Intelligence Integration in Distributed Knowledge Management

Dariusz Król
Wroclaw University of Technology, Poland

Ngoc Thanh Nguyen
Wroclaw University of Technology, Poland

Information Science REFERENCE

INFORMATION SCIENCE REFERENCE
Hershey · New York

Director of Editorial Content:	Kristin Klinger
Managing Development Editor:	Kristin M. Roth
Senior Managing Editor:	Jennifer Neidig
Managing Editor:	Jamie Snavely
Assistant Managing Editor:	Carole Coulson
Copy Editor:	Lanette Ehrhardt
Typesetter:	Jeff Ash
Cover Design:	Lisa Tosheff
Printed at:	Yurchak Printing Inc.

Published in the United States of America by
Information Science Reference (an imprint of IGI Global)
701 E. Chocolate Avenue, Suite 200
Hershey PA 17033
Tel: 717-533-8845
Fax: 717-533-8661
E-mail: cust@igi-global.com
Web site: http://www.igi-global.com

and in the United Kingdom by
Information Science Reference (an imprint of IGI Global)
3 Henrietta Street
Covent Garden
London WC2E 8LU
Tel: 44 20 7240 0856
Fax: 44 20 7379 0609
Web site: http://www.eurospanbookstore.com

Copyright © 2009 by IGI Global. All rights reserved. No part of this publication may be reproduced, stored or distributed in any form or by any means, electronic or mechanical, including photocopying, without written permission from the publisher.

Product or company names used in this set are for identification purposes only. Inclusion of the names of the products or companies does not indicate a claim of ownership by IGI Global of the trademark or registered trademark.

Library of Congress Cataloging-in-Publication Data

Intelligence integration in distributed knowledge management / Dariusz Krol and Ngoc Thanh Nguyen, editors.

p. cm.

Includes bibliographical references and index.

Summary: "This book covers a broad range of intelligence integration approaches in distributed knowledge systems, from Web-based systems through multi-agent and grid systems, ontology management to fuzzy approaches"--Provided by publisher.

ISBN 978-1-59904-576-4 (hardcover) -- ISBN 978-1-59904-578-8 (ebook)

1. Expert systems (Computer science) 2. Intelligent agents (Computer software) 3. Electronic data processing--Distributed processing. I. Krol, Dariusz. II. Nguyên, Ngoc Thanh.

QA76.76.E95I53475 2009

006.3--dc22

2008016377

British Cataloguing in Publication Data
A Cataloguing in Publication record for this book is available from the British Library.

All work contributed to this book set is original material. The views expressed in this book are those of the authors, but not necessarily of the publisher.

If a library purchased a print copy of this publication, please go to http://www.igi-global.com/agreement for information on activating the library's complimentary electronic access to this publication.

Chapter VIII
How Can We Trust Agents in Multi-Agent Environments?
Techniques and Challenges

Kostas Kolomvatsos
National and Kapodistrian University of Athens, Greece

Stathes Hadjiefthymiades
National and Kapodistrian University of Athens, Greece

ABSTRACT

The field of Multi-agent systems (MAS) has been an active area for many years due to the importance that agents have to many disciplines of research in computer science. MAS are open and dynamic systems where a number of autonomous software components, called agents, communicate and cooperate in order to achieve their goals. In such systems, trust plays an important role. There must be a way for an agent to make sure that it can trust another entity, which is a potential partner. Without trust, agents cannot cooperate effectively and without cooperation they cannot fulfill their goals. Many times, trust is based on reputation. It is an indication that we may trust someone. This important research area is investigated in this book chapter. We discuss main issues concerning reputation and trust in MAS. We present research efforts and give formalizations useful for understanding the two concepts.

INTRODUCTION

The technology of Multi-agent systems (MAS) offers a lot of advantages in computer science and more specifically in the domain of cooperative problem solving. **MAS** are systems that host a number of autonomous software programs that are called agents. **Agents** act on behalf of their owners giving them access to information resources easily and efficiently. Users state their requirements and agents are responsible to fulfill them. Hence, MAS include many entities trying

Copyright © 2009, IGI Global, distributing in print or electronic forms without written permission of IGI Global is prohibited.

to solve their problems that are beyond of their capabilities. For this reason, in many cases, agents must cooperate with others in order to find the appropriate information and services to achieve their goals.

It is obvious that **MAS** are dynamic and distributed environments where agents may cooperate and communicate with others in order to complete their tasks. A key challenge arises from this nature of MAS. In such open systems, entities change their behavior dynamically. Thus, there is a requirement for trust between agents when they must exchange information Therefore, the basic question in such cases is: How and when can we trust an agent? Agents, in the majority of cases are selfish and their intentions and beliefs change continually.

We try to address this dilemma throughout this chapter. Specifically, we cover the fields of reputation and trust in MAS. This is an active research area, which is very important due to the fact that these two concepts are used in commercial applications. However, open issues exist in many cases, as it is difficult to characterize an agent as reliable or not.

In our work, we try to provide a detailed overview of reputation and trust models highlighting their importance to open environments. Due to the abundance of the relevant models, only the basic characteristics of models are discussed. We discuss basic concepts concerning MAS, reputation and trust. Accordingly, we present efforts, formalizations, and models related to the mentioned concepts. Finally, we discuss about trust engineering issues and we present future challenges and our conclusions.

BACKGROUND

Multi-Agent Systems (MAS)

Software agents and agency have been active research areas for many years due to their importance in various domains. The Web and the recently emerged Semantic Web are the most appropriate examples of such systems. In this section, basic characteristics of MAS are described. Our goal is to provide necessary knowledge about these systems and their requirements for security.

With the rapid evolution of the Internet, Software agents are a very important research area in Computer Science. **Software agents** are components of software or hardware which are capable of acting on behalf of a user in order to accomplish tasks (Nwana, 1996). The owner of an agent may be a human or another computational entity. Tasks are requested by the owners of agents in order to fulfill their needs. There are different kinds of agents. One can meet information agents that search for information sources, mobile agents that move from an environment to another, intelligent agents that can learn from their owners and the environment and so forth. For an extensive discussion of the different types of agents one can refer to Nwana (1996).

In the most cases, agents must deal with complicated tasks that demand cooperation with others. A **Multi-agent system (MAS)** can be defined as a loosely coupled network of problem solvers that interact to solve problems that are beyond the individual capabilities or knowledge of each problem solver (Durfee & Lesser, 1989). In such systems agents can cooperate or compete with others to complete their tasks. We must note that such systems are open. An open system is one in which the structure of the system is capable of dynamically changing (Sycara, 1998). In open MAS, the basic components may change over time such as information sources or agents' behaviors. From this point of view, it can be assumed that in open MAS (Huynh, Jennings, & Shadbolt, 2006):

- Agents have different owners and for this reason they are selfish and may be unreliable;

- There is no knowledge about the environment in which agents must interact with each other; and
- There is no central authority that controls the agents.

The last point is important for the cooperation among agents. Cooperation is often presented as one of the key concepts which differentiates MAS from other systems (Doran, Franklin, Jennings, & Norman, 1997). Through cooperation agents are able to obtain the necessary information needed for their tasks. Of course, interactions are the key issue for the cooperation.

It is obvious that there is an increasing need for the definition of trustworthy entities. In an open environment like MAS, agents change their intentions, goals and behaviors continually thus rendering imperative the need to define methods based on which each agent can be enabled to recognize nontrustworthy entities. The most important thing in such cases is to find ways to acquire information related to others' behavior. For example, an agent must communicate with the candidate partner or with others, in order to infer its trustworthiness. We describe methods to achieve this goal, and we give their basic characteristics.

Reputation in MAS

Reputation is an important factor in many research fields. Especially in computer science, reputation mechanisms are used either in research efforts or in commercial applications. In MAS, agents have to interact with others in order to fulfil their owners' needs for information. In such cases, reputation plays an important role.

According to a dictionary "**reputation is the state for a person of being held in high esteem and honour.**" From a social point of view "**reputation is the general estimation that the public has for a person**" (Wordnet, http://wordnet.princeton.edu).

In MAS, reputation refers to a perception that an agent has of the intentions and norms of another (Mui, Halberstadt, & Mohtashemi, 2002). This is critical for the cooperation among autonomous components in open environments, where the knowledge about the plans of others is limited.

One can find a categorisation of reputation in Wang and Vassileva (2003). Authors distinguish reputation models as centralised or decentralised, according to who has the responsibility to derive a reputation value. It should be noted that authors consider that trust is elicited through reputation. Therefore, their categorisation concerns both reputation and trust models.

- **Centralised.** In centralised reputation and trust models, the system is responsible to collect ratings for agents and publish them. Through this procedure, all ratings are evident to all members of the community and there is little need for communication among agents. Also, an aggregation procedure is performed by the system. The aggregation procedure aims to combine the different opinions in a final reputation level. Centralised models are characterised by simplicity and are mainly encountered in the area of e-commerce, where the main transactions are between sellers and buyers.
- **Decentralized or Distributed.** In decentralised systems there is not a central responsible authority and for this reason each agent develops its own reputation level for other community members. This means that there is an increased need for interactions between agents. Through them, agents form a subjective trust in their potential partners.

Mui et al. (2002) discern reputation based on which experiences and ratings are taken into consideration, and through what procedure information for the opponents is extracted. According to authors, reputation models can be divided into the following:

- **Individual Reputation.** Individual reputation is the description of the reputation level of a simple entity by another. This level is computed based on actions and information related to an agent and not a group of agents.
- **Group Reputation.** Group reputation depicts the social dimension of reputation. In these models, reputation is a function of the aggregated ratings taken from a group of entities. Entities rate others having their own experiences. These ratings may be utilised to provide information to an agent, when it needs to cooperate with others.
- **Direct Reputation.** Direct reputation is based on straight experiences with an entity. Usually, these observations are taken by interactions held between two entities. Direct reputation may be *observed* or *encounter-derived*. We have observed reputation when feedbacks through direct experiences of others consists a reference of the reputation of an agent. On the other hand, entities' ratings, after an interaction with others, may affect the reputation level of an agent. In this case, we have encounter-derived reputation.
- **Indirect Reputation.** With the lack of direct experiences for an entity, reputation can be derived from information gathered indirectly. There are three basic models for indirect reputation. The first model uses prior beliefs that agents carry about their interactions with strangers while the second model takes into consideration the group that an agent belongs to. Finally, the third model uses the information taken about an agent from the entities in the environment.

Trust in MAS

Trust is a common theme in computer science research, and refers to a range of different issues. It has important impact on domains such as security, e-commerce and Semantic Web. Trust is also an important concept for MAS. While in general trust refers to an aspect of the relationship of individuals, the concept has a completely different meaning depending on the context is used (Deriaz, 2006). Hence, trust has different meaning when we use it to characterize that humans' actions are trusted or when an agent decides to rely on another in order to obtain some resources.

Trust can be seen as the extent to which one entity intends to depend on somebody in a given situation (McKnight & Chervany, 1996). Trust can be defined as the belief that one can rely on someone else to accomplish a task. There is, however, a possibility that unfavourable issues can arise from interactions with a trusted person.

In this point, we describe a list of trust categories found in the literature.

According to Ramchourn, Huynh, and Jennings (2004) trust may be categorised, based on the part which decides the grade of trust, into the following:

- **Individual Level Trust:** Each agent decides which entity can be trusted based on its beliefs. These beliefs derive from interactions held between agents. Individual level trust can be further divided into:
  a. **Learning based.** Agents may interact with each other many times before deciding to trust someone. From this procedure, useful conclusions can be derived for the potential partners. Through repeated games, agents are able to analyze their opponents' moves in order to reach a conclusion. There are different kinds of metrics used in such models, as bi-stable values (good or bad) or fuzzy mechanisms with which one can decide to trust someone else for various acts.
  b. **Reputation based.** Reputation-based models use ratings from the members

of a community in order to derive a trust level. Important issues concerning this type of trust are the collection of ratings, their aggregation and the diffusion of members' opinions. First of all, an agent must collect ratings from the members of the community through the use of *referrals*. Referrals are the opinions of community members about a certain entity. After that, he must use an aggregation method in order to extract a result. In this point, there are issues that can complicate the whole procedure, as the lying witnesses or the absence of ratings. Finally, an important issue is the propagation of reputation in a community based on reputation scores that agents have for a set of other entities.

c. **Socio-cognitive based.** Contrary to previous types, where trust is computed taking into consideration the results and the components of interactions between agents, socio-cognitive-based trust is computed based on beliefs that an agent has for their opponents. These beliefs are: competence, willingness, persistence, and motivation belief.

- **System Level Trust:** Agents are selfish components which want to obtain as much profit as possible. For this reason, it is imperative to force them to follow some rules when interacting in the context of a system. This is the system level trust. In consequence, they will be trustworthy, thus minimizing the danger to interact with liars. System level trust can be further divided into:

a. **Truth-eliciting protocols.** These protocols may be used to elicit trustworthy behaviour of an agent. Agents must conform to certain protocols' steps in order to complete transactions in the system.

b. **Reputation mechanisms development.** Reputation may be used in system level trust. There are rules posed by the system concerning the three key elements of reputation models (collection of ratings, aggregation and propagation). In such systems, the entities responsible to store ratings may be centralised or decentralised. All agents working in the system have access to these entities either to read ratings or to publish their own.

c. **Security mechanisms development.** In these model types, a number of features are taken into consideration in order to provide a reliable security mechanism that ensures trust in system entities. The essential elements that make an agent trustworthy can be identity proof, access permissions, content integrity and content privacy (Poslad, Calisti, & Charlton, 2002). Additionally, certificates may be used to provide a higher level of security. The system forces members to give the necessary information as for the aforementioned elements in order to have an acceptable degree of safety.

Artz and Gil (2006) note that there are two methods to derive trust:

- **Based on credentials.** Credentials are elements that can be used to elicit information for an entity. A credential may be simple as a signature or complex relationships between elements in an open environment as the Semantic Web. For example, an agent may have an identifier with which may interact with others. This identifier may be used in a system to provide to an agent permissions or rights to work with specific information sources. An extensive review of systems that use credentials-based trust models is presented in Artz and Gil (2006).

- **Based on Reputation.** This model uses reputation to assign trust to members of a community. An agent utilizes personal experiences taken from interactions held between potential partners, and ratings from other members of the system. There are two ways to extract the trust level for an entity. One can rely on a central authority to have access in reputation ratings or on himself. Few efforts in literature use the first method. The second one describes the decentralised model where each entity must develop methods for the aggregation of ratings taken from the community.

In Osman (2006), authors specify the difference between trusting an agent, and trusting an interaction. They provide a categorization and a specific connection with the categories presented in Ramchourn et al. (2004). Two new categories are presented:

- **Local Deontic Level.** It concerns the constraints and permissions that an agent must follow when interacting in a multi-agent system. Agents are dynamic components and their goals, intentions and plans change continually. This means that when they work in an open environment, they have obligations, permissions and prohibitions, posed by the system. Through this, the system defines a security level concerning the transactions held among the potential partners.
- **Global Interaction Level.** Apart from the internal deontic model of each agent, there is another interaction model that specifies the rules based on which interactions are held. Every agent must conform to these rules in order to gain access to interactions with others, useful for the completion of its goals. Specifically, the interaction model is a protocol that determines steps to carry out interactions.

Grandison and Sloman (2000) presented a set of trust classes which are:

- **Provision Trust**. It describes the trust that an entity may have to a service provider.
- **Access Trust.** It describes the trust that an entity may have for the purposes of accessing resources.
- **Delegation Trust.** It describes the trust that an entity may have to an agent that works on its behalf.
- **Identity Trust.** It describes the belief that an identity is as claimed.
- **Context Trust.** It describes that the relying entity has confidence in a system in which transactions are held. Moreover, each entity can rely on system, when problems may arise in transactions.

Finally, another categorisation found in Wang and Vassileva (2003) has already been presented, where trust quantification may be held either by a central authority or by each agent individually.

A significant amount of work on trust has been performed in the area of Normative MAS (NMAS). NMAS is an extension of classical MAS which combines traditional MAS with normative systems where concepts such as obligations, commitments, permissions and rights are used to describe the behaviour of an entity (Boella, Van Der Torre, & Verhagen, 2006). Thus, every agent acting in a community has some obligations and commitments that should be fulfilled. In such systems, norms are defined to describe when the behaviour of an agent is acceptable. While the agent follows these norms its trust level increases in the community. In reverse, the agent's trust level decreases when its actions do not conform to the specified rules of normal behaviour. These rules aim to force agents to do the right thing cooperating with others in the broader environment. However, norms can be violated for various reasons and thus there is a dynamic trust valuation (Boella & Van Der Torre, 2005).

ISSUES CONCERNING REPUTATION

Reputation level can be derived based on three elements: the experiences of the evaluator, the referrals of others and the combination of the experiences and the referrals (Josang et al., 2006). There are methods that deal with all these three issues. Generally speaking, in MAS a set of agents A={a_1, a_2, …, a_n} want to interact with others in order to complete their goals. For each potential partner, every agent must calculate its reputation degree which is extracted through a reputation function:

$$\text{Reputation\_value}=f(R, E, S) \qquad (1)$$

where R represents ratings from other members of the community, E are the individual experiences taken from direct interactions with the target entities, and S represents ratings retrieved from the system.

The factor R is computed based on aggregation of ratings. Namely, there is an aggregation function which derives a final value from a set of witness agents WA={wa_1, wa_2, …, wa_m}.

$$R=g(r_1, r_2, \dots, r_m) \qquad (2)$$

where r_i denotes the referrals of the i_{th} witness agent. In literature, there are models that use only one of the above mentioned elements or a combination of them. As discussed below, every reputation model uses a function in order to calculate the final result, which follows the general form depicted in (1). For example, an agent may be based only on direct experiences with the target agent, without paying attention to the ratings of others. In order to define efficiently final reputation value, a model must be based on a combination of the above mentioned features (e.g., referrals, the system's ratings and direct interactions). The majority of systems in the literature follow this direction. Their difference is located to the form of the referred functions and the type of the computed values. Hence, in all models we can find a reputation function that produces a value that can be either discrete (e.g., "Confident," "Non-Confident") or continuous (represented through a real number).

A short description of reputation models follows. Furthermore, we present our point of view related to their advantages and disadvantages.

Simple Mathematical Models

They are the simplest models. In these models simple calculations are used in order to compute reputation values. For example, the system may store the number of positive and negative opinions for agents and compute the final score. If the positive opinions are P={p_1, p_2, …, p_k} and the negative are N={n_1, n_2, …, n_m} then the final score is:

$$Score = |P| - |N| \qquad (3)$$

where |x| denotes the cardinality of set x. The higher the value of *Score* is the more reliable the agent is considered. For example, if an agent has received 10 positive and 2 negative referrals, it has a reputation degree of 8. This agent is more reliable than another that has a reputation degree of 5. However, these models do not take into consideration the initial numbers from which the final result is computed. Let us examine an agent that may have received 100 positive and 90 negatives opinions. This means that the agent has approximately 47% negative opinions in the community. Nevertheless, this agent is more reliable than another with 10 positives and 1 negative referral (approximately 9% negative opinions).

In order to cover these disadvantages, advanced mechanisms use a weighted sum to compute an average which shows the reputation level. These mechanisms use the information related to the ratings such as the age of each rating, the distance between rating and current reputation value, and so forth (Josang, Roslam, & Colin, 2006).

It should be noted that such systems do not take into consideration critical issues concerning the selfish and dynamic nature of agents. It is possible that agents may form coalitions in order to exchange positive marks thus achieving better reputation scores. Furthermore, the simple mathematical models do not examine in depth the referrals retrieved from the community members in order to extract useful information about the behaviour of an agent.

Bayesian Reputation Systems

Bayesian systems are based on statistics. They compute reputation using the Beta probability density function. This function can be used to describe probability distributions of binary events. For simplicity, we give only a short description of such systems.

A Bayesian reputation system takes binary ratings and uses the *a priori* reputation score and the current ratings to compute the *a posteriori* result (Josang & Ismail, 2002; Mui, Mohtashemi, Ang, Szolovtis, & Halberstadt, 2001; Whitby, Josang, & Indulska, 2004). In agent systems, we can describe the behaviour of an agent as "Honest" vs "Dishonest," or as "Reliable" vs "Unreliable" which constitute binary events. If, for one agent, there are x positive and y negative observations, then the reputation score can be computed as follows:

$$\alpha=x+1,\ \beta=y+1 \text{ with } x,y \geq 0 \qquad (4)$$

the probability expectation value is:

$$E(p) = \frac{\alpha}{\alpha + \beta} \qquad (5)$$

for the Beta distribution, which can be expressed as:

$$B(p|\alpha,\beta) = \frac{\Gamma(\alpha + \beta)}{\Gamma(\alpha)\cdot\Gamma(\beta)} p^{\alpha-1} \cdot (1-p)^{\beta-1} \qquad (6)$$

where $p \in [0..1]$, $\alpha,\beta>0$, $p \neq 1$ when $b<1$ and $p \neq 0$ when $a<1$.

When the a priori probability does not exist, then we consider α=1 and β=1. From this function, each agent can compute the possibility that a potential partner is reliable based on previous values of reputation. For example, if the expectation value E(p) has a score of 0.9 means that the most likely value of positive outcomes in the future is 0.9, but the actual outcomes are uncertain.

Bayesian models give a computational theoretical framework for the reputation score good for autonomous computational entities as agents are. It is an efficient mechanism to combine evidences. An entity must keep track of the outcomes of others and compute the reliability possibility through the above referred functions. A full description of a system representative of this kind of model can be found in Josang and Ismail (2002). However, these models are complicated due to the calculations that must be performed in order to derive a final reputation value. Also, the definition of a priori probability used for the calculations is necessary. The probability value is important for these models and must be derived by a subjective method.

Social Networks

Social networks are originated in sociology. Social networks can be represented as graphs that depict relations between members of a community. Social networks analysis emerged as a set of methods for the analysis of social structures (Sabater & Sierra, 2002). In MAS, agents must retrieve data concerning the relation among the members of the system in order to decide the reputation level of a potential partner. However, it is difficult to use methods taken from sociology in order to poll information for the network architecture. For example, sociologists use methods as the opinion poll or interviews. In cases where autonomous computational components are the nodes of a social network, more "computational" ways must be found to extract the necessary information.

The procedure that agents adopt for building the network is critical for the success of such systems. They must describe as much relational data as they can in order to have a significant view of the system. We must also note, that these networks change dynamically due to the open nature of MAS. Agents may enter or leave at every time and, moreover, may alter their goals, behaviours and intentions.

Generally speaking, in systems based on social networks there is a set A={a_1, a_2, a_3, …, a_n} of agents that want to obtain information from others in order to complete their goals. Each agent builds a social network G={d_1, d_2, …, d_n}, with nodes d_i representing the members of the system. Edges that connect nodes show a relation between them. For example, edge $e_{i,j}$ denotes that there is a relation among nodes i and j. Next, it finds its potential partners and tries to collect information about them. It may be based on referrals or on the system or on a combination of them. Referrals are taken from other members that have interaction history with the target agents, and after a careful selection. These referrals must be aggregated in order to extract a final value through which reputation is computed.

Three critical factors must be taken into account: the possibility that agents may tell lies, the possibility that agents may conceal information about others and the careful selection of referral agents. In these cases, social networks may be constructed based on false values as for the reputation degree of each member. Also, there are cases in which agents ally with others and for this reason may hide the bad reputation that an examined agent acquired. A method to alleviate the problem of lying in MAS is presented in Schillo, Funk, and Rovatsos (2000).

As mentioned above, in MAS social networks agents appear as nodes and their relationships as edges. Each edge has a value which represents the weight of the relationship between the two connected agents. After the graph creation, the construction of the network and the computation of reputation follow. Such computation is based on the weights assigned to edges. An extensive survey on reputation mechanisms based on social networks can be found in Ramchourn et al. (2004).

In this point, we present two simple examples of social networks models. A first example of such model is presented in Pujol and Sanguesa (2002). Authors describe an algorithm which is named *NodeRanking* and is used to extract a reputation value for members in a community. Its main idea is that every node and, respectively, agents have an authority degree which is an importance measure. For example, the authority of a node x is computed as a function of the total measure of authority presented in the network and the authority of the nodes pointing in x. The main rationale is that if a node has a lot of edges pointing to it, this means that the node is important in the community because this means that it cooperates with many other members. Another critical issue is that every authority value is propagated through the out-edges. The reputation value is computed based on the authority value of each node, taking into consideration the importance and hence the number of agents that are related to the examined entity.

Another system that uses social information in order to compute reputation is REGRET (Sabater & Sierra, 2002). Reputation in REGRET has three dimensions, the *individual*, the *social*—according to the source of the information used to extract a reputation value—and the *ontological* dimension, which helps to transfer the reputation between related contexts. While the individual dimension takes into consideration the results of direct interactions between potential partners, the social dimension utilizes information taken from the other members of the community. In the second dimension of reputation, agents may use witnesses from others or consider neighbourhood reputation. Furthermore, the system assigns a reputation level to every role defined in it. Hence, agents that have a specific role in the system inherit the reputation level assigned to the role.

System reputation is the easiest to compute but is dangerous because a role held by an agent does not convey information about its intentions. In the REGRET system, reputation is combined with a domain and calculated using a table in which rows are the potential roles and columns are the reputation types. More complicated are the remaining two methods. Witnesses reputation uses the referrals of others in order to establish a reputation level. REGRET gives the opportunity to an agent to define a set of witnesses and aggregate their referrals based on fuzzy rules. Of course, witnesses are entities that have interacted in the past with the target agent and they are taken into account based on the same event, if it is possible. Neighbourhood reputation is not related to the physical location of the agents but to the links created by interactions. These interactions and the relations between agents are very useful to compute reputation level for a target agent. Fuzzy rules are also used in this case.

Belief Theory Models

Belief theory characterizes the remaining of the subtraction between 1 and the summary of the possibilities of the all possible outcomes, as uncertainty. In these models, agents use their beliefs about the behaviour of another entity. In Yu and Singh (2002), authors propose the use of Dempster-Shafer theory (Dempster, 1968; Shafer, 1976) for the computation of reputation degree. There are two kinds of belief in their model: *Local* and *Total*. Local belief is obtained from direct interactions with the target agent and Total belief is extracted from the opinions of others combined with local belief. Witnesses from others are necessary when interactions are not available. Each agent models the information from others using belief functions. There are two outcomes related to the reliability of an agent: *Trustworthy* or *Not Trustworthy,* each of them has a belief value m(T) and m($\neg$ T) respectively, taken from the correspondent belief function. The reputation score for an agent A is:

$$\Gamma(A)=\beta_A(\{T_A\})-\beta_A(\{\neg T_A\}) \qquad (7)$$

where β_A is the cumulative belief result computed using the testimonies from a set of L neighborhoods. When no testimonies are available, then the reputation score is 0. Also, authors present the reputation value of a set of K agents, which is:

$$\text{Group\_Reputation} = \frac{1}{K}\sum_{i\in[1..K]}\Gamma(A_i) \qquad (8)$$

These models are able to exhibit the beliefs of agents, accumulated from past experiences or others, in functions that combine them and produce the final result. However, the definition of a threshold value is critical. This threshold defines when an agent may be characterized as trustworthy or not. Moreover, the assumption of only two possible outcomes limits the model.

Fuzzy Models

Fuzzy models try to catch the subjective point of view of an agent related to another member of a community. In these models, reputation is presented through linguistic fuzzy values in contrast to other models in which common reputation levels are defined by means of real numbers. For example an agent may be characterised as "reliable" or "not reliable." Fuzzy logic (Zadeh, 1989) is very important because it provides reasoning techniques for the extraction procedure. Rules may have the next form:

IF andecent THEN consequent

where *andecent* is represented with fuzzy sets.

Fuzzy logic techniques depend on subjective criteria that may lead an agent to cooperate based on its thoughts about others. This means that if an agent has very optimistic views of the community he may rely on others that have bad intentions.

In Rubiera, Molina Lopez, and Muro (2001) a method for the computation of reputation based on a fuzzy model is presented. Each agent retrieves opinions only from entities that are highly appreciated. Based on their answers it computes a value that is extracted from a fuzzy set according to its point of view. The result is the weight of an agent's opinion. Furthemore, there is interest on combination of the new and the old reputation values. The old reputation score of the candidate partner is taken into consideration and, hence, the final result is the average of two fuzzy values: the old and the new one. The agent is responsible to decide on defection. Usually, if a threshold is reached, cooperation is held.

Another system that relies on fuzzy rules is REGRET, which was briefly described in the previous section.

Role-Based Reputation

In some models, reputation can be seen as a value of a role fulfilment. A role is a set of obligations and actions that an entity has in the community. If an agent acts and behaves as a role dictates, then it has the reputation level that this role offers. In Carter and Ghorbani (2004) a framework for the role fulfilment measurement is presented. Three roles are investigated: The Assistant, the Provider and the Citizen. A general overview of how measurements take place in each role is given by the authors. In order to compute the final value of reputation, authors examine the satisfaction degree of each role for a specific entity and combine these partial results. The reputation is a weighted sum of each value that reflects the fulfilment of each role.

The main problem with these models is that they do not examine in depth the intentions that agents have. Meeting the requirements of a role by an agent does not mean that the agent will not change its behaviour.

Unfair Ratings: Deception

As mentioned above in distributed reputation models, when a central authority is absent each agent that wants to cooperate with others must collect ratings from the environment in order to decide the reputation degree of an entity. In these cases important issues are:

- The possibility that some agents provide unfair ratings for others. These ratings may be unfairly positive or unfairly negative (Dellarocas, 2000).
- The possibility that an agent deludes others.

According to Whitby et al. (2004), methods of avoiding unfair ratings are divided into *endogenous* and *exogenous*.

Endogenous methods are based on the statistical analysis of the rating values. They can give or exclude ratings that are possible to be unfair. Classical examples of this kind of systems are Bayesian reputation systems. Authors describe an algorithm that filters unfair ratings in a Bayesian model. Exogenous methods are based on factors that are related to external elements such as the reputation of the witness. The main idea is that an entity with low reputation is likely to give unfair ratings and vice versa. In the relevant literature, one can find a lot of works falling in the aforementioned categories.

Another algorithm for the detection of deceptive agents is proposed in Yu and Singh (2003). This method uses exogenous characteristics of the witnesses as we presented above. The algorithm assigns weights to witnesses and makes a prediction based on the weighted sum of their ratings. The second idea is to tune these weights when a prediction fails. In this case, the weight of successful witnesses is increased and the weight for the unsuccessful is decreased. We must note that the ratings that an agent takes are belief functions and for this reason the algorithm maps belief functions to probabilities in order to be able to compute and update the weights of each witness. Moreover, authors study the number of witnesses and its effect to the system's prediction values.

ISSUES CONCERNING TRUST

Trust is usually researched in the security domain. The main reason is that these two concepts are related, but they have different orientations. However, trust and security provide protection against malicious components. In this sense, trust can be considered as a *soft security* mechanism. This term first appeared in Rasmusson and Jansson (1996). Authors discern *hard* and *soft* security mechanisms. Hard tools are authentication, cryptography, and so forth, and soft tools are those that take into consideration social control issues, as they are trust and reputation.

In MAS, trust plays an important role because agents need to cooperate with other members of the community. The importance of trust in MAS is shown in Castelfranchi and Falcone (1998). Critical questions arise such as: When can I trust another entity? Which entity is trustworthy? What are the elements that can be used in order to conclude a trust level? What methods should be used to conclude trust level? Such questions are addressed in the fourth section of our chapter.

Discussion

The main differences between trust and reputation are:

a. Usually, trust is a score that reflects the subjective view of an entity from another, whereas reputation is a score that reflects the view of the community.
b. In trust systems, transitivity is considered explicitly while in reputation systems is seen implicitly (Wang, Hori, & Sakurai, 2006).

The common element between the two concepts is that both of them try to help someone that wants to find trustworthy partners to achieve its goals through cooperation. However, trust is more complicated concept that involves many parameters. For this reason, it is very difficult to assign a strict definition to trust.

As mentioned above, trust is a subjective view of an entity. It is based on some beliefs that an entity has for another, but it is not clear where such beliefs originated. This means that an agent may be reliable only for a set of other agents and not for all of them. The level of trust is also depended on the context in which it is being studied. For example, an agent may be trustworthy when providing information but it is nontrustworthy when selling products. These two factors are basic to open systems and must be taken into consideration. Moreover, trust is dynamic. An agent may consider another entity as reliable in a specific time but its opinion may change accordingly based on the behaviour of the target entity.

The simplest form of trust is centralized. In such systems, there is a central authority that keeps the trust level of entities who rate each other after every transaction. Soft mathematical calculations are held to provide the final result. It is a scheme that must take into account issues concerning lying entities or unfair ratings. Also, there must be a high security level to prevent violations in central database where ratings are kept. On the other hand, decentralized trust models are complex and require effort from the side of each entity that tries to find partners. In such systems, critical issues are the storage of trust values, the location of witnesses and the inference procedure.

In general, a trust function has the following parameters: the beliefs of the examiner (B), the reputation of the examinee (R), previous trust values (P) and the context (C).

$$\text{Trust\_value}=f(B, R, P, C) \qquad (9)$$

An interesting value is B. B may be extracted from direct experiences through communication or past experiences with the target entity. It may be a positive or a negative belief. Relation (9) concerns a general function form. As we discuss in the following paragraphs all the described models use a function that follows this general form and takes into consideration one or more values from B, R, P or C.

The result of the referred function may be discrete or continuous values. For example, in discrete models, as fuzzy models are, a trustworthy behaviour may be characterized as "Very Trustworthy," "Trustworthy," "Untrustworthy," or "Very untrustworthy," for direct trust between two entities, or "Very good," "Good," "Bad," or "Very bad" for recommender trust (Abdul-Rahman & Hailes, 2000). Either discrete or continuous, the final trust value reflects a confidence over the knowledge we have about an entity.

Trust mechanisms vary from these that use simple computations to those that use more complicated characteristics of the entities involved in such situations. However, the common procedure among them is that they map a set of features to trust information. In the following paragraphs, we give a description of some important categories of trust and examples of each one.

A key issue concerning trust is its dynamic nature. Trust evolves over time as entities cooperate with others. For this reason, it is critical to define a trust update procedure. Especially in open environments like MAS, where goals, intentions and beliefs of each agent change continually, there is a need for dynamic adaptation of the trust level. This means that in every model that is used to compute trust, developers must take into consideration how trust levels evolve over time and transactions. The evolution of trust may be based on the experiences of the trustor or on new information taken from other members of the community.

In conclusion, in the computing trust procedure, the phases that an agent may handle are:

a. Trust Discovery Phase (TDP);
b. Trust Aggregation Phase (TAP); and
c. Trust Evolution Phase (TEP).

In TDP, each agent tries to find the appropriate sources for referrals and may communicate with the target agents in order to elicit useful information about their behaviour. In this phase it is important to have mechanisms to identify if a group of agents have formed a coalition and share good referrals among them. In TAP, the most important issue is to use an appropriate aggregation function in order to derive the final value of trust. This function may take into consideration the results of direct experiences and of course the referrals of peers. Finally, the TEP is a continuous procedure through which an initial trust level is evolving over time based on observations. Its great importance relies on the dynamic nature of MAS. If an agent is trusted in a specific time this does not mean it is to be trusted forever. Agents are selfish and may change their behaviour without warning or may ally with others.

Trust Propagation

MAS can be viewed as graphs where their agents are represented by nodes. Edges consist of their relationship in the community and weights represent the trust value between the two connected nodes. Graphs may be used to transfer trust information among members. Every agent trusts some others in the network. *Estimated trust belief* is derived through the trust network based on inferences while *expected trust belief* is the ideal target (Ding, Kolari, Ganjugante, Finin, & Joshi, 2004). It is very difficult to achieve the expected trust belief due to the lack of global knowledge of the community and its members. As the trust network evolves over time and more information is gathered from the agents, the ultimate goal is gradually approached.

Propagation of trust is very important in such networks because it gives the opportunity to derive beliefs about agents through the combination of values taken from multiple sources. The most common method for trust propagation is referrals. Referrals have been investigated in reputation mechanisms (see Section "Issues Concerning Reputation"). An agent, having collected opinions from a set of peers, needs an aggregation method in order to define the final estimated value of trust. Through this procedure, trust information can be propagated over the network. Additionally, we must take into consideration that agents may ally with others and give positive recommendations for their allies. However, when an agent establishes trust based on recommendations from others, this trust value cannot be greater than the trust value between the agent and the recommender, and neither the trust value between the recommender and the target agent (Lindsay, Yu, Han, & Ray Liu, 2006). Another effort on trust propagation is discussed in Guha, Kumar, Raghavan, and Tomkins (2004).

Simple Trust Models

Simple trust models try to determine relations that depict the behaviour of an entity as a function of positive and negative opinions. The simplest form, found in Deriaz (2006), is:

$$\text{Trust\_score} = \frac{|\,Positive\ Ratings\,|}{|\,Positive\ Ratings\,| + |\,Negative\ Ratings\,|} \tag{10}$$

where |Positive Ratings| and |Negative Ratings| represent the number of positive and negative ratings, respectively.

Whenever an certain agent has only positive ratings and no negatives, then the trust score is equal to 1. Therefore, *Trust_score* can take values ranging from 0 to 1. An extension to this model that take into consideration the time in which these ratings are provided gives more efficient trust computation because it can exclude obsolete ratings. Furthermore, trust can be computed if we scale recent events. Accordingly, if in Deriaz's model we set a=b=c=1, which are the default values of parameters a, b, c, then the following holds:

$$Trust\_score = \frac{|\,PosRatings\,|\cdot(1+\delta_p)+A}{|\,PosRatings\,|\cdot(1+\delta_p)+|\,NegRatings\,|\cdot(1+\delta_n)+B} \tag{11}$$

where

$$A = \frac{1}{T}\sum_{i\in[1..|PosRatings|]} t(po_i) \tag{12}$$

$$B = \frac{1}{T}\Big(\sum_{i\in[1..|PosRatings|]} t(po_i) + \sum_{i\in[1..|NegRatings|]} t(pn_i)\Big) \tag{13}$$

and δ_p, δ_n are the statistical variance of the positive and negative outcomes, T is the current time, po_i, pn_i are positive and negative rating received at time i, |PosRatings| and |NegRatings| are the cardinality of positive and negative ratings, respectively.

The disadvantage of this mechanism is that it does not take into account the context in which these ratings were made. The above forms deal with the positive and the negative opinions without separating them into the context fields for which ratings are formulated. Furthermore, the model does not notice cases where entities may ally with others in order to elicit positive outcomes or the case that an entity is neutral to another.

Entropy-Based & Probability-Based Trust Model

In this section, we present two models of trust based on the work reported in Lindsay et al. (2006). Authors, influenced by information theory, present *Entropy-based* and *probability-based* trust. The trust relationship between two entities is represented by T(S,A,AC), where S is the subject which examines the trust level of the agent A for an action AC. Similarly, the probability that and agent A will perform the action AC in the subject's S point of view is represented by P(S,A,AC). The entropy based value of trust is calculated as follows:

$$T(S,A,AC) = \tag{14}$$

$$\begin{cases} 1 + p \cdot \log_2(p) + (1-p) \cdot \log_2(1-p) & 0.5 \le p \le 1 \\ -p \cdot \log_2(p) - (1-p) \cdot \log_2(1-p) + 1 & 0 \le p < 0.5 \end{cases}$$

where

$$p = P(S,A,AC) \tag{15}$$

The final trust value is a real number in the interval [-1,1]. Some important examples that show the trust level of an agent are:

$$C = \begin{cases} \textit{Subject trusts the agent} \quad p = 1 \;\; and \;\; T = 1 \\ \textit{Subject distrusts the agent the most} \quad p = 0 \;\; and \;\; T = -1 \\ \textit{Subject has no trust} \quad p = 0.5 \;\; and \;\; T = 0 \end{cases} \tag{16}$$

In general, the following holds:

$$T(S,A,AC) = \begin{cases} < 0 & when \;\; p \in [0..0,5) \\ > 0 & when \;\; p \in (0,5..1] \\ = 0 & when \;\; p = 0,5 \end{cases} \tag{17}$$

We should note that the probability P(S,A,AC) represents the view of a specific subject which means that different agents have different opinions about a target agent.

The entropy-based model depends on the trust value described above. Especially for the propagation of trust, a simple product is used where the two factors are the recommendation value of another agent multiplied by the trust value of the recommender. For multipath recommendations, the final result is the weighted sum of each recommendation. In the probability-based model, the probability that an agent will perform the specific tasks combined with the probability that a recommender make correct recommendations is adjusted.

Reputation-Based Trust Models

Reputation based trust models are used in distributed systems where there is little information on the overall network. If an entity has a high reputation level in a community then others may trust it more easily than another that has lower reputation value. For the computation of trust, an agent depends on opinions of a set of community members. Important issues in these cases are the collection method of ratings and the aggregation procedure. It should be reminded that trust is a concept derived from direct interactions between two entities, while reputation is the view of a member from the community side.

Reputation-based models rely on methods that give the opportunity to an entity to gather referrals from other members and apply an aggregation function in order to calculate the final value of trust. It is wiser to combine these results with values obtained from direct experiences. In Ramchourn et al. (2004), authors give an extensive review describing methods for the retrieval of ratings and the aggregation procedure from a social network point of view.

Reputation models are discussed in previous section.

Bayesian Network Trust Models

A Bayesian network is a network where probability relationships of some entities are presented (Ben-Gal, 2007). The final trust value is represented by the root of the network and leafs are the sources in which the beliefs of the examiner are based. A formalization of Bayesian trust networks is given in Melaye and Demazeau (2005). Each agent forms its basic beliefs investigating a number of belief sources. We have:

$$Basic\_Belief_i = f(B_{i1}, B_{i2}, \ldots, B_{iN}) \tag{18}$$

where $i \in [1..number\_of\_basic\_beliefs]$ and N are the number of belief sources. An important point is that the function f is depended on conditional probabilities, which means that a tree node may be more influential than another. Each trust component is associated with a probability of satisfaction. Accordingly, the final trust value is calculated as follows:

$$\text{Trust\_value}=g(BB_1, BB_2, \ldots, BB_N) \qquad (19)$$

where BB_i is the i-th basic belief taken from (18). It is obvious that in such cases a bottom-up approach is adopted, starting from the belief sources and concluding with the final result.

Another representative example is shown in Wang and Vassileva (2003). Each agent builds a Bayesian network for every potential partner. Each network has a root with two values. The first represents the satisfaction level while the second the nonsatisfaction degree. The satisfaction level is derived from the number of the successful interactions divided by the total number of interactions. The leaf nodes in the network represent the different capabilities that potential partners have. In this model, the recommendations of other agents are taken into consideration.

Belief and Fuzzy Models

In belief models, we meet methods that use the beliefs of an agent that another entity is trustworthy or not. In belief theory, each opinion may be represented as a triplet (belief, disbelief, uncertainty). The sum of probabilities of these three values is 1:

$$\text{belief}+\text{disbelief}+\text{uncertainty}=1 \qquad (20)$$

Agents use their and others' beliefs in order to extract the final score. This score is computed through the use of belief theory and consists of a subjective certainty of the pertinent beliefs. An example of a belief trust model one can be found in Josang (1999, 2001). Josang names his trust model "subjective logic" and combines belief theory with Bayesian probabilities. The forms used for this purpose are:

$$\text{belief} = \frac{p}{p+n+2} \qquad (21)$$

$$\text{disbelief} = \frac{n}{p+n+2} \qquad (22)$$

$$\text{uncertainty} = \frac{2}{p+n+2} \qquad (23)$$

where p and n are the positive and negative ratings, respectively. These two parameters are also used in the beta probability density function (see Section titled "Bayesian Reputation Systems"). For the combination of beliefs, external or internal, an aggregation function is used. "Majority consensus" functions are well-known for handling beliefs with discrete values while numerical functions are more appropriate for handing beliefs with continuous values (Ding et al., 2004). The authors in Josang (1999, 2001) use operators for the combination of opinions that are not based on Dempster-Shaffer theory as Yu and Singh do (2002).

In fuzzy models trust and reputation are described with linguistic fuzzy values. Reasoning is used in order to achieve the definition of a trust level. The most important and completed example is the system REGRET, which is described in the "Social Networks" section.

Role-Based Trust

These models are based on the notion of role and its assigned permissions to operate in a system. Hence, credentials are used to define access to a system such as identity, authentication, and so forth. Thus, an entity can be uniquely identified by the system and can obtain a specific role. Roles are used to give information about an entity. The key is the trust management mechanism that employs different languages and engines for reasoning on rules for trust establishment. It is a model used in access control systems and tries to determine the trust level of an entity based on credentials and security policies. A framework based on roles is presented in Li and Mitchell (2003).

REPUTATION AND TRUST ENGINEERING

Modeling trust requirements is the most important issue in developing efficient systems. Especially in cases where open systems are examined (e.g., MAS), this feature receives more attention. This section of our work aims to show the basic elements in which a requirements engineering procedure must be based.

As mentioned, MAS are open systems with members that change their behaviours continually. They are characterized by openness, heterogeneity, and dynamic character. Selfish agents try to locate partners in order to achieve their goals. Main issues that must be taken into consideration in MAS are:

- **Agents' identity.** Each autonomous component that interacts in a system should have a unique name and should be able to prove its identity. Identity is a requirement when agents communicate with others, because it shows to the potential partner that the component is a registered user of the system. Of course, a critical issue is the administration of the names. For example, an agent having bad reputation in a community may change its name in order to avoid the consequences.
- **Agents' communication.** Agents' communication is also important. Developers of MAS should introduce standards for communication. A security policy is necessary in order to avoid problems in the exchange of messages. Corrupted messages should be recognised by the recipients. Mechanisms that can be use for safe communication are authentication, cryptography, and so forth.
- **Agents' context.** The context in which each agent is activated should be defined. Mechanisms that take the context into consideration should be developed. This could provide efficiency in the cooperation procedure.
- **Agents' behaviour.** Agents' behaviour should be observed by the interested members and from the system. It is imperative to have the opportunity to observe and recognize bad or good behaviours in the system. Based on behaviour, trust is developed and members obtain a high reputation value in the community. Constructive trust must be promoted through the observation of interactions of an agent in the society. Also, from the systems' point of view, mechanisms that "punish" bad behaviours should be developed.

In Wong and Sycara (1999), the authors present a number of possible threats in MAS and their potential solutions. The basic threats that MAS may meet are related to corrupted naming of agents, insecure communication channels, insecure delegation and lack of accountability. Synoptically, the proposed solutions are:

- The use of trusted agent name servers and matchmakers.
- The provision of methods for the unique identification of each agent and proofs for their identifications.
- Secure the communication channels through the authentication of messages.
- Force agents to prove their owners.
- The provision of methods through which owners of the agents may be liable for their actions.

Authors give a full description of these solutions and explain the mechanisms with which these goals will be achievable.

A methodology for agent-based software development is described in Giorgini, Massacci, and Zannone (2005) and Giorgini, Mouratidis, and Zannone (2007). In TROPOS there are phases through which the trust establishment is feasible. The first phase is the requirement phase. In this stage, the functional and the nonfunctional requirements are determined in two subphases:

the *early requirements phase* and the *late requirements phase*. The key concepts in secure TROPOS are:

- **Actor.** It is an entity that represents a physical or software agent as well as a role or position, having its goals and intentions.
- **Goal.** Represents actors' interests that they wish to accomplish.
- **Plan.** It concerns a number of steps targeting to achieve a goal.
- **Resource.** It is a physical or an information entity.
- **Dependency.** Indicates that an actor depends on some other entity in order to complete their goals.

Accordingly, in TROPOS four new relationships are defined, which are: **Ownership** which indicates that an actor has a goal, **provisioning** which is the capability of an actor to achieve a goal or to have a plan or to provide some information, **trust** which indicates the belief of an actor that another entity will perform some task according to their goals and plans and **delegation** which shows that an actor delegates to some other to achieve its goals.

In TROPOS methodology, basic operations are:

- The definition of the actor's model and the dependency model. The essential actors should be recognised as well as the dependencies among them from the point of view of achieving their goals.
- The trust and delegation models. They determine the relationships between actors.
- The goal and plan models. Such models identify, from the viewpoint of actors, goals and plans that are necessary to achieve (sub)goals. Moreover, the resources needed to this procedure are recognised.

An extension to classic TROPOS is the security constraint modeling, which involves security constraints posed by the actors and the system. The architectural design development process for this extension relies on the following:

1. Secure Architectural style model.
2. Actor model, Goal/plan model, and security constraint model.
3. Capability and secure capability model.
4. Agent model.

In this chapter, due to the space constraints, we have presented only a short description of the TROPOS methodology. For a full discussion the interested reader should refer to the Giorgini et al. (2005, 2007).

FUTURE DIRECTIONS

Reputation and trust are important concepts in today's dynamic systems. However, there are some open issues that must be addressed in order to develop efficient methods for adoption in MAS.

First of all, for the construction of a theoretical framework that covers all the aspects of reputation and trust generation, manipulation and propagation is necessary. This model will set basic specifications in which developers may be based on, in order to construct efficient and productive systems. Such a framework will set the essentials that will allow the comparison of existing models. Today, this comparison is very difficult to accomplish, because the existing models come from specific domains and they are not based on a common theoretical framework.

The social network dimension in MAS presents new opportunities in reputation and trust management. However, it must be validated in real applications in order to discover its advantages and disadvantages. In social networks, there is an extensive need to deal with problems related to strategic lying and strategic coalition formation. This domain must be further studied in order to produce effective methods to deal with. Moreover, there is a need to define basic mechanisms through which opinions of members can be stored and secured in order to provide a higher security level.

Interactions of agents are held in a system under specific context. Context must be taken into consideration when defining the trust and reputation level of an entity. To the best of our knowledge, only a few works deal with this issue. Additionally, concern should be posed in trust propagation in specific contexts. Propagation allows the building of relations between all the agents communicating in a system. New methods for propagation should be developed with regard to the combination of the aforementioned models such as statistical functions, belief and fuzzy theory.

CONCLUSION

This chapter introduces the reader to the domain of reputation and trust in Multi-agent systems. It presents the existing reputation and trust quantification methods that are used in commercial and research applications. We show the importance of these two concepts especially in open systems where control over the actions of agents is limited. Also, the environment, where agents act, may change at any time due to the nature of the involved entities, which are autonomous components trying to serve their owners. For this reason, they are selfish and change their behavior subject to new conditions. A lot of models have been proposed for reputation and trust extraction in specific domains. We shortly present the most important of them, giving their basic characteristics. Certain models are very simple, while others are more sophisticated and utilize statistical functions, belief or fuzzy theory. Finally, we discuss key contribution in the domain of trust and reputation engineering. It is a critical field that leads to more efficient and productive systems. The engineering process must drive developers to construct systems that pay special attention to issues that ensure secure communication and fair ratings for all.

Reputation and trust will play an important role in future systems. Such concepts will be extensively adopted and used for gaining access to information sources in open environments like MAS and the Semantic Web.

REFERENCES

Abdul-Rahman, A., & Hailes, S. (2000). Supporting trust in virtual communities. In *Proceedings of the Hawaii International Conference on System Services* (Vol. 6, pp. 6007).

Artz, D., & Gil, Y. (2006). *Survey of trust in computer science and the Semantic Web.* Submitted for publication, Information Sciences Institute, University of Southern California. Retrieved April 3, 2008, from http://www.isi.edu/~dono/pdf/artz06survey.pdf

Ben-Gal, I. (2007). Bayesian networks. In F. Ruggeri, F. Faltin, & R. Kenett (Eds.), *Encyclopedia of statistics in quality and reliability.* John Wiley & Sons.

Boella, G., & Van Der Torre, L. (2005). Normative multiagent systems and trust dynamics. *Trusting agents for trusting electronic societies, LNAI 3577,* (pp. 1-17).

Boella, G., Van Der Torret, L., & Verhagen, H. (2006). Introduction to normative multiagent systems. *Computational & Mathematical Organisation Theory, 12*(2-3), 71-79.

Carter, J., & Ghorbani, A. A. (2004). Value centric trust in multiagent systems. In *Proceedings of the IEEE/WIC International Conference on Web Intelligence*, (pp. 3-9).

Castelfranchi, C., & Falcone, R. (1998). Social trust: Cognitive anatomy, social importance, quantification, and dynamics. In *Proceedings of the First International Workshop on Trust,* Paris, France, (pp. 72-79).

Dellarocas, C. (2000). Immunizing online reputation reporting systems against unfair ratings and discriminatory behavior. In *Proceedings of the ACM Conference of Electronic Commerce*, (pp. 150-157).

Dempster, A. P. (1968). A generalisation of BaysianBayesian inference. *Journal of the Royal Statistical Society, Series B, 30*, 205-247.

Deriaz, M. (2006). *What is trust? My own point of view.* University of Geneva. Retrieved April 3, 2008, from http://cui.unige.ch/ASG/publications/TR2006/

Ding, L., Kolari., P., Ganjugunte, S., Finin, T., & Joshi, A. (2004). Modeling and evaluating trust network inference. In *Proceedings of the 7th International Workshop on Trust in Agent Societies at AAMAS 2004,* New York.

Doran, J. E., Franklin, S., Jennings, N. R., & Norman, T. J. (1997). On cooperation in multi-agent systems. *The Knowledge Engineering Review, 12*(3), 309-314.

Durfee, E. H., & Lesser, V. (1989). Negotiating task decomposition and allocation using partial global planning. In L. Gasser & M. Huhns (Eds.), *Distributed artificial intelligence* (Vol. 2, pp. 229-244). San Francisco: Morgan Kaufmann.

Giorgini, P., Massacci, F., & Zannone, N. (2005). Security and trust requirements engineering. *Foundations of security analysis and design III—tutorial lectures, LNCS 3655* (pp. 237-272). Springer-Verlag.

Giorgini, P., Mouratidis, H., & Zannone, N. (2007). Modelling security and trust with secure TROPOS. *Integrating security and software engineering: Advances and future vision.* Hershey, PA: Idea Group.

Grandison, T., & Sloman, M. (2000). A survey of trust in Internet applications. *IEEE Communications Surveys and Tutorials, 4th Quarter, 3*(4), 2-16.

Guha, R., Kumar, R., Raghavan, P., & Tomkins, A. (2004). Propagation of trust and distrust. *In Proceedings of the 13th International Conference on World Wide Web (WWW 2004)*, New York, (pp. 403-412).

Huynh, D., Jennings, N. R., & Shadbolt, N. R. (2006). An integrated trust and reputation model for open multi-agent systems. *Autonomous Agents and Multi-Agent Systems, 13*(2), 119-154.

Josang, A. (1999). Trust-based decision making for electronic transactions. In L. Yngstrm & T. Scensson (Eds.), *Proceedings of the 4th Nordic Workshop on Secure Computer Systems.* Stockholm, Sweden: Stockholm University Report 99-005.

Josang, A. (2001). A logic for uncertain probabilities. *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems, 9*(3), 279-311.

Josang, A., & Ismail, R. (2002). The Beta reputation system. In *Proceedings of the 15th Bled Conference on Electronic Commerce,* Slovenia, (pp. 324-337).

Josang, A., Roslam, I., & Colin, B. (2006). A survey of trust and reputation systems for online service provision. *Decision support systems.*

Li, N., & Mitchell, J. C. (2003). RT: A role-based trust-management framework. In *Proceedings of the 3rd DARPA Information Survivability Conference and Exposition (DISCEX III).*

Lindsay, Y., Yu, W., Han, Z., & Ray Liu, K. J. (2006). Information theoretic framework of trust modelling and evaluation for the ad hoc networks. *IEEE International Journal on Selected Areas in Communications, 24*(2), 305-317.

McKnight, D. H., & Chervany, N. L. (1996). The meanings of trust (Tech. Rep.). University of Misessota, Management Information Systems Research Center. Retrieved April 3, 2008, from http://www.misrc.umn.edu/workingpapers/

Melaye, D., & Demazeau, Y. (2005). Bayesian dynamic trust model. In *Proceedings of Multi-agent Systems and Applications IV: 4th International Central and Eastern European Conference on Multi-agent Systems, CEEMAS 2005*, (pp. 480-489). Springer-Verlag, LNCS 3690.

Mui, L., Halberstadt, A., & Mohtashemi, M. (2002). Notions of reputation in multi-agent systems: A review. In *Proceedings of the 1st International Joint Conference on Autonomous Agents and Multiagent Systems,* Bologna, Italy, (pp. 280-287).

Mui, L., Mohtashemi, M., Ang, C., Szolovtis, P., & Halberstadt, A. (2001). Ratings in distributed systems: A bayesian approach. In *Proceedings of the Workshop on Information Technologies and Systems (WITS)*, Miami, Fl.

Nwana, H. S. (1996). Software agents: An overview. *The Knowledge Engineering Review, 11*(3), 205-244.

Osman, N. (2006). *Formal specification and verification of trust in multi-agent systems.* School of Informatics, University of Edinburgh. Retrieved April 3, 2008, from http://homepages.inf.ed.ac.uk/s0233771/trust.pdf

Poslad, S., Calisti, M., & Charlton, P. (2002). Specifying standard security mechanisms in multi-agent systems. In *Proceedings of the Workshop on Deception, Fraud and Trust in Agent Societies, AAMAS 2002*, Bologna, Italy, (pp. 122-127).

Pujol, J. M., & Sanguesa, R. (2002). Reputation measures based on social networks metrics for multi agent systems. In *Proceedings of the 4th Catalan Conference on Artificial Intelligence CCIA-01,* Barcelona, Spain, (pp. 205-213).

Ramchourn, S. D., Huynh, D., & Jennings, N. R. (2004). Trust in multi-agent systems. *The Knowledge Engineering Review, 19*(1), 1-25.

Rasmusson, L., & Jansson, S. (1996). Simulated social control for secure Internet commerce. In C. Meadows (Ed.), *Proceedings of the 1996 New Security Paradigms Workshop,* (pp. 18-26).

Rubiera, J. C., Molina Lopez, M. J., & Muro D. J. (2001). A fuzzy model of reputation in multi-agent systems. In *Proceedings of the 5th International Conference on Autonomous Agents*, Montreal, Quebec, Canada, (pp. 25-26).

Sabater, J., & Sierra, C. (2002). Reputation and social network analysis in multi-agent systems. In *Proceedings of the 1st International Joint Conference on Autonomous Agents and Multiagent Systems*, Bologna, Italy, (pp. 475-482).

Schillo, M., Funk, P., & Rovatsos, M. (2000). 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.

Shafer, G. (1976). *A mathematical theory of evidence*. Princeton University Press.

Sycara, K. P. (1998). Multiagent systems. *Artificial Intelligence Magazine, 19*(2), 79-92.

Wang, Y., Hori, Y., & Sakurai, K. (2006). On securing open networks through trust and reputation–architecture, challenges and solutions. In *Proceedings of the 1st Joint Workshop on Information Security*, Seoul, Korea.

Wang, Y., & Vassileva, J. (2003). Bayesian network-based trust model. In *Proceedings of IEEE International Conference on Web Intelligence,* Hallifax, Canada.

Whitby, A., Josang, A., & Indulska, J. (2004). Filtering out unfair ratings in bayesian reputation systems. In *Proceedings of the AAMAS 2004*, New York.

Wong, H. C., & Sycara, K. (1999). Adding security and trust to multi-agent systems. In *Proceedings of Autonomous Agents '99 Workshop on Deception, Fraud, and Trust in Agent Societies*, (pp. 149-161).

Yu, B., & Singh, P. M. (2002). An evidential model of distributed reputation management. In *Proceedings of the 1st International Joint Conference on Autonomous Agents and Multiagent Systems*, Bologna, Italy, (pp. 294-301).

Yu, B., & Singh, P. M. (2003). Detecting deception in reputation management. In *Proceedings of the 2nd International Joint Conference on Autonomous Agents and Multiagent Systems*, Melbourne, Australia, (pp. 73-80).

Zadeh, L. A. (1989). Knowledge representation in fuzzy logic. *IEEE Transactions on Knowledge and Data Engineering, 1*(1), 89-100.