of these eras are generated randomly as the following: 0.45, 0.25, and 0.30, for Era1, Era3, and Era7 respectively. These weight-era pairs are registered for the combination of Era1, Era3 and Era7 in the *weights table*. The next step is the generation of true values: true values of each era part of the painting are generated with the same formula with the formula used to generate true value of a painting in the original ART Testbed. Let's say true values for the era parts become: 1000, 2000, and 1500 for Era1, Era3, and Era7 respectively. The overall true value is the weighted sum of partial true values:
$(0.45 * 1000) + (0.25 * 2000) + (0.30 * 1500) = 1400$

Agents are only informed about the overall appraisal error, overall true value, and overall appraised value of the assigned painting. That is, an agent learns the error it made in evaluating a painting, but it does not learn the individual errors in different era(s). This corresponds to the case that a consumer does not like a composed service but does not give feedback about which individual parts have failed. Whereas in original ART Testbed, agents try to find the most suitable provider for each painting, in modified ART Testbed, agents need to find the most suitable group of agents for each painting.

Example 4 Continuing from the above example, an agent requests opinions of a group of agents such that one agent will be responsible from one era of the painting. Let's say, appraised values of the agents in the group are 1100, 1980, and 1560 for Era1, Era3, and Era7 respectively. Then the overall relative appraisal error is weighted sum of partial appraisal errors, and calculated by weights, partial true values, and partial appraised values:
$(0.45 * ((1100 - 1000)/1000)) + (0.25 * ((2000 - 1980)/2000)) + (0.30 * ((1560 - 1500)/1500)) = 0.0545$

After some time, if the simulator creates a painting with the same era group Era1, Era3, Era7, then it considers the weights of this combination from the weights table and determines the new weights for the current painting: 0.40, 0.27, and 0.33 for Era1, Era3 and Era7. The weight of Era1 is decreased by 0.05, the weight of Era3 is increased by 0.02, and the last weight is also increased by 0.03 randomly one by one. The sum of these weights is again 1 and the absolute rate of difference of the weights of an era in different paintings with the same era group is at most 0.05. Simulator behaves this way whenever a painting, which is belong to Era1, Era3 and Era7, is created.

Agent Side Modification: Agents are assigned paintings which belong to one or more eras in modified ART Testbed. Appraising a value for any era part of the painting can be thought as a subtask in a composed service. Agents ask the opinion of one agent for each era of assigned paintings in modified ART. The total number of opinions asked for a painting is equal to the number of eras which this painting is belong to.

In the original testbed, when requesting an opinion, an agent asks the opinion of other agents by sending the era name for which an opinion is requested. In modified testbed, when requesting an opinion, in addition to the era name for which an opinion is requested, the agent is informed that it would be involved in a composed service consisting of a particular era group with a particular group of agents. All information is included in the opinion request message of the opinion protocol.

In ART domain, *noncooperativeness property* is defined as the following: when the agent is requested its opinion about an era of the painting which is in the noncooperativeness list of this agent, then this agent will send the worst opinion creation order to mimic the fact that the agent cannot do well in this composed service. Actually, non-cooperative behavior emerges due to *teamwork effect* mentioned before.

3.4 Opinion Request Strategy

Opinion request strategy of the agent for a particular painting depends on the number of eras of the painting. If the number of eras is one, agent uses basic trust model to find the most suitable agent to request opinion. Otherwise, it uses the group trust tools to decide the group of agents.

Remember that the basic trust model includes all agent-era pairs. When the agent is assigned a painting, it selects the most trusted agent for the era of this painting. The most trusted is measured with a weighted sum of certainty of agents and the expertise level of agents for the era. The weights of the certainty and expertise values changes during the game and their sum is 1.0. Actually, these weights are adaptable parameters and their value depends on the current timestep. The weight of the expertise is increased by a small amount in certain timesteps, while the weight of certainty decreases by the same amount directly. Since, the importance of the agent's own experiences increases as the number of experiences increases, the agent with the highest combination of certainty and expertise value is selected to request opinion.

Finding a trusted group of agents for a painting which belongs to more than one era is more complicated. There are four alternative strategies: (1) using the exact experience which is higher than a certain threshold from the group trust model instances, (2) using a set of experienced edges of the service graph if edges with high weights exist in the graph, (3) using inexperienced edges whose group instances have an expertise value higher than a certain threshold, and the last alternative is (4) using the basic trust model. The agent pursues these alternatives one by one in this order. Once it finds a suitable group of agents in any of the steps, it finalizes the opinion request procedure. In order an agent to use the first and second strategies, it should have successful past experiences with the corresponding painting. We explain these strategies next.

First strategy is similar to the strategy used for paintings with one era, but in this case there is no certainty information of a group. Instead, the agent uses the expertise values of the group trust models with the same painting type. If there is an appropriate experience, which is higher than a certain threshold, for exactly the same painting type, the same group can be used.

Second strategy uses the service graph information, namely the weights of the edges. If there exists a set of edges whose average is higher than a certain threshold value, the agent selects these edges, where the union of eras of these edges' source nodes is equal to the eras of the assigned painting. Then, the agent looks at its group trust models to find an appropriate group instance for each selected edges' source node. Finding suitable group instances is similar to first strategy.

Third strategy uses the service graph information, namely the edges of the graph rather than the weights of the edges. The agent finds all possible edge sets by using these edges to obtain a final group for the current painting, and then finds the best groups instances for each possible edge set by looking at its group trust model instances, that is its past experiences. The edge set, which has the highest average expertise based on the expertise weights of the group instances, is selected. If the selected highest expertise is higher than a certain threshold value, these group instances of the selected edges' source nodes are used.

If the agent cannot find a group of agents at the end of first three strategies, it uses the last strategy. In this case, it selects agents one by one for each era of the painting by using basic trust models without considering any threshold value, since no alternative strategy remains in this step. This strategy works well with the agents those have *ideal behavior* mentioned before.

Note that if the agent uses second or third strategies in the current timestep, it keeps the selected edges for the corresponding painting to update the edge weights of service graph at the beginning of the next step, where the appraisal errors are received from the simulator. Additionally, the agent keeps the group and painting pairs to update the group trust model at the beginning of the next step.

3.5 Agent Strategies

We explain the representation of three models, namely basic trust model, group trust model, service graph model, geometric update procedure, non-cooperative behavior of agents, and opinion request strategies so far. In this section, we present how the game evolves based on the ART Testbed messages:

1. *prepareReputationRequests():* The agent doesn't use reputation information. Hence, it doesn't send any reputation request message. Opinion replies are received in this step, the agent updates basic trust models, group trust models and the service graph based on the replies.
2. *prepareReputationAcceptsandDeclines():* The agent accepts all reputation requests.
3. *prepareReputationReplies():* The agent generates reply messages for the agents that request reputation information based on how the agent modeled the agent whose reputation information is being asked.
4. *prepareCertaintyRequests():* We set a high value for the maximum number of certainty messages. Though, the agent sends a certainty request for each era of each assigned painting. The agent prefers to ask certainty especially from agents whose certainty value is unknown about the related era.
5. *prepareCertaintyReplies():* The agent replies the certainty messages according to its real expertise value of the related era.
6. *prepareOpinionRequests():* The agent uses basic trust model for paintings with one era. On the other hand, the agent uses group trust models and service graph for paintings with more than one era.
7. *prepareOpinionCreationOrders():* Noncooperativeness property of the agent emerges in this step. When the agent's opinion is asked about an era of a painting which exists in its noncooperativeness list, then agent order an opinion value of 1 from the simulator via sending a message of type OpinionOrderMsg. If the non-cooperative list doesn't contain this painting type, then the agent orders an opinion value of 10.
8. *prepareOpinionProviderWeights():* Weights don't have any effect in group model setting, since the agent asks one opinion from only one agent for an era of a painting. This is the only opinion that effects the appraisal of the painting about particular era. Formally, the agent sends 1 as a weight to the simulator via WeightMsg.
9. *prepareOpinionReplies():* The agent sends messages of type OpinionReplyMsg by finding the appropriate opinions that are already sent to the simulator.

4 Experimental Results

So far, we have explained how an agent can decide on the trustworthiness of teams using group trust model. Now, through experiments, we evaluate how well such an agent can indeed model a team. To understand this, we compare the performance of the group trust model with the basic trust model. Our agents are named as GMA (Group Modeling Agents) and the other agents are called SMA (Single Modeling Agents), respectively.

SMA agents are modeled by excluding the group trust model and service graph of the GMA agents. SMA agents use exactly the same basic trust model, the same opinion request strategy for paintings with one era, same adaptable parameters, same reward and penalty weights in the update procedure with the GMA agents. The only difference is that when requesting an opinion, SMA agents select agents to request opinion one by one for each era of the painting by using basic trust model.

In addition to SMA agent, we use honest agents, which randomly select the agents to request opinion from. Our experimental setup contains 9 agents: 3 GMA agents, 3 SMA agents and 3 honest agents in the environment. All agents have noncooperativeness behavior and the same noncooperativeness level is used for these agents. The game continues 100 timesteps. We have repeated our experiments with larger populations and our results are similar.

Threshold parameter values used in strategy (1), (2) and (3) are adapted by the agents during the game based on the current timestep. Different threshold parameters are used for these strategies. The expertise weights of group trust models and the weights of the service graph are expected to increase for better groups during the game. Thus, an agent increases the threshold values by a small amount in certain periods.

Example 5 For group trust models, threshold values are adapted as shown in Figure **??**.

```
if $currentTimestep < 10$ then
    $threshold = 0.50$
else if $currentTimestep < 20$ then
    $threshold = 0.55$
else if $currentTimestep < 30$ then
    $threshold = 0.60$
else if $currentTimestep < 40$ then
    $threshold = 0.65$
else if $currentTimestep < 50$ then
    $threshold = 0.70$
else if $currentTimestep < 60$ then
    $threshold = 0.75$
else
    $threshold = 0.80$
end if
```

Fig. 3. Adapting the Threshold Values

We compare GMA and SMA agents based on their bank balances and the behavior of their final bank balances with respect to changing noncooperativeness level. GMA agents considerably outperform SMA agents by using group trust modeling tools. Note that honest agents are not represented in the graphs because they behave randomly and have no strategy.

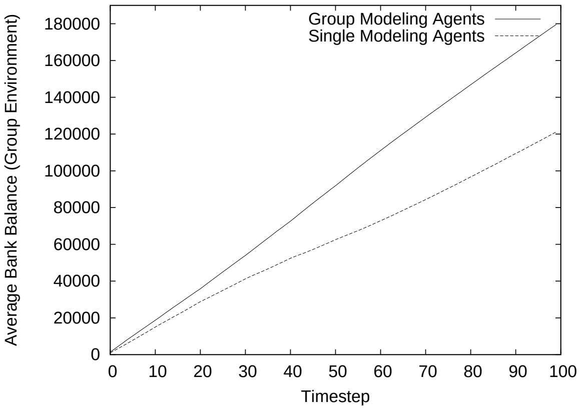


Fig. 4. Bank Balances of GMA vs. SMA

Figure **??** depicts the bank balances of GMA and SMA agents with noncooperativeness value of 0.3. This setting is run 50 times and the average value of bank balance are used in the graph. Accordingly, for each run average bank balances of GMA agents and SMA agents for each timestep are used.

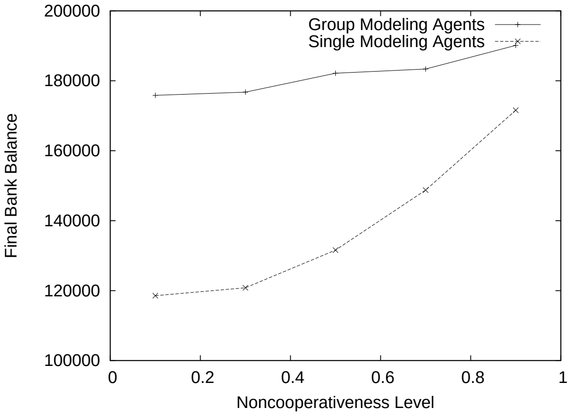


Fig. 5. Noncooperativeness level vs. Final Bank Balances

In Figure **??**, final bank balances of GMA and SMA agents with respect to different noncooperativeness levels are shown. For small values of noncooperativeness level, the difference between the bank balances of GMA and SMA agents has the largest values. This difference decreases as the noncooperativeness level increases and the cooperativeness level of the agents decreases. Decreasing cooperativeness means that agents, from which opinions are requested, send opinions properly for very limited set of paintings and they choose not to cooperate for most of the paintings in the environment. Hence, GMA agents start to misclassify groups as noncooperativeness level increases.

On the other hand, SMA agents cannot handle even the smaller noncooperativeness levels. Since they just use basic trust models, they assume that agent behave the same way for any environment and for any painting. However, since agents behave noncooperatively with different paintings, projection of their behavior to teams is not always successful. Another important result is that higher noncooperativeness levels produce higher bank balances. Remember that noncooperative behavior is paying the smallest amount to the simulator via opinion creation order in ART domain. Hence, as the noncooperativeness level increases, opinion costs (the amount paid by the opinion provider to generate the opinion) decrease, and provider agents save their money.

5 Discussion

Most approaches to trust model consider a single agent and predict its trustworthiness accordingly. However, in many real-life settings, an agent has to interact with a group of agents to receive a composed service. This paper proposes a group trust model to understand the behavior of such teams that carry out a composed service.

Barber [?] presents a trust-based mechanism for team formation problem where agents selectively pursue partners of varying trustworthiness in a market-based environment, where a job consists of multiple subtasks and agents have different skills which correspond to subtasks. A certain percentage of the agents are randomly selected as contractors at each round and they decide to continue their current job or a new job, which turns to establish teams to work on their new job, by using a greedy heuristic. Candidate members of the team have different tendencies towards completing an assigned task. Results show that an agent may utilize better by selecting less trustworthy partners with comparison to more trustworthy partners. In contrast to our group trust model, this study proposed a trust model with the aspect of the participants of teamwork, the agents are modeled individually based on the tendency to complete a subtask and considers subtasks requiring different number of rounds to complete, and maximizing the profit. The behavior of the agents in the team doesn't differ based on the team or teamwork, instead they have certain characteristics to continue or leave their current job based on maximizing their profit.

TRAVOS [?] is a probabilistic trust model that considers both trust and reputation in order to handle the possibility of inaccurate reputation information. Self-interested agents may betray the trust by not performing the requested action as required. In TRAVOS, trust is calculated using probability theory between agents considering the past interactions. Whenever there is little or no interaction with an agent, the agent uses the reputation information gathered from third parties. This study especially handles the possibility of inaccurate reputation information based on the interactions with the agent whom requests the reputation information. However, TRAVOS does not provide a modeling mechanism to evaluate teamwork.

Another solution [**?**] is developed by using Bayesian approach and deals with the sequential decision making problem of agents operating in computational economies. It allows agents to incorporate different trust priors and explore optimally with respect to their beliefs when choosing potential service or information providers. The trustworthiness of the agents in the environment is uncertain. A generic Bayesian Reinforcement Learning algorithm is applied to the exploration-exploitation problem where agents decide whether to keep interacting with the same "trusted" agents or keep experimenting by trying other agents with whom they haven't had much interaction so far. This algorithm considers the expected value of perfect information of an agent's actions to take optimal sequential decisions; it's applied to the ART Testbed scenario.

The proposed solution in Blizzard [**?**] is an action-based approach for modeling the environment; and it is also developed in ART Testbed. Blizzard differs from traditional agent-based trust models by modeling actions of the agent and their effect on the environment instead of models all agents individually. Q-learning method which originally deals with actions and states is used by removing state mapping since there is no state info in ART. Three versions of the Blizzard is developed and compared with Frost agent which is an agent-based trust model in the evaluation part, and it dramatically outperforms the agent-based approaches during evaluations.

References

1. K. Fullam, T. B. Klos, G. Muller, J. Sabater, Z. Topol, K. S. Barber,J. S. Rosenschein, and L. Vercouter. The agent reputation and trust (ART) testbed architecture. In *The Workshop on Trust in Agent Societies at The Fourth International Joint Conference on Autonomous Agents and Multiagent Systems*, 50–62. (2005)
2. P. Yolum, and M. P. Singh. Service Graphs for Building Trust. In *International Conference on Cooperative Information Systems (CoopIS)*, 509–525. (2004)
3. O. Kafalı, and P. Yolum. Trust strategies for ART Testbed. In *In Ninth International Workshop on Trust in Agent Societies, AAMAS*, 43–49. (2006)
4. O. Kafalı, and P. Yolum. Action-Based Environment Modeling for Maintaining Trust. In *Eleventh International Workshop on Trust in Agent Societies, AAMAS*, 23–32. (2008)
5. W. T. L. Teacy, J. Patel, N. R. Jennings, and M. Luck. TRAVOS: Trust and reputation in the context of inaccurate information sources. In *JAAMAS*, 183–198. (2006)
6. W. T. L. Teacy, G. Chalkiadakis, A. Rogers and N. R. Jennings. Sequential Decision Making with Untrustworthy Service Providers. In *AAMAS*, 755–762. (2008)
7. C. L. D. Jones, K. K. Fullam, S. Barber Exploiting Untrustworthy Agents in Team Formation. In *International Conference on Intelligent Agent Technology*, 299-302. (2007)

Selecting Trustworthy Service in Service-Oriented Environments

Chung-Wei Hang and Munindar P. Singh

Department of Computer Science
North Carolina State University
Raleigh, NC 27695-8206, USA
{chang,singh}@ncsu.edu

Abstract. Most of current service selection approaches in service-oriented environments fail to capture the dynamic relationships between services or assume the complete knowledge of service composition is known as a prior. In these cases, problems may arise when consumers are not aware of the underlying composition behind services. We propose a distributed trust-aware service selection model based on a Bayesian network for consumers to maintain their knowledge of the environment locally. Results show our model can punish and reward services in terms of QoS properties accurately with incomplete observations so that consumers can prevent themselves from interacting services with unsatisfying QoS.

1 Introduction

In service-oriented computing environments, computing resources are managed as *services*, which can be used directly or composed into larger services. Service-oriented architecture has been widely adopted in modern distributed environments, such as for cloud computing. We address the problem of selecting services based on criteria such as user requirements and service qualities.

The dynamism of quality of service (QoS) is the first challenge of service selection. For example, the number of requests to a shopping service is much higher in the holiday sale season than usual. Traditional approaches, for example, *Web Service Definition Language* or *WSDL*, describe the functionalities of services statically for users to match services to their needs. However, These approaches lack capabilities of capturing the non-functional aspect of services.

The research on *trust* modeling in artificial intelligence provides us with a promising solution to service selection. Trust is a basis of interactions, indicating the relationships between parties in large, open systems. Two parties must trust each other sufficiently to be willing to carry out desired interactions. In service-oriented context, a party Alice trusts another party Bob, because Alice expects Bob will provide desired service under the expected QoS. The trustworthiness of the parties can be defined by both functional and non-functional properties. Selecting desired services based on trust is called *trust aware service selection.*

Maximilien and Singh [1] develop a trust-aware approach to select services based on well-defined ontologies that provide a basis for describing consumers' requirements and providers' advertisements. Their approach also captures the dynamism by taking QoS properties into account. However, the other aspect of dynamism comes from service composition. Unfortunately, Maximilien and Singh's approach fails to take service composition into consideration. Services may be composed into larger services. The underlying services of composed services may not be shown externally to the consumers. Service composition can be divided into many scenarios [2] and these scenarios can be nested. This makes QoS properties difficult to collect and evaluate. Consequently, our service selection is more complicated than selection without considering compositions because the consumers may not even know with whom they are interacting.

An ideal trust aware service selection should be able to (1) reward/punish underlying services in an appropriate way so that consumers and composed services will become reluctant to interact with low reputation services, and (2) suggest suitable composition.

This paper aims to provide a trust aware service selection model that can capture the dynamism from not only non-functional QoS properties but also service composition in service-oriented environments. We formalize a Bayesian service selection model, develop approaches for consumers to monitor and explore desired service composition. In this paper, we will show that how our approach rewards/punishes the services dynamically with incomplete knowledge of the composition. The suggestion of better service composition will be left as one of our future work.

2 Related Work

Milanovic and Malek [3] compare various modern web service composition approaches. They also conclude four necessary requirements for service composition: connectivity, nonfunctional QoS properties, correctness, and scalability. However, these approaches poorly define QoS properties. Our approach deals with QoS properties separately. No QoS properties are pre-defined.

Menascé [2] studies how QoS properties are aggregated in different service composition scenarios. However, this approach requires the knowledge of the composition. For example, service A invokes service B, which may invoke C and D with probability p_c and p_d. This information is not always available because of two reasons. First, the providers have no incentive to give such information. Second, modeling the invocation probabilities is not trivial. By contrast, our service composition model makes no assumptions. Our approach monitors and explores the desired services dynamically.

Wu *et al.* [4] use Bayesian networks to model a consumer's assessment of a service's QoS. Their approach provides consumers to combine different QoS attributes. Our model uses Bayesian networks to model service composition to evaluate the QoS properties of the composed services. Instead of combining these properties based on the trustworthiness of each QoS property, we may use *multiattribute utility theory* for decision making, which is beyond our scope.

Yue *et al.*'s approach is the closest work to ours. Yue *et al.* [5] propose a Bayesian network-based approach to model the causal relationships between elementary services. Their approach construct a web service Bayesian networks (WSBN) based on the invocations between the services. Then the service composition guidance can be made from the *Markov Blanket* of a given service. However, this approach fails to consider the dynamism because the guidance remains unchanged if the causal relationships are fixed. Our model captures the dynamism by updating the Bayesian network, which will eventually affect the trustworthiness of a service.

3 Service Selection Model

We propose a trust-aware service selection model based on a Bayesian network. We represent trust based on the *beta distribution*, which can be integrated with Wang and Singh's model [6, 7]. The trustworthiness of services should be estimated based on both direct and indirect experience. Direct experience is referred to the previous quality of service received from the target, whereas indirect experience comes from referrals by peers. To model trust from indirect experience, which can be found in [8], is beyond our scope.

Estimating trust from direct experience is not straightforward in a service composition setting, because some services may not expose details of their composition to their clients directly. A client may interact with a composed service without knowing other underlying services. In such a case, evaluating the trustworthiness of services is no longer an easy task. For example, a client books an itinerary from a composed travel agent service, which interacts with other underlying services like flight services, hotel services, and transportation services. Suppose the client is not satisfied with the composed service because of its late response time. The model should penalize the composed service, as well as the underlying ones. If the hotel service, for instance, is reported to be the cause of unsatisfying QoS, the model should reflect the changes in the way that clients or other composed services become reluctant to interact with it. Also, as the experience increases, the model should be able to suggest appropriate composition.

Our service selection method models causal relationships between services with a Bayesian network. Each consumer maintains its own local model to guide itself to reward or penalize services based on direct interactions. The trust information can be also aggregated with referrals from other consumers. Figure 1 shows the architecture of our trust-aware service selection model. Our model is two-fold. First, a consumer keeps interacting with the services, constructs and updates its local service composition model, and get composition suggestions from the model. Second, consumers may exchange referrals with each other. This indirect evidence can be aggregated with the trust information in our service selection model, helping consumers discover strangers and identify desired services more quickly. The integration of the indirect evidence is our future work.

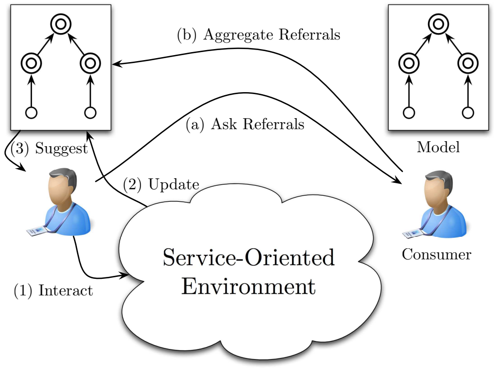


Fig. 1. Trustworthy service selection architecture

3.1 Bayesian Networks

The purpose of modeling service composition is to model how a certain QoS property of a component service can affect the whole composition. For example, the reliability of a composed travel service may be affected by the reliability of the underlying hotel service and flight service. If the underlying service is not reliable, the composed service is very likely not reliable either. Thus, the composition model should be able to not only represent the relationships between (composed) services, but also capture the causal factors between them. Of course, QoS properties of underlying services may not have influence on the composed services. For example, the reliability of the composed service may stay the same no matter how a particular underlying service performs. In other words, the trustworthiness regarding the reliability of the composed service should not correspond to the trustworthiness of that underlying service.

We introduce a Bayesian network-based service selection model, which can be constructed from the *incomplete observations* (direct experience) of a consumer. Here, we emphasize incomplete observations because not all QoS properties are observable from the consumers' point of view. An observation of a particular QoS property of a service d at time t can be represented as a number x_d^t between 0 and 1. Some QoS properties, say, error, can be simply considered as positive 1 or negative 0. Other quantitative QoS properties like up-time should be further projected to an real number from 0 to 1. An observation D^t of the whole composition at time t can be written as $D^t = (x_1^t, x_2^t, \ldots, x_d^t)$, where d is the number of services in the composition.

A Bayesian network is an acyclic directed graph $G = \langle V, R \rangle$ with random variables V as nodes, and edges R as the direct relationships between variables. A conditional probability associated to each node represents the probability of the node variable given its parent variable value. Let each node in the Bayesian network be the probability of getting good service (in terms of a particular QoS property) captured from a composed or elementary service. An edge represents the relationship of composition. For example, in Figure 2, a composed hotel service H is composed of Four Season hotel service f, i.e., f is a child of H. Then the conditional probability of node H is the probability of getting good service in terms of a particular QoS property from H, given f provides good service. T, a travel service, is composed of composed hotel and car rental services H and C, which is also a composed service composed of Enterprise car service e.

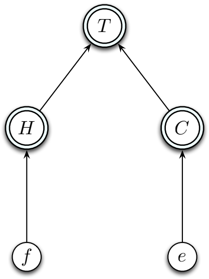


Fig. 2. Service composition example

The Bayesian network models the causal relationships between services. The conditional probability table associated with each node provides consumers a basis of how much responsibility an underlying service should take behind a service composition. The network can be constructed and the conditional probabilities can be learned from the consumers' direct experience.

3.2 Parameter (Trust) Estimation

Given an acyclic Bayesian network graph G over d variables, x_1, x_2, $\ldots x_d$, the associated joint distribution is written as

$$P(x_1, \ldots, x_d) = \prod_{i=1}^{d} P(x_i|x_{pa_i}) = \prod_{i=1}^{d} \theta_i \tag{1}$$

The conditional probability $P(x_i|x_{pa_i}) = \theta_i$ can be estimated by n observations, $D = \{(x_1^t, \ldots, x_d^t), t = 1, \ldots, n\}$, where x_{pa_i} is the set of parent variables of x_i. In fully observable environments, θ_i can be learned from the observed data by *maximum likelihood estimation* (MLE) [9].

In our model, each variable θ_i represents the probability of getting a good service from x_i given getting a good service from x_{pa_i}. The likelihood function can be then defined as [10],

$$P(D|\theta) = \prod_{t=1}^{n} P(x_1^t, \ldots, x_d^t|\theta) \tag{2}$$

$$= \prod_{t=1}^{n} \prod_{i=1}^{d} \theta_i \tag{3}$$

$$= \prod_{i=1}^{d} \prod_{x_i, x_{pa_i}} \theta_i^{n(x_i, x_{pa_i})} \tag{4}$$

$$= \prod_{i=1}^{d} \theta_i^{m_i} (1 - \theta_i)^{l_i} \tag{5}$$

where $n(x_i, x_{pa_i})$ is the number of observations that satisfy the variable setting, and $m_i = n(x_i, x_{pa_i})$ and $l_i = n(x_{pa_i}) - m_i$. Then, the parameters that maximize the likelihood is $\hat{\theta}_i = \frac{m_i}{m_i + l_i}$.

3.3 Bayesian Inference

Note that, when the number of observations is small, MLE may yield over-fitted results. Consider an extreme case where $x_i^t = 1$ for $t = 1, \ldots, n$. The parameter $\hat{\theta}_i$ maximizing the likelihood is 1, which is not reasonable. Thus, we use *Bayesian inference* to treat this problem by introducing a beta distribution $P(\theta_i)$ over the parameter θ_i as a conjugacy prior.

$$P(\theta_i) = \frac{\Gamma(\alpha_i + \beta_i)}{\Gamma(\alpha_i)\Gamma(\beta_i)} \theta_i^{\alpha_i - 1} (1 - \theta_i)^{\beta_i - 1} \tag{6}$$

where α_i and β_i are *hyperparameters* controlling the distribution of parameter θ_i. The expected value or mean of θ_i is given by $E(\theta_i) = \frac{\alpha_i}{\alpha_i + \beta_i}$. Bayesian inference uses observations to update the prior. The parameters θ_i can be learned using Bayes rule.

$$P(\theta_i | D) = \frac{P(D|\theta_i) P(\theta_i)}{P(D)} \tag{7}$$

That is, the posterior distribution $P(\theta_i|D)$ is propositional to the multiplication of the prior $P(\theta_i)$ and the likelihood function $P(D|\theta_i)$. Now we can put equations 5, 6, and 7 together and normalize it,

$$P(\theta_i | D) = \frac{\Gamma(m_i + \alpha_i + l_i + \beta_i)}{\Gamma(m_i + \alpha_i)\Gamma(l_i + \beta_i)} \theta_i^{m + \alpha_i - 1} (1 - \theta_i)^{l_i + \beta_i - 1} \tag{8}$$

Note that the posterior distribution is also a beta distribution. Here we assume the values of x_i are independent of θ_i, i.e., $P(D|\theta_i) = \theta_i$. Then the predictive distribution of x_i given the observations D is defined by the mean of θ_i given the observations D.

$$P(x_i | D) = \int_0^1 P(x_i|\theta_i) P(\theta_i|D) d\theta_i \tag{9}$$

$$= \int_0^1 \theta_i P(\theta_i|D) d\theta_i \quad (10)$$

$$= E(\theta_i|D) \quad (11)$$

$$= \frac{m_i + \alpha_i}{m_i + \alpha_i + l_i + \beta_i} \quad (12)$$

3.4 Dealing with Incomplete Data using Expectation Maximization

In service-oriented settings, some variables may not be observable, which means data is incomplete. In this case, we can use *expectation maximization* (EM) algorithm to calculate a optimal parameter estimation [11, 12].

The idea is that, since some variables are not observable, we can consider those variables without data as latent variables and calculate the expected values of those variables. Let $D_{observed}$ and $D_{missing}$ be the observed and missing data, respectively. Then probabilistic inference can be used to infer $P(x_i^t|D_{observed}, \theta_i^t)$, where $x_i^t \in D_{missing}$ and θ_i^t is the current parameter estimation. We can complete the *counts* (i.e., m_i and l_i) by $P(x_i^t|D_{observed}, \theta_i^t)$. This is called the E step of EM algorithm. For example, suppose there is a composed travel service T, which is composed of an underlying hotel service h. If a consumer observes T has reliability 1 at timestep t (i.e., $x_T^t = 1$) but does not observes the reliability of h, then we can use the expected reliability of h, which is $P(h = 1, T = 1)$, as the observation (i.e., $x_h^t = P(h = 1, T = 1)$). The completed data, i.e., $(x_T^t, x_h^t) = (1, P(h = 1, T = 1))$, can be used as the observations in the M step to update parameter estimation by Bayesian inference. The new parameter estimation of θ_i^{t+1} can be calculated by the posterior mean of θ_i^t. The E and M steps are executed iteratively until the convergence of the estimation. This EM process, which can be viewed as a sequential (on-line) learning method, can be repeated whenever the consumer has new observations.

3.5 Example

We can implement a sequential approach to construct and learn the service composition model from observations. Take the scenario in Figure 2 as an example, Table 1 shows the incomplete observations from a consumer in terms of response time. In the first observation, the consumer interacts with hotel service H with a satisfying response time. The consumer is also aware of the existing underlying Four Season hotel service f and its good response time. In the second observation, the consumer interacts with the car rental service C but with a bad response time. This time the consumer is not aware of any underlying services behind C. In the third observation, the consumer directly interacts with the travel service T with positive experience. It also realizes the presence of the two underlying services H and C. Service H is reported good, whereas service C is reported bad. Service C further reports the bad response time is caused by its underlying Enterprise service e.

Table 2 show the parameters estimation using Bayesian inference. The parameters are represented as a pair of hyperparameters α_i, β_i of the corresponding

t	x_f^t	x_e^t	x_H^t	x_C^t	x_T^t
1	1		1		
2	(0.67)		(0.61)	0	
3	(0.67)	0	1	0	1

Table 1. An example of observation from a consumer's experience

beta distribution. The numbers in the parentheses in Table 1 are the inferred counts to complete the missing data in E step. For example, $n(x_f^2 = 1) = E(\theta_f^1) = \frac{\alpha_f^1}{\alpha_f^1+\beta_f^1} = 0.67$. Then $n(x_H^2 = 1)$ can be inferred by

$$n(x_H^2 = 1) = n(x_H^2 = 1|x_f^2 = 1) + n(x_H^2 = 1|x_f^2 = 0) \tag{13}$$
$$= P(x_H^2 = 1|x_f^2 = 1)P(x_f^2 = 1) + P(x_H^2 = 1|x_f^2 = 0)P(x_f^2 = 0) \tag{14}$$
$$= 0.5 \times 0.33 + 0.67 \times 0.67 = 0.61 \tag{15}$$

Then the completed data can be used to update the parameter estimation. For example, the new estimation θ_H^2 (including $\theta_{H|f=0}^2$ and $\theta_{H|f=1}^2$) is given by

$$(\alpha_{H|f=1}^2, \beta_{H|f=1}^2) \tag{16}$$
$$= (\alpha_{H|f=1}^1 + n(x_H^2 = 1, x_f^2 = 1), \beta_{H|f=1}^1 + n(x_H^2 = 0, x_f^2 = 1)) \tag{17}$$
$$= (2 + P(x_H^2 = 1|x_f^2 = 1) \times x_f^2, 1 + P(x_H^2 = 0|x_f^2 = 1) \times x_f^2) \tag{18}$$
$$= (2.44, 1.22) \tag{19}$$
$$(\alpha_{H|f=0}^2, \beta_{H|f=0}^2) \tag{20}$$
$$= (\alpha_{H|f=0}^1 + n(x_H^2 = 1, x_f^2 = 0), \beta_{H|f=0}^1 + n(x_H^2 = 0, x_f^2 = 0)) \tag{21}$$
$$= (1 + P(x_H^2 = 1|x_f^2 = 0) \times (1 - x_f^2), 1 + P(x_H^2 = 0|x_f^2 = 0) \times (1 - x_f^2)) \tag{22}$$
$$= (1.17, 1.17) \tag{23}$$

t	θ_f^t	θ_e^t	$\theta_{H\|f=0}^t$	$\theta_{H\|f=1}^t$	$\theta_{C\|e=0}^t$	$\theta_{C\|e=1}^t$
0	(1,1)		(1,1)	(1,1)		
1	(2,1)		(1,1)	(2,1)	(1,1)	
2	(2.67,1.33)	(1,1)	(1.17,1.17)	(2.44,1.22)	(1,2)	(1,2)
3	(3.33,1.67)	(1,2)	(1.5,1.17)	(3.11,1.22)	(1,3)	(1,2)

Table 2. The parameter estimation based on the observations

Note that some parameters may not exist until particular observation because the consumer is not aware of the corresponding random variables. For example, service C does not exist until the second observation. The conditional dependencies may change because some underlying services may not be discovered in the first place. For example, $\theta^1_{C|e=0}$ actually means θ^1_C in the first observation because service e does not exist. However, θ^2_C changes to $\theta^2_{C|e=0}$ and $\theta^2_{C|e=1}$ is initialized because service e and the dependency on service C are discovered in the third observation. In these cases, the Bayesian network is updated at the same time to reflect new discovery.

4 Evaluation

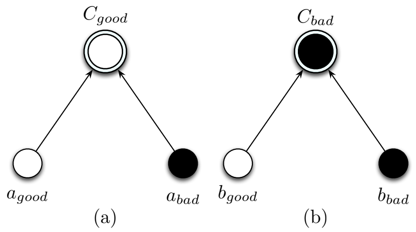


Fig. 3. Two basic experiment scenarios

4.1 Service Selection with Missing Data

Now we evaluate our trust aware service selection model by showing it can reward/punish underlying services in an appropriate way so that consumers and composed services will become reluctant to interact with low reputation services. Two basic scenarios are considered as shown in Figure 3. Shaded nodes represent bad services, which provides unsatisfiable QoS with high probability 0.8, whereas good services provide 80% satisfying QoS. To enable our service composition assumption that underlying services may not be exposed to the consumer, we introduce δ as the percentage of missing data. In each scenario, the consumer interacts with the composed service for d times. Each time the composed service may report the QoS metric of each of its underlying service with independent probability δ. The service selection model is updated sequentially. We measure the quality of the estimation by *root mean square error* (RMS). We also show how the trust (i.e., parameter θ) in services changes over time, and how the performance of underlying services affect the composed services and the whole composition by comparing the parameter θ and the joint probability.

Figure 4 shows the error of trust in a_{good} service in the first experiment scenario for 20% and 40% missing data (i.e., $\delta = 0.2$ or 0.4). The total number of observations d is 100. The trust learned from 40% missing data captures a_{good}'s behavior more slowly than the one learned from 20% missing data. Also, other results show that our approach successfully reward or punish underlying services based on incomplete observations. For example, with $\delta = 0.4$, $P(C_{good} = 1|a_{bad} = 0)$ and $P(C_{good} = 1|a_{good} = 1)$ are 0.78 and 0.76, respectively. $P(C_{bad} =$

$1|b_{good} = 1)$ and $P(C_{bad} = 1|b_{bad} = 0)$ are 0.12 and 0.13, respectively. This result indicates our model correctly evaluates underlying services with 40% missing data, regardless of the goodness or badness of the composed service.

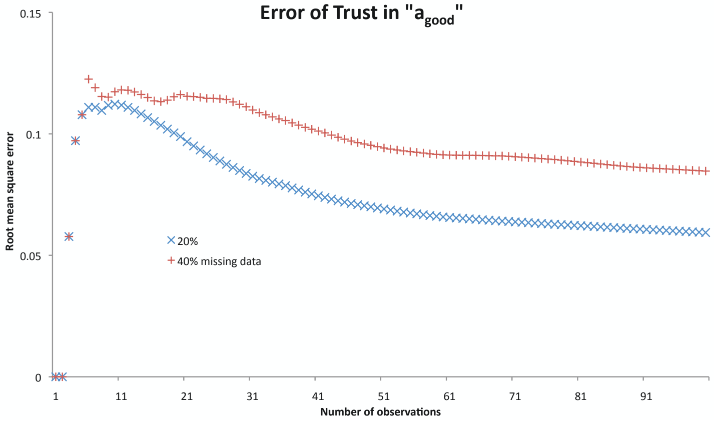


Fig. 4. Error of trust in a_{good} for 20% and 40% missing data

4.2 Service Selection with Dynamic Service Providers

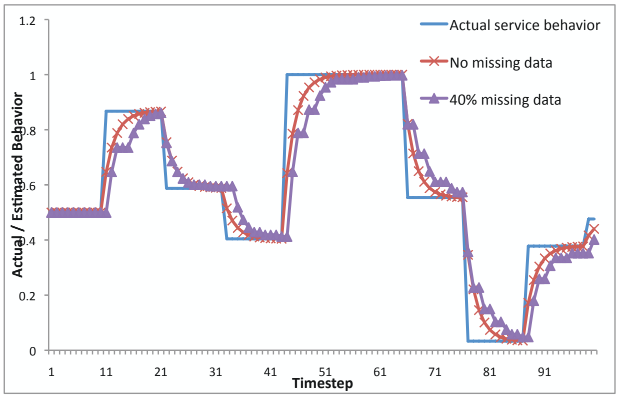


Fig. 5. Tracking a random walk service for different percentages of missing data

Our second evaluation examines the ability of tracking the dynamic behavior of services. We introduce two behavior profiles: *random walk* and *damping*. The random walk profile models the general predictable behavior. The random walk service changes behavior every certain period. Its current behavior x^t depends on the previous behavior x^{t-1}, defined as $x^t = x^{t-1} + \gamma U(-1, 1)$, where γ is a real number between 0 and 1, and $U(-1,1)$ represents the uniform distribution from -1 to 1. In our settings, the random walk service changes behavior every ten timesteps, and $\gamma = 0.8$. The damping profile models the service who turns bad once its reputation is built. Its behavior is defined as $x^t = 1$ when $t \leq T$, and $x^t = 0$ otherwise, where T is the total number of timesteps. Additionally, a discount factor ϕ is used when we calculate posterior distribution in Equation 7, which becomes $P(x_i|D) = \frac{m_i+\phi\alpha_i}{m_i+\phi\alpha_i+l_i+\phi\beta_i}$. The discount factor is a common idea in trust and reputation systems. The estimate reflects the overall behavior if it is high; otherwise, the estimate depends more on the recent behavior. The study of the effect of the discount factor can be found in [13]. Here we set $\phi = 0.6$.

Figure 5 shows how our trust values track the actual behavior of the random walk service with 0% and 40% missing data. The result shows our approach captures the dynamism of the random walk service, although the missing data does slow down the convergence. Figure 6 shows the similar result of tracking damping service.

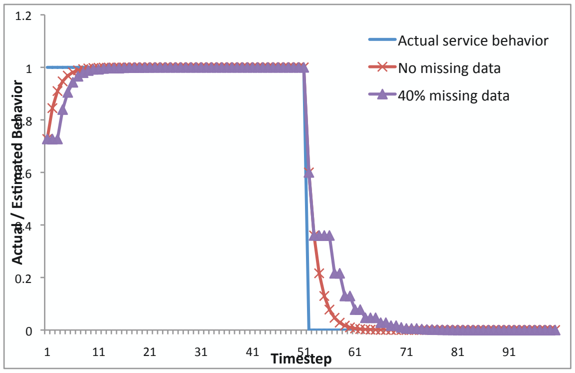


Fig. 6. Tracking a damping service for different percentages of missing data

5 Conclusion

This paper present a trust-aware service selection model in service-oriented environments. The model is built on a Bayesian network to capture the relationships between services. The trust information, which can be integrated with our previous trust model, is learned sequentially from both direct observations and indirect evidence in terms of QoS properties. The main feature of this model is it can deal with incomplete observations, which is as a result of the fact that the underlying services behind service composition may not be observable. Consumers maintain its own knowledge of the environment locally and exchange information each other. Our model rewards services with good QoS metrics and punishes those with bad metrics in the way that consumers will be reluctant to interact with services with low reputation.

Our future work is to refine and enhance an existing QoS ontology from [1] to fit it into our approach. This ontology will be able to capture SLAs as well as the requirements of consumers and advertisements from services regarding SLAs. Both domain-independent and domain-specific QoS properties can be defined in our ontology. Thus, we can further evaluate the QoS properties by comparing the QoS metrics and SLAs, and the sociability of referrers by our trust framework. Knowing the sociability can yield more accurate trust information from referrals. We will study how referrals improve the convergence of trust estimation. We will also apply multiattribute utility theory for decision-making, based on the trust information. Finally, the EM-based parameter estimation in our model can be upgraded to *Structural EM* [14], which can not only learn the trust information but also the graph structure. The learned structure can be used as a suggestion of service composition.

Acknowledgement

This work is supported by the U.S. Army Research Office (ARO) under grant W911NF-08-1-0105 managed by NCSU Secure Open Systems Initiative (SOSI).

References

1. Maximilien, E.M., Singh, M.P.: A framework and ontology for dynamic web services selection. IEEE Internet Computing **8**(5) (September 2004) 84–93
2. Menascé, D.A.: Composing web services: A QoS view. IEEE Internet Computing **8**(6) (2004) 88–90
3. Milanovic, N., Malek, M.: Current solutions for web service composition. IEEE Internet Computing **8**(6) (2004) 51–59
4. Wu, G., Wei, J., Qiao, X., Li, L.: A Bayesian network based QoS assessment model for web services. In: IEEE International Conference on Services Computing. (2007) 498–505
5. Yue, K., Liu, W., Li, W.: Towards web services composition based on the mining and reasoning of their causal relationships. In: APWeb/WAIM. (2007) 777–784
6. Wang, Y., Singh, M.P.: Trust representation and aggregation in a distributed agent system. In: Proc. of the AAAI, Menlo Park, (2006) 1425–1430
7. Wang, Y., Singh, M.P.: Formal trust model for multiagent systems. In: Proc. of IJCAI, Detroit, (2007) 1551–1556
8. Hang, C.W., Wang, Y., Singh, M.P.: Operators for propagating trust and their evaluation in social networks. In: Proc. of AAMAS. (2009)
9. Buntine, W.L.: Operations for learning with graphical models. Journal of Artificial Intelligence Research **2** (1994) 159–225
10. Bishop, C.M.: Pattern Recognition and Machine Learning. Springer (August 2006)
11. Lauritzen, S.L.: The EM algorithm for graphical association models with missing data. Computational Statistics & Data Analysis **19**(2) (1995) 191–201
12. Singh, M.: Learning Bayesian networks from incomplete data. In: Proc. of AAAI, AAAI Press (1997) 534–539
13. Hang, C.W., Wang, Y., Singh, M.P.: An adaptive probabilistic trust model and its evaluation. In: Proc. of AAMAS. (2008)
14. Friedman, N.: The Bayesian structural EM algorithm. In: Proc. of UAI '98, San Francisco, CA, USA. (1998) 129–138

From cognitive trust theories to computational trust*

Jomi F. Hübner[1], Emiliano Lorini[2], Laurent Vercouter[1], and Andreas Herzig[2]

[1] ENS Mines Saint Etienne, France
{hubner,boissier,vercouter}@emse.fr
[2] IRIT, Toulouse, France
{lorini,herzig}@irit.fr

Abstract. Among the several categories of trust models, cognitive models have important features. Initially these models were only informally defined, but formalizations were recently proposed. The concepts of the models are thus sufficiently well defined to be implemented and evaluated. In this paper, the cognitive trust model proposed by Castelfranchi and Falcone is integrated into a BDI (belief, desire, intention) agent architecture and implemented with the *Jason* programming language. The ART testbed scenario is then used to experiment and evaluate both the model and the implementation.

1 Introduction

The concept of trust is important for recent application domains where agent technologies are relevant, such as information retrieval, e-commerce, and peer-to-peer systems. It has been in the focus of many research projects during the last few years, and many theoretical models and systems have been developed. One of the most prominent theoretical model is the *cognitive* model of trust proposed by [2], henceforth abbreviated C&F. Their informal definition of trust is formulated as an individual *belief* about some properties of the trustee.

In this paper we develop further this approach, with the aim of bridging the gap between C&F's cognitive theory and computational models. A first formalisation of C&F trust is proposed in [7], where the definition is refined step by step into more primitive concepts, namely actions, agency, preference and choice (Section 2 briefly presents this formalisation). We here evaluate this definition by means of the ART scenario, which is commonly used as a testbed for trust models (Section 3). We first present an implementation of the C&F definition in a BDI (belief, desire, intention) agent programming language (Section 4). The implementation of that conceptualisation of trust is suitable for such languages since both rely on the same concepts such as beliefs and goals. Besides showing that an agent equipped with the C&F concept of trust performs quite well against other agents of the ART testbed, an important result is that all the trustee's properties included in the C&F definition of trust is shown to be useful in the experiments (Section 5).

* The first two authors are financed by the ANR Project ForTrust (ANR-06-SETI-006)

2 Trust Definition

According to C&F, trust has four ingredients: a truster i, a trustee j, an action α of j, and a goal φ of i.[3] C&F provide a definition of trust which is based on four primitive concepts: capability, intention, power, and goal. In their definition, "i trusts j to do α in order to achieve φ" if and only if:

1. i has the *goal* φ;
2. i believes j is *capable* to do α;
3. i believes j has the *power* to achieve φ by doing α;
4. i believes j *intends* to do α.

For example, when i trusts j to send product P in view of satisfying i's goal of possessing P then (1) i wants to possess P, (2) i believes that j is capable to send P, (3) that j's sending P will result in i possessing P, and (4) that j has the intention to send P. C&F stress the importance of the goal component: it makes no sense to say that I trust j to do α when α is completely irrelevant for my goals.

In [7] this concept was detailed in two types: occurrent trust and dispositional trust. In the former case, the truster has a certain goal and believes that the trustee is going to act here and now in such a way that its goal will be achieved. In the latter case, the truster thinks to be possible that it will have a certain goal in the future and believes, whenever it will have such a goal, the trustee will act in such a way that the goal will be achieved. In this paper only the former type of trust is considered, and it is defined by:

$$\begin{aligned} \mathit{OccTrust}(i, j, \alpha, \varphi) \overset{\text{def}}{=} \; & \mathit{Goal}(i, \varphi) \wedge \\ & \mathit{Believes}(i, \mathit{OccCap}(j, \alpha)) \wedge \\ & \mathit{Believes}(i, \mathit{OccPower}(j, \alpha, \varphi)) \wedge \\ & \mathit{Believes}(i, \mathit{OccIntends}(j, \alpha)) \end{aligned} \tag{1}$$

2.1 The underlying BDI logic

The definition of occurrent trust presented in the previous section has been initially formalised in [10], where a modal logic for reasoning about trust in multi-agent system has been proposed. This logic enables us to specify the five predicates for belief, goal, capability, intention and power on the right hand side of the definition of occurrent trust (definition (1)), namely the predicates Goal, $\mathit{Believes}$, OccCap, $\mathit{OccPower}$ and $\mathit{OccIntends}$. The proposed logic (called $\mathcal{L}$) is a multimodal logic which combines the expressiveness of dynamic logic [6] with the expressiveness of a so-called BDI logic of agents' mental attitudes (see [4] for instance).

It is not the aim of this work to discuss the precise semantics of the modal operators of the logic $\mathcal{L}$. We just present them in an informal way in order to help the reader to understand the relationship between the logical specification of our trust model and its implementation in the *Jason* architecture.[4]

[3] We use α to denote actions and φ to denote goals.

[4] See [10] for an analysis of the semantics of these operators, their relationships, and their correspondence with the structural conditions on the models of the logic $\mathcal{L}$.

The syntactic primitives of the logic $\mathcal{L}$ are the following: countable sets of atomic formulas $ATM = \{p, q, \ldots\}$, agents $AGT = \{i, j, \ldots\}$ and actions $ACT = \{a, b, \ldots\}$. The language of $\mathcal{L}$ is the set of formulas defined by the following BNF:

$$\varphi ::= p \mid \neg\varphi \mid \varphi \vee \varphi \mid \mathtt{After}_{i:\alpha}\,\varphi \mid \mathtt{Does}_{i:\alpha}\,\varphi \mid \mathtt{Bel}_i\,\varphi \mid \mathtt{Pref}_i\,\varphi$$

where p ranges over ATM, α ranges over ACT and i ranges over AGT. Thus, the logic $\mathcal{L}$ has four types of normal modal operators:[5] $\mathtt{Bel}_i$, $\mathtt{Pref}_i$, $\mathtt{Does}_{i:\alpha}$, and $\mathtt{After}_{i:\alpha}$.

These operators have the following intuitive meaning. $\mathtt{Bel}_i\,\varphi$: the agent i believes that φ; $\mathtt{After}_{i:\alpha}\,\varphi$: after agent i does α, it is the case that φ ($\mathtt{After}_{i:\alpha}\,\bot$ is read: agent i cannot do action α); $\mathtt{Does}_{i:\alpha}\,\varphi$: agent i is going to do α and φ will be true afterward ($\mathtt{Does}_{i:\alpha}\,\top$ is read: agent i is going to do α); $\mathtt{Pref}_i\,\varphi$: agent i prefers that φ holds.

Operators for actions of type $\mathtt{After}_{i:\alpha}$ and $\mathtt{Does}_{i:\alpha}$ are normal modal operators satisfying the axioms and rules of inference of system K [3]. Operators of type $\mathtt{Bel}_i\,\varphi$ are just standard doxastic operators satisfying the axioms and rules of inference of system KD45. Therefore, positive and negative introspection over beliefs is supposed, and it is assumed that an agent cannot have inconsistent beliefs. Finally, operators of type $\mathtt{Pref}_i$ are used to express an agent's binary preference. These are similar to Cohen & Levesque's operators [4]. It is supposed that every operator $\mathtt{Pref}_i$ satisfies the axioms and rules of inference of system KD, that is, it is assumed that an agent cannot have conflicting preferences (i.e. an agent cannot prefer φ and $\neg\varphi$ at the same time).

The most important relationships between the four types of operators are expressed by the following logical axioms.

Active: $\bigvee_{i \in AGT, \alpha \in ACT} \mathtt{Does}_{i:\alpha}\,\top$

Inc$_{Act,PAct}$: $\mathtt{Does}_{i:\alpha}\,\varphi \rightarrow \neg\mathtt{After}_{i:\alpha}\,\neg\varphi$

IntAct1: $(\neg\mathtt{After}_{i:\alpha}\,\bot \wedge \mathtt{Pref}_i\,\mathtt{Does}_{i:\alpha}\,\top) \rightarrow \mathtt{Does}_{i:\alpha}\,\top$

IntAct2: $\mathtt{Does}_{i:\alpha}\,\top \rightarrow \mathtt{Pref}_i\,\mathtt{Does}_{i:\alpha}\,\top$

Axiom **Active** ensures that the world is never static, i.e. at every moment there exists an agent i and action α such that i performs α. This is the reason why the operator $\mathtt{X}$ for *next* of LTL (linear temporal logic) can be defined as follows:
$\mathtt{X}\varphi \overset{\mathtt{def}}{=} \bigvee_{i \in AGT, \alpha \in ACT} \mathtt{Does}_{i:\alpha}\,\varphi$. According to **Inc**$_{Act,PAct}$, if i is going to do α and φ will be true afterward, then it is not the case that $\neg\varphi$ will be true after i does α. Axioms **IntAct1** and **IntAct2** relate preferences with actions. Note that $\neg\mathtt{Does}_{i:\alpha}\,\varphi \rightarrow \mathtt{After}_{i:\alpha}\,\neg\varphi$ is not valid. According to **IntAct1**, if i has the preference to perform action α and can do action α then, i is going to do α. According to **IntAct2**, an agent is going to do action α only if it has the preference to perform action α: an agent's *doing* is by definition intentional. Similar axioms have been studied in [11] in which a logical model of the relationships between intention and action performance is proposed.

2.2 The logical definition of trust

The five predicates on the right hand side of the definition of occurrent trust (definition (1)) can be specified in the logic $\mathcal{L}$ as follows:

[5] We here typographically distinguish the informal predicates $Goal$, $Believes$, $OccCap$, $OccPower$ and $OccIntends$ in the definition of occurrent trust from the modal operators of the logic $\mathcal{L}$ written in typewriter font.

$$\begin{aligned}
\mathit{Believes}(i,\varphi) &\overset{\text{def}}{=} \mathtt{Bel}_i\,\varphi\\
\mathit{Goal}(i,\varphi) &\overset{\text{def}}{=} \mathtt{Pref}_i\,\mathtt{X}\varphi\\
\mathit{OccCap}(i,\alpha) &\overset{\text{def}}{=} \neg\mathtt{After}_{i:\alpha}\,\bot\\
\mathit{OccPower}(i,\alpha,\varphi) &\overset{\text{def}}{=} \mathtt{After}_{i:\alpha}\,\varphi\\
\mathit{OccIntends}(i,\alpha) &\overset{\text{def}}{=} \mathtt{Pref}_i\,\mathtt{Does}_{i:\alpha}\,\top
\end{aligned}$$

Thus, agent i has the goal that φ, if and only if i prefers φ to be true in the next state; i has the capability to do α if and only if, i can do α (i.e. at the actual world there exists a possible occurrence of α performed by i); i intends to do α if and only if, i prefers to do α.

It is worth noting that, from Axioms **IntAct1**, **IntAct2**, and **Inc**$_{Act,PAct}$ it follows that the following logical equivalence is a theorem of the logic $\mathcal{L}$: $(\neg\mathtt{After}_{i:\alpha}\,\bot \wedge \mathtt{Pref}_i\,\mathtt{Does}_{i:\alpha}\,\top) \leftrightarrow \mathtt{Does}_{i:\alpha}\,\top$. Therefore, i's occurrent capability and i's occurrent intention to perform action α are together equivalent to the fact that i performs action α, that is:

$$(\mathit{OccCap}(i,\alpha) \wedge \mathit{OccIntends}(i,\alpha)) \leftrightarrow \mathtt{Does}_{i:\alpha}\,\top \tag{2}$$

This is the reason why the definition of occurrent trust given in the previous section can be simplified as follows:

$$\begin{aligned}
\mathit{OccTrust}(i,j,\alpha,\varphi) \overset{\text{def}}{=}\ & \mathit{Goal}(i,\varphi) \wedge\\
& \mathit{Believes}(i, \mathit{OccAct}(j,\alpha)) \wedge\\
& \mathit{Believes}(i, \mathit{OccPower}(j,\alpha,\varphi))
\end{aligned} \tag{3}$$

where OccAct is a predicate used to express action occurrence defined by:
$\mathit{OccAct}(j,\alpha) \overset{\text{def}}{=} \mathtt{Does}_{j:\alpha}\,\top$.

This formalisation of occurrent trust expresses a fundamental aspect of the trust concept, namely the fact that the truster has a goal that φ and believes that the trustee is going to ensure φ by performing action α.

2.3 From binary trust to graded trust

In a recent extension of the previous logic of trust [9], the authors moved from binary trust (i.e. either i trusts j or does not) to graded trust (i.e. agent i trusts agent j with a certain strength x). To this aim, the doxastic operators of the form $\mathtt{Bel}_i$ were generalised to normal operators for graded beliefs of the form $\mathtt{Bel}_i^x$ where $i \in AGT$ and $x \in [0,1]$. A formula $\mathtt{Bel}_i^x\,\varphi$ means: agent i believes φ at least with strength x. Therefore $\mathtt{Bel}_i^1\,\varphi = \mathtt{Bel}_i\,\varphi$.

At the semantic level, every operator $\mathtt{Bel}_i^x$ is interpreted according to a corresponding accessibility relation R_i^x over possible worlds $w, w', ...$ It is supposed that, given two possible worlds w and w', if $x > y$ then $R_i^y \subseteq R_i^x$. Thus, for every agent i, the accessibility relations in $\{R_i^x | x \in [0,1]\}$ induce a so-called system of spheres [8]. This constraint on the accessibility relations R_i^x corresponds to the following logical axiom:

Inc$_{Bel}$ $\quad \mathtt{Bel}_i^x\,\varphi \rightarrow \mathtt{Bel}_i^y\,\varphi$

That is, if $x > y$ and i believes φ at least with strength x then i also believes φ at least with strength y. More generally, the logic of graded beliefs validates:

$$(\mathtt{Bel}_i^{x_1}\,\varphi_1 \wedge ... \wedge \mathtt{Bel}_i^{x_m}\,\varphi_m) \rightarrow \mathtt{Bel}_i^{min(x_1...,x_m)}\,(\varphi_1 \wedge ... \wedge \varphi_m)$$

Such operators of graded belief can be used to represent truster's beliefs with different strengths about different properties of the trustee. As we will show in Section 4, this aspect is important when moving from the abstract model of trust reasoning to the implementation in *Jason*. For example, one would like to say that i (the truster) believes at least with strength x that j (the trustee) will perform action α, or that i believes at least with strength y that j has the power to achieve φ by doing α. These two facts are respectively represented by the formulas $\mathtt{Bel}_i^x\,\mathtt{Does}_{j:\alpha}\,\top$ and $\mathtt{Bel}_i^y\,\mathtt{After}_{j:\alpha}\,\varphi$.

3 Occurrent trust applied to ART scenario

We apply the definition of trust presented in Section 2 to the ART testbed scenario (`http://art-testbed.net`). This scenario is proposed by the trust community as a common testbed for experimentation and evaluation of multi-agent trust models. The ART scenario consists in a simulation of painting appraisals. Several agents are in competition and each agent has a few paintings to evaluate. A painting belongs to a given era and the agents have different level of expertise allowing them to be more or less skilled in the evaluation of a painting's rating according to its era. At each time-step, an agent receives from simulated clients a set of paintings to evaluate. An agent cannot evaluate all the paintings from its own clients and it has to rely on other agents to do it. This is called the *opinion* protocol, where an agent asks other agents an appraisal (or an opinion) for its paintings. In order to choose who to ask for opinions, an agent can use its past direct experiences, and/or two interaction protocols: (i) the *certainty* protocol, according to which, the agent directly asks other agents about their own expertise; (ii) the *reputation* protocol, according to which, the agent asks other agents what is the reputation of a third agent. Every agent has the possibility to lie when communicating in these protocols. Agents are payed when appraising paintings and if they were accurate, they receive more clients at the next step so that the accurate appraisers earn more money.

An agent i may use the concept of trust presented in Section 2 to select a partner j to whom to ask for an appraisal for a painting. To use that conceptualisation, we need to identify the actions and goals in the context of ART. All agents share the same set of possible actions: to appraise paintings of a specific era. The goal of each agent is to give the best possible evaluation for its clients' paintings. In order to achieve this, an agent must select partners to ask for appraisals. Thus, the sentence 'agent i trusts j to appraise a painting p' can be written as follows:

$$\begin{aligned}&\mathit{OccTrust}(i, j, \text{appraise(p)}, \text{good\_eval(p)}, min(x, y))\\ &\overset{\text{def}}{=}\ \mathit{Goal}(i, \text{good\_eval(p)}) \wedge\\ &\quad\ \mathit{Believes}(i, \mathit{OccAct}(j, \text{appraise(p)}), x) \wedge\\ &\quad\ \mathit{Believes}(i, \mathit{OccPower}(j, \text{appraise(p)}, \text{evaluate(p)}), y)\end{aligned}$$

where x and y are the strengths of the two beliefs used in the formula; and $Believes(i, \varphi, x) = \mathtt{Bel}_i^x\, \varphi$.

Having identified the actions and goals for ART, the next and more complex step is to develop some mechanisms which allow agent i to infer those beliefs about the properties of j which are relevant for the achievement of its goal of giving the best possible evaluation for the paintings. Namely, these mechanisms should allow i to evaluate whether the predicates $OccPower(j, \alpha, \varphi)$ and $OccAct(j, \alpha)$ hold in such a way that i can assess the trustworthiness of j. The former predicate denotes j's power to appraise a painting that will help i's goal to give the best possible evaluation for its paintings, whereas the latter denotes that j is going to provide its opinion about the paintings.

A first mechanism is to obtain information about the power of j by means of the *certainty* protocol available in ART. Of course, agents may lie about their expertise possibly leading to incorrectness in i's belief about the predicate $OccPower$. A second mechanism, that can also be applied for $OccAct$, consists in using previous experiences of interaction with j. For instance, if in previous collaborations (when j was asked to respond), j has provided appraisals for i's paintings, then it is concluded that now j has the power to provide appraisals and is going to respond (given that i has asked him). While the first mechanism concerns sincerity issues, the second mechanism concerns all problems related with learning. A third mechanism is to ask other agents their opinions about j's properties, that is, to ask other agents whether the predicates $OccPower(j, \alpha)$ and $OccAct(j, \alpha, \varphi)$ hold. In other words, this third mechanism consists in discovering the reputation of j. However, issues related to reputation are not considered yet in the current stage of our work.

To sum up, in the ART scenario an agent i can exploit various sources of information in order to assess the trustworthiness of some target agent j: communication with j, direct experiences with j, and the reputation of j.

4 From the abstract model to an agent implementation

This section describes how the definition of trust presented in the previous section can be designed and implemented for an agent that participates in the ART scenario. Once the concept of trust is defined on the basis of cognitive ingredients (beliefs, goals, etc.), a suitable agent architecture and programming language should be chosen. For this work, the BDI architecture and the *Jason* programming language were chosen [1]. The main reason to select this language is that it is perfectly suitable for an implementation of the formal definition of trust discussed in Section 2. *Jason* is selected since it is both based on logic programming and on the BDI architecture. Other kinds of architecture and language could be chosen. However, the goal here is to concretely show that the concept can be implemented in at least one configuration.

Figure 1 illustrates the main components of the agent architecture. Briefly, there are data structures for the agent's beliefs, goals, plan library (set of possible plans to achieve goals), and intentions (current plans in execution to achieve the goals of the agent). The *perceive process* updates the belief base from the incoming messages and the *act process* selects an action to be performed from the current set of intentions. The *trust inference* has to decide whether to trust an agent or not. For that purpose a *theoretical* *reasoning* may be enough, in the case where a conclusion can be draw from the current beliefs (for instance, from past experiences). However, in some circumstances a kind of *practical reasoning* may be necessary, i.e. some sequence of actions are required to obtain the necessary information for the trust decision (as in the case where the reputation of an agent has to be asked to others). In this latter case, a new intention is created to perform those actions and obtain the required information.

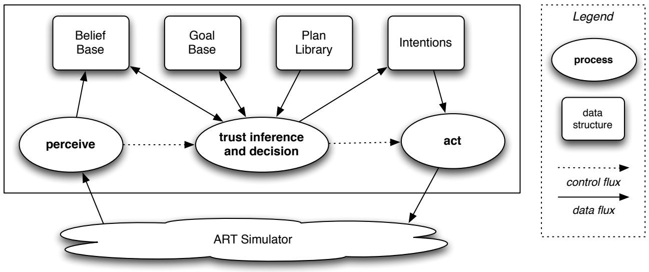


Fig. 1. General agent architecture for trust

The first requirement for the development of our agent is the integration of the ART testbed agent architecture (where the agent have to be coded in Java) and a *Jason* agent architecture that allows the programming of the agent using BDI primitives. *Jason* provides a suitable support to allow this kind of customisation. Roughly, this component provides as perception all data that come from the ART simulator and translates the agent's actions into suitable messages to the simulator. When some particular decision is required for the agent, a new goal is introduced into the reasoning cycle of the agent. For instance, when the simulator requires that the agent performs all the certainty requests, a new goal `!prepareCertaintyRequests` is created. During the agent reasoning process, a suitable plan will be selected to try to achieve this goal resulting in the execution of actions that correspond to a reputation request.

The perception provided by the architecture is translated to first-order predicates and included in the belief base with a special annotation that indicates that they correspond to the agent's perception. Older belief-perceptions are removed accordingly. Among these beliefs, the following are given by the ART simulator and used in the sequel:

- $painting(e, p, t)$: represent that the painting p of era e is allocated to the agent at the current step t of the simulation. The agent's paintings are perceived at the begin of each simulation step.
- $opinion(j, e, v_g, v_r, t)$: represents the appraisal produced by partner j for a painting of era e; the real value of the painting as defined by the simulator is v_r and the opinion provided by agent j is v_g. The quality of the opinion provided by j is based on the difference between v_g and v_r. This sort of information is provided to the agents at the end of each simulation step so that they can evaluate their selection of partners.

All beliefs described above are considered as having strength 1.

In each simulation step for the ART scenario, our agent receives several paintings to evaluate. For each painting it initially assigns n partners using exploitation and exploration strategies (n is the maximum number of opinions an agent can ask for a painting). The exploitation strategy tries to select the n most trustworthy agents in the corresponding era of the painting. If there are not enough trustful agents, partners are randomly selected among the sincere agents (exploration strategy).

The identification of trustful agents uses the definition of occurrent trust (definition 3), i.e. trust is inferred from the agent's goals and beliefs (see the code of Figure 2).[6] The first component of the trust definition is $Goal(i, \varphi)$. In order to check whether this predicate holds, we simply consult all the intentions of the agent. The second component, the belief about $OccAct(j, \alpha)$, is inferred using the following implication ($\alpha = appraise(p)$):

$$\begin{aligned} Believes(i, OccAct(j, \alpha), x) \leftarrow\ & Believes(i, opinions\_count(j, a, g), 1) \wedge \\ & a > 0 \ \wedge\ x = \tfrac{g}{a} \ \wedge\ x > \epsilon \end{aligned} \tag{4}$$

where $opinions\_count(j, a, g)$ is the fact that there were a opinions that were asked to j, and g opinions provided by j. Thus, agent i believes that j is going to collaborate if i has previously interacted with j ($a > 0$) and the percentage of answers provided by j is greater than ϵ ($\epsilon = 0.9$ in our experiments). The strength x of the belief about $OccAct$ is $x = \frac{g}{a}$. Although we use only direct experiences to infer $OccAct(j, \alpha)$, the very particular mechanism used for that could be more complex and efficient. The goal in this paper however is not to optimise the mechanism, but rather to illustrate the implementation of the concept and to compare its influence in the agent performance differentiating agents that consider the predicate $OccAct(j, \alpha)$ in their trust reasoning from those that do not consider it.

The third component of the trust definition, the belief about the property $OccPower(j, \alpha, \varphi)$, is inferred by the following implication when the goal is to have a good evaluation for a painting p ($\varphi = good\_eval(p)$) and the action is to appraise the painting ($\alpha = appraise(p)$):

$$\begin{aligned} Believes(i, OccPower(j, \alpha, \varphi), y) \leftarrow\ & Believes(i, sincere(j), 1) \wedge \\ & Believes(i, painting(e, p), 1) \wedge \\ & y = image_t(j, e) \ \wedge\ y > \delta \end{aligned} \tag{5}$$

where $sincere(j)$ holds when j is believed to be sincere (based on previous interactions with j); $painting(e, p)$ is given as perception by the simulator and is used here to retrieve the era e of painting p; and $image_t(\alpha, e)$ is a function ($image_t : AGT \times ERA \rightarrow [0, 1]$) that maps each agent and era of the simulation step t to the corresponding agent's image. The strength of j power is the same value as its ($y = image_t(j, e)$). Thus, agent i believes that j has power to give a good evaluation on some era if j is sincere and currently has an image greater than δ ($\delta = 0.5$ in our experiments).

The definition of the image function is inspired by reinforcement learning techniques and the Q-Learning algorithm [15]. The reward of asking opinions to j in a simulation step t is given by the mean of all errors in j's opinions:

$$r_t(j, e) = \frac{1}{\#O_t^{j,e}} \sum_{(v_g, v_r) \in O_t^{j,e}} 1 - \frac{|v_g - v_r|}{v_r}$$

[6] The purpose of adding this excerpt of code is twofold: to provide some details of the functioning of the agent and to show how our proposal is implemented in a BDI approach. We do not have the space here to introduce the language; however, we added comments in the code to explain the meaning of the main parts.

```
// trust inference rule, e.g. Act=appraise(p1), Goal=good_eval(p1)
trust(J,Act,Goal)[strength(C)] :-
   .intend(Goal) &                      // I have the goal
   occ_act(J,Act)[strength(X)] &        // J is capable and intend
   occ_power(J,Act,Goal)[strength(Y)] & // J has the power
   C = math.min(X,Y).                   // computes the strength of the trust
   // the strength of beliefs are represented by annotations, enclosed by [ and ]

// when a painting is allocated to me, to evaluate it is a goal
+painting(Era,P) <- !good_eval(P).

// capability and intention are based on the percentage of responses to requests
occ_act(J,appraise(P))[strength(X)] :-
   opinions_count(J,Asked,Provided) & Asked > 0 & X = Provided/Asked & X > 0.9.

// power is based on image and sincerity
occ_power(J, appraise(P), _)[strength(Y)] :-
   sincere(J) & painting(Era,P) & image(J, Era, Y) & Y > 0.5.
   // the image function is implemented as a belief where the third term is
   // the value of the image of agent J

// whenever I receive an opinion from J
+opinion(J, Era, GivenValue, RealValue)
   <- Error = math.abs(RealValue - GivenValue) / RealValue;
      if (Error > 10) {  // huge errors means insincerity
         +˜sincere(J)    // add a belief that J is not sincere
      };
      N = .count(opinion(J,Era,_,_)); // number of opinions
      R = (1-Error)/N;                // reward for the opinion
      ?image(J, Era, Img);            // consult current image
      NewImg = 0.5*Img + 0.5*R;       // compute new image
      -+image(J, Era, NewImg).        // update image belief
```

Fig. 2. Excerpt of the implementation of the trustfulness evaluation in *Jason*

where $O_t^{j,e}$ is the set of all opinions provided by agent j to our agent in paintings of era e and simulation step t; $\#O_t^{j,e}$ is the cardinality of this set; and each element of the set is a pair (v_g, v_r) where v_g is the value provided by j and v_r the real value of the painting.

Considering t as the current simulation step, the current image of j is calculated from the reward of asking opinions to j and the previous image of j:

$$image_t(j,e) = \begin{cases} 0.5 & \text{if } t = 0 \\ image_{t\text{-}1}(j,e) & \text{if } O_t^{j,e} = \emptyset \\ \gamma\ r_t(j,e) + (1\text{-}\gamma) image_{t\text{-}1}(j,e) & \text{otherwise} \end{cases}$$

The first case of the function, when $t = 0$, represents the initial image of j, i.e. 0.5. The second case is selected when no opinion was provided by j in step t, the image of the previous step is then used. The third case uses the reward of asking opinions to j and the previous image. The value of γ ($0 \leq \gamma \leq 1$) represents a discount for past images. We use $\gamma = 0.5$ meaning that the current experiences have the same importance than past experiences.

The above implementation is then used by our agent in each simulation step as follows. (1) For each painting that the agent has to evaluate, assign n partner agents. (2) Participate in reputation protocol. In this implementation, our agent does not ask for any reputation information. It simply answers to reputation requests using the internally build image of others. (3) Participate in certainty protocol. Besides providing answers to requests, where our agent is always sincere, the certainty of the partners are requested. (4) Participate in opinion protocol. In this phase, our agent asks partners for opinions and accepts to provide opinions for every request. The accuracy of the opinion provided by our agent depends on the sincerity of the requester (more sincere agents receive more accurate opinions). (5) Update some beliefs based on the information available in the end of the simulation step: check whether the partners have provided or not an opinion for my paintings and update the $opinions\_count$ belief accordingly; update the $image$ of the partners based on the quality of the opinion they have produced; and update the sincerity property of the agents.

5 Experiments

Two experiments were done with our agent in the ART testbed. In both cases we used the configuration of the 2008 contest and, to produce the graphs, the mean of 10 executions is considered.

In the first experiment the four better placed agents of the 2008 AAMAS ART Contest were included (Uno, Connected, ForPrefect, and Next). We also added one cheating agent (that does not collaborate) and one honest agent (that always does the best for the partners). The result is shown in Figure 3. Our agent, identified by 'ForTrust', is in the group of agents placed second. Although it shows that our agent works quite well, the final performance of the agent is strongly dependent on the particular mechanism used to infer $OccAct$ and $OccPower$ and some parameters like ϵ, δ, and γ. As said before, we are not looking for the optimisation of those parameters here.

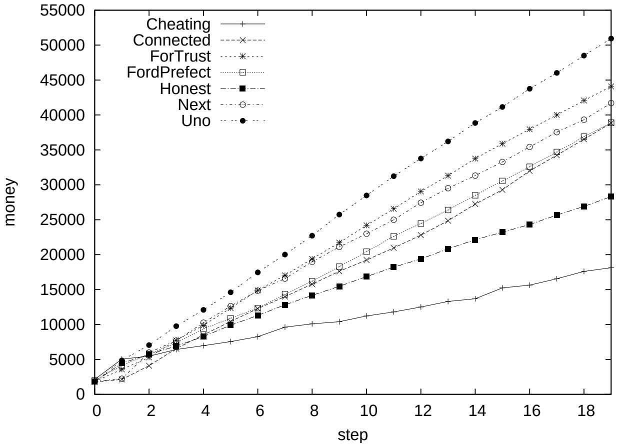


Fig. 3. Simulation results for our agent against some participants of the ART 2008 contest.

In the second set of experiments we intend to identify how the $OccAct$ and the $OccPower$ components of the trust definition interfere in the agent performance. For such an evaluation, four configurations of our agent were created:

Type1: this agent uses the complete definition of trust as presented in Sec 3.
Type2: the trust inference is based on $OccPower$.

Type3: the trust inference is based on $OccAct$.
Type4: this agent trusts in everybody.

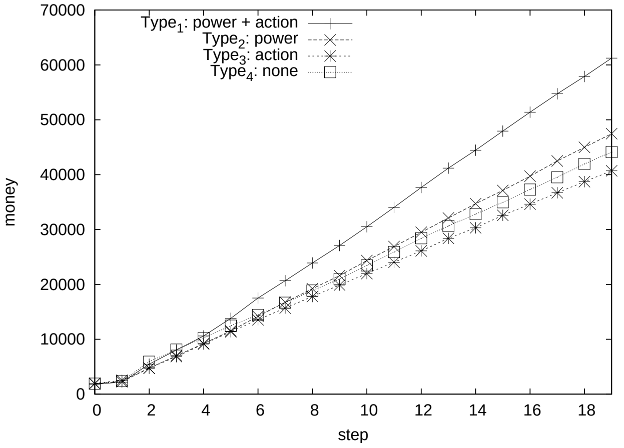


Fig. 4. Comparison of agents that use different ingredients of trust.

Against these four agents, we put four honest agents, one cheating agent, and two lazy agents. Lazy agents are those that promise to provide an opinion (in ART we simulate this by the lazy agents asserting that they are experts), but that do not provide an opinion when someone ask them to do that. Briefly, lazy agents have the power to provide opinions but do not have the intention. The comparative performance of the four types of agents is shown in Figure 4. We can see that Type1, that uses the all the ingredients of the concept of trust performs better. To explain the result, we have to take a closer look at the partners this type of agent. Figure 5 shows how many requests of opinions were done at each simulation step by the agents of Type1 and Type2 respectively —the graph represents thus with whom the agent is interacting. After the exploration phase (around the step 8), the agents start to exploit their trust on other agents. While the agent Type1 rarely interacts with lazy agents since it also considers $OccAct$, the agent of Type2 continues to interact with lazy agent as often as with honest agents.

6 Discussion and related works

The ART scenario brings out some advantages for our experiments since it is well known by the community. It provides useful tools for the analysis of the experiments and other agents (from previous contests) to be included in the simulation and that we can then compare against our proposal. Nevertheless, some constraints might be cited. First, an important feature of the C&F definition of trust is to allow the truster to deal with different goals and actions, in ART however there is only one relevant type of action and it is the same for all agents. Second, the BDI architecture and the *Jason* language are suitable for environments where the agents have to be pro-active, while the ART simulator forces the agents to be just reactive to the protocols of each simulation step.

Although several trust models exist in the literature (a survey is presented in [14]), few of them are based on cognitive concepts. Not only few cognitive models of trust exist, but their integration into an agent architecture is rare. A first work in this direction is [12]. In that work, Pinyol and Sabater propose the integration of the concept of image from Repage reputation model with a BDI agent architecture. Their proposal consists of identifying how the reputation of other agents can influence the beliefs, desires, and intentions of the agent. Although we do not consider reputation in our proposal, our contribution is to use a general concept of trust (where the reputation can be integrated), propose an implementation, and evaluate the proposal in the ART scenario.

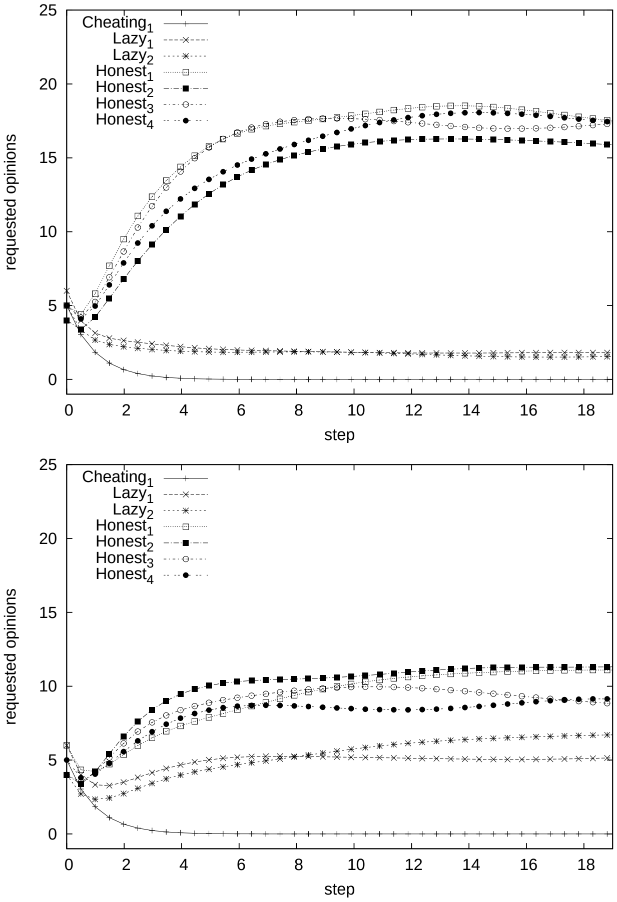


Fig. 5. Partners of the agent Type1 (top) and agent Type2 (bottom).

An important feature of our proposal is that the integration considers two directions: from trust to BDI and vice-versa. For example, when the agent intends to ask an opinion, the trust model is used; conversely, the trust reasoning may trigger new intentions to support the trust decision.

7 Conclusions

We conclude that the *cognitive* concept of trust as proposed by Castelfranchi and Falcone and formalised in Section 2 can be implemented and used by a concrete agent architecture. That concept is particularly suitable to be implemented in a BDI based language as provided by *Jason*. Although we do not take into account other BDI languages, the same conclusion may likely be drawn for similar languages like 2APL [5] and Jadex [13].

Our agent performed well against those of the 2008 ART Competition (2nd rank). The experiments in the ART testbed showed that, with certain types of agents (as the lazy agents used in the experiment), an agent that uses a concept of trust that considers all the ingredients proposed by C&F (goal, capability, power and intention), performs better than an agent that uses only a subset of these ingredients. Some features of our proposal are however not well explored and evaluated due to the limitations of the ART scenario. Future works will include the evaluation of our proposal in more complex scenarios. We also plan to include *reputation* as an important source of information to decide whether the trustee is going to act for the truster's goal and has to power to do that.

References

1. R. H. Bordini, J. F. Hübner, and M. Wooldridge. *Programming Multi-Agent Systems in AgentSpeak using Jason*. Wiley Series in Agent Technology. John Wiley & Sons, 2007.
2. C. Castelfranchi and R. Falcone. Social trust: A cognitive approach. In C. Castelfranchi and Y. H. Tan, editors, *Trust and Deception in Virtual Societies*, pages 55–90. Kluwer, 2001.
3. B. F. Chellas. *Modal logic: an introduction*. Cambridge University Press, 1980.
4. P. R. Cohen and H. J. Levesque. Reasons: Belief support and goal dynamics. *Artificial Intelligence*, 42:213–61, 1990.
5. M. Dastani. 2APL: a practical agent programming language. *Autonomous Agent and Multi-Agent Systems*, 16:241–248, 2008.
6. D. Harel, D. Kozen, and J. Tiuryn. *Dynamic Logic*. MIT Press, Cambridge, 2000.
7. A. Herzig, E. Lorini, J. F. Hübner, J. Ben-Naim, C. Castelfranchi, R. Demolombe, D. Longin, L. Vercouter, and O. Boissier. Prolegomena for a logic of trust and reputation. In *Proc. of 3rd International Workshop on Normative Multiagent Systems (NorMAS 2008)*, pages 143–157, 2008.
8. D. Lewis. *Counterfactuals*. Basil Blackwell, 1973.
9. E. Lorini and R. Demolombe. From binary trust to graded trust in information sources: a logical perspective. In *Trust in Agent Societies 2008*, LNAI, pages 205–225. Springer-Verlag, 2008.
10. E. Lorini and R. Demolombe. Trust and norms in the context of computer security. In *Proc. of the Ninth International Conference on Deontic Logic in Computer Science (DEON'08)*, LNCS, pages 50–64. Springer-Verlag, 2008.
11. E. Lorini and A. Herzig. A logic of intention and attempt. *Synthese*, 163(1):45–77.
12. I. Pinyol and J. Sabater. Cognitive social evaluations for multi-context BDI agents. In *Proc. of Ninth Annual International Workshop Engineering Societies in the Agents World (ESAW'08)*, 2008.
13. A. Pokahr, L. Braubach, and W. Lamersdorf. Jadex: A BDI reasoning engine. In R. H. Bordini, M. Dastani, J. Dix, and A. El Fallah Seghrouchni, editors, *Multi-Agent Programming: Languages, Platforms, and Applications*, number 15 in Multiagent Systems, Artificial Societies, and Simulated Organizations, chapter 6, pages 149–174. Springer, 2005.
14. J. Sabater and C. Sierra. Review on computational trust and reputation models. *Artificial Intelligence Review*, 24:33–60, 2008.
15. R. S. Sutton and A. G. Barto. *Reinforcement Learning: An Introduction*. MIT Press, 1998.

Building a Trust-based Social Agent Network

Sarah N. Lim Choi Keung and Nathan Griffiths

Department of Computer Science
University of Warwick
Coventry CV4 7AL, United Kingdom
{slck, nathan}@dcs.warwick.ac.uk

Abstract. Agents evolving in a multi-agent system interact with one another to achieve their individual goals. In trust-based agent models, agents form a local view of their environment from their direct interactions, and base their interaction decisions on the trustworthiness of the other agents. Agents can also obtain recommendations about other agents from third parties, either directly or indirectly. Reputation complements trust from direct interactions in providing information for agent selection. While trust and reputation ensure that an agent selects and interacts with the most appropriate provider, we believe that the agent can learn about the agent relationships and interconnections at the same time. By building a network of agents it interacts with, and with information about interaction details, trustworthiness, recommendation chain and reputation, the agent is in a better position to extract emergent information, such as potential new customers, suppliers, its competitors and potentially collusive groups of agents. In this paper we propose a mechanism for agents to build a representation of their local environment based on direct interactions, trust and reputation.

1 Introduction

Agents in an open and dynamic multi-agent system (MAS) interact with a group of agents within their area of interest. For instance, in an e-supply chain for computer hardware, an agent representing a part-built computer manufacturer may only be interested in a number of suppliers for computer parts and customer agents. To ensure that it selects the most appropriate agents for interaction, an evaluator can use the concepts of trust and reputation to minimise the uncertainty associated with agent interactions. It gathers trust information from the direct interactions it has with agents. This can be supplemented with reputation information from third party agents when the direct trust information is insufficient or not available. Reputation information is built from both direct and indirect recommendations along a recommendation chain.

We focus on the links that are formed when an evaluator interacts with other agents, both for service provision and recommendations. From direct interactions and direct recommendations, the evaluator has a local view of its environment. By further knowing who is involved in giving indirect recommendations along a recommendation chain, the evaluator can obtain an extended view of its environment. Together with information related to the strength of agent relationships, trustworthiness, reputation, experience and recency of interactions, the evaluator can deduce emergent information that is valuable for future transactions. Emergent information includes the knowledge of potential new customers, new suppliers, who are its competitors and which group of agents are colluding.

The main contributions of this paper are: (i) describing how agents can build an extended view of their local environment, based on trust and reputation, and (ii) proposing potential uses of the emergent information that can be extracted from the extended view of the agent environment. The remaining parts of this paper are organised as follows. Section 2 outlines the related work in the areas of trust and agent networks and in collusion detection as a possible application of the emergent information from the agent network. In Section 3, we describe our mechanism for data collection and network building, which form part of our trust-based agent network model for decision making under uncertainty. Section 4 provides a discussion of how the emergent information can be used to further reduce an agent's interaction uncertainty, and finally Section 5 presents some conclusions and future work.

2 Related Work

Agents interacting in an environment naturally form networks, which potentially hold useful information about the network members and their relationships with one another. The evolution of networks brought more dynamic characteristics and the questions of how the networks are formed, maintained and used are still active research areas. Uncertainty is a characteristic property of interactions among self-interested agents. Thus, agents need to reliably predict the behaviour of other agents to ensure a high level of successful interactions. The concepts of trust and reputation have been proposed to improve the prediction of agent behaviour. We outline the relevant work on agent networks, trust and reputation in agent-based systems and discuss the issues that still need to be addressed, such as the accurate prediction of agent behaviour and collusion.

2.1 Social Networks

The search for relevant information involves finding the right sources, for instance, the agents who have the desired information or expertise. The social network is important in discovering these relevant information sources. An agent is only aware of a portion of the social network to which it belongs [1]. Additionally, due to issues such as privacy, agents will not list their social relationships on a central repository. Agents can however gather this information from distributed searches via referrals. Referrals are important for information flow. Studies of the phenomenon of word-of-mouth found referrals to be very effective in communicating product information among consumers and influencing their purchasing choices [2]. Further evidence that referrals are effective in searching large social networks has been demonstrated for instance by Milgram [3, 4], leading to the concept of *Six Degrees of Separation*. Milgram examined the social connectivity among people and his study involved asking participants to send a packet to a given individual with some information about the person. The participants had to send the packet through individuals they knew by their first name, hence the participants had to choose the most likely intermediary in the chain. Milgram concluded that the individuals within the study were separated by an average of six intermediaries, or six degrees of separation.

Link Prediction. The high dynamism of social networks suggests the addition of new interactions and deletion of old links in the underlying social structure, thus making the understanding of the mechanisms of evolution of social networks important. Liben-Nowell and Kleinberg [5] study *link prediction* as a basic computational problem underlying social network evolution. They describe the problem as involving the accurate prediction of the edges that will be added to the network, during the interval from a time t to a given future time t'. They seek to discover the extent to which the evolution of a social network can be modelled using features intrinsic to the network itself. The link prediction problem is also relevant to the company environment, where the company can benefit from the interactions occurring within the informal social network among its members. These interactions serve to supplement the official hierarchy imposed by the organisation [1, 6]. In our view, the link prediction problem has parallels with the discovery of emergent information about an agent's environment through the agents' local views, which can be overlapped to some extent to give a wider perspective of the other agents in the system, their transactions and social links. Agents often have a notion of the agents in their environment from their direct interactions, in acquiring services or opinions. Indirect service interactions and recommendations are also useful in predicting the relationships among agents. For instance, an evaluator can infer from an indirect recommendation that the secondary recommender has used the target as a service provider. Although the aim of the recommendation request is to evaluate the trustworthiness of the target, the evaluator is also able to draw a link between the secondary recommender and the target, thus building a more complete view of its environment.

2.2 Trust and Reputation

Trust and reputation models have been developed to improve the success of interactions by minimising uncertainty. Many of the models are based on Marsh's trust formalism [7], in using trust to assess the likelihood that an agent honours its promises. Several of the existing models use the notion of an agent neighbourhood, the more relevant ones are briefly described below.

ReGreT is a model of trust and reputation with three dimensions of information: *individual*, *social* and *ontological*. The social dimension includes information on the experiences of other members of the evaluator's group, or neighbourhood, which is assumed to be a group of agents with some common knowledge. FIRE [8, 9] is a modular approach that integrates up to four types of trust and reputation from different information sources, according to availability: *interaction trust*, *role-based trust*, *witness reputation*, and *certified reputation*. The notion of neighbourhood is used by FIRE in its witness reputation module for searching for relevant witnesses. This is based on Yu and Singh's referral system for multi-agents, enabling them to share referrals for the location of relevant information [10]. Other trust-based network models include TrustNet [11], and Histos [12].

2.3 The Collusion Problem

Despite the ongoing research into agent systems, there remains some open issues that still need to be resolved to make multi-agent systems more widely used in real-life systems. The problem of collusion is a complex issue, especially in decentralised systems. Collusion is defined as a collaborative activity of a subset of users that grants its members benefits otherwise not gained as individuals [13]. We view collusion as occurring in centralised and decentralised systems, and within each, various solutions have been proposed to address collusion issues.

Collusion in Centralised Systems. Centralised systems include centralised reputation systems, such as eBay [1] and Amazon [2], where reputation values about individual agents are collected and managed by a central system and every user in the system sees the same reputation value for another user. In these centralised systems, members have a global view of the entire system and this view is unique to all. Jurca [14] proposes a method for designing incentive-compatible, collusion-resistant payment mechanisms, by using several reference reports. The idea behind deterring lying coalitions is to design incentive-compatible rewards that make honest reporting the unique or at least the "best" equilibrium. Meanwhile, Lian *et al.* [13] report on the analysis and measurement results of user collusion in Maze, a large-scale peer-to-peer (P2P) file-sharing system. Their aim is to observe user collusion in P2P networks that use incentive policies to encourage cooperation among nodes. They search for colluding behaviour by examining complete user logs and incrementally refine a set of collusion detectors to identify common collusion patterns. They found collusion patterns that are similar to those found in Web spamming.

Wang and Chiu [15, 16] propose to use social network analysis in online auction reputation systems to analyse the underlying structure of the accumulated reputation score and its corresponding transactional network. They demonstrate that network structures formed by transactional histories can be used to expose underlying opportunistic collusive seller behaviours. Transaction logs and social relationship structures are used to reconstruct the relationship profiles to supplement the lack of demographic data in the online environment. To identify ill-intended users, Wang and Chiu have used real world blacklist data, consisting of suspended fraudulent accounts collected from the Yahoo Taiwan Inc. online auction site. However, the lack of cooperation from online auction hosts limits data collection and the prediction capability.

Collusion in Decentralised Systems. In decentralised systems, such as P2P systems, trust and reputation information for members are collected and stored across the network by each individual member to help in predicting their future interactions. Moreover, individual members do not have a global view of the whole system. TrustGuard [17] is a framework designed to provide a dependable and efficient reputation system that focuses on the vulnerabilities of the reputation system to malicious behaviour, including

[1] http://www.ebay.com

[2] http://www.amazon.com

strategic oscillation of behaviour, shilling attacks, where malicious nodes submit dishonest feedback and collude with one another to boost their own ratings or bad-mouth non-malicious nodes, and fake transactions, which can lead to fake feedback. The main goal of TrustGuard's safeguard techniques is to maximise the cost that the malicious nodes have to pay in order to gain advantage of the trust system. The behaviour of non-malicious and malicious nodes are defined using game theory. The problem of fake transactions is tackled through having feedback bound to a transaction through a transaction proof, such that feedback can be successfully filed only if the node filing the feedback can show the proof of the transaction. To deal with the problem of dishonest feedback, a credibility factor is proposed that acts as a filter in estimating reputation-based trust value of a node in the presence of dishonest feedback.

Synthesis. Open issues, such as collusion, still need to be resolved in decentralised multi-agent systems. The main strategy to detect collusive behaviour, as used in centralised systems, is to have a global view of the system in order to identify the possible colluding agents. However, such a global view is not available to individual agents in a decentralised MAS, as there is no central management of agent information. Despite the limitations of an agent's local view of its environment, we believe that the local view can be complemented by recommendation information about other agents to form an extended view, so that individual agents can have access to a relevant set of information concerning their own transactions. Trust and reputation information, together with the agent network, can build and maintain the extended localised view of the agent environment.

3 Multi-agent Network Model

Our multi-agent network model is designed to capture the dynamic behaviours of agents, their interactions and any emergent behaviour and information. The model consists of three main components: (i) data collection, (ii) network building, and (iii) analysis of interaction data. The data collection module is largely presented in our previous work on agents using trust, as well as direct and indirect recommendations to better inform their decision making for agent interactions [18]. We supplement the history of past interactions with a history of relevant recommendations, and using these to build a network of the agent environment. With the combined information, agents are aware of a wider view of their environment, beyond their local view. We believe that analysing this extended view can help agents discover emergent information, that will allow them to take decisions on issues, such as collusion.

3.1 E-supply Chain Scenario

We consider the case of a computer hardware e-supply chain, where the component suppliers provide products to customers, which include computer systems manufacturers, computer shops and computer parts resellers. In a two-stage supply chain, a customer obtains components directly from the supplier, for instance the memory card and hard disk. A customer typically needs to purchase different types of components and there are several suppliers that can do the job. In an e-supply environment, many computer manufacturers and resellers need to interact with various suppliers to source the necessary components to build or sell their systems. Customers can also act as suppliers for partly-assembled components, for example, a computer shop sells partly-built computers, to which components, such as hard disks and memory chips need to be added on. In this competitive industry, there are many stakeholders and they each try to get the most benefits and attain their individual goals and objectives. In an environment where suppliers have variable performance and reliability, a customer needs to ensure that it interacts with the most trustworthy supplier for the required product to minimise costs and production times. A computer systems manufacturer, denoted as Customer C_1, needs to purchase computer monitors and there are 3 suppliers, Supplier S_1, S_2 and S_3, with different offers. The cheaper supplier is not necessarily the best choice as it might also be the one providing the worse quality products. Using our model of trust and reputation, C_1 can make the decision on which supplier to use, based on previous interactions and recommendations from other agents.

Trust from Direct Interactions. An evaluator assesses another agent's direct trustworthiness from its history of past interactions. For instance, the evaluator, Customer C_1 wants to assess which of the 3 suppliers is the most trustworthy for future transactions. It has interacted with 2 of the suppliers previously, S_1 and S_2. From its interaction history, C_1 can assess how trustworthy each supplier has been, based on service characteristics, such as successful delivery, timeliness and cost. For a similar number of interactions, supplier S_1 has been trustworthy in all the important service characteristics 90% of the time, compared to 50% for supplier S_2. From this comparison, C_1 can decide to use supplier S_1 for its next order of computer monitors.

Reputation from Direct Recommendations. Customer C_1 also requires supplies of hard disks, a recent addition to the component parts it needs. There are 2 suppliers for this component, namely S_3 and S_4. C_1 has purchased from S_3 once before and has not interacted with S_4 previously. With insufficient past interactions to reliably assess the trustworthiness of either supplier, C_1 can complement information from direct trust with recommendations from agents that have previously interacted with S_3 and S_4. C_1 has a regular customer C_2, a computer shop, which resells computers and computer parts. Since C_2 stocks hard disks for resale from both suppliers, C_1 can obtain its opinion about these suppliers.

Reputation from Indirect Recommendations. Considering the case where C_1 wants to assess the trustworthiness of suppliers S_3 and S_4, but it has insufficient direct interactions with them to make an informed decision about whom to approach for the next order. This time, customer C_2 has not interacted with either suppliers, but it knows another agent C_3, which has interacted with both S_3 and S_4. C_2 therefore gives an indirect recommendation about the suppliers to C_1, based on C_3's experience.

3.2 Data Collection Component

Let us consider the representation of a customer agent, a_c, acting as an evaluator. Agent a_c records a partial history of provider interactions, $H_{i_s} = (\mathbb{P}\, i_s, count^+, count^-, ST, ST_c)$, where $i_s = (a_c, a_p, s, t)$ is a service interaction. The provider agent is a_p, s is the service performed at time t, $count^+$ and $count^-$ are the number of positive and negative interactions experienced by a_c respectively. ST is the situational trust in a_p and ST_c is the confidence in the situational trust value. The service s is defined as the service type and a set of dimensions, each defined as: $d = (d_{type}, d_e, d_a)$, where d_{type} is the dimension, d_e is the expected value, and d_a is the actual value following an interaction.

The evaluator a_c also holds a history of the recommendations, obtained from direct and indirect witnesses: $H_{i_r} = (\mathbb{P}\, i_r, count^+, count^-, RT, RT_c)$, where $\mathbb{P}\, i_r$ is the set of recommendations, RT is the recommendation trust in the witness and RT_c is the confidence in that trust. Recommendations are defined as $i_r = (a_c, a_t, a_r, s, t, r)$ where a_t is the target, a_r is the witness who gives recommendation r at time t, and s is the service recommended. Recommendations can be direct, $r^d = (s, a_r, count^+, count^-)$ or indirect, $r^i = (a_{r'}, r^d_{a_{r'}}) \vee (a_{r'}, r^i_{a_{r''}})$, where $a_{r'}$ is an indirect recommender and $r^d_{a_{r'}}$ is the direct recommendation of $a_{r'}$, and $r^i_{a_{r''}}$ is the indirect recommendation of the next witness in the recommender chain $a_{r''}$.

As the evaluator takes into consideration recommendations to decide about provider selection, it updates its recommendation trust in the witnesses and also records the interaction results in its history. The interaction history gives a reflection of the relevant past transactions of an agent. The evaluator applies a decay function to the older interactions to give higher importance to the more recent ones. More details on the performance evaluation using trust and reputation can be found in [19].

3.3 Network Building Component

As an evaluator interacts with providers and witnesses it gathers information about interactions and relationships to build an agent network to better understand its environment. We consider three graph structures to represent an agent's environment: provider graph, witness graph, and a combined provider-witness graph. The nodes represent agents and the edges correspond to links between agents, including the strength of the link in terms of experience. For both the provider and witness graphs, these are further differentiated into service-oriented or agent-oriented graphs. Service-oriented graphs concern interactions and recommendations about a particular service, whereas agent-oriented graphs concern the agents in general. The agent-oriented provider graph is an example of a combined provider-witness graph as an evaluator constructs it from its own direct interactions and inferred interactions between other agents from the recommendations it receives.

Algorithm 1 shows how part of the agent graphs is constructed and updated, where r_μ is the currently processed recommendation. For a direct recommendation, an edge is created for each new recommender and the recommendation count is incremented. Indirect recommendations are updated recursively, with edges created or updated from the further recommender $a_{r''}$ in the chain to a closer one $a_{r'}$. Moreover, the evaluator a_c also updates its provider graph to include the link between $a_{r'}$ and $a_{r''}$, since $a_{r'}$

Algorithm 1 Provider and Witness Graph Updates for Indirect Recommendations

```
for all indirect recommendations $r^i$ do
    if $r^i.a_{r'} \not\in \mathbb{P}\, a_{r'}$ then
        add edge($a_{r'}, a_c$) in $a_c.witnessGraph$
    increment $count_{response}$
    repeat
        if $r^i.a_{r''} \not\in \mathbb{P}\, a_r$ then
            add edge($a_{r''}, a_{r'}$) in $a_c.providerGraph$
            increment $count_{response}$
    until $r_\mu = r^d$
```

obtained a direct recommendation from $a_{r''}$. Every time an edge is added or updated, the number of accurate, inaccurate or unused recommendations is incremented; this is represented by $count_{response}$ in the algorithm.

The evaluator agent continuously maintains its provider and witness graphs throughout the period of interaction with other agents. The graphs contain a summary of the links between two agent nodes. For instance, the graph edges in provider graphs record the number of positive and negative interactions between the two agents. Meanwhile, the witness graph edges consist of the number of accurate and inaccurate recommendations by the witnesses, both for direct and indirect opinions. As in our trust model, where trust values are decayed according to how recent they are, the graph data is also subject to decay, but the decay function is applied when the data is used, rather than when it is recorded, since the agent might choose to apply different decay functions at different times.

4 Discussion: Analysis of Interaction Data Component

In this section, we give an overview of the third component of our model, which involves the analysis of the emergent data from the agent graphs. The collection of interaction data over medium to long term transaction periods of an agent enables it to make decisions about numerous aspects, particularly with the view to increase the success of its interactions and maximising its benefits. Besides using trust and reputation to efficiently select interaction partners and witnesses, that information, together with agent network details can bring more insight into other aspects of the agent environment. For instance, agent networking information helps in reinforcing the trust in the roles of witnesses to give accurate information. We believe that emergent information obtained from the multi-agent network can be used to find solutions to the issues of collusion. We discuss how the agent network can be analysed to extract clues to categorise potentially colluding agents.

4.1 Example Usage: Collusion Detection

Collusion detection is one of the potential uses of the emergent information from the agent network. Examples of simple collusion include: witness and target collusion,

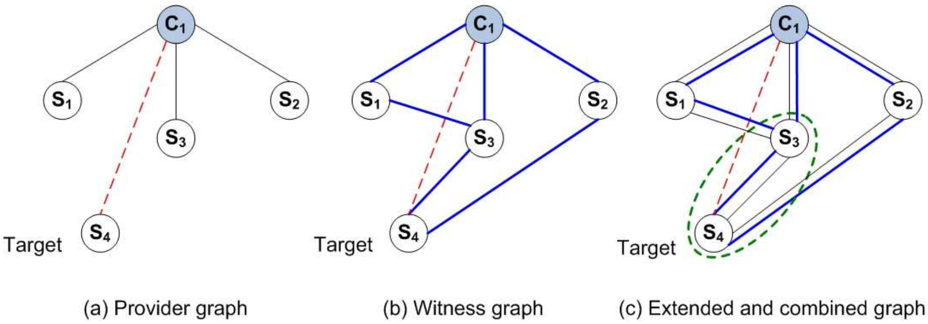


Fig. 1. Collusive behaviour between target and witness

where the witness promotes the target, collusion among witnesses to manipulate a target's reputation, and provider collusion over price. Collusive behaviour is characterised by elements such as heavy agent interactions or similar responses to queries as witnesses. Witness and target collusion is depicted in Figure 1, based on the e-supply chain scenario described in Section 3.1. The evaluator is Customer C_1, which is already using the services of three providers, Supplier S_1, S_2, and S_3. Now C_1 needs a new type of service, which is offered by Supplier S_4. However, C_1 has never interacted with S_4 and therefore decides to request for recommendations from agents who have. Figure 1(a) shows C_1's provider graph. The solid lines represent direct interactions between two agents, while the dashed line shows the target agent that the evaluator is considering for interaction. Agent C_1's witness graph, Figure 1(b) shows the recommenders it uses, through the bold solid lines in the diagram. For instance, S_1 has not interacted directly with S_4 and therefore only gives an indirect recommendation to C_1, via S_3.

The combination of the provider and witness graphs gives Figure 1(c), from which the evaluator can extract information not previously known about certain agent relationships. An additional provider graph edge, between S_1 and S_3 can be derived from the provider and witness graphs. Since S_1 has provided an indirect recommendation to C_1 and S_3 is the only secondary witness, this implies that S_1 and S_3 have direct service interactions. The dashed line circling S_3 and S_4 shows potential collusion between the witness S_3 and target S_4. C_1 requests recommendations about target S_4 from its three service providers, S_1, S_2 and S_3, who can be considered to be trustworthy enough to take their opinions into consideration. From the combined graph Figure 1(c), the evaluator C_1 observes over a period of interaction that S_1 and S_3 have similar recommendations about S_4, as compared to the recommendations of S_2. The emergent information is that S_1's indirect recommendation has been obtained along a recommendation chain of length 2, via S_3. Subsequently, as the recommendations from S_3 are more positive than that of S_2, and from its own initial direct interactions with $S4$, C_1 can suspect that S_3 is colluding with S_4 to promote S_4 as a trustworthy provider. Without the agent network, the evaluator, using only trust and recommendations, would eventually have a low recommendation trust in both witnesses S_1 and S_3, without identifying that S_3 was the dishonest agent. Recommendation trust ensures that the evaluator can distinguish between those witnesses giving accurate opinions, when these are compared to the actual interaction with the target, if the recommendation is followed. However, low recommendation trust gives no indication of the reason behind the inaccuracy, whether it is only due to differing experiences or due to malicious intent.

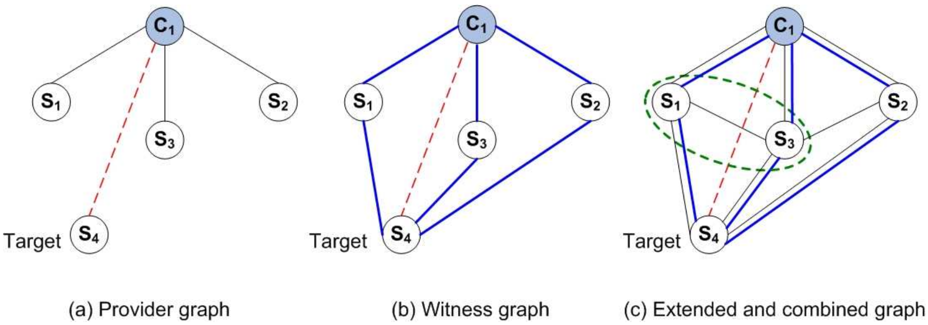


Fig. 2. Collusive behaviour between witnesses

Figure 2 shows an example of collusive behaviour among witnesses. The evaluator C_1 obtains recommendations about target S_4 from providers S_1, S_2, and S_3. Again, C_1 has had no past interactions with S_4. Figure 2(a) shows C_1's provider graph, with the solid lines representing direct service interactions and the dashed line indicates C_1's interest to interact with S_4. Figure 2(b) is different from Figure 1(b) as the recommendations obtained are all direct recommendations about S_4. The extended and combined graph, Figure 2(c) shows the additional information that the evaluator C_1 can infer from the trust and reputation information gathered. Frequent similarity of recommendations from S_1 and S_3, compared to other recommenders could suggest a potential case of collusion between these witnesses, especially if the opinions are inaccurate compared to the actual agent interaction. This is depicted by the dashed line circling S_1 and S_3 in Figure 2(c). Although S_2 and S_3 appear to have similar links as S_1 and S_3, the comparison of their recommendations helps determine that S_1 and S_3 are potentially collusive, while S_2 and S_3 are not considered in this category. Witnesses collude, for example, to lower the trustworthiness of the target as viewed by the evaluator to prevent the target from being swamped with interaction requests, which could potentially increase competition for the witnesses' to interact with the target as a supplier.

As part of the analysis of emergent data to detect collusion, Algorithm 2 outlines the partial collusion detection process after target a_β has just provided service s_β following recommendations. Initially, the set of potential colluders will include all the direct recommenders for target a_β about the service s_β. This set then needs to undergo further selection to ultimately obtain the smallest group of potential colluders. Based on this information, the evaluator can decide on subsequent interactions with the members of the suspected collusive group.

Algorithm 2 Partial Witness and Target Collusion Detection

```
for all direct recommendations $r^d$ do
    if ($r^d.a_t = a_\beta$) AND ($r^d.s = s_\beta$) then
        for all dimensions $d \in r^d.s$ do
            if $d_a < d_e$ then
                add $a_r$ to $\mathbb{P}\,colluders$
```

5 Conclusions and Future Work

In this paper, we have presented the component of our multi-agent network model, where agents build a network of their local environment. Using interaction data and recommendations, agents can maintain their own representation of their neighbourhood. Their local view is extended from the inferences that can be made from the trust and reputation information available. Whilst existing models mention using some form of social network without specifying how this is done, we go further and show how the network is built and maintained through provider and witness graphs.

We have also outlined the third component of our model, involving the analysis of the emergent network data. We have an implementation of the first two components, that is, the data collection and network building modules. Our ongoing work focuses on the analysis of the network data to extract useful information about agent relationships, in particular, those involving the detection of some forms of collusion. We believe that using available trust and reputation information as a way to learn more about the agent environment is a new approach to solving issues that are usually solved through global access to agent information. With an extended view to the individual agent neighbourhood, agents are closer to make informed decisions about issues that do not necessarily concern only its immediate neighbours.

Future research in the field could further explore the actions to be followed after emergent information has been discovered about the agent environment. For instance, following collusion detection, an evaluator can decide to incorporate the knowledge of collusive agents into its decision making regarding future interactions with the agents concerned.

References

1. Kautz, H., Selman, B., Shah, M.: Referral Web: Combining social networks and collaborative filtering. Communications of the ACM **40**(3) (1997)
2. Brown, J.J., Reingen, P.H.: Social ties and word-of-mouth referral behavior. Journal of Consumer Behaviour **14**(3) (1987) 350–362
3. Milgram, S.: The small world problem. Psychology Today **2** (1967) 60–67
4. Travers, J., Milgram, S.: An experimental study of the small world problem. Sociometry **32**(4) (1969) 425–443
5. Liben-Nowell, D., Kleinberg, J.: The link prediction problem for social networks. In: Proceedings of the 12th International Conference on Information and Knowledge Management. (2003) 556–559

6. Raghavan, P.: Social networks: From the Web to the enterprise. IEEE Internet Computing **6**(1) (2002) 91–94
7. Marsh, S.: Formalising Trust as a Computational Concept. PhD thesis, Department of Computer Science, University of Stirling (1994)
8. Huynh, T.D., Jennings, N.R., Shadbolt, N.: An integrated trust and reputation model for open multi-agent systems. Journal of Autonomous Agents and Multi-Agent Systems **13**(2) (2006) 119–154
9. Huynh, T.D., Jennings, N.R., Shadbolt, N.: Developing an integrated trust and reputation model for open multi-agent systems. In: Proceedings of the 7th International Workshop on Trust in Agent Societies, New York, USA (2004) 65–74
10. Yu, B., Singh, M.P.: Searching social networks. In: Proceedings of the 2nd International Joint Conference on Autonomous Agents and MultiAgent Systems (AAMAS 2003), ACM Press (2003) 65–72
11. Schillo, M., Funk, P., Rovatsos, M.: Using trust for detecting deceitful agents in artificial societies. Applied Artificial Intelligence, Special Issue on Trust, Deception, and Fraud in Agent Societies **14**(8) (2000) 825–848
12. Zacharia, G., Maes, P.: Trust management through reputation mechanisms. Applied Artificial Intelligence **14**(9) (2000) 881–907
13. Lian, Q., Zhang, Z., Yang, M., Zhao, B.Y., Dai, Y., Li, X.: An empirical study of collusion behavior in the Maze P2P file-sharing system. In: Proceedings of the 27th International Conference on Distributed Computing Systems (ICDCS 2007), IEEE Computer Society (2007) 56
14. Jurca, R.: Truthful Reputation Mechanisms for Online Systems. Phd thesis, 3955, Ecole Polytechnique Fédérale de Lausanne (2007)
15. Wang, J.C., Chiu, C.C.: Recommending trusted online auction sellers using social network analysis. Expert Systems with Applications **34**(3) (2008) 1666–1679
16. Wang, J.C., Chiu, C.C.: Detecting online auction inflated-reputation behaviors using social network analysis. In: Annual Conference of the North American Association for Computational Social and Organizational Science (NAACSOS 2005). (2005)
17. Srivatsa, M., Liu, L.: Securing decentralized reputation management using TrustGuard. Journal of Parallel and Distributed Computing **66**(9) (2006) 1217–1232
18. Lim Choi Keung, S.N., Griffiths, N.: Towards improved partner selection using recommendations and trust. In Falcone, R., et al., eds.: Trust in Agent Societies (TRUST 2008). Volume 5396 of Lecture Notes in Computer Science. Springer-Verlag Berlin Heidelberg (2008) 43–64
19. Lim Choi Keung, S.N., Griffiths, N.: Using recency and relevance to assess trust and reputation. In: Proceedings of AISB 2008 Symposium on Behaviour Regulation in Multi-Agent Systems. Volume 4., The Society for the Study of Artificial Intelligence and Simulation of Behaviour (2008) 13–18

Effects of expressiveness and heterogeneity of reputation models in the ART testbed: Some preliminary experiments using the SOARI architecture

Luis G. Nardin[1], Guillaume Muller[1], Anarosa A. F. Brandão[1], Laurent Vercouter[2], and Jaime S. Sichman[1]

[1] Laboratório de Técnicas Inteligentes - EP/USP
Av. Luciano Gualberto, 158 – trav. 3
05508-900 – São Paulo – SP – Brasil
{luis.nardin,anarosa.brandao,jaime.sichman}@poli.usp.br,
Guillaume.Muller@freesurf.fr
[2] École Nationale Supérieure des Mines de Saint-Étienne
158, cours Fauriel, 42023 Saint-Étienne Cedex 2, France
Laurent.Vercouter@emse.fr

Abstract. Trust and reputation have proved to help protect societies against harmful individuals. Inspired by these principles, many computational models have been and new ones continue to be proposed in the literature to protect multi-agent systems. In an open system, where few assumptions can be made on the internals of the agents, it is possible that different agents use different trust and reputation models. Since agents have to exchange information to make their trust and reputation models more robust, and since the models use different internal concepts and metrics, it is very important to consider the interoperability of these models. Based on experiments, this paper illustrates the usefulness of the SOARI architecture, which allows heterogeneous agents to interoperate more expressively about reputation.

1 Introduction

Agents present the capabilities of both acting autonomously and engaging in social activities. In open environments, where agents can enter or leave the environment at any time, taking part in such social activities may expose them to risks, for instance, when taking decisions based on information provided by malevolent agents. In order to avoid such risks, solutions based on trust models where implemented [5, 17, 6, 14, 13, 10]. Most of these models are based on the concept of reputation.
In order to accelerate the reputation evaluation and to improve the robustness of their reputation models, the agents generally exchange information about the reputation of third parties. However, since there is no consensus about a single unifying reputation definition, the semantics associated with reputation differs from one model to another. This semantic heterogeneity raises an interoperability problem among existing reputation models, which is addressed by SOARI [11] architecture.

In this paper, we present results of experiments where SOARI is used to enable interoperability of two reputation models: Repage [13] and L.I.A.R. [10]. These experiments evaluate the impact that the reputation models interoperability may cause on agents evaluation accuracy. More specifically, this paper answers the following two questions: (1) is there any improvement in the reputation evaluation accuracy when enabling a more expressive communication? (2) How does the heterogeneity influence the evaluation accuracy of the dishonest agents' reputation?

The rest of the document is organised as follows. Section 2 presents briefly the platforms used to run the experiments (ART and FOREART testbeds) as well as the SOARI architecture. In Section 3, the results and analysis of the experiments are shown. Finally, our conclusions and future work are presented in Section 4.

2 Background Work

The ART testbed (Agent Reputation and Trust testbed) [7] is currently the unique platform freely available to perform benchmarks with heterogeneous reputation models. We first briefly present its scenario, because it is the basis for the experiments. However, this platform does not allow the agents to communicate about reputation using their distinct semantically reputation model concepts, thus losing expressiveness. The FOREART testbed [3], which is an extension of the ART testbed, allows a more expressive communication among the agents. In order to reach this goal, this latter platform uses FORE (Functional Ontology of Reputation) [4] as a common vocabulary. In this platform, interoperability is obtained by translating concepts from a source model (expressed in ontological terms) to concepts of FORE, and then by translating the result from FORE into concepts of a target model (also expressed in ontological terms). The SOARI architecture is then used to implement the FOREART testbed's agents thus enabling a more expressive communication about reputation among them. The resulting platform is the basis of the experiments described in the next section.

2.1 The ART testbed

In AAMAS'04 TRUST workshop, it was admitted that the diversity in the internals and metrics employed by current models of trust and reputation made it difficult to establish objective benchmarks. In order to design a testbed platform to enable comparison, the ART testbed initiative was launched.

The resulting testbed platform (programmed in Java) simulates an art appraisal game, where agents evaluate paintings for clients and gather opinions and reputations from other agents to produce accurate appraisals. More precisely, a game proceeds as a series of the following time steps[3]: (i) the platform assigns clients (i.e. paintings) to each appraiser. Appraisers receive larger shares of clients (thus larger amount of money) if they have produced more accurate appraisals in the past; according to the era each painting belongs to, an appraiser is more or less accurate in its evaluations; (ii) reputation transactions occur, where appraisers can exchange reputation information about third parties for given eras; (iii) certainty transactions occur, where appraisers can exchange

[3] Those time steps refer to ART testbed platform used on the competition of 2008, which implements slightly different time steps sequence than the one of the previous years' platform, described in [8].

how certain they are about a specific era; (iv) opinion transactions occur, where appraisers can exchange expert opinions about a specific painting; (v) finally, the appraisers are required to send weights to the platform; those weights represent the intensity with which the appraiser considers the opinion of each other appraiser; the platform then computes the final appraisal of each appraiser as a weighted mean of the opinions it has purchased; this step ensures the same computation for everybody, therefore, (a) only the trust models are evaluated (not the expertise in art) and (b) cheating is impossible. The winner of the game is the agent that has the higher final bank balance.

In this scenario, the need for reputation modelling comes from the duality of the need for cooperation to evaluate some of the paintings (because the agents are only competent in some eras) and the competition to earn the biggest part of the client pool.

More details about the ART testbed can be found in [8].

2.2 The FOREART testbed

The interaction which involves the reputation transaction is a moment where agents have to exchange information from their reputation models, meaning that interoperability among reputation models is required. In the current version of the platform, interoperability is obtained by asking the developers of each agent to map their reputation model evaluations into a single value in the domain [0:1]. This common model is too simple and the mapping of complex internal reputation models into a simplistic one results in loss of expressiveness and details. It is thus impossible to perform finer agent interactions about reputation.

The addition of semantic data to this common model may improve the agent performance during the process of reputation building, while allowing interoperability between different reputation models. Therefore, the FOREART testbed platform was implemented as an extension of ART by modifying its engine to allow the exchange of messages related to reputation transactions that involve semantic content. The messages' content is a string (instead of couples *(agent, painting era)* or numerical value) and it is expected that this string is queries and answers written in an ontology query language. The chosen query language is related to the inference engine that is used to reason about the queries. The first version of FOREART uses nRQL [9] and Racer [12]. Nonetheless, FOREART agents were implemented according to the general agent architecture proposed to support reputation interaction with semantic content [16]. The general architecture main modules are the Interaction Module (IM), the Reputation Mapping Module (RMM) and the Reputation Reasoning Module (RRM). These modules are responsible for dealing with the translation between FORE and the agent internal reputation model expressed as ontology, and the reasoning about exchanged messages.

More information on this platform can be found in [15, 16, 3, 2].

2.3 SOARI: Service Oriented Architecture for Reputation Interaction

However, because of some drawbacks of the general agent architecture [16], the SOARI architecture was proposed (Figure 1). The SOARI is a service-oriented architecture to support the semantic interoperability among agents that implement heterogeneous reputation models. The main underlying idea of SOARI is that the mapping between different ontologies (by using FORE as an interlingua) may be realised off-line, and be available on-line as a service for the agents that use the same reputation model. Hence, it extends the FOREART agent architecture in two ways: (i) it subdivides the Reputation Mapping Module (RMM) in two distinct and specialised modules: the Ontology Mapping Service (OMS) and the TRANSLATOR module (in grey in the figure), and (ii) it performs the ontology mapping and translation functions as a service outside the agent architecture.

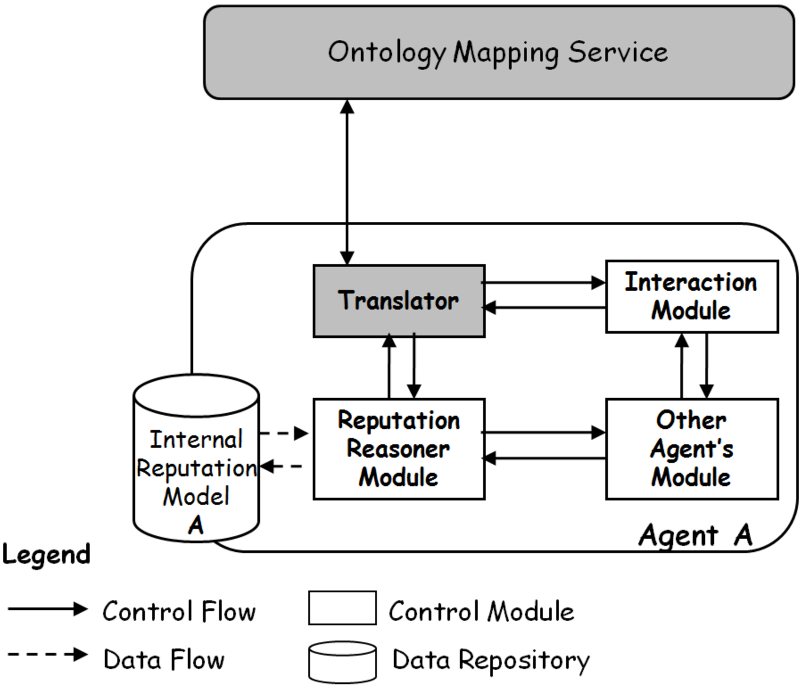


Fig. 1. Service Oriented Architecture for Reputation Interoperability

The OMS module is a service outside the agent that implements the mapping and translation ontology functions and presents two main functionalities: (i) to map concepts from the target's reputation model ontology to the concepts of the common ontology; and (ii) to answer concept translation requests from the TRANSLATOR module.
The TRANSLATOR module resides inside the agent and it translates reputation messages. It has four main activities: (i) to translate the reputation messages from the common ontology to the target agent's reputation model ontology whenever the message comes from the Interaction Module (IM); (ii) to translate the reputation messages from

the agent's reputation model ontology to the common ontology whenever the message is sent to IM; (iii) to trigger some function in the Reputation Reasoning Module (RRM) based on the interpretation of messages written using the reputation model ontology; and (iv) to create a message using the reputation model ontology whenever requested by RRM.

More information on this architecture can be found in [11].

3 Experiments

This section intends to answer two questions: (1) is there any improvement in the accuracy of the agents' reputation evaluation when enabling more expressive communication about reputation? (2) how does the heterogeneity influence the accuracy of dishonest agent's reputation evaluation?

In order to answer those questions, some experiments were performed using the ART and FOREART testbeds and the SOARI architecture. In those experiments, one agent deliberately lies about the other agents' reputation and about paintings evaluation. The analysis was performed to determine how accurate the other agents are in the evaluation of the reputation of the liar agent.

In a practical point of view, all the experiments were performed using a modified FOREART testbed. In the remaining, the term ART thus refers to situations when the reputation communication among the agents is limited to numeric (numeric communication) and FOREART when it is performed using strings (symbolic communication).

The experiments include two types of agents: *Honest* and *Dishonest*. The *Honest* agents answer to the requests only when they have expertise about the requested painting era and with information coherent to their internal state. The *Dishonest* agents answer to all the requests, even when they do not have expertise about that painting era and they never answer the requests with information coherent to their internal state.

3.1 Agent Model

The agent models in the testbed platforms are implemented by extending the abstract *Agent* class and filling up the methods that describe the agent's behaviour [8]. These methods correspond globally to the steps described on section 2.1.

In the begin of each time step, a set of paintings is assigned to the agent for appraisal. For each painting assigned, the agent performs reputation transactions. First, it requests to other agents in the testbed platform the reputation of possible appraisers of that painting. Then, it answers to reputation requests received from other agents. If it is a *Dishonest* agent, it accepts all the requests. Otherwise, it accepts the requests only if it has the expertise higher than a predefined expertise threshold ($expertisethreshold = 0.7$). To all the accepted requests, the agent answers with a reputation value, which does not reflect its internal reputation evaluation if it is a *Dishonest* agent.

After performing the reputation transaction, the agent performs certainty transactions. It first selects a group of agents and requests to them their certainty about a specific painting era. In the sequence, it answers to certainty requests received from other agents. If it is an *Honest* agent and its expertise is higher than a predefined expertise threshold ($expertisethreshold = 0.7$), it answers with its expertise value. However, if it is a *Dishonest* agent, it answers with the maximum between 1 and its expertise value plus 0.5.

After performing the certainty transactions, the agent requests the opinion of the agents it trusts (i.e. which reputation value is higher than a trust threshold that in Repage is $Image >= 0.5$ and/or $Reputation >= 0.8$, and L.I.A.R. is $X >= 0.7$, where $X = \{DIbRp,\ IIbRp,\ ObsRcbRp,\ EvRcbRp\ or\ RpRcbRp\}$) or the agents from which it received a certainty value higher than a predefined certainty threshold ($certaintythreshold = 0.5$).

Finally, in order for the simulator to compute the opinions, the agents provide to it the weight of each opinion provided by the other agents.

3.2 Experiments Description

The main objective of these experiments was to identify the mean value of the reputation assigned by the *Honest* agents to the *Dishonest* agent. In order to enable comparison between the experiments, the initial painting era knowledge and clients distribution were identical in all the experiments. Moreover, all the agents used the same configuration parameters (Table 1) and agent model (see Section 3.1) in all the simulations.

To reach this goal, we considered the execution of 10 simulations ($p = 10$) for each experiment with 100 cycles each. Each simulation was composed of 11 agents ($n = 11$), where 10 agents were *Honest* and 1 agent was *Dishonest* ($i = [1, 10]$ and $j = 11$). The mean value of the reputation assigned to the *Dishonest* agent by each *Honest* agent (r_j) considered only the value obtained in the last simulation cycle ($l = 100$ and $m = 100$). The value of the last simulation cycle was used because we considered it the most accurate reputation evaluation.

Table 1. Testbed's configuration parameters

Parameter	Value
averageClientsPerAgent	4
numberOfPaintingEras	20
cp_opinionCost	10
cp_certaintyCost	2
f_clientFee	100
nb_certaintyMsg	20
nb_opinionMsg	5

Formally, consider a set of n agents, where $i = \{1, 2, \ldots, n-1\}$ are *Honest* agents and $j = n$ is a *Dishonest* agent. Moreover, consider that r_{ij}^{sk} is the reputation value assigned by the agent i to the agent j in cycle k on simulation s. Typically, the reputation value assigned by agent i to agent j on simulation s corresponds to the mean reputation value of a set of cycles. Thus, $r_{ij}^{s} = \frac{\sum_{k=l}^{m} r_{ij}^{sk}}{m - l + 1}$, where l and m represents, respectively, the lower and upper cycle limits. The mean reputation value assigned by the *Honest* agents to the *Dishonest* agent on simulation s is $r_{j}^{s} = \frac{\sum_{i=1}^{n-1} r_{ij}^{s}}{n-1}$. Finally, given a set of simulations $s = 1, \ldots, p$ that compose an experiment, the mean value of the *Dishonest* agent is $r_j = \frac{\sum_{s=1}^{p} r_{j}^{s}}{p}$.

The experiments performed were classified based on two dimensions: (1) reputation models used by the agents in the experiment (Repage, L.I.A.R. or both), and (2) reputation communication method (numeric or symbolic) (Table 2). Moreover, the mixed experiments are split in two others based on the reputation model of the *Dishonest* agent. This distinction is indicated by the *D/L.I.A.R.* and *D/Repage* suffix in the experiment's name. In the other experiments, the *Dishonest* agent uses the same reputation model than the *Honest* agents.

Table 2. Summary of experiments

ID	Experiment name	Reputation Model	Reputation Communication
exp1	ART/L.I.A.R.	L.I.A.R.	Numeric
exp2	ART/Repage	Repage	Numeric
exp3.1	ART/Mixed-D/L.I.A.R.	L.I.A.R. and Repage	Numeric
exp3.2	ART/Mixed-D/Repage	L.I.A.R. and Repage	Numeric
exp4	FOReART/L.I.A.R.	L.I.A.R.	Symbolic
exp5	FOReART/Repage	Repage	Symbolic
exp6.1	FOReART/Mixed-D/L.I.A.R.	L.I.A.R. and Repage	Symbolic
exp6.2	FOReART/Mixed-D/Repage	L.I.A.R. and Repage	Symbolic

3.3 Experiments Results and Analysis

Here, we present an analysis of the results obtained from the experiments in order to answer the two questions posed at the beginning of this section. The complete raw results data can be obtained at http://www.lti.pcs.usp.br/results.pdf. The analysis methodology used to answer the questions raised on this section is based on the Student's T-Test [1].

The analysis performed on this section was based on the L.I.A.R. and Repage reputation models attributes. For reputation model attribute, we mean the different concepts of reputation defined in each reputation model. The L.I.A.R. reputation model defines five different types of reputation: *Direct Interaction-based Reputation* (DIbRp); *Indirect Interaction-based Reputation* (IIbRp); *Observation Recommendation-based Reputation* (ObsRcbRp); *Evaluation Recommendation-based Reputation* (EvRcbRp); and *Reputation Recommendation-based Reputation* (RpRcbRp). Further details about those types of reputation can be obtained in [10].
The Repage reputation model defines two reputation concepts: *Image* and *Reputation*. Further details about those can be obtained in [6].

Effect of the expressiveness of communication. In order to analyse the effects of the more expressive communication, it was verified if the mean value of the *Dishonest* agent's attributes (r_j) obtained on the numerical experiments (ART experiments) were higher than the similar ones obtained on the symbolic experiments (FOREART experiments). If so, then it means that the *Dishonest* agent was better identified in the symbolic experiments than in the numerical experiments. Thus, using Student's T-Test, a set of hypotheses was required to demonstrate it. The general form of the hypotheses is:

The mean value of the reputation model attribute from ART experiments is higher than the same attribute's mean value from the FOREART experiments. This hypothesis, from the point of view of the reputation model attribute is expressed mathematically as $Q^X_{ART} > Q^X_{FOReART}$, where X is a L.I.A.R. or Repage reputation model attribute.
In order to validate this hypothesis using the Student's T-Test, the following test is performed:
$H0 : Q^X_{ART} <= Q^X_{FOReART}$
$H1 : Q^X_{ART} > Q^X_{FOReART}$

The complete set of hypotheses to demonstrate the effects of the more expressive communication are presented on Table 3.

Table 3. Expressiveness hypotheses

Hypothesis	Reputation Model	Attribute
A	L.I.A.R.	DIbRp
B	L.I.A.R.	IIbRp
C	L.I.A.R.	RpRcbRp
D	Repage	Image
E	Repage	Reputation

When applied to the results of the following pairs of experiments: (exp1, exp4), (exp2, exp5), (exp3.1, exp6.1), (exp3.1, exp6.2), (exp3.2, exp6.1) and (exp3.2, exp6.2), considering the risk level (α) of 0.01 and the degree of freedom of 18, those hypotheses generate the results presented in Table 4 (✔ means that $H0$ was rejected, which confirms the hypothesis; ✘ means that $H0$ was not rejected, thus the hypothesis cannot be confirmed; and $-$ (dash) means that the hypothesis is not applicable for the pair of experiments).

Table 4. Expressiveness hypotheses result

Pair	Hypotheses				
	A	B	C	D	E
(exp1, exp4)	✘	✘	✘	-	-
(exp2, exp5)	-	-	-	✔	✔
(exp3.1, exp6.1)	✘	✘	✘	✔	✔
(exp3.1, exp6.2)	✘	✘	✘	✔	✔
(exp3.2, exp6.1)	✘	✘	✘	✘	✔
(exp3.2, exp6.2)	✘	✘	✘	✘	✔

Analysing the information in Table 4, we can verify that in most of the cases the hypotheses D and E reject the $H0$ (indicated by ✔) confirming those hypotheses, while the hypotheses A, B and C do not (indicated by ✘). From the reputation model point of view, the hypotheses D and E are associated to the Repage reputation model ($Image$ and $Reputation$ attributes), while the hypotheses A, B and C are associated to the L.I.A.R. reputation model ($DIbRp$, $IIbRp$ and $RpRcbRp$ attributes). Therefore, we can conclude that a more expressive communication about reputation has a positive effect in the accuracy of the reputation evaluation to agents that use the Repage reputation model. However, it was not possible to infer that the more expressive communication benefits or harms the agents that use the L.I.A.R. reputation model.
Based on these results, we conclude that the Repage reputation model has some intrinsic or some implementation characteristics that enables it to benefit from the more expressive communication.

Effect of the reputation model heterogeneity. The analysis of the effect of reputation model heterogeneity was performed by testing if the mean value of the *Dishonest* agent's reputation model attributes (r_j) obtained on experiments with homogeneous reputation model were higher than the similar ones obtained on mixed experiments. Thus, to demonstrate it using Student's T-Test a set of hypotheses was required. The general form of the hypotheses is:

The mean value of the reputation model attribute from experiments with homogeneous reputation model is higher than the same attribute's mean value from mixed experiments. This hypothesis, from the point of view of the reputation model attribute is expressed mathematically as $Q^X_{P/M} > Q^X_{P/Mixed}$, where M is the reputation model (L.I.A.R. or Repage), X is its attribute and P is the testbed platform (ART or FOREART).

In order to validate this hypothesis using the Student's T-Test, the following test is performed:

$H0 : Q^X_{P/M} <= Q^X_{P/Mixed}$
$H1 : Q^X_{P/M} > Q^X_{P/Mixed}$

The complete set of hypotheses to demonstrate the effects of heterogeneous reputation models are presented on Table 5.

Table 5. Heterogeneous hypotheses

Hypothesis	Reputation Model	Attribute	Platform
F	L.I.A.R.	DIbRp	ART
G	L.I.A.R.	IIbRp	ART
H	L.I.A.R.	RpRcbRp	ART
I	Repage	Image	ART
J	Repage	Reputation	ART
K	L.I.A.R.	DIbRp	FOREART
L	L.I.A.R.	IIbRp	FOREART
M	L.I.A.R.	RpRcbRp	FOREART
N	Repage	Image	FOREART
O	Repage	Reputation	FOREART

When applied to the results of the following pairs of experiments: (exp1, exp3.1), (exp1, exp3.2), (exp2, exp3.1), (exp2, exp3.2), (exp4, exp6.1), (exp4, exp6.2), (exp5, exp6.1) and (exp5, exp6.2), considering the risk level (α) of 0.01 and the degree of freedom of 18, those hypotheses generate the results presented in Tables 6 and 7 (✔ means that $H0$ was rejected, which confirms the hypothesis; ✘ means that $H0$ was not rejected, thus the hypothesis cannot be confirmed; and $-$ (dash) means that the hypothesis is not applicable for the pair of experiments).

Table 6: Hypotheses result ART

Pair	Hypotheses				
	F	G	H	I	J
(exp1, exp3.1)	✘	✘	✘	-	-
(exp1, exp3.2)	✘	✘	✘	-	-
(exp2, exp3.1)	-	-	-	✘	✘
(exp2, exp3.2)	-	-	-	✘	✘

Table 7: Hypotheses result FOREART

Pair	Hypotheses				
	K	L	M	N	O
(exp4, exp6.1)	✘	✘	✔	-	-
(exp4, exp6.2)	✘	✘	✘	-	-
(exp5, exp6.1)	-	-	-	✘	✘
(exp5, exp6.2)	-	-	-	✘	✘

Analysing the Tables 6 and 7, we can infer that in most of the cases the hypotheses did not reject $H0$ (indicated by ✘). This leads us to the conclusion that reputation model heterogeneity does not have any effect on the accuracy of the *Dishonest* agent reputation evaluation.

4 Conclusions

In this paper we presented some experiments using the SOARI architecture integrated into the FOREART testbed. Those experiments were performed to answer two questions: (1) is there any improvement in the reputation evaluation accuracy when enabling a more expressive communication? and (2) how does the heterogeneity influence the evaluation accuracy of the dishonest agents' reputation? The results obtained do not allow us to conclude that a more expressive communication about reputation or reputation model heterogeneity provides an accurate reputation evaluation of other agents. However, the results have shown the Repage reputation model benefits from the symbolic communication, which leads us to think that there are some intrinsic or implementation model's characteristics that provided it.

Since those were some preliminary experiments, the results obtained may be related to the fact that the experiments may not be the ideal ones to assess the effects of communication expressiviness and reputation model heterogeneity. Therefore, as a future work, we intend to design better experiments using and not using the ART and FOREART testbeds.

Moreover, we intend to perform experiments using more and different reputation models, thus expanding the analysis related to the effects of heterogeneity on the accuracy of reputation evaluation. Based on those results, we expect to have enough information to perform a detailed analysis to identify the relationship between the reputation models characteristics and the benefits of using the SOARI architecture.

Acknowledgements

This project is partially supported by FAPESP/Brazil. Guillaume Muller is supported by CNPq/Brazil under grant 382737/2008-3. Anarosa A. F. Brandão is supported by CNPq/Brazil under grant 310087/2006-6. Laurent Vercouter was partially supported by FAPESP/Brazil under grant 2005/02902-5. Jaime S. Sichman is partially supported by CNPq/Brazil.

References

1. S. Boslaugh and P. A. Watters. *Statistics in a Nutshell*. O'Reilly Media, Inc., 2008.
2. A. A. F. Brandão, L. Vercouter, S. J. Casare, and J. Sichman. Exchanging reputation values among heterogeneous agent reputation models: an experience on ART testbed. In *AAMAS '07: Proceedings of the* 6^{th} *International Joint Conference on Autonomous Agents and Multi-Agent Systems*, pages 1–3, New York, NY, USA, 2007. ACM Press.

3. A. A. F. Brandão, L. Vercouter, S. J. Casare, and J. S. Sichman. Extending the art testbed to deal with heterogeneous agent reputation models. In C. Castelfranchi, S. Barber, J. Sabater, and M. Singh, editors, *Proceedings of the* 10^{th} *Workshop on Trust in Agent Societies*, Honolulu, Hawaii, 2007.
4. S. Casare and J. S. Sichman. Using a functional ontology of reputation to interoperate different agent reputation models. *Journal of the Brazilian Computer Society*, 11(2):79–94, 2005.
5. C. Castelfranchi and R. Falcone. Principles of trust in mas: Cognitive anatomy, social importance, and quantification. In *ICMAS '98: Proceedings of International Conference on Multi-Agent Systems*, pages 72–79, Washington, USA, 1998. IEEE Computer Society.
6. R. Conte and M. Paolucci. *Reputation in Artificial Societies. Social Beliefs for Social Order*. Kluwer, Boston, 2002.
7. K. K. Fullam, T. B. Klos, G. Muller, J. Sabater-Mir, A. Schlosser, Z. Topol, K. S. Barber, J. Rosenschein, L. Vercouter, and M. Voss. A specification of the Agent Reputation and Trust (ART) testbed: Experimentation and competition for trust in agent societies. In F. Dignum, V. Dignum, S. Koenig, S. Kraus, M. P. Singh, and M. Wooldridge, editors, *AAMAS '05: Proceedings* 4^{th} *International Conference on Autonomous Agents and Multi-Agent Systems*, pages 512–518. ACM Press, Utrecht, The Netherlands, July 2005.
8. K. K. Fullam, T. B. Klos, G. Muller, J. Sabater-Mir, Z. Topol, K. S. Barber, J. S. Rosenschein, and L. Vercouter. The Agent Reputation and Trust (ART) testbed architecture. In C. Castelfranchi, K. S. Barber, J. Sabater-Mir, and M. P. Singh, editors, *Proceedings* 8^{th} *Workshop on Trust in Agent Societies*, pages 50–62, Utrecht, The Netherlands, July 2005.
9. V. Haarslev, R. Möller, and M. Wessel. Querying the semantic web with racer + nrql. In *ADL '04: Proceedings of the KI-2004 International Workshop on Applications of Description Logics*, 2004.
10. G. Muller and L. Vercouter. L.i.a.r. achieving social control in open and decentralised multi-agent systems. Technical Report 2008-700-001, École Nationale Supérieure des Mines de Saint-Étienne, Saint-Étienne, France, 2008.
11. L. G. Nardin, A. A. F. Brandão, J. S. Sichman, and L. Vercouter. *SOARI: A Service Oriented Architecture to Support Agent Reputation Models Interoperability*, volume 5396 of *Lecture Notes in Computer Science*. Springer, Heidelberg, Germany, 2008.
12. Racer Systems GmbH and Co. KG, Hamburg, Germany. *RACERPro User's Guide Version 1.9.2*, 2007.
13. J. Sabater-Mir, M. Paolucci, and R. Conte. Repage: Reputation and image among limited autonomous partners. *Journal of Artificial Societies and Social Simulation*, 9(2), 2006.
14. J. Sabater-Mir and C. Sierra. Social regret, a reputation model based on social relations. *SIGecom Exch.*, 3(1):44–56, 2002.
15. L. Vercouter, S. J. Casare, J. S. Sichman, and A. A. F. B. ao. An experience of reputation models interoperability using interactions based on a functional ontology. In *Iberagents 2006*, Ribeirão Preto, SP, Brazil, 2006.
16. L. Vercouter, S. J. Casare, J. S. Sichman, and A. Brandão. An experience on reputation models interoperability based on a functional ontology. In *IJCAI '07: Proceedings of the* 20^{th} *International Joint Conference on Artificial Intelligence*, pages 617–622, Hyderabad, India, 2007.
17. G. Zacharia and P. Maes. Trust management through reputation mechanisms. *Journal of Applied Artificial Intelligence*, 14(9):881–907, 2000.

Towards the Definition of an Argumentation Framework using Reputation Information

Isaac Pinyol and Jordi Sabater-Mir

IIIA - Artificial Intelligence Research Institute
CSIC - Spanish Scientific Research Council
Bellaterra, Barcelona, SPAIN
{ipinyol,jsabater}@iiia.csic.es

Abstract. Reputation mechanisms have been recognized one of the key technologies when designing multi-agent systems. They are specially relevant in complex open environments, becoming a non-centralized mechanism to control interactions among agents. Agents tackling such complex societies must use reputation information not only for selecting partners to interact with, but also in dialog processes, like negotiation or persuasion. Some of these processes rely on *arguments* that support agent's points of view. This is the focus of this paper. Taking as a base Repage, a cognitive reputation model, and its integration in a BDI architecture, we highlight on the necessary elements to build an argumentation framework (AF) that includes reputation information. We propose a general AF that allows graded influence among arguments, and we instantiate it in the particular case of Repage. Finally, to show the potential of the framework we illustrate a possible exchange of arguments between two agents where one of them is seeking for new information.

1 Introduction

Reputation mechanisms have been recognized one of the key technologies when designing multi-agent systems (MAS) [1]. In this relatively new paradigm, reputation models have been adapted to confront the increasing complexity that open multi-agent environments bring. Thus, the figure of agents endowed with their own private reputation model, takes special relevance as a non-centralized mechanism to control interactions among agents. Following this line, cognitive agents using cognitive reputation models arise as one of the most complete and generic approaches when facing very complex societies. Usually, cognitive agent's architectures, like BDI (*Belief, Desire, Intention*), follow logic-based reasoning mechanisms, providing then a high flexibility and theoretically well-founded reasoning.

Such complex agents must use reputation information not only for selecting partners to interact with, but also in the dialog process itself. For instance, in negotiation processes where it is important to establish and defend certain position, reputation information may be useful to strength such position by *justifying* it. But at the same time these reputation values can be justified as well. Argumentation frameworks (AF) have been proved to be useful when facing these kinds of dialogs in MAS ([2];[3]). Usually an argument is a set of *elements* of a concrete knowledge base (for instance, the set of an agent's attitudes) from which can be deduced another element that wants to be justified. An AF provides a formal framework to relate arguments among each other.

We deal with these issues by proposing a roadmap towards an argumentation framework that includes Repage information. Repage is a reputation system based on a cognitive theory of reputation that has been used in logical BDI reasoning processes [4] (BDI+Repage), offering then an integrated reasoning framework. In this paper, we focus on how reputation information can be included in argumentation frameworks, hinting at a global and integrated AF. We take advantage of the facilities that multicontext systems (MCS)[5] specifications offer. The construction of arguments is done by considering the mutlicontext specification of the BDI+Repage model, following the approach defined in [2], and by defining the appropriate influence relations among these arguments.

In section 2 we briefly introduce some preliminary concepts related to argumentation frameworks, multicontext systems and how to use them to build AF. Also, we introduce a new generic argumentation framework to deal with graded influences among arguments that will be useful when defining Repage arguments. In section 3 we specify Repage as a MCS to define the set of possible arguments. In the same section we specify how these arguments attack and support each other. In section 4 we put our AF to work by stating a simple example. Finally we conclude in section 5 with the conclusions and future work.

2 Abstract Argumentation Frameworks

An argumentation system gives a formal framework to reason over a knowledge base with possible inconsistent information. Many formalisms have appeared in literature to deal with argumentation ([6];[2];[7];[3]). Dung defines in [6] an abstract argumentation system as follows:

Definition 1. *An argumentation system is a tuple $AF = \langle A, R \rangle$, where A is a set of arguments and R is a binary attack relation where $R \subseteq A \times A$.*

So, let $\alpha, \beta \in A$ be two arguments, if $\alpha R \beta$ holds we say that the argument α attacks argument β. From this abstract definition, several categories of argumentation frameworks have appeared. In this case we interested in epistemic arguments, because they are built under consequence relations and can be easily extended to logical consequences, which is a key point in logic-based reasoning.

2.1 Epistemic Arguments

This category of arguments is constructed over a knowledge base with a consequence relation. Taking the formal definition proposed in [7]:

Definition 2. *An epistemic argument α is a tuple $\alpha = \langle B, b \rangle$ such that $B \subseteq K$ where K is the knowledge base, and $B \vdash b$ where $\vdash$ is a suitable consequence relation defined for K.*

Also, we can define the abstract argumentation system for epistemic arguments as follows:

Definition 3. *Let K be a knowledge base, we define the argumentation framework $AF_K = \langle A_K, R_K \rangle$ where A_K is the set containing all possible consistent epistemic arguments, and R_K the attack relation on arguments of A_K.*

Then, to construct an argumentation framework it is only necessary to decide *which* arguments are possible, and how they *attack* each other. Regarding this last issue, several kinds of *attack* have been defined: (1) **rebuttal** referring to arguments with contradictory conclusions, (2) **assumption** where the conclusions of an argument contradicts a premise of the other, and (3) **undercut**, where the conclusion of an argument contradicts an inference rule used in the other argument.

2.2 Extending Argumentation Frameworks

The abstract argumentation systems presented above lacks in some important features if we consider underlying languages whose propositions have a fuzzy or probabilistic interpretation. On the one hand, if the knowledge base has no boolean formulas, the definition of contradictory formulas and in extension, the idea of attack may be somehow more difficult to define. On the other hand, the previous definition does not take into account arguments that *support* other arguments. This concept is very useful when the argumentation framework wants to be used in multi-agent communication.

To cover these necessities we define an extended argumentation framework that subsumes the previous one. The idea is to redefine the binary attack relation by means of a binary function. Thus, the semantics of attack is replaced by the semantics of *influence* in certain degree. The degree must capture both the semantics of attack and support. Formally,

Definition 4. *An extended argumentation system is a tuple $AFe = \langle A, G \rangle$, where A is a set of arguments and G is the binary influence function such that $G : A \times A \to M$, where M is a partially ordered set defining a lattice.*

Examples of M could be $[-1, 1] \subset I\!R$ or $\{attack, none, support\}$ where $attack \leq none \leq support$. For instance, assuming that $M = [-1, 1]$ we could consider that 1 is the maximum support, -1 is the maximum attack and 0 is the non-influence value. In this example, let $\alpha, \beta \in A$, if $G(\alpha, \beta) = 0.3$ indicates that α influences β in a degree of 0.3, or following the defined semantics, that α supports β in a degree of 0.3.

Notice that this definition subsumes the previous one. It offers more capabilities without losing any of the previous properties. Formal definitions and properties regarding this framework should be carefully studied by reformulating the original Dung's framework. We plan to investigate it in the future.

In the next section we show how arguments can be generated using multicontext systems, which is the framework we use in this paper. From now on, when we use the term argumentation framework (AF) we are referring to the extended version.

2.3 Multicontext Systems and Argumentation

In this section we introduce the notion of multicontext system and how argumentation frameworks can be built using them.

Multicontext systems (MCS) provide a framework to allow several distinct theoretical components to be specified together, with a mechanism to relate these components [5]. These systems are composed of a set of contexts (or units), and a set of bridge rules. Each context can be seen as a logic and a set of formulas written in that logic. Bridge rules are the mechanisms to infer information from one context to another.

Giunchiglia and Serafini [5] proposed the following formalization of MCS: Let I be the set of context names, a MCS is formalized as $\langle \{C_i\}_{i \in I}, \triangle_{br} \rangle$:

- $C_i = \langle L_i, A_i, \triangle_i \rangle$, where L_i is a formal language with its syntax and semantics, A_i is a set of axioms and $\triangle_i$ the set of inference rules. Thus, L_i and A_i define an axiomatic formal system, a logic for the context C_i. Beside axioms, it is possible to include a theory T_i as predefined knowledge. All A_i, $\triangle_i$ and T_i are written in the language L_i.
- $\triangle_{br}$ is a set of bridge rules.

Bridge rules can be seen as inference rules among contexts. Each one has a set of antecedents (or preconditions) and a consequent (or postcondition). When each formula in the antecedent is true in its respective context, the consequent becomes true as well (also in its context). A bridge rule is graphically represented as follows:

$$\frac{C_{i_1} : \varphi_1, \ldots, C_{i_n} : \varphi_n}{C_{i_x} : \varphi_x}$$

where $C_{i_k} \varphi_k$ indicates that formula φ_k belongs to the context C_{i_k}, formulas $\varphi_1 \ldots \varphi_n$ are the antecedents and φ_x is the consequent.

We follow the approach given by Parsons *et al.* in [2] to define argumentation systems using MCS. In this approach, an argument is a set of deductive steps. Each deductive step is an expression $\Gamma \vdash_d \varphi$ with $d = \{s_1, \ldots, s_t\}$, meaning that the formula φ is deduced by agent i from the set of formulas Γ using the inference rules $s_1, \ldots, s_t$. Thus, an argument will be composed of a set of deduction expressions (as grounds for the argument). More formally and paraphrasing [2]:

Definition 5. *An argument for the formula φ is a pair $\langle \mathcal{P}, \varphi \rangle$ where $\mathcal{P}$ is a sorted set of grounds $\{g_1, \ldots, g_l\}$ such that for $1 \leq k \leq l$:*

1. $g_l = \Gamma_l \vdash_{d_l} \varphi$
2. *for every g_j where $j \leq l$ where*
   - $g_j = \Gamma_j \vdash_{d_j} \Upsilon_j$ *such that every element $e \in \Gamma_j$ is in the theory of the agent or is Υ_h where $h < j$* ***or***
   - $g_j = \Upsilon_j$ *and Υ_j is in the theory of the agent.*

Notice that this kind of arguments can be seen as epistemic arguments. From now on we will use the previous definition when referring to AF. Multicontext systems then, can be used to specify how arguments can be generated, since the behavior of bridge rules is somehow equivalent to inference rules. Finally, to fully define an AF, it is also necessary to define the binary *influence* function G, which is totally context-dependent.

Once all the components are specified as a MCS, it is possible to construct arguments following the previous definition to support, for instance, the actions that agents perform in terms of intentions, desires and beliefs, but also, in terms of the internal elements of the reputation model. In the following section, we explain the Repage system and one possible MCS specification for it.

3 Argumentation for Repage System

In this section we focus on argumentation issues for Repage predicates. The idea is to specify Repage as a multicontext system, and use it to build arguments on reputation information, using the generic argumentation framework for multicontext systems defined in section 2.3. The original Repage architecture defined at [8] has already a modular specification that makes easy this step.

3.1 The Repage System

Regae is a computational system based on the cognitive theory of reputation described in [9]. This theory describes a model of imAGE, REPutation and their interplay. Although both are social evaluations, image and reputation are distinct objects. Image is a simple evaluative belief; it tells that a target agent is *good* or *bad* with respect to a norm, a standard, or a skill. Reputation is a belief about the existence of a communicated evaluation. Consequently, to assume that a target j is assigned a given reputation implies only to assume that j is reputed to be *good* or *bad*, i.e., that this evaluation circulates, but it does not imply to share the evaluation.

The Repage architecture is composed of a set of elements. The relevant one for this paper is the memory, which stores information in terms of predicates.

In the memory, predicates are conceptually organized in distinct levels of abstraction and inter-connected. Each predicate that belongs to one of the main types (*image, reputation, shared voice, shared evaluation, valued communication* and *outcome*) contains an evaluation that refers to a certain agent in a specific role. The value associated to a predicate as a tuple of five positive values (summing to one and representing a probability distribution), that we call $weights$, plus a strength value: $\{w_1, \dots, w_5, s\}$. Each value has an associated label in a rating scale: *Very Bad* (VB), *Bad* (B), *Neutral* (N), *Good* (G) and *Very Good* (VG). The network of dependences specifies which predicates contribute to the values of others.

The strength associated to each predicate is function of its antecedents and of the intrinsic properties of each kind of predicate. As a general rule, predicates that resume or aggregate a bigger number of predicates will hold a higher strength. However, strength is closely related to bias factors, rules that for instance, give more importance to direct experiences than indirect experiences, and that may come from sociology or psychology theories.

At the first level of the Repage memory we find a set of predicates not evaluated yet by the system: communications. They can be related to two different aspects: *communicated image*, and *communicated reputation*. In level two we have two kinds of predicates:

- *Valued communication*: The information contained in communications is modulated depending on the *credibility* of the communication sources. This is done by considering the images that the agent have about the source agents as *informants*. Once the communications are modeled, they become *valued communication*s.
- *Outcome*: The agent's subjective evaluation of a direct interaction.

In the third level we find two predicates that are only fed by valued communications. On one hand, a *shared voice* will hold the information received about the same target and same role coming from communicated reputations. On the other hand, *shared evaluation* is the equivalent for communicated images.

Shared voice predicates generates *candidate reputation*s, and *share evaluation*s together with *outcome*s, *candidate image*s. In this fourth level *candidate reputation* and *candidate image*s aren't strong enough to become a full *reputation* and *image* respectively. New communications and new direct interactions will contribute at this level to enrich these predicates and therefore "jump" to images and reputations. For a more detailed explanation we refer to [8].

3.2 Preliminaries: Notation to Describe Repage Information

Let $A = \{i_1, \ldots, i_N\}$ and $R = \{r_1, \ldots, r_M\}$ be a set of agents and a set of roles, and $L \in \mathbb{N}$, $L > 0$ the number of labels that evaluations have[1]. We define the set *Eval* of all possible evaluations as

$$Eval = \{< i, r, v > | i \in A, r \in R, v \in \{[x_1, \ldots, x_L]\}\}$$

where $x_1 \ldots x_L \in [0,1] \subset \mathbb{R}$ and $\sum_{k=1}^{L} x_k = 1$ (a probability distribution). We assume that the special role $\mathcal{I} \in R$ is predefined, referring to the *informant* role. Then, let $e \in Eval$, $i, j \in A$, $t \in \mathbb{N}$ and $s \in \mathbb{R}$, the set P of possible predicates are:

$Img_i(e), Rep_i(e)$	Image/Reputation of agent i with evaluation e
$CI_i(e,s), CR_i(e,s)$	Candidate Image/Reputation of i with evaluation e and strength s
$ShE_i(e,s), ShV_i(e,s)$	Shared Evaluation/Voice of i with evaluation e and strength s
$O_i(e,t,s)$	Outcome of i at the instant t with evaluation e and strength s
$vcI_{i,j}(e,t,s), vcR_{i,j}(e,t,s)$	Valued Comm Image/Reputation of i from j at the instant t with evaluation e and strength s
$cI_{i,j}(e,t,s), cR_{i,j}(e,t,s)$	Communicated Image/Reputation of i from j at the instant t with evaluation e and strength s

For example, the predicate $Img_{john}(< laura, seller, [0.6, 0.3, 0.1] >)$ indicates that $john$ has an image about $laura$, indicating that as a $seller$ she acts *bad* with a probability of 0.6, *neutral* with a probability of 0.3, and *good* with a probability of 0.1.

We state now some functions that will be helpful when defining attack and support relationships.

Definition 6. *Let $\varphi \in P$, $type(\varphi)$ returns the type of the predicate and $\mathrm{e}(\varphi)$ returns the evaluation object of the predicate. Assuming that $\mathrm{e}(\varphi)e =< i, r, v >\in Eval$ then $\varphi.target = i$, $\varphi.role = r$, $\varphi.val = v$*

3.3 Specifying Repage as a multicontext System

Following the Repage architecture specified in [8] it is possible to define it as a multicontext system. To do it, we have to specify the contexts and the bridge rules. Let $e \in Eval$, $i, j \in A$ and $t \in \mathbb{N}$, the next table shows the list of contexts and their respective information

[1] The original definition of Repage considers only five values, but this is easy to generalize.

Context Name	**Formulas**
Rep-Image(RIC)	$Img_i(e), Rep_i(e)$
Candidate(CaC)	$CI_i(e,s), CR_i(e,s)$
Third Party(TPC)	$ShE_i(e,s), ShV_i(e,s)$
Direct Experience(DEC)	$O_i(e,t,s)$
Valued Communication(VCC)	$vcI_{i,j}(e,t,s), vcR_{i,j}(e,t,s)$
Communication(CC)	$cI_{i,j}(e,t,s), cR_{i,j}(e,t,s)$

Each context has a first-order logic restricted to horn clauses. Thus, it is possible to express Repage predicates. The calculation of such predicates is done by means of bridge rules. They should follow the specifications of the Repage architecture. In Figure 1 we show a graphical representation of the Repage as a MCS where arrows are bridge rules. The definition of each one of the bridge rules regarding MCS-Repage is shown in figure 1. Bridge rules A_I, A_R and B appear in the BDI model [4]. The later (B) refers to the external beliefs that influence Repage. Rules A_I and A_R transform image and reputation predicates into beliefs in the BDI+Repage model. In this way, the normal BDI deductive process incorporates Repage information. They are completely defined in [4], but are out of the scope of this paper.

Rule $1r$ and $2r$ deal with communications. After agent i receives a communicated image (or reputation) from j, agent i takes into account the image as informant (role $\mathcal{I}$) of j before considering it. To do it, it modifies the strength of the predicate by the function f_s, defined in [8]. Basically it considers that communications from agents whose image as informants are bad, will generate valued communications with a low strength, and consequently, will have low influence.

Rules $3r$ and $4r$ aggregate valued communications referring to the same target agent and role, generating shared evaluations and shared voices respectively, as explained in section 3.1. Here, function f refers to the aggregation function defined for Repage. Distinct functions can be defined, but all of them should be based on weighted means. A deep study on aggregation functions regarding this kind of information can be found at [10] and [11].

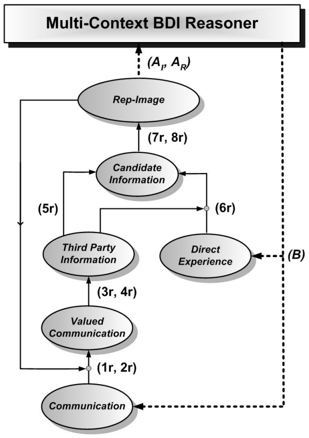
Rule $5r$ generates candidate reputations from shared voices without taking into account any other information. Instead, rule $6r$ considers a shared evaluation of a given target agent and role plus all outcome predicates over the same agent and role, to generate a candidate image. The aggregation functions are the same as before.

Finally, rules $7r$ and $8r$ generate image and reputation predicates by considering the exact same value as candidate image/reputation but that has a strength higher than certain threshold.

3.4 Defining a Repage Argumentation Framework

The previous specification allows us to define the set of all feasible arguments in the Repage system. Let F be a particular instance of a Repage-MCS system, the argumentation framework for F is $A_F = \langle H_F, G_F \rangle$ where H_F is the set of all possible arguments that can be build from F (as defined above). $G_F : H_F \times H_F \rightarrow M$ is the influence function that we define in the following lines. We take $M = [-1, 1]$ where 1 indicates the maximum support, -1 the maximum attack and 0 the neutral influence.

We define first the influence relation among single Repage predicates:



1r,2r $$\frac{CC : cI_{i,j}(e,t,s) \quad RIC : Img_i(< j, \mathcal{I}, v >)}{VC : vcI_{i,j}(e,t,f_s(s,v))}, \quad \frac{CC : cR_{i,j}(e,t,s) \quad RIC : Img_i(< j, \mathcal{I}, v >)}{VC : vcR_{i,j}(e,t,f_s(s,v))}$$

3r $$\frac{VC : vcI_{i,k_1}(< j, r, v_1 >, t_1, s_1) \quad \ldots \quad VC : vcI_{i,k_n}(< j, r, v_n >, t_n, s_n)}{TPC : ShE_i(< j, r, f(v_1, s_1, \ldots, v_n, s_n) >, f_s(s_1, \ldots, s_n))}$$

4r $$\frac{VC : vcR_{i,k_1}(< j, r, v_1 >, t_1, s_1) \quad \ldots \quad VC : vcR_{i,k_n}(< j, r, v_n >, t_n, s_n)}{TPC : ShV_i(< j, r, f(v_1, s_1, \ldots, v_n, s_n) >, f_s(s_1, \ldots, s_n))}$$

5r $$\frac{TPC : ShV_i(e,s)}{CaC : CR_i(e,s)}$$

6r $$\frac{TCP : ShE_i(< j, r, v >, s) \quad DEC : O_i(< j, r, t_1, v_1 >, s_1) \quad \ldots \quad DEC : O_i(< j, r, t_n, v_n >, s_n)}{Cac : CI_i(< j, r, f(v_1, s_1, \ldots, v_n, s_n) >, f_s(s, v))}$$

7r,8r $$\frac{CaC : CI_i(e,s), s \geq th_I}{RIC : Img_i(e)}, \quad \frac{CaC : CR_i(e,s), s \geq th_R}{RIC : Rep_i(e)}$$

Fig. 1. The Repage system specified as a MCS and the bridge rules in detail. Functions f and f_s are the aggregation functions defined for Repage.

Definition 7. *The influence relation $R_I \subseteq P_F \times P_F$ (where P_F is the set of all Repage predicates in the system F) is a relation that indicates which Repage predicates influence each other. In our case, predicates of the same type that refer to the same target agent and the same role influence each other. More formally: Let $\varphi, \phi \in P_F$, $\varphi R_I \phi \leftrightarrow type(\varphi) = type(\phi)$ and $e(\varphi).agent = e(\phi).agent$ and $e(\varphi).role = e(\phi).role$*

Furthermore, having an argument $\alpha = \langle B, b \rangle$, we define $con(\alpha) = b$ (it returns the conclusion of the argument) and $supp(\alpha) = B$ (it return the supporting set).

Having this, G_F is defined as follows:

Definition 8. *Let $\alpha, \beta \in H_F$ then,*

- ***Influence in the conclusion**: $G_F(\alpha, \beta) = con(\alpha).val \ominus con(\beta).val \leftrightarrow con(\alpha) R_I con(\beta)$*
- ***Influence in the premises**: $G_F(\alpha, \beta) = \Sigma(con(\alpha).val \ominus con(g).val | con(\alpha) R_I con(g))$ where $g \in supp(\beta)$*
- ***No influence**: $G_F(\alpha, \beta) = 0 \leftrightarrow$ it is not the case that $con(\alpha) R_I con(\beta)$, and $\nexists g \in supp(\beta)$ such that $con(\alpha) R_I con(g)$*

Function $\ominus : V \times V \to [-1, 1]$ (where V is the set of possible tuples $\{[v_1, \ldots, v_L]\}$ such that $v_1, \ldots, v_L \in [0,1] \subset I\!R$) calculates the degree of similitude between two evaluation values mapping it into the set $[-1, 1] \subset I\!R$. Several functions can be defined for this purpose. For instance, in [11] a difference function is defined using the concept of center of mass and momentum.

Function $\Sigma : 2^{[-1,1]} \to [-1, 1]$ is an aggregation function. It is used for the case in which an argument influences in more than one place the premises of the other argument. Examples for this function could be the average, the maximum and the minimum, but each of them would carry different consequences:

– **Average**: If Σ is defined as the average, agent would consider as no influence an argument that attack and support in the same grade another argument. The advantage is that all grades would be taken into account for the final value.
– **Max/Min**: In this case, the agent would consider only the maximum/minimum graded of all the set of influences. Again, this carries some complications, since values closed to zero does not bring much information.
– **MaxAbs**: Another possibility is to define Σ as the maximum but in absolute value. Then, the final value would be close to -1 if the maximum attack is higher that the maximum support, and close to 1 in the other way around. Therefore, only the most extreme grade would be consider.

Thus, playing with function $\ominus$ and Σ multiple frameworks can be defined. We plan as future work to investigate and characterize these functions.

3.5 The BDI+Repage Specification

The previous Repage specification can be placed in a multicontext BDI model, defining then a complete multicontext system. A possible base framework can be found in [4]. There, the BDI+Repage model is specified as MCS. It has one context for each basic attitude: beliefs, desires and intentions, but also, it is endowed with a planner and a communication context. The set of bridge rules performs the BDI reasoning taking into account the Repage information. Through rules A_I and A_R (see figure 1) image and reputation predicates are introduced into the belief context. These beliefs in combination with the desires and the planner context generate intentions. From the set of intentions, the model instantiates concrete actions, that would represent the *best* possible action. Further explanation can be found in [4]. The integrated argumentation framework should be able to justify any action from the intention it was generated from, this intention from the set of desires, plans, and beliefs, and each belief from their respective image and reputation information (if they come from Repage). Also, these Repage predicates can be justified by the internals of Repage defined above. Since the multicontext scpecification of the system is already done, it should not be difficult to incorporate the A_F framework in a more generic one where also beliefs, desires and intentions are present.We let the formal definition for this complete argumentation framework as future work.

4 Putting the Model to Work: Seeking for Information

In this section we show how the argumentation framework that we have defined in the previous sections can be used when agents cannot decide which action to perform because they lack in information.

4.1 The Initial Scenario

In this example, we consider the agent i whose architecture is a BDI+Repage where Repage has been specified as a MCS as shown in section3. Let's assume the agent desires focus on obtaining a very good car. However, given the current knowledge (the set of beliefs) the agent has about the two possible sellers (*alice* and *bob*), it can only generate the following intention[2]: $(I_i[buy(alice)]VGCar, 0)$.

An intention with grade 0 in the BDI+Repage architecture indicates that the positive effects of achieving it are cancelled by the negative ones. In other words, there is not benefit for the agent in pursing that intention[4]. In this situation the agent usually would ask for more information trying to generate a new intention with positive grade or to change the grade of this one. With an argumentation framework it has another alternative: ask for help to detect if there is something wrong in the reasoning process that has generated that intention.

Using the Repage+BDI architecture is easy to build an argument for the generated intention. An example could be $\langle Q, (I_i[buy(alice)]VGCar, 0)\rangle$ where Q is:

$$\{Img_i(< alice, seller, [0.3, 0.1, 0.6] >)\} \vdash_{A_I} (B_i[buy(alice)]VGCar, 0.3) \tag{1}$$

$$\{Img_i(< alice, seller, [0.3, 0.1, 0.6] >)\} \vdash_{A_I} (B_i[buy(alice)]VBCar, 0.6) \tag{2}$$

$$\{1, 2\} \vdash_{bdi} (I_i[buy(alice)]VGCar, 0) \tag{3}$$

At the same time, the justification of $alice$'s image can be obtained from Repage. In our example, a possible justification could be $\alpha = \langle P_\alpha, Img_i(< alice, seller, [0.3, 0.1, 0.6] >)\rangle$ where P_α is[3]

$$\{\emptyset\} \vdash_B cImg_{i,debra}(< alice, seller, [1, 0, 0] >, t_1, 0.8) \tag{4}$$

$$\{\emptyset\} \vdash_B cImg_{i,charly}(< alice, seller, [1, 0, 0] >, t_2, 0.8) \tag{5}$$

$$\{\emptyset\} \vdash_B O_i(< alice, seller, [0, 0, 1] >, 1) \tag{6}$$

$$\{4, Img_i(< debra, \mathcal{I}, [0, 0, 1] >)\} \vdash_{1r} vcI_{i,debra}(< alice, seller, [1, 0, 0] >, t_1, 0.8) \tag{7}$$

$$\{5, Img_i(< charly, \mathcal{I}, [0, 0, 1] >)\} \vdash_{1r} vcI_{i,charly}(< alice, seller, [1, 0, 0] >, t_2, 0.8) \tag{8}$$

$$\{8, 7\} \vdash_{3r} ShE_i(< alice, seller, [1, 0, 0] >, 0.8) \tag{9}$$

$$\{6, 9\} \vdash_{6r} CI_i(< alice, seller, [0.3, 0.1, 0.6] >, 0.9) \tag{10}$$

$$\{10\} \vdash_{7r} Img_i(< alice, seller, [0.3, 0.1, 0.6] >) \tag{11}$$

We recall here that the previous argument could be extended, since for instance, $Img_i(< debra, \mathcal{I}, [0, 0, 1] >)$ has not been justified. Agents can build arguments as long or short as they want depending on the level of detail they want to provide to the partner.

4.2 Building Counterarguments

Now, let's assume that i has sent the previous argument to j that is using the same argumentation framework. Agent j can try building arguments supporting or attacking α. In general, three possibilities arise[4]:

1. **No influential arguments**: In this case, agent j is not able to construct any argument that influences this one. It means that she does not have information about $alice$ as a $seller$, but furthermore, no information about $debra$ or $charly$ as $informants$ etc. Formally, in this case for all argument γ different of α, $G(\alpha, \gamma) = 0$.

[2] this means that agent i has the intention to achieve a VGCar(very good car) by archiving the action $buy(alice)$. Such notation is the logical language defined in [4]

[3] $\{\emptyset\} \vdash_B$ indicates that the information comes from outside the Repage system, like communications or outcomes

[4] This classification leads to the definition of several classes of arguments defined in [2].

2. **Only influent arguments in premises**: This is the case in which j can build an influential argument for the supporting set of α. For instance, she may have the argument $\beta = \langle P_\beta, Img_j(< debra, \mathcal{I}, [0.7, 0.3, 0] >)\rangle$ where P_β is

$$\begin{aligned} \{\emptyset\} &\vdash_B cImg_{j,peter}(< debra, \mathcal{I}, [0.7, 0.3, 0] >, t_1, 0.8) && (12) \\ \{12, Img_j(< peter, \mathcal{I}, [0, 0, 1] >)\} &\vdash_{1r} vcI_{j,peter}(< debra, \mathcal{I}, [0.7, 0.3, 0] >, t_1, 0.8) && (13) \\ \{13\} &\vdash_{3r} ShE_j(< debra, \mathcal{I}, [0.7, 0.3, 0] >, 0.8) && (14) \\ \{14\} &\vdash_{6r} CI_j(< debra, \mathcal{I}, [0.7, 0.3, 0] >, 0.8) && (15) \\ \{15\} &\vdash_{7r} Img_j(< debra, \mathcal{I}, [0.7, 0.3, 0] >) && (16) \end{aligned}$$

   In this case, notice that the image that i had about $debra$ on the role $\mathcal{I}$ was very good ([0,0,1]), but for agent j it is mostly bad ([0.7,0.3,0]). A possible G function should give a value tending to -1.
3. **Influent arguments in conclusion**: This is the case in which j is able to build an argument on the conclusion.

There are many heuristics an agent could use to assign relevance to counterarguments (and therefore take actions according to them). For example, giving priority to arguments of type 3 in front of arguments of type 2 and using the strength of the influence to stablish an order among arguments of the same type. Probably a dynamic heuristic that can adapt to the context will be necessary for agents in real open MAS. This is something that requires further study.

4.3 Acting in Consequence

After i receives the counterarguments from j she can do several things, some of them depending on the trust it has towards j. One possibility, if the trust on j is very high, is to accept j counterarguments and modify the knowledge base. Another possibility is to try to attack/support j's counterarguments. She can decide to ignore counterarguments from which she can build influential arguments with high attacking grades or, on the contrary, modify the knowledge base if she can build arguments that give support to what j is arguing. Depending on the dialog protocol, agent i even could send again this new arguments to j, to check whether j is capable to respond. The kind of arguments i can build are the same we have explained for j in the previous section, that is: no influential arguments, influent arguments in premises and influent arguments in conclusion.

5 Conclusions and Future Work

In this work we have presented a roadmap towards a specification of an argumentation framework for a cognitive BDI agent using the Repage system. Form the research explained in this paper, it should be clear that reputation models play a crucial role in the next generation of open multiagent systems. As a future work we plan to study in detail the influence of these models into negotiation and persuasion dialogues and extend the study also at the BDI model. In [2] a multicontext BDI model is used to build arguments and negotiate. We think that this model could be a good starting point for this.

We have also shown the importance of allowing granularity in the reputation models. From a cognitive point of view we argue that in some cases, the path that the information follows to build a final social evaluation is as important as the final value. In this paper we hint at the importance of these *paths*, by using them as arguments to support final values. Notice that without an argumentation system, seeking for information can be reduced at asking other agents for final values, increasing then the changes to obtain an *IDontKnow* answer.

Regarding the new extended argumentation framework, we plan to formally work on it in the context of argumentation framework theory. Although no exhaustive formal study was presented in this paper we have shown how our framework subsumes Dung's [6]. However a much deeper study is necessary to put it in context with other argumentation models, above all, those that deal with fuzzy arguments, like the one defined in [12].

6 Acknowledgments

This work was supported by the EC under the FP6 programme [eRep project CIT5-028575]; the Spanish Education and Science Ministry [AEI project TIN2006-15662-C02-01, AT project CONSOLIDER CSD2007-0022, INGENIO 2010] and the Generalitat de Catalunya [2005-SGR-00093].

References

1. Luck, M., McBurney, P., Shehory, O., Willmott, S.: Agent Technology: Computing as Interaction (A Roadmap for Agent Based Computing). AgentLink (2005)
2. Parsons, S., Sierra, C., Jennings, N.: Agents that reason and negotiate by arguing. Journal of Logic and Computation **8**(3) (1998) 261–292
3. Rahwan, I., Amgoud, L.: An argumentation based approach for practical reasoning. In: Proc. of AAMAS '06, New York, NY, USA, ACM (2006) 347–354
4. Pinyol, I., Sabater-Mir, J.: Pragmatic-strategic reputation-based decisions in bdi agents (to appear). In: Proc. of the AAMAS'09, Budapest, Hungary. (2009)
5. Giunchiglia, F., Serafini, L.: Multilanguage hierarchical logic (or: How we can do without modal logics). Journal of AI **65** (1994) 29—70
6. Dung, P.M.: On the acceptability of arguments and its fundamental role in nonmonotonic reasoning, logic programming and n-person games. AI **77**(2) (1995) 321–358
7. Simari, G.R., Loui, R.P.: A mathematical treatment of defeasible reasoning and its implementation. Artif. Intell. **53**(2-3) (1992) 125–157
8. Sabater-Mir, J., Paolucci, M., Conte, R.: Repage: Reputation and image among limited autonomous partners. JASSS **9**(2) (2006)
9. Conte, R., Paolucci, M.: Reputation in artificial societies: Social beliefs for social order. Kluwer Academic Publishers (2002)
10. Yager, R.: On the determination of strength of belief for decision support under uncertainty-part ii: fusing strengths of belief. Fuzzy Sets and Systems **1** (2004) 129—142
11. Sabater-Mir, J., Paolucci, M.: On representation and aggregation of social evaluations in computational trust and reputation models. International Journal of Approximate Reasoning **46**(3) (2007) 458–483
12. Krause, P., Ambler, S., Elvang-Goransson, M., Fox, J.: Logic of argumentation for reasoning under uncertainty. Computational Intelligence **11** (1995) 113–131

Empirical Testing of the “TrustWorthiness ANtecendents” (TWAN) Scale

Ellen Rusman[1], Jan van Bruggen and Martin Valcke

[1] Open University of the Netherlands, CELSTEC, Valkenburgerweg 167,
6401 DL Heerlen, The Netherlands
{ellen.rusman@ou.nl, janvanbruggen@ou.nl; martin.valcke@ugent.be}

Abstract. We have earlier proposed an extended model for trustworthiness between project team members in a symmetric work relationship, consisting of several additional antecedents to the commonly accepted antecedents of ability, benevolence and integrity. Examples of these additional antecedents are “communality” and “accountability”. By reviewing literature on the measurement of trust and trustworthiness, we have operationalized and specified this model in several scales. In this paper we describe the resulting draft version of a measurement instrument of trustworthiness, called “TrustWorthiness ANtecendents” scale (TWAN). We also present the first results of an empirical evaluation of this instrument.

Keywords: Trust; Trustworthiness; Virtual teams; Collaboration; Measurement Instrument

1 Introduction

In this paper we describe the operationalization and evaluation of an extended conceptual model for trustworthiness between project members in vocational contexts. Research on trust has predominantly used the model of (Mayer, Davis, & Schoorman, 1995) to operationalize and measure trustworthiness in organizational settings. The key antecedents of trustworthiness in this model are ability, benevolence and integrity. Mayer et.al.’s (1995) model was based on extensive literature research and developed within a particular domain, namely management, but with the purpose of integrating various content perspectives. The selection of (inclusion, deletion) of several antecedents mentioned in the literature was based on a conceptual analysis and ‘common sense’. Many researchers have used and accepted this model to define and measure trustworthiness, without setting up an empirical study to test the model. In a recent article (Schoorman, Mayer, & Davis, 2007), the original authors of the model urge researchers to reconsider and elaborate the model, with a special emphasis on the issue of measuring trust and trustworthiness in various settings. We have developed such an alternative model (Rusman et.al., submitted), based on an extensive, interdisciplinary literature review of antecedents of trustworthiness. We started this development because we were looking for a suitable measurement instrument to measure the effect of a personal identity profile on the development of trust between virtual project team members in a symmetric work relationship. We expect that impressions of trustworthiness formed in teams which have the availability of a profile (designed to foster trustworthiness) in the first two weeks of a project are more detailed but less extreme than those formed in teams without the availability of such a profile (Hancock & Dunham, 2001). In order to measure this we needed an instrument suitable to measure trustworthiness on a personal level (between dyads in a symmetric work relationship) between project team members and also with sufficient detail to determine the effects of a profile. While reviewing existing measurement instruments we discovered that they were either designed to measure other constructs (e.g. trust propensity, overall trust, trust in asymmetric relationships) or had insufficient level of detail in order to serve our purposes. Thus, we developed an alternative model to the commonly used models and operationalized it into a draft version of a scale. In this paper we shortly present this model and elaborate on the draft version of this "TrustWorthiness ANtecendents" scale (TWAN). Moreover, we report the first results of an empirical study that aims to validate this scale. An elaboration of these results will be presented during the workshop.

1.1 An Alternative Model for Trustworthiness: Development of a Measurement Instrument

Trustworthiness is the individual's assessment of how much and for what type of performance a trustee can be trusted (Hardin, 2002). People assess trustworthiness by collecting signs of particular characteristics of another person and these are 'tested' against their conceptual model of trustworthiness. In this way one determines for instance whether this person is friendly, open, reliable, consistent, etc. We have developed an alternative model. To develop an instrumental and operational version of this alternative model, we reviewed an additional 43 articles (see attachment 1 for an overview of reviewed literature) that reports on the development or use of specific scales or instruments to determine trust and trustworthiness. This resulted in an inventory of key methodological questions when measuring the antecedents of trustworthiness. Based on this inventory, we developed subscales consisting of four items/questions in view of each antecedent. Positive and negative phrasing of questions were balanced. This resulted in an instrument with the following operational structure (figure 1) and related questions (table 1), called "TrustWorthiness ANtecendents" scale (TWAN). The draft version of the TWAN scale consists of 92 questions.

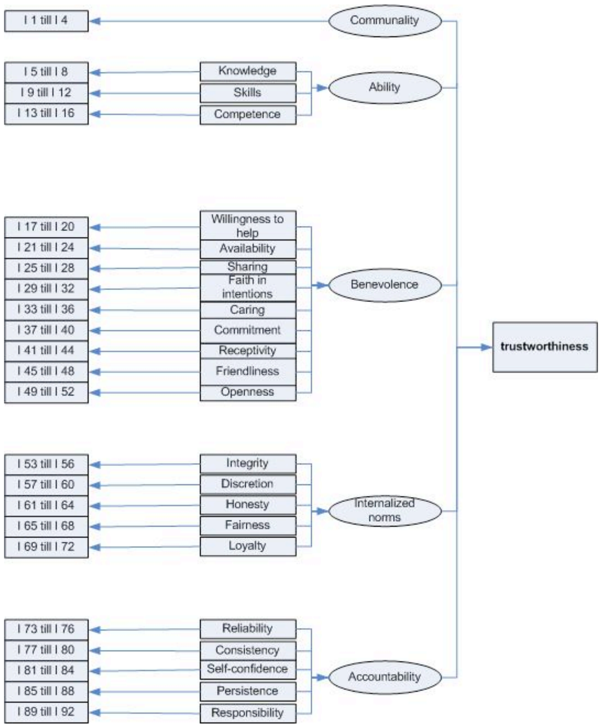


Fig. 1. Operationalization of trustworthiness in a draft version of the TWAN scale

Table 1. Draft version of trustworthiness measurement instrument TWAN

Antecedents of trustworthiness (AT), items and variable names
Communality (COM)
I 1: I trust …. because he/she shares the same interests (AT_COM_int) I 2: I trust ….because he/she shares my expectations and goals of the project (AT_COM_goal) I 3: I don't trust … because he/she has a different communication style than mine (AT_COM_com)[*] I 4: … work values are not very similar to mine (AT_COM_work)*
Ability
Knowledge (KNOW)
I 5: I trust … to contribute relevant expertise to this project (AT_KNOW_expert) I 6: I trust ……… to indicate the limitations of his/her knowledge (AT_KNOW_limit) I 7: … is not very knowledgeable about his/her discipline (AT_KNOW_discip)* I 8: … has not so much knowledge which is relevant for the work that needs to be done (AT_KNOW_work)*
Skills (SKIL)
I 9: In his/her job … seems to work efficiently (AT_SKIL_effic)

[*] Question posed negative

I 10: I have full confidence in the skills of …. (AT_SKIL_conf) I 11: …… does not perform his/her tasks with skill (AT_SKIL_perf)* I 12: I cannot rely on the task-related skills of … (AT_SKIL_rel)*
Competence (COMP)
I 13: ……… does things competently (AT_COMP_comp) I 14: ……… does things in a capable manner (AT_COMP_cap) I 15: I feel that … is not good at what he/she does within the project (AT_COMP_good)* I 16: … seems to be unsuccessful in the professional activities he/she undertakes (AT_COMP_unsuc)*
Benevolence
Willingness to help (HELP)
I 17: If I got into difficulties with work I know …. would try and help me out (AT_HELP_dif) I 18: I can trust… to lend me a hand if needed (AT_HELP_hand) I 19: If I required help, … would not do his/her best to help me (AT_HELP_best)* I 20: I feel that I can not count on … to help me with a crucial problem (AT_HELP_count)*
Availability (AV)
I 21: It's usually hard for me to get in touch with ……… (AT_AV_touch)* I 22: … is available when I need him/her (AT_AV_avail) I 23: I can usually reach … when I need him/her (AT_AV_reach) I 24: I am not able to contact readily … when it is required (AT_AV_con)*
Sharing (SHA)
I 25: Even if I didn't ask ... to share knowledge with me I feel certain that he/she will (AT_SHA_share) I 26: I feel that …. keeps information from me (AT_SHA_keep)* I 27: … does not pass information or ideas on that can be helpful to you or the project team (AT_SHA_pass)* I 28: … timely shares any relevant information (AT_SHA_time)
Faith in intentions (FI)
I 29: I think that ….. takes advantage of me (AT_FI_advant)* I 30: I feel that … takes advantage of people who are vulnerable (AT_FI_vuln)* I 31: I can rely on ……….. to react in a positive way when I expose my weakness to him/her (AT_FI_weak) I 32: Sound principles seems to guide the behaviour of … (AT_FI_princ)
Caring (CA)
I 33: If I share my problems with him/her, … will respond constructively and caringly (AT_CA_constr) I 34: … does not keep my interests in mind when making decisions (42, adapted) (AT_CA_decis)* I 35: … cares about the well-being of others (25, adapted) (AT_CA_others) I 36: … is primarily interested in his/her own welfare (16, item 1 adapted) (AT_CA_own)*
Commitment (COMIT)
I 37:… makes considerable investments in our working relationship (AT_COMIT_inv) I 38: … is not strongly committed to the project (AT_COMIT_com)* I 39: … does not do everything within his/her capacity to help our team perform (AT_COMIT_cap)* I 40: … does everything what is possible in order to meet the project goals (AT_COMIT_goal)
Receptivity (REC)
I 41: … makes an effort to understand what I have to say (AT_REC_eff) I 42: … is sincere in his/her attempts to meet my point of view (AT_REC_sinc) I 43:… often fails to listen to what I say (AT_REC_list)* I 44: Often … does not pay full attention to what I am trying to tell him/her (AT_REC_atten)*

Friendliness (FRI)

I 45: If I make a mistake, … is willing to forgive (AT_FRI_mis)
I 46: … is friendly and approachable (AT_FRI_appr)
I 47: If …. unexpectedly laughed at something I did or said, I would wonder if he/she was being critical and unkind (AT_FRI_crit)*
I 48: If … asks why a problem occurs, I will not speak freely when I am partly to blame (AT_FRI_speak)*

Openness (OPEN)

I 49: …. lets me know what's on his/her mind (AT_OPEN_mind)
I 50: ….. shares his/her thoughts with me (AT_OPEN_share)
I 51: … doesn't tell me what is really going on (AT_OPEN_tel)*
I 52: … is secretive (AT_OPEN_secr)*

Internalized norms

Integrity (INT)

I 53: ….. can not be corrupted (AT_INT_nocor)
I 54: … is a corruptible person (AT_INT_cor)*
I 55: I have faith in the integrity of … (AT_INT_fait)
I 56: … is not honest in describing his/her experience and abilities (AT_INT_hon)*

Discretion (DISC)

I 57: If I give ….. confidential information, he/she keeps it confidential (AT_DISC_conf)
I 58: ….. does not tell others about things if I ask that they be kept secret (AT_DISC_secr)
I 59: I lack confidence in the overall discretion of … (AT_DISC_discr)*
I 60: … talks too much about sensitive information that I give him/her (AT_DISC_sens)*

Honesty (HON)

I 61: I feel that … works with us honestly (AT_HON_hon)
I 62: I think that … does not mislead me (AT_HON_mis)
I 63: Even when … makes excuses which sound rather likely, I am not confident that he/she is telling the truth (AT_HON_conf)*
I 64: Sometimes … changes facts in order to get what he/she wants (AT_HON_fac)*

Fairness (FAIR)

I 65: ……… treats me fairly (AT_FAIR_fair)
I 66: … treats me on an equal basis with others (AT_FAIR_equ)
I 67: ….. treats others better than he/she treats me (AT_FAIR_bett)*
I 68: … is unfair in dealings with me (AT_FAIR_unfair)*

Loyalty (LOY)

I 69: I can discuss problems with … without having the information used against me (AT_LOY_prob)
I 70: …. would never intentionally misrepresent my point of view to others (AT_LOY_misp)
I 71: If I make a mistake, … will use it against me (AT_LOY_mist)*
I 72: If … didn't think I had handled a certain situation very well, he/she would criticize me in front of other people (AT_LOY_crit)*

Accountability

Reliability (REL)

I 73: Keeping promises is a problem for … (AT_REL_keep)*
I 74: ….. does things that he/she promises to do for me (AT_REL_prom)
I 75: If …. promised to do me a favour, he/she would follow through (AT_REL_fav)
I 76: I feel that …. will not keep his/her word (AT_REL_word)*

Consistency (CONS)
I 77: …… behaves in a very consistent manner (AT_CONS_con)
I 78: I sometimes ignore ……….. because he/she is unpredictable and I fear writing or doing something which might create conflict (AT_CONS_unpr)*
I 79: I seldom know what …… will do next (AT_CONS_nex)*
I 80… responds the same way under the same conditions at different times (AT_CONS_dif)
Self-confidence (SEC)
I 81: .… has high self esteem (AT_SEC_est)
I 82: I think that …. is very self-confident (AT_SEC_conf)
I 83: I feel that … is insecure of her/himself (AT_SEC_insec)*
I 84: … has low self esteem (OI) (AT_SEC_lowest)*
Persistence (PER)
I 85: Even in hard working circumstances, I can count on …. to follow through on work commitments (AT_PER_com)
I 86: In the face of difficulties I can count on …. to solve problems and meet work commitments in time (AT_PER_prob)
I 87: In difficult working circumstances … fails to follow through on work commitments (AT_PER_fai) *
I 88: When encountering problems … lacks the courage to constructively start working on them (AT_PER_constr)*
Responsibility (RES)
I 89: I can rely on …. not to make my work more difficult by careless work (AT_RES_dif)
I 90: I feel that…. tries to get out of his/her work commitments (AT_RES_com)*
I 91: … would go on with his/her work even if nobody checked it (AT_RES_work)
I 92: … readily denies responsibility for problems incurred by his/her mistakes (AT_RES_prob)*

To test our alternative model, we focused on the following questions:

- Is the proposed conceptual model valid? (construct validity)
- Do the questions in the questionnaire measure what they intend to measure? Can the proposed measurement instrument indeed distinguish between different levels of trustworthiness (content validity)?
- Do the questions represent the underlying latent variable which they intend to measure (unidimensionality)? Can they also distinguish between the other latent variables underlying trustworthiness (discriminant validity)?
- Is the questionnaire internally consistent (reliability)?

2 Method

In order to test our model and measurement instrument we set up an empirical study at the Ghent University, Belgium. To consider the language background of the respondents, we translated the English version of the questionnaire into Dutch. This translation was checked independently by two experts.

2.1 Nature of the Research Instrument

The questionnaire – presented above - contained open, as well as close-end questions. Open-ended questions referred to background variables of respondents, such as gender, age, organization, function, duration of the project they were working in, goals of the project, number of people participating in the project work, degree of personal acquaintance with other project team members and the means of communication within the project (e.g., face to face/audioconferencing/videoconferencing). In this paper we concentrate on the results of the close-ended questions. These questions referred to the antecedents of trustworthiness and reflect the items represented in Table 1. All items/questions were shuffled in the final version of the questionnaire, to prevent bias interference when replying to subsequent items. Respondents could not recognize the relationship between questions and a specific antecedent. Respondents were asked to react to individual questions with a 7-point Likert scale: (1) Strongly disagree , (2) Disagree, (3) Slightly disagree, (4) Neutral, (5) Slightly agree, (6) Agree and (7) Strongly agree. Respondents were asked to react to the questionnaire items twice by keeping a person in mind which they trusted most and least within one specific project context. We expected, in this way, to trace differences in measurement related to trustworthiness, but limiting respondents in their choice of extreme examples while they could only choose one project context.

2.2 Procedure

The data were collected by trained bachelor level research-students, enrolled in the Educational Sciences program at the Ghent University. These students worked in groups of 10 and were given a detailed research portfolio containing all practical materials needed to perform the study. Students received questionnaires, an instruction, a list with the scales and items and the names of the variables together with a predefined excel-file, containing the variable names fixed in the columns and in the same order as the questions were posed in the questionnaire. They were instructed in the guest lecture, but also received paper instructions on how to perform the research. Each group gathered data from 50 professionals having concrete experience with project work in their professional context. It has to be stressed that this sampling technique results in a "convenience sample", and does not guarantee that a specific stratification of the population is mirrored in the final sample. All data had to be entered and processed on the base of the pre-structured excel files. All materials, including the paper versions of the questionnaires, were handed in. The data collection process was organized during a period of 5 weeks.

Respondents were told that the responses on this questionnaire would be kept anonymous and that it would take about 15 minutes to complete the questionnaire. They were instructed to fill out the questionnaire, whilst keeping in mind a project they had encountered during work or study in which they had to collaborate with at least two other people in order to achieve an objective and were strongly dependent on these other project members. They were also asked to choose this context based on a difference in the trustworthiness of the different team members. Their project experience could be positioned in a face to face setting, an online setting or in a mix of these forms.

2.3 Characteristics of the Research Sample

On the base of the procedure, described in the former section, a data set was obtained from 1180 respondents. Because the questionnaire also contained a number of open questions (e.g. about the organization and job function of the respondent), the legitimacy of the data records was screened (e.g., including an analysis of the handwriting). Although initially it seemed that four respondents failed to fill the questionnaire well, it turned out to be a data processing error. None of the respondents needed to be excluded, leaving all 1180 respondents.

52% of the respondents were male, 48% female. The age of respondents varied between 17 and 71, with a mean age of 39, although the majority of respondents fell between 20 and 55 (see figure 2). We observe two broad types of respondents: advanced students who probably already experienced project work during their study (age group 17 till 29) and employees of professional organizations (age group 30 till 71 (see figure 2). The majority of the respondents belonged to the latter group. Group size of the project teams was reported by 82% of respondents to be between 2 to 13 project team members.

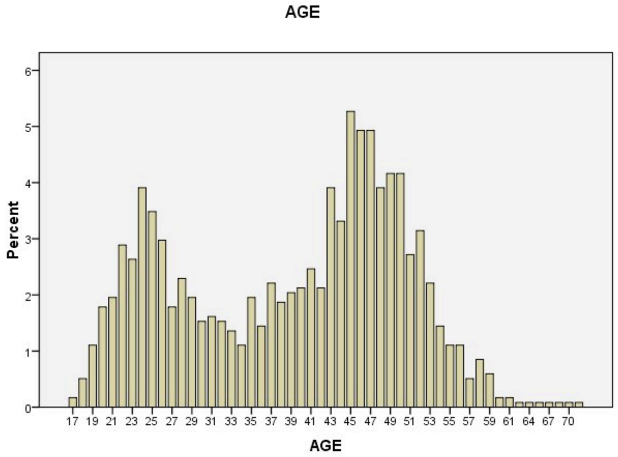


Fig. 2. Age of respondents

2.4 Data Analysis

Reliability of the TWAN scale will be assessed by determining the internal consistency of the questionnaire, using Cronbach's alpha or the mean inter-item correlation. Reliability will be considered for each of the clusters of items in the instrument, reflecting the same antecedent.

Additional data analysis is aimed at the validation of concepts (content and construct validity) of the TWAN scale. This will be done by confirmatory factor analysis and structural equation modeling (SEM). We can also compare the scores of the most trustworthy person and least trustworthy person and analyze the nature of observed differences.

First analysis indicate that our model underlying the TWAN scale will partly hold after empirical evaluation and that trustworthiness can be determined by measuring four latent variables of ability (6 items), benevolence (12 items), internalized norms (10 items) and accountability (7 items) by in total 25 items. We did not succeed to operationalize the latent variable 'communality' in an internal consistent manner (Cronbach's alpha of 0.45).

3 Conclusion and Future Work

On the basis of an earlier study (Rusman et.al., submitted), we developed an alternative model to the model presented by Mayer, Davis and Schoorman's (1995). Next to a description of this extended model and the operationalization in the TWAN scale, we described the approach to involve a large scale convenience sample in an empirical test of this scale. Current analysis of the available data shed a clear light on the appropriateness of the respondent group. We will further evaluate the reliability of the instrument and next – and foremost – the structure validity of the instrument. Based on structural equation modeling, it will become possible to analyze the relationships between the antecedents and dependent variables.

In future work we also intend to apply the – empirically tested model - to study the impact of profiling techniques that are hypothesized to foster and support trustworthiness decisions in virtual project teams in the initial phases of a project. We expect that team members with a personal profile will develop in a more rapid way a completer image/impression of trustworthiness and underlying antecedents as compared to team members without a clear profile. We also expect these people to express less extreme judgments about trustworthiness of project team members.

Acknowledgement

We would like to thank the three anonymous reviewers who helped to improve this paper with their valuable comments.

References

Bhattacherjee, A. (2002). Individual Trust in Online Firms: Scale Development and Initial Test. *Journal of Management Information Systems 19*(1), 211 - 241

Butler, J. K. (1991). Towards understanding and measuring conditions of trust: evolution of a condition of trust inventory. *Journal of management, 17*(643-663), 643-663.

Cook, I., & Wall, T. (1980). New work attitude measures of trust, organizational commitment and personal need nonfulfillment. *Journal of Occupational Psychology, 53*, 39-52.

Cummings, L. L., Bromiley, P., Kramer, R. M., & Tyler, T. R. (1996). *The Organizational Trust Inventory (OTI): Development and validation*. Thousand Oaks, CA, US: Sage Publications, Inc.

Doyal, L., & Gough, I. (1991). *A Theory of Human Need.* . London: : Macmillan.

EATMP. (2003a). *Guidelines for Trust in Future ATM Systems: A Literature Review*. Brussels: Eurocontrol.

EATMP. (2003b). *Guidelines for Trust in Future ATM Systems: Trust Measures*. Brussels: Eurocontrol.

Feng, J., Lazar, J., & Preece, J. (2004). Empathy and online interpersonal trust: a fragile relationship. *Behaviour and information technology*.

Gillespie, N. (2003). *Measuring trust in working relationships: the behavioural trust inventory*. Paper presented at the Academy of Management Meeting. from http://www.mbs.edu/index.cfm?objectid=951E38F4-123F-A0D8-42A4CE244A85F4BE.

Glaeser, E. L., Laibson, D. I., Scheinkman, J. A., & Soutter, C. L. (2000). Measuring trust. *The quarterly journal of economics*, 811-846.

Hancock, J. T., & Dunham, P. J. (2001). Impression Formation in Computer-Mediated Communication Revisited. *Communication research, 28*(3), 325-347.

Hardin, R. (2002). *Trust and trustworthiness*. New York: Russell Sage Foundation.

Hoy, W. K., Tschannen-Moran, M. . (2003). The conceptualization and measurement of faculty trust in schools: the Omnibus-T-Scale. In W. K. Hoy, Miskel, C.G. (Ed.), *Studies in Leading and Organizing Schools*.

Illes, K. (2006). Trust Questionnaire. Retrieved 20th of October, 2006, from http://btc-server.btc.anglia.ac.uk/phpsurveyor/?sid=3

Jeanquart-Barone, S. (1993). Trust differences between supervisors and subordinates: examining the role of race and gender. *Sex roles, 29*(1-2), 1-11.

Jian, J.-Y., Bisantz, A. M., & Drury, C. G. *Towards an empirically determined scale of trust in computerized systems: distinguishing concepts and types of trust*. Paper presented at the Human Factors and Ergonomics Society 42nd Annual Meeting, Chicago.

Jian, J. Y., Bisantz, A. M., & Drury, C. G. (1997). *Towards an empirically determined scale of trust in computerized systems: distinguishing concepts and types of trust*. Paper presented at the Human Factors and Ergonomics Society 42nd Annual Meeting, Chicago, Il.

Jiang, X., Khasawneh, M. T., Reena Master, Bowling, S. R., Gramopadhye, A. K., Melloy, B. J., et al. (2004). Measurement of human trust in a hybrid inspection system based on signal detection theory measures. *International Journal of Industrial Ergonomics, 34*, 407-419.

Johnson-George, C., & Swap, W. C. (1982). Measurement of specific interpersonal trust: Construction and validation of a scale to assess trust in a specific other. *Journal of Personality and Social Psychology, 43*(6), 1306-1317.

Kanawattanachai, P., & Yoo, Y. (2005). Dynamic nature of trust in virtual teams. *Sprouts: Working papers on Information Environments, systems and organizations, 2*(2), 41-58.

Kramer, R. M. (1999). Trust and distrust in organizations: Emerging perspectives, enduring questions. *Annual Review of Psychology, 50*(1), 569.

Lagace, R. R. (1991). An Exploratory Study of Reciprocal Trust Between Sales Managers and and Salespersons *Journal of Personal Selling & Sales Management, 11*(2), 49-58.

Larzelere, R. E., & Huston, T. L. (1980). The dyadic trust scale: Toward understanding interpersonal trust in close relationships. *Journal of Marriage and the Family*, 595-604.

Lewicki, R. J., Bunker, B. B., & Rubin, J. Z. (1995). Trust in relationships: A model of development and decline, *Conflict, cooperation, and justice: Essays inspired by the work of Morton Deutsch*. (pp. 133-173). San Francisco, CA, US: Jossey-Bass.

Lewicki, R. J., Bunker, Barbara Benedict, Kramer, Roderick M., Tyler, Tom R. (1996). *Developing and maintaining trust in work relationships*. Thousand Oaks, CA, US: Sage Publications, Inc.

Madsen, M., & Gregor, S. (2000). *Measuring human-computer trust*. Paper presented at the Proceedings of Eleventh Australasian Conference on Information Systems 6-8 December, Brisbane.

Mayer, R. C., & Davis, J. H. (1999). The effect of the performance appraisal system on trust for management : A field quasi-experiment. *Journal of applied psychology, 84*(1), 123-136.

Mayer, R. C., Davis, J. H., & Schoorman, D. (1995). An integrative model of interorganizational trust. *Academy of management review, 20*(3), 709-734.

McAllister, D. J. (1995). Affect and cognition-based trust as foundations for interpersonal cooperation in organisations. *Academy of management journal, 38*(1), 25-59.

McAllister, D. J., Lewicki, Roy J., Chaturvedi, Sankalp. (2006). Trust in developing relationships: from theory to measurement. *Academy of Management Proceedings*, G1-G6.

McKnight, D. H., Choudhury, V., & Kacmar, C. (2002). Developing and Validating Trust Measures for e-Commerce: An Integrative Typology. *Information Systems Research, 13*(3), 334-359.

McKnight, D. H., Cummings, L. L., & Chervany, N. L. (1998). Initial trust formation in new organizational relationships. *Academy of Management Review, 23*(3), 473-490.

Muir, B. M., & Moray, N. (1996). Trust in automation: II. Experimental studies of trust and human intervention in a process control simulation. *Ergonomics, 39*(3), 429-460.

Pearce, J. L., Sommer, S. M., Morris, A., & Frideger, M. A. (1992). *Configurational approach to interpersonal relations: profiles of workplace social relations and task interdependence*. Irvine: University of California

Rempel, J. K., Holmes, J. G., & Zanna, M. P. (1985). Trust in close relationships. *Journal of personality and social psychology, 49*(1), 95-112.

Rotter, J. B. (1967). A new scale for the measurement of interpersonal trust. *Journal of personality, XXXV*, 651-665.

Rozendaal, v. C. (1997). *Vertrouwen in leidinggevenden. Een vergelijkende literatuurstudie naar definities van het vertrouwen in leidinggevenden en de inhoudsvaliditeit van meetprocedures*. Heerlen: Open Universiteit Nederland.

Rusman, E., Bruggen, van, J., Sloep, P., Koper, R. (submitted). Fostering trust in virtual teams: towards a design framework. [available online: http://dspace.ou.nl/dspace/handle/1820/1791].

Schoorman, F. D., Mayer, R. C., & Davis, J. H. (2007). An integrative model of organizational trust: past, present and future. *Academy of Management Review*. *32*(2), 344–354.

Sheridan, T. B. (1988). Trustworthiness of command and control systems. . *IFAC Man-Machine Systems*, 429-431.

Sims, H. P., Szilagyi, A. D., & Keller, R. T. (1976). The measurements of job characteristics. *Academy of Management Journal, 19*(2), 195-212.

Taylor, R. M., Shadrake, R., & Haugh, J. (1995). *Trust and adaptation failure: An experimental study of uncooperation awareness*. Paper presented at the The Human-Electronic Crew: Can we Trust the Team?, 3rd Int. Workshop on Human-Computer Teamwork.

Wheeless, L. R., Grotz, Janis. (1977). The measurement of trust and its relationship to self-disclosure. *Human Communication Research, 3*(3), 250-257.

Wrightsman, L. S. (1991). Interpersonal trust and attitudes toward human nature. In J. P. S. Robinson, P.R.; Wrightsman, L.S. (Ed.), *Measures of personality and social psychological attitudes*. (Vol. 1, pp. 373-412). San Diego: Academic press, inc.

Yamagishi, T., Cook, K. S., & Watabe, M. (1998). Uncertainty, Trust, and Commitment Formation in the United States and Japan (Vol. 104, pp. 165-194).

Zolin, R., Hinds, P. J., Fruchter, R., & Levitt, R. E. (2002). Trust in Cross-functional, global teams. *CIFE*

Appendix 1

Nr.	Reference
1.	(Rotter, 1967)
2.	(Butler, 1991)
3.	(Glaeser, Laibson, Scheinkman, & Soutter, 2000)
4.	(Feng, Lazar, & Preece, 2004)
5.	(Rempel, Holmes, & Zanna, 1985)
6.	(Illes, 2006)
7.	(Hoy, 2003)
8.	(Zolin, Hinds, Fruchter, & Levitt, 2002)
9.	(Johnson-George & Swap, 1982)
10.	(Cummings, Bromiley, Kramer, & Tyler, 1996)
11.	(Rozendaal, 1997)Bromiley, Butler en Cook and Wall)
12.	(Cook & Wall, 1980)
13.	(Jian, Bisantz, & Drury, 1997)
14.	(Kramer, 1999)
15.	(Lagace, 1991)
16.	(Jeanquart-Barone, 1993)
17.	(Larzelere & Huston, 1980)
18.	(Yamagishi, Cook, & Watabe, 1998)
19.	(McKnight, Cummings, & Chervany, 1998)
20.	(Pearce, Sommer, Morris, & Frideger, 1992)
21.	(Sims, Szilagyi, & Keller, 1976)
22.	(Rempel et al., 1985)
23.	(Mayer & Davis, 1999)
24.	(Mayer et al., 1995)
25.	(McAllister, 1995)
26.	(McKnight, Choudhury, & Kacmar, 2002)
27.	(Jiang et al., 2004)
28.	(Gillespie, 2003)
29.	(Bhattacherjee, 2002)
30.	(Kanawattanachai & Yoo, 2005)
31.	(Muir & Moray, 1996)
32.	(Lewicki, Bunker, & Rubin, 1995)
33.	(McAllister, Lewicki, Roy J., Chaturvedi, Sankalp, 2006)
34.	(Madsen & Gregor, 2000)
35.	(Lewicki, Bunker, Barbara Benedict, Kramer, Roderick M., Tyler, Tom R., 1996)
36.	(Jian, Bisantz, & Drury)
37.	(EATMP, 2003a)
38.	(Taylor, Shadrake, & Haugh, 1995)
39.	(EATMP, 2003b)
40.	(Wrightsman, 1991)
41.	(Doyal & Gough, 1991)
42.	(Sheridan, 1988)

Detecting and Dealing with Naive Agents in Trust-aware Societies

Amirali Salehi-Abari and Tony White

School of Computer Science, Carleton University,
1125 Colonel By Drive, Ottawa, Ontario, K1S 5B6, Canada
{asabari, arpwhite}@scs.carleton.ca

Abstract. Autonomous agents, in ways analogous to humans, require trust and reputation concepts in order to identify communities of agents with which to interact reliably. Through the introduction of naive agents, this paper shows empirically that while learning agents can identify malicious agents through direct interaction, naive agents compromise utility through their inability to discern malicious agents. Moreover, the impact of the proportion of naive agents on the society is analyzed. The paper demonstrates the need for witness interaction trust to detect naive agents in addition to the need for direct interaction trust to detect malicious agents. By proposing a set of policies, the paper demonstrates how learning agents can isolate themselves from naive and malicious agents.

1 Introduction

Trust is a crucial concept driving decision making and relationships in human and artificial societies. According to Jarvenpaa et al.[5], trust is an essential aspect of any relationship in which the trustor does not have control over the actions of a trustee, the decision is important, and the environment is uncertain. This paper uses the same definition of trust presented by Mui et al. [7]: “Trust is a subjective expectation an agent has about another’s future behavior based on the history of their encounters”. This definition is consistent with *image* discussed by Sabater et. al [9].

Trust and reputation models assist agents in deciding how, when and who to interact with in a specific context [8]. In other words, an agent must be able to model trustworthiness of potential interaction partners and make decisions based on that. It is the position of this paper that the main utility of trust and reputation models is minimizing the risk of interacting with others by avoiding interacting with malicious agents. With this view in mind, the principal objective of such models is the detection of untrustworthy agents.

Most computational trust and reputation models are designed and evaluated based on the assumption that the agent society only embraces two types of agents: trust-aware and malicious. In our view, an agent society should include another type of agent called *a naive agent*. Naive agents are naive in terms of deciding how, when and who to interact with while always cooperating with other agents. The effects of naive agents on trust-aware individuals and the whole of society have not been analyzed to date. This observation motivates the work reported in this paper.

Our contributions include the introduction of the concept of a naive agent, analyzing the impact of this agent class on agent societies using a game-theoretic model on a simulation platform, and a strategy proposal for trust-aware agents to deal with them. While ART [3] aims to provide a unified platform for trust model evaluation it does not consider variables that are central to the evaluation proposed in this paper. Therefore, in order to evaluate our model, we design our own testbed which is described in section 4.

The remainder of this paper proceeds as follows. After describing the related work in Section 2, we discuss the environment model of agents in Section 4. We describe the agent model in Section 5, and experiments in Section 6. Finally, conclusions and future work are explained in Section 8.

2 Related Work

The body of research on trust and reputation models is large; a review of which can be found in [8] and [11]. Here we limit our discussion to models that incorporate multiple information sources or express the importance of doing so.

Regret [10] is a decentralized trust and reputation system which takes into account three different sources of information: direct experiences, information from third party agents and social structures. The direct trust, witness reputation, neighborhood reputation and, system reputation are introduced in Regret.

Yu and Singh developed an approach for social reputation management, in which they represented an agent's trust ratings regarding another agent as a scalar and combined them with testimonies [15]. Yu et al. have proposed the trust model in peer-to-peer systems in which each peer has its own set of acquaintances [14]. The acquaintance's reliability and credibility are included in this model but are not used to drive the selection of new acquaintances as proposed here.

Huynh et al. introduce a trust and reputation model called FIRE that integrates a number of information sources to estimate the trustworthiness of an agent [4]. Specifically, FIRE incorporates interaction trust, role-based trust, witness reputation, and certified reputation to provide a trust metric. FIRE does not consider malicious witness providers because it assumes agents are honest in exchanging information. The research reported here explicitly deals with inaccurate witness providers.

In the Social Interaction Framework (SIF) [12], agents are playing a Prisoner's Dilemma set of games with a partner selection phase. Each agent receives the results of the game it has played plus the information about the games played by a subset of all players. An agent evaluates the reputation of another agent based on observations as well through other witnesses. However, SIF does not describe how to find witnesses, which the model reported here does.

There are few trust models which consider the existence of an adversary in providing witness information and present solutions for dealing with inaccurate reputation, essentially the problem of naive agents of interest here. TRAVOS [13] models an agent's trust in an interaction partner. Trust is calculated using probability theory that takes account of past interactions and reputation information gathered from third parties while coping with inaccurate reputations. Yu and Singh [16] is similar to TRAVOS, in that it rates opinion source accuracy based on a subset of observations of trustee behavior.

3 Naive Agent

We define a naive agent as follows: a naive agent is incapable of properly deciding how, when and with whom to interact. In this sense, it fails to detect and stop interacting with untrustworthy agents due to the lack of proper assessment of other agents. They are optimistic such that they consider all other agents completely trustworthy and always cooperate with every member of the society. Naive agents usually do not have any malicious intention.

Examples of naive agents can be seen in many places. On eBay, sellers receive feedback (+1, 0, -1) in each auction and their reputation is calculated as the sum of those ratings over the last six months. It can be observed that there are many users (buyers) who do not receive satisfactory goods or services but they rate the sellers highly and even continue interacting with them. We see these users as naive users. In peer-to-peer file sharing systems free riding is a well-documented problem (e.g., BitTorrent). Free-riders do not share enough or appropriate files while benefiting from the society by downloading files from peers. It is observable that there are some users in these systems who are incapable of detecting free-riders and share all of their files to everyone in the society. These peers follow our definition for naive agents.

4 Environment Model

The majority of open distributed computer systems can be modeled as multi-agent systems (MAS) in which each agent acts autonomously to achieve its objectives. In the model presented here, heterogeneous agents interact in a game theoretic manner. The model is described in the following 3 subsections.

4.1 Interactions

An agent interacts with a subset of all agents. Two agents are *neighbors* if both accept each other as a neighbor and interact with one another continuously. An agent maintains the *neighborhood* set which is dynamic, changing based upon the agent's decisions. Agents can have two types of interactions with their neighbors: *Direct Interaction* and *Witness Interaction.*

Direct Interaction. Direct interaction is the most popular source of information for trust and reputation models [11, 8]. Different fields have their own interpretation and understanding of direct interaction. In the context of e-commerce, direct interaction might be considered to be buying or selling a product.

Witness Interaction. An agent can ask for an assessment of the trustworthiness of a specific agent from its neighbors and then the neighbors send their ratings of that agent to the asking agent. We call this asking for an opinion and receiving a rating, a ***Witness Interaction***.

4.2 Games: IPD and GPD

We have modeled direct interaction and witness interaction using two extensions of the Prisoner's Dilemma. The Prisoner's Dilemma is a non-zero-sum, non-cooperative, and simultaneous game in which two players may each "cooperate" with or "defect" from the other player. In the iterated prisoner's dilemma (IPD) [1], the game is played repeatedly. Therefore, each player has the opportunity to "punish" the other player for previous uncooperative play. The IPD is closely related to the evolution of trust because if both players trust each other they can both cooperate and avoid mutual defection. We have modeled the direct interaction using IPD.

Witness Interaction is modeled by the Generalized Prisoner's Dilemma (GPD) [2]. GPD is a two-person game which specifies the general forms for an asymmetric payoff matrix that preserves the social dilemma. GPD is compatible with client/server structure where one player is the client and the other one is the server in each game. It is only the decision of the server which determines the ultimate outcome of the interaction.

4.3 Cooperation and Defection

We define two kinds of **Cooperation** and **Defection** in our environment: (1) Cooperation/Defection in Direct Interaction (CDI/DDI) and (2) Cooperation/Defection in Witness Interaction (CWI/DWI).

CDI/DDI have different interpretations depending on the context. In the context of e-commerce, defection in an interaction can be interpreted as the agent not satisfying the terms of a contract, selling poor quality goods, delivering late, or failing to pay the requested amount of money to a seller [8]. CWI means that the witness agent will provide a reliable and honest rating for the asking agent regarding the queried agent. In contrast, DWI means that the witness agent provides a false rating or hides its rating for the asker agent regarding the queried agent.

5 Agent Model

In this section, we present two types of trust variables which help agents determine with whom they should interact. Furthermore, three types of policies will be presented: direct interaction policy, witness interaction policy, and connection policy which assist agents in deciding how and when they should interact with another agent.

5.1 Trust Variables

Based on the two kinds of cooperation/defection explained in section 4.3, two dimensions of trust are considered. The motivation for having two trust variables is that we believe trustworthiness has different independent dimensions. For instance, an agent who is trustworthy in a direct interaction is not necessarily trustworthy in a witness interaction.

Each trust variable is defined by $T_{i,j}(t)$ indicating the trust rating assigned by agent i to agent j after t interactions between agent i and agent j, while $T_{i,j}(t) \in [-1, +1]$ and $T_{i,j}(0) = 0$. One agent in the view of the other agent can have one of the following levels of trustworthiness: *Trustworthy*, *Not Yet Known*, or *Untrustworthy*. Following Marsh [6], we define for each agent an upper and a lower threshold to model different levels of trustworthiness. The agent i has its own upper threshold $-1 \leq \omega_i \leq 1$ and lower threshold $-1 \leq \Omega_i \leq 1$. Agent j is *Trustworthy* from the viewpoint of agent i after t times of interactions if and only if $T_{i,j}(t) \geq \omega_i$. Agent i sees agent j as an *Untrustworthy* agent if $T_{i,j}(t) \leq \Omega_i$ and if $\Omega_i < T_{i,j}(t) < \omega_i$ then the agent j is in the state *Not Yet Known*.

Direct Interaction Trust (DIT). Direct Interaction Trust (DIT) is the result of CDI/DDI. Each agent maintains $DIT_{i,j}(t)$ variables for the agents having had direct interactions with them. We used the following trust updating scheme motivated by that proposed in [15]:

$$DIT_{i,j}(t+1) =$$
$$\begin{cases} DIT_{i,j}(t) + \alpha_D(i)(1 - DIT_{i,j}(t)) & DIT_{i,j}(t) > 0 \text{ , } CDI \\ (DIT_{i,j}(t) + \alpha_D(i))/(1 - min(|DIT_{i,j}(t))| \text{ , } |\alpha_D(i)|) & DIT_{i,j}(t) < 0 \text{ , } CDI \\ (DIT_{i,j}(t) + \beta_D(i))/(1 - min(|DIT_{i,j}(t))| \text{ , } |\beta_D(i)|) & DIT_{i,j}(t) > 0 \text{ , } DDI \\ DIT_{i,j}(t) + \beta_D(i)(1 + DIT_{i,j}(t)) & DIT_{i,j}(t) < 0 \text{ , } DDI \end{cases}$$

Where $\alpha_D(i) > 0$ and $\beta_D(i) < 0$ are positive evidence and negative evidence weighting coefficients respectively for updating of the direct interaction trust variable of agent i. The value of $DIT_{i,j}(t)$, ω_i^{DIT} and Ω_i^{DIT} determine that the agent j is either *trustworthy*, *Not Yet Known* or *Untrustworthy* in terms of direct interaction from the perspective of agent i.

Witness Interaction Trust (WIT). Witness Interaction Trust (WIT) is the result of the cooperation/defection that the neighbors of an agent have with the agent regarding witness interaction (CWI/DWI). Agent i maintains a $WIT_{i,j}(t)$ variable for the agent j from whom it has received witness information. The updating scheme of $WIT_{i,j}(t)$ is similar to the one presented for $DIT_{i,j}(t)$ but CDI and DDI should be replaced by CWI and DWI respectively and $\alpha_D(i) > 0$ and $\beta_D(i) < 0$ is replaced with $\alpha_W(i) > 0$ and $\beta_W(i) < 0$ respectively. Where $\alpha_W(i) > 0$ and $\beta_W(i) < 0$ are positive evidence and negative evidence weighting coefficients respectively for updating of the witness interaction trust variable of agent i. The value of $WIT_{i,j}(t)$, ω_i^{WIT} and Ω_i^{WIT} determine that the agent j is either *Trustworthy*, *Not Yet Known* or *Untrustworthy* in terms of witness interaction from the perspective of agent i.

5.2 Agent Policy Types

The perceptions introduced above allow agents to determine the trustworthiness of other agents. Policies use this information to decide upon future interactions.

Direct Interaction Policy (DIP). This type of policy assists an agent in making decisions regarding its direct interactions.

Witness Interaction Policy (WIP). This type of policy exists to aid an agent in making two categories of decisions related to its witness interactions. First, agents should decide how to provide the witness information for another agent on receiving a witness request. Should they manipulate the real information and forward false witness information to the requester (an example of defection) or should they tell the truth? The second decision is related to when and from whom the agent should ask witness information.

We defined two sub witness interaction policies: Answering policy (AP) and Querying policy (QP). The former covers the first category of decisions mentioned above while the latter is for the second category.

Connection Policy (CP). This type of policy assists an agent in making decisions regarding whether it should make a request for connection to other agents and whether the agents should accept/reject a request for a connection.

Disconnection Policy (DP). DP aids an agent in deciding whether or not it should drop a connection to a neighbor.

5.3 Experimentally Evaluated Policies

We here explain policies employed in our experiments.

Direct Interaction Policies. Three kinds of DIPs used in our experiments are: Always Cooperate (AC), Always-Defect (AD), and Trust-based Tit-For-Tat (TTFT)[1]. Agents using the AC policy for their direct interactions will cooperate with their neighbors in direct interactions regardless of the action of their neighbor. In contrast, agents using the AD policy will defect in all neighbor interactions. Agents employing TTFT will start with cooperation and then imitate the neighbors' last move as long as the neighbors are neither trustworthy nor untrustworthy. If a neighbor is known as untrustworthy, the agent will defect and if a neighbor is known as trustworthy, the agent will cooperate with it.

Connection Policy. Three kinds of connection policies are used in our experiments: Conservative (C), Naive (N), and Greedy (G). There is an internal property for each of these policies called Socializing Tendency (ST) which dramatically effects decisions for making a connection request and the acceptance of the connection request. All three connection policies use Algorithm 1 with different values for the ST variable. According to Algorithm 1, any connection request from other agents will be accepted regardless of value of ST but the agent will acquire unvisited agent IDs if its number of neighbors is less than ST. We set the value of ST to be 5, 15, and 100 for Conservative, Naive, and Greedy connection policies respectively.

Witness Interaction Policy. We have specified three kinds of **answering policies (AP)**: Honest (Ho), Liar (Li), and Simpleton (Si). All these sub-policies use the pseudo-code presented in Algorithm 2 while differentiating in the assignment of opinion variable (refer to * in Algorithm 2). The asterisk should be replaced by $DIT_{i,j}(t)$, "$-1 * DIT_{i,j}(t)$", or 1 for Honest, Liar, or Simpleton policy respectively. An agent employing the Liar policy gives manipulated ratings to other agents by giving high ratings for untrustworthy agents and low ratings for trustworthy ones. The Simpleton policy ranks all other agents as trustworthy

[1] Always Cooperate and Always Defect have been called unconditional cooperation and unconditional defection respectively in game theory literature.

Algorithm 1 Connection Policies

```
{CRQ is a queue containing the connection requests}
if CRQ is not empty then
   j = dequeue(CRQ)
   connectTo(j)
end if
if $size(neighborhood) < ST$ then
   j = get unvisited agent from list of all known agents
   if $\exists j \neq null$ then
      requestConnectionTo(j)
   end if
end if
```

but the Honest policy always tells the truth to everyone. CWI/DWI will be sent based on whether the forwarding opinion is in contradiction with the internal trust value of an agent or not. If the difference between them is less than the Discrimination Threshold (DT), an agent will send CWI otherwise DWI is sent. In this sense, Liar always defects, Honest always cooperates, and Simpleton sometimes defects (by rating high untrustworthy agents) and sometimes cooperates (by rating low trustworthy agents) in providing the witness information. Note that DT is set to the value of 0.25.

Algorithm 2 Answering Policy

```
if receiving a witness request about $j$ from $k$ then
   $opinion = *$
   send opinion to $k$
   if $|opinion - DIT_{i,j}(t)| < DT$ then
      Send CWI to $k$ after $T_W$ time steps
   else
      Send DWI to $k$ after $T_W$ time steps
   end if
end if
```

Using the **querying policy (QP)** presented in Algorithm 3, the agent asks for witness information from its all neighbors regarding one of the untrustworthy agents which has already interacted with the given agent. As a result, the agent can understand which neighbors are capable of detecting untrustworthy agents.

Disconnection Policy. We have utilized three kinds of disconnection policies in our experiments: Lenient (Le) , Moderate (Mo), and Strict (St). Using the Lenient policy, an agent will never drop any connections. An agent which uses the Moderate policy will disconnect from the neighbor known as an untrustworthy agent in terms of direct interaction. An agent with the Strict connection policy disconnects from the neighbor which is known to be untrustworthy either in direct interactions or in witness interactions.

Algorithm 3 Querying Policy

```
{BlackList: a list of known untrustworthy agents in terms of direct interactions}
if BlackList is not empty then
    $j$ = select randomly $j$ from BlackList
    Ask for witness information about $j$ from all neighbors
end if
```

6 Experiments

We have empirically analyzed our agent types on both microscopic and macroscopic levels. On the macro level, we studied how society structure changes over interactions. On the micro level, the utility of agents is examined.

$\overline{U_{AT}(i)}$, the average of utilities for agents with the type of AT at time step i, is calculated by: $\overline{U_{AT}(i)} = \frac{\sum_{a \in AT} U_{Avg}(a,i)}{N_{AT}}$, where $U_{Avg}(a,i)$ is the average of utility of agent a over its interactions at time step i and N_{AT} is the total number of agents in the society whose type is AT. The utility of each interaction is calculated as follows: If agent i defects and agent j cooperates, agent i gets the Temptation to Defect payoff of 5 points while agent j receives the Sucker's payoff of 0. If both cooperate each gets the Reward for Mutual Cooperation payoff of 3 points, while if both defect each gets the Punishment for Mutual Defection payoff of 1 point.

We have used the agent types presented in Table 1 for all experiments.

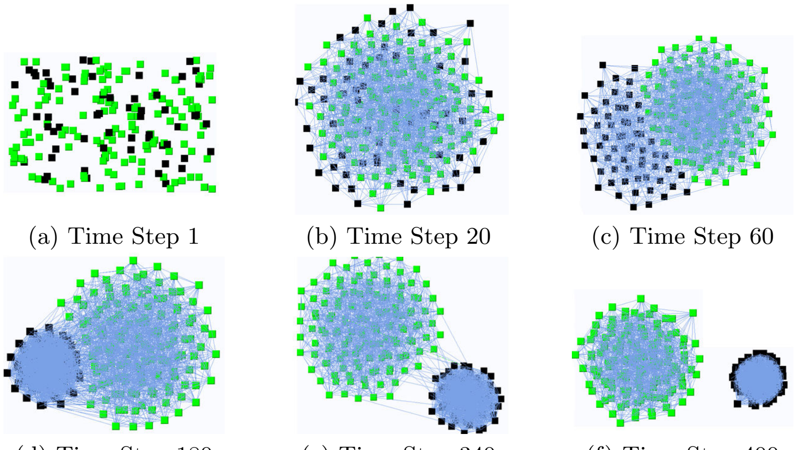
Name	Naive	Malicious	Trust-Aware(TA)	Trust-Aware$^+$(TA^+)
Trust	-	-	DIT	DIT&WIT
DIP	AC	AD	TTFT	TTFT
CP	N	G	C	C
DP	Le	Le	Mo	St
AP	Si	Li	Ho	Ho
QP	-	-	-	QP

Table 1. Agent Types and Specifications

Experiment 1. We run the simulation with the population size of 200 agents where TA agents cover 66% of population and the rest are Malicious agents. The objective of this experiment is to understand whether cooperation emerges between TA agents while they isolate themselves from Malicious agents.

Different stages of this simulation are depicted in Figure 1, where TA agents and Malicious agents are in green (light gray in white-black print) and in black respectively. Starting from an initially unconnected society (Figure 1a) Malicious agents are quickly discovered (Figure 1c) and are completely isolated by time step 400 (Figure 1f).

Experiment 2. We run 200 agents where 55%, 11% and 34% of population are TA, Naive and Malicious agents respectively. The structure of the agent society after 400 time steps is presented in Figure 2a. Malicious and Trust-Aware agents are shown with the same colors of the previous experiment and blue squares with white "+" represent Naive agents. With the introduction of


(a) Time Step 1

(b) Time Step 20

(c) Time Step 60

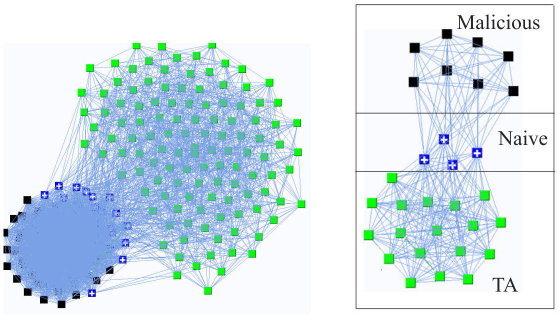
(d) Time Step 180

(e) Time Step 340

(f) Time Step 400

Fig. 1. Structural changes of Agents Society in Experiment 1

Naive agents, we could not achieve separation of Malicious and TA agents seen in Experiment 1. Since TA agents perceived Naive agents as trustworthy agents in direct interaction so they maintain their connections with Naive agents. On the other hand, since Naive agents accept all connection requests and do not drop any connections, they will be exploited by Malicious agents. As illustrated in Figure 2a, TA agents are connected indirectly to Malicious agents by means of Naive agents. Figure 2b shows Naive agents acting a buffer between the 2 other agent communities for a 30 agent simulation.



(a) 200 Agents

(b) 30 Agents

Fig. 2. The Final Society Structure in Exp. 2

Figure 3 shows the $\overline{U}$ of each agent type over the course of the simulation. $\overline{U}_{TA}$ increases over the simulation with small fluctuations. The more $\overline{U}_{TA}$ gets close to 3, the higher the proportion of interactions of TA agents are mutual cooperation. $\overline{U}_{Malicious}$ is increasing due to connecting to more Naive agents. The $\overline{U}_{Naive}$ drops over the course of simulation since the number of their connections with Malicious agents increases. All three graphs stabilize before time step 350, which is the result of not establishing new connections by any agents. Not requesting any connections can be the result of reaching the ST threshold (e.g., Naive and Trust-Aware) or scanning all of the agents (e.g., Malicious agents).

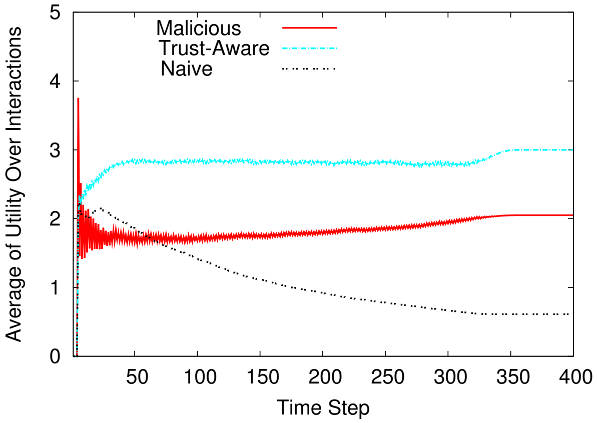


Fig. 3. $\overline{U}$ of agent types over simulation

Experiment 3. This experiment intends to show the effect of a varying proportion of Naive agents. We have run five simulations of 200 agents with different proportions of Naive and Trust-Aware agents while maintaining Malicious agents unchanged as shown in Table 2.

Agent Type	Population				
	Pop1	Pop2	Pop3	Pop4	Pop5
Malicious	34%	34%	34%	34%	34%
Naive	0%	11%	22%	33%	44%
Trust-Aware	66%	55%	44%	33%	22%

Table 2. Population Distributions of Experiment 3

Figure 4 presents $\overline{U}$ of each agent type at time step 400 for each of the runs. By increasing the proportion of Naive agents, $\overline{U}_{Malicious}$ increases considerably although the proportion of Malicious agents is unchanged. $\overline{U}_{TA}$ in all runs stays at 3 indicating that the proportion of Naive agents does not influence $\overline{U}_{TA}$. $\overline{U}_{Naive}$ increases slightly because Malicious agents have more choices to connect to Naive agents and to satisfy their ST threshold. For Pop5, the $\overline{U}_{Malicious}$ exceeds that of TA agents. As a consequence, in such societies, there is no incentive to be a Trust-aware agent since Malicious agents have better utility, that is all the outcome of having a high proportion of Naive agents in the society.

Experiment 4. We run 200 agents where 55%, 11% and 34% of the population are Trust-Aware^{+} (TA^{+}), Naive and Malicious agents respectively. The structure of the agent society at three points in the simulation are presented in Figure 5. Malicious and Naive agents are shown with the same colors of previous experiments and TA^{+} agents are presented in green. It is interesting to observe that Naive and Malicious agents are isolated from the TA^{+} agents. By using multi-dimensional trust (DIT and WIT) and the Strict disconnecting policy, TA^{+} agents could identify both Malicious and Naive agents to isolate them from their community. Naive agents are detected based on their failure to provide the appropriate witness information while Malicious agents are recognized by their defections in direct interactions.

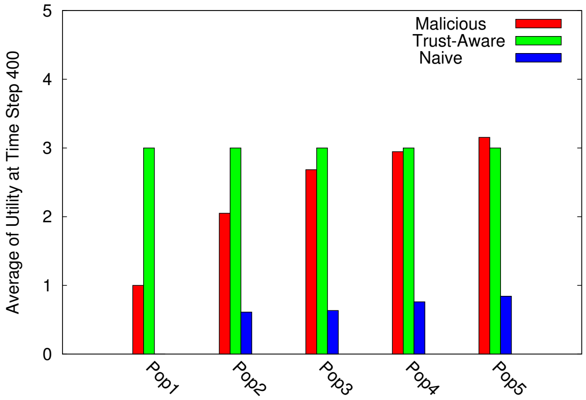
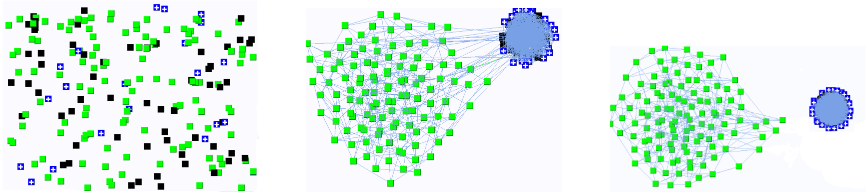


Fig. 4. $\overline{U}$ for five runs of Exp. 3



(a) Time Step 1 (b) Time Step 180 (c) Time Step 400

Fig. 5. Structural changes of Agents Society in Experiment 4

7 Discussion

The isolation of untrustworthy agents from a society of agents is considered one of the main objectives of trust models [15]. Experiment 1 demonstrates that malicious agents can be isolated using DIT when naive agents are absent. Experiments 2 and 3 demonstrate how the proportion of naive agents affects the utility of malicious agents and society structure. When this proportion exceeds some threshold, malicious agents have the best utility in the society and consequently there is no incentive for trust-aware agents to stay trustworthy. In contrast, they are motivated to be malicious to exploit naive agents too. Experiment 4 shows how adding WIT allows naive agents to be detected. In this sense, TA$^+$ agents assessed the ability of their neighbors in detecting malicious agents. Those agents which fail in this assessment turn out to be naive agents.

8 Conclusion and Future Work

Naive agents strongly degrade the value of DIT in trust-aware agent societies. Our results demonstrate that naive agents help malicious agents survive by cooperating with them directly (by providing good services) and indirectly (by giving a good rating for them).

By proposal of a set of policies and trust variables, we show that trust-aware agents need multi-dimensional trust models to separate malicious and naive agents from the trustworthy community.

We plan to extend the proposed trust model for other sources of information such as observed interactions. Then, we are interested in modeling agents which are naive in observing the results of interaction. It would be interesting to see the effect of naive agents in reputation variable (systems) where the ratings regarding the specific agents will be gathered from naive neighbors.

References

1. Robert Axelrod. *The Evolution of Cooperation.* New York: Basic Books, 1984.
2. Michal Feldman, Kevin Lai, Ion Stoica, and John Chuang. Robust incentive techniques for peer-to-peer networks. In *EC '04*, pages 102–111, New York, NY, USA, 2004. ACM.
3. Karen K. Fullam, Tomas B. Klos, Guillaume Muller, Jordi Sabater, Andreas Schlosser, Zvi Topol, K. Suzanne Barber, Jeffrey S. Rosenschein, Laurent Vercouter, and Marco Voss. A specification of the agent reputation and trust (art) testbed: experimentation and competition for trust in agent societies. In *AAMAS '05*, pages 512–518, New York, NY, USA, 2005. ACM.
4. Trung Dong Huynh, Nicholas R. Jennings, and Nigel R. Shadbolt. An integrated trust and reputation model for open multi-agent systems. *Autonomous Agents and Multi-Agent Systems*, 13(2):119–154, 2006.
5. Sirkka L. Jarvenpaa, Noam Tractinsky, and Michael Vitale. Consumer trust in an internet store. *Inf. Technol. and Management*, 1(1-2):45–71, 2000.
6. S. Marsh. Formalising trust as a computational concept, 1994.
7. L. Mui, M. Mohtashemi, and A. Halberstadt. A computational model of trust and reputation for e-businesses. In *HICSS '02*, page 188, Washington, DC, USA, 2002. IEEE Computer Society.
8. Sarvapali D. Ramchurn, Dong Huynh, and Nicholas R. Jennings. Trust in multiagent systems. *Knowl. Eng. Rev.*, 19(1):1–25, 2004.
9. J. Sabater, M. Paolucci, , and R. Conte. Repage: Reputation and image among limited autonomous partners. *Journal of Artificial Societies and Social Simulation*, 9(2), 2006.
10. Jordi Sabater and Carles Sierra. Regret: A reputation model for gregarious societies. In *Fourth Workshop on Deception Fraud and Trust in Agent Societies*, pages 61–70, 2001.
11. Jordi Sabater and Carles Sierra. Review on computational trust and reputation models. *Artif. Intell. Rev.*, 24(1):33–60, 2005.
12. Michael Schillo, Petra Funk, and Michael Rovatsos. Using trust for detecting deceitful agents in artificial societies. *Applied Artificial Intelligence, Special Issue on Trust, Deception and Fraud in Agent Societies*, 14(8):825–848, September 2000.
13. W. T. Luke Teacy, Jigar Patel, Nicholas R. Jennings, and Michael Luck. Coping with inaccurate reputation sources: experimental analysis of a probabilistic trust model. In *AAMAS '05*, pages 997–1004, New York, NY, USA, 2005. ACM.
14. Bin Yu, M.P. Singh, and K. Sycara. Developing trust in large-scale peer-to-peer systems. *Multi-Agent Security and Survivability, 2004 IEEE First Symposium on*, pages 1–10, 30-31 Aug. 2004.
15. Bin Yu and Munindar P. Singh. A social mechanism of reputation management in electronic communities. In *CIA '00*, pages 154–165. Springer-Verlag, 2000.
16. Bin Yu and Munindar P. Singh. Detecting deception in reputation management. In *AAMAS '03*, pages 73–80, New York, NY, USA, 2003. ACM.

Comprehensive Trust Management

Sandip Sen, Nick Malone, and Kuheli Chakraborty

Department of Computer Science
University of Tulsa
sandip@utulsa.edu

Abstract. Trust is an essential aspect that influences human interactions in societies. By extension, trust has been viewed as an integral component of agent decision making in the context of multiagent systems (MASs). Various formal and semi-formal trust schemes, motivated by diverse considerations and influenced by various fields of study, has been proposed, implemented, and evaluated. We believe that there still exists a pressing need for developing a comprehensive trust management scheme that addresses most, if not all, issues surrounding trust development, maintenance, and use. Accordingly, we present a general and comprehensive trust management scheme. In the process we provide our own operational definition of trust motivated by uncertainty management and utility optimization. We identify the various components required of a comprehensive trust management scheme and their relationships. We elaborate on the necessary properties and requirements of such a scheme and illustrate it by referring to existing literature on trust in MAS. We also identify Engagement as a relatively unexplored area of trust management and demonstrate how learning techniques for balancing exploration and exploitation can satisfy such a requirement. We also analyze the well-known ART testbed, developed for evaluating and comparing trust management schemes, fosters the development of comprehensive trust schemes as proposed in our framework. We further illustrate the effects of different Engagement decision modules in a procurement domain.

1 Introduction

The routine operation of human societies critically depend on the smooth and effective functioning of individual agents and their interactions. It is hard to overstate the significance of key aspects of human cognition, inference and reasoning mechanisms like learning, problem solving, planning, language understanding, etc. on the richness, robustness and vitality of human behavior in diverse societal interactions. A core component of human reasoning in societal settings is the use of *trust*. Trust is truly a multi-dimensional and multi-faceted, even somewhat nebulous, concept and is used to capture a somewhat loosely related set of influences on human decision-making. We will discuss some attempts to characterize trust and its influence on decision-making, though no one definition appears to encompass all aspects of trust in human societies as we normally perceive it. This is not particularly uncommon though among anthropomorphic concepts that have been studied in the context of computational models and is not necessarily a problem for artificial intelligence and multiagent systems researchers. For example, research on learning by AI and MAS researchers have been quite productive while branching into somewhat disjoint tracks such as supervised and unsupervised learning.

Research in computational trust models started in earnest only in the mid to late 1990s. While some researchers have attempted to formalize the role of trust in multiagent interactions, others have proposed trust models that allow agents to represent, update, and use their trust in other agents and services in their environment. Though notable advances have been made, we believe it would be productive to reflect on the aspects of trust that have received fair treatment from researchers and those that have been relatively unexplored. This reflective evaluation of the requirements of trust model and the availability of matching trust models can identify the critical needs that need to be met and fuel future trust research. Our goal in this research is then to analyze and recommend the necessary components of a comprehensive trust management scheme (CTMS) that can be used by researchers to both evaluate existing trust models and develop the next-generation trust management schemes that will be more robust and effective in handling a rich set of decision-making contexts.

We begin our analysis by considering some alternate definitions of the concept of *trust* from a computational perspective. We present some oft-quoted definitions and discuss why they are not adequate for our requirements to develop required specifications for a CTMS. Accordingly, we put forth our own definition of computational trust which captures the fundamental need of an agent to effectively handling uncertainty and optimizing performance.

Next, we consider representative examples of real-life human and agent interactions to differentiate broad categories of trust-related decisions that autonomous entities routinely undertake. We discuss, in particular, how these decisions are correlated and must be considered holistically to construct a CTMS. We, therefore, present a generic architecture for a CTMS module that identifies the relationships between these trust components. We also provide an adaptation of a generic agent architecture and discuss the integral role of the CTMS module in deciding how an agent should behave and interact with other agents in its environment.

We then review some of the well-known computational models developed by MAS researchers to identify how they match up with our proposed CTMS specifications. From this analysis we identify certain CTMS features that have been under-represented in the MAS community. In particular, we consider the *engagement* component of a CTMS whose goal is to create situations and interactions that will elicit further trust-related information about other agents. We develop an experimental framework that allows us to study the usefulness of a well-crafted engagement component on the viability of an agent in a competitive environment. We pose the engagement decision as an instance of the classical explore-exploit dilemma. Though other models of this decision are possible, we emphasize that this component of trust management needs more attention than has been received in the multiagent systems community.

We also consider the Agent Reputation and Trust (ART) testbed, a yearly competition designed to allow evaluation of competing trust schemes, as a testing ground for candidate CTMSs. Our analysis shows that the ART testbed is well-suited for carefully studying and developing effective CTMSs.

2 Trust as a Concept

The goal of individual computing entities, or agents, is to maximize local utility. To do this effectively and consistently, individual agents will need to coordinate, collaborate, and work with other agents. This often means that agents have to rely on other agents' decisions, e.g., that they fulfill negotiated commitments. Without any centralized authority or enforcement mechanisms in most of these open environments, commitments are non-binding. In addition, the likelihood of external offers and opportunities may provide short-term incentives to deviate from commitments. Hence, agents in open environments need to rely on distributed reputation and trust mechanisms that encourage agents to fulfill their commitments. Distributed trust schemes produce and maintain agent reputations reflecting their performance and trustworthiness and can support and sustain mutually beneficial medium to long-term relationships between self-interested agents.

Various definitions of trust exist in literature focusing on either the philosophical or pragmatic aspect of the concept [7, 14]. We use the following operational characterization that captures what it means for an agent to trust another agent (also see Figure 1):

> Trust in another agent reduces the uncertainty over that agent's independent actions which positively correlates with the truster's utility.

According to this interpretation of trust, trust in another agent can both reduce uncertainty about outcomes and improve performance. From a decision theoretic perspective, given a set of outcomes influenced by another agent, when the agent is trusted its behavior results in higher utility outcomes becoming more probable (correspondingly lower utility outcomes becoming less likely) and hence results in higher expected utility. If we consider risk neutral agents, then we can consider agents to choose actions according to the Maximum Expected Utility (MEU) principle [18]:

$$\arg\max_{a \in A} \sum_{o \in O} Pr(o|a, M)U(o),$$

where A is the set of actions available to the agent, O is the set of outcomes possible, M is the world model of the agent and U is its utility function over outcomes. In the context of trust management, we consider the set of outcomes to be also dependent on other agents in the environment. For the current discussion we will concentrate on bilateral interactions, and hence, outcomes are

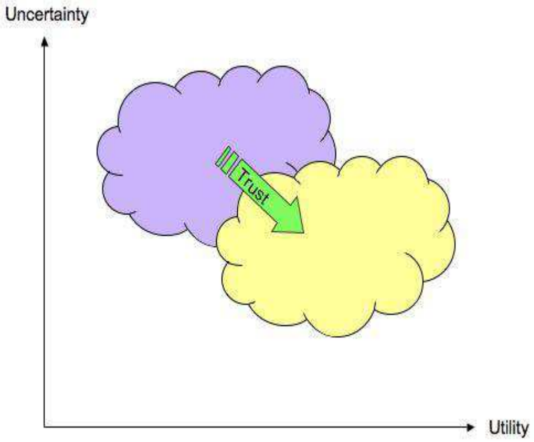


Fig. 1. The set of outcomes for an agent changes from the dark region to the light region when interacting with a trusted agent, thereby reducing uncertainty and increasing utility.

determined by the current agent and another agent and influences the trust between them. Assuming prior knowledge of the set of actions A, the set of possible outcomes, O, and the utility function, a trust model in another agent will then estimate the outcome probabilities, $Pr(o|a, M)$. Either a frequency based approach can be used to estimate these probabilities or one can use Bayesian priors and associated update rules for model updating. Often these outcomes will depend on unobservable parameters and may involve time dynamics. In such cases, Dynamic Bayesian networks with efficient approximate inference schemes like particle filtering [4] may be used to estimate these probabilities.

Typically an agent develops trust estimates of another agent both from direct interactions with that agent and from trust values reported by other agents (also called *reputation*). In particular, for various reasons often cited in favor of multiagent systems, including flexibility of use, low infrastructural overhead, robustness, etc. we are interested in reputation frameworks that are distributed and peer-level rather than centralized and monolithic.

The need for distributed trust schemes also arises in distributed systems susceptible to security threats. Malicious sources can compromise the nodes of a distributed system to undermine the performance of the entire system. The problem is compounded when multiple nodes are compromised and collude to adversely affect the system. Distributed trust schemes can be used to screen and identify irregular activities in distributed systems exposed to intrusion threats and take responsive measures to limit damage to the system from malicious intruders.

Trust is also a resource that can be leveraged to gain influence. When agent interactions are based on trust, trustworthy agents will have a larger influence

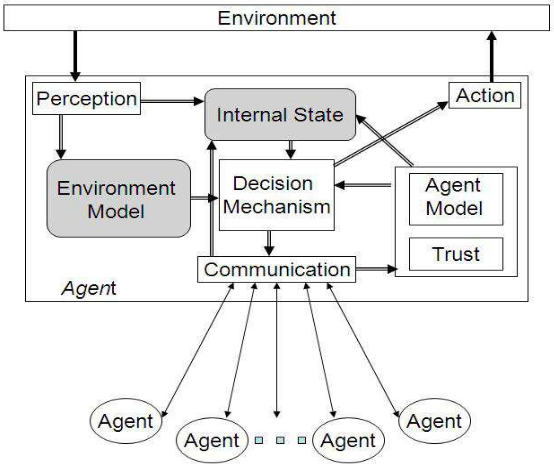


Fig. 2. Agent architecture with embedded trust module.

on negotiated outcomes. For example, agents who are trusted to provide higher quality service may demand larger fees for their services. Trust often has to be earned at a cost. For example, a manufacturing agent may have to spend extra time and resources to meet stringent delivery deadlines when upstream suppliers delay delivery of raw materials. If, however, improved trustworthiness is rewarded with additional profitable contracts, the cost expended can be recouped many times over. In such scenarios, establishing a high reputation may be a priority for rational agents. Strategic reasoning involving trust considerations will trade-off the cost of establishing and maintaining trust in the community with the future expected profits from leveraging the trust earned.

We believe that trust is a complex, multifaceted concept and involves more than merely evaluating other's trustworthiness. A more integrated approach is necessary and should additionally address engagement of others, creating situations to evaluate trust, investing resources and time to establish your own trustworthiness, strategic use of trust information, etc. Though prior research have proposed and evaluated various trust and reputation approaches that evaluates the trustworthiness of other agents, little attention is paid to a comprehensive trust management scheme. Our proposed CTMS scheme will address trust modeling, exploration, learning, as well as both tactical and strategic reasoning to achieve the desired properties of reducing uncertainty and increasing utility.

3 Comprehensive Trust Management

We now outline the basic framework of our proposed research. As mentioned above, we believe a comprehensive trust management scheme will not only address trust evaluation, but also trust establishment and use. In Figure 2 we show an agent architecture with an embedded trust management, i.e., CTMS, module. This module stores models of other agents and runs a trust management process that interfaces both with the communication module and the decision selection mechanism. Next, we further elaborate on the components of this CTMS module. These subcomponents are pictorially described in Figure 3 and their functionalities are described below:

Evaluate: *The evaluation module is in charge of evaluating the trustworthiness of another agent given its history of interaction.* This is the most frequently cited and studied aspect of trust management in the literature. A *post facto* analysis of the trustworthiness of another agent is a valuable component of agent decision making.

Establish: Trust establishment is in some respect the flip side of evaluation. *This module determines the actions and the resources to be invested to establish our agent to be trustworthy to another agent.* For example, a new supplier in the market has to determine how much time, effort, and resources to allocate to process the task/contract awarded by a lucrative customer. In some sense, this module plays as critical a role in the viability of a social agent as the trust evaluation module. Unfortunately, there is very little research existing that addresses this central trust issue.

Engage: *The Engage module enables rational agents to choose carefully and with strategic intention to interact and engage other agents for the purpose of evaluating their trustworthiness.* In practice, agents cannot depend primarily on accidental and circumstantial interactions to judge another agent's trustworthiness, i.e., they cannot be passive evaluators. Rather, an agent must make conscious decisions about which other agent to interact with. In addition, agents may have to create situations and decide on task allocations that allow for trust evaluation. For example, to evaluate a new supplier in a supply chain, a company may choose to award it some contracts. The nature, timing, and importance of the contract must be carefully chosen to allow evaluation of the competence of the new supplier without jeopardizing the production schedule or delivery deadlines for the company. The strategic creation of trust interactions that will allow for establishment or evaluation of trust is a key component of a CTMS.

Use: *This module determines how to select future courses of action based on the trust models of other agents that have been learned.* Trust considerations can influence agent decisions both in the short and the long term. Developing trust models is key but, of necessity, be coupled with an effective decision procedure to utilized this knowledge. Both tactical and strategic use of trust information is key to the competitiveness of agents in open environments. In particular, careful attention must be given to the confidence in the trust values. For example, given different interaction histories with different agents, an agent must carefully balance exploitation of existing trust knowledge and investment in exploration for gathering trust models about relative newcomers in the environment.

To demonstrate the emphasis on the "evaluate" sub-module, we briefly discuss two recently developed trust models that are often cited:

FIRE: The FIRE [16] model is primarily a utility evaluation model because of

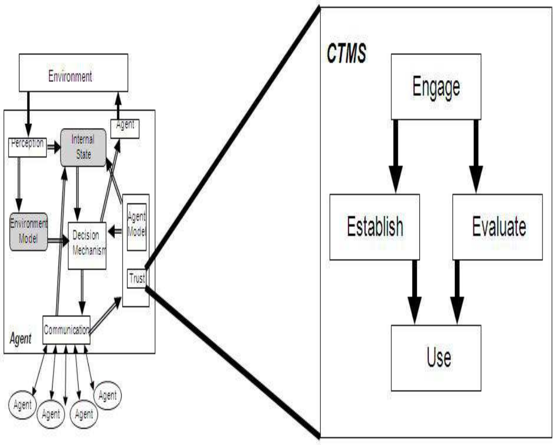


Fig. 3. Principal components of the embedded trust module in an agent architecture.

the following assumptions: all agents are honest and all agents are willing to share. These two assumptions mean there is no need to utilize a central aspect of the CTMS, establishing trust. The current FIRE model just gathers utility information via four methods, direct experience, witness information, role based rules and third party reference. FIRE calculates a weighted mean of each of these information types and then creates a composite score. FIRE creates trust situations (engage submodule) via a Boltzmann distribution exploration strategy. It will chose to either explore a new provider create a trust situation) or use its trust knowledge to select the provider that delivers the highest utility.
TRAVOS: TRAVOS [29], like FIRE, is primarily a trust evaluation model. It includes lying agents but does not seek to establish trust. The model does not include any strategic reasoning about if it should lie or attempt to tell the truth. TRAVOS uses direct trust and reputation to evaluate the trustworthiness of an agent. Furthermore the TRAVOS model does not consider the utilization of the trust calculation. However, the reputation methodology is classified as creating trust situations.

4 Existing Trust Models

Multiagent systems are studied to help us understand how agents should behave in the presence of other agents. These interactions might be either cooperative, competitive or simply co-existential in nature. With the need to coordinate their actions, agents should be able to communicate with one another and learn from their interactions. Researchers have worked on developing effective communication protocols and efficient learning algorithms in different social environments. One of the most challenging issues in open multiagent environment is the issue of trust and reputation among agents [7, 23, 27, 31, 16, 10]. Therefore, one of the critical research issues in multiagent systems involves how we learn to trust other agents and how we build and maintain reputations in an agent society.

Castelfranchi and Falcone have argued the necessity of trust in social interactions between agents with complex mental attitudes and identified the benefits of being trusted [7, 8]. They argue that trust can be based on mental background, and though it necessarily entails risks for delegation and collaboration, considerations of morality and use of reputation can be used to mitigate that risk. Castelfranchi, Conte, and Paolucci use normative reputation [6] to enhance the performance of agents that comply with social norms. An example of using morality to promote social relationships, as suggested by Castelfranchi and Falcone, can be found in the SPIRE framework developed by Grosz and collaborators [15, 28].

The application of learning to the problem of trust and reputation management has received increasing attention from researchers in intelligent systems and game theory [2, 5]. Fullam [12] has shown how environmental rewards can be used to learn comprehensive trust strategies. Her work identifies interdependencies which exist since agents can influence each other by exchanging reputation. To reduce such interdependencies, assumptions are introduced, and rewards are attributed to the decision which facilitates reinforcement learning of trust decision strategy. Other researchers, such as Sen [24], showed that adaptive probabilistic reciprocity strategies can be used to develop and sustain trust and cooperation between self-interested agents. Even though reciprocity do not address the problem of task specific learning we are interested in, it shows how trust relations between agents can be developed and sustained in order to facilitate efficient decision making and identify exploitative agents in a system.

Referral systems have recently received increasing attention among multiagent researchers. In [33] Yu and Singh study a referral system when an agent helps a human user find relevant expertise and protect him/her from too many irrelevant requests. Singh and his students have also studied the management of reputation in such distributed referral systems [30, 32]. Sen and his students have studied the use of referrals to locate service providers when an agent first enters a new community with no prior knowledge of the quality of service providers or the reliability of the referrers [3, 27]. More recent work on trust models incorporate divergent approaches including information-theoretic and fuzzy approaches to trust metrics [9, 16, 19, 21].

A significant body of work by mathematical biologists or economists on the evolution of altruistic behavior deals with the idealized problem called the Prisoner's dilemma [20] or some other repetitive, symmetrical, and identical 'games.' To consider a well-known study in this area, Axelrod demonstrates that a simple, deterministic reciprocal scheme or the *tit-for-tat* strategy is quite robust and efficient in maximizing local utility [1]. We have argued that the simple reciprocative strategies are inappropriate for most real-life situations because the underlying assumptions are violated [24, 25]. We have shown that agents with complementary expertise can learn to form stable, mutually beneficial partnerships [11]. The evaluation framework used by Axelrod considers an evolving population composition by allowing propagation of more successful behaviors and elimination of unsuccessful ones. We have studied the emergence of dominant or evolutionarily stable behaviors in such environments [22, 26]. We have also considered the use of reciprocity to promote beneficial relationships between agent groups [17].

Most of the trust research referenced above focus primarily on the Evaluate and Use aspects of the CTMS framework. We believe there is significant research issues that need sustained research focus on the somewhat neglected aspects of comprehensive trust models. In particular, engagement and establishment need to be carefully studied with integrated trust based decision schemes that are informed by and, in turn, inform the usage and evaluation of trust of other agents in the environment.

5 Trust decisions in the ART Testbed

To better ground the discussion of our techniques, we first briefly review the Agent Reputation and Trust (ART) testbed [13], an international trust-competition testbed, that has been proposed to evaluate alternative trust strategies. The ART competition uses an artwork appraisal domain, where agents evaluate paintings for their clients. In each run, an appraiser agent has a set of clients, and the agent's work is to provide an appraisal for a painting presented by its client. A given painting may belong to a finite set of eras, and appraisers have varying level of expertise in each era. An appraiser's expertise is described by a normal distribution of the error between the appraiser's opinion and the true painting value. This normal distribution has a mean of zero and a standard deviation s given by $s = \left(s^* + \frac{\alpha}{c_g}\right) t$ where s^* is unique for each era, t is the true value of the painting to be appraised, α is a parameter, and c_g is the cost expended by the agent to generate the opinion. Later we discuss strategic investment in such costs to improve trust ratings.

To improve their performance, agents might seek opinions from other agent(s) where it will incur a cost of c_p per opinion request. Before seeking help from a particular agent, agents can seek the reputation of that agent from other agents in the system. The cost associated with this reputation transaction is c_r and the following holds true: $c_r \ll c_p \ll f$, where f is the fixed fee paid by the clients for each appraisal request.

Initially clients are evenly distributed among appraisers. Those appraisers whose final appraisals were most accurate are rewarded with a larger share of the client base in subsequent time steps. The rewarding scheme in ART first computes the average relative error ϵ_a for an appraiser a as $\epsilon_a = \frac{\Sigma_{c \in C_a} \frac{|p_c^* - t_c|}{t_c}}{|C_a|}$, where C_a is the number of clients a has in the current time step, and $\frac{|p_c^* - t_c|}{t_c}$ is its performance error for client c. Then client shares are altered based on the relative accuracy of agent performances.

To illustrate the benefits of the CTMS framework, we focus on the following requirements of agents in the ART framework and identify the CTMS modules active in fulfilling those requirements: (a) an agent needs to carefully decide on how to react to opinion requests from other agents (Establish/Engage), (influences its reputation in the community), (b) an agent needs to develop trust models of other agents who might individually or in colluding groups report false opinions to adversely affect the agent's performance and hence its trustworthiness to customers (Evaluate), (c) agents need to strategically invest locally to generate opinions as that influences its performance and trustworthiness to customers (Establish), (d) given its understanding of its own competency and others' trustworthiness, decide how to evaluate a painting be combining own effort and seeking others opinions (Use/Engage).

6 Procurement Domain

We now introduce a domain which will be used to evaluate alternative trust engagement decision functions that can be used by agents to create interaction opportunities to better evaluate trustworthiness of possible trading partners. The domain consists of purchasing agents trying to procure required amounts of needed items (this can be goods or services depending on the domain of application) from a set of service agents with varying capabilities. The goal of the purchasing agents is to fulfill their requirements at minimum cost. We will describe the trust engagement strategies from the view of a single purchasing agent, P.

When a purchasing agent hires the services of a service agent, it incurs a fixed cost, C. The service agent, S, provides a quantity, U_S, of the requested item. U_S is drawn from a Normal Distribution, $N(\mu_S, \sigma_S)$. Agent P can only observe U_S but not μ_S or σ_S. The total requirement of the purchasing agent is given by R.

For our experiments, we draw μ_S from the uniform distribution, $U(0,1)$, and σ_S is set to $\frac{\mu_S}{\Delta \times Max\sigma + Min\sigma}$, where Δ is drawn from the Uniform distribution $U(0,1)$ and $Max\sigma$ and $Min\sigma$ are system parameters.

The utility function is as follows:

$$U = |(\Sigma_{S\epsilon\rho}(U_S) - R)| - C \cdot |\rho|,$$

where ρ is the set of service agents the purchasing agent has selected to buy from this round.

We also assume an open environment, allowing for x% of the service agents to be replaced by new agents every iteration. The purchasing agent's goal is then to maximize its utility by learning expected utility of interacting with different service agents, and subsequently choose to interact those with the highest expected utility. While this characterization emphasizes the Evaluate and Use modules of CTMS, we emphasize that accurately estimating the expected utility of a service agent requires strategic behavior and falls within the purview of the Engage module of CTMS. In particular, the Engage module must build the initial estimates of existing service agents and continue to "explore" newly arriving service agents, to guarantee long-term viability, without risking significant short-term utility loss.

7 Engagement strategies

This balancing act of long-term and short-term priorities can be addressed by alternate engagement strategies of the CTMS framework. Recall that the goal of the CTMS scheme is to trade with a set of service providers to fulfill requirements while minimizing cost. A myopic approach to this problem would be to develop initial estimates of the existing service agents through random sampling and then contracting with only the most profitable service providers. This strategy is ineffective in practice because of the turnover in the service provider population.

A key question is what to do once an initial estimate of the service provider population is obtained via random interactions. The purchasing agent must then decide how to utilize the expected utilities to minimize cost while still fulfilling its requirement. This is the purview of the Engagement module of the CTMS. We now present four different Engagement strategies with varying risk attitudes: Risk Averse, Risk Neutral, Risk Seeking, and Random. Each strategy only differs in the manner of provider selection and consequently the amount of exploration.

All strategies initially random chooses $\frac{requirement}{\hat{\mu}}$ service agents, where $\hat{\mu}$ is the average expected quantity returned by service agents. $\hat{\mu} = .5$ since μ_S's are drawn from $U(0, 1)$. We use $\frac{R}{\hat{\mu}_S}$ as the average number of agents needed to fulfill requirement, and $\frac{R}{\hat{\mu}_S} \times C$, as the average cost needed to fulfill the requirement.

```
Sort known Providers by $\mu$ from highest to lowest
for Sorted Providers do
    if Requirement Met > Total Requirement then
        break;
    end
    Requirement Met ← Requirement Met + $(\hat{\mu}_{Provider} - \hat{\sigma}_{Provider})$
end
while Number of Providers Selected < $\frac{requirement}{\hat{\mu}}$ do
    Randomly select an additional provider to buy from
end
```

Algorithm 1: Risk Adverse Agent's Provider Selection Algorithm

The Risk Averse, Risk Neutral and Risk Seeking strategies use the utilities received from selected service providers to estimate $\hat{\mu}_S$ and $\hat{\sigma}_S$ for each provider. They then sort the providers in descending order by $\hat{\mu}_S$. The *Risk Averse* Engagement strategy traverses down the ordered list summing up $(\hat{\mu}_S - \hat{\sigma}_S)$ from each service provider until the sum is greater than or equal to R (see Algorithm 1). The risk averse agent then asks more agents, thereby incurring additional costs, than is warranted to ensure that its requirement is fulfilled. If the number of providers selected is less than the average number of providers needed to fulfill the requirement, more providers are selected randomly. The selected providers are then sent service requests. Each provider returns an amount based on their personal capabilities. The purchasing agent can then use these utilities to update its model about each selected provider.

The Risk Neutral Agent follows the same algorithm as the Risk Averse except it sums the $\hat{\mu}_S$ values from each agent until the requirement is met. Similarly the Risk Seeking Agent sums $(\hat{\mu}_S + \hat{\sigma}_S)$. The random agent, on the other hand, selects providers randomly and increments the expected quantity by $\hat{\mu}_S$ until the cumulative expected amount exceeds R. In the next section, we present experimental results to evaluate these alternative Engagement strategies.

8 Experimental Results

For our experiments, we use 50 service providers and $C = 0.1$. Also, 5% of the providers were replaced every iteration. Results, averaged over 1000 runs, are presented for different requirements and σ_S values in Figures 4 through 9. All the engagement strategies converge to a fairly stable utility returns within about 200 iterations.

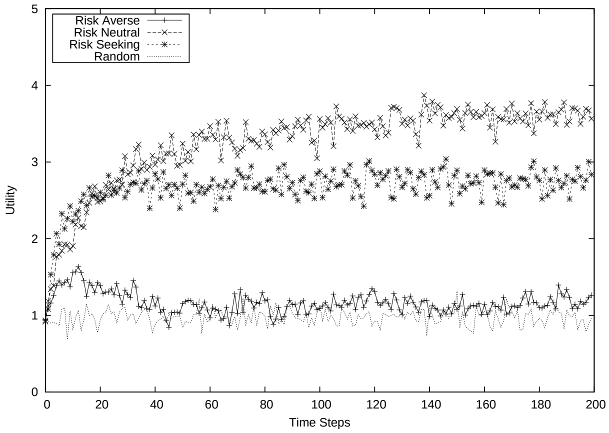


Fig. 4. Constant Requirement (3) with $\sigma = \frac{\mu}{\Delta \times 3+1}$, $\Delta \in$ [0, 1], $\mu \in$ [0,1]

The first three figures correspond to the purchasing agent having a fixed requirement of $R = 3$ at each iteration. The following three figures correspond to the purchasing agent's requirement varying from iteration to iteration, each time drawn from the uniform distribution, $U(1, 4)$. While the former corresponds to purchasing agents operating in a stable economy with fixed demands, the latter case corresponds to volatile environments with highly variable demands.

We observe that the performance of all engagement strategies are impacted by the sigma spread used to generate the providers' performance functions. Overall, we find the Risk Neutral strategy is superior to the other strategies. Furthermore,

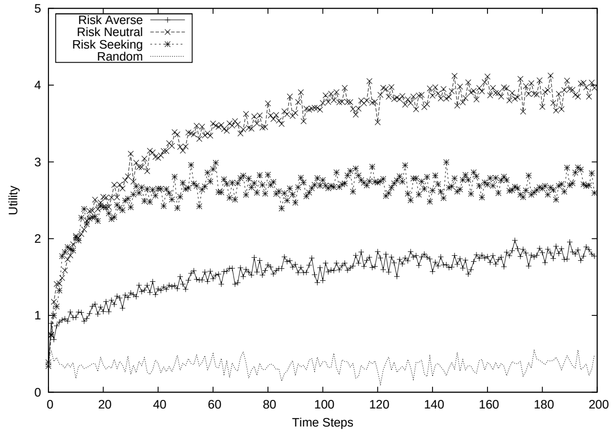


Fig. 5. Constant Requirement (3) with $\sigma = \frac{\mu}{\Delta \times 9+1}$, $\Delta \in [0, 1]$, $\mu \in [0,1]$

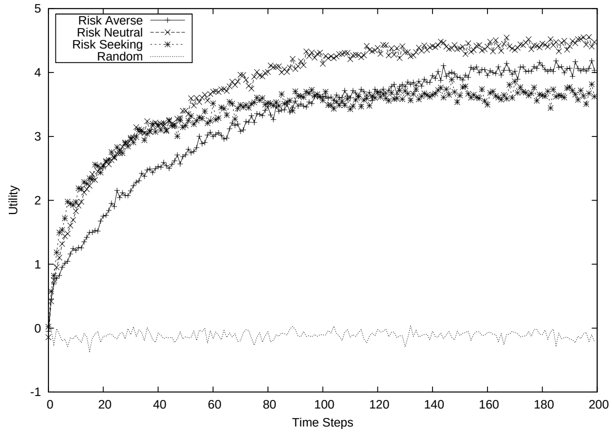


Fig. 6. Constant Requirement (3) with $\sigma = \frac{\mu}{\Delta \times 10+15}$, $\Delta \in [0, 1]$, $\mu \in [0,1]$

as the sigma variance becomes smaller, the performance of both the Risk Averse and Risk seeking strategies approach that of the Risk Neutral strategy. As σ_s reduces, in effect, the overlap between the service providers selected by the three strategies increases.

It is interesting to note that the Risk Averse strategy actually gains more with decreasing σ_S. In particular, it performs very poorly, almost equivalent to the random agent, for relatively high performance deviations (see Figure 4). The reason for this phenomena is that the risk averse strategy pessimistically expects little return from even providers with high μ_S values. Hence it has to ask more

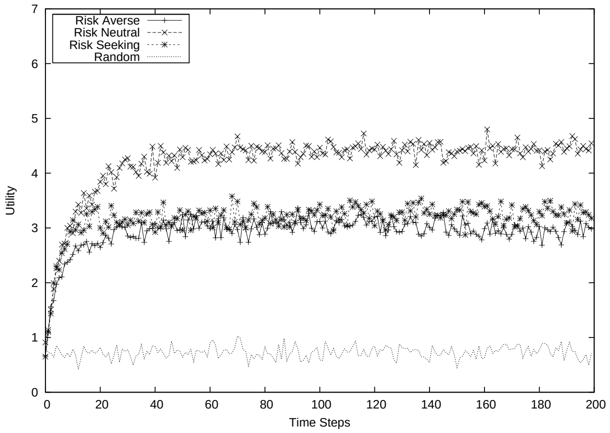


Fig. 7. Changing Requirement, drawn from $U(1, 4)$, with $\sigma = \frac{\mu}{\Delta \times 3+1}$, $\Delta \in [0,\ 1]$, $\mu \in$ [0,1]

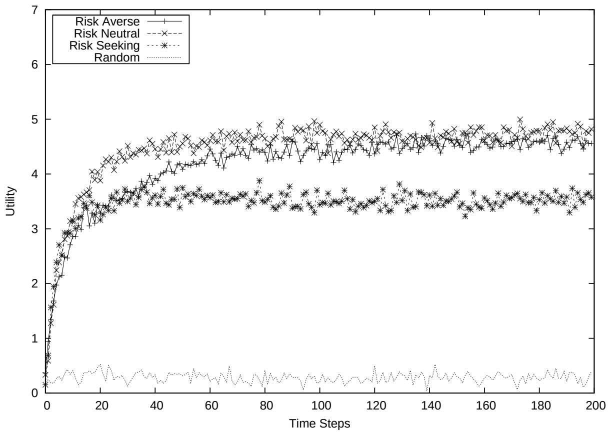


Fig. 8. Changing Requirement, drawn from $U(1, 4)$, with $\sigma = \frac{\mu}{\Delta \times 9+1}$, $\Delta \in [0,\ 1]$, $\mu \in$ [0,1]

agents from the ordered list of agents and has little opportunity to engage and hence evaluate newly arriving providers. Thus, in its haste to improve short-term utility, the risk averse agent loses out on long-term utility. As new agents replace current agents, assuming their ids, the risk averse agent sticks to the same agents. This explains the initial rise and subsequent fall of the utilities obtained by the risk averse agent. Also, as newly arriving agents are assigned random μ_S values, the performance of the risk averse agents converges to that of random engagement decisions.

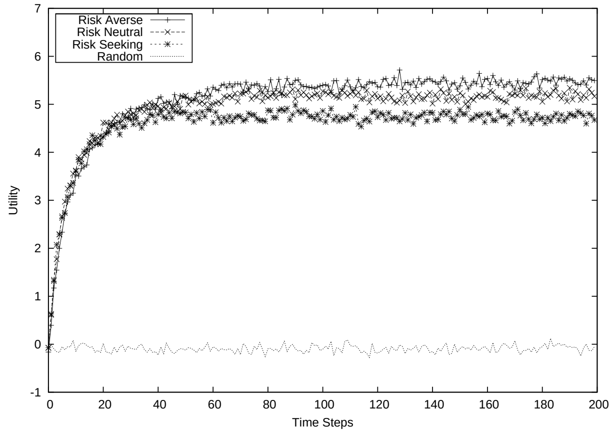


Fig. 9. Changing Requirement, drawn from $U(1, 4)$, with $\sigma = \frac{\mu}{\Delta \times 10 + 15}$, $\Delta \in$ [0, 1], $\mu \in$ [0,1]

Similar results can be observed with changing requirement. However, the Risk Averse Engagement strategy does better under changing requirement than it does with constant requirement. With changing requirement, the Risk Averse strategy is sometime forced to explore more than at other times. This additional exploration is sufficient to improve its performance to match that of the Risk Seeking Engagement strategy for the environment with the highest variation in service provider performance (see Figure 7).

While it appears that σ is causing the strategies to differ, all of these observations are connected directly to the method of exploration, an immediate consequence of the Engagement strategies used here. For example, under the constant requirement conditions, the number of random explorations (agents engaged after expected fulfillment of requirement), in increasing order is Risk Averse $\longrightarrow$ Risk Neutral $\longrightarrow$ Risk Seeking $\longrightarrow$ Random. Exploration in the system highly impacts the performance of the buyer agents as it enables the purchasing agent to develop better estimates of all service providers in the population (the random agent does not utilize this learned knowledge and hence continues to perform poorly). The well-recognized exploration-exploitation dilemma in intelligent systems is hence highly influential in the efficacy of engagement decisions and should be carefully considered when designing the trust components of intelligent agents.

9 Conclusions

We have argued for the development and use of a comprehensive trust management system as an integral component of intelligent agent architectures. We introduced a holistic conceptual view of trust decisions as reducing uncertainty and improving utility and shown that such a characterization nicely dovetails into a decision-theoretic design of a rational agent. We analyzed the requirements of an effective CTMS design and identified the corresponding fundamental modules. We reviewed some well-known trust mechanisms and existing trust literature to highlight the current emphasis on only some of the identified CTMS modules. In particular, engagement and establishment decisions decisions are often neglected or unspecified, but can be determining factors behind the success or failure of implemented trust-based systems. We then evaluate the effects of several engagement decision mechanisms in an open trading environment.

This paper begins a dialog for a more comprehensive treatment of trust mechanisms and their use in intelligent agents. Further investigation is needed to identify important submodules of the four major trust modules identified here, and may be even other modules of significance. Novel, innovative engagement and establishment decision mechanisms need to be developed and tested in conjunction with Evaluate and Use modules used by existing frameworks. Moreover, a holistic treatment of trust with clear identification of and contribution to the dual goals of uncertainty reduction and utility maximization should lead to principled trust module designs that work seamlessly with other components of the intelligent agent architecture such as learning, negotiation, etc. We believe that such integrated trust-based reasoning is essential towards developing more robust and flexible agent designs for future, challenging applications.

Acknowledgment This work is supported in part by a DOD-Army Research Office Grant #W911NF-05-1-0285.

References

1. R. Axelrod. *The Evolution of Cooperation.* Basic Books, 1984.
2. B. Banerjee and S. Sen. Selecting partners. In *Game Theory and decision theory in agent-based systems*, pages 29–42. Simon Parsons, Piotr Gmytrasiewicz and Michael Wooldridge, Kluwer, 2002.
3. T. Candale and S. Sen. Effect of referrals on convergence to satisficing distributions. In F. Dignum, V. Dignum, S. Koenig, S. Kraus, M. P. Singh, and M. Wooldridge, editors, *Proc. 4th International Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS 2005)*, pages 347–354. ACM, 2005.
4. O. Cappe, E. Moulines, and T. Ryden. *Inference in Hiden Markov Models.* Springer, 2005.
5. D. Carmel and S. Markovitch. Exploration strategies for model-based learning in multi-agent systems. *Autonomous Agent and Multi-agent Systems*, 2(2):141–172, 1999.
6. C. Castelfranchi, R. Conte, and M. Paolucci. Normative reputation and the costs of compliance. *Journal of Artificial Societies and Social Simulation*, 1(3), 1998.
7. C. Castelfranchi and R. Falcone. Principles of trust for MAS: Cognitive autonomy, social importance, and quantification. In *Proceedings of the Third International Conference on Multiagent Systems*, pages 72–79, Los Alamitos, CA, 1998. IEEE Computer Society.

8. C. Castelfranchi, R. Falcone, and F. Marzo. Being trusted in a social network: Trust as a relational capital. In *Proceedings of iTrust*, pages 19–32, 2006.
9. R. K. Dash, S. D. Ramchurn, and N. R. Jennings. Trust-based mechanism design. In *Proceedings of the Third Intenational Joint Conference on Autonomous Agents and Multiagent Systems*, pages 748–755, New York, NY, 2004. ACM Pres.
10. S. D.Ramchurn, D. Huynh, and N. R. Jennings. Trust in multiagent system. *The Knowledge Engineering Review*, 19(1):1–25, 2004.
11. P. S. Dutta and S. Sen. Forming stable partnerships. *Cognitive Science Research*, 4(3):211–221, 2003.
12. K. Fullam. Learning complex trust decision strategies. In *Autonomous Agent and Multiagent Systems Conference*, page 1241, 2006.
13. K. K. Fullam, T. B. Klos, G. Muller, J. Sabater, A. Schlosser, Z. Topol, K. S. Barber, J. S. Rosenschein, L. Vercouter, and M. Voss. A specification of the agent reputation and trust (ART) testbed: Experimentation and competition for trust in agent societies. In *Proceedings of the Fourth Intenational Joint Conference on Autonomous Agents and Multiagent Systems*, pages 512–518, New York, NY, 2005. ACM Pres.
14. D. Gambetta. *Trust*. Basil Blackwell, Oxford, 1990.
15. A. Glass and B. Grosz. Socially conscious decision-making. In *Proceedings of the Fourth International Conference on Autonomous Agents*, pages 217–224, New York, NY, 2000. ACM Press.
16. T. D. Huynh, N. R. Jennings, and N. R. Shadbolt. An integrated trust and reputation model for open multi-agent systems. *Journal of Autonomous Agents and Multi-Agent Systems*, 13(2):119–154, 2006.
17. P. P. Kar, S. Sen, and P. S. Dutta. Effect of individual opinions on group interactions. *Connection Science*, 14(4):335–344, 2002.
18. J. V. Neumann and O. Morgenstern. *Theory of games and economic behavoir*. Princeton University Press, New Jersey, 1944.
19. S. D. Ramchurn, N. R. Jennings, C. Sierra, and L. Godo. Devising a trust model for multi-agent interactions using confidence and reputation. *Applied Artificial Intelligence*, 18(9-10):833–852, 2004.
20. A. Rapoport. Prisoner's dilemma. In J. Eatwell, M. Milgate, and P. Newman, editors, *The New Palgrave: Game Theory*, pages 199–204. Macmillan, London, 1989.
21. J. Sabater and C. Sierra. Reputation and social network analysis in multi-agent systems. In *Proceedings of the First Intenational Joint Conference on Autonomous Agents and Multiagent Systems*, pages 475–482, New York, NY, 2002. ACM Pres.
22. S. Saha and S. Sen. Predicting agent strategy mix of evolving populations. In *Proceedings of the Fourth Intenational Joint Conference on Autonomous Agents and Multiagent Systems*, pages 1075–1082, 2005.
23. M. Schillo, P. Funk, and M. Rovatsos. Using trust for detecting deceiptful agents in artificial societies. *Applied Artificial Intelligence*, 14:825–848, 2000.
24. S. Sen. Reciprocity: a foundational principle for promoting cooperative behavior among self-interested agents. In *Proceedings of the Second International Conference on Multiagent Systems*, pages 315–321, Menlo Park, CA, 1996. AAAI Press.
25. S. Sen. Believing others: Pros and cons. *Artificial Intelligence*, 142(2):179–203, 2002.
26. S. Sen and P. S. Dutta. The evolution and stability of cooperative traits. In *Proceedings of the First Intenational Joint Conference on Autonomous Agents and Multiagent Systems*, pages 1114–1120, New York, NY, 2002. ACM Pres.

27. S. Sen and N. Sajja. Robustness of reputation-based trust: Boolean case. In *Proceedings of the First Intenational Joint Conference on Autonomous Agents and Multiagent Systems*, pages 288–293, New York, NY, 2002. ACM Pres.
28. D. G. Sullivan, B. Grosz, and S. Kraus. Intention reconciliation by collaborative agents. In *Proceedings of the Fourth International Conference on Multiagent Systems*, pages 293–300, Los Alamitos, CA, 2000. IEEE Computer Society.
29. W. T. L. Teacy, J. Patel, N. R. Jennings, and M. Luck. Travos: Trust and reputation in the context of inaccurate information sources. *Journal of Autonomous Agents and Multi-Agent Systems*, 12(2):183–198, 2006.
30. P. Yolum and M. P. Singh. Engineering self-organizing referral networks for trustworthy service selection. *IEEE Transactions on System, Man, and Cybernetics*, 2005.
31. B. Yu and M. P. Singh. Distributed reputation management for electronic commerce. *Compuational Intelligence*, 18(4):535–549, 2002.
32. B. Yu and M. P. Singh. Detecting deception in reputation management. In *Proceedings of the Second Intenational Joint Conference on Autonomous Agents and Multiagent Systems*, pages 73–80, New York, NY, 2003. ACM Pres.
33. B. Yu and M. P. Singh. Searching social networks. In *Proceedings of the Second Intenational Joint Conference on Autonomous Agents and Multiagent Systems*, pages 65–72, New York, NY, 2003. ACM Pres.