Doctor of Philosophy

TRUST MANAGEMENT FOR COMPLEX AGENT GROUPS

A THESIS SUBMITTED TO AUCKLAND UNIVERSITY OF TECHNOLOGY
IN PARTIAL FULFILMENT OF THE REQUIREMENTS FOR THE DEGREE OF
DOCTOR OF PHILOSOPHY

Supervisors
Dr Quan Bai
A.Prof. Russel Pears

August 2017

By
Doan Tung Nguyen
School of Engineering, Computer and Mathematical Sciences

Copyright

Declaration

I hereby declare that this submission is my own work and that, to the best of my knowledge and belief, it contains no material previously published or written by another person nor material which to a substantial extent has been accepted for the qualification of any other degree or diploma of a university or other institution of higher learning.

Signature of candidate

Acknowledgements

The completion of this PhD thesis is the result of a long journey learning and searching for new knowledge. With great help along the way, I have the opportunity here to express my gratitude to those who assisted me in my academic career to date.

First and foremost, they are my supervisors, Dr Quan Bai and Prof. Russel Pears, who have encouraged and guided me through my research. Not only have they given me numerous opportunities to develop my skills, but also provided great advice on the many issues of a PhD student. Therefore, I am greatly indebted to all their knowledge and support, which are essential to the success of my work.

Second, I am grateful to my colleagues, both current and past, within the school of Engineering, Computer and Mathematical Sciences at the Auckland University of Technology, who have helped me during my course of study, Minh Nguyen, Lester Khoo, Weihua Li, and Saide Lo, etc.

Last but not least, I extend my sincere appreciation to all my family members for their blessings, encouragement, and also unconditional sacrifices. Without their love, my arduous journey would have been much harder to conquer.

Abstract

In Multi-agent Systems, there are complex problems that cannot be solved by a single agent. Therefore, agent groups are formed to deal with the problems more effectively. These groups can have various types, structures, and complex behaviours. Meanwhile, the openness characteristic of multi-agent systems frequently introduces uncertainty into the environments. In such situations, it is hard or impossible to establish trust between agents and it can greatly affect the operation in the systems. Several trust models have been proposed to assist agents to make decisions during moments of uncertainty. Nevertheless, most of the existing works focus on trust evaluation for individual agents. Since complex agent groups are becoming ubiquitous in agent societies, these trust models have been exposing several shortcomings in addressing different aspects of trust in agent groups. This includes the highly dynamic and complex behaviours of them. Therefore, it is necessary to have a comprehensive trust management mechanism regarding these groups. To this end, our thesis presents a trust management stack which consists of three components, i.e., CoTrust, DBATE, and GEM. The three components manage trust in agent groups in three stages, i.e., group formation, group performance, and group disbandment. The following contributions have been made.

It provides an efficient trust establishment method regarding group formation: CoTrust is a trust establishment method devised to address the lack of effective trust mechanism in group formation. It includes a protocol and a preference reasoning method that helps agents to improve the success rate of cooperation requests and also

the satisfaction of the cooperation.

It enhances the robustness of existing trust/reputation systems in MASs: GEM is a group evidence management method to address the inconsistency of evidence generated by the rating convention when the systems consist of agent groups. GEM provides an efficient rating distribution method for group members. By using an evidence-based approach, GEM can not only guarantee the fairness of the distribution but can also significantly improve the robustness of reputation systems.

It enables agent group trust evaluation in MASs: DBATE is a trust evaluation method which aims to address shortcomings of current trust evaluation models targeting ad-hoc groups. The approach combines the power of the dynamic Bayesian network and contextual information for improving the accuracy of the evaluation. All of the proposed methods come with empirical analyses to confirm their effectiveness.

Publications

1. Nguyen, T. D. & Bai, Q. (2014). Accountable individual trust from group reputations in multi-agent systems [Book Section]. In Pricai 2014: Trends in artificial intelligence (p. 1063-1075). Springer.

2. Nguyen, T. D. & Bai, Q. (2017b, mar). Bitrust: A comprehensive trust management model for multi-agent systems. In R. Fenghui & K. Fujita (Eds.), Modern approaches to agent-based complex automated negotiation. Springer International Publishing AG.

3. Nguyen, T. D. & Bai, Q. (2016). Trust management for composite services in distributed multi-agent systems with indirect ratings: (extended abstract). In Proceedings of the 2016 international conference on autonomous agents & multiagent systems (pp. 1295–1296). Richland, SC: International Foundation for Autonomous Agents and Multiagent Systems

4. Nguyen, T. D., Bai, Q. & Li, W. (2016). Capability-aware trust evaluation model in multi-agent systems. In R. Booth & M.-L. Zhang (Eds.), Pricai 2016: Trends in artificial intelligence: 14th pacific rim international conference on artificial intelligence, phuket, thailand, august 22-26, 2016, proceedings (pp. 771–778). Springer International Publishing.

5. Nguyen, T. D. & Bai, Q. (2017c, may). Enhance trust management in composite services with indirect ratings. In The computer journal. Oxford University Press.

Contents

List of Tables

List of Figures

Chapter 1

Introduction

Trust is an important concept in human society. It helps people be confident about their decisions of daily activities. For example, one could give their car key to a friend because he is aware that this action would not do any harm to his property. In other words, he trusts his friend and believes the friend can drive safely. The potentials of trust have made itself available in several disciplines outside of social science (Helbing, 1994; Möllering, 2001), including psychology (Cook et al., 2005), economics (Granovetter, 1985; Huang, 2007), and computer science (Marsh, 1994; Huynh, Jennings & Shadbolt, 2006a; Castelfranchi & Falcone, 2010; Wang, Hang & Singh, 2011; Jøsang, 1996).

From the early days, the term 'trusted' was used mostly with the meaning of 'known to be safe' in information security (Abdul-Rahman & Hailes, 1997). Work in the area of agent-based computing has made trust a relevant research topic for computer scientists (Marsh, 1994; Castelfranchi & Tan, 2002). Since the fitrst attempt of modelling trust for agents (Marsh, 1994), a significant amount of research on trust has been proposed in the last decade to address these problems in Multi-agent Systems (MASs). Nevertheless, the rapid evolution of new technologies has brought new and unfamiliar problems to researchers. Thus, it is necessary for the definition of trust to be refreshed in order to adapt to new situations.

Amongst open problems related to trust, trust management for complex agent groups is very challenging due to several reasons. The complexity may come from the dynamic behaviours of ad-hoc groups, trust evaluation under different preferences and un-unified standards, the loosely coupled groups, and the dynamic environment. Agent groups has been playing important roles in MASs. They are responsible for the flexibility and scalability of MASs because they can solve various complex problems that single agent cannot. The ubiquity of agent groups has increased the demand for trust in the application domain.

This thesis presents a trust management stack to address the shortcomings of existing trust management models. As an introductory chapter, Section 1.1 briefly goes through some basic concepts of MASs. Section 1.2 presents the computational trust in MASs including definitions, properties, and challenges for trust management in MASs The motivations of my work, research questions, and contributions are given in Section 1.3 - 1.6, respectively.

1.1 Agents and Multi-agent Systems

MASs are important paradigm for dynamic and highly scalable distributed systems. MASs have shown their strength in modelling real-world problems, such as human society simulation (Holland, 2006), distributed systems (Jennings, 2000, 2001), social networks (Hamill & Gilbert, 2009), peer-to-peer network (Kamvar, Schlosser & Garcia-Molina, 2003), e-market places (Shoham & Leyton-Brown, 2008), virtual organisations (Rodriguez et al., 2011), and decision-support systems (Luo, Liu & Davis, 2002).

At a glance, a MAS is a computer system composed of multiple interacting intelligent agents within an environment. Typical agents are assumed to be goal-driven, self-interested and autonomous (Wooldridge, 2009). They can interact with each other in any method necessary to achieve defined goals. Thus, agents often possess the abilities to communicate, reason and make decisions to achieve goals. Wooldridge (2009) and Sycara (1998) summarised the following characteristics for MASs.

- **Local view**: agents are considered to have local views. They have incomplete information about the environment as well as limited understanding of other agents' states.
- **Limited capabilities**: agents have limited capabilities. Their abilities are restricted by the design or energy constraints.
- **Autonomous**: agents are at least partially independent and autonomous. They can make decisions and take action in some circumstances.
- **Heterogeneous**: agents are diverse in models and communication languages. They may belong to various organisational structures or topologies.
- **Decentralised**: the heterogeneity of agents does not allow a global control in MASs. Data is also decentralised and is generally stored by individual agents or local data storages. Examples of such domains include wireless sensor networks (Vinyals, Rodriguez-Aguilar & Cerquides, 2011), distributed information retrieval (Chau, Zeng, Chen, Huang & Hendriawan, 2003), smart grid system (Pipattanasomporn, Feroze & Rahman, n.d.), etc.
- **Openness:** MASs are considered open systems as they have external interactions. They often allow the members to join and leave easily, e.g., buyers and sellers of an e-commerce system

1.1.1 Agent-based approach

When investigating some complex problems in open and dynamic environments, the agent-based approach is preferable by the research community. It is a microscale model simulating the simultaneous operations and interactions of multiple agents in an attempt to re-create and predict the appearance of complex phenomena (Gustafsson & Sternad, 2010).

Most agent-based models are composed of: (1) numerous agents specified at various scales; (2) decision-making heuristics; (3) learning rules or adaptive processes; (4) an interaction topology; and (5) an environment.

In agent-based approaches, agents are considered to be the core units built with some architectures. If an agent is self-interested, the agent has its own description of states (preferences), and its actions are motivated by the preferences. Regarding agents' decisions, if an agent is greedy, it will go for the option with the highest utility value.

1.2 Trust in Multi-agent Systems

The openness characteristic of MASs enables the robustness and scalability of the systems (McArthur et al., 2007). An example for this robustness is that a system can tolerate failure due to the ability to find alternative agents for tasks. Scalability of MAS originates from its modularity. It is easier to extend the capability of MASs by adding new agents than traditional monolithic systems.

Nevertheless, a major issue of MASs also comes from the openness. We often witness joining and leaving activities in on-line social networks or e-commerce systems, such as eBay [1] or Amazon[2]. This phenomenon introduces much chaos to the systems as it may cause changes to social structures and, therefore, decrease the stability. Interactions of agents are put to risk because of the uncertainty caused by those changes. In such situations, **trust** has been considered as an effective tool for agents to initiate interactions under uncertainty.

[1] https://www.eBay.com
[2] https://www.amazon.com

Why do we need trust in MASs? Simply speaking, trust is needed in the situations that are the outputs of interactions not only depend on the capability of trustee agents but also in their intentions. In particular, when agents are developed with interactions and autonomous actions, they have demonstrated behaviours that are close to those found in human society. For example, agents can involve themselves in on-line auctions, or they can act on behalf of human users (H. Yu, Shen, Leung, Miao & Lesser, 2013; Shoham & Leyton-Brown, 2008). The level of delegation requires agents to make decisions themselves. Meanwhile, there are potential risks in such scenarios, e.g., being deceived by interacting agents.

In the following example gives a scenario in which $Alice$ wants to buy a product p from a seller in an e-marketplace. $Alice$'s agent can query and negotiate with all available sellers for p. The agent expresses $Alice$'s expectation of the product, while other sellers may try to make offers corresponding to the expectation. The question is from whom to buy p if there is more than one seller claiming that they can offer p. As a rational agent, it wants to go for the most reliable seller. However, if $Alice$ does not know any seller in person, how does the agent compare the reliability among the candidates?

In the above scenario, $Alice$'s agent has to decide which seller to rely on. As an example, all the sellers could promise to be capable of providing good products. However, the uncertainty does not allow the agent to assume that the candidates' proposals are accurate regarding competencies and capabilities. Namely, the agent must accept the fact that sellers may be intentionally deceptive, spreading false information, or having other deceptive manners for their benefits. $Alice$ must decide to purchase with some degree of risk, especially when interacting with strangers.

The environment, where $Alice$'s agent and $sellers$ reside and communicate, can be modelled by a MAS. The risks mentioned are real threats. They can cause severe breakdown of the operation and threaten the long-term well-being of the system (H. Yu, Shen, Leung et al., 2013). To diminish the uncertainty in the behaviours of partners, trust was introduced to MASs as a computational concept.

1.2.1 Concepts and definitions of trust

Trust has been studied in diverse disciplines. Each discipline may define trust from different perspectives. In the next section, some selected concepts and definitions of trust, its properties, and the relationship between trust and reputation will be discussed.

In daily interactions, there are two dominant situations involve trust. These are demonstrated in the following scenarios.

Scenario 1: $Alice$ has difficulty when dealing with a task T_k. $Alice$ asks Bob, a person she knows, to help her with the task and he accepts. In this situation, $Alice$ believes that Bob is capable of providing her with a good outcome with the task when Bob accepts her request. We may say $Alice$ trusts Bob in solving task T_k. We often refer to this situation as *direct interaction trust*.

Scenario 2: $Alice$ asks Bob to help with the T_k. Bob says he cannot do it, but he knows $Charlie$ who can. Bob introduces $Charlie$ to $Alice$. $Alice$ trusts $Charlie$ as a recommendation from Bob. In this situation, Bob indirectly helps $Alice$ in solving her problem by giving a recommendation to a suitable person.

Trust in the above scenarios can be interpreted as the truster believes that the trusted party will perform some expected actions to support its claims or plans. The concept of trust is understood by humans. However, there is no formal consensus on the definition of trust even in dictionaries. The Merriam-Webster's Dictionary defines trust as *"assured reliance on the character, ability, strength, or truth of someone or something."*[3]. Meanwhile, Dictionary.com describes trust as the *"firm reliance on the integrity, ability, or character of a person or thing"*[4].

[3]https://www.merriam-webster.com/dictionary/trust
[4]http://www.dictionary.com/browse/trust

In sociology, Morton Deutsch's definition is more widely accepted than others. It states that *"trusting behaviour occurs when an individual perceives an ambiguous path, the result of which could be good or bad, and the occurrence of the good or bad result is contingent on the actions of another person; finally, the bad result is more harming than the good result is beneficial. If the individual chooses to go down that path, he can be said to have made a trusting choice, if not, he is distrustful"* (Deutsch, 1962).

In economics, Fehr (2009) claims that trust is apparent when *"an individual (trustor or investor) voluntarily places resources at the disposal of another party (the trustee) without any legal commitment from the latter. Also, the act of trust is associated with an expectation that the act will pay off regarding the investor's goals"*.

In computing literature, Marsh (1994) is among the first to introduce the concept of trust in distributed artificial intelligence. He referred trust as *"the belief an agent has about another agent in terms of future action"*. However, the most commonly used trust definition is the one from Gambetta (2000) who defined trust as "*a particular level of the subjective probability with which an agent or group of agents will perform a particular action, both before [we] can monitor such action (or independently of his capacity of ever to be able to monitor it) and in a context in which it affects [our] own actions*".

Some studies present variations on and additions to Gambetta's definition. Although there are differences between definitions, all of them follow the similar and interrelated concepts of the expectations of trusters to trustees' actions. For example, Falcone and Castelfranchi (2001) considered risk and delegation. Trust can be defined as the belief of agents, e.g. *"an agent's belief in another agent's capabilities, honesty and reliability based on its own direct experiences"* (Wang & Vassileva, 2003), *"the belief that an entity is capable of acting reliably, dependably, and securely in a particular case"* (H. Li & Singhal, 2007), or *"the relation between beliefs and dependence on actions of others"* (Jøsang, 1996).

Similarly, Ramchurn, Huynh and Jennings (2004) modified the definition to fit their purpose as *"a belief of an agent has that the other party will do what it says it will (being honest and reliable) or reciprocate, given an opportunity to defect to get higher payoffs"*. This definition assisted the use of game theory in modelling trust. Mui et al. (2002) defined trust as *"a subjective expectation an agent has about another's future behaviours based on the history of their encounters"*. These definitions classify trust as a subjective probability and expectation, which are more convenient to model with probability theory.

In this thesis, I also give my definition of trust in Section 1.5, which governs the direction of this research toward agent groups.

1.2.2 Properties of trust

Properties of trust vary from context to context. Listed below are some properties commonly applied in computer science. Firstly, trust is not inherently transitive (non-transitive). Namely, if A trusts B and B trusts C, it is not plausible to conclude that A also trusts C. However, under some conditions, we can say A trust C to some extent, e.g., the trust discount in subjective logic (Jøsang, Hayward & Pope, 2006) or conditional transitivity as proposed by Abdul-Rahman (Abdul-Rahman & Hailes, 1997). It is largely based on the recommendations of B and how much A can judge the recommendations.

Secondly, trust is typically asymmetric. Namely, if A trusts B, it does not imply that B trusts A. This phenomenon is the results of the differences in roles and preferences between A and B. It is particularly common in open MASs where agents can be heterogeneous by beliefs, opinions, perceptions, and expectations. However, B. Yu and Singh (2000) found that when both agents are trustworthy, repeating interactions will build up their mutual trust to high values.

Thirdly, a trust value may change over time (dynamic), and the most recent value of trust is the most informative (S. Liu, Yu, Miao & Kot, 2013; Ramchurn et al., 2004; W. T. L. Teacy, Patel, Jennings & Luck, 2006; Marsh, 1994; Mui et al., 2002; Commerce, Jøsang & Ismail, 2002). Due to the dynamic behaviour of agents, the trustworthiness of an agent may vary from time to time. Thus, it is essential to update frequently.

Fourthly, trust can be subjective and self-reinforcing. Namely, when an agent chooses to trust or distrust another agent, it can be a personal choice. Each agent may have distinct preferences or interests (subjectivity) that influence their trust evaluation. For example, one may trust agent P because of the quality of some products whilst many other agents do not trust P regarding on-time delivery. However, the first agent may change their belief if they experience bad service. In other words, individuals continue to trust (self-reinforcement) beyond the point where the evidence points to the contrary (Anand, Gai & Marsili, n.d.; Fang, Zhang, Şensoy & Thalmann, 2012).

Finally, trust is context-aware (W. T. L. Teacy et al., 2006; Toivonen, Lenzini & Uusitalo, 2006; Haibin, Yan & Xiuzhen, n.d.; X. Liu & Datta, 2012). Namely, different scenarios many have different types of trust. Moreover, trust does not stand alone; it is bonded with some relationship. Trust is evaluated and established through the connections between two or more entities.

1.2.3 Trust VS. Reputation

In many studies, trust and reputation frequently go together. Sometimes, they are used interchangeably due to the close relationship between them. In this section, instead of clarifying these two concepts, only some related definitions of reputation and its relationship with trust are listed.

The Merriam-Webster dictionary defines reputation as an *"overall quality or character as seen or judged by people in general"*[5]. It is similar to Abdul-Rahman and Hailes (2000)'s definition, where they state that it is the collection of opinions received from other users or an expectation of someone's behaviour based on previous interactions indicated by others . Kreps and Wilson (1982) defined reputation as a characteristic attributed to someone by another person or community. Meanwhile, Mui et al. (2002) defined reputation as *"the perception agents create through past actions about their intentions and norms, being related to expectations held by others"*.

In a society where reciprocity is normal, trust and reputation are linearly related (Dasgupta, 1988). Figure 1.1 illustrates the relationship between trust, reputation, and reciprocity. They are embedded in a social network where they are expected to follow these rules: (1) the increase of reputation of agent a_i can lead to the increase of trust from the other agents for a_i; (2) the increase of trust in agent a_j for agent a_i should increase the likelihood that a_j will reciprocate positively to a_i's actions; (3) the increase of a_i's reciprocating actions to other agents should increase a_i's reputation (Mui et al., 2002).

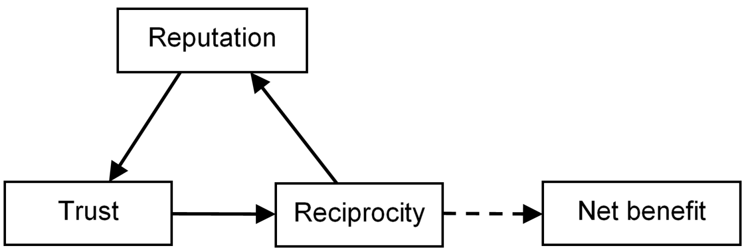


Figure 1.1: Trust and reputation relation (Mui et al., 2002)

Marsh (1994) described reputation as "*the amount of trust inspired by a particular person in a specific setting or domain of interest*". Reputation is generally associated with the past performance of an agent (e.g., past cumulative rating scores) (Jøsang et al., 2006). While trust is the individual subjective opinion about someone, reputation

[5]https://www.merriam-webster.com/dictionary/reputation

is more about referential opinions collected from different agents. Thus, the trust value is more important for an agent's decision because the value reflects its subjective evaluation. In the other hand, the reputation value can be a good reference.

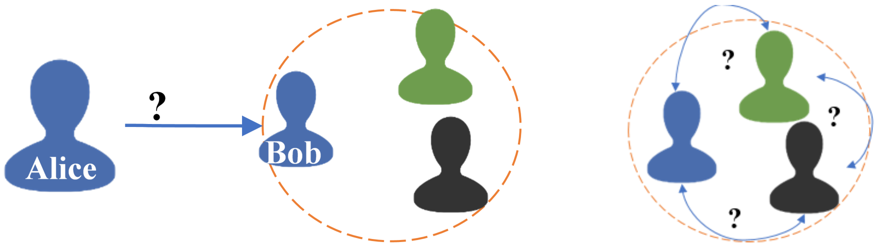
The trust and reputation relationship can be briefly summarised as follows.

- A high reputation does not coincide with a high trust value due to the multi-facet of trust. For example, if B is reputable for doing T_1 and bad at doing T_2, then A may distrust B when doing T_2.
- In the same context, low reputation often leads to low trust. It is a direct result of the previous property.
- The reputation of an agent can be built up by a cumulation of trust from other agents. This property can be inferred from some definitions.
- Like trust and distrust, a reputation can be either good or bad. If most agents experienced bad service from provider P, then P is said to have a bad reputation.

1.3 Research Motivations

Reconsidering the scenario in which $Alice$ is going to buy a product from Bob. A question is how $Alice$ evaluates the trustworthiness of Bob as a seller? Obviously, the accuracy of the evaluation depends on the amount of accurate information which $Alice$ has about Bob. In a normal situation, $Alice$ would see Bob as a single individual who is in charge of all aspects of the product, from handling requests to delivery. However, in reality, Bob can also be a name representing a group of several agents working together.

Apparently, Bob is the one who has the most information about their performance. If something happens during processing the request, whilst Bob will hear of it, $Alice$



(a) Different views of Alice and Bob

(b) Trust-aware group composition

Figure 1.2: Trust in different scenarios

will not. Being unaware of such events may result in an inaccurate trust evaluation from $Alice$ regarding interaction decisions with Bob.

From the above scenario, we can also observe two interaction scopes which involve trust, i.e., (1) $Alice$ - Bob, and (2) Bob and his team members. The former is often referred to the consumer-provider (or seller-buyer) relationship. In this case, $Alice$ needs to be more confident that Bob will act honestly, i.e., Bob will fulfil his commitment by providing $Alice$ a service as advertised.

The latter can be seen as a cooperative relationship. Namely, Bob and his team members collaborate in order to provide the service to $Alice$ (Figure 1.2a). The quality of the service depends not only on Bob but also on all other members. How can $Alice$ accurately evaluate the trustworthiness of the whole group of Bob?

For example, to deal with $Alice$'s request, Bob may invite $Charlie$ to join the group (team) that is managing $Alice$'s request. The situation is referred to "group formation" or "team composition". In this case, both Bob and $Charlie$ need to evaluate the quality of their cooperation. Again, $Charlie$ can be either individual agents or a group of agents. The problems that Bob encounters in a situation like this include the trust or distrust between Bob and Charlie during their cooperation, estimating the trustworthiness of the cooperation and maintaining the evidence after the group disbands (see Figure 1.2b).

In MASs, trust is important not only for individual agents but also for agent groups.

Nevertheless, there is little research on trust management for agent groups in MASs (see Chapter 2 for the detailed literature review). Current trust models have exposed shortcomings in addressing complex agent groups.

Below is a summary of some critical challenges regarding trust management for agent groups:

- **Dynamism**: agent groups are formed due to specific requirements. They may exist for a short period before disbanding (ad-hoc groups). Group formation may happen rapidly. The lifetime of a typical ad-hoc group may not be long enough to allow the establishment of trust for members inside or outside of the groups (Jarvenpaa & Leidner, 1999). Additionally, as members can join and leave the system, it reduces the chance for those members to interact and form groups again, especially in a system where the population is large or the topology is sparse. Consequently, the phenomenon affects the incentives to develop long-term trust relationships as well as cooperative behaviours (Meyerson, Weick & Kramer, 1996).

- **Diversity**: agents are diverse in many aspects, e.g. architecture, belief, intention. Agents may have different experience which leads to several distinct opinions. As a result, ad-hoc groups often lack good coordination and communication protocols that can sometimes be found in other well-established groups (Lewicki, Tomlinson & Gillespie, 2006).

- **Instability**: several factors can affect the performance of agent groups. Internally, a member can be a stranger to other group members. This means that they may have little or no time to develop a sense of trust before collaboration. Externally, to other agents, ad-hoc groups are often seen as unstable and risky to interact with (Comfort, Ko & Zagorecki, 2004). It is also hard for group themselves to ensure their commitment even when all group members behave honestly. Therefore, the applications which involve ad-hoc groups often contain higher potential risks and cost of failure.

Agent groups are complicated entities and have much more dynamic behaviours compared to individual agents. Employing approaches, which are designed for individual trust evaluation, can be ineffective for evaluating the trust of agent groups. Many assumptions, which are frequently adopted in studies of trust for individual agents, do not hold for agent groups. Furthermore, there are several aspects of trust which are ignored in individual trust evaluation yet cannot be ignored for agent groups (Sen, 2013; Tong & Zhang, 2009). All these aforementioned challenges of trust in complex agent groups have led us to the topic of this thesis.

1.4 Research Questions and Objectives

Outside of the literature review in Chapter 2, I have investigated trust in different stages of an agents group in order to identify the research questions for trust management of complex agent groups. Generally, the life cycle of an agent group goes through (1) group formation, (2) group performance, and (3) group disbandment. Each stage involves different types and roles of trust. To effectively address the trust in complex agent groups, I propose a trust management stack which consists of three corresponding components, i.e., trust establishment for group formation, managing trust evidence of agent groups, and trust evaluation of agent groups. The stack is illustrated in Figure 1.3.

1.4.1 Research questions

The objectives of this research are embodied in the following three questions:

Research question 1: How does trust play a role in the formation of agent groups?

Group formation indicates the process where agents start to search for suitable

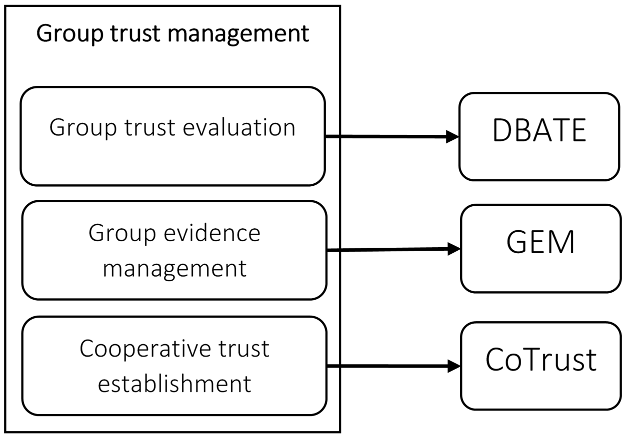


Figure 1.3: Trust management stack for agent groups

partners and agree to work together as a team. This question attempts to discover an appropriate trust mechanism used to initiate agent cooperation for group formation. Some sub-questions that can be considered are: what are the differences between trust in cooperation context and trust in consumer-provider relationships? How can an agent gain trust from other agents? What is a mechanism that can be used to support the trust establishment among group members?

Research question 2: How can trust evidence be managed in MASs that consist of complex agent groups?

Quality evidence is essential for any trust and reputation systems. The differences between a group and an individual require evidence to be managed differently. The question attempts to find out the problem associated with evidence management in agent groups. Without a proper method, it may affect the accuracy of trust evaluation for future interaction as well as group formation.

Research question 3: How can the trustworthiness of agent groups be evaluated accurately?

As the behaviours of complex agent group are dynamic and have a high uncertainty level, this question investigates an efficient method that evaluates the trustworthiness of agent groups. Further questions include: What are factors causing the dynamic behaviours? How can factors be utilised to improve the accuracy of the trust evaluation for groups?

1.4.2 Research scope

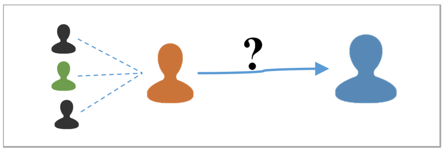
To answer the research questions, it is essential to clarify the scope of this research.

What type of agents to be considered? In many places of this research, examples of agents can be drawn from human society in order to demonstrate some ideas. However, agents in this research represent software agents, which are computer programs that act on behalf of users. They can either be of buyer agents or shopping agents (Aref & Tran, 2015), user or personal agents (H. Yu, Miao, An, Shen & Leung, 2014), monitoring-and-surveillance agents, and data-mining agents (Haag, Cummings & Phillips, 2007).

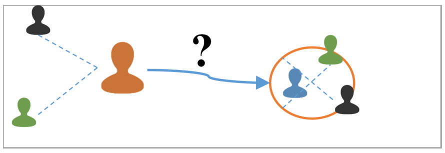
What type of agent groups are considered? This research considers ad-hoc agent groups as the main targets. An ad-hoc agent group is a group formed to solve an immediate problem. The lifetime of an ad-hoc group, from the formation to disbandment, is considered shorter than other types of groups, e.g. an organisation. Furthermore, the members of an ad-hoc group may change frequently. This thesis uses the term agent groups with the same implication.

What trust relationships are considered? In this research, we only investigate the trust built up from agents' commitments and its actual actions. Namely, trust value is calculated based on evidence rather than socio-cognitive approaches (refer to Chapter 2). This research does not involve trust with internal characteristics, e.g. mental states of goals and interests (Falcone & Castelfranchi, 2001; Castelfranchi & Falcone, 2010), or trust involving security, organisational protocol or platforms as seen in Kollingbaum and Norman (2003); Huynh et al. (2006a); Hermoso, Billhardt and Ossowski (2010), or trust of information sources (Huynh et al., 2006a; Jiang & Bai, 2013a; Fang et al., 2012). This means that trust between humans and agents is not in the scope of this research. Throughout this thesis, trust between agents or agent groups is based on my definition as found in Section 1.5.

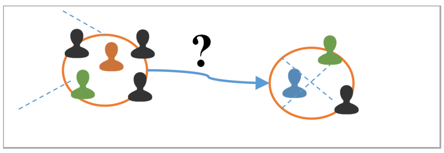
What type of trust evaluation for groups are considered in this thesis? Figure 1.4 illustrates the three common types of evaluation. There are two types of group trust evaluations: (1) One-to-many *(1-to-m)* refers to the case when an individual evaluates the trustworthiness of an agent group (see Figure 1.4b). (2) Many-to-many *(m-to-n)* refers to the case when an agent group evaluates the trustworthiness of another group. While most previous studies only measure the trust from one agent to another agent (Figure 1.4a), this thesis addresses *1-to- m* group trust evaluation, which models trust in terms of collaboration. Many to many *(m-to-n)* is even more complicated compared to *1-to-m*. For that reason, I intend to leave it for future work. Thus, *m-to-n* trust is not in the scope of this research.



(a) $one - to - one$ trust evaluation



(b) $1 - to - m$ trust evaluation



(c) $m - to - n$ trust evaluation.

Figure 1.4: From individual to group trust evaluation.

1.5 The Proposed Approach

To realise the research objectives, first, a definition of trust toward groups in MAS is given. Then, I will answer each research question one by one in a corresponding chapter by elaborating the problem and possible solutions. I also provide the evaluation of the proposed method in the same chapter.

1.5.1 Trust definition toward agent groups

In this thesis, I use a definition of trust, which is adapted from (Gambetta, 2000; Chang, Dillon & Hussain, 2005; H. Li & Singhal, 2007; Romano, 2003). The definition of trust that governs my research direction toward agent groups is stated as follows:

Definition 1.5.1. Trust (in the context of MASs) is a harmonisation of subjective assessments of a rational entity, who is either an individual agent or an agent group, using evidence collected from its connections over another entity regarding the quality of output that the entity will perform in a given situation at a given time.

The definition highlights three factors to be investigated in this study. Firstly, trust in this study is not restricted to individual agents. Rational entities can be either an individual agents or agent groups.

Secondly, the trust evaluation in this study will use an evidence-based approach. Trust is not only about the subjective assessment of an entity; it is also about the harmonisation between many other information sources collected by an individual agent or agent group. The term “connections” in the definition implies that social relations among agents may also be taken into account for trust evaluation.

Thirdly, trust is context and temporal-dependent. My research will take contextual information and temporal factors into consideration. Though these concepts are proposed by some researchers, they have not been thoroughly discovered in the context of complex agent groups.

1.5.2 Evaluation methods of this research

Agent-based modelling is used in simulated environments and evaluations. Agents' decisions are regulated using parameterized utility functions adopted from game theory. Table 1.1 summarises some factors to be considered when constructing the evaluation environments.

Table 1.1: Evaluation setting considerations

	Factors	Arguments
Agent	- Autonomous - Adaptive - Limited capability - Self-interest	Components of agent architecture used in this thesis
Trust dynamism	- Temporal factor - Member switching - Member relationship	How to better utilise these factors to enhance the accuracy of trust evaluation?
Trust establishment	- Satisfaction measure - Subjective preferences - Protocol	How can an agent gain trust from other agents with these approaches
Trust evaluation	- Evidence-based approach - Distributed environments - Data collection - Learning model - Evidence management	What is the suitable data collection and learning algorithm for trust measure in distributed environments?
Dataset and evaluations	- Simulation	

As our trust management stack consists of three components, it would be ideal if the evaluations could be conducted with systems which implements the all the components of the proposed stack. Unfortunately, there is not an equivalent work on trust management for agent groups for a comprehensive comparison. Also, the three components are applied in three different stages of agent groups. Therefore, I will assess the effectiveness of each component separately by comparing with state-of-the-art approaches with the assumption that if all components perform well in each stage then the whole trust management stack can perform well.

For experiments, I choose to use the sample application from the service-oriented systems because service composition problems are popular in the domain. Also, it is more intuitive for demonstrating agent cooperation and group formation. Each evaluation requires different assessment criteria. This can include the accuracy, consistency, and satisfaction, which are elaborated in the relevant chapters.

1.6 Contributions

By fulfilling the objectives defined in the previous sections, the contributions of this work to the field are summarised as follows.

- **It provides an efficient trust establishment method regarding group formation:** Trust establishment plays a significant role in group formation. Nevertheless, there is not effective approach that is applicable for group formation. Therefore, CoTrust is devised to address the need of trust during group formations. It can improve the success rate of request and satisfaction of cooperation. To my best knowledge, no equivalent work for trust establishment for agent groups has been done before. The related results have been published in Nguyen and Bai (2017a).
- **It enables agent group trust evaluation in MASs:** Dynamism is one of the most challenging problems for trust evaluation of (ad-hoc) agent groups (refer Section 1.3). DBATE is a flexible and effective trust evaluation method dedicated for agent groups. DBATE uses features from contextual data to evaluate the trustworthiness of the targets. The relevant work has been published in Nguyen, Bai and Li (2016).
- **It enhances the robustness of existing trust/reputation systems in MASs:** Trust and reputation systems need quality evidence to function. However, the

presence of complex agent groups can break down the consistency of the evidence and affect the well-being of the systems. To this end, GEM is an effective evidence management method which mainly addresses the rating distribution among group members. The distribution method not only can guarantee the fairness of the distribution but also enhance the robustness of the reputation systems by diminishing the inconsistency of rating evidence. This work has been published in Nguyen and Bai (2016, 2017b).

- **It extends the applicability of trust in MASs:** With the proposed trust management stack, trust can now be used to deal with a wider range of interaction situations. Its application is not restricted to only agent groups. For example, a single agent can be seen as a group of one member. Therefore, DBATE can work perfectly fine with the case of individual agent.

1.7 Thesis Structures

The remainder of the thesis is structured as follows.

- **Chapter 2** provides a detailed review of several state-of-the-art studies of computational trust in MASs.
- **Chapter 3** presents CoTrust, a trust establishment mechanism for agent cooperation. It helps to improve the success rate of cooperation requests and satisfaction of cooperation.
- **Chapter 4** introduces a method for managing evidence in agent groups. The work enhances the accuracy for trust management in MAS with the presence of agent groups.

- **Chapter 5** presents DBATE, an effective and flexible trust evaluation method, to measure the trustworthiness of agent groups.
- **Chapter 6** summarises the contributions of this thesis, the advantages and limitations of proposed approaches, as well as outlines for future work.

Chapter 2

Literature Review

Studies of trust and its applications in MASs have drawn much attention in the last decade. Trust models have been devised to "*support the accumulation of member reputation information and leverage this information to increase the likelihood of successful member interactions and to better protect the community from fraudulent members*" (Gal-Oz, Grinshpoun, Gudes & Friese, 2010; Grinshpoun, Gal-Oz, Meisels & Gudes, 2009). There are several successful systems assisted by trust, e.g., Web search with PageRank (Page, Brin, Motwani & Winograd, 1999), finding most reliable sellers in E-Commerce systems (eBay, Amazon), finding good answers for questions (Stack Exchange [1] and Quora [2]), etc.

This chapter presents a detailed review of studies related to trust management in MASs. Section 2.1 gives an introduction of trust modelling and relevant criteria. Sections 2.2 and 2.3 classify existing studies of trust into several categories based on trust information sources, trust propagation techniques, and contextual information used in each study. Finally, the chapter is summarised in Section 2.4.

[1]https://stackexchange.com
[2]https://www.quora.com

2.1 Trust modelling in Multi-agent Systems

The vital role of trust in MASs is to reduce the uncertainty in interactions and help agents make faster and better decisions (W. T. L. Teacy, Patel, Jennings & Luck, 2005; Gambetta, 2000; H. Yu, Miao, An, Leung & Lesser, 2013). Figure 2.1 depicts a typical trust model and its relationships with other trust-aware components (Dondio & Longo, 2011). A trust model often comes with its domain application, evidence used for computing trust, and the algorithm to generate trust values. The obtained trust values can be combined with other components, e.g., risk management and disposition, and then be used for agent's decision making.

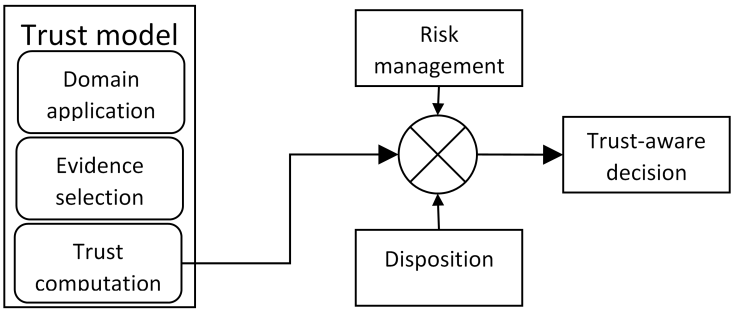


Figure 2.1: A trust model and related modules (Dondio & Longo, 2011)

As trust is a multi-faceted concept, each researcher may define trust from a different aspect and with a different focus (see Section 1.2). In turn, each definition can open up a series of issues as well as solutions for trust management in MASs. The design objectives of trust models are also different. For example, some models target at the evaluation of trust; some target at the use of trust, i.e., how to use trust in decision makings.

Sen (2013) summarised four research areas that are requires to achieve comprehensive trust management, i.e., trust evaluation, trust establishment, trust engagement, and trust use.

- Trust evaluation is to measure the trustworthiness of another agent given its history of interaction. This is the most frequently cited and studied aspect of trust management in literature.

- Trust establishment is opposite to trust evaluation. It emphasises how an agent actively gains trust from others. More specific, this module determines the actions and the resources to be invested by an agent in order to establish trust relationships with others. For example, a new provider in the market needs to determine how much time, effort, and resources to allocate to processing the task (or contract) awarded by a lucrative customer. In this thesis, I further elaborate the term into passive and active type.

- Trust engagement enables rational agents to select carefully and with strategic intentions interact and engage other agents for evaluating their trustworthiness. In other words, agents can actively create circumstantial interactions to judge another agent's trustworthiness. It is essential for new agents to have a chance to gain trust. Existing agents may have to create situations or allocate tasks for initiating this trust for a newcomer. This module is important for exploring new alternative partners.

- The use of trust 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 long term. Developing trust models is the key. However, out of necessity, they must be coupled with an effective decision procedure to utilise the knowledge. For example, given different interaction histories with different agents, an agent must carefully consider using the existing trust information or exploration for new trustee agents in the system.

In the four aspects, trust evaluation is the most studied one. The use of trust is the direct application of trust evaluation. Therefore, there has also been significant research undertaken that is related to the use of trust. In contrast, trust establishment and engagement are less investigated. Despite of this, they have huge potentials for future research.

Starting from trust evaluation, I summarise some important criteria to consider when designing a trust model. The criteria are extracted from a study of test-bed framework (Fullam, Park & Barber, 2005) and up-to-date surveys of trust management in MASs, e.g., H. Yu, Shen, Leung et al. (2013); Granatyr et al. (2015).

1. Accurate for long-term performance: A trust model must be an accurate predictor of another agent's future behaviour. The accuracy can be measured by the difference between the value generated by trust models and the trustee's trustworthiness. The model should reflect the confidence of a given reputation value and be able to distinguish between a new entity of unknown quality and an entity with bad long-term performance.

2. Efficiency: Time and effort for constructing models should be minimal. Computational efficiency can be gauged by time to complete a trust model update. Trust evaluation sometimes is expensive due to the limited capability of agents while there is a huge amount of data to collect and process. A good model should allow calculating trust value quickly with reasonable effort. For example, a known seller's reputation should be calculated in faster way compared to a new one.

3. Toward latest behaviour: The model should be able to capture the latest trends in the performance of the target. This objective is similar to short-term performance evaluation that focuses on the trustworthiness of an agent at the time of making the request. This criterion is also essential to address the dynamism of trust.

4. Fast converging and adaptation: Trust modelling algorithms must be able to create usable new models quickly when unknown agents enter the system. Trust models

must be able to accommodate dynamic trustworthiness characteristics of other agents, who might suddenly lose competence or maliciously employ strategies to vary trustworthiness.

5. Robustness: The model should be able to be stable when working under different critical conditions, such as Sybil or the whitewashing attack. This is necessary in order to identify the trustworthiness of agents accurately. The model should be able to resist any attempt to manipulate ratings. Namely, ratings should reflect the true behaviours or quality of services.

6. Explainable to statistical evaluation: Small amount of new ratings should not have a big impact on the reputation value. This property can prevent negative affect of biased ratings. However, it is not good at capturing the dynamic behaviours of trustees. More importantly, if there is any change in the evaluation, it should be explainable. Namely, it should be possible to identify the factors that produced such changes. This criterion highlights the ability to distinguish the honesty and performance influential factors. It is important to improve the evaluation in different contexts of interaction.

7. Understandable and verifiable: The produced reputation values should be presented in a manner that is easily understood for human users. Moreover, it should be possible for users to reproduce the value with existing data.

8. Multiple trust dimensions: Trust models must be able to distinguish between another agents' varied trustworthiness characteristics across multiple categories.

9. Privacy: Agents' preference should be private and can be obtained by direct interactions. This objective is designed to avoid the exploitation of the personal information and behaviours.

2.1.1 Common assumptions in trust models

Generally, a trust model is devised based on defined objectives, scopes, assumptions. Currently, there is no commonly accepted trust model that allows for any context. Each trust model is designed to address some specific problem. Researchers often use assumptions to enable the trust model to work in certain conditions.

There are some assumptions, which are accepted by most researchers in the field:

(a) Agents are self-interested;

(b) Agents prefer to interact with the most trustworthy agent (greedy);

(c) Identifying a trustee agent requires at least an identity.

Beside the common assumptions, some researchers also provide some special assumptions to simplify their research problems. For example:

(i) Trust can take binary value (trust or distrust) (W. T. L. Teacy et al., 2006; Commerce et al., 2002).

(ii) Agent interactions happens in discrete (or equal) time steps (W. T. L. Teacy et al., 2006; Commerce et al., 2002; Kerr & Cohen, 2006).

(iii) The majority of ratings or third party's testimonies are reliable (Commerce et al., 2002; Jøsang & Haller, 2007).

(iv) The public characteristics of a trustee agent are useful for predicting agent's behaviour.

(v) A truster agent selects only one trustee each time (W. T. L. Teacy et al., 2006; Commerce et al., 2002; Sabater & Sierra, 2001a; Su, Zhang, Mu & Bai, 2013; Wang & Singh, 2007).

(vi) The number of requests accepted by a trustee agent in a time step does not affect the performance of the agent.

(vii) The result of an interaction can be assessed as soon as the interaction is completed.

Though these assumptions help to simplify the models, they restricts the applications of trust models. Researchers have relaxed some assumptions as the application domain evolves. For example, assumption (i) was relaxed in Jøsang and Haller (2007), assumption (ii) was relaxed in S. Liu, Kot, Miao and Theng (2012), assumption (iii) was relaxed in W. T. L. Teacy et al. (2006, 2005), assumption (vi) was relaxed in H. Yu et al. (2014), etc.

2.2 Trust research categorisations

In this section, I firstly summarise the existing reviews on computational trust, especially for trust evaluation (see Table 2.1). After that, I will provide a graph (see Figure 2.2) which summarises the different trust research areas other than trust evaluation.

Below, these reviews summarises several aspects of trust which are used mainly in trust evaluation models. They are reviews from different researchers in the past ten years, namely, Sabater and Sierra (2005); Lu, Lu, Yao and Yip (2009); Pinyol and Sabater-Mir (2013); H. Yu, Shen, Leung et al. (2013); Granatyr et al. (2015). Each review contains a set of categories or dimensions. Firstly, their work is summarised in Table 2.1. The categorisation of trust/reputation studies is presented afterwards.

Table 2.1: Existing reviews of trust/reputation models in MASs

(Ramchurn et al., 2004)	(Sabater & Sierra, 2005)

Individual level - Socio-cognitive models - Reputation models - Evolutionary and learning System level - Trustworthy interaction mechanisms - Reputation mechanisms - Distributed security mechanisms	Paradigm type Information sources Visibility Granularity Cheating assumptions Type of exchange information Reliability measure Type model
(Lu et al., 2009)	(H. Yu, Shen, Leung et al., 2013)
Semantics Architecture Mathematical model Trust network Reliability Risk Dimension	Direct trust Indirect \Reputation-based trust Organizational trust Socio-cognitive trust
(Pinyol & Sabater-Mir, 2013)	(Granatyr et al., 2015)

Trust	Paradigm type
Cognitive	Information sources
Procedural	Cheating assumptions
Generality	Trust semantics
	Trust preferences
	Delegation
	Risk Measure
	Incentive Feedback
	Initial trust
	Open environment
	Hard security

The review of Granatyr et al. is the most recent. It combines different trust dimension from other reviews with some other dimensions defined by the authors. Nevertheless, the review mainly investigates the evaluation aspects of trust management. In comparison, my classification of trust management studies includes both four research directs adapted from Sen (2013).

Figure 2.2 illustrates my categorisations as adapted from previous works. It is not difficult to see which area has been investigated extensively and which is not. This categorisation is based on the view of comprehensive trust management proposed by Sen (2013), i.e. evaluation, establishment, engagement, and use.

2.2.1 Trust patterns

The patterns of trust, also know as paradigm types, are coarse categorizations for trust based on its representations proposed by Sabater (Sabater & Sierra, 2005). In this work, trust is classified as two paradigm types, i.e., the cognitive and numerical paradigms.

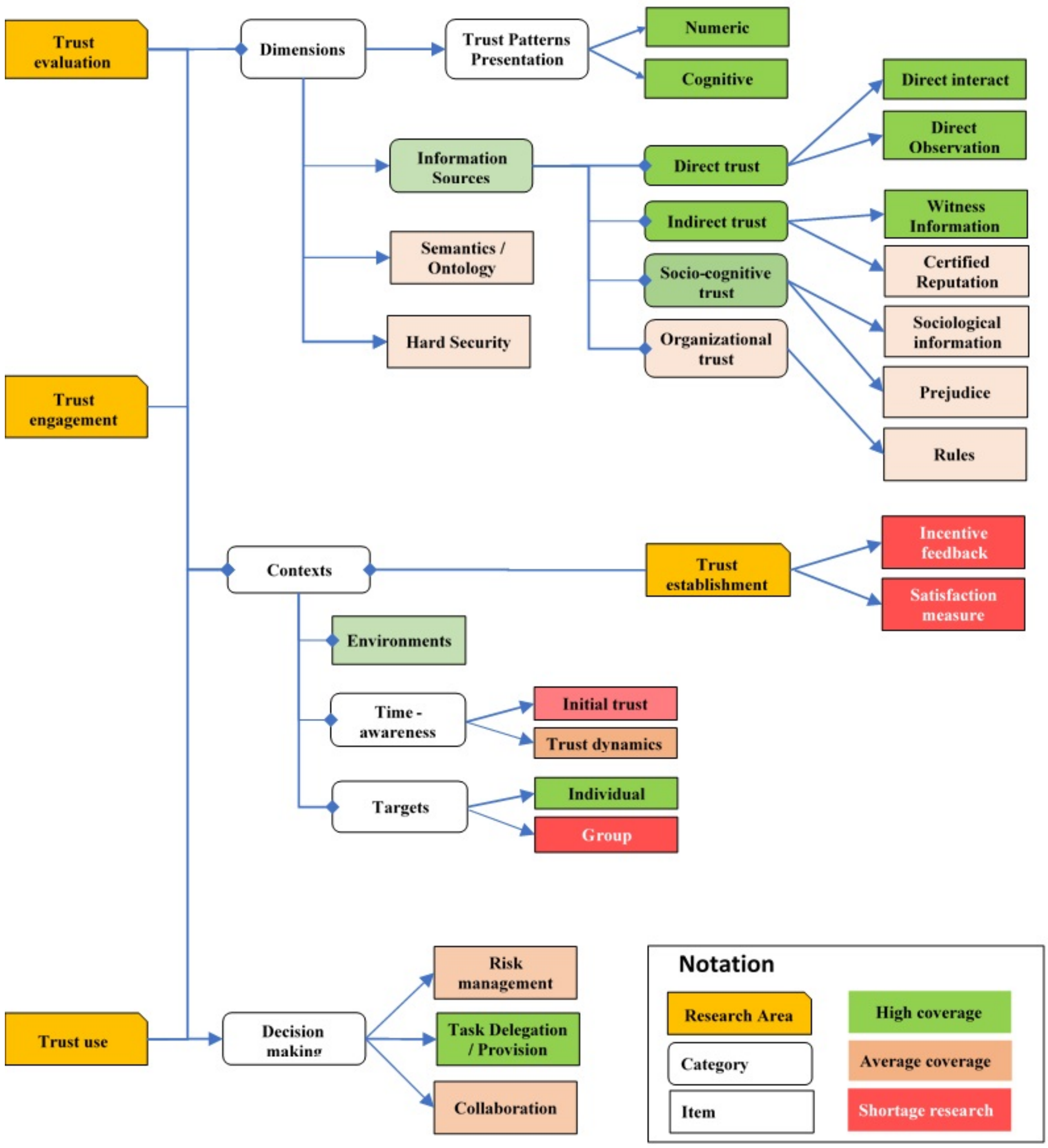


Figure 2.2: Trust research categorisation

The cognitive paradigm is based on the beliefs and mental states of agents. Relevant trust models contains agents that their behaviours is closest to human behaviours. The way that trust is calculated is also similar to human' mind. The works of Falcone and Castelfranchi (2001); J. Zhang, Ghorbani and Cohen (2007); Villata, Boella, Gabbay and van der Torre (2013) are typical cognitive paradigms.

The numerical paradigm is based on numerical representations using the aggregation of past interactions. This approach often involves subjective probabilities for the performance of a targeted agent. The commonly used techniques can be Bayesian probabilities (W. L. Teacy, Luck, Rogers & Jennings, 2012; Wang et al., 2011; Jøsang, 2007), and Dempster-Shafer functions (S. Liu et al., 2012).

Using numeric values to present trustworthy levels of agents is preferable because it is more intuitive and understandable to humans. Trust values can be either discrete or continuous. It is also easier for agents to compare trust and make decisions (Sabater & Sierra, 2005).

2.2.2 Information sources for evaluation models

The most relevant data for calculating trust and reputation values is from the evidence collected from the environment. It is also the most influential ingredient for building a trust model. Regarding evidence collection, researchers often consider two sources of information, i.e., direct trust evidences and indirect trust evidences. The former is often seen as the experience of direct interaction or the observation of the truster agent itself. The latter is referred to as trust reports from third parties. The use of third-party reports can be divided further into two major types, i.e., evidence aggregation and filtering approaches. There are also approaches which are based on evidence analysis, e.g., socio-cognitive and organisational models.

Direct trust evidence

Direct trust refers to models with accessible and verifiable evidence. Namely, the evidence comes from agents' own experience (repeated direct interactions) or from witnesses (direct observations).

Direct interaction often refers to the case when A gains experience by interacting directly with B. Through observing the outcomes of past (repeated) interactions between A and B, A can decide whether to carry on future interactions. Typical models, which use solely direct experience in order to estimate the reliability of the agent's task delegation, include Marsh (1994) and Griffiths (2005). Although direct experience is considered as the most relevant and accurate information, trust models of this type are not applicable for agents with no previous interactions.

Direct observation is more popular than direct interaction because it is helpful for evaluating trust in the first interaction. It overcomes the shortcoming of the direct interaction model when there is no experience between agents to compute trust. For example, when A wants to buy a product from B, A can observe B's behaviour through B's past interactions with others. Ratings from verified buyers on Amazon or eBay is a good example of the direct observation information source. Users of the websites can use the ratings to decide whether to buy products from sellers.

Direct observation can be in different forms. Sierra and Debenham used direct observation which is based on the fulfilment of the contract using information theory in the context of agent negotiation (Sierra & Debenham, 2005). Rettinger, Nickles and Tresp (2008) used data from the eBay feedback system and proposed the Infinite Hidden Relational Trust Model (IHRTM) to extract some hidden information using combined machine learning techniques. Nevertheless, the accuracy of trust value depends on the quality of feedback. To that end, W. T. L. Teacy, Chalkiadakis, Rogers and Jennings (2008) proposed a model that filters the inaccurate evidence.

Wang et al. (2011) used direct observation for their trust model to quantify the uncertainty from obtained evidence. The authors considered the case that agent A has few good experiences with agent C and the success rate is high (e.g., 90%). Another agent B has transacted with C for many times but the success rate is low (e.g., 60%). The question here is determining which evidence set is better for estimating the trustworthiness of C. Wang and Singh (2007) addressed this question by calculating the uncertainty in evidences using a probability distribution of rating values, which can be either positive or negative. The study points out that the uncertainty is low when the evidence space is small and when the conflicts of feedbacks are high.

Indirect trust evidence

Undoubtedly, direct interaction and observation are the most relevant sources for evaluation. However, the information is not always available. Another type of evidence is required to help truster agents to evaluate the trustee agent. The evidences are found in reports, recommendations or opinions from other agents about a targeted agent. It is important to emphasise that those agents may not have direct experience with the target. Thus, there are two main approaches, (1) recommendations or opinions, and (2) aggregation with subjective logic.

Recommendations are useful when there is no direct trust information. To obtain information about B, A may query C, who knows B, for C's opinion about B. This method is also known as the indirect information or reputation-based model as found in H. Yu, Shen, Leung et al. (2013). The trust is typically evaluated by aggregating the opinions of agents who have knowledge about the targeted agent.

An early trust model that tries to derive trustworthiness through combining feedback from multiple resources is the Beta Reputation System (BRS) (Commerce et al., 2002). Inspired by the Beta distribution, BRS measures a trustee agent's trustworthiness by projecting its past interaction experience into the future. The trustworthiness of an agent is calculated by the expectation value of the distribution of the good and bad feedbacks about the agent. To overcome the limitation of binary outcomes of BRS, Jøsang and Haller (2007) used the Dirichlet distribution for a multinomial scale rating (Dirichlet Reputation System). However, both approaches are prone to biased opinions, such as the Sybil attack (Douceur, 2002), and they require sufficient amounts of evidence to operate effectively.

To address inaccurate feedback, W. T. L. Teacy et al. (2006) proposed the TRAVOS model. This model does not assume that the majority of feedback (or ratings) is reflective of true information, unlike BRS. TRAVOS calculates the posterior probability of the likelihood function of trust values based on the beta distribution. However, in order to handle inaccurate reputation, TRAVOS assesses each source individually using the accuracy observed from past opinions.

To further tune the performance, Griffiths (2005) adopted a multi-dimensional trust with four aspects. The trustworthiness was then measured by a weighted average of these dimensions. The weights are set based on the personal preferences of truster agents. Similarly, Su et al. (2013) introduced the PBTrust model which presents trust as a multiple value vector rather than combining the multi-attribute evaluation into a single trust value. Trustee and truster agents' communication involves the distribution of interest over different attributes of a service. By doing so, truster agents can better select service providers based on separate attribute reputation. Furthermore, the reputation of trustee agents can be maintained better than using trust with the single value representation. For example, if there is a delay in delivery, only the reputation for the delivery attribute will be affected.

Trust aggregation

Trust models often take both direct and indirect evidence into account. The common method introduced when combining these two sources is to assign a weight α and

$(1 - \alpha)$ to direct and indirect evidence, respectively. The value of α can be predefined or continually adjusted.

Predefined α values are adopted in static approaches. The value of α can be 0.5 for a balanced approach, e.g., in S. Liu, Zhang, Miao, Theng and Kot (2014); Weng, Miao and Goh (2006). Jonker and Treur (1999) used values of 0 or 1 for α to exclusively filter out one resource. The empirical results in Barber and Kim (2003) have shown that direct evidence has a long-term effect on the truster agent while indirect evidence helps to build an accurate picture of trustee agents more quickly. However, the precondition is having no biased testimonies.

The static approaches are not efficient in addressing the dynamism of trust. Therefore, researchers have devised some methods to adjust α dynamically. For example, Mui used the number of direct observations available to a truster agent for α. When $\alpha = 0$, it indicates that an agent relies on indirect evidence only. It is when a truster agent has no direct experience with the targeted agent. When the number of direct interactions increases, the value of α increases accordingly. This approach does not attempt to filter out biased rating; instead, when there is enough direct evidence, the evaluation can be estimated without an indirect one. In practice, when the value of α reaches 1, it implies that agent behaviour is fixed. Obviously, this assumption does not hold in all cases and limits the applicability under more dynamic scenarios.

Trying to address the dynamism of MASs, Fullam and Barber (2007) introduced the combination of a predefined value set for α by basing it on experts' opinions to select an appropriate value. A Q-learning-based method is used to choose the suitable α value that may give the highest reward. Nevertheless, because the α value is predefined, the accuracy of the approach depends on experts that form α values.

Evidence Filtering: Even until now, incorrect testimonies are one of the most serious issues for trust evaluation. There is still no effective solution for all problems. Some proposed approaches are often supported by some particular infrastructures in the environment.

Sabater and Sierra (2001a) created the ReGreT model which assesses the credibility of witnesses. ReGreT uses the social relationships combined with the predetermined *fuzzy rules* to estimate the trustworthiness of witnesses. A testimony is then discounted by the trustworthiness of the corresponding witness when aggregating trust. Nevertheless, the relevant social network information must be available in order to make this work.

Whitby, Jøsang and Indulska (2004) proposed a solution that is closely related to BRS. Each rater is assumed to have a cumulative rating vector that is then filtered by lower and upper quantile. As model assumes that the majority of the evidences reflects the correct trustworthiness. However, it suffers from similar weaknesses to BRS when inaccurate information increases. Thus, it is not useful in the environment in which most witnesses are malicious.

Weng et al. (2006) proposed an entropy-based approach which measures the deviation of evidence to its belief before deciding whether to update its current one. The purpose of the approach is to use third-party testimonies to estimate trust when there is not enough direct evidence. However, the model has shown a shortcoming as the prior belief depends on direct experience rather than fully using indirect ones.

S. Liu et al. (2014) introduced a filtering model that considers the majority of past interactions that contribute to agents' reputations. The approach applies local and global clustering over the evidence space, which supports multinomial rating levels. The global clustering is triggered when the local one fails to meet the threshold condition. Similar to Whitby et al. (2004), it is not useful in the environment where the majority of witnesses are malicious.

Fang et al. (2012) considered the variation of ratings as the result of subjectivity. The authors proposed the SARC method which tries to adjust the accuracy of the reputation value by analysing the level of subjectivity of trustee agents.

A common approach for filtering is to use temporal discount factor (Commerce et al., 2002; Su et al., 2013; Wang et al., 2011; S. Liu et al., 2013). This method is to discard outdated ratings/evidence rather than unfair ratings. Wang et al. (2011) proposed a mechanism that allows a truster agent to update the trust (to a trustee) continuously. It considers trust combined with certainty. These two factors can incrementally change when each new piece of evidence arrives. Furthermore, parameters can be tuned dynamically without human interception.

Generally, the temporal filtering approach can help estimate reputation in some circumstances. However, for the systems that the transactions are rare/sparse, this method significantly affects the estimation efficiency by discarding some useful information. In another case, determining the reasonable discount factor is also a challenging question.

Socio-cognitive trust evaluation

Neither socio-cognitive nor organisational methods strictly require direct and indirect evidences. Organisational trust involves hardware, software, protocols, rules to regulate the behaviours. Joining in an environment can guarantee an agent a certain level of trust even without evidences. Meanwhile, socio-cognitive reasoning methods can involve mental states, intrinsic properties, meta-data or social/environmental norms. Thus, those approaches are the complement of evidence-based trust models in situations with insufficient evidences for trust evaluation.

Castelfranchi, Falcone and Pezzulo (2003) used the *fuzzy cognitive maps* concept for a trust decision model. Each truster agent can determine the values of causal links between several factors of their preferences. Ashri et al. (2005) modelled the relationships among agents as market relationships. To identify the relationships, the approach first analyses interactions using an agent-based market model. Only the most relevant relationships are then used for interpreting trust values.

One of the first social network-based trust inference models was the SUNNY model

(Kuter & Golbeck, 2010). Firstly, SUNNY generates a probabilistic trust network through modelling confidence of each node in the social network. The nodes are then filtered by a probabilistic logic sampling to select the input for trust computation.

Burnett, Norman and Sycara (2010) investigated the bootstrapping problem facing evidence-based trust models. The authors assumed that at system bootstrap, there is no experience information available for newcomers. By using intrinsic properties of a trustee agent, the authors claimed to reveal the trustworthiness of targeted agents to some extent. The stereotypes are learned using a decision tree. Trustee agents can be sorted to corresponding stereotype categories. Finally, a stereotypical reputation value is generated.

Noorian, Marsh and Fleming (2011) enriched trust evaluation models by incorporating human dispositions, i.e., optimism, pessimism and realism into the opinion selection process. The intuition is that unfairness can also be caused by such dispositions other than just deception. The approach, therefore, introduces a two-layered filter. The first layer evaluates the competency of neighbours with their experience to be labelled as advisers. The second layer then assesses the credibility of the advisers to identify useful opinion sources.

A *fuzzy logic* based approach was proposed by S. Liu et al. (2013) to avoid unfair ratings caused by deliberate alterations, changing situations and subjectivity. The model takes into account the quantity (for measuring confidence), the temporal factors of the reports, and the similarity between witnesses and the current truster. The three factors are then passed to a fuzzy logic system to assign a weight for each report, and then the reputation value is calculated using the weighted average.

Fang, Guo and Zhang (2015a) integrated four different interpersonal aspects of trust in social science, i.e., benevolence, competence, integrity and predictability, for computing trust. The model is based on users' historical ratings, while impersonal aspects are formulated from the perspective of user connections in trust networks. Two logistic regression models are developed and trained by accommodating these factors, and then applied to predict users' continuous trust values. However, the study depends on the independence of each trust aspects which is difficult to verify.

2.2.3 Contextual information

All trust models need to work within some context. However, not many trust models can entirely make use of contextual information. In this section, three important categories of contextual data will be discussed. They are ienvironmental factor, temporal factor, and interaction targets. These factors are corresponding to "where", "when", and "who" in interaction contexts.

Environmental factors

Environment information has been used in the research to identify the application domain. Different environments affect trust management differently. For example, environments can be open or closed systems and the risks in open systems are usually higher than those in closed ones (Huynh, Jennings & Shadbolt, 2006b). Some researchers placed trust models in more specific environments, e.g., online trading systems (Jøsang & Haller, 2007; X. Liu & Datta, 2012; H. Yu et al., 2014), service-oriented systems (Wang et al., 2011; Zheng, Wang & Orgun, 2013), computer networks (Kamvar et al., 2003), social networks (Sabater & Sierra, 2001b; Kuter & Golbeck, 2010; Nepal, Sherchan & Paris, 2011), and crowdsourcing (H. Yu, Shen, Miao & An, 2013).

Regarding the topologies, two sub-categories are commonly considered, i.e., distributed and centralised. In centralised environments, it is easier to retrieve evidences about users and perform trust evaluation with algorithms that require large datasets. For example, machine learning techniques can be applied in centralised approaches, i.e., Bayesian networks (Regan, Poupart & Cohen, 2006; Jiang & Bai, 2013a). On the other hand, trust management in distributed environments is more challenging. Evidence collection is often less cost-efficient due to several issues, e.g. the sparseness of network topologies. Furthermore, trust evaluation in this environment may require additional steps. For example, Moe, Tavakolifard and Knapskog (2008) used reinforcement learning to obtain more data from curtain environments, while X. Liu, Tredan and Datta (2014) used a combination of local data for training.

Interaction targets

All trust evaluation models need to specify their targets. In current literature, the targets are referred to as trustees. Trustees can be individual agents or agent groups (Gambetta, 2000). However, most research ignores the case of agent groups and considers only individual agents as evaluated targets. Obviously, two types of targets impact trust models differently. For example, the behaviours of agent groups are more dynamic and harder to evaluate (refer to Section 1.3).

The individual agent is the most studied interaction subject in literature. It is considered as the fundamental solution to solve more complicated problems. However, there is another reason to develop a trust model targeting individual agents: the uncertainty about the evaluated target. As mentioned in the example at the beginning of Section 1.3, the lack of information about the evaluation target has led to the adoption of several assumptions. For example, when the trustee is unknown, it is reasonable to consider the trustee as a single individual instead of an agent group. Similarly, when agents' internal activities are unknown, it is fine to assume that the quality of the outcome is influenced by the intention of the trustee.

The immediate drawback of considering all targets as individuals is that it decreases the accuracy of trust models. Not only in terms of intentions, but that there may be internal and external factors which can impact the interaction outcome. In other words, it is impossible for current individual trust evaluation models to discern between the benevolence and the factors influencing the output.

Agent groups are common in agent society. In psychology or economics, there have already been studies conducted that researched trust in groups and related aspects (Hackman & Morris, 1975; Korsgaard, Schweiger & Sapienza, 1995).

Despite many common characteristics between agent groups and groups in human society, trust management for agent groups has not received enough attention for many years (Tong & Zhang, 2009). Agent groups, as the evaluated targets, were mentioned briefly in Marsh (1994). Later, the concept was included in the definition of trust by Gambetta (2000). However, there is no significant study to explore this research area.

Group reputation was introduced by Halberstadt and Mui (2001). However, instead of group trust evaluation, this study used the reputation of groups to discuss the trustworthiness of group members, i.e., individual agents. Similarly, Ezhei and Tork Ladani (2013) proposed the GTrust model which uses the meta-graph for modelling group-based trust. It utilised common features of groups to infer about individual's trustworthiness.

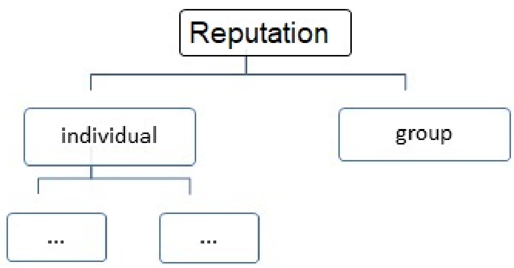


Figure 2.3: Reputation typology of (Mui, 2002)

Mui (2002) introduced group reputation as part of the reputation typology (refer to Figure 2.3). Mui also mentioned a simple group reputation estimation as the average reputation of its members.

The work in Tong and Zhang (2009) is one amongst the first studies of computational trust for a group of agents. However, the approach sets the group reputation value based on the number of agents joining or leaving the group, which is not adequate and lacks theoretical support.

Temporal factors

Time is a crucial factor in trust modelling. It is often used to address the dynamic aspect of trust. Namely, in different timestamps, an agent can have different trust values. However, researchers normally consider trust in two different phrases, i.e., the initial phase and stabilised phase. The initial phase deals with trust management at system startup (bootstrap), or with the newcomers of systems. On the other hand, in a stabilised phase, trust management deals with agents who have been in the system for a period.

The majority of the research work has focused on the stabilised phase. These studies use timestamps to control the age of evidence. For example, a timestamp is used for calculating the discount factor of the evidence weight (Jøsang, 2007; W. T. L. Teacy et al., 2006; Su et al., 2013).

Time is also related to trust-aware decision making. The dynamics of trust often requires agents to choose 'when' to take some action. Most trust-aware systems tend to select agents immediately after the highest trust value is obtained. However, such an approach is trivial compared with real situations. In practice, some events may happen in the middle of the evaluation process, which may affect the accuracy of decision making. One example is the reputation lag problem introduced in Kerr and Cohen (2009). Trust and decision models should take temporal factors into account so that the up to date value is used at the time of taking action.

2.2.4 Trust establishment

Methods to establish trust relationship between agents can be categorised into two main types, i.e., passive trust establishment and active trust establishment. Passive trust establishment methods build trust relationship between agents by predefined rules for agents in the system to follow, e.g., assigning default trust values for newcomers, providing authentication, cryptography, or protocol (Fullam, Park & Barber, 2005; Singh, Juneja & Sharma, 2010). On the other hand, the active trust establishment, defined by Sen (2013), is an establishment method that trustee agents should also invest their effort in to gain trust from other agents rather than to simply follow rules of the system passively.

Malik and Bouguettaya (2009) introduced a trust establishment for newcomers. The approach assigns initial reputation value to newcomers and asks the community to assess the newcomers and update the reputation value. By undertaking this method, agents can quickly receive trust values. This method described is similar to the trust engagement method proposed by Sen (2013). However, in Sen's work, the evaluation of new partners should also be conducted by trustee agents.

Tran, Cohen, Langlois and Kates (2014) proposed one the first establishing models of the active type. The authors introduced a satisfaction criteria and measure. By estimating and providing what a customer needs, a trustee can then actively gain trust. Aref and Tran (2015) then improved this work by introducing satisfaction demand and index to increase average trust levels.

2.2.5 Trust use

Trust is normally used for risk management. For example, agents have something to lose when the interaction fails. Risk can be quantised to potential loss before interacting with another agent, especially in situations where task delegation is necessary (Aljazzaf, Perry & Capretz, 2010). Falcone and Castelfranchi (2001) addressed this fact regarding the risks of failure, frustration, and effort lost when an agent trusts another, especially in delegated tasks.

The quantity of gain/loss is often measured by using utility functions. Ramchurn and Jennings (2004) used probabilities to estimate risk through functions that estimate the agent utility loss. If the function returns a high value, the risk of interaction is also high. Similarly, Wang et al. (2008) used utility functions based on agents' attitudes toward risks. This function returns the uncertainty degree about the interaction, which can then be used for decision making, i.e., trust-aware decision making (Burnett, Norman & Sycara, 2011; H. Yu et al., 2014; H. Yu, Miao et al., 2013).

It is worth to mention that trust-aware decision models depend on the robustness of trust models. Many decision-making studies have come back to modify trust models to fit their application contexts. For example, the "accept-when-requested" is an assumption adopted when modelling agents interaction. However, this assumption is not practical in real-world situations where agents have limited capabilities. Using the assumption in decision making can affect their reputation seriously (H. Yu, Shen, Leung et al., 2013). Therefore, practical factors should be included into a trust-aware decision model to coordinate the actions of trustee agents (H. Yu, Miao et al., 2013).

2.3 Propagation techniques and model evaluation

There are different techniques to enable truster agents to estimate the trustworthiness of targeted agents. For processing trust information, we have four main approaches, i.e., subjective probability, machine learning based, fuzzy logic, and socio-cognitive (Table 2.2). Each technique has its advantages and disadvantages.

The earliest methods of trust computation are based on probability. The rationale is that interaction risk can often be interpreted as the probability of being deceived by the targeted partner. Meanwhile, the probability can be calculated by using the outcome records of past interactions. These methods normally assume that the majority will reflect the true trust value. The advantage of using probability is that it is suitable for modelling the subjective aspect of trust. Therefore, transforming from definition to computation is more intuitive. Nevertheless, probability approaches require sufficient direct evidence to predict the trust value accurately. Furthermore, they are not very effective in addressing the trust dynamics and are, therefore, vulnerable to unfair ratings (Commerce et al., 2002; Jøsang & Haller, 2007; W. T. L. Teacy et al., 2006).

Machine learning methods can be powerful for improving the accuracy of the trust evaluation. For example, the Bayesian network can utilise the hierarchical relation between third party testimonies and trust value (Fang et al., 2012; Kuter & Golbeck, 2010). The disadvantages of using machine learning approaches are the increasing complexity and the requirement of the huge amount of high-quality data for training.

Fuzzy logic is another tool used in trust reasoning. Researchers, who use socio-cognitive approaches, often find it useful. Sabater and Sierra (2001a) introduced the ReGreT model, which uses the social relationship information among members under the assumption of the availability of relevant social network information. Castelfranchi et al. (2003) proposed a trust model using a fuzzy cognitive map. The internal characteristics of a trustee agent and external factors from the environment can affect the performance of the trustee agent. The causal links between factors and preferences must be presented in order to construct the maps. Nevertheless, it is not easy to make the information available and to verify these links in many systems.

Trust Model Evaluations

The assessment of trust models can be based on simulations, real-world data, or the combination of both. Each method has its pros and cons.

Simulation methods used in different studies are diverse. In order to standardise the assessments using simulations, Fullam, Klos et al. (2005) proposed the Agent Reputation and Trust Testbed, which provides an environment with typical activities for agents. However, the testbed only covers parts of the big picture of trust computation.

Subjective probability-based	Machine learning based	Fuzzy logic based	Socio-cognitive
(Commerce et al., 2002)	(W. T. L. Teacy et al., 2006)	(Castelfranchi et al., 2003)	(Castelfranchi et al., 2003; Castelfranchi & Falcone, 2010)
(Jøsang & Haller, 2007)	(Huynh et al., 2006a)	(S. Liu et al., 2013)	(Noorian et al., 2011)
(Griffiths, 2005)	(Huynh et al., 2006a)		(Kuter & Golbeck, 2010)
(Su et al., 2013)	(Fang, Zhang & Magnenat Thalmann, 2014)		(Burnett et al., 2010)
(Wang & Singh, 2007)	(Rettinger, Nickles & Tresp, 2011)		(Fang et al., 2012, 2014)

Table 2.2: Computational techniques used in trust models

Consequently, researchers still prefer creating flexible environments that fit their specific requirements. So far, the survey of trust research in the past decade shows that most researchers prefer using simulations for experiments and evaluation (H. Yu, Shen, Leung et al., 2013).

Another evaluation method is to use real-world data collected from existing applications. Currently, three popular real datasets used in trust model assessment are Epinions (Epinions, n.d.), Advogato (*Advogato*, n.d.), and FilmTrust (Guo, Zhang & Yorke-Smith, 2013). Established in 1999, Epinions.com was a site for general consumer review and it was acquired by eBay in 2005. In Epinions, visitors could read old and new reviews about a variety of items to help them decide on a purchase. While Epinions' reputation system is not abuse-proof and contains no graph information (relation), Advogato's goal is to be attack-resistant; however, the effectiveness of the algorithms is controversial [3]. The computation of the trust metric is performed relative to a "seed" of trusted accounts and links are expanded from these seeds.

Each dataset is suitable for different evaluation purpose. For example, Kuter and

[3]http://www.advogato.org/article/928.html

Golbeck (2010) used the FilmTrust dataset for the SUNNY model because the dataset contains social network relationships. The Epinions dataset by Massa and Avesani (2007) was used by L. Li and Wang (2010); Massa and Avesani (2007) to test the model performance regarding bootstrapping because the dataset contains indirect trust information (recommendation). Rating data of eBay is used in (Rettinger et al., 2011) for statistical analysis.

Undoubtedly, real-world datasets are the complement of simulation data in assessments. They could bring the insight to researchers of the performance of proposed models under realistic conditions. However, the shortcoming of using real-world datasets is the lack of flexibility for testing dynamic behaviours of agents and the environment. The datasets often contain no ground truth and behavioural information, e.g., rating data from eBay or Amazon. Namely, there is no efficient way to evaluate a trust value generated by a trust model and the true reliability of a trustee agent. The lack of accurate behavioural and intentional data restricts comprehensive analysis of the trust models.

Ideally, there should have been a dataset containing essential information for trust study, e.g., preferences and behaviours, so that researchers could both test and verify the trust model. Unfortunately, trust is also a sensitive research area. Obtaining such information from available industries is almost impossible.

2.4 Summary

This chapter provided a detailed review of more than 120 papers in the field of trust management in MASs. The studies are sorted into different categories based on trust information sources, contextual information and propagation techniques used in each study. By summarising the pros and cons of each study, this chapter has depicted a comprehensive picture of trust management in MASs, which is helpful to identify research gaps and directions. The findings from the literature review are summarised as follows.

1. In the four major trust research areas, trust evaluation has been investigated extensively while trust establishment and trust engagement would need more attention.

2. There are several trust information sources. Direct trust and indirect trust sources are used in most models. However, there is not a convincing method to combine different information sources to produce a single trust value.

3. Most trust evaluation models and their applications have focused on individual agents whilst ignoring agent groups. Trust research should also cover different aspects of trust in complex agent groups, as there are increasingly important applications assisted by these groups.

4. There are four main techniques used for processing trust information. However, contextual data can be better utilised for building robust trust management models.

5. Using simulation for evaluations is preferable in research community due to the flexibility in creating testing environments.

The literature review has pointed out several research gaps of trust management in MASs. It also confirms the need for trust research for complex agent groups. As introduced in Chapter 1, this thesis aims to address the shortcomings of current trust models when dealing with different trust-related issues from agent groups. The solution presented in this thesis is a trust management stack. It not only involves different stages of agent groups' life-cycles but it also takes into account several trust dimensions, e.g., trust establishment, trust evaluation, and trust use. The realisation of three components of the stack is presented in the following chapters.

Chapter 3

CoTrust: a cooperative trust establishment mechanism

This chapter presents the realisation of CoTrust, the first component of the trust management stack introduced in Chapter 1. It focuses on the roles of trust and relevant mechanisms in the group formation stage. When forming a group, agents must go through a partner selection process. Although current trust evaluation models can evaluate the trustworthiness of potential partners, simply sending a request to the partners may not result in successful cooperation. Unfortunately, there is not an effective trust mechanism dedicated to assist agents in this process. Therefore, the CoTrust mechanism proposes to assist group formation by improving the success rate of requests and satisfaction of cooperation.

3.1 Overview

Why cooperation? Many tasks cannot be completed by a single agent, but they can be solved effectively by multiple agents. Therefore, agent groups are preferable. However, gathering agents to work together as a team is not an easy process, especially for self-interest intelligent agents.

For example, there could be a situation where agent A receives a request req_i from agent B to solve a task. A is aware that he cannot solve req_i alone. Therefore, A decides to ask other agents to work alongside him in order to solve the request. Suppose that C is an agent that A considers to invite, A must estimate the performance of C when they cooperate. As a rational agent, C has some expectation over the cooperation. Therefore, when C receives an invitation to join A's group, C may not consider A as a good choice. Namely, there would be no cooperation between them.

In this study, the terms "group formation" and "agent cooperation" are used interchangeably. These two terms both mean that agents work together as a group to solve tasks. Furthermore, we refer to the situation where both A and C evaluated each other before deciding to cooperate as a two-sided trust evaluation rather than a one-sided trust evaluation as adopted in most current trust models.

A two-sided trust evaluation is important for the success of cooperation because risk can arise from untrustworthy behaviours of both the truster or trustee agents. For example, online auctions may fail due to the bad behaviours of bidders who refuse to pay after winning. Distributed denial of service (DDOS) attacks succeed due to the failure to identify requests from zombie agents.

The previous examples have suggested that truster agents also need to be evaluated for their credibility. Classic one-sided trust evaluations are inadequate for maintaining the effectiveness of reputation systems in the long term. Therefore, having a proper trust mechanism, that can also regulate the behaviours of both truster and trustee agents, is crucial for agent cooperation. Unfortunately, there is not an existing trust mechanism to address this gap. To this end, I propose the CoTrust (Cooperative Trust) mechanism. It consists of a protocol and a reasoning method for agents to select and establish trust relationships with potential partners. The proposed trust establishment method adopts a stable matching game which can improve the satisfaction of the cooperation. Also, it can give an incentive to all agents to behave honestly.

The remainder of this chapter is structured as follows. Section 3.2 reviews some trust evaluation models and trust establishment methods regarding agent cooperation. Section 3.3 presents the CoTrust mechanism. An empirical analysis of our method is presented in Section 3.5. Finally, the chapter is summarised in Section 3.6.

3.2 Related Work

Group formations (group composition) have long been studied in MASs. The book written by Dunin-Keplicz and Verbrugge (2010) provides a comprehensive review and suggests formal approaches for teamwork in MASs. It states that "*the existence of joint motivation attitudes is a necessary condition for a loosely coupled group of agents to become a strictly cooperative team*". Furthermore, agents need to trust each other during group formation.

The condition has exposed a major shortcoming of existing trust models regarding agent cooperation as they were originally designed to perform one-sided trust evaluations. These studies often came with a simplifying assumption, i.e., "accept when requested". Namely, if agent c_i trusts agent p_j, there will be a transaction/cooperation between the two agents. This assumption can also cause the reputation damage problem described in H. Yu, Miao et al. (2013).

Current trust mechanisms provide incentives for trustee agents to behave honestly, but not for truster agents. In reality, truster agents can also act maliciously, e.g., flooding fake requests or denying to complete the transaction. Nevertheless, such misbehaviours of truster agents have never been considered directly by any trust model. Taking lying agents in the TRAVOS model (W. T. L. Teacy et al., 2006) as an example, the agents can be truster agents, and they may give wrong feedback after interactions. Instead of marking the agents who give false feedback as malicious agents, the model only learns to identify biased ratings. Namely, the malicious agents still have chances to spread wrong information. As a result, the approach performs poorly when the percentage of unfair raters increase. In this situation, without a strategic trust establishment method, trustee agents may fail to interact with potential trusters.

The work on trust establishment proposed in Tran et al. (2014) and Aref and Tran (2015) helps trustee agent to gain trust from truster agent by measuring the satisfaction criteria of truster agents. However, there is no warranty for trustee agents' satisfaction in the transaction. Therefore, it is not suitable for establishing trust in group formation.

To our best knowledge, there is not a current trust model which supports trust establishment in the context of agent groups. CoTrust includes a protocol and a reasoning method that assists with trust establishment and help to improve the satisfaction of agents involved in the cooperation.

3.3 The Cooperative Trust Establishment Model

The architecture described by Sen (2013) suggests that each agent should be embedded with a trust management module. This module consists of several sub-components working together. The subcomponents of the trust management module, in compliance with Sen (2013) are in charge of trust evaluation, trust establishment, trust engagement, and trust use. The term trust establishment is different from passive trust establishment introduced by Singh et al. (2010). Sen defined trust establishment as the "flip side of the evaluation", which focuses on how trustees can gain trust from truster agents actively.

The work of this chapter acts as a mechanism for supporting agent group formation (agent cooperation). It contributes to active trust establishment and trust use. Some trust evaluation methods involved in the trust establishment process. However, this work does not introduce a new trust evaluation method. Although a group can be formed by the cooperation of two single agents or between an agent and another group, this chapter only investigates the first case, i.e., agent-agent cooperation.

3.3.1 The matching game

In order to establish trust in the process of agent cooperation, we borrowed some knowledge from stable matching games in Game theory. The matching theory has become a leading area in economic theory with a mathematical framework. It attempts to describe the formation of mutually beneficial relationships over time. The stable marriage problem is a well-known matching game which is stated as follows:

A stable marriage I consists of the same number of n men and n women. Each person has a sorted list of all members of the opposite sex ordered by their preference. We denote $w_i \succ_{m_k} w_j$ when a man m_k prefers w_i to w_j ($1 \leq i, j, k \leq n$). A similar notation is used for women's preferences.

A matching M is a set of man-woman pairs in I. When there are n matched of disjoint pairs, we have a perfect matching. We denote $M(m_k) = w_i$ and $M(w_i) = m_k$ for a matching M of a man m_k and a woman w_i.

A matching is stable when after matching, there is not a pair of a man m_k and a woman w_i who both prefer each other to their current partner. Formally, it can be defined using the term "blocking pair". A pair of a man m_k and a woman w_i is a blocking pair if the matching comprises of the following three conditions: (1) $M(m_k) \neq w_i$; (2) $m_k \succ_{w_i} M(w_i)$; (3) $w_i \succ_{m_k} M(m_k)$. A matching is stable if there is no blocking pair, and *unstable* otherwise.

The solution for stable matching often results in the improvement of satisfaction during the interaction (Irving, Leather & Gusfield, 1987). Algorithm 1 is a matching solution proposed by Gale and Shapley (1962). It involves a number of "rounds".

There are two influential factors in the stable matching of CoTrust, (1) the evaluation of both sides and (2) the rejection of proposals. CoTrust adopts the game as part of

Algorithm 1 Gale-Shapley stable matching

Free all $m \in M$and $w \in W$
while exists free man m_k who has a woman w_i in the list **do**
  w_i = is the woman whom m_k has not proposed
  if w_i is free **then**
    $M(m_k) = w_i$ and $M(w_i) = m_k$
  else if $M(w_i) = m_l$
    if $m_k \succ_{w_i} m_l$ **then**
      set m_l free
      $M(m_k) = w_i$ and $M(w_i) = m_k$
    else
      No new match is made
    End if
  End if
End while

the communication protocol. However, in our study, the number of request-makers (requesters) and responders are generally not equal. Therefore, it is not a perfect matching game but another variant of the classic stable matching game.

3.3.2 Definitions and assumptions

CoTrust considers *"trust each other"* as the key factor of cooperation. It means that both the truster and trustee agents need to trust their interacting partners in the course of the cooperation. It is assumed that agents have preferences regarding cooperation outcomes. Agents make decisions based on utility values governed by the preferences. They evaluate the trustworthiness of partners using available evidence, i.e., feedback of the past transactions. CoTrust does not follow prevailing *accept-when-request* assumption. It can be distinguished from the DRAFT approach (H. Yu, Miao et al., 2013) in that trustee agents in CoTrust reject requests based on a trust-aware utility gain rather than an assessment of their own limitation.

For the rest of this chapter, we omit the terms *"truster"* and *"trustee"* as both need to be trusted by others in order to continue a transaction. Instead, the terms *"requester"* and *"responder"* are used accordingly. They are both abstract concepts and can be used in a wide range of contexts. On the one hand, they could be a "buyer-seller", service "consumer - provider". On the other hand, they could be two arbitrary agents who are going to work together, etc.

Figure 3.1 illustrates the conceptual architecture of an agent in CoTrust. Each agent (e.g., a_i) has the following five components:

1. *The communication module* is in charge of communicating and exchanging information with other agents and the surrounding environments.

2. *The profile* is a database containing information of a_i which is public to all agents in the system. The information includes a_i's identity and transaction records in D_{a_i} (see Definition 3.3.3) that can be used as evidence for other agents to reason about the performance of a_i.

3. *The private knowledge base* is a database containing private information which can be accessed only by a_i. The private information includes belief, intentions, and preferences.

4. *The learning module* uses information from the above three components to learn and reason about the trustworthiness of interacting partners.

5. *The decision module* combines the information from the above four components to make a decision on which strategies and actions are to be taken.

Every cooperation starts with an initiation. We call the agent, who initiates a request for cooperation, a **requester**. A requester is denoted as (c_j). The types of request vary to the cooperation contexts.

A **responder** (p_i) is an agent who receives a proposal from requesters and responds with an accept or deny message. Before accepting a request, a responder also evaluates all of the requesters. Similar to requesters, responders have their preferences which influence their behaviours.

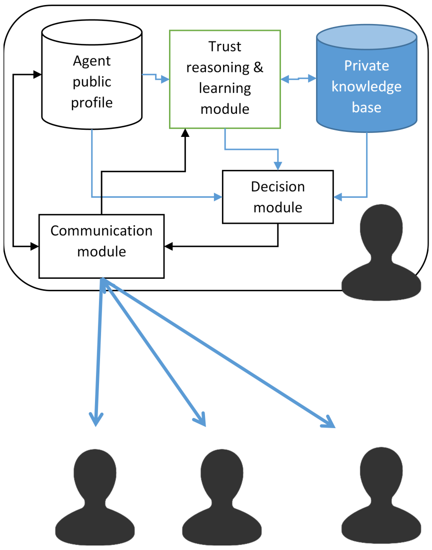


Figure 3.1: The architecture of an individual agent

A message $msg_{a_i}^{a_j} = \{msg_t, msg_c\}$ is the information sent from a_i to a_j regarding cooperation, where msg_t and msg_c is the type and the content of the message.

There are two types of messages, i.e., request and response. The request message contains a proposal for the cooperation, which includes requirements and offers (potential benefits). We term the requirements as cooperation attributes. We denote $At_{c_i p_j} = \{at_1, at_2, \ldots, at_n\}$ for the set of attributes of the cooperation between a_i and a_j. For example, the requester, who requests for a delivery service, can prefer in fast delivery (short time) or the cost of the service. Therefore, time and cost can be included as attributes of the proposal. The content of a response message can be as simple as to accept or reject the request.

Definition 3.3.1. A **cooperation preference** (or just preference for short) is the private information of an agent regarding a situation of cooperation. The preference of a_i regarding cooperation with a_j is a vector of weights over different attributes $Pr_{a_i}^{a_j} = (\omega_1, \omega_2, \ldots, \omega_n)$, where $\omega_i (1 \leq i \leq n)$ is the weight of i^{th} attribute of a cooperation.

The preference is an important part of an agent's architecture. It influences the behaviours of agents on selecting partners and also providing feedback. Therefore, knowing the preference of partners can help to improve the quality of cooperation. However, to avoid being exploited by malicious agents, most trust models consider preferences as private information. The preference reasoning method would require learning ability. ω_i values between [0, 1]. 0 and 1 represent the least and the most preferred attribute. For example, if we have $Pr_{c_i}^{p_j} = (0.3, 0.8, 0.6)$ represents the weights of cost, delivery, and quality, it means that c_i cares about the delivery the most. How to use preferences for selecting partners will be discussed in Section 3.4.

Definition 3.3.2. Feedback $R_{a_i}^{a_j}$ represents an assessment of agent a_i over the experience given by agent or agent group a_j.

The most common type of feedback is rating. The experience can be a quality of a product or a service that the agent received. In the context of agent groups, it represents the satisfaction of cooperation. In this study, we also consider R as a vector of real values in $[0, 1]$, i.e., $R_{a_i}^{a_j} = (r_1, r_2, \dots, r_n)$, where n is the number of attributes defined in the proposal. 0 and 1 represent the worst and the best satisfaction of the experience. All feedback for a_i is received and stored in D_{a_i}, which is a local database.

Definition 3.3.3. The ***transaction history*** D_{a_i} is a set of past transactions for agent a_i, i.e. $D_{a_i} = \{\tau_{a_{i1}}, \tau_{a_{i2}}, \dots\}$. D_{a_i} is visible and verifiable to other members of the system, and it is maintained by the agent itself.

Each transactional record $\tau_{a_{ik}}$ is potential evidence ($k \geq 1$), which can be used for supporting a_i in trust evaluations. $\tau_{a_{ik}}$ contains information of a transaction with a_i, i.e., who was involved in the transaction, timestamp, transaction feedback. We also assume that records in D_{a_i} are verifiable. This assumption enables the model to work in both centralised and distributed environments in terms of evidence collecting.

Definition 3.3.4. The **reputation** of agent a_i is a vector of real values indicating the performance of agent a_i over the set of n attributes, i.e., $Rep_{a_i} = (rep_1, rep_2, \dots, rep_n)$ where $rep_i \in [0, 1]$ and $(1 \leq i \leq n)$.

In this study, we assume that there is already a reputation model provided by the system. The reputation value of agent a_i is calculated by using the reputation model and evidence in D_{a_i}. The reputation information of an agent can be obtained via their public profile (refer to Figure 3.1).

Definition 3.3.5. A **cooperation strategy** of agent a_i regarding a cooperation with agent a_j is a set of possible options that a_i choose to improve the success rate of the cooperation. It is denoted as $St_{a_i}^{a_j} = \{st_{a_i}^{a_j|at_k}\}$ where $st_{a_i}^{a_j|at_k}$ is the strategy of a_i to a_j regarding attribute at_k $(1 \leq k \leq n)$.

As agents have limited capabilities, it is important for them to have suitable strategies that maximise their utility gain. For example, if b is the budget of c_i for a cooperation with two attributes $At_{c_ip_j} = \{at_1, at_2\}$, one possible strategy can be $st_{a_i}^{a_j|at_1} = (b, 0)$ which c_i allocates all the budget for attribute at_1. For simplicity, this study only consider a fix set of strategy for each attribute.

Next, we introduce the protocol which supports message exchange and trust establishment.

3.3.3 The cooperation protocol

Figure 3.2 illustrates the procedures to enable CoTrust in which both the consumer and provider need to evaluate each other before the transaction. The providers and truster agents must comply with the protocol described below.

1. Requester c_i queries and evaluates available partners for cooperation.

2. Requester c_i send a proposal to each potential partner. Each proposal can be different for each partner. The requester's reasoning is described in Subsection 3.4.1.

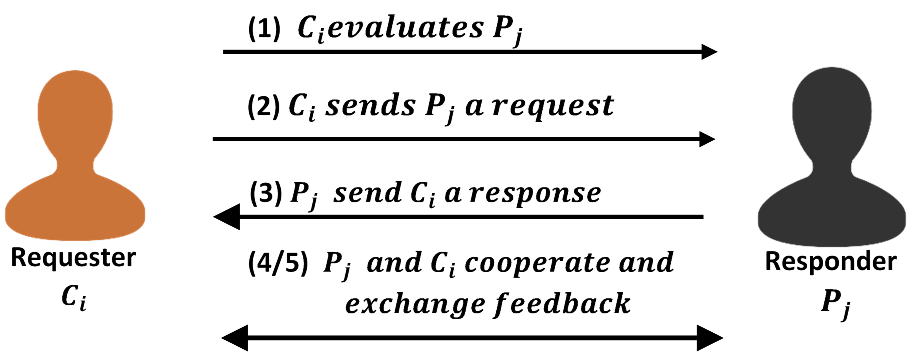


Figure 3.2: The CoTrust protocol

3. After receiving a request, responder p_j evaluates the trustworthiness of requester c_i based on evidence stored in D_{c_i}. The responder can accept or deny the request from c_i. The responder's reasoning is described in Subsection 3.4.3.

4. After receiving all responses, requester c_i chooses the most preferred partner and sends another response message to confirm the cooperation.

5. After the cooperation is completed, both partners give feedback of the cooperation and store it in the database.

In each transaction round, a requester can send requests to many responders with different proposals. The number of responders and requesters in the system can be unequal. Thus, the problem can be modelled as an imperfect stable matching game (Gusfield & Irving, 1989).

The act of partners giving feedback about the other is not new, and can be found in many e-commerce platforms, such as eBay and Amazon. Nevertheless, the use of sellers' feedback is unclear in practice. Furthermore, most studies of trust in MASs use "accept-when-requested" interaction methods, which again leave out the sellers' feedback. To fully utilise the available information, the feedback of partners will be used in two evaluation phases of CoTrust.

One of the persistent issues of feedback is inaccuracy. Namely, having more feedback does not coincide with gaining more certainty (Wang et al., 2011). For example, A gives positive feedback to B, but B may give negative feedback to A. The situation is common in MASs. However, in terms of cooperation, the cooperation between A and B can be considered unsuccessful. In cooperative agent groups, the outcome of the cooperation should be agreed by both parties. Namely, the feedback of both should be based on the comparison between the actual outcome and the proposal. By doing so, CoTrust can prevent some conflict of feedback.

3.4 Partner selection mechanism

We will now introduce partner selection mechanisms for requesters and responders for cooperation. The mechanism involves the trust reasoning and learning modules (refer to Figure 3.1).

A requester intends to cooperate with the most preferred partner. To achieve that, the requester should appear to be a good choice for the responder. Requesters can improve their chance of being accepted by learning about preferences of targeted partners. As preference and behaviour have a strong relationship, in CoTrust, preferences are learned via investigating feedback. Requesters and responders have different reasoning methods, which are discussed below.

3.4.1 Requesters' reasoning

Before sending a proposal to p_j, c_i need to evaluate the trustworthiness of p_j. In this stage, c_i wants to know whether agent p_j is capable to complete a defined task. The trust value is generated by a trust evaluation model in the system.

To have a suitable partner, a requester must find responders whose reputations meet the requester's preferences. As the reputation values and preference attributes are in the form of vectors, it is convenient for requesters to filter out potential partners by calculating the cosine similarity between its preference and the targeted partner's reputation. The similarity can be calculated using Equation 3.1.

$$sim(Pr_{c_i}^{p_j}, Rep_{p_j}) = \frac{Pr_{c_i} \cdot Rep_{p_j}}{|Pr_{c_i}| \cdot |Rep_{p_j}|} \tag{3.1}$$

In Equation 3.1, $0 \le sim(Pr_{c_i}, Rep_{p_j}) \le 1$. p_j is qualified if $sim(Pr_{c_i}, Rep_{p_j}) \ge \theta_{c_i}$, where θ_{c_i} is the threshold set by c_i.

As depicted in Figure 3.3, the $Pr_{c_i}^{p_j}$ and Rep_{p_j} are close if the angle between the two vectors are small. If the cosine similarity is greater than a predefined threshold θ_{c_i}, then the agent will be accepted. The shaded area indicates all the agents that have similarity values greater than θ_{c_i}. The requester only needs to send proposals to qualified responders. For example, if $Pr_{c_i} = (0.3, 0.8, 0.6)$, $Rep_{p_i} = (0.8, 0.4, 0.5)$), $Rep_{p_k} = (0.4, 0.7, 0.5)$ then p_k will be selected.

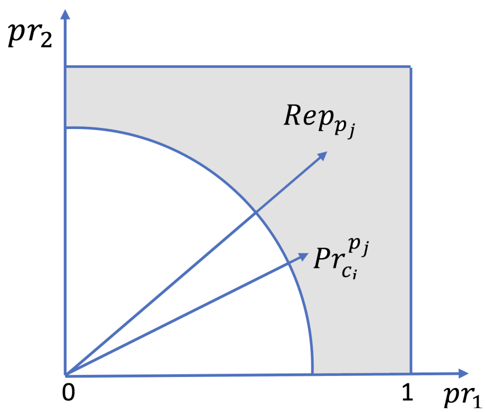


Figure 3.3: Partner acceptance space

Using cosine similarity for selecting partners can be better than using a utility value, which is calculated by the ratio of reputation and cost (W. T. L. Teacy et al., 2005). Therefore, cosine similarity can be useful to distinguish two candidates with the same utility value calculated by the ratio of reputation and cost.

Let $O_{c_i}^{p_j}$ be the cost that c_i invest to the cooperation with p_j and $G_{c_i}^{p_j}$ be the gain if the cooperation is successful. Equation 3.2 shows the utility function of requester c_i regarding cooperation with p_j.

$$U_{c_i}^{p_j} = \frac{G_{c_i}^{p_j}}{O_{c_i}^{p_j}} \cdot sim(Pr_{c_i}^{p_j}, Rep_{p_j}) \tag{3.2}$$

The utility value of the requester is proportional to the gain and the suitability of the targeted partner. Likewise, the value is inversely proportional to the cost of investment.

3.4.2 Estimate Responder Preferences

Suppose that p_j has passed c_i's evaluation, the next task for c_i is to select a suitable strategy when propositioning p_j. In order to increase the chance of being accepted by p_j, it is necessary for c_i to learn about p_j's preferences.

Each requester can send proposals to many partners in one transaction round but can cooperate (engage) with only one partner. Requesters can discover a responder's preferences through analysing feedback records of the responder to his previous partners.

In CoTrust, we assume that agents' preferences influence their rating behaviours. Namely, instead of checking whether a rating is accurate, we consider all feedback is correct regarding the preference. A benefit of doing so is that biased ratings and behaviours are not treated separately. For example, if agent p_j rates the attribute at_m lower than the majority ratings, p_j is considered to have a higher expectation for attribute at_m compared with other agents. From that, requester c_i can reason with p_j's preferences (weight distribution) through comparing p_j's ratings with the average rating of his previous partners on the same aspect. Their preference weight is measured by the ratio between ratings of attribute at_m of responder p_i with the ratio of the reputation of providers in transaction records of the same aspect.

$$\omega_{p_j,c_k}^{m} = \frac{rep_{p_j}^{m}}{r_m} \tag{3.3}$$

Equation 3.3 shows how to calculate the preference weight $\omega^m_{p_j,c_k}$ that a responder p_i placed in attribute at_m. Note that, $\omega^m_{p_j,c_k}$ has a positive real value, i.e., $\omega^m_{p_j,c_k} \geq 0$. When $\omega^m_{p_j,c_k} \geq 1$, the rating of p_j is smaller than the reputation value of attribute at_m of c_k. Namely, p_j has a higher expectation for this attribute in comparison with the average expectation. If the consumer has no transaction record, $\omega^m_{p_j,c_k}$ is set to 1. We perform the analysis over the transaction records of p_j to get the mean weight $\overline{\omega}^m_{p_j}$ of responder p_j for each attribute:

$$\overline{\omega}^m_{p_j} = \frac{\sum_{c_k \in D_{p_j}} \omega^{a_m}_{p_j,c_k}}{h} \tag{3.4}$$

In Equation 3.4, h is the number of cooperation records in D_{p_j}. $\overline{\omega}^m_{p_i} \geq 1$, which indicates that p_i has lower expectation for attribute at_m and normally rate at_m higher than the average agents in the previous transactions. Regarding the reputation gain, requesters can benefit from cooperating with responders of this type. Likewise, when p_i has a high expectation for attribute at_m, responders may risk their reputation. To comply with Definition 3.3.1, the value of $\overline{\omega}$ is then normalized to (0, 1) by using Equation 3.5.

$$\widehat{\omega}^m_{p_i} = \frac{\overline{\omega}^m_{p_i}}{\max\{\overline{\omega}^m_{p_i} | at_m \in Atc_i p_j\}} \tag{3.5}$$

By using Equation 3.5, requesters can obtain useful information about responders' interested attributes. If p_j prefers at_k, then the strategy $st^{p_j|at_k}_{c_i}$ will be used to propose p_j.

A good strategy can also improve the satisfaction of the cooperation, which is reflected by feedback. The expected rating from p_j to an attribute at_k under strategy $st^{p_j|at_k}_{c_i}$ is estimated by using Equation 3.6.

$$r^{at_k}_{p_j,c_i} = \widehat{\omega}^k_{p_i} \cdot \overline{rep}^{at_k|st^{at_k}_{c_i}}_{c_i} \tag{3.6}$$

, where $\overline{rep}_{c_i}^{at_k|st_{c_i}^{at_k}} = \frac{\sum_{l=1}^{h} R_{p_l|at_m}^{at_j}}{h}$ is the mean ratings of attribute at_k calculated with all the transactions using strategy $st_{c_i}^{at_k}$.

To improve the effectiveness of proposals, a reinforcement learning approach is applied as shown in Algorithm 2. This algorithm uses the results of the proposals (response messages) for the training preference estimation. Namely, if c_i's proposal is accepted by p_j, c_j can assumed that he has passed p_j's evaluation (refer to Subsection 3.4.3). Likewise, strategy $st_{c_i}^{p_j|at_k}$ works for p_j. Otherwise, c_i must consider improving its current reputation status or changing strategy in order to be accepted in the future.

Algorithm 2 Requester selection and trust establishment

Require: predefined utility threshold $\theta_{c_i}^{p_j}$, a set of strategies St_{c_i}, a set of responders P.

for each $p_j \in P$ **do**
  if p_i exists in D_{c_i} and satisfied with $st_{c_i}^{p_j|at_k}$ **then**
    propose p_i with previous strategy $st_{c_i}^{p_j|at_k}$.
  else
    Estimate p_i's prefences using Equation 3.5
    Assign new strategy $st_{c_i}^{p_j|at_l}$
    Estimate $U_{c_i}^{p_j}$ using Equation 3.2
    if $U_{c_i}^{p_j} \geq \theta_{c_i}^{p_j}$ **then**
      propose p_j with $st_{c_i}^{p_j|at_k}$
    else
      Ignore p_j
    end if
  end if
  $msg_{p_j}^{c_i} \leftarrow$ response from p_i
  if p_j accepts the proposals **then**
    return
  else
    change strategy for p_i
  end if
end for
if p_k accepts the propsal and $U_{c_i}^{p_k} = \max\{U_{c_i}^{p_*}\}$ **then**
  cooperate with p_k and reject other responder
end if

This algorithm can achieve the following:

- In one transaction round, there is an equal number of requesters and responders cooperating. This is easy to verify as one requester cooperates with one responder. However, there can be a requester or a responder who is not being cooperated with (free).

- The cooperations are stable. Let c_i and p_j both be engaged, but not to each other. Upon completion of the algorithm, it is not possible for both c_i and p_j to prefer each other over their current partners. If c_i prefers p_j to c_i's current partner, c_i must have proposed to p_j before c_i proposed to its current partner.

3.4.3 Responder's reasoning

For responders, there are two tasks to consider when receiving a proposal, i.e. evaluating the proposal and the trustworthiness of the requester. Evaluating the proposal is to verify whether it is suitable for the responder's preferences. Meanwhile, evaluating the trustworthiness of the requester is to know whether the proposal will be done as proposed. Unlike requesters, responders do not have to reason about the preference of requesters.

CoTrust constructs the utility function from these two evaluation outputs. If an attribute in the proposal meets the requirement of the responder, a reward will be added to the utility value. Therefore, responder p_i's utility function regarding a proposal of c_j is calculated by Equation 3.7.

$$U_{p_j}^{c_i} = \sum_{k=1}^{n} R_{c_i}^{at_k} \cdot U_{p_j}^{at_k} \tag{3.7}$$

In Equation 3.7, $U_{p_j}^{c_i}$ is a reward, which is defined according to p_j's preference, if the proposed attribute meets the expectation of p_j. The reward is then adjusted by the current reputation of requester c_i of the corresponding attribute. By doing so, it ensures that the proposal should agree with the reputation to some degree. For example, if the proposal of c_i assigns 1 to attribute at_k but the reputation of c_i for at_k is 0 then the gain for p_j regarding the attribute is 0.

3.5 Experiments

Several experiments have been conducted to evaluate the effectiveness of CoTrust. The objectives of the experiments were to confirm that (1) CoTrust can help requesters improve the success rate of proposals, and that (2) CoTrust can improve the satisfaction of cooperation.

For simplifying purposes, there is one type of cooperation for all agents. Each cooperation has two attributes, i.e., reward (at_1) and rating (at_2). There are two corresponding types of agents, i.e., a benefit sensitive BSA and a reputation sensitive agent (RSA). The $RSAs$ are agents who place the reputation as the highest priority and may sacrifice their benefit (reward amount) in a transaction. For this type of partner, the result of cooperation regarding ratings is more important. By contrast, $BSAs$ consider benefit as the highest priority. The preferences for two attribute of c_i are $\omega_{c_i}^{at_1}$ and $\omega_{c_i}^{at_2}$. If c_i is reputation sensitive, the value $\omega_{c_i}^{at_2}$ will be larger than $\omega_{c_i}^{at_1}$.

The system was simulated with 100 agents of 50 requesters and 50 responders. Agents were profiled with performance and cost which can be either of high (H), normal (N), or low (L) values. Accordingly, there are three values for the expectation that influence agent's rating behaviours, i.e., high expectation (HE), normal expectation (NE) and low expectation (LE). The higher expectation for an attribute may cause the lower rating for that attribute if it is not satisfied.

A random number of requests for cooperation was generated each round. The simulation stopped when it reached 3000 transactions. The obtained results were compared with the common approach with the $accept-when-requested$ (AWR)

Table 3.1: Requester acceptance rate in stabilised stage

	HE	NE	LE
RS	38.2%	68.8%	71.4%
BS	91.8%	95.2%	96.6%

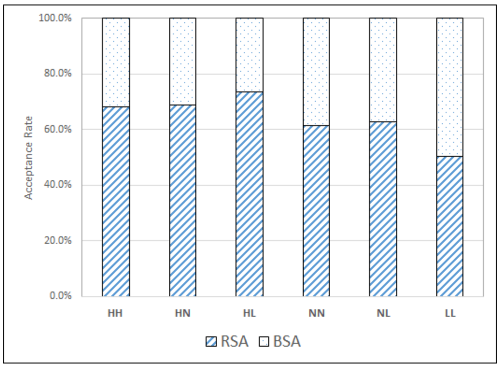
assumption regarding the success rate of proposals and satisfaction. AWR is a common interaction method adopted by several trust models, such as (Wang et al., 2011; Su et al., 2013; Jøsang, 2007).

Requesters use Equation 3.5 to estimate the preference of responders and select a suitable strategy to make requests. Responders use Equation 3.7 to calculate the utility gain.

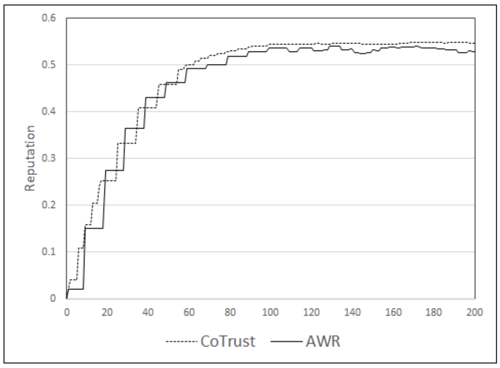
The first experiment examined how profiles influence the success rate of proposals. The obtained data was analysed at two different stages, i.e., at system bootstrap and after agents have some transaction records (stabilised stage). Figure 3.4a shows the success rate of two types of agents, i.e., reputation sensitive (RSAs) and benefit sensitive (BSAs) agents at the system bootstrap. As all agents had no experience at the time, there was no strategy applied in proposals. Requesters used their own preferences for proposals. The figure shows that proposals from high-quality and low-cost requesters (HL) have the most success rate (73.4%), the low quality, low cost (LL) requesters have the lowest success rate of 49.7%. Meanwhile, there is no significant difference between NN and NL.

Table 3.1 illustrates which requesters are likely to be accepted by responders in the stabilised stage. Consumers with high expectation for the interested attribute are more likely to be rejected by reputation sensitive agents compared with normal, and low expectation ones. However, they are all likely to be accepted at the similar rate for benefit sensitive agents.

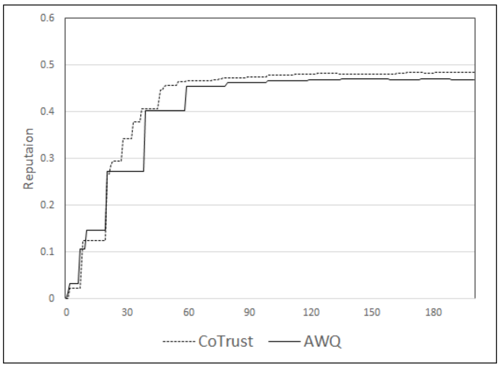
The next experiment investigates the satisfaction of cooperation. As the ratings reflect the satisfaction, we can also examine satisfaction via reputation information.



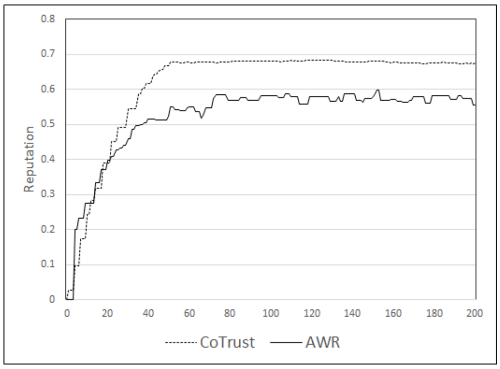
(a) Acceptance rate at system bootstrap



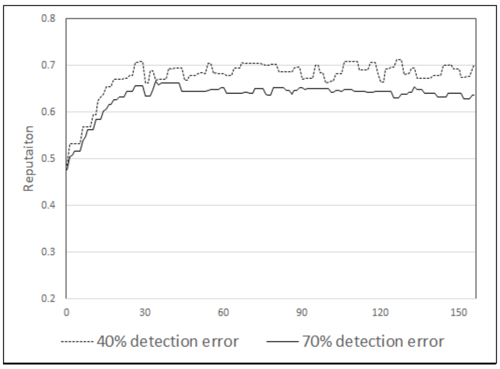
(b) CoTrust VS. AWR (50% HE responders, BS requesters)



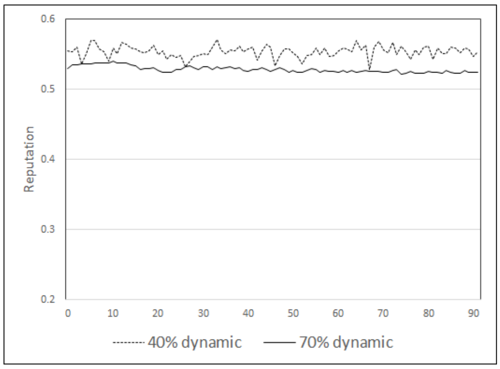
(c) CoTrust VS. AWR (75% HE responders, BS requesters)



(d) CoTrust VS. AWR (50% HE responders, RS requesters))



(e) CoTrust reputation under estimation error



(f) CoTrust reputation under dynamic settings

Figure 3.4: Experimental results

First, we increased the percentage of HE responders in the system by 50% and 75%. Then, we compared the reputations of requesters with the AWR model where responders always accept requests.

Figures 3.4b and 3.4c show the experimental results of benefit sensitive requesters (i.e., $\omega_{c_i}^{at_1} < \omega_{c_i}^{at_2}$) in the environment with 50% and 75% HE responders, respectively. The rejection rates of these two cases are 38.2% and 21.09%; the preference estimation accuracy is 96.7% and 98.4%, respectively. Because requesters are benefit sensitive, partners' rating is not their biggest concern (small value of $\omega_{c_i}^{at_2}$). Consequently, there is not much difference between AWR and CoTrust. However, with the increases in the percentage of HE responders, the average reputation of requesters drops significantly in these two cases, which is around 0.55 and 0.485 in a steady state.

Figure 3.4d illustrates the reputation in cases where responders are reputation sensitive RS. The reputation difference between CoTrust and AWR is larger in comparison with the case of benefit sensitive responders. We can see that under the AWR setting, the agents' reputation fluctuates significantly. Especially when the number of HE responders increases, the reputation values of those requesters are reduced considerably. Namely, CoTrust can improve the satisfaction more effectively in comparison with traditional approaches.

Next, we analyse the reputation of requesters regarding errors in estimating responders' preferences. Figure 3.4e shows reputation values in two cases, i.e., 40% and 70% estimation errors from NE responders. The reputation values converge to the predefined reputation value (rating threshold $\theta_{r_m} = 0.6$) when the detection error rate increases. Similar results can be obtained in the case of a highly dynamic environment (Figure 3.4f) where we allow responders to join and leave freely. The rejoined responders will have no transaction record, which leads to a higher estimation error.

3.5.1 Discussions

From the experiments, we can see that CoTrust can improve the success rate of proposals and improve the satisfaction of cooperation. However, in some circumstances, a requester can constantly be rejected by some providers. It is because their reputation or proposals do not meet the criteria of responders. To be accepted, the requester can consider changing their proposals or profiles. However, as mentioned in the previous section, CoTrust does validate the coherence between agents' reputation and proposals. Therefore, improving their current reputation can benefit agents in future cooperation. In other words, CoTrust brings incentives to both sides of cooperation to behave honestly.

The potential applications of CoTrust can be extended to address the bias ratings. Currently, many trust models rely on third-party testimonies to calculate trust values. Nevertheless, the credibility of advisers, who give opinions, is frequently not evaluated. As a result, filtering inaccurate opinions can be costly. If we consider advisers as responders who cooperate with responders to solve a trust evaluation of another agent, we can see that CoTrust will be able to enforce the advisers to give more accurate information.

As can be seen from the proposed protocol, it is the message exchange between two agents. However, the protocol can be used to form groups with more than two members by considering one member as the representative of the group to send a request for another agent to join the group. This approach has a limitation as it ignores the consensus among members. However, this case is out of the scope of this thesis as stated in Chapter 1.

3.6 Summary

This chapter presented CoTrust to address the lack of an effective trust establishment method regarding agent cooperation. To the best of my knowledge, this work is among the first research on active trust establishment in a group context (see Section 2.1).

CoTrust contains two main components, i.e., a protocol and a preference reasoning method. Compared to most current trust mechanisms, CoTrust can bring the incentives for both agents to behave honestly. In previous cases, only trustee agents have had this incentive. It is because of the protocol, which includes the two-sided trust evaluations, is suitable for a context of cooperation as it requires mutual trust from both partners. Meanwhile, the preference reasoning method helps to improve the success rate of cooperation requests.

In CoTrust, matched partners for cooperation are stable. Consequently, the satisfaction of agents can be improved with a higher gain of reputation. The proposed protocol and reasoning method of CoTrust are straightforward and easy to integrate with any system which already has a single-side trust evaluation.

The work in this chapter has been published in Nguyen and Bai (2017a).

Chapter 4

GEM: a trust evidence management method for agent groups

This chapter investigates the problem of inconsistent rating evidence of agent groups during group disbandment to enhance the robustness of trust and reputation systems. To achieve that, this chapter introduces an evidence management method which focuses on the rating distribution for group members. The proposed method can guarantee the fairness among members. Furthermore, it can enhance the robustness of trust and reputation systems by improving the consistency of rating evidence.

4.1 Overview

In Chapter 3, we introduced CoTrust, a trust establishment mechanism to assist the formation of agent groups. The mechanism involves trust evaluation during the partner selection process. It is worth to mention that the accuracy of the trust evaluation largely depends on the quality of evidence. Therefore, it is important to address possible issues related to evidence.

In MASs, truster agents often rely on trustee agents for their resources (or services).

The requested services can be diverse in types and sizes. A simple service can be delivered by a single trustee agent. However, some complicated services often require several agents working together to complete. Such services are often referred to as composite services, whilst the agent groups which provide the composite services are referred to as composite service providers. As the systems evolve, there is increasing demand for complex services. The significance of these services in MASs's robustness has attracted much research to improve the quality of agent groups by selecting the right members for the groups (refer to Chapter 3).

The problem we found when monitoring transactions in MASs with agent groups was the high possibility of missing evidence for trust computing. This happens when truster agents give feedback about the quality of the outcome without knowing the presence of contributing group members. Namely, all trustee agents, which are in charge of the component services, are not evaluated individually by truster agents. Since ratings are the main ingredient of trust evaluation (H. Yu, Shen, Leung et al., 2013; Wahab, Bentahar, Otrok & Mourad, 2015; Granatyr et al., 2015), the missing of ratings data can considerably affect the accuracy of trust evaluation and decision making.

Fig. 4.1 demonstrates a composite service provided by agent group G_j and its components. Conventionally, truster agent c_i rates composite service G_j after consuming it. How members of G_j can get rated is not an easy task, especially when given ratings can damage their reputation. Under uncertainty, group members may argue that they performed well and lay the blame on others for any undesirable rating of the group. The described situation has motivated us to find an efficient evidence management method which is capable of distributing fairly the rating of a group to its members under uncertainty. The problem somehow looks like the "fair-division" problems in game theory, which also deals with fairness and satisfaction to ensure envy-free division (Brams & Taylor, 1996). However, our problem differs from the games in many ways, from preliminary settings to procedures, that we will discuss more in Section 4.2.

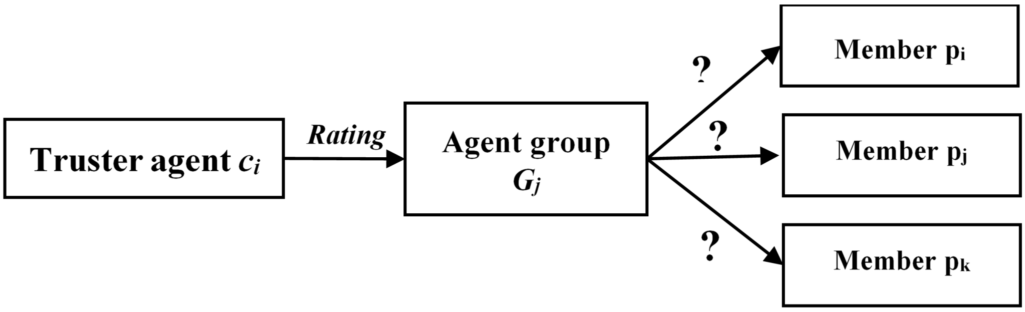


Figure 4.1: Rating distribution in a composite service

This chapter presents a group evidence management method (GEM). It consists of a rating distribution mechanism for group members, which is explainable and fair regarding members' interest. GEM maintains the consistency of evidence by diminishing the chance of losing rating data. The experimental results show the effectiveness of GEM in improving the accuracy of reputation systems and promising results against biased ratings.

The rest of this chapter is organised as follows. Section 4.2 reviews related work. The problem of evidence management is presented in Section 4.3. The rating distribution method is presented in Section 4.4. The experimental settings and analytical results are discussed in Section 4.5. Finally, the chapter is summarised in Section 4.6.

4.2 Related Work

In this section, we review some work related to the evidence management for trust/reputation models in MASs, especially in the context of agent groups. Despite the importance of trust evidence in trust/reputation systems, little research has been invested in evidence management. Some common problems of evidence that are addressed by several trust models are inaccurate ratings or biased opinions, e.g. as found in W. T. L. Teacy et al. (2005) and Fang, Guo and Zhang (2015b).

Only some rating patterns and value scales have been discussed and integrated into corresponding trust evaluation methods. For example, ratings can take binary values, i.e., good or bad (Commerce et al., 2002), ternary such as eBay rating system [1], or multi-nominal (values of 1 to n) in Jøsang and Haller (2007). Rating values can also be continuous or multi-dimensional (H. Yu, Shen, Leung et al., 2013).

4.2.1 Ratings and agent groups

Regarding agent groups, most published research focuses on improving the quality of group composition. For example, T. Zhang et al. (2014) proposed service composition by evaluating the trust degree of each service based on a lattice-based trust model. Jiang and Bai (2013b) proposed a provenance-based trust estimation approach for service compositions. The evidence used for service compositions was analysed using historical provenance information. Similarly, Wang, Chen, Cho, Chan and Swami (2013) and Dai and Wang (2009) developed a trust-based service composition respecting maximising the quality of service and information without developing a trust model for composite services. As can be seen, these works are the flip side of our problem. Namely, they focus on how to use evidence in the group formation stage, while the study in this chapter focuses on how to manage evidence at the group disbandment stage.

There is little study on trust management for agent groups in MASs. The study in Al-Oufi, Kim and Saddik (2012) created an extended version of the Advogato trust graph to determine a person's reliability according to the strength of her/his relationships and the group of neighbours trusted by an individual agent. Nevertheless, the term "group" in Al-Oufi et al. (2012) does not coincide with the "agent group" which provides composite services in this chapter. The work of Tong and Zhang (2009) is one of the first works on trust of agent groups. However, the results are still very limited.

Managing evidence for agent groups is important not only for the trust evaluation

[1] https://pages.ebay.com/help/feedback/scores-reputation.html

models but it can also provide proper evidence to enable these models to work. For example, the trust models of Commerce et al. (2002), Huynh et al. (2006b), and Wang and Singh (2010) are not applicable for reasoning trust in the context of groups because they lack a reasoning mechanism with *indirect ratings* (refer to Definition 4.3.2). FIRE (Huynh et al., 2006b) combines four evaluation methods, i.e., direct experience, witness information, role-based rules and third-party reference (certified reputation model). However, the uncertainty of group ratings and members' contribution within a group makes none of these methods work.

4.2.2 Cooperative games and fair division

Regarding sharing among group members, the cooperative game and fair division are well-known concepts in game theory. The Shapley value (Shapley, 1953) is one solution to distribute the total gains (payoff of the coalition S) to the players (members) regarding "fairness", assuming that they all collaborate. In a particular game v, the players involved are contained in any carrier N, which is a subset of U (all players) such that $v(S) = v(S \cap N)$. It is a "fair" distribution in the sense that it is the only distribution complying three axioms listed below:

1. The symmetry axiom requires that players who are treated identically by the characteristic function are to be treated identically by the value.

2. The carrier axiom requires that the sum of the payoff $\phi_i(v)$ over all players i in any carrier N are equal $v(N)$.

3. The additivity axiom requires that for any games v and w, $\phi(v)+\phi(w) = \phi(v+w)$

In A. Liu, Li, Huang and Wen (2012), the authors proposed a Shapley value-based reputation management model for web service composition. The approach assumes that the contribution of each service component to the QoS is computable.

The assumption hardly holds in real-world applications under uncertainty of rating. It is not applicable for distributing the rating of the group to members because several preliminary conditions are not met: (1) The rating value does not comply with the additivity axiom: in general, the sum of rating of the members does not equal the rating of the group. Additionally, a rating is not an item in either "divisible" or "indivisible" categories of fair division games. We cannot simply divide the group rating by the number of members. For example, it is not reasonable to divide a 5-star rating to 3 members; each gets a rating of 1.7-stars. (2) The contribution of each group member (sub-service provider) in our study is assumed to be unknown because there is a weak link between composite service's rating with its sub-services.

Hereafter, this chapter will discuss the distribution of ratings for members of an agent group with an evidence-based approach. To improve the satisfaction of rated members, the proposed method also considers several supporting facts for the distribution, i.e., the performance graph extracted from collective evidence to evaluate the performance of group members. We develop a process for retrieving relevant evidence from historical records, analysing the performance of members in groups, then distribute the rating based on supporting evidence so that it can diminish conflicts of individuals' interest.

4.3 Problem Definitions

Managing evidence in MASs is a non-trivial task. Not including the inaccuracy of information, it also needs to deal with the consistency of evidence as mentioned in the previous section. In this study, the evidence to be considered is the ratings from previous transactions of agents. Furthermore, I focus mainly on the rating distribution method among group members. A challenging problem when distributing group rating is how to resolve the conflict of interest among group members. The dissension may occur among group members if there is not a proper distribution method. For example, when a group receives a rating of 0.6, it is the overall rating for the quality of the group. However, each member of the group could perform better or worse than the received group rating. Therefore, to distribute the rating and minimise the conflict of interest between members, we choose an evidence-based approach to support our rating scheme.

This study considers the following preliminaries:

1. A rating from a truster agent (consumer) to an agent group indicates the quality of service provided by the whole group, not for a particular member.

2. The contribution of each group member to either the quality of service or the rating from a truster agent is unknown by the group.

These conditions highlight the uncertainty between group rating and members' performance. Different truster agents may have different criteria when rating the same service. According to Yaari and Bar-Hillel (1983), the empirical experiments on how people define the concept of objective fairness leads to inconclusive results. Thus, rating distribution with objective fairness is not feasible. Like most current research on fairness, this study also adopts a concept of subjective fairness. The *fairness* of rating distribution is defined by the following principles: (1) Members with the same set of supporting evidence will get the same rating value (2) members can deny a rating only if their provided evidence can prove otherwise.

Definition 4.3.1. A ***composite service*** (G) is a service which contains several simple services provided by different trustee agents, i.e., $G = \{p_1, p_2, \dots, p_n\}$ $(n \in \mathbb{N}^*)$.

A composite service G_i is provided by a group of agents. Meanwhile, a simple service (ss) is a service that is provided by a single trustee agent. Each trustee agent is represented by a 2-tuple, i.e., $p_i = (ss_i, D_{p_i})$, in which D_{p_i} is the transaction history (refer to Definition 3.3.3) of p_i. In this chapter the terms *"composite service provider"* or *"agent group"* are used interchangeably.

The formation of a group depends on the required functionalities for the composite service. The trustee agent with the highest reputation value for a specific service will be selected for the group. For example, p_3 is selected from the three candidates $p_1(t_1 = 0.8), p_2(t_1 = 0.6), p_3(t_1 = 0.9)$ for service ss_m.

Definition 4.3.2. An ***indirect rating*** (in the context of agent groups) is a rating generated by a group (see Definition 4.3.1) to its members, based on the rating of a truster agent to the group.

Direct ratings are the most common type of ratings that we can find in many trust and reputation models. They are assessment values generated by truster agents to trustee agents. However, in most cases, the rating for a composite service cannot just be passed to the group members. It is because the rating represents the assessment of the quality of service (QoS) of the whole group, not for a particular member. Furthermore, a group does not consume its own services, so it cannot rate its members like actual truster agents. To maintain the consistency of the ratings, a group should not rate its members with arbitrary values. Instead, the ratings of members must be related to the rating of the truster agent given to the group. We call this kind of rating *indirect rating*.

Note that indirect ratings are inferred values from direct ratings. They do not belong to the indirect evidence (or recommendations) category mentioned in Chapter 2.

Definition 4.3.3. The ***group rating*** R_{G_i} is a real value between 0 and 1, representing the QoS of group G_i evaluated by a truster agent. 0 and 1 represent the ratings of the least and most satisfied values, respectively.

There are various ways to represent ratings. For example, multi-dimensional values can also be used to rate the quality, delivery or any other criteria defined by users (Mehdi, Bouguila & Bentahar, 2015; H. Yu, Shen, Leung et al., 2013). However, due to the assumption of the uncertain contribution mentioned at the beginning of this section, this study uses single values for ratings.

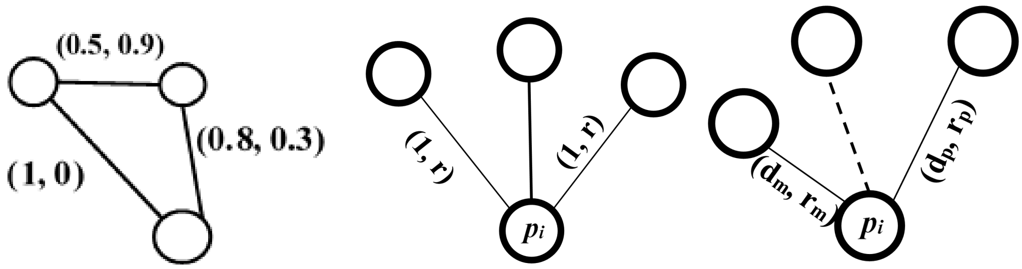
Each transaction record of agent a_i is stored in D_{a_i} (see Definition 3.3.3). A transaction record is represented by a 2-tuple, i.e., $\tau_{i_j} = (G_j, R_{G_j})$, where G_j is a group p_i once participated in, and R_{G_j} is the rating of G_j in providing the service. The agent itself maintains the historical record for future references. Visibility and verifiability are conditions enabling the cross-validation to reduce the chance of deceit or alternating evidence.

Definition 4.3.4. An ***extended evidence space*** is a set of evidence provided by all the members of a composite service, including the current transaction of the group as a record.

The extended evidence space is an important concept in this study because it can assure each member has at least one piece of evidence, which is the transaction of the current group. Therefore, this approach can overcome some obstacles of trust reasoning for new trustee agents in the system. The extended evidence space will be elaborated on further in Section 4.4.

Definition 4.3.5. The ***performance graph*** of a group (G_i) is an ordered pair $PG(G_i) = (V, E)$ comprising of set V of nodes or trustee agents together with a set of undirected weighted edges, i.e., E, indicating the degree of the performance relationship between a pair of nodes from all evidence provided by group members.

From the evidence space, we construct a performance graph of group members. Nodes represent trustee agents, while their edges contain two pieces of information: (1) the closeness (distance) between two trustee agents, and (2) the performance weight as a combination of the two members, i.e., $e_{p_i,p_j} = (d_{p_ip_j}, r_{p_ip_j})$. The distance $d_{p_ip_j}$ reflects the cohesion between trustee agents, it is calculated by the number of times they cooperate. The weight $r_{p_ip_j}$ is the pairwise performance of these members when they work together. Figure 4.2a visualises a performance graph of a group containing three trustee agents. We can see the relations of three members regarding interaction frequency and performance. The parameters presented at the edges of the graph are not geodesic distance but are real values representing cooperation frequency between nodes. Unlike pseudo-logarithmic distance introduced in Dekker (2005), here, the smaller value represents the closer relation.



(a) 3 members group (b) No evidence (c) With evidence

Figure 4.2: Examples of groups with extended evidence space and their performance graphs

Definition 4.3.6. The ***unaccountability*** of member p_i in group G, i.e. $UA(p_i|G)$, is a real value indicating the degree of the uncertain contribution of member p_i to group G in respect to the group rating.

The performance of a member can be influenced by the various factors in its group, e.g., the performance of group members, workflow, coordination protocol, etc. Under uncertainty, no conclusion about a member's performance can be drawn. Namely, if the group receives a high rating, a member could say it is because of his presence. Likewise, one could blame another members' presence for the bad rating of the group. Both arguments are considered equal under uncertainty unless they can be proved wrong. Therefore, the unaccountability factor is introduced to resolve the distribution fairness for self-interested members. The lower the unaccountability coefficient, the less responsibility a member has if there is an undesirable rating of the group.

Definition 4.3.7. The ***evidential performance*** of agent p_i to group G, denoted as $ep(p_i|G)$, is a real value indicating the potential performance of p_i in group G. It is calculated from the extended evidence space for supporting p_i to obtain rating reward from a group G.

An evidential performance can be used to reason the possible contribution of members with respect to a group rating. Its concept is similar to a trustee's trust evaluation, as it analyses the provided evidence. Additionally, outdated evidence should not be used. The detail of how to manage evidential performances will be discussed in Subsection 4.4.4.

4.4 Trust evidence management method

We now present the detailed implementation of GEM. It consists of a protocol and a subtle rating distribution method for agent groups. The ratings of group members should be based on the performance of them in the groups. Unfortunately, one group rating is insufficient to conclude the quality (performance) of each member of the group.

Sparrowe, Liden, Wayne and Kraimer (2001) found that the performance of a provider can be discovered by analysing the services that they attended to and the relationships among group members. For example, a company hired to make cameras for iPhone 7, which is a highly-rated phone brand in the market, can be expected to produce a good quality camera for other cell phone brands too. In this case, the ratings of iPhone are taken into account to assess the camera manufacturer. The fact is that the rating for composite services sometimes means the rating for their components in general. It is known as the ontological dimension in the REGRET model (Sabater & Sierra, 2001b). Intelligent entities can share an understanding of other's ratings, e.g., a good phone might imply not only a good build quality but also good quality of the voice or the design. On the other hand, the performance of a service provider sometimes depends on their partners. For example, if provider p_i and provider p_j have been working in high rated groups, it is reasonable to expect the combination of p_i and p_j will likely be good in future cooperation. Thus, the distributed ratings of members in a group are influenced by their evidential performance (calculated from previous evidence) adjusted by the uncertainty (Δ_r) of the group G.

$$R(p_i|G) = ep_{p_i} + \Delta_r(p_i|G) \tag{4.1}$$

The uncertainty factor Δ_r can be considered a rating regulator of p_i. In the context of a group, unknown factors can cause the variation in the performance of members. Consequently, they may affect the group rating and the rating distribution. To resolve Δ_r, our approach makes use of the data obtained from the composite service including group features and members' relationships. Hence, the proposed rating distribution method consists of three corresponding processes, i.e., the evidential performance evaluation, the unaccountability evaluation, and the rating distribution (see Figure. 4.3).

1. *The evidential performance evaluation* uses evidence provided by p_i to measure the base performance of p_i based on a current composition of members (refer to Subsection 4.4.2).

2. *The unaccountability evaluation* deals with uncertainty. It resolves the chance of conflict by adjusting the rating based on the relevance of evidence and the current group context (refer to Subsection 4.4.3).

3. *The rating distribution* combines the actual rating of a group with the results generated by above processes to give the rating to each group member (refer to Subsection 4.4.4).

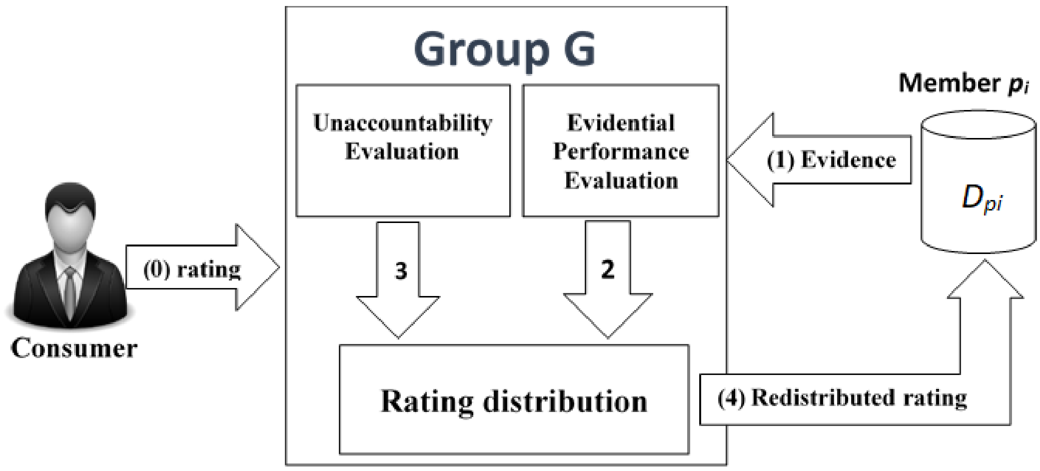


Figure 4.3: The proposed group evidence management

4.4.1 The rating distribution protocol

Figure 4.3 illustrates the rating distribution process: (0) receiving rating, (1) collecting evidence, (2) calculating evidential performance, (3) the unaccountability factor, and (4) distributing rating. The members of a group must comply with the following protocol to enable the rating distribution. It consists of the five following steps.

1. Truster agent C makes a request to trustee agent B. B then needs to form an agent group (G), as B cannot deliver the service individually.

2. Group G is formed by selecting members according to the required functionalities and the current reputation values of available trustee agents.

3. After receiving a rating from C, if group G has more than one member, it asks each member of G (e.g., p_i) to provide supporting evidence $\{\tau_{i,j}\}$ from its historical records (D_{p_i}) to calculate the distributed rating.

4. G uses all evidence related to p_i and the rating of this group to distribute the rating to member p_i. The distribution rewards more for members with better evidence.

5. p_i receives the distributed rating from G, updates its reputation with the existing mechanism of the system, and leaves the group.

This protocol can be adapted easily to existing trust management systems to evaluate the trustworthiness of every trustee agent since it has no particular requirement. In the case that a group has only one member, it then turns out to be a classic trust evaluation problem.

4.4.2 The evidential performance

The evidential performance evaluation (refer to Definition 4.3.7) discovers the potential performance of members considering the presence of other members of the group. For example, if evidence shows that the composition of two trustee agents p_i and p_j have more good ratings than bad ratings, then this fact can support both p_i and p_j to get better ratings. As it is relationship-based, we use a graph (refer to Definition 4.3.5) constructed from the merged evidence space to discover the performance of members of a certain composition.

By using the extended space, it guarantees that all nodes in the graph are connected, i.e. any two nodes have an edge. We discuss advantages of the extended history more in Subsection 4.4.3. An edge's length is calculated by Equation 4.2.

$$d_{p_i p_j} = \frac{1}{1 + \log(k)} \tag{4.2}$$

where k is the number of evidence (records) that contains both p_i and p_j. $d_{p_i p_j}$ has value in (0, 1]. The logarithm adopted here is to denote the closeness based on communication frequency (Dekker, 2005). Figure 4.4a depicts the edge length based on the number of transactions between p_i and p_j. It is easy to verify that the distance of a member, who has only one piece of evidence after joining the current group, to other members is $d = 1$.

Figure 4.2b demonstrates the case where the member p_i has no acquaintance before joining the group. Thus, all edges from p_i have length 1. Since the edge length indicates the cooperation frequency between nodes, it can also be considered as the confidence indicator for the performance of them.

Figure 4.2c shows that p_i has edges with different distances. It can be seen that the performance edge is not addictive, which means $r_{p_ip_j} \neq r_{p_ip_k} + r_{p_kp_j}$ generally. To assess the performance of p_i in a group, we first assess the pairwise performance of p_i and p_j by the following formula.

$$r_{p_ip_j} = \frac{\sum_{l=1}^{n} r_{\{G_n|G_n \in D_i\}} . \delta_l}{\sum_{l=1}^{n} \delta_l} \tag{4.3}$$

where δ_l is the discount factor that controls the weight of evidence, based on time stamp. The more recent the evidence is, the more important it is. For instance, high rated but outdated evidence should not be used, i.e., filtered by δ_l, which can be calculated by Equation 4.4

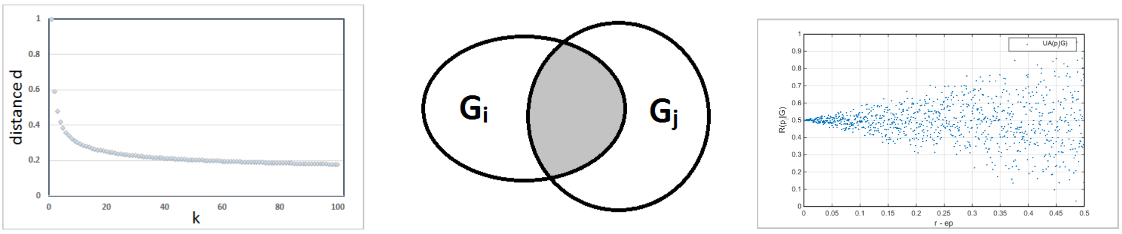
$$\delta_l = e^{-\frac{\Delta_t}{\lambda}} \tag{4.4}$$

where Δ_{t_i} is the time difference between the time when the i^{th} evidence was recorded and the current time, and λ is a coefficient to adjust the discounting speed.

The evidential performance of the p_i in G is drawn from provided evidence containing p_i. Using Equation 4.2 and 4.3, the evidential performance of p_i in the certain combination is formulated as:

$$ep(p_i|G) = \frac{\sum_{j=1}^{j=n} d_{p_ip_j}^{-1} r_{p_ip_j}}{\sum_{j=1}^{j=n} d_{p_ip_j}^{-1}} \tag{4.5}$$

In Equation 4.5, $ep(p_i|G)$ indicates the evidential performance of p_i in the presence of other members of group G. The closer distance trustee agents means the evidential performance has more weight.



(a) Edge length (social distance) $e_{p_ip_j}$

(b) Group feature similarity

(c) $R(p_i|G)$ when $ep \rightarrow r_G$

Figure 4.4: Components of the proposed rating distribution algorithm

4.4.3 The unaccountability

The frequent dispute over distribution problems is when a member receives a rating lower than expected, as the rating can affect their reputation negatively. As there is always a difference (positive or negative) between the group rating and the evidence-based performance ($\Delta_p = ep - r_G$), a member can avoid punishment if his evidence can support the fact that he is less than likely to be the cause of the poor performance. To this end, we introduce an unaccountability factor (refer to Definition 4.3.6). This approach measures the unaccountability by comparing the difference between feature sets (F_{G_i}) extracted from two groups. The Jaccard coefficient (Levandowsky & Winter, 1971) is used to calculate the group's similarity.

$$Sim(G_i, G_j) = \frac{|F_{G_i} \cap F_{G_j}|}{|F_{G_i} \cup F_{G_j}|} \tag{4.6}$$

It can be seen that $0 \leq Sim_i(F_{G_i}, F_{G_j}) \leq 1$, where "0" means two groups have no common feature and "1" means two groups have exactly the same feature set.

The term "feature" of a group is flexible and context-dependent. It can mean the service types, structures or workflow pattern of a service or any combination of them. To improve the accuracy of similarity measurements, features should be carefully selected by the system administrators. Figure 4.4b demonstrates the similarity of two groups with a Venn diagram; the shaded area indicates their common features. It also indicates the relevancy between two pieces of evidence, while the white areas indicate the uncertainty that produces the performance variation of the two groups. In other words, the white areas support the unaccountability.

In the case that the group has a new member with no transaction history, the similarity will be 0. This is an important boundary value that influences the consistency of our approach (Equation 4.7). However, by using the extended evidence space mentioned in Subsection 4.4.2, we have subtly handled this issue. The extended evidence space now includes both old attended groups and the current group, making the number of records always greater than 0. The similarity between two groups indicates the supporting degree of the evidence. The lower value of similarity that there is, the less relevant to provided evidence there will be.

The relevancy of the extended evidence of p_i and the current group G is formulated as follows.

$$Rel(p_i|G) = \frac{\sum_{j=1}^{k} Sim(F_G, F_{G_{ij}})}{k} \tag{4.7}$$

where k is the amount of evidence (including current group G) provided by p_i, G_{ij} is evidence from D_i. In the case that member p_j has no previous history, the extended evidence space makes $Rel(p_i|G) = 1$. We have the unaccountability of p_i:

$$UA(p_i|G) = 1 - Rel(p_i|G) \tag{4.8}$$

4.4.4 Rating distribution

Our approach uses evidential performance and relevancy factors to distribute the actual rating of a group to its members. The distributed rating has a value between the evidential performance and the group rating. The rating differences vary due to the differences between provided evidence and the current group. It means that if the group

Table 4.1: Example of distributed rating calculation

Evidence	$G_{i_1}(r = 0.8)$	$G_{i_2}(r = 0.75)$	$G_{i_3}(r = 0.9)$
$Sim(G, G_{ij})$	1.0	0.6	0.6
$Rel(p_i\|G)$	$UA(p_i\|G)$	$ep(p_i\|G)$	r
0.73	0.27	0.81	0.5
$R(p_i\|G)$	0.726		

rating differs from the evidential performance of a member, then the distributed rating should be adjusted by the contextual differences. Thus, the distributed rating depends on the rating of the current group, the difference between evidential performance and this group rating value, and the relevancy of evidence.

Regarding conflict resolution, the evidence-based approach guarantees that only relevant evidence provided by group members is considered. A member is not allowed to cite other groups' ratings that are not related to the current group. Here, we developed a formula satisfying those conditions that will give a better reward for the member with better evidence. Applying Equation 4.5 and 4.8 to Equation 4.1, we can obtain the rating distribution for each member of group G.

$$R(p_i|G) = \begin{cases} ep_{p_i} + (1 - UA(p_i|G)) \cdot (r_G - ep_{p_i}) & \text{if } ep \le r_G \\ ep_{p_i} + UA(p_i|G) \cdot (r_G - ep_{p_i}) & \text{if } ep > r_G \end{cases} \tag{4.9}$$

where r_G and ep_i are the reference values coordinating the distributed rating to member p_i. When $r_G - ep \ge 0$, we call the second part of the equation (containing $r_G - ep_{p_i}$), the "rating reward". Otherwise, the part can be considered as the "rating punishment", which is associated with the unaccountability factor.

Table 4.1 shows an example of rating calculation for p_i. G_{i1}, G_{i2}, G_{i3} are extracted from D_{p_i}, where the group ratings are 0.8, 0.75 and 0.9 respectively. The similarity evaluation gives them 1.0, 0.6 and 0.6. Finally, Equation 4.9 distributes a rating of 0.726 for p_i.

Proposition 4.4.1. $\min(r_G, ep_{p_i}) \le R(p_i|G) \le \max(r_G, ep_{p_i})$

Proof. From Equation 4.7 and 4.8

$\because 0 \le UA(p_i|G) \le 1$ and $0 \le r_G \le 1$

Case $r_G > ep$

$$\begin{aligned} R(p_i|G) &= ep + UA(p_i|G) \cdot (r_G - ep) \\ &\le ep + 1 \cdot (r_G - ep) = r_G \\ R(p_i|G) &= ep + UA(p_i|G) \cdot (r_G - ep) \\ &\ge ep + 0 = ep \end{aligned}$$

Case $r_G \le ep$

$$\begin{aligned} R(p_i|G) &= ep + (1 - UA(p_i|G)) \cdot (r_G - ep) \\ &\le ep + 0 = ep \\ R(p_i|G) &= ep + (1 - UA(p_i|G)) \cdot (r_G - ep) \\ &\ge ep - 1 \cdot (ep - r_G) = r_G \end{aligned}$$

□

Proposition 4.4.2. $R(p_i, G) = r_G$ when member p_i has no transaction history before joining this group G.

Proof. From Equation 4.5 : $ep_{p_i} = r_G$

From Equation 4.7: $\because p_i$ has no history $\therefore Rel(p_i) = 1$

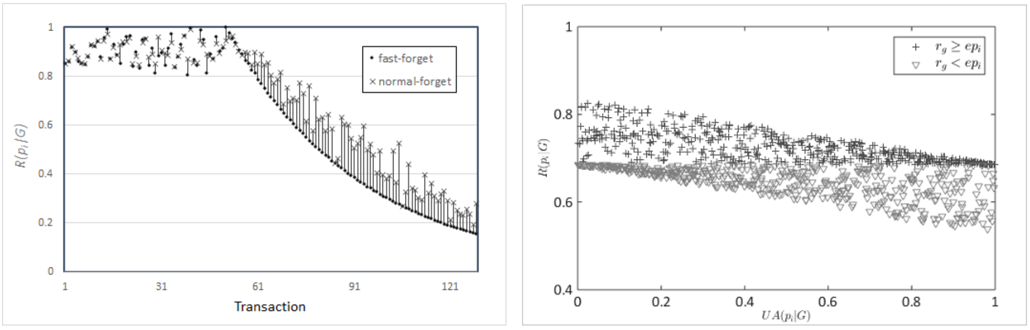
$$\therefore UA(p_i|G) = 0$$

Apply to Equation 4.9:

$$\begin{aligned} R(G, p_i) &= ep + UA(p_i|G) \cdot (r_G - ep) \\ &= r_G + 0 \cdot (r - r) = r_G \end{aligned}$$

□

The intuition of Proposition 4.4.1 and 4.4.2 is to guarantee that group members will receive the distributed rating value within the evidential performance and group rating. Moreover, the value is at least the same rating as the group rather than 0 when a member has no supporting evidence. It is a reasonable distribution for the new trustee agents because, under uncertainty, no evidence means no adjustment.



(a) Discount factors (b) Unaccountability factors

Figure 4.5: The $R(p_i|G)$ with different discount and unaccountability factors

Proposition 4.4.3. The distributed rating value gets close to the current group rating r_G when the evidential performance value approaches the group rating.

Proof.
$$\lim_{ep \to r} R(p_i|G) = r_G + UA(p_i|G) \cdot \lim_{ep \to r} (ep - r_G)$$
$$= r_G + UA(p_i|G) \cdot 0 = r_G \qquad \square$$

Proposition 4.4.4. The distributed rating value gets close to the evidential performance $ep(p_i|G)$ (or group rating r_G) when the unaccountability coefficient approaches 0 (or 1).

The proof of this proposition is similar to Proposition 4.4.3. Figure 4.4c illustrates the case $ep \to r_G$, ratings of G to p_i converges to the rating of group $(r_G = 0.5)$ if its similarity is close to the group average similarity. The similar result can be achieved when the evidence trust gets closer to the group rating r_G.

The propositions 4.4.3 and 4.4.4 yield that the approach rewards more and punishes less for members with lower unaccountability. The result can also be interpreted as given any unexpected ratings, the reward or punishment is small when the provided evidence is more relevant to the current group. It is the key to counter the biased rating problem.

The extended evidence space benefits the reputation system since it gives no better rating for any coalition between group members. Since the space is unique, it, therefore, eliminates the case in which some evidence of p_i benefits only p_i and not p_j. It makes the second condition of objective fairness defined in Section 4.3 feasible.

Proposition 4.4.5. $R(p_i, G) = r_G$ when member p_i is the only member of group G.

The proposition can be proved simply by using the protocol defined in Subsection 4.4.1. It implies that the proposed distribution for an agent group is a generalised case of individual ratings for trust and reputation systems. It provides a feasible solution for the one-to-many (one truster agent to many agent members of a group) rating scheme, and it also fits the one-to-one rating system.

4.5 Experiments and Discussions

Several experiments have been conducted to evaluate the effect of the distribution on the accuracy of the reputation system and the satisfaction of rated members. Specifically, we investigate the impact of introduced factors (e.g. unaccountability and discounting), the consistency of reputation systems, the satisfaction of the rating distribution method, and the ability to resist the impact of biased ratings.

To analyse the consistency, the experiments were carried out in two distinct environmental settings: one with a rating distribution ("RD" system) and one without a distribution method ("NRD" system). To evaluate the satisfaction of rated members and the impact of different factors, we compared the proposed approach with two other methods with the same settings in the RD environment. The two methods to compare are the ones adapted from A. Liu et al. (2012) and Nguyen and Bai (2014).

Without the loss of generality, each trustee agent was created with only one type of service. Ten simple service types were defined, and 60 trustee agents are initialized with various performance types (low, medium, high) associated with these services. 500 honest truster agents were created to make requests for composite services. The system operates with several transaction rounds; a random number of agent groups were formed to handle requests for composite services each round. Each experiment session contained more than 20,000 transactions. A trustee agent could join only one group at a time and be selected based on its reputation. All experiments used the same reputation management to calculate the reputation values of trustee agents after they received ratings.

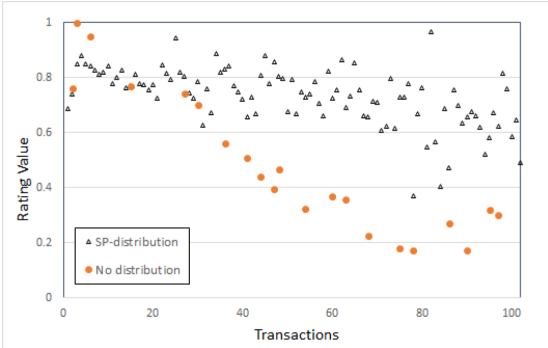
At the system bootstrap, all agents had no transaction records, and members received the same ratings as the group rating each transaction. After some transactions, the distribution method gave additional adjustments for members' ratings as shown in the following experiments.

4.5.1 Effect of unaccountability and discount factor

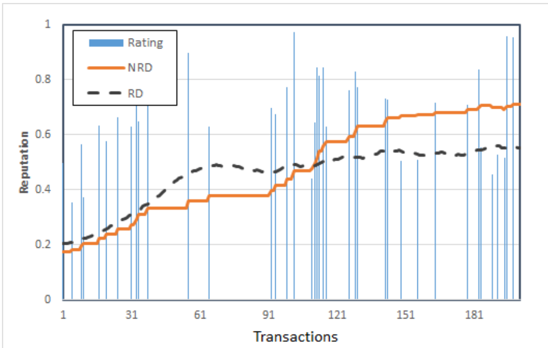
In this experiment, we investigated the impact of discount and unaccountability factors to rating distributions and members' satisfaction. All transactions complied with the protocol defined in Subsection 4.4.1 and Equation 4.9 in order to distribute the rating among group members.

Figure 4.5a illustrates the effect of the discount factors when distributing ratings in two cases: the small and regular discount factors. When the value of δ is small, the system will forget old evidence more quickly (fast-forget). Consequently, the distributed rating comes closer to the group rating. If we choose a very small value for λ ($\lambda \to 0^+$), all the previous evidence will be ignored, and only the current group rating will be used for all the members. It supports the idea that for trustee agents with more evidence, more adjustment will be made for distributed ratings.

The effect of the unaccountability factor $UA(p_i|G)$ to $R(p_i|G)$ is demonstrated in Figure 4.5b in two cases, i.e., $r_g \geq ep_i$ and $r_g \leq ep_i$. The experiment used a fixed value of ep_i and varied the received rating (r_g) of the composite service. When $UA(p_i|G)$ is



(a) The missing ratings



(b) Reputation lag issue

Figure 4.6: Consistency analysis in the case of missing ratings

small ($UA(p_i|G) \to 0$), the mechanism rewards more for p_i when $r_g \geq ep_i$. Likewise, when $UA(p_i|G)$ is large, it punishes more when $r_g \leq ep_i$. The rationale of these is that rational agents want to minimise their responsibilities for any fault whenever their group receives a bad rating. When the fault is unknown, the unaccountability factor shows the eligible proportion of denial of responsibility for a member. Similarly, if a group member wants to claim for the reward of their contribution for a good rating of the composite service, the relevancy $Rel(p_i|G)$ is used to validate the claim.

4.5.2 Consistency evaluation

This experiment focuses on how distribution affects ratings and reputation management. We monitor the reputation of agents with a fixed performance and trustworthiness in four combinations of settings: with and without a rating distribution method under (60%)

Table 4.2: Reputation variation on decision makings

Composite Service Requests		NRD	RD
0%	Var NRD	0.0845	0.0845
	Missing ratings	0	0
60%	Var RP	0.1611	0.0717
	Missing ratings	120	0

requests for composite services and requests for atomic services only. We can evaluate the effectiveness of the rating distribution to reputation management by comparing the intrinsic trustworthiness of an agent with its reputation calculated by systems. If the reputation values of monitored agents are closer to their actual trustworthiness, it is better. Table 4.2 shows the statistical analysis for the four settings.

It is easy to see when there is no request for composite services, i.e. when there are only requests for simple services. The two systems are the same in terms of reputation management because there is no need for distributing rating values. The ratings from truster agents can reach trustee agents directly, and there is no missing rating evidence. As a result, the variations of reputations are the same for both systems. On the other hand, when 60% of the requests are for composite services, the similar proportion of missing evidence is presented for the NRD system: 80 records out of 200 transactions. The analytical result shows the significant difference between the two systems with the reputation variation of 0.16 and 0.07 for NRD and RD respectively. It means that the RD system can make a decision based on the reputation that is closer to the service's trustworthiness.

Figure 4.6 illustrates the effect of the rating distribution on NRD and RD systems. However, when there is an increasing number of requests for composite services (60% of requests), the NRD system suffers greatly from the missing rating problem. Without a rating distribution, the evidence space of this trustee agent is 60% less than its total transaction (Figure 4.6a).

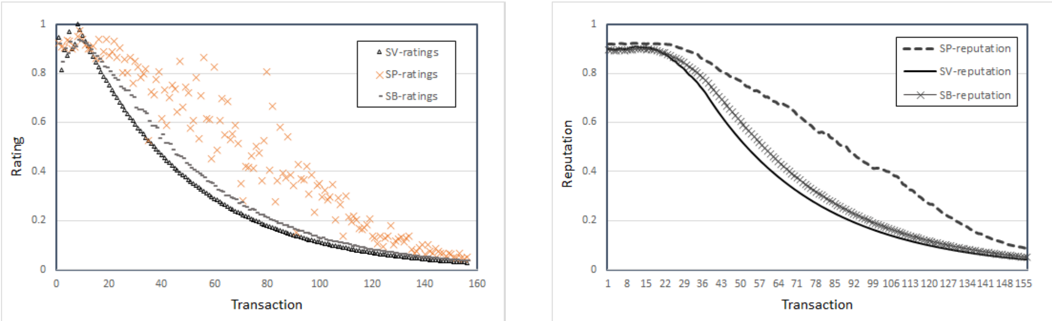
As a consequence, the reputation of the trustee agent in the NRD system suffers from a phenomenon similar to the reputation lag problem (Kerr & Cohen, 2009), i.e., reputation values are calculated from an incomplete rating collection. The incompleteness of information is illustrated by plateaux in Figure 4.6b. During a period of a plateau, the actual performance of the trustee agent may be changed, but the agent can still be selected to join different groups based on the outdated reputation. Whereas, the RD system with a distributed method can significantly reduce the degree of the plateaux, and reflect the reputation trend more accurately.

4.5.3 Satisfaction evaluation

The satisfaction of members is measured under the self-interested and greedy assumption. Namely, rational members prefer higher distributed ratings when the fairness conditions are met. We investigate how different distribution methods harmonise group ratings based on trustee agents' evidence.

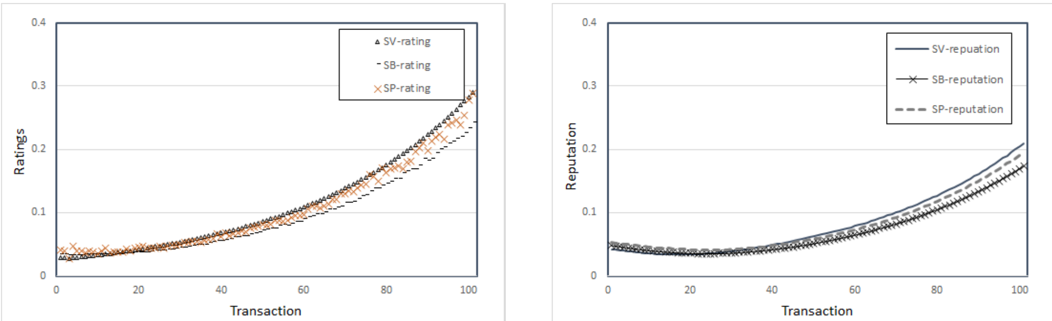
We denote our approach as the "SP" method as it takes members' social relationship into consideration. The experimental results are compared with another two approaches, one from Nguyen and Bai (2014), which rates members based on simple services combination ("SB" method) without social performance analysis and the other from A. Liu et al. (2012) which is a Shapley value-based approach ("SV" method). As the information about QoS contribution is not available in these settings, the SV approach leads to the equal probability of members-only marginally contributing. As a result, the SV approach assigns the same rating to all the members within the group.

In the first scenario, we focused more on trustee agents with good profiles as they are more sensitive to bad ratings, i.e. the group received decreasing ratings. The statistical results in Table 4.3 show that the SV method distributes the same rating of the group



(a) Rating Distribution (b) Reputation

Figure 4.7: Rating distribution and reputation with decreasing ratings



(a) Rating Distribution (b) reputation

Figure 4.8: Rating distribution and reputation with increasing ratings

to its members while both the SB and SP methods show the additional adjustment for ratings. However, the SB method rewards less than SP in this setting because SP has more supporting clues to receive better ratings, i.e., the inter-member relations. Consequently, the reputation values of the SV and SB method decline relatively fast, while the reputation of the SP method decreased slower (Figure 4.7b). It means that the SP method gave better satisfaction to rated members as SP can give a relatively higher reputation (average 0.546) compared to the other two (0.379 and 0.407).

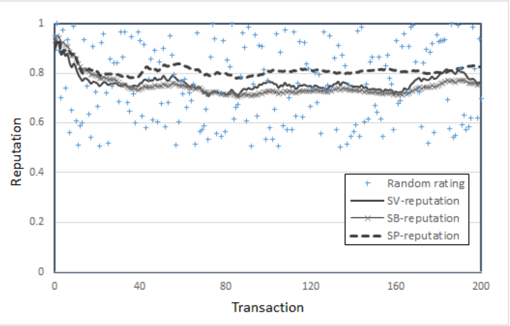
In the second setting, we observe trustee agents with low reputations under increasing group rating values (see Figure 4.8a and 4.8b). The distributed ratings of the SP method lie in between the other two. The SP method could not give ratings as high as the NR method because the evidential performance ep_i was smaller than the group

Table 4.3: Reputation under different rating patterns

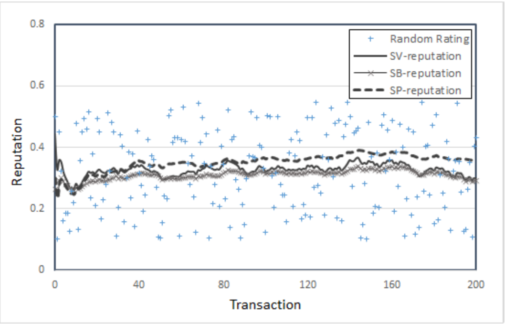
Rating pattern	**Reputation**	**SV**	**SB**	**SP**
Increasing	Mean	0.0843	0.0726	0.08079
	Var	0.00273	0.00165	0.00208
Decreasing	Mean	0.3798	0.4075	0.5460
	Var	0.0.09587	0.09578	0.08271
Random (high)	Mean	0.7603	0.7494	0.8112
	Var	0.00122	0.00202	0.00055
Random (low)	Mean	0.3276	0.0.3098	0.3505
	Var	0.00041	0.00025	0.00091

rating r_g. It means that the evidence of p_i does not support p_i to get a higher rating. This experimental result also agrees with Proposition 4.4.1, however, it can still guarantee a reward and some increment of reputation.

Under a dynamic setting, the experiment allows agents to leave and rejoin the system. Trustee agents can reset their reputation values on rejoining, and truster agents are allowed to give more arbitrary ratings. As can be seen from Table 4.3, the overall reputation of trustee agents in our approach is higher in both cases of low ratings and high ratings. It means agents have more satisfaction with our distribution mechanism.



(a) Random high ratings



(b) Random low ratings

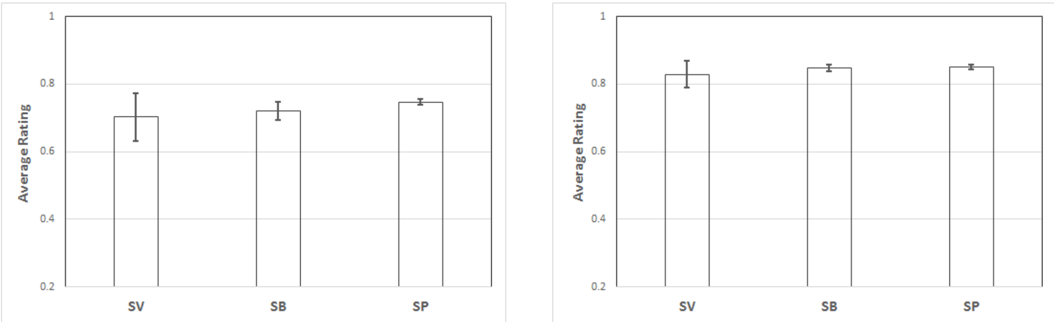
Figure 4.9: Rating distribution and reputation under randomised rating patterns

4.5.4 Stability evaluation

The dynamic nature of open systems can make the reputation unstable. This experiment investigates two (external and internal) factors that influence the performance stability of groups. The external factor to consider is the biased ratings of truster agents, and the internal factor is the variations of group composition. Biased rating is a serious issue of trust and reputation systems in open systems. It leads to inaccurate trust values and wrong decisions. The ability to resist the impact of biased ratings is measured via the stability of reputation under biased ratings. We can assume that under the same composition and setting, the performance of members is supposed to be the same. The results of three common biased rating patterns, namely, decreasing (or negative bias), increasing (positive bias), and random ratings are shown in Figure 4.7, 4.8, and 4.9, respectively. The results show the strong ability to reduce the impact of biased ratings of SP. The reason for this is the unaccountability factor that helps members to resist to an impulsive change of rating. Even though the reputation can still be manipulated under the long-term bias (even p_i's performance is stable in groups).

For the internal factor, we examine the stability of reputation respecting different service compositions patterns. There are two main patterns considered: (1) groups of the same service composition but with different members, (2) groups of the same fellow members (the same service composition and trustee agents who always work together). Suppose that some certain combination of some members (or services) always give the same fixed output result (good or bad), e.g., a group containing $\{p_i, p_j, p_k\}$ always receive good ratings. This setting allows acquaintances to separate and form a new group with other trustee agents.

Figure 4.10 shows the results of agent groups with the same service composition. In the case of same service composition, the result indicates that the SB method has the smallest rating variation amongst the three (see Figure 4.10a). It is because the



(a) Same service composition (b) Same member composition

Figure 4.10: Rating distribution in different service composition patterns

SB method consists of fewer variables since it considers only the service types in composition whilst ignoring the difference in member composition. However, in the other case, the result shows that our approach has the least variation in both rating and reputation of same member composition (see Figure 4.10b). These results indicate that our SP method can give better protection for some members that always work together in groups. They are less likely to receive unexpected ratings than other approaches because they have a higher relevancy factor.

4.6 Summary

This chapter focused on the problem of inconsistent data rating in the context of agent groups. An efficient trust evidence management method for agent groups, GEM, was proposed. GEM distributes the group rating fairly amongst group members by using evidence provided by group members. By introducing the evidential performance and the unaccountability factors, the rating distribution mechanism can reduce the conflict of individual interest and improve the satisfaction of rated members.

The integration of GEM into any existing reputation system is simple as it does not have special requirements. With GEM, reputation systems can better maintain the consistency of rating evidence in every transaction including ones with agent groups.

Consequently, it can enhance the accuracy of trust/reputation systems by reducing the reputation lag problem and the impacts of biased ratings.

The work of this chapter has been published in the Computer Journal Nguyen and Bai (2017b) and Nguyen and Bai (2016).

Chapter 5

DBATE: a trust evaluation model for agent groups

This chapter discusses the third component of the trust management stack introduced in Chapter 1. Specifically, it proposes a trust evaluation model, DBATE, which aims to overcome the shortcomings of current trust evaluation models when evaluating agent groups, i.e., how to accurately capture the trustworthiness of a group with dynamic behaviour. DBATE uses a dynamic Bayesian network approach to address the dynamic behaviours of complex agent groups. The experimental results show that DBATE can achieve higher accuracy than other approaches when evaluating agent groups in certain settings.

5.1 Overview

In the real world, we frequently witness trustee agents forming groups to deliver sophisticated services to truster agents (consumers). Such agent groups introduce more complexity to the trust evaluation compared with the individual agent. The performance of agent groups may be affected by several factors, e.g., the changing of group members (an internal factor), or changing environmental conditions (external factors). All these internal and external factors may happen anytime to agent groups, thereby causing the "dynamic behaviours" of agent groups. The challenge is how to capture these dynamic behaviours to improve the accuracy of the trust evaluation of agent groups. It is essential for a trust evaluation model to take extra care of the aforementioned factors.

Recently, incorporating contextual information has received considerable attention due to the promising results in many situations, e.g. mining computational trust (Y. Zhang & Yu, 2012) and recommender systems (Adomavicius & Tuzhilin, 2011; Wang, Li & Liu, 2015). Proper use of additional data has shown the effectiveness of improving the accuracy of the evaluation (H. Yu, Shen, Leung et al., 2013; Hoelz & Ralha, 2016).

In this chapter, I introduce a novel trust evaluation model using a dynamic Bayesian network approach for building a flexible and responsive trust evaluation model. Specifically, it allows user-defined criteria of trust combined with contextual data obtained from the current transaction. The data is mapped to observation features and then put into the trust prediction model, which is trained by a combinatorial method.

The contributions of this work are (1) a trust evaluation model (DBATE) is proposed to address the dynamics of agent groups; (2) DBATE is applicable for either individual agents or agent groups; (3) it demonstrates the potential to relax some assumptions.

The rest of the chapter is organised as follows. Section 5.2 reviews some recent approaches on trust evaluation. The DBATE approach is presented in Section 5.3. The experiments and discussion are outlined in Section 5.4. Finally, the summary is given in Section 5.5.

5.2 Related Work

There are already several models to evaluate the trustworthiness of potential partners (see Chapter 2). Nevertheless, these models contain two major drawbacks when evaluating the complex targets. Firstly, trustee agents are treated rigidly as individual agents. Uncertainty often leads to the adoption of several simplifying assumptions, such as the idea that each trustee agent can only handle one task at a time (single-tasking) and that the number of requests does not affect the performance of trustee agents, etc. However, these conditions may not hold in some application domains (H. Yu, Shen, Leung et al., 2013; Nguyen & Bai, 2016). The work of Tong and Zhang (2009) is one of the first works on trust of agent groups. The advantage of this work is its simplicity. It can use the current approaches to individual trust to calculate the trustworthiness of a group by the average reputation of its members. However, the limitation is it leaves out the dynamic behaviour of group members and contextual information of groups especially when they work together.

The second drawback of most current approaches is the lack of the ability to discern between true trustworthiness and causes of the performance observed. Chapter 2 has mentioned two main causes of risks, i.e., intention and unknown influential factors. The intentional risks are the risks posed by the deceiving acts of trustees. Meanwhile, risks resulting from unknown factors are ones that are not expected to happen by both truster and trustee agents. Unfortunately, existing trust evaluation models often combine all these causes into one concept: uncertainty. By doing so, it can simplify the models, but it may also affect the evaluation accuracy.

The studies in Jøsang and Haller (2007) and Huynh et al. (2006b); Wang et al. (2011) are some typical trust evaluation models designed for evaluating individual trustee agents. These studies are applicable to a single entity whose performance is assumed to be influenced by trustee agents' intentions. In an open MAS, besides intention, there are also several factors that may affect the performance of trustee agents. Relying on only feedback from truster agent may not be able to accurately capture the trustworthiness of trustee agents in some certain situations. For example, if agent p_i has a good reputation and on a certain day p_i is ill, there is a higher chance that all the transactions with p_i may fail. Without knowing the condition of p_i, the trust evaluation models cannot give the accurate trustworthiness value of p_i.

The contextual information can be useful in such situation. For example, in Korovaiko and Thomo (2013), the study links features, such as age, gender, and service categories, and then applies machine learning techniques to learn about provider's trustworthiness. Wang et al. (2015) utilised the relationship information among agents for trust evaluation. However, the difficulty of verifying the relationship information is one of its drawbacks. In Regan et al. (2006) and W. L. Teacy et al. (2012), the authors demonstrated how to apply the Bayesian network approach (BNA) to tackle the changing behaviours using observed features. However, the creation of the trustee network is not computationally efficient. Also, finding the similar context for training parameters would fail if other contextual information also changes frequently.

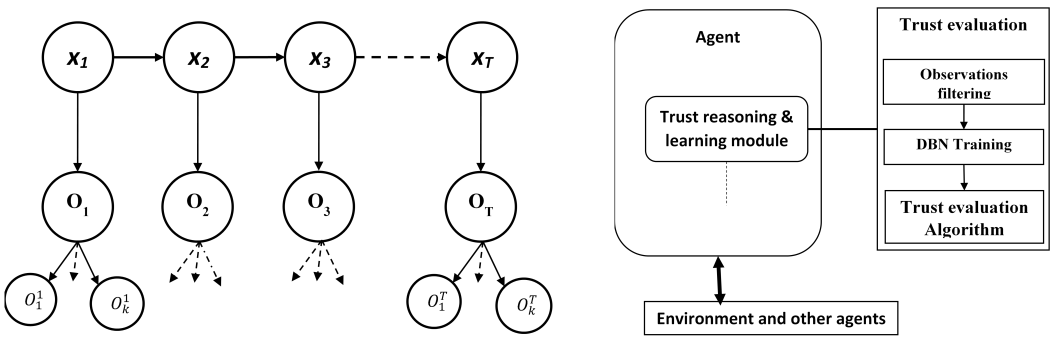
In this study, I also used contextual information for DBATE. The success of the Dynamic Bayesian Network (DBN) has motivated me to choose it as a tool for agent group trust evaluation. By using DBN, DBATE can better capture the dynamic behaviours of agent groups, while it can have faster training.

5.3 The DBATE trust evaluation model

The following example describes a situation in which truster agent c_m needs a service from a potential trustee agent p_n at time t (now). c_m wants to know whether the probability that p_n will deliver the expected service at the time $(t + 1)$ if p_n accepts the request of c_m. Agent c_m believes that the current states of p_n may affect the future outcome, e.g. if p_n is in the trustworthy state, it is more likely that c_m will receive good service. Since the state of p_n is hidden to c_m and there are continuous changes of contextual data, the described process is suitable for space-time modelling with discrete-time, i.e., time t is an integer (see Appendix A).

To evaluate the trustworthiness p_n, we can firstly find the hidden state of p_n at the time t. Thus, it can be mapped to an inference problem of DBN introduced in Section A.3. DBATE treats the contextual transaction data as a source for feature selection and observation, e.g. environment, targeted partner, or agent's intrinsic properties. When there are multiple features, it is convenient that DBN allows more than one observation variable at each time step (see Figure 5.1a). I will go through some important definitions and assumptions.

It is imperative to understand that more data about the group could help improve the accuracy of evaluation. Therefore, I proposed a method based on a variation of the dynamic Bayesian network (DBN) which makes use of multiple observations obtained from the transaction context. In this research, I adopted the Multiple Observation Hidden Markov Model (MOHMM). The basic idea of the MOHMM is that each state can have a set of observations instead of just a single observation (refer to the base model demonstrated in Figure 5.1a).



(a) DBN MOHMM

(b) Trust evaluation module

Figure 5.1b illustrates the building blocks of the DBATE trust evaluation module adapted from Figure 3.1. It consists of three components, i.e., observation filtering, parameters training, and a trust evaluation algorithm.

5.3.1 Definitions and assumptions

A truster (service provider) in this study can be an individual agent or an agent group. A provider receives requests from consumers, processes these requests, and delivers the service to consumers (trustee agents). An agent group is a combination of several agents, but it can be identified by a single identity (see Section 2.1.1). For example, in online marketplaces, customers make requests to a seller called "EZ Booking". The name represents a service provider. But "EZ Booking" can also be seen as an agent group who handle different tasks of booking, i.e., from receiving orders to delivering tickets.

A *transaction record* between consumer c_m and provider p_n is represented by a 3-tuple $\tau_{(c_m,p_n)} = \{t_r, t_d, R_{c_m,p_n}\}$, where t_r, t_d, and R_{c_m,p_n} represent the timestamp of the request, the timestamp of the delivered service, and the transaction outcome (see Definition 5.3.1), respectively.

Records are stored in the local database of all participants (see Definition 3.3.3). Provider p_j has $D_{p_n} = \{\tau^i_{c_k,p_n}\}$, where c_k is a consumer in the system. The transaction records are public and verifiable.

Definition 5.3.1. A *transaction outcome* of consumer c_m and provider p_n is a value of a predefined value set, which indicates the degree of satisfaction of c_m to the performance of p_n as an interaction partner.

The outcome is denoted as a set of discrete values $R = \{R_1, R_2, \ldots, R_L\}$. Most studies consider the set as rating values (feedback) of consumers to the performance of providers. For example, eBay uses $R = [-1, 0, +1]$ to indicate negative, neutral, positive ratings, respectively. Feedback can be biased or subjective. However, I assume that feedback is accurate in this study.

Definition 5.3.2. The *multiple sequence observation* of a consumer is a set $O = \{O^{(1)}, O^{(2)}, \ldots, O^{(T)}\}, O^{(t)} = \{o_k^t \mid t = 1..T, k = 1..L\}$, where o_k^t is the t^{th} symbol of the k^{th} observation variable ($o_k^t \in V_k$).

The observations are from the contextual data of transactions. Each selected feature w_k can produce a set of symbols V_k, where L is the number of features (see Appendix A). The number of observations depends on the visible scope of the truster agents (evaluators). For example, if c_i orders a pizza using the on-line ordering system, the only information c_i may know is the ratings of the pizza shop from previous orders. At the same time, another customer c_j is at the shop, where he can have the information about how the current staff works, the number of customers in the queue, etc. Obviously, c_j can have more evidence to evaluate the quality of the service at this moment compared to c_i.

Each observation is a set of selected features (see Figure 5.1a). These features are expected to be able to differentiate between the levels of the transaction outcome. The details of processing the observations are discussed in Subsection 5.3.3.

Definition 5.3.3. The *hidden states* of providers are random variables denoted by $X = \{x_1, x_2, \ldots, x_T\}, x_t \in S, t = 1..T$, where $S = \{S_1, S_2, \ldots, S_N\}$ is a set of N distinct states of the Markov process.

At each time step, a provider may change their state from S_k to S_l $(k, l = 1..N)$. These states follow the Markov process and they are corresponding to the observation sequences. In the experimental section, I demonstrate a system in which trustee agents have two states, however, it is more flexible to allow implementers to decide how many states are needed based on specific requirements.

5.3.2 Trust value calculation

The Viterbi algorithm (see Appendix A) can be used effectively for estimating the hidden state sequence that is the best explanation for the observation. However, this study only focuses on the state at the end of the observed sequence, i.e., $P(s_t = S_k \mid O, \lambda)$, because the current state s_t of provider is more related to future performances. This problem is also known as filtering and can be handled by using Bayesian rules and recursion.

If O_{w_i} is the sequence produced by feature w_i, the Bayesian rule gives:

$$P(x_t = S_k|O_{w_i}, \lambda) = \frac{P(x_t = S_k, O_{w_i}, \lambda)}{P(O_{w_i}, \lambda)} \tag{5.1}$$

The $P(x_t, O_{w_i}, \lambda)$ is the joint probability that O_{w_i} is observed and the state x_t of the provider is S_k. Since λ is independent to O_{w_i}, the equation can be simplified by removing the λ. Equation 5.1 can be rewritten as:

$$P(x_t = S_k|O_{w_i}) = \frac{P(x_t = S_k, O_{w_i})}{P(O_{w_i})} \tag{5.2}$$

While the probability of $P(O_{w_i})$ can be calculated using the following equation.

$$P(O_{w_i}) = \sum_{k=1}^{T} P(O_{w_i}|x_t = S_k)P(x_t = S_k) \tag{5.3}$$

Applying the following recursion

$$P(x_t = S_k) = \sum_{k=1}^{T} P(x_t = S_k|x_{t-1}).P(x_{t-1}) \tag{5.4}$$

to Equation 5.3, we then have:

$$P(O_{w_i}) = \sum_{k=1}^{T} \pi_{x_1} B_{x_1, o_{w_i 1}} A_{x_1, x_2} \ldots A_{x_{T-2}, x_{T-1}} B_{x_{T-1}, o_{w_i T-1}} \tag{5.5}$$

A and B are the state transition probability matrix and the observation probability matrix described in Appendix A. Obviously, the described solution is not efficient regarding computation. Therefore, the forward algorithm adapted from the dynamic programming as seen in Murphy (2002) can be used. When

$$\alpha_t(x_t) = P(x_t, O_{w_i}) = \sum_{x_{t-1}} P(x_t, x_{t-1}, O_{w_i})$$

is the joint probability of the observation $O_{w_i} = \{o_{w_i1}, o_{w_i2}, \ldots, o_{w_it}\}$ and $x_t = S_k$ at time t, then using the chain rule to expand $P(x_t, O_{w_i})$, we have:

$$\alpha_t(x_t) = \sum_{x_{t-1}} p(o_{w_it}|x_t, x_{t-1}, o_{1:t-1})p(x_t|x_{t-1}, o_{w_i1:t-1})p(x_{t-1}, o_{w_i1:t-1}) \tag{5.6}$$

Because o_t is conditionally independent to all except x_t, in-turn x_t is conditionally independent to all except x_{t-1}, Equation 5.6 can be simplified as:

$$\alpha_t(x_t) = p(o_{w_it}|x_t) \sum_{x_{t-1}} p(x_t|x_{t-1})\alpha_{t-1}(x_{t-1}) \tag{5.7}$$

Since $p(y_t|x_t)$ and $p(x_t|x_{t-1})$ are given by the model's emission distributions and transition probabilities, we can calculate $\alpha_t(x_t)$ from $\alpha_{t-1}(x_{t-1})$ while avoiding exponential computation time caused by the recursion. Then, the prediction for the outcome of the next transaction at time $(t+1)$ can be computed with just one more step by summing up all paths from state s_t to R_l.

$$P(r_{t+1} = R_l) = \sum_{k=1}^{N} P(x_t = S_k)A_{S_k,R_l} \tag{5.8}$$

The trust value $T(c_m, p_n)$ of consumer c_m over p_n is considered as the probability that p_n will provide the acceptable outcome $R_x (R_x \in R)$, i.e $P(o_{t+1} \geq R_x)$. It also indicates that the model allows consumers to define their satisfaction criteria for trust values. Therefore, the trust value can be calculated by summing up all the probability of corresponding symbols shown below.

$$P(o_{t+1} \geq R_x) = \sum_{R_j \in R, R_j \geq R_x} P(o_{t+1} = R_j) \tag{5.9}$$

Since different consumers may have different satisfaction criteria, DBATE can generate different trust values with the same set of evidence. This result is different with many existing trust/reputation models which are based purely on evidence whilst ignoring truster agents' preferences. The trust value of c_m to p_n can be defined as the normalised value shown below:

$$T_{(c_m,p_n)}^{R_x,w_i} = \frac{P(o_{t+1} \geq R_x)}{\sum_{R_j \in R} P(o_{t+1} = R_i)} \tag{5.10}$$

As each observation sequence can generate a different value of trust, after normalising the trust values generated from different sequences, the trust value can be obtained using Equation 5.11.

$$T_{(c_m,p_n)}^{R_x} = \frac{1}{L}\sum_{i=1}^{L} T_{(c_m,p_n)}^{R_x,w_i} \tag{5.11}$$

Nevertheless, the accuracy of the prediction also depends on the trained parameters which are discussed in the next section.

5.3.3 A combinatorial training method for DBATE

The training process is illustrated in Figure 5.1. It consists of three stages: observation filtering, feature selection, and parameters training.

Contextual data can be sorted into three broad categories (refer to Section 2.2.3): the environmental factors (e.g. online market, social networks, centralised or distributed), targets (e.g. their services, feedback, relationship, etc.) and temporal factors. The agent preferences have been used in defining the trust value in the previous section. The data obtained from the environment and partners are used as feature sources.

Figure 5.1: Feature selection and training process

In the observation filtering stage, the consumer will filter out irrelevant observation data, including the removing of unfair testimonies or adjusting the subjectivity of third-party testimonies. As data is assumed to be accurate, this step can be skipped. The feature selection stage further optimises the relevancy of features and improves the efficiency of computation in the training stage.

(A) *Feature selection*

To select the most relevant features from contextual data, I adopted a filtering method introduced in L. Yu and Liu (2004). If the potential features extracted are represented as $\Omega = \{w_1, w_2, \ldots, w_L\}$, then analysing the transaction database D_{p_n} gives us the probability distribution $P = \{p_1, p_2, \ldots, p_L\}$ corresponding to L outcome symbols of R. The uncertainty of the transaction is represented by the entropy:

$$H(D_{p_i}) = \sum_{j=1}^{L} P(tr_j) log \frac{1}{p_i} \tag{5.12}$$

The mutual information quantifies the a mount of information obtained about one random variable through the other variable. The mutual information (information gain) of selecting one feature w_j is:

$$I(D_{p_i}, w_j) = H(D_{p_i}) - H(D_{p_i} \mid w_j) \tag{5.13}$$

By comparing the information gain, the most relevant features can then be selected by setting a threshold. The dependency between features then can be filtered using measuring the distance correlation (Székely, Rizzo, Bakirov et al., 2007).

(B) *Training*

After selecting features, I use the Baum-Welch algorithm for training parameters of each feature, which is summarised below.

Initiate $\lambda = (A, B, \pi)$ with random values or prior information.

Forward procedure: Let $\alpha_i(t) = P(Y_1 = y1, \ldots, Y1 = y_t, X_t = i \mid \lambda)$ be the probability of seeing $y_1, y_2, \ldots, y_t$ and being in state i at time t.

- $\alpha_i(t) = \pi_i b_i(y_1)$
- $\alpha_j(t+1) = b_j(y_{t+1}) \sum_{i=1}^{N} \alpha_i(t) a_{ij}$

Backward procedure: Let $\beta_i(t) = P(Y_{t+1} = y_{t+1}, \ldots, Y_T = y_T \mid X_t = i, \lambda)$ be the probability of the ending partial sequence $y_{t+1}, \ldots, y_T$ given starting state i at time t.

- $\beta_i(T) = 1$
- $\beta_i(t) = \sum_{j=1}^{N} \beta_j(t+1) a_{ij} b_j(y_{t+1})$

Update the temporary variables using Bayes' theorem as:

$$\gamma_i(t) = \frac{\alpha_i(t)\beta_i(t)}{\sum_{j=1}^{N} \alpha_j(t)\beta_j(t)} \tag{5.14}$$

which is the probability of being in state *i* at time *t* given the observation sequence Y and the parameter θ.

$$\xi ij(t) = \frac{\alpha_i(t)a)_{ij}\beta_j(t+1)b_j(y_(t+1))}{\sum_{i=1}^{N}\sum_{j=1}^{N} \alpha_i(t)a_{ij}\beta_j(t+1)b_j(y_{t+1})} \tag{5.15}$$

Repeating update λ as follows.

(a) The expected frequency spent in state i at time 1 can be calculated by using the equation below:

$$\pi_i^* = \gamma_i(1) \tag{5.16}$$

(b) The expected number of transactions from state i to state j compared to the expected total number of transactions away from state i.

$$a^*_{ij} = \frac{\sum_{t=1}^{T-1} \xi_{ij}(t)}{\sum_{t=1}^{T} \gamma_i(t)} \tag{5.17}$$

(c) The expected number of time that the output observations have been equal to v_k while in state i over the expected total number of times in state *i*.

$$b^*_i = \frac{\sum_{t=1, y_t=v_k}^{T} \gamma_i(t)}{\sum_{t=1}^{T} \gamma_i(t)} \tag{5.18}$$

5.4 Experiments

Several experiments have been conducted to evaluate the effectiveness of DBATE in addressing the trust evaluation of agent groups regarding dynamics. For better results, agent and groups are designed with more practical settings, e.g. agents with limited processing capability, multitasking agents, and group structures. The dynamics of groups are mainly caused by the variations of group members. The prediction accuracies are compared with two other approaches, i.e., an HMM-based trust adapted from Moe et al. (2008) (using feedback without contextual data) and evidence-based trust (EBT) (Wang & Singh, 2010). Alongside this, I also discuss some potential applications of DBATE such as relaxation of some common assumptions or assisted decision making in some real-world scenarios.

5.4.1 Experimental settings

The system was populated with 500 consumer agents and 60 provider agents. Agents are generated with different profiles and simulated behaviours. For simplifying purposes, each provider provides one in ten predefined simple services. Consumers can request for composite services which are groups of simple services. These groups are formed by inviting suitable providers, and their members can dismiss them after the job is done.

The number of requests is generated randomly in each round. Providers are set with two latent states, i.e. $S = \{\text{trustworthy}, \text{untrustworthy}\}$, and also two levels of outcome $R = \{0, 1\}$ representing the unsatisfied and satisfied feedback, respectively.

Some typical features obtained from the context are selected, e.g. feedback, the number of tasks, and group members. To extract the capability feature, we can monitor the relationship between the time of making requests and the time of service delivery, and other activities in-betweens as shown in Figure 5.2.

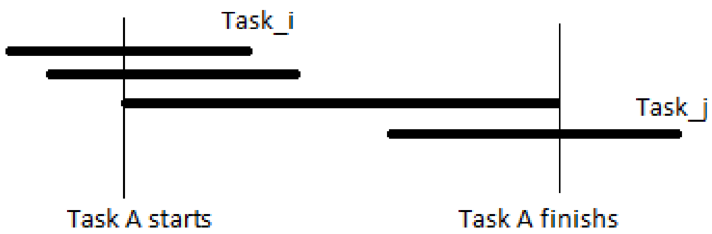


Figure 5.2: Task handling in multi-tasking mode

Providers, who can accept new requests before finishing the old ones, are considered as multi-tasking. In practice, multi-tasking models are ubiquitous and preferable as they can improve group productivity significantly. For example, in a group, when one agent (a group member) is processing requests, another agent can handle other tasks rather than just waiting. In this setting, multi-tasking is enabled for groups of agents, whilst using the single-tasking model for individual trustee agents.

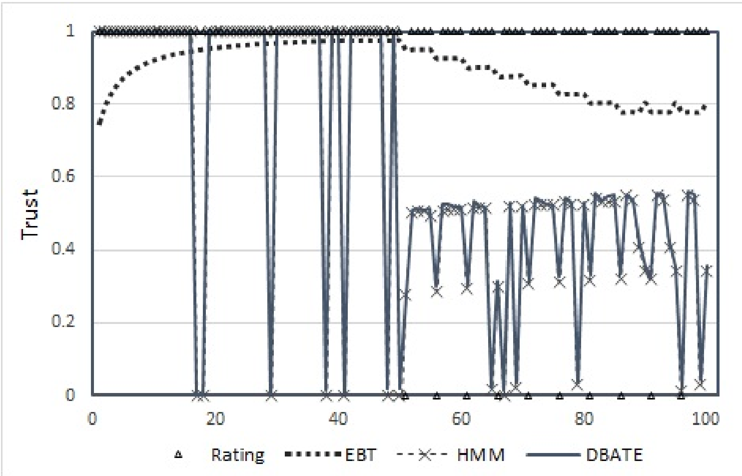
Consumer agents should understand that delegating a task to a provider may change the provider's current state and affect the future outcome. The calculation for the number of concurrent tasks (ct) when c_m makes a request to p_n (supposed that D_{p_n} already has $l - 1$ records) is given by:

$$ct^l_{p_n} = 1 + \left|\{\tau_k \in D_{p_n}\}\right|; tr_l \leq tr_k \leq td_l \tag{5.19}$$

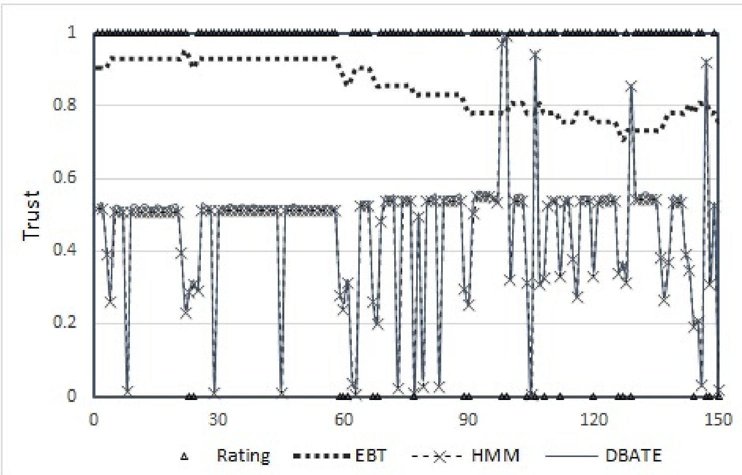
, where τ_k is a request for the k^{th} transaction in D_{p_i}, td_m denotes the delivery timestamp of the m^{th} transaction.

Initially, providers had no transaction record and transactions were carried based on a default trust value. After some transactions, the change of trust values for each entity is monitored, even in the case of no incoming requests.

5.4.2 Results and Discussions

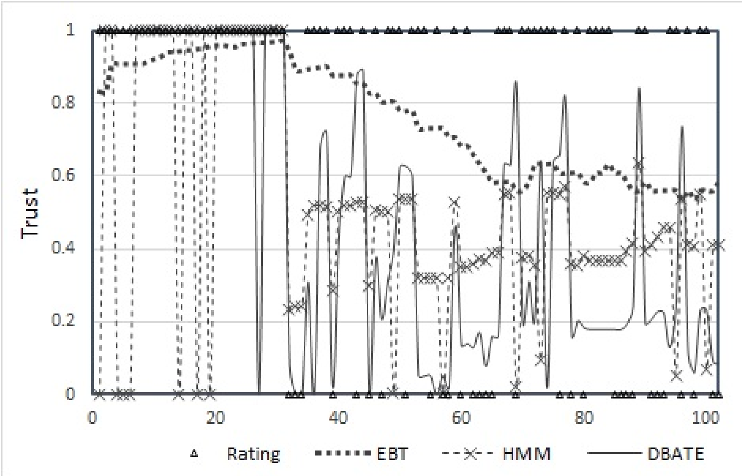


(a) Patterned deceits

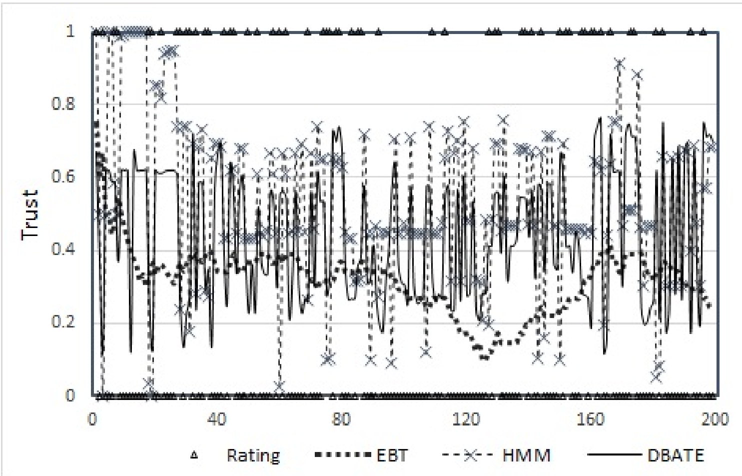


(b) Unpatterned deceits

Figure 5.3: EBT, HMM, and DBATE under deceiving



(a) Group of fixed members

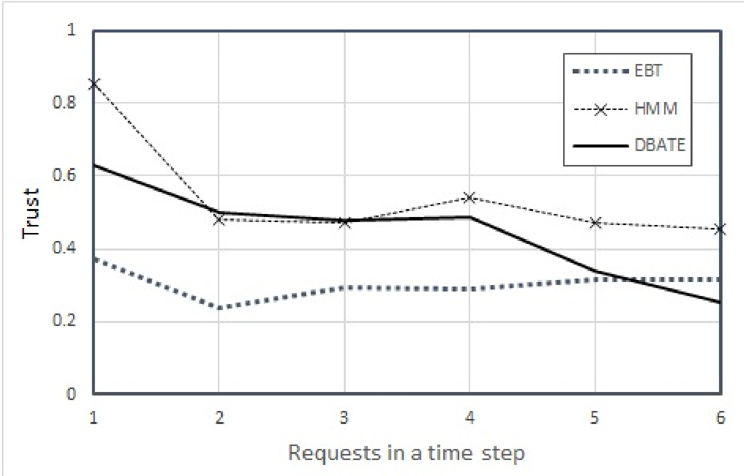


(b) Group of changeable members

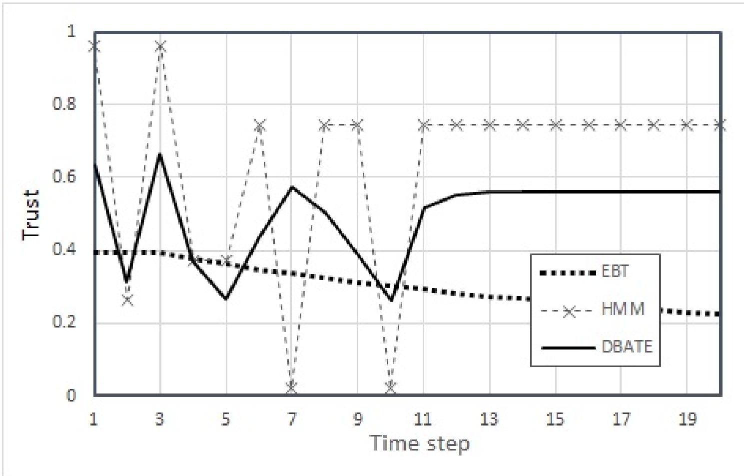
Figure 5.4: EBT, HMM, and DBATE with different member patterns

Experiment 1: Single observation

The first experiment investigated the trust evaluation for individual trustee agents considering only one feature. The setting was simple: The rating was used as an observed feature. Initially, providers must have some good ratings to be selected for the interactions. Then, providers start to deceive. The evaluated providers were tested in two scenarios, i.e., patterned deceit and random deceit. Under the patterned deceit, the providers cheated once in every a few transactions at a fixed rate. Whereas, in the second scenario, providers deceived more randomly.



(a) Average trust value under different concurrent tasks



(b) Trust value in a idle period

Figure 5.5: EBT, HMM, and DBATE under different number of concurrent tasks

Figures 5.3a and 5.3b illustrate the comparison results of the two cases between DBATE, EBT, and HMM. The graphs show that EBT responded slower in comparison to DBATE and HMM. On another hand, HMM and DBATE learned and had good responses to the deceit pattern. Since there is only one observation, the performance of HMM and DBATE are quite similar in both scenarios.

The statistical data of the experiment is summarised in Table 5.1. The accuracy of DBATE, HMM, and EBT are measured by the *false positive rate* (the transaction is unsuccessful, but an algorithm predicted that it would be successful). The statistics show

Table 5.1: Accuracy (false positive) in different scenarios

Observation	Setting	Trans	EBT	HMM	DBATE
One	Patterned deceit	20	1.0	1.0	1.0
		50	1.0	0.67	0.67
		100	1.0	0.42	0.42
	Randomised deceit	20	1.0	1.0	1.0
		50	1.0	0.726	0.726
		100	1.0	0.612	0.612
Multiple	Fixed members	20	1.0	1.0	1.0
		100	1.0	0.272	0.265
		150	1.0	0.163	0.183
	Changeable members (a)	20	0.486	1.0	0.857
		100	0.388	0.588	0.519
		200	0.282	0.469	0.437
	Changeable members (b)	20	0.486	0.785	0.692
		100	0.388	0.495	0.367
		200	0.282	0.316	0.219

that HMM and DBATE outperformed EBT in predicting the untrustworthy behaviours. The accuracy decreased in cases of unpatterned deceit compared to the patterned one. The performance of DBATE in the unpatterned case can be improved by increasing the number of training iterations.

Experiment 2: Multiple observations

This experiment investigated the performance of the three methods with more complicated settings. Two distinct patterns of composite service providers were observed: (1) groups of fixed members, and (2) groups with changeable members. More observable features are used including the multi-tasking mode. As can be seen, the setting of the second case introduces more dynamics than the first one. The results are shown in Figure 5.4a and 5.4b and the statistical results are presented in Table 5.1 with fixed members and changeable members.

Table 5.1 shows that DBATE has a higher prediction accuracy when groups have fixed members. The statistics of the fixed members in the table also shows that both methods predicted poorly at the beginning. It was due to the inadequate training samples. Later, the learning ability of HMM and DBATE helped to improve the accuracy of predictions significantly in the case of fixed member groups. We can see that the accuracy of EBT is lower than the other two in this case. Furthermore, EBT does not show much difference throughout the number of transactions like HMM or DBATE.

In the second part of this experiment, we observed groups with changeable members. To further explore the performance of DBATE, we also examined two settings of parameters: one with average training iterations and one with intensive training as demonstrated in Table 5.1 with changeable members (a) and (b), respectively. The test with changeable members experienced higher error rates of both algorithm compared to the counterpart of fixed members. However, we can see the considerable improvement of DBATE (0.219 to 0.437) in the intensive training mode.

I then examined the relations between the performance of providers and the number of requests during the runtime as well as idle periods. Figure 5.5a demonstrates the average trust values under different numbers of concurrent tasks (each time step). As can be seen, EBT and HMM show a small response to other contextual data. On the other hand, DBATE shows the trend of trust: the trustworthiness reduces when the number of the concurrent tasks increases. A following significant result is shown in Figure 5.5b when providers enter idle states (the state with no ongoing processing task or new incoming requests). From Figure 5.5b, it can be seen that when a provider had no incoming requests, the trust values calculated by HMM and EBT stay the same (in the case of no time discounting factors), or it gradually reduced after some periods of time. On the other hand, DBATE keeps decoding little while longer based on the observed features. The result indicates that the trust value can increase during the idle period rather than decrease strictly like some other approaches.

Discussions

Applications on contextual decision-making: DBATE helps to improve the accuracy of trust evaluation as it can learn about potential features affecting the performance. The demonstration of trust value in idle period shows the effectiveness of DBATE compared with other approaches. One potential application is contextual decision making. By observing the contextual data, DBATE can suggest when agents should interact with the targets.

Alongside this, in some situations, deciding to go for the provider with a lower reputation could be better than choosing highly reputable ones. As a consequence, some agents with lower reputations also have higher chances of making transactions. This phenomenon can help reduce the local optimisation problem, i.e., only the highest reputable agents are selected. Thus, it can also reduce the reputation damage problem caused by an excessive number of requests as described in H. Yu, Miao et al. (2013).

Assumption relaxation: The uncertainty of internal/external activities of providers led to the adoption of several simplifying assumptions, e.g. single-tasking and unlimited processing capability (UPC) (H. Yu, Shen, Leung et al., 2013). Many trust evaluation models have worked based purely on ratings. However, by considering additional information, e.g., timestamps from transaction records, Equation 5.19 can confirm the multi-tasking feature in a non-heuristic way. As UPC and single-tasking assumptions can be relaxed, it enables DBATE to work in more realistic environments.

Stationary VS. Non-stationary DBN: DBATE can work efficiently when there are sufficient amounts of data for training. However, most Bayesian models work under the stationary assumption. We are aware of the degraded performance in non-stationary environments and will investigate the problem in future work.

5.5 Summary

This chapter presented DBATE, a novel trust evaluation method based on the dynamic Bayesian network for agent groups. It focused on addressing the dynamic behaviours of agent groups to improve the accuracy of trust evaluation. To the best of my knowledge, it is one of the first trust evaluation approaches for agent groups by considering dynamic features of agent behaviours and the environment.

DBATE is a flexible trust evaluation model. It allows trustee agent to define evaluation criteria. The features can be selected according to specific environments. The experimental results show that DBATE responds well to trust dynamics of agent groups. It outperformed other approaches with intensive training. Furthermore, it illustrates the potential to relax some assumptions when the relevant data can be observed or inferred.

With DBATE, this work has provided MASs with an effective trust evaluation method that is capable of evaluating the trustworthiness of agent groups. Furthermore, as agent groups are the generalised case of individual agents, DBATE is also able to evaluate individual agent.

The related work of this chapter has been published in Nguyen et al. (2016).

Chapter 6

Conclusions

This chapter summarises the findings of this thesis in trust management for complex agent groups in open dynamic environments, which is an important and challenging problem in MAS research. A management stack of three components was developed. It takes into account three main aspects of trust in agent groups, i.e., trust establishment in group formation, evidence management for agent groups, and group trust evaluation.

I summarise the contributions of this thesis in Section 6.1. The limitations of the stack and possible directions for the future work are discussed in Section 6.2.

6.1 Research Contributions

Agent groups are important parts of MASs due to their ability to solve more complex tasks that cannot be done by a single agent. However, the presence of agent groups in the systems brings not only benefits but also challenges. As the applications of agent groups become more and more popular, the demand for comprehensive trust management also increases. Meanwhile, current trust management research is insufficient to address several issues related to agent groups. Therefore, the objectives of this thesis are to address the shortcomings. Three research questions have been identified to investigate trust in three fundamental stages of agent groups, i.e., group formation, formed group performance, and group disbandment. Consequently, a trust management stack for agent groups, which consists of three components, i.e., CoTrust, DBATE, and GEM, was devised.

This research contributes to the field in the following ways:

- **It extends the applicability of trust in MASs:** individual agents and agent groups are two research objects defined in the early days of computational trust in MASs. While the former has received excessive studies, the achievement in trust management for agent groups is insufficient to address several issues associated with agent groups. Therefore, the works of this thesis not only addresses the increasing demand for trust management of agent groups when they are becoming ubiquitous, but it also broadens the applicability of trust to another important part of MASs.

- **It equips agents in MASs with an efficient trust establishment method regarding group formation:** By analysing the essential factors during the formation of groups, this thesis developed a trust establishment method, CoTrust, that helps agents to gain trust from other agents actively. With CoTrust, agents can achieve a better success rate of requests and higher satisfaction regarding cooperation.

- **It enhances the robustness of trust/reputation systems in MASs:** This research has pointed out an issue related to evidence in agent groups. GEM is a group evidence management method which aims to maintain the consistency of ratings in systems that contain agent groups. GEM benefits any trust/reputation that operates using rating evidence.

- **It enables agent group trust evaluation in MASs:** dynamism is one characteristic of agent groups and the open MASs which seriously affect the accuracy of trust evaluation models. DBATE is a novel trust evaluation model that is suitable for evaluating agent groups.

6.1.1 CoTrust

CoTrust investigates the roles of trust in group formation. Agents in a cooperation context need mutual trust in order to work together as a group. Consequently, the requirements for a trust mechanism in group formation are:

1. It should support trust evaluation of both parties.

2. Agents should have a mechanism to establish trust relationships with potential agents when initiating requests for cooperation.

3. It should be able to improve the satisfaction of the cooperation.

To achieve that, CoTrust consists of a protocol and a preference reasoning mechanism for requesters when making proposals to potential partners. As the protocol adopts a stable matching game, it can guarantee that matched agents for cooperation are satisfied with the proposal. Alongside this, the mechanism also brings incentives for agents to behave appropriately ad in a trustworthy manner.

6.1.2 GEM

Agent groups are formed to solve complicated tasks of truster agents. However, group members are not rated individually by the truster agents. Namely, the feedback for a service delivered by an agent group is not for a specific member of the group. Therefore, it requires a proper method to guarantee the fairness and accuracy of the feedback.

GEM, an agent group evidence management method, was proposed to solve the rating distribution of groups to their members. GEM uses the evidence-based approach combined with evidential performance to assess the performance of each member in the group. GEM can achieve the following objectives:

1. The members who have no previous transaction record receives the same rating as the group.

2. The members with experience receive the rating based on evidential performance adjusted by the accountability factors.

3. The rating distribution is fair regarding proof of evidence.

With GEM, agents who participate in agent groups receive individual feedback, which can then be used for trust evaluation. Therefore, it can reduce the reputation lag problem. The implementation of GEM is simple and can be integrated into any existing trust/reputation systems to enable trust evaluation to work in systems which consist of agent groups.

6.1.3 DBATE

The behaviours of agent groups are more dynamic compared to individual agents. It is hard to capture the trustworthiness of agent groups using existing trust models. To this end, DBATE, a trust evaluation model, was proposed to overcome the shortcomings of current trust evaluation models when evaluating agent groups.

DBATE combines agent-defined criteria with contextual information related to the groups, in order to generate more accurate trust values. DBATE can also work for evaluating individual agents as individual agent can be considered as a group consisting of one member.

DBATE is a flexible trust evaluation model. It allows trustee agent to define evaluation criteria. The features can be selected according to specific environments. It utilises the contextual data to reason with the trustworthiness of the targeted group using a dynamic Bayesian network approach.

The experimental results show that DBATE can achieve better accuracy compared with other trust evaluation models in dynamic settings.

6.1.4 Potential applications

Previously, I have listed the contributions of CoTrust, GEM, DBATE to trust management regarding agent groups. It is essential to illustrate the relation of these three components in action.

Figure 6.1 illustrates the relationship between these components during a life-cycle of an agent group. The interdependencies of these components are described as below.

1. In the group formation stage, the CoTrust mechanism is used to establish trust between agents. During this process, agents evaluate each other using a trust evaluation method which DBATE can be a good one.

2. To work effectively, DBATE requires contextual data, in which rating is the most important source.

3. To have the quality rating evidence, the trust management system should be able to achieve two objectives. Firstly, agents should have the incentive to leave accurate feedback. Secondly, all agents should receive ratings whether they work individually or in agent groups. These two requirements are covered by CoTrust and GEM.

As can be seen, the three components have a close relationship, the existence of one component has positive effects on the other components.

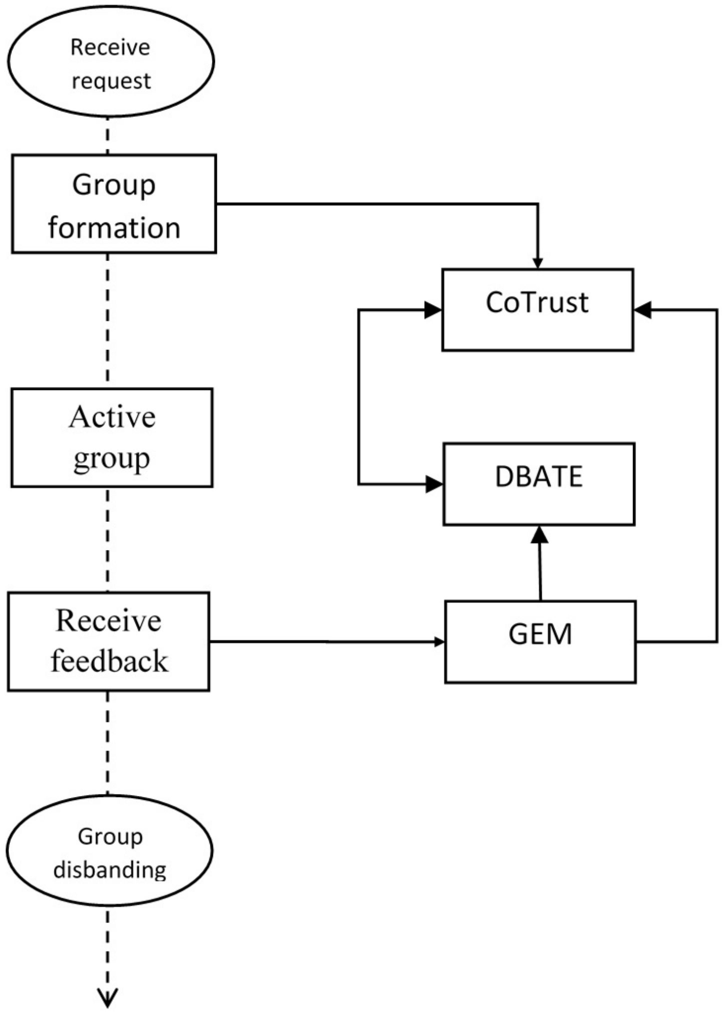


Figure 6.1: An ad-hoc group life-cycle and the trust management stack

As our research focuses on agent groups, it can be applied for trust-aware service composition or team composition. The main problems in this domain are selecting suitable components for the composite services and evaluating the trustworthiness of the composed services.

6.2 Limitations and Future directions

There are some possible improvements of the proposed trust management stack to be considered for future work.

Firstly, CoTrust's protocol is suitable for two agents negotiation. To form a large group that has more than two agents, it ignores the evaluation/opinion of existing members. I will continue to improve CoTrust when the "many-to-one" trust evaluation is address in the future. Besides, CoTrust works under the predefined preference settings of agents. The settings can benefit the preference reasoning. However, one possible problem is that some agents may constantly be rejected by responders as their preferences cannot pass the responders' evaluations. In the future, I will consider adaptive agents who can learn and change their preferences. Correspondingly, it is necessary to devise a better preference reasoning method that is applicable to these adaptive agents.

Secondly, regarding the trust evaluation method, DBATE has demonstrated its flexibility in capturing the dynamism of trust using contextual information. As DBATE uses a machine learning technique, it requires sufficient amount of data to work. Therefore, DBATE is not suitable to use at system bootstrap when the data for training may be inadequate. Also, DBATE works under the assumption that the behaviours of agents are stationary. In the future, I will investigate a suitable method for evaluating agent groups at the system bootstrap and consider other behavioural models of agents in groups.

In this thesis, the effectiveness of each proposed component is evaluated separately. However, in the future, I will find a suitable environment as well as criteria for evaluating the whole trust management stack.

Besides the specific issues mentioned above, the future work could also involve extending the scope of the research. Currently, the scope of this thesis covers one case in agent group interactions (see Chapter 1), i.e. one-to-many, besides, many-to-many or many-to-one. The two cases can take interaction between two agent groups cooperation, or one agent group evaluating another agent group. These directions could be much more challenging compared to the one-to-many case as each member of a group can have a different opinion about a targeted entity; the evaluation and cooperation between groups would be more complicated since it may involve the negotiation and consensus between members.

References

Abdul-Rahman, A. & Hailes, S. (1997). A distributed trust model. In *Proceedings of the 1997 workshop on new security paradigms* (pp. 48–60). New York, NY, USA: ACM. Retrieved from http://doi.acm.org/10.1145/283699.283739 doi: 10.1145/283699.283739

Abdul-Rahman, A. & Hailes, S. (2000). Supporting trust in virtual communities. In *Proceedings of the 33rd hawaii international conference on system sciences-volume 6 - volume 6* (pp. 6007–). Washington, DC, USA: IEEE Computer Society. Retrieved from http://dl.acm.org/citation.cfm?id=820262.820322

Adomavicius, G. & Tuzhilin, A. (2011). Context-aware recommender systems. In F. Ricci, L. Rokach, B. Shapira & B. P. Kantor (Eds.), *Recommender systems handbook* (pp. 217–253). Boston, MA: Springer US. Retrieved from http://dx.doi.org/10.1007/978-0-387-85820-3_7 doi: 10.1007/978-0-387-85820-3_7

Advogato. (n.d.). Retrieved from http://www.advogato.org/

Aljazzaf, Z. M., Perry, M. & Capretz, M. A. M. (2010, Sept). Online trust: Definition and principles. In *2010 fifth international multi-conference on computing in the global information technology* (p. 163-168). doi: 10.1109/ICCGI.2010.17

Al-Oufi, S., Kim, H.-N. & Saddik, A. E. (2012). A group trust metric for identifying people of trust in online social networks. *Expert Systems with Applications*, *39*(18), 13173 - 13181. doi: http://dx.doi.org/10.1016/j.eswa.2012.05.084

Anand, K., Gai, P. & Marsili, M. (n.d.). Financial crises and the evaporation of trust [Journal Article]. *arXiv preprint arXiv:0911.3099.*

Aref, A. & Tran, T. (2015). A trust establishment model in multi-agent systems. In *Workshops at the twenty-ninth aaai conference on artificial intelligence.*

Ashri, R., Ramchurn, S. D., Sabater, J., Luck, M. & Jennings, N. R. (2005). Trust evaluation through relationship analysis. In *Proceedings of the fourth international joint conference on autonomous agents and multiagent systems* (pp. 1005–1011). New York, NY, USA: ACM. Retrieved from http://doi.acm.org/10.1145/1082473.1082625 doi: 10.1145/1082473.1082625

Barber, K. S. & Kim, J. (2003). *Soft security: isolating unreliable agents from society* [Conference Paper]. Springer-Verlag.

Brams, S. J. & Taylor, A. D. (1996). *Fair division: From cake-cutting to dispute resolution*. Cambridge University Press.

Burnett, C., Norman, T. J. & Sycara, K. (2010). Bootstrapping trust evaluations through stereotypes. In *Proceedings of the 9th international conference on autonomous agents and multiagent systems: Volume 1 - volume 1* (pp. 241–248). Richland, SC: International Foundation for Autonomous Agents and Multiagent Systems. Retrieved from http://dl.acm.org/citation.cfm?id=1838206.1838240

Burnett, C., Norman, T. J. & Sycara, K. (2011). Trust decision-making in multi-agent systems. In *Proceedings of the twenty-second international joint conference on artificial intelligence - volume volume one* (pp. 115–120). AAAI Press. Retrieved from http://dx.doi.org/10.5591/978-1-57735-516-8/IJCAI11-031 doi: 10.5591/978-1-57735-516-8/IJCAI11-031

Castelfranchi, C. & Falcone, R. (2010). *Trust theory: A socio-cognitive and computational model* (1st ed.). Wiley Publishing.

Castelfranchi, C., Falcone, R. & Pezzulo, G. (2003). *Trust in information sources as a source for trust: a fuzzy approach* [Conference Paper]. ACM. doi: 10.1145/860575.860590

Castelfranchi, C. & Tan, Y.-H. (2002, April). The role of trust and deception in virtual societies. *Int. J. Electron. Commerce*, *6*(3), 55–70. Retrieved from http://dl.acm.org/citation.cfm?id=1286960.1286965

Chang, E., Dillon, T. S. & Hussain, F. K. (2005, July). Trust and reputation relationships in service-oriented environments. In *Third international conference on information technology and applications (icita'05)* (Vol. 1, p. 4-14 vol.1). doi: 10.1109/ICITA.2005.168

Chau, M., Zeng, D., Chen, H., Huang, M. & Hendriawan, D. (2003). Design and evaluation of a multi-agent collaborative web mining system [Journal Article]. *Decision Support Systems*, *35*(1), 167-183.

Comfort, L. K., Ko, K. & Zagorecki, A. (2004). Coordination in rapidly evolving disaster response systems. *American Behavioral Scientist*, *48*(3), 295-313. Retrieved from http://dx.doi.org/10.1177/0002764204268987 doi: 10.1177/0002764204268987

Commerce, B. E., Jøsang, A. & Ismail, R. (2002). The beta reputation system. In *In proceedings of the 15th bled electronic commerce conference.*

Cook, K. S., Yamagishi, T., Cheshire, C., Cooper, R., Matsuda, M. & Mashima, R. (2005). Trust building via risk taking: A cross-societal experiment. *Social Psychology Quarterly*, *68*(2), 121-142. Retrieved from http://www.jstor.org/stable/4148787

Dai, G. & Wang, Y. (2009). Trust-aware component service selection algorithm in service composition. In *Proceedings of the 2009 fourth international conference on frontier of computer science and technology* (pp. 613–618). Washington, DC, USA: IEEE Computer Society. Retrieved from http://dx.doi.org/10.1109/FCST.2009.41 doi: 10.1109/FCST.2009.41

Dasgupta, P. (1988). Trust as a commodity. In D. Gambetta (Ed.), *Trust: Making and breaking cooperative relations* (pp. 49–72). Blackwell.

Dekker, A. (2005). Conceptual distance in social network analysis. *Journal of social structure*, *6*(3).

Deutsch, M. (1962). Cooperation and trust: Some theoretical notes. *In M. R. Jones (Ed.), Nebraska Symposium on Motivation*, 275–319.

Dondio, P. & Longo, L. (2011). Trust-based techniques for collective intelligence in social search systems. In N. Bessis & F. Xhafa (Eds.), *Next generation data technologies for collective computational intelligence* (pp. 113–135). Berlin, Heidelberg: Springer Berlin Heidelberg. Retrieved from http://dx.doi.org/10.1007/978-3-642-20344-2_5 doi: 10.1007/978-3-642-20344-2_5

Douceur, J. R. (2002). The sybil attack. In P. Druschel, F. Kaashoek & A. Rowstron (Eds.), *Peer-to-peer systems: First internationalworkshop, iptps 2002 cambridge, ma, usa, march 7–8, 2002 revised papers* (pp. 251–260). Berlin, Heidelberg: Springer Berlin Heidelberg. Retrieved from http://dx.doi.org/10.1007/3-540-45748-8_24 doi: 10.1007/3-540-45748-8_24

Dunin-Keplicz, B. M. & Verbrugge, R. (2010). *Teamwork in multi-agent systems: A formal approach* (1st ed.). Wiley Publishing.

Durbin, J. & Koopman, S. J. (2012). *Time series analysis by state space methods* (Vol. 38). OUP Oxford.

eBay. (n.d.). *Feedback scores, stars, and your reputation.* Retrieved from http://pages.ebay.com/help/feedback/scores-reputation.html

Epinions. (n.d.). Epinions [Journal Article]. Retrieved from http://www.epinions.com/

Ezhei, M. & Tork Ladani, B. (2013). Gtrust: a group based trust model. *The ISC International Journal of Information Security*, *5*(2), 155–170. Retrieved from http://www.isecure-journal.com/article_39143.html doi: 10.22042/isecure.2014.5.2.4

Falcone, R. & Castelfranchi, C. (2001). Social trust: A cognitive approach. In C. Castelfranchi & Y.-H. Tan (Eds.), *Trust and deception in virtual societies* (pp. 55–90). Dordrecht: Springer Netherlands. Retrieved from http://dx.doi.org/10.1007/978-94-017-3614-5_3 doi: 10.1007/978-94-017-3614-5_3

Fang, H., Guo, G. & Zhang, J. (2015a). Multi-faceted trust and distrust prediction for recommender systems. *Decision Support Systems*, *71*, 37 - 47. Retrieved from http://www.sciencedirect.com/science/article/pii/S0167923615000068 doi: http://dx.doi.org/10.1016/j.dss.2015.01.005

Fang, H., Guo, G. & Zhang, J. (2015b, March). Multi-faceted trust and distrust prediction for recommender systems. *Decis. Support Syst.*, *71*(C), 37–47. Retrieved from http://dx.doi.org/10.1016/j.dss.2015.01.005 doi: 10.1016/j.dss.2015.01.005

Fang, H., Zhang, J., Şensoy, M. & Thalmann, N. M. (2012). Sarc: Subjectivity alignment for reputation computation. In *Proceedings of the 11th international conference on autonomous agents and multiagent systems - volume 3* (pp. 1365–1366). Richland, SC: International Foundation for Autonomous Agents and Multiagent Systems. Retrieved from http://dl.acm.org/

citation.cfm?id=2343896.2344008

Fang, H., Zhang, J. & Magnenat Thalmann, N. (2014). Subjectivity grouping: Learning from users' rating behavior. In *Proceedings of the 2014 international conference on autonomous agents and multi-agent systems* (pp. 1241–1248). Richland, SC: International Foundation for Autonomous Agents and Multiagent Systems. Retrieved from http://dl.acm.org/citation.cfm?id=2615731.2617446

Fehr, E. (2009). On the economics and biology of trust. *Journal of the European Economic Association*, *7*(2-3), 235–266. Retrieved from http://dx.doi.org/10.1162/JEEA.2009.7.2-3.235 doi: 10.1162/JEEA.2009.7.2-3.235

Fullam, K. K. & Barber, K. S. (2007). *Dynamically learning sources of trust information: experience vs. reputation* [Conference Paper]. ACM. doi: 10.1145/1329125.1329325

Fullam, K. K., Klos, T. B., Muller, G., Sabater, J., Schlosser, A., Topol, Z., ... Voss, M. (2005). A specification of the agent reputation and trust (art) testbed: Experimentation and competition for trust in agent societies. In *Proceedings of the fourth international joint conference on autonomous agents and multiagent systems* (pp. 512–518). New York, NY, USA: ACM. Retrieved from http://doi.acm.org/10.1145/1082473.1082551 doi: 10.1145/1082473.1082551

Fullam, K. K., Park, J. & Barber, K. S. (2005, April). Trust-driven information acquisition for secure and quality decision-making. In *International conference on integration of knowledge intensive multi-agent systems, 2005.* (p. 303-310). doi: 10.1109/KIMAS.2005.1427100

Gale, D. & Shapley, L. S. (1962). College admissions and the stability of marriage. *The American Mathematical Monthly*, *69*(1), 9-15. Retrieved from http://www.jstor.org/stable/2312726

Gal-Oz, N., Grinshpoun, T., Gudes, E. & Friese, I. (2010, May). Tric: An infrastructure for trust and reputation across virtual communities. In *2010 fifth international conference on internet and web applications and services* (p. 43-50). doi: 10.1109/ICIW.2010.14

Gambetta, D. (2000). Can we trust trust. *Trust: Making and breaking cooperative relations*, 213-237.

Granatyr, J., Botelho, V., Lessing, O. R., Scalabrin, E. E., Barthès, J.-P. & Enembreck, F. (2015, October). Trust and reputation models for multiagent systems. *ACM Comput. Surv.*, *48*(2), 27:1–27:42. Retrieved from http://doi.acm.org/10.1145/2816826 doi: 10.1145/2816826

Granovetter, M. (1985). Economic action and social structure: The problem of embeddedness. *American Journal of Sociology*, *91*(3), 481-510. Retrieved from http://www.jstor.org/stable/2780199

Griffiths, N. (2005). Task delegation using experience-based multi-dimensional trust. In *Proceedings of the fourth international joint conference on autonomous agents and multiagent systems* (pp. 489–496). New York, NY, USA: ACM. Retrieved from http://doi.acm.org/10.1145/1082473.1082548 doi: 10.1145/1082473.1082548

Grinshpoun, T., Gal-Oz, N., Meisels, A. & Gudes, E. (2009, Sept). Ccr: A model for sharing reputation knowledge across virtual communities. In *2009 ieee/wic/acm international joint conference on web intelligence and intelligent agent technology* (Vol. 1, p. 34-41). doi: 10.1109/WI-IAT.2009.13

Guo, G., Zhang, J. & Yorke-Smith, N. (2013). A novel bayesian similarity measure for recommender systems. In *Proceedings of the 23rd international joint conference on artificial intelligence (ijcai)* (p. 2619-2625).

Gusfield, D. & Irving, R. W. (1989). *The stable marriage problem: Structure and algorithms*. Cambridge, MA, USA: MIT Press.

Gustafsson, L. & Sternad, M. (2010). Consistent micro, macro and state-based population modelling. *Mathematical Biosciences*, *225*(2), 94 - 107. Retrieved from http://www.sciencedirect.com/science/article/pii/S0025556410000283 doi: https://doi.org/10.1016/j.mbs.2010.02.003

Haag, S., Cummings, M. & Phillips, A. (2007). *Management information systems for the information age* (6th ed.). McGraw-Hill Irwin.

Hackman, J. & Morris, C. G. (1975). Group tasks, group interaction process, and group performance effectiveness: A review and proposed integration1. In L. Berkowitz (Ed.), (Vol. 8, p. 45 - 99). Academic Press. Retrieved from http://www.sciencedirect.com/science/article/pii/S0065260108602488 doi: http://dx.doi.org/10.1016/S0065-2601(08)60248-8

Haibin, Z., Yan, W. & Xiuzhen, Z. (n.d.). A trust vector approach to transaction context-aware trust evaluation in e-commerce and e-service environments [Conference Proceedings]. In *Service-oriented computing and applications (soca), 2012 5th ieee international conference on* (p. 1-8). doi: 10.1109/SOCA.2012.6449461

Halberstadt, A. & Mui, L. (2001). Group reputation in multi-agent systems. In *In proceedings of the goddard/jpl workshop on radical agents concepts, nasa goddard space flight center.*

Hamill, L. & Gilbert, N. (2009). Social circles: A simple structure for agent-based social network models [Journal Article]. *Journal of Artificial Societies and Social Simulation*, *12*(2), 3.

Helbing, D. (1994). A mathematical model for the behavior of individuals in a social field. *Journal of Mathematical Sociology*, *19*(3), 189–219.

Hermoso, R., Billhardt, H. & Ossowski, S. (2010). Role evolution in open multi-agent systems as an information source for trust. In *Proceedings of the 9th international conference on autonomous agents and multiagent systems: Volume 1 - volume 1* (pp. 217–224). Richland, SC: International Foundation for Autonomous Agents and Multiagent Systems. Retrieved from http://dl.acm.org/citation.cfm?id=1838206.1838237

Hoelz, B. W. & Ralha, C. G. (2016). Towards a cognitive meta-model for adaptive trust and reputation in open multi-agent systems. In *Proceedings of the 2016 international conference on autonomous agents & multiagent systems* (pp.

624–625). Richland, SC: International Foundation for Autonomous Agents and Multiagent Systems. Retrieved from http://dl.acm.org/citation.cfm?id=2936924.2937016

Holland, J. (2006). Studying complex adaptive systems [Journal Article]. *Journal of Systems Science and Complexity*, *19*(1), 1-8. Retrieved from http://dx.doi.org/10.1007/s11424-006-0001-z doi: 10.1007/s11424-006-0001-z

Huang, F. (2007). Building social trust: A human-capital approach. *Journal of Institutional and Theoretical Economics JITE*, *163*(4), 552–573.

Huynh, T. D., Jennings, N. R. & Shadbolt, N. R. (2006a). Certified reputation: How an agent can trust a stranger. In *Proceedings of the fifth international joint conference on autonomous agents and multiagent systems* (pp. 1217–1224). New York, NY, USA: ACM. Retrieved from http://doi.acm.org/10.1145/1160633.1160854 doi: 10.1145/1160633.1160854

Huynh, T. D., Jennings, N. R. & Shadbolt, N. R. (2006b, September). An integrated trust and reputation model for open multi-agent systems. *Autonomous Agents and Multi-Agent Systems*, *13*(2), 119–154. Retrieved from http://dx.doi.org/10.1007/s10458-005-6825-4 doi: 10.1007/s10458-005-6825-4

Irving, R. W., Leather, P. & Gusfield, D. (1987, July). An efficient algorithm for the “optimal” stable marriage. *J. ACM*, *34*(3), 532–543. Retrieved from http://doi.acm.org.ezproxy.aut.ac.nz/10.1145/28869.28871 doi: 10.1145/28869.28871

Jarvenpaa, S. L. & Leidner, D. E. (1999, June). Communication and trust in global virtual teams. *Organization Science*, *10*(6), 791–815. Retrieved from http://dx.doi.org/10.1287/orsc.10.6.791 doi: 10.1287/orsc.10.6.791

Jennings, N. R. (2000). On agent-based software engineering [Journal Article]. *Artificial Intelligence*, *117*(2), 277-296. Retrieved from http://www.sciencedirect.com/science/article/pii/S0004370299001071 doi: http://dx.doi.org/10.1016/S0004-3702(99)00107-1

Jennings, N. R. (2001). An agent-based approach for building complex software systems [Journal Article]. *Commun. ACM*, *44*(4), 35-41. doi: 10.1145/367211.367250

Jiang, J. & Bai, Q. (2013a). Provenance-based trust estimation for service composition [Book Section]. In S. Cranefield & A. Nayak (Eds.), *Ai 2013: Advances in artificial intelligence* (Vol. 8272, p. 68-73). Springer International Publishing. Retrieved from http://dx.doi.org/10.1007/978-3-319-03680-9_7 doi: 10.1007/978-3-319-03680-9_7

Jiang, J. & Bai, Q. (2013b). Provenance-based trust estimation for service composition. In *Ai 2013: Advances in artificial intelligence* (Vol. 8272, p. 68-73). Springer International Publishing. doi: 10.1007/978-3-319-03680-9-7

Jonker, C. M. & Treur, J. (1999). Formal analysis of models for the dynamics of trust based on experiences. In *Proceedings of the 9th european workshop on modelling autonomous agents in a multi-agent world: Multiagent system engineering* (pp. 221–231). London, UK, UK: Springer-Verlag. Retrieved from http://dl.acm.org/citation.cfm?id=646910.710813

Jøsang, A. (1996). The right type of trust for distributed systems. In *Proceedings of the 1996 workshop on new security paradigms* (pp. 119–131). New York, NY, USA: ACM. Retrieved from `http://doi.acm.org/10.1145/304851.304877` doi: 10.1145/304851.304877

Jøsang, A. (2007). Trust and reputation systems [Book Section]. In A. Aldini & R. Gorrieri (Eds.), *Foundations of security analysis and design iv* (Vol. 4677, p. 209-245). Springer Berlin Heidelberg. Retrieved from `http://dx.doi.org/10.1007/978-3-540-74810-6_8` doi: 10.1007/978-3-540-74810-6_8

Jøsang, A. & Haller, J. (2007, April). Dirichlet reputation systems. In *Availability, reliability and security, 2007. ares 2007. the second international conference on* (p. 112-119). IEEE. doi: 10.1109/ARES.2007.71

Jøsang, A., Hayward, R. & Pope, S. (2006). Trust network analysis with subjective logic. In *Proceedings of the 29th australasian computer science conference - volume 48* (pp. 85–94). Darlinghurst, Australia, Australia: Australian Computer Society, Inc. Retrieved from `http://dl.acm.org/citation.cfm?id=1151699.1151710`

Kamvar, S. D., Schlosser, M. T. & Garcia-Molina, H. (2003). The eigentrust algorithm for reputation management in p2p networks. In *Proceedings of the 12th international conference on world wide web* (pp. 640–651). New York, NY, USA: ACM. Retrieved from `http://doi.acm.org/10.1145/775152.775242` doi: 10.1145/775152.775242

Kerr, R. & Cohen, R. (2006). *Modeling trust using transactional, numerical units* [Conference Paper]. ACM. doi: 10.1145/1501434.1501460

Kerr, R. & Cohen, R. (2009). Smart cheaters do prosper: Defeating trust and reputation systems. In *Proceedings of the 8th international conference on autonomous agents and multiagent systems - volume 2* (pp. 993–1000). Richland, SC: International Foundation for Autonomous Agents and Multiagent Systems. Retrieved from `http://dl.acm.org/citation.cfm?id=1558109.1558151`

Kollingbaum, M. J. & Norman, T. J. (2003). Supervised interaction: A form of contract management to create trust between agents. In *Proceedings of the 2002 international conference on trust, reputation, and security: Theories and practice* (pp. 108–122). Berlin, Heidelberg: Springer-Verlag. Retrieved from `http://dl.acm.org/citation.cfm?id=1762128.1762137`

Korovaiko, N. & Thomo, A. (2013). Trust prediction from user-item ratings. *Social Network Analysis and Mining*, *3*(3), 749–759. Retrieved from `http://dx.doi.org/10.1007/s13278-013-0122-z` doi: 10.1007/s13278-013-0122-z

Korsgaard, M. A., Schweiger, D. M. & Sapienza, H. J. (1995). Building commitment, attachment, and trust in strategic decision-making teams: The role of procedural justice. *The Academy of Management Journal*, *38*(1), 60-84. Retrieved from `http://www.jstor.org/stable/256728`

Kreps, D. & Wilson, R. (1982). Reputation and imperfect information. *Journal of Economic Theory*, *27*(2), 253-279. Retrieved from `http://EconPapers.repec.org/RePEc:eee:jetheo:v:27:`

y:1982:i:2:p:253-279

Kuter, U. & Golbeck, J. (2010, June). Using probabilistic confidence models for trust inference in web-based social networks. *ACM Trans. Internet Technol.*, *10*(2), 8:1–8:23. Retrieved from http://doi.acm.org/10.1145/1754393.1754397 doi: 10.1145/1754393.1754397

Levandowsky, M. & Winter, D. (1971). Distance between sets. *Nature*, *234*(5323), 34-35. (10.1038/234034a0)

Lewicki, R. J., Tomlinson, E. C. & Gillespie, N. (2006). Models of interpersonal trust development: Theoretical approaches, empirical evidence, and future directions. *Journal of Management*, *32*(6), 991-1022. Retrieved from http://dx.doi.org/10.1177/0149206306294405 doi: 10.1177/0149206306294405

Li, H. & Singhal, M. (2007, February). Trust management in distributed systems. *Computer*, *40*(2), 45–53. Retrieved from http://dx.doi.org/10.1109/MC.2007.76 doi: 10.1109/MC.2007.76

Li, L. & Wang, Y. (2010). Subjective trust inference in composite services. In *Proceedings of the twenty-fourth aaai conference on artificial intelligence* (pp. 1377–1384). AAAI Press. Retrieved from http://dl.acm.org/citation.cfm?id=2898607.2898827

Liu, A., Li, Q., Huang, L. & Wen, S. (2012). Shapley value based impression propagation for reputation management in web service composition. In *Proceedings of the 2012 ieee 19th international conference on web services* (pp. 58–65). Washington, DC, USA: IEEE Computer Society. Retrieved from http://dx.doi.org/10.1109/ICWS.2012.85 doi: 10.1109/ICWS.2012.85

Liu, S., Kot, A. C., Miao, C. & Theng, Y.-L. (2012). A dempster-shafer theory based witness trustworthiness model to cope with unfair ratings in e-marketplace. In *Proceedings of the 14th annual international conference on electronic commerce* (pp. 99–106). New York, NY, USA: ACM. Retrieved from http://doi.acm.org/10.1145/2346536.2346555 doi: 10.1145/2346536.2346555

Liu, S., Yu, H., Miao, C. & Kot, A. C. (2013). A fuzzy logic based reputation model against unfair ratings. In *Proceedings of the 2013 international conference on autonomous agents and multi-agent systems* (pp. 821–828). Richland, SC: International Foundation for Autonomous Agents and Multiagent Systems. Retrieved from http://dl.acm.org/citation.cfm?id=2484920.2485051

Liu, S., Zhang, J., Miao, C., Theng, Y.-L. & Kot, A. C. (2014). An integrated clustering-based approach to filtering unfair multi-nominal testimonies. *Computational Intelligence*, *30*(2), 316 - 341. Retrieved from http://ezproxy.aut.ac.nz/login?url=http://search.ebscohost.com/login.aspx?direct=true&db=bth&AN=95876767&site=eds-live

Liu, X. & Datta, A. (2012). Modeling context aware dynamic trust using hidden markov model. In *Proceedings of the twenty-sixth aaai conference on artificial intelligence* (pp. 1938–1944). AAAI Press. Retrieved from http://dl.acm.org/citation.cfm?id=2900929.2901002

Liu, X., Tredan, G. & Datta, A. (2014, November). A generic trust framework for large-scale open systems using machine learning. *Comput. Intell.*, *30*(4), 700–721. Retrieved from `http://dx.doi.org/10.1111/coin.12022` doi: 10.1111/coin.12022

Lu, G., Lu, J., Yao, S. & Yip, Y. J. (2009, March). A review on computational trust models for multi-agent systems. *The Open Information Science Journal*, *2*, 18–25. Retrieved from `http://eprints.hud.ac.uk/4102/` (Copyright © 1994 - 2009 Bentham Open)

Luo, Y., Liu, K. & Davis, D. N. (2002, Jan). A multi-agent decision support system for stock trading. *IEEE Network*, *16*(1), 20-27. doi: 10.1109/65.980541

Malik, Z. & Bouguettaya, A. (2009, Jan). Reputation bootstrapping for trust establishment among web services. *IEEE Internet Computing*, *13*(1), 40-47. doi: 10.1109/MIC.2009.17

Marsh, S. P. (1994). *Formalising trust as a computational concept* (Unpublished doctoral dissertation). University of Stirling.

Massa, P. & Avesani, P. (2007). Trust-aware recommender systems. In *Proceedings of the 2007 acm conference on recommender systems* (pp. 17–24). New York, NY, USA: ACM. Retrieved from `http://doi.acm.org/10.1145/1297231.1297235` doi: 10.1145/1297231.1297235

McArthur, S. D., Davidson, E. M., Catterson, V. M., Dimeas, A. L., Hatziargyriou, N. D., Ponci, F. & Funabashi, T. (2007). Multi-agent systems for power engineering applications—part i: Concepts, approaches, and technical challenges. *IEEE Transactions on Power systems*, *22*(4), 1743–1752.

Mehdi, M., Bouguila, N. & Bentahar, J. (2015). Trust and reputation of web services through qos correlation lens. *IEEE Transactions on Services Computing*, *PP*(99).

Meyerson, D., Weick, K. E. & Kramer, R. M. (1996). Swift trust and temporary groups. *Trust in organizations: Frontiers of theory and research*, *166*, 195.

Möllering, G. (2001). The nature of trust: From georg simmel to a theory of expectation, interpretation and suspension. *Sociology*, *35*(2), 403-420. Retrieved from `http://dx.doi.org/10.1177/S0038038501000190` doi: 10.1177/S0038038501000190

Moe, M. E. G., Tavakolifard, M. & Knapskog, S. J. (2008). Learning trust in dynamic multiagent environments using hmms. In *The 13th nordic workshop on secure it systems (nordsec2008). copenhagen, denmark. october 9-10, 2008.*

Mui, L. (2002). *Computational models of trust and reputation: Agents, evolutionary games, and social networks* (Unpublished doctoral dissertation). Massachusetts Institute of Technology.

Mui, L., Mohtashemi, M. & Halberstadt, A. (2002). A computational model of trust and reputation for e-businesses. In *Proceedings of the 35th annual hawaii international conference on system sciences (hicss'02)-volume 7 - volume 7* (pp. 188–). Washington, DC, USA: IEEE Computer Society. Retrieved from `http://dl.acm.org/citation.cfm?id=820745.821158`

Murphy, K. P. (2002). *Dynamic bayesian networks: Representation, inference and learning* (Unpublished doctoral dissertation). (AAI3082340)

Nepal, S., Sherchan, W. & Paris, C. (2011, Nov). Strust: A trust model for social networks. In *2011ieee 10th international conference on trust, security and privacy in computing and communications* (p. 841-846). doi: 10.1109/TrustCom.2011.112

Nguyen, T. D. & Bai, Q. (2014). Accountable individual trust from group reputations in multi-agent systems [Book Section]. In *Pricai 2014: Trends in artificial intelligence* (p. 1063-1075). Springer.

Nguyen, T. D. & Bai, Q. (2016). Trust management for composite services in distributed multi-agent systems with indirect ratings: (extended abstract). In *Proceedings of the 2016 international conference on autonomous agents & multiagent systems* (pp. 1295–1296). Richland, SC: International Foundation for Autonomous Agents and Multiagent Systems. Retrieved from `http://dl.acm.org/citation.cfm?id=2936924.2937127`

Nguyen, T. D. & Bai, Q. (2017a). Bitrust: A comprehensive trust management model for multi-agent systems. In K. Fujita et al. (Eds.), *Modern approaches to agent-based complex automated negotiation* (pp. 3–16). Cham: Springer International Publishing. Retrieved from `http://dx.doi.org/10.1007/978-3-319-51563-2_1` doi: 10.1007/978-3-319-51563-2_1

Nguyen, T. D. & Bai, Q. (2017b, June). Enhance trust management in composite services with indirect ratings. In *The computer journal.* Oxford University Press. doi: https://doi.org/10.1093/comjnl/bxx026

Nguyen, T. D., Bai, Q. & Li, W. (2016). Capability-aware trust evaluation model in multi-agent systems. In R. Booth & M.-L. Zhang (Eds.), *Pricai 2016: Trends in artificial intelligence: 14th pacific rim international conference on artificial intelligence, phuket, thailand, august 22-26, 2016, proceedings* (pp. 771–778). Springer International Publishing. Retrieved from `http://dx.doi.org/10.1007/978-3-319-42911-3_65` doi: 10.1007/978-3-319-42911-3_65

Noorian, Z., Marsh, S. & Fleming, M. (2011). Multi-layer cognitive filtering by behavioral modeling. In *Proceedings of the 10th international conference on autonomous agents and multiagent systems - volume 2* (pp. 871–878). Richland, SC. Retrieved from `http://dl.acm.org/citation.cfm?id=2031678.2031742`

Page, L., Brin, S., Motwani, R. & Winograd, T. (1999). The pagerank citation ranking: Bringing order to the web [Journal Article].

Pinyol, I. & Sabater-Mir, J. (2013, June). Computational trust and reputation models for open multi-agent systems: A review. *Artif. Intell. Rev.*, *40*(1), 1–25. Retrieved from `http://dx.doi.org/10.1007/s10462-011-9277-z` doi: 10.1007/s10462-011-9277-z

Pipattanasomporn, M., Feroze, H. & Rahman, S. (n.d.). Multi-agent systems in a distributed smart grid: Design and implementation [Conference Proceedings]. In *Power systems conference and exposition, 2009. psce '09. ieee/pes* (p. 1-8). doi: 10.1109/PSCE.2009.4840087

Rabiner, L. R. (1989, Feb). A tutorial on hidden markov models and selected applications in speech recognition. *Proceedings of the IEEE*, *77*(2), 257-286. doi: 10.1109/5.18626

Ramchurn, S. D., Huynh, D. & Jennings, N. R. (2004, March). Trust in multi-agent systems. *Knowl. Eng. Rev.*, *19*(1), 1–25. Retrieved from `http://dx.doi.org/10.1017/S0269888904000116` doi: 10.1017/S0269888904000116

Ramchurn, S. D. & Jennings, N. R. (2004). Devising a trust model for multi-agent interactions using confidence and reputation. *International Journal of Applied Artificial Intelligence*, *18*, 2004.

Regan, K., Poupart, P. & Cohen, R. (2006). Bayesian reputation modeling in e-marketplaces sensitive to subjecthity, deception and change. In *Proceedings of the 21st national conference on artificial intelligence - volume 2* (pp. 1206–1212). AAAI Press. Retrieved from `http://dl.acm.org/citation.cfm?id=1597348.1597380`

Rettinger, A., Nickles, M. & Tresp, V. (2008). A statistical relational model for trust learning. In *Proceedings of the 7th international joint conference on autonomous agents and multiagent systems - volume 2* (pp. 763–770). Richland, SC: International Foundation for Autonomous Agents and Multiagent Systems. Retrieved from `http://dl.acm.org/citation.cfm?id=1402298.1402330`

Rettinger, A., Nickles, M. & Tresp, V. (2011). Statistical relational learning of trust [Journal Article]. *Machine learning*, *82*(2), 191-209.

Rodriguez, S., Julián, V., Bajo, J., Carrascosa, C., Botti, V. & Corchado, J. M. (2011, August). Agent-based virtual organization architecture. *Eng. Appl. Artif. Intell.*, *24*(5), 895–910. Retrieved from `http://dx.doi.org/10.1016/j.engappai.2011.02.003` doi: 10.1016/j.engappai.2011.02.003

Romano, D. M. (2003). *The nature of trust: Conceptual and operational clarification* (Thesis).

Sabater, J. & Sierra, C. (2001a). Regret: Reputation in gregarious societies. In *Proceedings of the fifth international conference on autonomous agents* (pp. 194–195). New York, NY, USA: ACM. Retrieved from `http://doi.acm.org/10.1145/375735.376110` doi: 10.1145/375735.376110

Sabater, J. & Sierra, C. (2001b, December). Social regret, a reputation model based on social relations. *SIGecom Exch.*, *3*(1), 44–56. Retrieved from `http://doi.acm.org/10.1145/844331.844337` doi: 10.1145/844331.844337

Sabater, J. & Sierra, C. (2005). Review on computational trust and reputation models. *Artificial Intelligence Review*, *24*(1), 33–60. Retrieved from `http://dx.doi.org/10.1007/s10462-004-0041-5` doi: 10.1007/s10462-004-0041-5

Sen, S. (2013). A comprehensive approach to trust management. In *Proceedings of the 2013 international conference on autonomous agents and multi-agent systems* (pp. 797–800). Richland, SC: International Foundation for Autonomous Agents and Multiagent Systems. Retrieved from `http://dl.acm.org/citation.cfm?id=2484920.2485046`

Shapley, L. S. (1953). *A value for n-person games* (Vol. 28; H. W. Kuhn & T. A. W., Eds.). Princeton University Press.

Shoham, Y. & Leyton-Brown, K. (2008). *Multiagent systems: Algorithmic, game-theoretic, and logical foundations*. New York, NY, USA: Cambridge University Press.

Sierra, C. & Debenham, J. (2005). An information-based model for trust. In *Proceedings of the fourth international joint conference on autonomous agents and multiagent systems* (pp. 497–504). New York, NY, USA: ACM. Retrieved from http://doi.acm.org/10.1145/1082473.1082549 doi: 10.1145/1082473.1082549

Singh, A., Juneja, D. & Sharma, A. K. (2010, June). Introducing trust establishment protocol in contract net protocol. In *2010 international conference on advances in computer engineering* (p. 59-63). doi: 10.1109/ACE.2010.13

Sparrowe, R. T., Liden, R. C., Wayne, S. J. & Kraimer, M. L. (2001). Social networks and the performance of individuals and groups. *The Academy of Management Journal*, *44*(2), pp. 316-325.

Su, X., Zhang, M., Mu, Y. & Bai, Q. (2013, August). A robust trust model for service-oriented systems. *J. Comput. Syst. Sci.*, *79*(5), 596–608. Retrieved from http://dx.doi.org/10.1016/j.jcss.2012.11.005 doi: 10.1016/j.jcss.2012.11.005

Sycara, K. P. (1998). Multiagent systems [Journal Article]. *AI magazine*, *19*(2), 79.

Székely, G. J., Rizzo, M. L., Bakirov, N. K. et al. (2007). Measuring and testing dependence by correlation of distances. *The Annals of Statistics*, *35*(6), 2769–2794.

Teacy, W. L., Luck, M., Rogers, A. & Jennings, N. R. (2012, December). An efficient and versatile approach to trust and reputation using hierarchical bayesian modelling. *Artif. Intell.*, *193*, 149–185. Retrieved from http://dx.doi.org/10.1016/j.artint.2012.09.001 doi: 10.1016/j.artint.2012.09.001

Teacy, W. T. L., Chalkiadakis, G., Rogers, A. & Jennings, N. R. (2008). Sequential decision making with untrustworthy service providers. In *Proceedings of the 7th international joint conference on autonomous agents and multiagent systems - volume 2* (pp. 755–762). Richland, SC: International Foundation for Autonomous Agents and Multiagent Systems. Retrieved from http://dl.acm.org/citation.cfm?id=1402298.1402329

Teacy, W. T. L., Patel, J., Jennings, N. & Luck, M. (2006). Travos: Trust and reputation in the context of inaccurate information sources [Journal Article]. *Autonomous Agents and Multi-Agent Systems*, *12*(2), 183-198. Retrieved from http://dx.doi.org/10.1007/s10458-006-5952-x doi: 10.1007/s10458-006-5952-x

Teacy, W. T. L., Patel, J., Jennings, N. R. & Luck, M. (2005). *Coping with inaccurate reputation sources: experimental analysis of a probabilistic trust model* [Conference Paper]. ACM. doi: 10.1145/1082473.1082624

Toivonen, S., Lenzini, G. & Uusitalo, I. (2006). Context-aware trust evaluation functions for dynamic reconfigurable systems. In *Mtw.*

Tong, X. & Zhang, W. (2009, Aug). Group trust and group reputation. In *2009 fifth international conference on natural computation* (Vol. 5, p. 561-565). doi: 10.1109/ICNC.2009.738

Tran, T., Cohen, R., Langlois, E. & Kates, P. (2014). Establishing trust in multiagent environments: realizing the comprehensive trust management dream. In *Proceedings of the 17th international workshop on trust in agent societies.*

Villata, S., Boella, G., Gabbay, D. M. & van der Torre, L. (2013). A socio-cognitive model of trust using argumentation theory. *International Journal of Approximate Reasoning*, *54*(4), 541 - 559. Retrieved from `http://www.sciencedirect.com/science/article/pii/S0888613X12001600` doi: http://dx.doi.org/10.1016/j.ijar.2012.09.001

Vinyals, M., Rodriguez-Aguilar, J. A. & Cerquides, J. (2011, March). A survey on sensor networks from a multiagent perspective. *Comput. J.*, *54*(3), 455–470. Retrieved from `http://dx.doi.org/10.1093/comjnl/bxq018` doi: 10.1093/comjnl/bxq018

Wahab, O. A., Bentahar, J., Otrok, H. & Mourad, A. (2015). A survey on trust and reputation models for web services: Single, composite, and communities. *Decision Support Systems*, *74*, 121 - 134. Retrieved from `http://www.sciencedirect.com/science/article/pii/S0167923615000809` doi: http://dx.doi.org/10.1016/j.dss.2015.04.009

Wang, Y., Chen, I. R., Cho, J. H., Chan, K. S. & Swami, A. (2013, Nov). Trust-based service composition and binding for tactical networks with multiple objectives. In *Milcom 2013 - 2013 ieee military communications conference* (p. 1862-1867). doi: 10.1109/MILCOM.2013.314

Wang, Y., Hang, C.-W. & Singh, M. P. (2011, January). A probabilistic approach for maintaining trust based on evidence. *J. Artif. Int. Res.*, *40*(1), 221–267. Retrieved from `http://dl.acm.org/citation.cfm?id=2016945.2016952`

Wang, Y., Li, L. & Liu, G. (2015). Social context-aware trust inference for trust enhancement in social network based recommendations on service providers. *World Wide Web*, *18*(1), 159–184. Retrieved from `http://dx.doi.org/10.1007/s11280-013-0241-5` doi: 10.1007/s11280-013-0241-5

Wang, Y., Li, M., Dillon, E., g. Cui, L., j. Hu, J. & j. Liao, L. (2008, April). A context-aware computational trust model for multi-agent systems. In *2008 ieee international conference on networking, sensing and control* (p. 1119-1124). doi: 10.1109/ICNSC.2008.4525384

Wang, Y. & Singh, M. P. (2007). *Formal trust model for multiagent systems* [Conference Paper]. Morgan Kaufmann Publishers Inc.

Wang, Y. & Singh, M. P. (2010, November). Evidence-based trust: A mathematical model geared for multiagent systems. *ACM Trans. Auton. Adapt. Syst.*, *5*(4), 14:1–14:28.

Wang,

In *Proceedings of the 3rd international conference on peer-to-peer computing* (pp. 150–). Washington, DC, USA: IEEE Computer Society. Retrieved from `http://dl.acm.org/citation.cfm?id=942805.943810`

Weng, J., Miao, C. & Goh, A. (2006, September). An entropy-based approach to protecting rating systems from unfair testimonies. *IEICE - Trans. Inf. Syst.*, *E89-D*(9), 2502–2511. Retrieved from `http://dx.doi.org/10.1093/`

ietisy/e89-d.9.2502 doi: 10.1093/ietisy/e89-d.9.2502

Whitby, A., Jøsang, A. & Indulska, J. (2004). Filtering out unfair ratings in bayesian reputation systems. In *Proceedings of the 7th international workshop on trust in agent societies* (Vol. 6).

Wooldridge, M. (2009). *An introduction to multiagent systems* [Book]. Wiley Publishing.

Yaari, M. & Bar-Hillel, M. (1983). *On dividing justly*. Department of Applied Economics, University of Cambridge. Retrieved from https://books.google.co.nz/books?id=hakpHAAACAAJ

Yu, B. & Singh, M. P. (2000). A social mechanism of reputation management in electronic communities. In *Proceedings of the 4th international workshop on cooperative information agents iv, the future of information agents in cyberspace* (pp. 154–165). London, UK, UK: Springer-Verlag. Retrieved from http://dl.acm.org/citation.cfm?id=647785.735555

Yu, H., Miao, C., An, B., Leung, C. & Lesser, V. R. (2013). A reputation management approach for resource constrained trustee agents. In *Proceedings of the twenty-third international joint conference on artificial intelligence* (pp. 418–424). AAAI Press. Retrieved from http://dl.acm.org/citation.cfm?id=2540128.2540189

Yu, H., Miao, C., An, B., Shen, Z. & Leung, C. (2014). *Reputation-aware task allocation for human trustees* [Conference Paper]. International Foundation for Autonomous Agents and Multiagent Systems.

Yu, H., Shen, Z., Leung, C., Miao, C. & Lesser, V. R. (2013). A survey of multi-agent trust management systems. *IEEE Access*, *1*, 35-50. doi: 10.1109/ACCESS.2013.2259892

Yu, H., Shen, Z., Miao, C. & An, B. (2013). A reputation-aware decision-making approach for improving the efficiency of crowdsourcing systems. In *Proceedings of the 2013 international conference on autonomous agents and multi-agent systems* (pp. 1315–1316). Richland, SC: International Foundation for Autonomous Agents and Multiagent Systems. Retrieved from http://dl.acm.org/citation.cfm?id=2484920.2485201

Yu, L. & Liu, H. (2004, December). Efficient feature selection via analysis of relevance and redundancy. *J. Mach. Learn. Res.*, *5*, 1205–1224. Retrieved from http://dl.acm.org/citation.cfm?id=1005332.1044700

Zhang, J., Ghorbani, A. A. & Cohen, R. (2007). A familiarity-based trust model for effective selection of sellers in multiagent e-commerce systems. *International Journal of Information Security*, *6*(5), 333–344. Retrieved from http://dx.doi.org/10.1007/s10207-007-0025-y doi: 10.1007/s10207-007-0025-y

Zhang, T., Ma, J., Xi, N., Liu, X., Liu, Z. & Xiong, J. (2014, June). Trustworthy service composition in service-oriented mobile social networks. In *2014 ieee international conference on web services* (p. 684-687). doi: 10.1109/ICWS.2014.102

Zhang, Y. & Yu, T. (2012). Mining trust relationships from online social networks.

Journal of Computer Science and Technology, *27*(3), 492–505. Retrieved from http://dx.doi.org/10.1007/s11390-012-1238-8 doi: 10.1007/s11390-012-1238-8

Zheng, X., Wang, Y. & Orgun, M. A. (2013, June). Modeling the dynamic trust of online service providers using hmm. In *2013 ieee 20th international conference on web services* (p. 459-466). doi: 10.1109/ICWS.2013.68

Appendix A

Statistical models

This appendix provides more detailed information about the Hidden Markov Model and the Dynamic Bayesian Network Model used in Chapter 5.

A.1 State-space models

In the real world, there are many systems which produce sequential data. The data may either be a time series, generated by a dynamical system, or a sequence generated by a spatial process, as seen in Murphy (2002). The data can be obtained in real-time or the data has previously been collected.

If we denote $y_{1..t} = (y_1, ..., y_t)$ as the observed sequence, then the prediction is $P(y_{t+h}|y_{1..t})$, where $h > 0$ and it indicates how far into the future we want to predict. An example can be predicting the weather in the next 10 days.

The prediction can be calculated using either classical time series modelling or state-space modelling. It has been proved that space-time models are more efficient in many scenarios (Durbin & Koopman, 2012; Murphy, 2002). A state-space model assumes that there is some underlying hidden state that generates the observations. This hidden state evolves in time, possibly as a function of the inputs. The goal is to infer the hidden state that produced the observations up to time t. The Hidden Markov Model (HMM) and Dynamic Bayesian Network(DBN) are well-known state-space models which are capable of solving the problem.

A.2 Hidden Markov Model

The Hidden Markov Model(HMM) is a statistical model in which the system being modelled is assumed to be a Markov process with unobserved (hidden) states. HMMs have found applications in many areas in relation to signal processing and in particular speech processing (Rabiner, 1989).

A discrete HMM is defined as a tuple $\lambda = (S, V, \pi, A, B)$:

- S is the set of N distinct states, $S = \{S_1, S_2, \ldots, S_N\}$.
- V is the set of M output symbols, $V = \{V_1, V_2, \ldots, V_M\}$.
- $\pi = \{\pi_i\}_{N\times 1}$ is the initial state distribution, i.e. $\pi_i = P(s_i)$
- A is the state transition probability matrix:
  $A = \{a_{ij}\}_{N\times N}$, where $a_{i,j} = P(x_{t+1} = S_j \mid x_t = S_i)$, $1 \le i, j \le N$.
- B is the observation (emission) probability matrix:
  $B = \{b_j(k)\}_{N\times M}$, where $b_j(k) = P(V_k \mid x_t = S_j), j = 1..N, k = 1..M$.

O is a series of observed outputs of length T, $O = \{o_1, o_2, \ldots, o_T\}$, $o_t \in V, t = 1..T$, and the corresponding (hidden) state sequence $X = \{x_1, x_2, \ldots, x_T\}$, $x_i \in S$.

There are three common problems for state-space inference, i.e., evaluation, decoding and learning.

Evaluation

The evaluation problem is to find the probability $P(O|\lambda)$ when O and a model λ are given. In other words, it is to find the probability of the observation sequence given a model. This problem can be viewed as one of evaluating how well a model predicts a given observation sequence.

The probability of the observations O for a specific state sequence X is:

$$P(O|X,\lambda) = \prod_{t=1}^{T} P(o_t|x_t,\lambda) = b_{x_1}(o_1)bx_2(o_2)\dots b_{q_T}(o_T) \tag{A.1}$$

An effective solution for this is using the forward algorithm, which is a dynamic programming induction. The algorithm can reduce the redundant calculations compared with the recursive method. The details of the algorithm are provided in Subsection 5.3.2.

Decoding

The decoding problem is to find X that maximise the probability $P(O|\lambda)$ when O and a model λ is given.

The aim of decoding is to discover the hidden state sequence that was most likely to have produced a given observation sequence. One solution to this problem is to use the Viterbi algorithm to find the single best state sequence for an observation sequence. The Viterbi algorithm is quite similar to the forward algorithm, except that the transition probabilities are maximised at each step, instead of summed.

We denote the probability of the most possible state sequence for the observations as:

$$\delta(i) = \max_{x_1,\dots,x_{t-1}} P(x_1 x_2 \dots, x_t = S_i \mid \lambda) \tag{A.2}$$

The Viterbi algorithm is described as follows:

1. Initiation:

$$\begin{aligned} &\delta_1(i) = \pi_i b_i(o_1), \qquad 1 \leq i \leq N \\ &\psi_1(i) = 0; \end{aligned} \tag{A.3}$$

2. Recursion:

$$\begin{aligned} \delta_t(j) &= \max_{1 \leq i \leq N}[\delta_{t-1}(i)a_{ij}]b_j(o_t), \qquad 2 \leq t \leq T, 1 \leq j \leq N \\ \psi_t(j) &= \max_{1 \leq i \leq N}[\delta_{t-1}(i)a_{ij}], \qquad 2 \leq t \leq T, 1 \leq j \leq N \end{aligned} \tag{A.4}$$

3. Termination:

$$\begin{aligned} P* &= \max_{1 \leq i \leq N}[\delta_t()] \\ s_T^* &= arg \max_{1 \leq i \leq N}[\delta_t(i)] \end{aligned} \tag{A.5}$$

4. Optimal sequence backtracking:

$$s_t^* = \psi_{t+1}(s_{t+1}^*), \qquad ,t = T-1, T-2, \ldots, 1. \tag{A.6}$$

In the recursion step, the probability of each state is maximised rather than summed like in the forward algorithm. Backtracking allows the best state sequence to be found from the back pointers stored in the recursion stop.

Learning

The learning problem is how to adjust λ to maximise probability $P(O|\lambda)$ when O is given. In other words, we want estimate the model parameters $\lambda = (A, B, \pi)$ that best describe that process.

There are two standard training methods depending on the samples. If the training sample contains both inputs and output of a process, the supervised training can be applied. For other cases, we have the unsupervised training method. The supervised training methods are often simpler than unsupervised training. While supervised training can be a simple method such as counting frequency, unsupervised methods are often more complicated. A well-known approach for unsupervised training is the Baum-Welch algorithm.

A.3 Dynamic Bayesian Networks

Dynamic Bayesian Network (DBN) is a well-known tool for a wide range of data mining applications, including speech recognition, digital forensics, and bioinformatics. The term dynamic in DBN describes a system which changes or evolves dynamically over time. Murphy (2002) introduced DBN in different forms of hidden Markov process.

A.3.1 DBN VS HMM

A Dynamic Bayesian Network is a Bayesian network which relates variables to each other over adjacent time steps. Typically the variables can be partitioned into the input, hidden and output variables of a state-space model in three-tuple $Z_t = (U_t, X_t, Y_t)$. Murphy (2002) defined a DBN as a directed acrylic graph (DAG) that relates two-slice temporal Bayesian networks (see Figure A.1).

$$P(Z_t \mid Z_{t-1}) = \prod_{i=1}^{N} P(Z_t^i \mid Pa(Z_t^i)) \tag{A.7}$$

where Z_t^i is the i^{th} node at time t, which could be a component of Xt, Yt or Ut, and $Pa(Z_t^i)$ are the parents of Z_t^i in the graph. Each node in the second slice of the temporal Bayesian network has an associated conditional probability distribution, which defines $P(Z_t^i \mid Pa(Z_t^i))$ for all $t > 1$. The parents of a node, $Pa_(Z_t^i)$ can be in the same time slice or in the previous time slice which can be assume to be the first-order Markov.

The main differences of DBNs and HMMs are:

- An HMM represents the state using single discrete random variable, $X_t \in$

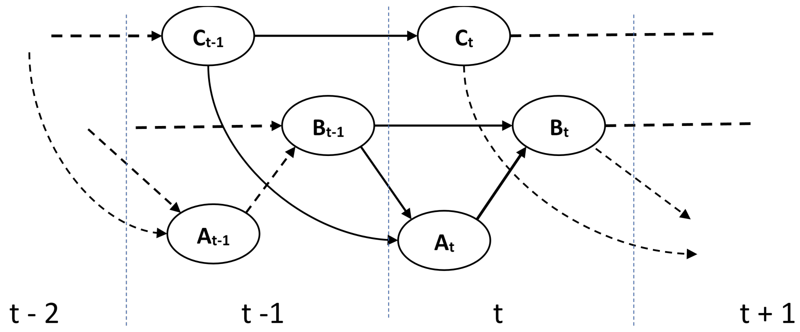


Figure A.1: Example of a DBN composed of 3 variables.

$\{1, \ldots, K\}$

- A DBN represents the state of the world using a set of random variables, i.e., $X_t^1, \ldots, X_t^D$
- A DBN represents $P(X_t \mid X_{t-1})$ in a compact way using a parameterised graph.

The DBN may have fewer parameters, and the inference in DBN may be much faster than its corresponding HMM. In the simplest case, an HMM can be presented by a DBN graph as depicted in Figure A.2.

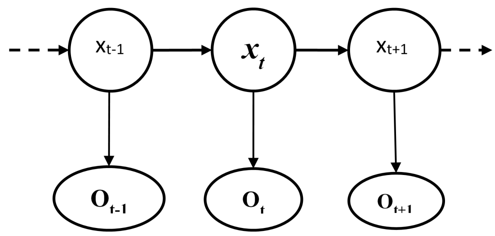


Figure A.2: An HMM represented as an instance of a DBN