TRUST EVALUATION AND ESTABLISHMENT FOR MULTI-AGENT SYSTEMS

by

Abdullah Aref

Thesis submitted to the
Faculty of Engineering
In partial fulfillment of the requirements
For the Ph.D. degree in
Computer Science

School of Electrical Engineering and Computer Science
Faculty of Engineering
University of Ottawa

Abstract

Multi-agent systems are increasingly popular for modeling distributed environments that are highly complex and dynamic such as e-commerce, smart buildings, and smart grids. Often in open multi-agent systems, agents interact with other agents to meet their own goals. Trust is considered significant in multi-agent systems to make interactions effectively, especially when agents cannot assure that potential partners share the same core beliefs about the system or make accurate statements regarding their competencies and abilities. This work describes a trust model that augments fuzzy logic with Q-learning, and a suspension technique to help trust evaluating agents select beneficial trustees for interaction in uncertain, imprecise, and the dynamic multi-agent systems. Q-Learning is used to evaluate trust on the long term, fuzzy inferences are used to aggregate different trust factors and suspension is used as a short-term response to dynamic changes. The performance of the proposed model is evaluated using simulation. Simulation results indicate that the proposed model can help agents select trustworthy partners to interact with. It has a better performance compared to some of the popular trust models in the presence of misbehaving interaction partners.

When interactions are based on trust, trust establishment mechanisms can be used to direct trustees, instead of trustors, to build a higher level of trust and have a greater impact on the results of interactions. This work also describes a trust establishment model for intelligent agents using implicit feedback that goes beyond trust evaluation to outline actions to guide trustees (instead of trustors). The model uses the retention of trustors to model trustors' behaviours. For situations where tasks are multi-criteria and explicit feedback is available, we present a trust establishment model that uses a multi-criteria approach to help trustees to adjust their behaviours to improve their perceived trust and attract more interactions with trustors. The model calculates the necessary improvement per criterion when only a single aggregated satisfaction value is provided per interaction, where the model attempts to predicted both the appropriate value per criteria and its importance. Then we present a trust establishment model that integrates the two major sources of information to produce a comprehensive assessment of a trustor's likely needs in multi-agent

systems. Specifically, the model attempts to incorporates explicit feedback, and implicit feedback assuming multi-criteria tasks. The proposed models are evaluated through simulation, we found that trustees can enhance their trustworthiness, at a cost, if they tune their behaviour in response to feedback (explicit or implicit) from trustors. Using explicit feedback with multi-criteria tasks, trustees can emphasize on important criterion to satisfy need of trustors. Trust establishment based on explicit feedback for multi-criteria tasks, can result in a more effective and efficient trust establishment compared to using implicit feedback alone. Integrating both approaches together can achieve a reasonable trust level at a relatively lower cost.

Acknowledgements

I would like to express my deepest gratitude and thank my research advisor, Dr. Thomas Tran, for mentoring me during my PhD study and research. I have learned a lot from him. Without his persistent help, patience and guidance, this dissertation would not have materialized. I cannot thank him enough for his sincere and overwhelming help and support. Also, I would like to thank the faculty and staff of the School of Electrical Engineering and Computer Science at the University of Ottawa for their help and support.

I must express my gratitude to Eman, my wife, for her continued support, encouragement, and help to get through this agonizing period in the most positive way. I am grateful to my mother, father, brother and sister who have experienced all of the ups and downs of my research and have provided me through moral and emotional support. I am also grateful to my other family members and friends who have supported me along the way.

Finally, I would like to thank all people who made this possible.

Dedication

This dissertation is dedicated to the memory of my father, Majed Aref, his endless love, care and support have sustained me throughout my life.

Table of Contents

List of Tables

List of Figures

List of Symbols

Abbreviations

Symbol	Description	Page
AMT	Amazon Mechanical Turk	48
ART	Agent Reputation and Trust	71
BSI	Basic Suspension Interval	57
CR	Certified Reputation	47
DiReCT	Dirichlet-based Reputation and Credential Trust management	39
DRAFT	Distributed Request Acceptance approach for Fair utilization of Trustees	48
DT	Direct Trust	46
DTMAS	Decentralized Trust Evaluation Model for Multi-Agent Systems Using Fuzzy Logic and Q-Learning	56
FLS	Fuzzy Logic System	15
FQ-Learning	Fuzzy logic-based Q-Learning	17
FrT	Fraudulent Threshold	59
H	High	57

HoT	Honesty Threshold	59
IoA	Internet of Agents	151
IoT	Internet of Things	151
ITE	An Integrated Trust Establishment Model for MASs	99
L	Low	57
M	Medium	57
MAS	Multi-Agent Systems	2
MCTE	Multi-Criteria Trust Establishment Model Using Explicit Feedback	91
MDP	Markov Decision Process	14
MF	Membership Function	15
PATROL	comPrehensive reputAtion-based TRust moOdeL	43
PATROLF	comPrehensive reputAtion-based TRust moOdeL with Fuzzy subsystems	41
RI	Reputational Incentive	51
RL	Reinforcement Learning	14
RM	Relevancy Matrix	38
RT	Role based Trust	47
SG	Smart Grid	151
SWORD	Social Welfare Optimizing Reputation-aware Decision making	48
TD	Temporal-Difference	14
TEIF	Trust Establishment Using Implicit Feedback	88
TRAVOS	Trust and Reputation in the Context of Inaccurate Information Sources	41

UG	Utility Gain	22
VAMET	Vehicular Ad Hoc Networks	26
WBSI	Witness Basic Suspension Interval	58
WCFLS	Witnesses Credibility Fuzzy Logic Subsystem	69
WM	Weight Matrix	38
WT	Witness Trust	47

.

Mathematical Symbols

Symbol	Description	Page
α_1	The cooperation factor for calculating DT_x^y	59
α_2	The engagement increment factor in TEIF	91
α_3	The engagement increment factor in MCTE	97
α_4	The engagement increment factor in ITE	101
β_2	The engagement decrements factor in TEIF	91
β_3	The engagement decrements factor in MCTE	97
β_4	The engagement decrements factor in ITE	101
γ_1	The cooperation factor for calculating TW_x^w	69
γ_3	The attraction-factor-scaling in MCTE	97
γ_4	The attraction-factor-scaling in ITE	102
κ_3	The decrements factor of relative weight	95
λ_1	The decaying factor	61
λ_3	The increment factor of relative weight	95
$\mu_{FT_x'^y}(ft)$	The membership function of the aggregated fuzzy set FT	68

Ω_2	The trustee's y's engagement threshold	89
Ω_3	The SATISFACTION-THRESHOLD used in MCTE	96
Ω_4	The trustee's y's engagement threshold	101
ω_4	The weight of implicit feedback in ITE	99
$\omega_{K(r_i)}$	The rating weight function that calculates the relevance or the reliability of the rating r_i	47
$\overline{Ret(s,t)}$	The most recently calculated average retention rate	88
ϕ_3	The RELATIVE-WEIGHT-THRESHOLD used in MCTE	96
ρ_2	The degree of decay for calculating R_y^x	86
σ_i^d	The standard deviation for the Gaussian MF	62
Θ_2	The trustee's y's un-engagement threshold	89
Θ_3	The DIS-SATISFACTION-THRESHOLD used in MCTE	96
Θ_4	The trustee's y's un-engagement threshold	101
$max_{i \in X'}(R_y^i)$	The maximum retention of an individual trustor in the set X'	87
ζ_1	The cooperation factor for calculating TW_x^w	69
ζ_3	The profit-making-factor-scaling in MCTE	97
ζ_4	The profit-making-factor-scaling in ITE	102
$A_{patrolf}$	The weight of for trust evaluation calculated by the the trustor itself for $PATROL_F$	43
AUG'	The average of time decayed utility gain within the last G interaction with a trustee y	61

B_i^d	The fuzzy set that can be interpreted as v is a member of the fuzzy set d	62
c_i^d	The mean values for the corresponding Gaussian MF	62
$d_x(s_{c_i})$	The demand level of criterion c_i of task s as determined by trustor x	93
DT_x^y	Direct Trustworthiness Estimation of the trustee y as calculated by the trustor x.	59
DT_x^y	The non-cooperation factor for calculating DT_x^y	60
$E\_Imp_x^y(s_{c_i}, req)$	The improvement calculated by y for criterion s_{c_i} of task s in response to request req by trustee x based on Explicit feedback	99
FT_w^y	w's fuzzy trust evaluation of y	60
FW_w^k	The fuzzy credibility of witness w	61
G	The size of the historical window considered for calculating AUG'$_x^y$	61
$g^y(s,t)$	The general retention trend function of y	87
$I\_Imp_x^y(s_{c_i}, req)$	The improvement calculated by y for criterion s_{c_i} of task s in response to request req by trustee x based on implicit feedback	99
$Imp_x^y(s_{c_i}$	The improvement for individual criterion	97
$Improvement_x^y(s, req)$	The calculated improvement by y for task s in response to the request req made by x	22
IT_x^y	Indirect Trust estimation of y by x	44
$MinUG^y(s)$	The maximum possible proposed utility gain that y may deliver to task s	22

$MinUG^y(s)$	The minimum possible proposed utility gain that y may deliver to task s	22
N_w	The number of consulted witnesses	61
$N_y^x(s,t)$	The number of transaction between x and y during the last Ht time units before t for task s.	85
$pug_x^y(s_{c_i})$	The proportional utility gain change per criterion	95
$R_{K(x,y,c)}$	The set of ratings collected by component K (i.e DT, WT, RT,CR) for calculating $T_{K(x,y,c)}$	47
R_y^x	The time decayed retention of trustor	85
$ret_y^x(s,t)$	The private retention function of x to y	85
RT_w^y	The reported testimony of witness w about trustee y	44
rw	The relative weight	93
$SAT_x^y(s,tra)$	The satisfaction of trustor x as a result of transaction tra with y for task s	92
$T_{K(x,y,s)}$	The trust value of component K (i.e DT, WT, RT,CR) that trustor x has in trustee y with respect to task s.	47
t_{xj}	The time elapsed since the transaction j took place with x	86
TW_x^w	The credibility of their witnesses w	68
$UG_x^y(s,req)$	The proposed aggregated utility gain by trustee y in response to the request req made by trustor x, to do task s	22
$ug_x^y(s_{c_i},tra)$	The utility gain provided by y to x for criterion c_i of task s in the transaction tra	93

v_{r_i}	The value of rating r_i	47
W_I	The weight of for witness category I for $PATROL_F$	43
$wg_x(s_{c_i})$	The weight of criterion c_i of task s as determined by trustor x	93
X'	The set of trustors $\in X$ having interacted with the y so far within the last Ht time units	86
$\overline{Ret(s,t')}$	The average retention rate calculated immediately before $\overline{Ret(s,t)}$ has been calculated	88

Chapter 1

Introduction

1.1 Overview

In human relationships, trust is believed to be a significant factor of social life, and therefore, an important emphasis of sociology [23]. Trust can be associated with scenarios where an action is required on behalf of some other entity, and the fulfillment of such action is vital [124]. Consequently, it is considered to be directed towards the future and involves expectations concerning another entity that it will not try to betray what is entrusted for, despite being able to do so [23]. Trust allows forming expectations about the outcomes of interactions and guiding decisions about what is to be done in those interactions [10]. This implicitly, also, assumes that there is no direct control over the trusted entity; otherwise, the possibility of betraying would not be an issue [124]. Originally, trust was a research topic for theology, psychology, and philosophy [23]. During the last decade of the last century, a high volume of trust research was conducted in sociology, economy, and organizational theory [151]. Recently, working on trust attracted the attention of researchers in the field of computing [153]. While ensuring data integrity and data confidentiality are major objectives of cryptographic security techniques, trust, on the other hand, offers a way to help protect against misuse from other malicious entities [151].

Recently, considerable research in the field of artificial intelligence has been conducted for

designing intelligent software agents to perform problem-solving on behalf of human users and organizations to reduce the processing required [157]. These agents try to collaborate with other agents in the environment, in order to achieve their own objectives. Such agents are generally assumed to be autonomous, self-interested, and goal-driven. Multi-agent systems (MAS) arise when these agents coexist [157]. MASs are an increasingly popular paradigm for the study and implementation of distributed systems that are highly complex and dynamic [157], such as auctions, virtual organizations, and the grid [11].

Trust has been defined in many ways in different domains [91]. For this work the definition used in [50] for trust in MASs, will be adapted. An agent's trust is considered as a measurement of the agent's possibility to do what it is supposed to do. The agent that needs to rely on another agent is referred to as the trustor (also known as service consuming agent in the literature or buyer in the context of e-marketplace). The agent that provides resources or services to trustors is referred to as the trustee (also known as service providing agent in the literature or seller in the context of e-marketplace) [39].

Sen [108] distinguished between trust establishment and evaluation. Where the first aims to help trustees to determine the proper actions to be trustworthy to others in the environment, the second aims to help trustors to evaluate how trustworthy a potential interaction partner (trustee) in the environment. The term trust management considered to be a generic term that describes trust related modelling including trust establishment and evaluation.

We would like to highlight that the term "trust establishment" is used in different ways in the literature of trust management. Many researchers in the domain of ad-hoc networks, such as [104], use the term to refer to trust evaluation of trustees. Others in the domain of service-oriented computing, such as [67], use it to refer to the boot strapping or the cold start problem. With this in mind, few works in the literature addresses "trust establishment" to mean directing trustees to become more trustworthy, as used in this work.

This work addresses trust evaluation and establishment in dynamic multi-agent systems in

situation where a task fulfilled by each trustee is self contained rather than being composed of multiple ”sub tasks” performed other agents. Composite tasks, such as composite web services, are considered outside the scope of this work.

In this chapter, we present the motivations for our work, the problem statement, and an outline for the thesis.

1.2 Motivations

In many systems that are common in virtual contexts, such as peer-to-peer systems, e-commerce, and the grid, entities act in an autonomous and flexible way in order to meet their goals. Such systems can be molded as MASs [91]. Agents can come from any setting with heterogeneous abilities, organizational relationships, and credentials. Furthermore, the decision-making processes of individual agents are independent of each other [10]. As each agent has only bounded abilities, it may need to rely on the services or resources of other agents in order to meet its objects [153]. Agents cannot take for granted that other agents share the same core beliefs about the system, or that other agents are making accurate statements regarding their competencies and abilities. Also, agents must accept the possibility that others may intentionally spread false information, or misbehave in different ways, to meet their own aims [11]. In such environments, trust management is regarded valuable to help agents make decisions that reduce the risk associated with interacting with others in the environment [151]. In order to evaluate how trustworthy a trustee is, a trustor may use its direct prior interaction experience with the trustee [130]. However, when individual trustors are not likely to have enough direct trust evidence with a trustee to generate a reliable evaluation of their trust, trustors may use testimonies from third-party witnesses to help evaluate trust of trustees [153]. Many research works propose models that use the history of interactions with a trustee as a major component for evaluating trust [81]. While doing so, usually, they consider recent interactions more important that old ones [87]. Such mechanisms allow a recently well-behaving trustee, to betray the trust more than once until its long-term trust

evaluation make it undesirable. Responding quickly to dynamic changes, therefore, becomes an important issue that needs to be resolved in order to make effective trust decisions. It is argued that trust mechanisms in multi-agent systems can be used not only to enable trustors to make better trust evaluations but also to provide an incentive for good behavior among trustees [10], [14]. Castelfranchi et al. [14] argued that trusted agents have better chances of being selected as interaction partners and can increase the minimum rewards they can obtain in their interactions. If agents' interactions are based on trust, trustworthy agents will have a greater impact on the results of interactions [108]. The potential cost of acquiring trust may be compensated if improved trust leads to further profitable interactions [108]. In such situations, building a high trust may be an advantage for rational trustees. Though previous research has suggested and evaluated several trust and reputation techniques that evaluate the trust of trustees, slight consideration has been paid to trust establishment [108], which is a principal objective of this work. It is possible that trustors are not willing to provide explicit feedback information to trustees, for many reasons, such as unjustified cost or being self-centred. It is argued in [109] that agents can act independently and autonomously when they do not share problem-solving knowledge. Such agents are not affected by communication delays or misbehaviour of others, and therefore, may result in a more general-purpose MASs. For similar situations, trustees need a trust establishment model that depends on indirect feedback from trustors to help honest trustees enhance their reputation and attract more interactions with trustors, and hopefully achieve more profit, in the long run. Also, the situations where explicit feedback is available from trustors, need to be addressed.

1.3 Problem Statement

Often in MASs, agents must rely on others for the fulfilment of their own goals which may accidentally or intentionally behave in a harmful way. Trustors should be able to address the uncertainty in trustees' behaviour and quickly adapt to environment changes, without scarifying long term goals. Unfortunately, existing models do not allow trustors to respond quickly enough

to reduce effects of misbehaving trustees. We aim to address this shortcoming by combining reinforcement learning with fuzzy logic and the use of temporary suspension.

When interactions are based on trust, trust establishment mechanisms needed to direct trustees, instead of trustors, to build a higher level of trust and have a greater impact on the results of interactions. Existing trust establishment models assume the exitance of explicit feedback, and do not address cases where services are multidimensional with trustors having the option to assign different weights for each dimension. We aim to propose approaches for such situations using the retention function in the absence of explicit feedback and attempting to predict relative weights of different criterion for multi-criteria services.

1.4 Peer Reviewed Publications

1.4.1 Papers in Journals

1. Abdullah Aref, Thomas Tran: An Integrated Trust Establishment Model for Internet of Agents. Submitted to Knowledge and Information Systems

2. Abdullah Aref, Thomas Tran: A Hybrid Trust Model Using Reinforcement Learning and Fuzzy Logic. Computational Intelligence (accepted, October 2017, available online December 2017)

3. Abdullah Aref, Thomas Tran: Multi-criteria trust establishment for Internet of Agents in smart grids. Multiagent and Grid Systems 13(3): 287-309 (2017)

4. Abdullah M Aref, Thomas T Tran: A decentralized trustworthiness estimation model for open, multiagent systems (DTMAS). Journal of Trust Management 2015, 2:3 (11 March 2015)

1.4.2 Papers in Conferences

1. Abdullah Aref, Thomas Tran: Acting as a Trustee for Internet of Agents in the Absence of Explicit Feedback. MCETECH 2017: 3-23

2. Abdullah Aref, Thomas Tran: Modeling Trust Evaluating Agents: Towards a Comprehensive Trust Management for Multi-agent Systems. AAAI Workshop: Incentives and Trust in Electronic Communities 2016

3. Abdullah Aref, Thomas Tran: RLTE: A Reinforcement Learning Based Trust Establishment Model. TrustCom/BigDataSE/ISPA (1) 2015: 694-701

4. Abdullah Aref, Thomas Tran: A Trust Establishment Model in Multi-Agent Systems. AAAI Workshop: Incentive and Trust in E-Communities 2015

5. Abdullah Aref, Thomas Tran: Using Fuzzy Logic and Q-Learning for Trust Modeling in Multi-agent Systems. FedCSIS 2014: 59-66

6. Abdullah Aref and Thomas Tran. A Decentralized Trustworthiness Estimation Model for Open Multi-Agent Systems. Third International Workshop on Incentives and Trust in E-Communities (WIT-EC-14), July 2014

1.5 Organization

The thesis is organized as follows; Chapter 2 will present some background about trust management and other techniques used through our research. Chapter 3 describes selected relevant works. However, it is not meant to be a comprehensive survey by itself. A decentralized trust evaluation model is presented in chapter 4, together with performance analysis. Chapter 5 describes our trust establishment models both using explicit and implicit feedback with performance analysis. Chapter 6 presents a discussion of the work and the last chapter summaries

contributions and discusses potential directions for future research on trust evaluation and trust establishment.

Chapter 2

Background

2.1 Introduction

This chapter introduces some background topics used throughout the remainder of the thesis. Readers who are familiar with any of these topics can skip the corresponding sections, without any loss of information regarding the contributions of this work itself as the information presented here does not cover any of the main contributions of this thesis. Those who are unfamiliar with any of these topics, though, may use this chapter to gain an initial understanding of these topics that will aid in comprehending the work presented in the remainder of this thesis. We start by overviewing the concept of trust (Section 2.2), which is core to this work, then we briefly describe Multi-Agent Systems (MASs) (Section 2.3). Furthermore, we overview reinforcement learning (Section 2.4) and fuzzy logic (Section 2.5); two techniques we used for developing our trust evaluation and trust establishment models. Then, an overview of combing the two techniques is presented (Section 2.6). Finally, Section 2.7 presents the general architecture of the system used in the thesis.

2.2 Trust

2.2.1 Insights into Trust

Trust has been studied in many disciplines, such as psychology, sociology, and economics [157]. The views from different disciplines provide a lot of valuable observations and theories for researchers in computer science [141]. In societies, trust is a reality of daily life [23]. Humans use trust to rationally reason about everyday life decisions complexities [70]. Trust as a social relationship between people is usually addressed by sociologists, whereas economists examine trust from the viewpoint of utility (profit and cost) [141].

Characteristics of Trust

Trust is useful in environments characterized by uncertainty when entities need to depend on each other to achieve their goals. Trust would be of no significance in a perfectly controlled environment, where each entity would commit to the action that should be taken [70].

- Context-dependent: Typically, we trust an expert in his/her area of expertise, but not necessarily in other areas [157]. For example, it is generally acceptable to trust a doctor for health-related issues, but not for car engine problems.

- Subjectivity [70]: The evaluation of trust depends on how the behaviours of trustees are perceived by the trustors, such a perception often depends on some internal characteristics of the trustor [151]. Suppose x_1,y and x_2 are agents in a MAS; if x_1 trusts y to do task S, it is not necessary that x_2 trusts y to do the same task, even though the behaviour of y is the same. For example, one week can be considered as "fast enough" delivery of a product to one customer, but it may be considered too late for a different customer of the same product.

- Unidirectional [70]: The subjective nature of trust implies that it is unidirectional; that is if x trusts y, it is not necessary that y trusts x in the same way.

Why Trust at All?

Trust enhances cooperation and makes it less complicated because it removes the incentive to check up on others [10] and, therefore, can reduce the cost [23]. That is, if x has high trust in y, x can use simple or no control mechanism when interacting with y, but if x has low trust in y, then x may decide to invest more monitoring and controlling y for an interaction [10]. By enabling the selection of interaction partners with a higher potential for achieving the desired outcome, trust can reduce the risk of harmful or non-beneficial interactions [23]. Also, trust can be used for complexity reduction by enabling the division of complex tasks among a set of entities [23].

Motivation for Being Trustworthy

In economics, it is argued that what can be gained through fulfillment of contracts can never be gained by fraud [23]. A trustee may act trustworthy for selfish reasons because of fear of penalty mechanisms, or because of a large potential for gain in the longer term. Another motivation can be whether a trustee can see the overall advantage and will take into account common benefit(s) [23]. In the context of multi-agent systems, Sen [108] argues that if agents' interactions are based on trust, trustworthy agents will have a greater impact on the results of interactions. In such situations, building a high trust may be an advantage for rational trustees [108].

2.2.2 Trust in computing

Trust in Keys

One of the earliest areas of information technology where the concept of trust used is public key cryptography, where trust refers to being sure of the identity of a certain user [57]. When public key cryptography is used, a user owns a pair of keys; a public one and a corresponding private one. The private key is kept secret for the user, whereas the public one is publicly published. Data encrypted with one key can be decrypted with the corresponding one [98]. Trust is highly related to binding a public key to its owner's identity. Digital certificates are used for this purpose. Trusted third parties are entities authorized to issue such certificates, and the issue of whether a certain key really belongs to a particular user is replaced by finding out whether the issuer of the certificate is sufficiently trusted to make this statement [23].

Trust in Platforms

A hardware-based trusted computer platform is a simple, passive, integrated circuit chip designed to enhance the degree of trust that a user has in a particular aspect of computing systems and networks and to create a framework for trusted applications [116] The chip is intended to be a trustworthy device for certain low-level security services, such as device identity, and software integrity. When a trusted platform boots up, the chip can measure the integrity of each step of the boot process and, therefore, determine whether or not to proceed forward to the next step [23].

Trust in Multi-Agent systems

As pointed out earlier, MASs are widely used for the study and implementation of distributed systems and virtual online communities. Trust has long been recognized as a vital field of interest in MASs, where agents are self-interested, diverse, and deceptive. In these systems, agents have to accept the possibility that their peers may be intentionally, or accidentally, misleading,

spreading false information, or otherwise behaving in a harmful manner, in order to achieve their own goals [10]. Trust allows agents to resolve some of the uncertainty in their interactions, and form expectations about the behaviours of others [50]. Furthermore, it is considered the parameter that reflects agents' risk level when they want to rely on the provided information or service of other agents for the fulfillment of their own goals [49].

2.3 Agents and Multi-Agent Systems

2.3.1 Agents

An agent is an intelligent entity that is capable of autonomous action in some environment, in order to achieve its delegated goals [145]. Being autonomous is the fundamental point about agents, as they are capable of independent action [145]. A particular agent uses its built-in capabilities to maintain an ongoing interaction with its environment reactively or proactively, using a decision selection mechanism [50]. Generally, agents have limited capabilities, which constrains them to working with other agents to accomplish complex tasks [49].

Based on the modular architecture of agents described in [108], we assume that each agent composed of multiple modules or components, such as the communication module. One of these modules is the trust management module or component. The trust management module stores models (not modules) of other agents and interfaces with both the communications module and the decision selection mechanism. The trust management module itself is composed from multiple modules such as the trust evaluation module and the trust establishment module. Based on the [108], trust management module is composed of:

- Evaluation module: This component is responsible for evaluating the trust of other agents using different information sources such as direct experience and witness testimonies. Trust models described in [130], [102, 123, 150] are well-known models that belong

mainly to the evaluation component. The proposed model in Chapter 4 designed to address trust evaluation.

- Establishment module : This component is responsible for determining the proper actions to establish the agent to be trustworthy to others. The work of Tran, et al. [129] is an example of a model designed mainly to address this component. The proposed models in Chapter 5 designed mainly to address trust establishment.

- Engagement module: This component is responsible for allowing rational agents to decide to interact and engage with others with the aim of evaluating their trust. In the literature, this component is usually referred to as trust bootstrapping and the cold start problem. Bootstrapping Trust Evaluations through Stereotypes [8] is an example model that can address engagement.

- Use module: This component is responsible for determining how to select prospective sequences of actions based on the trust models of other agents that have been learned.

2.3.2 Multi-Agent Systems

A multi-agent system is composed of multiple interacting intelligent agents, which interact with each other [145]. MASs are broadly used in distributed environments [157]; they are used as an alternative to centralized problem solving [49]. A common reason for this is that problems are themselves distributed. Alternatively, the use of multi-agent systems may result a more efficient way to organize the process of problem solving [49]. Analyzing multi-agent systems is like analyzing human or animal behaviours in the sense that they are self-independent, and their selfish actions can influence the environment in unpredictable ways. The main criteria obtained by developing MASs is flexibility in proactive environments. Due to self-managed nature of MASs, they are considered to be rapidly self-recovering and failure-proof [49].

2.4 Reinforcement Learning

Reinforcement Learning (RL) attempts to solve the problem of learning from interaction to achieve an objective. An agent starts by observing the current state u of the environment, then performs an action on the environment, and later on receives feedback f from the environment. The received feedback is also called a reinforcement or reward. Agents aim to maximize the cumulative reward they receive [4]. There are three well-known, fundamental classes of algorithms for solving the reinforcement learning problem, namely dynamic programming, Monte Carlo, and Temporal-Difference (TD) learning methods. Unlike other approaches, TD learning algorithms can learn directly from experience without a model of the environment (contrary to Dynamic Programming) and are incremental in a systematic sense (contrary to Monte Carlo methods). However, unlike Monte Carlo algorithms, which must wait until the end of an episode to update the value function (only then is the return f known), TD algorithms only need to wait until the next time step. TD algorithms are considered incremental in a systematic sense [130].

One of the most widely used TD algorithms is known as the Q-learning algorithm. Q-learning works by learning an action-value function based on the interactions of an agent with the environment and the instantaneous reward it receives. For a state u, the Q-learning algorithm chooses an action a to perform such that the state-action value $Q(u, a)$ is maximized. If performing action a in state u produces a reward f and a transition to state u', then the corresponding state-action value $Q(u, a)$ is updated accordingly. State u is now replaced by u' and the process is repeated until reaching the terminal state [130]. The detailed mathematical foundation and formulation, as well as the core algorithm of Q-learning, can be found in [120] therefore, it is not repeated here. Q-learning is an attractive method of learning because of the simplicity of the computational demands per step and also because of proof of convergence to a global optimum, avoiding all local optima, as long as the Markov Decision Process (MDP) requirement is met; that is, the next state depends only on the current state and the taken action (it is worth noting that the MDP requirement applies to all RL methods) [28]. It is this special characteristic of reinforcement

learning that makes it a naturally suitable learning method for trust-evaluating agents in MASs [127]. A MAS can be considered as an uncertain environment, where the environment may change anytime. Obviously, RL considers the problem of an agent that learns from interaction with an uncertain environment in order to achieve a goal, where the learning agent needs to discover which actions yield the most reward via a trial-and-error search instead of being advised which actions to take as supervised machine learning. It is this special characteristic of RL that makes it a naturally suitable learning method for trust evaluation and establishment in MASs. Furthermore, the suitability of RL can also be seen if we note that a trustor observes the trustees, selects a trustee, and receives the service from that trustee. If we take a close look at the activities of a trustee in a MAS. The trustee observes trustors' feedback (direct or indirect), proposes utility gain in response to collaboration requests of trustors, interact with a trustor and receives some utility from that transaction. In general, these agents get some input, take an action, and receive some feedback. Which is the same framework used in reinforcement learning.

2.5 Fuzzy Logic Systems (FLSs)

Fuzzy logic has been extensively applied with success in many diverse application areas due to its similarity to human reasoning, and its simplicity [28]. Fuzzy logic systems offer the ability to handle uncertainty and imprecision effectively and, therefore, have been considered ideally suited for reasoning about trust [33]. Fuzzy inference deal with imprecise inputs and allows inference rules to be specified using imprecise linguistic terms, such as "very high" or "slightly low"[33]. A Fuzzy logic system provides a nonlinear mapping of crisp input data vector into a crisp scalar output. A Fuzzy logic system has four components: fuzzy logic rules, a fuzzifier, an inference engine, and a defuzzifier [74]. Fuzzy logic systems have been used successfully in trust evaluation modeling in [12, 96, 117, 122] among others. In fuzzy logic systems, sets used for expressing input and output parameters are fuzzy, and they are based on the concept of a Membership Function (MF) which defines the level to which a fuzzy variable is a member of

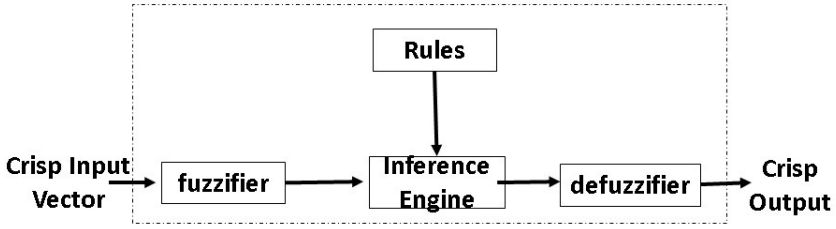


Figure 2.1: Fuzzy Logic System [74]

a set [33]. One represents the full membership, whereas zero represents no membership [33]. A MF provides a measure of the level of similarity of an ingredient to the fuzzy subset. An ingredient can reside in more than one fuzzy set to varying levels of association, which can't happen in crisp set theory. Triangular, trapezoidal, piecewise linear and Gaussian, are commonly used shapes for MFs [74]. Fuzzy logic rules may be implemented by experts or can be derived from numerical data. In either case, fuzzy logic rules are represented as a collection of IF-THEN statements. MFs map input values into the interval [0,1] by the process known as "fuzzification" [33]. The fuzzifier maps crisp inputs into fuzzy sets, to stimulate rules which are in terms of linguistic variables. Fuzzy logic rules define the relationship between inputs and outputs. The inference engine handles the way in which rules are combined. The membership levels of the conclusion are aggregated by superimposing the resultant membership curves. The defuzzifier maps fuzzy output sets into crisp numbers to be collected at the output [74]. Figure 2.1 presents the general architecture of a fuzzy logic system. For each fuzzy rule, the inference engine determines the membership level for each input, then measures the degree of relevance for each rule based on membership levels of inputs and the connectives (such as AND, OR) used with inputs in the rule. After that, the engine drives the output based on the calculated degree of relevance and the defined fuzzy set for the output variable in the rule [84]. The Mamdani min-max method [68] is a well-known direct inference method, where the degree of membership of rule conclusions is clipped at a level determined by the minimum of the maximum membership values of the intersections of the fuzzy value antecedent and input pairs. This ensures that the degree of membership in the inputs is reflected in the output [33]. In this work, Mamdani's method is

used. Several defuzzification methods have been proposed in the literature. The centroid, also known as center of gravity, defuzzification is an appealing one, which has been used widely in the literature [100]. The method takes the center of gravity of the final fuzzy space in order to produce an output sensitive to all rules. A disadvantage of this method is that it is computationally expensive [84]. The survey presented in [101] provides more insight on existing work in the field of defuzzification techniques. In this work, the centroid defuzzification method is used.

2.6 Fuzzy Q-Learning

Amongst neural networks, evolutionary computation, and RL, the last one is considered the preferred computational intelligence technique to work with FLS for trust evaluation on MASs. Neural networks need a large diversity of training for real-world data, and they require a lot of computational resources and high processing time for large neural networks [79], which is not necessarily available for trust evaluating agents. Genetic drift is considered a major challenge for using genetic algorithms technique [79]. Obtaining a training data set that is representative of all situations in MASs can difficult task. Q-learning does not depend on the use of complete and accurate knowledge to take proper actions, and therefore, agents can learn from their own experience to perform the best actions [130]. Unfortunately, the optimization process can be sensitive to the selection of the reinforcement signal and the fact that the system states must be visited a sufficient number of times, which is alleviated by the use of temporal suspension described earlier.

The combination of fuzzy logic and RL can be used to improve the behaviour of FLSs and enables incorporation of prior knowledge in the fuzzy rules to speed up the learning process [79]. A Fuzzy logic-based Q-Learning algorithm (FQ-Learning) is proposed in [6] to extend the applications of Q-Learning to fuzzy environments in which goals and/ or constraints are fuzzy themselves [6]. In the context of trust evaluation, goals can be fuzzy such as trust estimation is high or low.

FQ-Learning extends the original $Q(u, a)$ value to $FQ(u, a)$, where actions can have fuzzy constraints [6]. The algorithm chooses for each state u, to perform action a that attempts to maximize the fuzzy state-action value $FQ(u, a)$. When action a is performed in state u to produce a transition to state u' and a reward f, the corresponding fuzzy state-action value $FQ(u, a)$ is updated to reflect the change of state, such that state u is now replaced by u' and the process is repeated until reaching the terminal. Unfortunately, as with Q-Learning, the fuzzy extension may suffer from slow learning rate to some extent [6]. The detailed mathematical foundation and formulation, as well as the core algorithm of FQ-learning, can be found in [6] therefore, it is not repeated here. The combination of FLSs and Q-Learning algorithm used successfully in [79, 78] for dynamic traffic steering and load balancing in wireless networks as well as in [16] for admission control in heterogeneous wireless networks. A Hierarchical extension described in [19].

2.7 System Overview

In this section, we will present some common notation, and outline the necessary components and assumptions we make about the underlying trust establishment model, which will be referred to in the remainder of this chapter.

2.7.1 Agent Architecture

Based on agent's architecture described in [108], we assume that each agent has an embedded trust management module. This module stores models of other agents and interfaces with the communication module and the decision selection mechanism. The trust management module includes an evaluation module, which is responsible for evaluating the trust of other agents, and an establish sub-components, which is responsible for determining the proper actions to establish the agent to be trustworthy to others.

2.7.2 Agents, Tasks, and Interactions

Agents are autonomous software components that can represent humans, devices, an abstract service provider, or a web service. We assume a MAS consisting of a set of m interconnected, possibly heterogeneous, agents A = $\{a_1, a_2, ..., a_m\}$. Communication protocols used for implementing MAS is outside the scope of this work. Each agent has its own trust management module, responsible for all trust related modeling. A task is what a trustor wants the trustee to do. On an abstract level, a task can be a service or a capability that is offered by a trustee, such as pieces of information, a file in peer to peer systems, or goods offered by an online shop. Interactions are actions of performing tasks. When both interaction parties have an expectation of the outcome, the interaction is referred to as a bidirectional interaction. For example, when a trustee y sells something to trustor x, x has an expectation about the provided product or service by y, and y has an expectation of receiving compensation or rewards, such as money, in return. On the other hand, when trustors only have expectations about the outcome of interactions, such interaction referred to as a uni-directional one. For example, in a peer to peer file sharing scenario, only the trustor that receives the file has an expectation about the quality of the received file, but not the trustee [97].

Each trust management module has a trust evaluation module and a trust establishment module. Figure 2.2 presents a general agent architecture with trust management module. Agents are independent, mobile, if needed, autonomous enough to identify specific actions to take, and sufficiently intelligent to reason about their environments and to interact directly with other agents in the MAS. Also, we assume a set of possible tasks S =$\{ s_1, ..., s_k\}$,

Each task $s \in S$ has a set of criteria. The set of criteria is denoted as C =$\{ c_1; c_2; ...c_p\}$. Individual trustors may have various levels of demand for each criterion, such as response time or transaction cost. Furthermore, individual criterion of a service may be of different importance or weight such that WG =$\{wg_1, ..., wg_g\}$ where $0 \leq wg_i \leq 1$. This allows a trustor to have a mixed configuration of demand and importance. For example, a trustor may be highly demanding

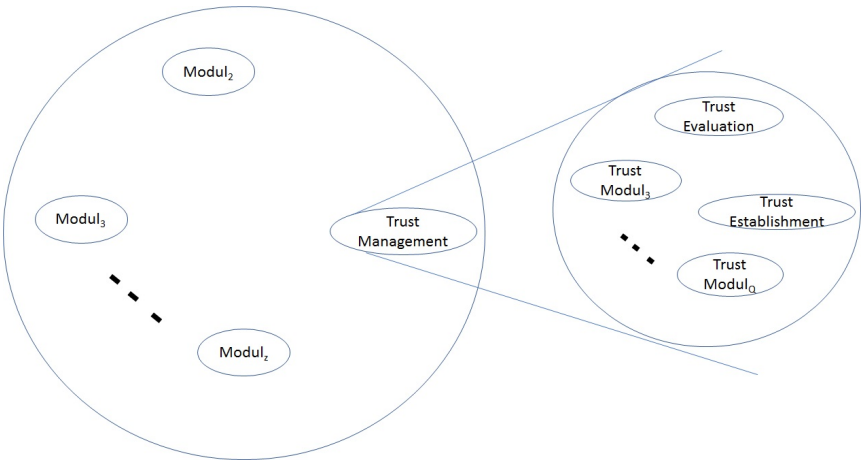


Figure 2.2: General Agent Architecture with Trust Management Module [108]

with respect to a criterion, such as response time should be within one seconds, but this criterion is not so important compared to other criteria such as stability or environmental impact. The nature of tasks in S is application dependent, and individual trustees may be able to perform zero or more tasks. Also, individual trustors may be interested in zero or more tasks, any number of times.

2.7.3 Roles

Agents in the system can be trustees or trustors. Trustors are agents who consider relying on others to fulfill a task. Such roles are not mutually exclusive, an agent can be both a trustor and a trustee [39]. Trustors who report their experience of interacting with trustees are referred to as witnesses. In this chapter, the set of trustors is referred to as $X = \{ x_1, ..., x_n\}$ and the set of trustees as $Y = \{ y_1, ..., y_q\}$ such that $X \cup Y \subseteq A$. The set of witnesses $W = \{w_1, ..., w_g\}$ such that $W \subseteq X$. Figure 2.3 presents a simplified scheme for roles of agents.

2.7.4 Process Overview

In this subsection, we provide an overview of the process of trust evaluation of potential interaction partners and selecting a trustee based on trust. A trustor $x \in X$ that desires to see a task

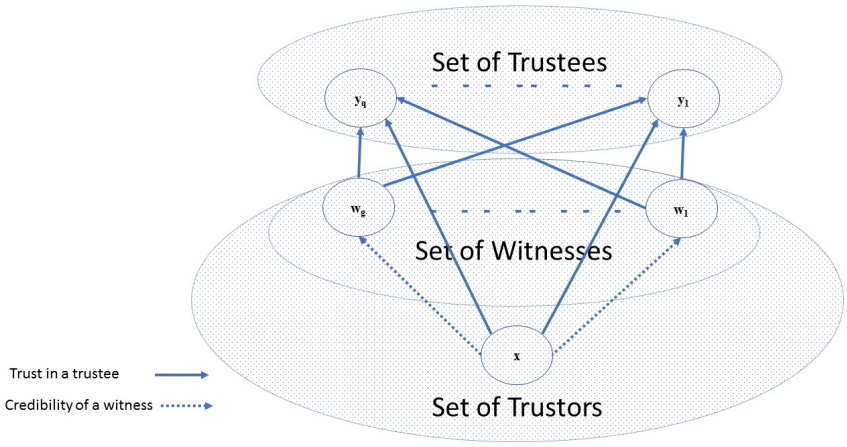


Figure 2.3: Simplified scheme for roles of agents

$s \in S$ accomplished, considers depending on a trustee $y \in Y$ to perform the task on its behalf. For this purpose, x may consult some witnesses.

Locating Trustees

Trustors need to know how to find trustees in a MAS before interacting with any of them. MAS may used a directory service for registering trustees. Alternatively, trustees may choose to advertise their selves in their communities. Locating trustees is assumed to be outside the scope of this work.

Locating Witnesses

Witnesses locating was described in [81] to reflect a mechanism for locating third-party information sources. Few general-purpose trust models, such as FIRE [39] use a distributed approach for this purpose. Naïv broadcasting of witnesses' information, system wide inquiry about witness, and referral networks are among approaches used to locate witnesses. A semi-hierarchical structure, coupled with the notion of the "small world" [143] can be used to help reducing the communication overhead associate locating witnesses. Locating witnesses is assumed to be outside the scope of this work.

Evaluating Trust of Trustees

Trustor x considers the evaluation of direct evidence and testimonies from witnesses about trustees. This includes the utility gain of an interaction by considering the expected benefits that are associated with the interaction by the trustor. In this work, the cost of not interacting is assumed to be too high and should be avoided. The selection of an appropriate trustee may be based on various strategies, such as using trust score only, or using expected utility gain. In this work it is assumed that the trustor selects the candidate with the best expected utility gain.

Interaction

When a trustor x decides on which trustee y to interact with, x requests to interact with y. In response to req made by trustor x to do task s at time instance t, trustee y proposes to deliver a Utility Gain (UG) for x by a transaction as follows

$$UG_x^y(s, req) = Improvement_x^y(s, req) + MinUG^y(s) \tag{2.1}$$

- $UG_x^y(s, req)$ is the proposed aggregated utility gain by trustee y in response to the request req made by trustor x, to do task s.
- $MinUG^y(s)$ and $MaxUG^y(s)$ are the minimum and maximum possible proposed utility gain that y may deliver to task s.
- $Improvement_x^y(s, req)$ is the calculated improvement by y for task s in response to the request req made by x. Where

$$0 \quad \leq \quad Improvement_x^y(s, req) \quad \leq \quad (MaxUG^y(s) \; - \; MinUG^y(s)) \tag{2.2}$$

Satisfaction Evaluation

After an interaction, x evaluates how satisfying the transaction was, and update the trust evaluation of y, and potentially, the credibility of witnesses. The satisfaction evaluation is added to the evidence from past interactions, which allows for building a history from the behaviour of a trustee. The aggregation of satisfaction over previous interactions represents the direct trust evaluation of a trustee. Because of the distributed nature of MAS, no central entity or authority exists to facilitate trust-related communications. Consequently, each trustor manages the histories of all the interactors and witnesses it has previously encountered by itself.

Distribution of Interaction Result

Trustors may be willing report how satisfying was the interaction was with a trustee. For centralized systems, like eBay and Amazon, this is simply reported to the central authority, and becomes available to all other agents. In distributed architectures, like peer to peer systems and web services, it can be provided in response to requests from other trustors in a reactive way or, alternatively, sent as a broadcast to every member in the system in a proactive way. However, the later option can be highly ineffective in terms of messages exchange.

Figure 2.4 summarize these steps for the process of trust evaluation

2.7.5 Basic Assumptions

This section presents a set of basic assumptions used for designing the proposed model in the study.

- Locating trustees: trustors need to know how to find trustees in a MAS before interacting with any of them. It is assumed that a mechanize for locating trustees exists, and agents are willing to use it.

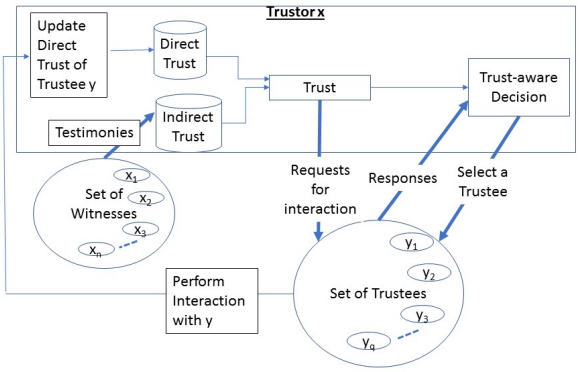


Figure 2.4: General Trust Evaluation Process

- Interactions between agents: A fundamental assumption is that agents have an interest and the possibility to interact with others, as it is the reason for evaluating trust.

- Collusion and composite services: We assume no collusion attacks or composite services. This mean no group of gents in the system coordinate to harm other agent(s). Also, a trustee can fulfill a task by itself, rather than further delegating the task fully or partially.

- Distinguishable agent: Agents assumed to be recognizable by others, and evidences from previous interactions can be linked to agents. Typically, this can be accomplished by having a unique identifier for each agent that cannot be used by any other agent.

- Satisfaction computing: It is assumed that after an interaction, the trustor is capable of computing its own satisfaction of an interaction, which is necessary in order to track trust evaluation.

- Storing satisfaction results: Agents assumed to have the capabilities for storing satisfaction results derived from past interactions. To reduce the memory requirements, trustors may choose to aggregate such values from previous interactions per trustee.

- Possibility of repeated interactions: The environment assumed to allow for repeated interactions between agents. While this sounds trivial at first glance, there are different

scenarios in which this assumption is not met [97]. For example, in situations where the cost of joining a MAS is negligible, agent may exit and re-enter the system using different identity; a security attack known as white washing.

- Unlimited Processing Capacity: We assume that trustees can service an infinite number of requests from trustors during a time unit without negatively affecting their performance. This assumption is referred to as the Unlimited Processing Capacity, which may be justified in computerized services or resources [155].

2.8 Summary

This chapter presented an overview of topics used throughout the remainder of this thesis, with references to further reading for those interested. Based on background topics discussed in this chapter, we present our trust evaluation model in the next chapter and our trust establishment models in the one that follows.

Chapter 3

Literature Review and Related Work

3.1 Introduction

Recently, the scientific research in the field of trust evaluation modelling considerably increased [87] and resulted in a variety of trust models that can be used in a wide range of situations [103]. Most models are domain specific, context dependent, and often use particular variables from the environment. For example, some models developed for e-commerce use variables like price, quality, or delivery time which may be useful in this context, but not necessarily in the context of Vehicular Ad Hoc Networks (VANETs) where congestion and safety may be more appropriate [31].

In (Section 3.2) we do not aim to provide a comprehensive systematic review of existing trust models, instead we aim to adapt a general classification of trust evaluation models and a review for selection of notable models related to our work.

It is argued that trust mechanisms in multi-agent systems can be used not only to enable trustors to make better trust evaluations but also to provide an incentive for good behaviour among trustees [11, 14]. Castelfranchi et al. [14] argued that trusted trustees have better chances of being selected as interaction partners and can increase the minimum rewards they can obtain

in their interactions. Due to the relatively small amount of directly related work to trust establishment, in Section (3.3) we attempted to situate our work within the relatively larger context of trust management and present a review for selection of trust establishment models related to our work.

3.2 Trust Evaluation

3.2.1 Classification Dimensions

Trust models can be analyzed from different perspectives [103]. Various dimensions to classify and characterize trust models have been presented in different surveys, reviews, and theoretical evaluation frameworks exist in the literature [54]. Some of them are based on online reputation systems such as [41], others based on trust in peer-to-peer networks such as [53], or VANETs such as [158]. Some reviews focus on concrete aspects or functionalities like attack and defence techniques [36]. Others are more general like [32, 77, 103, 35, 62, 81, 87, 153, 31]. A recent review of reputation system evaluations approaches presented in [54]. One of the most cited and used general purpose surveys is the one developed by Sabater and Sierra [103]. It has been used as a base in other reviews such as [87, 31]. The proposed dimensions of analysis were prepared to enhance the properties of the Regret model [102] and show the basic characteristics of trust models and considered a good starting point [87].

Classification Dimensions in The Literature

A considerable number of dimensions for analysis presented is a set of well-known reviews in the literature. Each one of these frameworks takes into account several aspects to categorize the models, and the authors define a set of characteristics aiming to indicate how trust evaluation models in the literature behave according to those characteristics. Here, we list classification dimensions presented is a set of well known reviews in the literature. In the next subsection we

present an aggregated list of classification dimensions adapted from those in the litterateur, and highlight different terms used in different studies to describe similar dimensions.

- Grandison and Sloman [32]
  - Provision trust: address situations where a trustor relies on a trustee to provide or deliver a set of services or sources [32].
  - Access trust: addresses the concern of resources' owners for allowing a particular partner to access their resources or not [41].
  - Certification, authentication, or identity trust: describes the belief that the identity of an trustee is as claimed [41].
  - Infrastructure, system, or context trust: address concerns of the trustor that the necessary system components are in place to support the interaction [41].
- Sabater and Sierra [103]
  - Conceptual model: refers different methods used build trust models, these methods referred to as computation engines [81], conceptual model in [103] or paradigm type [31].
  - Information sources: refers to sources used for trust related information such as direct experience and testimonies form witnesses
  - Visibility: indicate weather feedback of trustors regarding an interaction with trustees can be seen as a global property shared by all trustees or as a subjective property assessed particularly by each individual [87]
  - Granularity: refers to the level of details addressed by trust models.
  - Cheating assumptions: refers to whether or not the model assumes agents capable or willing to cheat [102].

  - Type of exchange information: refers to the type of information expected from witnesses such as Boolean or real numbers.
  - Reliability measure: refers to whether or not a trust model provides a measure of how reliable is the trust evaluation.

- Pinyol and Sabater [87]
  - Trust: The authors believe that "trust implies a decision", consequently they consider a model that provide evaluation without a decision as non-trust model or simply a reputation one.
  - Cognitive: refers to whether or not a model has clear representations of trust in terms of cognitive elements such as beliefs and goals.
  - Procedural: refers to whether or not a model has explanations on how it bootstraps.
  - Generality: refers to whether or not a model is a general purpose.

- Lu et al [62]
  - Semantics: whether the model address access trust, certification trust, provision trust, or infrastructure trust.
  - Architecture refers to the way agents are interconnected and describes whether a model assumes a centralized, distributed or a hybrid architecture.
  - Mathematical model: this is similar to the conceptual model used by [103].
  - Trust network: refers to relationships between agents i the system and whether ot not the model use them.
  - Reliability: as with [103].

  - Risk: refers to whether or not a model takes risk int account before selecting a trustee to interact with.

- Noorian and Ulieru [81]

  - Rating approaches: this is similar to granularity described in [103].
  - Witness Locating Approaches: refers to how a trustor can find witness to consult about a trustee.
  - Reputation Computation Engine: this is similar to the conceptual model used by [103].
  - Information Sources: this is similar to the corresponding one used by [103]. Context Diversity Checking: refers to whether or not a model can check different contexts and preferably map between them.
  - Adaptability: refers to bootstrapping and dynamic response to changes in the system.
  - Reliability and Honesty Measurement: as with [103].
  - Basic reputation measurement parameters: refers to some parameters such as time and transitivity.

- Ramchurn et al. [91]

  - Socio-cognitive models: refers to whether or not models use social relationship among agents and their roles in the community, which influences relationships with others.
  - Reputation models: refers to whether or not models use testimonies from witnesses.
  - Evolutionary and learning: refers to whether or not models can be learned or evolved over multiple interactions.

- Truth-eliciting interaction protocols: this refers to interaction protocols with rules to create situations where agents have no better choice than being honest.
- Reputation mechanisms: this refers to mechanisms used to encourage witnesses to provide testimonies or create reputation scores for trustees.
- Distributed security mechanisms: this is similar to access, certification and infrastructure trust described in [32].

An Aggregated List of Classification Dimensions

In this section, we did not consider all the dimensions proposed by the authors in the literature; because some of them are repetitive among the papers or they are not applicable to our study. The following list based mainly on the work of [31] and [103].

- Trust Class

  In order to be more specific about trust semantics, we will distinguish between a set of different trust classes according to Grandison et. al. [32], who differentiated between access trust, certification trust, provision trust, and infrastructure trust. While the first two are of interest for computer information security communities [31], the last one is of interest for system engineers, whereas the provision trust considered the focus of the trust models described in this study.

  - Provision trust address situations where a trustor relies on a trustee to provide or deliver a set of services or sources. Provision trust can be used to help trustors avoid malicious or unreliable partners. Trusting a partner to act and make decision on behalf of a trustor is considered a special form of service provision referred to as delegation trust [32].
  - Access trust addresses the concern of resources' owners for allowing a particular partner to access their resources or not [41].

- Certification, authentication, or identity trust describes the belief that the identity of an trustee is as claimed [41]. Pretty Good Privacy (PGP) [27] is a well-known example of identity trust.

- Infrastructure, system, or context trust address concerns of the trustor that the necessary system components are in place to support the interaction. Things like insurance, legal system, law enforcement and stability of society in general can be considered as an example of such components [41].

- Architecture

  From an architectural point of view, we can differentiate between centralized and distributed trust evaluation models. Typically, a central entity manages the trust and reputation related information of all agents in the first approach, but each agent performs its trust related computations without a central entity in the second approach.

  In the centralized approach trust related information are publicly seen by all the members fo the society [103] where trustors report the performance of their transaction partners to a central entity which is regarded as an authority [41]. Such authority is assumed to be truthful and available upon request [142]. The cents of the society [103] where trustors report the performance of their transaction partners to a central entity which is regarded as an authority [41]. Such authority is assumed to be truthful and available upon request [142]. The centralized architecture is widely used in online auction sites like eBay and Amazon where buyers learn about the trustfulness of sellers before initiating direct interaction [81]. Whether the central authority implements a calculation engine to derive the final trust evaluation for trustees as in [43, 156] or individual trustors derive final trust evaluation of trustees based on information from the central authority, the architecture remains a centralized one.

  The centralized approach is considered practical when all members of a community recognize and trust the central authority; which is not necessarily the case of Multi-Agent

Systems (MASs), where no central authority control all the agents in an MAS or when agents may question the credibility of such centralised entities and decide not to use them [39]. Moreover, a centralized architecture contradicts with the characteristics of a dynamic environment where the population of agents can vary over time and subsequently when the number of agents grows in a distributed environment, the operational cost of reputation models can be large [81]. Typically, single points of failure and performance bottlenecks are major concerns for centralized model [142], when the central authority is not directly involved in transactions, the problem of misleading testimonies is exaggerated in centralized approaches, and malicious agents can trick the system [81]. Therefore, such mechanisms are not well suited to computational agents, which must usually make decisions autonomously [39].

When testimonies from witnesses are aggregated equally, as in typical centralized models, the models cannot adapt well to changes in agents' performance, whether these changes were accidental or strategic [39].

Generally speaking, decentralized models are more complex than centralized ones. Each trustor is capable of storing trust related information locally and derive the trust evaluation for trustees as needed, using its own computation engine tailored to its requirements without depending on a central entity for managing information about the performance of trustees [41]. Such approach can provide good scalability and robustness [142]. The avoidance of single point of failure assures the accessibility of services in all situations [81].

However, such decentralized environment also creates challenges of weight washing where malicious agents can gain benefits without being identified and punished. For example, a dishonest service provider may gain competitive advantage by providing false information about its service quality [142].

Hybrid approaches attempt to combine the benefits of both approaches, such as the model

described by Wang et al. [141, 142]. Their model attempts to overcome the single point of failure problem using sets of super agents or brokers responsible for trading trust related information upon request [142]. Unfortunately, the performance of super agents may vary, and trustors should be given techniques for distinguishing between honest and fraudulent super agents. Compared with the centralized models, hybrid models can perform better in dynamic environments. However, when the population size grows very large, hybrid models may suffer from the high costs of locating super agents [81].

- Information Sources

  In dynamic environments, information required to calculate trust evaluation may be scarce. Therefore, many trust models take into account a variety of sources for trust information [39]. It is possible that not all trustors use all information sources, as it will depend on the type of application and the trustor's architecture. Widely used sources of information include:

  1. Direct experience: This is one of the most valuable sources of information for trustors, where a trustor uses its previous experiences in interacting with a trustee to evaluate its trust [103]. It is referred as interaction trust in [39]. Trust models like [70, 130] are among those that use only direct experience for trust evaluation. Because direct experience alone cannot be used to predict how trustworthy a trustee is when a trustor has never interacted with the trustee, many trust models rely on other information sources [31].

  2. Direct Observation: Unlike direct interaction, there is no direct interaction between the evaluator and the trustee to compute trust, rather than that, the evaluator predicts the trustee's performance based on observation of others' interactions with the target [103]. Trust models described in [13, 95, 51] perform direct observation through characteristics perceived in a trust situation, such as the percentage of positive reviews. Salehi and White [106] presented a model that prevents the exposure of the

observed trustee by allowing the observation of direct interaction only to the confident neighbours. A variation of observation is referred to as prejudice, which relates to criteria or signs that identify individuals as members of groups [103], which are context dependent [31]. The trust models presented in [9] uses observation of characteristics to build stereotypes of trustees to address the cold start problem whereas models like [102] use institutional structures to address the same problem.

3. Witness Information: Like direct observation, no direct interaction between the trustor and the trustee is necessary to compute trust, where evaluators ask others about the trust of the target that use witnesses [31]. The aggregation of testimonies from witnesses who have had some interaction with the target is widely referred to as reputation in the literature. Some trust models like [7, 52, 60] use pure reputation information. However, many others, such as [123, 43, 4], use information form witnesses together with information from other sources.

4. Certificates: In the literature two variations of certificates can be differentiated based on the issuer: 1) Certified reputation [38] and 2) Endorsement [71]. Certified reputation is similar to traditional recommendation letter from a professor about a student which can be used to clarify the credibility of the student. It occurs when a trustee has a list of witnesses who can testify about itself [38, 39]. To prevent fraud, encryption and digital signature can be used. The main benefit of certified reputation would be the low cost to find witnesses, which can be a performance problem for models that use witnesses' information to compute trust. On the other hand, an authority, or a well trusted entity, can endorse trustors by giving approval or support, in a way similar university certificates. This can be very useful for new comers to bootstrap their reputation [71].

5. Sociological Information: Similar to direct interaction, there must be many interactions between agents because it requires social network analysis, this source of

information relates to the social relationship among agents and their roles in the community, which influences relationships with others [103]. Regret [102] implements graphs called sociograms to indicate the relationship among entities. Sociograms group entities to get information from those who are more representative. Models like [51] focuses on the trusted community concept using adaptive agents, since high degrees of confidence are expected from members belonging to the same community. Others like [119, 39] base trust on social structures as friendship and familiarity. In the literature, sociological information also referred to as rules [31, 39].

- Computation Engine

  Trust models attempt to model the behaviour of potential interaction partners before actual interactions. For this purpose, they may exploit different methods to build such model, these methods referred to as computation engines [81], conceptual model in [103] or paradigm type [31]. In general, trust models can be classified as numerical, cognitive, or hybrid [31]. Numerical models do not have any explicit representation of cognitive attitudes to describe trust. On the other hand, cognitive build trust and reputation based on beliefs and their degree [87]. Whereas hybrid models attempt to integrates both approaches [31].

  The model described in [24] based on trust as a result of a mental state composed of some cognitive components, such as objective, ability, competence, and dependence. The model described in [13] takes into account the combination of self-esteem, reputation, and familiarity among the agents. The model presented in [119] introduces friendship concepts such as best friends and casual friends. This model is based on trust expression through emotions. The model by Singh [113] uses the BDI architecture to support intuition and the semantic concept of a rich vocabulary to represent the evaluations. The BDI and Repage model [88] includes concepts of image as a composition of the set of beliefs that are based on specific goals independent of the reputation value. An agent may have a low reputation value related to a certain dimension, but its image may be good for other agents who

see it from another perspective. Other examples about cognitive based models include [90, 133, 15] which are based on the work of [24], and cognitive maps are used as the trustors reasoning mechanism.

Numerical trust models do not have any explicit representation of cognitive attitudes to describe trust, but they use numerical aggregation of past interactions and presents a set of subjective probabilities that trustees will correctly perform as expected. Some numerical trust models, such as [39, 147], use a deterministic approach where trust evaluation calculated from handcrafted formulas. It is argued in [81] that this approach enables trust models to aggregate the essential factors they have been considered in their models such as the credibility of witnesses and time factors as in [39] and even context and criteria similarity rate as in [75]. Other numerical trust models, such as [122], use fuzzy logic where common parameters of trust models can be represented by fuzzy sets, while membership functions describe in what degree a parameter belongs to each set, and fuzzy rules are used to predict the trust of trustees and even the credibility of witnesses [81]. Machine learning is also used as the base of trust models in [128, 130, 5, 149]. Bayesian probability theory also used in trust models such as [43, 42, 139, 94, 125], where the posterior probability that a trustee fulfills behave as expected towards the trustor in light of the previous outcomes experienced by witnesses or the trustor itself [81].

- Context Handling

  Even though trust information is considered context dependent, most models of trust evaluation allow the communication of trust-related information regardless of their contexts [75, 31]. Context aware models calculate various aspects of interactions along with the corresponding evaluation values for a single trustee [31]. Models like [75] not only consider multiple contexts, but also describe adapting values between contexts which may lead to more accurate evaluation of trustees; where contexts might be totally different or have some criteria in common. To measure the effect of a particular context, two individual metrics were defined:

- A Weight Matrix (WM) that includes the importance level of each criterion of context.
- A Relevancy Matrix (RM) that indicates the degree of similarity of each criterion in the first context with the corresponding one in the second context.

The WM is a 1 *n matrix, where n is the number of context criteria and matrix entry w_i is in [0, 1]. The RM is an n *1 matrix where matrix entry v_i is in [0, 1] and refers to the degree of importance of the corresponding criterion. For example, an agent, called B1, may consider the "fast-enough" delivery of a specific transaction very important and uses 0.9 as the corresponding value in its WM. Similarly, another agent, called B2, may also consider the "fast-enough" delivery of another transaction very important and uses 0.9 as the corresponding value in its WM. However, for B1, "fast enough" means: within one week. On the other hand, for B2, "fast enough" means: within one day. Therefore, B1 will use a lower value for "delivery time" criterion in its RM (e.g. 0.2) whereas, B2 will use a higher value for "delivery time" criterion in its RM (e.g. 0.7).

Generally speaking, context aware models can be more complicated than the classical single context models and demands higher computational power. However, properly handling various contexts can lead to more accurate trust evaluation [81].

- Adaptability

  Typically, trust models should consider an adaptive approach to deal with the dynamic nature of systems where they operate, where it may be impossible to predict all the potential events [31]. Trust models themselves may become the target of security attacks and be compromised. Properly responding to dynamic changes increases reliability and robustness of trust models. Some security attacks, like betrayal and inconsistency attacks are associated with behavioural change of trustees, other attacks like ballot stuffing and bad mouthing, are related to misbehaving witnesses. Successful trust models are expected to be able to deal with unanticipated events such as changes in trustees or information sources

attitudes, whether the change was accidental or intentional. Many trust models, like those described in [128, 39, 75] use time decaying factors to emphasize that information based on recent transaction weigh more compared to old transactions, which helps trustors adapt to recent changes, without ignoring historical behaviours. The computational models of Marsh and Briggs [69] use the idea of regret and forgiveness to adapt to dynamic changes in trustees' behaviour. The trust estimation at time instance $t + 1$ depends of the current estimation (at time t) and a forgiveness function. Moreover, trust models should be able to operate properly in the absence of some information sources. Even though models like those described in [128, 69] relies on direct experience only, many others like those described in [123, 45, 4] use both direct experience as well as indirect experiences in the form of testimonies from witnesses. The framework presented in [39] is designed to integrates multiple sources of information to address potential absences information sources.

Dealing with identity change, also referred to as Whitewash and Sybil attacks, is more challenging, where agents creates a new identity to avoid its bad history with others. Some models like those described in [1, 82, 2] rely on public key authentication mechanism to address identity change and Sybil attacks where the certificate issuing authority only allows one identity per entity. Aboulwafa and Bahgat described Dirichlet-based Reputation and Credential Trust management framework (DiReCT) [1] that integrates trust modelling with hard security mechanizes to resist whitewashing. Zacharia and Maes [156] proposed assigning low trust values to newcomers, removes the motivation of re-entry.

- Cheating Behaviour Assumption

  While some agents behave properly at all times, others might misbehave either always or from time to time. Various agents' misbehaving are possible, however the motivations of such improper behaviours are not necessarily the same. Sabater and Sierra [103] highlighted various assumptions of trust models regarding cheating behaviour of agents that report indirect trust information. Some trust models such as Regret [102] may assume that witnesses are cooperative, and non-cheating. On the other end, trust models like [5] may

assume that agents can intentionally cheat. Between those two extremes, trust models can assume that agents may misbehave accidentally, but not intentionally such as the model described in [75]. Models that assume non-cheating agents simply use reported information by witness in their calculations, while trust models that assume potential cheating witnesses, usually reduce the weight of their testimonies or avoid using their feedback at all.

- Visibility

  Feedback of transactions and models of trustors needs, and behaviours can either be seen as a global property shared by all trustees, as with online reputation models used in Amazon or Ebay, or as a subjective property assessed particularly by each individual [87], as in work of Tran and Cohen [128]. In the first case, the trust evaluation is based on testimonies of the witnesses that in the past had interacted with the trustee being evaluated. These values are publicly available to all members of the community and updated each time a member issues a new evaluation of the trustee. On the other extreme, each trustor builds its own subjective model to its potential partners each partner in the community based on direct experiences and what the others have said to it personally, potentially among other things, to calculate the trust score of the trustee being evaluated.

  Generally, global visibility of trust scores goes well with centralized models, where individual trustors report their scores to the central entity, otherwise the cost of shearing trust evaluation will be high. The main challenge using global visibility is the lack of personalization of trust value [103]. Theoretically, it is possible to design a trust model with distributed architecture and global trust scores, however synchronizing trust evaluations can be complicated.

  As trust is subjective by nature, many trust models adapt a subjective approach, which is more appropriate for when agents frequently interact [87]. Generally, subjective visibility of trust scores goes well with distributed architectures. Theoretically, it is possible to

design a trust model with centralized architecture and subjective trust evaluation where a centralized entity can manage things like identity trust, while individual trustors calculate subjective trust evaluations depending on various sources of trust information [103].

- Incentive Feedback

  Many trust models use witnesses' information; however, it is possible that witnesses have no interest in cooperating or in providing true information about their interactions [31]. Wang et. al [142] proposed trading reputation information to encourage witnesses to provide their feedback. Wang and Zhang [138] proposed increasing reputation of honest and cooperative witnesses and decreasing reputation for non-cooperative or unhonest witnesses.

3.2.2 Overview of Selected Trust Evaluation Models

This subsection overviews a selection of decentralized models and is not meant to provide a comprehensive survey of trust modeling literature in MASs. Presented models are either tightly related to our work such as the work of Tran and Chohen [128] and comPrehensive reputAtion-based TRust moOdeL with Fuzzy subsystems (PATROL-F) [122], or because they are well cited models in the literature such as Trust and Reputation in the Context of Inaccurate Information Sources (TRAVOS) [123] and FIRE [39]. Recent surveys such as [32, 77, 103, 35, 62, 81, 87, 153, 31] provide further detailed overviews of the literature.

Models Based on Reinforcement Learning

- A reinforcement learning based trust evaluation model for buying and selling agents in a dynamic, uncertain and untrusted e-marketplace is described in [128] and further elaborated in [130], where buyers model the trust of the sellers. Sellers are partitioned into trustworthy, untrustworthy and neutral sets. A buying agent chooses to purchase from a trustworthy seller. If no trustworthy seller available, then a seller from the list of non-

untrustworthy sellers is chosen. Initially all sellers are considered neutral. The seller's trust evaluation is updated based on whether the seller meets the expected value for the demanded product with proper quality. If the seller meets or exceeds the demanded product value, then the seller is considered cooperative and its trust evaluation is incremented. The seller is considered trustworthy if the trust evaluation is above a threshold value. If the seller fails to meet the demanded product value, then the seller is considered uncooperative and its trust evaluation is decremented. The seller is considered untrustworthy if the trust evaluation falls below another threshold value. Sellers with trust evaluation in between the two thresholds are considered neutral (neither trustworthy nor untrustworthy). In this model, buyers and sellers make decisions based only on their own experiences. The model builds trust slowly and the buyer has to interact with a seller several times before the trust of the seller crosses the threshold value [4].

- A decentralized extension to the model used in [128] is describe in [92, 93], to enable indirect trust evaluation. Witnesses are partitioned into trustworthy, untrustworthy and neutral sets to address buyers' subjectivity in opinions. As with [128], sellers are partitioned into trustworthy, untrustworthy and neutral sets. A buyer requests information only when it has no previous experience with the seller. A buyer evaluates the influence degree of witnesses simply based on their deviations from its own opinions about some known agents. After purchasing from a seller, the trust evaluations of witnesses are adjusted based on their predictions about the sellers.

- A combined model based on direct and indirect trust evaluation is described in [5] as an extension to [92, 93]. As with [92, 93] advising agents are partitioned into trustworthy, untrustworthy and neutral sets. As with [128], sellers are partitioned into trustworthy, untrustworthy and neutral sets. Two variants presented in [5] where opinions of trustworthy witnesses and their trust evaluation of sellers are utilized in one variant, along with the buyer's own point of view for the trust of a seller. However, in the second variant, only

opinions of trusted witnesses are used; to address situations where interaction between buyers and sellers are infrequent. Reinforcement learning is used to adapt an witness's trust evaluation. To address the subjectivity of witnesses, the mean of the differences between the witness's trustworthiness estimation and the buyer's trust evaluation of a seller is used to adjust the witness's trust evaluation for that seller in the first variant. Both direct and indirect trust evaluation used. Furthermore, the number of previous interactions as well as the recentness of interactions are taken into consideration [4].

Comprehensive Reputation-Based Trust Model with Fuzzy Subsystems

The comPrehensive reputAtion-based TRust Model with Fuzzy subsystems (PATROL-F) [122] is the fuzzy logic extension to the trust model called a comPrehensive reputAtion-based TRust moOdeL (PATROL) [121].

A trustor x evaluate the trust of a trustee y based on it direct experience with y as well as testimonies from witnesses. Witnesses are categorized as Neighbours, Friends, Strangers, and different weight s assigned to testimonies of each category. As a starting point $PATROL_F$ calculates $Trust_x^y$ as follows:

$$Trust_x^y = A_{patrolf} * Trust_x^y + \sum_{I \in \{Neighbours, Friends, Strangers\}} W_I \frac{\sum_{\forall w \in I} TW_x^w * RT_w^y}{\sum_{\forall w \in I} TW_x^w} \quad (3.1)$$

Where

- $A_{patrolf}$ is the weight of for trust evaluation calculated by the trustor itself.
- W_I is the weight of for category I which can be $Neighbours, Friends, Strangers$ where

$$A_{patrolf} + \sum_{I \in \{Neighbours, Friends, Strangers\}} W_I \quad (3.2)$$

- TW_x^w is the credibility of the agent reporting trust value.
- RT_w^y is the reported testimony of witness w about trustee y

To address the credibility of testimonies from witnesses, $PATROL_F$ depends on the following metrics, as its predecessor, PATROL:

- Activity: Depending on the total number of interactions of a witness with other trustees, this metric meant to indicate the activity level of a witness and how much its information is reliable and up-to-date [122]. Activity of witness w denoted as $Act(w)$ and calculated as

$$Act(w) = \frac{\sum Interaction\_from_x}{\sum Interaction\_from_{all\ agents}} \tag{3.3}$$

- Similarity: This metric used to indicate how similar the trust evaluation conducted by trustor x and witness w. The higher the similarity, the more accurate the testimony assumed to be [122]. Similarity between x and w denoted as Sim_x^w and calculated as

$$Sim_x^w = 1 - \sqrt{\frac{\sum_{y \in common\ trustees} (Trust_x^y - Trust_w^y)^2}{25 * |common\ trustees|}} \tag{3.4}$$

  where 25 is used normalize Sim_x^w as the maximum possible trust value in $PATRL_F$ is 5 and the minimum is 0.

- Popularity. This metric used to indicate how much a trustee is liked and how much its services are asked for in the system. This metric can be used for witnesses if a witness can play the role of trustee in the system [122]. Popularity of witness w denoted as $Pop(w)$ and calculated as

$$Pop(w) = \frac{\sum Interaction\_with_x}{\sum Interaction\_with_{all\ agents}} \tag{3.5}$$

- Cooperation: This metric used to indicate the willingness of a witness to respond to requests for testimonies and services (if it provide services). Cooperation of witness w de-

noted as $Co(w)$ and calculated as

$$Co(w) = \frac{\sum Interaction\_with_w}{\sum requests\ for\ testimonies\ and\ services} \tag{3.6}$$

Consequently, TW_x^w is calculated as

$$TW_x^w = a_{patrolf} * Sim_x^w + b_{patrolf} * Act(w) + c_{patrolf} * Pop(w) \tag{3.7}$$

Were

- $a_{patrolf}$, $b_{patrolf}$ and $c_{patrolf}$ are factors chosen by trustor x such that

$$a_{patrolf} + b_{patrolf} + c_{patrolf} = 1 \tag{3.8}$$

$PATROL_F$ uses fuzzy logic to account for humanistic and subjective concepts such as the importance of a transaction, the decision in the uncertainty region, and setting the result of interaction in distributed systems [122]. The model uses three consecutive fuzzy subsystems.

- The first is used to set the importance factor of an interaction. This helps the trustor doing the evaluation to decide how many witnesses to consult and can be used as an input to the second fuzzy subsystem.
- The second subsystem is the core component of the model. It uses the personality of the trustor, in addition to the importance of the interaction and the most recent trust evaluation of the trustee, if any, to evaluate the trust of the trustee. This subsystem is used only if the trustor is not sure whether the trustee is good enough or bad.
- The last subsystem is used to estimate the result of interaction and to figure out whether the interaction was good or bad and consequently update the trust evaluation of the trustee.

The model is decentralized and takes into account direct and indirect trust factors.

Trust and Reputation model for Agent-based Virtual Organizations (TRAVOS)

Trust and Reputation model for Agent-based Virtual Organizations (TRAVOS) [123] is a well-known decentralized, general purpose trust evaluation model for MASs, that uses two information sources to calculate the trust evaluation of trustees: direct experience and testimonies from witnesses. The model depends on beta distributions to predict the likelihood of honesty of a trustee [48]. Interestingly, this model depends greatly on direct experiences and seek advice from witnesses for evaluating a trustee only if the trustor 's confidence in its own evaluation is below a predefined threshold [81, 47]. Furthermore, the model allows to weight testimonies from witnesses based on the accuracy of the past testimonies of each witness instead of calculating the reliability of the provided testimony based on its deviation from testimonies of other witnesses [81]. That is, when x evaluates the trust of y, and x considers the testimonies of w, then, the credibility of w in the context of providing testimonies is only derived from the accuracy of $w's$ previous testimonies about y. Consequently, trustors store not only the evidence about the outcome of the past interactions per trustee, but also the testimonies per witness per trustee, which can increase memory requirements as the number of witnesses and trustees increase [97].

TRAVOS summarize outcomes of interactions a single value to represent the trust evaluation [81] and trust evaluation considered a private and subjective property that each trustor builds. The model computes the reliability of witnesses via the direct experience of interaction between the trustor and witnesses and discards inaccurate testimonies [50]. Unfortunately, it takes certain time for the trustor to recognize the inaccuracy of the provided reports from the previously trusted witness agents [50].

FIRE

FIRE [39] is a well-known decentralized trust evaluation model for MASs. The model uses diverse sources of information namely:

- Direct experience called Interaction Trust or Direct Trust (DT).

- Witness Reputation (WR).
- Role-based Trust (RT).
- Certified Reputation (CR).

Certified Reputation considered to be one of the distinguishing features of the FIRE, where a certified reference disclosed by a third-party witness made available upon request of an inquiring trustor. Such approach can reduce the cost of locating witnesses [81].

A trustor can view a trustee as trustworthy, untrustworthy, or not yet known. The trust value is calculated as the sum of all the available ratings weighted by the rating relevance and normalized it (by dividing the sum by the sum of all the weights). Trust value for component K (DT, WR, RT or CR) is given by [39]

$$T_{K(x,y,s)} = \frac{\sum_{r_i \in R_{k(x,y,s)}} \omega_{k(r_i)}.v_i}{\sum_{r_i \in R_k(x,y,s)} \omega_{k(r_i)}} \tag{3.9}$$

Where

- $T_{K(x,y,s)}$ is the trust value of component K that trustor x has in trustee y with respect to task s.
- $R_{K(x,y,c)}$ is the set of ratings collected by component K for calculating $T_{K(x,y,c)}$.
- $\omega_{K(r_i)}$ is the rating weight function that calculates the relevance or the reliability of the rating r_i ($\omega_{K(r)} \geq 0$)
- v_{r_i} is the value of rating r_i .

To locate witnesses, FIRE uses a variant of the referral system used by [150]. FIRE assumes that addressing subjectivity of witnesses is application dependent. To address resources limitations; the branching factor and the referral length threshold parameters were used. The first used

to limit the breadth of search and the second is used to limit the depth of search [39]. An important aspect of FIRE is that a trustor does not create a trust graph, as in [150], and therefore may quickly evaluates the trustee's trust value using a relatively small number of transactions [50].

3.3 Trust Establishment

From an architectural point of view, we can differentiate between centralized and distributed models. Typically, a central entity manages or facilitates trust establishment of all agents in the first approach, but each agent performs its trust establishment related operations without a central entity in the second approach.

A centralized architecture goes well with online crowdsourcing, such as Amazon Mechanical Turk (AMT) . Typically, the central authority implements a calculation engine to derive the necessary trust establishment actions, as in the Social Welfare Optimizing Reputation-aware Decision making (SWORD) [155]. The model acts as a delegation broker for trustors to dynamically balance the workload between similar trustees SWORD [155].

Decentralized models are more complex than centralized ones. Each trustee is capable of handling trust establishment related information locally and derives the necessary adjustments as needed, using its own computation engine tailored to its requirements without depending on a central entity. Distributed Request Acceptance approach for Fair utilization of Trustees (DRAFT) [152] is considered as the decentralized variant of SWORD.

Trustees should be able to model the behaviour of potential interaction partners. For this purpose, they may exploit different methods to build such model; these methods referred to as computation engines or paradigm type. To the best of our knowledge, existing trust establishment models are numerical models. They do not have any explicit representation of cognitive attitudes to describe satisfaction, but they use numerical aggregation of past interactions and presents a set of subjective probabilities that trustors will be satisfied in a future interaction. Numerical

trust models may use a deterministic approach where trust evaluations are calculated from hand-crafted formulas. This approach enables models to aggregate the essential factors they have been considered in their models such as time factors and context and criteria similarity rate. Other numerical trust models, may use fuzzy logic where common parameters of trust establishment model can be represented by fuzzy sets, while membership functions describe in what degree a parameter belongs to each set, and fuzzy rules are used to predict the satisfaction of trustors and appropriated performance adjustment of a trustee. Also, reinforcement learning can be used as the base of trust establishment model.

Trust establishment models can depend on a single criterion or multiple criteria. In single-criterion approach, trustees receive or calculate a single aggregated value that represents the subjective opinions of and interactions partner. In the multiple-criterion approach, trustees tend to receive or calculate different aspects of their interactions along with the corresponding evaluation values. This help trustees to learn about the needs of trustors per criteria in order to make informed decisions on updating their performance if necessary. Generally speaking, multi-criterion approach can be more complicated than the single one and demands higher computational power. However, it can lead to more accurate modeling of trustors. Most existing trust establishment models [10, 155, 152] use single-criterion approach.

Trust establishment models have different assumptions regarding processing capacity of trustees. Some models, implicitly or explicitly, assume that trustees can service an infinite number of requests from trustors during a time unit without negatively affecting their performance. This assumption is referred to as the Unlimited Processing Capacity (UPC), which may be justified in computerized services or resources like P2P systems [155]. However, when the number of requests a trustee effectively fulfill per unit time is limited, the trustee can be overloaded with requests and not only its performance may decline, but also its reputation [152]. Instead of the accept-when-requested approach assumed in [10], SWORD [155] and DRAFT [152] assume limited capacity trustees.

Feedback of transactions from trustors can be explicit or implicit. Explicit feedback can

be more accurate if trustors are not lying. However, they may generate extra communication overhead. The use of implicit feedback can be appropriate for situations where explicit feedback is not possible or not desirable. SWORD [155] and DRAFT [152] do not use feedback from trustors.

3.3.1 Overview of Selected Trust Establishment Models

Modeling buying agents' needs in e-marketplace described in [128] and the use of trust-gain as an incentive mechanism for honesty in e-marketplace environments as in [159] are considered preliminary works for developing models for trustees to establish trust.

Social Welfare Optimizing Reputation-aware Decision Making (SWORD)

In the literature, most existing trust evaluation models choose a selfish interaction approach where trustors select the most trustworthy trustee known as often as possible. Such models, implicitly or explicitly, assume that trustees can service an infinite number of requests from trustors during a time unit without negatively affecting their performance, which may be justified in computerized services [155]. However, when the number of requests a trustee effectively fulfill per unit time is limited, the trustee can be overloaded with requests and not only its performance may decline, but also its reputation [152]. Instead of the accept-when-requested approach traditionally assumed in trust evaluation models, Social Welfare Optimizing Reputation-aware Decision making (SWORD) approach described in [155] to avoid overwhelming a small number of relatively more trustworthy trustees. SWORD attempts to balance the workload among trustworthy trustees, while taking into account variations in their trust, by acting as a delegation broker for trustors. Due to its centralized architecture, SWORD can experience scalability and transparency issues [153].

Distributed Request Acceptance approach for Fair Utilization of Trustees (DRAFT)

Distributed Request Acceptance approach for Fair utilization of Trustees (DRAFT) [152] is a reputation management model for trustees based on distributed constraint optimization to help each resource constrained trustee decide, dynamically, whether or not further requests for interaction should be accepted. The aim is to prevent the resulting trust evaluation from being affected by factors out of the trustees' control. DRAFT attempts to balance between maximizing the potential reward for the trustee agent and minimizing the degradation of performance experienced by the requesting trustors.

The Reputational Incentive (RI)

The Reputational Incentive (RI) model described in [10] is based on the notion that untrustworthy trustees will have less chance of being selected, and so must work for fewer compensations than trustworthy ones in order to remain competitive. However, in doing this, those trustees obtain less reward(s) from interactions, while having to expend the same efforts performing the task. The potential losses or gains associated with reputational changes can provide an incentive for trustees to select a particular performance level when interacting with trustors. RI uses accumulated reputation, represented as real numbers, as the single-criterion to adjust the performance of trustees. Each trustee builds its own models of interactions' partners. To maintain an effective operation, it continuously monitors the trustee's reputation and adopts learning techniques using handcrafted formulas with the purpose of adjusting respective parameters tailored to the current situation. RI is a decentralized, generic model that can be instantiated and applied in a wide range of applications. When interacting for the first time, trustors are assumed to be neutral.

The Model of Tran. et. al [129]

The decentralized trust establishment models described in [129] targets selling agents (trustees) for the e-commerce application, in single context environment, to help them better understand

the needs of buying agents (trustors) based on direct Boolean feedback from buyers to indicate whether or not they are satisfied with results of the interaction. Trustees use reinforcement learning to categorize buyers based on two criteria; namely price and quality. Trustees attempt to classify trustors as price-sensitive, who are more interested in a low price than high quality, or quality-sensitive who are interested in high quality more than in a low price, or balanced buyers who consider price and quality equally important. Each trustee builds and maintains its own models of interactions' partners. First-time buyers assumed to be neutral. When buyers change their behaviour, the model responds by updating their categories.

3.4 Summary

In summary, a wide variety of approaches have been presented in the literature to ensure agents encounter trustworthy partners. Consequently, there exists considerable attempts to classify them. In this chapter, we presented major classification dimensions of trust models described in notable surveys and adapted an aggregated list of dimensions for classify trust models to help understanding the research directions in the field. While we did not present a comprehensive survey of the large litterateur, we pointed out some recent surveys that were dedicated for that purpose and described relevant trust evaluation models as will as some well known models in the literature. Then we described relevant trust establishment models before concluding the chapter with this summary section.

Within the domain of trust evaluation, both reinforcement learning, and fuzzy logic deemed to be a suitable for trust evaluating agents in MASs as agents discover which actions yield the most reward via a trial-and-error search in the first approach [130] and the simplicity and similarity to human reasoning of the second approach [28]. A well-known issue with reinforcement learning based trust models is that trustors do not quickly recognize the environment changes an d adapt with new settings. In the next chapter, we aim to address this shortcoming by combining reinforcement learning with fuzzy logic and the use of temporary suspension.

Due to the relatively small amount of directly related work to trust establishment, we attempted to situate our work within the relatively larger context of trust management and present a review for selection of trust establishment models related to our work.

Chapter 4

A Decentralized Trust Evaluation Model for Multi-Agent Systems Using Fuzzy Logic and Q-Learning (DTMAS)

4.1 Introduction

Ensuring that the identity of an entity is as claimed, also referred to as identity trust, and authenticating entities to access resources, referred to as access trust, are usually areas of interest for the information security community [31]. However, the measurement of an entity's possibility to do what it is supposed to do is referred to as it provisions trust [32], which is the class of trust considered in this chapter.

An agent that needs to rely on another agent is referred to as the trustor (also known as service consuming agent in the literature or buyer in the context of e-marketplace). The agent that provides resources or services to trustors is referred to as the trustee agent (also known as service providing agent in the literature or seller in the context of e-marketplace) [39]. These roles tend to be situational rather than fixed since an agent may act as either a trustor or a trustee

under different circumstances [153].

Trustworthiness evaluation should be accurate enough to allow trustors to avoid ”bad” partners and identify ”good” ones in their systems, where ”bad” refers to trustees that produce outcome less than expected when interacting with them, and ”good” refers to those who produce at least the expected result of their partners. The trust evaluation model should be able to dynamically respond to changes of the environment. Also, failure of any node must not lead to the failure of the entire system.

As highlighted in the previous chapter, many existing trust evaluation models used testimonies from witnesses to select the trustworthy trustees. Unfortunately, not all of these models take in account that witnesses themselves may misbehave intentionally (i.e. being malicious), or accidentally, due to lack of information for example. Therefore, a general-purpose model should evaluate the credibility of witnesses in regard to their suggested trustee. Then according to the witnesses' testimonies about their suggested trustees, the model measures the trust of trustees and finally selects the most trustworthy agent among potential candidates.

A suspension technique is used and combined with reinforcement learning to improve the model responses to dynamic changes in the system. This component is used to address the short-term relationship between a trustor and the trustee under consideration. It helps the trustor to address a recently malfunctioning trustee that used to be honest for a relatively large number of transactions. The idea is that a trustor will stop interacting with a misbehaving trustee immediately and wait until it is clear whether this misbehaviour is accidental, or it is a behavioural change. Because the suspension is temporary, and because trustor uses information from witnesses, the effect of accidental misbehaviour will phase out, but the effect of a behaviour change will be magnified.

Fuzzy Logic Systems has been widely used for trust evaluation of web services [111], for cloud computing [107], for social networks [114], and for multi-agent systems [122, 12, 110]. This indicates the usefulness of Fuzzy Logic System (FLSs) for trust management due to their

similarity to human reasoning, and their simplicity [28]. Fuzzy Logic can be used to translate human knowledge into a set of basic rules, representing the mapping of the input to the output in linguistic terms [100]. Different approaches have been used in the literature to create, adapt or refine rules, when knowledge is not available, such as using neural networks, genetic algorithms and RL [79]. Q-Learning is a Reinforcement Learning (RL) technique that works by learning an action-value function based on the interactions of an agent with the environment and the immediate reward it receives [130]. This method has been successfully applied to trust evaluation in [128, 131, 4, 92, 93]. The combination of fuzzy logic and Q-Learning algorithm is a powerful mechanism proposed in [6] where the Q-Learning allows to adapt the system that facilitates the modeling at a higher level of abstraction [79]. The combination of fuzzy logic and Q-Learning algorithm used successfully in [79, 78] for dynamic traffic steering and load balancing in wireless networks.

The chapter is organized as follows: a general overview of the system and assumptions are presented in the following section. We present the details of the proposed suspension technique in section 4.2.1 followed by performance analysis, including performance measures, simulation environment, and parameters are described in Section 4.3.

4.2 A Decentralized Trust Evaluation Model for Multi-Agent Systems using Fuzzy Logic and Q-Learning (DTMAS)

In this section we describe the use of fuzzy inferences to aggregate trust factors from various sources. Suspension will be used as a short-term response to dynamic changes and Q-Learning to evaluate trust on the long term. Trustors use Q-Learning to evaluate the trust for trustees based on direct experience, testimonies of witnesses and the average of time decayed utility gain within the last G interaction with the trustee. For those trustees that are not suspended, trustors consult witnesses for their testimonials about trustees. Two fuzzy subsystems are used, one for

the credibility estimation of witness and the main one for trust evaluation of trustees.

The design of a FLS involves defining input signals, the fuzzy sets and membership functions of the input signals and defining the linguistic rule base which determines the behaviour of the FLS [79]. Three fuzzy sets have been defined for each input to achieve a reasonable granularity in the input space while keeping the number of states small to reduce the set of control rules. In particular, each input can Low (L), Medium (M), or High (H).

4.2.1 The Use of Temporary Suspension

To reduce the side effects associated with a trustee that misbehave after building a good reputation, trustors may temporarily suspend interacting with such a trustee immediately after any unsatisfactory transaction. The duration of suspension may depend on the harm caused by that transaction. The temporary suspension allows the trustor to re-evaluate the trust of the trustee without any further unpleasing transactions. The idea is that accidentally misbehaving trustees keep their reputation in the community, unlike those who misbehave due to long term behaviour change. Similarly, an agent may suspend the use of testimony from some witnesses.

Trustor x suspends the use of y for a period of time determined by equation (4.1).

$$SUS_x^y(s,t) = SUS_x^y(s,t-1) + BSI * (UG_x^y(s,t) - MinUG(s)) \tag{4.1}$$

- $SUS_x^y(s,t)$ is the suspension penalty associated with y at instant time instance t.
- The Basic Suspension Interval (BSI) is application dependent and could be days in an e-marketplace or seconds in a robotics system that has a short life time.
- $UG_x^y(s,t)$ is the Utility Gain (UG) if the unsatisfactory transaction between x and y for task s.
- $MinUG(s)$ is the minimum possible value for task s.

Whenever a trustor x consults an honest witness w regarding the trust of y, the witness reports the lowest possible credibility score for y, when y is suspended. Furthermore, an honest witness will reduce the reported trust evaluation of y after the end of the suspension period. This reduction is inversely proportional to the time elapsed since the end of suspension. In other words, the reported trust evaluation of a trustee will equal the calculated trust evaluation of y, based on the history of interaction between the witness and y, minus a penalty (i.e. punishment) amount related to time elapsed since the end of suspension period.

When considering the credibility of witnesses, a suspension policy similar to the one implemented in the direct trust component of the model is used. That is, a trustor x will suspend the use of any witness whose advice is the opposite of the actual result of interaction with the trustee. The suspension period increases as the transaction importance increases.

$$SUS_x^w(s,t) = SUS_x^w(s,t-1) + WBSI * (UG_x^y(s,t) - MinUG(s)) \tag{4.2}$$

- $SUS_x^w(t)$ is the suspension penalty associated with y at instant Time$_t$.
- The Witness Basic Suspension Interval (WBSI) is application dependent and could be days in an e-marketplace or seconds in a robotics system that has a short life time.
- $UG_x^y(s,t)$ is the Utility Gain (UG) if the transaction between x and y, where y is the trustee for which the testimony of w was not correct.

When a trustor wants to interact with a trustee at time t, the trustor avoids any trustee that is suspended and may evaluate the trust of trustees using recognized, available information sources

4.2.2 Input Parameters

We define three input parameters for the main FLS as follows:

- The calculated direct trust evaluation

- Indirect trust
- The average of time decayed utility gain

Direct Trust Evaluation DT_x^y

Trustors use RL to estimate the direct trust of trustees in a way similar to the process in [130, 128], such equations are presented here for convenience. If a trustor is satisfied by the interaction with a trustee, Eq. (4.3) is used to update the trust of the trustee as viewed by the trustor.

$$DT_x^y = min(DT_x^y + \alpha_1(1 - |DT_x^y|), 1) \tag{4.3}$$

- DT_x^y is the direct trust estimation of the trustee y as calculated by the trustor x. The value of DT_x^y varies from -1 to 1, where one mean fully trustworthy, and -1 means fully untrustworthy.
- The cooperation factor α_1 is positive ($< 0\alpha_1 \leq 1$)
- The initial value of the direct trust evaluation is set to zero (neutral). A Trustor x will consider a trustee y as being cooperative if the resulting UG of the transaction is greater than or equal the trustor's satisfactory threshold.

A trustee is considered trustworthy if the trust evaluation is above an Honesty Threshold (HoT), which is similar to the cooperation threshold in [69]. The trustee is considered untrustworthy if the trust evaluation value falls below a Fraudulent Threshold (FrT), which is similar to the forgiveness limit in [69]. Trustees with trust evaluation values between the two thresholds are considered neutral.

If the trustor is not satisfied by the interaction with the trustee, the trustor updates the credibility of the trustee as follows:

$$DT_x^y = max(DT_x^y + \beta_1(1 - |DT_x^y|), -1) \tag{4.4}$$

- β_1 is a negative factor called the non-cooperation factor ($0 > \beta_1 \geq -1$).
- A Trustor x will consider a trustee y as being non-cooperative if the resulting UG of the transaction is less than the x's satisfactory threshold.

Furthermore, and unlike existing work in [130, 128], the trustor immediately suspends the use of the trustee for a period of time related to value of the unsatisfactory interaction as in Eq. (4.1).

We believe that the cooperation and non-cooperation factors are application dependent and should be set by each agent independently. In general, we agree with [130] that the factors should be related to the value gain of the transaction.

When a trustor wants to interact with a trustee at instant Time$_t$, the Trustor avoids any trustee that is untrustworthy (i.e., $DT_x^y < FrT$) or suspended (i.e., $SUS_x^y > t$).

Indirect Trust IT_x^y

As with many other models, such as [4, 39, 122], the model takes into consideration testimonials of witnesses, and the credibility of witnesses. A trustor always asks testimonies of witnesses if such witness exist and were not suspended. If no such witness found, the neutral value (0) for IT_x^y is assumed. To evaluate indirect trust, a trustor x consults other witnesses who interacted previously with the trustee y. To reduce the effect of fraudulent witnesses, a trustor tracks the credibility of witnesses and weighs each testimony by the credibility of the witness. An honest witness w reports its testimony FT_w^y is w's fuzzy trust evaluation of y.

The trustor x calculates the indirect trust IT_x^y as

$$IT_x^y = \frac{\sum_{w=1}^{N_w} FW_x^w * FT_w^y}{\sum_{w=1}^{N_w} FW_x^w} \tag{4.5}$$

- N_w is the number of consulted witnesses.
- FT_w^y is the testimony of witness w about y
- FW_x^w is the fuzzy credibility of witness w.

Calculating FW_x^w is explained later in the following subsection.

The Average of Time Decayed Utility Gain

UG is the net benefit that a trustor x achieves from the transaction. Time decaying is used to emphasize that UG from recent transaction weigh more compared to UG from old transactions if they have the same absolute value. This input signal is used to indicate that not only the number of transactions is important, but also the benefits from those transactions also counts. This helps avoid being fooled by a high rate of successful, but not important transactions. The average of time decayed utility gain within the last G interaction with a trustee y is referred to as AUG' , as calculated by equation 4.6.

$$AUG'^y_x = \frac{\sum_{j=1}^{G} e^{-\lambda_1 \Delta_1 T} UG^y_{jx}}{G} \tag{4.6}$$

- Δ_1T = Current Time - Time of the transaction,
- λ_1 is the decaying factor
- $UG_j$is the utility gain for transaction j between x and trustee y being evaluated
- G is the size of the historical window considered for calculating AUG'^y_x.

4.2.3 Fuzzification

Values for input parameters described in section 4.2.2 are real numbers. These crisp numerical values of input parameters need to be transformed into the equivalent membership values of

appropriate fuzzy sets. A fuzzification operator or a Membership Function (MF), transforms the continuous input into fuzzy sets by calculating the degree of membership to each fuzzy set. Regardless of input parameter or value, the output of a MF is usually a degree of membership in the related fuzzy linguistic sets within the interval $[0, 1]$. Triangular, Trapezoidal, piecewise linear and Gaussian, are commonly used shapes for MFs [74].

The three input parameters; DT_x^y, IT_x^y, and $AUG_x^{'y}$ described in section 4.2.2 are fuzzified according to Gaussian MFs as follows:

$$\mu_{B_i^d}(v) = e^{-(\frac{(c_i^d - v)^2}{2(\sigma_i^d)^2})} \tag{4.7}$$

where

- B_i^d (i = 1, 2, 3 and d=L, M, H) represents the fuzzy set that can be interpreted as v is a member of the fuzzy set d (e.g., B_i^d is the fuzzy set of low direct trust when i = 1 and d = L).
- c_i^d and σ_i^d are the mean and standard deviation values for the corresponding Gaussian MF, respectively.

For example, Figure 4.2.3 shows that the point representing 0.2 Directs Trust (DT) is projected onto the membership function curve which describes the linguistic set $Medium\ DT$, and reach the membership value $\mu = 0.9$ for the fuzzy set $Medium\ DT$.

The Gaussian MF defined for the output fuzzy set as follows:

$$\mu_{E^d}(v) = e^{-(\frac{(c_i^d - v)^2}{2(\sigma_i^d)^2})} \tag{4.8}$$

where

- E^d (d=L, M, H) represents the fuzzy set interpreted as v is a member of the fuzzy set d.

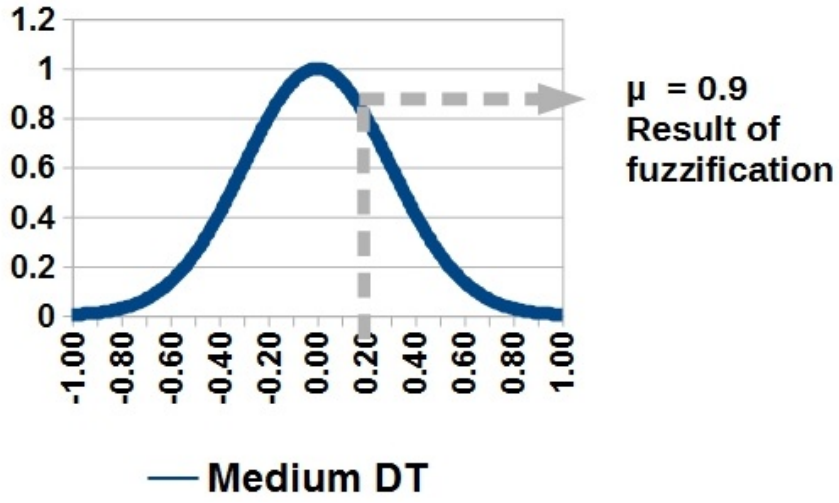


Figure 4.1: Fuzzifying input variable Direct Trust

- c_i^d and σ_i^d are the mean and standard deviation values for the corresponding Gaussian MF, respectively.

4.2.4 Inference Rules

The next step is defining the rules for the inference engine of the FLS, which allows expressing the knowledge in linguistic terms by a set of rules following a syntax of the type IF-THEN to set the control strategy [74]. Then, the inference engine identifies the activated rules and calculates the fuzzy output values for each rule. An example rule for main FLS is the following:

"If DT is HIGH and the IT is HIGH, and the $AUG_x'^y$ is HIGH then the Fuzzy Trust (FT) is HIGH."

This rule intuitively states that if the estimated direct trust is high and the indirect trust is high, and the average utility gained by interacting with this trustee y over the last G interactions is HIGH then trustee y is expected, to be honest and the transaction result is expected to be high based on local experience of trustor x, subject to the constraint that trustee y is NOT suspended.

Using Fuzzy Operators

When the fuzzy inference system contains more than one input variable, the antecedent of If-Then rule might always be defined by more than one fuzzy linguistic set, because in most cases each input variable has one corresponding fuzzy set based on which to figure out the degree of membership [100]. In the main FLS, the antecedent of Rules consists of three fuzzy linguistic sets combined with AND operator to obtain one numerical value that represents the result of the antecedent for the rule. The most common fuzzy operators are AND operation and OR operation. To formulate these logical operation, function min and function max are applied. Although other functions, such as product and probabilistic OR, are also applicable in expressing these fuzzy operators, function min and function max are always simple, and widely used [100].

Figure 4.2 shows the AND operation via function min for the following rule "If DT_x^y is MEDIUM and the IT_x^y is low, and the $AUG_x^{'y}$ is HIGH then the FT_x^y is MEDIUM." Three different fuzzy sets of the antecedent in the Rule yielded the fuzzy membership values 0.9, 0.41, and 0.25 respectively, and the minimum of them is 0.25, is picked out as the result of the antecedent of Rule.

Implication

The consequent part of If-Then rule is another fuzzy linguistic set defined by an appropriate membership function. Unlike the result from antecedent part of If-Then rule that a single numerical value is generated, the inference method in Then-part is to reshape the fuzzy set of consequent part according to the result associated with the antecedent or say the single number. This process is called implication method. The AND operation is implemented which truncates the fuzzy set of consequent part. The extent of deformation of the output fuzzy set in each rule should depend on the specific single number coming from the matching antecedent of the rule

[100]. The output fuzzy set of the i^{th} rule is $FT_{xi}^{'y}$ such that

$$\mu_{FT_{xi}^{'y}}(ft) = \mu_{DT_{xi}^{y}}(dt) \bigwedge \mu_{IT_{xi}^{y}}(it) \bigwedge \mu_{AUG_{xi}^{'y}}(aug') \tag{4.9}$$

Figure 4.3 shows an example of implication for the same rule used in the previous point.

Aggregation

Each rule whose antecedent has a non-zero matching degree contributes a fuzzy set output. Aggregating these fuzzy sets into a single fuzzy set is necessary before defuzzification. Function max, sum, and probabilistic OR are all applicable for aggregation operation, but function max is widely accepted for rule aggregation operation [85]. The max function calculates the overall fuzzy output from the set of individual outputs by taking the maximum value where one or more terms overlap [85].

In the main FLS, three truncated fuzzy sets coming from twenty-seven rules respectively are operated through aggregation method by function max, to create a new fuzzy set representing the outcome for output variable 'Trust' before the defuzzification process.

$$\mu_{FT_{xi}^{'y}}(ft) = max_{1 \leq i \leq 27} \mu_{FT_{xi}^{'y}}(ft) \tag{4.10}$$

The proposed rules for the main FLS are in table 4.1. We insisted that a trustee with any low component cannot have a high direct trust value. The idea is that a trustor will avoid being fooled by a trustee with a high rate of low UG successful transactions.

4.2.5 Defuzzification

The last step of fuzzy inference process is defuzzification, through which the combined fuzzy set from aggregation process will output a single scalar quantity. As the name implies, defuzzi-

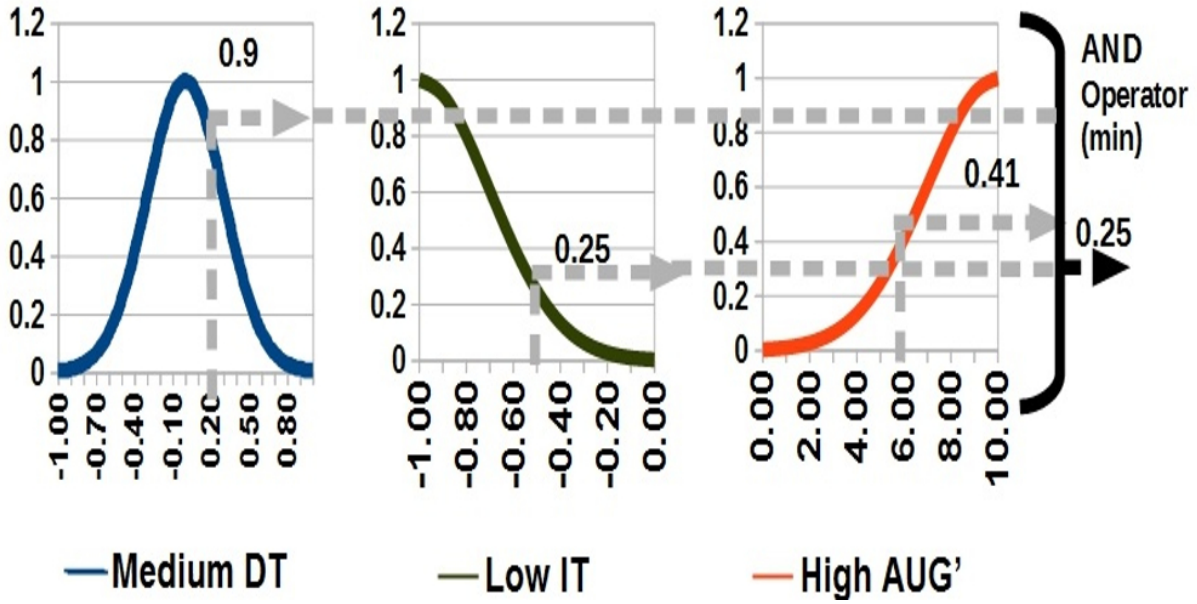


Figure 4.2: The AND operation

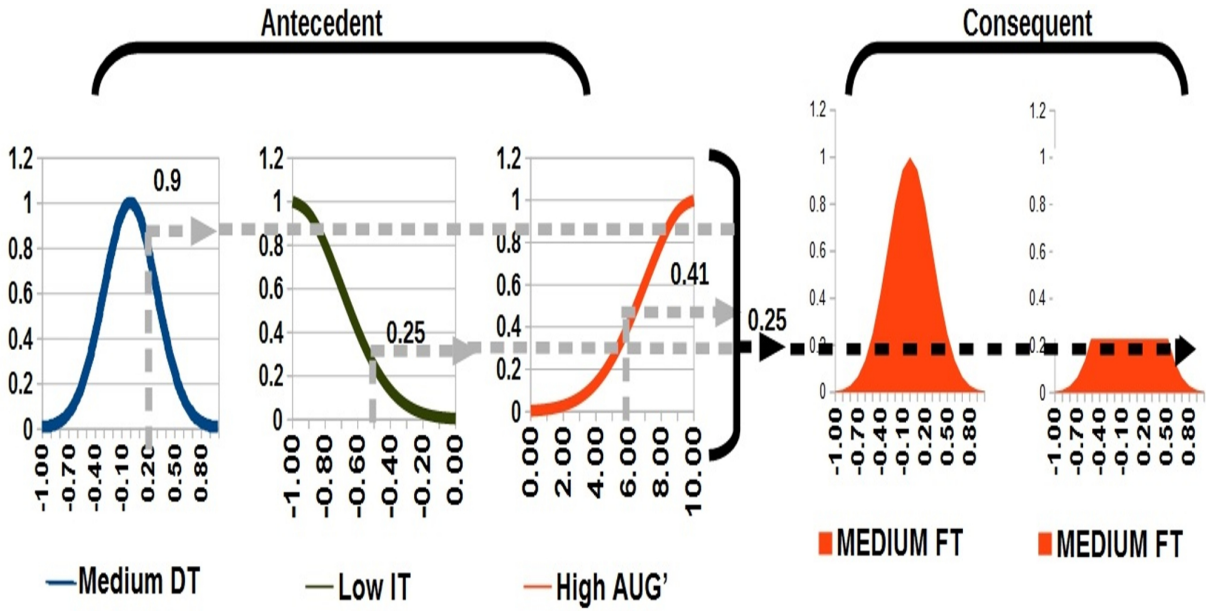


Figure 4.3: Implication

Table 4.1: Main Fuzzy Subsystem Rules

Rule	DT	IT	AUG'	Output
1	L	L	L	L
2	M	L	L	L
3	H	L	L	M
4	L	M	L	L
5	M	M	L	L
6	H	M	L	M
7	L	H	L	L
8	M	H	L	M
9	H	H	L	M
10	L	L	M	L
11	M	L	M	M
12	H	L	M	M
13	L	M	M	L
14	M	M	M	M
15	H	M	M	H
16	L	H	M	M
17	M	H	M	M
18	H	H	M	H
19	L	L	H	L
20	M	L	H	M
21	H	L	H	M
22	L	M	H	M
23	M	M	H	M
24	H	M	H	H
25	L	H	H	M
26	M	H	H	H
27	H	H	H	H

fication is the opposite operation of fuzzification. Since in the first procedure the crisp values of input variables are fuzzified into a degree of membership about fuzzy sets, the last procedure extracts a precise quantity out of the range of fuzzy set to the output variable. The Centroid Method (also called center of area or center of gravity) is an appealing approach for defuzzification [84]. It is calculated as follows:

$$FT_x^y = \frac{\int_{ft} \mu_{FT_x'^y}(ft)(ft.dft)}{\int_{ft} \mu_{FT_x'^y}(ft)(dft)} \tag{4.11}$$

where ft' is the output variable, and $\mu_{FT_x'^y}(ft)$ is the membership function of the aggregated fuzzy set $FT_x'^y$.

4.2.6 Credibility of Witnesses

Trustors track the credibility of their witnesses TW_x^w . Each trustor updates its rating for the witnesses after each interaction as follows

- If the transaction was satisfactory for the trustor and a witness w had recommended the trustee y or
- If the transaction was NOT satisfactory and $w's$ opinion was "not recommend".

Then the credibility of w is incremented as follows:

$$TW_x^w = min(TW_x^w + \gamma_1(1 - |TW_x^w|), 1) \tag{4.12}$$

- Otherwise, the credibility of witness w (TW_x^w) is decremented as follows:

$$TW_x^w = max(TW_x^w + \zeta_1(1 - TW_x^w)|), -1) \tag{4.13}$$

Table 4.2: Witnesses Credibility Fuzzy Logic Subsystem Rules

Rule	TW	UG	Output
1	L	L	L
2	M	L	L
3	H	L	M
4	L	M	L
5	M	M	M
6	H	M	M
7	L	H	L
8	M	H	M
9	H	H	H

- γ_1 and ζ_1 are positive and negative factors respectively and chosen by the trustor as cooperation and noncooperation factors.
- The value of TW_x^w varies from -1 to 1.

The calculated TW_x^w is used as an input to the Witnesses Credibility Fuzzy Logic Subsystem (WCFLS), together with the utility gained from the transaction in order to calculate the credibility of a witness. Each input can be either Low (L), Medium (M), or High (H). The rules for this FLS are presented in (table 4.2). The centroid method used for calculating the output in this work. Table 4.3 presents summary of DTMAS Equations and Their Use.

4.3 Performance Analysis

It is often difficult to find suitable real-world data set for comprehensive evaluation of trust models since the effectiveness of various trust models need to be assessed under different environmental conditions and misbehaviours [153]. Simulation-based evaluations offer the possibility of experimentally evaluating different trust models based on the same set of environmental parameters and scenarios, which enable the use of performance measurements for comparisons [54]. In trust modeling for MASs research field, most of the existing trust models are assessed using

Table 4.3: Summary of DTMAS Equations and Their Use

Equation	Use
$SUS_x^y(t) = SUS_x^y(t-1) + BSI * (UG_x^y(s,t) - MinUG(s))$	Used by trustor x to suspend the use of y after any unsatisfactory transaction.
$SUS_x^w(t) = SUS_x^w(t-1) + WBSI * (UG_x^y(s,t) - MinUG(s))$	Used by trustor x to suspend the use of w after any incorrect testimony.
$DT_x^y = min(DT_x^y + \alpha_1(1 - \lvert DT_x^y\rvert), 1)$	Used by trustor x to increment the trust of y after satisfactory transaction using Q-Learning.
$DT_x^y = max(DT_x^y + \beta_1(1 - \lvert DT_x^y\rvert), -1)$	Used by trustor x to decrement the trust of y after unsatisfactory transaction using Q-Learning.
$IT_x^y = \frac{\sum_{w=1}^{N_w} FW_x^w * FT_w^y}{\sum_{w=1}^{N_w} FW_x^w}$	Used by trustor x to evaluate the trust of y using Q-Learning testimonies from witnesses.
$AUG'^y_x = \frac{\sum_{j=1}^{G} e^{-\lambda_1 \Delta_1 T} UG_{jx}^y}{G}$	Used by trustor x to calculate the average of time decayed utility gain within the last G interaction with y
$\mu_{B_i^d}(v) = e^{-(\frac{(c_i^d - v)^2}{2(\sigma_i^d)^2})}$	Used by the fuzzifier inside trustor x to fuzzify input parameters to the fuzzy system
$\mu_{FT'^y_{xi}}(ft) = \mu_{DT_{xi}^y}(dt) \bigwedge \mu_{IT_{xi}^y}(it) \bigwedge \mu_{AUG'^y_{xi}}(aug')$	Implication used by the fuzzy engine inside trustor x
$\mu_{FTx'^y}(ft) = max_{1 \le i \le 27} \mu_{FT'^y_{xi}}(ft)$	Aggregation used by the fuzzy engine inside trustor x
$FT_x^y = \frac{\int_{ft} \mu_{FT'^y_x}(ft)(ft.dft)}{\int_{ft} \mu_{FT'^y_x}(ft)(dft)}$	Used by the defuzzifier inside trustor x to produce crisp output
$TW_x^w = min(TW_x^w + \gamma_1(1 - \lvert TW_x^w\rvert), 1)$	Used by trustor x to increment the credibility of w after a correct testimony using Q-Learning.
$TW_x^w = max(TW_x^w + \zeta_1(1 - TW_x^w)\rvert), -1)$	Used by trustor x to decrement the credibility of w after a incorrect testimony using Q-Learning

simulation or synthetic data [153]. Experiments differ among various works. Usually, they focus on the performance of trust models, but there is no agreed-on benchmarks to enable comparing different results [54]. One of the most popular simulation testbeds for trust models is the Agent Reputation and Trust (ART) test-bed proposed in [26]. However, even this test-bed does not claim to be able to simulate all experimental conditions of interest. For this reason, many researchers design their own simulation environments when assessing the performance of their proposed trust models [153].

4.3.1 Simulation Environment

We use a scenario-simulator approach [144] to compare the performance of different models. Scenarios are pre-generated and not modified during the simulation phase. The generation of scenario files encodes different parameters, such as the number of trustees, the number of trustors, the profile of each agent and so on. These scenarios are then given to the simulator. Thus, multiple simulations, implementing different models, can be run from the same scenario. By fixing every aspect of a scenario run, the only differences between runs will be trust model specific.

Generally speaking, a scenario is a sequence of ”request for interactions” followed by interactions. Each request specifies a service name and the agent seeking that service. The choice of a trustee to interact with will be made at simulation runtime by trust evaluation model. The value of the selected trustee’s response is determined by the trustee at simulation time.

For simulation, we use the discrete-event MAS simulation toolkit MASON [63] with trustees, which provide services, and trustors, that consume services. For the Fuzzy subsystems, we used the jFuzzyLogic Java package [17]. As with [39], we assume that the performance of a trustee in a particular service is independent of that in another service. Therefore, to reduce the complexity of the simulation environment, it is assumed that there is only one type of service in the system simulated and all trustees offer the same service with, possibly, different performance.

When comparing different trust models, the higher the percentage of good transactions, the

more successful the model in predicting trustworthy partners. This is also referred to as success rate or accuracy of the model, which is calculated as the summation of all transactions took place in the system where the demands of trustors fulfilled completely.

To study the performance of the proposed trust model for evaluating the trust of trustees, we compare the proposed model with the most relevant work comPrehensive reputAtion-based TRust moOdeL with Fuzzy subsystems (PATROL-F) [122] under different misbehaving scenarios of trustees and witnesses. Our simulations of PATROL-F is based on the main reference in which they were proposed. To reduce the effect of bootstrapping, for the first 1 percentage of simulation study interval, trustors select interaction partners randomly. Network communication effects are not considered in this simulation. Each agent can reach each other agent. The simulation step is used as the time value of interactions. Transactions that take place in the same simulation step are considered simultaneous. Locating trustees and witnesses are not part of the proposed model; therefore, trustors locate trustees and witnesses through the system.

Having selected a trustee, the trustor then interacts with the selected trustee and gains some utility from the transaction (UG). The value of UG is in -10, 10 and depends on the level of performance of the trustee in that transaction. A trustee can serve many users at a time. A trustor does not always use the service in every round. The probability it needs and requests the service, called its activity level, is selected randomly when the agent is created.

After each transaction, the trustor updates the credibility of the trustee participated in the transaction, as well as the credibility of a witness in the case of the proposed model. It is assumed that trustees and witnesses may be selfish, liars, non-cooperative or simply malfunctioning. The percentage of interactions whose results fully satisfy the needs of trustors is referred to as the success rate and used as the main metric in this study.

Trustees can be in one of three types: good, ordinary, or bad. Each of them has a mean level of performance. The actual performance follows a normal distribution around this mean which is in the range of (5,10] for good trustees, [0, 5] for ordinary trustees and [-10,0) for bad trustees

Table 4.4: Values of Used Parameters

Parameter	Value
Total number of trustees	100
Total Number of trustors per model	100
Percentage of good trustees	30%
Percentage of bad trustees	30%
Percentage of ordinary trustees	40%
Percentage of High Demand Trustors	30%
Percentage of Low Demand Trustors	40%
Percentage of Normal Demand Trustors	30%
Percentage of Highly Active Trustors	30%
Percentage of Regular Active Trustors	40%
Percentage of Low Active Trustors	30%
trustee cooperation factor	0.1
trustee non-cooperation factor	-0.3
Witnesses cooperation factor	0.1
Witnesses non-cooperation factor	-0.3
Degree of decay	0.1
BSI	100
WBSI	100
FrT	-0.5
HoT	0.5
Satisfaction Threshold	0.5 of demand level

or random in [-10,10] in case of random attack.

Since agents are owned and controlled by various stakeholders, the performance of an agent may not be consistent over time. A trustee may change its behaviour. In this simulation study, the performance of a trustee can be changed by a randomly selected amount with a probability selected randomly when the agent is created. When responding to service request, an honest (good or ordinary) trustee responds with its utility gain value it will provide if the transaction tack place. A bad (unhonest) trustee promises to provide a positive value for its utility gain, but the utility gain that the corresponding trustor can get is the true utility gain that the bad trustee can afford (negative value or random one).

To account for different level of services, 30% of trustors were configured with high demands, and another 30% of trustors were configured with low demands, were as the remain 40% of trustors were configured with regular demands. Highly demanding trustors requires at least 75% of the maximum possible UG, low demanding trustors requires no more than 25% of the maximum possible UG and regular demanding trustors requires something in between.

Similarly, highly active trustors request the service with a probability of at least 75%, low active ones request the service with a probability of at most 25%, while trustors with a regular level of activity request the service with probability in between. As with demand levels, 30% of trustors were configured with high activity level. Another ,30% of trustors were configured with low activity level and the remaining 40% of trustors were configured with regular activity level.

4.3.2 Experimental Results:

Base Scenario

First of all, we would like to study the performance of the proposed model in a generic scenario where neither trustees nor witnesses attempt to cheat or misbehave. This is considered the base line, were any good trust model should perform well. The base scenario has a mix of good (30%),

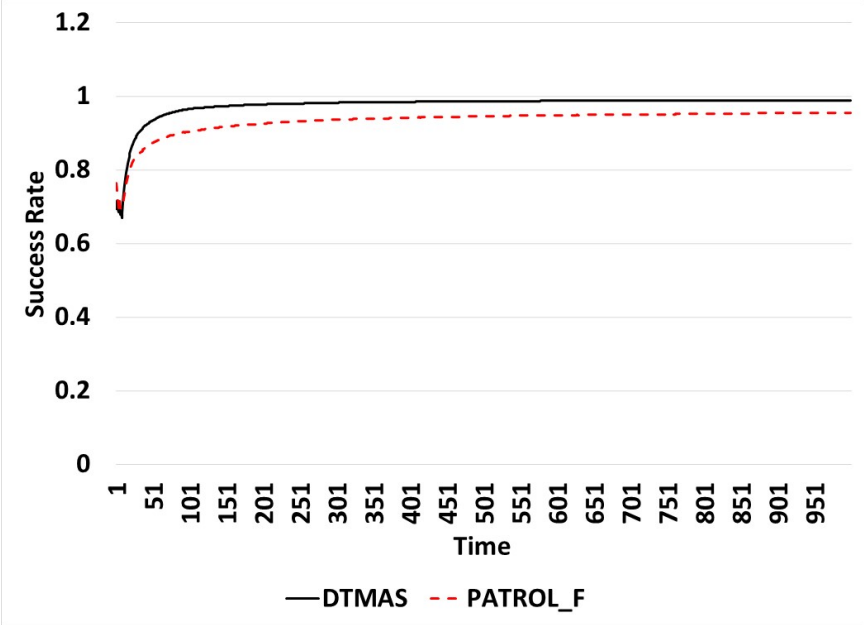


Figure 4.4: Overall Success Rate in the Base Scenario

ordinary (40%) and bad (30%) trustees, but witnesses are honest and fully cooperative. Trustors do not always use the service in every round; the activity level in the range $[0 - 1]$.

Figure 4.4 describes the overall success rate of the simulated models as the number of simulation steps increases 1 to 999. The figure shows that, eventually, both models can achieve high success rates. However, DTMAS and can do that faster. PATROL-F and DTMAS assumes a unidirectional service model and integrate multiple sources of information. They do not tend to avoid feedback form witnesses, and they do not tend to overemphasize such feedback at the price of direct experience. The integration of multiple sources of information helps PATROL-F and DTMAS achieving a high accuracy rates within the base scenario configuration. As each point in 4.4 represent the average of ten experiments, we conducted variance statistical analysis. The low variance shown in figure 4.5 does not invalidate the result of the experiments. We used separate figure to present the variance to enhance readability of results.

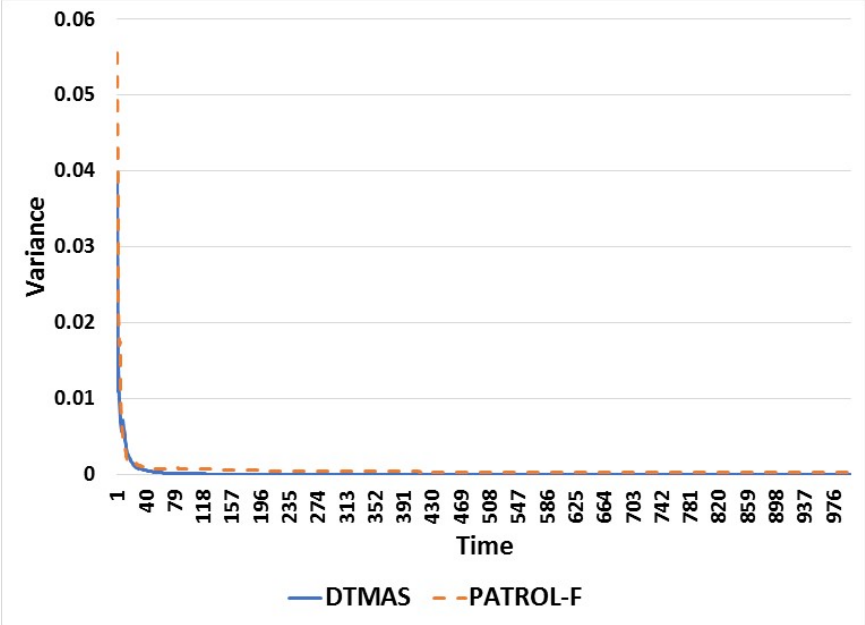


Figure 4.5: Overall Success Rate in the Base Scenario

Misbehaving Trustees Scenarios

When agents are not cheating, models, generally, can achieve a high success rates. However, trustees may misbehave in different ways. A good trust model should guide trustors to interact with honest trustees, if any, rather than misbehaving ones. However, nothing can be done when none of the trustees is honest. In this set of experiments, only ten percent of trustees are honest, while others are misbehaving. In the literature, a wide range of attacks have been investigated by researchers to examine their proposal of trust evaluation models or by independent survey, interested reader may refer to [36] for more information on types of potential attacks on trust evaluation models. Due to the scope of this work, we studied only a sample of what we believe to be basic attacks that trust evaluation models should be able to address. To study the effect of different cheating behaviours, also referred to as attacks, we assume all non-honest trustees use the same cheating strategy. We study the following trustee attacks:

- Attack Type 1: Acting purely in a random way, regardless of time, reputation, or trustors.
- Attack Type 2: Acting randomly sometimes and honestly some others. We study the case

when trustees behave randomly with 50% probability.

- Attack Type 3: Strategic cheating, where they attempt to build good reputation then misbehave before they try to build their reputation again. This is also referred to as "cyclical behaviour" in [105]. We study the case when trustees cheat once each five transactions.

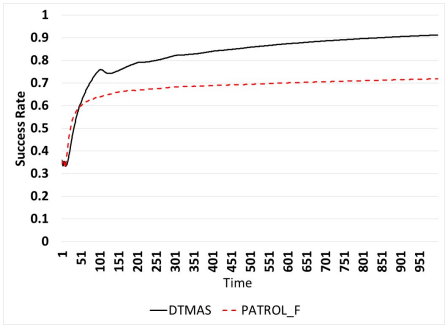
- Attack Type 4: Targeting subset of trustors where trustees misbehave only with this subset but interact honestly with others. We study the case when all cheating trustees target the same half of the community of trustors.

- Attack Type 5: Cheat all times, without paying attention to their partners.

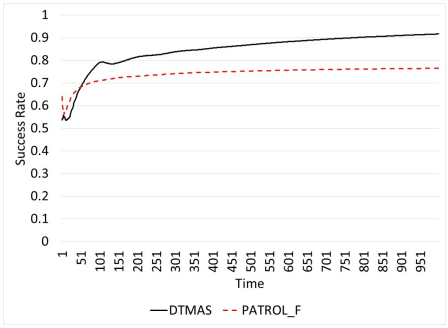
These Scenarios have ninety percent of trustees are misbehaving. Trust evaluating agents, do not always use the service in every round. The activity level is in the range $[0 - 1]$. As we are studying the effects of trustee attacks, we assume all witnesses are honest and fully cooperative.

Figure 4.6 describes the overall success rate of the simulated models as the number of simulation steps increases from 1 to 999.

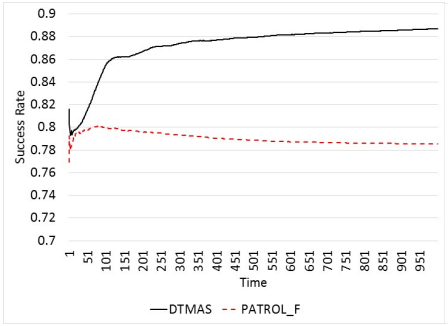
Sub-figure (a) shows that, figuring out trustworthy agents from a set of trustees with random pattern of cheating is challenging due to lack of systematic behaviour. Fuzzy logic works fine is this case, however the use on immediate suspension helps DTMAS to outperform the other model. The performance when cheaters behave randomly only fifty percent of the times is presented in Sub-figure (b), which is, generally, less sever than the purely random attacks; as the number of cheating attempts is reduced, and a partial systematic behaviour can be seen, but still DTMAS outperforms the other model. Sub-figure (c) shows that strategic cheating is hard to detect as it happens once a while, this agrees with the findings of [105]. However, the combination of fuzzy logic with immediate suspension helps DTMAS resist such types of attacks, and let reinforcement direct the long-term learning curve. Sub-figure (d) shows that performance of models when attackers target a subset of trustors only. The similarity selection of PATROL-F



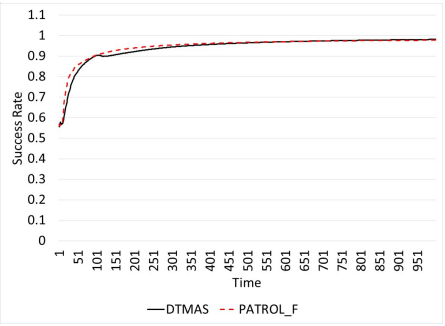
(a) Purely Random Behaviour of Trustees



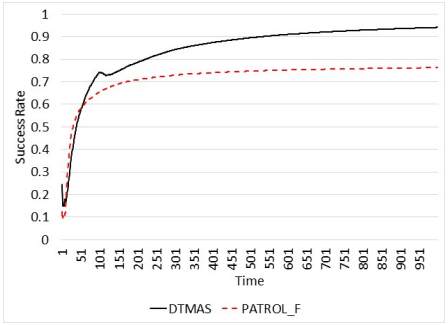
(b) Random Behaviour of Trustees Probability 50%



(c) Trustees Strategic Cheating



(d) Trustees Targeting Subset



(e) Trustees Always Cheat

Figure 4.6: Effect of Trustee Attacks on Success Rate

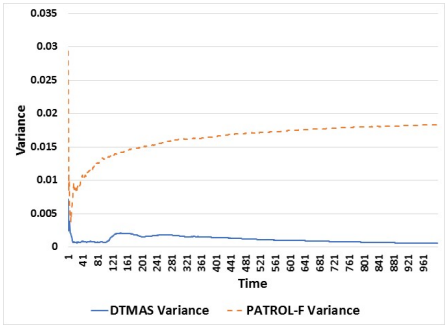
helps the model, slightly, outperforms DTMAS. Sub-figure (e) shows the effect of always cheating trustees on the success rates of the models. The immediate suspension helps DTMAS to outperform PATROL-F.

As each point in 4.6 represent the average of ten experiments, we conducted variance statistical analysis. The low variance, less than 00.2, shown in figure 4.7 does not invalidate the result of the experiments. We used separate figure to present the variance to enhance readability of results.

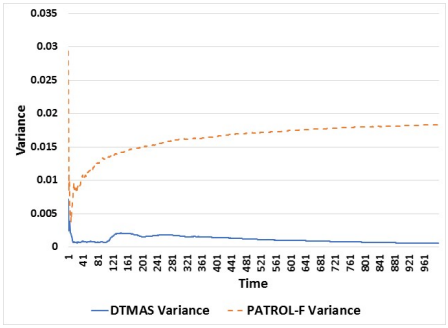
Misbehaving Witnesses Scenarios

Not only trustees may misbehave, but also witness may misbehave in different ways. A good trust model should guide trustors to interact with honest trustees, even in the existence of misbehaving witnesses. Trust models based on direct experience only are not affected at all by potential misbehaviour of witnesses, as they do not use testimonies. However, such models can be at a disadvantage when credible witnesses exist. Models differ in their abilities to resist potential attacks from witnesses. As with the base scenario, this set of experiments has a mix of good (30%), ordinary (40%) and bad (30%) trustees. However, unlike the base scenario, 90% of witnesses are misbehaving. To study the effect of different witnesses' attacks, we assume all misbehaving witness use the same attacking strategy. We study the following witnesses attacks:

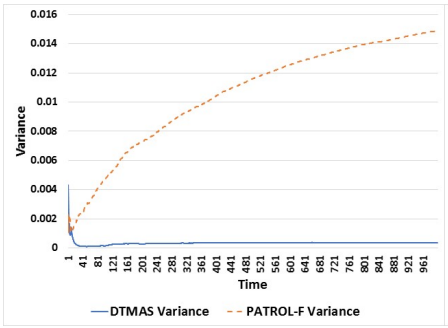
- Witnesses attack type 1: Purely random attack, regardless of time, trustee, or trustor.
- Witnesses attack type 2: Probabilistic random attack where agents behave randomly with a probability, and honestly otherwise. We study the case when trustees behave randomly with 50% probability.
- Witnesses attack type 3: Strategic cheating, where agents cheat once in a while. We study the case when trustees cheat once each ten requests.



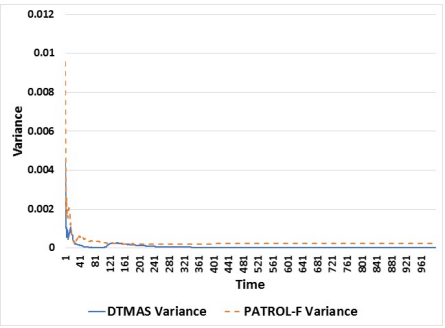
(a) Purely Random Behaviour of Trustees



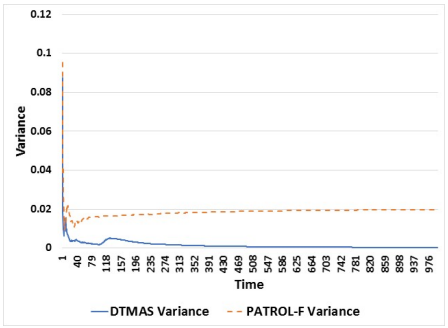
(b) Random Behaviour of Trustees Probability 50%



(c) Trustees Strategic Cheating



(d) Trustees Targeting Subset



(e) Trustees Always Cheat

Figure 4.7: Effect of Trustee Attacks on Success Rate

- Witnesses attack type 4: Targeting subset of trustors where witnesses cheat only with this subset but respond honestly to others. We study the case when all cheating trustees target the same half of the community of trustors.

- Witnesses attack type 5: Bad mouthing, giving negative testimonies about all trustees in response to any request from any trustors.

- Witnesses attack type 6: Positive mouthing, giving positive testimonies about all trustees in response to any request from any trustor.

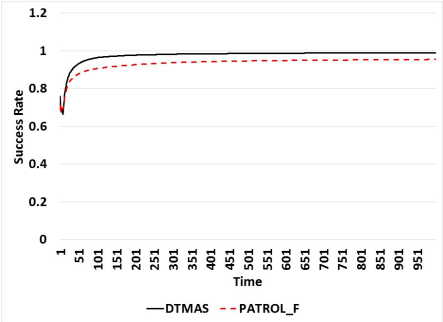
Trust evaluating agents, do not always use the service in every round. The activity level is in the range $[0 - 1]$. As we are studying the effects of witnesses' attacks, we assume all trustee are honest, with different levels of performance.

Figure 4.8 describes the overall success rate of the simulated models as the number of simulation steps increases from 1 to 999.

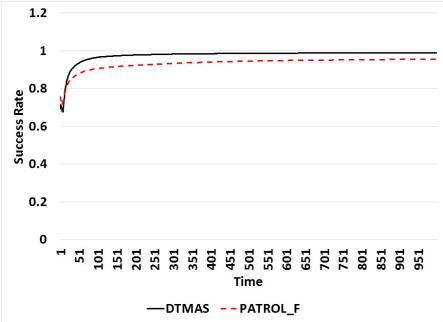
The figure shows that, generally, pure random cheating and targeting a subgroup of trustors may leads to the worst success rate, where as positive mouthing and probabilistic random are less sever. None of the studied models depend on witnesses' testimonies alone; rather this information source is combined with direct experience, which alleviates the effect of witnesses attacks.

The figure shows that DTMAS seems to reasonably resist attacks of witnesses as it temporally suspends interacting with any misbehaving witness, and let reinforcement direct the long term learning curve. Also, the use of fuzzy logic helps DTMAS resist random witnesses' attacks as shown in Sub-figures (a and b). As PATROL-F evaluates the credibility of witnesses continuously to avoid cheating ones, it also has good success rates as shown in Sub-figures (a - f). Sub-figures (e and f) suggests that good mouthing can be less harmful than bad mouthing attack for the two models.

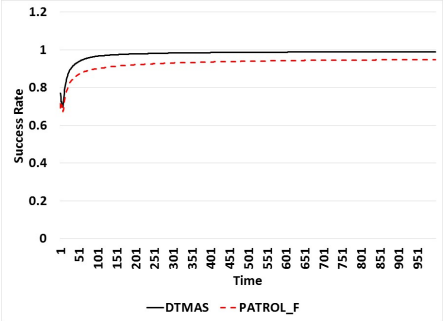
Even though the proposed model doesn't show an extensive superior performance when no



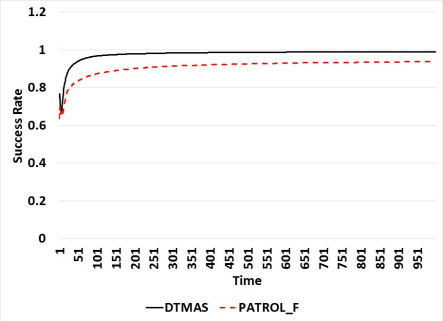
(a) Purely Random Behaviour of Witnesses



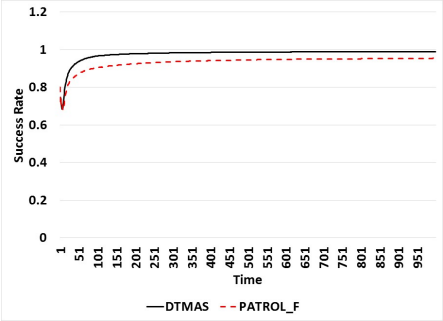
(b) Random Behaviour of Witnesses Probability 50%



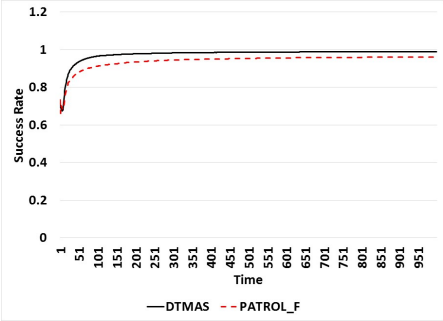
(c) Witnesses Strategic Cheating



(d) Witnesses Targeting Subset



(e) Bad Mouthing



(f) Positive Mouthing

Figure 4.8: Effect of Witnesses Attacks on Success Rate

trustee is cheating, it is interesting to note that the proposed model can be useful for situations where misbehaving trustees expected especially attacks like random behaviour or cyclic attacks. Please note that the proposed model works better for situations where repetitive interactions take place as it is not outperforming comparable models with low repetitive interactions. The accuracy rate of the proposed model can be enhanced by increasing the honesty threshold and the satisfaction threshold at the risk of not finding proper trustee to interact with. This can also happen if trustors increase the absolute value of the non-cooperating factors and basic suspension intervals. While increasing the cooperation factor for witnesses and trustors can make trustors mores friendly, this may increase the risk of being fooled by misbehaving trustees, especially cyclic attackers.

4.4 Summary

In this chapter, we presented a trust model for MASs that combines the use of Q-learning and two fuzzy subsystems to evaluate trust and help selecting collaboration partners. The presented model allows direct and indirect sources of trust information to be integrated to provide a collective trust estimation. The proposed model has been simulated using MASON with the use of the jFuzzyLogic package. The results indicate that the model can help trustors identify good trustees to interact with, and that the proposed model ha potentials to resist attacks. In short, we believe the proposed model can provide a trust measure that is sufficiently useful to be used in MASs. Further comparisons will be presented in (Chapter 6)

Chapter 5

Trust Establishment

5.1 Introduction

Existing trust establishment models for Multi-Agent Systems(MASs) such as [129] allow trustees to adjust their behaviour based on explicit feedback from trustors in order to satisfy trustors' demands. The Reputational Incentive (RI) described in [10] depends on a trustee's reputation for calculating the offered Utility Gain (UG). However, calculating reputation requires explicit feedback from trustors. It is possible that trustors are not willing to explicitly provide feedback to trustees, for many reasons, such as unjustified cost or being self-centred. It is argued in [109] that agents can act independently and autonomously when they do not share problem-solving knowledge. Such agents are not affected by communication delays or misbehaviour of others. Therefore, the resulting MASs can be more general-purpose. In such situations, trustees need a trust establishment model that can function properly without the existence of explicit feedback. Section 5.3 describes an alternative for situations where explicit feedback is not available or not preferred.

Although some researchers have treated the concept of trust as a single and abstract concept, most agreed that the concept of trust is multidimensional [73]. To address the multi-criteria nature of tasks, section 5.4 describes a model that uses a general multi-criteria approach based

on explicit feedback from trustors. The model allows the associations of weights for criteria to enable different importance of various criteria and aiming to address trustors' preferences and expectations.

Then we aim to integrate booth approaches together to benefit from both sources of information in section 5.5; that is, we use explicit and implicit feedback, assuming a multi criteria nature of tasks.

5.2 Retention Functions

5.2.1 Private Retention Function

A Trustee y can model the retention of a trustor x in the system using the function $ret_y^x(s,t) : X \rightarrow [0,1]$, which is called the private retention function of x to y. Initially, y sets the retention rating $ret_y^x = 0$ for every trustor $x \in X$. After responding to a collaboration request with a trustor x, y will update (increase or decrease) ret_y^x depending on whether or not this response results in an actual transaction.

When the environment of MAS is dynamic, old transactions become less important due to changes in the environment [39]. Therefore, trustees scale the impact of old transactions using a decaying factor, to reduce the influence of old transactions adaptively.

Let us define the time decayed retention of trustor x as R_y^x to reflect this

$$R_y^x(s,t) = \sum_{j=1}^{N_y^x(s,t)} e^{(-\rho_2 \, \Delta t_{xj})} \tag{5.1}$$

where

- $N_y^x(s,t)$ is the number of transaction between x and y during the last Ht time units before t for task s.

- ρ_2 is the degree of decay. Trustees can specify the degree of decay ρ_2 ($0 \leq \rho_2 \leq 1$) based on their policies.
- t_{xj} is the time elapsed since the transaction j took place with x.
- t is the time instance of processing the request.

Assuming that rational trustors do not frequently interact with untrustworthy partners, we define the retention function $ret_y^x(s,t)$ to combine both:

- The time decayed retention of x relative to the average time decayed retention per trustor among those having interacted with y previously.
- The time decayed retention of x relative to the maximum time decayed retention per trustor among those having interacted with y previously.

The retention function $ret_y^x(s,t)$ uses the retention relative to the maximum time decayed retention to address the cases of trustors whose retention above average. However, it uses to average time decayed retention to alleviate situations where the maximum retention is very large compared to that of most trustors. If the trustor has no retention value (for being new, or not interacting for along time), then the retention of the trustors is set zero. The multi-line equation used to avoid dividing by zero when the average and maximum retention is zero.

$$ret_y^x(s,t) = \begin{cases} \frac{R_y^x(s,t)}{\underset{i \in X'}{max}(R_y^i(s,t))} + \frac{R_y^x(s,t)}{\underset{i \in X'}{avg}(R_y^i)(s,t)}, & R_y^x(s,t) > 0 \\ 0, & \text{otherwise} \end{cases} \tag{5.2}$$

where

- X' is the set of trustors $\in X$ having interacted with the y so far within the last Ht time units.

- $\max_{i \in X'}(R_y^i)$ is the maximum retention of an individual trustor in the set X'.
-

$$\underset{i \in X'}{avg}(R_y i(s,t)) = \begin{cases} \frac{\sum_{j=1}^{|X'|} R_y^j(s,t)}{|X'|}, & |X'| > 0 \\ 0, & \text{otherwise} \end{cases} \tag{5.3}$$

- $|X'|$ is the size of the set X'.

$ret_y^x(s,t)$ is meant to indicate the willingness of x to interact with y at time t for task s. If a trustor x doesn't interact with y for task s for longer than Ht time period, then it becomes inadvisable to y and $R_y^x(s,t)$ becomes zero and its retention is replaced with the average retention if it comes back. We agree that the $ret_y^x(s,t)$ says nothing about the relation of the x with other trustees, i.e. competitors, as well as other trustors in the community. However, such information might not always be available due to selfishness, privacy, or lack of authorized providers, among other possible reasons.

5.2.2 General Retention Function

In addition to attracting trustors by promising higher utility gain for trustors with low retention rate, trustees can control the offered utility gain in order to enhance profit. When there is a general trend of dissatisfaction among trustors, the trustee y should increase the improvements it makes of proposed utility gain for trustors to avoid becoming untrusted trustee from their perspective, especially if trustors use testimonies from witnesses to evaluate the trust of trustees. Furthermore, y should be very careful when attempting to make profit from those engaged trustors in these circumstances to reduce the chances of extra dissatisfaction and further reduced retention. Therefore, y models the general trend of retention among trustors in the society using the function $g^y(s,t) : G \rightarrow [0,1]$, which is called the general retention trend function of y. Trustee y

may update $g^y(s,t)$ periodically or after each response to a collaboration request with a trustor x.

An increase in the average retention rate, indicates a general trend of satisfaction among trustors. In this case, y expects that a low retention rate of a trustor x is an individual behaviour rather than a general dissatisfaction. As a result, y can still enhance improvements it makes of proposed utility gain for x. However, y can reduce the improvements it makes of proposed utility gain. The general retention trend function of y is:

$$g^y(s,t) = \frac{\overline{Ret(s,t)}}{\overline{Ret(s,t')}} \tag{5.4}$$

- $\overline{Ret(s,t)}$ is the most recently calculated average retention rate. $\overline{Ret(s,t)} = \frac{\sum_{x=1}^{N_{tr}^y(s,t)} NT_x^y(s,t)}{N_{tr}^y(s,t)}$ where $N_{tr}^y(s,t)$ is the number of trustors interacted with y for task s before time t and $NT_x^y(s,t)$ is the number of transactions between trustor x and trustee y for task s before time t.
- $\overline{Ret(s,t')}$ is the average retention rate calculated immediately before $\overline{Ret(s,t)}$ has been calculated.

5.3 Trust Establishment using Implicit Feedback (TEIF)

Trustee y can use reinforcement learning to adjust the UG it provides when interacting with each trustor depending on the most recently proposed UG and the average number of transactions performed by that trustor relative to the average number of transactions performed by all trustors interacting with this trustee. Trust Establishment Using Implicit Feedback (TEIF) does not depend on explicit feedback, nor depends on the current reputations of trustees in the community, and does not require trustors to use a particular evidence-based trust evaluation model. The model is designed to help honest trustees attract more transactions with trustors, and hopefully achieve more profit, in the long run.

If the retention rate of trustor x is less than the average retention rate by all trustors, this indicates that x is not so happy with the UG provided by y. In response, by TEIF , y should put in some effort to encourage x to interact with it later. For this purpose, y categorizes trustors into three non-overlapping sets, and responds with different enhancements to trustors in various categories. When the retention of x is way above the average, then y may carefully attempt to increase its profit. However, if the retention rate is around the average, then y retains its level of performance, generally, unchanged. If the average retention rate by all trustors is decreasing, this indicates a general disappointment among the trustors against y. This situation should encourage y to deliver a higher UG to trustors or increase its rate of enhancement and reduce profit-making. Otherwise, y can, carefully, attempt to increase its profits.

5.3.1 Categorizing Trustors

After responding to a collaboration request with x, y will update (increase or decrease) $ret_y^x(s,t)$ depending on whether or not this response results in an actual transaction. A trustor x is considered engaged by trustee y if $ret_y^x(s,t) \geq \Omega_2$, where Ω_2 is y's engagement threshold $(0.5 < \Omega_2)$. Trustor x is considered un-engaged by y if $ret_y^x(s,t) \leq \Theta_2$, where Θ_2 is y's un-engagement threshold $(0 \leq \Theta_2 \leq 0.5)$. A trustor x with $\Theta_2 < ret_y^x(s,t) < \Omega_2$ is neither engaged nor un-engaged to trustee y.

The set of engaged trustors X_e is the set of frequently returning trustors. Trustee y believes that $\forall x \in X_e$, x does trust y and prefers to interact with it over other trustees if they propose similar utility gain. Therefore, y carefully attempts to make profits, in other words, reduces the proposed UG. By "carefully" we mean small, gradual changes.

On the other hands, the set of un-engaged Trustors X_{ue} is the set of low frequency returning trustors. Trustee y believes that $\forall x \in X_{ue}$, x has no preference for y over other trustees, and such trustors may not be willing to interact with y. Therefore, y puts some efforts attempting to attract trustors in this set. Trustee y believes that $\forall x \in (X - (X_e \cup X_{ue}))$, x has an average

willingness to interact with y, but x has not decided to be loyal to y. Therefore, y does not change the improvement provided in the last transaction.

5.3.2 Improvement Calculation

When trustee y proposes to deliver utility gain for trustor x by a transaction, the proposed utility gain is calculated by equation 2.1, section 2.7. The equation is repeated here for convenience:

$$UG_x^y(s, req) = Improvement_x^y(s, req) + MinUG^y(s)$$

- $UG_x^y(s, req)$ is the proposed aggregated utility gain by trustee y in response to the request req made by trustor x, to do task s.
- $MinUG^y(s)$ are the minimum and maximum possible proposed utility gain that y may deliver to task s.
- $Improvement_x^y(s, req)$ is the calculated improvement by y for task s in response to the request req made by x. Where

  $0 \leq Improvement_x^y(s, req) \leq (MaxUG^y(s) - MinUG^y(s))$

The improvement efforts calculated by y in response to the request req made by trustor x to do task s, at time instance t depends on:

- the category of x : whether it is engaged, un-engaged or neutral
- the value of the engagement factors
- the utility gain improvement proposed for the most recent transaction with x with respect to task s

$$Improvement_x^y(s, req(t)) \begin{cases} (1+\alpha_2) * Improvement_x^y(s, tra') & , x \in X_{ue}^y \\ Improvement_x^y(s, tra') & , x \notin (X_{ue}^y \cup X_e^y) \\ (1+\beta_2) * Improvement_x^y(s, tra') & , x \in X_e^y \end{cases} \quad (5.5)$$

- If $Improvement_x^y(s, req(t))$ calculated by equation 5.5 is greater than $(MaxUG^y(s) - MinUG^y(s))$, $Improvement_x^y(s, req(t))$ will be reset to $(MaxUG^y(s) - MinUG^y(s))$.
- the attraction-factor α_2 is a small increment factor such that $0 \leq \alpha_2 \leq 1$. Trustees use α_2 to increase proposed utility gain for un-engaged trustors.
- the profit-making-factor β_2 is a small decrement factor, such that $-1 \leq \beta_2 \leq 0$. Trustees use β_2 to propose utility gain for engaged trustors and increase the profits they make.
- It is clear that values of those factors are application dependent. As a general guideline, we propose that $\mid \alpha_2 \mid \geq \mid \beta_2 \mid$, to emphasize that satisfying the needs for trustors is more important than making profits.
- We refer to α_2 and β_2 as the engagement factors

5.4 Multi-Criteria Trust Establishment Model using Explicit Feedback (MCTE)

Although some researchers have treated the concept of trust as a single and abstract concept, most agreed that the concept of trust is multidimensional [73]. This section presents a Multi-Criteria Trust Establishment Model Using Explicit Feedback (MCTE) . For a task $s \in S$ the set of criteria can be denoted as $C = \{ c_1; c_2; ...c_p\}$. Individual trustors may have different levels of demand for each criterion, such as response time or transaction cost. Furthermore individual criterion of a service may be of different importance or weight such that $WG = \{ wg_1, ..., wg_p\}$ where $0 \leq wg_i \leq 1$. This allows a trustor to have a mixed configuration of demand and importance.

For example, a trustor may be highly demanding with respect to a criterion, such as response time should be within one seconds, but this criterion is not so important compared to other criteria such as quality of service. Therefore, in response to the request req made by trustor x to do task s at time instance t, trustee y proposes to deliver a utility gain for x by a transaction as follows

$$UG_x^y(s, req) = \sum_{i=1}^{p} Improvement_x^y(s_{c_i}, req) + MinUG^y(s_{c_i}) \tag{5.6}$$

- $UG_x^y(s, req)$ is the proposed aggregated utility gain by trustee y in response to the request req made by trustor x at time instance t, to do task s.

- $MinUG^y(s_{c_i})$ and $MaxUG^y(s_{c_i})$ are the minimum and maximum possible proposed utility gain that y may deliver to criterion c_i of task s.

- $Improvement_x^y(s_{c_i}, req)$ is the calculated improvement by y for criterion c_i of task s in response to the request req made by x at time instance t. Where

$$0 \quad \leq \quad Improvement_x^y(s_{c_i}, req) \quad \leq \quad (MaxUG^y(s_{c_i}) \; - \; MinUG^y(s_{c_i})) \tag{5.7}$$

After interacting with a trustee y, trustor x then gains some benefits from the transaction requested at time instance t. The outcomes of each task is a real number in [0,1] representing the percentage of satisfaction (SAT), such that

$$SAT_x^y(s, tra) = \sum_{i=1}^{p} \frac{wg_x(s_{c_i}) * ug_x^y(s_{c_i}, tra)}{d_x(s_{c_i})} \tag{5.8}$$

where

- $SAT_x^y(s, tra)$ is the satisfaction of trustor x as a result of transaction tra with y for task s.

- $wg_x(s_{c_i})$ is the weight of criterion c_i of task s as determined by trustor x. We assume it is not time or transaction dependent; such that $0 \leq wg_x(s_{c_i}) \leq 1$.

- $ug_x^y(s_{c_i}, tra)$ is the utility gain provided by y to x for criterion c_i of task s in the transaction tra. This may be different for in different transactions.

- $d_x(s_{c_i})$ is the demand level of criterion c_i of task s as determined by trustor x. We assume it is not time or transaction dependent; such that $d_x(s_{c_i}) \geq 1$. If x is not interested in s_{c_i}, then the corresponding weight is set to zero.

We assume that trustors are able and willing to provide explicate feedback to trustees regarding their general satisfaction, but not their satisfaction per criterion.

The case when trustors are willing to provide detailed and accurate feedback, including the level of satisfaction, and the importance or weight per criterion is, generally, straightforward. Using the proposed value of each criterion (by the trustee) and the satisfaction level (of the trustor) of the same criterion, trustees can determine the necessary performance enhancement. However, when only a single aggregated, or general, satisfaction value is provided per transaction, both the appropriate value per criteria and its importance need be predicted.

The proposed trust establishment model uses the provided feedback from trustors regarding how satisfied they were with recent transactions, to adjust the behaviour of trustee(s). If a trustor x is highly unsatisfied, then the performance of its transaction partner y need to be enhanced, particularly concerning to important criteria. On the other hand, if a trustor x is highly satisfied, then the performance of its transaction partner y can be, carefully, reduced, particularly in respect to nonimportant criteria. Let us define the relative weight (rw) of a criterion as the actual weight $w_x(s_{c_i})$ of the criterion, divided by the level of demand for that particular one $d_x(s_{c_i})$. i.e

$$rw_x(s_{c_i}) = \frac{wg_x(s_{c_i})}{d_x(s_{c_i})} \tag{5.9}$$

The satisfaction equation 5.8 can be rewritten as:

$$SAT_x^y(s, tra) = \sum_{i=1}^{p} rw_x(s_{c_i}) * ug_x^y(s_{c_i}, tra) \tag{5.10}$$

5.4.1 Updating Relative Weights

To break symmetry, initially, values for relative weight are set to random value in the range of $[0, 1]$. Upon each transaction, information related to this transaction will be stored in the trustee's local database. Such information will be retrieved when needed for responding to service requests. Since agents may have limited resource (i.e. memory), storing all information about transactions is not necessarily an option. Therefore, each agent will only store information related to transactions for a maximum of Ht time interval. Here Ht is referred to the history size. This parameter is adjustable according to a particular agent's situation.

When the change in satisfaction is greater than or equal to the corresponding proportional change in utility gain for a particular criterion, this indicates that x highly values this criterion. Therefore, the predicted relative weight of the criterion needs to be increased. On the other hand, when the relative change in satisfaction is less than the corresponding proportional change in utility gain for the criteria, this indicates that x does not highly appreciate this criterion. Therefore, the predicted relative weight of the criterion needs to be decreased. The following equations detail the changes

Let us define satisfaction change as

$$\Delta SAT_x^y(s) = SAT_x^y(s, tra^{'}) - SAT_x^y(s, tra^{''}) \tag{5.11}$$

Where $SAT_x^y(s, tra^{'})$ and $SAT_x^y(s, tra^{''})$ are the satisfaction of x in the most recent transaction and the one before it, respectively

Let us define the proportional utility gain change per criterion $pug_x^y(s_{c_i})$ as

$$pug_x^y(s_{c_i}) = rw_x(s_{c_i}) * (ug_x^y(s_{c_i}, tra^{'}) - ug_x^y(s_{c_i}, tra^{''})) \tag{5.12}$$

Then

$$rw_x(s_{c_i}) = \begin{cases} min((1+\lambda_3) * rw_x(s_{c_i}), 1), & \Delta SATx^y(s) >= pug_x^y(s_{c_i}) \\ max((1+\kappa_3) * rw_x(s_{c_i}), 0) & \text{otherwise} \end{cases} \tag{5.13}$$

- λ_3 is a small increment factor such that $0 \leq \lambda_3 \leq 1$. Trustees use λ_3 to increase the predicted relative weight of a criteria.
- κ_3 is a small decrement factor such that $-1 \leq \kappa_3 \leq 0$. Trustees use κ_3 to decrease the predicted relative weight of a criteria.

Note that the relative weight in equation (5.10) is a prediction calculated by y rather than the true value used by x.

5.4.2 Improvement Calculation

The improvement efforts calculated by y in response to the request req made by trustor x to do task s based on explicit feedback depends on:

- The satisfaction level of x in the most recent transaction with y for task s, $SAT_x^y(s, tra^{'})$
- The most recent predicted relative weight of the criterion under consideration $rw_x(s_{c_i})$ as calculated by y for transaction $tra^{'}$ with x on task s.
- The most recent proposed utility gain improvement $Improvement_x^y(s_{c_i}, tra^{'})$ calculated by y for transaction $tra^{'}$ with x on task s.

Table 5.1: Distinct Sets Based on Satisfaction Feedback

Case	Satisfaction(SAT)	Relative Weight(rw)	Interpretation
1	$SAT_x^y(s, tra^{'}) < \Theta_3$	$rw_x(s_{c_i}) \geq \phi_3$	x is not satisfied and c_i is important
2	$SAT_x^y(s, tra^{'}) \geq \Omega_3$	$rw_x(s_{c_i}) < \phi_3$	x is satisfied and c_i is not so important
3	$SAT_x^y(s, tra^{'}) < \Theta_3$	$rw_x(s_{c_i}) < \phi_3$	x is not satisfied and c_i is not important
4	$SAT_x^y(s, tra^{'}) \geq \Omega_3$	$rw_x(s_{c_i}) \geq \phi_3$	x is satisfied and c_i is important

To help a trustee decide which set of criteria needs to be enhanced for a particular trustor, the model uses the aggregated satisfaction feedback provided by the trustor and the relative weight of individual criteria to classify criteria into four distinct sets, show on table 5.1, based on the following two rules:

- Satisfaction Rule: When satisfaction level of the last transaction is bellow the average satisfaction of $y's$ transaction partners, this means that x is not satisfied. Otherwise, x is considered satisfied. Instead of the average satisfaction, SATISFACTION-THRESHOLD (Ω_3) and DIS-SATISFACTION-THRESHOLD (Θ_3) are used for the purpose of generality, where trustees can target a particular level of satisfaction in the range $[0 - 1]$.

- Relative Weight Rule: For a particular trustor x, when the relative weight of a criterion c is above or equal the average relative weight of all criteria, this means that x is the criterion is relatively important to x. Otherwise, c is considered relatively not important. Instead of average relative weight, the RELATIVE-WEIGHT-THRESHOLD (ϕ_3) is used for the purpose of generality.

When trustor x is not satisfied by the last transaction, and the criterion is important, the trustee y should rapidly increase the proposed value for this criterion in future transaction with this particular trustor x. On the other hand, when x is satisfied, and the criterion is not so important, y can slowly decrease the proposed value for this criterion in future transaction with x. When x is not satisfied, and the criterion is not so important y can slowly increase the proposed value for this criterion in future transaction with x. Lastly, when x is satisfied, and the criterion is important y can very slowly decrease the proposed value for this criterion in future transaction

with x. Consequently, based on feedback from x, y can propose the following improvement for individual criterion ($Imp_x^y(s_{c_i})$): .

$$Imp_x^y(s_{c_i}, req) = \begin{cases} (1 + (1 - SAT_x^y(s, tra^{'})) * \alpha_3) * Imp_x^y(s_{c_i}, tra^{'}), & SAT_x^y(s, tra^{'}) < \Theta_3 \\ & \text{and} \quad rw_x(s_{c_i}) \geq \phi_3 \\ (1 + SAT_x^y(s, tra^{'} * \beta_3) * Imp_x^y(s_{c_i}, tra^{'}), & SAT_x^y(s, tra^{'}) \geq \Omega_3 \\ & \text{and} \quad rw_x(s_{c_i}) < \phi_3 \\ (1 + \gamma_3 * (1 - SAT_x^y(s, tra^{'}) * \alpha_3) * Imp_x^y(s_{c_i}, tra^{'}), & SAT_x^y(s, tra^{'}) < \Theta_3 \\ & \text{and} \quad rw_x(s_{c_i}) < \phi_3 \\ (1 + \zeta_3 * SAT_x^y(s, tra^{'} * \beta_3) * Imp_x^y(s_{c_i}, tra^{'}), & SAT_x^y(s, tra^{'}) \geq \Omega_3 \\ & \text{and} \quad rw_x(s_{c_i}) \geq \phi_3 \end{cases} \tag{5.14}$$

- $Imp_x^y(s_{c_i}, req)$ is the calculated improvement by y for criterion c_i of task s in response to the request req made by x at time instance t based on x's feedback. Where

$$0 \leq Imp_x^y(s_{c_i}, req) \leq (MaxUG^y(s_{c_i}) - MinUG^y(s_{c_i})) \tag{5.15}$$

- The attraction-factor α_3 is a small increment factor such that $0 \leq \alpha_3 \leq 1$. Trustees use α_3 to increase proposed utility gain for unsatisfied trustors with respect to important criteria.

- The profit-making-factor β_3 is a small decrement factor, such that $-1 \leq \beta_3 \leq 0$. Trustees use β_3 to propose utility gain for unsatisfied trustors with respect to less important criteria to increase the profits they make.

- The attraction-factor-scaling γ_3 and the profit-making-factor-scaling ζ_3 are small scaling factors, such that $0 \leq \gamma_3 \leq 1$ and $0 \leq \zeta_3 \leq 1$.

- We refer to α_3 and β_3 as the engagement factors
- It is clear that values of those factors are application dependent. As a general guideline, we propose that $\mid \alpha_3 \mid \geq \mid \beta_3 \mid$ to emphasize that satisfying the needs for trustors is more important than making profits.

Because $SAT_x^y(s, tra')$ is a percentage, equation (5.14) uses $(1 - SAT_x^y(s, tra'))$, as a factor for enhancing the performance when x is not satisfied. This way, improvement is higher for lower values of $SAT_x^y(s, tra')$, taking into account that α_3 cannot be negative. On the other hand, by the same equation, improvement is reduced more for higher values of $SAT_x^y(s, tra')$ because β_3 cannot be positive.

To reduce the effect of trustors with extreme demands and help defend against potential misleading information provided by a particular trustee, MCTE considers the general needs of all transaction partners. The provided utility gain improvement for a particular trustor x regarding a specific criterion is a function of the improvement calculated for x and the average improvement provided for all partners for the same criterion.

$$Improvement_x^y(s_{c_i}, req) = \chi * Imp_x^y(s_{c_i}, req) + (1 - \chi) * \frac{\sum_{j=1}^{N_{tr}^y} Imp_j^y(s_{c_i}, tra_j')}{N_{tr}^y} \quad (5.16)$$

Where

- $Imp_j^y(s_{c_i}$ is the calculated total improvement by y for criterion c_i of task s for the most recent transaction between y and j
- N_{tr}^y is the number of trustors having interacted so far with y.

5.5 ITE: An Integrated Trust Establishment Model for MASs

The previous section addressed the multi-criteria nature of tasks for trust establishment using explicit feedback from trustors. The one before it addressed trust establishment in the absence of explicit feedback from trustors. In this section we aim to integrate booth approaches together to benefit from both sources of information. We present An Integrated Trust Establishment Model for MASs (ITE) .

ITE calculates the utility gain improvement per criterion in response to request req made by trustor x for task s based on both implicit and explicit feedback as

$$Improvement_x^y(s_{c_i}, req) = \omega_4 * E\_Imp_x^y(s_{c_i}, req) + (1 - \omega_4) * I\_Imp_x^y(s_{c_i}, req) \quad (5.17)$$

- $I\_Imp_x^y(s_{c_i}, req)$ is the improvement calculated by y for criterion s_{c_i} of task s in response to request req by trustee x based on implicit feedback.
- $E\_Imp_x^y(s_{c_i}, req)$ is the improvement calculated by y for criterion s_{c_i} of task s in response to request req based on explicit feedback from trustor x in previous transactions.
- ω_4 is a positive factor, chosen by y, which determines the weight of each component in the model.

As with TEIF, ITE uses the retention of trustors to adjust the behaviour of trustee(s), such that, if the retention rate of x is less than the unengagement threshold, this indicates that x is not so happy with previous transaction(s). In response, y should put some more effort to encourage x to interact with itself later, especially on important criteria. On the other hand, when the retention of x is above the engagement threshold, then the performance of its transaction partner y can be, carefully, reduced, especially with respect to nonimportant criteria.

5.5.1 Updating Relative Weights

When trustor x is not satisfied by the last transaction, i.e reported satisfaction is less than the satisfaction threshold, and the predicted relative weight of a criterion is low, it is likely that the criterion is more important than what the trustee predicted so far, consequently the trustee y should rapidly increase the predicted relative weight for this criterion in future transaction with this particular trustor x. On the other hand, when x is satisfied, and the predicted relative weight of a criterion is low, y can decrease the predicted value of the relative weight of the criterion in future transaction with x.

The case when x is not satisfied, and the predicted relative weight of a criterion is high can be less obvious, while x can be unsatisfied because of other criterion, this one could have influenced such dissatisfaction if x considers this criterion even more important that what y predicted. On the other hand, it is possible that increasing the predicted relative weight of such a criterion will have no effect of the satisfaction of x, if x is already satisfied with the respect to this criterion. As a compromise, we suggest that y increase the predicted weight slightly in this case. The case when x is not satisfied, and the predicted relative weight of a criterion is high can be seen from a similar perspective and y slightly decrease the predicted value of the relative weight of the criterion in future transaction with x.

$$rw_x^y(s_{c_i}, req) = \begin{cases} (1+\alpha_4) * rw_x^y(s_{c_i}, tra^{'}), & SAT_x^y(s, tra^{'}) < \Omega_4 \\ & \text{and} \quad rw_x(s_{c_i}, tra^{'}) \geq \phi_4 \\ (1+\beta_4) * rw_x^y(s_{c_i}, tra^{'}), & SAT_x^y(s, tra^{'}) \geq \Theta_4 \\ & \text{and} \quad rw_x(s_{c_i}, tra^{'}) < \phi_4 \\ (1+\gamma_4 * \alpha_4) * rw_x^y(s_{c_i}, tra^{'}), & SAT_x^y(s, tra^{'}) < \Omega_4 \\ & \text{and} \quad rw_x(s_{c_i}, tra^{'}) < \phi_4 \\ (1+\zeta_4 * \beta_4) * rw_x^y(s_{c_i}, tra^{'}), & SAT_x^y(s, tra^{'}) \geq \Theta_4 \\ & \text{and} \quad rw_x(s_{c_i}, tra^{'}) \geq \phi_4 \\ (1+\lambda_4 * \alpha_4) * rw_x^y(s_{c_i}, tra^{'}), & Otherwise \end{cases} \tag{5.18}$$

- $rw_x^y(s_{c_i}, req)$ is the calculated relative weight by y for criterion c_i of task s in response to the request req made by x at time instance t based on x's feedback.

- $MaxImp^y(s_{c_i})$ is the maximum possible improvement such that

$$MaxImp^y(s_{c_i}) = MaxUG^y(s_{c_i}) - MinUG^y(s_{c_i}) \tag{5.19}$$

- Ω_4 is y's engagement threshold $(0.5 < \Omega_4)$.

- Θ_4 is y's un-engagement threshold $(0 \leq \Theta_4 \leq 0.5)$.

- α_4 is a small increment factor such that $0 \leq \alpha_4 \leq 1$. Trustees use α_4 to increase the predicted relative weight of a criteria.

- β_4 is a small decrement factor, such that $-1 \leq \beta_4 \leq 0$. Trustees use β_4 to decrease the predicted relative weight of a criteria.

- The attraction-factor-scaling γ_4 and the profit-making-factor-scaling ζ_4 are small scaling factors, such that $0 \leq \gamma_4 \leq 1$ and $0 \leq \zeta_4 \leq 1$.
- λ_3 is a small factor to slightly attract trustors that are neither satisfied nor unsatisfied $0 \leq \lambda_4 \leq 1$
- We refer to α_4 and β_4 as the engagement factors
- It is clear that values of those factors are application dependent.

  As a general guideline, we propose that $\mid \alpha_4 \mid \geq \mid \beta_4 \mid$ to emphasize that satisfying the needs for trustors is more important than making profits.

5.5.2 Improvement Calculation

Trustee y updates (increase or decrease) ret_y^x depending on whether or not this response results in an actual transaction. As with TEIF, a trustor x is considered engaged by trustee y if ret_y^x is greater than or equal y's engagement threshold. Trustor x is considered un-engaged by y if ret_y^x is less than or equal y's un-engagement threshold. Otherwise, the trustor x is neither engaged nor un-engaged to trustee y. The predicted relative weight can be used in a way similar to the one described in subsection 5.4.2, that is if $x \in X_{ue}^y$ and the relative weight of a criterion is above the LOW-WEIGHT threshold (ϕ_4), this indicates that x is not satisfied, and the criterion is important. Therefore, the trustee increases the proposed value for this criterion in future responses to request from x. On the other hand, when s $x \in X_e^y$, and the relative weight of a criterion is below HIGH-WEIGHT threshold (ψ_4), this indicates that x is satisfied, and the criterion is not so important. Therefore, y can slowly decrease the proposed value for this criterion in future responses to request from x.

$$I\_Imp_x^y(s_{c_i}, req) = \begin{cases} (1+\alpha_4) * I\_Imp_x^y(s_{c_i}, tra^{'}) & , x \in X_{ue}^y \\ & \text{and} \quad rw_x(s_{c_i}) > \phi_4 \\ (1+\beta_4) * I\_Imp_x^y(s_{c_i}, tra^{'}) & , x \in X_e^y \\ & \text{and} \quad rw_x(s_{c_i}) < \psi_4 \\ I\_Imp_x^y(s_{c_i}, tra^{'}) & , \text{otherwise} \end{cases} \tag{5.20}$$

- If $I\_Imp_x^y(s_{c_i}, req)$ calculated by equation 5.20 is greater than $(MaxUG^y(s_{c_i}) - MinUG^y(s_{c_i}))$, $I\_Imp_x^y(s_{c_i}, req)$ will be reset to $(MaxUG^y(s_{c_i}) - MinUG^y(s_{c_i}))$.

Based on predicted relative weights, $E\_Imp_x^y(s_{c_i}, req)$

$$E\_Imp_x^y(s_{c_i}, req) = rw_x^y(s_{c_i}) * MaxImp^y(s_{c_i}) + MinUG^y(s_{c_i}) \tag{5.21}$$

- $E\_Imp_x^y(s_{c_i}, req)$ is the calculated improvement by y for criterion c_i of task s in response to the request req made by x at time instance t based on x's feedback. Where

$$0 \leq E\_Imp_x^y(s_{c_i}, req) \leq (MaxUG^y(s_{c_i}) - MinUG^y(s_{c_i})) \tag{5.22}$$

5.6 Updating the Engagement Factors

The engagement factors α_k and β_k, $k \in 2, 3, 4$, are used to adjust the improvement of proposed utility gain for criteria per trustor. In addition to their role in attracting trustors by promising higher utility gain for those with a low satisfaction rate, the use of the engagement factors can help trustees control the offered utility gain in order to enhance profit. When there is a general trend of dissatisfaction among trustors, trustee y should speed up the improvements it makes of proposed utility gain for trustors to avoid becoming an untrusted partner from their perspective,

especially if trustors use testimonies from witnesses to evaluate the trust of trustees. Furthermore, y should be very careful when attempting to make a profit to reduce the chances of extra dissatisfaction. Therefore, y models the general trend of satisfaction among trustors using the function g^y, which we call the general satisfaction trend function of y. Trustee y may update g^y periodically or after each response to a collaboration request with a trustor x.

An increase in the average satisfaction rate indicates a general trend of satisfaction among trustors. In this case, y expects that a low satisfaction level of a trustor x is an individual behaviour rather than a general attitude. As a result, y can still enhance the improvements of proposed utility gain it makes for x. However, y can decelerate the improvements of proposed utility gain it makes.

When Explicit feedback available, the general trend of satisfaction function of y is:

$$g^y(s,t) = \frac{\overline{\hat{SAT_x^y}(s,t)}}{\overline{SAT_x^y(s,t)}} \tag{5.23}$$

- $\overline{SAT_x^y(s,t)}$ is the most recently calculated average satisfaction rate such that $\overline{SAT_x^y(s,t)} = \frac{\sum_{x=1}^{Ntrax^y(s,t)} SAT_x^y(s,tra)}{Ntra_x^y(s,t)}$ where

  where $Ntrax^y(s,t)$ is the number of transactions between x and y for task task s up to and including time t.

- $\overline{SAT_x^y(s,t')}$ is the average satisfaction rate calculated time t.

Consequently, α_k is updated as

$$\alpha_k = (\alpha_k - \ln(g^y(s,t))) \tag{5.24}$$

and the profit-making-factor β_k is calculated as

$$\beta_k = (\beta_k + \ln(g^y(s,t))) \tag{5.25}$$

- The natural logarithmic function used as it has negative values for input parameters in (0,1) and its value changes quickly in that range, bur slower after that $(1, \infty)$ which allows for dynamic response of the model to changes in g^y.

5.7 Performance Analysis

The effectiveness of various models needs to be assessed under different environmental conditions, research that present performance analysis of trust establishment models evaluate them in a proprietary manner based on simulation, as it is challenging to obtain suitable real-world data sets for the comprehensive evaluation. Simulation experiments differ among various works, not only in the simulation parameters but also in nature. Generally, there is no agreed-on benchmarks to enable comparing different results.

5.7.1 Performance Measures

If transactions are based on trust, trustworthy trustees will have a greater impact on the results of transactions. In such situations, building a high trust may be an advantage for rational trustees. Rational trustees need to trade off the cost of building and keeping trust in the community with the future gains from holding the trust acquired. Therefore, to study the performance of the proposed model, we use the following measures:

- Trust evaluation: This metric is calculated as the average direct trust evaluation for all trustees that use a particular model. We agree that exact trust evaluation is highly dependent on the adapted trust evaluation model used by trustors, however we include this metric in this study as indicator in situations where trustors use a general purpose, probabilistic evaluation model.

- Average delivered utility gain: This measure indicates the efforts needed to achieve the enhancement in the percentage of overall transactions. It is arguable that an honest trustee

y can achieve a higher percentage of overall transactions if it is committed to delivering the highest possible utility gain transaction. However, providing greater utility gain usually incurs extra cost. Therefore, a rational trustee will attempt to achieve a higher number of transactions while keeping the provided utility gain as low as possible. In this work, the average of delivered utility gain per transaction is defined as the summation of all delivered utility gain values in all transactions be the group of trustees empowered with a particular model divided by the number of those transactions.

$$AverageUG_l(s) = \frac{\sum_{y \in Y} \sum_{x \in X} \sum_{k=1}^{NTR_y^x} UG_x^y(k)}{\sum_{y \in Y} \sum_{x \in X} NTR_y^x} \tag{5.26}$$

Where

- l is the model used by trustees.
- NTR_y^x is the number of transactions between x and y.

- Total number of good transactions in the system: When comparing different trust establishment models, the higher the number of good transactions, the more successful the model in predicting the needs of trustors. This metric is calculated as the summation of all transactions took place in the system where the demands of trustors fulfilled completely. Other metrics like percentage of good transactions and the average number of good transactions per trustee, can be seen as derived metric form this one, or just a different presentation based on this metric.

- Total number of bad transactions in the system: When comparing different trust establishment models, the lower the number of bad transactions, the more successful the model in predicting the needs of trustors. This metric is calculated as the summation of all transactions took place in the system where the demands of trustors are not fulfilled completely, i.e either fulfilled partially or not fulfilled at all. Other metrics like percentage of bad

transactions and the average number of bad transactions per trustee, can be seen as derived metric form this one, or just a different presentation based on this metric.

5.7.2 Simulation Environment

We use a scenario-simulator approach [144] to compare the performance of different models. Scenarios are pre-generated and not modified during the simulation phase. The generation of scenario files encodes different parameters, such as the number of trustees, the number of trustors, the profile of each agent and so on. These scenarios are then given to the simulator. Thus, multiple simulations, implementing different models, can be run from the same scenario. By fixing every aspect of a scenario run, the only differences between runs will be trust establishment specific.

Generally speaking, a scenario is a sequence of request for transactions followed by transactions. Each request specifies a service name and the agent seeking that service. The choice of a trustee to interact with will be made at simulation runtime by trust evaluation model. The value for different criterion of the selected trustee's response is determined by the trust establishment model at simulation time.

For simulation, we use the discrete-event MAS simulation toolkit MASON [63] with trustees providing services, and trustors consuming services. We assume that the performance of an individual trustee in a particular service is independent of that in other services. To reduce the complexity of the simulation environment, it is assumed that there is only one type of services in the simulated system. All trustees offer the same service with, possibly, different performances. Network communication effects are not considered in this simulation. Each agent can reach each other agent. The simulation step is used as the time value of transactions. Transactions that take place in the same simulation step are considered simultaneous. Locating trustees and other agents are not part of the proposed model, as agents locate each other through the system. Trustors can request any number of trustees to bid. No trust certification mechanism exists, and

third-party witnesses are assumed, to be honest.

For trust evaluation, trustors assumed to use a simple probabilistic trust evaluation

$$trust_x^y = 0.5 * directTrus_x^y t + 0.5 * indirectTrust_x^y \tag{5.27}$$

- Direct trust, or direct experience, is the aggregated percentage of satisfaction from transactions performed so far with the trustee.
- Indirect trust, or indirect experience, represents the reputation of the trustee in the community, and is calculated as the average direct trust value of those who previously interacted with the trustee.

Initially, trustors use neutral trust rating for every trustee. Therefore, initial trust is set to

$$trust_x^y = \frac{minTrust + maxTrust}{2} \tag{5.28}$$

Having selected an transaction partner y, agent x then interacts with y and gains utility. A trustee can serve many users at a time. A trustor does not always use the service in every round. The probability a trustor requests a service is referred to as activity level. After a transaction, x updates the trust evaluation of its transaction partner. At the end of each simulation step, trustees update the average retention rates of trustors if necessary. Charted values calculated as the averaged value for ten different randomly generated scenarios with the same base configuration.

Table 5.2 presents the base values for the number of agents and other parameters used in the proposed model and those employed in the environment. When testing the effect of a particular parameter, others are set to those base values.

Table 5.2: Base values for simulation parameters

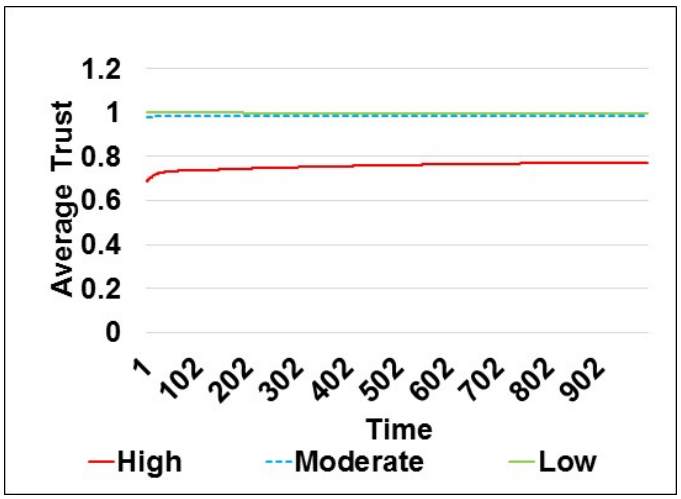
Parameter	Value
Number of Agents	100
Number of Trustees	10
Number of Trustors	90
Percentage of High Demand Trustors	30%
Percentage of Low Demand Trustors	40%
Percentage of Normal Demand Trustors	30%
Percentage of Highly Active Trustors	30%
Percentage of Regular Active Trustors	40%
Percentage of Low Active Trustors	30%
Number of simulation steps	1000
Number of criteria	10
Ω_2	0.9
Θ_2	0.5
Ω_3	0.9
Θ_3	0.9
Ω_4	0.9
Θ_4	0.5
ψ_4	0.5
ϕ_4	0.5
Initial value of α_k	0.01
Initial value of β_k	-0.01
μ	1.5
ν	0.5
MinUG	0.1
MaxUG	1

5.7.3 Experimental results

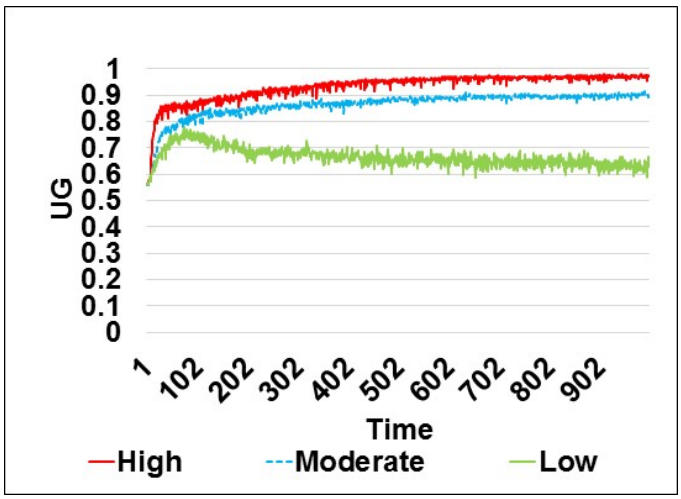
Effects of Trustors' Demand Level

The level of service required to satisfy needs of trustors can vary within the same environment, as well as in different environments. For example, in the context of a smart grid, the various electrical vehicle may require different service levels of charging stations depending on factors like distance to destination, urgency of the trip and so on. To study the effects of trustors' levels of demand on the behaviour of the proposed model, we analyze three extreme cases where all trustors belong to the same demand category: all with high demand, all with low demand or all intermediate (normal) demand such that highly demanding trustors requires at least 65% of the maximum possible UG for each criterion, low demanding trustors requires no more than 35% of the maximum possible UG for each criterion and regular demanding trustors requires something in between.

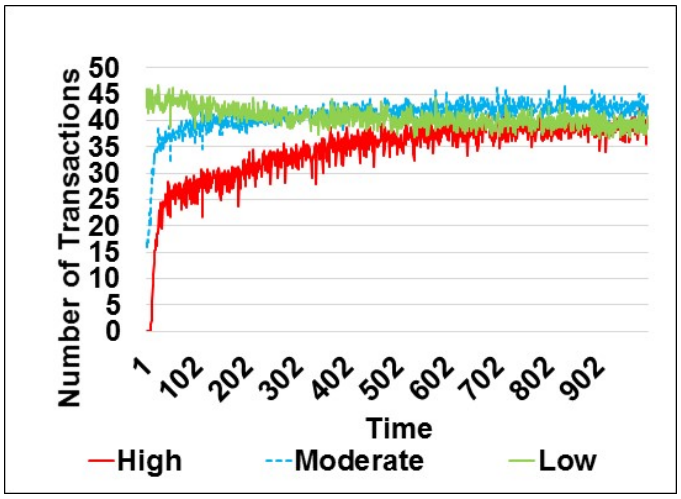
Figure 5.1 presents the performance TEIF under different levels of demands. Sub-figure (a) shows that trustees empowered with TEIF have a high level of average trust. Even though the model causes trustees to provide higher UG for highly demanding trustors, these trustors result in the lowest level of trust compared to low and moderate demanding trustors scenarios, as shown in sub-figures (a and b). Even though the model response differently to various levels of demands, the response of the model is not as dynamic as the demands of trustors. This can be seen by noting the UG provided for low demanding trustors that requires no more than 35% of the maximum possible UG, but the model provides more than 60% in its transactions. Something similar can be noticed for trustors with moderate demands. As would be expected, the number of good transactions is related to the levels of UG provided, whereas the number of bad one is reversely related to the levels of UG provided as shown in sub-figures (b, c, and d). This is because TEIF depends on retention of trustors, and once the retention indicator of a trustor reaches the engagement threshold, the trustee reduces the UG it provides, which may result in a (slightly) bad transaction.



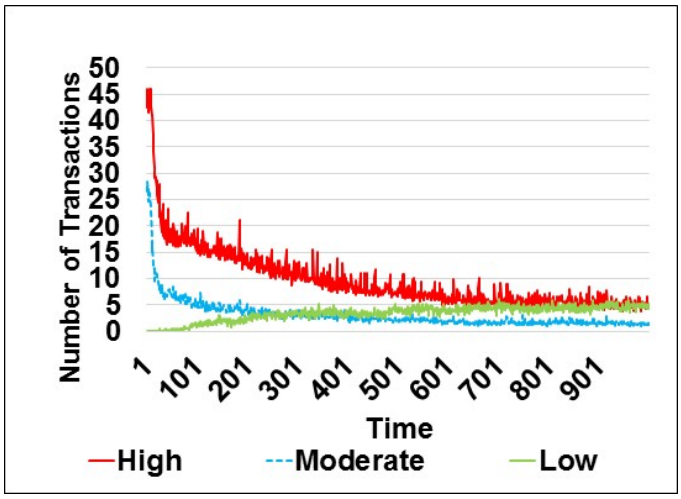
(a) Average Trust



(b) UG



(c) Number Of Good Transaction

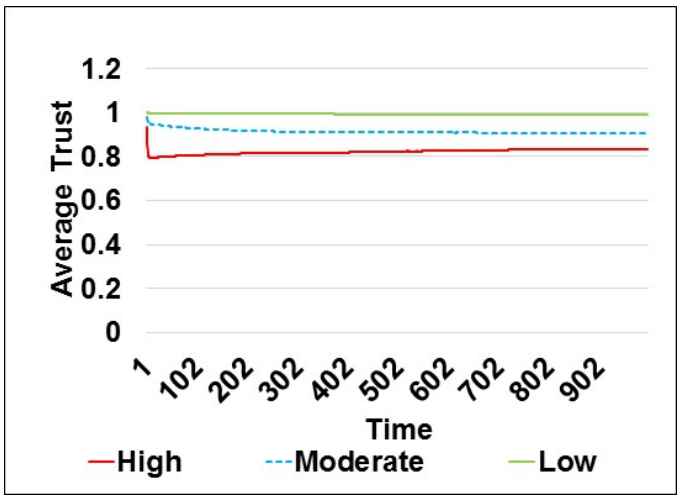


(d) Number Of Bad Transaction

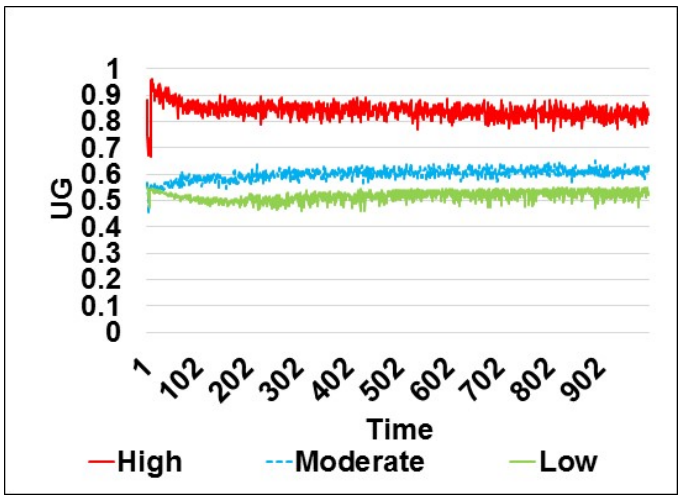
Figure 5.1: Effect of Trustors' Demand - TEIF

Figure 5.2 presents the performance MCTE under different levels of demands. Sub-figure (a) shows that trustees empowered with MCTE have a high level of average trust. Even though the model causes trustees to provide higher UG for highly demanding trustors, these trustors result in the lowest level of trust compared to low and moderate demanding trustors scenarios, as shown in sub-figures (a and b). The model response differently to various levels of demands in way better than TEIF, and the response of the model is close to the demands of trustors. This can be seen by noting the UG provided for low demanding trustors that requires no more than 35% of the maximum possible UG, where the model provides more about 50% of maximum UG in its transactions, for trustors with moderate demands, the model provides about 60% of maximum UG and for highly demanding trustors, the model provides around 80% after a transaction period. The dynamic response of the models can be referred to the use of explicit feedback from trustor. However, because the provided UG depends on both the partner's feedback and the general feedback form the community, the model's response does not match the demand level of individual trustors. It worth noting that keeping a relatively high level of trust, despite the high number of "bad" transactions, and consequently, the low number of "good" transactions (sub-figures c and d) suggests that unfulfilled criterion either not important, or highly, rather than completely, fulfilled or both.

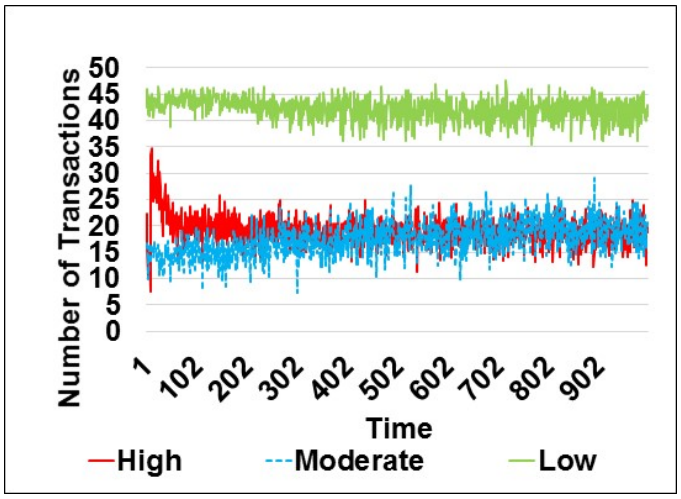
Figure 5.2 presents the performance ITE under different levels of demands. Sub-figure (a) shows that trustees empowered with ITE have a high level of average trust. Even though the model causes trustees to provide higher UG for highly demanding trustors, these trustors result in the lowest level of trust compared to low and moderate demanding trustors scenarios, as shown in sub-figures (a and b). The model response differently to various levels of demands in way better than TEIF and MCTE, and the response of the model is close to the demands of trustors. This can be seen by noting the UG provided for low demanding trustors that requires no more than 35% of the maximum possible UG, where the model provides more about 40% of maximum UG in its transactions after a transnational period, for trustors with moderate demands, the model provides slightly more than about 50% of maximum UG and for highly demanding trustors, the



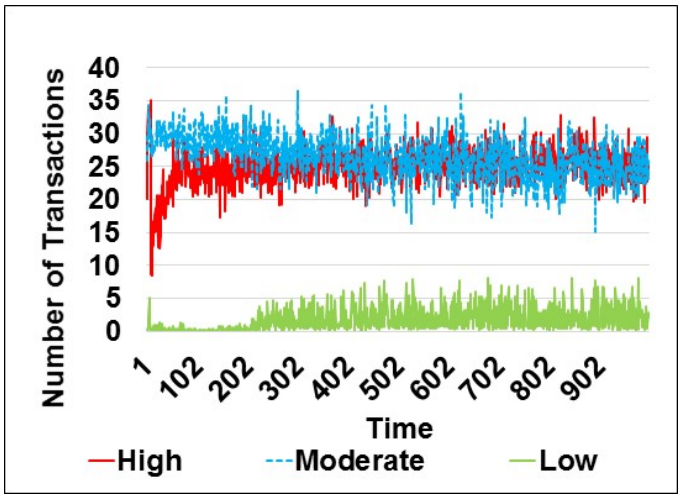
(a) Average Trust



(b) UG



(c) Number Of Good Transaction



(d) Number Of Bad Transaction

Figure 5.2: Effect of Trustors' Demand - MCTE

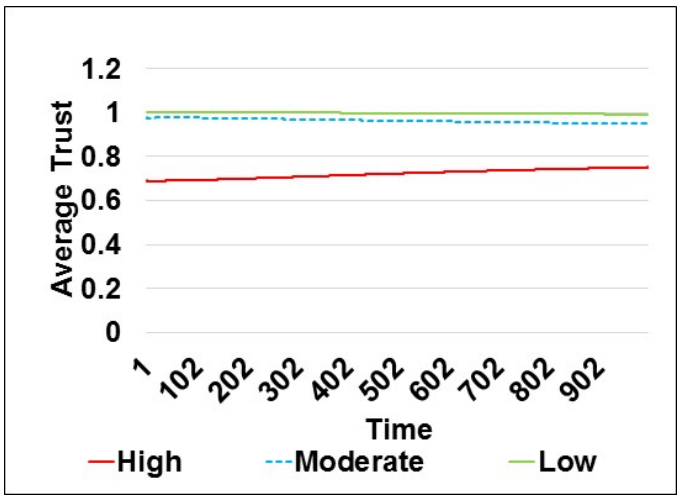
model provides around 70% after a transaction period. The dynamic response of the models can be referred to the use of explicit feedback from trustor to tune relative weights of criteria, which, in turn, used with the retention indicator of the requesting trustor to predict the provided UG depending ”mainly” on the partner's feedback. We say ”mainly” because the retention indicator depends partially of the average retention of other trustors.

It worth noting that keeping a relatively stable level of trust for low demand trustors, despite the decline in the number of good transactions, and consequently, the increase in the number of bad transactions (sub-figures c and d) suggests that unfulfilled criterion either not important, or highly, rather than completely, fulfilled or both. Detecting the importance of different criterion and providing various levels of UG accordingly helps trustees to achieve a relatively high level of trust at a reasonable provided UG.

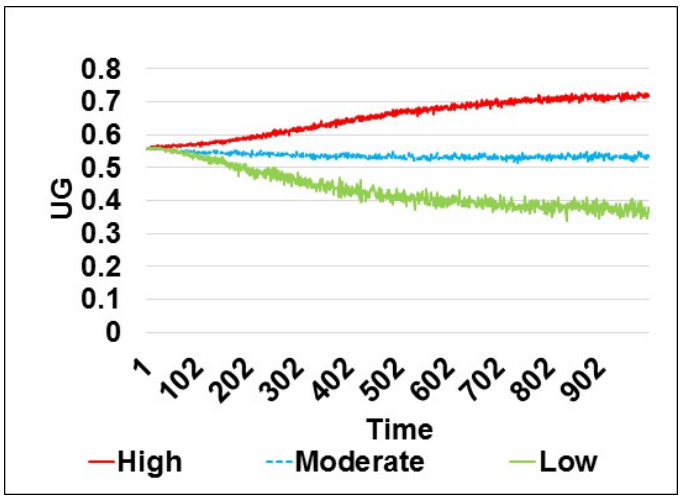
The Effects of Trustors' Activity Level

It is possible that trustors do not always use the service in every round. While some trustors can be highly active, some others can have a low or intermediate level of activity. To study the effects of trustors' activity level on the behaviour of the proposed model, we analyze three extreme cases where all trustors belong to the same demand category: all with high activity level, all with intermediate (normal) activity level, or all with low activity level. In each round, highly active trustors request the service with a probability of at least 65%, low active ones request the service with a probability of at most 35%, while trustors with a regular level of activity request the service with probability in between.

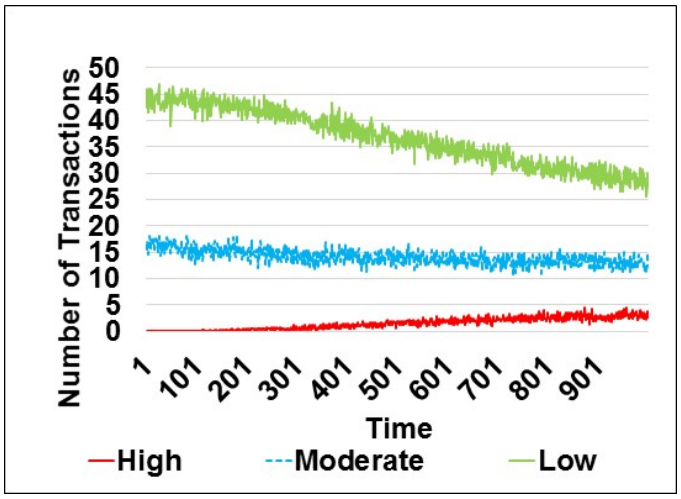
Figure 5.4 presents the performance TEIF under different levels of trustors' activity. Sub-figures (a and b) show that the variation of trustors' activity doe not have significant effect on the provided UG as well as on the level of trust for TEIF empowered trustees.This is because the model uses a historical window of most recent transaction, rather than using all transactions since its creation.



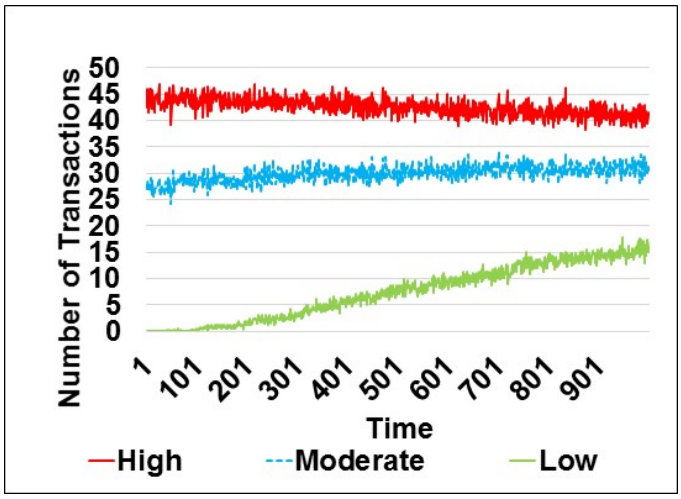
(a) Average Trust



(b) UG

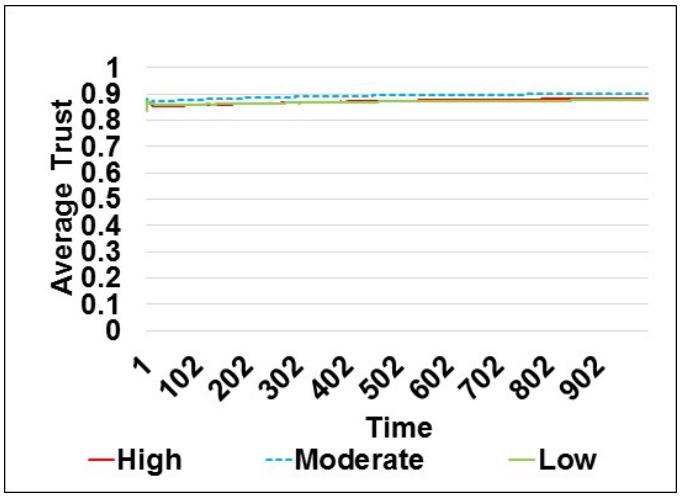


(c) Number Of Good Transaction

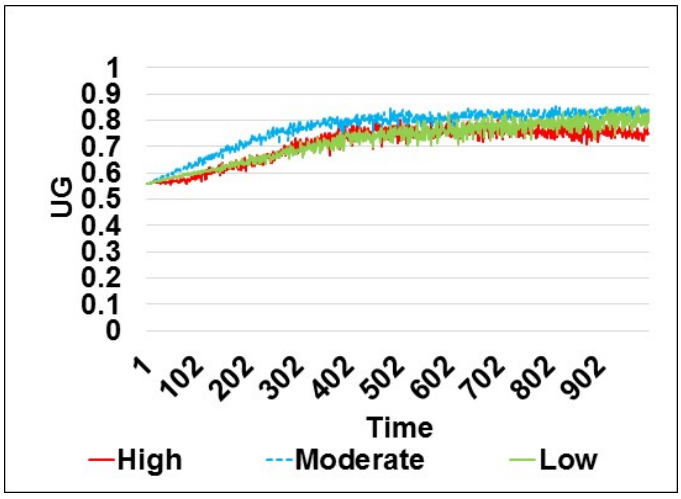


(d) Number Of Bad Transaction

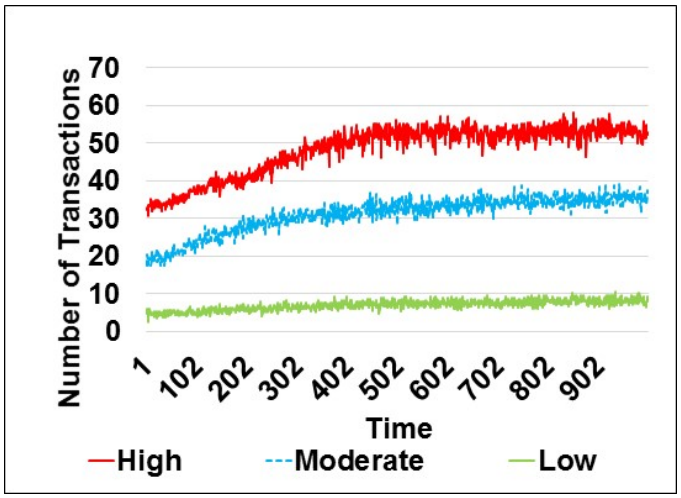
Figure 5.3: Effect of Trustors' Demand - ITE



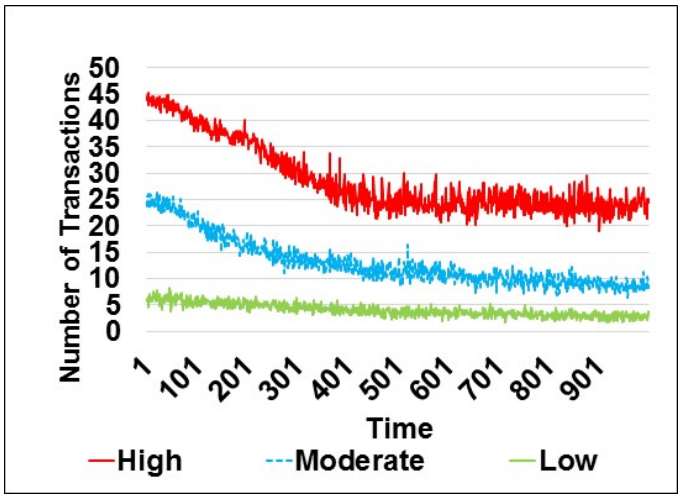
(a) Average Trust



(b) UG



(c) Number Of Good Transaction

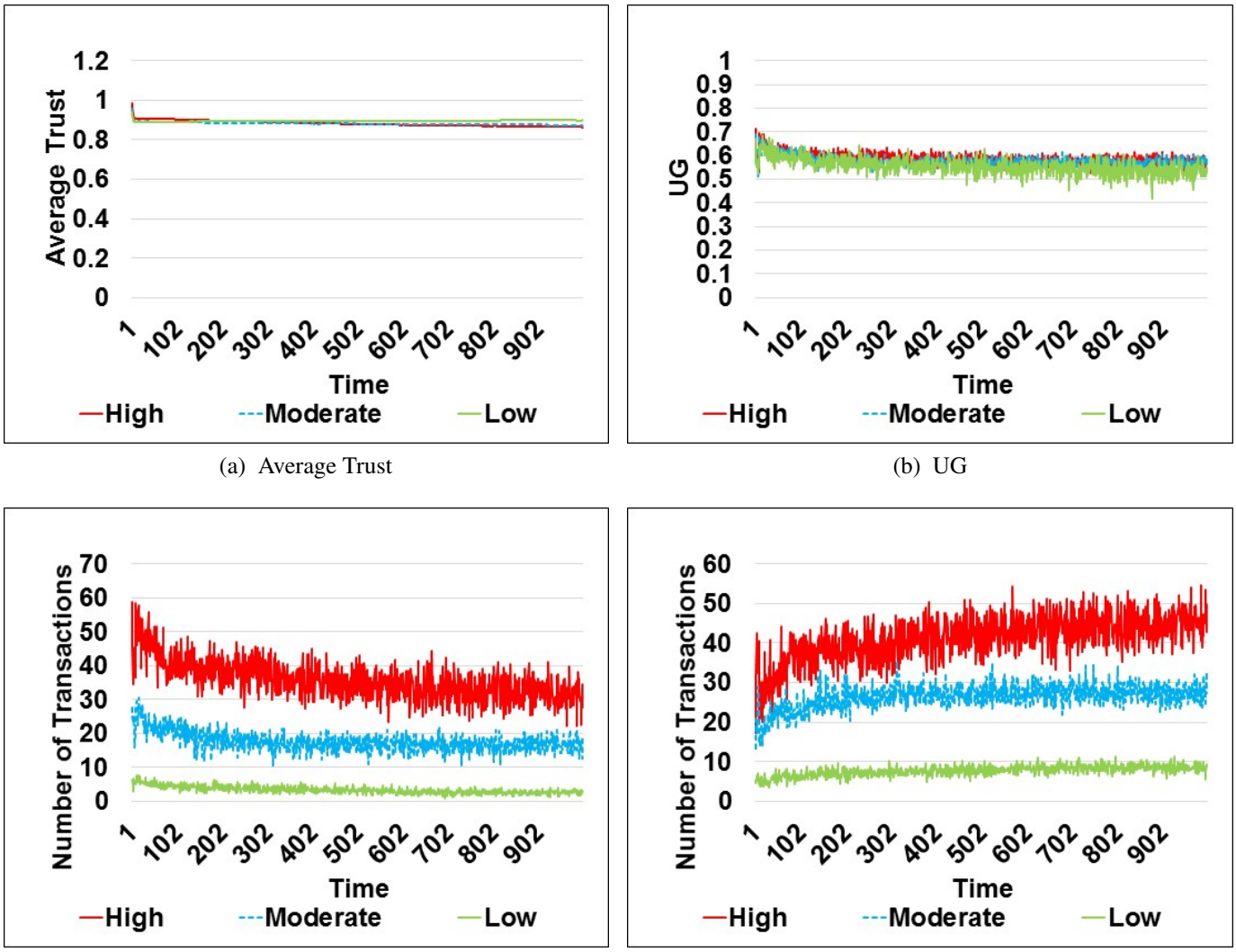


(d) Number Of Bad Transaction

Figure 5.4: Effect of Trustors' Activity - TEIF

As would be expected, the number of good transactions increases more rapidly for highly active trustors and the number of bad transactions declines faster as shown sub-figures (c and d), simply because the number of transactions, in total, is larger. Therefore, the general behaviour of the model is exaggerated.

Figure 5.5 presents the performance MCTE under different levels of trustors' activity. As with TEIF, the variation of trustors' activity doe not have significant effect on the provided UG as well as on the level of trust for MCTE empowered trustees, as shown sub-figures (a and b). This is because the model depends on explicit feedback and does not take retention into account



(a) Average Trust

(b) UG

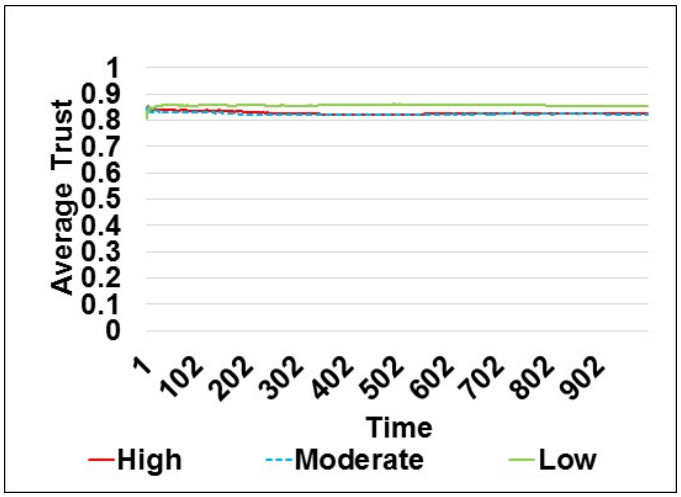
(c) Number Of Good Transaction

(d) Number Of Bad Transaction

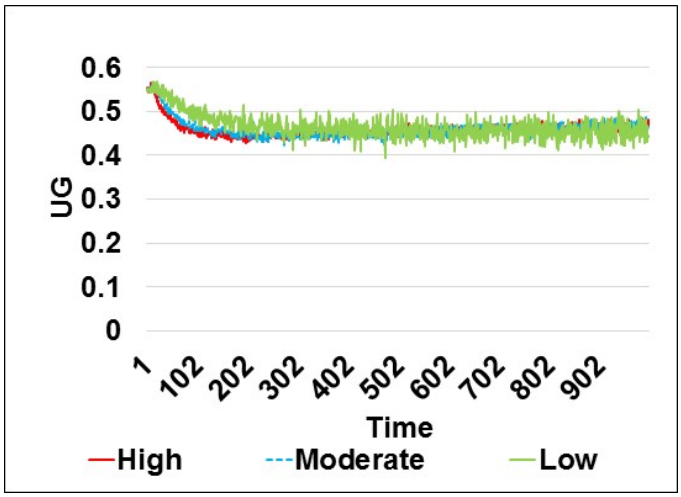
Figure 5.5: Effect of Trustors' Activity - MCTE

when calculating offered UG. The variation in the number of good and bad transactions presented in sub-figures (c and d) is a natural consequence of the increase in the number of transaction as the activity of trustors increases.

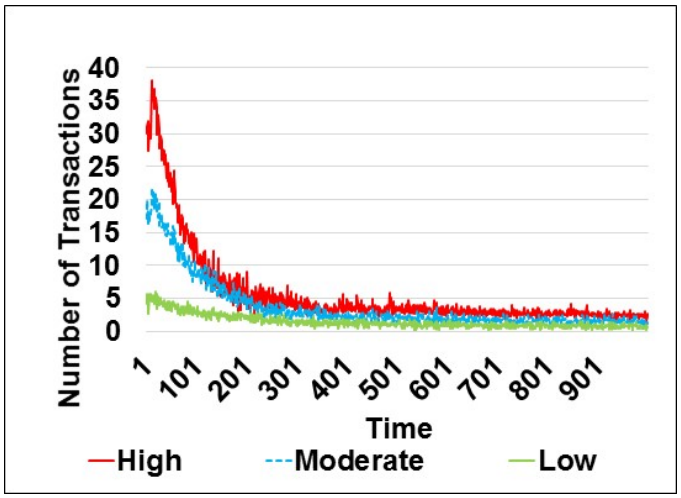
Figure 5.6 presents the performance ITE under different levels of trustors' activity. As with TEIF and MCTE, the variation of trustors' activity doe not have significant effect on the provided UG as well as on the level of trust for MCTE empowered trustees, as shown sub-figures (a and b) because the model depends on explicit feedback, and when considering retention, the model uses a historical window of most recent transaction, rather than using all transactions since its



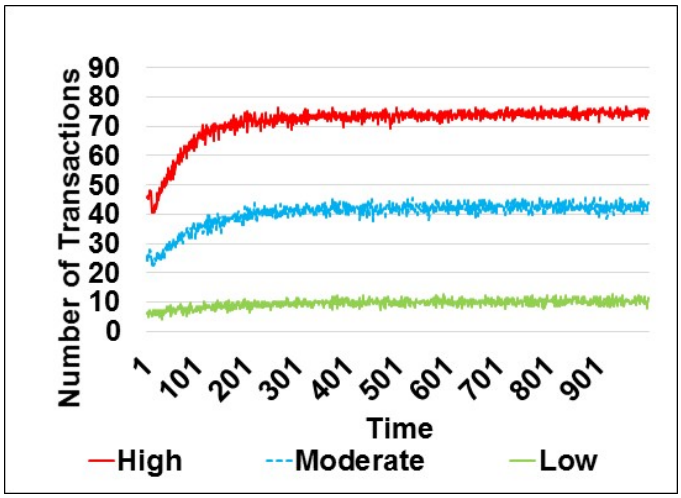
(a) Average Trust



(b) UG



(c) Number Of Good Transaction



(d) Number Of Bad Transaction

Figure 5.6: Effect of Trustors' Activity - ITE

creation for calculating offered UG.

It worth noting that the number of good transactions presented in sub-figures (c) decreases significantly and the rate of decreasing relate to the activity level of trustors. As mentioned earlier, because the model decrements provided UG when the trustor seems to be engaged. While the rate of decrements is relatively high for highly active trustors because of the increase in the number of transaction as the activity of trustors increases. The mirror of effect can be seen for the number of bad transactions presented in sub-figures (d) which increases significantly for highly active trustors.

TEIF can be promising for situations where explicit feedback doesn't exist, and task are, or can be treated as, tasks of single criterion. However, it can be costly if repetitive interactions don't tack place. Also, it is more effective for trustors with moderate demands than those with high demands. TEIF works better for honest trustors with homogeneous activity level, where interaction based on trust. This way, a reduced activity level can be attributed to reduction of satisfaction. Using high value for attraction factor can increase the level of trust for trustees at the price of extra cost in the form of extra utility gain provided.

On the other hand, MCTE can be promising for situations where explicit feedback exists, and task have multi criteria. However, it can be costly if trustors report misleading feedback. MCTE works better for honest trustors with homogeneous levels of demands as it depends, partially, on the average demand of partners to reduce the effect of trustors with extreme demands. As it depends on gradual enhancement, the models work better for situations where repetitive interactions tack place. Increasing the satisfactory threshold and reducing te importance threshold, each of which can enhance the trust level at the price of extra cost. As with other trust establishment models proposed in this work, using high value for attraction factor can increase the level of trust for trustees at the price of extra cost in the form of extra utility gain provided.

ITE attempt to benefit from both explicit and implicit feedback and reduce side effects for tasks that have multi criteria. The model can reduce the effects of a trustor reporting misleading explicit feedback because the model depends partially on implicit feedback, or the other way around. However, the model cannot tolerate both misleading explicit feedback and intentionally misleading return pattern. ITE works better for honest trustors with homogeneous activity and demand levels where it can achieve a good level of trust compared to MCTE at a reasonable cost. Increasing the satisfactory threshold and reducing te importance threshold, each of which can enhance the trust level at the price of extra cost.

5.8 Summary

In this chapter, we presented a trust establishment model for MASs using reinforcement learning and implicit feedback for situations where explicit feedback is not available, or not desirable. The presented model allows trustees to alter their behaviour based on both: the behaviour of the partner trustor, and use the general conduct of the community of trustors to speed up or slow down such adjustments. Then, we presented a distributed, multi-criteria trust establishment model for situations where trustors are willing to provide only aggregated explicit satisfaction feedback, without detailing the importance or weight of individual service criteria. The proposed model uses the provided feedback from trustors regarding how satisfied they were with recent transactions to predict the importance of different service dimensions for trustors and adjust the behaviour of trustee(s) accordingly. After that, we presented a trust establishment model that integrates direct and indirect sources of information. The presented model allows trustees to alter their behaviour based on the aggregated explicit satisfaction feedback as well as the behaviour of trustors. The aim of trustees is to enhance their trust with the hope to be selected for future transactions. The chapter presented the performance analysis, using simulation, for the presented models. Further comparisons will be presented in (Chapter 6). Even though we used Q-Learning to update the performance of trustees in response to feedback from trustors. We believe that a computational engine similar to the one used in the previous chapter based on fuzzy logic and the combination of fuzzy logic and Q-learning can have a promising performance due to the ability of fuzzy logic in modelling uncertainty.

Chapter 6

Discussion

6.1 Introduction

In Chapter 4 we described our trust evaluation model using fuzzy logic and reinforcement learning and evaluated our model using simulating in different configurations. In Chapter 5 we presented our trust establishment models using implicit and explicit feedback and evaluated our models using simulating in different configurations. In this chapter we present a discussion on the value of our models. We contrast our proposed models with other related models, and compare the performance using simulations. Then we present potential application areas, namely Web services and Internet of Agent.

6.2 Compare and Contrast of Trust Evaluation Model

Trust and Reputation in the Context of Inaccurate Information Sources (TRAVOS) [123] and FIRE [39] are well-known decentralized, general purpose trust evaluation models for Multi-Agent Systems(MASs) that takes into account direct experiences and third-party information. TRAVOS depends on beta distributions to predict the likelihood of honesty of a trustee [48]. If

the trustor's confidence in its evaluation is below a predefined threshold, the trustor seeks advises from witnesses.

TRAVOS computes the reliability of witnesses via the direct experience of interaction between the trustor and witnesses and discards inaccurate advises. Unfortunately, it takes certain time for the trustor to recognize the inaccuracy of the provided reports from the previously trusted witness agents [50].

FIRE takes into account multiple sources of information. FIRE categorizes trust components into four categories; direct experience called Interaction Trust, Witness Reputation, Role-based Trust and Certified Reputation. The model assumes that witnesses are honest and willing to cooperate and use weighted summation to aggregate trust components [39]. Unfortunately, if a trustee proposes some misbehaving referee for certified reputation, this source of information and be misleading to the trustor [50].

Reinforcement learning explicitly considers the problem of a trustor that learns from interaction within an uncertain environment to achieve its goal [131]. The trustor can discover which actions yield the most reward via a trial-and-error search rather than being told which actions to take as in most forms of machine learning [127]. Consequently, reinforcement learning deemed to be a naturally suitable learning method for trust evaluating agents in MASs because of this special characteristic [130].

A well-known issue with reinforcement learning based trust models is that trustors do not quickly recognize the environment changes and adapt with new settings. To address this shortcoming, Decentralized Trust Evaluation Model for Multi-Agent Systems Using Fuzzy Logic and Q-Learning (DTMAS) uses a technique similar to regret described in [69] to improve responses of reinforcement learning based models to dynamic changes in the system. We propose suspending the use of a trustee immediately as a response to an unsatisfactory transaction with the trustee. The suspension is temporary, and its period increases as the transaction importance increases. Similar to regret [69], it represents an immediate reaction of a trustor when it is not

satisfied with an interaction with a trustee. However, unlike the use of regret in [69], trustors do not depend on of any expressed feel of sorry from trustees, as the regret can be misleading if the trustee communicate false regret value. Furthermore, the suspension is more aggressive. No interactions are performed with suspended trustees as long as it is suspended. On the other hand, while forgiveness is used to reduce the effect of regret in [69], suspension decays with time only to avoid being misled by a false feel of sorry expressed by the trustee and to allow the effect of accidental misbehaviour to phase out and magnify the impact of a behaviour change as testimonies from witnesses are aggregated.

Fuzzy logic offers the ability to handle uncertainty and imprecision effectively and is, therefore, considered appropriate for reasoning about trust [33]. Fuzzy inference copes with imprecise inputs and allows inference rules to be specified using imprecise linguistic terms, such as ”very high” or ”slightly low” [33].

A multi-criteria trust model for web-based social networks based on fuzzy logic presented in [115]. Two-level hierarchical Fuzzy Logic Systems (FLSs) are used to address imprecise and uncertain information and determine trust evaluation. Various personality attributes of individuals like their attitude, behaviour, beliefs, honesty and responsibility were considered. As with many trust models, they aggregated direct and indirect experience trust at the second level of the FLS. Unlike most other trust models, the aggregation as accomplished using fuzzy logic. The proposed model targets web-based social networks and rely on specific criteria of the graph representation of these networks such as the in-degree and the out-degree.

The model known as comPrehensive reputAtion-based TRust moOdeL with Fuzzy subsystems (PATROL-F) [122] uses fuzzy logic to account for humanistic and subjective concepts such as the importance of a transaction, the decision in the uncertainty region, and setting the result of interaction in distributed systems. The model is decentralized and takes into account direct and indirect trust factors and considers the credibility of witnesses.

Fuzzy logic has been used is for trust evaluation in [12] and [117], among others. Unlike

existing fuzzy logic based trust evaluation models, DTMAS combines the advantages of both: fuzzy logic and Q-learning for trust evaluation in a distributed MAS, while using third-party information and considering the credibility of those third-party sources.

6.2.1 Experimental Comparison

To compare the performance of the proposed trust evaluation model with other models, we use a simulation environment similar to the one described in 4.3. We compare the proposed model with some of the well-cited generic trust models, namely BRS, TRAVOS, and FIRE as well a PATROL-F . Our simulations of PATROL-F and FIRE are based on the main reference in which they were proposed. For the other models, we adapted the implementation of [40].

Base Scenario

First of all, we would like to compare the performance of DTMAS to other models used in the study in a generic scenario with a mix of good, bad, and regular trustees as in 4.3.2. Witnesses do not attempt to cheat or misbehave. This is considered the base line, were any good trust model should perform well. In this scenario, trust evaluating agents, do not always use the service in every round. The activity level in the range $[0 - 1]$. Highly active trustors request the service with a probability of at least 75%, low active ones request the service with a probability of at most 25%, while trustors with a regular level of activity request the service with probability in between. As with demand levels, 30% of trustors were configured with high activity level. Another ,30% of trustors were configured with low activity level and the remaining 40% of trustors were configured with regular activity level. Similarly, 30% of trustors were configured with high demands, and another 30% of trustors were configured with low demands, were as the remain 40% of trustors were configured with regular demands. Highly demanding trustors requires at least 75% of the maximum possible Utility Gain (UG), low demanding trustors requires no more than 25% of the maximum possible UG and regular demanding trustors requires something in

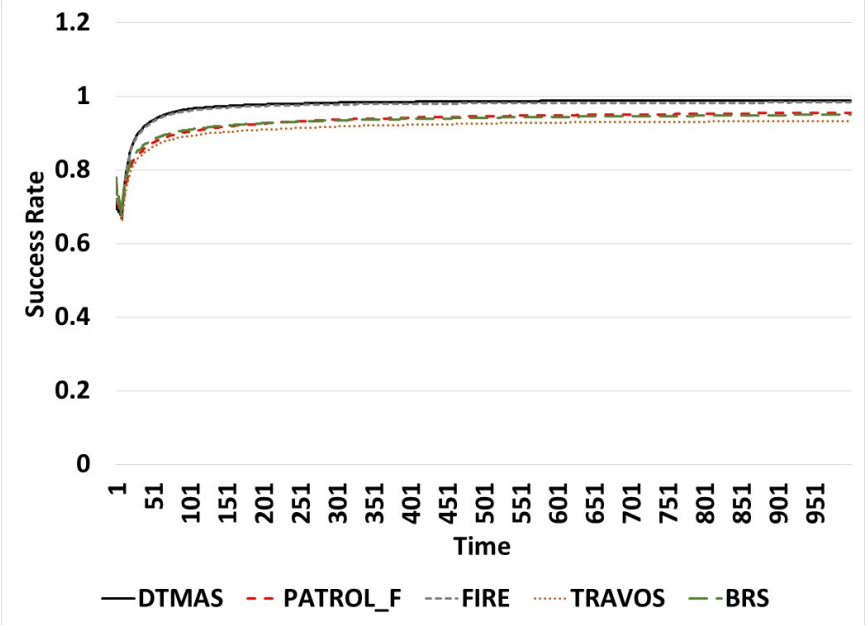


Figure 6.1: Overall Success Rate in the Base Scenario

between.

Figure 6.1 describes the overall success rate of the simulated models as the number of simulation steps increases 1 to 999. The figure shows that, eventually, all models can achieve high success rates. However, DTMAS and FIRE can do that faster. TRAVOS prefers direct experience over feedback from witnesses, when direct experience is available, and tends to reduce dependence of feedback from witnesses. Therefore, it does not fully benefit from such information sources. PATRO-F, DTMAS and FIRE assumes a unidirectional service model and integrate multiple sources of information. They do not tend to avoid feedback form witnesses, as in TRAVOS, and they do not tend to overemphasize such feedback at the price of direct experience as in BRS. The integration of multiple sources of information helps PATROL-F, DTMAS and FIRE achieving a slightly higher accuracy rates within the base scenario configuration.

Misbehaving Trustees Scenarios

When agents are not cheating, all models, generally, can achieve a high success rates. However, trustees may misbehave in different ways. A good trust model should guide trustors to interact

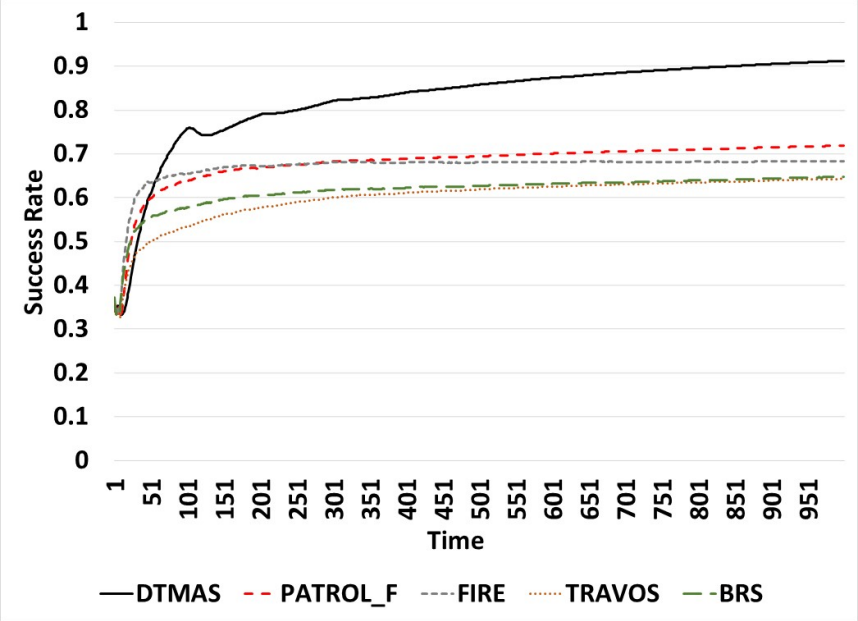


Figure 6.2: Success Rate of Trust Evaluation Models - Purely Random Behaviour of Trustees

with honest trustees, if any, rather than misbehaving ones. However, nothing can be done when none of the trustees is honest. In this set of experiments, only ten percent of trustees are honest, while others are misbehaving. To study the effect of different cheating behaviours, also referred to as attacks, we assume all non-honest trustees use the same cheating strategy. We study the trustee attacks described in 4.3

These Scenarios have ninety percent of trustees are misbehaving. Trust evaluating agents, do not always use the service in every round. The activity level is in the range $[0 - 1]$. As we are studying the effects of trustee attacks, we assume all witnesses are honest and fully cooperative.

Figure 6.2 shows that, figuring out trustworthy agents from a set of trustees with random pattern of cheating is challenging due to lack of systematic behaviour. Fuzzy logic works better than classical probabilistic approach is this case. Therefore, models with fuzzy logic components, such as DTMAS and PATROL-F, outperform probabilistic based models as TRAVOV and BRS. The tendency of FIRE to to explore unseen partners has negative effect of it performance in this case. The performance when cheaters behave randomly only fifty percent of the times is presented in figure 6.3, which is, generally, less sever than the purely random attacks; as the

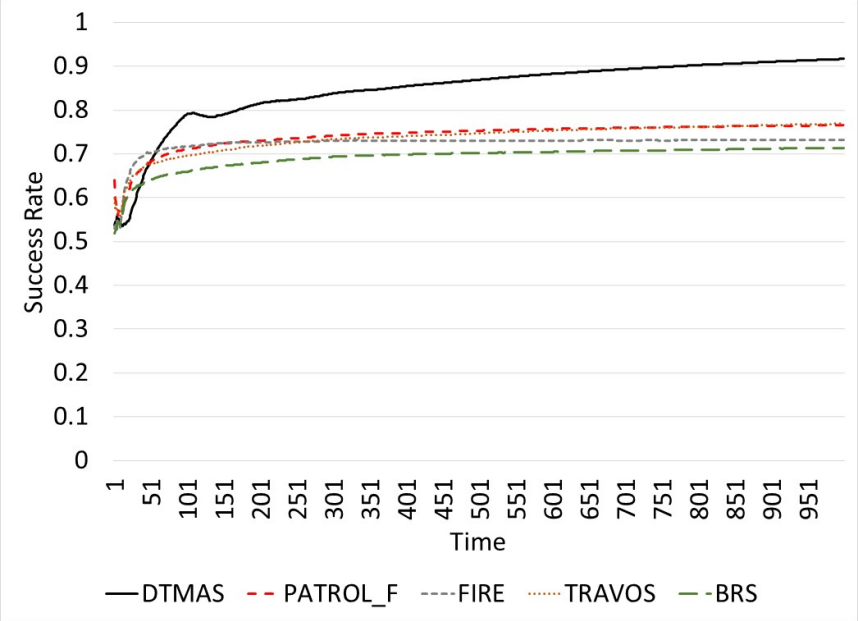


Figure 6.3: Success Rate of Trust Evaluation Models - Random Behaviour of Trustees Probabil ity 50%

number of cheating attempts is reduced, and a partial systematic behaviour can be seen. It worth noting that the bias toward using direct experience helped TRAVOS to achive comparable results to PARTORL-F in this case.

Figure 6.4 shows that strategic cheating is hard to detect as it happens once a while, this agrees with the findings of [105]. However, the combination of fuzzy logic with immediate suspension helps the proposed model resist such types of attacks, and let reinforcement learning direct the long-term learning curve. The use of confidence in calculated values helps TRAVOS perform relatively well compared to other models in the study under strategic attack of trustees.

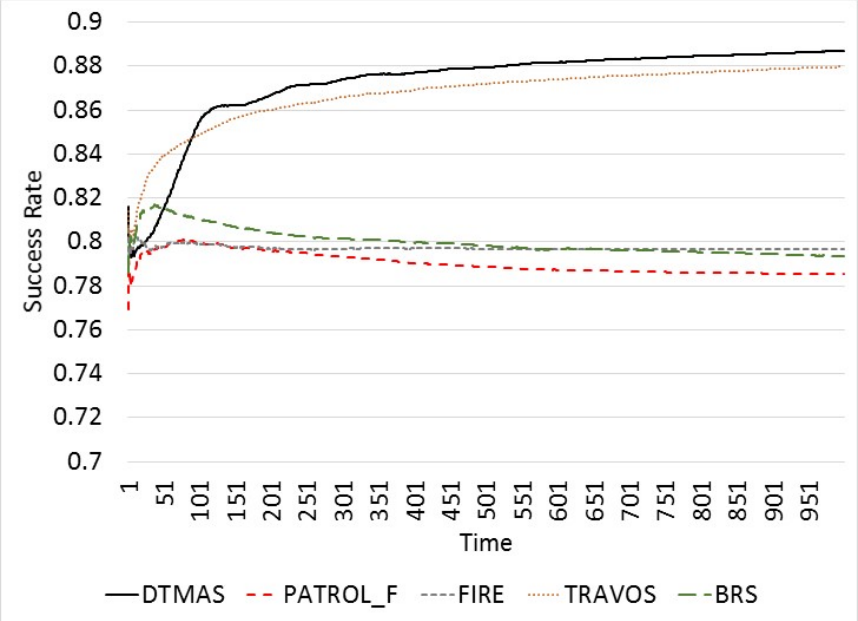


Figure 6.4: Success Rate of Trust Evaluation Models - Trustees Strategic Cheating

Figure 6.5 shows that performance of models when attackers target a subset of trustors only. The similarity selection of PATROL-F helps the model, slightly, outperforms DTMAS. DTMAS performs well because of its suspension of trustees, credibility checking of witnesses, and the tendency to avoid using information from witness whose testimonies led to unsatisfying interactions. Figure 6.6 shows the effect of always cheating trustees on the success rates of the models. Fuzzy logic based models outperforms probabilistic based ones, and the immediate suspension helps DTMAS to outperform PATROL-F.

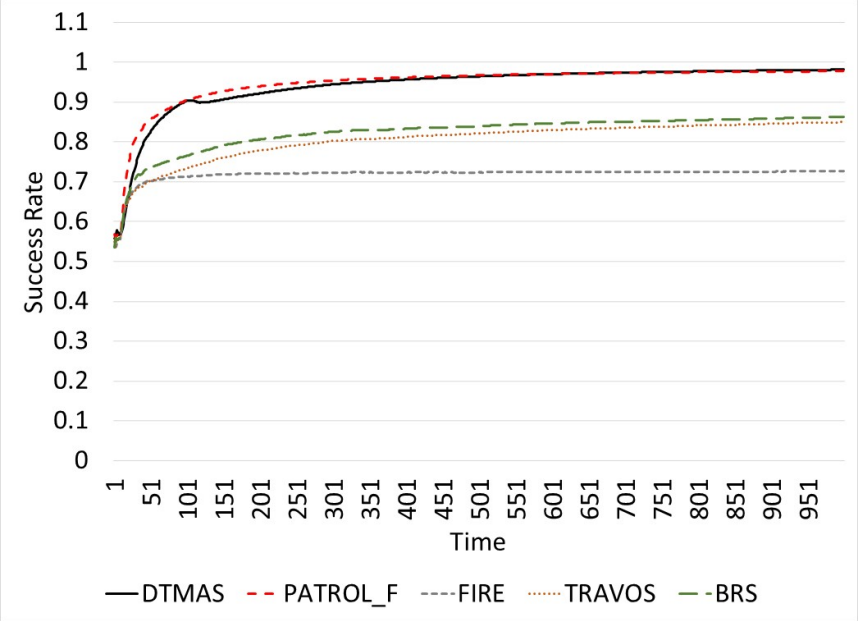


Figure 6.5: Success Rate of Trust Evaluation Models - Trustees Targeting Subset

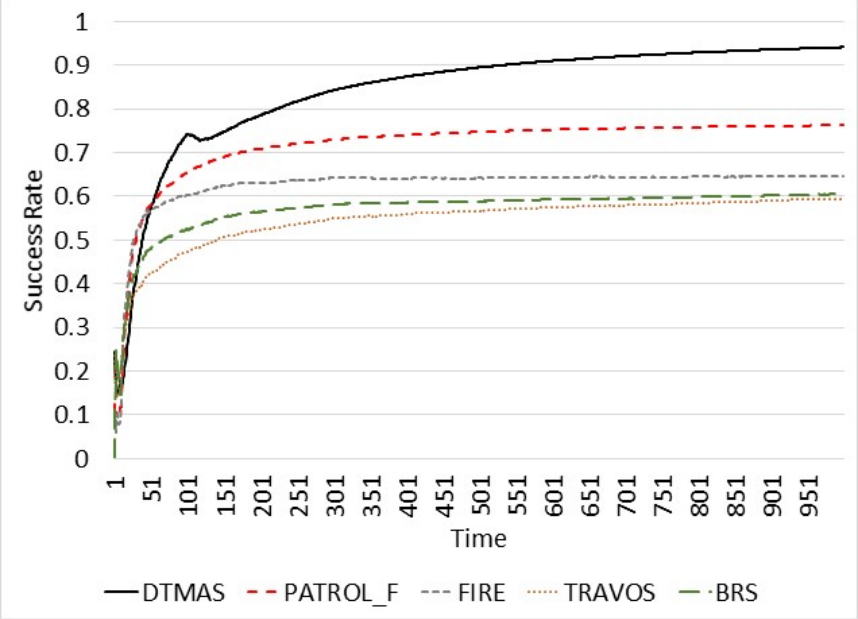


Figure 6.6: Success Rate of Trust Evaluation Models - Trustees Always Cheat

Misbehaving Witnesses Scenarios

Not only trustees may cheat, but also witness may misbehave in different ways. A good trust model should guide trustors to interact with honest trustees, even in the existence of misbehav-

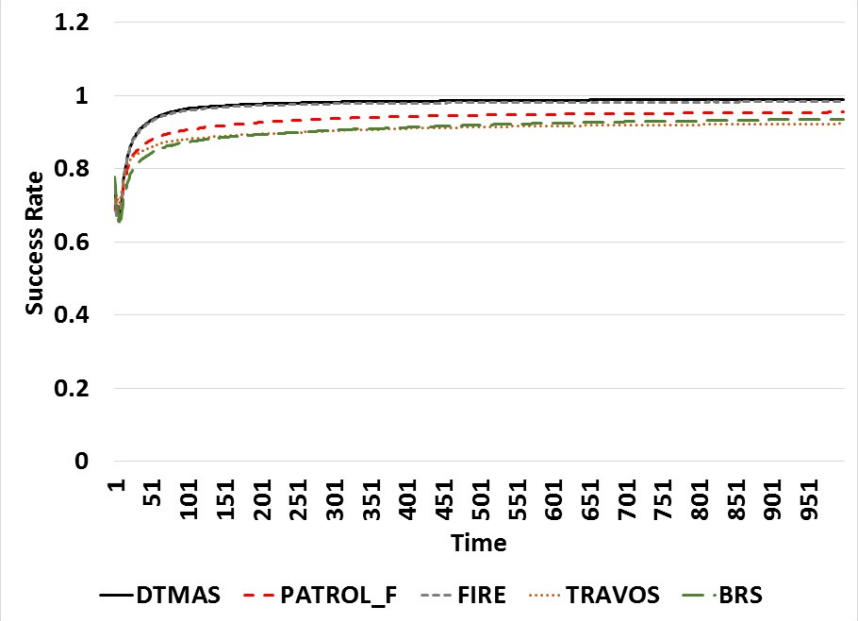


Figure 6.7: Success Rate of Trust Evaluation Models - Purely Random Behaviour of Witnesses

ing witnesses. Trust models based on direct experience only are not affected at all by potential misbehaviour of witnesses, as they do not use testimonies. However, such models can be at a disadvantage when credible witnesses exist. All the models used in the performance analysis study were designed to consider testimonies from wittiness, when such information exists, however models differ in their abilities to resist potential attacks from witnesses. In this set of experiments, only ten percent of trustees are honest, while others provide low performance without purposefully attempting to attack trustors, all witnesses are misbehaving. To study the effect of different witnesses' attacks, we assume all non-honest witness use the same attacking strategy. We study the witnesses' attacks described in 4.3

Trust evaluating agents, do not always use the service in every round. The activity level is in the range $[0 - 1]$. As we are studying the effects of witnesses' attacks, we assume all trustee are honest, with different levels of performance.

Generally, bad mouthing shown in figure 6.11 and targeting a subgroup of trustors shown in figure 6.10 leads to the worst success rate for most models, where as positive mouthing shown in figure 6.12 and probabilistic random shown in figure 6.8 are less sever. None of the studied

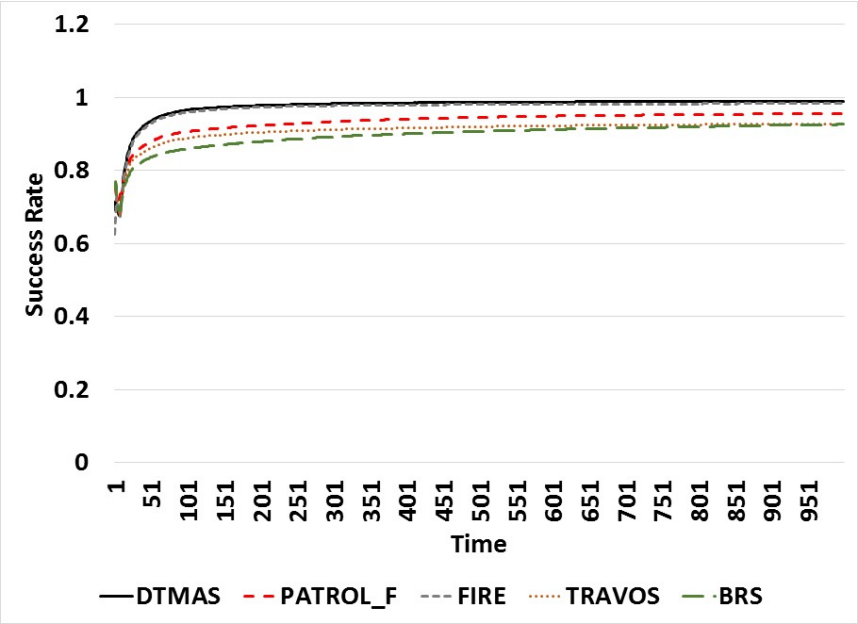


Figure 6.8: Success Rate of Trust Evaluation Models - Random Behaviour of Witnesses Proba bility 50%

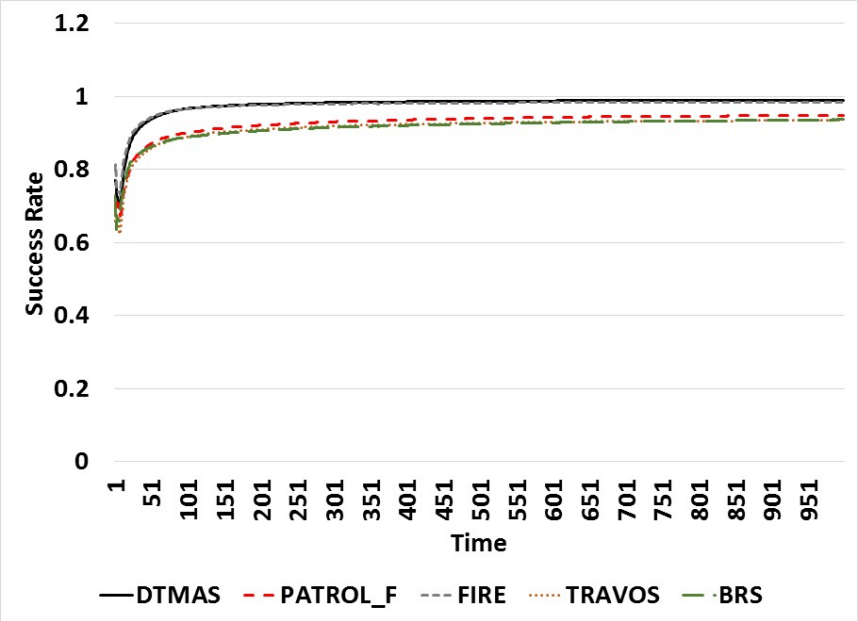


Figure 6.9: Success Rate of Trust Evaluation Models - Witnesses Strategic Cheating

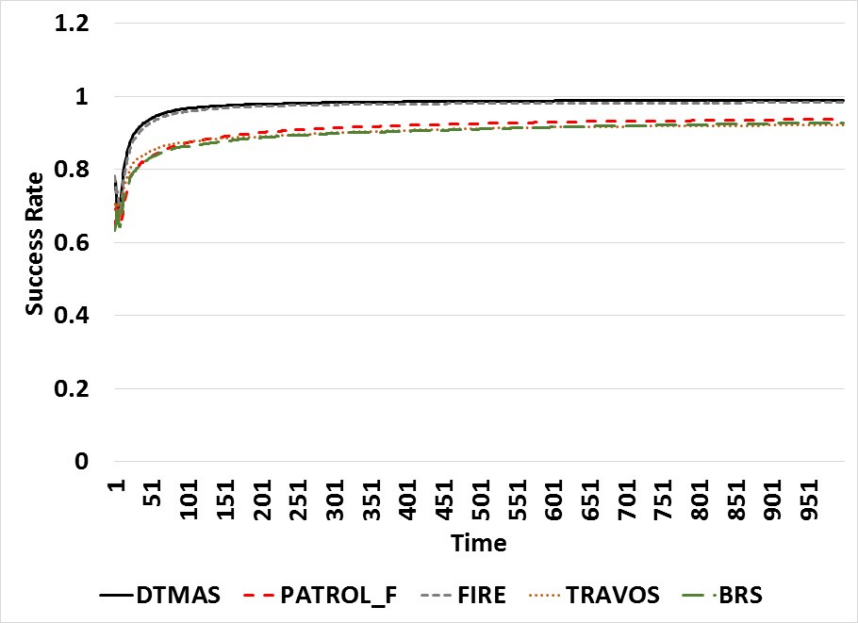


Figure 6.10: Success Rate of Trust Evaluation Models - Witnesses Targeting Subset

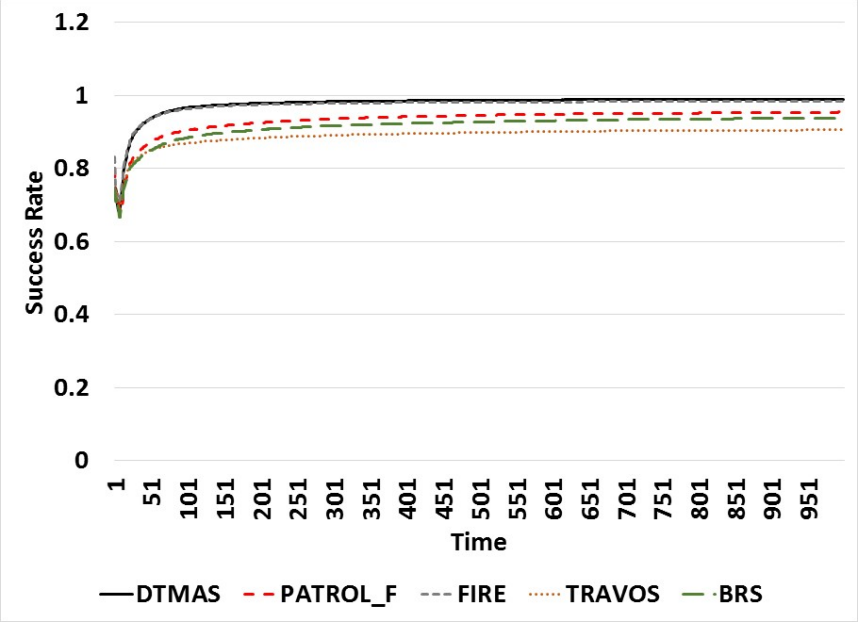


Figure 6.11: Success Rate of Trust Evaluation Models - Bad Mouthing

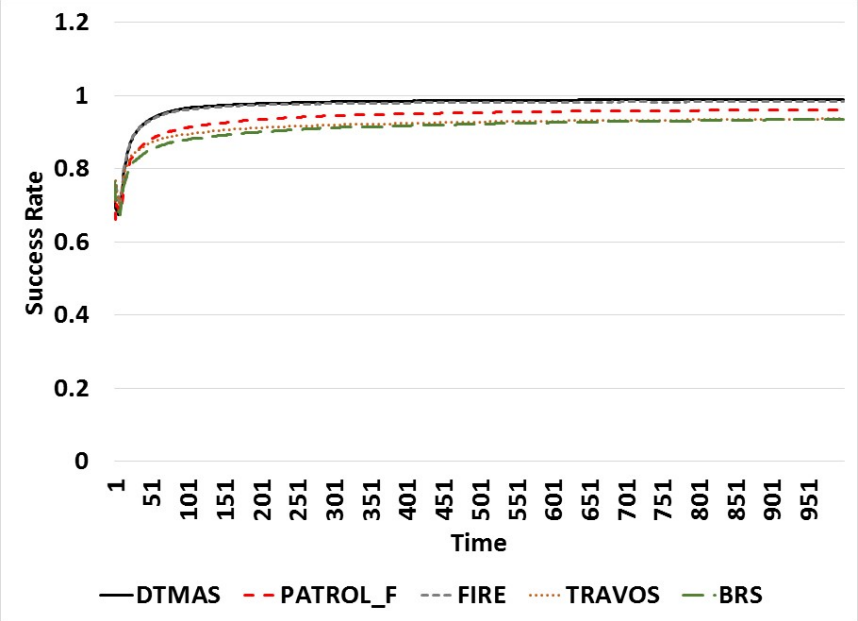


Figure 6.12: Success Rate of Trust Evaluation Models - Positive Mouthing

models depend on witnesses' testimonies alone; rather this information source is combined with direct experience, which alleviates the effect of witnesses' attacks.

The figures show that DTMAS seems to reasonably resist attacks of witnesses as it temporally suspends interacting with any misbehaving witness, and let reinforcement direct the long-term learning curve. Also, the use of fuzzy logic helps DTMAS resist random witnesses' attacks as shown in figures 6.7 and 6.8. The use of multiple information sources, as well as exploration strategy used in FIRE help resisting attacks of witnesses.

In general, models that evaluate the credibility of witnesses continuously to avoid cheating ones have good success rates, such as PATROL-F, BRS, and TRAVOS. However, figures 6.9, 6.10 and 6.12 suggests that probabilistic models, such TRAVOS and BRS, are more sensitive to strategic cheating, subset targeting, and bad-mouthing attacks compared to models that use fuzzy logic, such as DTMAS and PATROL-F. Figures 6.11 and 6.12 suggests that good-mouthing can be less harmful than bad-mouthing attack.

6.3 Compare and Contrast of Trust Establishment Models

Even though a centralized architecture for trust establishment models can go well with online crowd-sourcing, such as Amazon Mechanical Turk (AMT), such architecture contradicts with the characteristics of a dynamic environment where the population of agents can vary over time, as in MASs. Models with centralized architecture, such as Social Welfare Optimizing Reputation-aware Decision making (SWORD), can experience scalability and transparency issues [153]. Typically, single points of failure and performance bottlenecks are major concerns for a centralized model. On the other hand, the decentralized approach can provide good scalability, robustness and assures the accessibility of services [153], and can be more appropriate for modelling MASs.

Reputational Incentive (RI) [10] is a decentralized, generic model that can be instantiated and applied in a wide range of applications. The model can deal with the dynamic characteristics of MAS, such as changeability in trustors' behaviours. Even though the model allows environments with multiple contexts, context diversity checking and mapping is not available. The model assumes tasks with single criterion, or can be treated as such. Furthermore, the model does not use any defence mechanism against potential attacks on reputation; rather the model assumes that trustees can get an accurate evaluation of their reputation in the community.

The decentralized trust establishment models described in [129] assumes a single context environment and cooperative set of trustors that are willing to provide accurate explicit feedbacks. Even though the model has no explicit defense against misleading information, third-party information sources are not used, and buyers have no incentive to lie.

Trust Establishment Using Implicit Feedback (TEIF) is a decentralized trust establishment model that depends on implicit feedback from interactions' partners to address the situation where explicit feedback in not possible or not desirable. The model uses retention as a single criterion represented as a real number and implicitly assumes that trustors are neither cooperative nor liars. Context handling in not addresses in the model neither explicitly not implicitly.

Trustees dynamically update their prediction of interaction partners' behaviours.

Multi-Criteria Trust Establishment Model Using Explicit Feedback (MCTE) is a distributed, multi-criteria trust establishment model. The model is aimed at situations where trustors are willing to provide only aggregated explicit satisfaction feedback, without detailing the importance or weight of individual service criteria. MCTE uses the provided feedback from trustors regarding how satisfied they were with recent transactions to predict the importance of different service criteria for trustors and adjust the behaviour of trustee(s) accordingly.

As with MCTE, Integrated Trust Establishment Model for MASs (ITE) is a distributed, multi-criteria trust establishment model. ITE uses the provided feedback from trustors regarding how satisfied they were with recent transactions as well as their retention to predict the importance of different service criteria for trustors and adjust the behaviour of trustee(s) accordingly, rather than using retention only as in TEIF or explicit feedback only as MCTE. Context diversity checking can be achieved using multi criteria nature, and the use of weight and demand an be considered as the first step toward context mapping in ITE.

6.3.1 Experimental results

To compare the performance of the proposed trust evaluation model with other models, we use a simulation environment similar to the one described in 5.7. However, in this set of experiments we choose to present the percentage of good transactions and the percentage of bad transaction for readability. We compare the proposed trust establishment models with the most relevant work; namely RI model [10]. For the general simulation, each scenario is simulated, separately, with all trustees use the same trust establishment model. As we aim to compare the performance of trustees equipped with the proposed models and those equipped with the RI [10], all trustees are assumed, to be honest, and the only difference among them is the trust establishment model. This way we can relate the difference in performance to the model of trust establishment used. Table 5.2 presents the base values for the number of agents and other parameters used in the

proposed model and those employed in the environment. When testing the effect of a particular parameter, others are set to those base values.

Effects of Trustors' Demand Level

Figures 6.13 - 6.24 presents the performance of the models, under different levels of trustors' demands. The figures show that trustees empowered with ITE have a high average trust comparable to TEIF and MCTE and exceeding that of RI (figures 6.13 - 6.15) despite providing the lowest UG (figures 6.16 - 6.18) for intermediate and high demanding trustors, and a slightly higher than RI for low demanding trustors. This is because ITE tries to emphasis on improving its performance with respect to important criteria, and rely on explicit feedback to determine satisfaction of partners combined with the retention indicator. TEIF continuously provide more UG than all other models (figures 6.17 and 6.18), which is the reason for its trust level (figures 6.14 and 6.15). For easy going trustors, all models cause trustees to provide lower UG and stay trustworthy (figures 6.15), however ITE is the one with the lowest trust level, around 80% as ITE believes that such high level of satisfaction is good enough to start making profit (engagement threshold is 0.9 and Θ is 0.9 also). Detecting the importance of different criterion and providing various levels of UG accordingly helps trustees to achieve a relatively high level of trust at a reasonable provided UG.

It worth noting that keeping a relatively stable level of trust, despite the decline in the percentage of good transactions, and consequently, the increase in the percentage of bad transactions (figures 6.19 - 6.24) suggests that unfulfilled criterion either not important, or highly, rather than completely, fulfilled or both.

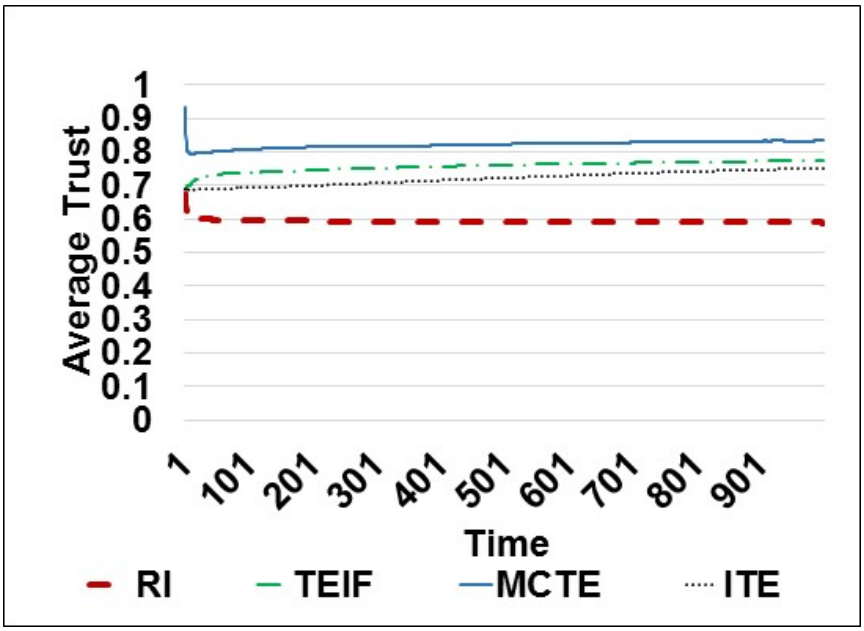


Figure 6.13: Average Trust: High Demanding Trustors

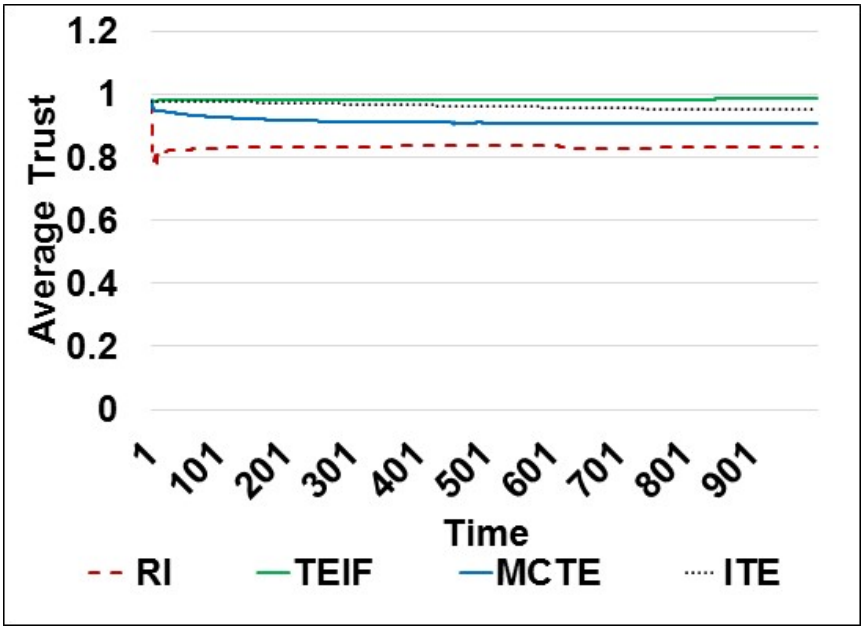


Figure 6.14: Average Trust: Regular Demanding Trustors

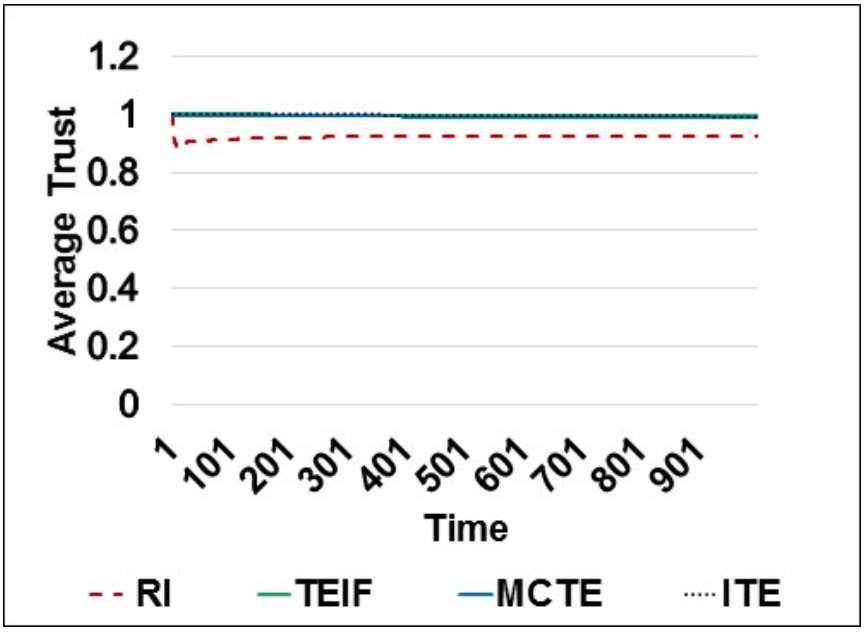


Figure 6.15: Average Trust: Low Demanding Trustors

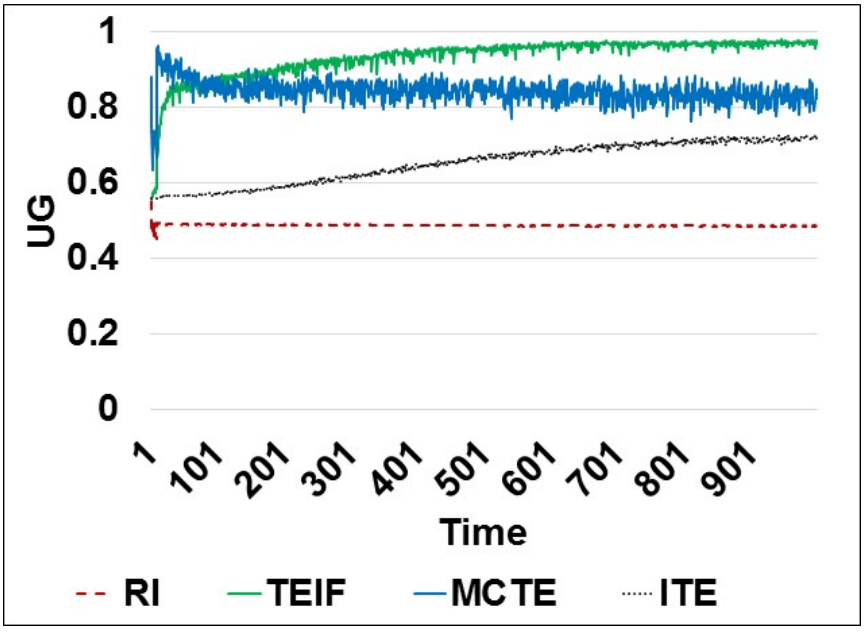


Figure 6.16: UG: High Demanding Trustors

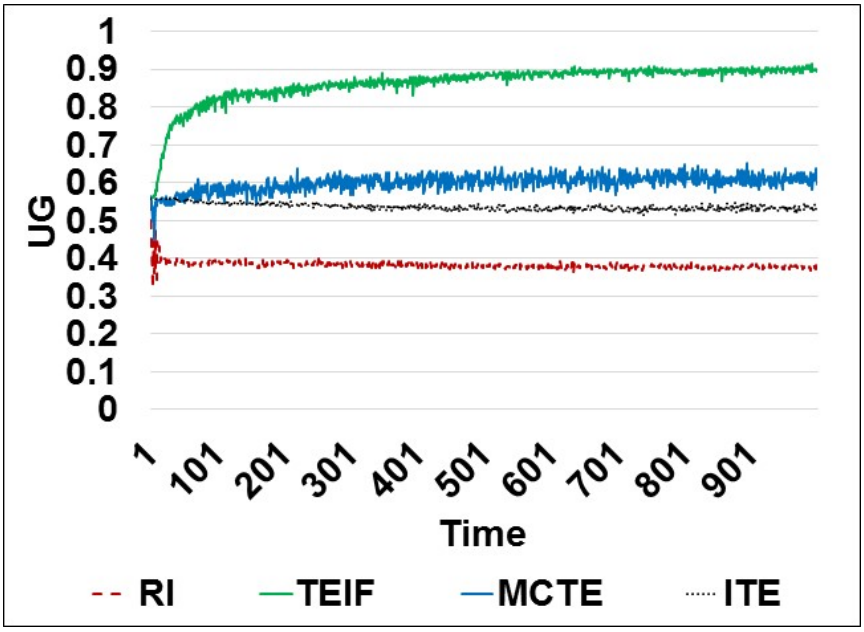


Figure 6.17: UG: Regular Demanding Trustors

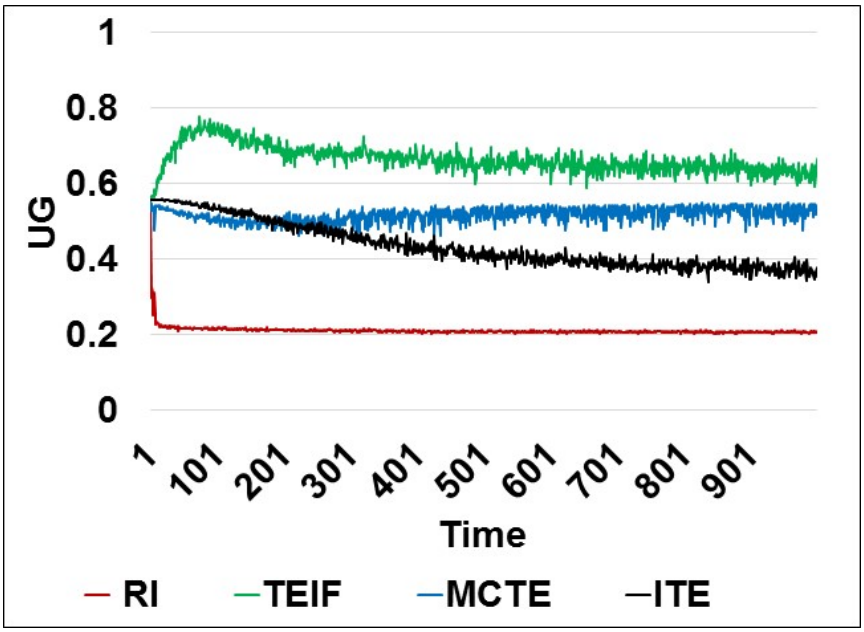


Figure 6.18: UG: Low Demanding Trustors

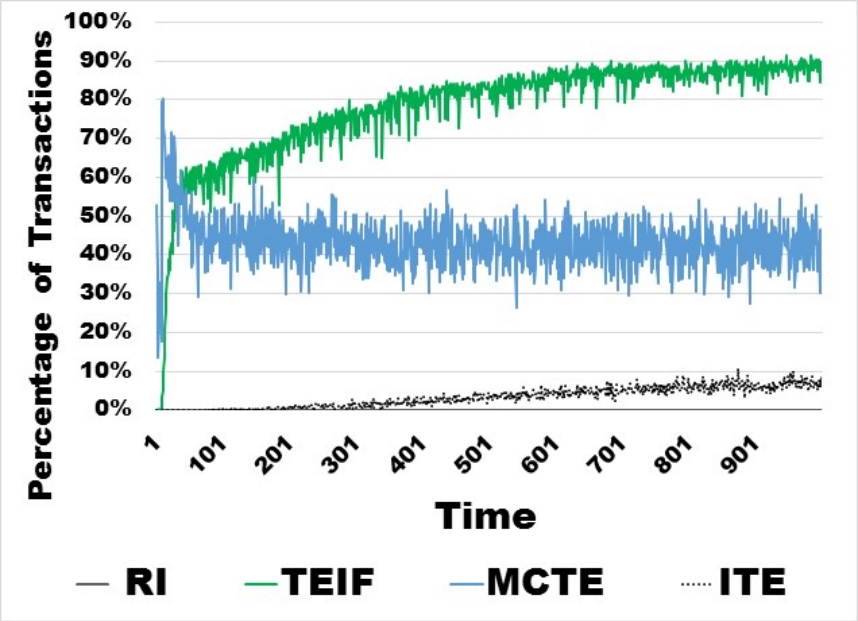


Figure 6.19: Number of Good Transaction: High Demanding Trustors

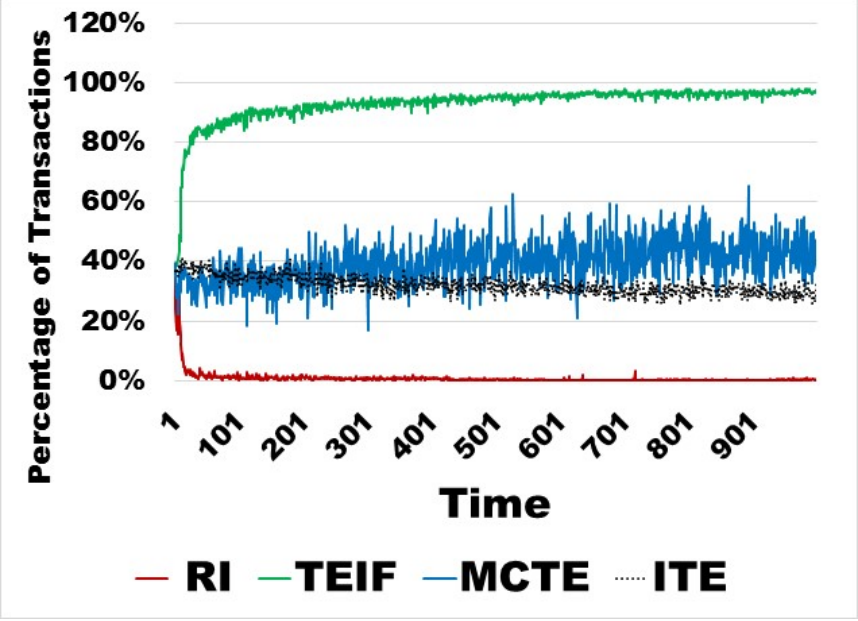


Figure 6.20: Number of Good Transaction: Regular Demanding Trustors

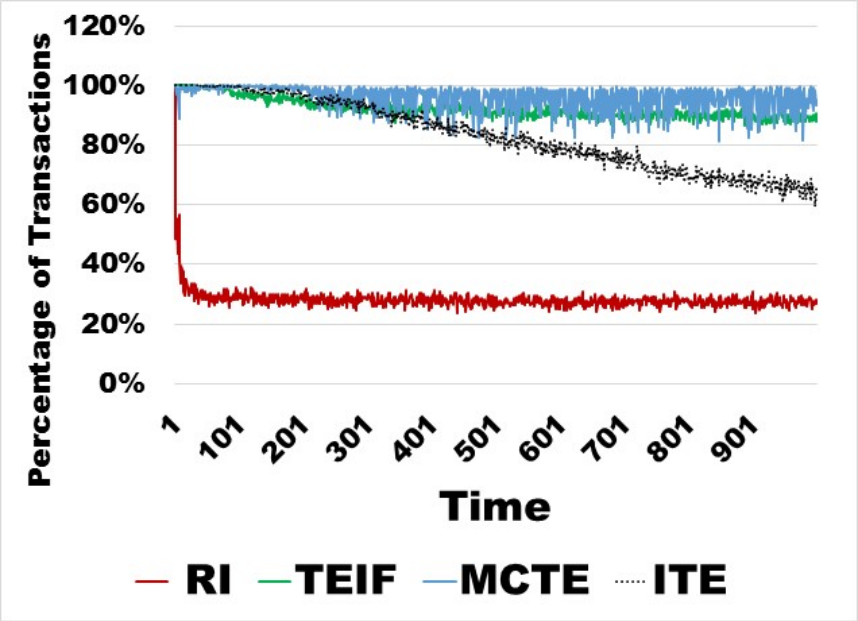


Figure 6.21: Number of Good Transaction: Low Demanding Trustors

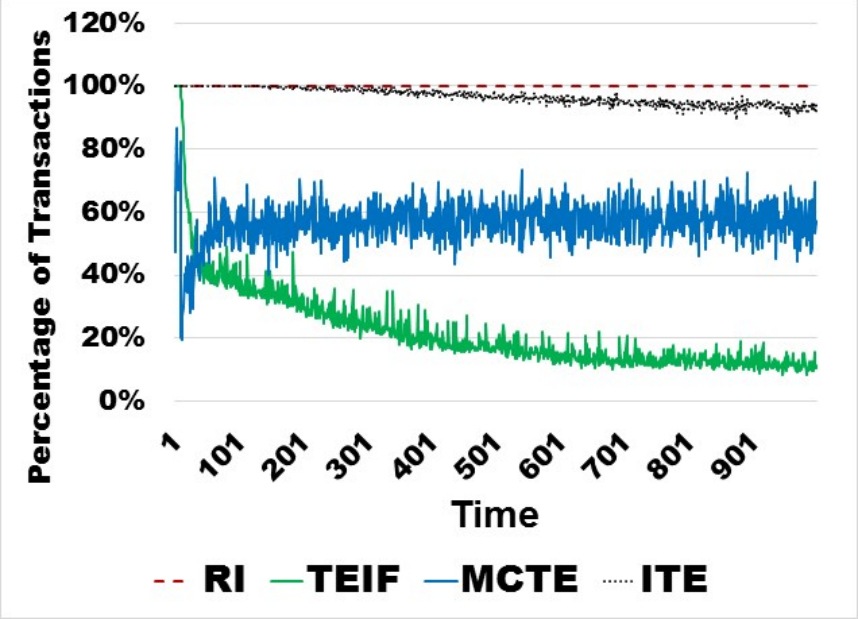


Figure 6.22: Number of Bad Transaction: High Demanding Trustors

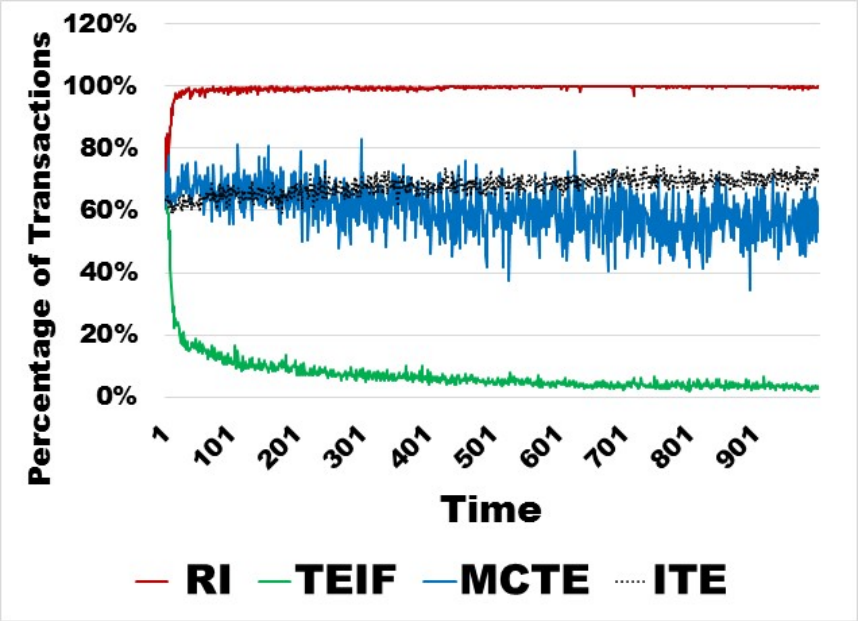


Figure 6.23: Number of Bad Transaction: Regular Demanding Trustors

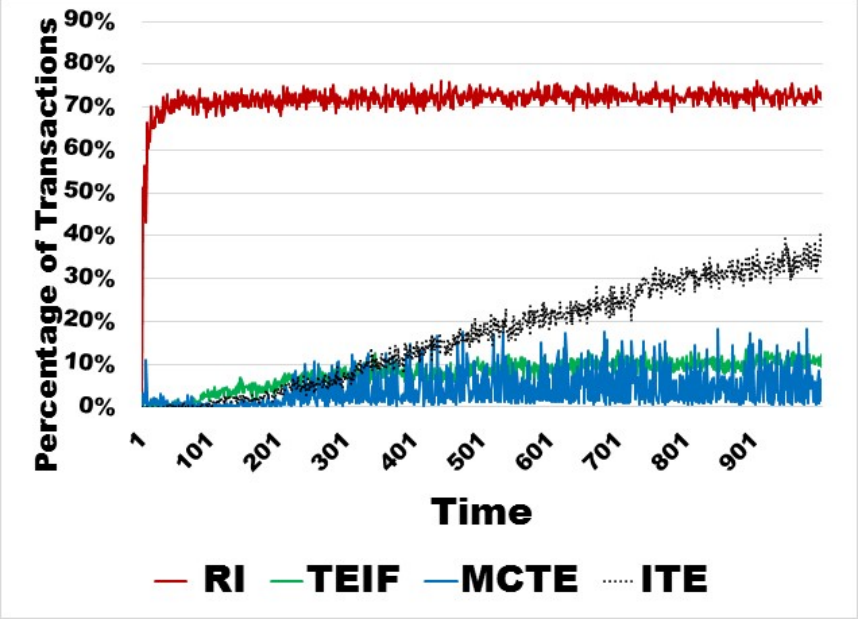


Figure 6.24: Number of Bad Transaction: Low Demanding Trustors

The Effects of Trustors' Activity Level

Figures 6.25 - 6.36 presents the performance of the models, under different levels of trustors' activity. Figures Figures 6.25 - 6.27 show no significant variation of trust for the models under different activity levels, especially MCTE, which doe not depend on retention. However, TEIF seems to provide the highest UG compared to other models as it depends on retention only and assumes a single criterion tasks. Generally, ITE can achieve a comparable level of trust against MCTE and TEIF under different levels of activity, despite having the highest level of bad transactions. It is interesting to note that ITE does not provide extra high UG as it depends on both explicit and implicit feedback and assumes multi-criteria tasks. Also, note that the UG provide TEIF can be affected by activity levels, as the models depends on retention alone.

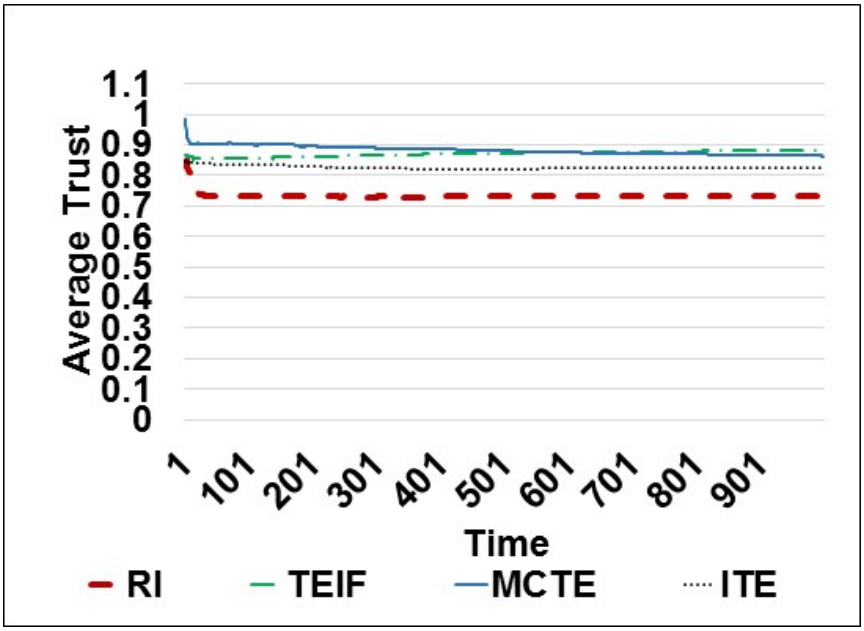


Figure 6.25: Average Trust: High Activity Trustors

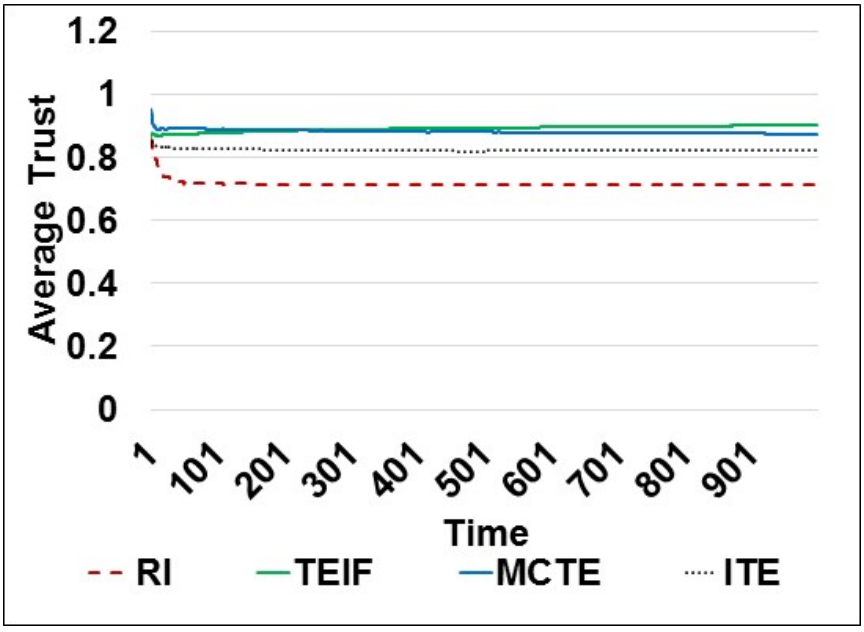


Figure 6.26: Average Trust: Regular Activity Trustors

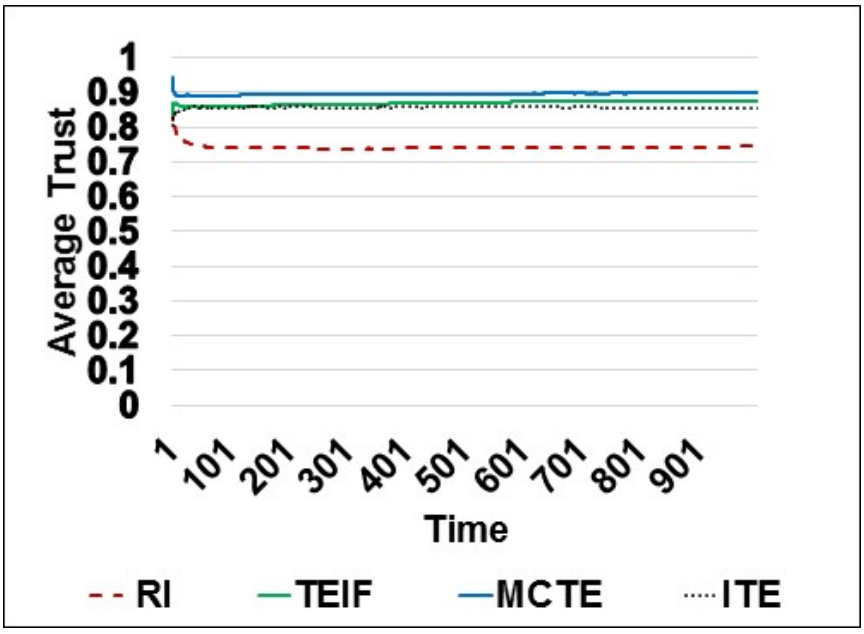


Figure 6.27: Average Trust: Low Activity Trustors

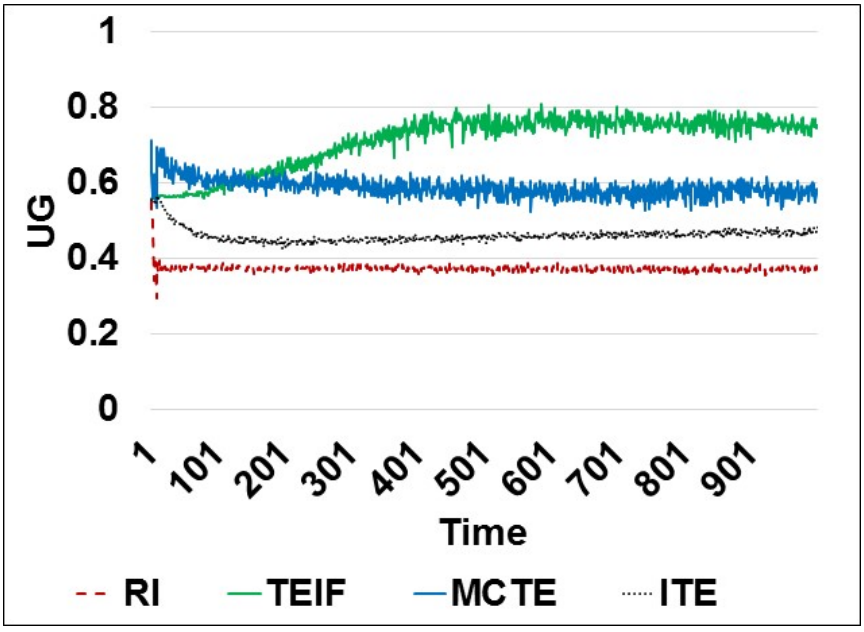


Figure 6.28: UG: High Activity Trustors

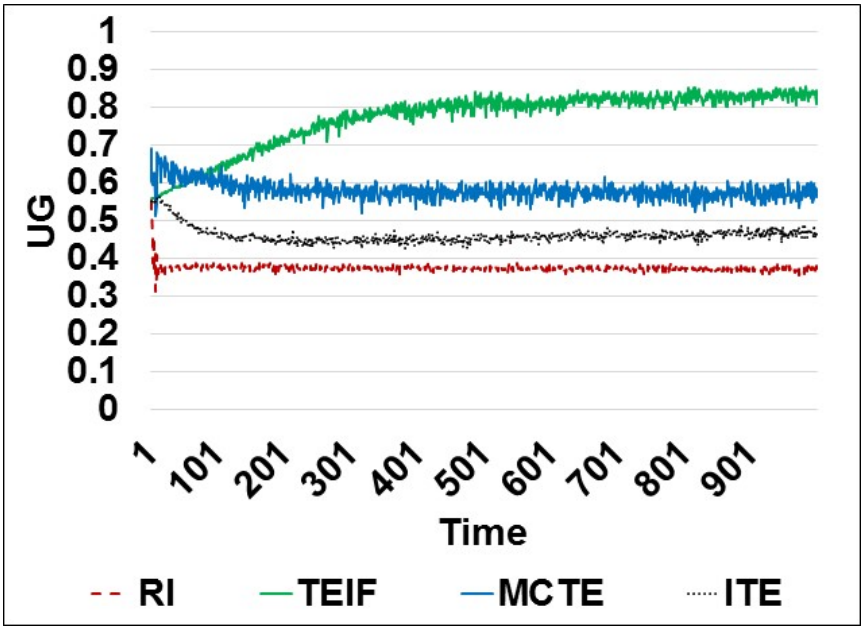


Figure 6.29: UG: Regular Activity Trustors

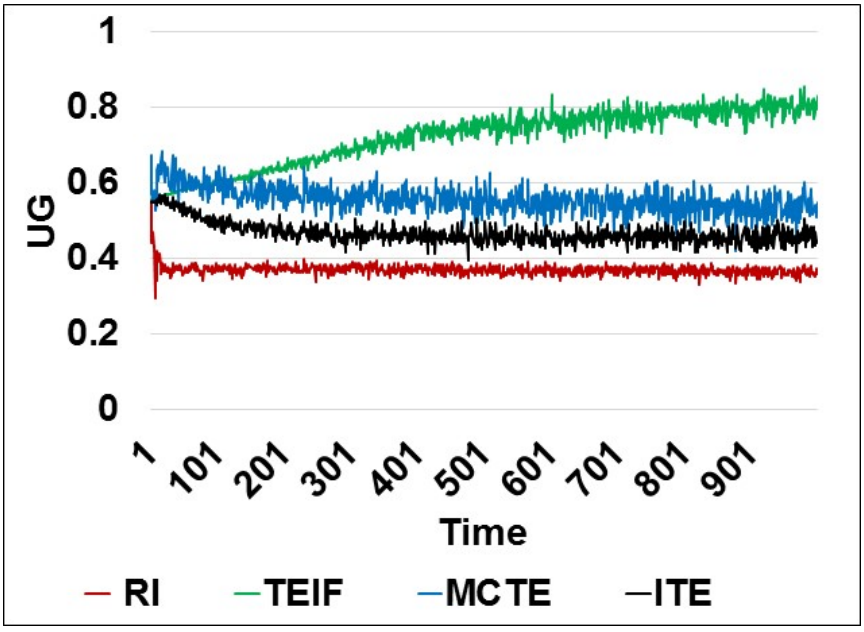


Figure 6.30: UG: Low Activity Trustors

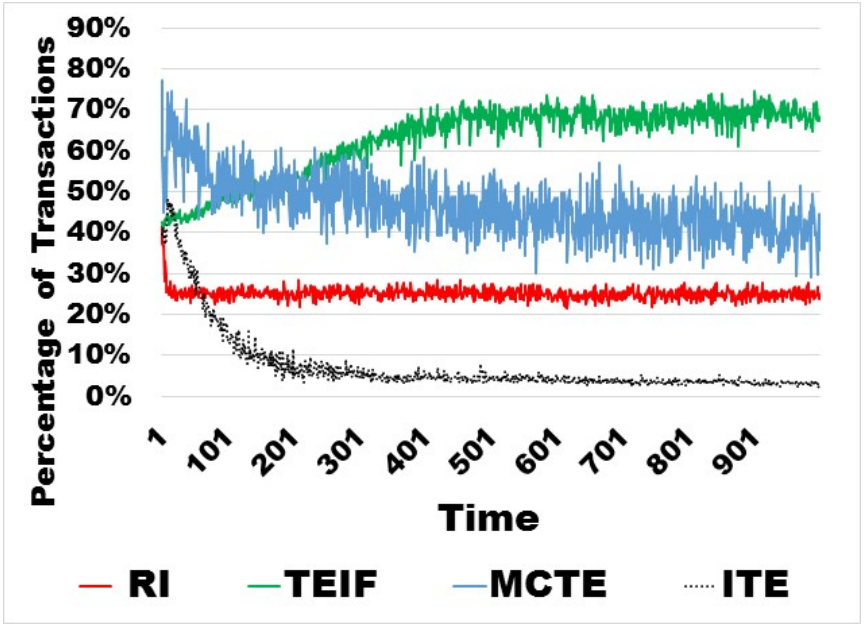


Figure 6.31: Number of Good Transaction: High Activity Trustors

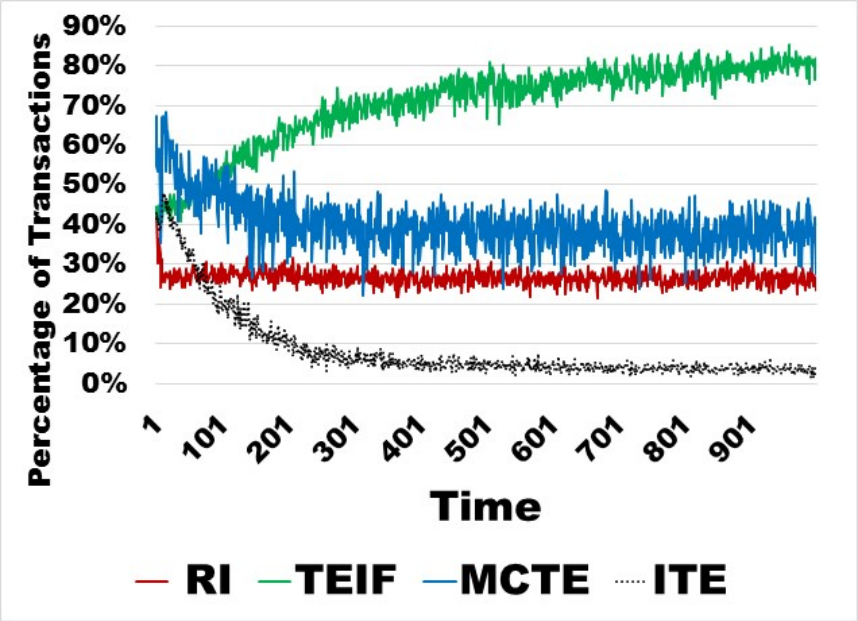


Figure 6.32: Number of Good Transaction: Regular Activity Trustors

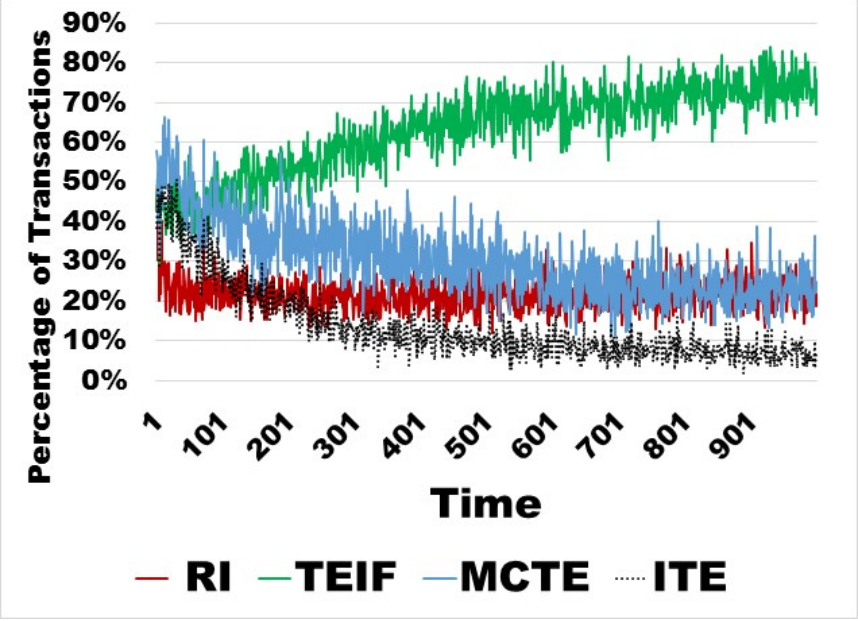


Figure 6.33: Number of Good Transaction: Low Activity Trustors

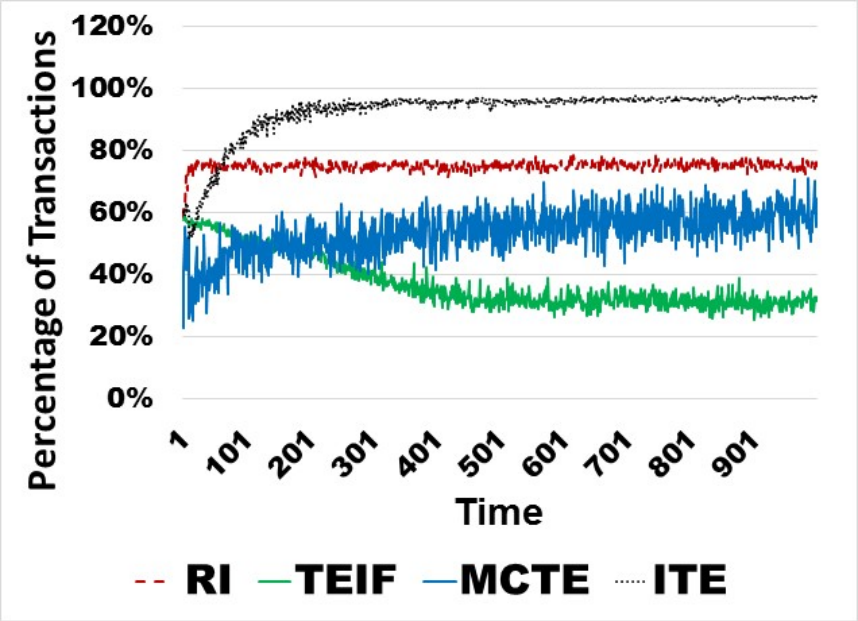


Figure 6.34: Number of Bad Transaction: High Activity Trustors

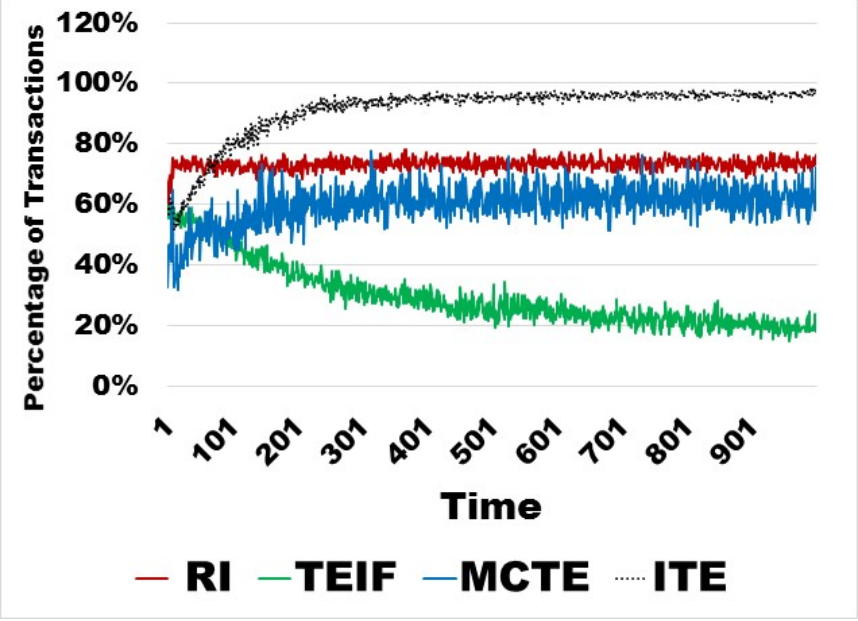


Figure 6.35: Number of Bad Transaction: Regular Activity Trustors

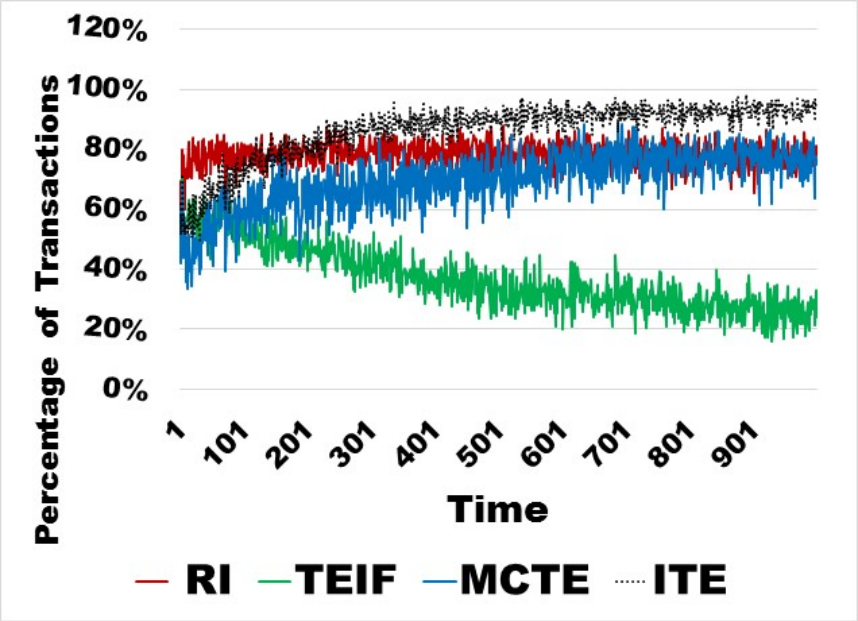


Figure 6.36: Number of Bad Transaction: Low Activity Trustors

6.4 Application Areas

6.4.1 Web Services

The amount of automated, distributed services available over the Web, such as web-based storage, social networks, and entertainment is continuously increasing [136]. Web services are modular, web-accessible software applications that can interact with each other over networks using a set of protocols [66]. They provide a flexible mean for integrating web-based software applications [112]. In fact, Web services can be used to address a wide range of requirements from simple requests for information to complicated business processes [137]. MASs deemed appropriate for modelling Web services [83, 59, 56, 29, 65, 136]. Wide adaptation of such services evolves the Web into collections of services automatically used by software agents instead of being a large set of static pages [22]. When a proper semantic markup, or machine-readable description, is provided to every resource, making it easily accessible by software agents, knowledge embedded within web pages may be accessed and exploited automatically by software agents using different reasoning mechanisms, and Web services become semantic ones [136].

We believe that fully automated environment of Web services can be modelled as a MAS where agents represent Web services and each web service can play the role of service consumer of services provided by others (i.e. the role of trustor) or the can provide service for others (i.e. the role of trustee) or both (i.e. a trustor for some tasks and a trustee for some other). Services providers expected to be rewarded for services they provide, which is not unlikely in the environment of web service. Agents representing Web services are self-interested whose goal is to maximize their own benefit.

The environment of Web services assumed to be interconnected where Web services, and their agents, assumed to have access to all other Web services (agents) at negligible cost. Generally, services providers are registered with a special purpose discovery service. Also, services request and delivery can be fulfilled by existing protocols of web service or a variation of them (which is out of scope of this work). However, in a large, open and dynamic environment where services can be published effortlessly, a service consumer needs to know not only what a service provider can do, but also how well a service provider can do that. Unfortunately, as service consumers cannot assure that service providers make accurate statements regarding their competencies and abilities, trust is considered essential for making interactions effective [140]. In fact, it is this aspect where the proposed trust evaluation model DTMAS can be used to help service consumers choose trustworthy service providers. To evaluate the trust of a service provide, the service consumers can use its prior direct experience with the provider, testimonies from witness and the average time decayed utility gain of previous transactions with the provider. Q-learning to be used to tune the learned trust based on direct experience, and testimonies from witnesses to be weighed by the credibility of each witness. Then the fuzzy logic system to be used for evaluating how trustworthy a provider is. When more then one provider can provide the same service, the one with the highest trust level will be selected. If the selected provider does not satisfy the needs for the consumer, the consumer will reduce its trust evaluation of the provider and suspend the use of that provider for a while.

Typically, a services provider tries to maximize its rewards by tuning the utility gain it pro-

vides and try to be selected for as many interactions as possible. As individual services can have multi-criteria, providers can use MCTE or ITE when consumers are willing to provide explicit feedback about how satisfying the transaction was. Consequently, providers can tune the service they provide to better meet the demands of trustors, reduce the cost of the provided service when possible, and become more trustworthy. When consumers are not willing to provide explicit feedback about how satisfying the transaction was, providers can use TEIF to tune the service they provide to better meet the demands of trustors and become more trustworthy. .

6.4.2 Internet of Agents

Interconnecting agents of different MASs representing various modern systems such as smart buildings, smart transportation systems, and Smart Grid (SG) , can be referred to as Internet of Agents (IoA) . Such systems can be interconnected through web-like technologies (web service like) or using Internet-like architecture. IoA is viewed as an extension of the concept of Internet of Things (IoT) by augmenting internal reasoning and intelligence capability to, traditionally, naïve things [86]. Agents of IoA not only can interact directly with other agents in the same or different MASs, but also, they can interact with other systems directly or through a mediator [80]. In the literature, IoA also referred to as Internet of Smart Things [132], Web of Things [18], Web for Agents [134], Agents of Things [80] as well as IoA [154].

One potential application of IoA is the Smart Grid (SG). Traditionally, large, centralized electrical generators produce electricity, which is transferred in bulk to substations that in turn distribute electricity to residential, commercial, and industrial customers [135]. The electrical system that generates, transmits, distributes, and controls electricity is usually referred to as the grid [25]. There is a global interest recently for using renewable energy, not only to replace fossil fuels for their negative environmental impacts but also to meet future demands for electricity [44]. Even though the price reduction of these renewable sources further encourages adoption of such sources, large-scale adoption of renewable sources can be challenging and require complex

control of diverse sources and storage [44], as renewable sources can be both discontinuous and distributed [99].

The term Smart Grid refers to an electric system that uses information and communication technologies integrated with a highly intelligent and distributed control across electricity generation, transmission, distribution and consumption [25]. SG represents a paradigm shift from the traditional centralized electricity generation and one-directional power flow into more reliable and sustainable energy systems [44]. They can be considered as the next generation electricity network [25] and expected to be consumer friendly, efficient, reliable, flexible enough to enables markets, and accommodate various generation and storage options [89]. Other terms for SG used in the literature, such as Future-Grid, smart power grid, and intelligent grid [34]

New energy technologies and electrical-powered devices come with embedded electronic intelligence that controls their operations and can interconnect them to the grid [61], which can grow into distributed ecosystems with a large number of intelligent, heterogeneous components interacted at the electricity and information levels to provide multidimensional services [135]. For example, distributed energy sources can be evaluated not only based on the price per unit they provide but also their stability, as well as, their environmental impacts. The importance of each criterion can vary from one consumer to another.

In addition to the applications of MASs in micro grids [55, 46, 21], MASs can be used in different categories of power engineering [76] in the context of SGs, such as modeling and simulation [37], planning [30], network management [20], SG operation [3] , electricity market [118], automation of substations [146], restoration [58], reconfiguration [64], load shedding [148], monitoring and diagnostic [72], and protection [126]. Interconnecting agents of different MASs of a SG, as well as, agents of other modern systems through IoA, enhances the services that could be derived from the interactions of agents.

We believe that in a fully automated IoA environment, agents may choose to rely on other agents to fulfill some tasks. Such environments assumed to be interconnected where agents as-

sumed to have access to all other agents at negligible cost. For example, an agent representing an appliance in a smart building MAS, may want to get energy for its operation, the agent representing the appliance may request energy from the agent representing an energy source in a power production MAS, such as a solar panel in the neighbourhood. Agents who perform tasks expected to be rewarded for their efforts. While rewarding mechanisms can be straight forward in some situations, such as selecting energy source, it may not be the case for other situations, such as monitoring the smart grid. Rewarding mechanisms, architecture, and discovery of other agents are open issues for the IoA research community (which is out of scope of this work).

Interestingly, the utility gain offered by different gents (trustees) may not be the same. In the previous example, energy sources may provide various levels of energy, stability, and availability. Also, an agent assigned a task may choose not to fulfill its commitments. For example, not providing the required energy with proper stability. Unfortunately, trustors can examine the utility gain (quality, delivery time, etc.) of the task only after delivery. In fact, it is this aspect where the proposed trust evaluation model DTMAS can be used to help requesting agents (trustors) choose trustworthy interaction partners (trustees). To evaluate the trust of a trustee, the trustor consumers can use its prior direct experience with the provider, testimonies from witness and the average time decayed utility gain of previous transactions with the provider. Q-learning to be used to tune the learned trust based on direct experience and testimonies from witnesses to be weighed by the credibility of each witness. Then the fuzzy logic system to be used for evaluating how trustworthy a provider is. When more then one provider can provide the same service, the one with the highest trust level will be selected. If the selected provider does not satisfy the needs for the consumer, the consumer will reduce its trust evaluation of the provider and suspend the use of that provider for a while.

With proper reward mechanisms, a rational trustee tries to maximize its rewards by tuning the utility gain it provides and try to be selected for as many interactions as possible. As individual services can have multi-criteria, providers can use MCTE or ITE when consumers are willing to provide explicit feedback about how satisfying the transaction was. Consequently, providers can

tune the utility gain they provide to better meet the demands of trustors, reduce the utility gain of the assigned task when possible, and become more trustworthy. When trustors are not willing to provide explicit feedback about how satisfying the transaction was, trustees can use TEIF to tune the service they provide to better meet the demands of trustors and become more trustworthy.

6.5 Summary

This chapter provides a discussion of the value of the proposed models in this thesis. After a brief introduction, we presented contrasts and compares of our trust evaluation model with other related, or well-known models. We first reviewed several models in contrast with our model, namely TRAVOS [123], FIRE [39], the reinforcement learning based model presented in [131] and elaborated in [130], and the fuzzy logic based model referred to as PATROL-F [122], among some others. We argue that the proposed trust evaluation model has faster response to dynamic changes. Then we compared our trust evaluation model with the most relevant work of [122] and the well-known TRAVOS [123] and FIRE [39] under different attack strategies. The simulation results show that our proposed model has better resistance for trustee attacks.

The third section of the chapter presented contrasts and compares of our trust establishment models with the most related work. We first reviewed the most related models in contrast with our models, namely SWORD [155], Distributed Request Acceptance approach for Fair utilization of Trustees (DRAFT) [152] and RI [10]. Then we compared our trust establishment models with the most relevant work of [10] under different environmental conditions. The simulation results show that our proposed models can perform better in terms of achieving higher, however thy may have to put extra efforts in terms of more providing more utility gain to trustors.

The fourth section presented potential application areas were trust management models can be used.

Chapter 7

Conclusions and Directions for Future Work

7.1 Introduction

In this thesis, a range of issues have been explored. However, there are many potential directions for future research. Here, we present our conclusions and identify several noteworthy directions to expand our current research, both in extending the trust evaluation and the trust establishment mechanisms we are building and in developing more extensive validation to demonstrate the value of our models.

7.2 Conclusions

The main conclusion of this work is that for situations where recurring transactions are possible, combining Q-learning with fuzzing logic and suspension can result in a trust evaluation model with better accuracy, especially when trustees and witnesses can change their behaviour. Such performance can be attributed to fuzzy logic ability to address uncertainty, the ability of Q-learning to learn on the long term and the short-term response of suspension.

Interestingly, trustees can enhance their trustworthiness, at a cost, if they tune their behaviour in response to feedback (explicit or implicit) from trustors. Using explicit feedback with multi-criteria tasks, trustees can emphasize on important criteria to satisfy needs of trustors. Trust establishment based on explicit feedback for multi-criteria tasks, can result in a more effective and efficient trust establishment compared to using implicit feedback alone. Both approaches can be integrated together to achieve a reasonable trust level at a lower cost which was done by the model called Integrated Trust Establishment Model for Multi-agent Systems (ITE).

7.2.1 Contributions

In this thesis, we considered two major components of trust management in Multi-Agent Systems (MASs): trust evaluation and trust establishment. We summarize our main contributions as follows:

- Trust Evaluation
  - Temporary suspension
    - A suspension technique is used and combined with fuzzy logic and reinforcement-learning trust evaluation to improve the model responses to dynamic changes in the system.
    - Suspension is used to address the short-term relationship between a trustor and a misbehaving trustee or witness. It allows trustors to re-evaluate the trust of trustees without any further unpleasing transactions. The idea is that accidentally misbehaving trustees keep their trust in the community, unlike those who misbehave due to long term behaviour change.
    - The duration of suspension may depend on the harm resulted from the transaction.

* Suspension also used to respond quickly to misbehaving witnesses. Even though the model does not use testimonies about credibility of witnesses, not using testimonies from accidentally misbehaving witnesses can help trustors better evaluates their credibility as as the impact of old transactions fade out.
* Our work regarding suspension is presented in paper 4 in subsection 1.4.1 and paper 6 in subsection 1.4.2.

.

– Computation Engine

* A computational engine for trust evaluation base on fuzzy logic and Q-learning is presented.
* We use two fuzzy subsystems, one for the credibility evaluation of witnesses and the main one for trust evaluation of trustees.
* We define three input parameters for the main fuzzy subsystem. The first one is the direct trust evaluation where trustors use Q-learning to evaluate the trust of trustees in a way similar to the process in [130, 128]. The proposed model takes into consideration testimonials of witnesses, and the credibility of witnesses. To evaluate indirect trust, a trustor x consults other witnesses who interacted previously with the trustee y. To reduce the effect of fraudulent witnesses, a trustor tracks the credibility of witnesses and weighs each testimony by the credibility of the witness. To indicate that not only the number of transactions is important, but also the benefits from those transactions also taken into counts, we use the average decayed utility gain of previous transaction. Time decaying is used to emphasize that utility gain from recent transaction weigh more compared to those from old transactions if they have the same absolute value.

* We define two input parameters for the credibility evaluation of witnesses fuzzy subsystem. The first one is the credibility of witnesses calculated using Q-learning in a way similar to evaluating the trust of trustees using Q-learning. The other parameter is the utility gain associated with the most recent transaction where the testimony of a witness is used.

* Three fuzzy sets have been defined for each input to achieve a reasonable granularity in the input space while keeping the number of states small to reduce the set of control rules. In particular, each input can Low (L), Medium (M), or High (H).

* The performance of the proposed model is evaluated using simulation for a basic scenario where all agent are honest, and for a set of scenarios with sample attacks. Simulation results indicate that the proposed model can help agents select trustworthy trustees to interact with. It has a better performance compared to some of the popular trust models in the presence of misbehaving trustees.

* Our work regarding the use of fuzzy logic and Q-learning combined with suspension is presented in paper 2 in subsection 1.4.1 and paper 5 in subsection 1.4.2.

- Trust Establishments

  - Trust establishments using implicit feedback: We propose a trust establishment model based on implicit feedback from trustors where trustees use the retention of trustors to model trustors' behaviour when interactions are based on trust and they occur frequently in the system. Such a model can be used when trustors are not willing to provide direct feedback information to trustees, for any reasons, such as unjustified cost or being self-centred. The performance of the proposed model is evaluated using simulation for a variation of trustors demand levels and activity

levels. Compared with the most relevant work in th literature, simulation results indicate that the proposed can help trustees to become more trustworthy at the price of providing considerable higher utility gain. Our work regarding the use of implicit feedback to establish trust is presented in papers 1,2, and 3 in subsection 1.4.2.

– Trust establishment using explicit feedback: We propose a multi-criteria trust establishment model inspired by the research on customer satisfaction in the field of marketing. Trustees attempt to improve their performance based on the importance of the satisfaction criterion importance and level of demand. Trustees enhance their performance for highly important and highly demanded features. On the other hand, they can reduce their performance for unimportant and not highly demanded feature(s) while the corresponding trustor(s) may still be satisfied. The performance of the proposed model is evaluated using simulation for a variation of trustors' demand levels and activity levels. Compared with the most relevant work in th literature, simulation results indicate that the proposed can help trustees to become more trustworthy at the price of providing a slightly higher utility gain for demanding trustors, which is not the case for non-demanding trustors. Our work regarding trust establishment using explicit feedback is presented in paper 3 in subsection 1.4.1 and paper 4 in subsection 1.4.2.

– Integrated trust establishment: We propose a multi-criteria trust establishment model using both implicit and explicit feedback inspired from trustors. Trustees attempt to improve their performance based on the importance of the satisfaction criterion, importance and demand. Trustees enhance their performance for highly important and highly demanded features. On the other hand, they can reduce their performance for unimportant and not highly demanded feature(s) while the corresponding trustor(s) may still be satisfied. The performance of the proposed model is evaluated using simulation for a variation of trustors' demand levels and activity levels. Compared

with the previous two models, simulation results indicate that the proposed can help trustees to become more trustworthy at the price of providing a slightly higher utility gain than that provided by the model that use explicate feedback only but still lower than that provided by the model that use implicit feedback only. Our work regarding integrated trust establishment is presented in paper 1 in subsection 1.4.1.

The proposed trust evaluation model can be useful especially when trustees and witnesses can change their behaviour. The idea of immediate and temporary suspension seems promising and modular. We believe that it can be integrated with other exiting and future trust evaluation models to enhance the resistance against attacks similar to the cyclic attack. As we successfully combined fuzzy logic and Q-learning for trust evaluation, we believe that properly combining different computational techniques can be used to enhance the accuracy of trust evaluation models which can open the door for a new technique to design computational engines of trust evaluation models in MASs.

Our work on trust establishment and the results obtained seem interesting, as we believe it can be considered a milestone in the research direction of trust establishment, as the literature widely emphasized on trust evaluation rather than trust establishment. Even though we based our work on the limited existing literature of trust establishment in MASs, to the best of our knowledge, we are the first to address some challenges of trust establishment, such as the absence of explicit feedback, and the multi-criteria nature of tasks. The proposed trust establishment models can be useful especially when trustors base their decisions for interaction on trust, and the community of multi-agent systems looks homogeneous in terms of demands and levels of activity. In this thesis, we proposed a model that can help trustees become more trust worthy when trustors do not provide explicit feedback. Also, the thesis addressed cases when each trustor is willing to provided an aggregated level of satisfaction for multi-criteria tasks rather than providing per-criterion feedback. This thesis can open the door for many research efforts in the field of trust establishment including the investigation of different computational engines such as fuzzy logic and game theory, and finding other information sources, especially in the absence of explicate

feedback.

Even though our proposal on integrating both sources of information to achieve a better trust establishment resulted in a reasonable compromise between the two other models, we believe it can open the door for a new technique to design computational engines of trust establishment in MASs.

7.2.2 Limitations

Although we have a number of methods for evaluation and trust establishing, there are still some open issues for which we do not provide a solution.

In particular, we have identified the following limitations.

- Decision independence assumption: In our proposed models, we assume that testimonies reported by witness are independent of each other; that is no coordination, collusion or intersection of interest. Also, we assume that trustors report explicit feedback independently. Similarly, we assumed that the decision of a trustor to return back and interact with a trustee is done independently without coordination, collusion or intersection of interest with others. However, when this assumption does not hold, other techniques should be used to avoid misleading trustors, in the case of trust evaluation, and trustees in that case of trust establishment.

- Homogeneous levels of demand and activity: The two proposed trust establishment models that use implicit feedback assumed a homogeneous level of activity among trustors, and attributed the deviation from the average retention to reduction in trust evaluation rather than level of activity. Also, the proposed models of trust establishment that use explicit feedback summed homogeneous levels of demand for trustors and used the average demand to reduce the effect of extremers. However, when this assumption does not hold, other techniques should be used to avoid misleading trustees.

- Tasks of single criterion: Our trust evaluation model assumes that tasks are either single criterion or can be treated as single criterion. However, when this assumption does not hold, the trust evaluation should take into account levels of demand and importance for each.

- Self-contained tasks: Proposed trust evaluation and establishment models assumes that tasks for which a trustor evaluates a trustee, or a trustee may try to become more trustworthy, are self-contained, or can be treated as one, rather than being composed of multiple tasks performed be different agents. However, when this assumption does not hold, different approaches needed to take into account the process of composition and agents that handle each sub-task.

7.3 Directions for Future Work

Although our approach makes a number of advances to the state of the art, there are still a number of ways in which this work can be further extended. Here we present a short list of topics we think would improve our approach.

7.3.1 Extending the Models

Computation Engines

A critical component of any trust evaluation or establishment model is its computation engine that draws the necessary conclusions based on inputs from the surrounding environment. The literature of trust evaluation is relatively rich with various computation engines. In section 3.2 we presented an overview of a wide range of computation engines used for trust evaluation.

As a future work, we would like to consider developing a generic framework for hierarchical fuzzy Q-Learning based trust modeling where the selection of criteria for trust evaluation is

dynamic and extendable, that is a trustor may choose what criteria to include from a pre-existing set, and also can add new criteria for evaluation on the fly. This change can be application or transaction dependent.

The trust evaluation model described in 4.2 use hard-coded boundaries and equal weights for rules. As a future work, we would like to explore the opportunities of using Q-Learning to dynamically adjust the boundaries for membership functions of the fuzzy logic system, or in other words, attempting to learn the most accurate rules. Furthermore, we would like to explore the opportunity of associating different weights with different rules and learning such weight dynamically.

Proposed models for trust establishment (chapter 5) use Q-Learning to update the performance of trustees in response to feedback from trustors. Examining other computational engines is considered a future work. Mainly we are interested in Fuzzy logic and the combination of fuzzy logic and Q-learning.

Due to imprecise nature of parameters like ”criteria impotence” and ”level of satisfaction,” we would like to evaluate the use of fuzzy logic for trust establishment using direct feedback both for deciding which criteria to enhance and how much enhancement is appropriate. In particular, we are interested in examining hierarchical fuzzy systems, where inputs for the first level fuzzy subsystems include criteria importance and level of satisfaction. We would like to find out the effects of such changes on the trust of trustees, and the cost of achieving good trust levels.

As explained earlier, combining fuzzy logic and Q-Learning is promising for trust management, as fuzzy logic can enhance the dynamic response of Q-Learning. Investigating opportunities for combining the advantages of fuzzy logic and Q-Learning for trust establishment, both using direct feedback and indirect is a future research direction. We would like to use Q-Learning to adjust the input signals for the fuzzy logic subsystems.

Robustness and Attacking Strategies

Robustness to attacks can be indicated by how fast the trust model can detect these attacks and recover from them. The trust evaluation model described in 4.2 uses suspension and evaluate the credibility of witnesses to reduce the effect of potential attacks. However, we would like to explore other potential attacks and immunize the model against them.

When a trustee interacts for the first time, trustors assume neutral trust level of those trustees. Unfortunately, this makes the model susceptible to whitewash attacks. Another related attack is the Sybil attack that occurs when a malicious agent in the system creates a large number of fake identities. A classical way to reduce the effect of these attacks is the use of an authentication mechanism that makes registering fake identities difficult. However, classical authentication mechanisms are centralized, which contradicts with the distributed nature of the MASs.

For the trust establishment models presented in (chapter 5), trustors assumed to either provide explicit feedback or do not provide any feedback, but they were not assumed to perform any attack on trustees to harm trustees or gain more benefits.

In addition to whitewash and Sybil attacks, trustors can maximize their gains by choosing different partners each time, if the model depends on retention like Trust Establishment Using Implicit Feedback (TEIF) and Integrated Trust Establishment Model for MASs (ITE). Also, groups of trustors may agree together to reduce interaction with a trustee for awhile and consequently reduce the general retention for that trustee.

One more attack would be providing inaccurate explicit feedback. We would like to investigate more advanced attacking strategies and find out ways against them in the future.

Dynamic Parameter Tuning

The proposed models use many parameters. Even though parametric models can be adapted to different applications, deciding on the optimal, or near optimal, values for such parameters may be challenging. As a future work, we would like to optimize dynamically those parameters.

Models described in (chapter 4.2) and (chapter 5) allow agents to adjust their predictions based on various sources of information. However, all proposed models depend on many parameters such as the engagement factors, the time-decaying factor, the engagement and unengagement thresholds, the historical window, and so on.

Currently, engagement factors can be tuned dynamically based on the change in general retention of trustors. We would like to relate the amount of increment or decrements in provided Utility Gain (UG) to the level of satisfaction.

We would like to examine options for updating engagement and unengagement thresholds such as the use of probability of interaction and the probability of satisfying the needs for the requesting trustor taking into account historical requests and actual interactions.

Investigation the potential of using embedded information in interaction requests is left as a future work. Currently, we are assuming the costs associated with requesting interaction and the corresponding reply are negligible; trustor can make unlimited requests. Consequently, whether or not a trustor makes a request is of limited use. However, if a nonnegotiable cost associated with initiating requests, trustors will be selective in making requests, and initiating a request to interact with a trustee can be seen as an indicator of trust levels. We would like to use this information to update the parameters of the proposed models.

Even though a simple weighted summation can be used to aggregate prediction based on explicit and implicit feedback in ITE, we believe that an adaptive approach is preferred, where the reliability of information can be accounted for.

Using Social Structures

We would like to explore various opportunities to use social structures of relations between agents to empower trust establishment.

When trust evaluation models use testimonies from witnesses, building up a higher trust level with influencers can help establishing trust with other trustors. This is especially useful

when trustors cannot contact all witnesses or when they consider some witnesses more credible than others. This concept is widely used in marketing, where advertiser relay on famous public figures, such as well-known soccer players, to promote their products.

As we are working with software agent, rather than humans, it seems interesting to considers schemes for finding those influencers or creating them if they do not exist. It worth noting that the use of influencer is not risk-free, as they can provide misleading information, or promote competitors. Unfortunately, the use of influencers exaggerates the ”man-in-the-middle” problem where trustees have to consider not only the needs for trustor but also the requirements for the influencers, which may add extra cost to end services.

One particle approach we are interested in is the use of referral mechanisms in a way similar to hierarchical marketing where trustor(s) can benefit because other trustors interacted with the particular trustee they recommend. Furthermore, those recommending agents can act as brokers and further benefits others if they pass their recommendation and such recommendation results on a transaction with the recommended trustee. Consequently, trustor considers not only the UG they get from a trustor by the interaction, but also the benefit they get because of recommendations.

We also would like to explore the use of recommendation chain to support predicting needs of trustor. In other words, if a trustor uses recommendations or testimonies from others; it is likely that they have similar needs. Even though knowing who provided testimonies is possible at a relatively low cost in a centralized architecture, doing that in a distributed manner not only may result in extra overhead but also may open the door for one more attack. Related to this research approach is attempting to detect groups or communities based on similar needs and tune the performance in light of the general needs of each group.

7.3.2 Developing More Extensive Validation

Even though parametric models can be adapted to different applications, investigating the impact of the variation of each parameter and setting parameters for specific domains is considered as a future work.

For future work, we would like to develop more extensive validation by involving agents with different attack types and advanced attacking strategies. We will also continue to examine the robustness of our models against various kinds of attacks and the scalability of our models for various applications.

Even though we outlined a simulation frameworks in Section 4.3 and in Section 5.7 to compare our models with other relevant models, we would like to develop a unified simulation framework to evaluate and compare different trust evaluation models and different trust establishment models for multi-agent systems.

The targeted framework should introduce performance metrics of robustness and scalability, which are two essential concerns in trust management, and incorporate sophisticated attack models to test the performance of trust establishment models thoroughly. Furthermore, the framework should be modular and easily extendable to accommodate new models and attacks as they become available. The targeted framework should be able to simulate specific scenarios to analyze which models will be more effective in which situations. Such simulation test-bed will be beneficial for other researchers in the field of trust modelling to analyze and compare their trust models with the purpose of improving their performance and by offering flexibility for them to adjust parameters of simulations according to their needs.

We would like to the simulation framework to be user friendly and consist of an input interface and an output interface. The input interface provides a convenient way for users to set up simulation parameters and to run customized experiments. The output should be capable of visually presenting simulation results as well as providing the otion of textual presentation. The core engine would be the central part of the simulation framework. Its main functionalities include:

bootstrapping the simulation process and displaying the input interface for users to configure the experiments; creating a simulation with a group of agent instances based on the configurations received from users; coordinating all the components of the simulation framework to accomplish simulation tasks; collecting simulation results and sending data to the output interface in textual and visual formats.

Despite the usefulness of evaluating trust models using simulation, real-life testing is necessary before putting any model in production. Implementing the models in real life pilot project and analyzing the performance of the proposed models in actual life conditions is considered as a future work.

7.4 Summary

The purpose of this research is to promote effective, efficient, and flexible trust management in decentralized multi-agent systems. In this thesis, we propose several ideas to achieve this purpose, including the use of suspention combined with fuzzy logic and reinforcement learning for trust evaluation, the idea of using implicit feedback alone, explicit feedback alone, and the combination of both for trust establishment. In this chapter, we presented our conclusions and a brief description of directions for future research in order to extend our current work beyond the scope of the current research project. We highlighted tow main directions for future work: extending the models them self by adding more criteria, evaluating other computational approaches and resitting more attacks. The other direction for improvement is extending the validation fo the proposed models both by exiting the framework of simulation and the variety of scenarios.

References

[1] Somaya Aboulwafa and Reem Bahgat. Direct: Dirichlet-based reputation and credential trust management. In *Informatics and Systems (INFOS), 2010 The 7th International Conference on*, pages 1–8. IEEE, 2010.

[2] Basmah Almoaber and Thomas Tran. A trust certificate model for multi-agent systems. In *International Conference on E-Technologies*, pages 166–179. Springer, 2015.

[3] Gillian Basso, Nicolas Gaud, Franck Gechter, Vincent Hilaire, and Fabrice Lauri. A framework for qualifying and evaluating smart grids approaches: Focus on multi-agent technologies. 2013.

[4] Sandhya Beldona. *Reputation based Buyer Strategies for Seller Selection in Electronic Markets*. PhD thesis, Electrical Engineering & Computer Science, University of Kansas, 2008.

[5] Sandhya Beldona and Costas Tsatsoulis. Reputation based buyer strategy for seller selection for both frequent and infrequent purchases. In *ICINCO-RA (2)*, pages 84–91, 2007.

[6] H.R. Berenji. Fuzzy q-learning: a new approach for fuzzy dynamic programming. In *Fuzzy Systems, 1994. IEEE World Congress on Computational Intelligence., Proceedings of the Third IEEE Conference on*, pages 486–491 vol.1, Jun 1994.

[7] Cristian Bertocco and Carlo Ferrari. Context-dependent reputation management for soft security in multi agent systems. In *Web Intelligence and Intelligent Agent Technology,*

2008. WI-IAT'08. IEEE/WIC/ACM International Conference on, volume 3, pages 77–81. IEEE, 2008.

[8] Chris Burnett, Timothy J. Norman, and Katia Sycara. Bootstrapping trust evaluations through stereotypes. In *Proceedings of the 9th International Conference on Autonomous Agents and Multiagent Systems: Volume 1 - Volume 1*, AAMAS '10, pages 241–248, Richland, SC, 2010. International Foundation for Autonomous Agents and Multiagent Systems.

[9] Chris Burnett, Timothy J. Norman, and Katia Sycara. Stereotypical trust and bias in dynamic multiagent systems. *ACM Trans. Intell. Syst. Technol.*, 4(2):26:1–26:22, April 2013.

[10] Chris Burnett, Timothy J. Norman, and Katia P. Sycara. Trust decision-making in multi-agent systems. In *IJCAI*, pages 115–120, 2011.

[11] Christopher Burnett. *Trust Assessment and Decision-Making in Dynamic Multi-Agent Systems*. PhD thesis, Department of Computing Science, University of Aberdeen, 2011.

[12] J. CARBO, J. M. MOLINA, and J. DAVILA. Trust management through fuzzy reputation. *International Journal of Cooperative Information Systems*, 12(01):135–155, 2003.

[13] Jonathan Carter and Ali A. Ghorbani. Value centric trust in multiagent systems. In *Proceedings of the 2003 IEEE/WIC International Conference on Web Intelligence*, WI '03, pages 3–, Washington, DC, USA, 2003. IEEE Computer Society.

[14] Cristiano Castelfranchi, Rino Falcone, and Francesca Marzo. Being trusted in a social network: Trust as relational capital. In *Proceedings of the 4th International Conference on Trust Management*, iTrust'06, pages 19–32, Berlin, Heidelberg, 2006. Springer-Verlag.

[15] Cristiano Castelfranchi, Rino Falcone, and Giovanni Pezzulo. Trust in information sources as a source for trust: a fuzzy approach. In *Proceedings of the second international joint conference on Autonomous agents and multiagent systems*, pages 89–96. ACM, 2003.

[16] Yung-Han Chen, Chung-Ju Chang, and Ching Yao Huang. Fuzzy q-learning admission control for wcdma/wlan heterogeneous networks with multimedia traffic. *Mobile Computing, IEEE Transactions on*, 8(11):1469–1479, Nov 2009.

[17] P. Cingolani and J. Alcala-Fdez. jfuzzylogic: a robust and flexible fuzzy-logic inference system language implementation. In *Fuzzy Systems (FUZZ-IEEE), 2012 IEEE International Conference on*, pages 1–8, June 2012.

[18] Simone Cirani and Marco Picone. Effective authorization for the web of things. In *2015 IEEE 2nd World Forum on Internet of Things (WF-IoT)*, pages 316–320. IEEE, 2015.

[19] K. Dai, H.M. Kammoun, and A.M. Alimi. H-fql: A new reinforcement learning method for automatic hierarchization of fuzzy systems: An application to the route choice problem. In *Intelligent Systems (IS), 2012 6th IEEE International Conference*, pages 54–59, Sept 2012.

[20] Euan M Davidson and Stephen DJ McArthur. Exploiting multi-agent system technology within an autonomous regional active network management system. In *Intelligent Systems Applications to Power Systems, 2007. ISAP 2007. International Conference on*, pages 1–6. IEEE, 2007.

[21] Aris L Dimeas and Nikos D Hatziargyriou. Operation of a multiagent system for microgrid control. *IEEE Transactions on Power Systems*, 20(3):1447–1455, 2005.

[22] Nicola Dragoni. Toward trustworthy web services - approaches, weaknesses and trust-by-contract framework. In *Proceedings of the 2009 IEEE/WIC/ACM International Joint Conference on Web Intelligence and Intelligent Agent Technology - Volume 03*, WI-IAT '09, pages 599–606, Washington, DC, USA, 2009. IEEE Computer Society.

[23] Michael Engler. *Fundamental models and algorithms for a distributed reputation system*. PhD thesis, 2007.

[24] Rino Falcone and Cristiano Castelfranchi. Trust and deception in virtual societies. chapter Social Trust: A Cognitive Approach, pages 55–90. Kluwer Academic Publishers, Norwell, MA, USA, 2001.

[25] Xi Fang, Satyajayant Misra, Guoliang Xue, and Dejun Yang. Smart grid the new and improved power grid: A survey. *IEEE communications surveys & tutorials*, 14(4):944–980, 2012.

[26] Karen Fullam, Tomas B. Klos, Guillaume Muller, Jordi Sabater, Andreas Schlosser, Zvi Topol, K. Suzanne Barber, Jeffrey S. Rosenschein, Laurent Vercouter, and Marco Voss. A specification of the agent reputation and trust (ART) testbed: experimentation and competition for trust in agent societies. In *4th International Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS 2005), July 25-29, 2005, Utrecht, The Netherlands*, pages 512–518, 2005.

[27] Simson Garfinkel. *PGP: pretty good privacy*. " O'Reilly Media, Inc.", 1995.

[28] Stylianos Georgoulas, Klaus Moessner, Alexis Mansour, Menelaos Pissarides, and Panagiotis Spapis. A fuzzy reinforcement learning approach for pre-congestion notification based admission control. In *Proceedings of the 6th IFIP WG 6.6 International Autonomous Infrastructure, Management, and Security Conference on Dependable Networks and Services*, AIMS'12, pages 26–37. Springer-Verlag, Berlin, Heidelberg, 2012.

[29] Nicholas Gibbins, Stephen Harris, and Nigel Shadbolt. Agent-based semantic web services. *Web Semantics: Science, Services and Agents on the World Wide Web*, 1(2):141–154, 2004.

[30] Edgard Gnansounou, Jun Dong, Samuel Pierre, and Alejandro Quintero. Market oriented planning of power generation expansion using agent-based model. In *Power Systems Conference and Exposition, 2004. IEEE PES*, pages 1306–1311. IEEE, 2004.

[31] Jones Granatyr, Vanderson Botelho, Otto Robert Lessing, Edson Emílio Scalabrin, Jean-Paul Barthès, and Fabrício Enembreck. Trust and reputation models for multiagent systems. *ACM Comput. Surv.*, 48(2):27:1–27:42, October 2015.

[32] Tyrone Grandison and Morris Sloman. A survey of trust in internet applications. *IEEE Communications Surveys & Tutorials*, 3(4):2–16, 2000.

[33] N. Griffiths, Kuo-Ming Chao, and M. Younas. Fuzzy trust for peer-to-peer systems. In *Distributed Computing Systems Workshops, 2006. ICDCS Workshops 2006. 26th IEEE International Conference on*, pages 73–73, July 2006.

[34] Rabab Hassan and Ghadir Radman. Survey on smart grid. In *Proceedings of the IEEE SoutheastCon 2010 (SoutheastCon)*, pages 210–213. IEEE, 2010.

[35] Andreas Herzig, Emiliano Lorini, Jomi Fred Hübner, Jonathan Ben-Naim, Cristiano Castelfranchi, Robert Demolombe, Dominique Longin, Laurent Vercouter, and Olivier Boissier. Prolegomena for a logic of trust and reputation. *NorMAS*, 8:143–157, 2008.

[36] Kevin Hoffman, David Zage, and Cristina Nita-Rotaru. A survey of attack and defense techniques for reputation systems. *ACM Comput. Surv.*, 42(1):1:1–1:31, December 2009.

[37] K. Hopkinson, Xiaoru Wang, R. Giovanini, J. Thorp, K. Birman, and D. Coury. Epochs: a platform for agent-based electric power and communication simulation built from commercial off-the-shelf components. *IEEE Transactions on Power Systems*, 21(2):548–558, May 2006.

[38] Trung Dong Huynh, Nicholas R. Jennings, and Nigel R. Shadbolt. Certified reputation: How an agent can trust a stranger. In *Proceedings of the Fifth International Joint Con-*

ference on Autonomous Agents and Multiagent Systems, AAMAS '06, pages 1217–1224, New York, NY, USA, 2006. ACM.

[39] Trung Dong Huynh, Nicholas R. Jennings, and Nigel R. Shadbolt. An integrated trust and reputation model for open multi-agent systems. *Autonomous Agents and Multi-Agent Systems*, 13(2):119–154, 2006.

[40] David Jelenc, Ramón Hermoso, Jordi Sabater-Mir, and Denis Trček. Decision making matters: A better way to evaluate trust models. *Know.-Based Syst.*, 52:147–164, November 2013.

[41] Audun Jøsang, Roslan Ismail, and Colin Boyd. A survey of trust and reputation systems for online service provision. *Decision support systems*, 43(2):618–644, 2007.

[42] Audun Jøsang and Walter Quattrociocchi. Advanced features in bayesian reputation systems. In *Proceedings of the 6th International Conference on Trust, Privacy and Security in Digital Business*, TrustBus '09, pages 105–114, Berlin, Heidelberg, 2009. Springer-Verlag.

[43] Audun Jsang and Roslan Ismail. The beta reputation system. In *Proceedings of the 15th bled electronic commerce conference*, volume 5, pages 2502–2511, 2002.

[44] YR Kafle, K Mahmud, S Morsalin, and GE Town. Towards an internet of energy. In *Power System Technology (POWERCON), 2016 IEEE International Conference on*, pages 1–6. IEEE, 2016.

[45] Sepandar D. Kamvar, Mario T. Schlosser, and Hector Garcia-Molina. The eigentrust algorithm for reputation management in p2p networks. In *Proceedings of the 12th International Conference on World Wide Web*, WWW '03, pages 640–651, New York, NY, USA, 2003. ACM.

[46] Abhilash Kantamneni, Laura E. Brown, Gordon Parker, and Wayne W. Weaver. Survey of multi-agent systems for microgrid control. *Engineering Applications of Artificial Intelligence*, 45:192 – 203, 2015.

[47] Reid Kerr. *Addressing the Issues of Coalitions and Collusion in Multiagent Systems*. PhD thesis, University of Waterloo, 2013.

[48] Reid C. Kerr. Toward secure trust and reputation systems for electronic marketplaces. Master's thesis, Computer Science, University of Waterloo, 2007.

[49] Babak Khosravifar. *Trust and Reputation in Multi-Agent Systems*. PhD thesis, Concordia University Montréal, Québec, Canada, 2012.

[50] Babak Khosravifar, Jamal Bentahar, Maziar Gomrokchi, and Rafiul Alam. Crm: An efficient trust and reputation model for agent computing. *Knowl.-Based Syst.*, 30:1–16, 2012.

[51] Lukas Klejnowski, Yvonne Bernard, Jorg Hahner, and Christian Muller-Schloer. An architecture for trust-adaptive agents. In *Self-Adaptive and Self-Organizing Systems Workshop (SASOW), 2010 Fourth IEEE International Conference on*, pages 178–183. IEEE, 2010.

[52] Andrew Koster, Jordi Sabater-Mir, and Marco Schorlemmer. Personalizing communication about trust. In *Proceedings of the 11th International Conference on Autonomous Agents and Multiagent Systems-Volume 1*, pages 517–524. International Foundation for Autonomous Agents and Multiagent Systems, 2012.

[53] Eleni Koutrouli and Aphrodite Tsalgatidou. Reputation-based trust systems for p2p applications: design issues and comparison framework. In *International Conference on Trust, Privacy and Security in Digital Business*, pages 152–161. Springer, 2006.

[54] Eleni Koutrouli and Aphrodite Tsalgatidou. Reputation systems evaluation survey. *ACM Computing Surveys*, 48(3):1–28, Dec 2015.

[55] AL Kulasekera, RARC Gopura, KTMU Hemapala, and N Perera. A review on multi-agent systems in microgrid applications. In *Innovative Smart Grid Technologies-India (ISGT India), 2011 IEEE PES*, pages 173–177. IEEE, 2011.

[56] Yinsheng Li, Weiming Shen, and Hamada Ghenniwa. Agent-based web services framework and development environment. *Computational Intelligence*, 20(4):678–692, 2004.

[57] John Linn. Trust models and management in public-key infrastructures. *RSA laboratories*, 20, 2000.

[58] Dong Liu, Yunping Chen, Guang Shen, and Youping Fan. A multi-agent based approach for modeling and simulation of bulk power system restoration. In *2005 IEEE/PES Transmission & Distribution Conference & Exposition: Asia and Pacific*, pages 1–6. IEEE, 2005.

[59] Fang Liu, Li Yao, Weiming Zhang, Hua Liu, and Hua Zhang. A conceptual model of agent mediated web service. In *Services Computing, 2004.(SCC 2004). Proceedings. 2004 IEEE International Conference on*, pages 638–642. IEEE, 2004.

[60] Siyuan Liu, Han Yu, Chunyan Miao, and Alex C Kot. A fuzzy logic based reputation model against unfair ratings. In *Proceedings of the 2013 international conference on Autonomous agents and multi-agent systems*, pages 821–828. International Foundation for Autonomous Agents and Multiagent Systems, 2013.

[61] Michela Longo, Mariacristina Roscia, and George Cristian Lazaroiu. Innovating multi-agent systems applied to smart city. *Research Journal of Applied Sciences, Engineering and Technology*, 7(20):4296–4302, 2014.

[62] Gehao Lu, Joan Lu, Shaowen Yao, and Yau Jim Yip. A review on computational trust models for multi-agent systems. *The open information science journal*, 2:18–25, 2009.

[63] Sean Luke, Claudio Cioffi-Revilla, Liviu Panait, Keith Sullivan, and Gabriel Balan. Mason: A multiagent simulation environment. *Simulation*, 81(7):517–527, July 2005.

[64] Robert Lukomski and Kazimierz Wilkosz. Modeling of multi-agent system for power system topology verification with use of petri nets. In *Modern Electric Power Systems (MEPS), 2010 Proceedings of the International Symposium*, pages 1–6. IEEE, 2010.

[65] Zakaria Maamar, Soraya Kouadri Mostefaoui, and Hamdi Yahyaoui. Toward an agent-based and context-oriented approach for web services composition. *IEEE Transactions on Knowledge and Data Engineering*, 17(5):686–697, 2005.

[66] Zaki Malik and Athman Bouguettaya. Reputation bootstrapping for trust establishment among web services. *Internet Computing, IEEE*, 13(1):40–47, 2009.

[67] Zaki Malik and Athman Bouguettaya. *Trust Management for Service-Oriented Environments*. Springer Publishing Company, Incorporated, 1st edition, 2009.

[68] E.H. Mamdani and S. Assilian. An experiment in linguistic synthesis with a fuzzy logic controller. *International Journal of Man-Machine Studies*, 7(1):1 – 13, 1975.

[69] Stephen Marsh and Pamela Briggs. Examining trust, forgiveness and regret as computational concepts. In *Computing with social trust*, pages 9–43. Springer, 2009.

[70] Stephen Paul Marsh. *Formalising trust as a computational concept*. PhD thesis, University of Stirling, 1994.

[71] E Michael Maximilien and Munindar P Singh. Reputation and endorsement for web services. *ACM SIGecom Exchanges*, 3(1):24–31, 2001.

[72] SDJ McArthur and EM Davidson. Multi-agent systems for diagnostic and condition monitoring applications. In *Power Engineering Society General Meeting, 2004. IEEE*, pages 50–54. IEEE, 2004.

[73] D. Harrison McKnight, Vivek Choudhury, and Charles Kacmar. Developing and validating trust measures for e-commerce: An integrative typology. *Info. Sys. Research*, 13(3):334–359, September 2002.

[74] J.M. Mendel. Fuzzy logic systems for engineering: a tutorial. *Proceedings of the IEEE*, 83(3):345–377, Mar 1995.

[75] Ehsan Mokhtari, Zeinab Noorian, Behrouz Tork Ladani, and Mohammad Ali Nematbakhsh. A context-aware reputation-based model of trust for open multi-agent environments. In *Proceedings of the 24th Canadian conference on Advances in artificial intelligence*, pages 301–312. Springer-Verlag, 2011.

[76] Mohammad H. Moradi, Saleh Razini, and S. Mahdi Hosseinian. State of art of multiagent systems in power engineering: A review. *Renewable and Sustainable Energy Reviews*, 58:814 – 824, 2016.

[77] Lik Mui, Mojdeh Mohtashemi, and Ari Halberstadt. Notions of reputation in multi-agents systems: a review. In *Proceedings of the first international joint conference on Autonomous agents and multiagent systems: part 1*, pages 280–287. ACM, 2002.

[78] P. Munoz, R. Barco, and I. de la Bandera. Optimization of load balancing using fuzzy q-learning for next generation wireless networks. *Expert Systems with Applications*, 40(4):984 – 994, 2013.

[79] P. Munoz, D. Laselva, R. Barco, and P. Mogensen. Dynamic traffic steering based on fuzzy q-learning approach in a multi-rat multi-layer wireless network. *Comput. Netw.*, 71:100–116, October 2014.

[80] A. M. Mzahm, M. S. Ahmad, and A. Y. C. Tang. Agents of things (aot): An intelligent operational concept of the internet of things (iot). In *2013 13th International Conference on Intellient Systems Design and Applications*, pages 159–164, Dec 2013.

[81] Zeinab Noorian and Mihaela Ulieru. The state of the art in trust and reputation systems: A framework for comparison. *J. Theor. Appl. Electron. Commer. Res.*, 5(2):97–117, August 2010.

[82] Beng Chin Ooi, Chu Yee Liau, and Kian-Lee Tan. Managing trust in peer-to-peer systems using reputation-based techniques. In *International Conference on Web-Age Information Management*, pages 2–12. Springer, 2003.

[83] Mike P Papazoglou. Agent-oriented technology in support of e-business. *Communications of the ACM*, 44(4):71–77, 2001.

[84] CostasP. Pappis and ConstantinosI. Siettos. Fuzzy reasoning. In EdmundK. Burke and Graham Kendall, editors, *Search Methodologies*, pages 437–474. Springer US, 2005.

[85] D T Pham and M Castellani. Action aggregation and defuzzification in mamdani-type fuzzy systems. *Proceedings of the Institution of Mechanical Engineers, Part C: Journal of Mechanical Engineering Science*, 216(7):747–759, 2002.

[86] P. Pico-Valencia and J. A. Holgado-Terriza. Semantic agent contracts for internet of agents. In *2016 IEEE/WIC/ACM International Conference on Web Intelligence Workshops, Omaha, NE, USA, October 13-16, 2016*, pages 76–79.

[87] Isaac Pinyol and Jordi Sabater-Mir. Computational trust and reputation models for open multi-agent systems: a review. *Artificial Intelligence Review*, 40(1):1–25, 2013.

[88] Isaac Pinyol, Jordi Sabater-Mir, Pilar Dellunde, and Mario Paolucci. Reputation-based decisions for logic-based cognitive agents. *Autonomous Agents and Multi-Agent Systems*, 24(1):175–216, 2012.

[89] Manisa Pipattanasomporn, Hassan Feroze, and S Rahman. Multi-agent systems in a distributed smart grid: Design and implementation. In *Power Systems Conference and Exposition, 2009. PSCE'09. IEEE/PES*, pages 1–8. IEEE, 2009.

[90] Michele Piunti, Matteo Venanzi, Rino Falcone, and Cristiano Castelfranchi. Multimodal trust formation with uninformed cognitive maps (uncm). In *Proceedings of the 11th International Conference on Autonomous Agents and Multiagent Systems - Volume 3*, AAMAS '12, pages 1241–1242, Richland, SC, 2012. International Foundation for Autonomous Agents and Multiagent Systems.

[91] Sarvapali D Ramchurn, Dong Huynh, and Nicholas R Jennings. Trust in multi-agent systems. *The Knowledge Engineering Review*, 19(01):1–25, 2004.

[92] Kevin Regan and Robin Cohen. Indirect reputation assessment for adaptive buying agents in electronic markets. In *Proceedings of business agents and the semantic web workshop (BASeWEB'05)*, volume 1, pages 121–130, Victoria, British Columbia, Canada, May 2005.

[93] Kevin Regan, Robin Cohen, and Thomas Tran. Sharing models of sellers amongst buying agents in electronic marketplaces. In *Proceedings of Decentralized, Agent Based, and Social Approaches to User Modelling Workshop*, volume 1, pages 75–79, Edinburgh, UK, July 2005.

[94] Kevin Regan, Pascal Poupart, and Robin Cohen. Bayesian reputation modeling in e-marketplaces sensitive to subjectivity, deception and change. In *Proceedings of the National Conference on Artificial Intelligence*, volume 21, page 1206. Menlo Park, CA; Cambridge, MA; London; AAAI Press; MIT Press; 1999, 2006.

[95] Achim Rettinger, Matthias Nickles, and Volker Tresp. A statistical relational model for trust learning. In *Proceedings of the 7th international joint conference on Autonomous agents and multiagent systems-Volume 2*, pages 763–770. International Foundation for Autonomous Agents and Multiagent Systems, 2008.

[96] Cédric Richier, Eitan Altman, Rachid Elazouzi, Tania Altman, Georges Linares, and Yonathan Portilla. Modelling view-count dynamics in youtube. *arXiv preprint arXiv:1404.2570*, 2014.

[97] Sebastian Ries. *Trust in ubiquitous computing*. PhD thesis, Technische Universität, 2009.

[98] Ronald L Rivest, Adi Shamir, and Leonard Adleman. A method for obtaining digital signatures and public-key cryptosystems. *Communications of the ACM*, 21(2):120–126, 1978.

[99] A Rogers, SD Ramchurn, and NR Jennings. Delivering the smart grid: challenges for autonomous agents and multi-agent systems research. In *Proceedings of the Twenty-Sixth AAAI Conference on Artificial Intelligence*, pages 2166–2172. AAAI Press, 2012.

[100] Timothy J. Ross. *Fuzzy Logic with Engineering Applications*. John Wiley & Sons, Ltd, 2010.

[101] Shounak Roychowdhury and Witold Pedrycz. A survey of defuzzification strategies. *International Journal of intelligent systems*, 16(6):679–695, 2001.

[102] Jordi Sabater and Carles Sierra. Regret: A reputation model for gregarious societies. In *Fourth workshop on deception fraud and trust in agent societies*, volume 70, 2001.

[103] Jordi Sabater and Carles Sierra. Review on computational trust and reputation models. *Artif. Intell. Rev.*, 24(1):33–60, September 2005.

[104] Radhika Saini and Ravinder Kumar Gautam. Establishment of dynamic trust among nodes in mobile ad-hoc network. In *Computational Intelligence and Communication Networks (CICN), 2011 International Conference on*, pages 346–349. IEEE, 2011.

[105] Amirali Salehi-Abari and Tony White. Trust models and con-man agents: From mathematical to empirical analysis. In *Twenty-Fourth AAAI Conference on Artificial Intelligence*, 2010.

[106] Amirali Salehi-Abari and Tony White. Dart: a distributed analysis of reputation and trust framework. *Computational Intelligence*, 28(4):642–682, 2012.

[107] K SANGEETA and GK PATRA. A fuzzy based hierarchical trust framework to rate the cloud service providers based on infrastructure facilities. *International Journal of Performability Engineering*, 12(1):55–62, 2016.

[108] Sandip Sen. A comprehensive approach to trust management. In *Proceedings of the 12th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2013)*, 2013.

[109] Sandip Sen, Mahendra Sekaran, John Hale, et al. Learning to coordinate without sharing information. In *AAAI*, pages 426–431, 1994.

[110] Yun Cheng Shen, Hong Zhang, and Xu Liang Duan. Trust model of p2p network based on fuzzy degree of credit. In *Applied Mechanics and Materials*, volume 738, pages 1231–1235. Trans Tech Publ, 2015.

[111] Hossein Shirgahi, Mehran Mohsenzadeh, and Hamid Haj Seyyed Javadi. A three level fuzzy system for evaluating the trust of single web services. *Journal of Intelligent & Fuzzy Systems*, 32(Preprint):1–23, 2016.

[112] Hossein Shirgahi, Mehran Mohsenzadeh, and Hamid Haj Seyyed Javadi. A three level fuzzy system for evaluating the trust of single web services. *Journal of Intelligent & Fuzzy Systems*, 32(1):589–611, 2017.

[113] Munindar P Singh. Trust as dependence: a logical approach. In *The 10th International Conference on Autonomous Agents and Multiagent Systems-Volume 2*, pages 863–870. International Foundation for Autonomous Agents and Multiagent Systems, 2011.

[114] Sarbjeet Singh and Jagpreet Sidhu. An approach for determining trustworthiness of individuals in a web-based social network. *Arabian Journal for Science & Engineering (Springer Science & Business Media BV)*, 41(2), 2016.

[115] Sarbjeet Singh and Jagpreet Sidhu. An approach for determining trustworthiness of individuals in a web-based social network. *Arabian Journal for Science and Engineering*, 41(2):461–477, 2016.

[116] Sean W. Smith. *Trusted computing platforms. Design and applications.* New York, NY: Springer, 2005.

[117] S. Song, Kai Hwang, Runfang Zhou, and Yu-Kwong Kwok. Trusted p2p transactions with fuzzy reputation aggregation. *Internet Computing, IEEE*, 9(6):24–34, Nov 2005.

[118] Junjie Sun and Leigh Tesfatsion. An agent-based computational laboratory for wholesale power market design. In *Power Engineering Society General Meeting, 2007. IEEE*, pages 1–6. IEEE, 2007.

[119] Alistair Sutcliffe and Di Wang. Computational modelling of trust and social relationships. *Journal of Artificial Societies and Social Simulation*, 15(1):3, 2012.

[120] Richard S. Sutton and Andrew G. Barto. *Introduction to Reinforcement Learning*. MIT Press, Cambridge, MA, USA, 1st edition, 1998.

[121] Ayman Tajeddine, Ayman Kayssi, Ali Chehab, and Hassan Artail. Patrol: a comprehensive reputation-based trust model. *International Journal of Internet Technology and Secured Transactions*, 1(1-2):108–131, 2007.

[122] Ayman Tajeddine, Ayman Kayssi, Ali Chehab, and Hassan Artail. Fuzzy reputation-based trust model. *Applied Soft Computing*, 11(1):345 – 355, 2011.

[123] W. T. Luke Teacy, Jigar Patel, Nicholas R. Jennings, and Michael Luck. TRAVOS: trust and reputation in the context of inaccurate information sources. *Autonomous Agents and Multi-Agent Systems*, 12(2):183–198, 2006.

[124] WT Luke Teacy. *Agent-based trust and reputation in the context of inaccurate information sources*. PhD thesis, University of Southampton, 2006.

[125] WT Luke Teacy, Michael Luck, Alex Rogers, and Nicholas R Jennings. An efficient and versatile approach to trust and reputation using hierarchical bayesian modelling. *Artificial Intelligence*, 193:149–185, 2012.

[126] Yasushi Tomita, Chihiro Fukui, Hiroyuki Kudo, Jun Koda, and Kuniaki Yabe. A cooperative protection system with an agent model. *IEEE Transactions on Power Delivery*, 13(4):1060–1066, 1998.

[127] Thomas Tran. *Reputation-Oriented Reinforcement Learning Strategies for Economically-Motivated Agents in Electronic Market Environments*. PhD thesis, Waterloo, Ontario, 2004.

[128] Thomas Tran and Robin Cohen. Improving user satisfaction in agent-based electronic marketplaces by reputation modelling and adjustable product quality. In *AAMAS*, pages 828–835, 2004.

[129] Thomas Tran, Robin Cohen, and Eric Langlois. Establishing trust in multiagent environments: realizing the comprehensive trust management dream. In *Proceedings of the 17th International Workshop on Trust in Agent Societies,*, 2014.

[130] Thomas T. Tran. Protecting buying agents in e-marketplaces by direct experience trust modelling. *Knowl. Inf. Syst.*, 22(1):65–100, 2010.

[131] Thomas T. Tran and Robin Cohen. A reputation-oriented reinforcement learning strategy for agents in electronic marketplaces. *Computational Intelligence*, 18(4):550–565, 2002.

[132] Leo van Moergestel, Melvin van den Berg, Marco Knol, Rick van der Paauw, Kasper van Voorst, Erik Puik, Daniel Telgen, and John-Jules Meyer. Internet of smart things - a study on embedding agents and information in a device. In *Proceedings of the 8th International Conference on Agents and Artificial Intelligence - Volume 1: ICAART,*, pages 102–109, 2016.

[133] Matteo Venanzi, Michele Piunti, Rino Falcone, and Cristiano Castelfranchi. Facing openness with socio-cognitive trust and categories. In *Proceedings of the Twenty-Second international joint conference on Artificial Intelligence-Volume Volume One*, pages 400–405. AAAI Press, 2011.

[134] Ruben Verborgh, Erik Mannens, and Rik Van de Walle. The rise of the Web for Agents. In *Proceedings of the First International Conference on Building and Exploring Web Based Environments*, pages 69–74, January 2013.

[135] Pavel Vrba, Vladimír Mařík, Pierluigi Siano, Paulo Leitão, Gulnara Zhabelova, Valeriy Vyatkin, and Thomas Strasser. A review of agent and service-oriented concepts applied to intelligent energy systems. *IEEE transactions on industrial informatics*, 10(3):1890–1903, 2014.

[136] Hai H Wang, Nick Gibbins, Terry Payne, Alina Patelli, and Yangang Wang. A survey of semantic web services formalisms. *Concurrency and Computation: Practice and Experience*, 27(15):4053–4072, 2015.

[137] Hongbing Wang, Joshua Zhexue Huang, Yuzhong Qu, and Junyuan Xie. Web services: problems and future directions. *Web Semantics: Science, Services and Agents on the World Wide Web*, 1(3):309–320, 2004.

[138] Ping Wang and Zili Zhang. A computation trust model with trust network in multi-agent systems. In *Proceedings of the 2005 International Conference on Active Media Technology, 2005.(AMT 2005).*, pages 389–392. IEEE, 2005.

[139] Yao Wang and Julita Vassileva. Bayesian network-based trust model. In *Web Intelligence, 2003. WI 2003. Proceedings. IEEE/WIC International Conference on*, pages 372–378. IEEE, 2003.

[140] Yao Wang and Julita Vassileva. Toward Trust and Reputation Based Web Service Selection: A Survey. *International Transactions on Systems Science and Applications (ITSSA) Journal*, 3(2), 2007.

[141] Yao Wang, Jie Zhang, and Julita Vassileva. Super-agent based reputation management with a practical reward mechanism in decentralized systems. In *Trust Management*, 2010.

[142] Yao Wang, Jie Zhang, and Julita Vassileva. A super-agent-based framework for reputation management and community formation in decentralized systems. *Computational Intelligence*, 30(4):722–751, 2014.

[143] D. J. Watts and S. H. Strogatz. Collective dynamics of'small-world'networks. *Nature*, 393(6684):409–10, 1998.

[144] Andrew G West, Sampath Kannan, Insup Lee, and Oleg Sokolsky. An evaluation framework for reputation management systems. *Trust Modeling and Management in Digital Environments: From Social Concept to System Development*, pages 282–308, 2010.

[145] Michael Wooldridge. *An Introduction to MultiAgent Systems*. Wiley Publishing, 2nd edition, 2009.

[146] QH Wu, JQ Feng, WH Tang, and J Fitch. Multi-agent based substation automation systems. In *Power Engineering Society General Meeting, 2005. IEEE*, pages 1048–1049, 2005.

[147] Li Xiong and Ling Liu. Peertrust: Supporting reputation-based trust for peer-to-peer electronic communities. *IEEE Transactions on Knowledge and Data Engineering*, 16(7):843–857, 2004.

[148] Yinliang Xu, Wenxin Liu, and Jun Gong. Stable multi-agent-based load shedding algorithm for power systems. *IEEE Transactions on Power Systems*, 26(4):2006–2014, 2011.

[149] Liangjun You. *An Adaptive Reputation-based Trust Model for Intelligent Agents in e-Marketplace*. PhD thesis, Arlington, TX, USA, 2007. AAI3288896.

[150] Bin Yu and Munindar P. Singh. An evidential model of distributed reputation management. In *Proceedings of the First International Joint Conference on Autonomous Agents and Multiagent Systems: Part 1*, AAMAS '02, pages 294–301, New York, NY, USA, 2002. ACM.

[151] Han Yu. *Situation-aware trust management in multi-agent systems*. PhD thesis, 2014.

[152] Han Yu, Chunyan Miao, Bo An, Cyril Leung, and Victor R. Lesser. A reputation management approach for resource constrained trustee agents. In *Proceedings of the Twenty-Third International Joint Conference on Artificial Intelligence, Beijing, China, August 03 - 09, 2013*, pages 418–424.

[153] Han Yu, Zhiqi Shen, C. Leung, Chunyan Miao, and V.R. Lesser. A survey of multi-agent trust management systems. *Access, IEEE*, 1:35–50, 2013.

[154] Han Yu, Zhiqi Shen, and Cyril Leung. From internet of things to internet of agents. In *Green Computing and Communications (GreenCom), 2013 IEEE and Internet of Things (iThings/CPSCom), IEEE International Conference on and IEEE Cyber, Physical and Social Computing*, pages 1054–1057. IEEE, 2013.

[155] Han Yu, Zhiqi Shen, Chunyan Miao, and Bo An. A reputation-aware decision-making approach for improving the efficiency of crowdsourcing systems. In *Proceedings of the 2013 International Conference on Autonomous Agents and Multi-agent Systems, St. Paul, MN, USA, May 06 - 10, 2013*, pages 1315–1316.

[156] Giorgos Zacharia and Pattie Maes. Trust management through reputation mechanisms. *Applied Artificial Intelligence*, 14(9):881–907, 2000.

[157] Jie Zhang. *Promoting honesty in electronic marketplaces: Combining trust modeling and incentive mechanism design*. PhD thesis, University of Waterloo, 2009.

[158] Jie Zhang. Trust management for vanets: Challenges, desired properties and future directions. *Int. J. Distrib. Syst. Technol.*, 3(1):48–62, January 2012.

[159] Jie Zhang and Robin Cohen. Design of a mechanism for promoting honesty in e-marketplaces. In *Proceedings of the Twenty-Second AAAI Conference on Artificial Intelligence, July 22-26, 2007, Vancouver, British Columbia, Canada*, pages 1495–1500, 2007.