



HAL
open science

Managing uncertainty through trust in resource sharing systems

Ningkui Wang

► **To cite this version:**

Ningkui Wang. Managing uncertainty through trust in resource sharing systems. Automatic Control Engineering. Université de Lille, 2020. English. NNT : . tel-04317280

HAL Id: tel-04317280

<https://theses.hal.science/tel-04317280>

Submitted on 1 Dec 2023

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

CENTRALE LILLE

THESE

Présentée en vue d'obtenir le grade de

DOCTEUR

Spécialité : Automatique, Informatique

Ningkui WANG

DOCTORAT DELIVRE PAR CENTRALE LILLE

Managing uncertainty through trust in resource sharing systems

Gestion de l'incertitude par la confiance dans les systèmes de partage de ressources

Soutenue le 17 décembre devant le jury d'examen:

Présidente *Evren SAHIN, Professeur, Ecole Centrale de Paris, Laboratoire Génie Industriel*
Rapporteur *Jean-Charles BILLAUT, Professeur, Université de Tours*
Rapporteur *Olga BATTAIÀ, Professeur, Université de Bordeaux*
Examineur *Hayfa ZGAYA, HDR. MCF, ILIS/Université de Lille, CRISAL*
Directeur *Slim HAMMADI, Professeur, Ecole Centrale de Lille/Centrale lille, CRISAL*
Co-directeur *Philippe MATHIEU, Professeur, IUT/Université de Lille, CRISAL*

Thèse préparée dans le Laboratoire CRISAL

Ecole Doctorale SPI 072



Abstract

The service-oriented networked manufacturing known as cloud manufacturing is an innovative manufacturing paradigm. It transforms manufacturing resources and products into manufacturing services, which can be managed and operated in an intelligent and unified way to allow the full sharing and flowing of these services between customers and suppliers. Therefore, the selection of the suitable manufacturing service plays an essential role in cloud manufacturing implementation and development. The Cloud manufacturing is a kind of parallel, networked, and distributed system consisting of an integrated and inter-connected manufacturing services. In this context, we adopt the multi-agent approach to model and manage the cloud manufacturing in order to provide solutions for all different users involved in the whole lifecycle of manufacturing. A user is thus represented by an autonomous agent that interacts with the other agents in the system to share and also benefit from the services available "in the cloud". Thus, an agent makes his choice based on reputation and trust. This thesis deals with service cloud selection from the viewpoint of trust. With Dempster-Shafter theory of evidence, this work considers direct trust that originated from direct transactions, mainly highlighting the factor of uncertainty caused by incompleteness, randomness, fading property, transaction frequency and service' fluctuant performance. The proposed trust model also estimates the indirect reputation presented by third-party agents, emphasizing the confidence in applying the received recommendations from the perspective of personal reliability, entropy-based recommendation certainty, and distance-based recommendation quality. Additionally, recommendations presented by trusted agents with similar service requirements are also analyzed in a random walk process to select cloud manufacturing resources. The experimental results of this thesis, implemented in agent-based cloud manufacturing, proved the efficiency of our trust estimation model in cloud service selection. Furthermore, it should be noted that this thesis presents a promising perspective for distributed cloud manufacturing control.

Keywords: Cloud manufacturing, distributed system, multi-agent system, trust, uncertainty, service selection.

Résumé

La fabrication en réseau orientée services, connue sous le nom de "fabrication en nuage" ou "fabrication cloud", est un paradigme de fabrication innovant. Ce nouveau paradigme transforme les ressources et capacités de fabrication en services unifiés et accessibles via un "cloud", facilitant la gestion des ressources, la pertinence des données et la circulation des flux. Ces services peuvent donc être gérés et exploités d'une manière pertinente et rationnelle afin de permettre un partage harmonieux des prestations ainsi qu'une circulation efficace d'informations entre les différents acteurs du système. Dans ce contexte, la sélection du service approprié joue un rôle essentiel dans la mise en uvre et le développement des services de fabrication en nuage. En effet, ces services sont intégrés dans un système distribué en réseau, constitué de services interconnectés. Ce qui les rend facilement évolutifs par rapport aux changements éventuels et donc efficaces et robustes. Cette thèse traite la sélection des services en nuages du point de vue de la confiance. Les utilisateurs sont donc représentés sous forme d'entités autonomes et interactives échangeant des informations concernant ces services. Dans ce contexte, l'approche multi-agent est adoptée pour modéliser, gérer et fournir des solutions à tous les utilisateurs impliqués dans le cycle de vie complet de la production de ces services. Un utilisateur est donc représenté par un agent autonome qui interagit avec les autres agents du système pour partager et aussi bénéficier des services disponibles "en nuage". Ainsi, un agent fait son choix en se basant sur la réputation et confiance. Dans ce contexte, nous nous sommes basés dans ce travail de thèse sur la théorie des preuves de Dempster-Shafter qui considère la confiance directe et indirecte. La confiance directe provient des transactions directes, en mettant principalement en évidence le facteur d'incertitude causé par l'incomplétude, le caractère aléatoire, le facteur d'oubli, la fréquence des transactions et la performance fluctuante du service. Ce modèle de confiance estime également la réputation indirecte présentée par des agents tiers, en soulignant la confiance dans l'application des recommandations reçues du point de vue de la fiabilité personnelle, de la certitude des recommandations basée sur une entropie et de la qualité des recommandations basée sur la distance. En outre, les recommandations présentées par des agents de confiance ayant des exigences de service similaires sont également analysées dans un processus de marche aléatoire

pour sélectionner les services en nuages. Les résultats expérimentaux dans cette thèse, mis en uvre dans le contexte du partage des services en nuages basé sur des agents, ont prouvé l'efficacité de notre modèle d'estimation de la confiance dans la sélection de ces services. En outre, il convient de noter que cette thèse présente une perspective prometteuse pour le contrôle distribué des services.

Mots-clés: Fabrication en nuage, systèmes distribués, systèmes multiagents, confiance, incertitude, selection de services

Contents

Abstract	i
Résumé	iii
Contents	v
List of Figures	ix
List of Tables	xiii
General Introduction	1
1 Resource sharing problems	5
1.1 An introduction to sharing	5
1.2 Drivers of resource sharing	7
1.3 Shared resources	9
1.4 Manufacturing	11
1.4.1 Manufacturing industry	11
1.4.2 Cloud Manufacturing	13
1.4.3 Sharable resources in manufacturing	16
1.5 Trust	18
1.6 Goal and assumptions	21
1.7 Conclusion	24
2 Trust management in resource sharing systems	25
2.1 Agent-based cloud manufacturing	25
2.1.1 Agents and multiagent systems	25
2.1.2 Multiagent cloud manufacturing	28
2.2 Trust models	31
2.2.1 Trust management in multiagent systems	33
2.2.2 Numerical models	36
Direct experience	36
Indirect reputation	38

Trust and reputation framework	40
2.2.3 Trust in resource sharing systems	41
2.3 Conclusion	44
3 An integrated evidence-based trust model in multiagent systems	45
3.1 Background	45
3.2 Preliminary	47
3.2.1 Why Dempster-Shafer theory of evidence	48
3.2.2 Dempster Shafer theory of evidence	49
3.3 Proposed trust model in a distributed multiagent system	54
3.3.1 Model direct trust	55
3.3.2 Method 1: Modelling indirect trust by the amount of information (MITAI)	59
3.3.3 Method 2: An overall indirect reputation model	62
Evidence consistency	64
Credibility of individual witness	65
Model certainty	66
Model indirect reputation	68
Model overall Trust	69
Update Credibility	70
Incentive	72
3.4 Resource reservation and resource sharing	73
3.5 Discussion	73
3.6 Conclusion	75
4 Random walk based resource selection in a distributed environment	77
4.1 Introduction: Problem statement	77
4.2 Related works	79
4.2.1 Recommender system	79
4.2.2 PageRank algorithm	80
4.2.3 Random walk algorithm	82
4.2.4 PageRank algorithm in recommender systems	83
4.3 Random walk for manufacturing service selection	85
4.3.1 Cloud manufacturing services	85
4.3.2 Preference representation	87
4.3.3 Trust estimation in the proposed distributed multiagent systems	91
Edge weights–Trust relationships	92
4.3.4 Interactions and decision making	96
4.3.5 Evaluation and dynamic network structure	99

4.4	Conclusion	100
5	Application and validation	103
5.1	Trust estimation models	103
5.1.1	Evidence generation	103
5.1.2	Model indirect trust by information amount VS. Beta reputation system	105
	Simulation setting	106
	Simulation Results	107
5.1.3	Manage trust certainty with entropy	108
	Certainty rises with increasing experiences under fixed conflict	110
	Certainty rises with increasing conflict under fixed experience .	110
	Comparison and discussion	112
5.1.4	The overall trust model	113
5.1.5	Limitation and discussions	117
5.2	Random walk process for service selection	119
5.2.1	A test of Simhash	119
5.2.2	Trust-based cloud manufacturing recommendation	121
5.3	Conclusion	127
	Conclusion and Future works	129
	Reference	151

List of Figures

1-1	Drivers of resource sharing	7
1-2	Shared cultural taxonomy and examples for each category	9
1-3	Manufacturing process	12
1-4	Articles on cloud manufacturing by year and country from 2010 to June 2020.	14
1-5	An example of service transaction flow (Yan, Cheng, and Tao, 2016).	15
1-6	The classification of cloud manufacturing resources	17
1-7	An example of functional trust and recommender trust.	20
1-8	An example of manufacturing resource sharing.	22
2-1	Agents and the agent environment (Wooldridge, 1999)	26
2-2	Research domains of multiagent systems	28
2-3	Manufacturing service agent communication architecture	31
2-4	Trust models and the related information	33
2-5	From complete distrust to trust	36
3-1	A game of picking ball which can be handled by probability theory	48
3-2	A game of picking ball where probability theory is unapplicable but evidence theory is able to handle.	48
3-3	Trust estimation processes: from experience to trust value	55
3-4	Trust evaluation in multiagent systems	56
3-5	The assigned uncertainty changes with the value of frequency and averaging fluctuation	59
3-6	The entropy when the frame of discernment is $\Omega = \{T, nT\}$ and it is in the range of $[0, 2]$	67
3-7	Intergrated confidence	69
4-1	A basic principle of PageRank, the face sizes indicate the importance	80
4-2	Random walk process for provider selection	86
4-3	An example: Manufacturing indexes building procedure	89
4-4	A sequence diagram of agents communication	97

5-1	The recent five-round cumulative absolute error, obtained under different probabilities and changes performance in different interactive rounds: Figure. 5-1(a): 10 rounds and the probability equals 0.1 or 0.9; Figure. 5-1(b): 100 rounds and the probability equals 0.1 or 0.9; Figure. 5-1(c): 10 rounds and the probability equals 0.3 or 0.7; Figure. 5-1(d): 100 rounds and the probability equals 0.3 or 0.7;	105
5-2	the RFS comparison between the MINTAI and the BRS	108
5-3	RFS comparison of A) our proposed entropy-based weighted approach MITAI (high weight assigned to low-entropy BPAs) with B) the analogous version of MITAI with high weight assigned to high-entropy BPAs	109
5-4	Evidence certainty increases with the amount of interaction ($r + s$) when satisfactory 0.5 is fixed. X-axis: Amount of interaction; Y-axis: Certainty degree	111
5-5	Evidence certainty varies with satisfactory when the amount of interaction 20 is fixed, and the satisfactory 0.5 leads to the lowest certainty.	111
5-6	Change in average credibility for each information sharing agent.	115
5-7	Change in average credibility for each information sharing agent when five agents change their performance.	116
5-8	Change in average credibility for each information sharing agent when one agent changes its performance, the change process is as defined in (Yu, Kallurkar, and Flo, 2008).	117
5-9	Change in average credibility for each information sharing agent when five agent change their performances, the change process is as defined in (Yu, Kallurkar, and Flo, 2008).	118
5-10	The effect of the proposed trust-based recommendation: It is obtained by the satisfactory degree of the proposed method minus the satisfactory degree obtained by randomly selection. Figure. 5-10(a): one factor is considered to represent the preference; Figure. 5-10(b) : two factors is considered to represent the preference; Figure. 5-10(c) : three factors is considered to represent the preference. Figure. 5-10(d): four factors are considered to represent the preference.	123

5-11	The proposed trust-based recommendation method's effect under different percentages of dishonest agents: We use the proposed method's accumulative satisfaction minus the satisfactory degree obtained by randomly selecting a potential provider. Figure. 5-11(a): All agents are honest; Figure. 5-11(b): 60% of the agents are honest; Figure. 5-11(c): 40% of the agents are honest; Figure. 5-11(d): All agents are dishonest;	124
5-12	The effect of preference similarity in the proposed trust-based method: We use the proposed method's satisfactory degree minus the satisfactory degree obtained by randomly selecting a potential provider. Figure. 5-12(a): preference similarity is considered and 20% of agents perform misleading information; Figure. 5-12(b): preference similarity is not included, and high-trust resource provider agents are recommended in the random walk, and 20% of agents present misleading information.	125
5-13	The impact of the number of agents in the proposed trust-based approach: We use the proposed method's satisfactory degree minus the satisfactory degree obtained by randomly selecting a potential provider. Figure. 5-13(a): The simulation is conducted with 10 agents; Figure. 5-13(b): The simulation is conducted with 50 agents; Figure. 5-13(c): The simulation is conducted with 500 agents;	126

List of Tables

2-1	Trust models	34
3-1	Comparisons with trust models	76
5-1	Certainty computed by differentiated approaches for different satisfactory with fixed amount of interaction 4.	112
5-2	Parameters in the simulation.	114
5-3	Parameters in the simulation.	122

General introduction

This dissertation is related to trust estimation in the distributed solving of resource sharing systems. In this chapter, we describe the general issues of this dissertation. After a presentation of the context and a description of this research's motivations, our objectives and contribution are detailed. We end this chapter with the structure of this dissertation.

Context and motivations

Sharing is a popular social activity that can be traced back to ancient times and has also attracted widespread attention in modern society. Resource sharing aims to share a set of resources with a group of users who have needs or preferences to save time, energy, costs, etc. Some typical applications include Airbnb, car sharing, bike sharing, and cloud manufacturing. In Industry 4.0, cloud manufacturing was first introduced to facilitate the sharing of distributed manufacturing resources. In such a system involving enormous resources, a multiagent system is usually employed to resolve the sharing problem. However, agents' trust has a significant impact on sharing efficiency and has been widely studied. This thesis mainly investigates the concept of trust for agent-based manufacturing cloud selection in a distributed environment.

This dissertation focuses on uncertainty management in trust estimation models, which would be employed for service provider agent selection to facilitate resource sharing. Ordinarily, trust is on the basis of evidence, including interactive and organizational evidence. However, lots of uncertainty involved brings potential decision-making errors and difficulties to resource sharing. In the interaction-based trust estimation models, insufficient evidence, uncertainty, randomness, volatility, dynamic, etc., makes it challenging to maintain the direct experience. Simultaneously, ensuring the received indirect experience trustworthy is also problematic since the third-party agents could be deceptive, uncertain, and insufficient. Besides, how to obtain and maintain reliable indirect experience is also challenging. Additionally,

trust and privacy issues need to be emphasized in a distributed resource sharing environment. We address these central notions in this dissertation for resource sharing to enhance efficiency and lower risks.

Objectives and contributions

In this dissertation, we seek to design and improve trust estimation models to facilitate resource sharing. We assume that agents have interactions to realize direct trust. That is to say, the proposed trust model defaults to having at least one interaction. Many researchers have studied the fading and dynamic property. Nevertheless, analyzing and using uncertainty have seldom been considered. Thus, uncertainty causes in interaction-based trust estimation models have to be stressed and employed to achieve useful information. Also, many studies have considered evidence confidence. However, no one has investigated a comprehensive confidence approach.

Our main contribution first has overcome the limitations and improved the Dempster - Shafter theory of evidence-based trust estimation model, from both direct trust and indirect reputation perspectives. The uncertainty caused by the listed factors is analyzed and represented by basic probability assignment. What follows, evidence confidence is emphasized by entropy-based evidence certainty, distance-based evidence quality, and previous experience-based credibility. We also use credits to incentive agents to provide reliable information. In the resource sharing system, trusted and similar clients are employed in a random walk process to recommend and select the valid resource provider agent.

Thesis structure

Chapter 1: Resource sharing problems. This chapter presents the general context related to resource sharing problems. Their main characteristics of resource sharing are discussed, including the definitions, the drives, and generally shared resources. We finally introduce the notion of trust, and this chapter ends with a detailed description of the thesis goals.

Chapter 2: Trust management in resource sharing systems. This chapter presents the relevant literature review and related works. We split the relevant contents into two parts: agent-based manufacturing and trust management in multiagent systems. A discussion about these frameworks is included in this chapter.

Chapter 3: An integrated evidence-based trust model in multiagent systems. This chapter presents an adaptive trust estimation model that can maintain excellent performance in a distributed multiagent system involving deceptive agents. This approach is based on the Dempster-Shafter theory of evidence, and we have analyzed the shortages in the previous Dempster-Shafter theory of evidence-based trust estimation models. Then we improve the trust of direct trust and indirect reputation perspectives. Both direct trust and indirect reputation are integrated for trust estimation. This chapter begins by specifying direct trust that emphasizes the main factor of uncertainty, and subsequently, studies indirect reputation in the same way as to direct trust. This chapter analyzes recommender trust from the social-cognitive perspective and introduces a near-perfect model that considers consistency, credibility, and certainty to capture the confidence. Furthermore, a trust model generally solicits third-party agents for information and accurate information. We, therefore, motivate agents to be trustworthy, thanks to a credit mechanism.

Chapter 4: Random walk based resource selection in a distributed environment. This chapter proposes a random walk based resource sharing approach in a distributed environment. We first defined what kind of resources are shared based on the preference representation. Then, we proposed a random walk based manufacturing resource sharing model in the heterogeneous information network. The agent's trust degree and client agents' preference are studied to determine the weight to select a reliable and desirable agent for resource sharing.

Chapter 5: Application and validation. This chapter mainly gives simulation results to demonstrate the efficiency of the proposed method in chapter 3 and chapter 4. Besides, we also give real examples of manufacturing in the multiagent system to practically use the proposed trust model for resource sharing.

We conclude this thesis by summarizing our main contributions. The limitations of this paper are also described by the description of extensions and future works, which seem interesting.

Chapter 1

Resource sharing problems

In the past few decades, significant attention has been devoted to resource sharing. This dissertation deals with the management of uncertainty of resource sharing in a distributed environment through the concept of trust. In this chapter, we present the relevant content of resource sharing, including an introduction to sharing and its development, the drives of resource sharing, and so on. After discussing and describing the goals and assumptions, the motivation is detailed.

1.1 An introduction to sharing

Sharing has been the most basic way of economic distribution in hominid societies for several hundred thousand years (Price, 1975), and it is a distinct, ancient, and increasingly indispensable consumer research topic (Winterhalder, 1986; Belk, 2010). For instance, numerous anthropological researches have investigated patterns of food sharing, primarily in forager societies, to improve resource use efficiency and enhance security (Gurven, 2006). Likewise, a general decentralized approach is often used in home and small office networks, where every user makes their local folders and printers available to others (Kleinrock, 2002). These kinds of shared tasks mainly appeared in small geographic areas to satisfy nearby participants' needs and requirements. Nowadays, rituals of sharing are even more popular worldwide. Individuals and companies have recognized the immense potential of sharing tangible and intangible resources, principally due to the continuous advances in communication technologies and the rapid expansion of the Internet in terms of E-commerce (Ahadipour and Schanzenbach, 2017). As a result, diverse business models for resource sharing have been developed from both academic and practical lives aspects (Di Amato, 2016; Kumar, Lahiri, and Dogan, 2018). Among these sharing models, the shared resources range from consumer levels, such as the well-known car-sharing system in West Europe and North America (Shaheen, Sperling, and Wagner, 1998; Sprei et al., 2019), to a business or industrial level, such as groupage systems in

transportation logistics (Shi et al., 2019; Kuklina et al., 2020), cloud and grid computing in information processing (Shen et al., 2017a; Sun et al., 2019), and cloud manufacturing platforms for manufacturing resource sharing (Tao et al., 2011; Ghomi, Rahmani, and Qader, 2019; Ren et al., 2015).

There have been various research trying to define sharing (Belk, 2010). Benkler sees sharing as nonreciprocal pro-social behavior (Benkler, 2004). Sharing is defined as the act and the process of distributing what is ours to others or can be understood as a process of receiving something from others for our own use (Belk, 2007). Rather than identifying what is mine or yours, sharing explains some items as ours (Lamberton and Rose, 2012). Two or more individuals, groups, and even nations may experience the earnings (or expenses) from occupying a thing in a sharing. In this dissertation, sharing is interpreted as providing ours to others for their use or obtaining some things from others for our use, thereby benefiting both sides. Participants may share anything, ranging from a private house, a personnel vehicle, a bicycle, or a traditional family meal, to knowledge, experience, and ideas. Each of these sharing involving some costs or benefits of an object. Parts features concerning sharing from (Katrini, 2018) are enumerated as following,

- **High-participatory:** Participants desire to share peer to peer or use a pool of resources instead of owning them.
- **Co-management:** The participants collaborate in the management of the practice with a range of capacities, and ideally, power asymmetries are avoided through a series of organizational mechanisms.
- **Social relationships:** The practice helps build certain social relationships over time between the participants that go beyond a one-time interaction for the sharing of resources.
- **Sustainability:** The practice model is sustained through time, as well as agile and resilient.
- **Accessibility:** The practice strives to be open to all, and its identity is redefined as people join.
- **Physicality:** The practice takes place in the physical space and has spatial manifestations.

As shown above, all participants share their possessions, especially surplus resources, in a shared system to obtain certain benefits for demanders to obtain the right to use the resources within a specific period. Sharing does not exchange the

ownership. After the transaction, the resources still belong to the original owner. The literature also appears to use various terms to describe and discuss resource sharing (Yu et al., 2011; Daudi, 2018). The terms include collaborative sharing (Gonzalez-Feliu and Morana, 2011), sharing economy (Hamari, Sjöklint, and Ukkonen, 2016; Puschmann and Alt, 2016), and collaborative consumption (Jiang and Tian, 2018). The sharing economy and collaborative consumption are even popular because they mostly involve sharing by means of renting idle or underutilized resources. In what follows, we discuss the driving factors of resource sharing.

1.2 Drivers of resource sharing

Resource sharing can become highly embedded in the everyday, and thus it is vital to recognize its emergence and evolution over time. Many drivers have promoted the rapid expansion of resource sharing today, which can be interpreted from the technology, economic, social, and biological perspectives displayed in Figure 1-1. These aspects are interrelated, complementary, and mutually restrictive, and they are compatible with what Gesing and Puschmann described as the drivers of sharing economy (Gesing, 2017; Puschmann and Alt, 2016). The sharing economy drivers include changed consumer behaviors and the increasing use of social networks, electronic marketplaces, mobile devices, and electronic services. At the same time, achievable trust and increasing transparency have boosted resource sharing (Paajanen, 2019). More details are explained in the following paragraphs.

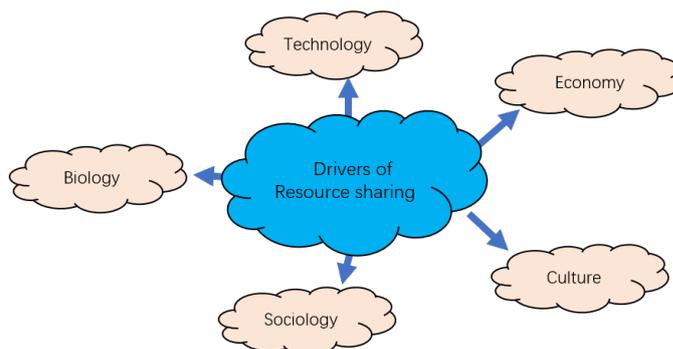


FIGURE 1-1: Drivers of resource sharing

Societal factors raise a need to share resources. Shove et al., describe the emergence of new elements that can be caused by the death of other elements (Shove, Pantzar, and Watson, 2012). Within this framework, we could see the driver of resource sharing in correlation to the demise of relevant services provided by the state

or the market within specific contexts, which are synonymous with what Shove et al., discuss as drivers of the sharing culture (Katrini, 2018). On one side, the nation lacks the competencies and materials to provide citizens with excellent quality services due to policy changes and lack of support. On another side, citizens might not afford some services rendered by the market because of a lack of money, time, or even being not satisfied with the available alternatives (Katrini, 2018). Furthermore, as a result of changing times, people today, young and old, are opting for experiences over possessions, and a staggering 74 percent of Americans prioritize experiences over owning products as reported in (Morgan, 2019). In those contexts, an alternative resource sharing emerges to fill the gap in the market, the state's lack of support, and fulfilling people's daily lives. For example, according to the State Information Center (Administration Center of China E-government Network)'s recent report, the COVID-19 outbreak at the beginning of 2020 facilitated the transaction volumes in the fields of shared healthcare and shared education (The State Information Center, 2020), due to the lock-down and social distancing. Here, shared health brings together clinical experts from across the country to deliver a patient-centered, accessible, responsive health system that people can count on, while shared education is defined as schools from different sectors working in partnership to provide opportunities for pupils, staff and the community to engage in collaborative and meaningful learning experiences.

In the previous paragraph, we analyzed the promotion of resource sharing by social factors. In this part, we understand the resource sharing driver from an ecological perspective. Paris Agreement commits participating parties to "holding the increase in the global average temperature to well below 2°C above pre-industrial levels and pursuing efforts to limit the temperature increase to 1.5°C above pre-industrial levels" (Dimitrov et al., 2019). According to the Intergovernmental Panel on Climate Change's (IPCC's) Fifth Assessment Report, to achieve even a 2°C goal, world greenhouse gas (GHG) emissions must peak between 2010 and 2020 and 2050 emissions should be 41-72% below 2010 levels (Zhou et al., 2019). Achieving these international goals, reducing the use of energy resources, and improving the rational use of energy will be applicable. However, an increase in population has resulted in a generation of more wastes, which has nowhere to go (Buczynski, 2013). In this respect, sharing transportation vehicles, for example, may leverage the number of vehicle trucks deployed on the road while also reducing the amount of fuel usage and CO₂ emissions (Daudi, 2018). Of course, it has many other influential factors, and we do not discuss all of them in detail here. In what follows, the shared resources are emphasized in the following parts.

1.3 Shared resources

In recent years, the concept of sharing has turned from a community-based practice into a profitable business model (Katrini, 2018). In the traditional sharing activities, people in communities throughout the world regularly adopt productive methods to meet their everyday demands by sharing and collaboration, often to create alternative resolutions capable and more socially engaging. Groups of people who immerse themselves in these modes have been defined as creative communities (Meroni, 2007; Katrini, 2018), and are favorably presenting resolutions to daily needs on a local scale through more substantial applications of human, environmental, and economic resources. For instance, Bird-David et al. note the strong emphasis on sharing in hunter-gatherer societies and anthropologists who study them (Bird-david, Widlok, and Tadesse, 2005). In (Katrini, 2018), the authors have listed ten taxonomical categories of sharing are food, shelter, work, caregiving, knowledge, well-being, resources, mobility, leisure (Figure. 1-2).

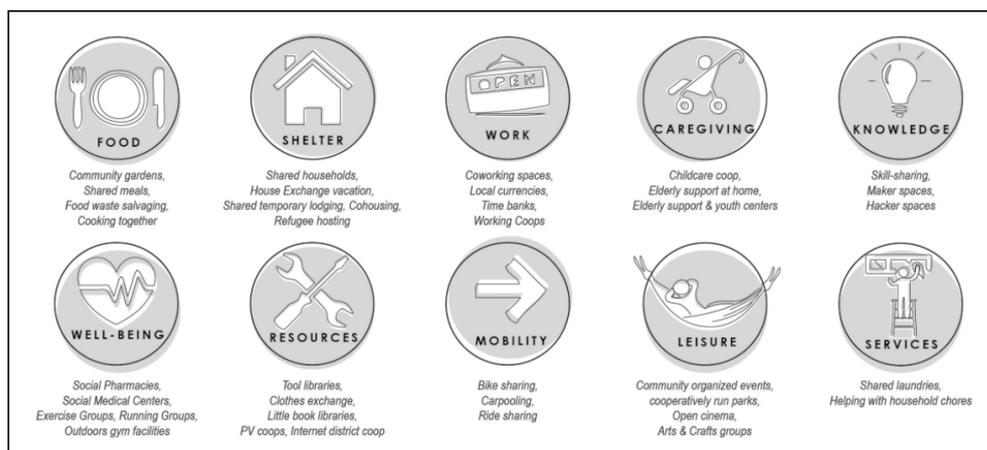


FIGURE 1-2: Shared cultural taxonomy and examples for each category

In recent years, more and more individuals and companies have noticed the immense potential of resource sharing, and new technological possibilities such as the Internet of Things have offered innovative approaches to control the use of resources dynamically inside and outside company borders. Examples of resource sharing on a consumer-level can be discovered, among others, in mobility and housing (Botsman and Rogers, 2010). Especially car-sharing services have become popular in Western Europe and North America (Lee and Cho, 2015; Ciari, Weis, and

Balac, 2016), and non-dockless bicycle sharing in China (Zheng and Du, 2019). Besides consumer use, resource sharing is also utilized at a business or industrial level. Typical applications are groupage systems in transportation logistics (Mao, 2008; Becker and Hendrik, 2016), cloud computing in information processing (Liu et al., 2016). For instance, in computing, a resource is any physical or virtual component of limited availability within a computer system. Every device connected to a computer system is a resource. Every internal system component is a resource. Virtual system resources include files (concretely file handles), network connections (concretely network sockets), and memory areas. The application of grid computing has approached maturity, and now more scientific research and practice are focused on cloud computing.

In summary, the resources that are being shared can be anything that can be shared; it consists of financial capital, technical and managerial skill, or other corresponding assets (Lin, Yang, and Arya, 2009). (Lavie, 2006) differentiate between two types of resources, namely tangible resources and intangible resources. As well, resource sharing also emerges in human resources, and the shared resources thus are concluded as follows,

(i) **Tangible resources**

Tangible resources refer to the fixed and current assets of an organization that has a fixed long-run capacity. They can be touched physically and transmitted without loss of integrity, such as financial capital and equipment.

(ii) **Intangible resources**

An intangible resource does not physically exist that can be classified as assets and skills. In terms of intangible assets, something as we referred to as trust and patent. As for intangible skills, for example, corporate culture, know-how, technology, and management (Hall, 1992).

(iii) **Human resources**

Recently, human-beings have started to contribute skills and personal time through shareable modes, facilitated by digital platforms (Gesing, 2017)

It is crucial to distinguish tangible from intangible resources since they function differently in resource sharing. Compared with tangible resources, intangible resources are less mobile, imitable, and substitutable (Das and Teng, 2000). Thus intangible resource has more advantages in term of resource sharing. Many studies have focused on the impact of tangible and intangible resources on resource sharing,

especially in the alliance networks. (Ghobadi and D'Ambra, 2012) shows that competition for tangible resources was found to positively affect the cooperative communication of individuals, whereas competition for intangible resources (political competition) had negative impacts on resource sharing and task orientations. Research works on how trust affects alliance performance for resource concludes that goodwill trust matters more to tangible than intangible resource sharing, whereas competence trust matters more to intangible than to tangible resource sharing (Jiang et al., 2015). In this dissertation, we tend to resource sharing in manufacturing, then the tangible resources such as pure manufacturing resources, and intangible resources, such as manufacturing capability, are encapsulated as services that sharable in the pool. In the following, more content related to manufacturing is expressed.

1.4 Manufacturing

These sharing are in the models of Business-to-Consumer (B2C) (Jiang and Wang, 2005), and Consumer-to-Consumer (C2C) (Teubner et al., 2014) and Business-to-Business (B2B) (Neu and Brown, 2005; Jeong, Oh, et al., 2017). The C2C is the creation of a product or service with a specific promotional strategy for consumers to share that product or service with others, such as warehouses, accommodation, and vehicles (trucks). In their entirety, these profitable business models enjoy emerged digital platforms, which facilitate matchmaking and coordination. The use of digital platforms reduces the scale for viable hiring transactions or participation in consumer hiring markets. Sharing economy platforms and applications are already being used widely in the B2C markets, such as Uber and Airbnb (Ert, Fleischer, and Magen, 2016; Di Amato, 2016), but sharing solutions for the B2B markets still carries much potential. B2B manufacturers hold a unique position in the business market.

1.4.1 Manufacturing industry

The manufacturing industry is the basis of a nation's economy and powerfully affects people's livelihood (Zhong et al., 2017). According to the global manufacturing scorecard (West and Lansang, 2018), manufacturing constitutes 27 percent of China's overall national output, which accounts for 20 percent of the world's manufacturing output. In the United States, it represents 12 percent of the nation's output and 18 percent of the world's capacity. In Japan, manufacturing is 19 percent of the country's national output and 10 percent of the world's total. From another view, manufacturing provides high-wage jobs, and it is the primary source of commercial innovation. Besides, manufacturing can significantly reduce the nation's trade

deficit and make a disproportionately large contribution to environmental sustainability (Helper, Krueger, and Wial, 2012). However, industrialized economies' manufacturing output growth has been continuously diminishing since the end of 2018, primarily due to the prevailing trade and tariff uncertainties in commodity trading in the US and EU economies. Global manufacturing output growth has recently registered a sharp decline of 6.0 percent in the first quarter of 2020 due to economic lockdown measures motivated by the COVID-19 pandemic (United nations industrial development organization, 2020). Several suggestions are raised to improve the manufacturing environment, one of which is to unlock 21st-century tools such as Big Data, automation, and artificial intelligence (West and Lansang, 2018).

Manufacturing is the making of goods by hand or by machines that, upon completion, the business sells to a customer. Items used in manufacture may be raw materials or component parts of a larger product. The manufacturing usually happens on a large-scale production line of machinery and skilled labor (Buzacott and Yao, 1986). Figure. 1-3 shows the transformation of materials into items of higher value by one or more processing and assembly operations. Customers and suppliers are two necessary components in manufacturing. When customers necessitate particular merchandise, the best supplier discovery is a critical challenge. Generally, the following phases are employed to find an appropriate supplier (Thomas, 2020):

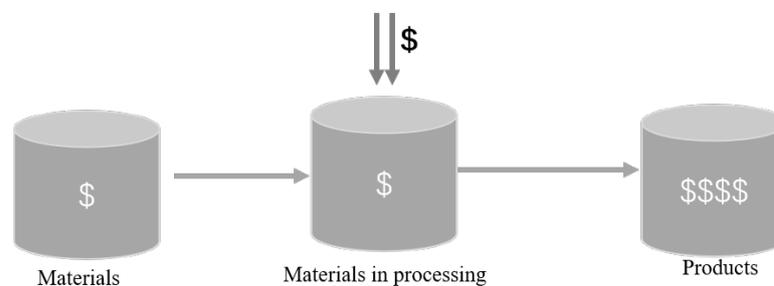


FIGURE 1-3: Manufacturing process

1. The company defines a need for a product or service. Before discovering suppliers, the initial minimum supplier requirements must be defined, including direct spendings such as raw materials, components, and services, and indirect spendings such as computers and office supplies.
2. Supplier discovery leads to a shortlist. A customer search for information and list possible suppliers. Afterward, the customer asks each of them to provide more detailed information.

3. Suppliers provide required information If suppliers meet the minimum requirement in the first step, they reply to offer more information.
4. Advanced supplier review At this step, all potential suppliers are evaluated thoroughly, including their capability, reliability, willingness, costs, timelessness, and so forth.
5. Acceptance the winner receives the opportunity to be the manufacturer partner.

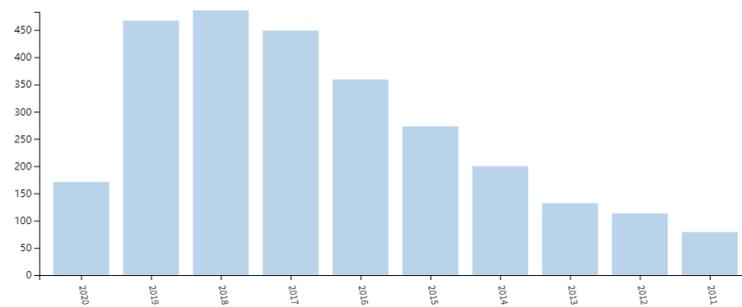
Scientific research of resource sharing in manufacturing are not sufficient, and most of them focus on resource cooperation, such as sharing of demand information or joint activities within research and development (Garrette, Castañer, and Dussauge, 2009). From another view, more resource sharing in manufacturing is conducted in the context of the supplier chain. For example, Sandberg investigated the situation of information sharing for Swedish manufacturing companies within supply chains, and results showed a correlation between the intensity of the collaboration and its success (Sandberg, 2007). (Ma et al., 2018) explored the feasibility of implementing resource sharing in the manufacturing stage of the garment supply chain. A key performance indicator based on an extended analytic network process approach was designed to evaluate the overall performance of different resource sharing scenarios. Also, a simulation model of crowdsourced manufacturing with a resource model and an agent-based negotiation algorithm to evaluate manufacturing effectiveness based on delivery and machine usage is proposed in (Kaihara et al., 2017).

1.4.2 Cloud Manufacturing

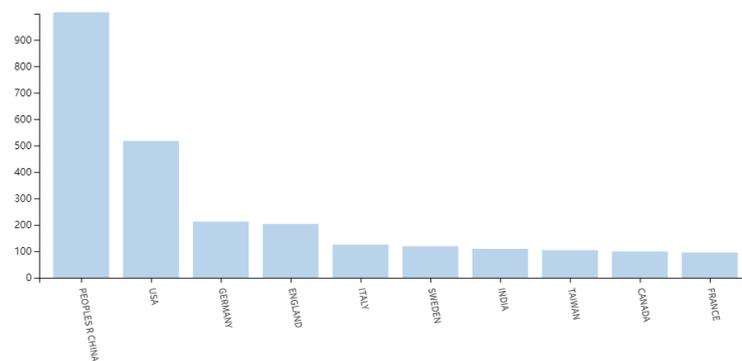
Now, we are stepping into the generation of Industry 4.0. It holds the promise of increased flexibility in manufacturing, mass customization, better quality, and improved productivity (Zhong et al., 2017). Manufacturing systems in Industry 4.0 are updated to an intelligent level, and some Intelligent manufacturings, including cloud manufacturing and IoT-enabled manufacturing emerge (Tao et al., 2011; Zhong et al., 2017; Bohu, Lin, and Xudong, 2020).

Cloud manufacturing can be interpreted as an advanced manufacturing model under the support of cloud computing, the Internet of things, virtualization, and service-oriented technologies, which transforms manufacturing resources into services that can be comprehensively shared and circulated (Zhong et al., 2017). In recent ten years, researchers have drawn significant attention to cloud manufacturing. Figure 1-4(a) displays the number of cloud manufacturing articles by year from

2010 to June 2020 (resources from the Web of Science), which symbolizes an inclination that research on cloud manufacturing is becoming increasingly widespread. Figure 1-4(b) indicates the affiliations of researchers in cloud manufacturing and the review observed that the enormous majority of authors in the cloud manufacturing community are from China, a nation where manufacturing is viewed as a pillar industry in the national economy (He and Xu, 2015). More in detail, most active researches in cloud manufacturing come from Chinese universities, such as Beihang University, Wuhan University, Zhejiang University, and so forth. Outside mainland China, researchers are mainly from the University of Auckland, KTH Royal Institute of Technology, Hongkong university and etc. In summary, cloud manufacturing is becoming increasingly popular and accepted worldwide.



(a) Articles by year from 2010 to June 2020



(b) Published articles on cloud manufacturing by country

FIGURE 1-4: Articles on cloud manufacturing by year and country from 2010 to June 2020.

In a cloud manufacturing system, there are primarily two kinds of participants (Tao, Hu, et al., 2008): resource service providers who announce their resources, products, ability and manufacturing resource services to meet the others' requirements. Those who receive these resources are resource service customers (Tao et al.,

2011). The transactions between resource providers and consumers are performed in the cloud manufacturing service platform, and a specific service transaction process in (Yan, Cheng, and Tao, 2016) is shown in Fig. 1-5. As can be seen, resource consumers and providers post their requirements and resources to the system, and an optimal provider is selected for the transaction.

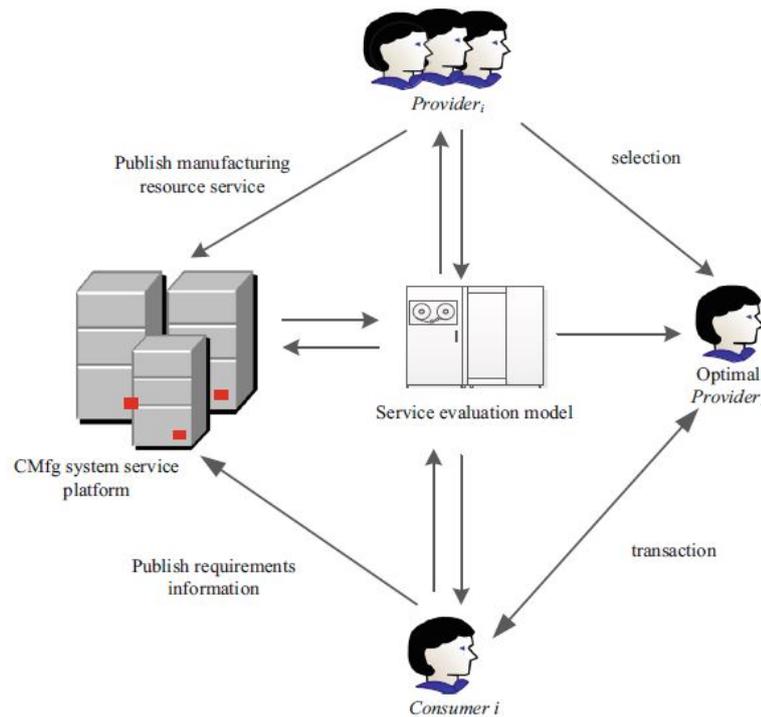


FIGURE 1-5: An example of service transaction flow (Yan, Cheng, and Tao, 2016).

Some practical case of the new manufacturing paradigms emerges in recent years, such as the swiss Biryuellefabrik (*swiss Biryuellefabrik* 2019). This system constitutes base enterprises with fundamentally complementary resources and competences. They concentrate on manufacturing novel products and prototypes, which cannot be accomplished without specific abilities. The orders are placed, and the parts of the task are distributed between the manufacturers by specialized brokers. The system is designed to execute orders together to supplement each other's competences and resources. The framework participants could not efficiently use their resources without working together (Szaller, Egri, and Kádár, 2020).

These developments also bring obstructions and challenges in manufacturing. Firstly, additional transportation and subcontracting costs can undertake the risk of

high expenses. In a cloud manufacturing system, the platform is built on the Internet, and there can be a lack of precise information between transactions, so transaction members have trouble learning specific, reliable, and complete information mutually. What is more, in a system where participants share resources and cooperate mutually, the participants' trustworthiness is always a critical factor in fostering transactions. In the next part, the notion of sharable resources is well studied in manufacturing.

1.4.3 Sharable resources in manufacturing

The service-oriented networked manufacturing known as cloud manufacturing, which drew much inspiration from cloud computing, was first proposed in 2010 to maintain complex and large-scale manufacturing problems (Liu et al., 2019b). Nowadays, the significance of cloud manufacturing to the manufacturing industry has been recognized from both academic research and practical industry implementation aspects. Up to date, academic research concerning cloud manufacturing falls in the architecture and platform construction of systems (Zhou et al., 2018; Škulj et al., 2017), the resource virtualization and use (Laili et al., 2012; Wang, Wang, and Gördes, 2018), and the scheduling and matching of resources (Tao et al., 2012), but the study on the service trust evaluation is relatively few.

Researchers with distinguishing backgrounds understand cloud manufacturing from different perspectives. Thus, many definitions of cloud manufacturing have been proposed; nevertheless, a standard one that all people agree with does not exist (Fisher et al., 2018). To better understand resource sharing in cloud manufacturing, we first go through its definition from the perspective of cloud computing, which helps investigate cloud manufacturing.

Manufacturing is a concept that migrates the concept, operation model, and technologies of cloud computing to the field of manufacturing rather than applies cloud computing to manufacturing (Xu, 2012). Therefore, cloud manufacturing is logically being interpreted as a manufacturing version of cloud computing (Liu et al., 2019b). Three aspects, including resources, application, and technology, are employed to differ cloud manufacturing from cloud computing (Liu et al., 2019b), namely:

- Resources: In the context of cloud computing, the resources consist of general computing resources (e.g., CPU, storage); In terms of cloud manufacturing, there are three different resources, which are general computing resources, as well as manufacturing-oriented computing resources, and pure manufacturing resources, such as machine, tools.

- Applications: Compared to cloud computing, cloud manufacturing consists of general computing applications and manufacturing-oriented computing resources and completely manufacturing applications.
- Technology: Cloud manufacturing maintains the general technology that cloud computing holds; at the same time, it has some other technologies, including the Internet of Things.

Thus, cloud manufacturing covers up all aspects in terms of resources, technology, and application compared with cloud computing. More in detail, Figure 1-6 shows the resources in cloud manufacturing, where resources in cloud manufacturing consist of manufacturing resources and manufacturing capability (Zhang et al., 2014). Manufacturing resource includes soft resource and hard resource where the soft resources are referred to as software, data, and knowledge, and hard resource consists of requirements of manufacturing, logistic, and so on. Manufacturing capability is a combination of manufacturing resources, human, and knowledge involved in the manufacturing process (Zhang et al., 2014).

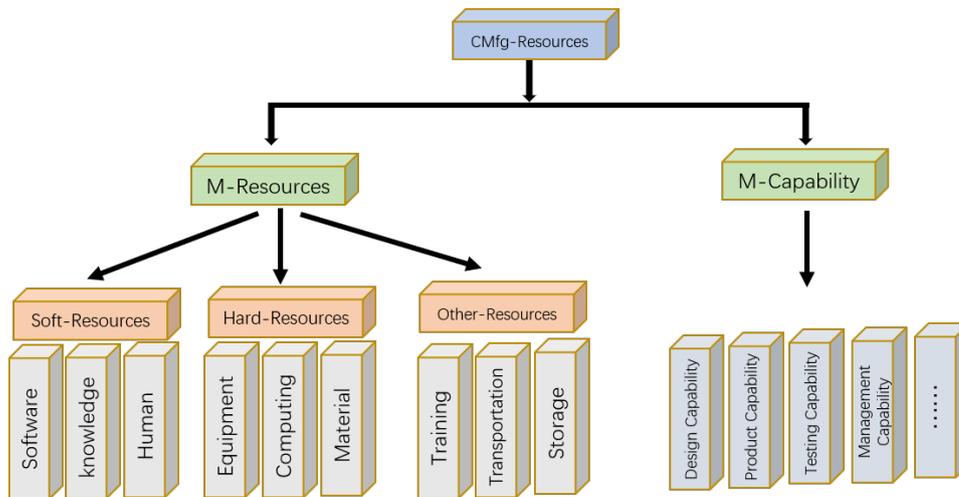


FIGURE 1-6: The classification of cloud manufacturing resources

Manufacturing resources and manufacturing capabilities can be encapsulated to form the cloud service, which will later be registered and published in the cloud manufacturing platform to provide services. Therefore, a cloud service generally provides two different services, namely, manufacturing resources and capabilities endowed from different enterprises to transform materials to products, i.e., pure manufacturing resource services, on the one hand. On the other hand, the operator uses computing resources and manufacturing-oriented computing resources to facilitate

resource planning and scheduling. Accordingly, there are ordinarily three components in cloud manufacturing, namely, service providers, service consumers, as well as cloud service providers (Yan, Cheng, and Tao, 2016). Notably, in distributed cloud manufacturing systems, we also say that two components consist of a cloud manufacturing system, that is, service provider and service customers (Li, Barenji, and Huang, 2018; Szaller, Egri, and Kádár, 2020).

It is necessary to understand the way of obtaining or providing services in cloud manufacturing. In terms of providing service to both consumers and providers, computing resources and manufacturing-oriented computing resources can generally be provided in the same way as general computing resources in cloud computing (Liu et al., 2019b). That is to say; consumers can unilaterally obtain these resources as needed without human interaction with service providers (Lu and Xu, 2015). However, it is incredibly complicated to provide pure manufacturing resources such as machine tools and robots. The main reason is that resource consumers commonly have to interact with resource providers to determine detailed information such as design solution and intellectual property ownership (Ren et al., 2015). Thus, resource sharing in cloud manufacturing involves two modes, namely the modes without direct interaction or involve direct interaction, respectively. From the perspective of service providers, the resource provider can completely transform the authority of its resources to the provider. That is to say, the resource provider only retains its ownership, and the operator manages everything on its behalf, on the one hand. On the other hand, the resource provider only registers its resources in the cloud manufacturing system, and manages and schedules everything (Liu et al., 2019b). It is necessary to note that resources in the latter situation are usually core manufacturing resources such as machine tools and robots, and usually, providers involvement is required. In this thesis, resource providers register their resources in the cloud manufacturing platform, and they communicate with resource customers directly to share resources.

1.5 Trust

In the large-scale resource sharing systems, numerous resource providers, resource clients, and resources are involved. Thus, finding the best resource provider is extremely challenging. For instance, cloud manufacturing services have the characteristics of interoperability, self-organization, and self-adaption (Zhang et al., 2014). However, it is still difficult for users and developers to discover, request, assemble, and use cloud manufacturing resources. In this circumstance, autonomous and flexible agents and multiagent systems are suitable tools for negotiating user access,

automating resource and service discovery, composition, trading, and cloud manufacturing resources. However, agents are usually mobile, goal-oriented, independent, etc., and much uncertainty stems from many factors, including randomness, incompleteness, insufficient evidence, and willingness can dramatically affect the decision-making process and efficiency.

As a result, the concept of trust is employed to stress these issues. Trust is a relationship between two entities in some environments. A commonly accepted notion to denote these two entities are truster and trustee (Nguyen and Bai, 2018; Pan et al., 2020). Scholars and practitioners widely acknowledge the importance of trust, and academic research on trust could date back to 1967 (Giffin, 1967; McKnight and Chervany, 1996) and even earlier. Trust is studied and applied in various disciplines, including sociology, psychology, economics, and computer science (Ruan and Durresi, 2016). As claimed, a straightforward way to comprehend something is by firstly proffering a definition. However, it is not an effortless task as the focus of trust differs from distinctive disciplines (McKnight and Chervany, 1996). Later in 1994, trust was employed in computer science, and a definition of trust was recently given as follows (Cho, Chan, and Adali, 2015),

- "Trust is the willingness of the truster to take a risk-based on a subjective belief that a trustee will exhibit positive behavior to maximize the truster's interest under uncertainty of a given situation based on the cognitive assessment of previous experience with the trustee (Cho, Chan, and Adali, 2015).

As the definition shows, trust is defined as the probability that a trustee would perform as committed to maximizing the truster's interest. Also, trust can be categorized into different classifications due to the varying criteria; for example, in (McKnight and Chervany, 1996), impersonal/structural trust, dispositional trust, and personal/interpersonal trust are three distinctive filed trust. What we want to highlight here is that in (Abdul-Rahman and Hailes, 2000), on the basis of behavior types, trust is further divided into functional trust and recommender trust. Functional trust describes how reliable an agent is when it is appointed to implement some tasks. An example in Figure 1-7 is applied to identify functional trust and recommender trust. When Alice decides to purchase resources from Bob, functional trust exhibits how reliable Bob would present the expected resource. Recommender trust usually measures how reliable an information provider is about its recommendation. We could also interpret the recommender trust as to how much we can trust the trust (Gambetta et al., 2000), for instance, Alice asks Charles for preceding quality estimation of Bob's resources, and the reliability of the provided estimation is considered as recommender trust. It has to be emphasized that recommender trust is distinct from

the perception of indirect reputation. In fact, indirect reputation can be known as a kind of functional trust provided by third-party agents, which is functional trust-2 in Figure. 1-7, and we appreciate recommender trust from a confidence perspective in Chapter 3.

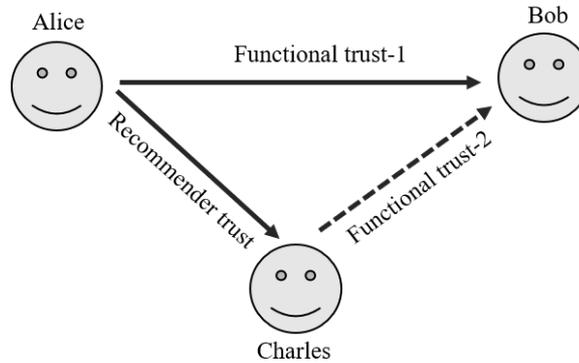


FIGURE 1-7: An example of functional trust and recommender trust.

Trust has various qualities, such as subjective, dynamic, asymmetric, context-dependent, transitive, composable, and so on (Sherchan, Nepal, and Paris, 2013; Nguyen and Bai, 2018). Similar to its definition, different disciplines highlight distinct aspects of their properties, and more details related to trust property can see (Sabater and Sierra, 2005; Sherchan, Nepal, and Paris, 2013). What I would highlight in this thesis are the following listed properties,

- 1.) Trust is dynamic. The value of trust might increase and decrease over time, interactions, and experiences (Staab et al., 2004). Moreover, the reliability of experiences might fade over time, and previous experiences might become obsolete or irrelevant (Sherchan, Nepal, and Paris, 2013). From another perspective, an agent's trust might go from genuine to malicious or the other way around to hide the actual performance or to achieve the maximum profit. In multiagent systems, trust evaluation should be performed periodically to guarantee that the estimated trust values are applicable at present. Many research articles are focused on modeling agents' dynamicity in computer science (Jøsang and Ismail, 2002; Jiang et al., 2016). Chapter 3 proposed a dynamic trust model that has declared the involved properties for agent trust estimation.
- 2.) The propagative of trust. For example, if Alice trusts Bob, and the latter trusts Charles, while Alice does not possess any information about Charles. Logically speaking, it is affirmed that Alice can obtain some degree of trust in Charles

according to its trust towards Bob and the trustworthiness that Bob has over Charles. Because of its propagative nature, trust information can be transferred from one agent to another in a social network, formulating a trust chain (Sherchan, Nepal, and Paris, 2013). Various trust models have taken advantage of the prerogative property, such as "the friends of our friends are also our friends" in (Yu and Singh, 2002a; Jøsang, Gray, and Kinatader, 2003). However, it is necessary to distinguish the propagative property from the transitive property, transitivity implies propagation, but the reverse is not true (Sherchan, Nepal, and Paris, 2013). That is to say, if Alice trusts Bob, and Bob trusts Charles, we cannot infer directly from the information that Alice trusts Charles. We employ the notion of confidence to investigate the propagative property in Chapter 3. Also, in chapter 4, the propagative feature is studied to emphasize that a message from a trusted information provider is pretty reliable.

- 3.) Trust is context-specific. Trust happens between two agents in a fixed context, that is to say, trust only exists in a fixed individual object, and trust may not exist if the object is distinctive. For example, Alice trusts Bob as a doctor; however, there might be no trust in Bob if it acts as a driver. Trust is different from the trust context, which customarily refers to the environment where a trust relationship exists (Toivonen, Lenzini, and Uusitalo, 2006; Liu and Datta, 2012; Wang, Li, and Liu, 2015). A trust model in a heterogeneous network is investigated in Chapter 4.
- 4.) Trust is asymmetric. If Alice trusts Bob, there is no rule to urge Bob to trust Alice. Much information results in asymmetric property since agents act different roles, goals, and preferences in the multiagent systems.

Trust is a notion with complicated characteristics. We investigate trust in resource sharing systems to enhance sharing efficiency and avoid uncertainty in this dissertation.

1.6 Goal and assumptions

As mentioned, much uncertainty stems from many factors, including randomness, incompleteness, insufficient evidence, and willingness can dramatically affect the decision-making process and efficiency. In this thesis, the concept of trust is employed to stress this uncertainty in resource sharing systems to select the best resource provider to improve efficiency and accuracy. Resource owners are modeled as intelligent agents that occupy a certain amount of distinguishing resources. Also,

each agent has specific goals and tasks to be accomplished, and when a task is on the waiting list, agents need to communicate mutually with others to accomplish the task. Thus, the task can be accomplished within this multiagent system, like companies trying to generate revenue from completing jobs. In the presented thesis, agents are motivated to perform as many tasks as possible and utilize resources as much as possible, but the financial aspects are not stressed in detail. However, an agent is capable of achieving a task unless it is used to perform excellently, which would result in high-trust value. This trust value functions as the core for a service provider selection. Figure 1-8 displays an example of the process of resource sharing. Multiple agents receive an ordered stream consisting of a series of tasks outside the system and collaborate by sending and receiving requests, offers and messages.

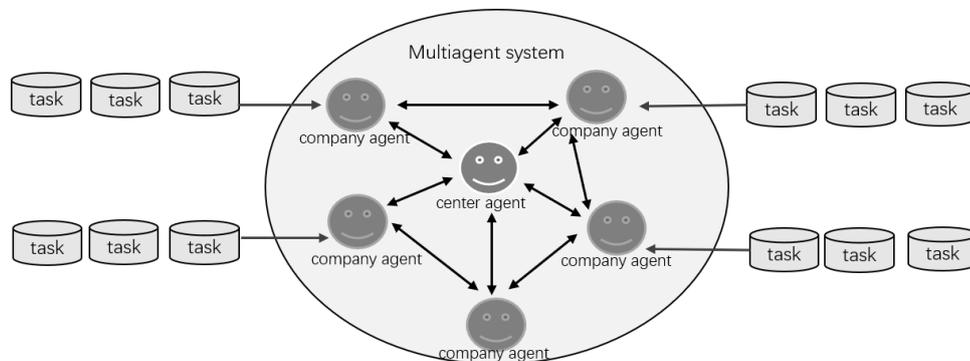


FIGURE 1-8: An example of manufacturing resource sharing.

Since this thesis' principal goal is to investigate the effect of trust in resource sharing, and for simplicity reasons, resources are considered to be continuously served in the models. Interactive feedback-based trust is quantified as a numerical number to reveal the agent's trustworthiness (Sabater and Sierra, 2002). Generally, interactive feedback refers to the amount of positive and negative interactions or the interaction's evaluations (Song and Hwang, 2004). However, it is also challenging to determine the exact value, and the main reasons range from identifying influential factors, achieving the evaluations to reach trust values, and choosing with which agent to interact. Some indicators, such as quality of service (QoS) and security of the website (SoW), are fuzzy factors that are difficult to evaluate accurately. Also, agents' performances differ from periods in the dynamic environment. Furthermore, as stated, trust is not a black and white matter. Much uncertainty exists in providing an agent's trustworthiness; for example, it is hard for us to determine an agent's trust employing only one interaction. Numerous factors determining the trust value make the system a dynamic, multi-dimensional, and multi-level index task.

A sound reputation mechanism is expected to reveal participants' public reputation in a large-scale system involving many suppliers and customers. It can be used when the consumer and the provider have no transaction history or with inefficient direct trust. This mechanism should be flexible enough to get updated and provide accurate reputation ratings regularly. In fact, the obtained third-party information regarding one's reputation would greatly influence participants' decision-making process to look for high-quality service and accurate information. However, trusting the information received directly can cause significant failures. Firstly, trusted participants might contribute some misleading information because of the dynamic environment. For instance, Alice trusts Bob strongly while Bob does not realize the environmental changes. Bob's information thus is no more trustworthy. Secondly, a group of agents can present misleading information jointly because agents are capable of learning other's attitudes. Thirdly, an information provider needs to be sure of its information. Thus, a reputation has to be proved accurate before usages. In this thesis, we tried to investigate recommender trust taking advantage of all the third-party information received. This action is expected in multiagent networks within which reputation plays an essential role in agents' transactions. What is more, agents expect to acquire reliable information, and mechanisms are needed to motivate agents to contribute accurate information in a multiagent environment. As a result, this thesis's primary goal is to produce and maintain secure trust estimation approaches that optimally function in multiagent systems with dynamic environmental attributes such as agent goals, credibility, and the proposed trust model is applied for resource sharing in a distributed cloud manufacturing system. The main objectives are categorized as follows:

- 1.) Understand the uncertainty in resource sharing precisely and design a near-perfect direct trust-based framework in a fully distributed environment to accurately consider the involved factors and provide an excellent trust estimation manner.
- 2.) Introduce a dependable indirect reputation mechanism by analyzing the received information to increase its accuracy and eliminate group deception and environmental changes.
- 3.) Apply agent trust-based resource sharing approaches in cloud manufacturing to guarantee agent satisfaction to the achieved resources.
- 4.) Investigate a recommender system-like resource discovery approach in cloud manufacturing, taking advantage of the propagation of trust and agents' tasks.

This thesis considers trust in different chapters and analyzes the details within different environments to fulfill resource sharing. In the direct trust-based framework, we mainly focus on uncertainty, the parameters involved in this framework reflect the resource suppliers' trustworthiness, and the proposed direct trust is appropriate to dynamic multiagent systems. We also apply the others' recommendations to capture trust, and different approaches are employed to obtain the exact trust.

1.7 Conclusion

This chapter presented a general introduction to resource sharing problems. We first reviewed that sharing is a widespread notion that exists in our practical daily life, and its development is then investigated according to the drives of resource sharing. We also studied the shared resources in modern society, including tangible resources, intangible resources, and human resources.

In what follows, we introduced resource sharing in manufacturing and cloud manufacturing. Cloud manufacturing can be understood as an advanced manufacturing model under the support of cloud computing, the Internet of things, virtualization, and service-oriented technologies. Since numerous resources involving in cloud manufacturing and multiagent systems are appropriate tools that have been employed to facilitate manufacturing resource sharing.

In the agent-based manufacturing resource sharing systems, agents' trust plays an imperative role in selecting trustworthy resource provider agents. We, therefore, introduced the definition and characteristic of trust and emphasized the uncertainty problems. We concluded this chapter with the goals and assumptions of this thesis.

Chapter 2

Trust management in resource sharing systems

Trust received increasing attention owing to the critical roles it performed in resource sharing. Following this attention, many literary works have contributed to this field. Our primary concern in this chapter lies in discussing more specific multiagent systems for resource sharing based on the analysis of the previous fundamental works. This chapter starts with an introduction to the multiagent system and then applies the multiagent system to cloud manufacturing. After that, this chapter introduces the previous work on trust in multiagent systems and cloud manufacturing. This chapter ends with a summarization of the emphasized issues. In a word, the discussion throughout this chapter comprises a crucial introduction to the previous work of the entire thesis.

2.1 Agent-based cloud manufacturing

The term agent is famous in artificial intelligence and has a wide variety of applications. This section focus on two main aspects, first, more details about agents and multiagent systems are discussed. Afterward, based on the characteristic, we apply agents to cloud manufacturing and investigate the term of agent-based cloud manufacturing.

2.1.1 Agents and multiagent systems

The intuitive way to understand agents should firstly furnish a definition. However, it is challenging to present a specific interpretation that all researchers accept, and the main reason is that the agent's behavior has a different emphasis in varying fields, for example, (Srinivasan, 2010) and (Ye, Zhang, and Vasilakos, 2016) stress the properties from the perspectives of communication and reasoning respectively

to define an agent. In the literatures (Wooldridge and Jennings, 1995; Dorri, Kanhere, and Jurdak, 2018), a commonly accepted definition is given as follows,

Definition 1 (Agents) *An agent is an autonomous entity placed in an environment and senses the environment using various observation parameters, and it is capable of acting with intelligence on behalf of another entity.*

More in detail, entities are modeled as agents, which can be software or hardware components. They are placed in an environment, perceive and receive information from the environment or their acquaintances to perform appropriate actions for the fulfillment of achieving a determined goal. Furthermore, in (Wooldridge, 1999), an abstract figure of view of agents is given as Figure 2-1 shows. As is exhibited, an agent is capable of perceiving information from the environment. Correspondingly, it can produce corresponding reactions in response to the changes.

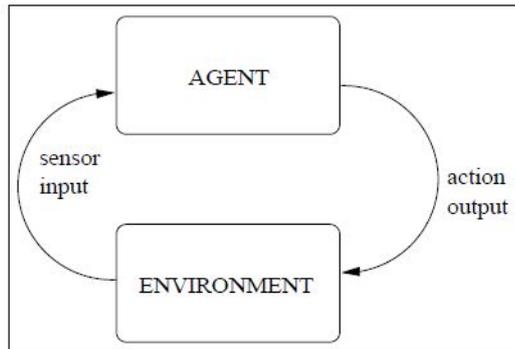


FIGURE 2-1: Agents and the agent environment (Wooldridge, 1999)

Here, it is necessary to stress that agents are different from intelligent agents. Agents could have distinguishing properties due to the varying needs. In brief, an agent could have the following general properties due to different environments (Dorri, Kanhere, and Jurdak, 2018).

- **Autonomy:** Operate on an owner's behalf but without any interference from that ownership entity. An autonomous agent can process information within a restricted domain giving it autonomy and then taking action based on the rules it has been furnished or acquired.
- **Sociability:** Agents can communicate mutually to exchange knowledge to improve reactions in achieving their goals.

- Proactivity: Agents can use the acquired information to reason and predict possible reactions of other agents to make the most beneficial actions.

Typically, one single agent system should be simple to resolve particular problems. The agent is assigned as a controller and collects information, processes optimize, and makes decisions. However, it has too many constraints; for instance, weak robustness. Thus, multiple agents usually work collaboratively to overcome the limitations, solve more complex problems, and benefit greatly from the notion of agents, the system is known as multiagent system.

Multiagent systems are composed of multiple agents and their environment (Stone and Veloso, 2000). The concept of multiagent systems has been studied as an independent field since about 1980 (Wooldridge, 2009). The rapid development and popularity of computers and Internet result in the widespread adoption of multiagent systems in the mid-1990s (Wooldridge, 2009).

The conspicuous features of a multiagent system, in summary, including high efficiency, time and energy saving, flexibility, and privacy, make the system an appropriate tool to solve complex problems (Wooldridge, 2009). In a multiagent system, an agent usually has limited capabilities conferred and restricted by the design or energy (Coelho et al., 2017). When facing a complex problem, it is often disaggregated into multiple smaller tasks and each of which is assigned to an individual agent separately (Dorri, Kanhere, and Jurdak, 2018). At this moment, an individual agent can process the assigned task, makes decisions and take actions autonomously with the information perceived or interrogated from the environment and acquaintance. It has to be emphasized that agents are considered to have local views, and logically, it is generally unable for one agent to perceive the entire system's overall information, which has greatly improved the privacy of agents in multiagent systems (Such, Espinosa, and García-Fornes, 2014). Compared to solving the complex problem in a centralized way, dismembering the complex problem achieves the purpose of saving time, energy, and consumption.

Graph theory, as is known, has often been employed to perform mathematical analysis on multiagent systems in computer science (Merris, 1994). Generally, a vertex in the graph is used to represent an agent in multiagent systems, and the relationships between two agents are modeled as edges. An edge indicates that the two agents are communicating with each other. A multiagent system in a directed network is usually with switching topologies since agents in the system have external interactions, and agents are allowed to enter and leave the system freely, just as customers and sellers in e-commerce systems (Zheng and Wang, 2012). Besides, the final decision made by an agent applies to the corresponding graph might change

the relationship of the edges. More details of graphs related to multiagent systems can refer to (Merris, 1994).

Because of the features of multiagent systems declared above, the multiagent system has full applications in various fields. According to the survey presented in (Dorri, Kanhere, and Jurdak, 2018), multiagents are mainly applied to five research domains, as is shown in Figure.2-2, and they are computer networks (including for cloud computing (Sim, 2011), social networks (Gatti et al., 2013; Ma and Zhang, 2014), security applications (Gorodetski and Kotenko, 2002), and routing (Bendjima and Feham, 2015)), Robotics (Cruz and Yu, 2017), Modeling complex systems, City and the built environment (Anagnostopoulos et al., 2020) and smart grids (Klaimi et al., 2018). In this thesis, we propose to apply agents to cloud manufacturing to facilitate transactions in cloud manufacturing.

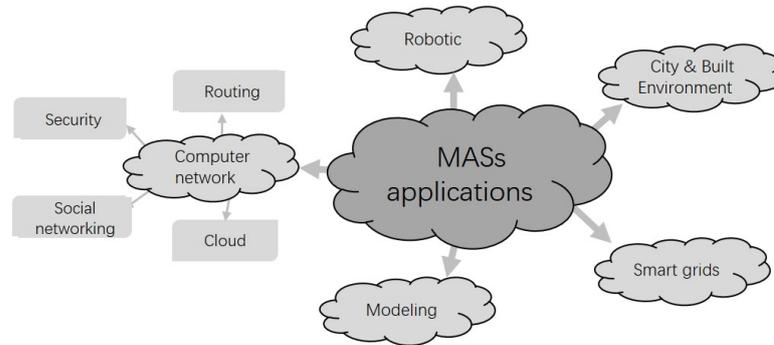


FIGURE 2-2: Research domains of multiagent systems

2.1.2 Multiagent cloud manufacturing

The literature is rich in applying agents in manufacturing (Vatankhah Barenji and Vatankhah Barenji, 2017; Arbib and Rossi, 2000; Hussain and Ali, 2019), and also, many attempts are conducted to combine agents and clouds. Cloud computing and agents are complementary technologies. The features of Cloud computing can provide advanced computing properties for multiagent systems. Conversely, applying multiagent systems to the Cloud platform makes it easier to combine different functionalities, such as inference and learning ability (Prieta et al., 2019). Talia combines cloud computing and agents from two general perspectives (Talia, 2011). Namely, 1) agents use clouds where multiagents use or directly deployed in cloud computing services, and 2) cloud computing systems adopt agents to manage their services. This work is based on a limited number of references at that time, and it is somehow general and straightforward, but it is a positive feature in some way (Prieta

et al., 2019). In (Prieta et al., 2019), a survey of agent-based cloud computing applications is carried out, and the authors propose to classify the relationship between cloud computing and multiagent systems from the perspective of Cloud computing. Their work proposes the use of the roles proposed by NIST (Liu et al., 2011) as a new taxonomy for classifying systems that combine multiagent systems and cloud computing.

Cloud manufacturing is a concept that migrates the concept, operation mode, and technologies of cloud computing to manufacturing, and it is rational and profitable to combine agents to cloud manufacturing (Wu et al., 2013b). From another perspective, manufacturing resources satisfy consumers in the form of services in cloud manufacturing. The cloud manufacturing resources, including manufacturing resources and manufacturing capability, are encapsulated as manufacturing services. Service platform manages these services, such as integration, sharing, and distribution, and enables the service's interaction, collaboration, composition, trading freely (Zhao, Luo, and Zhang, 2020; Zhao, Wang, and Zhang, 2020). A service is defined as a network unit, which is independent, open, and non-related with systems. The service is established in a distributed environment, which has advantages in flexibility, reusability and sustainability (Yu and Han, 2006). A Cloud manufacturing platform can meet consumers' requirements by providing manufacturing service access from providers.

The research activities in the area of cloud manufacturing mainly focus on the following five aspects nowadays (Ghomi, Rahmani, and Qader, 2019), 1)designing the architecture and platform of cloud manufacturing, 2)resource description and encapsulation, 3)service selection and composition, 4)resource allocation and service scheduling, and 5)services searching and matching. In summary, we can say that in Cloud manufacturing, the main focus of research is on the efficient use of cloud manufacturing resources. Autonomous and flexible agents and multiagent systems are suitable tools for negotiating user access, automating the resource and service discovery, and composition, trading, and harnessing of Cloud resources. Autonomous agents can make cloud manufacturing smarter in the interaction with consumers and more efficient in allocating processing and storage to applications.

In the state of the art, there are some studies on agent-based cloud manufacturing. Notwithstanding, this is an emerging trend, and the studies and applications in this field become gradually increasing. The term agent is used for scheduling, including matching, discovery, selection of manufacturing resources in cloud manufacturing (Liu et al., 2019c). In (Zhao, Luo, and Zhang, 2020), a combination

of service and agent is proposed and is applied in the cloud manufacturing simulation platform to show the phenomenon in cloud manufacturing. Similarly, service agents in a simulation platform can drive the manufacturing service to form a self-organization network to conduct transactions and cooperate spontaneously. Thus, the service network of a service agent and the collaboration network of service agents are formed (Zhao, Wang, and Zhang, 2020). What is more, security and trust are critical issues in Cloud manufacturing as data are deposited, accessed and run on machines that are not owned or directly managed by owners of the data. The agent-based model can be very profitable for trust estimation in cloud manufacturing. For example, in (Szaller, Egri, and Kádár, 2020), a distributed collaboration framework of manufacturing agents is introduced, where the members with resource shortages can request resources from others, divide requests among multiple agents, reorganize their production to be able to complete a request and cancel an undertaken task if needed. In (Meng et al., 2014), a five-dimensional trust evaluation system is built, and a reputation-based trust evaluation algorithm is also proposed for agent trust estimation in cloud manufacturing.

Taking inspiration from service agents, where service agents can be interpreted as an intelligent encapsulation or service driver of manufacturing service (Zhao, Luo, and Zhang, 2020), we also adopt manufacturing service agents in this thesis. These service agents can act as service providers as well as service consumers (or service clients). Particularly, the service agents in cloud manufacturing often perform both the two roles to guarantee they have profitable opportunities (Zhao, Luo, and Zhang, 2020). However, the manufacturing resources are varied, and their description and access methods are much more complicated than the computational resources in the cloud computing (Zhao, Luo, and Zhang, 2020). In a cloud manufacturing platform, a manufacturing pool is established to maintain the manufacturing resources. These resources and some digital operating interfaces will be encapsulated as virtualized manufacturing resources and resource functions, which are later achievable as services through the application system (Zhao, Luo, and Zhang, 2020). However, it is not our focus in this thesis. Our goal is to characterize the service agent's trustworthiness by studying the performance in past transactions and analyzing the uncertainties caused by incompleteness, randomness, and deception of the system, then choose the most trustworthy agent to raise transaction to obtain maximum benefits and to minimize risks. Thus, the communication architecture of service agents are more important.

Manufacturing service agents have the ability to communicate independently. The communication architecture of manufacturing service consists of four layers

which is shown in Figure 2-3 (Zhao, Luo, and Zhang, 2020): they are manufacturing resource layer, manufacturing service layer, manufacturing service application layer, and network layer. The network layer protocol is not restrained, and it can use HTTP, TCP/IP, UDP, and other protocols. The manufacturing resource application agent, which can be regarded as a high-level function of manufacturing services. A manufacturing service agent has the ability to raise complex functions, such as negotiation. However, the manufacturing service layer takes actions based on manufacturing services, while the manufacturing services are typically based on manufacturing resources. In this way, the manufacturing service agent can communicate with each other, query and share the possible information together in cloud manufacturing.

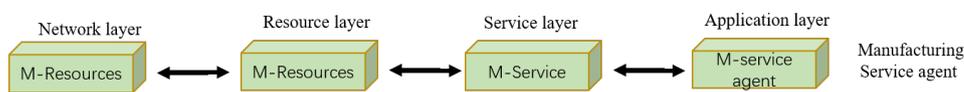


FIGURE 2-3: Manufacturing service agent communication architecture

2.2 Trust models

Coinciding with advantages of multiagent systems also spawned many challenges. First of all, agents in the systems are all goal-determined entities. That is to say, most of the agents are self-interested, and the chief purpose is to achieve their tasks. From another perspective, some agents are inherently deceptive, and it is the nature or personality of these agents to provide uncertain information. Thus, it is unstoppable that some misleading information would be provided for some agents in the systems. Furthermore, multiagent systems are mostly considered to be open, where agents can freely enter and exit the system, as we usually experience, users can repeatedly register a new account to reoperate in the system of Amazon. Such qualities make the multiagent system full of uncertainties, and decision-making agents are exposed to risks during interactions. In order to avoid decision-making risks and reduce personal utility losses, trust measurement is regarded as a useful risk avoidance tool in multiagent systems.

The notion of trust is so vital and has been investigated extensively in many social science research disciplines, as well as in people's daily life (McKnight and Chervany, 1996; Ruan and Durresti, 2016). However, why trust is imperative in the multiagent system? Generally speaking, trust is central to how agents interact with

each other, and it makes cooperative endeavors take place to reach any positive interaction feedback outcomes (Ruan and Durresi, 2016). In particular, when an agent is involved in online social communities and tries to raise interaction with the others, then the agent is considered of complicated human behaviors, and its trust can be affected by many factors. For instance, agents can be modeled to conduct commercial trade on behalf of human users in e-commerce. Agents have to decide with which agent to raise an interaction according to their acquired information. Anyhow, it has to deal with the uncertainty contained in the system and bear the consequences of being deceived. The following examples are given to indicate the importance of trust in multiagent systems.

Consider a cloud-based resource distribution multiagent system where load balancing is performed to utilize the computational resources efficiently (Premarathne and Rajasingham, 2020). Agents are assumed to be mutually trustworthy (Mohamed and Al-Jaroodi, 2015). In such a system, cooperative load balancing solutions rely on the trustworthiness of agents (Bajo et al., 2016). Agents retrieve data and share useful information about the runtime features (e.g. CPU usage, current load level etc.) to coordinate their tasks. In such a system with malicious agents, misleading information on the load conditions is shared. As a result, load balancing will fail (Das and Islam, 2011). Therefore, in such multiagent systems, the corporate decision making for task coordination relies on the mutual trustworthiness of the agents. Also, the reliability of the cooperative decision-making on load balancing is significantly affected by the malicious performances of some agents (Papadimitriou and Garcia-Molina, 2010). The trust estimation will thus be suitable to distinguish and discard malicious agents.

Let us suppose there is another task allocation system in which agent Alice needs a particular resource to accomplish its tasks. Naturally, it asks its friends if they have the required resources to meet its needs. Coincidentally, a group of agents, for example, Bob, Charles, and David all have the relevant resources that Alice required. As self-benefit agents, they need to perform their best to obtain the authority to cooperate with Alice. At this moment, the question lies in, to whom should Alice better distributed its task. As a rational agent, before Alice decides to negotiate with these agents or discuss the specific contact, it should first acknowledge each agent's trust and, mostly, it will cooperate with the most reliable agent.

Whatever resource Alice chooses to purchase subsequently, although all resource provider agents may provide substantial resources. However, uncertainty in decision-making will still exist due to the above analysis. In other words, Alice might obtain resources from a provider agent who furnishes uncertainty or a fake resource that

does not match the claimed quality. Alice must tolerate the risks to a certain extent caused by inconsistent resources while arising interactions. Therefore, proper trust management can help agents interpret uncertainty, reduce risks, and maximize profits.

2.2.1 Trust management in multiagent systems

Trust was first introduced in computer science in the 1990s (Marsh, 1994). Since then, numerical mathematical models have arisen for trust estimation in multiagent systems. Of these trust models, they can be summarized by different standards. For instance, based on the theoretical knowledge used for trust estimation, trust models could be logic models that are mainly based on logical (Liau, 2003). Numerical models that the trustworthy can be represented by a real number, which can also be interpreted as a rate of successful interaction. Generally, this value is in the range of 0 to 1 (Jøsang and Ismail, 2002; Teacy et al., 2006). When considering the received information for trust estimation, the trust models could be categorized into two conventional trust models. Namely, firstly is recommendation based and reputation-based trust models and follows by evidence-based and policy-based trust models (Basheer et al., 2015). Here we have to explain what exactly these standards are. Including what they are? To be precise, what are direct trust, indirect reputation, social cognitive information (Shmatikov and Talcott, 2005), and organizational trust. For us, we would establish numerical models based on direct experience and indirect experience for trust estimation in multiagent systems. Table. 2-1 shows some of the necessary trust models and Figure.2-4 indicated the information as well as the trust models in this thesis.

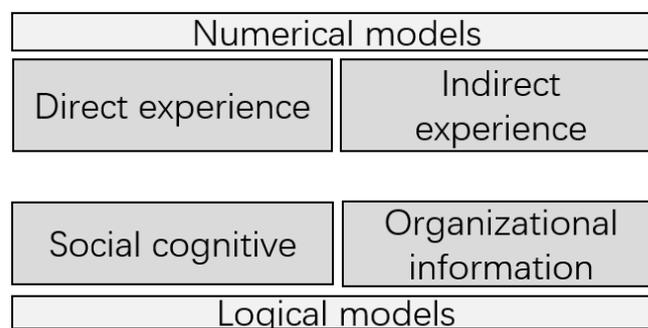


FIGURE 2-4: Trust models and the related information

It is necessary to emphasize the meaning of the headers in Table. 2-1. The details are showing as follows,

TABLE 2-1: Trust models

	Structure	Evidence	Weight	Incentive	Frequency	Quality	Learning	Dynamic changing
BRS (Jøsang and Ismail, 2002)	Distributed	Indirect	x	x	x	x	x	Static
TRAVOS(Teacy et al., 2006)	Distributed	Indirect or Direct	Dynamic	x	x	√	√	Static
FIRE(Huynh, Jennings, and Shadbolt, 2006)	Distributed	Indirect & Direct	Static	x	x	x	x	Static
Yu & Singh (Yu and Singh, 2002a; Yu and Singh, 2002b; Yu and Singh, 2003)	Distributed	Indirect	Dynamic	x	x	x	x	Static
(Yu, Kallurkar, and Flo, 2008)	Distributed	Indirect	Dynamic	x	x	x	x	Static
Wang and Singh (Wang and Singh, 2006; Wang and Singh, 2010)	Distributed	Indirect	x	x	x	√	x	Static
(Basheer et al., 2015)	Distributed	Indirect	x	x	x	√	x	Static
ACT model (Yu et al., 2014)	Distributed	Indirect & direct	Dynamic	x	x	x	√	Static
Zuo and Liu (Zuo and Liu, 2017)	Distributed	Indirect	Static	x	x	x	x	Static
ITEA (Elham, HosseinNikravan, and Zillesa, 2019)	Distributed	Indirect	Dynamic	√	√	x	√	Dynamic
Adaptive Trust (Shehada et al., 2018)	Distributed	Indirect & Direct	Dynamic	√	√	x	√	Dynamic

- **Structure:** The structure of multiagent systems can be either centralized or distributed. In a centralized system, it is the controller that restores information, data, and experience, and the controller takes responsibility for trust estimation since it has an overall knowledge about the system; on the contrary, information and data are stored by an individual agent in a distributed environment. Agents would interact mutually to achieve trust estimation information in a distributed environment since they only obtain local knowledge. Compared to the centralized environment, a distributed system performs more robustly, privately and safely.
- **Evidence:** Evidence is referred to as direct experience or indirect experience; The direct interaction between a truster and a trustee can be interpreted as a direct experience. Generally, direct experience is sufficient for trust estimation. However, considering indirect experience is profitable if with insufficient direct experience or in dynamic systems.
- **Weight:** When both direct experience and indirect experience are taken into consideration, it is a problem to decide their weight. Of the previous studies, to set fixed weight, machine learning approaches are employed to determine the weight between direct experience and indirect experiences.
- **Incentive:** Indirect experience is always required when the direct experience is insufficient or in a dynamic environment; However, a truster not only desires to achieve indirect experience but also to attain reliable indirect experience. Thus, the incentive should be employed to motive agents to provide reliable information.
- **Quality:** This parameter is used to check if the received information is of good quality and if it can be put into usage.
- **Frequency:** The frequency determines if the received information is certain; Logically, the experiences originated from multiple interactions are of high confidence.
- **Learning:** This factor is adopted to observe if an agent is smart enough to learn from the other to achieve reliable trust estimation values.
- **Dynamic changing:** A system can be dynamic; that is to say, in a stable system, an agent performs consistently as it first registered in the system. However, an agent used to collect information, data, and experience, and will always try to act profitably.

the beta probability density functions that rely on the amount of negative and positive interaction feedbacks (Jøsang and Ismail, 2002). The probability expectation is used to express the trust value. TRAVOS models trustworthiness probabilistically through a Beta distribution computed from the outcomes of all the interactions a trustor has observed, and it uses the probability of successful interaction between two agents to capture trust (Teacy et al., 2006). The literature of (Basheer et al., 2015) adopts the number of positive interactions and certainty to measure trustworthiness. Also, some references evaluate the direct trustworthiness by taking advantage of recent performance, which is expressed by real numbers in the range 0 (or -1) to 1. In the literature (Shehada et al., 2018; Elham, HosseinNikravan, and Zillesa, 2019), they achieve trust through the evaluation of recent performance and historical evaluations. SecuredTrust (Das and Islam, 2011) model trust by assigning dynamic weights to historical trust and current performance. FCTrust average the evaluations to estimate the trustworthiness (Hu, Wu, and Zhou, 2008). SFTrust weighs its performance by introducing a fading factor in dynamic multiagent systems (Zhang, Chen, and Yang, 2009). Of the two separate approaches, benefiting from the amount of successful and failed interactions is intuitive and straightforward, but it takes much energy (memory) to memorize the number of interactions. However, it is also challenging to decide the appropriate weight when considering recent and current performance.

A decision-maker also has to keep in mind that there is no trust without risk and uncertainty, and risk and uncertainty are requisite analytic preconditions of trust (Jøsang and Ismail, 2002; Castelfranchi and Falcone, 2010). In the range of 0 (or -1) and 1 in Figure 2-5, it is not only a matter of degree of trust, but it must be analyzed by scientific composition and reasons. Logically, trust assessment should encounter uncertainty, which deviates from cognitive impairment, insufficient evidence, randomness, and incomplete information (Berenji, 1988; Barber, Fullam, and Kim, 2002). For example, interactions raised recently would be of more significant importance or reference compared to that constructed rounds ago toward the last trustworthiness. Therefore, several approaches employ the definition of forgetting factor (or fading factor) to entertain these detailed characteristics (Zhang, Chen, and Yang, 2009). At the same time, various methods proposed to assign higher weights to the recent interactive feedback, especially in dynamic multiagent systems. Another reason for uncertainty derived from inadequate information also makes sense. From one aspect, we could not receive enough experience from full perspectives, and inadequate resources might involve much mental and affective attitude. From the other aspect, insufficient interactions can hardly convey an integrated evaluation. For instance,

an agent has only interacted with a prospective agent once, and whatever the interactive feedback is, it is unconvincing to make decisions compared to the agents with numerous interactions. Subsequently, the feature of the multiagent systems is also significant as rational agents might switch their performance to conceal their actual property.

Wang and Singh use evidence theory to present uncertainty for trust estimation (Yu and Singh, 2002b; Yu and Singh, 2003; Yu, Kallurkar, and Flo, 2008). They portray direct trust on the basis of interactive feedback, which is expressed by real numbers in the range 0 and 1. Subsequently, they set two different parameters in (0,1), and the value between the two variables is appointed as uncertainty. That is to say, vague assessments are the dominant reasons for uncertainty. From a different perspective, uncertainty can be understood as the conflict of positive and negative evaluation. In Zuo and Liu's model (Zuo and Liu, 2017), they adopt a three-value opinion (belief, disbelief, uncertainty) in the opinion-based model for trust estimation, which is very similar to evidence theory. However, as explained, uncertainty stems from cognitive impairment, inefficient evidence, randomness, and incomplete information (Berenji, 1988; Barber, Fullam, and Kim, 2002). What is more, our purpose is not only to represent uncertainty during trust estimation but also to achieve meaningful information conveyed in the uncertainty. Therefore, a more exhaustive exploration and exploiting the uncertainty need to be carried out.

Indirect reputation

It is natural and reasonable to query for third-party information whenever direct experience is sufficient. However, it is risky to make a final decision relying entirely and directly on the gathered information in a distributed multiagent system, especially with deceptive agents involved. Thus, we need to make sure that the accumulated information is in a reliable way before putting it into usage. Therefore, the concept of **confidence** is adopted to capture the reliability of gathered information (Yu and Singh, 2003; Teacy et al., 2006; Wang and Singh, 2007; Wang and Singh, 2010; Ramchurn et al., 2003; Basheer et al., 2015; Shehada et al., 2018; Elham, HosseinNikravan, and Zillesa, 2019). We define confidence as a probability that the trust is applicable, supported by evidence, consistency, credibility, and certainty. However, most of these models only take some aspects into consideration as a separate trust model in multiagent systems.

Consistency– Firstly, some trust estimation models assume that most third-party agents are reliable (Jøsang and Ismail, 2002). Thus those opinions that are consistent with the majority can be regarded as reliable (Jøsang and Ismail, 2002; Das and Islam, 2011). Others identify unreliable agents by comparing them with direct experience

(Yu and Singh, 2003; Teacy et al., 2006). PeerTrust (Xiong and Liu, 2003; Xiong and Liu, 2004) employs personalized similarity measure to calculate the confidence of recommenders, and it applies this similarity measure to weight each feedback presented by third-party agents. However, PeerTrust has to regain all the interaction within recent time, and all interactions are weighted equally seems inappropriate, especially in dynamic multiagent systems. SecuredTrust (Das and Islam, 2011) overcomes the shortages and proposes to measure confidence with evidence conflicts. From our perspective, only by studying the similarity or conflict between evidence to determine confidence might hide implicit dangers. For example, a group of agents joint and cheat together. In this thesis, we define this characteristic of acquiring confidence from similarity or conflict as **consistency**. Consistency displays the differences or similarities of some evidence between other evidence. It is helpful, especially when those agents whose trust and certainty are unknown. However, most of our reviewed articles did not pay much attention to consistency for trust estimation in multiagent systems.

Credibility is widely applied in distributed multiagent systems to capture confidence. Most of the circumstances, credibility can be interrupted as a weight that indicates how much an agent is reliable (Elham, HosseinNikravan, and Zillesa, 2019). It is updated according to the provided information and the interactive feedback (Yu and Singh, 2003; Yu, Kallurkar, and Flo, 2008; Shehada et al., 2018; Elham, HosseinNikravan, and Zillesa, 2019). However, in the literature (Elham, HosseinNikravan, and Zillesa, 2019; Yu and Singh, 2003), the credibility of third party agents decreases gradually; that is to say, the truster would no longer accept the agents with sufficient interactions. In the literature (Yu, Kallurkar, and Flo, 2008), the Dempster-Shafter theory of evidence is used to reach trust. However, it is unfair and illogical that the credibility of the third-party agents increases if the feedback is a success, and the other way round. In a dynamic multiagent system, false messages could also be real information as the error is caused by dynamic performance. Thus, improving agents' credibility by current interaction might be questionable and not enough. From another perspective, measuring confidence by analyzing credibility seems to be insufficient, especially in dynamic multiagent system with deceptive agents.

Certainty is "a measure of confidence that an agent may place in the trust information (Basheer et al., 2015)." Measuring certainty can filter out reliable information efficiency (Bilgin et al., 2012). As stated in the previous part, trust always involves uncertainty caused by the lack of certainty, a lack of right and complete information, or randomness, etc. (Fraser, 2011). Consequently, the information that accompanied by different degrees of certainty may affect confidence. For instance, agent Alice always trusts agent Bob, yet agent Bob is uncertain about its information

towards some event. Thus, agent Alice would hesitate to entrust the collected information from agent Bob. In the literature (Ramchurn et al., 2003), the trust model uses fuzzy sets and assign certainty to each of the evaluations. In (Basheer et al., 2015), the model assumes that certainty and evidence are independent. That is to say, the third-party agents present not only their opinion but also the certainty towards the information.

From the perspective of opinions, where the opinion is defined as a combination of belief, disbelief, uncertainty, the uncertainty can be interpreted as confidence in the opinion. (Nogoorani and Jalili, 2016) proposed a method to determine the uncertainty in the framework of evidence theory. However, they understand uncertainty mainly from the conflict between positive and negative interactive feedback. Entropy is a measure of uncertainty in physics, and information entropy is used to characterize uncertainty in the literature (Anisi and Analoui, 2011). Similar to other papers explained, their uncertainty also comes from conflicts between feedbacks. As mentioned in this chapter, many reasons might convey uncertainty, including but not limited to the lack of certainty, a lack of right and complete information, or randomness. Therefore, in the following chapter, we analyze the certainty contained in the testimony caused by the above reasons from the perspective of entropy. In what follows, I would display some of the recent trust estimation models that related to confidence estimation.

Trust and reputation framework

FIRE is an integrated trust and reputation model for open multiagent systems according to Interaction Trust (IT), Role-based trust (RT), witness trust (WT) and certified trust (CT) (Huynh, Jennings, and Shadbolt, 2004a). IT is direct trust calculated by direct experience, RT, WT, and CT are about witness experience. FIRE presents good performance. However, the model does not stress uncertainty and assumes that all agents provide accurate information. Besides, many static parameters are required to put the model into use that limits its application (Noorian and Ulieru, 2010; Shehada et al., 2018). SecuredTrust is a dynamic trust computation model in multiagent systems (Das and Islam, 2011). The model addresses many factors, including differences and similarities. However, feedback credibility generated from similarity is used to capture the degree of accuracy of the feedback information, which is the same as consistency. However, certainty and personal credibility are not well stressed. Meanwhile, an incentive mechanism should be used to encourage agents to be reliable. The actor-critic model selects reliable witness by reinforcement learning (Yu et al., 2014). It dynamically selects credible witnesses as well as the

parameters associated with the direct and indirect trust evidence based on the interactive feedback. Nevertheless, when only one known witness is available for the indirect reputation, bootstrapping error occurs (Rishwaraj, Ponnambalam, and Kiong, 2017). What is more, capturing uncertainty in a perfect trust estimation model in a dynamic system is necessary. By the analysis of uncertainty, we could obtain the degree of certainty and filter out reliable agents efficiently. An adaptive trust model for mobile agent systems is established. The model combines direct trust and indirect reputation, and also assess the credibility of witnesses. Moreover, the model considers uncertainty when generating direct trust is reasonable. However, witnesses' certainty is necessary for a trust model, especially in a dynamic trust model. Despite the advantages presented by the aforementioned models, these trust models in a system of multiagent still imperfect and need to be extended. The following chapter proposes an evidence theory-based trust model. It firstly improves the inappropriate witness credibility updating approach in the previous articles. When capturing direct trust, the model manages uncertainty from the perspective that cognitive impairment, inefficient evidence, randomness, and incomplete information (Berenji, 1988; Barber, Fullam, and Kim, 2002). What more, consistency, credibility and certainty are declared to achieve confidence. An encouragement mechanism is used to inspire agents providing reliable information.

2.2.3 Trust in resource sharing systems

Resources sharing remains imperative, and it carries many challenges simultaneously. Among the essential unresolved challenges encountered in sharing resources are maintaining trust, transparency, liability, and insurance (Daudi, 2018; Daudi, Thoben, and Hauge, 2018), and we would main study the notion of trust in resource sharing.

Many studies have adopted the concept of trust in decision-making, and trust is the basis of social coordination and cooperation and considered by many to be the very foundation of high-functioning societies (Argandoña, 1999; Chang et al., 2014). For example, in (Chard et al., 2011), the authors proposed leveraging trust relationships to form a dynamic "Social Cloud," thereby enabling users to share heterogeneous resources within the context of a social network.

In cloud manufacturing, the cloud manufacturing services, which have the same or similar capabilities, perhaps do not have identical quality. Therefore, a prerequisite for a transaction between trading entities, including the cloud manufacturing service providers and clients, is to have a good relationship of trust. There are few trust estimation models in this field, and some of them are listed and explained as follows,

In (Yang et al., 2019), a trust evaluation model for cloud manufacturing, which is based on service satisfaction by monitoring the data of historical service evaluation, is conducted. A comprehensive value of trust comprises of three parts, and they are direct satisfaction, indirect satisfaction degree provided by friends, as well as the platform satisfaction. In this trust estimation model, a time decay function was employed to portray service satisfaction changes over time. Meanwhile, service satisfaction volatility is used to correct the final trust value in a long-term transaction environment.

In (Li et al., 2014), a framework of the trust evaluation system in cloud manufacturing, which consists of seven-tuple and six evaluation indexes, was established. In this trust estimation model, the overall trust value consists of two parts, direct trust service, which is designed through a discrete method and recommendation trust service, which is extracted through the cloud theory model and cloud focus evaluation method. This model aims at the problem of reputation evaluation and selection of third-party regulatory roles, a trusted third-party regulatory service model of cloud manufacturing that considers security has been developed.

In (Yan, Cheng, and Tao, 2016), the authors reviewed trust estimation models and developed a new trust estimation model by introducing a third-party trust evaluation model on the basis of existing direct trust, recommended trust models, and a time decay factor. In this way, the proposed trust model becomes more practical.

In (Gan and Duan, 2012), the authors built a direct trust prediction model based on trading experience and weighted moving average method to resolve the trust evaluation problem between multiagent systems in cloud manufacturing.

In (Liu and Chen, 2019), a clustering-based and trust-aware approach is proposed to achieve higher prediction accuracy and recommendation quality in cloud manufacturing. The value of trust is further divided into local trust and global trust in this model, and more in detail, local trust is explicitly generated based on previous interactions, while global trust is estimated by averaging the local trust values from direct neighbors in the network.

In (Huang and Wu, 2020), an index system is established, and a fuzzy synthetic evaluation is adopted to compute the final comprehensive score of evaluation object. This trust evaluation aims to solve the problem of selecting cloud manufacturing resources to meet the needs of manufacturing enterprises.

Feng and Huang performed a hybrid collaborative filtering algorithm that has taken the location and set similarity into consideration to manage data sparsity (Feng and Huang, 2018). A hierarchical configurable reputation evaluation model was authorized based on the characteristics of the cloud manufacturing service platform.

In (Meng et al., 2014), aiming at providing a flexible enough trust model and reflecting agents' subjective preferences in cloud manufacturing, a five-dimensional trust evaluation system is proposed. The authors also introduced a reputation-based trust estimation algorithm that the direct trust rank, friend reputation, and platform reputation of the agent are integrated into the trust rank with the weight vector through that algorithm.

In summary, trust estimation models inside and outside the multiagent resource sharing systems and cloud manufacturing have some issues that need to be stressed.

1. Most of the previous research simply weight the trust indicators without considering the subjectivity, fuzziness, and uncertainty of trust, as well as the dynamic and big data characteristics of the cloud manufacturing service platform. For instance, trust originated from previous experience might fade over time. Moreover, insufficient interactions and randomness can also result in uncertainty. The evaluation index and relationships need to be further improved.
2. Although the recommended reputation considered in these methods, trust preferences of users are ignored. In other words, the resource client is most likely to choose from recommenders with similar trust preferences rather than other users.
3. Although the recommended reputation from similar clients is considered from some aspects, how much can the truster trust the recommendation is also a challenge. From one side, how to determine the received recommendation is correct. From another side, we have to see if the recommender is surely certainty towards its recommendation.
4. Under the circumstance that recommendation is adopted for trust estimation, and the previous works used to assume that the information providers are honest. However, it is impractical in real life. As a result, measures have taken to encourage information providers to present reliable information.
5. In the cloud manufacturing environment, the service quality provided by manufacturing providers in the long-term service is fluctuant, which will affect the trust assessment and the service cooperation relationship. However, few studies on improving the stability of the service cooperation between users in the cloud manufacturing field.

With the before-mentioned shortcoming for the previous trust estimation in cloud manufacturing, we propose two trust evaluation models in Chapter 3 and Chapter 4 to stress these issues. In chapter 3, trust uncertainty (items 1 and 4) caused by many factors is studied, represented, and used. Besides, we also employ the notion

of confidence to capture how to trust a friend's recommendation (items 2 and 3). In chapter 4, we mainly employing recommendation and product similarity for a trust estimation for service provider selection.

2.3 Conclusion

The present chapter has described multiagent systems and cloud manufacturing. Cloud manufacturing is a concept that transfers the concept, mode, and technologies of cloud computing to manufacturing. Cloud computing provides large-scale infrastructures for high-performance computing that are elastic since they are able to adapt to user and application needs. multiagent system represents a distributed computing paradigm based on multiple interacting agents that are capable of intelligent behavior. Cloud computing and agent are complementary technologies, the feature of cloud computing can provide advanced computing properties for multiagent systems, while cloud platform applies agent to make it easier to combine different functionalities. In this chapter, we applied agent to cloud manufacturing to make the cloud manufacturing smarter. We also explained the core issues in agent-based cloud manufacturing, which is to ensure the trust and security of agents. Besides, we analyzed the state of the art of trust, as well as review some basic trust estimation models in multiagent systems.

At the end of this chapter, we went over state of the art concerning trust estimation in resource sharing. We find that the existing references conducted trust estimation models, mostly in centralized structures. Meanwhile, agents are always honest and desire to share their information. In addition, uncertainty is not stressed in these trust models and accordingly, we emphasized the main issues about trust estimation in resource sharing discussed in this chapter.

Chapter 3

An integrated evidence-based trust model in multiagent systems

Multiagent systems have widespread applications thanks to the advantages bestowed by intelligent agents. This chapter presents an adaptive trust estimation model that can maintain excellent performance in a distributed multiagent system involving deceptive agents. In other words, service provider agents and information provider agents (witness), might determine to go from genuine to malicious or the other way around to occasionally hide their actual performance. Both direct trust and indirect reputation are integrated for trust estimation. This chapter begins by specifying direct trust that emphasizes the main factor of uncertainty, and subsequently, studies indirect reputation in the same way as to direct trust. However, indirect reputation derives from direct experience from the third-party agents must have reasonable confidence to be useful. This chapter analyzes trust from the social-cognitive perspective and introduces a near-perfect model that considers consistency, credibility, and certainty to capture the confidence. Furthermore, a trust model generally not only solicits the third-party agents for information but also for accurate information. We therefore motivate agents to be trustworthy, thanks to a credit mechanism. During the research parts of the contents are mainly based on the literature (Wang et al., 2018; Wang et al., 2020a).

3.1 Background

Trust and reputation assessment is a typical daily social behavior. Before purchasing distinct items, customers accumulate information to determine how the quality of the prospective product is. The final decision would be firmly on the basis of the aggregated trustworthiness of the seller to present high-quality products. We go to the bus station after work as we believe that the bus would always be on schedule. Similarly, the same mechanism fits for agents in multiagent systems, especially for multiagent systems with deceptive agents in a distributed environment.

Trust management refers to the relevant approaches to maintain and process trust information to formulate an attitude or opinion towards a prospective interactive partner. As discussed in the previous chapter, numerous protocols have been proposed to accentuate the trustworthiness issues in multiagent systems. Logical models describe how one agent trust another according to mathematical logic. The advantage of these methods is the use of robust methods that can conceive inferences and rationalize conceptual equipment. Social-cognitive approaches estimate the trustworthiness by taking inspiration from human psychology. Organizational models model trust by personal relationships in the system considered while numerical models understand trust from the perspective of mathematical probability (Liau, 2003; Jøsang and Ismail, 2002). Each of the four branches offers advantages toward trust estimation. Our proposed trust estimation model falls into the intersection of social-cognitive and numerical models, stressing that the statical trust should be modified from the social-cognitive perspectives.

Numerical models have a fixed goal to implement trust in automatic systems independent from the representational framework. Generally, the trustworthiness of an agent is represented by a real number, which implies the possibility of successful interaction, or of providing accurate information, or the expectation to perform as promised. The service provider with a high degree of trust is more likely to be selected. For instance, the trustworthiness of the agents Alice and Bob are a and b from the perspective of the agent Charles to undertake one task. In general, the task would rationally be delegated to Alice if $a > b$. To achieve the trustworthiness of an agent, statistical models need to gather some information with respect to the corresponding performance.

In general, in terms of the required information, we can distinguish direct experience from indirect experience. Direct trust is intuitive, and simple agents' sources yield from previous direct interactions with other agents and with the system. This experience is strongly on the basis of the agent's observant apparatus (Burnett and Oren, 2012; Yu et al., 2013). However, without uncertainty and risk, there is no trust (Castelfranchi and Falcone, 2010). Thus, the qualification of direct experience must involve numerous negative aspects, including the lack of sufficient knowledge, mistrust, hesitation, doubts, uncertainty, and so forth. Indirect experience sources can be articulately understood by the third-party agents' experience directly communicated or made by central or decentral mechanisms and types of general reasoning and deduction (Jøsang and Ismail, 2002; Jøsang, Ismail, and Boyd, 2007); However, as stated, the collected information from third-party agents must involve significant uncertainty, and the obtained information should have rational confidence to be serviceable. Consequently, we have to declare that the obtained information is in a

reliable way before trust estimation in multiagent systems.

In terms of distinguishing serviceable information received from the third-party agents, some assume that the majority agents are dependable (Jøsang and Ismail, 2002). Some recognize dishonest agents by figuring them with truster's direct experiences (Teacy et al., 2006) or filter out the unreliable recommendation and discount their evidence by assigning adaptive weights (Teacy et al., 2006; Jiang, Zhang, and Ong, 2013; Teacy et al., 2012; Elham, HosseinNikravan, and Zillesa, 2019). Although some of them present excellent security features, most of them assume that a static list of service providers is contacted. The assumption is that these service providers are all trusted and maintain their honest behavior consistently. What is more, the testimony provided by third-party agents is always reliable. However, there is no guarantee that service providers and witnesses would always present static behavior. They might decide to go from genuine to malicious or the other way around. Thus, the general assumption might be insufficient due to the dynamic behavior of agents in multiagent systems. Furthermore, most systems that we reviewed only consider either reliability, certainty, or consistency as a separate model in multiagent systems. These models do not exploit an integrated model concerning all aspects of trust for trustworthiness estimation in multiagent systems. To provide a near-perfect model of trust, especially in dynamic multiagent systems in a distributed environment, we extend the existing model concerning consistency, credibility, certainty. Meanwhile, we also motivate agents to provide valuable information for trust estimation in this chapter.

With respect to the factor of certainty of the witness to some provided testimony, some references decline that certainty and witness are independent (Castelfranchi and Falcone, 2010; Basheer et al., 2015) and trust should be interpreted from the perspectives of evaluation and certainty. However, we believe that certainty is almost always accompanied by evidence. For this reason, we propose to investigate certainty by interpreting uncertainty advertised in evidence thanks to the concept of entropy. The rest of this chapter is organized as follows; Some related works are introduced in section 2; the primary tool used in this chapter is manifested in section 3. The proposed trust model is detailed explained in section 4. Comparisons and conclusions are given in the last section.

3.2 Preliminary

In this section, we would mainly present the tool adopted in this chapter, namely the Dempster-Shafer theory of evidence. The main content goes to why we select this theory as well as some detailed introductions.

3.2.1 Why Dempster-Shafer theory of evidence

The theory of belief functions, also known as Dempster-Shafer theory of evidence or evidence theory, can be regarded as a generalization of the Bayesian theory of subjective probability. It was first introduced by Arthur P. Dempster (Dempster, 1967) and later developed by Glenn Shafer into a general framework for uncertainty measurement (Shafer, 1976). This theory enables one to fuse evidence from various sources and reach a degree of belief that takes into account all the possible sources. Compared to Bayesian theory, evidence theory has the advantage of describing ignorance and combine multiple sources of evidence, an example of picking balls from different boxes is used to show the efficiency to model uncertainty of evidence theory (Deng, 2016),

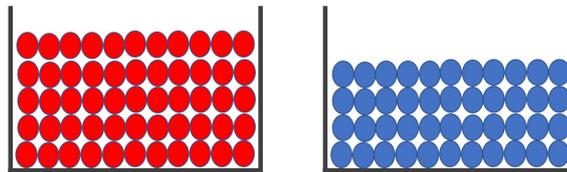


FIGURE 3-1: A game of picking ball which can be handled by probability theory

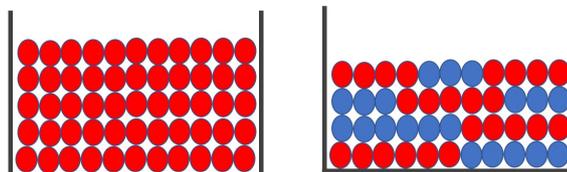


FIGURE 3-2: A game of picking ball where probability theory is unapplicable but evidence theory is able to handle.

There are two boxes shown in Figure 3-1. As can be seen, the left box is filled with several red balls, and the right box is filled with blue balls. However, the exact amount in each of the two boxes is unknown. A ball picking game is conducted to pick balls from the two boxes randomly, and we set a probability of 0.7 to select the left one, while the right one is selected with a probability of 0.3. Therefore, it can be rationally inferred that the probability of picking a red ball is 0.7 on the basis of probability theory, namely, $p(\text{red}) = 0.7$. Similarly, the probability of picking a blue ball is 0.3, that is to say, $p(\text{blue}) = 0.3$.

Now, the game situation is changed, which is shown in Figure 3-2. As shown, the left box is the same as described in Figure 3-1, which fills with red balls. To the

right box, we can find that not only blue balls but also red balls are filled in the box. However, the precise number of balls in each box is still unknown. In this circumstance, what keeps similarly is the probability of picking balls from the two boxes, namely, with a probability of 0.7 to choose the left box while with a probability of 0.3 to select the right one. The problem is that we would like to estimate the probability of picking blue and red balls, respectively. Due to the lack of knowledge in the right box, the probability theory is no longer suitable in the current case. However, we can easily use the Dempster-Shafer theory of evidence to present this information with ignorance involved.

3.2.2 Dempster Shafer theory of evidence

In this part, how to represent the ignorance as described above in the pick-up game, and more details related to Dempster-Shafer theory are introduced.

Definition 2 *The frame of discernment and BPAs*

Evidence theory is defined in the frame of discernment denoted by Ω which consists of n mutually exclusive and collectively exhaustive elements. The power set represented by 2^Ω is the set of all subsets of Ω which is represented by $2^\Omega = \{\emptyset, \{\theta_1\}, \{\theta_2\}, \dots, \{\theta_n\}, \{\theta_1, \theta_2\}, \dots, \{\theta_1, \theta_2, \dots, \theta_n\}\}$. The elements of the power set can be employed to describe propositions in terms of the actual state of the system, by containing all and only the states in which the proposition is correct. A basic probability assignment (BPA), also known as mass function, is adopted to assign a belief mass to each element of the power set. Mathematically, the function of BPA is defined as

$$m : 2^\Omega \rightarrow [0, 1]. \quad (3.1)$$

This function has two properties. Firstly, the mass function of the empty set is zero, namely,

$$m(\emptyset) = 0, \quad (3.2)$$

where \emptyset is the empty set. Then, the sum of all the masses of the remaining members of the power set is exactly 1, that is

$$\sum_{A \in 2^\Omega} m(A) = 1. \quad (3.3)$$

The mass $m(A)$ signifies the proportion of all relevant and available evidence that supports the claim that the actual state belongs to A . That also means that $m(A)$ does not imply any relationship with the subset of A . As defined, each element in

the power set has its own mass. Formally, a subset $A \in 2^\Omega$ such that $m(A) > 0$ is named as focus element, and the set of all focus elements is called core.

Let us use the Dempster-Shafer theory of evidence to present the ignorance described in Figure 3-2. There are two kinds of defined balls in the two boxes. Logically, our judgment could only base on the two balls. Thus, the frame of discernment could be presented by $\Omega = \{red, blue\}$. Four possible propositions constitute the power set 2^Ω , which is the basis for making judgments, where $2^\Omega = \{\emptyset, red, blue, (red, blue)\}$. We can employ the BPA $m(red) = 0.7$ and $m(red, blue) = 0.3$ in the framework of evidence theory to represent the ignorance in Figure 3-2. In this way, $m(red, blue)$ determines the uncertainty of whether the proposition is *red* or *blue*.

From the BPA, upper and lower bounds of a probability interval can be defined. This interval holds the specific probability of a set of interest, and is bounded by two non-additive continuous measures called belief function (*Bel*) and plausibility function (*Pl*):

Definition 3 *Belief function*

The belief $Bel(A)$ for a focal element A is defined as the sum of all the masses of subsets of the set, formally,

$$Bel(A) = \sum_{B \in 2^\Omega: B \subseteq A} m(B) \quad (3.4)$$

Definition 4 *Plausibility function*

The plausibility $Pl(A)$ is the sum of all the masses of the sets B that intersect the set of interest A , namely,

$$Pl_m(A) = \sum_{B \in 2^\Omega: B \cap A \neq \emptyset} m(B) \quad (3.5)$$

One of the advantages of the Dempster-Shafer theory of evidence is to combine two independent sets of BPAs in specific situations. Dempster's rule of combination is the appropriate fusion operator when different sources express their beliefs over the frame in terms of belief constraints. This rule acquires the commonly shared belief between multiple sources and ignores all the conflicting (non-shared) beliefs through a normalization factor. Given two BPAs m_1 and m_2 , the Dempster's combination rule denoted by $m = m_1 \oplus m_2$ can fuse them as follows,

Definition 5 *Dempster's combination rule*

$$m(\emptyset) = 0 \quad (3.6)$$

$$m_{12}(A) = m_1 \oplus m_2(A) = \frac{1}{1 - K} \sum_{B \cap C = A} m_1(B)m_2(C) \quad (3.7)$$

where $K = \sum_{B \cap C = \emptyset} m_1(B)m_2(C)$, A , B and C are elements in 2^Ω . The normalization factor as shown above, namely $1-K$, has the effect of completely ignoring conflict and attributing any mass associated with conflict to the empty set. The normalization constant K also shows the conflict degree between defined BPAs. If $K = 1$, the two pieces of evidence are completely conflicting, and the Dempster's combination rule is not applicable. Correspondingly, the evidence is non-conflicting while $K = 0$. This combination rule for evidence can generate counterintuitive results.

Definition 6 *Evidence Distance*

With the full application of evidence theory, how can one know if some evidence bodies are similar to others? Distance measures for bodies of evidence determine the conflict or dissimilarity between two BPAs, and several definitions of distance in evidence theory have been proposed, for example Zouhal and Denoeux in (Zouhal and Denoeux, 1998) introduced a distance based on the mean square error between pignistic probabilities. Jousselme's distance (Jousselme, Grenier, and Bossé, 2001), Wen's cosine similarity (Wen, Wang, and Xu, 2008), and Smets' transferable belief model (TBM) global distance measure (Ristic and Smets, 2006). Jousselme's distance (Jousselme, Grenier, and Bossé, 2001) is one of the rational and popular definitions of distance, which is identified based on Cuzzolin's geometric interpretation of evidence theory (Cuzzolin, 2008). The power set of the frame of discernment space, a distance, and vectors are defined with the BPA as a particular case of vectors (Jousselme, Grenier, and Bossé, 2001). With m_i , m_j being two BPAs under the frame of discernment Ω , Jousselme's distance $d(m_i, m_j)$ that estimates the distance between m_i and m_j is defined as

$$d(m_i, m_j) = \sqrt{\frac{1}{2}(\vec{m}_i - \vec{m}_j)^T D (\vec{m}_i - \vec{m}_j)} \quad (3.8)$$

where $(\vec{m}_i - \vec{m}_j)$ is a vector determined by the two BPAs, D is a matrix and the element in D is defined as $D(A, B) = |A \cap B| / |A \cup B|$, $A, B \in 2^\Omega$, and $|A|$ represents cardinality. An example of suspect identification is given in Example 1, let us take the BPAs m_1 and m_2 as an example to determine the dissimilarity between m_1 and m_2 in more details. We have $m_1(Paul) = 0.6$, $m_1(John) = 0.3$, $m_1(Paul, John) = 0.1$ in Example 1, thus $\vec{m}_1 = (m_1(Paul), m_1(John), m_1(Paul, John)) = (0.6, 0.3, 0.1)$. As a result, $\vec{m}_1 - \vec{m}_2 = (m_1(Paul) - m_2(Paul), m_1(John) - m_2(John), m_1(Paul, John) - m_2(Paul, John))$. The vector $\vec{m}_1 - \vec{m}_2$ shows their difference. Correspondingly, the detailed element in the matrix D can be determined by $D(A, B)$ as well, where A and B could be each of the elements in the power set. For example, since $Paul \cap (Paul, John) =$

$Paul$ and $Paul \cup (Paul, John) = (Paul, John)$, thus $D(Paul, (Paul, John)) = \frac{|(Paul)|}{|(Paul, John)|} = \frac{1}{2}$. Jousselme's distance is an efficient tool to measure the dissimilarity of two BPAs.

Definition 7 Entropy of Dempster-Shafter theory

Entropy, which was first introduced by the German physicist Rudolf Clausius in 1850 (Harmanec and Klir, 1994; Yao and Ke, 2014), is the measure of a system's thermal energy per unit temperature that is unavailable for doing useful work. Because work is obtained from ordered molecular motion, the amount of entropy is also a measure of the molecular disorder, or randomness, of a system. In 1948, the concept of information entropy was introduced by Claude Shannon in his published paper "A Mathematical Theory of Communication" (Shannon, 1948). The information entropy, often just entropy, is a primary quantity in information theory associated with any random variable, which can be interpreted as the average level of "information," "surprise," or "uncertainty" inherent in the variable's possible outcomes.

Given a random variable X , with possible outcomes x_i , each with probability $P_X(x_i)$, the entropy $h(X)$ of X is defined as follows:

$$h(X) = - \sum_i p_X(x_i) \log(p_X(x_i)) \quad (3.9)$$

Evidence theory is a generation of the Bayesian theory of probability. However, the uncertainty estimation of BPAs of the Dempster-Shafter theory of evidence still needs trial and error. Since the first entropy definition of a BPA was given by Höhle in 1982 (Höhle, 1982), many definitions of entropy of Dempster-Shafter theory have raised, for instance, the definitions proposed by Smets (Smets, 1983), Yager (Yager, 1983), Nguyen (Nguyen, 1987), Dubois and Prade (Dubois and Prade, 1987), Deng (Deng, 2016) and so on. Given a BPA m_i , Deng entropy (Deng, 2016; Fei, Deng, and Mahadevan, 2015) is defined as follows,

$$E_d(m_i) = - \sum_{F \subseteq 2^\Omega} m_i(F) \log_2 \left(\frac{m_i(F)}{2^{|F|} - 1} \right). \quad (3.10)$$

where F is the focal element in the BPA and $|F|$ is the cardinality of F . However, these definitions are still imperfect. For example, Deng entropy has some limitations when the propositions are of the intersection. Thus, several methodologies have been proposed for Deng entropy improvement (Özkan, 2018; Cui et al., 2019). Recently, Jiroušek and P. Shenoy proposed a new definition, five of six listed properties of entropy are satisfied (Jiroušek and Shenoy, 2018). Though if there exists a definition that satisfies the six properties is still an open issue, the proposed entropy in (Jiroušek and Shenoy, 2018) is acceptable and could be regarded as near-perfect.

As a result, we would use the definition of entropy of Dempster-Shafer theory in literature (Jiroušek and Shenoy, 2018) for uncertainty estimation in this chapter. For the frame of discernment defined in Ω where $x \in \Omega$ and F is a focal element, the entropy of the BPA m is defined as follows,

$$H(m) = H_s(Pl_P_m) + H_d(m) = \sum_{x \in \Omega} Pl_P_m(x) \log\left(\frac{1}{Pl_P_m(x)}\right) + \sum_{F \in 2^\Omega} m(F) \log(|F|) \quad (3.11)$$

Where the first part is employed to measure the conflict between elements in the frame of discernment (Jiroušek and Shenoy, 2018; Shannon, 1948), and $Pl_P_m(x)$ is defined as the plausibility (Definition 4) transform (Cobb and Shenoy, 2006; Jiroušek and Shenoy, 2018), where,

$$Pl_P_m(x) = \frac{Pl_m(x)}{\sum_{x \in \Omega} (Pl_m(x))} \quad (3.12)$$

The second part in equation 3.11 can be regarded as non-specificity of the BPA m (Dubois and Prade, 1987). That is to say, the uncertainty of the Dempster-Shafer theory of evidence can be captured from two aspects, namely the conflicts between exclusive elements in set Ω and the non-specificity uncertainty. Both components are on the scale $[0, \log(|\Omega|)]$, thus, the maximum entropy of a BPA is $2\log(|\Omega|)$ (Jiroušek and Shenoy, 2018).

Example 1 An example of Dempster-Shafer theory of evidence is given as follows. Supposing a murder happened yesterday night and two suspects are Paul and John. Then the frame of discernment is presented as $\Omega = \{Paul, John\}$ and three witnesses show their judgements. Their opinions are shown as

- 1.) $m_1(Paul) = 0.6, m_1(John) = 0.3, m_1(Paul, John) = 0.1$
- 2.) $m_2(Paul) = 0.4, m_2(John) = 0.4, m_2(Paul, John) = 0.2$
- 3.) $m_3(Paul) = 0.1, m_3(John) = 0.4, m_3(Paul, John) = 0.5$

In terms of evidence distance, we have the following calculation process,

1. $\vec{m}_1 = \{Paul, John, (Paul, John)\} = (0.6, 0.3, 0.1)$. Similarly, $\vec{m}_2 = (0.4, 0.4, 0.2)$, $\vec{m}_3 = (0.1, 0.4, 0.5)$;

$$2. D = \begin{matrix} & & Paul & John & (Paul, John) \\ Paul & & 1 & 0 & 0.5 \\ John & & 0 & 1 & 0.5 \\ (Paul, John) & & 0.5 & 0.5 & 1 \end{matrix}$$

3. $(\vec{m}_1 - \vec{m}_2) = (0.2, -0.1, -0.1)$;
4. $d(m_1, m_2) = \sqrt{\frac{1}{2}(\vec{m}_1 - \vec{m}_2)^T D(\vec{m}_1 - \vec{m}_2)} = 0.158$. In a similar way, $d(m_1, m_3) = 0.361$ and $d(m_2, m_3) = 0.212$.

The three witnesses are all not one hundred percent sure who the murder is. Nevertheless, we may study their degrees of uncertainty respectively. Let's take the first witness as an example,

1. $Pl_{P_m}(Paul) = \frac{Pl_m(Paul)}{\sum_{x \in \Omega}(Pl_m(x))} = \frac{0.6+0.1}{0.6+0.1+0.3+0.1} = \frac{7}{11}$. In the same manner, $Pl_{P_m}(John) = \frac{4}{11}$;
2. Thus, $H(m_1) = \frac{7}{11} \log_2(\frac{7}{11}) + \frac{4}{11} \log_2(\frac{4}{11}) + 0.6 * \log_2(|Paul|) + 0.3 * \log_2(|John|) + 0.1 * \log_2(|(Paul, John)|) = 1.046$. In the same way, $H(m_2) = 1.2$, and $H(m_3) = 1.47$. As we can see, the testimony given by the third witness is the most vague and uncertain.

3.3 Proposed trust model in a distributed multiagent system

In this section, we propose a trust model for trust estimation in multiagent systems, where agents' behaviors might vary over time. The proposed method not only rely on direct trust learned from direct experience but also indirect reputations by integrating all direct experience provided by third-party agents. We study the confidence of witnesses and combine both the direct trust and the indirect reputation. In what follows, a detailed explanation of the proposed trust model is presented.

Firstly, direct trust statement based on Dempster-Shafer theory is illustrated to manage interactive experiences, mainly stressing the fact that recent interactions are more important by assigning higher weight. In this way, the proposed method is applicable for dynamic trust estimation. What is more, the uncertainty that stems from randomness, incomplete information, and inefficient interactions are stressed. We then studied the confidence of individual witnesses to estimate indirect reputation. Afterward, direct trust and indirect reputation are combined, and we could reach the trustworthiness of each resource provider. The agent with the highest trust value is selected to raise interactions. After interaction, the credibility of witnesses is updated by not only the interactive feedback but also the average evaluation. Finally, credits are reassigned to witnesses, with which they are able to purchase information for trustee evaluations. In this way, information providers are motivated to present accurate information. For the sake of simplicity, the detailed processes are shown in Figure 3-3.

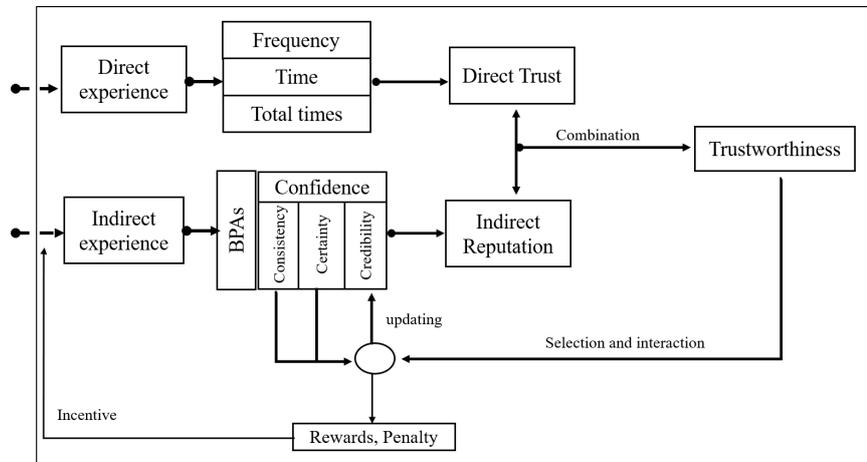


FIGURE 3-3: Trust estimation processes: from experience to trust value

3.3.1 Model direct trust

Direct trust is a reliable evaluation of the elected agent based on direct experience, which was given after interactions. The furnished interactive feedback could be binary, i.e., success or failure, or any real number in the range $[0,1]$, indicating that the service received is satisfactory. The real number could also represent a percentage reflecting the trustworthiness of the information or service provider. For the sake of clarity, the main actors are defined as follows,

1. We name the service client c_i who requires services or information or resources as client agent;
2. We name the service provider s_j who offers services or information or resources as provider agent;
3. We name the third-party agent w_k who presents testimony to the provider for the trust estimation as witness agent;

As is shown in Figure. 3-4, the client agent (evaluator) c_i colored in light gray would like to estimate the trustworthiness of s_j colored in dark gray according to its direct experience and the indirect reputation provided by witness w_k represented by sad faces.

After an interaction, service client agents update the corresponding satisfaction degree of the elected service provider agent, according to its previous trust value, and current feedback outcomes. Of the previous literature, they mostly average

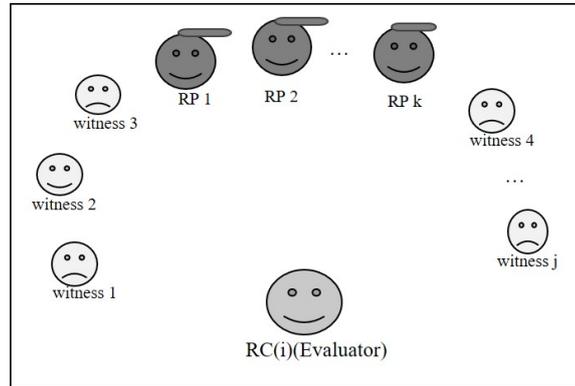


FIGURE 3-4: Trust evaluation in multiagent systems

or weight all the evaluations to obtain the ultimate satisfactory degree. From our perspective, two significant issues should be emphasized in direct trust estimation. First, recent interactions have great reference value compared to the previous ones, which is extendable for dynamic trust evaluation. Secondly, in a dynamic multiagent system, the inevitable uncertainty caused by randomness, incomplete information, insufficient interactions, environmental changes, etc, should be taken into account. To the best of our knowledge, of the current literature, at least in the context of Dempster-Shafer theory of evidence, lacks a complete interpretation of these two aspects. In (Yu and Singh, 2002b), they classify the evaluation into trust, uncertainty, and distrust by giving two parameters, and the medium evaluations are assigned as uncertainty. It could work, but we argue that in dynamic multiagent systems, recent interactions should significantly affect the final estimation. Later in the literature (Jøsang and Ismail, 2002; Teacy et al., 2006), assigning higher weights to recent interactions looks reasonable. Nevertheless, it lacks an in-depth analysis of the uncertainty caused by different factors. Therefore, in this chapter, we estimate direct trust considering a fading factor and analyzing the main causes of uncertainty from the perspective of interaction order, interaction volume, and interaction frequency.

Let T mean that the service client agent considers the service provider agent to be trustworthy. Accordingly, nT indicates that the supplier agent to be untrustworthy. Thus, the frame of discernment is represented by $\Omega = \{T, nT\}$. When c_i needs to appraise the performance of s_j , $S_{ij}(t)$ and $S_{cur_{ij}}(t)$ are employed to represent the satisfaction before the t th interaction and of the t th interaction respectively. Then the satisfaction is updated as follows,

$$S_{ij}(t+1) = cS_{ij}(t) + (1-c)S_{cur_{ij}}(t) \quad (3.13)$$

As mentioned above, $S_{cur_{ij}}(t)$ indicates the current interactive feedback which could be any number in the range of $[0, 1]$, namely,

$$S_{cur_{ij}}(t) = \begin{cases} 1, & c_i \text{ is completely satisfied with } s_j; \\ 0, & c_i \text{ is completely unsatisfied with } s_j; \\ v \in (0, 1); & \text{others.} \end{cases} \quad (3.14)$$

It is generally natural to average the current satisfaction or adopt the ratio of successful interaction for satisfaction representation. However, on the one hand, it takes much energy to remember all the interactions. From other perspectives, it is inappropriate to average the accumulated satisfaction, especially in a dynamic system. For example, two agents had 100 interactions, and the resource provider performed excellently in each of the interactions. However, from the 101 interaction on, the resource provider begins to perform poorly. The above approaches could hardly recognize its changed performance, which can lead to decision errors in the following interactions.

Nevertheless, in Equation 3.13, it is challenging to decide an appropriate value of c . In the literature (Jøsang and Ismail, 2002), a fading factor is adopted, which is proper for the beta reputation system. (Shehada et al., 2018) believe that weight is related to interaction amount and frequency. A. Das et al. hold that c should be dependent on the accumulated deviation (Das and Islam, 2011). From our point of view, we assume that c is mainly determined by its past value of trust and the current error $\Delta_{i,j}(t+1)$, which is obtained from estimated trust and actual performance. We trust its past trustworthiness if an agent acts consistently; otherwise, we rely more on the current interaction feedback if an agent performs different operations to hide its actual characteristic. For example, if the agent always has consistent performance, we have $c = 0.9$, so it is reasonable and intuitive to employ its current performance to estimate its expected trust in practical life. Thus, the parameter c is defined as follows,

$$c = \frac{e^{-1+\Delta_{i,j}(t+1)}}{e^{S_{ij}(t)} + e^{-1+\Delta_{i,j}(t+1)}} \quad (3.15)$$

here $\Delta_{i,j}(t+1) = |S_{cur_{ij}}(t) - S_{ij}(t)|$. In this way, the reliability of experience vanishing over time is well-stressed thanks to the parameter c .

With the method discussed above, the satisfaction and dissatisfaction of the service provider s_j from the perspective of the client c_i can be represented. From a certain point of view, it can be used as the trustworthiness of the provider directly for the reference of making decisions. However, as explained, much uncertainty is involved when generating the evidence due to incomplete information, interactive frequency, and randomness. Let us see the following examples,

- 1.) The agent Alice has interacted with Bob for only once. Thus, it is questionable to estimate the trustworthiness of Bob based on that interaction feedback;
- 2.) The agent Alice has plenty of interactions with Bob, but all of them have occurred for quite a long time ago.
- 3.) The agent Alice has plenty of interactions with Bob recently. Nevertheless, at the same time, Alice has raised more interactions with Charles.
- 4.) The system involves many deceptive agents who always present fake resources or information.

It is a challenge to distinguish a provider to be trusted or distrusted in the circumstances described. Therefore, the generated evidence includes not only trust $m(T)$, distrust $m(nT)$ but also the uncertainty expressed by $m(T, nT)$, that is, we do not know to assign the mass to T or nT . In order to express the uncertainty caused by these factors, the frequency factor $Fc_{ij}(t)$ is used. In this chapter, uncertainty are mainly caused by interaction amount r_{ij} , interaction frequency (r_{ij}/R_i) where R_i indicates the total interaction that c_i has. Property of the dynamic system can be denoted by averaging fluctuation, namely $\Delta_{i,j}(t+1)/r_{ij}$. Thus, $Fc_{ij}(t)$ is defined as

$$Fc_{ij}(t+1) = e^{-r_{ij}/R_i} * e^{-\Delta_{i,j}(t)/r_{ij}} \quad (3.16)$$

As a result, the generated BPAs for direct trust estimation after the r th interaction (before the $(r+1)$ th interaction) can be represented as follows,

$$\begin{cases} m_{ij}^{dir}(T) = S_{ij}(t+1) * Fc_{ij}(t+1) \\ m_{ij}^{dir}(nT) = (1 - S_{ij}(t+1)) * Fc_{ij}(t+1) \end{cases} \quad (3.17)$$

Then $m_{ij}^{dir}(T, nT) = 1 - (m_{ij}^{dir}(T) + m_{ij}^{dir}(nT))$, from this perspective, uncertainty caused by the lack of information and certainty, randomness are all distributed to $m_{ij}^{dir}(T, nT)$, the direct experience can also be rewritten as $m_{ij}^{dir} = (m_{ij}^{dir}(T), m_{ij}^{dir}(nT), m_{ij}^{dir}(T, nT))$. More in details, the assigned uncertainty changes with the value of frequency and averaging fluctuation are shown in Figure 3-5. As is shown in the figure, frequency and fluctuation are in the range of 0 and 1. As exhibited, all satisfactory are assigned to trust or distrust if the average fluctuation is 0 as the green line displays. That is to say, if agents act consistently, no randomness has accrued from the aspects of the system. Likewise, when the frequency is close to 0, which indicates there are only a few interactions between the two agents, then a large part of mass would be assigned to uncertainty (as the red line shows).

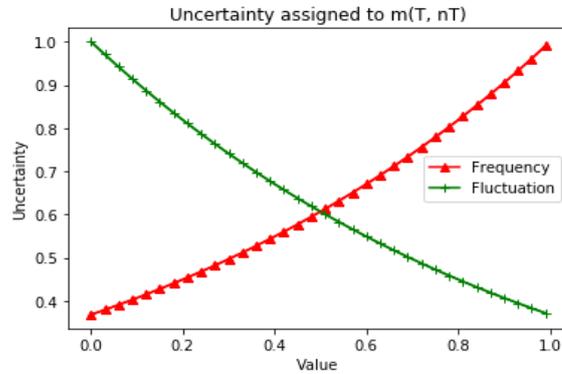


FIGURE 3-5: The assigned uncertainty changes with the value of frequency and averaging fluctuation

3.3.2 Method 1: Modelling indirect trust by the amount of information (MITAI)

The previous Dempster-Shafer theory-based trust models have both advantages and disadvantages (Yu and Singh, 2002b; Xu et al., 2007; Qiu et al., 2010; Feng et al., 2011) to manage indirect experience. For example, in (Yu and Singh, 2002b), Dempster-Shafer theory is used for distributed reputation management of electronic commerce. The method relies on either direct trust or indirect reputation, and an indirect reputation is not necessary if the direct trust is obtained. Meanwhile, Dempster's combination rule is directly applied to integrate multiple materials. Swift trust in a virtual temporary system based on Dempster-Shafer theory is recognized in the literature (Xu et al., 2007). The proposed method classifies the swift trust in three sub-objectives (the vulnerability, uncertainty, and venture) supported by five influencing factors, and Dempster's rule of combination is employed to fuse all influencing factors. In the literature (Qiu et al., 2010), trust transitivity is stressed by the Dempster-Shafer theory. They described the trust relationship considering uncertainty and established a trust transition model based on the trust features and trust relationship types. In (Feng et al., 2011), the authors utilized the Dempster-Shafer theory to address the network security problem in wireless sensor networks. The BPA is defined by some trust factors, such as the received packets rate, packets forwarding rate. The evidence similarity is processed as weights to modify evidence, and the modified parts are designated as unknown. In terms of limitations, the property that evidence reliability vanishing over time is not well stressed when generating BPAs. In addition, the standard Dempster's combination rule is not applicable directly when combining conflicting evidence. Recently, evidence distance

and entropy-based models have been used to combine conflicting evidence outside of the multiagent system community (Deng, Xiao, and Deng, 2017; Tang et al., 2017; Cui et al., 2019; Liu et al., 2019a). However, we realize that the assigned weight is directly proportional to the entropy of the evidence (the higher the entropy, the greater the weight) in data fusion models (Yuan et al., 2016; Tang et al., 2017; Cui et al., 2019). We argue that great weights are assigned to the high-entropy evidence is inappropriate. Additionally, to the best of our knowledge, to cope with the combination of conflict evidence with the amount of information conveyed by BPAs has rarely been stressed, especially in the field of trust estimation. Therefore, in this subsection, we improve the entropy-based weighted approach when combining evidence for indirect trust estimation in multiagent systems which is defined as MITAI.

Facing the two limitations, as explained in the last subsection, the evidence reliability vanishing over time is well stressed by the factor c in Equation 3.15. However, the combination of conflicting evidence would result in inconsistent and irrational results. The inconsistent results can mislead the customer to choose a partner that is not trustworthy. In the literature (Deng, 2015), a method based on BPA preprocessing is proposed. Supposing that the client agent c_i has received L BPAs $m_{qj}^{t.ind}(q = 1, \dots, L)$, each of which can be regarded as an indirect reputation provided by the third-party customer c_q . the pretreatment of these BPAs (Deng, 2015) is as following

$$MAE(F) = \sum_{q=1}^L \omega_{qj} m_{qj}^{t.ind}(F) \quad (3.18)$$

where ω_{qj} is the weighted degree of $m_{qj}^{t.ind}$ and F is the element in the union of the focal elements in the L BPAs. $MAE(F)$ represents the weighted average BPA of all the primitive evidence. After preprocessing, the classic Dempster's combination rule is used $L - 1$ times to combine $MAE(F)$ to receive the final result (Deng, 2015). However, it is challenging to decide the appropriate weight of $m_{qj}^{t.ind}$. In the previous methods (Deng et al., 2004; Deng, 2015; Qian, Guo, and Deng, 2017; Liu et al., 2019a), especially outside multiagent systems, evidence distance and entropy-based approaches and average weight approaches are dominant for data fusion. However, we recognize that the assigned weight is directly proportional to the entropy value (Yuan et al., 2016; Tang et al., 2017; Cui et al., 2019), that is, the high weight is assigned to the evidence with high entropy, and it is not reasonable. Besides, the entropy of Dempster-Shafer theory of evidence is rarely used alone for conflict evidence combination.

As explained, the generation of BPAs involves much uncertainty. Generally speaking, uncertainty can also be interpreted as the amount of information conveyed

by the message, and it can be utilized to construct weights (Klir and Lewis, 2008; Han et al., 2011). Compared to the news "the sun rises from the east," it is accepted that "the sun rises from the west" carries more knowledge. Namely, when the data source produces a low-probability value, the event conducted more "information" than when a high-probability event occurs. Deng entropy shows the information contents contained by the BPAs (3.10). Therefore, the higher the entropy value, the greater the uncertainty the evidence is, and accordingly, the less information the evidence contains. Thus, lesser weight should be assigned. Moreover, on one side, the frame of decrement consists of two elements, i.e., $\Omega = \{T, nT\}$; thus, the propositions have no intersection, and the Deng entropy values can be employed to capture uncertainty. On the other side, considering the same entropy could be used to illustrate that the high weights need to be assigned to low-entropy BPAs. As a result, Deng entropy is used in this part, and we suppose that L BPAs $m_{qj}^{t:ind}$ ($q = 1, \dots, L$) are received by the client agent c_i , then the detailed processes to fuse the L indirect reputation are shown as follows,

- (i) Compute Deng entropy of BPA $m_{qj}^{t:ind}$ by (3.10) and results are represented by $E_d(m_{qj}^{t:ind})$. It must be emphasized that because of the existence of uncertainty and fading functions, the evidence of 100% negative or positive does not exist, for instance, at time t , ($m_{qj}^{t:ind}(T) = 1, m_{qj}^{t:ind}(nT) = 0, m_{qj}^{t:ind}(T, nT) = 0$) does not exist;
- (ii) Normalize the Deng entropy $E_d(m_{qj}^{t:ind})$, in order to observe obvious difference between the Deng entropy $E_d(m_{qj}^{t:ind})$, we propose to use the equation below and results are represented by $\hat{E}_d(m_{qj}^{t:ind})$

$$\hat{E}_d(m_{qj}^{t:ind}) = \frac{\max_{k=1}^L e^{E_d(m_{kj}^{t:ind})} - e^{E_d(m_{qj}^{t:ind})}}{\sum_{p=1}^L (\max_{k=1}^L e^{E_d(m_{kj}^{t:ind})} - e^{E_d(m_{pj}^{t:ind})})} \quad (3.19)$$

- (iii) Afterward, each BPA is weighted with its normalized entropy, we therefore have $\omega_{qj} = \hat{E}_d(m_{qj}^{t:ind})$ in (3.18). Thus, we modified the entropy-based weighted approach to combine multiple BPAs, that is, a great weight is appointed to the low-entropy BPA. From another perspective, agents tend to trust the third-party agents that provide more information in the trust estimation model, or third-party agents with higher certain indirect reputations are more trustworthy. If all the entropies are the same, the weights are set equally.
- (iv) Finally, combine the modified BPA $L - 1$ times as (Deng, 2015) stressed to obtain the final result.

In this way, the received L BPAs are combined and the combined indirect reputation is represented by $(M_{ij}^{t:ind}(T), M_{ij}^{t:ind}(nT), M_{ij}^{t:ind}(T, nT))$. An example is given as follows to illustrate its processes.

Example 2 Assume $\Omega = \{A, B\}$, three BPAs are given below,

- $m_1(A) = 0.9, m_1(B) = 0.1, m_1(A, B) = 0$
- $m_2(A) = 0, m_2(B) = 0.8, m_2(A, B) = 0.2$
- $m_3(A) = 0.4, m_3(B) = 0.4, m_3(A, B) = 0.2$

We use the normalized uncertainty degree of BPAs to represent the weight degree, the fusion of m_1, m_2 and m_3 can be computed below.

Step 1: The Deng entropy values are obtained by Equation. 3.10, where $E_d(m_1) = 0.469, E_d(m_2) = 1.039$ and $E_d(m_3) = 1.839$;

Step 2: Normalize the Deng entropy by Equation. 3.19, and $nE_d(m_1) = 0.575, nE_d(m_2) = 0.424$ and $nE_d(m_3) = 0$;

Step 3: Modify the BPAs by Equation. 3.18 with weights equal to the normalized Deng entropy, and $MAE_{123}(A) = 0.518, MAE_{123}(B) = 0.397$ and $MAE_{123}(A, B) = 0.085$;

Step 4: Combine the modified BPA $MAE(m)$ two (3-1) times, the final results are $m_{123}(A) = 0.660, m_{123}(B) = 0.338$ and $m_{123}(A, B) = 0.002$;

3.3.3 Method 2: An overall indirect reputation model

In section 3.3.1, we discussed how to generate direct trust according to historical interactive feedback, and in section 3.3.2, we discussed improving Dempster's combination rule by and only by the information amount conveyed by the BPAs, and the main contribution can be concluded from two aspects; from one hand, we tested that the Dempster's combination rule can be revised by entropy from the perspective of information amount. That is to say, the evidence that contains a different amount of information should contribute differently. On the other hand, we have revised that the high weight should be assigned to low-entropy BPAs. However, we have stressed the inevitable uncertainty factor that originated by interactive frequency, randomness, or incomplete information during trust estimation. In this subsection, we would focus on indirect reputation management. Some literature employs direct trust only if it has; otherwise, it would demand indirect reputations. From our perspectives, we have to make full use of both direct trust and indirect reputation for trust estimation, especially in dynamic and distributed multiagent systems. However, it is challenging to rely on the received information directly since some agents

might be dishonest, self-interested, and selfish. Thus, we first need to have confidence in each of the collected information, and we would estimate indirect reputation from the following aspects,

1. The trust estimation presented by the witness: This aspect is the same as capturing direct trust. The framework of the Dempster-Shafer theory of evidence is adopted to estimate its direct trust from the perspective of information provider agents (witness).
2. Consistency. As explained, both witnesses and service providers might present misleading or fake information, and an agent has to figure out the high-quality information to get rid of serious consequences. For instance, the client c_i trust the witness w_k . However, w_k does not have sufficient testimony about the service provider s_j or w_k desires to provide false information because of the conflict of interest. In this case, directly applying the information received may lead to negative results. The average dissimilarity between received information is employed to identify the distinguishes between evidence, and on the contrary, the similarities between direct trust are adopted to decide the quality of the provided information.
3. The credibility of witnesses makes wonderful sense since agents tend to trust reliable friends. A witness's credibility is represented by a real number in the range of $[0, 1]$, and a value of one indicates that the witness is entirely reliable. Otherwise, the credibility of the witness is low, and trust should be avoided. We initialize the value to a fixed number and gradually update it based on interactive feedback.
4. The certainty of a witness is also meaningful in interaction-based trust estimation amongst multiagent systems. For instance, the agent Alice completely trusts the agent Bob. However, it is questionable to rely on the agent Bob if Bob is uncertain to the presented evidence about the agent Charles. In this chapter, we believe that certainty is dependent on evidence, and we can refine certainty from the received evidence.

Therefore, these four aspects are used to estimate indirect reputation, and evaluators are more confident in the high-quality and high-certainty evidence provided by highly reliable witnesses. We use $(m_{kj}^{dir}(T), m_{kj}^{dir}(nT), m_{kj}^{dir}(T, nT))$ to express the direct trust of the agent s_j from the perspective of the witness w_k . As is asserted in the previous part, agents should have a rational confidence on the received information. In our opinions, evidence availability changes accompanied by confidence, and

the low-confidence evidence will be greatly reduced in the final decision-making process, and the high-confidence evidence plays a more important role for the final decision. We assume that the confidence of the received information is represented by $\mathbf{Conf}(c_i, w_k, s_j)$, then the modification of the evidence by the degree of confidence can be determined as follows:

$$\begin{aligned} m_{ij}^{ind}(T) &= m_{kj}^{dir}(T) * \mathbf{Conf}(c_i, w_k, s_j); \\ m_{ij}^{ind}(nT) &= m_{kj}^{dir}(nT) * \mathbf{Conf}(c_i, w_k, s_j); \end{aligned} \quad (3.20)$$

then $m_{ij}^{ind}(T, nT) = 1 - m_{ij}^{ind}(T) - m_{ij}^{ind}(nT)$. Here $m_{ij}^{ind}(T)$ shows the indirect reputation evaluation of the agent s_j from the perspective of the agent c_i according to the information provided by the witness w_k . As defined, the discounted parts are interpreted as uncertainty. It is logical as if the provided information is not convincing, and more information is assigned as uncertainty. Thus, more information is required to induce uncertainty. After the modification, the Dempster's combination rule is used to merge all the evidence.

However, it is not easy to decide an appropriate value of confidence, especially in a dynamic environment. (Yu and Singh, 2003) assign weights to witnesses that represent its credibility. It could work in a stable system, but it provides some opportunity for joint deception. That is to say, a group of agents jointly deceived. Certainty is also adopted to present confidence (Basheer et al., 2015). However, it is not enough to consider these aspects separately to decide confidence. As explained, using information consistency to characterize confidence will allow group deception to take advantage. Employing agents' credibility to describe confidence may lead to wrong decisions due to the misinformation from trusted agents. At the same time, if there is high uncertainty towards high-quality information provided from reliable alternatives, it is also impossible to make the right decision. Also, we have rarely found work that integrates consistency, credibility, and certainty to determine confidence in our reviewed works. As a result, to provide a near-perfect trust estimation model, consistency, credibility, and certainty are studied in this chapter.

Evidence consistency

Distance suggests how far two objects are away from each other in practical life. The distance of BPAs shows the dissimilarity between two BPAs. The shorter distance means that the two BPAs are more likely to support the same target (service provider). For trust estimation models in multiagent systems, we adopt average distance to capture the consistency of some evidence. Assume that the client received BPAs from L different witnesses w_k , each of which is represented by m_{kj}^{dir} . As a result,

the consistency (dissimilarity) between these pieces of evidence are explored below. Firstly, the average distance of m_{kj}^{dir} is represented by $Dis(m_{kj}^{dir})$ which is defined as

$$Dis(m_{kj}^{dir}) = \frac{\sum_{p=1, p \neq k}^L d(m_{kj}^{dir}, m_{pj}^{dir})}{L - 1} \quad (3.21)$$

where $d(m_{kj}^{dir}, m_{pj}^{dir})$ indicates the distance of m_{kj}^{dir} and m_{pj}^{dir} , and $Dis(m_{kj}^{dir})$ represents the average distance to the evidence provided by the witness w_k , which also expresses its conflict or difference in the group of L evidence. Obviously, the distance between two evidence falls in the range $[0, 1]$ in this chapter. Meanwhile, we claim that the opposite of difference is similarity. Consequently, we can receive the similarity by the following operations,

$$Sim(m_{kj}^{dir}) = 1 - Dis(m_{kj}^{dir}) \quad (3.22)$$

$Sim(m_{kj}^{dir})$ represents the consistency towards the service provider s_j from the perspective of the witness w_k in the group of L witnesses. We use consistency to distinguish bad-quality information. Let us see an example, imagine that agents Alice, Bob and Charles all recommend Emma while Davis is extreme against Emma. Therefore, Alice, Bob, Charles reach a high degree of average similarity. Thus, the opinion provided by Alice, Bob and Charles is regarded as high-quality evidence, and they tend to be trusted. However, in order to avoid joint deception by a group of agents in this case, we define another individual credibility to characterize the degree of confidence.

Credibility of individual witness

In this part, the credibility of an individual witness in multiagent systems is analyzed in detail. For a group of witnesses $w_1, w_2, \dots, w_k \dots w_L$, they provide testimony to the resource client c_i if c_i urges to evaluate the trustworthiness of the resource provider s_j . From the perspective of c_i , to rely on the witness or not is a thought-provoking question. Thus, agents' credibility is adopted to indicate how much the witness is reliable. In this chapter, we force the witness credibility to be in the range $[0, 1]$, and the value is initialized to 0.5 if no cooperations have ever occurred. With interactions go by, the credibility is updated according to the interactive feedback.

It is necessary to explain how credibility is updated. Of the previous Dempster-Shafer theory based trust estimations models, the value of credibility decreases all the time over interactions if the witness is found lying in (Yu, Singh, and Sycara, 2004). This model is under the assumption that all agents are acting in a constant

way. Besides, never forgiving an agent or providing it a second chance also seems irrational. In the literature (Yu, Kallurkar, and Flo, 2008), credibility is learned mainly from two aspects, namely, the positive feedback and negative feedback. That is to say, rewards are expressed to all agents when positive feedback is reached while punishing all agents when negative feedback is collected. However, some witnesses may have opposing attitudes when positive feedback is raised, and so it is the same when negative feedback is received. Thus, the credibility of these witnesses needs to be updated separately. For example, Alice, Bob and Charles present their trust evaluation about Davis, where Alice and Bob trust Davis, while Charles does not. So Davis was chosen to initiate an interaction, and it was later proved to be a positive result. At this time, when updating personal credibility, we cannot enhance the credibility of all agents, at least the value of Charles should be decreased because its information provided may cause decision-mistakes.

What is more, it is sometimes questionable to modernize the value only according to the current interactive feedback, especially in the dynamic systems. Anyhow, individual reliability has a great influence on indirect reputation. We consider the credibility $Cred(c_i, w_k)$ of the witness w_k from the perspective of c_i when generating indirect reputation. More details about learning from the experience to renew personal credibility are discussed in the following sections.

Model certainty

The witness's certainty also counts for trust estimation in multiagent systems. To effectively achieve uncertainty, the definition has to capture the following vital intuitions (Wang and Singh, 2007; Wang and Singh, 2010).

1. Effect of evidence. Certainty raises as evidence increases (for a fixed ratio of positive and negative observations).
2. Effect of conflict. Certainty declines as the extent of conflict increases in the evidence.

It is stressed that uncertainty caused by a lack of certainty and complete information is imperative when generating BPAs. Thus, certainty is independent of evidence(BPAs). From our perspective, certainty comes from two aspects. Firstly, certainty decreases as the extent of conflict increases in the evidence (We manage certainty by the ratio of positive and negative observations, namely, trust and distrust). Therefore, certainty decreases as the ratio of positive and negative interactions increases. Secondly, certainty decreases as unknown information $m(T, nT)$ increases. As defined in the direct trust part (Equations 3.16 and 3.17), unknown information

is determined by the interactive frequency and average difference. That is to say, more interactions would result in less unknown information. Thus, certainty raises as evidence increases for a fixed ratio of positive and negative observations. Meantime, we force the value of certainty to fall at the range of $[0, 1]$. Due to the listed reasons, entropy is an idle tool for uncertainty measurement.

Many definitions of entropy have been proposed for the framework of the Dempster-Shafter theory of evidence. For instance, Deng entropy (Deng, 2016), Dubois and Prade's definition (Dubois and Prade, 1987). In this part, we adopt the entropy presented by Radim and Prakash (Jiroušek and Shenoy, 2018). The main reasons can firstly be that five of the six listed properties are satisfied, which could be regarded as near-perfect in some perspective. Moreover, the entropy given by Radim and Prakash is designed from two components. The first component is designed to measure conflicts in the BPAs, and the second is used to measure non-specificity in BPAs, which fits for the main reasons of uncertainty in this chapter. As explained, we believe that uncertainty yield from 1:) the conflict of positive and negative interaction, and 2:) a lack of information could result in uncertainty. In this chapter, we have $\Omega = \{T, nT\}$, and $m(T) = x$, $m(nT) = y$ and $m(T, nT) = 1 - x - y$ which is under the conditions of $0 \leq x \leq 1$, $0 \leq y \leq 1$, and $0 \leq 1 - x - y \leq 1$. Then the relation between the entropy (Jiroušek and Shenoy, 2018) is presented in Figure 3-6. As is shown, uncertainty is at the range of $[0, 2 * \log_2|\Omega|]$. Thus, we define the certainty of the information m_{kj}^{dir} as follows,

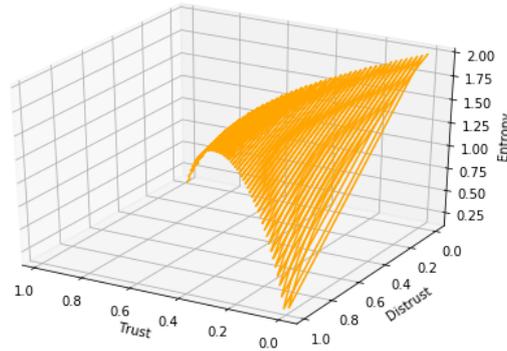


FIGURE 3-6: The entropy when the frame of discernment is $\Omega = \{T, nT\}$ and it is in the range of $[0, 2]$

$$Cer(m_{kj}^{dir}) = \frac{2 - H(m_{kj}^{dir})}{2} \quad (3.23)$$

According to Figure. 3-6, for example, a witness is one hundred percent confident in its provided evidence if the information is represented by the BPA of $(1, 0, 0)$ or $(0, 1, 0)$. However, if the evidence is $(0, 0, 1)$, the witness has one hundred percent uncertainty in his judgment.

Model indirect reputation

In this subsection, we would explain how the ultimate indirect reputation is calculated. As explained, indirect reputation comes from two aspects, namely the quality evaluation, which is based on the resource provider agent's previous performance, and the confidence of the witness. The value of confidence is influenced by three sub-factors, i.e., the consistency of the provided information (equation 3.21), the credibility of agents (subsection 3.3.3) and the certainty (equation 3.23).

The three sub-factors are used to revise the generated evidence, namely, to estimate $\mathbf{Conf}(c_i, w_k, s_j)$ in equation.3.20. Here, we use $\mathbf{Conf}(c_i, w_k, s_j)_t$, $Sim(m_{kj}^{dir})_t$, $Cred(c_i, w_k)_t$ and $Cer(m_{kj}^{dir})_t$ to represent the value of confidence, the consistency, the credibility and the certainty for pretreatment of the t th interaction respectively. Detailed modification is defined as follows,

$$\mathbf{Conf}(c_i, w_k, s_j)_t = Sim(m_{kj}^{dir})_t * Cred(c_i, w_k)_t * [(Cer(m_{kj}^{dir})_t)^{|\Delta|}] \quad (3.24)$$

Where $\Delta = \min\{0, Sim(m_{kj}^{dir})_t * Cred(c_i, w_k)_t - \frac{1}{L} \sum (Sim(m_{kj}^{dir})_t * Cred(c_i, w_k)_t)\}$ is the smaller value between 0 and the difference of confidence obtained from consistency and credibility. More in detail, the confidence of a piece of evidence provided by the witness is directly proportional to the credibility of a witness, quality, and certainty of the evidence. In terms of certainty, as Δ defines, the confidence $\mathbf{Conf}(c_i, w_k, s_j)_t$ of the evidence provided by the witness w_k decreases with the certainty $Cer(m_{kj}^{dir})_t$ if its credibility $Cred(c_i, w_k)_t$ and quality $Sim(m_{kj}^{dir})_t$ are below the average level, which is calculated by $Sim(m_{kj}^{dir})_t * Cred(c_i, w_k)_t - \frac{1}{L} \sum (Sim(m_{kj}^{dir})_t * Cred(c_i, w_k)_t)$. Otherwise, the confidence is determined by only the credibility and quality. That is to say, the resource customer agent is more confident in the good-quality indirect experience provided by credibility agents as presented in Figure 3-7. Those experiences that originate from unreliable agents require to be discounted and no longer reliable. After modification, agents' indirect reputation are fused by Dempster's combination rule shown in the equation. 3.6.

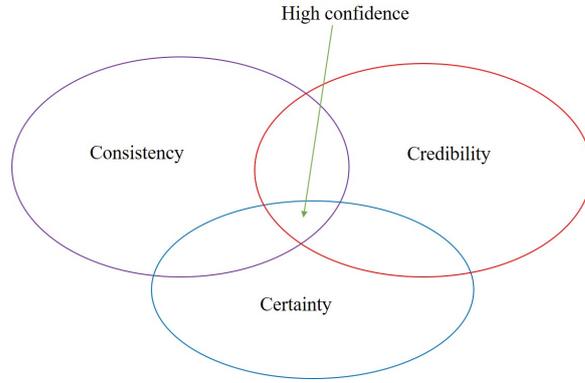


FIGURE 3-7: Intergrated confidence

Model overall Trust

In this section, we would explain how to model the overall value of trust. The overall evaluation of the service provider s_j from the perspective of the service client c_i derives from two parts, namely direct trust, and indirect reputation. In general, a client trusts itself more compared to the information provided by the third-party agents. Anyhow, we are not self-conception, and an agent might better trust others if a system is unstable. Thus, the pretreatment of direct trust and indirect reputation proposed in (Deng, 2015) is adopted, equation 3.25 shows the detailed processes, and we use m_{ij}^{pre} to indicate the modified BPA. In the equation, ψ indicates the weight of direct trust, while $1 - \psi$ corresponds to the indirect reputation. ψ depends totally on the stability in the multiagent systems; a small ψ indicates agents' performances often change. Generally, $\psi > (1 - \psi)$, thus, we have

$$\begin{aligned}
 m_{ij}^{pre}(T) &= \psi m_{ij}^{dir}(T) + (1 - \psi) m_{ij}^{ind}(T) \\
 m_{ij}^{pre}(nT) &= \psi m_{ij}^{dir}(nT) + (1 - \psi) m_{ij}^{ind}(nT) \\
 m_{ij}^{pre}(T, nT) &= \psi m_{ij}^{dir}(T, nT) + (1 - \psi) m_{ij}^{ind}(T, nT)
 \end{aligned} \tag{3.25}$$

After the pretreatment, Dempster's combination rule is used to combine $(m_{ij}^{pre}(T), m_{ij}^{pre}(nT), m_{ij}^{pre}(T, nT))$ once as (Deng, 2015) defined, and we can receive the final evaluation about the service provider. Afterward, the resource client selects one of the most reliable service provider to raise interaction. The interactive feedback outcome is received after the interaction, and it could be either binary results (either success or failure) or an evaluation score in the range $[0, 1]$.

The general rule for the customer agent c_i to estimate the trust of all the potential resource suppliers could be translated into the Algorithm 1, which is also named

as Evidence theory-based Trust Estimation Model (ETE). Our proposed ETE algorithm has two inputs: \hat{S} which is the set of prospective supplier agents, and \hat{C} is the set of third-party customers who provide indirect reputation to a certain supplier s_j . Specifically, for each potential resource supplier $s_j \in \hat{S}$, the customer c_i receives recommendations from other customers (lines 4-6). Then indirect reputation is calculated with the proposed trust model (lines 7-11). Likewise, direct trust can be computed in the same way (line 12). Direct trust and indirect reputation are combined to obtain the ultimate trustworthy value of s_j (lines 13-14). Finally, c_i achieves the trust value of all suppliers. In the following section, we would describe how personal credibility is updated.

Algorithm 1 ETE algorithm

```

1: Input:  $\hat{S}, \hat{C}$ .
2: Output: Trustworthiness of each  $s_j$  ( $s_j \in \hat{S}$ ).
3: for  $s_j$  in  $\hat{S}$  do
4:   for  $c_q$  in  $\hat{C}$  do
5:      $c_i$  asks  $c_q$  for indirect reputation;
6:   Endfor
7:    $c_i$  receives customers' evaluations.
8:   Compute the evidence consistency by Equation. 3.21 and Equation. 3.22;
9:   Update the credibility of witness as section 3.3.3 explained;
10:  Compute the certainty of each evidence by Equation 3.23;
11:  Combine the received indirect reputation by Equation 3.24;
12:  Obtain the direct trust;
13:  Pretreat direct trust and indirect reputation by Equation 3.25;
14:  Combine the pretreatment and obtain the trust value of the resource provider
    agent from the perspective of resource customer agent.
15: Endfor
16: return the trustworthiness of  $s_j$ 

```

Update Credibility

We must first distinguish whether the witnesses w_k recommend or oppose the service provider s_j before updating the corresponding credibility. In short, an agent updates personal credibility according to the witness's contribution to the decision-making process. $(m_{kj}^{dir}(T), m_{kj}^{dir}(nT), m_{kj}^{dir}(T, nT))$ represents the evaluation of s_j provided by w_k , where $m_{kj}^{dir}(T)$ indicates that w_k trusts s_j , and $m_{kj}^{dir}(nT)$ represents that the resource provider to be untrustworthy. $m_{kj}^{dir}(T, nT)$ shows the uncertainty caused by delay, interactive frequency, incomplete information. It is essential to highlight

that uncertainty doesn't mean knowing nothing, it could be resigned to trust or distrust according to Equation 3.12 as follows,

$$\begin{aligned} Pl_P_m(T) &= \frac{Pl_m(T)}{Pl_m(T)+Pl_m(nT)} \\ Pl_P_m(nT) &= \frac{Pl_m(nT)}{Pl_m(T)+Pl_m(nT)} \end{aligned} \quad (3.26)$$

$Pl_P_m(T)$ greater than 0.5 symbolizes that the service provider is reliable from the witness's perspective. Otherwise, the witness w_k concludes that the client can not rely on the service provider s_j . Therefore, the credibility of an agent needs to be updated by the difference between the information provided and the feedback outcome. The interaction outcome is represented by a success $(1, 0, 0)$ or failure $(0, 1, 0)$ in binary evaluation systems, otherwise $(s, 1 - s, 0)$. As has been emphasized, agents require resources in multiagent systems where agents might act unstably. That is to say, the obtained incorrect information can probably not because of the witness's unreliability. For example, Alice trusts Bob, and Bob also provided a trusted evaluation about Charles, but at this time, Charles provided fake resources, then Alice firstly evaluates the quality of information by evidence similarity (Equation. 3.21 and Equation. 3.22) before updating Bob's credibility. The client agent believes the witness is still reliable if its information is similar to other testimony provided by other agents. Then the credibility of w_k from the perspective of c_i is updated with the following four manners:

- If the interactive feedback is positive:

case 1: $Pl_P_m(T) \geq 0.5$, $Cred(c_i, w_k)_{t+1} = Cred(c_i, w_k)_t * (1 + v \frac{\bar{D}_{c_i, w_k}}{2})$;

case 2: $Pl_P_m(T) < 0.5$, $Cred(c_i, w_k)_{t+1} = Cred(c_i, w_k)_t * (1 - v \bar{D}_{c_i, w_k})$;

- If the interactive feedback is negative:

case 3: $Pl_P_m(T) \geq 0.5$, $Cred(c_i, w_k)_{t+1} = Cred(c_i, w_k)_t * (1 - v \bar{D}_{c_i, w_k})$;

case 4: $Pl_P_m(T) < 0.5$, $Cred(c_i, w_k)_{t+1} = Cred(c_i, w_k)_t * (1 + v \frac{\bar{D}_{c_i, w_k}}{2})$.

\bar{D}_{c_i, w_k} is the difference of feedbacks, which can be calculated from two aspects. One is from its evaluation and the actual feedback. For instance, the evaluation of service provider s_j given by witness w_k is represented by $(m_{kj}^{dir}(T), m_{kj}^{dir}(nT), m_{kj}^{dir}(T, nT))$, then the first aspect is represented by

$$\bar{D}_{c_i, w_k}^1 = d(m_{kj}^{dir}, bpa(outcome)) \quad (3.27)$$

where $d(m_{kj}^d, bpa(outcome))$ indicates the differences between the provided testimony and the feedback. It could also be interrupted as a consequence by relying

on the received information. The second aspect is the conflict of the evidence in the group of L evidences. From our perspective, it is necessary to emphasize these differences, especially in distributed multiagent systems with deceptive agents where agents may deliberately react to hide their actual attributes. Thus, the second aspect is familiar with the definition with Equation.3.21 which can be illustrated as follows,

$$\bar{D}_{c_i, w_k}^2 = 1 - \frac{\sum_{p=1, p \neq k}^L d(m_{kj}^{dir}, m_{pj}^{dir})}{L - 1}. \quad (3.28)$$

In summary, D is updated with the equation as follows, where ζ indicates the weight of the difference of its evaluation and the feedback. In general, $\zeta > 0.7$, if multiagents do not change their behaviors frequently.

$$\bar{D}_{c_i, w_k} = \zeta * \bar{D}_{c_i, w_k}^1 + (1 - \zeta) * \bar{D}_{c_i, w_k}^2. \quad (3.29)$$

Incentive

One of the significant challenges in multiagent systems involving deceptive agents is to motivate agents to bestow accurate information. In this chapter, we are inspired by the credit mechanism to achieve this operation. Each of the agents in multiagent systems has a fixed number of credits, no matter it acts as a resource client, a witness, or a resource provider, which is administered and displayed by a blackboard. However, credits only make sense when an agent acts as a resource client or a witness. A witness is only allowed to contribute its testimony unless the resource client has credits. Simultaneously, a witness might receive or lose credits according to the interactive feedback of the resource client. As a resource client, an agent could no longer solicit others for any testimonies once it runs out of credits since it acts as a witness. Of course, it has the opportunity to urge for information by presenting convincing information and accumulating credits.

We suppose that to gain or to lose credits is affected by the interactive feedback. The specific amount is determined by its original credit and the confidence value decided by credibility value and certainty. In order to explain the credit mechanism in detail, we apply CR_i to represent the credit of agent c_i . We force the value to be in the range $[0, 2]$, and it is initialed to 1. It is capable of demanding information unless CR_j is bigger than the threshold, for example, $CR_j > 0.2$. After the interaction, the credits of all witnesses are updated as follows,

1. If the interactive feedback matches the testimony, then

$$CR_i = CR_i[1 + Cred(c_i, w_k) * Cer(m_{kj}^{dir})]$$

2. If the interactive feedback does not match the testimony, then

$$CR_i = CR_i[1 - Cred(c_i, w_k) * Cer(m_{kj}^{dir})]$$

It is necessary to explain what "match" means here. Namely, the witness supports the resource provide if the feedback was a success and the other way around. We set two bounds as the witness is not allowed to accumulate or lose too many credits in a dynamic multiagent systems. Meanwhile, a high-credit agent loses credits rapidly if it keeps lying. Another advantage of our trust estimation model is that another "second chance" mechanism is not required. As soon as a dishonest agent reaches the minimum credit level, it understands that it can no longer demand information for trust estimation. Thus, to maximize its own social welfare, it has to provide honest information to gain credit.

3.4 Resource reservation and resource sharing

We study the trust of agents to ensure efficient sharing of resources In multiagent systems, to get rid of potential deceptive resource provider agents. An agent in the proposed method makes decisions locally and prioritizes its utility function. Therefore, a well-evaluated agent in MARA is more likely to be selected more than once. However, resource sharing can not be interrupted once they begin processing. Besides, an agent has no priority over other agents. Hence, we offer solutions guided by the concept of reservations to ensure a fair allocation of resources.

The customer agent attempts to reserve the required resource provided by the most reliable supplier agent s_j . c_i sends a message to inform its unique identity and reservation time. The supplier s_j acts generous and preserves the corresponding available resource. Once the exact resource is retained, no one has the right to assign it before the reservation period. In this way, the resource customer who first reserves the resource is prioritized to obtain the resource. "First come, first served" guarantees the fairness and normal operation of the proposed system.

3.5 Discussion

In this chapter, we have conducted a trust and reputation model for trustworthiness estimation in multiagent system. Afterward, the proposed trust model is applied in multiagent resource sharing for resource provider agents selection. The proposed trust method offered a near-perfect trust estimation model from the following aspects:

1. Trust estimation in multiagent systems always involves uncertainty yield from a lack of certainty, incomplete information, or the dynamic environment. Of the previous dempster-Shafter theory-based trust models, uncertainty comes from the neutral evaluation. From our perspective, it should be improved from comprehensive aspects. Simultaneously, the trust generation needs to be sensitive to recent interactive feedback, especially from dynamic trust estimation in multiagent systems.
2. It is necessary to capture the confidence of the provided information for indirect reputation estimation. Of the most works we have reviewed, they have taken into account credibility, certainty, and consistency in a separate trust model. In this chapter, we proposed two approaches to reach indirect reputation. First, we understand evidence uncertainty from the entropy perspective and evidence is combined according to the information amount conveyed by BPAs. Then, we believe that a perfect trust model is still required to take all aspects (credibility, certainty, and consistency) into consideration. As a result, we manage evidence confidence from these three aspects, thanks to the Dempster-Shafter theory. This theory is capable of quantifying the similarity from other evidence. When qualifying certainty, we believe that certainty is independent of evidence and uncertainty mainly comes from the positive and negative interactions. Thus, employing entropy of evidence theory is suitable to manage certainty. What is more, the maximum entropy has also been approved, and it is rational to capture the value of certainty in the range $[0, 1]$.
3. In the trust estimation models in multiagent systems, one of the significant challenges is to motivate agents sharing honest information. We are motivated by the credit mechanism, and employ a blackboard to manage and display credits. Agents' credits are initialed to a fixed number while it changes over interactions. When the credit is lesser than the minimum level, it is not allowed to achieve information. In this way, witnesses have to share honest information to gain credits from the perspective of their utility.

To further evaluate the proposed model and compare it to the existing trust estimation models, we propose a comparison framework according to the following aspects:

1. Types of evaluation: Some models use direct experience to make the decision. While others rely on indirect reputation. In a dynamic system with deceptive agents, combining both evaluations adds flexibility to the system and provides a more generic and trusted evaluation technique.

2. The ability to capture uncertainty: Uncertainty is necessary for trust estimation caused by a lack of certainty, incomplete information, interactive frequency, and so forth. As a result, it is required to take into account the factor for trust estimation in multiagent systems.
3. Consistency: This factor, from one aspect, shows the quality of the provided information. From another aspect, we may distinguish the false information (if it is) caused by dishonesty or lack of cognition.
4. Credibility: Personal performance makes great sense in trust estimation, especially in dynamic multiagent systems. It reflects how much we may rely on a witness if it provides good-quality and certain information.
5. Certainty: It is logical to believe that A trusts B from the perspective of C mainly because of C's trustworthiness and certainty in its testimony. Meanwhile, we not only study uncertainty when generating evidence but also analyze uncertainty to obtain useful knowledge.
6. Incentive: To motivate agents to share reliable information is essential for multiagent systems with deceptive agents. Agents are willing to live in an environment where all agents are encouraged to be reliable.
7. Stability: Most of the reviewed multiagent systems assume that agents hold consistent performance constantly. Yet, some rational agents in practical lives react differently occasionally for the sake of personal interests or to protect their actual performance. Thus, considering that agents might perform differently is also an essential factor for trust estimation in multiagent systems.

Table 2-1 shows the comparison of the proposed model and some of the reviewed works in the state of art part.

3.6 Conclusion

This chapter is about studying and analyzing trust and reputation in multiagent systems for the fulfillment of filtering reliable agents for resource sharing. The proposed trust model is an interactive feedback based trust framework to secure multiagent systems in which agents continuously communicate. Our model has overcome the limitations or disadvantages of the Dempster-Shafer theory based trust model, and it has improved the trust model from the following aspects: Firstly, the proposed model has measured uncertainty caused by many factors, including inefficient interaction, incomplete information, randomness, and so forth. Direct trust derived

TABLE 3-1: Comparisons with trust models

	Uncertainty	Evidence	Consistency	Credibility	Certainty	Incentive	Stability
BRS (Jøsang and Ismail, 2002)	×	Indirect	×	×	×	×	Static
TRAVOS (Teacy et al., 2006)	×	Indirect or Direct	×	×	×	√	Static
FIRE (Huynh, Jennings, and Shadbolt, 2006)	×	Indirect & Direct	×	×	×	×	Static
Yu & Singh (Yu and Singh, 2002b)	√	Indirect	×	√	×	×	Static
ACT model (Yu et al., 2014)	√	Indirect & direct	×	×	×	×	Static
Zuo and Liu (Zuo and Liu, 2017)	×	Indirect	×	×	×	×	Static
ITEA (Elham, HosseinNikravan, and Zillesa, 2019)	×	Dynamic	√	√	×	√	Dynamic
Adaptive Trust (Shehada et al., 2018)	√	Indirect & Direct	×	√	×	×	Dynamic
Proposed Model	√	Indirect & Direct	√	√	√	√	Dynamic

from previous experience can be extended to dynamic trust estimation. Then, two approaches are proposed to reach indirect reputation. We first employ Deng entropy to study evidence certainty or information amount conveyed by BPAs, which would later be used to estimate the final indirect reputation. At the same time, we proposed a near-perfect trust model that used multiple information to capture indirect reputation. As explained, direct experience presented by the third-party agents must have the confidence to be useful. Therefore, we define the similarity between the received reputations to eliminate the agent's sudden performance change. We defined the credibility of agents to get rid of group deception. Besides, we have studied the uncertainty caused by various factors and analyzed the cause of uncertainty. What is more, we learned the certainty of the received evidence and made full use of the information presented. According to these three factors, we proposed that we should be more confident in the good-quality and high-certainty indirect experience provided by high-credibility agents. Thirdly, it is stressed that in a system of multiagent, agents not only desire to receive information for trust estimation but also desire to receive reliable information. Therefore, we propose an incentive mechanism that motivates agents to provide reliable information based on the credibility system. From another perspective, we improve the Dempster-Shafer theory-based trust model by modifying some limitations, mainly for uncertainty management and indirect reputation estimation.

When agents' trust is reached, we use a reservation approach to help agents obtain the required resource, which can improve the satisfaction towards the received resources.

Chapter 4

Random walk based resource selection in a distributed environment

With the increasing volumes of manufacturing services in a system, reliable manufacturing services selection for a target user has become one of the most significant challenges. In this chapter, a manufacturing service selection approach is proposed in a distributed environment where Simhash (Sadowski and Levin, 2007) is employed to obtain the similarity of cloud manufacturing services. Trust value is used to maintain the uncertainty and relationship between cloud manufacturing communities. This approach is mainly based on the fact that cloud manufacturing is more likely to be selected if recommended by trusted clients with similar requirements. This chapter is on the basis of the articles (Wang et al., 2020b).

4.1 Introduction: Problem statement

The cloud manufacturing paradigm is designed to realize the sharing of manufacturing resources distributed in different areas (Zhang et al., 2017). Distributed manufacturing resources and manufacturing capabilities are virtualized and encapsulated as cloud services obtainable by service clients in a pay-as-you-go mode (Ren et al., 2015; Liu et al., 2019b). Besides, manufacturing enterprises are encouraged to provide manufacturing services that lead to an increasing number of manufacturing service providers and their compositions (Guerrini and Yamanari, 2019). As a result, it is a tremendous challenge for resource clients to select and schedule manufacturing services to fit their requirements.

Generally speaking, there are two manners to select an ideal manufacturing service (Wu et al., 2013a). The first one is a centralized operating mode, which is the current mainstream of cloud manufacturing (He et al., 2019). In this operation mode,

cloud manufacturing consumers announce their requirements to the platform, and the cloud platform selects the most suitable cloud manufacturing services that meet the requirements. Another model is a distributed operation mode, which may be the future trends for manufacturing industries (Li, Barenji, and Huang, 2018). In this mode, manufacturing consumers select and pay the available services directly in need. That is to say, manufacturing resources providers and clients communicate and negotiate mutually to reach a resource-sharing agreement (He et al., 2019). The centralized operation mode has more powerful control over distributed resources. It is more conducive to handle complex requirements, while the distributed operation mode enables a more flexible and scalable cloud manufacturing, which can actively co-evolve with its environment (Škulj et al., 2017).

As described, the rapidly increasing numbers of manufacturing services lead to a complex manufacturing service ecosystem (Cardoso and Camarinha-Matos, 2013). In this kind of system, it is challenging to select the manufacturing services that target manufacturing consumer interests, and service recommendation techniques could be employed to facilitate the selection. In the case of the distributed operation mode, the manufacturing system, including both manufacturing clients and providers, can be represented as a set of autonomous and interactive entities, called agents, as is stated in section 2.1. Thus, cloud manufacturing agents have to communicate mutually to achieve this purpose. Then two significant issues are arisen, firstly, with whom the target agents should communicate. Secondly, how to pick the right manufacturing service in a heterogeneous information system involving various services?

Generally, similar resource clients are employed to make recommendations (Rong et al., 2014). A similar resource client discovery is usually based on the historical manufacturing service usage data stored by a central organization. Thus it is easy to determine similar clients in this kind of system. However, in the distributed operation mode, manufacturing service usage data are often monitored by geographically distributed sensors and stored in different cloud platforms (Qi et al., 2017). That is to say, historical data are distributed stored rather than centralized. Such a distributed environment needs data sharing to increase collaboration between different cloud manufacturing agents in the system. However, most cloud manufacturing agents are unwilling to contribute their data because this kind of data sharing can bring privacy leakage (Shen et al., 2017b). Simultaneously, cloud manufacturing consumers do not desire to announce their detailed requirements widely in the system. Furthermore, cloud manufacturing consumers' preference has seldom been considered. Due to these reasons, this chapter intends to consider linguistic preference, and the Simhash approach is employed to find clients with similar manufacturing

requirements. When searching for a manufacturing service, trust is regarded as the main factor for finding a client, and then the recommendation would be given by the trusted clients according to its local trust and similarity value. In what follows, some related works, including the recommender system, the PageRank algorithm and random walk algorithm are explained.

4.2 Related works

In this part, we present the related contents of this chapter. Firstly, the recommender system is discussed. What follows, the PageRank algorithm, especially the random walk approach, is introduced. We end this part by investigating the PageRank and random walk-based recommender systems.

4.2.1 Recommender system

In recent years, the Internet has played an indispensable role in everyone's daily life. It is a massive network that connects millions of computers globally. People communicate with any other people employing the Internet to search for service, information, resources, to buy and sell products, to complete different kinds of tasks. The Internet has, in essence, become a social network that links people, organizations, and knowledge (Wellman, 2001). Besides, Internet users have benefited a lot from the Internet to chase various daily entertainments that they usually do in the real world (Walter, Battiston, and Schweitzer, 2008).

This development also brings obsessions to the users as they would be faced with the problem of filtering one of the most suitable information that fits their requirements among the enormous database. The growing number of users accessing the Internet has also dramatically increased the amount of information and services related to the transactions. Thus, it is becoming difficult (even impossible) to manage systems by only human resources. As a result, intelligent information and service recommender systems are needed to satisfy the users by recommending desirable information autonomously. Recommender systems have been used to provide users with recommendations for products or services in various domains, such as e-commerce platforms and social networks. As stated in (Jamali and Ester, 2009), there is a set of users U and a set of services S in a service recommender system. The ratings expressed by users on services are given in a rating matrix $R = [R_{u,s}]_{m \times n}$. In this matrix, $R_{u,s}$ denotes the rating of the user u on the service s . The task of a recommender is as follows: Given a user $u_0 \in U$ and an item $i_0 \in I$ for which the rating

R_{u_0, i_0} is unknown, predict the rating for u_0 on item i_0 . Traditional recommendation approaches include (Lü et al., 2012):

1. Content-based filtering (CB): These approaches employ the content information to recommend items that are similar to those previously preferred by the target user.
2. Collaborative filtering (CF): These approaches recommend items to users based on the preferences that other users have expressed for those items.
3. Hybrid approaches: These approaches combine collaborative with content-based methods or with different variants of other collaborative methods.

In essence, the recommender systems are to rank the dissimilar items and resources according to its or the other previous experience. PageRank algorithm, which is famous for ranking webpages, has been used in recommender systems. What follows, more information concerning the PageRank algorithm is discussed.

4.2.2 PageRank algorithm

The PageRank algorithm accumulates the number and quality of links to a page to estimate the website pages' importance. The core assumption is that the more important websites receive more links from other websites (Page et al., 1999). From another perspective, PageRank can be understood as a "vote" by counting all the ballots from other pages. A website's click works as a vote of support, and there is no support with links amongst websites. A fundamental principle to illustrate the PageRank is presented in Figure 4-1. In this figure, it can be seen that the blue face is much larger than other faces, which shows that it plays a more critical role in that social community.

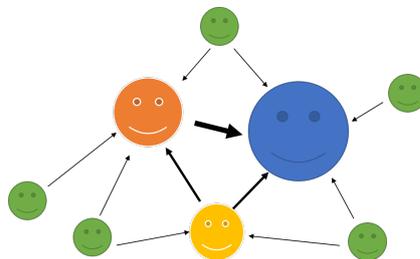


FIGURE 4-1: A basic principle of PageRank, the face sizes indicate the importance

In the network consisting of N pages, the adjacency matrix of the network is represented by $MatrixA_{ij}$, then the element in the matrix is defined as

$$MatrixA_{ij} = \begin{cases} 1 & \text{a link between } i \text{ and } j; \\ 0 & \text{Otherwise.} \end{cases} \quad (4.1)$$

It has to be stressed that a node that does not link to other nodes that we call dangling nodes. Indeed, a dangling node has no direct effect on other nodes in the ranking process. We may treat the dangling as webpages linking to all pages averagely in the system from another perspective. Then it might vote each of the pages randomly. That is, a dangling node can jump (move) to any other pages with equal possibility.

In this way, a network system can be modeled as a graph, where the webpages are represented by nodes, and the edges connecting node i and node j indicate that there is a link linking these two nodes. Then we use D_i to represent the total out link number of the node i , the transition matrix can be defined as follows,

$$MatrixP_{ij} = \begin{cases} \frac{1}{D_i} & \text{If } MatrixA_{ij}=1; \\ 0 & \text{If } MatrixA_{ij}=0; \\ 1/n & \text{If } D_i=0. \end{cases} \quad (4.2)$$

It is necessary to emphasize that the transition probability of a dangling node is set as $\frac{1}{n}$ uniformly for each of the pages in the system.

The PageRank theory supposes that a hypothetical viewer randomly clicking on a link will eventually stop clicking. The damping coefficient of α influences the probability that the person will continue at any step. Various studies have experimented with different damping factors, but it is generally assumed that the damping factor would be better set around 0.85 (Srivastava, Garg, and Mishra, 2017).

Subtract the damping factor from 1 (The result is divided by the number of documents in the collection (N) in some variants of the algorithm), and then add this to the damping factor and the incoming PageRank score. That is, Google's PageRank transition Matrix $Matrix\hat{P}_{ij}$ with the damping factor α as:

$$Matrix\hat{P}_{ij} = (1 - \alpha)MatrixP_{ij} + \frac{\alpha}{n}E \quad \text{Where } E = 1_{n*n} \quad (4.3)$$

The transition matrix $Matrix\hat{P}_{ij}$ is employed to illustrate how transitions are made between two states. It is applied when events are more or less likely depending on the previous events. With the transition matrix, we can obtain the dominant

eigenvector based on the power iteration. Then every value in the vector indicates the final importance of the corresponding page. The PageRank algorithm adopts an arbitrary $\pi^0 = (\frac{1}{n}, \dots, \frac{1}{n})$ as the starting vector and compute the PageRank vector π^{k+1} through the Power Iteration process until converge.

In terms of the Power Iteration Process, we can explain a little bit more, and it is the principal idea in the PageRank algorithm. For a given matrix \hat{P} , it can create a number λ and also a nonzero vector π such that

$$\hat{P}\pi = \lambda\pi \quad (4.4)$$

It is essential to declare that the Power Iteration is a recursive method based on the above eigenvalue algorithm. It commences with a vector π^0 which can be an approximation to the dominant eigenvector or a random vector, that is

$$\pi^{k+1} = \frac{\hat{P}\pi^k}{\|\hat{P}\pi^k\|} \quad (4.5)$$

Typically, the vector will converge to the dominant eigenvector after multiple iterations, where at each iteration, the vector is multiplied by the transition matrix and normalized. Under this circumstance, a threshold ϵ can be defined, once $\|\pi^{k+1} - \pi^k\| < \epsilon$, the vector is said to converge to its dominant eigenvector. The power iteration method is the central pillar in the PageRank algorithm.

4.2.3 Random walk algorithm

As explained, we could receive the final ranking according to the vector π^{k+1} . However, we have to discover the entire system structure and the corresponding transition matrix. In this way, the working mechanism implies a centralized mechanism. Since we could hard to obtain the overall structure, we would like to employ the PageRank algorithm from a distributed perspective. Thus, we can understand the algorithm from the aspect of random walk models. The random walks are assumed to be Markov chains. A Markov chain is described as a set of states. The process starts in one of the states and then moves from one state to another. If the chain is currently at one state, it jumps to another state the next step with a probability that it has no connection to its current state. In general, two random walks are widely adopted and accepted. The first type is merely walking randomly. That means the next state is selected averagely. In the second type, the random walk jumps from one state to another following some probability distribution, which indicates that every state would be visited with a different probability. This is the scenario used in this chapter.

Monte Carlo method can be used to random walk method as random walk estimate importance starting from random pages. The Monte Carlo method is a kind of technology that employs a statistical sampling method to achieve issues that are difficult or impossible to accomplish by other mathematical methods. It has been used a lot in numerous fields already, such as sampling methods to estimate the event frequency to determine the probability. One of its early applications is to estimate the value of π by dropping needles on a floor made of parallel and equidistant strips. It functions like people can generate a number of each trail by rolling a dice several times. As computer technics develops, the increasing computation speed makes this simulation widely used.

In mathematical aspects, it has been proved that the Monte Carlo method works well on PageRank for ranking pages (Avrachenkov et al., 2007). For the given transition matrix $Matrix\hat{P}$ and the PageRank vector π , we know that

$$\pi Matrix\hat{P} = \pi \quad (4.6)$$

As explained, the transition matrix can be represented by the adjective matrix and dangling nodes, that is, $Matrix\hat{P} = (1 - \alpha)MatrixP + \frac{\alpha}{n}E$, we have,

$$\begin{aligned} (1 - \alpha)\pi MatrixP + \frac{\alpha}{n}\pi E &= \pi \\ \Rightarrow (1 - \alpha)\pi MatrixP + \frac{\alpha}{n}\mathbf{1}_{1*n} &= \pi \\ \Rightarrow \pi[I - (1 - \alpha)MatrixP] &= \frac{\alpha}{n}\mathbf{1}^T \\ \Rightarrow \pi &= \frac{\alpha}{n}\mathbf{1}^T [I - (1 - \alpha)MatrixP]^{-1} \\ \Rightarrow \pi &= \frac{\alpha}{n}\mathbf{1}^T \sum_{k=0}^{\infty} (1 - \alpha)^k MatrixP^k \\ \Rightarrow \pi &= \frac{\alpha}{n}\mathbf{1}^T \sum_{k=0}^{\infty} ((1 - \alpha)MatrixP)^k \end{aligned} \quad (4.7)$$

In this way, the PageRank value could be obtained by numerous walks in a distributed way. In the next part, some previous works related to PageRank based recommender systems are presented.

4.2.4 PageRank algorithm in recommender systems

As explained, the main objective of the recommender system is to rank and recommend the top k items to the system users. PageRank (Page et al., 1999) is a popular algorithm to rank vertices in a graph. In essence, PageRank is an instance of the

random walk algorithm used in various fields, including social recommender systems (Gupta et al., 2013). It is usually applied to estimate the relevance between two nodes in recommender systems. An influential factor in these methods success is how to set the weight of two separate users properly. The similarity between the items represents a kind of associate degree between two objects that is often used in the recommender system. SimRank is a general similarity measure between two vertices in a graph (Jeh and Widom, 2002). It can be interpreted as the expected meeting distance of two walkers, which starts from two distinct objects, respectively, randomly and simultaneously walk step by step (Jeh and Widom, 2002; Nguyen et al., 2015). Based on the SimRank algorithm, a method called UniWalk, which is based on the random walk method, is proposed for items recommendation in large-scale and distributed graph frameworks (Song et al., 2018; Shao et al., 2015). Paudel, B et al. deal with updating recommendations on evolving graphs for interactive applications using random walk techniques in recommender systems (Paudel et al., 2017). In the literature (Deng, Huang, and Xu, 2014), an extended random walk algorithm is studied to earn recommendation results in the web service recommender system. In the literature (Cooper et al., 2014), Cooper et al. proposed a recommendation system based on short random walks. Three algorithms, namely P^3 , P^5 and P_α^3 , are proposed in the graph. P^3 and P^5 start random walks at a node for fixed lengths three and five respectively, and P_α^3 raises the transition probabilities to the random walk process. These methods adopt the random walk method by setting the weight of links to the similarity between two items in recommender systems.

As a human characteristic, trust is also widely applied to computer science to elaborate on the expectation of the target agent. In the literature (Kerchoue and Dooren, 2008), the PageTrust system considers a random walk that traverses trust edges but remembers the list of all distrustful nodes seen thus far makes sure never to visit such a node in the remainder of the walk. TrustWalker (Jamali and Ester, 2009) analyzes a random walk model combining the trust-based and the collaborative filtering approach for the recommendation. From the literature (Andersen et al., 2008), we can find that the trust and random walk-based recommender system aims at generating personalized recommendations by integrating the other users in the social network. A social web-based service recommendation method with trust enhancement known as RelevantTrustWalker is proposed in (Deng, Huang, and Xu, 2014). Firstly, a matrix factorization method is used to obtain the trust degree and item similarity between any two users in the social network, and then the extended random walk algorithm is utilized to assess recommendation results, fixed six steps are predefined in each random walk process.

Most random walk-based recommender systems perform random walk with

fixed length 3, 5, or 6 (Cooper et al., 2014; Deng, Huang, and Xu, 2014), or move and terminate with a transition probability. The transfer matrix could be either on the basis of the similarity measures or the trust relationship. However, service recommender systems, which base on random walk algorithm and feedback-based trust in a distributed multiagent system, have seldom been emphasized. Compared to the previous works, we propose a service recommender system in a distributed multiagent environment. An agent is an autonomous and interactive entity representing a node of the considered graph in this context. Meanwhile, we use the feedback-based approach to obtain the trust degree between any two agents, and the trust value is in the range of 0 and 1. It is to say, the trust degree only relates to direct interactive feedback. Linguistic preference is employed to express the clear preference, and Simhash is considered as the tool to capture the similarity. Finally, we conduct a random walk algorithm for recommendations based on trust values. An agent keeps moving forward with the transition probability c if it has credible knowledge. Otherwise, it terminates the walking. More details are discussed in the next parts.

4.3 Random walk for manufacturing service selection

This section presents the proposed multiagent system in detail for manufacturing cloud sharing in a distributed environment. First, we define the manufacturing cloud service that agents maintain in a multiagent system, followed by explaining a resource requirement from a resource client agent's perspective. What follows, we define the trust estimation approach between a manufacturing cloud provider and manufacturing client agents, on which our proposed system is based. Then we introduce the random walk based resource sharing approach in a distributed environment. The main process is displayed in Figure. 4-2.

4.3.1 Cloud manufacturing services

One of the significant challenges in cloud manufacturing systems is selecting and scheduling multiple cloud resources with distinct objectives regarding time, consumption, quality, and reliability. So far, numerous researches have studied cloud manufacturing selection problems in central systems. In the centralized operation mode, the cloud platform collects all resources and requirements and selects all required services that fit the query. Some criteria have been well studied in the previous work, ranging from cost-related, time-related indices, quality-related indices, and risk-related indices reliability or trust-related indices, environment-related indices (He et al., 2019). However, in the distributed operation mode, individual

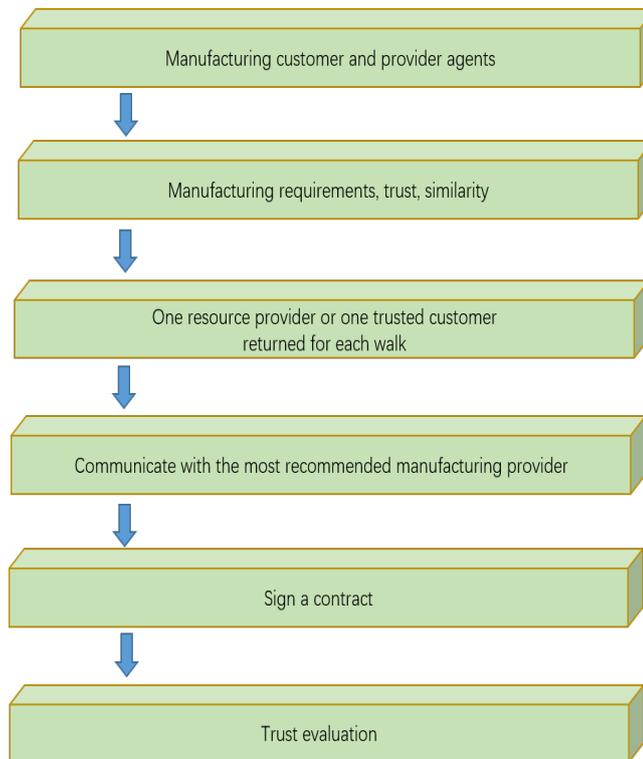


FIGURE 4-2: Random walk process for provider selection

agents maintain their manufacturing cloud, and no agent maintains full information. Thus, query similar and trusted clients for recommendations related to the manufacturing requirement is meaningful for cloud selection.

Generally, the first issue to follow a recommendation is to express its preferences, and the recommendation from similar clients with a similar experience is always estimable. It is easy to find similar clients if the general information is at hand. However, in a distributed cloud manufacturing system, the cloud manufacturing services, as well as interaction feedback, are stored distributed. Therefore, it is not easy to ask for data sharing in such a kind of system. Even worse, it may bring additional privacy leakage risk and decreases the feasibility of recommendation, as stated by (Shen et al., 2017b). On the other side, even some personal data are shared selflessly, and it can also be hard to guarantee that the provided information is correct. Thus, in this chapter, we use a linguistic description to express the requirements, and Simhash is employed to capture the similarity of service and requirement. When asked for a recommendation, a random walk approach is used, which will be discussed in section 4.3.4, and we hold that the recommendations from trusted and similar clients are valuable.

In our opinions, services relate to multi-attributes and their corresponding values. The attribute is defined as a quality or feature regarded as a natural or inherent part of someone or something. When talking about services, the first attribute restricts the service type, which can be any manufacturing types in this chapter, such as food manufacturing, textile industry, furniture manufacturing, automotive manufacturing, etc. The rest attributes are used for detailed descriptions. For instance, materials and functions can be employed to describe furniture manufacturing. In summary, manufacturing services in our system of resource sharing can be formally summarized as follows.

- $R: \{r_1, r_2 \dots r_j \dots r_{card(R)}\} \Rightarrow$ Finite set of manufacturing types;
- $\forall r_j \in R: c_{r_j}: \{c_{(j,1)}, c_{(j,2)} \dots c_{(j,k)} \dots c_{(j,card(c_{r_j}))}\} \Rightarrow$ Finite set of characters to resource type r_j and $c_{(j,k)}$ represents the k th character;

Corresponding, to the agent a_i , it owns several manufacturing clouds and the associated characters can be represented by

- $R_i: \{r_1, r_2 \dots r_j \dots r_{card(R_i)}\} \Rightarrow$ Finite set of all acquired resource types;
- $C_i = \bigcup_{j=1}^{card(R_i)} (c_{r_j}) \Rightarrow$ Finite union set of all characters to the agent a_i .

As is explained, manufacturing capability and resources with distinguishing attribute values are placed among agents. To the agent a_i , all its resource types R_i , the corresponding characters C_i and their values are summarized in the following list,

1. $\{Rt_1 = r_1, c_{(1,1)} = value_{(1,1)}, c_{(1,2)} = value_{(1,2)} \dots c_{(1,k)} = value_{(1,k)}\}$
2. $\{Rt_2 = r_2, c_{(2,1)} = value_{(2,1)}, c_{(2,2)} = value_{(2,2)} \dots c_{(2,k)} = value_{(2,k)}\}$
3. ...
4. $\{Rt_i = r_i, c_{(i,1)} = value_{(i,1)}, c_{(i,2)} = value_{(i,2)} \dots c_{(i,k)} = value_{(i,k)}\}$
5. ...

where Rt_i means the i th resource type, $value_{(i,k)}$ denotes the exact value to the character $c_{(i,k)}$.

4.3.2 Preference representation

In this subsection, the Simhash approach, which is adopted for privacy-preserving, is introduced in detail. Simhash is a technique for estimating how similar two descriptions are. That is to say, a client uses Simhash to capture the similarity between the required manufacturing cloud and its previous experiences.

For each of the manufacturing cloud agents, a_i , its manufacturing resources and manufacturing capability are represented by keywords and the related captures, as stated previously. In the distributed environment, a_i does not send a detailed requirement worldwide. Alternatively, the services' hash values are announced to the cloud platform to find potentially similar clients for privacy-preserving. This method's main idea is that the agents who have experienced the same manufacturing cloud can be regarded as potentially similar clients. After that, a random walk is conducted starting from these potential agents. The details of the hash process are presented as follows,

1. Firstly, build manufacturing field indexes offline, including the weights of the keywords. In the proposed method, all manufacturing clouds are represented by linguistic descriptions (text), and the descriptions build the font library. It has to be noticed that different keywords in this font library are of distinct weights, and the cloud platform conducts this process. Some applications, such as NLTK (Bird, Klein, and Loper, 2009) and Jieba (Junyi, 2015), can be employed to segment a document and find keywords and the related weights. For example, for the manufacturing requirement, Ms_j described by *I need to manufacture computers and smartphones*. Would find that computers and smartphones are the keywords in this manufacturing requirement.
2. Secondly, for each of the keywords kw_k in the service Ms_j , we convert it to 64-bit hash values, which is represented by $h(kw_k)$. For the given example of producing smartphones and computers, we suppose their weights of keywords are $w(\text{smartphone}) = 3$ and $w(\text{computer}) = 2$, respectively. With the same hash function, we have $h_1(\text{smartphone}) = (0, 1, 0, 0, 1, 0)$ and $h_1(\text{computer}) = (1, 1, 0, 0, 0, 1)$ for example.
3. What follows, we rewrite the 0-1 dimensions and replace value "0" by value "-1", after revision, a new dimension h_2 is received. That is, $h_1(\text{smartphone}) = (0, 1, 0, 0, 1, 0)$ after modification should be $h_2(\text{smartphone}) = (-1, 1, -1, -1, 1, -1)$.
4. After that, the weights of the related keywords are employed to rewrite the revised dimensions. Each of the values, both 1 and -1, multiply by the weight, thus for the keyword smartphone, we have $h_3(\text{smartphone}) = (-3, 3, -3, -3, 3, -3)$, and in the same way, we have $h_3(\text{computer}) = (2, 2, -2, -2, -2, 2)$;
5. Next, for both the keywords, we calculate the sum of their columns, and a Hash function can represent this manufacturing requirement, that is to say, for this requirement, we have $h_4(Ms_j) = (-1, 5, -5, -5, 1, -1)$. Finally, we transfer the

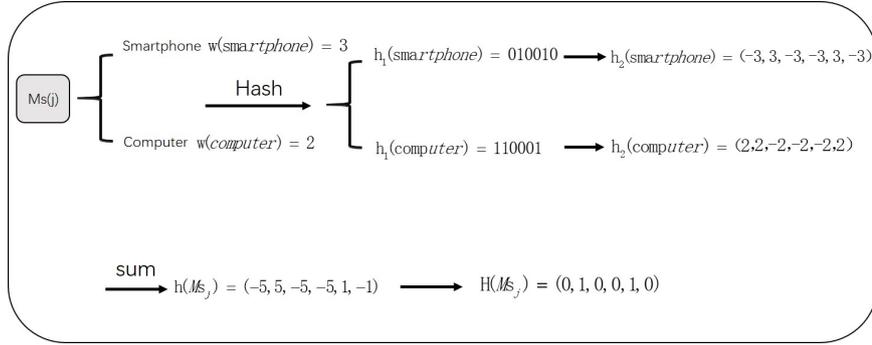


FIGURE 4-3: An example: Manufacturing indexes building procedure

positive values to 1 and the negative values to 0, and we could see that the final hash function can be represented by $H(Ms_j) = (0, 1, 0, 0, 1, 0)$, which can be regarded as the index for the manufacturing requirement Ms_j according to Simhash theory. In this way, we can build indexes for all the manufacturing services, and this index would be sent to other manufacturing providers from the perspective of privacy-preserving. For a better explanation, Figure.4-3 shows the detailed process of hash a service.

In this way, according to the same hash function in the previous process, we can obtain all manufacturing service indexes in the manufacturing user's request set and all possible manufacturing services that one provider can present. Thus, a manufacturing request can be represented by a hash number. As stated, manufacturing consumers can query a needed service and adopt the recommendations from trusted similar clients, thus, we have to obtain the similarity of the requests and the agent's historical experiences. Humming Distance, denoted by $D(H(Ms_j), H(Ms_k))$, shows the distance between the manufacturing service Ms_j and Ms_k . Here, the manufacturing services are represented by a 64-dimensional vector $(1, 1, 0 \dots v_{j,r} \dots)$ and the Humming distance is calculated as follows,

$$D(H(Ms_j), H(Ms_k)) = \frac{\sum b_x}{64} \quad (1 \leq x \leq 64) \quad (4.8)$$

where, b_x is a Boolean value by the following equation,

$$b_x = \begin{cases} 0, & v_{j,r} \neq v_{k,r} \\ 1, & v_{j,r} = v_{k,r} \end{cases} \quad (4.9)$$

According to the definition of Simhash (Shen et al., 2017b), we can conclude that the two manufacturing services are similar if $D(H(Ms_j), H(Ms_k)) < 3/64$ holds. In

other words, the two manufacturing services are probably the same, and the manufacturing users (manufacturing user and manufacturing client) have similar experiences, and the recommendation from similar clients can be usable.

As described in section 4.3.1, the manufacturing resources are represented by a set of characters, and the first character is used to define the detailed manufacturing types while the other characters are employed to capture the detailed information. A task T_k , in the cloud manufacturing, can be decomposed into multiple subtasks ($subT_{(k,1)} \dots subT_{(k,j)} \dots$), and each of the subtasks $subT_{(k,j)}$ can be transformed into a Hash value $H(subT_{(k,j)})$. Thus, the similarity of the resource client and the client with previous experience can be calculated by

$$Sim_{task}(T_k, T_{k'}) = 1 - \max_{j=1}^m D(H(subT_{(k,j)}), H(subT_{(k',j')})) \quad (4.10)$$

where $T_{k'}$ is the agent a_j 's previous experience, and $subT_{(k',j')}$ is the subtasks of the task $T_{k'}$.

Pratically, cloud manufacturing clients have personal preferences over the required resources, as stated in section 4.3.3. The detailed weights related to time, cost, quality, and security are always different from agents. Thus, the preference similarity is necessary to present a recommendation. Imagining that a cloud manufacturing client needs good-quality resources, and the recommender has a similar experience, but it values cost much than time, then the recommendation from this client might not fit the client's requirement. Supposing that the exact weight regarding time, cost, quality, and security are represented by $\omega_t^i, \omega_c^i, \omega_q^i, \omega_s^i$. In term of the similar client, its weights are denoted as $\omega_t^j, \omega_c^j, \omega_q^j, \omega_s^j$. the similarity of their preferences are defined as follows,

$$sim_{pre}(a_i, a_j) = \sqrt{\frac{\sum_{h=t,c,q,s} (\omega_h^i - \omega_h^j)^2}{4}} \quad (4.11)$$

Thus, a manufacturing service user would try to ask this probably similar client for recommendations. However, some deceptive agents might present misleading information, which may result in decision errors. In the next part, we would discuss trust estimation in a distributed environment.

4.3.3 Trust estimation in the proposed distributed multiagent systems

The proposed manufacturing system consists of manufacturing units that are distributed located in the systems. Each of the manufacturing units contains numerous manufacturing resources or manufacturing capabilities that are of distinguishing features. Any two agents can raise interactions in two ways, one is by requiring manufacturing resources when acting as a resource client agent, and another is to recommend required information from the perspective of third-party agents. Thus Let \mathcal{X} be the system as $\mathcal{X} = \langle \mathcal{A}, \mathcal{I} \rangle$, where

- \mathcal{A} : A finite set of agents, where $\mathcal{A} = \{a_1, a_2, a_3 \dots a_i \dots\}$;
- \mathcal{I} : A finite set of interactions, where $\mathcal{I} = \{\mathcal{I}_{(a_1, \cdot)}, \mathcal{I}_{(a_2, \cdot)} \dots \mathcal{I}_{(a_i, \cdot)} \dots\}$;

The finite set of agents \mathcal{A} consists of all agents in the proposed system. In a social network, any participants are potential service consumers because of daily needs, and the others might offer available recommendations to win social welfare, social influence, or merely building trust, credibility and respect. That is to say, all agents in the proposed system could be service provider agents and client agents, or simultaneously both. The participation or leave of an agent leads to the variability of the number $card(\mathcal{A})$.

Meanwhile, we could understand the system from another perspective. As stated previously, agents occupy different kinds of services, and they communicate, negotiate, cooperate, to reach an agreement on resource sharing. Thus links between two agents should be a mapping, such that two connected agents share information or services over the before-mentioned items. For instance, the agent a_j pays a_i for smartphone manufacturing resources; The agent a_j queries the agent a_i to recommend reliable phone manufacturers. Therefore, the system can also be interpreted as an information network. In order to better understand the information network, we first introduce the following definitions,

Definition 8 *An information network is defined as a direct graph that comprises agents and/or methods organized to collect, process, transmit, and disseminate data which can be represented by $\mathcal{G} = \langle \mathcal{R}, \mathcal{E} \rangle$ (Shi et al., 2016). An object type mapping function $\varphi : \mathcal{R} \rightarrow \mathcal{A}$ and a link type mapping function: $\psi : \mathcal{E} \rightarrow \mathcal{I}$.*

Each object $o \in \mathcal{R}$ belongs to one particular object type in the object set $\mathcal{R} : \varphi(o) \in \mathcal{A}$, and each link $e \in \mathcal{E}$ belongs to a particular relation type $\mathcal{I} : \psi(e) \in \mathcal{I}$. If two links belong to the same relation type, the two links share the same starting object type as well as the ending object type. If the types of objects $|\mathcal{R}| > 1$ or the types

of relations $|\mathcal{E}| > 1$, that information network is called heterogeneous information network; otherwise, it is a homogeneous information network. The properties of this information can be summarized as follows,

- (1.) Non-reflexivity. There is no relationship pointing to agents themselves, such as $e(a_i, a_i)$. That is to say, we don't accept using their own resources as a sharing;
- (2.) Asymmetry. The binary relationship among users is directional, which means the relation $e(a_i, a_j)$ does not infer the relation $e(a_j, a_i)$.
- (3.) Non-transitivity. Namely, $\exists e(a_j, a_i) \wedge \exists e(a_j, a_k) \not\Rightarrow e(a_i, a_k)$

Most of the contemporary information network reviews have a basic assumption that the type of objects or links is unique (Sun and Han, 2012). That is to say, the homogeneous network contains the same type of objects and links, such as the author collaboration network and the friendship network (Lichtenwalter, Lussier, and Chawla, 2010), and in the cloud manufacturing system, all agents have only one kind of manufacturing cloud and share this service mutually. In what follows, we would discuss the trust relationship that connects two agents with on less than one links in the proposed multiagent system. Firstly, a trust relationship between any agent pairs in the homogeneous information system is studied. Afterward, we analyze the trust relationship in a heterogeneous information network.

Edge weights–Trust relationships

Trust is a complicated concept in our everyday life, the main reasons for studying trust can be summarized from the following aspects (Yan, Zhang, and Vasilakos, 2014)

1. Trustee's objective characteristics, such as a trustee's safety and dependability. Notably, reputation is a public estimation of the trustee concerning its previous behaviors and performance.
2. Trustee's individual properties, such as trustee honesty, benevolence, and morality.
3. Trustor's individual properties, such as trustor disposition and willingness to trust.
4. Trustor's physical properties, such as the trustor's criteria or policies for a trust decision.

5. Context that the trust relationship remains, such as the purpose of the trust, the environment of trust (e.g., time, location, activity, devices being used, their operational mode, etc.), and the risk of trust.

In this chapter, we mainly focus on agents' trustworthiness and the willingness to provide real information or service, namely, the first two aspects as enumerated. We first generalize how trust is updated with previous experiences in the heterogeneous information network of providing information or contributing services. The detailed process is discussed as follows.

We analyze the trust relationship between any agent pairs, and each agent a_i keeps track of two trust value TV^{mr} and TV^{if} ($TV \in [0, 1]$) to the agent who has provided it manufacturing resources and information, respectively. It is essential to emphasize that trust only exists between agents directly interacting by providing resources or sharing information. Furthermore, if two agents are strangers or without sufficient evidence, then the indirect trust can be inferred by "The recommendations given by reliable recommenders tend to be trustworthy (The friends of my friends are also my friends)".

Interactions between any two agents can be viewed as an independent trial. If each trial can result in two possible interactive feedback outcomes, we call one of these outcomes a success, and the other, is a failure. The trials are independent and keep the same probability of success. Then this trial satisfies the Bernoulli distribution, and multiple trials, therefore, satisfy Binomial distribution. As a result, Beta reputation can be employed to study the individual probability of success. Similarly, Dirichlet distribution can also be used to study the probability of a fixed feedback outcome if more than two results are available in one trial.

Mathematically, the Beta reputation system and the Dirichlet system are suitable to qualify trust in multiagent systems (Jøsang and Ismail, 2002; Fung et al., 2011). However, these systems are on the foundation of multiple interactions and sufficient prior experiences. Some interactions can not be displayed multiple times, such as earthquakes or even resource sharing. What is more, in a multiagent system, an agent can have oscillating behaviors because of their characteristics or the changing environment. That is to say, the probability of success might differ from circumstances. Moreover, an agent pair might not raise many interactions in a system consisting of numerous agents. As a result, the Beta reputation system and the Dirichlet system are no longer appropriate in practical life. Thus, we propose this interactive feedback based trust model, where the feedback outcome of the r th interaction raised by a_i with a_j over the resource s_k is represented by the evaluation $Eva_{(a_i, a_j, s_k)}(r)$ which is in the range of 0 and 1, zero means completely bad performance, and one means extremely excellent performance.

The performance evaluation of the received resources can also be interpreted as the main reason for trust, which indicates the probability of a second interaction. For instance, the agent a_j provided the service s_k to the agent a_i , and a_i evaluates a_j to be an excellent agent that satisfied it. If a_i searches for the same resource the second time, obviously, a_j would become its prospective resource provider. Now the problem is how to decide the exact trust value according to the feedback outcome in a dynamic multiagent system. We propose to use the equation below as

$$TV(r+1) = \delta(r) \times TV(r) + (1 - \delta(r)) \times Eva(r) \quad (4.12)$$

where TV (including TV^{mr} and TV^{if}) indicates the trust estimation after the (r) th interaction (before the $(r+1)$ th interaction). The present trust $TV(r+1)$ is determined by the previous trustworthiness $TV(r)$ and the current performance $Eva(r)$.

It is also challenging to determine the exact values of the parameters $Eva(r)$ and $\delta(r)$. In terms of $Eva(r)$, the detailed value of providing cloud manufacturing resources differs from presenting information. Firstly, we present how to decide $Eva_{(a_i, a_j, s_k)}^{mr}(r)$, the detailed evaluation of providing cloud manufacturing resources.

As stated in (Meng et al., 2014; Huang and Wu, 2020), we perceive time, cost, quality, and security are primary factors that employed for platform trust estimation. Thus, inspired by (Meng et al., 2014), the evaluation of the a_j from the perspective of a_i in terms of the resource s_k is defined in a weighted average approach. To the influential factors time, cost, quality and security are respectively represented by S_t , S_c , S_q , S_s and their corresponding weights are ω_t^i , ω_c^i , ω_q^i , ω_s^i , and $\sum_h \omega_h = 1$ and $\omega_h^i \geq 0 (h = t, c, q, s)$, thus the evaluation is calculated as follows,

$$Eva_{(a_i, a_j, s_k)}^{(r)} = \sum_{h=t, c, q, s} \omega_h^i \times S_h \quad (4.13)$$

where $S_h \in \{0, 1\}$ and $S_h = 1$ indicates that the service demander is satisfied with the received resource in terms of time, cost, quality, and security.

Regarding the parameter δ when evaluating the detailed performance of the received manufacturing cloud, we use the same definition in Equation 3.15 of section 3.3.1. That is to say, the direct trust of the manufacturing cloud is influenced by the past value of trust and performance fluctuation.

Next, we have to determine the value of trust TV^{if} when presenting recommendations, and it is decided by the current performance related to the received resources. The current performance of the recommendation $Eva_{(a_i, a_j, s_k)}^{if}$, recommended by the agent a_j to the agent a_i in terms of the resource s_k owned by the agent a_j , is defined as $Eva_{(a_i, a_j, s_k)}^{if} \in \{0, 1\}$, and $Eva_{(a_i, a_j, s_k)}^{if} = 1$, if $Eva_{(a_i, a_j, s_k)}^{mr}$ is greater than 0.5,

otherwise, $Ev a_{(a_i, a_j, s_k)}^{if} = 0$.

From our perspective, many factors influence the exact value of δ , including the total interaction amount, the oscillation of the current evaluation and actual performance, and the particularized trust concerning the other resources. In what follows, we investigate recommendation trust in a heterogeneous information network.

As explained in the previous part, the trust value of an agent is estimated by $TV_{(a_i, a_j, s_k)}^{(if)}$ if the agent a_i and the agent a_j have interactions over recommending the resource s_k . Any agent pairs can communicate and share multiple resources information more than once. As discussed, trust is known as the willingness, desire, or ability to share the correct information. It is a kind of intrinsic property of an agent deeply. Thus, the probability of sharing some exact information affects the trust of sharing other different information. Therefore, we propose the following equation to model the trustworthiness in heterogeneous information networks. Each time, we keep the following instructions in mind,

- 1.) For those agents who used to act fluctuating, an evaluator would rely more on the previous trustworthiness, otherwise, both previous trust and current performance are important.
- 2.) Service providers tend to be trustworthy unless they are reliable when sharing other different resources.

Thus, we define the value of δ by

$$\delta_{(a_i, a_j, s_k)}^{if}(r+1) = (1 - \bar{\Delta}^{if}(r)) \frac{\sum_{p=1, p \neq k}^K TV_{(a_i, a_j, s_p)}^{if}}{K} \quad (4.14)$$

$\bar{\Delta}(r)$ is the average oscillation of the previous evaluation and the actual performance. Each round, the agent a_i records the total interaction amount r with the agent a_j on the resource s_k , thus for the next interaction,

$$\bar{\Delta}(r+1) = \frac{r * \bar{\Delta}^{if}(r) + |TV_{(a_i, a_j, s_p)}^{if}(r) - Ev a_{(a_i, a_j, s_p)}^{if}(r)|}{r+1} \quad (4.15)$$

$\frac{\sum_{p=1, p \neq k}^K TV_{(a_i, a_j, s_p)}^{if}}{K}$ indicates the average difference between the estimated trustworthiness and the actual evaluation of performance.

In this way, $TV_{(a_i, a_j, s_p)}^{(r+1)}$ is obtained to represent the trustworthiness value of the agent a_i to the agent a_j on sharing the resource s_k . Recall the definition of trust, trust value is computable, agents could dynamically build or lose trust. This trust value would be of great use later for potential manufacturing cloud providers selection for resource sharing.

4.3.4 Interactions and decision making

The agent-based manufacturing cloud system proposed in this chapter is highly dependent on autonomous agents' roles, especially communication technology and reasoning. In this part, we design an interaction protocol that enables agents to dialogue and transmit information to select appropriate manufacturing services—usually the top-ranking alternative. The intelligent agents' actions are firstly analyzed. If an agent acts as a cloud manufacturing client, then

1. it generates a query;
2. it studies the trust relationship and transmits it to the prospective service provider in the system;
3. it chooses the right service provider and proposes to raise an interaction;
4. it updates its memory, both for resource providers and information recommenders.

What if an agent acts as a service provider, then

1. it studies the preference and responds to the query;
2. it recommends a good-evaluated resource provider or transmits the query to trusted trustees;
3. it provides the required services.

Figure 4-4 shows a sequence diagram of the agents' communication in the proposed system for manufacturing service sharing. Each time a service client agent (SCA) generates a query, it sends the query to the other trusted service provider agents (SPA). A SPA firstly modernizes its local trust and similarity according to its previous experiences, and if the SPA has ever demanded the to-be-requested resource, then the resource provider agent is reported. Otherwise, this SPA has no similar requirement as the resource client agent and therefore recommends a reliable client to the service client agent. In this way, the service client agent remembers the recommendation and continues querying the approved client agent for more information. Each round, the SCA receives a corresponding recommended provider agent or a reliable client agent. By repeating the process several times, the global trust relationship and agents' preference are studied, and negotiation and cooperation are raised with the best-evaluated service cloud provider agent. Finally, the service cloud client agent updates its memory of the service provider agents according to the interactive feedback.

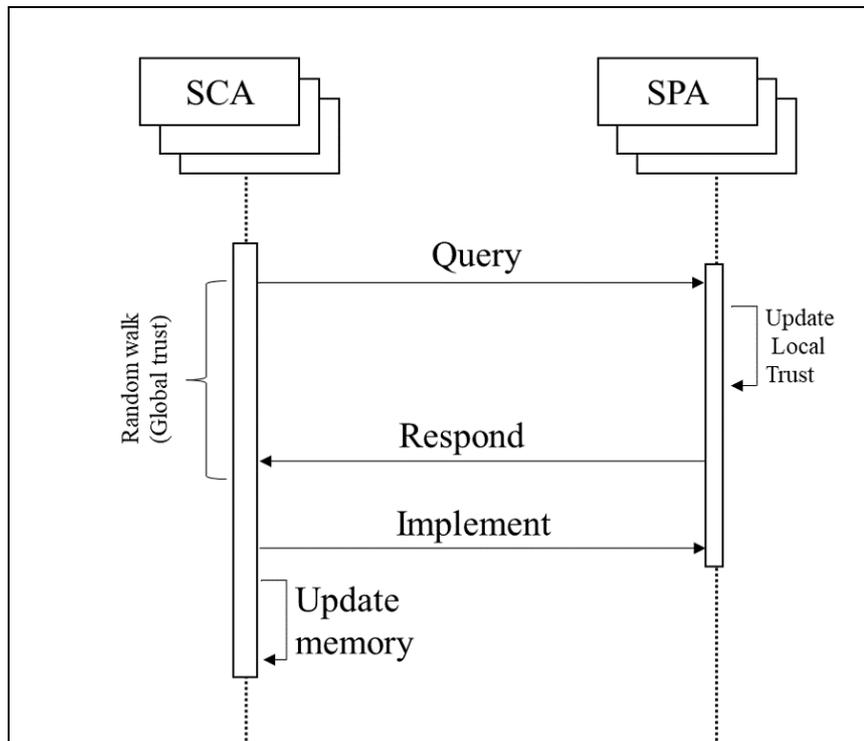


FIGURE 4-4: A sequence diagram of agents communication

As is discriminated in section 4.2.4, when confronting the problem of filtering a recommendation, it is not only decided by its interactive performance but also the trust values. We can imagine that a patient agent is looking for a dentist agent while one passenger recommends it one of its trusted doctors. At the same time, the patient agent's doctor asks it to see another professional dentist agent. I suppose that the patient agent will follow the recommendation from its doctor agent. Here we use the Monte Carlo and random walk method for recommendation selection.

Recall the **MC complete path** algorithm in (Avrachenkov et al., 2007), we conduct a random walk process in a cloud manufacturing system to select service provider agents based on agent trust and manufacturing service similarity. When manufacturing service provider agents receive a query in the proposed distributed environment, it first examines its previous experience and presents a resource provider agent that can provide the required service or one of its trusted client agents. If it has neither a similar experience nor trust client agents, no more information/recommendation is presented. As a result, a random walk at agent a_0 could either provide service recommendations or terminate if no trusted agents and similar requirements, which is more reasonable than that of jumping to a random page when a node has no neighbors and this is also why we trust the **MC complete path** in (Avrachenkov et al.,

2007).

In summary, our method reaches a final result through numerous iterations, each of which a random walk starts from a prospective provider agent a_0 in the weighted multiagent systems, the next walk will reach a particular agent a_j . If the agent a_j occupies or has ever asked the required resource, Then the client agent, as well as the similarity and trust, are reported directly as a result of this iteration; otherwise, the random walk process terminates at the current node. In this way, according to the received recommendation, an excellent resource provider agent can be selected.

At present, the existing methods usually choose the target node randomly, which means each agent whom the target agent trusts have an equal chance of being selected. Also, some references proposed to select an agent based on similarity and trust. However, agents might have no same experience related to the target agent. Thus, we propose to first select an agent according to the trustworthiness concerning the same manufacturing cloud. Otherwise, the agent would move to a target that has general trust over other resources. We propose that the target agent a_0 for the next step from the current agent a_i to the next agent a_j is selected according to the following probability:

$$PS_{(a_i, a_j)} = \begin{cases} (Sim_{task} * Sim_{pre} * TV^{mr})^{\frac{1}{2}}, a_j \text{ has interactions over the resource} \\ TV^{if}, a_j \text{ has no interactions over the resource} \\ 0, a_j \text{ has no trusted clients;} \end{cases} \quad (4.16)$$

Sim_{task} is the similarity between the manufacturing cloud s_k and a_j 's previous experiences, Sim_{pre} indicates the preference similarity between the target agent and the to-be-recommended agent. TV^{mr} is the trust value of the received cloud manufacturing. That is to say, if the to-be-asked agent has similar experience, it would recommend the most similar and trusted one. Otherwise, it follows the trust value TV^{if} of presenting recommendation and presents a trusted client. Under each of the circumstances, if the agent a_i has more than one trusted agent, it uses the roulette wheel algorithm to select the best agent. In this way, a reliable recommendation, either a to-be-asked cloud provider agent or a trusted client agent, is received. Finally, the resource client agent selects one potential agent to sent a cooperation message. They, therefore, dialogue, negotiate, and reach an agreement for resource sharing.

Typically, all recommendation consumers are likely to cooperate with the top-ranking agent. However, service providers have limited space and energy to provide enough services for all agents. For instance, an experienced doctor is impossible at work seven-eleven, it needs to take rest well, and it has a personal life. Furthermore,

the agents without enough interactive experiences would rank poorly, and it is hard to show their performances and increase their ranking. However, we also accept that more satisfied/experienced service providers should raise more transactions. As a result, here in this chapter, we adopt the roulette wheel algorithm to select an agent to get rid of always choosing the top-ranking one and provide more chances to the inexperienced agents. Roulette wheel selection is a general algorithm used to select an object proportional to its probability (Lipowski and Lipowska, 2012). The critical factor of the roulette wheel algorithm's success is knowing the probability of reaching each target object. In this chapter, we assign the estimated personal PageRank value, namely the cumulative recommendations $(Sim_{task} * Sim_{pre} * TV^{mr})^{\frac{1}{2}}$, as the probability to the service provider. We use the roulette wheel algorithm ψ times, and one agent is selected each time. As a result, ψ agents are selected in total, and the selected agent is the one that appears the most. Agents' selection process is presented in RWtoSA algorithm (Algorithm 2) in details as follows,

Algorithm 2 RWtoSA algorithm

```

1: L: the length of PRV;
2: R: Generate L random number, each number is between (1, $\psi$ )
3: for i in range( $\psi$ ) do
4:   sumPRV=0;
5:   J: Generate a random integer between (1,L);
6:   while sumPRV < R(i) do
7:     sumPRV+ = PRV(mod(J - 1, L) + 1)
8:     J+ = 1
9:   EndWhile
10:  Record the selected trusted agent  $a_j$ ;
11: EndFor
12: Return The agent which appears the most in  $\psi$  times;

```

4.3.5 Evaluation and dynamic network structure

We propose the recommender system-like approach to provide recommendations for cloud manufacturing consumers, and one of the goals is to provide as much satisfactory service as possible. Therefore, we evaluate the performance of the proposed system through global satisfaction, which can be obtained through $Eva_{(a_i, a_j, s_k)}^{mr}$. Thus, system success rate SR is defined as

$$SR = \frac{\sum_k Eva_{(a_i, a_j, s_k)}^{mr}}{\text{number of interactions}}$$

This gives us a measure for quantitatively comparing the performance of the proposed trust-based recommender system. An algorithm for intuitive displaying the proposed model's process is shown in the *WalkRS* Algorithm (Algorithm 3) as follows.

Algorithm 3 *WalkRS* Algorithm

Input: maximum rounds; c : preset probability;

Output: SR : Satisfaction

```

1: if  $a_i(a_i \in A)$  needs services then
2:   for agent in  $A$  do
3:     round=0
4:     while round < maximum rounds: do
5:       #Run a single random walk;
6:       Starting Node=[agent]
7:        $c$ =random number between (0,1);
8:       while  $c$  < preset probability & starting node has trusted clients or
          similar experiences; do
9:         Random walk and record the client;
10:         $c$ =random number between (0,1);
11:      EndWhile
12:      Record the path of the single random walk;
13:      round+=1;
14:    EndWhile
15:    Record the random walk path;
16:  EndFor
17: EndIF
18: (#Data analysis)
19: Estimate the PageRank value;
20: Select an agent through PageRank value;
21: Update the memory;
22: Update the edge weight;
23: Calculate the satisfaction
24: return Satisfaction

```

4.4 Conclusion

The detailed and appropriate manufacturing service representation is a significant step for efficient resource sharing in the distributed system of manufacturing cloud selection, especially from the privacy-preserving perspective. Also, reaching and evaluating the potential service provider agents according to the recommendations given by third-party agents is an arduous task, especially for strange provider agents

in a distributed environment to facilitate their recognition, communication, and further interaction. It is essential to ensure the reliability of the shared resource and prevent potential threats. However, traditional multiagent manufacturing cloud systems have rarely considered trust and pay less attention to provider agents' reliability evaluation.

In this study, we proposed a preference and trust-based manufacturing cloud sharing approach, which comprises the following aspects: (1) We first defined what exactly resources are shared in a distributed multiagent system, following by displaying how to present a manufacturing service and how to capture the difference or similarity between cloud services. These aspects are necessary for a recommender system-like resource sharing since the recommendation given by *A* might not fit for *B*. With the detailed definition of resource and preference, the suggestions provided by trusted and similar third-party client agents would be more usable. (2) A trust evaluation approach. The traditional trust model takes an agent's performance in some aspects as the agent's entire trust degree. However, agents might have different performance over different aspects. Thus, we regard the system as a heterogeneous information network and propose capturing trust from different interactions. (3) We use the Monte Carlo based random walk approach to filter excellent service provider agents; agents' trust and preference similarity are used to calculate the probability of selecting a traversal target at each step. To conclude, our approach uses trust and personal requirement similarity to select superior resource sharing providers in a distributed environment. Our proposed trust and similarity-based agent detection method can be used in further distributed resource sharing systems for practical significance.

Chapter 5

Application and validation

This chapter presents two main issues, namely the simulation and the validation. The application indicates how the proposed methods can be used appropriately for trust estimation in multiagent systems and applied in cloud manufacturing resource sharing, while the validation would be used to study the proposed method's effectiveness. The rest of the chapter is structured as follows. The simulation result analysis begins this chapter, in what follows, by studying comparisons with other methods to present its validation, including the theoretical aspects, the advantages, and limitations.

5.1 Trust estimation models

In order to examine the performance of the proposed model, it is tested using simulation experiments from the following aspects. Firstly, we have to see if the BPA generation approach is sensitive to an unexpected performance change. After that, we check the entropy-based certainty approach runs correctly and rationally. Finally, the assigned credibility method in the proposed method is compared to the previous works to see its efficiency for trust estimation.

5.1.1 Evidence generation

The proposed model aims to maintain the trust of agents, in which agents can react differently over time. That is to say, both resource provider agents and information provider agents (witnesses) can decide to move from genuine to malicious or the other way around, as stated in section 1. As a result, when an agent switches its performance deliberately, the proposed method enables agents to notice the agility difference and take sharp responses. This part mainly examines if the resource client agent can react sharply, facing an unstable resource provider agent, namely, to investigate the efficiency of the proposed direct trust generation approach.

We fix one resource provider agent and one resource client agent in the experiment. Each round, the resource client agent solicits the resource provider agent

for some resources, and correspondingly, the resource provider agent contributes its resources cooperatively. Besides, concerning the resource provider agent, we also fix the probability of distributing satisfied resources to the decimal number of P_s ($P_s \in [0, 1]$), which is unknown to the resource client agent. After a fixed number of interactions, for example, N_{i1} , the resource provider agent determines to go from genuine to malicious (or from malicious to genuine). Thus, we fix the probability of distributing satisfied resources to $1 - P_s$ correspondingly.

Each round, the client agent employs the transaction feedback to estimate the trust of the resource provider, and the transaction feedback could be a success if a randomly generated number is smaller than P_s . Otherwise, interaction feedback is a failure. As a comparison, we compared to (Jøsang, 1998), since they map the evidence (r, s) to the trust represented by $(\frac{r}{r+s+1}, \frac{s}{r+s+1}, \frac{1}{r+s+1})$, in terms of the absolute error, here r and s indicate the number of success interactions and failed interactions respectively. We pay much attention to the estimated value of trust in the proposed method when performance is changed. Figure 5-1 shows the accumulation of absolute errors (five rounds starting with N_{i1}) due to the resource provider changing its performance.

As we can see, the accumulative absolute errors obtained by the proposed method are smaller than the $\frac{r}{r+s+1}$ in Figure 5-1, no matter 10 or 100 rounds are conducted. That is to say, the proposed method has a sharp perception of the sudden change of the resource providers. This improvement enables agents to manage varying performance in a dynamic multiagent system.

Nonetheless, some issues have to be stressed in this experiment. Firstly, we fixed the number of resource provider agents and resource client agents to one that also fits any other number of agents. We simplify the number of agents because we want to check the performance facing sudden changes. Secondly, the proposed method generally acts well when the probability of presenting high-quality services is high or low (0.7 and 0.9). However, if the resource provider agent gets custom to act averagely (for example, $P_s = 0.5$), the estimated trust by (Jøsang, 1998) can have a better performance because it seems no difference when the agent decides to go from malicious ($P_s = 0.5$) to genuine ($1 - P_s = 0.5$). As a comparison, we run the test 1000 times, and we set $P_s = 0.5$ and $N_{i1} = 10$. We say it is a win if the accumulated five-round absolute error by the proposed method is smaller than that of $\frac{r}{r+s+1}$ starting with N_{i1} . We received an average of 0.47 for winning the comparison, and the method in (Jøsang, 1998) wins by a tiny margin. From another perspective, the proposed direct compete with probability when agents act stably. However, the proposed method wins with 0.74 and 0.964 if $P_s = 0.7$ and $P_s = 0.9$ respectively. Nevertheless, rational agents tend to cooperate with the well-performed resource

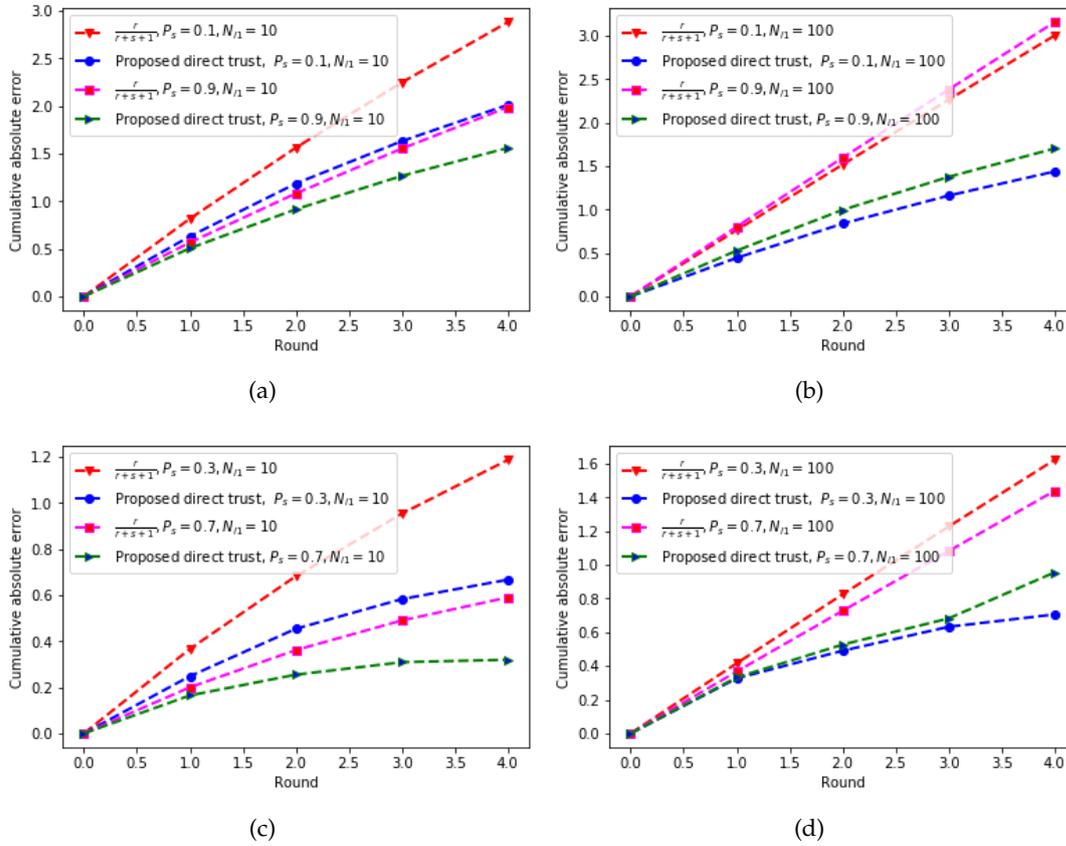


FIGURE 5-1: The recent five-round cumulative absolute error, obtained under different probabilities and changes performance in different interactive rounds: Figure. 5-1(a): 10 rounds and the probability equals 0.1 or 0.9; Figure. 5-1(b): 100 rounds and the probability equals 0.1 or 0.9; Figure. 5-1(c): 10 rounds and the probability equals 0.3 or 0.7; Figure. 5-1(d): 100 rounds and the probability equals 0.3 or 0.7;

provider agents rather than the averagely performed agents in practical life. Thus, this is why we focus on 0.1, 0.3, 0.7, and 0.9 to present satisfying resources in the test. In the next part, we see the first approach to estimate trust, namely combine evidence by the information amount conveyed by BPAs.

5.1.2 Model indirect trust by information amount VS. Beta reputation system

Of the existing computational trust models, the Beta reputation system (BRS) is widely applied in different fields and somehow similar to our trust method. It is a probabilistic model that relies on the number of successful and failed interactions

(Jøsang and Ismail, 2002). We compare the performance in terms of the relative frequency of successful interactions (RFS) (Yu et al., 2014). RFS, which evaluates the system based on how many interactions are executed to complete a target number of successful interactions. Specifically speaking, we exhibit the fraction of a fixed number of successful interactions over the total required amount of interactions. Here, it is necessary to state that a successful interaction for an agent corresponds to an interaction that selects it as a provider, and the latter has properly provided the service.

Simulation setting

Our simulation uses one resource client agent, s_{num} resource provider agent (set \hat{S}) and c_{num} third-party resource client agents (set \hat{C}). Each provider $s_j \in \hat{S}$ has a constant trust value of providing satisfactory resources, which is sampled uniformly from the values of 0.1, 0.2, ..., 0.9. This value indicates the probability of a successful outcome when interacting with this provider agent. For preprocessing, we let all the third-party clients interact with the resource providers so that they can achieve direct trust information about the providers. Five thousand interactions are conducted to assure the data to be near-accurate, and each of the 5000 interactions a provider agent $s_j \in \hat{S}$ is chosen randomly from the ten resource provider agents, and a third-party client agent c_q is randomly selected from the ten resource client agents. Each resource client agent c_q remembers, for each resource provider agent s_j , the interaction order, and the corresponding interactive feedback, which can be either a success or a failure.

Afterward, resource clients $c_q \in \hat{C}$ estimate their (direct) trust in any resource provider agents employing the previous information. Two different direct trust values, computed by the MITAI and the BRS are received respectively. In terms of the BRS, the client agent c_q models its trust in the provider agent s_j using the numbers of successful p_{qj} and failed n_{qj} interactions raised by c_q with s_j . Then the trust value of s_j from the perspective of c_q can be represented by $brs(p_{qj}, n_{qj}) = \frac{p_{qj}+1}{p_{qj}+n_{qj}+2}$ (Jøsang and Ismail, 2002). With regards to the MITAI, (3.17) is then adopted to generate the direct trust $m_{qj}^{t.d}$. An honest third-party client c_q will simply report the actual evaluation when asked for direct experience with respect to the provider agent s_j .

However, it must be emphasized that there are many deceptive agents in the multiagent system. In terms of the third-party resource client agent settings, a partly client agent first picks the resource provider agents to provide distorted recommendations. Each provider agent s_j could be picked with a probability of $p_{dh} = 50\%$. For all the other provider agents, c_q would always be honest. If a provider agent s_j is in the list of providing distorted information, c_q yields two random numbers

n_{q1} and n_{q2} ($n_{q1}, n_{q2} \in [0, 1]$) with which it reports the distorted recommendation $m_{qj}^{t,d}(T) = n_{q1} * n_{q2}$, $m_{qj}^{t,d}(nT) = (1 - n_{q1}) * n_{q2}$, and $m_{qj}^{t,d}(T, nT) = 1 - m(T) - m(nT)$. c_q also reports $n_{q1} * H_q$ and $(1 - n_{q1}) * H_q$ rounded to 1 instead of the actual amounts of successful and failed interactions, where H_q is the number of total interactions that c_q has raised.

Subsequently, the resource client agent c_i tries to interact with a resource client agent chosen randomly from the 10 resource provider agents. c_i starts with an empty set of direct trust, and then the experiment proceeds in rounds to reach a fixed number of successful interactions. Each round, we force c_i to assess the trustworthiness only by indirect reputation provided by c_q ($c_q \in \hat{C}$). When c_i collected all indirect reputation at the selection step, two resource client agents would be selected to conduct interactions based on the trustworthiness estimated by the MITAI and the BRS, respectively. Accordingly, we record the two selected client agents and the corresponding feedback.

With regard to demonstrating the effectiveness of the revised entropy-based weighted approach for multiple BPAs combination, we use $\frac{e^{E_d(m_{qj}^{t,ind})}}{\sum_{p=1}^L e^{E_d(m_{pj}^{t,ind})}}$ to replace (3.19) in our proposed MITAI. As (3.19) shows, we hold that great weights should be appointed to the small-entropy BPAs. However, (Yuan et al., 2016; Tang et al., 2017; Cui et al., 2019) employ $\frac{E_d(m_{qj}^{t,ind})}{\sum_{p=1}^L E_d(m_{pj}^{t,ind})}$, or $\frac{E_d(m_{qj}^{t,ind})}{\max(E_d(m_{pj}^{t,ind}))}$ for data fusion outside the system of multiagent. In other words, the weight assigned to BPA is proportional to the entropy value. The simulation terminates when the number of successful interactions reaches at 100 with the proposed method.

Simulation Results

Figure. 5-2 presents the RFSs obtained by the MITAI (blue cuboid) and the BRS (red cuboid) under $P\%$ $P \in [0, 50]$ dishonest agents, respectively. We ran 100 simulations and randomly select 10 times under 0 and 50% deceptive agents due to space constraints and similar results. As is shown in Figure. 5-2, RFSs are nearly the same when several deceptive agents are involved in the multiagent system. That is to say, the MITAI part competes with the BRS for trust estimation. However, we have overcome the limitations in the context of the Dempster Shafer theory-based trust estimation models as discussed in the last paragraph of section 3.4.2, from both generating BPAs and combing multiple BPAs aspects.

Figure. 5-3 exhibits the performances of the two entropy-based weighted approach for indirect reputation combination. We use the RFS value v_1 obtained by the proposed MITAI, minus the RFS value v_2 obtained by replacing (3.19) with $\frac{e^{E_d(m_{qj}^{t,ind})}}{\sum_{p=1}^L e^{E_d(m_{pj}^{t,ind})}}$

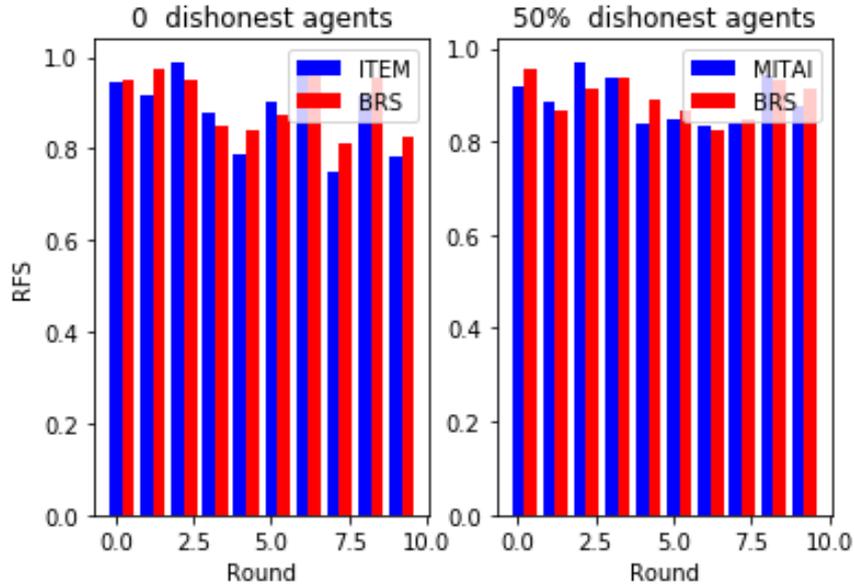


FIGURE 5-2: the RFS comparison between the MITAI and the BRS

in MITAI. Therefore, each round, we receive a $v_1 - v_2$, and the red line is hired to show the results. The blue line indicates the cumulative RFS differences. As displayed, we have $v_1 - v_2 > 0$ in each round, which means the proposed entropy-based weighted approach could always receive a better RFS value. Thus, great weights assigned to low-entropy BPAs for multiple BPAs combination is correct.

In summary, the MITAI approach is an ideal indirect trust estimation model, and it is as efficient as BRS. However, there are at least three advantages in MITAI. First, we considered the fading property when generating evidence by previous experience (Equation (3.17)), which has fulfilled the Dempster-Shafer theory of evidence-based trust system. This approach is extendable for dynamic trust estimation. Secondly, we performed an information amount based approach to combine multiple indirect reputations, that have rarely been emphasized before in multiagent systems for trust estimation. Lastly, we revised the entropy-based weighted approaches for BPAs combination. That is, high weights should be assigned to small-entropy BPAs.

5.1.3 Manage trust certainty with entropy

In the previous sections, we have discussed that trust is primarily on the basis of evidence consisting of positive and negative feedback, the amount of interaction, and the fluctuation (section 3.3.1). We utilize the satisfactory degree of S_{ij} to evaluate the performance in this thesis, which could also be interpreted as the numbers of

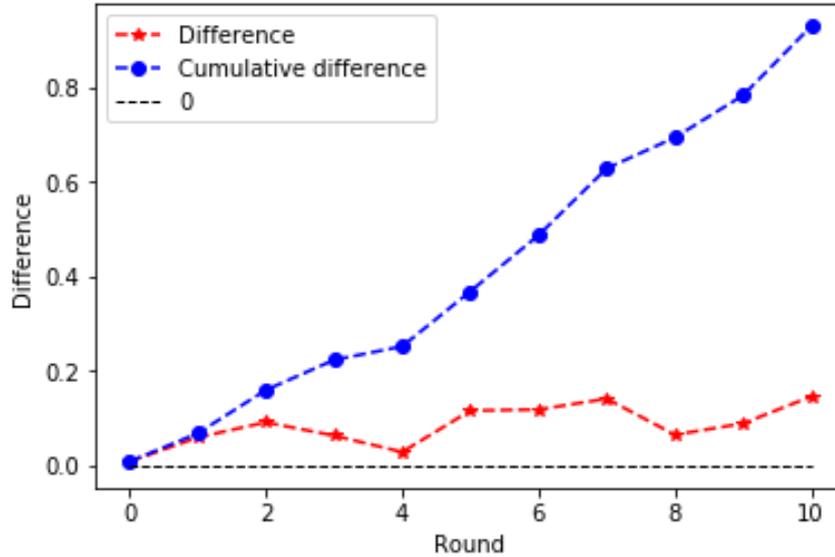


FIGURE 5-3: RFS comparison of A) our proposed entropy-based weighted approach MITAI (high weight assigned to low-entropy BPAs) with B) the analogous version of MITAI with high weight assigned to high-entropy BPAs

positive and negative feedback that are represented by r and s , respectively. Accordingly, we have $S_{ij} = \frac{r}{r+s}$, and then the related evaluation is generated by Equations 3.16 and 3.17. This section would mainly stress that the evidence certainty using the concept entropy of Dempster-Shafer theory of evidence.

Of the previous works, some literature assumes that certainty is independent of evidence (Basheer et al., 2015), and a third-party agent provides evidence as well as the relevant certainty towards the evaluation. In terms of the evidence (r, s) , (Jøsang, 1998) map this evidence to a trust represented by $(\frac{r}{r+s+1}, \frac{s}{r+s+1}, \frac{1}{r+s+1})$ and $\frac{1}{r+s+1}$ is designated as uncertainty. Yu and Singh used two parameters in the range of 0 and 1 to model all evidence (evaluations) to positive, negative, or neutral by the Dempster-Shafer theory of evidence (Yu and Singh, 2002b), and neutral is regarded as uncertainty. However, it fails to embody that conflicting evidence (the amount of positive or negative interaction) increasing uncertainty. Furthermore, the amount of interaction also has no impact on certainty. Later in (Wang and Singh, 2006; Wang and Singh, 2007), certainty is a statistical measure defined on a probability-certainty density function. However, this transformation from evidence to trust has neglected to implement an overall analysis of uncertainty caused by a fading factor, lacking information, incompleteness, and randomness. Besides, these works listed above are

on the condition that resource providers have stable performance. We transform the evidence to trust, providing an overall evaluation appropriate for dynamic agents' trust estimation, and the certainty of the evidence is well-studied thanks to the entropy definition, and we have to reveal that the certainty estimated with entropy also has the following general properties list in section 3.3.3, more details are expressed in the following parts.

Certainty rises with increasing experiences under fixed conflict

Assuming that the resource client agent a_i perceives evidence with a fixed satisfactory degree S_{ij} towards the resource provider agent a_j , we could ascertain how evidence certainty varies with an increasing amount of interaction. Logically, evidence certainty grows with the increasing amount of interaction. For example, compare 3 positive feedback out of 5 with 12 positive feedback out of 20, and we could recognize that the satisfactory degree S_{ij} is the same ($\frac{3}{5} = \frac{12}{20} = 0.6$) in both cases. Nevertheless, the certainty in the second case is intuitively greater than the first one. Definition 3.17 defines the generated BPAs, and Equation 3.23 is employed to calculate the certainty with the entropy of the Dempster-Shafter theory of evidence. We have certainty of 0.60 for (3,2), and as a comparison, certainty of 0.69 for (12,8). More in detail, Figure 5-4 indicates how certainty changes with an increasing amount of interaction (X-axis: from 0 to 20) and the fixed satisfactory $S_{ij} = 0.5$.

Certainty rises with increasing conflict under fixed experience

Under the condition of a fixed interaction amount ($r + s$), another characteristic of certainty, that certainty degree differs from the satisfactory, is also imperative. For instance, two agents have interacted ten times, and the feedback (10,0) is indeed more specific (certain or confident) than that of (5,5). Thus, certainty increases when satisfactory approaches 1 or approaches 0, and decreases when the satisfactory degree approaches 0.5. Figure. 5-5 displays the relevant result.

Model in detail, we consider four interactions are conducted between the resource client agent a_i and the resource provider agent a_j , and the number of positive feedback increases from 0 to 4 (the amount of negative feedback decreases from 4 to 0), Table 5-1 shows the certainty calculated by Yu and Singh (Yu and Singh, 2002b), Jøsangsang et al. (Jøsang, 1998), and Wang and Singh (Wang and Singh, 2006; Wang and Singh, 2007).

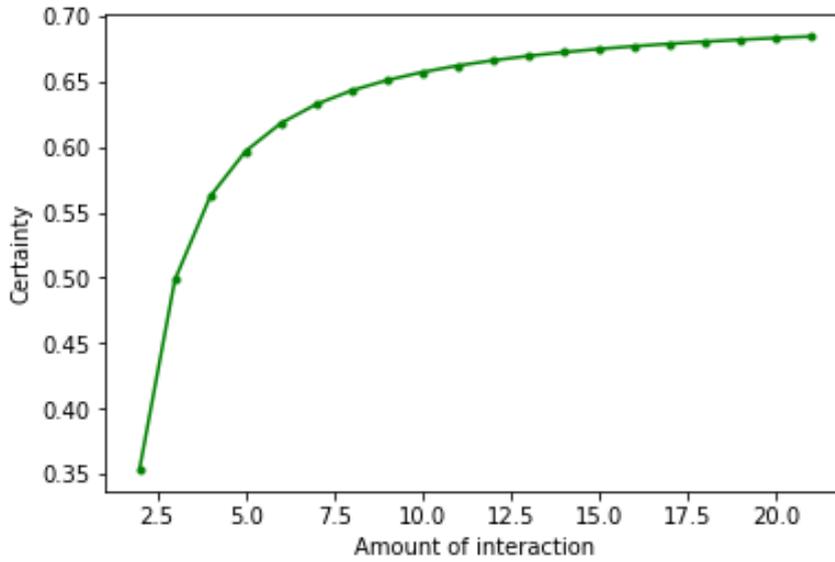


FIGURE 5-4: Evidence certainty increases with the amount of interaction ($r + s$) when satisfactory 0.5 is fixed. X-axis: Amount of interaction; Y-axis: Certainty degree

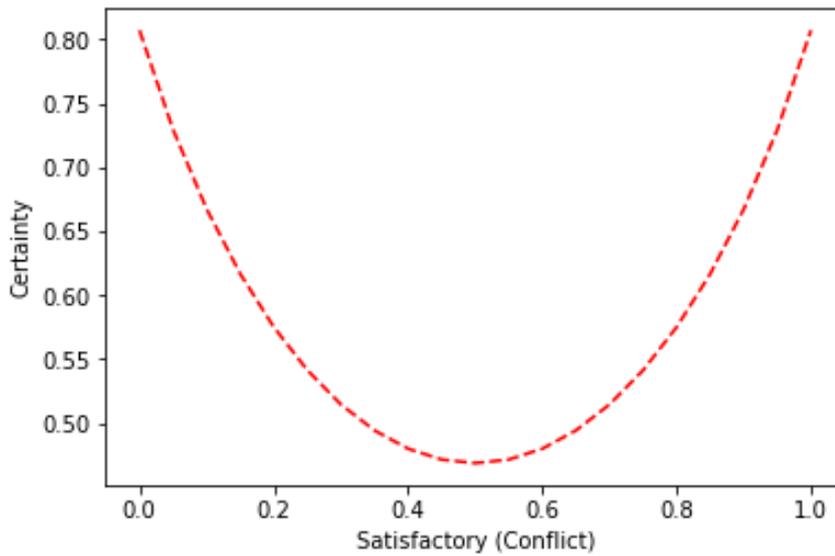


FIGURE 5-5: Evidence certainty varies with satisfactory when the amount of interaction 20 is fixed, and the satisfactory 0.5 leads to the lowest certainty.

TABLE 5-1: Certainty computed by differentiated approaches for different satisfactory with fixed amount of interaction 4.

	(4, 0)	(3, 1)	(2, 2)	(1, 3)	(0, 4)
(Yu and Singh, 2002b)	0	0	0	0	0
(Jøsang, 1998)	0.8	0.8	0.8	0.8	0.8
(Wang and Singh, 2006) (Wang and Singh, 2007)	0.54	0.35	0.29	0.35	0.54
Proposed entropy-based approach	0.69	0.62	0.59	0.62	0.69

Comparsion and discussion

This part is meant to discuss the trust certainty from the perspective of entropy of evidence. The previous literature study certainty from the observed trust, namely the amount of positive or negative feedback. For instance, (Wang and Singh, 2006; Wang and Singh, 2007) analyze certainty from the perspective of probability-certainty density function (Jøsang, 1998), and what follows, the estimated certainty is employed to generate evidence. That is to say, a piece of trust represented by belief, disbelief, and unknown should derive from the amount of positive and negative feedback, and the estimated uncertainty will be employed to obtain the trust of the unknown. From our perspective, we provide another perspective to make use of certainty; we firstly analyzed that multiple reasons would result in uncertainty, including lacking certainty, randomness, incompleteness, and varying performances. BPAs could represent this observation according to the Dempster-Shafer theory of evidence. However, how much we could rely on (or trust) the generated evidence when making decisions is a great challenge, and one of the significant aspects is the certainty of the evidence provider estimated through entropy.

Shannon entropy (Shannon, 1948) is a well-known information-theoretic measure of uncertainty that is based on a discrete probability distribution over a finite set of alternatives. However, entropy is said to be inappropriate for certainty estimation to the observed evidence represented by the binary event (r, s) (Wang and Singh, 2006; Wang and Singh, 2007). From one aspect, Shannon entropy's value ranges from 0 to ∞ , which does not fit our intuition. From another aspect, the confidence placed in a probability estimation is required. Lastly, the defined Shannon entropy cannot be employed to capture the probability estimation's uncertainty represented by binary events (positive and negative feedback). In this thesis, we used the entropy to estimate the certainty of the observed evidence represented by BPAs, and this certainty is an essential part of making full use of BPAs. Firstly, entropy models bits of missing information, which can also be interpreted as the information provided by the evidence, and more information conveyed by the evidence should

be more valuable (illustrated by method 1 in section 3.3.2). Secondly, the estimated entropy value is in the range of 0 to 1 that can be employed to exhibit the evidence's certainty. Last but most importantly, in our opinions, the uncertainty of a trust derived from the two events (r, s) consists of two main aspects: the conflict between each judgment (conflict between r and s) and the uncertain part of the judgment (assigned to uncertainty). Then the entropy definition (Definition 3.11) (Jiroušek and Shenoy, 2018) is designed to capture the certainty of BPAs from these two aspects that fit well to learn uncertainty. Furthermore, as exhibited in this section, the comparison results have proved the two essential qualities of certainty. That is, certainty rises with increasing experiences beyond a fixed conflict (section 5.1.3), and uncertainty rises with the increasing conflict under a fixed amount of interactions (section 5.1.3). Thus, the entropy is appropriate in measuring the uncertainty of the generated evidence by the Dempster-Shafer theory of evidence.

5.1.4 The overall trust model

In this section, a simulation test is designed to estimate the proposed trust model in distributed multiagent systems for manufacturing cloud selection. The test raises interactions between different cloud manufacturing agents, and each interaction would evaluate the different manufacturing cloud provider agents and filter with which provider agent to interact.

The simulation is assumed to have $Wnum$ cloud manufacturing client agents, and $Pnum$ cloud manufacturing provider agents. Once the interaction is launched, each client first interacts with each of the cloud manufacturing provider agents and collects and records the feedback three times. This information will later be used as evidence and be shared with other information clients. Simulation returns tell how to utilize our model, and the system can adjust to fluctuations in the information provider agents' behavior.

At the beginning of the simulation, all cloud manufacturing client agents are assumed to be honest. That is to say, once the other manufacturing cloud agents demand recommendations, an honest agent would contribute whatever it has to the demander to support it make correct decisions. However, after several interactions $Cn1$ (for example, 30 interactions), some of the agents (for example, the client agents a_1 and a_2) become malicious and become entirely dishonest to share information. In other words, they would present excellent evaluations of the bad-performance agents and vice versa. By using the proposed method, the malicious information provider agents are recognized, and their average credibility decreases gradually, which means that their contribution or recommendation will affect lesser

and lesser to the final decision. Although these malicious agents are always providing recommendations, a rational decision-maker would never trust or rely on the recommendations to make decisions. After a while (for example, at $Cn2$ rounds and $Cn2 = 70$), the malicious agents decide to become normal and contribute honest information again. When it decides to change its performances, its average credibility increases and their contributions are more influential. Simulation results and figures are discussed in detail in the following paragraph. Table 5-2 explains the detailed parameters in the simulation. As we can see, there are ten cloud manufacturing client agents, 25 manufacturing cloud provider agents. All agents act honestly in the first 30 round ($Cn1 = 30$). From round 30, some agents become dishonest and share complete incorrect evidence, and later they would become honest again ($Cn2 = 70$). Each round, a client agent might decide to arise interaction with a provider agent with a probability of 50%. One of the ten client agents is pointed as the evaluator each round, and seven agents ($Snum = 7$) are selected to share their information, and the evaluator might be included in the information provider agents. $Cnum$ best-evaluated manufacturing cloud provider agents are selected to carry interactions. The evaluation process is conducted 200 times, and each of the evaluation is managed for ten times. Since we find the same results and Figure 5-6 is used to show the results of the client agents' average credibility.

TABLE 5-2: Parameters in the simulation.

Parameters	Value	Explanation
$Wnum$	10	Number of client agents
$Pnum$	25	Number of provider agents
$Snum$	7	Number of selected agent to share information
$Cnum$	3	Number of selected providers to conduct interactions
$Cn1$	30	From when some agents become malicious
$Cn2$	70	from when the malicious return honest
Rounds	200	Total round

Figure 5-6 shows the average credibility for the cloud manufacturing client agents, and one of which, defined as a_1 , changes its performance from honest to malicious or the other way around, in round $Cn1$ and $Cn2$, respectively. The agent's credibility is updated at interaction 30 ($Cn1 = 30$), where it is redefined as a malicious information provider agent. As a result, we can recognize that its average credibility from other client agents' perspectives is degraded and, therefore, rational information demander agents decreases the confidences towards the provided information. Forty rounds later, that malicious agent rebecomes honest and restarts telling the truth. As

a result, its credibility is gradually rebuilt, the other agents begin to trust its recommendations. However, we also find that the average credibility decreases fast (takes around 40 rounds from 0.75 decreases to 0.15) and increases very slow (takes around 130 rounds from 0.15 decreases to 0.7), reflecting the fact that trust is easy to lose and hard to achieve.

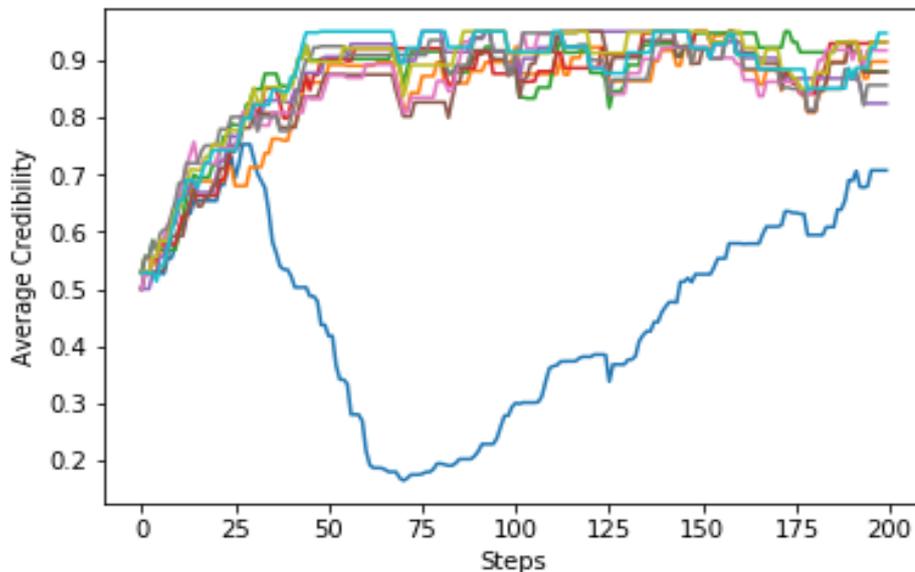


FIGURE 5-6: Change in average credibility for each information sharing agent.

More in detail, we can find that only one agent out of 10 information provider agents has changed its performance in round 30; that is to say, most agents also provide correct information to facilitate making the right decisions. As explained, some trust estimation models assume that most information provider agents are honest and present correct information (Jøsang and Ismail, 2002). As a result, in round 30, we force five honest agents to become malicious and begin sharing incorrect information, Figure 5-7 shows the result. As is shown, the average credibility of the five agents all decreased from round 30 and increased at round 70. That is to say, a rational evaluator can distinguish honest information providers, and if they keep telling an untruth, the information provider agents would no longer be trustworthy. However, if they give correct information, their provide information would become valuable again.

As a comparison, in terms of applying the Dempster-Shafer theory of evidence to trust estimation in multiagent systems, (Yu and Singh, 2003; Yu, Kallurkar, and Flo, 2008) have taken personal credibility into consideration. However, in (Yu and

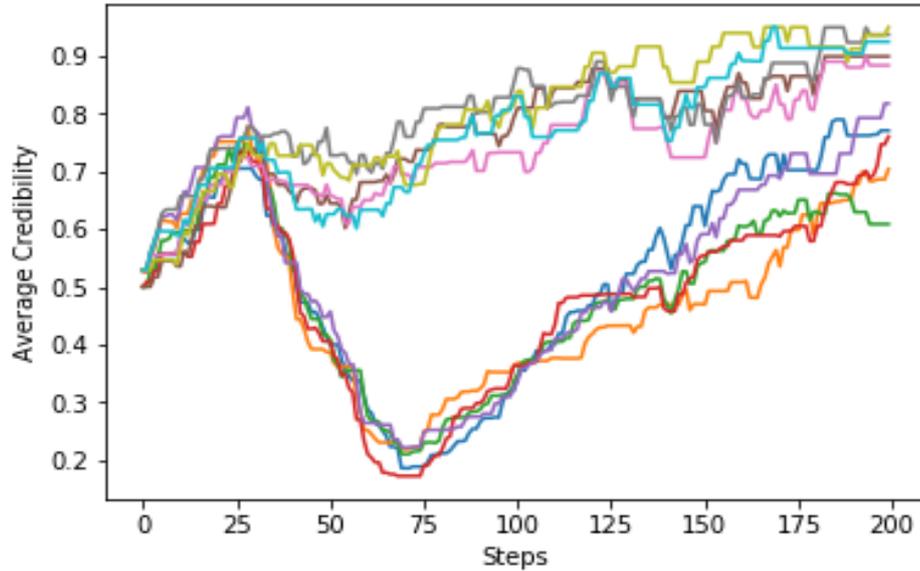


FIGURE 5-7: Change in average credibility for each information sharing agent when five agents change their performance.

Singh, 2003), we do not discuss if the BPAs generation is appropriate, especially in a dynamic and open multiagent systems, we would focus on how the weight of the third-party agent's is updated. As explicated, the weight is updated by $w'_i = \theta * w_i$, where w_i is the agent's weight and $\theta < 1$. Thus, the credibility of third-party agents decreases gradually. As a result, it seems inappropriate in an open and dynamic system. In (Yu, Kallurkar, and Flo, 2008), the agents' weight is defined increasing gradually to one if positive feedback is received; on the contrary, the weight decreases linearly for negative. Even evidence consistency, as well as evidence certainty, are not considered in this literature, we also simulate in the same testbed as explained in Table 5-2. Figure 5-8 indicates the average credibility of each information provider from the other perspective. As we can see, even the agent a_1 choose to present dishonest information from round 30 to round 70, its average credibility also increased. That is to say, no evaluator has recognized this cheater. The main reason for receiving increasing average credibility is that most honest agents share correct information that has helped the decision-maker select the well-evaluated cloud manufacturing provider agent. As a comparison, Figure 5-9 shows the average credibility change-ment when five agents are set to change their performance in round 30 and round 70.

As we can see in Figure 5-9, all agents' average credibility decreased from round

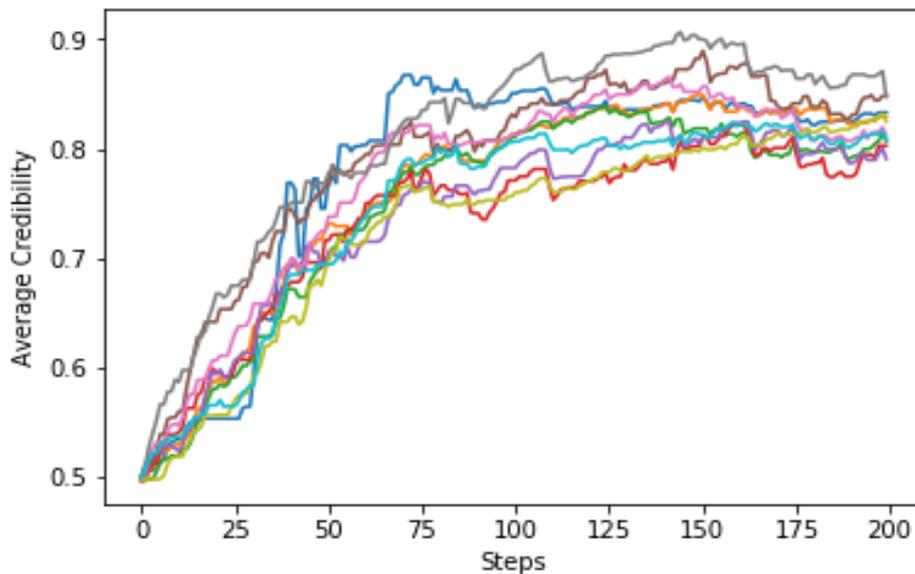


FIGURE 5-8: Change in average credibility for each information sharing agent when one agent changes its performance, the change process is as defined in (Yu, Kallurkar, and Flo, 2008).

30 to round 70. That is to say, in this period, the evaluator can hardly select well-performance cloud provider agents and it can also hardly recognize the dishonest information provider agents.

5.1.5 Limitation and discussions

In this section, we have expressed some significant characters of the proposed method in chapter 3. Some of the beneficial features of the proposed method are discussed as follows,

1. The model has analyzed uncertainty caused by many factors, including insufficient evidence, randomness, incompleteness, and so on.
2. The proposed model uses both direct trust and indirect reputation to achieve an overall evaluation;
3. The proposed model can cope with the witness' unstable behaviors in the system.
4. The proposed method has considered evidence consistency to select reliable evidence.

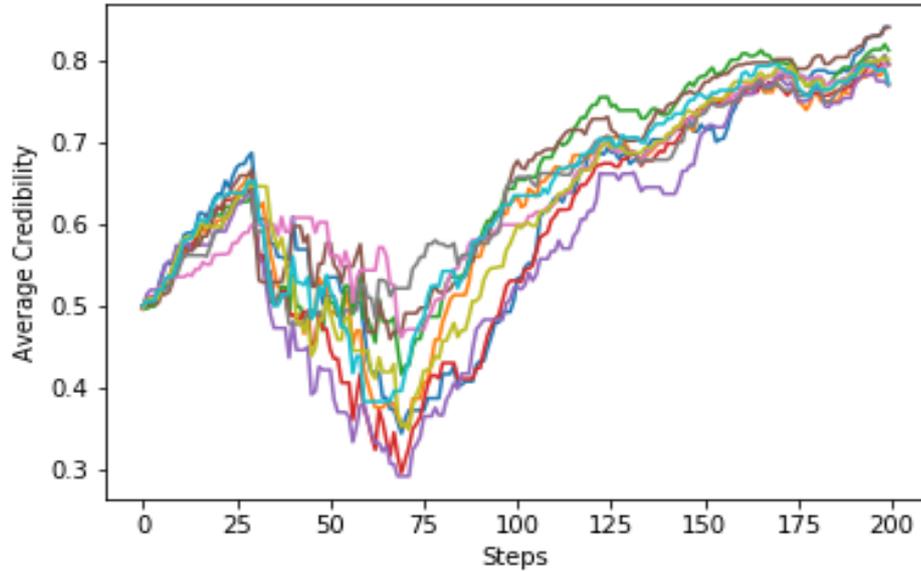


FIGURE 5-9: Change in average credibility for each information sharing agent when five agent change their performances, the change process is as defined in (Yu, Kallurkar, and Flo, 2008).

5. The proposed method has considered the witness' credibility to select reliable evidence.
6. The proposed method has considered the evidence' certainty from the perspective of entropy to select reliable evidence.
7. The proposed method relies on the good-quality and high-certainty evidence provided by trusted client agents.
8. The proposed method updates personal credibility from the perspective of interactive feedback and the quality of the provided information.
9. All witnesses have the right to provide testimony, but the evaluator would trust the trusted clients more and underrate the untrusted evidence.

Besides these positive perspectives, the proposed method is also limited to some assumptions. The limitations and assumptions are discussed as follows. Firstly, we assume that each of the witnesses has at least one interaction with each resource provider, and this information would be shared with the other agents to help make suitable decisions. That is to say, trust is on the basis of evidence. There will be no trust without any evidence (interactions). Besides, we also assume that some of the agents would present real experiences. Finally, when we redefine an honest agent

to be dishonest in some round, we assume that it changes its performance towards all other agents. From one aspect, trust can be interpreted as the will to present the required information, thus it is acceptable that an agent acts honestly, then they are honest with all. From another perspective, personal emotion does exist in the practical life for trust estimation. For example, the agent a_1 does not like the agent a_2 , but it likes the agent a_3 . Thus, agents are malicious, and they act maliciously with all users seems incorrect.

5.2 Random walk process for service selection

In this section, we present the simulation of Chapter 4 to illustrate the efficiency of the proposed model for cloud manufacturing agent selection. As discussed in Chapter 4, tasks can be divided into multiple subtasks, and each of the subtasks can be transformed into a hash value thanks to the Simhash approach. When demanding similar cloud manufacturing, interaction feedback based trust value is employed to estimate a trusted resource provider agent or a trusted recommender agent. So firstly, we have to discuss the Simhash approach is appropriate for privacy-preserving and similarity qualification.

5.2.1 A test of Simhash

Our principal goal is to discuss how the Simhash approach is applied to similarity measurement in this subsection. Simhash is employed for natural language processing, and it is on the foundation of text segmentation. Natural language processing (Bird, Klein, and Loper, 2009) is a suite of open-source Python modules, data sets, and tutorials supporting research and development that can be employed to segment a text to keywords and the corresponding probability or can also be interpreted as the related weight. Correspondingly, Jieba is a Chinese text segmentation, which can return weights of the keywords. Of course, these modules are scalable for other language segmentation. More importantly, the developers are capable of designing their own dictionary and importance. That is to say, in the cloud manufacturing systems, the cloud platform can design a dictionary of all manufacturing fields using the big data obtained from manufacturing client and client agents. For instance, the term food manufacturers cover a series of industrial activities directed at the processing, conversion, preparation, preservation, and packaging of foodstuffs. Thus, these words food, processing, conversion, preparation, preservation, and packaging are frequently employed for food manufacturing and therefore are added to the dictionary, and their corresponding weights can be obtained from the frequency or

appearance probability. However, our principal focus is the agents' trust in this thesis, and we paid less attention to the natural language process. Consequently, we adopt the module of Jieba, as an example, to explain the similarity estimation in this part.

A manufacturing client agent has a requirement of primary food processing, namely, turns the agricultural products, including wheat kernels and paddy, into something that eventually eatable. Thus, two subtasks described as

1. *I need to process wheat kernels, or wheat kernels processing.*
2. *I need to process paddy or paddy processing.*

This requirement is divided into two subtasks, and two Hash values are broadcasted to discover potential similar resource provider agents. According to the module Jieba and the default dictionary, we find influential keywords and the related weights as follows,

- (*'wheat'*, 3.4156), (*'kernels'*, 3.4156), (*'need'*, 1.7078), (*'process'*, 1.7078), (*'processing'*, 1.7078)
- (*'paddy'*, 4.7819), (*'need'*, 2.3910), (*'process'*, 2.3910), (*'processing'*, 2.3910)

Thus, according to Simhash, two hash values are widely separated. We suppose that one of the other agents have had some experience, and some are discussed as

- *I process wheat kernels, for wheat kernels processing.* and
- *I need to process paddy, to process paddy.*

Thus, according to the same hash function and Equation 4.10, each subtask is compared to the subtasks in the agents' previous experiences to achieve the minimum distances. Two distances, 0.125 and 0.140625, are received in this example. As a result, the similarity of the tasks and the to-be-recommended resources is calculated by Equation 4.10,

$$Sim = 1 - \max(0.125, 0.140625) = 0.859375$$

As a result, in terms of the divisible tasks that cannot be separated processing, we use the Simhash approach to estimate the similarity between the requirements and the to-be-recommended resources for privacy-preserving in a distributed environment. However, this approach relies enormously on the default dictionary as well as task descriptions. Thus these aspects need to be stressed in the next step. Since our primary focus is trust estimation in distributed multiagent systems, we

would prove the proposed trust-based model for manufacturing recommendation is correct and efficient. In the following simulation setting, the similarity between two requirements could only be 1 or 0, namely, completely the same or completely different.

5.2.2 Trust-based cloud manufacturing recommendation

In this part, we implement the proposed trust-based recommender system to prove its efficiency for the cloud manufacturing selection in a distributed environment, some preset parameters displayed in Table 5-3 for the simulation are discussed as follows.

This part presents the background of the simulation. Practically, all kinds of cloud manufacturing can be involved, ranging from transportation manufacturer, chemical industry, paper industry to furniture, toys, and so forth. In this simulation part, we rely solely on food manufacturing service selection. Each service provider has technical skills because of uneven education, experiences, and willingness regarding time, cost, safety, and reliability. For example, a resource provider agent always has quick responses to any requests, but the cost is extremely unacceptable. Thus, a manufacturing client agent who values time much would interact with this provider agent. In the simulation part, each resource provider agent is configured beforehand with random probability from these four aspects, which is unknown to the public. A higher probability means to provide more satisfactory services. Public resource client agents select service provider agents by other clients agent' recommendations, where the recommendations can be the value of trust of previously experienced provider agents or just trusted client agents.

In the simulation setting, N_a agents are employed, each of which, denoted by a_i , maintains different manufacturing resources that can be shared. Similarly, they regularly generate requirements for different resources. Of these requirements, they attach importance to n_p factors regarding the four manufacturing selection factors, including time, cost, security and quality. For instance, the agent a_i considers time and cost to be the most important, and then heavier weight would be assigned to these two factors. Besides, ($p\%$) of these agents are appointed to be dishonest when it acts as an information provider. In the random walk process, the preset probabilities are generated in the range of 0.1 to 0.9. When one of the agents needs a particular requirement, the proposed method is employed to filter the best resource provider by investigating recommendations. Each random walk is repeated φ times starts from each of the prospective service providers. In the PageRank algorithm, the PageRank value of a service provider is calculated by the total number (recommendation) that visits it divides the total number of visits during all random walks. In

this method, we use the target agent's total recommendations that received divides the overall recommendations during all random walks. We also employ the roulette wheel algorithm to ensure that all agents are contributable and get rid of always selecting the top-ranking agent. We ran a set of different tests to exhibit that our proposed method operates correctly and adequately. Due to the randomness, we run each test N_{tt} times, and the average results are used for the comparison. Firstly, we run a test to show that considering trust in the recommender system is essential. We compare the satisfactory rate generated by the proposed method to randomly selecting a resource provider. Table 5-3 explains the detailed variable in this simulation, and Figure 5-10 shows the results.

TABLE 5-3: Parameters in the simulation.

Parameters	Value	Explanation
N_a	50	Number of agents
a_i		An agent
$p\%$	20%	Percentage of dishonest agents
φ	5	Number of each random walk
N_{tt}	20	Total time each round
Rounds	100	Total round

As can be seen in Figure 5-10, all the obtained cumulative satisfaction represented by the blue lines is around 0.58, no matter one, two, three, or four factors are employed to describe a preference. As a comparison, the red lines in Figure 5-10 indicates the average ability. That is to say, satisfaction should be the same as the red lines if randomly selecting a resource provider agent. More in detail, in each round, each agent obtains some manufacturing resources, which are capable of implementing various tasks in terms of time, cost, safety, and quality. What follows, in each of the N_{tt} round, one agent is appointed as a resource client agent, and it has different preferences. Then the proposed method is used to select a resource provider to raise interaction, and in this way, we could receive a satisfactory degree of this interaction. Since its randomness, the average satisfaction of the N_{tt} rounds would be regarded as an estimated result for further comparisons. It has to be emphasized that if only one out of four-factor is considered for preference representation in the random walk process, the recommendation mechanism degenerates to a trust-based recommendation approach. That is to say, an information recommender has completely the same preference, and then a recommendation, namely a trust value, would be presented, and the resource client agent makes final decisions according to the received trust. In summary, in the simulation conducted by 50 agents, and 20% of which present

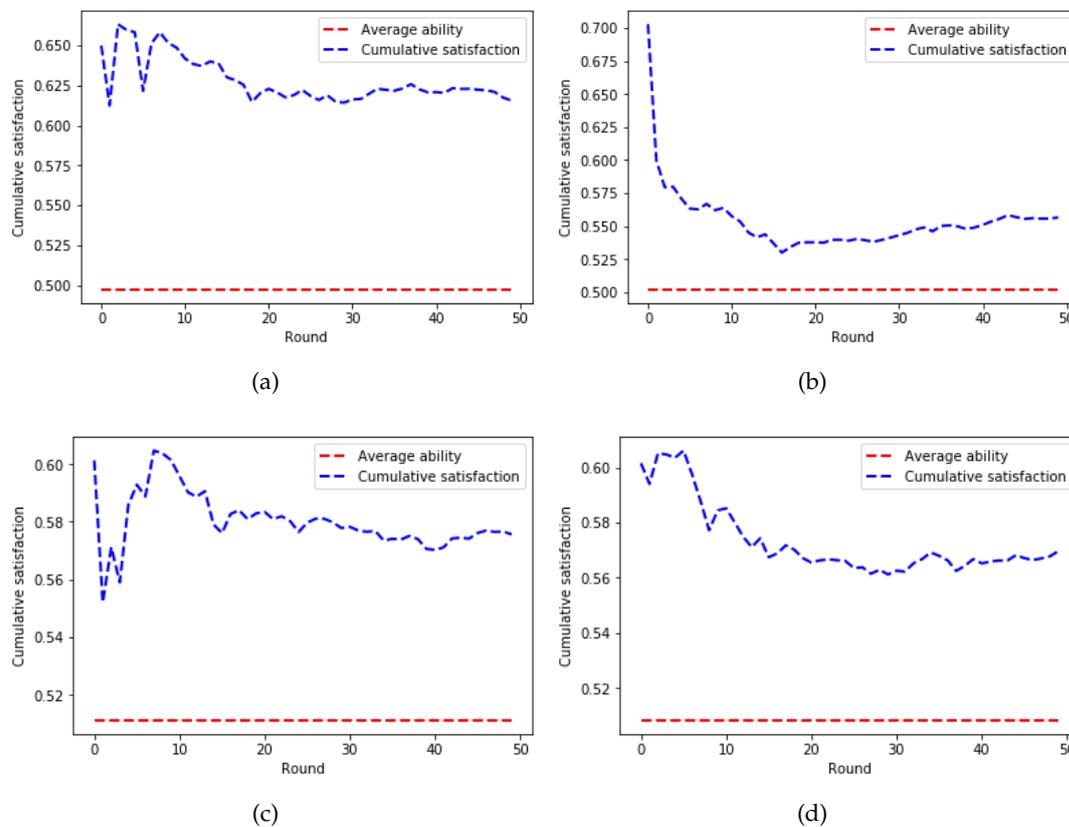


FIGURE 5-10: The effect of the proposed trust-based recommendation: It is obtained by the satisfactory degree of the proposed method minus the satisfactory degree obtained by randomly selection. Figure. 5-10(a): one factor is considered to represent the preference; Figure. 5-10(b) : two factors is considered to represent the preference; Figure. 5-10(c) : three factors is considered to represent the preference. Figure. 5-10(d): four factors are considered to represent the preference.

misleading information, from one perspective, the average cumulative satisfaction is better than that of randomly selecting a manufacturing resource provider, and the proposed method can help achieve correct and better resources. From another perspective, no matter how a task preference is represented, similar client agents' recommendations should be reliable.

The previous simulation is conducted under the condition that 20% of agents are dishonest. Figure 5-11 displays the results of 100%, 40%, 60% and 0 honest agents. Under these conditions, a dishonest agent randomly recommend a similar or a trusted client agent, and Figure 5-11 shows the result.

As we can see from Figure 5-11, the proposed method can achieve correct and

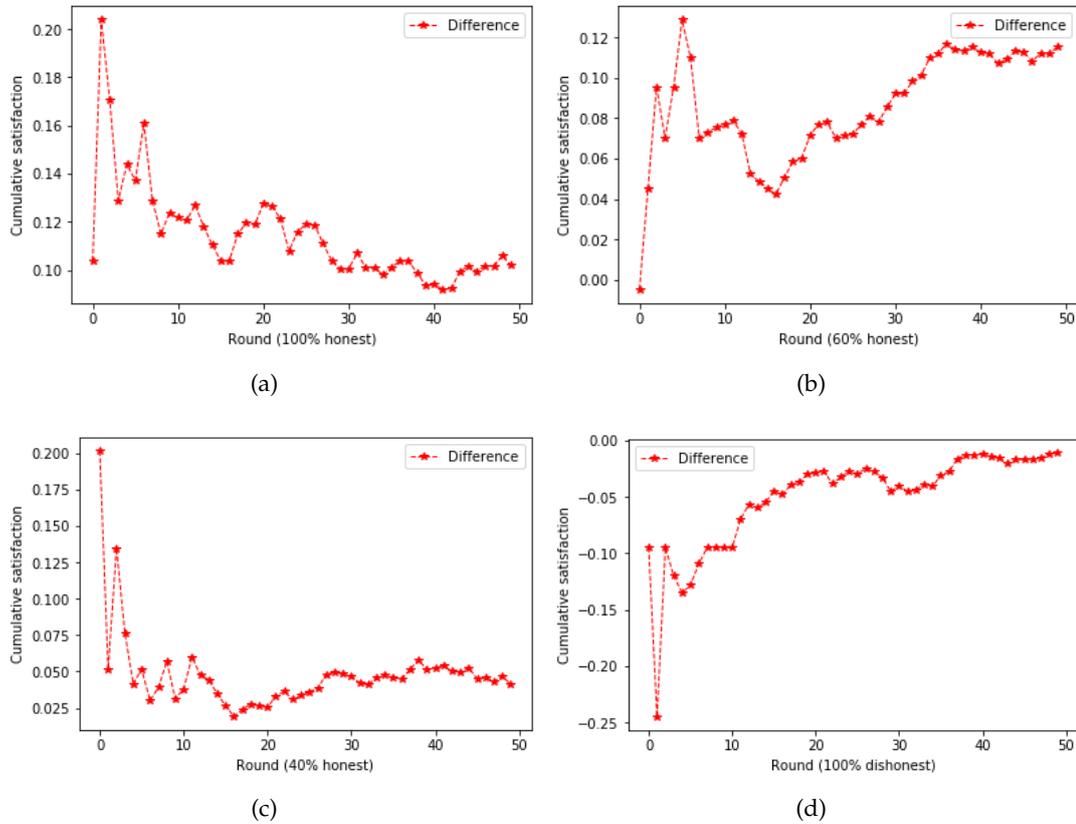


FIGURE 5-11: The proposed trust-based recommendation method's effect under different percentages of dishonest agents: We use the proposed method's accumulative satisfaction minus the satisfactory degree obtained by randomly selecting a potential provider. Figure. 5-11(a): All agents are honest; Figure. 5-11(b): 60% of the agents are honest; Figure. 5-11(c): 40% of the agents are honest; Figure. 5-11(d): All agents are dishonest;

better resources when honest agents are involved. In these simulations, a dishonest agent randomly recommends a target and a value of trust. We could also find that the difference decreases from 100% honest agents (Figure 5-11(a)) to 40% agents (Figure 5-11(b)), and when all agents are dishonest, the difference in Figure 5-11(d) approaches 0, which indicates that it is nearly the same as randomly selecting.

It has to be emphasized that the proposed trust-based resource recommendation has also intensely relied on similarities. As explained, our approach holds that recommendations provided by trusted and similar client agents are reliable. In this thesis, two similarities are discussed. The first one is based on the Simhash approach to capture task similarity from a privacy-preservation perspective. Since a manufacturing dictionary still needs further investigation, we employ the Hash value of the manufacturing keywords in this simulation, which means that two manufacturing

tasks could be neither complete the same or not, namely, Sim_{task} in Equation 4.16 is either one or not. An agent regards entirely the same tasks ($Sim_{task} = 1$) as similar and would be considered for further recommendations. Another similarity is about the preference similarity discussed in the Equation 4.11, we have to check whether the similarity is essential for recommendations, and Figure 5-12 shows the result.

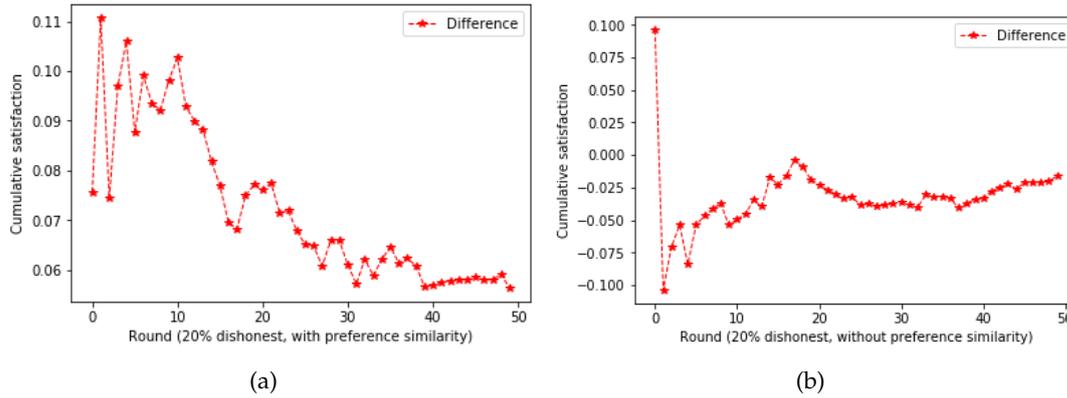


FIGURE 5-12: The effect of preference similarity in the proposed trust-based method: We use the proposed method's satisfactory degree minus the satisfactory degree obtained by randomly selecting a potential provider. Figure. 5-12(a): preference similarity is considered and 20% of agents perform misleading information; Figure. 5-12(b): preference similarity is not included, and high-trust resource provider agents are recommended in the random walk, and 20% of agents present misleading information.

From Figure 5-12, we could find that task similarity, especially the preference similarity, plays a remarkable role in the recommendation. In Figure 5-12(a), agents can receive correct and satisfied resources if both preference similarity and trust are considered. As a comparison, In Figure 5-12(b), the satisfaction approaches 0, which is the same as a random selection if preference similarity is not included. In summary, both task and preference similarity and trust are required for a resource recommendation.

All the previous simulations are conducted with 50 agents and ten manufacturing resources. We also run a test to see the influence of the number of agents. Figure 5-13 displays the results. As is shown, we do not find a noticeable difference between 10, 50 and 500 agents adopted in the simulations. However, it has to point out that the required time increases rapidly with the increasing number of agents, where seconds is required for 10 agents, but nearly 20 minutes for 500 agents. The main reason for the increasing time is that more prospective agents are received. Thus more random walks have to be conducted to find the final results.

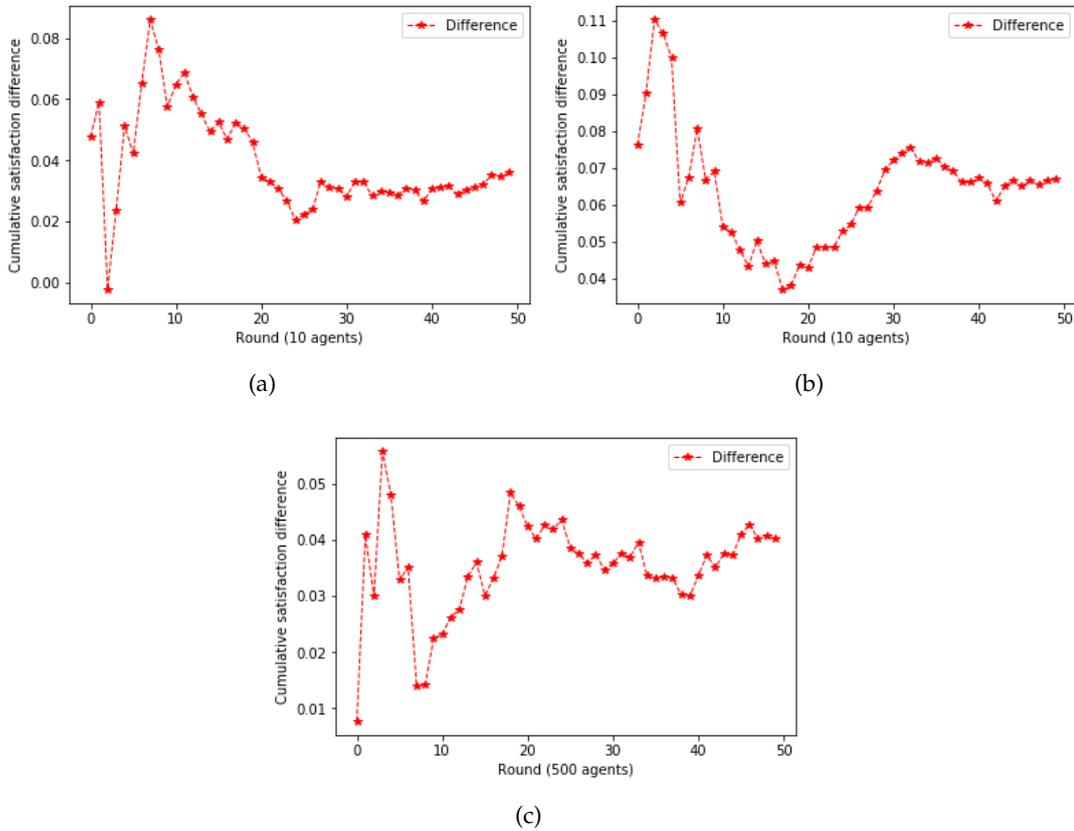


FIGURE 5-13: The impact of the number of agents in the proposed trust-based approach: We use the proposed method's satisfactory degree minus the satisfactory degree obtained by randomly selecting a potential provider. Figure. 5-13(a): The simulation is conducted with 10 agents; Figure. 5-13(b): The simulation is conducted with 50 agents; Figure. 5-13(c): The simulation is conducted with 500 agents;

In this random walk-based recommendation approach for service selection in a distributed environment, the preset probability is defined as continuing walking from the present agent. That is to say, the present agent can choose to terminate the walk (recommendation) with this preset probability. Higher probability means more recommendation and the other way around. We do not desire too many recommendations because of the enormous number of calculations; meanwhile, we hope sufficient recommendations could be received. As a result, we use 0.5 in the previous simulations. Similarly, the number of each random walk is employed to decide how many walks should be conducted from each prospective agent, and we set it 0.5 in the previous simulations. These two factors do not impact the final results but significantly influence the simulation time. Thus, further discussion of these factors is not discussed here.

5.3 Conclusion

We have presented the simulation results in this chapter, mainly from two aspects. From the first perspective, the proposed Dempster-Shafter theory of evidence-based trust model is proved to be efficient for selecting trust client agents. We first checked the proposed direct trust estimation approach is correct and effective, especially in dynamic multiagent systems where agents might be deceptive and go from honest to dishonest to hidden their actual performance. To model indirect reputation, we then compare the improved trust model to the beta reputation system in terms of the relative frequency of successful interactions, proving that the improved trust model completes the Beta reputation system. What more, we concluded the great weights should be assigned to the low-entropy BPAs inside the system of trust estimation is correct compared to some references discussed which assign a high weight to high-entropy BPAs outside the system of multiagents. The overall trust model is then conducted for resource provider agent selection, emphasizing that entropy-based certainty and the improved credibility assignment are correct and rational. Later we simulated the random walk-based trust model to manufacturing agents selection, we checked the influences of preference length, similarity, the number of agents, and the percentage of dishonest agents in the simulations. We could find that trust and similarity-based resources can help find correct and well-valued manufacturing resources.

Conclusion and future works

This chapter re-discussed the main issues and presents the conclusions of this thesis. We summarize our contributions through the results we learned, followed by describing the limitations of this thesis by presenting some interesting future works.

Context

Resource sharing problems last and have been discussed for a long time. Inside the manufacturing industry, multiple manufacturing resources are shared to fulfill requirements to improve efficiency, reduce cost, etc. In Industry 4.0, cloud manufacturing was first introduced, usually by means of centralized techniques, which is the mainstream of the present manufacturing sharing. However, because of the manufacturing particularities-producers and demanders have to keep in touch for manufacturing details, the centralized techniques are not well-adapted to various applications, and the distributed manufacturing cloud will be the future answers. Multiagent systems are often employed to solve difficult problems. However, the agents' trust has a notable impact on efficiency and has been studied for decades. In this thesis, we concentrate on the concept of trust for agent-based manufacturing cloud selection in a distributed environment.

The objective of this thesis is to design and improve trust models in a distributed and dynamic mechanism. Generally, trust is based on evidence, including direct experience, third-party agents' recommendations, social-cognitive trust, and organizational trust. For this purpose, we rely on the first two cases for trust estimation and also acknowledge that some criteria are necessary for a dynamic trust model: Uncertainty, evidence certainty, consistency, and credibility. Besides, to capture the agents' trust is not only our purpose, we also need to motivate agents to become cooperative in multiagent systems. Additionally, trust and privacy-preserving are emphasized for a cloud manufacturing selection in a distributed environment.

Contributions

We have designed and improved two trust models for agents' trust estimation in a multiagent system in order to identify with which agents to raise interaction. One is on the foundation of Dempster-Shafter theory, and another is based on the random walk approach.

We have overcome the limitations and improved it from the following aspects in the adaptive Dempster-Shafter theory of evidence-based trust estimation model. As stated that trust is based on evidence, and agents' performance can vary over time. Thus, evidence uncertainty can significantly influence decision making. This uncertainty comes from various aspects, ranging from memorization fading over time to uncertainty, randomness, and incomplete information. Thus, we obtain a satisfactory degree, which decides the value of trust thanks to a fading factor influenced by the previous evaluation and current performance. Besides, we generate the basic probability assignment, which estimates the value of trust in the adaptive model, uncertainty caused by these stated factors is appointed to unknown. Since we use direct trust and indirect reputation to achieve the final trust, combining the recommended indirect reputation is also a challenge. Of the previous works, some works use the Dempster' combination rule directly, some assigned weights to basic probability assignment. However, these aspects are inefficient to achieve an overall adaptive trust model, we improved the trust combination from the following aspects: Firstly, we not only assigning uncertainty when achieving a direct trust but also we have to make full use of this uncertainty. The assigned uncertainty can be interpreted as a certainty of the recommended evidence. Thus, the entropy of Dempster-Shafter (Deng, 2016; Jiroušek and Shenoy, 2018) is employed to study the evidence certainty. The Deng entropy (Deng, 2016) is adopted to emphasize that assigned weight should inversely proportional to the entropy inside trust models. The new entropy (Jiroušek and Shenoy, 2018) is used to state the certainty of the evidence. Simulation results indicate that capturing certainty (of the generated evidence) with the new entropy satisfies two main certainty features. This also has broken the assertion that entropy is inappropriate for certainty estimation in (Wang and Singh, 2006; Wang and Singh, 2007). Secondly, we use evidence distance (similarity) to decide evidence quality in case of the resource providers' sudden changes. This evidence consistency is also used to determine the witness's credibility. Thirdly, we modify the weight of the witness assigned in (Yu and Singh, 2003; Yu, Kallurkar, and Flo, 2008). Compared to decreasing the assigned weight or decreasing when negative feedback is received, and the other way around, our approach, reward to correct recommendation, and punishment for incorrect recommendation proved to

be convincing. What is more, we encourage agents to be trustworthy in the proposed method by a credit mechanism, and unlike the "second chance" in (Shehada et al., 2018), agents' credit increases gradually when correct recommendations are received. With the improved trust model, we adopt a reservation approach to help agents find excellent resource providers.

In the random walk based approach, we capture the direct trust in the same way with the improved adaptive Dempster-Shafter theory of evidence, aims at stressing the uncertainty caused by those factors. In the recommendation mechanism, we proposed that recommendations from trusted and similar friends are adoptable. As a result, finding trusted and similar friends are quite necessary. Besides, privacy-preserving is an essential aspect of resource sharing in a distributed environment. Therefore, we use the Simhash and preference to find perspective resource providers. In the random walk process, we follow **MC complete path** algorithm proposed in (Avrachenkov et al., 2007), and it has an advantage and reasonable to terminate the random walk once reaching an agent who has neither similar experience nor trusted friends. Finally, the resource client will try to negotiate and cooperate with the most recommended resource providers. Results indicate that the resource clients are satisfied with the selected resource provider and the related resources with the proposed trust-based random walk approach.

With these trust models, we can now achieve adaptive trust values in multiagent systems, no matter the agents have varying performances or the third-party agents providing misleading information. Besides, agents are motivated to be trustworthy. Otherwise, they have problems demanding recommendations for trust estimation. What is more, privacy is stated in a distributed environment, enabling the cloud manufacturing systems to be more applicable and scalable in more applications.

Limitations and future works

In future works, we propose to study more scalable trust models with various evidence. In this thesis, we learn the value of trust by investigating interactive feedback, namely, direct experience and indirect reputation recommendation by the others. In other words, the trust between agents is carried out under the condition that they interact at least once. However, various evidence (factors), including social cognitive, organizational experience, even personal emotion, and social roles, significantly impact trust estimation. In addition, as the amount of available information related to agent performance continues increasing, we can learn from these big data to discover relationships and measure trust.

We used to assume that agents have consistent performance towards all the other agents. Actually, agents might be honest to a part of agents while acting horribly to the others. That is to say, trust estimation needs to be considered from agents to agents and situations to situations. This aspect has to be greatly improved for future trust estimation, and the improvement seems to be of interest to us.

Of the previous trust estimation model, especially in the field of cloud manufacturing systems for platform trust estimation or provider evaluation. They provide numerical examples to indicate its efficiency, including the proposed simulation models in this thesis. However, these approaches have limitations or not enough for practical applications. In the future, more manufacturing resources sharing in the cloud have to be stated, and more practical examples have to be employed to prove the efficiency of the trust estimation.

In this thesis, we use a Simhash approach to discover similar prospective friends in manufacturing resource sharing. The proposed method relies highly on the task (subtask) or preference descriptions. This approach is applicable, but not scalable. As a result, we first have to establish a comprehensive keywords dictionary in the field of manufacturing. More investigation related to natural language processing has to be explored for similar friend discovery in the future.

Reference

- Abdul-Rahman, Alfarez and Stephen Hailes (2000). "Supporting trust in virtual communities". In: *Proceedings of the 33rd annual Hawaii international conference on system sciences*. IEEE, 9–pp.
- Ahadipour, Ava and Martin Schanzenbach (2017). "A survey on authorization in distributed systems: Information storage, data retrieval and trust evaluation". In: *2017 IEEE Trustcom/BigDataSE/ICSS*. IEEE, pp. 1016–1023.
- Anagnostopoulos, Theodoros et al. (2020). "A multi-agent system for distributed smartphone sensing cycling in smart cities". In: *Journal of Systems and Information Technology*.
- Andersen, Reid et al. (2008). "Trust-based recommendation systems: an axiomatic approach". In: *Proceedings of the 17th international conference on World Wide Web*. ACM, pp. 199–208.
- Anisi, Mohammad and Morteza Analoui (2011). "Multinomial agents trust modeling using entropy of the dirichlet distribution". In: *International Journal of Artificial Intelligence & Applications (IJAA)* 2.3, pp. 1–11.
- Arbib, Claudio and Fabrizio Rossi (2000). "Optimal resource assignment through negotiation in a multi-agent manufacturing system". In: *Iie Transactions* 32.10, pp. 963–974.
- Argandoña, Antonio (1999). "Sharing out in alliances: Trust and ethics". In: *Journal of Business Ethics* 21.2-3, pp. 217–228.
- Avrachenkov, Konstantin et al. (2007). "Monte Carlo methods in PageRank computation: When one iteration is sufficient". In: *SIAM Journal on Numerical Analysis* 45.2, pp. 890–904.
- Bajo, Javier et al. (2016). "A low-level resource allocation in an agent-based Cloud Computing platform". In: *Applied Soft Computing* 48, pp. 716–728.
- Barber, K Suzanne, Karen Fullam, and Joonoo Kim (2002). "Challenges for trust, fraud and deception research in multi-agent systems". In: *Workshop on Deception, Fraud and Trust in Agent Societies*. Springer, pp. 8–14.
- Basheer, Ghusoon Salim et al. (2015). "Certainty, trust and evidence: Towards an integrative model of confidence in multi-agent systems". In: *Computers in Human Behavior* 45, pp. 307–315.

- Becker, Till and Stern Hendrik (2016). "Impact of resource sharing in manufacturing on logistical key figures". In: *Procedia CIRP* 41, pp. 579–584.
- Belk, Russell (2007). "Why not share rather than own?" In: *The Annals of the American Academy of Political and Social Science* 611.1, pp. 126–140.
- (2010). "Sharing". In: *Journal of consumer research* 36.5, pp. 715–734.
- Bendjima, Mostefa and Mohamed Feham (2015). "Multi-agent system for a reliable routing in WSN". In: *2015 Science and Information Conference (SAI)*. IEEE, pp. 1412–1419.
- Benkler, Yochai (2004). "Sharing nicely: On shareable goods and the emergence of sharing as a modality of economic production". In: *Yale LJ* 114, p. 273.
- Berenji, Hamid R (1988). "Treatment of uncertainty in artificial intelligence". In: *Machine intelligence and autonomy aerospace systems* 115, pp. 233–247.
- Bilgin, Aysenur et al. (2012). "Dynamic profile-selection for zlices based type-2 fuzzy agents controlling multi-user ambient intelligent environments". In: *2012 IEEE International Conference on Fuzzy Systems*. IEEE, pp. 1–8.
- Bird, Steven, Ewan Klein, and Edward Loper (2009). *Natural language processing with Python: analyzing text with the natural language toolkit*. " O'Reilly Media, Inc."
- Birddavid, N, T Widlok, and W. G Tadesse (2005). *The property of sharing: western analytical notions, Nayaka contexts*.
- Bohu, Li, Zhang Lin, and Chai Xudong (2020). "Introduction to cloud manufacturing". In: *ZTE Communications* 8.4, pp. 6–9.
- Botsman, Rachel and Roo Rogers (2010). "Beyond Zipcar: Collaborative Consumption". In: *Harvard Business Review* 88.10.
- Buczynski, Beth (2013). *Sharing is good: How to save money, time and resources through collaborative consumption*. New Society Publishers.
- Burnett, Chris and Nir Oren (2012). "Sub-delegation and trust". In: *Proceedings of the 11th International Conference on Autonomous Agents and Multiagent Systems-Volume 3*. International Foundation for Autonomous Agents and Multiagent Systems, pp. 1359–1360.
- Buzacott, John A and David D Yao (1986). "Flexible manufacturing systems: a review of analytical models". In: *Management science* 32.7, pp. 890–905.
- Cardoso, Tiago and Luis M Camarinha-Matos (2013). "Pro-active service ecosystem framework". In: *International Journal of Computer Integrated Manufacturing* 26.11, pp. 1021–1041.
- Castelfranchi, Christiano and Rino Falcone (2010). *Trust theory: A socio-cognitive and computational model*. Vol. 18. John Wiley & Sons.

- Chang, Liu et al. (2014). "Multi-criteria decision making based on trust and reputation in supply chain". In: *International Journal of Production Economics* 147, pp. 362–372.
- Chard, Kyle et al. (2011). "Social cloud computing: A vision for socially motivated resource sharing". In: *IEEE Transactions on services computing* 5.4, pp. 551–563.
- Cho, Jin-Hee, Kevin Chan, and Sibel Adali (2015). "A survey on trust modeling". In: *ACM Computing Surveys (CSUR)* 48.2, pp. 1–40.
- Ciari, Francesco, Claude Weis, and Milos Balac (2016). "Evaluating the influence of carsharing stations' location on potential membership: a Swiss case study". In: *Euro Journal on Transportation & Logistics* 5.3, pp. 345–369.
- Cobb, Barry R and Prakash P Shenoy (2006). "On the plausibility transformation method for translating belief function models to probability models". In: *International journal of approximate reasoning* 41.3, pp. 314–330.
- Coelho, Vitor N et al. (2017). "Multi-agent systems applied for energy systems integration: State-of-the-art applications and trends in microgrids". In: *Applied energy* 187, pp. 820–832.
- Cooper, Colin et al. (2014). "Random walks in recommender systems: exact computation and simulations". In: *Proceedings of the 23rd International Conference on World Wide Web*. ACM, pp. 811–816.
- Cruz, David Luviano and Wen Yu (2017). "Path planning of multi-agent systems in unknown environment with neural kernel smoothing and reinforcement learning". In: *Neurocomputing* 233, pp. 34–42.
- Cui, Huizi et al. (2019). "An improved deng entropy and its application in pattern recognition". In: *IEEE Access* 7, pp. 18284–18292.
- Cuzzolin, Fabio (2008). "A geometric approach to the theory of evidence". In: *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)* 38.4, pp. 522–534.
- Das, Anupam and Mohammad Mahfuzul Islam (2011). "SecuredTrust: a dynamic trust computation model for secured communication in multiagent systems". In: *IEEE Transactions on Dependable and Secure Computing* 9.2, pp. 261–274.
- Das, T. K. and B. S. Teng (2000). "A Resource-Based Theory of Strategic Alliances". In: *Journal of Management* 26.1, pp. 31–61.
- Daudi, M Sc Morice (2018). "Trust in Sharing Resources in Collaborative Logistics". In:
- Daudi, Morice, Klaus-Dieter Thoben, and Jannicke Baalsrud Hauge (2018). "An Approach for Surfacing Hidden Intentions and Trustworthiness in Logistics Resource Sharing Networks". In: *Working Conference on Virtual Enterprises*. Springer, pp. 524–536.

- Dempster, Arthur P. (1967). "Upper and Lower Probabilities Induced by a Multivalued Mapping". In: *Annals of Mathematical Statistics* 38.2, pp. 325–339.
- Deng, Shuiguang, Longtao Huang, and Guandong Xu (2014). "Social network-based service recommendation with trust enhancement". In: *Expert Systems with Applications* 41.18, pp. 8075–8084.
- Deng, Xinyang, Fuyuan Xiao, and Yong Deng (2017). "An improved distance-based total uncertainty measure in belief function theory". In: *Applied Intelligence* 46.4, pp. 898–915.
- Deng, Yong (2015). "A threat assessment model under uncertain environment". In: *Mathematical Problems in Engineering* 2015.
- (2016). "Deng entropy". In: *Chaos, Solitons & Fractals* 91, pp. 549–553.
- Deng, Yong et al. (2004). "Combining belief functions based on distance of evidence". In: *Decision support systems* 38.3, pp. 489–493.
- Di Amato, Alessio (2016). "Uber and the sharing economy". In: *Italian LJ* 2, p. 177.
- Dimitrov, Radoslav et al. (2019). "Institutional and environmental effectiveness: Will the Paris Agreement work?" In: *Wiley Interdisciplinary Reviews: Climate Change* 10.4, e583.
- Dorri, Ali, Salil S Kanhere, and Raja Jurdak (2018). "Multi-agent systems: A survey". In: *IEEE Access* 6, pp. 28573–28593.
- Dubois, Didier and Henri Prade (1987). "Properties of measures of information in evidence and possibility theories". In: *Fuzzy sets and systems* 24.2, pp. 161–182.
- Elham, Parhizkar, Mohammad Hossein Nikravan, and Sandra Zillesa (2019). "indirect is simple to establish". In: pp. 3216–3222.
- Ert, Eyal, Aliza Fleischer, and Nathan Magen (2016). "Trust and reputation in the sharing economy: The role of personal photos in Airbnb". In: *Tourism Management* 55, pp. 62–73.
- Fei, Liguang, Yong Deng, and Sankaran Mahadevan (2015). "Which is the best belief entropy". In: *Journal of Latex Class Files* 13.9, pp. 1–4.
- Feng, Renjian et al. (2011). "A trust evaluation algorithm for wireless sensor networks based on node behaviors and ds evidence theory". In: *Sensors* 11.2, pp. 1345–1360.
- Feng, Yu and Biqing Huang (2018). "A hierarchical and configurable reputation evaluation model for cloud manufacturing services based on collaborative filtering". In: *The International Journal of Advanced Manufacturing Technology* 94.9-12, pp. 3327–3343.
- Fisher, Oliver et al. (2018). "Cloud manufacturing as a sustainable process manufacturing route". In: *Journal of manufacturing systems* 47, pp. 53–68.

- Fraser, Malcolm (2011). "How to Measure Anything: Finding the Value of" Intangibles" in Business". In: *People & Strategy* 34.2, pp. 58–60.
- Fung, Carol J et al. (2011). "Dirichlet-based trust management for effective collaborative intrusion detection networks". In: *IEEE Transactions on Network and Service Management* 8.2, pp. 79–91.
- Gambetta, Diego et al. (2000). "Can we trust trust". In: *Trust: Making and breaking cooperative relations* 13, pp. 213–237.
- Gan, Jia and Gui-jiang Duan (2012). "Method of cloud manufacturing service trust evaluation". In: *Computer Integrated Manufacturing Systems* 18.7, pp. 1527–1535.
- Garrette, Bernard, Xavier Castañer, and Pierre Dussauge (2009). "Horizontal alliances as an alternative to autonomous production: Product expansion mode choice in the worldwide aircraft industry 1945–2000". In: *Strategic Management Journal* 30.8, pp. 885–894.
- Gatti, Maíra et al. (2013). "Large-scale multi-agent-based modeling and simulation of microblogging-based online social network". In: *International Workshop on Multi-Agent Systems and Agent-Based Simulation*. Springer, pp. 17–33.
- Gesing, Ben (2017). "Sharing economy logistics: Rethinking logistics with access over ownership". In: *DHL Customer Solutions% Innovation*.
- Ghobadi, Shahla and John D'Ambra (2012). "Knowledge sharing in crossfunctional teams: a cooperative model". In: *Journal of Knowledge Management* 16.2, pp. 285–301.
- Ghomi, Einollah Jafarnejad, Amir Masoud Rahmani, and Nooruldeen Nasih Qader (2019). "Cloud manufacturing: challenges, recent advances, open research issues, and future trends". In: *The International Journal of Advanced Manufacturing Technology* 102.9-12, pp. 3613–3639.
- Giffin, Kim (1967). "The contribution of studies of source credibility to a theory of interpersonal trust in the communication process." In: *Psychological bulletin* 68.2, p. 104.
- Gonzalez-Feliu, Jesus and Joëlle Morana (2011). "Collaborative transportation sharing: from theory to practice via a case study from France". In: *Technologies for supporting reasoning communities and collaborative decision making: Cooperative approaches*. IGI Global, pp. 252–271.
- Gorodetski, Vladimir and Igor Kotenko (2002). "The multi-agent systems for computer network security assurance: frameworks and case studies". In: *Proceedings 2002 IEEE International Conference on Artificial Intelligence Systems (ICAIS 2002)*. IEEE, pp. 297–302.

- Guerrini, Fábio Müller and Juliana Suemi Yamanari (2019). "A Systematic Review of Collaborative Networks: Implications for Sensing, Smart and Sustainable Enterprises". In: *Working Conference on Virtual Enterprises*. Springer, pp. 69–80.
- Gupta, Pankaj et al. (2013). "Wtf: The who to follow service at twitter". In: *Proceedings of the 22nd international conference on World Wide Web*. ACM, pp. 505–514.
- Gurven, Michael (2006). "The evolution of contingent cooperation". In: *Current Anthropology* 47.1, pp. 185–192.
- Hall, Richard (1992). "The Strategic Analysis of Intangible Resources". In: *Knowledge & Strategy* 13.2, pp. 135–144.
- Hamari, Juho, Mimmi Sjöklint, and Antti Ukkonen (2016). "The sharing economy: Why people participate in collaborative consumption". In: *Journal of the association for information science and technology* 67.9, pp. 2047–2059.
- Han, Deqiang et al. (2011). "Weighted evidence combination based on distance of evidence and uncertainty measure". In: *J. Infrared Millim. Waves* 30.5, pp. 396–400.
- Harmanec, David and George J Klir (1994). "Measuring total uncertainty in Dempster-Shafer theory: A novel approach". In: *International journal of general system* 22.4, pp. 405–419.
- He, Wei et al. (2019). "Multi-Objective Service Selection and Scheduling with Linguistic Preference in Cloud Manufacturing". In: *Sustainability* 11.9, p. 2619.
- He, Wu and Lida Xu (2015). "A state-of-the-art survey of cloud manufacturing". In: *International Journal of Computer Integrated Manufacturing* 28.3, pp. 239–250.
- Helper, Susan, Timothy Krueger, and Howard Wial (2012). "Why does manufacturing matter? Which manufacturing matters? A policy framework". In: *Metropolitan Policy Program at Brookings*.
- Höhle, Ulrich (1982). "Entropy with respect to plausibility measures". In: *Proc. of 12th IEEE Int. Symp. on Multiple Valued Logic, Paris, 1982*.
- Hu, Jianli, Quanyuan Wu, and Bin Zhou (2008). "FCTrust: A robust and efficient feedback credibility-based distributed P2P trust model". In: *2008 The 9th International Conference for Young Computer Scientists*. IEEE, pp. 1963–1968.
- Huang, Lei and Chaoyan Wu (2020). "Selection Approach of Cloud Manufacturing Resource for Manufacturing Enterprises Based on Trust Evaluation". In: *2020 Prognostics and Health Management Conference (PHM-Besançon)*. IEEE, pp. 309–313.
- Hussain, Mohd Shaaban and Mohammed Ali (2019). "A Multi-agent Based Dynamic Scheduling of Flexible Manufacturing Systems". In: *Global Journal of Flexible Systems Management* 20.3, pp. 267–290.
- Huynh, T Dong, Nicholas R Jennings, and N Shadbolt (2004a). "Developing an integrated trust and reputation model for open multi-agent systems". In:

- Huynh, Trung Dong, Nicholas R Jennings, and Nigel Shadbolt (2004b). "FIRE: An integrated trust and reputation model for open multi-agent systems". In:
- Huynh, Trung Dong, Nicholas R Jennings, and Nigel R Shadbolt (2006). "An integrated trust and reputation model for open multi-agent systems". In: *Autonomous Agents and Multi-Agent Systems* 13.2, pp. 119–154.
- Jamali, Mohsen and Martin Ester (2009). "Trustwalker: a random walk model for combining trust-based and item-based recommendation". In: *Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, pp. 397–406.
- Jeh, Glen and Jennifer Widom (2002). "SimRank: a measure of structural-context similarity". In: *Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, pp. 538–543.
- Jeong, Miyoung, Haemoon Oh, et al. (2017). "Business-to-business social exchange relationship beyond trust and commitment". In: *International Journal of Hospitality Management* 65, pp. 115–124.
- Jiang, Baojun and Lin Tian (2018). "Collaborative consumption: Strategic and economic implications of product sharing". In: *Management Science* 64.3, pp. 1171–1188.
- Jiang, Siwei, Jie Zhang, and Yew-Soon Ong (2013). "An evolutionary model for constructing robust trust networks". In: *Proceedings of the 2013 international conference on Autonomous agents and multi-agent systems*. International Foundation for Autonomous Agents and Multiagent Systems, pp. 813–820.
- Jiang, Wei-Jin et al. (2016). "Multi agent system-based dynamic trust calculation model and credit management mechanism of online trading". In: *Intelligent Automation & Soft Computing* 22.4, pp. 639–649.
- Jiang, Xu et al. (2015). "How does trust affect alliance performance? The mediating role of resource sharing". In: *Industrial Marketing Management* 45, pp. 128–138.
- Jiang, Zhong Zhong and Ding Wei Wang (2005). "Research on Optimal Design of Logistic Distribution Center for B2C E-Commerce". In: *Journal of Northeastern University*.
- Jiroušek, Radim and Prakash P Shenoy (2018). "A new definition of entropy of belief functions in the Dempster–Shafer theory". In: *International Journal of Approximate Reasoning* 92, pp. 49–65.
- Jøsang, Audun (1998). "A subjective metric of authentication". In: *European Symposium on Research in Computer Security*. Springer, pp. 329–344.
- Jøsang, Audun, Elizabeth Gray, and Michael Kinateder (2003). "Analysing topologies of transitive trust". In: *Proceedings of the First International Workshop on Formal Aspects in Security & Trust (FAST2003)*. Pisa, Italy, pp. 9–22.

- Jøsang, Audun and Roslan Ismail (2002). "The beta reputation system". In: *Proceedings of the 15th bled electronic commerce conference*. Vol. 5, pp. 2502–2511.
- Jøsang, Audun, Roslan Ismail, and Colin Boyd (2007). "A survey of trust and reputation systems for online service provision". In: *Decision support systems* 43.2, pp. 618–644.
- Jousselme, Anne-Laure, Dominic Grenier, and Éloi Bossé (2001). "A new distance between two bodies of evidence". In: *Information fusion* 2.2, pp. 91–101.
- Junyi, Sun (2015). *Jieba Python Library*.
- Kaihara, Toshiya et al. (2017). "Simulation model study for manufacturing effectiveness evaluation in crowdsourced manufacturing". In: *CIRP Annals* 66.1, pp. 445–448.
- Katrini, Eleni (2018). "Sharing Culture: On definitions, values, and emergence". In: *The Sociological Review* 66.2, pp. 425–446.
- Kerchove, Cristobald de and Paul Van Dooren (2008). "The pagetrust algorithm: How to rank web pages when negative links are allowed?". In: *Proceedings of the 2008 SIAM International Conference on Data Mining*. SIAM, pp. 346–352.
- Klaimi, Joelle et al. (2018). "A novel loss-based energy management approach for smart grids using multi-agent systems and intelligent storage systems". In: *Sustainable cities and society* 39, pp. 344–357.
- Kleinrock, Leonard (2002). "On resource sharing in a distributed communication environment". In: *IEEE Communications Magazine* 40.5, pp. 58–64.
- Klir, George J and Harold W Lewis (2008). "Remarks on "Measuring ambiguity in the evidence theory"". In: *IEEE Transactions on Systems, Man, and Cybernetics-Part A: Systems and Humans* 38.4, pp. 995–999.
- Kuklina, Vera et al. (2020). "Mobilizing Benefit-Sharing Through Transportation Infrastructure: Informal Roads, Extractive Industries and Benefit-Sharing in the Irkutsk Oil and Gas Region, Russia". In: *Resources* 9.3, p. 21.
- Kumar, V, Avishek Lahiri, and Orhan Bahadir Dogan (2018). "A strategic framework for a profitable business model in the sharing economy". In: *Industrial Marketing Management* 69, pp. 147–160.
- Laili, Yuanjun et al. (2012). "A study of optimal allocation of computing resources in cloud manufacturing systems". In: *The International Journal of Advanced Manufacturing Technology* 63.5-8, pp. 671–690.
- Lamberton, Cait Poynor and Randall L Rose (2012). "When is ours better than mine? A framework for understanding and altering participation in commercial sharing systems". In: *Journal of Marketing* 76.4, pp. 109–125.

- Lavie, Dovev (2006). "The competitive advantage of interconnected firms: An extension of the resource-based view". In: *Academy of management review* 31.3, pp. 638–658.
- Lee, Sunme and Yooncheong Cho (2015). "Exploring Utility, Attitude, Intention to Use, Satisfaction, and Loyalty in B2C/P2P Car-Sharing Economy". In: *Social Science Electronic Publishing*.
- Li, Changsong et al. (2014). "Trust evaluation model of cloud manufacturing service platform". In: *The international journal of advanced manufacturing technology* 75.1-4, pp. 489–501.
- Li, Zhi, Ali Vatankhah Barenji, and George Q Huang (2018). "Toward a blockchain cloud manufacturing system as a peer to peer distributed network platform". In: *Robotics and Computer-Integrated Manufacturing* 54, pp. 133–144.
- Liau, Churn-Jung (2003). "Belief, information acquisition, and trust in multi-agent systems a modal logic formulation". In: *Artificial Intelligence* 149.1, pp. 31–60.
- Lichtenwalter, Ryan N, Jake T Lussier, and Nitesh V Chawla (2010). "New perspectives and methods in link prediction". In: *Proceedings of the 16th ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 243–252.
- Lin, Zhiang, Haibin Yang, and Bindu Arya (2009). "Alliance partners and firm performance: resource complementarity and status association". In: *Strategic management journal* 30.1, pp. 921–940.
- Lipowski, Adam and Dorota Lipowska (2012). "Roulette-wheel selection via stochastic acceptance". In: *Physica A: Statistical Mechanics and its Applications* 391.6, pp. 2193–2196.
- Liu, Fan et al. (2019a). "Generalized belief entropy and its application in identifying conflict evidence". In: *IEEE Access* 7, pp. 126625–126633.
- Liu, Fang et al. (2011). "NIST cloud computing reference architecture". In: *NIST special publication* 500.2011, pp. 1–28.
- Liu, Jian and Youling Chen (2019). "A personalized clustering-based and reliable trust-aware QoS prediction approach for cloud service recommendation in cloud manufacturing". In: *Knowledge-Based Systems* 174, pp. 43–56.
- Liu, Qin et al. (2016). "Achieving reliable and secure services in cloud computing environments". In: *Computers & Electrical Engineering*, S004579061630502X.
- Liu, Xin and Anwitaman Datta (2012). "Modeling context aware dynamic trust using hidden markov model". In: *Twenty-Sixth AAAI Conference on Artificial Intelligence*.
- Liu, Yongkui et al. (2019b). "Cloud manufacturing: key issues and future perspectives". In: *International Journal of Computer Integrated Manufacturing* 32.9, pp. 858–874.

- Liu, Yongkui et al. (2019c). "Scheduling in cloud manufacturing: state-of-the-art and research challenges". In: *International Journal of Production Research* 57.15-16, pp. 4854–4879.
- Lü, Linyuan et al. (2012). "Recommender systems". In: *Physics reports* 519.1, pp. 1–49.
- Lu, Yuqian and Xun Xu (2015). "Protecting intellectual property in a cloud manufacturing environment: Requirements and strategies". In: *IFIP International Conference on Advances in Production Management Systems*. Springer, pp. 404–411.
- Ma, Ke et al. (2018). "A resource sharing solution optimized by simulation-based heuristic for garment manufacturing". In: *The International Journal of Advanced Manufacturing Technology* 99.9-12, pp. 2803–2818.
- Ma, Lizhu and Yu Zhang (2014). "Hierarchical social network analysis using multi-agent systems: A school system case". In: *2014 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*. IEEE, pp. 1412–1419.
- Mao, Tai Tian (2008). "Regional Logistics Information Resource Sharing Model Under Network Environment". In: *Commercial Research*.
- Marsh, Stephen Paul (1994). "Formalising trust as a computational concept". In: McKnight, D Harrison and Norman L Chervany (1996). "The meanings of trust". In: Meng, Wei et al. (2014). "Reputation-based multi-dimensional trust model in cloud manufacturing service platform". In: *Multiagent and Grid Systems* 10.4, pp. 233–246.
- Meroni, Anna (2007). *Creative Communities. People inventing sustainable ways of living*. Edizioni Polidesign.
- Merris, Russell (1994). "Laplacian matrices of graphs: a survey". In: *Linear algebra and its applications* 197, pp. 143–176.
- Mohamed, Nader and Jameela Al-Jaroodi (2015). "MidCloud: an agent-based middleware for effective utilization of replicated Cloud services". In: *Software: Practice and Experience* 45.3, pp. 343–363.
- Morgan, Blake (2019). *NOwnership, No Problem: An Updated Look At Why Millennials Value Experiences Over Owning Things*. <https://www.forbes.com/sites/blakemorgan/2019/01/02/nownership-no-problem-an-updated-look-at-why-millennials-value-experiences-over-owning-things/#561ac229522f>.
- Muller, Guillaume, Laurent Vercouter, and Olivier Boissier (2003). "Towards a general definition of trust and its application to openness in MAS". In: *Proc. of the AAMAS-2003 Workshop on Deception, Fraud and Trust*.
- Neu, Wayne A and Stephen W Brown (2005). "Forming successful business-to-business services in goods-dominant firms". In: *Journal of service research* 8.1, pp. 3–17.
- Nguyen, Hung T (1987). "On entropy of random sets and possibility distributions". In: *The Analysis of Fuzzy Information* 1, pp. 145–156.

- Nguyen, Phuong et al. (2015). "An evaluation of SimRank and Personalized PageRank to build a recommender system for the Web of Data". In: *Proceedings of the 24th International Conference on World Wide Web*. ACM, pp. 1477–1482.
- Nguyen, Tung Doan and Quan Bai (2018). "A Dynamic Bayesian Network approach for agent group trust evaluation". In: *Computers in Human Behavior* 89, pp. 237–245.
- Nogoorani, Sadegh Dorri and Rasool Jalili (2016). "Uncertainty in trust: a risk-aware approach". In: *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems* 24.05, pp. 703–737.
- Noorian, Zeinab and Mihaela Ulieru (2010). "The state of the art in trust and reputation systems: a framework for comparison". In: *Journal of theoretical and applied electronic commerce research* 5.2, pp. 97–117.
- Özkan, Kürşad (2018). "Comparing Shannon entropy with Deng entropy and improved Deng entropy for measuring biodiversity when a priori data is not clear". In: *FORESTIST* 68.2, pp. 136–140.
- Paaajanen, Salla (2019). "Business-to-Business Resource Sharing". In: *VTT Technical Research Centre of Finland Business, Innovation and Foresight: Espoo, Finland*.
- Page, Lawrence et al. (1999). *The PageRank citation ranking: Bringing order to the web*. Tech. rep. Stanford InfoLab.
- Pan, Yiteng et al. (2020). "Learning adaptive trust strength with user roles of truster and trustee for trust-aware recommender systems". In: *Applied Intelligence* 50.2, pp. 314–327.
- Papadimitriou, Panagiotis and Hector Garcia-Molina (2010). "Data leakage detection". In: *IEEE Transactions on knowledge and data engineering* 23.1, pp. 51–63.
- Paudel, Bibek et al. (2017). "Updatable, accurate, diverse, and scalable recommendations for interactive applications". In: *ACM Transactions on Interactive Intelligent Systems (TiiS)* 7.1, p. 1. DOI: 10.1145/2955101.
- Premarathne, US and S Rajasingham (2020). "Trust based multi-agent cooperative load balancing system (TCLBS)". In: *Future Generation Computer Systems*.
- Price, John A (1975). "Sharing: The integration of intimate economies". In: *Anthropologica*, pp. 3–27.
- Prieta, Fernando De la et al. (2019). "Survey of agent-based cloud computing applications". In: *Future Generation Computer Systems* 100, pp. 223–236.
- Puschmann, Thomas and Rainer Alt (2016). "Sharing economy". In: *Business & Information Systems Engineering* 58.1, pp. 93–99.
- Qi, Lianyong et al. (2017). "Privacy-preserving distributed service recommendation based on locality-sensitive hashing". In: *2017 IEEE International conference on web services (ICWS)*. IEEE, pp. 49–56.

- Qian, Jin, Xingfeng Guo, and Yong Deng (2017). "A novel method for combining conflicting evidences based on information entropy". In: *Applied Intelligence* 46.4, pp. 876–888.
- Qiu, Xiang et al. (2010). "A Trust Transitivity Model Based-on Dempster-Shafer Theory". In: *Journal of Networks* 5.9, p. 1025.
- Ramchurn, Sarvapali et al. (2003). "A computational trust model for multi-agent interactions based on confidence and reputation". In:
- Ren, Lei et al. (2015). "Cloud manufacturing: from concept to practice". In: *Enterprise Information Systems* 9.2, pp. 186–209.
- Rishwaraj, G, SG Ponnambalam, and Loo Chu Kiong (2017). "An efficient trust estimation model for multi-agent systems using temporal difference learning". In: *Neural Computing and Applications* 28.1, pp. 461–474.
- Ristic, Branko and Philippe Smets (2006). "The TBM global distance measure for the association of uncertain combat ID declarations". In: *Information fusion* 7.3, pp. 276–284.
- Rong, Huigui et al. (2014). "User similarity-based collaborative filtering recommendation algorithm". In: *Journal on Communications* 35.2, pp. 16–24.
- Ruan, Yefeng and Arjan Durresi (2016). "A survey of trust management systems for online social communities—trust modeling, trust inference and attacks". In: *Knowledge-Based Systems* 106, pp. 150–163.
- Sabater, Jordi and Carles Sierra (2002). "Reputation and social network analysis in multi-agent systems". In: *Proceedings of the first international joint conference on Autonomous agents and multiagent systems: Part 1*, pp. 475–482.
- (2005). "Review on computational trust and reputation models". In: *Artificial intelligence review* 24.1, pp. 33–60.
- Sadowski, Caitlin and Greg Levin (2007). "Simhash: Hash-based similarity detection". In: *Technical report, Google*.
- Sandberg, Erik (2007). "Logistics collaboration in supply chains: practice vs. theory". In: *The International Journal of Logistics Management*.
- Shafer, Glenn (1976). "A Mathematical Theory of Evidence". In: *A Mathematical Theory of Evidence* 20.1, p. 242.
- Shaheen, Susan, Daniel Sperling, and Conrad Wagner (1998). "Carsharing in Europe and North American: past, present, and future". In:
- Shannon, Claude E (1948). "A mathematical theory of communication". In: *Bell system technical journal* 27.3, pp. 379–423.
- Shao, Yingxia et al. (2015). "An efficient similarity search framework for SimRank over large dynamic graphs". In: *Proceedings of the VLDB Endowment* 8.8, pp. 838–849.

- Shehada, Dina et al. (2018). "A new adaptive trust and reputation model for Mobile Agent Systems". In: *Journal of Network and Computer Applications* 124, pp. 33–43.
- Shen, Jian et al. (2017a). "Anonymous and traceable group data sharing in cloud computing". In: *IEEE Transactions on Information Forensics and Security* 13.4, pp. 912–925.
- Shen, Jian et al. (2017b). "Block design-based key agreement for group data sharing in cloud computing". In: *IEEE Transactions on Dependable and Secure Computing*.
- Sherchan, Wanita, Surya Nepal, and Cecile Paris (2013). "A survey of trust in social networks". In: *ACM Computing Surveys (CSUR)* 45.4, pp. 1–33.
- Shi, Chuan et al. (2016). "A survey of heterogeneous information network analysis". In: *IEEE Transactions on Knowledge and Data Engineering* 29.1, pp. 17–37.
- Shi, Lei et al. (2019). "Study on the bike-sharing inventory rebalancing and vehicle routing for bike-sharing system". In: *Transportation research procedia* 39, pp. 624–633.
- Shmatikov, Vitaly and Carolyn Talcott (2005). "Reputation-based trust management". In: *Journal of Computer Security* 13.1, pp. 167–190.
- Shove, Elizabeth, Mika Pantzar, and Matt Watson (2012). *The dynamics of social practice: Everyday life and how it changes*. Sage.
- Sim, Kwang Mong (2011). "Agent-based cloud computing". In: *IEEE transactions on services computing* 5.4, pp. 564–577.
- Škulj, Gašper et al. (2017). "Decentralised network architecture for cloud manufacturing". In: *International Journal of Computer Integrated Manufacturing* 30.4–5, pp. 395–408.
- Smets, Philippe (1983). "Information content of an evidence". In: *International Journal of Man-Machine Studies* 19.1, pp. 33–43.
- Song, Junshuai et al. (2018). "UniWalk: Unidirectional Random Walk Based Scalable SimRank Computation over Large Graph". In: *IEEE Transactions on Knowledge and Data Engineering* 30.5, pp. 992–1006. DOI: 10.1109/TKDE.2017.2779126.
- Song, Shanshan and Kai Hwang (2004). "Dynamic Grid security with trust integration and optimized resource allocation". In: *Internet and Grid Computing Laboratory, University of Southern California, Los Angeles, CA 90089*.
- Sprei, Frances et al. (2019). "Free-floating car-sharing electrification and mode displacement: Travel time and usage patterns from 12 cities in Europe and the United States". In: *Transportation Research Part D: Transport and Environment* 71, pp. 127–140.
- Srinivasan, Dipti (2010). *Innovations in Multi-Agent Systems and Application–1*. Vol. 310. Springer.

- Srivastava, Atul Kumar, Rakhi Garg, and PK Mishra (2017). "Discussion on damping factor value in PageRank computation". In: *International Journal of Intelligent Systems and Applications* 9.9, p. 19.
- Staab, Steffen et al. (2004). "The pudding of trust". In: *IEEE Intelligent Systems* 19.5, pp. 74–88.
- Stone, Peter and Manuela Veloso (2000). "Multiagent systems: A survey from a machine learning perspective". In: *Autonomous Robots* 8.3, pp. 345–383.
- Such, Jose M, Agustín Espinosa, and Ana García-Fornes (2014). "A survey of privacy in multi-agent systems". In: *The Knowledge Engineering Review* 29.3, pp. 314–344.
- Sun, Ningning et al. (2019). "Research on cloud computing in the resource sharing system of university library services". In: *Evolutionary Intelligence* 12.3, pp. 377–384.
- Sun, Yizhou and Jiawei Han (2012). "Mining heterogeneous information networks: principles and methodologies". In: *Synthesis Lectures on Data Mining and Knowledge Discovery* 3.2, pp. 1–159.
- swiss Biryuellefabrik (2019). <https://www.virtuellefabrik.ch/>.
- Szaller, Ádám, Péter Egri, and Botond Kádár (2020). "Trust-based resource sharing mechanism in distributed manufacturing". In: *International Journal of Computer Integrated Manufacturing* 33.1, pp. 1–21.
- Talia, Domenico (2011). "Cloud Computing and Software Agents: Towards Cloud Intelligent Services." In: *WOA*. Vol. 11. Citeseer, pp. 2–6.
- Tang, Yongchuan et al. (2017). "An improved belief entropy-based uncertainty management approach for sensor data fusion". In: *International Journal of Distributed Sensor Networks* 13.7, p. 1550147717718497.
- Tao, Fei, Ye Fa Hu, et al. (2008). "Study on manufacturing grid & its resource service optimal-selection system". In: *The International Journal of Advanced Manufacturing Technology* 37.9-10, pp. 1022–1041.
- Tao, Fei et al. (2011). "Cloud manufacturing: a computing and service-oriented manufacturing model". In: *Proceedings of the Institution of Mechanical Engineers, Part B: Journal of Engineering Manufacture* 225.10, pp. 1969–1976.
- Tao, Fei et al. (2012). "FC-PACO-RM: a parallel method for service composition optimal-selection in cloud manufacturing system". In: *IEEE Transactions on Industrial Informatics* 9.4, pp. 2023–2033.
- Teacy, WT Luke et al. (2006). "Travos: Trust and reputation in the context of inaccurate information sources". In: *Autonomous Agents and Multi-Agent Systems* 12.2, pp. 183–198.

- Teacy, WT Luke et al. (2012). "An efficient and versatile approach to trust and reputation using hierarchical bayesian modelling". In: *Artificial Intelligence* 193, pp. 149–185.
- Teubner, Timm et al. (2014). "Understanding resource sharing in C2C platforms: The role of picture humanization". In: ACIS.
- The State Information Center (2020). *China's sharing economy*. <http://www.sic.gov.cn/News/568/10429.htm>.
- Thomas, Team (2020). *How Big Companies Choose New Supplier*. <https://blog.thomasnet.com/how-big-companies-choose-new-suppliers>.
- Toivonen, Santtu, Gabriele Lenzini, and Ilkka Uusitalo (2006). "Context-aware Trust Evaluation Functions for Dynamic Reconfigurable Systems." In: *MTW* 190.
- United nations industrial development organization (2020). *World Manufacturing Production Statistics for Quarter I, 2020*. <https://www.unido.org/resources-statistics/quarterly-report-manufacturing>.
- Urena, Raquel et al. (2019). "A review on trust propagation and opinion dynamics in social networks and group decision making frameworks". In: *Information Sciences* 478, pp. 461–475.
- Vatankhah Barenji, Ali and Reza Vatankhah Barenji (2017). "Improving multi-agent manufacturing control system by indirect communication based on ant agents". In: *Proceedings of the Institution of Mechanical Engineers, Part I: Journal of Systems and Control Engineering* 231.6, pp. 447–458.
- Walter, Frank Edward, Stefano Battiston, and Frank Schweitzer (2008). "A model of a trust-based recommendation system on a social network". In: *Autonomous Agents and Multi-Agent Systems* 16.1, pp. 57–74. DOI: 10.1007/s10458-007-9021-x.
- Wang, Ningkui et al. (2018). "An Agent-based Distributed Approach for Bike Sharing Systems". In: *International Conference on Computational Science*. Springer, pp. 540–552.
- Wang, Ningkui et al. (2020a). "An improved evidence theory-based trust model for multiagent resource allocation". In: *IEEE INTERNATIONAL CONFERENCE ON SYSTEMS, MAN, AND CYBERNETICS*.
- Wang, Ningkui et al. (2020b). "Preference-based Resource Reservation Method for Resource Allocation in Full Distributed System". In: *Multiagent and grid system*, Under review.
- Wang, Xi Vincent, Lihui Wang, and Reinhold Gördes (2018). "Interoperability in cloud manufacturing: a case study on private cloud structure for SMEs". In: *International Journal of Computer Integrated Manufacturing* 31.7, pp. 653–663.

- Wang, Yan, Lei Li, and Guanfeng Liu (2015). "Social context-aware trust inference for trust enhancement in social network based recommendations on service providers". In: *World Wide Web* 18.1, pp. 159–184.
- Wang, Yao and Julita Vassileva (2003). "Bayesian network-based trust model". In: *Proceedings IEEE/WIC International Conference on Web Intelligence (WI 2003)*. IEEE, pp. 372–378.
- Wang, Yonghong and Munindar P Singh (2006). *Trust via evidence combination: A mathematical approach based on certainty*. Tech. rep. North Carolina State University. Dept. of Computer Science.
- (2007). "Formal Trust Model for Multiagent Systems." In: *IJCAI*. Vol. 7, pp. 1551–1556.
- (2010). "Evidence-based trust: A mathematical model geared for multiagent systems". In: *ACM Transactions on Autonomous and Adaptive Systems (TAAS)* 5.4, pp. 1–28.
- Wellman, Barry (2001). "Computer networks as social networks". In: *Science* 293.5537, pp. 2031–2034. DOI: 10.1126/science.1065547.
- Wen, Chenglin, Yingchang Wang, and Xiaobin Xu (2008). "Fuzzy information fusion algorithm of fault diagnosis based on similarity measure of evidence". In: *International Symposium on Neural Networks*. Springer, pp. 506–515.
- West, Darrell and Christian Lansang (2018). "Global manufacturing scorecard: How the US compares to 18 other nations". In: *Brookings*, July 10.
- Winterhalder, Bruce (1986). "Diet choice, risk, and food sharing in a stochastic environment". In:
- Wooldridge, Michael (1999). "Intelligent agents". In: *Multiagent systems* 35.4, p. 51.
- (2009). *An introduction to multiagent systems*. John Wiley & Sons.
- Wooldridge, Michael and Nicholas R Jennings (1995). "Intelligent agents: Theory and practice". In: *The knowledge engineering review* 10.2, pp. 115–152.
- Wu, Dazhong et al. (2013a). "Cloud manufacturing: Strategic vision and state-of-the-art". In: *Journal of Manufacturing Systems* 32.4, pp. 564–579.
- Wu, Dazhong et al. (2013b). "Enhancing the product realization process with cloud-based design and manufacturing systems". In: *Journal of Computing and Information Science in Engineering* 13.4.
- Xiong, Li and Ling Liu (2003). "A reputation-based trust model for peer-to-peer e-commerce communities". In: *EEE International Conference on E-Commerce, 2003. CEC 2003*. IEEE, pp. 275–284.
- (2004). "Peertrust: Supporting reputation-based trust for peer-to-peer electronic communities". In: *IEEE transactions on Knowledge and Data Engineering* 16.7, pp. 843–857.

- Xu, Guangquan et al. (2007). "Swift trust in a virtual temporary system: A model based on the Dempster-Shafer theory of belief functions". In: *International Journal of Electronic Commerce* 12.1, pp. 93–126.
- Xu, Xun (2012). "From cloud computing to cloud manufacturing". In: *Robotics and computer-integrated manufacturing* 28.1, pp. 75–86.
- Yager, Ronald R (1983). "Entropy and specificity in a mathematical theory of evidence". In: *International Journal of General System* 9.4, pp. 249–260.
- Yan, Kai, Ying Cheng, and Fei Tao (2016). "A trust evaluation model towards cloud manufacturing". In: *The International Journal of Advanced Manufacturing Technology* 84.1-4, pp. 133–146.
- Yan, Zheng, Peng Zhang, and Athanasios V Vasilakos (2014). "A survey on trust management for Internet of Things". In: *Journal of network and computer applications* 42, pp. 120–134.
- Yang, Xingxing et al. (2019). "A service satisfaction-based trust evaluation model for cloud manufacturing". In: *International Journal of Computer Integrated Manufacturing* 32.6, pp. 533–545.
- Yao, Kai and Hua Ke (2014). "Entropy operator for membership function of uncertain set". In: *Applied Mathematics and Computation* 242, pp. 898–906.
- Ye, Dayong, Minjie Zhang, and Athanasios V Vasilakos (2016). "A survey of self-organization mechanisms in multiagent systems". In: *IEEE Transactions on Systems, Man, and Cybernetics: Systems* 47.3, pp. 441–461.
- Yu, Bin, Srikanth Kallurkar, and Robert Flo (2008). "A demspter-shafer approach to provenance-aware trust assessment". In: *2008 International Symposium on Collaborative Technologies and Systems*. IEEE, pp. 383–390.
- Yu, Bin and Munindar P Singh (2002a). "An evidential model of distributed reputation management". In: *Proceedings of the first international joint conference on Autonomous Agents and Multiagent Systems: Part 1*. ACM, pp. 294–301.
- (2002b). "Distributed reputation management for electronic commerce". In: *Computational Intelligence* 18.4, pp. 535–549.
- (2003). "Detecting deception in reputation management". In: *Proceedings of the second international joint conference on Autonomous agents and multiagent systems*. ACM, pp. 73–80.
- Yu, Bin, Munindar P Singh, and Katia Sycara (2004). "Developing trust in large-scale peer-to-peer systems". In: *IEEE First Symposium on Multi-Agent Security and Survivability, 2004*. IEEE, pp. 1–10.
- Yu, Chia-Hao et al. (2011). "Resource sharing optimization for device-to-device communication underlying cellular networks". In: *IEEE Transactions on Wireless communications* 10.8, pp. 2752–2763.

- Yu, Han et al. (2013). "A survey of multi-agent trust management systems". In: *IEEE Access* 1, pp. 35–50.
- Yu, Han et al. (2014). "Filtering trust opinions through reinforcement learning". In: *Decision Support Systems* 66, pp. 102–113.
- Yu, Jian and Yanbo Han (2006). "Service oriented computing-principle and application". In: *Qing Hua University publisher, Beijing*.
- Yuan, Kaijuan et al. (2016). "Modeling sensor reliability in fault diagnosis based on evidence theory". In: *Sensors* 16.1, p. 113.
- Zhang, Lin et al. (2014). "Cloud manufacturing: a new manufacturing paradigm". In: *Enterprise Information Systems* 8.2, pp. 167–187.
- Zhang, Yingfeng et al. (2017). "Research on services encapsulation and virtualization access model of machine for cloud manufacturing". In: *Journal of Intelligent Manufacturing* 28.5, pp. 1109–1123.
- Zhang, Yunchang, Shanshan Chen, and Geng Yang (2009). "SFTrust: A double trust metric based trust model in unstructured P2P system". In: *2009 IEEE International Symposium on Parallel & Distributed Processing*. IEEE, pp. 1–7.
- Zhao, Chun, Xiao Luo, and Lin Zhang (2020). "Modeling of service agents for simulation in cloud manufacturing". In: *Robotics and Computer-Integrated Manufacturing* 64, p. 101910.
- Zhao, Chun, Lihui Wang, and Xuesong Zhang (2020). "Service agent networks in cloud manufacturing: Modeling and evaluation based on set-pair analysis". In: *Robotics and Computer-Integrated Manufacturing* 65, p. 101970.
- Zheng, Yalin and Wenyu (Derek) Du (2019). "OFO Bicycle: Value of Convenience Added via Sharing Bicycles". In: *World Scientific Book Chapters*.
- Zheng, Yuanshi and Long Wang (2012). "Distributed consensus of heterogeneous multi-agent systems with fixed and switching topologies". In: *International Journal of Control* 85.12, pp. 1967–1976.
- Zhong, Ray Y et al. (2017). "Intelligent manufacturing in the context of industry 4.0: a review". In: *Engineering* 3.5, pp. 616–630.
- Zhou, Longfei et al. (2018). "An event-triggered dynamic scheduling method for randomly arriving tasks in cloud manufacturing". In: *International Journal of Computer Integrated Manufacturing* 31.3, pp. 318–333.
- Zhou, Nan et al. (2019). "A roadmap for China to peak carbon dioxide emissions and achieve a 20% share of non-fossil fuels in primary energy by 2030". In: *Applied energy* 239, pp. 793–819.
- Zouhal, Lalla Meriem and Thierry Denoeux (1998). "An evidence-theoretic k-NN rule with parameter optimization". In: *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)* 28.2, pp. 263–271.

-
- Zuo, Yanjun and Jigang Liu (2017). "A reputation-based model for mobile agent migration for information search and retrieval". In: *International Journal of Information Management* 37.5, pp. 357–366.

Titre en français

Gestion de l'incertitude par la confiance dans les systèmes de partage de ressources

Résumé en français

La fabrication en réseau orientée vers les services, connue sous le nom de "abrication en nuage", est un paradigme de fabrication innovant. Elle transforme les ressources et les produits de fabrication en services de fabrication, qui peuvent être gérés et exploités de manière intelligente et unifiée pour permettre le partage et la circulation complets de ces services entre les clients et les fournisseurs. Cette thèse présente des modèles et des méthodes avancés basés sur les système multi-agent avec prise en compte de l'incertitude et de la gestion intelligente de la confiance entre clients et fournisseurs pour fournir des solutions à tous les différents utilisateurs impliqués dans le cycle de vie complet de la fabrication.

Mots-clés: Fabrication en nuage, systèmes distribués, systèmes multiagents, confiance, incertitude, selection de services

Titre en anglais

Managing uncertainty through trust in resource sharing systems

Résumé en anglais

Service-oriented networked manufacturing, known as "cloud manufacturing," is an innovative manufacturing paradigm. It transforms manufacturing resources and products into manufacturing services that can be intelligently and unifiedly managed and operated to enable the full sharing and flow of these services between customers and suppliers. This thesis presents advanced models and methods based on multiagent systems with consideration of uncertainty and intelligent trust management between customers and suppliers to provide solutions to all the different users involved in the complete manufacturing lifecycle.

Mots-clés: Cloud manufacturing, distributed system, multi-agent system, trust, uncertainty, service selection.