Trust Evolution in IoT Networks with Multiple Attributes

by

Nuray Baltaci Akhuseyinoglu

M.S. in Information Systems, Middle East Technical University, 2014

B.S. in Industrial Engineering, TOBB University of Economics and Technology, 2011

Submitted to the Graduate Faculty of
the School of Computing and Information in partial fulfillment
of the requirements for the degree of

Doctor of Philosophy

University of Pittsburgh

2023

UNIVERSITY OF PITTSBURGH

SCHOOL OF COMPUTING AND INFORMATION

This dissertation was presented

by

Nuray Baltaci Akhuseyinoglu

It was defended on

April 3, 2023

and approved by

Dr. Prashant Krishnamurthy, School of Computing and Information, University of Pittsburgh

Dr. Amy Babay, School of Computing and Information, University of Pittsburgh

Dr. Konstantinos Pelechrinis, School of Computing and Information, University of Pittsburgh

Dr. Mai Abdelhakim, Department of Electrical and Computer Engineering, University of Pittsburgh

Dissertation Director: Dr. Prashant Krishnamurthy, School of Computing and Information, University of Pittsburgh

Trust Evolution in IoT Networks with Multiple Attributes

Nuray Baltaci Akhuseyinoglu, PhD

University of Pittsburgh, 2023

The Internet of Things (IoT) is a communication paradigm comprising millions of devices, a.k.a *things* or *nodes*, growing in number. Things are interconnected smart devices that operate with or without human intervention, such as sensors, actuators, RFID devices, wearable devices, or more powerful computing systems. The heterogeneity of devices, software components, and network infrastructure in IoT leads to increased attack surfaces. One of the significant security threats for IoT is untrustworthy data and operations that may arise due to device compromise, vulnerable transmission medium, or faulty sensors. It is essential to ensure trust in the data and operations in IoT, as it is fundamental for people to overcome perceptions of uncertainty and risk in using IoT services and applications. The lack of trust may have dire consequences for IoT. For example, an attacker compromising an IoT device can generate or report bogus data, boost the reputation of malicious nodes, and ruin that of benign nodes.

There are security mechanisms to defend against external attacks in IoT, such as cryptographic algorithms. Yet, they cannot identify internal attacks as a benign node could turn into a malicious node anytime after joining the network because of compromise or malfunction. Trust management solutions are essential for detecting misbehaving legitimate nodes in IoT when cryptographic measures are not available or applicable. IoT brings extra challenges to trust management due to ever-changing network topology, heterogeneity in devices and network topology, and limited resources of constrained devices. Promising solutions have been proposed for IoT trust management to address these challenges. Yet, they are limited in accommodating key trust properties and automated trust computation needs for IoT environments.

The research in this dissertation focuses on trust evolution in IoT networks, drawing upon trust research in social sciences. Towards this, we distill significant aspects of trust evolution in social sciences and capture them in solutions for IoT trust management through automated trust computations. Specifically, we propose an automated trust computation framework based on the Multi-Attribute Decision Making (MADM) approach and Evidence-based Subjective Logic (EBSL) to account for the multi-dimensionality and uncertainty aspects of trust. We evaluate the performance of the proposed MADM-EBSL framework concerning varying levels of network connectivity and trust problem size. Additionally, we propose to extend the trust model of this framework with trust attributes based on our review of social sciences trust literature. We compare the two frameworks to investigate the effect of including additional attributes in trust computations. Finally, we explore trust repair strategies for IoT and a model to reflect these on automated trust computations. We present the findings of our evaluation of the proposed trust repair model.

Table of Contents

List of Tables

List of Figures

Preface

I am sincerely grateful to everyone who contributed to the completion of this dissertation. The journey was challenging yet rewarding, and their support and encouragement were instrumental.

First and foremost, I extend my thanks to my advisor Dr. Prashant Krishnamurthy, for his invaluable guidance and unwavering support. His expertise, invaluable insights, and constructive feedback have significantly shaped the direction of this dissertation and improved its quality. I am also immensely grateful to the members of my dissertation committee, Dr. Konstantinos Pelechrinis, Dr. Mai Abdelhakim, and Dr. Amy Babay, for their valuable time, constructive criticism, and expert suggestions that greatly influenced this work.

I also want to express my deepest gratitude to my family. Their unconditional love, understanding, and encouragement have been the cornerstone of my perseverance and success throughout this academic journey. I dedicate this dissertation to the living memories of my father, Kamil Baltacı, being proud of all my success throughout his life. I am indebted to my precious mother, Neziha Baltacı, for her endless support, always being nearest to me, and praying for me. I also express my heartfelt gratitude to my sisters Öznur Baltacı Oral and Saliha Baltacı Akgün, for encouraging and trusting me all the time. I am grateful to my husband, Kamil Akhuseyinoglu, for his continued love and support in every step throughout my PhD journey. I am deeply grateful to have our little one Kemal Akhuseyinoglu join our family, bringing joy to us during this journey. He has been a source of motivation during both challenging and rewarding times. I also would like to extend my gratitude to my cousin Dilara Avcı for her continuous support and friendship.

Last but not least, I wish to extend my appreciation to my colleagues and friends who provided unwavering support, engaging discussions, and a stimulating academic environment. I would like to give a special shout-out to my beloved pets, Gofret, Caramel, and Latte, whose adorable presence and unconditional love provided much-needed stress relief and companionship throughout my PhD and the process of writing this dissertation.

1.0 Introduction

The Internet of Things (IoT) is a still-developing communication paradigm that comprises millions of devices, a.k.a *things* or *nodes*, growing in number [64, 40, 11]. Things are connected to the Internet [112, 11], either directly or through gateways. They are interconnected smart devices [193, 40, 11, 9, 1, 107] that operate with or without human intervention and include sensors, actuators, RFID devices, wearable devices, or more powerful computing systems that interact/collaborate to fulfill a common goal [193, 40, 11, 9, 188]. They collect data from their surroundings, process them, or act back on the physical world [193, 40, 11]. IoT allows interaction among users, devices, applications, and the environment [11, 9], and provides a platform for context-aware computations [40]. IoT has been having an impact on several applications/domains, including but not limited to smart cities [56, 4], environmental monitoring [192, 193, 107], manufacturing [55, 11], smart transportation [56, 193, 11], and smart healthcare [56, 193, 11].

IoT is characterized by the *heterogeneity* of devices, software components, and network infrastructure [193, 40, 108, 4, 11, 9, 1, 107]. The heterogeneity of devices corresponds to the variety in their capabilities — computing, storage, and power resources—, operating systems, vendor, functionality, etc. [40, 4, 11, 1, 188]. The heterogeneity of network infrastructure corresponds to different networking technologies that could be integrated into IoT networks, such as sensor networks, wireless local area networks (WLANs), wide area networks (WAN), Bluetooth, Zigbee, etc. [193]. The heterogeneity in IoT devices and network infrastructure poses a challenge to the security of IoT environments as it leads to increased security threats/attack surfaces [189, 11, 9]. Security threats to IoT include attacks on IoT devices, such as Man-In-The-Middle (MITM), distributed Denial of Service (DDoS), impersonation, wormholes, and physical node compromise attacks [11]. One of the significant security challenges for IoT is untrustworthy data and operations [4]. Estimating trust in data and devices is challenging as the number of things is rising [64]. The data collected by IoT nodes are transmitted through networks, analyzed in real-time, and used for actuation [108] and other decisions. Infringement of trust in any of these processes or the data may signal a possible compromise and arise from malfunction leading to redundant data copies, vulnerable transmission medium, or faulty sensors [108].

Ensuring trust in the data and operations, i.e., *trust management*, is significant for IoT due to several reasons. First of all, trust mechanisms are fundamental for people to overcome perceptions of uncertainty and risk in using IoT services and applications [49]. Also, trust is essential for reliable IoT services as service subscribers interact with a large amount of data without being aware of their source [40]. For instance, in a smart traffic scenario, sensors at intersections may connect to smart cars passing by and send notifications to the smartphones of blind or deaf pedestrians. The data coming from smart cars and sensors need to be reliable for preventing false alarms [1]. Finally, the lack of trust may have dire consequences for IoT. For example, if an attacker compromises/captures an IoT device, it may be able to command the generation or reporting of bogus data [108]. A malicious node may interrupt the proper functioning of an IoT network as it may boost the reputation of malicious nodes and ruin that of benign nodes [1]. The lack of trust may even disrupt the whole data management process and prevent the proper functioning of IoT devices [4].

1.0.1 Need for Trust Management in IoT

There are security mechanisms to defend against external attacks in IoT, such as cryptographic algorithms to provide confidentiality, integrity, and network design for availability of devices, communications, and data [154]. Yet, they do not address internal or insider attacks [154]. A benign IoT node could turn into a malicious one anytime after joining the network and behaving benign for some period [11]. Traditional cryptographic security mechanisms cannot identify such behaviors as insider nodes have already joined the network, successfully exchange cryptographic keys, and establish secure communications [11]. For instance, an internal attack can be launched by a compromised IoT device with a legitimate cryptographic key by altering the exchanged data or inserting bogus information without being recognized [188, 154]. Another attack that cannot be identified by traditional security mechanisms is a wormhole attack. A wormhole attack creates a tunnel between a victim and remote nodes by an attacker by capturing packets in the network and re-transmitting them to distant nodes [11]. This attack does not require any knowledge of cryptographic keys or network structure [11]. Also, strong security measures cannot be implemented on constrained IoT nodes [1]. Therefore, trust management solutions are needed for IoT environments [188]. Trust management solutions are essential for detecting misbehaving legitimate nodes [9, 154] and addressing attacks performed by masquerading nodes in IoT [11] when cryptographic measures are not available [107] or applicable. A trust management solution performs a quick check by measuring trust at the device or transaction level, allows data exchange only between trustworthy nodes, and minimizes the likelihood of data compromise [11].

IoT brings extra challenges to trust management due to its unique features [11, 4, 107]. One unique feature is the heterogeneity of network topology and devices, aforementioned. *Ever-changing network topology* is another challenge [108, 11], which arises from constantly moving or adding/removing IoT nodes [108, 11, 107]. *Data redundancy* corresponds to multiple copies of data kept for data availability purposes. It may lead to uncertainty and untrustworthy data and is another source of challenge for trust management. *Limited resources* of constrained IoT devices, i.e., computation, storage, and power resources, also poses a challenge for trust management [40, 11, 9, 1, 107, 154]. Due to these challenges in trust management and the unique characteristics of IoT, trust management solutions for traditional networks do not accommodate the requirements for IoT [40, 11].

1.0.2 Trust Management Schemes

Trust management solutions for IoT have been proposed to address some of the challenges mentioned above. Existing work on trust management in IoT is further discussed in Chapter 2. A trust management solution/trust model is targeted to establish trust among nodes in a network [126, 40, 11] or in different IoT networks [188] by assessing the trustworthiness of nodes [107, 188], by detecting malicious and faulty behaviors [126, 40, 11, 154], and selecting the most reliable trustee node for a trustor node [126, 40, 11, 1]. Towards this, trust management schemes typically use a trust metric(s) to output trust levels, which combines different trust sources (e.g., direct and indirect trust [106]) and different attributes [176], including contextual attributes, reputation (a.k.a indirect trust opinions [107]), experience/behavior history (a.k.a direct trust opinions [107]), and knowledge of IoT nodes [40, 11, 107, 176]. Trust levels are further assessed [40], such as by comparing them to a threshold value [107, 188], to make decisions, such as about service selection or actuation [107]. Yet, existing solutions have some drawbacks. For example, trust management schemes built on the Social IoT (SIoT) paradigm[1] could require human intervention [12] and may not be suitable for automated trust computation. [5]. Also, trust measurement approaches based on feedback and service recommendation are prone to the same drawback [5] as the dynamic nature of IoT and an unpredictable number of devices render feedback management challenging [129]. The application of decentralized trust measurement approaches may be limited as they may rely on resource-constrained IoT nodes for the computation and storage of trust values.

1.0.3 Focus of the Dissertation

In this dissertation, I explore *automated trust evolution* in IoT networks, i.e., stages that a trust relationship between two IoT devices goes through. The research comprises the following parts.

1. First, we have explored in [3] how trust has been studied in a set of social science research papers to extract the important aspects of trust evolution. For this purpose, we have distilled social sciences trust literature into a detailed, yet composable framework. Next, I apply these concepts to IoT as trust computation approaches in an automated way where possible.

2. We have developed a trust computation framework for IoT in [5], —*MADM-EBSL Framework* (Multi-Attribute Decision Making (MADM) - Evidence-Based Subjective Logic (EBSL) Framework) —, based on multiple attributes and examined their effectiveness in capturing malicious and faulty nodes.

[1] SIoT is an integration of social networking and IoT concepts [12]. In SIoT, nodes "act as autonomous agents" [1]. They can establish social relationships like humans and operate based on predefined rules set by human users. These relationships can be a co-owner relationship (devices of the same owner), friendship relationship (devices of owners who are friends), co-location relationship (devices in the same location), and parental relationship (devices produced by the same manufacturer) [12].

3. I comprehensively evaluate the MADM-EBSL framework concerning a subset of aspects of trust evolution, drawing upon our research in Step 1. These aspects are *network connectivity* and *trust problem size* (Chapter 5 includes definitions of these concepts).

4. I extend the MADM-EBSL trust framework with trust-related information to be used in automated trust computations of devices. I investigate the effect of additional trust attributes through experimental evaluations by comparing the trust scores of nodes as computed by the MADM-EBSL to those by the extended framework.

5. I explore trust repair actions for IoT devices and propose a trust repair model to reflect those on trust scores. I present experimental evaluations and discuss the results.

The outline of this dissertation is as follows: Chapter 2 provides a literature review on IoT trust management. Chapter 3 provides an overview of trust in social sciences and discusses implications for IoT networks. Chapter 4 introduces our proposed automated trust computation framework for IoT networks [5], which uses multiple attributes and Subjective Logic [77]. Chapter 5 presents the results of a comprehensive experimental evaluation of the MADM-EBSL, focusing on network connectivity and trust problem size. Chapter 6 extends the MADM-EBSL and compare trust scores computed by the original and the extended frameworks through experiments. Chapter 7 proposes a trust repair framework for IoT based on the MADM-EBSL and trust repair concepts we have distilled in our review of trust in social sciences in [3]. Finally, Chapter 8 concludes the dissertation and outlines future work.

2.0 Background and Related Work

In this chapter, I present a review of literature related to trust management in IoT. As we identified in our review of trust in social sciences [3], trust is *multi-dimensional, context-specific, dynamic,* and involves *uncertainty.* I frame the related work based on these key characteristics and additional perspectives, namely *architecture, trust transitivity,* and *automation capability.* Table 1 summarizes the IoT trust management solutions discussed in this chapter from these seven perspectives.

For the convenience in discussing the related work, and for the rest of the dissertation, I present the definition of two concepts below, which are significant for IoT trust management.

Definition 1 (Trustor node)**.** A node, a.k.a. service requestor node, that typically delegates a task to another IoT node [1]. It may evaluate the service after completion through node ratings or other means.

Definition 2 (Trustee node [1])**.** A node, a.k.a service provider node, that is trusted to provide the service(s) requested by a trustor node. These services are typically beyond the direct control of the trustor.

Note that each IoT node can be a trustor, trustee, or both [25].

2.1 Trust Management Solutions based on Key Trust Properties

2.1.1 Multi-Dimensionality

Trust is a complex and *multi-faceted*, or a *multi-dimensional*, phenomenon that depends on numerous factors as indicated in the social sciences literature [3]. This property also applies to trust in IoT environments. Therefore, multi-dimensionality is crucial for trust management solutions developed for IoT. At the same time, extracting, storing, and updating node attributes [1] and combining the values of them to obtain a single trust score [5] are

Paper	Key trust properties				Architecture	Trust transitivity	Automation capability	Evaluation
	Multiple dimensions	Context-specificity	Dynamism	Uncertainty				
[125]	no	no	yes	no	Distributed	None	None	yes
[126]	no	no	yes	no	Distributed	None	None	yes
[145]	yes	yes	yes	yes	Distributed	global	Semi	yes
[11]	yes	yes	yes	no	Distributed	global	Semi	no
[9]	NI	no	yes	no	Centralized	local	Full	no
[1]	yes	yes	yes	no	Hybrid	local	Semi	yes
[107]	no	yes	yes	yes	Distributed	NI	Semi	yes
[188]	yes	no	yes	yes	Hybrid	local	Semi	yes
[70]	no	no	yes	no	Hybrid	global	Full	yes
[94]	yes	no	yes	yes	Distributed	local	Full	no
[26]	no	no	yes	no	Hybrid	local	None	yes
[154]	yes	yes	yes	no	Centralized	None	None	yes

Table 1: Representative Trust Management Solutions for IoT

challenging. In the sequel, I review previous work concerning the multi-dimensionality of trust computation approaches proposed for IoT.

2.1.1.1 Uni-dimensional Approaches

Mendoza et al. [125] propose an IoT trust management scheme, which computes the trust of an IoT node based on the provision of a service. In the proposed solution, a node can provide multiple services and discover services provided by nearby nodes through announcement packets sent to them. The trust score of a node is incremented if it provides a service to a requesting node on time, decremented otherwise. The proposed solution has a uni-dimensional trust computation approach as there is only a single factor affecting trust scores. Similarly, Mendoza and Kleinschmidt [126] propose a distributed trust management scheme (DTMS) based on a single attribute, reward (penalty) of providing (not providing) a service on time.

Li et al. [107] propose a trust model for IoT services based on context information and reputation. Although multiple context attributes are used for “trust estimation” from given trust scores of an IoT service and the context attribute values under which a trust score is computed, the proposed model is uni-dimensional. The reason is that it only uses the type of trust-related events, i.e., node behaviors such as trusted, distrusted, and uncertain, in the computation of trust scores. The Adaptive IoT Trust protocol [26] proposed for SIoT environments also provides a uni-dimensional solution for trust computations. Only a single factor, user satisfaction on IoT device/service as a binary variable, is used for trust computations, making the proposed solution uni-dimensional.

ReTrust [70] is another trust management solution for IoT with a uni-dimensional trust computation approach, specifically proposed for medical sensor networks. It only uses the number of successful and failed transactions in direct trust computations as decided by a master node in a sub-network based on an outlier detection method applied on reported measurements by sensor nodes, which is a single dimension.

2.1.1.2 Multi-dimensional Approaches

Rafey et al. [145] propose CBSTM, which is a trust model for SIoT. CBSTM is based on social relationships between nodes (relationships such as co-location, co-work, co-owner, parental, etc., as proposed by Atzori et al. [12]) and their owners interacting through social media. In the proposed architecture, nodes and users can form communities based on social relationships and interest similarities. CBSTM includes multiple attributes in trust computations, namely the type of social relationship between two IoT nodes[1], service feedback of a trustor node for a trustee node, recommendations provided by other nodes, and confidence in recommendations by the trustor node.

Asiri and Miri [11] propose a trust and reputation model based on recommender system concepts and Probabilistic Neural Networks (PNN) to classify IoT nodes as trustworthy and malicious. The model utilizes a collaborative filtering technique, a recommender system design, to profile IoT nodes based on node characteristics. These characteristics are considered as factors/attributes contributing to trust computations in the proposed model. They include the number of packets delivered or dropped, transmission rates, battery life, CPU power, available memory, and the sensitivity/severity of transmitted data. To address the cold start problem[2], the proposed model uses average service ratings provided for nodes

[1] Value of this attribute is highest for co-owner relationship and lowest for no social relationship between two nodes. Also higher value of this attribute indicates higher trust.

[2] The cold start problem is a well-known issue for recommender systems, which is the challenge of generating recommendations at the starting phase when there are not enough item ratings by users. The same

other than the node that does not have any rating.

Another multi-dimensional trust model is proposed by Wu and Li [188], for multi-domain RFID systems. The proposed trust model focuses on computing the trust of RFID readers and assumes RFID tags are trusted. It enhances the authentication process of RFID readers by authentication center devices. The trust of RFID readers is computed using interaction events, i.e., node interaction behaviors obtained using watchdog agents implemented by RFID readers. There are three types of node interaction behaviors (discarding, tampering with, and replaying or forging data) at three levels (malicious, malfunctioning, and benign). As the proposed model considers multiple interaction types as factors affecting trust computations, it is a multi-dimensional model.

Aalibagi et al. [1] propose a trust management mechanism for SIoT, which includes a bipartite social network model with two sets of nodes — service provider (trustee) and requestor (trustor) nodes. Trustor nodes can further establish social relationships among themselves and form local networks. The proposed solution also includes a social trust model that uses multiple attributes to estimate trust scores. These attributes are node ratings for completed services by trustee nodes and similarity of trust pattern between (trusting behavior of) trustor nodes, and latent features of the trustor and trustee nodes. Trust pattern similarity is a combination of trustor node similarity and centrality (degree and betweenness centrality). Node similarity is computed using ratings and different metrics, such as the Hellinger distance metric and Bayesian similarity metric. For predicting trust scores, a matrix factorization

concept applies to recommendation/service feedback-based trust computations in IoT for nodes newly joining networks [11, 1]. This is also defined as *initial trust formation* in IoT trust management [3] and applies to trust computation approaches regardless of node recommendations/service feedback. Initial trust formation in IoT networks can be captured by two main types of strategies. These are *naive strategies* and *trust propensity-based strategies* [3]. *Naive* strategies include methods to set the trust of a node to the worst (lowest), best (highest), or a neutral (medium) possible value in a trust scale [3]. *Trust propensity-based* strategies utilize the tendency of an entity to trust others [3] to set an initial trust value. Initial trust formation is a well-explored research area for IoT networks. For instance, as a naive strategy, setting the initial trust of an IoT node to the worst (zero) [126] and a neutral (medium) value (0.5 in a $[0, 1]$ scale of trust) [145] have been proposed. As trust propensity-based strategies, researchers utilize the social networking paradigm in IoT, a.k.a Social IoT (SIoT, (section 1.0.2, Page 4)). For example, node centrality and social relationships between IoT nodes (e.g., co-owner relationship, friendship relationship, co-location relationship, and parental relationship) have been used to set initial trust levels [133]. Latent features of SIoT nodes extracted from a rating matrix and a matrix factorization approach have been utilized to deal with the cold-start problem and data sparsity [1]. These attributes reflect the trust propensity of nodes, i.e., generalized trust towards other nodes based on the properties of node relationships. As another propensity-based strategy, researchers have proposed to use the average service rating of a trustor node for the nodes it has previously interacted with to set the initial rating for the first encounter with a node.

technique is applied, which also addresses the cold-start/sparsity problem[2] for the trust rating matrix. Latent node features are extracted by the matrix factorization model.

Konwar et al. [94] propose a multi-dimensional trust model for wireless mesh networks based on the Multiple Criteria Decision Making (MCDM) approach, specifically the Technique for Order Preference by Similarity to Ideal Solution (TOPSIS) model [76]. Selected criteria are the probability that an agent will perform a particular action (packet forwarding and recommendation exchange), the number of packets to be forwarded, the number of packets successfully forwarded, and the Delivery Ratio Efficiency (DRE) of the agent. One limitation of this model is that each criterion in MCDM formulation has equal importance and thus equal weight. The total trust score is computed by simply adding individual and recommendation trust values without weighing them. This results in a trust value greater than the allowed limits set in the model.

Saied et al. [154] propose a trust management system for IoT with multi-dimensional trust computations. They compute the trust level of an IoT node based on recommendations from nodes that consume services provided by this node. As recommendations are further evaluated by the trust management system based on multiple contextual variables, the proposed solution has a multi-dimensional trust computation approach. These variables are the type of service consumed, the time, and the capabilities of the node consuming the service (resource capacity, memory/battery level, etc.).

2.1.2 Context-Specificity

Context has a fundamental role in assessing trust in social relationships [3] and IoT environments [189, 107, 8]. Context refers to information describing the situation of an entity, such as attributes of a trustor, trustee, and their environment [176]. Examples include location, time, velocity, and device capability for a smart-wildlife application [107]. Trust is context-specific [109, 107, 106] as a trustor may trust a trustee under a given context but may decide not to trust under others [109, 107]. For instance, a base station may provide fixed devices with more reliable service/stable connections than mobile devices [107]. In the sequel, I overview trust management solutions for IoT with a context-based trust computation approach.

Saied et al. [154] propose a context-aware trust management system for IoT. Here, the underlying design principle is that an IoT node provisions multiple services and the trust level for each of them should be evaluated separately. Thus, the trust towards a node and recommendations provided by a node are computed based on contextual variables such as the type of the service consumed, the time and the capabilities (resource capacity, ageing etc) of the node consuming the service. The CBSTM proposed in [145] is a context-based social trust model for IoT. CBSTM considers the context of a transaction performed by an IoT node in trust computations for that node. Context is considered for both direct and indirect trust, i.e., there are distinct trust values between each device for each specific context. Yet, there is no further discussion on possible contextual attributes/parameters used in CBSTM.

The IoT trust model proposed by Asiri and Miri [11] can be considered as a context-based model as it accommodates the data sensitivity of transmitted data. That is, it makes trust decisions by checking the severity of the data to be exchanged in a transaction between two IoT nodes. For example, the trust of nodes for the exchange of weather data and patient records are evaluated differently. If the data is not sensitive and publicly available, it allows a node transaction by comparing the service ratings of a service provider node to a threshold value. If the data is sensitive, it facilitates more complex trust computations based on neural network algorithms. The proposed solution is effective for heterogeneity of devices by considering different data sensitivity requirements and their type and functionality. Yet, it is a distributed trust model where each node stores trust rating matrix and applies ML algorithms on it. Thus, computation and storage overhead brought by ML algorithms may render it ineffective for resource-constrained nodes.

In the trust management scheme proposed by Aalibagi et al. [1], context is defined as the characteristics/type of a service provided by a trustor node and a factor that decides node service feedbacks. That is, a service requestor node could possibly evaluate the service provider node depending on the characteristics of the provided service. The proposed solution uses context information such that a separate bi-partite graph of IoT nodes (two node sets: service requestor (trustor) and provider (trustee)) is constructed for each service type, which is limited in number.

Another context-based IoT trust model is proposed by [107]. The proposed model does not utilize context attributes directly in trust score computations. Yet, it constructs a context vector with multiple context attributes (e.g., location, time, velocity, capability) to represent the state of a service provided by an IoT node. This context vector is stored with the trust score measured for a service provider node. Later, to estimate trust score of another service provided by this device, a trustor IoT node compares the current context vector of the trustee node to previous context vectors. Hence, based on the most similar (using Euclidean distance between context vectors) context, a new trust score is estimated.

2.1.3 Dynamism

Trust is a dynamic concept that evolves over time through interactions [106]. This holds for both trust in human relationships [3] and trust in IoT [11]. Hence, trust management frameworks that consider dynamic trust changes in trust computations are needed for IoT environments [9]. Within the context of IoT, *dynamism* of a trust computation approach corresponds to the adaptive adjustment of trust scores concerning the behavior of nodes [145]. As seen from Table 1, all of the IoT trust management solutions reviewed in this chapter are dynamic. In the following, I present a subset of these solutions as examples of how dynamism can be reflected on trust computations in IoT.

The CBSTM proposed by Rafey et al. [145] is a dynamic trust model as it adapts to changes in behavior of nodes, i.e., trust values are dynamically updated after each transaction. The trust model proposed by Asiri et al. [11] is another dynamic model. It is considered dynamic as it also updates trust score through node ratings and recommendations upon completion of transactions. Another dynamic trust model is proposed by Aalibagi et al. [1]. The proposed social trust model is dynamic as it takes into account node ratings that are provided by service requestor nodes and updated after receiving a service every time.

The central IoT trust management mechanism in [9] comprises an architecture in which trust messages are exchanged between IoT nodes in a cluster and the master node of that cluster. These include *receive* and *send* messages that are used by the central node (cluster nodes) to request (send) updated trust values. Thus, this architecture sets forth a dynamic trust computation approach.

The context-based IoT trust model proposed in [107] is also dynamic as it evaluates trust based on trust-related events, i.e., node behaviors as trusted, distrusted, and uncertain activities. Wu and Li [188] propose a trust model for RFID systems with multiple management domains. This model is dynamic because it includes node interaction behaviors (discarding, tampering with, and replaying or forging data) and feedback records for completed interactions to compute trust scores. The Adaptive IoT Trust protocol proposed in [26] is another dynamic trust computation approach. The proposed solution is dynamic as it updates trust scores of IoT devices through user feedback upon receiving services or user service recommendations before receiving a service. It also updates direct trust scores periodically in a predefined time interval using an exponential decay function, even if there are no transactions.

2.1.4 Uncertainty

Trust becomes more relevant in situations that are characterized by uncertainty [3, 106]. Uncertainties may occur due to risks that cannot be reduced for a given context or cannot be calculated due to imperfect information about a trustee and its behaviors [3, 106]. Given that uncertainty is needed for trust to become relevant, it is crucial to consider uncertainty in trust computations for IoT. There are several approaches to deal with uncertainty in trust computations [106], including Dempster-Shafer (D-S) Theory [162], Fuzzy Logic [91], and Subjective Logic [77]. In the following, I overview IoT trust management solutions that consider the notion of uncertainty in trust computations.

CBSTM proposed in [145] includes the confidence of a trustor node in recommendations provided by other nodes. The confidence attribute used in this model resembles uncertainty in trust recommendations, so we can consider that CBSTM captures uncertainty from this perspective. In the IoT trust management system proposed by Li et al. [107], trust evaluations are performed based on the entropy of trust-related events, i.e., behavior of nodes as trusted, distrusted, and uncertain activities. Thus, this model is one of the few IoT trust models that considers uncertainty in trust scores. Similarly, Konwar et al. [94] propose a trust model that computes trust scores by entropy and approaches trust as a function of uncertainty.

The trust model proposed by Wu and Li [188] is another model that truly takes uncertainty into account for trust computations. It addresses trust uncertainty by using Dempster-Shafer (D-S) evidence theory as D-S theory can express uncertainty. Our previously proposed IoT trust framework in [5] (see Chapter 4 for details) also accounts for uncertainty in trust values through Evidence-Based Subjective Logic (EBSL) [167].

2.2 Trust Management Solutions based on Architecture

Trust management solutions for IoT can be characterized based on their architecture, namely *centralized* (e.g. [154]), *decentralized* (e.g. [125, 26, 126, 107]), and *hybrid* schemes [189, 40]. Centralized schemes include a central authority for computing, storing, and sharing trust levels. On the other hand, decentralized schemes do not contain a central trust authority responsible for trust management/trust computations [145, 11, 107]. A distributed trust management scheme is a special type of a decentralized scheme. In a decentralized scheme, there can be pre-trusted nodes, which are typically more powerful IoT nodes, that are responsible for trust management in their virtual cluster of IoT nodes, such as in [145, 11]. In a distributed scheme, IoT nodes are responsible of maintaining trust data and computing trust scores on their own [145, 11]. It should be noted that IoT nodes in a distributed trust management scheme are either capable of computing and storing trust data or not, so there is no delegation of these tasks to a different entity in the network [145]. Hybrid schemes have an architecture choice that is in between centralized and decentralized [125].

Each type of architecture has its own advantages and limitations. Centralized trust management schemes offer an advantage of comprehensive understanding of the whole network and an improved accuracy of trust computations [107]. Yet, they have the drawback of *single point of failure* as they compute and store trust data on a remote central unit [145, 1], which in turn increases the likelihood of compromising the system [11, 1]. Deploying a central node for measuring and managing trust may lead to possible node delays [11] and may restrict the scalability of trust management for large IoT networks [1]. In contrast, distributed schemes are not prone to single point of failure, so they provide better availability [11]. Yet, decentralized schemes which are not fully decentralized and distribute trust computations on more powerful pre-trusted nodes, may still be prone to the single point of failure for node clusters of unavailable pre-trusted nodes. They also may be prone to a malicious pre-trusted node attack (see Section 2.5). Fully decentralized schemes suffer from trust computation burden put on lightweight nodes. These nodes are typically resource-constrained and cannot accommodate complex security solutions due to energy and memory consumption issues [107]. As discussed above, a node is either capable or not capable of trust computations in a fully decentralized scheme. This leads to interruptions in trust computations and the unavailability of trust data.

2.2.1 Centralized IoT Trust Management Solutions

Saied et al. [154] propose a trust management system for IoT with a centralized architecture. The underlying design principle is that an IoT node provisions multiple services and the trust level for each of them should be evaluated separately. A central trust manager computes the trust level of an IoT node based on recommendations from nodes which consume services provided by this node, in the form of evaluation reports on service quality. The trust manager also evaluates recommendations based on contextual variables (the type of the service consumed, the time, and the capabilities of the node consuming the service (resource capacity, memory/battery level, etc.)).

Alshehri and Hussain [9] propose a centralized mechanism for trust management in IoT. Although not detailing on a technical solution for trust computations, they propose an architecture and elaborate on the components, i.e., modules, design patterns, and methods to be used by network elements. In this architecture, there is a central trust management component named Super Node (SN). Nodes are grouped into clusters (Cluster Nodes (CN)). In each cluster, there is a local trust manager named Master Node (MN). MNs have a local trust repository to store trust values of CNs in a cluster. Trust messages are exchanged between CNs, MNs, and SNs to communicate trust value of CNs. Trust values are set from a pre-defined set of discrete values in [0,1], zero representing distrust, 0.5 being neutral and 1 being complete trust. SN has a central trust repository to store trust values of all MNs and CNs, and the network topology. It can monitor communications among MNs and CNs and exchanged trust data. SN is remotely accessible to any IoT application through an Application Programming Interface (API) so that applications can query and share the trust value of IoT nodes.

2.2.2 Decentralized IoT Trust Management Solutions

Based on the multi-service paradigm of the trust scheme proposed by Saied et al. [154], Mendoza et al. [125] propose a distributed trust management solution for IoT. Their proposed scheme relies on direct observations and direct communication among nodes for trust evaluation. Each node has a trust score incremented (or decremented) if it provides (or does not provide) a service to its neighbor node which requests the service. DTMS [126] is another distributed trust management scheme for IoT, which is based on direct observation of nodes and the evaluation of services provided by them. Similarly, in DTMS, a node is rewarded for the service provided in time and penalized otherwise. This reward/penalty information is recorded by the node requesting the service. These trust computation approaches that are based only on timely service provision have a drawback. The trust value reduces if a node only does not provide the requested service. A node may misuse this by providing a requested measurement, but with a bogus value. Also, as trust computations are performed by IoT nodes and trust scores are recorded by them, resource constraints may be an issue. The trust management scheme proposed in [94] has also a distributed architecture. Trust computations are performed by nodes without delegation to any other network entity. Yet, the authors do not discuss about or propose a solution for the issues related to distributed architecture for trust management, i.e., overhead of trust computations on resource-constrained nodes.

Wang et al. [185] propose a trust model, LogitTrust, for service-oriented MANETs. Trust is defined as the probability of a service provider (SP) to provide a satisfactory service in response to a request from a service requestor. They use logit regression to predict this probability with respect to behavior patterns of an SP node while providing a service. They define behavior pattern factors as energy-sensitivity (# neighbors sharing the channel), capability-limitation (packet traffic to SP), and profit-awareness (price for the service). They show the superior performance of LogitTrust to Bayesian inference-based trust model with belief discounting, in terms of trust accuracy, false positive rate, and resiliency against bad-mouthing and ballot-stuffing attacks. The limitation of LogitTrust for IoT environments is that nodes store history of service satisfaction and contextual variables (for each interaction) and send it to any node upon request. As the time progresses, the size of the historical records will increase and a limited IoT node may not be able to store them.

CBSTM [145] is a distributed context-based social trust management model for IoT. As it is distributed, each node computes and stored trust values for other nodes and there is no central authority. Even though there is a claim that there is no central authority and each node calculates and maintains and trust values itself "with no reliance on pre-trusted peers" [145], trust calculation and storage are deferred by more constrained nodes (e.g., RFID tags) to less constrained nodes (e.g., mobile phones). This still may not alleviate the single point of failure problem of centralized architectures. Authors propose a storage management strategy for constrained nodes such that most recent and the higher trust values would be stored to save space. Yet, this may be misused by malicious nodes by providing higher trust recommendations for their cooperators and increasing their trust values.

Asiri et al. [11] propose a trust model with a completely distributed architecture, i.e., each IoT node is responsible for storing trust data and computing the trust of a node of interest. These include storing a trust rating matrix that holds ratings from each service consumer node towards a direct neighbor node providing a service after the completion of a transaction, and computing trust scores using a ML (PNN) model. Nodes are clustered into virtual clusters concerning the similarity of node profiles, which comprise node characteristics such as functionality, type, transmission rates, the sensitivity of the data transmitted among nodes, etc. These similarity scores are used in a PNN as input parameters for training of the network. There are pre-trusted alpha nodes in each cluster that are responsible for profiling the nodes at the startup phase and collecting trust-related data constantly. These nodes are stronger concerning their storage, computation, and energy resources, such as control hubs connected to a power source. Even though claimed as fully-distributed, pre-trusted alpha nodes play the role of a central component in the architecture. They store copy of trust rating matrix within each cluster of nodes and update it. Hence, these nodes could still be a single point of failure for the trust management operations.

Wu and Li [188] propose a hierarchical two-layer trust model for RFID systems with multiple management domains. The bottom "RFID reader trust layer" manages trust of RFID readers, whereas the top "authentication center layer" evaluates the trust of authentication centers (i.e., a pre-selected device responsible for trust computations in a domain) in each domain. RFID trust layer is distributed such that each RFID reader in a domain records node interaction behavior of its neighbor nodes and computes "local trust scores" using these records. Also, there is an authentication center in each network management domain that computes "global trust score" of an RFID reader by combining "local trust scores" reported by its neighbors. The authentication center trust layer has a centralized trust evaluation scheme such that the trust of authentication center device in each domain is evaluated by a central trust manager device (named administration center in the paper).

The proposed trust management solution in [1] is described as being "locally centralized and globally distributed". That is, SIoT nodes are members of local physical groups (e.g., nodes of a smart home) that includes both lightweight (e.g., smart lights and smart thermostat) and at least one more powerful node (e.g., smart TV). Within each group, a powerful node is selected as a central node. This node collects node ratings from other nodes within a group and from external groups, performs trust computations, and provides trust scores to requesting nodes.

Li et al. [107] propose a trust model and claim that trust management is distributed because the model is designed to accommodate requirements of multiple trust domains. Yet, there is no further discussion in the paper about its architecture/system model.

2.2.3 Hybrid IoT Trust Management Solutions

ReTrust proposed in [70] is an example of trust management schemes with a hybrid architecture. It is a hybrid scheme because trust computations are performed by a master node in each network domain, i.e., a physical grouping of sensor nodes in a medical sensor network deployed on a patient's body, rather than by each IoT node. Another hybrid solution is proposed by Chen et al. [26] for SIoT environments. Despite the authors discussing the proposed solution as having a distributed architecture, they include "high-end devices", such as smartphones and laptops, to store and compute trust scores within each physical group of sensors belonging an IoT user. Thus, I argue that this is a hybrid architecture with a centralized trust management within each group of sensors and distributed trust management across different networks.

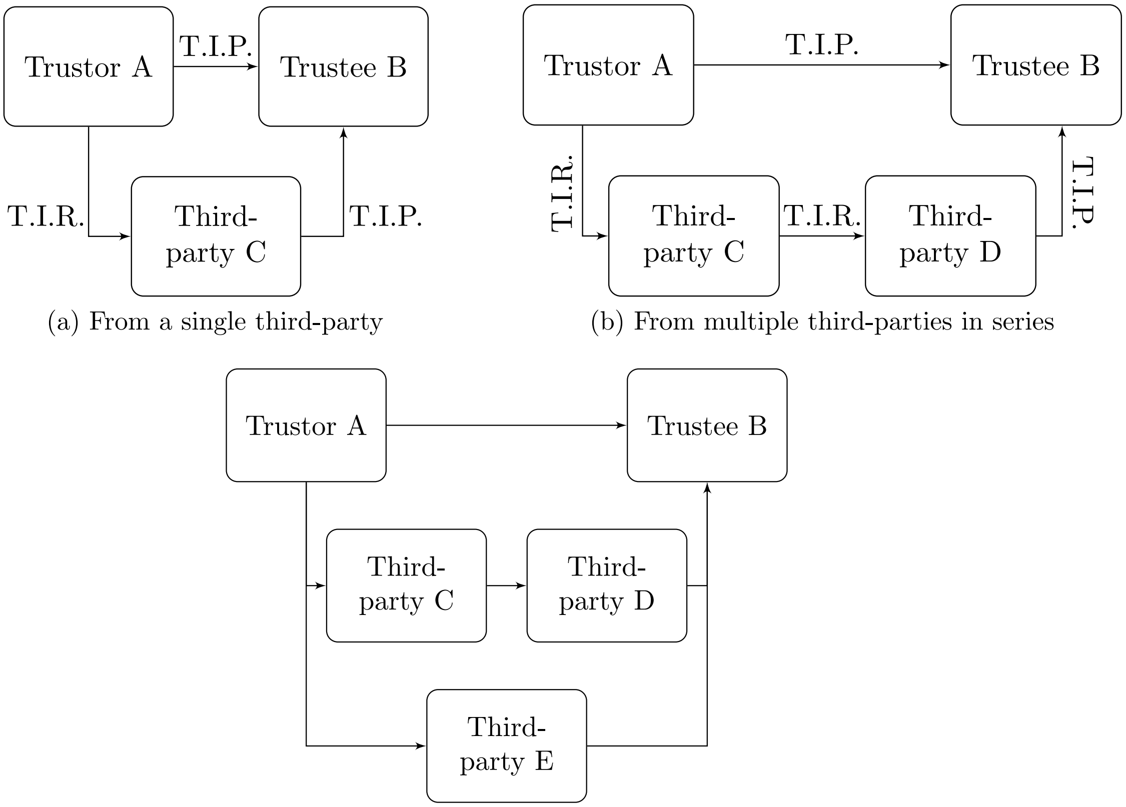
Truong et al. [179] propose an architecture of a trust platform as a service (TaaS) for trust management in SIoT. They use a semi-centralized approach to alleviate the drawbacks of centralized and distributed architectures. Their trust model imitates human information processing and incorporates recommendation, reputation, and knowledge as trust metrics.

2.3 Trust Management Solutions based on Trust Transitivity

In this section, first, I briefly explain trust transitivity through closely related concepts, such as recommendations, direct vs. indirect trust, and functional vs. referral trust. Then, I overview existing IoT trust management solutions based on trust transtivity.

Direct and *indirect* trust are the two ways to measure trust that are frequently cited in the IoT trust literature [40, 107, 106, 145]. The prior corresponds to the trust between two nodes that are directly connected, whereas the latter is the trust between indirectly connected nodes [40]. A trustor may combine direct and indirect trust to form trust in a trustee [106]. Direct trust, a.k.a *trust in performance* [106], is calculated based on direct interactions and observations by a trustor node of a trustee node [145, 106]. These observations provide evidence about the characteristics of a trustee to make trust decisions, such as performance, ability, and morality [106]. When there is no direct interaction, direct trust could be set to a default initial value or indirect interactions could be utilized [145].

For indirect trust, a trustor resorts to information from third parties rather than direct interactions with the trustee [106]. In IoT, nodes can obtain indirect trust through recom-



(a) From a single third-party

(b) From multiple third-parties in series

(c) From multiple third-parties in parallel

Figure 1: Indirect trust types (T.I.P: trust in performance, T.I.R: trust in relationship)

mendations provided by other nodes for the trustee node [145]. Indirect trust, a.k.a *trust in relationship* [106], can be obtained in following three ways in a trust management system: *i*) from a single third-party, *ii*) from multiple third-parties in series, and *iii*) from multiple third-parties in parallel [106]. Fig. 1 illustrates direct and indirect trust types. In all cases, the arrow from node A (trustor) to node B (trustee) denotes the direct trust of trustor A on trustee B, and is labelled as T.I.P. (trust in performance). Fig. 1a represents the case that a trustor resorts to a single third-party (which can be considered as a trust authority [106]) to form the trust in a trustee. The arrow from node A to node C (third-party C) represents T.I.R. (trust in relationship), i.e., the trust of node A on node C for its direct trust belief on node B [106]. Indirect trust of node A on node B is derived from the path on which node C resides.

Fig. 1b represents the case that a trustor resorts to multiple third-parties in series, a.k.a.

a trust chain [106], to form indirect trust. In this case, indirect trust is the combination of T.I.R.s and T.I.P. on the path from node A to B through nodes C and D. Fig. 1c represents the case that a trustor resorts to multiple third-parties in parallel to form trust. In this case, there are two paths formed by third-parties in parallel. The trustor aggregates opinions from multiple sources, a.k.a. *trust aggregation* [106]. Trust aggregation approaches include belief theory, Subjective Logic [77], and weighted sum [106]. All types of indirect trust involves *trust propagation* [106], or *trust transitivity*, such that trust propagates from one party to another until reaching to the trustor. As a result, to compute the total indirect trust of a trustor, trust *propagation* and *aggregation* can be utilized [40, 106].

Indirect trust is considered by some researchers (e.g., [145, 11]) to reflect the *reputation* of an IoT node in a community. *Reputation* is "the opinion of an entity about another" [145]. According to Jøsang et al. [78], there is a distinction between reputation and trust. They define reputation as what other people say or believe about the characteristics or standing of a person or entity. From this perspective, a person could be trusted even if they have a bad reputation, or the reverse. Similarly in IoT context, reputation is a value derived from the opinion of IoT nodes [11], which form a community, and reflects the indirect trust/recommendations in trust computations.

As aforementioned, indirect trust includes *trust transitivity* notion. I categorize trust computation approaches based on trust transitivity. Trust computation approaches that take into account indirect trust and trust transitivity are *transitive trust computation approaches*. Conversely, if a trust computation approach does not capture indirect trust but only direct trust, then it is a *non-transitive trust computation approach*. In a transitive trust computation approach, trust is allowed to be transferred from one node to another along paths in a network through recommendations. Recommendations could be *subjective* or *objective*. When nodes share their own view of trust for other nodes, such as through their owners in a social networking-based IoT paradigm (SIoT), recommendations are considered subjective, such as in [145, 11, 1]. When recommendations are derived from objective information or interaction parameters, they are considered as an objective recommendation, such as in [5].

Transitive trust computation approaches can be further categorized as *global* and *local*.

In a global approach, trust computations are performed by a central trust authority with the knowledge of the topology of the whole network using all existing links and paths among trustor and trustee nodes. In a local trust computation approach, typically there is no dependence on a remote central node [1], but there can still be trust transitivity. As such, trust ratings/recommendations may be shared among nodes in a local physical group of IoT nodes and trust can propagate along paths in the local network. Local trust computation approaches have been developed for addressing the *network sparsity* challenge that global approaches suffer from. Network sparsity challenge corresponds to a significant amount of missing links among nodes in a network as most of the nodes only interact with a small portion of the nodes. This poses a challenge for trust computations because trust scores may not be properly predicted due to the lack of data [1]. For instance, Wu and Li [188] observed that trust prediction accuracy drops in sparse networks. Local trust computation approaches address this challenge by considering immediate neighbors as most IoT nodes typically communicate with their neighbor nodes.

Next, I overview transitive and non-transitive trust computation approaches.

2.3.1 Transitive Schemes

He et al.[70] propose *ReTrust* trust management scheme for medical sensor networks with direct and indirect trust components. They suggest using the number of “successful” and “failed” interactions between two nodes to compute direct trust. They consider historical trust values to compute the current trust of a node, using “sliding window” concept.

CBSTM [145] is a context-based social trust model for IoT. It incorporates direct observations of a trustor node and indirect recommendations from other nodes to calculate a trust value of a transaction performed with a trustee node. As CBSTM includes recommendations as a way of transferring trust values among nodes, it can be considered as a transitive model. Direct trust is computed using type of social relationship between nodes (just used for setting the initial trust value, i.e., for $t = 0$) and subjective feedback of the trustor node for a service provided by the trustee node. Indirect trust is a weighted combination of recommendations by all other nodes in the network and confidence of trustor node in these recommendations.

Another transitive trust model for IoT is proposed in [11]. In this model, a service consumer node provides a rating for a service provider node after the completion of a transaction. These ratings are provided to a requesting trustor node as recommendations. Recommendations for a service provider node is weighted across nodes providing rates for this node. The proposed solution computes indirect trust between nodes that are not immediate neighbors through paths with multiple hops. Although a trust propagation or aggregation method is not discussed by authors, there is a proposed trust formula that computes trust between two nodes based on recommendations for these nodes and recommendation weight of each node. Recommendation weights are computed based on the quality of a recommendation (the accuracy of a recommendation based on past ratings of a node) and virtual clustering of nodes (based on similarity of node profiles generated using device functionalities and types). Recommendations from the same cluster are weighted more.

ReTrust [70] provides a transitive trust computation solution. In ReTrust, indirect trust is computed by multiplying the trust recommendation of a node on the path, with the trust of the node requesting recommendation on the recommender node for making correct recommendations. ReTrust has global transitivity because trust propagation is considered for computing indirect trust within the whole network, between two nodes that are not immediate neighbors and with a distance more than one hop.

The solution proposed in [94] for wireless mesh networks has a transitive and local trust model. In this model, there are two components; "individual trust" assigned by a node to its neighbors based on node behavior and "recommendation trust" broadcasted by the neighbors of a trustee node. As the immediate neighbors are taken into account for recommendations, trust is only transferred locally in the network. Another local-transitive trust management solution for IoT is proposed by Chen et al. [26]. The proposed solution has a transitive trust computation approach because it employs user recommendations towards IoT devices as a means of indirect trust. The type of transitivity is local because only users and devices with interactions/common interests/social relationships share recommendations. There is no global network information nor trust propagation.

2.3.2 Non-Transitive Schemes

The trust management scheme proposed in [125] (and DTMS [126], which utilizes the proposed solution in [125]) is a non-transitive trust management scheme proposed for IoT. Being a non-transitive trust management scheme, DTMS considers only direct observations for trust computations. More precisely, trust value of a node is computed based on rewards/penalties for providing/not providing requested services on time. Initially, when there is no direct observation for a trustee node, a trustor node sets the trust value for the trustee node to zero.

The central IoT trust management system proposed in [154] can be considered non-transitive as trust computations do not include trust transitivity or trust propagation. Nodes provide recommendations for the nodes that they receive a service in terms of service quality reports. Yet, these reports are not shared among nodes but are directly sent to a central trust manager. Thus, trust scores are not discounted as trust does not propagate among entities along paths on a network. Alshehri et al. [9] also proposed a central non-transitive trust management solution. In the proposed solution, IoT nodes do not share trust recommendations for other nodes. They only store their own trust values and send it to the master node in its cluster and the central super node in the architecture.

Wu and Li[188] propose a trust management scheme that is local and non-transitive as each node has a "local trust value" computed by its immediate neighbors in the same network management domain without considering trust propagation. Even though there is a "global trust value" of a node that is computed by aggregating local trust values reported by its neighbors (using Dempster-Shafer (D-S) evidence theory and Dempster knowledge rule), the scheme is not global as it does not consider non-neighbor nodes in a network domain.

The trust management mechanism proposed by Aalibagi et al. [1] is local and non-transitive. SIoT nodes are grouped into local physical clusters, such as smart home group, with a central node responsible for trust computations. Nodes in a local group can rate each other but trust is not propagated along paths in these groups. Instead, if there is no direct interaction between nodes, the central node predicts the missing rating based on a matrix factorization approach.

2.4 Trust Management Solutions based on Automation Capability

Providing network services in an automated and programmable way is critical as otherwise the time required for providing a service would be high [83]. Automation is also essential for IoT networks, given the ever-growing large scale of them. Considering trust management as a service for IoT environments, trust computations should be automated, as well. I argue that the *automation capability* of trust computation solutions can decided based on trust *recommendation* and *feedback* mechanisms.

Recommendation and *feedback* have been mechanisms frequently included in trust computation solutions (such as in [145]) for IoT networks in order to compute trust towards IoT nodes and their services or reported measurements.

Recommendations are requested by a *trustor* IoT node (or on behalf of it by a central trust management component in a central architecture) from other nodes for the services/measurements provided by a *trustee* IoT node. The nodes which have had direct interaction with the trustee node before in any context send their recommendation to the trustor node [145]. Based on the design of a trust computation solution, recommendations could be requested in two different cases. They could be requested only in case the trustor node does not have any direct interaction with a trustee node, but needs to be able to form a trust opinion for the trustee node. They could also be requested in case the trustee node has direct interaction before with the trustee node, but wants to complement its own direct trust observations with indirect evidence coming from other nodes in the network.

Feedback is the performance evaluation of an IoT node after the completion of a transaction or the use of a service provided by another node [145]. In other words, it is the evaluation of each transaction determined by a node itself [145], whereas a recommendation is the evaluation by other nodes. Hence, feedback is relevant to direct trust as a node keeps it for itself to compute direct trust and a recommendation is related to indirect trust as a node sends it to a requesting trustor node for indirect trust computations. A feedback could also be *subjective* or *objective* like a recommendation, as explained in Section 2.3. If the feedback for a transaction of an IoT node is provided by a human, it can be considered subjective, whereas it can be considered objective if computed based on objective information from the direct interaction between nodes, such as time delay or the deviation of a provided measure from a range.

Trust computation solutions that rely on recommendation or feedback mechanism may not have *automation capability*, i.e., they may require human intervention in the trust computation process. This is because nodes need to provide their recommendation based on some logic, which may require input from users. For example, in SIoT, devices and their owners form a social network and establish social relationships [12]. In an SIoT scenario, a device owner could provide its trust recommendation for a device owned by another user. On the other hand, a recommendation or feedback mechanism could be designed such that it does not require human intervention. For example, IoT nodes could send the most recent direct trust score they computed for the trustee node to the requesting node as their recommendation, such as in [145]. In this scenario, if direct trust computation also does not require human involvement, trust computations could be considered to have full automation capability.

We can relate the categorization of recommendations and feedback based on subjectivity to the automation capability of a trust computation solution. If a trust computation solution purely relies on *subjective* recommendation or feedback mechanism, it can be considered to have *no automation capability.* Conversely, if a trust computation solution purely relies on *objective* recommendation or feedback mechanism, it can be considered to have *full automation capability.* Anything in between, i.e., using partly objective and subjective recommendation or feedback mechanism, could be considered to have *semi-automation capability.*

2.4.1 Trust Management Solutions with No Automation Capability

Saied et al. [154] propose a trust management system for IoT. The proposed solution includes a trust model that computes trust scores for service provider nodes based on reports (a single number from the set $-1, 0, 1$) provided for evaluating the quality of provided services. These reports include subjective evaluation of a received service, i.e., “whether an assigned task is performed properly or not”. There is no method presented for how service quality reports are obtained from IoT nodes. It may be the case that it requires manual interventions from users during active operation of the system. There is no other component of trust computations that is based on automatically gathered trust-related information. Thus, the proposed solution may not have an automation capability.

The trust management schemes proposed by Mendoza et al. [125, 126] may be classified as solutions with no automation capability. These two proposed solutions use only direct observations for trust computations, and there is no recommendation mechanism. Direct observation is the feedback provided by an immediate neighbor of a service provider node after the service is completed. The authors formulate this as a reward point if the service is "successfully provided" and a punishment point if it is not successful. As the authors do not elaborate on how the success rate is decided, it is unclear if obtaining this information requires human intervention.

Chen et al. [26] propose a trust management protocol for SOA-based SIoT systems. They assume that IoT users are connected through social networks and relationships, so do the IoT devices they own. Trust comprises two components; direct trust from user satisfaction experiences on services provided by IoT devices and recommendations from other users. As both user feedback and recommendations depend on subjective information that requires human intervention, the proposed trust computation solution cannot be automated. In other words, trust scores of IoT devices cannot be computed if no human is involved in the process.

2.4.2 Trust Management Solutions with Semi-Automation Capability

The CBSTM model proposed in [145] could be considered as a semi-automated trust model. CBSTM computes trust of a node by combining direct observations and indirect recommendations. Direct observations correspond to one-to-one interaction between a trustor and trustee node. As a result of these interactions, a trustor node rates complete transactions with a trustee and stores it internally. These ratings are considered as subjective feedback by Rafey et al. [145]. As there is no further detail about how nodes rate each other and as CBSTM is proposed for SIoT, it is likely that there is a need for human intervention to collect feedback (maybe from device owners or users). Yet, recommendations in CBSTM are trust values computed by other nodes for the trustee node and sent to the trustor node. Hence, there is no obvious need for human intervention. Overall, CBSTM combines automated ways and human user interactions for trust computations, so it is a semi-automated trust scheme.

We consider the IoT trust model proposed in [11] as a semi-automated. The reason is that trust computations include node ratings for completed transactions, which could require human intervention, and node recommendations that could be automated. Similar to CBSTM [145], there is no further detail about how node ratings are obtained, i.e., they could be from device owners or users, which will require human intervention. Recommendations are node ratings sent by direct neighbors of a trustor node. As long as neighbor nodes store/-compute ratings for a trustee node, they send it to a requesting node, so recommendation process can be automated.

Wu and Li [188] propose a two-layer trust model for RFID systems, which comprises RFID reader and authentication center trust layers. This model can be deemed as semi-automated. The reason is that the part of trust computations in the RFID reader trust layer cannot be automated. That is, the authors propose two different trust computation approaches for this layer. The first trust evaluation approach can be automated as it computes trust scores through automated means, i.e., using interaction records collected through watchdog agents implemented by RFID readers. The second approach is based on the verification of interaction proofs. This approach is used when it is not possible to obtain node interaction behaviors through watchdogs due to limited communication range or if individual network domains (local networks) are sparse (because in the first proposed trust evaluation method, interactions only with neighbor nodes are used for trust computations). In this second type of trust evaluation method, a trustor node (RFID tag) provides a feedback (zero = not satisfied, 1 = satisfied) for a trustee node (RFID reader) after an interaction is completed. A feedback is evaluated by a third, intermediate node (RFID reader) to verify and proof that trustor feedback is not tampered with (using a hash function and certificates) while being transmitted to a local trust manager. The second trust evaluation approach may not be automated as feedback mechanism requires subjective evaluations (zero= not satisfied, 1 = satisfied) and may involve human in the process.

The proposed social trust model for SIoT in [1] could also be considered semi-automated. It uses both service ratings and node recommendations. Service ratings, i.e., feedback, are provided after completed services and reflect the experience of a trustor node with a trustee node. As the proposed solutions above, these ratings are assumed to be provided and no further discussion about how they are obtained is presented. Thus, they may require intervention from device owners. On the other hand, node recommendations do not require human intervention. They are service ratings sent by neighboring nodes of the trustor to the central IoT node in each local node cluster. As a result, this trust model can be partially-automated assuming that service ratings are provided.

Another semi-automated trust management mechanism is proposed in [107]. The proposed mechanism considers direct and indirect trust components in trust computation. Yet, it only uses indirect trust (recommendations) in case there is not enough direct observations for decision making. As the authors state that direct observations are subjective, we consider that direct trust computation process cannot be automated. Node recommendations are objective, i.e., obtained from direct trust scores for a trustee node sent by other nodes, and do not require human intervention. Overall, the proposed trust management mechanism is semi-automated as it combines subjective direct observations with objective indirect recommendations.

2.4.3 Trust Management Solutions with Full Automation Capability

ReTrust [70] is a trust management scheme with full automation capability. It can be fully automated because it does not require subjective feedback nor recommendations from nodes/users. Indeed, instead of feedback, the success and failure rate of interactions between two nodes are used to compute direct trust. This is performed by a master node in a local group of IoT nodes, based on an outlier detection method on readings reported by nodes. Recommendations used in indirect trust computations are objective as they are direct trust scores provided by recommender nodes computed for a trustee node. There is no need for human intervention for this, as well.

The trust management scheme proposed in [94] could be automated fully. The reason is two-fold. First, there is no feedback mechanism, so there is no manual intervention needed for collecting feedback. Second, recommendations are objective and do not require any human intervention, as well. Similar to the existing solutions discussed above, a recommendation is a direct trust score provided by a recommender node.

2.5 Attacks on Trust Management Schemes for IoT

As aforementioned, IoT has unique features that make it vulnerable to security threats and attacks. Attacks to IoT environments are various. For example, nodes can easily join a network, and malicious nodes can masquerade as benign nodes [11]. Node compromise is a type of physical attacks on IoT [11]. IoT is also prone to attacks that apply to traditional networks. For example, an attacker can listen to and analyze network traffic, selectively drop packets, change their sequence, and disrupt routing in a network [11].

There are specific attacks that pose threats to trust management schemes for IoT. In this section, I overview trust-related attacks for IoT environments.

Performance-related attacks: Attacks in this category relate to actions by individual malicious nodes. This group of attacks is typically about the under-performance of malicious IoT nodes or the actions they take to conceal their behaviors from a trust management scheme and boost their trust.

- *Individual malicious nodes* [44, 145]: These are the nodes that intentionally affect the confidentiality, integrity, and availability of IoT data and operations. They have under-performance, such as not providing requested service on time or modify the reported data.
- *Opportunistic service attacks* [1, 26]: In this attack, a malicious node opportunistically accumulate high reputation by behaving like a benign node at the beginning. After having enough reputation, it starts to demonstrate malicious behaviors.
- *Malicious pre-trusted nodes* [145]: Some trust management schemes, such as [81], deploy pre-trusted nodes in a network for trust computations and storage. In this attack, pre-trusted nodes turn into malicious ones over time.

- *On-and-Off attack/camouflage* [145, 107, 108, 23, 70]: This is a type of strategic attack where a node alters its malicious behaviors occasionally. In other words, a malicious node selectively demonstrates good (benign) and bad (malicious) behaviors to be able to cause damage and remain undetected. The *on* state of the attack is referred to as the *attack state*, in which the node performs an attack, such as not providing expected service with accepted quality. The *off* state of the attack is referred to as the *normal* state, in which the node behaves normally.
- *Sybil attack* [107, 134, 70]: This attack is about the ability of an adversary to generate multiple fake identities for an IoT node to deceive a trust management system. These identities may be used for whitewashing bad reputation or for other attacks.
- *Whitewashing attack* [145, 1, 25, 134]: In this type of attack, a malicious node disconnects and rejoins a network later with a different identity to disguise its bad reputation and to reset its history of poor performance in providing services. The attack happens if nodes can easily change their identity [1]. To prevent this, a trust mechanism should remember the trust of each identity (such as MAC address [1]) and punish inactivity for long period [25]. Doing so will prevent the trust data from being lost if a node leaves and rejoins the network [1]. Also, it is suggested to set the initial trust score of a newly joining node to a low value ([145]) to defend against this attack.

Recommendation/reputation-related attacks: This group of attacks is about recommendations provided by malicious IoT nodes. Recommendations by each IoT node need not be fully trusted [11], just as is the case for the data exchanged by nodes. Malicious nodes can provide false ratings for interactions between them and other nodes [11, 1, 154], and these ratings could be sent as false recommendations to other requesting nodes. Attacks in this category typically affect the reputation of nodes and could be *collective* (a.k.a *collusion/collaborative attack*), i.e., malicious nodes collectively work on interrupting or circumventing a trust management scheme [134, 25, 145, 26].

- *Self-promoting attack* [1, 25, 134, 26]: This attack occurs when a malicious node provides good recommendations for itself in order to be selected as a service provider. It provides a malicious service after being selected.

- *Bad-mouthing attack* [145, 11, 1, 107, 25, 108, 70, 26]: In this attack, a malicious node provides bad recommendations against a benign node and ruin the trust of it, which in turn prevents the benign node from being selected as a service provider.
- *Good-mouthing attack* [11, 107] (a.k.a *Malicious collectives* [44, 145]): In this type of attack, malicious nodes form a collective. They provide maximum trust value as recommendations for other malicious nodes so that malicious nodes can be selected as service providers. They always have inadequate performance in the services they provide.
- *Malicious collectives with camouflage* [44, 145]: This is a type of malicious collectives in which malicious nodes have unstable performance, but still provide best recommendations for other malicious nodes. More precisely, they have adequate performance for certain percentage of the time and inadequate performance other times for the services they provide but they always provide dishonest feedback.
- *Malicious collectives with spies* [44, 145] (a.k.a *ballot-stuffing* [11, 1, 108, 26]): In this collective attack, malicious spy nodes always perform adequately when providing a service so that they boost their own trust. They use their high trust to provide good recommendations for other malicious nodes that they collaborate (which always act malicious and provide dishonest node ratings) to increase their chance of being selected as a service provider.

Researchers (e.g., [145]) have empirically shown that using indirect recommendations could result in lower accuracy to detect collusive attacks.

3.0 Trust in Social Sciences and Implications for Trust in Internet of Things

In this chapter, I present our work [3] that reviews trust in social sciences. First, I present the motivation for this research and background information. Following this, I briefly explain our research methodology. Next, I overview trust measures, which is followed by a trust taxonomy. Then, I discuss several aspects of trust and challenges in assuring it, which include trust development over time, trust repair, trust transfer and reputation, the relationship of trust with context, distrust, risk, and uncertainty. Finally I discuss implications for IoT networks and present a conceptual trust management framework for IoT.

3.1 Motivation and Background

Trust is fundamental for interactions at a micro-level for individuals and a macro-level for societies [59, 52]. It shapes both social and economical exchanges [21, 148, 43, 52, 19, 35], interactions between people [37, 128, 35, 103, 118, 41, 148, 19, 52, 74], between people and technology [100, 73, 155], and between technological elements such as software programs and devices. Trust improves performance [30, 74], efficiency [118, 21, 52], costs [148, 21, 43, 52, 19, 35], and facilitates cooperation, collaboration, strong and close relationships [103, 74, 52, 19, 37, 41, 35, 128, 118], which in turn contribute to the wealth, health, and well-being of individuals and communities [50]. The lack of trust leads to undesirable outcomes as it impediments possible interactions or exchanges. Even worse, it leads to the loss of reputation and money, and more dire consequences for highly critical scenarios.

Trust has been a focus of research attempts in various disciplines [181, 88, 43], such as economics [10, 187], psychology [37, 152], sociology [103], business [88], management [181], marketing [128], and politics [90, 144, 72]. This broad exploration of trust has strengthened the trust literature [15, 153] as each discipline provided a unique insight into the definition of trust and how it develops [43]. Yet there are ongoing debates about several trust-related issues, including the constituents of trust and how to measure it [30, 88, 122]. The challenges in defining, investigating, and measuring trust arise from the nature of it, i.e., being complex, ambiguous, dynamic, and multi-faceted, adds to the difficulty of defining [68, 153, 138, 98]. Also, as I discuss in oncoming sections, there are under-investigated aspects of trust, such as trust repair and trust dynamics. Therefore, challenging and under-investigated trust aspects need further investigation, both in social sciences and IoT domains.

In this research, we contribute to the trust literature through a multi-disciplinary literature review of trust in various social science fields. Our goal is to utilize the existing body-of-knowledge for trust management and to develop trust measures for the IoT domain. Trust has been reviewed by researchers earlier, such as by Fulmer and Gelfand [52] in the organizational sciences literature. To the best of our knowledge, there is no attempt to date for a multidisciplinary review of trust that comprehensively reviews concepts, issues, and measures of trust in different social science disciplines. Also, existing reviews of trust in social science disciplines do not necessarily match trust concerns in IoT networks. We attempt to fill this gap by answering the following questions:

> ***RQ1.*** *What are the core concepts and key properties of trust that have been discussed in social sciences literature?*
> ***RQ2.*** *How is trust being studied and measured by social scientists?*
> ***RQ3.*** *How can we apply core trust concepts, properties, and measures to manage and measure trust in IoT networks?*

To address these research questions, we conduct a content analysis on reviewed articles. As an outcome of the content analysis, we identify core concepts/themes related to trust. Specifically, we identified key properties of trust, trust changes over time, distrust, risk and uncertainty, reputation, trust transfer, and trust repair as significant trust-related concepts. We also overview definitions of trust, constructs that are closely related to trust, research methods and measures used by social scientists in a complete paper ([3]). Yet in this chapter, I present a brief summary of subset of these topics that are highly relevant to the dissertation. We present a trust taxonomy that classifies different types of trust based on reviewed articles. We discuss the implications from our review for IoT networks. Specifically, we answer how those aforementioned significant trust concepts can relate to IoT networks and can be used to measure trust. Trust measures can be used by practitioners and researchers to monitor and evaluate trust within and among network administrative domains for better outcomes, i.e., security and reliability of data and operations. Distilling the information from implications for IoT networks, we propose a conceptual trust framework aimed as a guide to managing trust in IoT from various aspects. In the following subsections, I first present terminology relevant to trust in social sciences. Next, I overview trust definitions and key trust properties. Finally, I discuss trust-related constructs used in social science studies.

3.1.1 Terminology

Here I present the terminology relevant to the discussions in this chapter and used by trust-related articles in social sciences.

Construct: a.k.a variable, an abstract idea, theme, or subject matter that is measured using survey questions [99]. A construct can be measured using a single question or multiple questions, a.k.a items, or dimensions of a construct [99].

Trust antecedent: a.k.a trust predictor, a variable that is used in a theoretical model and is expected to influence and explain dynamics of trust ("cause of trust" or "condition leading to trust" [22]) where trust is an outcome variable. Examples of trust antecedents include leadership style [41], participation in decision making [41], the trustworthiness of a trustee [30], trust propensity of a trustor (disposition to trust) [30, 89], the perceived reputation of a brand [143, 89], perceived risk [89], perceived security and privacy [89], perceived information, service, and design quality [89] of an e-commerce website, social context and social support in social commerce [66].

3.1.2 Trust Definitions

Researchers have studied trust in various fields, including economics, sociology, and psychology. Trust has been approached from different aspects in these disciplines. For example, sociology studies have focused on institutional aspects whereas psychology researchers emphasized personal aspects of trust [88]. As trust has been studied in a variety of fields, it has diverse definitions [122, 73, 36]. Even though trust has been defined and measured in different ways, common elements are used to describe it [166, 84], such as belief, attitudes, or expectations towards acceptable outcomes resulting from actions of another individual, group, or organization [13, 105, 111] or that best interests of an actor will be served [84]. A conception of trust that has been implicitly accepted by most scholars is that it represents *a psychological state of positive expectations about the motives and actions of another party* [166]. In the following, I present widely adopted trust definitions.

One definition that has been accepted widely and commonly used in the *organizational science domain* [79] was proposed by Rousseau et al. [153], which describes trust as "a psychological state comprising the intention to accept vulnerability based upon positive expectations of the intentions or behavior of another". Mayer et al. [118] defined it as "the willingness of a party to be vulnerable to the actions of another party based on the expectation that the other will perform a particular action important to the trustor, irrespective of the ability to monitor or control that other party." Definitions by Rousseau et al. and Mayer et al. have two common components, which are *the intention to accept vulnerability* and *positive expectations.* Coleman [29] presents a formal definition of trust. Equation (1) is the formula for the expected payoff of a trustor from a trust relationship ($E(R)$). Trust is considered as a monotonically increasing function of $E(R)$ [183]. Let p be the probability that the trustee will be trustworthy, and G is the gain of the trustor in this case. Similarly, $1 - p$ is the probability that the trustee will be untrustworthy, and L is the associated loss of the trustor. If $p/(1-p) > L/G$, then the decision would be to trust. Conversely, if $p/(1-p) < L/G$, the decision would be not to trust. The case when $p/(1-p) = L/G$ would yield an indifference situation for the trustor. The expected payoff is hence defined as:

$$E(R) = p.G + (1 - p).L \tag{1}$$

In addition to the widely adopted trust definitions above, there are also other definitions of trust in specific *disciplines.* In the *psychology domain*, trust is defined as the "reliance upon the characteristics of an object, or the occurrence of an event, or the behavior of a person in order to achieve a desired but uncertain objective in a risky situation" [57]. Giffin [57] further explains the dimensions of trust as intelligence, good character, and goodwill that were set forth in Aristotle's Rhetoric [54] and respectively correspond to ability, integrity, and benevolence dimensions of trustworthiness in the model that was later proposed by Mayer et al. [118, 54] (see Section 3.1.4 for the model proposed by Mayer et al. [118]). Similarly, Nooteboom [135] describes trust as a concern for the ability and intentions of a partner to perform concerning agreements, which represents competence and goodwill dimensions. In *marketing literature*, trust is approached from two perspectives as *a behavioral intention* or *a set of beliefs* [128, 54] such that trusting beliefs lead to behavioral intentions for trusting [54]. Curral and Judge [35] approach trust from a behavioral intention perspective and describe it as "an individual's behavioral reliance on another person under a condition of risk". Schurr and Ozanne [158] approach trust from a belief perspective by defining it as "belief that a party's word or promise is reliable and that a party will fulfill his/her obligations in an exchange relationship". In the *politics literature*, trust of people in government is defined as the "rational or effective belief in the benevolent motivation and performance capacity of" the government [137] and the evaluation of the performance of government compared to the expectations by people [72]. Consumer behavior literature defines trust in a brand as "the willingness of the average consumer to rely on the ability of the brand to perform its stated function" [24].

3.1.3 Key Trust Properties

In this section, I present key features of trust discussed in social sciences literature.

P1. Trust represents the expectation from a trust relationship ($E(R)$ in (1)), so it is different from perception about the trustworthiness (p in (Equation 1)) [183, 29].

P2. Trust is a complex and "multi-faceted", or a "multi-dimensional", phenomenon that depends on numerous factors [103, 18, 54, 68, 110, 53, 118, 153, 22, 87, 35, 86, 36].

P3. Trust is context-specific [183, 18, 103, 184, 118], which means that it is associated with different contextual conditions [183, 18], i.e., its dimensions depend on the circumstance of an interaction [54, 173].

P4. Trust involves uncertainty as the trustor can only estimate the trustworthiness of a trustee (reflected by p in (1)) [183, 29].

P5. Trust is a dynamic process, which evolves over time [153, 115, 181, 116, 54, 68, 186, 43, 73, 160, 184]. It increases or decreases over time as being revised by the trustor as a result of the experience that the trustor gains by engaging in relationship with a trustee [183, 115] and as people are exposed to information [73].

P6. Trust is reciprocal [36, 22] such that a trustor demonstrates trusting behaviors if s/he expects trustworthy behaviors from a trustee, who in turn reciprocates it with trust expectations and trusting behaviors [22].

P7. Trust is easily broken and challenging to repair [102, 160, 168, 86, 42].

Overall, based on the key trust properties above, we define trust as:

> *A multi-faceted, complex, dynamic, and context-dependent concept that is driven by uncertainties about undesired outcomes and is the expectation of a trustor from a trusting relationship based on the probability of positive and negative interactions with associated losses and gains. Once this expectation turns into a trusting behavior, it may be reciprocated by the trustee. Once it is broken, it will not be easily recovered.*

3.1.4 Trust-Related Constructs

The trust literature has distinguished between various important trust-related constructs [30, 183]. These constructs are *trust, trustworthiness, trust propensity, trusting beliefs*, and *trusting intentions*. In what follows, I clarify these constructs.

3.1.4.1 Trust and Trustworthiness

Trustworthiness is distinguished from *trust* in social science disciplines, such as by Mayer et al. [118] and Ashraf et al. [10]. *Trustworthiness* is a crucial concept to understand *trust* because a person trusts another person as s/he is trustworthy and beliefs about trustworthiness fosters trust [45, 30, 10]. Mayer et al. [118] proposed that trustworthiness comprises three facets, i.e., characteristics of a trustee, —*ability, benevolence, and integrity*—, and is an antecedent of trust. This model is sometimes referred to as the *ABI model* [79]. As the ABI model is a widely cited social trust model [30, 123, 150, 157, 148], I overview its components below.

Ability: a.k.a *competence* [118, 36, 87, 38, 166, 22, 84, 57] corresponds to the perceptions of a trustor about the technical skills and knowledge, general wisdom and interpersonal skills of a trustee [87, 132, 30, 86], which are required to perform a job and be successful in an organization [30, 86].

Benevolence: is about the intention of a trustee for doing good without any profit motives [30, 178, 118], caring for and acting in the interests of the trustor [122, 132, 118, 58]. It is usually associated with loyalty, openness, caring, supportiveness [118], and the absence of opportunism [96].

Integrity: corresponds to the belief about the degree to which the trustee conforms to ethical principles [30, 122, 58] and principles that are found to be acceptable by the trustor [118, 87, 58, 178, 86], is honest [122, 132] or is willing to keep promises [88, 122, 132]. Some synonyms used in lieu of integrity include promising fulfillment [30], fairness [30, 58], justice [30], honesty [120, 148, 58], credibility [120, 148], and reliability [120, 148].

Even though both benevolence and integrity are ethical traits, benevolence is based on altruistic motives whereas integrity (which is about keeping promises and being reliable) may be due to utilitarian reasons [122], i.e., for personal gains. Some researchers consider benevolence and integrity together as a combined trust construct *goodwill* (e.g. [147, 36, 97]). Competence and integrity have been consistently found to be more important determinants of trustworthiness [86], whereas integrity is sometimes deemed as more significant than competence and vice versa [86].

3.1.4.2 Trusting Beliefs, Intentions, and Behaviors

As proposed by McKnight et al. [123, 121] and followed by other researchers (e.g. [173, 87, 132, 86, 42]), trust comprises two elements that are *trusting beliefs* and *trusting intentions*. Trusting beliefs are about the competence, benevolence, and integrity of another party [87, 132, 86, 42], which may lead to trusting intentions [87, 86, 42]. Trusting intentions correspond to the willingness to become vulnerable to the actions of another party under risky circumstances [173, 87, 86, 42, 96]. Trusting intentions are different from trusting beliefs because an individual may be unwilling to be vulnerable even though having trusting beliefs towards another individual, such as due to uncontrollable risks inherent to a situation [118, 173]. Both trusting beliefs and intentions are argued to be affected by the trust propensity of an individual [123] (see Section 3.1.4.3).

Based on the ABI model [118], trust is different from *trusting behaviors* and trustworthiness perceptions about a trustee [178]. Mayer et al. [118] note the fundamental difference between trust and trusting behaviors, such that the former is about willingness to be vulnerable and the latter is engaging in an action to take a risk. Trusting behavior is performing an action based on positive expectations about the trustee, such as the amount of money passed to a trustee in a trust game experiment [42, 160], or placing an order in an online shopping platform.

3.1.4.3 Trust Propensity

Trust depends on experience and enough time should pass to make decisions about trust [84, 30]. Yet, experience is not the only way of making trust decisions. There are dispositional factors that decide trust such as personality [84]. *Trust propensity* is one such factor that is connected to personality [118, 84]. This type of personality-based trust is referred to as with different names in the literature [30], namely *trust propensity* [118], *generalized trust* [171], *dispositional trust* [95], *disposition to trust* [121, 122], and *trusting/trust disposition* [110, 54].

Trust propensity is a "dispositional willingness to rely on others" [30], whereas trust is "the intention to accept vulnerability to a trustee based on positive expectations of his or her actions" [153]. McKnight et al. [121, 122] describe it as the degree that a person tends to be willing to depend on others. It can also be defined as the belief about the degree that people can be trusted in general [123, 96, 110], or a personal trait that reflects the tendency to trust others [132, 73]. Another definition of trust propensity was propounded by Rotter [151], which defines it as a general expectation that promises by other people can be relied on. Hence, trust propensity is not a situation-specific but a general inclination to trust others [118, 123, 73].

McKnight et al. [123] proposed that trust propensity can be depicted using two subconstructs, namely *faith in humanity* and a *trusting stance* (i.e., a trusting attitude towards others). Faith in humanity is the belief that others are usually well-meaning and upright

[122, 27]. McKnight et al. [123] proposed further subconstructs for faith in humanity, namely competence, benevolence, and integrity, which are personal attributes of others and the dimensions in the ABI model. Trusting stance refers to having a general trusting attitude towards others expecting better outcomes from relationships, regardless of belief in others' personal attributes [121, 122, 123]. Trusting stance is based on a calculative view of trust in economics-based trust research [27], i.e., trusting others regardless of being proven to be trusted and until they prove otherwise [122].

In the following section, I present a brief summary of our research method. More details are presented in our paper ([3]).

3.2 Research Methodology

We performed a multi-disciplinary literature review to explore trust in social sciences. Instead of compiling a complete set of studies for an in-depth review, we aimed to review papers that discuss a variety of trust-related topics in social sciences disciplines broadly. For this purpose, we followed the *three-pass* approach proposed by Keshav [85] to identify relevant articles. The three-pass approach to literature review requires identifying an initial set of *core articles* (3-5 articles) that can be considered seeds for a literature review. Core articles are selected from recent papers in a research area using an academic search engine and keywords. From core articles, a set of *key articles* are identified to read further by inspecting the related work section and reference list of core articles. Next, I present information about inclusion and exclusion criteria that we use to select core and key articles.

3.2.1 Selection Criteria

We utilized the Scopus database to select core articles[1]. We used "trust" as the search term in the article title field. We limited results to only English articles published in 2019-2020 for detecting recent core articles. Scopus provides different *subject areas*, each of which further branches into sub-topics. Based on the classification provided by Scopus, we repeated our search for the following *sub-topics of Social Sciences subject area*: i) arts and humanities, ii) business, management, and accounting, iii) decision sciences, iv) economics, econometrics, and finance, v) psychology, and vi) social sciences[2]. Hence, we performed six different searches to find core articles in each discipline. An example query term we used is as follows:

```
SUBJAREA(ARTS) TITLE(trust) AND (LIMIT-TO (PUBYEAR,2020) OR LIMIT-TO ( PUBYEAR
    ,2019)) AND (LIMIT-TO (LANGUAGE,"English")
```

For each sub-topic, we selected 3-5 *core articles* among the retrieved articles that have full-text available to us. For selecting these core articles, we reviewed the title, abstract, introduction, conclusion, and subtitles. If the article is about a study for factors affecting trust, trust changes over time, trust repair, or trust measures, and its topic does not significantly overlap with the previously chosen core articles (so that reading list is not focused on a single topic, e.g., trust in automation systems) we considered it. To select *key articles* from core articles, we examined their related work section and reference list. We noted repeated citations and author names (i.e., highly cited seminal works) in core articles to detect key papers. We did not impose any limit on publication year for key articles. We applied the same selection criteria that we considered for core articles.

[1] Limiting a literature review to a single database may lead to bias such that articles may be retrieved from a specific set of journals or publishers [98]. Yet, we utilized the Scopus database only for selecting core articles, which are like seeds for our search of additional key articles and constituted a small number of articles to be reviewed (3-5 articles in each research field). Thus, we believe that using other or additional databases should not significantly affect the final set of selected articles.

[2] Although we could not find information about what this sub-topic includes different than others, we anticipate that it may include articles that do not belong to other sub-topics or that are relevant to multiple sub-topics.

3.2.2 Analysis

We performed a *content analysis* on the articles in our reading list. Content analysis is performed by reading articles in full and extracting high-level and significant concepts arising from the content, which are known as *codes* or *themes* (see [22, 96] for more details about content analysis). This process is also known as *coding.* We read core and a subset of key articles[3] in our reading list in full to extract themes from them and the content relevant to each concept. While generating high-level codes from the content of articles, we focused on definitions and key properties of trust, factors affecting trust formation, theoretical bases, main contributions, research methods and measures for studying trust, and different trust types. These concepts are trust definitions, key properties of trust, trust-related constructs, trust changes over time, distrust, risk and uncertainty, reputation, trust transfer, and trust repair. I present more detailed information about them in the oncoming sections. In addition to the content analysis, I presented core articles during periodic meetings for our survey paper. During these meetings, a subset of committee members posed questions and brainstormed important trust concepts and their implications for computer networks. We present the output of this process as implications for IoT networks in Section 3.12 (and as future research directions in the original paper).

3.3 Trust Measures Used in Social Science Studies

Our ultimate goal is to utilize the knowledge we gain from reviewing the trust in the social sciences literature and apply it to IoT networks to devise trust computation methodologies. For this purpose, we reviewed measures used by social scientists for gauging trust. In this section, I overview trust measures used in the reviewed studies.

As informed by our review, researchers mainly use quantitative approaches to measure

[3] Although we initially planned on reading all the key articles in full, we realized that the information we obtain from 69 articles reached a point of data saturation, i.e., we started to see the same high-level trust concepts, definitions, and measures. This approach for stopping when data reach to saturation is a reasonable method suggested for qualitative research [31]. Hence, after that point, we read the remaining key articles partially as needed to explore and explain concepts.

trust in their studies, which are *questionnaires (surveys)* as *instruments* (a.k.a *scale* or *trust measure*). Typically, these approaches are known as an *instrument* (a.k.a *scale*) in a social science study, or as a *trust measure* in a trust study. Using a survey instrument to measure trust has several benefits, including quantifying trust levels, examining changes or variation over time and across populations, and the ease in incorporation in existing monitoring procedures [138]. Apart from surveys, other numeric measures can be used for gauging trust, such as the amount of money given by a trustor and returned by a trustee in behavioral economics studies [10]. There are also qualitative approaches used in social science studies to investigate trust relationships, such as interviews. Yet, as they do not have numeric scales, they are not regarded as trust measures.

A survey consists of single or various concepts, a.k.a *constructs* or *dimensions* [35]. A construct represents a conceptual definition of a variable [159]. Typically, a construct includes multiple relevant questions, a.k.a *items*, which are answered by study participants, a.k.a *subjects*. In the articles we review, trust is measured using *uni-dimensional* or *multi-dimensional* scales. A uni-dimensional scale captures a single trust construct, whereas a multi-dimensional scale consists of multiple trust constructs [138]. For example, Doney and Cannon [43] used a uni-dimensional scale as a measure of trust in their survey such that items are grouped under a single construct that reflects "trust of supplier firm" or "trust of salesperson". On the other hand, Gefen and Straub [54] used a four-dimensional scale in their e-commerce trust study. The dimensions (constructs) that captured trust were belief in ability, benevolence, integrity, and reliability of a vendor. Each construct consisted of items relevant to what is measured by it and collectively measured trust in a vendor. In their review of trust measures in health systems, Ozawa and Sripad [138] found that researchers used 12 questions on average, ranging from 4 to 59 questions, to develop a trust measure. It is crucial that constructs and items are rigorously prepared and reviewed before administering a survey. Constructs are typically determined based on a literature review and items could be determined based on interviews or adapted from previous surveys (see [35] for an example).

We present a representative set of trust measures in Table 12 of Appendix A to demonstrate how researchers measure trust and trust-related constructs (see our discussion in Section 3.1.4) in their studies. Survey items in the table have been widely adopted by researchers in their trust studies to measure trust and related constructs.

3.4 Trust Taxonomy

In this section, I discuss different trust types as identified by social science researchers and present a trust taxonomy. Previous research has suggested various criteria/factors to identify trust types. For example, trust has been classified based on the stage of a relationship (early, medium, mature) [148], the way that trustworthiness is perceived [33], how wide the scope of a trust type [33], the type of vulnerabilities addressed by trust [33], and the level of trust and trust analysis [52]. We consider the *formation process*, *communication channel*, *level of parties*, and the *type of trust referent* as criteria for categorizing trust. Figure 2 summarizes our taxonomy. I present the details in the following sections.

3.4.1 Trust based on Formation Process

Trust can be explained concerning its *formation process*, specifically as a *static* or *dynamic* concept [181]. The static view to trust formation considers the personality traits of a trustor [123] or characteristics of a trustee as suggested by the ABI model ([118]), namely ability, benevolence, and integrity. The dynamic view approaches trust as an "evolving experience" such that trust starts from an initial form or level and develops in time by progressing through several stages [181]. For example, in inter-organizational relationships, trust starts to develop at an individual level and then develops into the inter-organizational level through four stages, which are trustworthiness-related evidence gathering, personal interactions, transfer of trust, and institutionalization [156].

3.4.2 Trust based on Communication Channel

Another grouping of trust considers the communication channel. In various disciplines including human-computer interactions, business and commerce literature, trust is classified as *online* [110, 68, 33] (a.k.a *social* ([66]) and *offline* trust [68, 66, 33]. This distinction

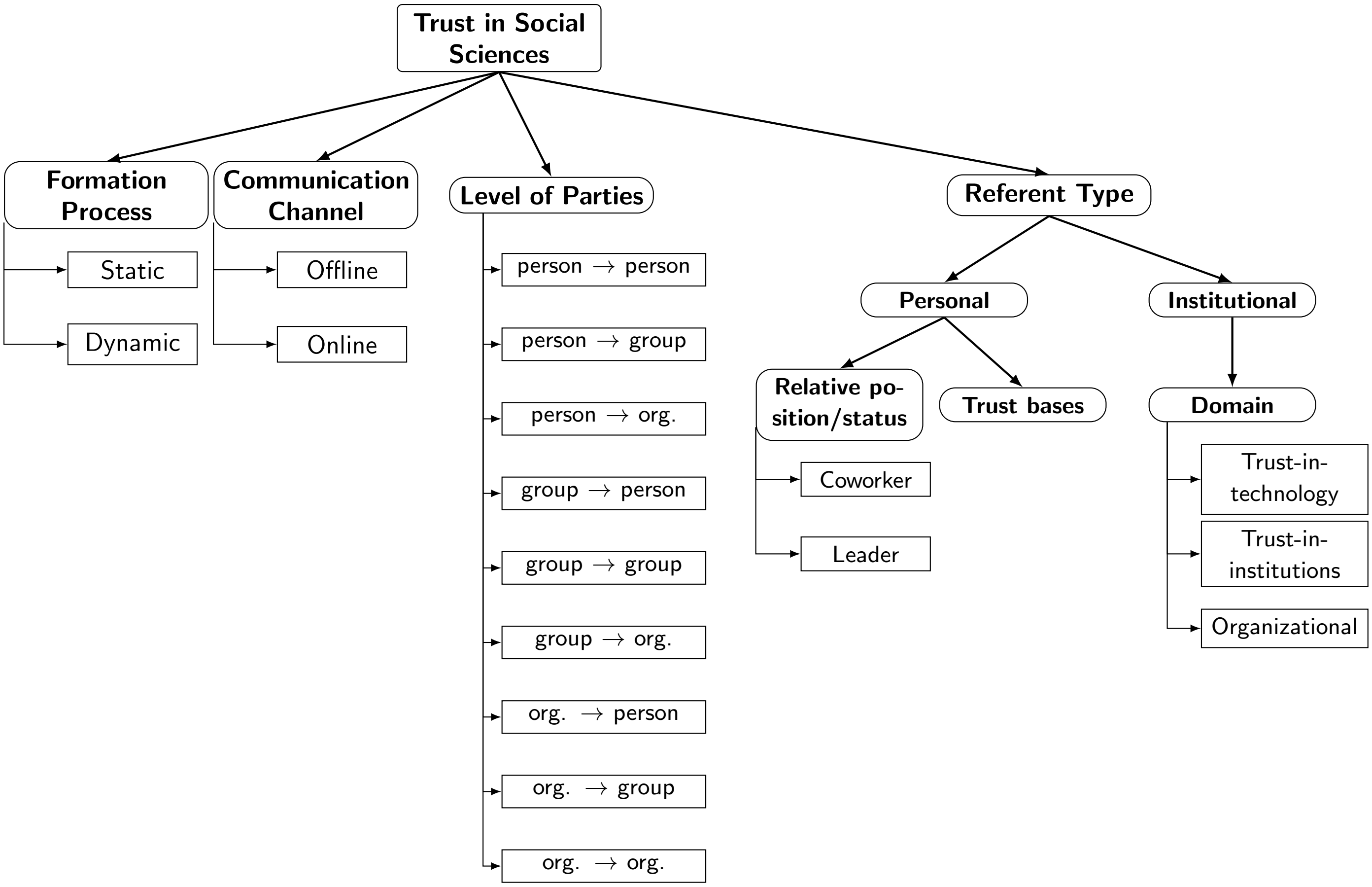


Figure 2: Classification of trust in social science disciplines

is based on the communication channel, usually between a customer/consumer and a business/vendor. In traditional high street shops, consumers spend time in physical stores and have actual human contact and sociability, whereas in online stores, there is an anonymous and impersonal relationship between consumers and a vendor [66]. These differences require trust to be approached differently as offline and online trust. Online trust corresponds to trust in online business environments and is defined as the expectation by consumers about the trustworthiness of online firms and that they will care for consumers with honesty without abusing characteristics of online environments for their benefits [88].

Online and offline trust are similar in many aspects, but they have differences in some key aspects [68]. For example, in an offline environment, partners can know each other better than in an online environment. Consumers may be hesitant to buy from online vendors as they may have uncertainty about vendors and find it risky to share personal information online [27]. Also, online vendors have lower barriers to enter and exit the business. This may arise from another key difference, which is the absence of physical elements in online businesses. That is, offline vendors invest in physical buildings and personnel and their customers have the opportunity to evaluate products physically. These are missing in online businesses, which may affect consumer trust in sellers [68].

3.4.3 Trust based on the Level of Trustor and Trustee

There are two parties in a trust relationship, which are a *trustor* and a *trustee* (trust referent). We have identified that trust can be at different levels concerning the trustor and the trustee. Das and Teng [36] discuss that trust can be at personal, organizational, inter-organizational, and even at international levels. Sitkin and Ruth [166] draw attention to the categorization of trust at interpersonal and institutional levels. Laan et al. [96] approach trust at interpersonal and inter-organizational levels. Stewart [173] also notes three levels of trust as a person/individual, a group, and an organization.

Although these researchers discuss trust at different levels, they do not touch upon the trust based on the level of a trustor and a trusted party separately. Curral and Inkpen [34] proposed to conceptualize the trust in international joint ventures at three levels of trustors and trustees, which are person, group, and firm. Fulmer and Gelfand [52] also shed light on this issue in their systematic review of trust in organizational sciences. They framed the previous work based on the trustor and trustee considered by researchers in reviewed articles. They grouped both trustor and trustee into three as *individual, team,* and *organizational*[4]. They emphasize the importance of distinguishing levels of a trustee as different trustee levels will result in different trust antecedents and consequences. For instance, the concerns of an employee with its coworker would be different than that of its organization.

From the perspective of these researchers ([52, 34, 36]), trust can be *i*) *interpersonal* (from a person to another person), *ii*) from a person to a group of people (e.g. team members), *iii*) *organizational* (from a person to an organization), *iv*) from a group of people to a person, *v*) from a group to another group, *vi*) from a group to an organization, *vii*) from an organization to a person, *viii*) from an organization to a group of people, and *ix*) *inter-organizational* (collectively from a group of people in an organization to another group of people in another organization). For example, in a model of trust between groups (denoted as $G \to G$ in [34]), the trustor could be a group of managers in a joint venture and the trustee could be another group of managers in a partner firm [34].

3.4.4 Trust based on the Type of Referent

Another prominent grouping of trust concerns the *type of trustee*, i.e., trust referent, for trust classification. At the highest level, this grouping distinguishes *personal/interpersonal trust* from *institutional/institution-based trust* [122, 148, 186, 110, 115, 155, 166]. *Personal trust* is described or conceptualized similarly in different research fields, even though it is named differently, such as trust-in-human [115, 73], trust-in-sellers [110], personal, interpersonal, or individual trust. Personal trust is about expectations of an individual about positive outcomes of another person, based on personal experience [166]. In organizational settings, the type of a referent could be based on the *relative position or status* of an em-

[4] Instead of trustor and trustee, they use the terms "trust at a level" and "trust in a referent". As they describe, trust at a level corresponds to the source/originator of the trust, so we call it trustor here. For example, trust levels could be an individual, team, and organization, which respectively refers to the trust of an individual, aggregated trust shared by team members, and aggregated trust shared by organization members. Trust referent corresponds to the target of the trust, so we call it trustee here. Similarly, trust referents could be an individual (e.g., leader, coworker negotiation partner), team, and organization.

ployee. For instance, trust relationships in organizations may be affected based on if the trustee is a co-worker or a leader [30, 41]. Personal trust can also be classified by different trust types, i.e., *trust bases*, [115]. Some trust bases proposed by researchers are *calculative-, cognitive-, emotional/affective, experience-, knowledge- personality-,* and *institution-based* trust [115, 68, 122, 10, 148, 33, 105, 119, 153, 187, 59, 178, 95].[5]

Institution-based trust is named or conceptualized differently in different *domains.* It is known as *organizational trust* in the organizational science literature, *trust-in-technology* in the online business [122, 115, 110, 148] and human-automation trust literature [155, 73], and *trust in institutions* in civic engagement/politics literature [186]. Interpersonal trust has been studied more widely compared to institutional trust [166]. In the sequel, we first discuss institutional trust and its further classification in different disciplines. Next, we present personal trust-related propositions and classifications.

3.5 Trust Over Time

Most researchers agree [43] that trust in relationships develops and improves over time [153, 115, 181, 116, 54, 68, 186, 73, 104, 34, 19, 160, 97, 35, 184, 118]. Vanneste et al. [183] investigate this common premise in their meta-analysis study. Results showed that there is a small correlation between trust and the duration of a relationship. Thus, they hypothesized that there are other mechanisms affecting trust changes over time, such as changes in the value of a relationship, i.e., gains/losses of a trustor. The interplay between them decides whether the trust will increase, decrease, or stay unchanged. These results lead us to a relevant question: *"Does trust increase over time through interactions by accumulating evidence rather than just with the increasing duration of a relationship?"*

[5] *Calculative trust* is about the rational behaviors of trustor and trustee such that both of them calculate their gains and loses in a trust relationship and decide to trust/be trustworthy or not to trust/be untrustworthy[187, 21, 59]. *Cognitive trust* is about the willingness of a trustor to depend on the ability and consistency of a trustee [127]. Thus, cognitive trust is evaluated based on the three dimensions of the ABI model, which are competence, benevolence, and integrity (see Section 3.1.4 for details) [88, 122, 118, 148]. *Emotional trust* is characterized by a single social experience [105, 119], an immediate emotional reaction [148], and a feeling of security and strength of a relationship [88]. For the rest of the trust bases, we refer the readers to [3].

In regards to this question, some researchers emphasize that time is sufficient for interpersonal trust to develop [119, 181]. Other researchers argue that the time alone is not sufficient and interactions are required for desired outcomes and trust [181, 93, 97]. Thus, trust builds gradually over extended periods of time through ongoing observations and interactions [54, 110, 186, 148, 43, 124, 87, 19, 97, 118] with other people, organizations, and the surrounding environment as interactions help a trustor to accumulate knowledge about a trustee and to form expectations about what they can do [54, 148, 43]. This is explained by the "prediction process", which is one of the five cognitive processes identified by Doney and Cannon [43] that helps to foster trust in business relationships. The prediction process is about the ability to forecast the behavior of another party and improves over time through experiences.

Even though it requires time to gather evidence on trustworthiness, trust decisions may need to be made before enough time has passed [30] in a trusting relationship. The early phases of a relationship are the most uncertain and tenuous stage, so it is challenging to develop trust initially [132]. This important challenge touched upon in the trust literature is known as *initial trust* [123, 73] that is about how to form trust in an unfamiliar trustee when there is not enough, meaningful, or credible information about the trustee [15]. Credible information accumulates over time when parties interact [122]. One of the most widely cited models that focuses on initial trust [68] was proposed by McKnight et al. [122]. Based on this model, trust propensity, or dispositional trust, is a factor that contributes to the formation of initial trust when parties do not have enough experience and unfamiliar actors present [15, 84]. Trust propensity comprises faith in humanity and a trusting stance (see Section 3.1.4). For example, having faith in clinicians, in general, would help a patient to trust a clinician at the first meeting [98]. Mechanic and Meyer [124] argue that initial trust may be based on reputation. For example, medical relationships start based on recommendations from family members and friends. Another proposition about initial trust is that some people demonstrate surprisingly high levels of trust at the early stages of relationships even without sufficient interactions [123, 58]. This is because those individuals assume that trust is warranted if there is no contrary evidence [123, 136], which is known as the "trustworthy until proven otherwise" assumption [87]. This assumption is related to the disposition of an individual to trust, i.e. personality [87]. There could be other reasons that may cause high levels of initial trust, such as the reputation of the trustee, stereotypes, institutional structures such as laws and regulations, etc. [123].

After an initial trust is established between a trustor and a trustee, trust evolves based on the perceptions/expectations and interactions between partners. Hoff and Bashir [73] coined the term "dynamic learned trust" to represent trust during this course of interaction. As interactions take place, a trustee adjusts its trust based on the competence of the trustee (or performance of an automation system as the trustee [73]). As the performance of a trustee may vary, trust is most likely to fluctuate [73]. More generally, the evolution of trust depends on the perceptions about trustee characteristics [184], i.e., dimensions of the ABI model. The ABI model proposed by Mayer et al. [118] explains trust evolution from the perspective of a trustor, i.e., unidirectional view, through a feedback loop between trustworthiness dimensions and risk-taking. As such, a trustor takes a risk with a trustee, observes the outcome, and reevaluates trust. In case of a positive outcome, the trust will be either maintained or increase based on prior beliefs about trustworthiness. In case of a negative outcome, the trust will decrease as a combination of the decrease in trustworthiness dimensions. Among the three dimensions, benevolence is stable in established relationships and less stable early in a relationship. According to the view that argues trust is reciprocal, trust develops between two individuals in a circular and mutually-reinforcing way [22]. More precisely, the process starts with expectations about the behaviors of another. If the expectation is towards trustworthy behaviors, one demonstrates trusting behavior, such as disclosing information, accepting influence, etc. The trustee responds with trusting behaviors upon perceiving the trust. This cycle continues by reinforcing expectations about trustworthy behavior and building trust. The contrary happens if the initial expectations point to mistrust such that one party expects untrustworthy actions and demonstrates this through withholding information, rejecting influence, etc. Perceiving this, the other party responds with expectations of untrustworthy behaviors, and so on. This cycle happens if the trust is assumed to be reciprocal [22].

3.6 Trust Repair

A trustor takes a trusting action not just because s/he trusts the trustee, but because of realizing gains if the trustee fulfills the expected part of the exchange [148]. This means that a trusting action causes vulnerability(ies) for a trustor because the result of this action is in the hands of a trustee [148]. It is not uncommon that positive expectations of the trustor will be violated [87]. The trustee may not fulfill its duties due to the lack of needed abilities or the lack of motivation, so may exploit the vulnerability of the trustor [148]. It is known as *trust violation* or *trust failure* when a trustee exploits the vulnerability of a trustor.

Trust repair is about increasing reduced trust in the aftermath of trust violations [87, 86, 42]. Previous research has shown that damaged trust is difficult to repair [160, 168, 42]. Trust repair is an important practical problem [160] as there are various challenges in repairing trust, and the process requires different strategies compared to initial trust-building [87, 86, 42]. First, due to the "trustworthy until proven otherwise" assumption [136, 123], trust levels could be high in the early stages of a relationship [123]. Yet after a trust violation, this assumption is invalidated, and trust may decrease to a level lower than initial trust. This makes the magnitude of required trust repair greater than the required initial trust levels [87]. Second, after trust violations, it is not enough to establish positive expectations, but negative expectations should be dealt with to repair trust [87, 42]. Overcoming negative expectations resulting from a failure is crucial because it conveys that actions have been taken that target direct and indirect contributors to prevent/avoid the same violation in the future [58]. If violations repeat, it is much more challenging to restore the trust [102]. Third, choosing the proper trust repair action is complicated because a mistrusted party may have to resolve issues about actions that it did not perform and about people who did not get direct harm from violations. Finally, different stages of a relationship impose different challenges for trust repair. Repairing broken trust in newly formed relationships may be more challenging than established relationships as in the latter, trustors have higher motivations towards the relationship [86, 42]. For established relationships, trust repair may be more challenging due to stronger emotions incurred by a trust violation [42].

The broken trust can be repaired to the degree depending on the response by the violator

[58]. *Type of trust repair action* is a determinant of the effectiveness of trust repair efforts [67, 58], namely, *verbal* and *substantive* strategies [178, 42, 18, 79] (see details later in this section). Effectiveness of trust repair efforts can also be determined by *trust violation type* [67, 79, 58, 87, 86, 42] that could be *competence-based*, *benevolence-based*, or *integrity-based* trust violations [79, 87, 86, 58], i.e., whether violation matters concerns of competence, benevolence, or integrity dimension of trust. Research has shown that violation in one dimension of the trust may cause a decline in other trust dimensions as well [87, 58], the concept called as "contamination" effect [58]. The same effect does not apply to trust repair efforts, i.e., repairing trust in one dimension does not necessarily improve other trust dimensions [58]. This is explained by the Schematic Model of Dispositional Attribution [146] that suggests people tend to weigh negative evidence more than positive evidence in general and for evaluating integrity, but do the reverse for ability [146, 169, 87]. Thus, different actions should be taken to repair different dimensions of trust [58]. Tomlinson and Mayer [178] argue that integrity is the most stable among the three dimensions of the ABI model [118]. Hence, if a trustor ascribes that low integrity is the cause of a negative outcome in a trusting relationship, then it is the hardest to repair.

Trust repair can be categorized into two as *trustor-centric* and *trustee-centric* [18]. Between these two, trustee-centric trust repair has been investigated more in the literature [18]. Trustor-centric trust repair concerns the role of the trustor in the process, such as forgiveness and emotions [18]. Tomlinson and Mayer [178] also emphasize the role of addressing emotions of a trustor in the trust repair process, such as reducing anger and fear. Trustee-centric trust repair focuses on the strategies that could be pursued by the trustee to repair broken trust [18]. The trustee-centric trust repair is further categorized into *verbal* and *substantive* strategies [178, 42, 18, 79]. Verbal strategies comprise apology [87, 18, 67, 79, 42, 86, 102, 160, 177, 178], denial [87, 178, 18, 79, 42], promises [18, 67, 79, 160], explanations [18, 79], excuses [178, 42], justification [178, 42], and communication [18]. Verbal strategy, especially apologies, may be discounted as "cheap talk" by a trustor as they are costless mere words [42]. A substantive strategy is more comprehensive than a verbal strategy [18], i.e., they are not mere words but involve tangible elements and involve a cost to the trustee [42]. Examples of substantive strategies include penance (i.e., paying penalty)

[178, 79, 18, 42], hostage posting (i.e., "providing victims the ability to monitor and sanction" [42]) [178, 18], organizational reforms after trust violations, such as changing rules and regulations [42, 18], manufacturing processes [18], organizational culture [18], and "legalistic remedies" such as policies, procedures, and monitoring [166].

3.7 Trust Transfer

Trust transfer is described as building trust impressions based on the cues provided by third parties instead of relying on one's own prior experience with a trustee [147], From a different angle, trust transfer is forming initial trust in an entity based on trust in other related entities (still own experience) or based on different contexts than a current encounter between a trustor and a trusted target, such as a different location [173]. Trust can be transferred from a better-known trusted entity with whom the trustor has little or no direct experience [43, 173]. For example, trust can be transferred from a marketplace to the sellers in that marketplace [110]. A patient who trusts in his/her clinician may trust institutions or the national health system more [98]. Reusen and Stouthuysen [147] hypothesized that third-party information has different levels and these levels affect trust differently in a buyer-supplier relationship. More precisely, third-party information could be *neutral* or *favorable*. Neutral information is knowing only that other buyer firms have done business with the supplier. Favorable information is the knowledge about the positive outcome of prior experiences by other firms. Neutral and favorable third-party information affect competence and goodwill dimensions, respectively.

Trust transfer is also known as *trust transitivity* [65, 61, 6] concept in networks. In network terms, trust transitivity is described such that if a node u trusts node v, which trusts node w, then node u may trust node w indirectly without having to have direct prior experience [61]. Trust prediction solutions for networks may rely on the trust transitivity concept such that trust is *propagated* on paths (a sequence of nodes linked together) in a network and *aggregated* from multiple paths [6]. Transitivity-based trust prediction approaches are criticized for being subject to uncertainties due to two reasons [6]. First, trust/trust-related evidence decays on long paths. In other words, “the strength of trust decreases with each degree of separation” [65]. Second, aggregation of trust from multiple paths may lead to contradictions. As a remedy to these two drawbacks, local solutions that consider direct neighbors of a node in a network for trust prediction, are advised [6]. However, this class of trust prediction methods may be subject to biases. In other words, if direct neighbors of a node are dishonest in the majority, they may harm the reputation of a target or bolster its reputation without deserving if they are collaborating.

3.8 Trust and Reputation

As with trust, reputation is also crucial in interactions and relationships such as in social and economic exchanges. For instance, it is important for buyer-seller relationships as a good reputation helps a firm to have higher levels of trust by customers [88]. Reputation is defined as the belief about the extent to which a trustee is honest and concerned about the trustor [43]. Reputation can also be described as a characteristic attributed to a person by another person, such as a reputation for being courteous [93]. Reputation embodies historic information about attributes of a trustee, including honesty, reliability, and dependability [120, 33, 148]. This historic information is supposed to have predictive power on, i.e., is indicative of, the future behavior of the person to whom an attribute is ascribed [93].

A *reputation network* is a network, in which reputation information disseminates among a group of connected people or entities. Reputation networks could be *traditional* or *online* [17] based on the medium over which entities communicate and reputation information is shared. One distinction between traditional and online reputation networks is their emphasis on the type of *reciprocity*, which could be *direct* and *indirect.* Traditional reputation networks are based on direct reciprocity such that trustors (buyers in electronic reputation markets [17]) obtain the reputation information from their interactions with the trustee (traders in electronic reputation markets [17]). In online reputation networks, trustors cull reputation information from the experience of others, which corresponds to indirect reciprocity [17].

The effect of reputation information on trust is influenced by several factors [148]. Examples of these factors are the topology of a social network [63], the cost of obtaining and sharing reputation information [148], and the trustworthiness of the reputation information itself [148]. Also, the degree of uncertainty in the environment that a trusting relationship takes place affects the concern for reputation by parties. As such, a higher degree of uncertainty leads to greater concern for reputation [93].

Many trust models include reputation [148]. Yet, if the trust is predicted based on only reputation information, such a model will be vulnerable to strategic misuse by untrustworthy actors to build a good reputation so that they can exploit the trust of a trustee or not perform the final transaction [148]. This type of behavior is commonly observed through the end of trust games, a.k.a end-game effect, played in laboratory experiments conducted in behavioral economics domain (such as proven by Bohnet and Huck [16] in their trust game experiment), and in online auction sites [148]. Also, in the computer networks domain, there are strategic attacks in which an attacker builds a good reputation to stay in the network without being recognized and perform attacks at later stages. Hence, reputation-based trust frameworks have the same drawbacks for computer networks.

3.9 Trust and Context

Trust is a context-dependent or context-specific phenomena [183, 18, 103, 173, 118], as already stated in Section 3.1.3. Context-dependency of trust means that trust and its dimensions depend on the circumstance of an interaction and parties interacting [54, 124, 173]. Context is the conditions or institutional structures under which an encounter between a trustor and a trusted target takes place [173]. For instance, Mayer et al. [118] propose that trust antecedents, i.e., perceptions about ability, benevolence, and integrity, are affected by contextual factors, such as organizational policies and the perceived similarity between a trustor and a trustee. Mechanic and Meyer [124] found that the trust of patients in doctors and healthcare plans is affected by the conditions of patients, such as sophistication and characteristics of their illness, their needs and access to information, and the degree of risk perception. The reason for trust to be context-specific is that the gain and loss of a trustor from a trust relationship depend on the situation [183], which is captured by payoffs (gain and loss) in the expected trust formula in (1) [183]. As demonstrated by previous work, contextual factors for an encounter may affect the trust, such as the location (neighborhood) and the channel (e.g. Internet) of an encounter [173].

3.10 Trust, Risk, and Uncertainty

Risk is seen as a key factor for the development of trust and trusting behaviors [153]. Some level of risk is needed as a basis for the development of trust [93, 59, 102, 95] or for the trust to become operational [43]. When the perceived risk is high, trust becomes more important [66]. Even it has been regarded as a precondition for trust such that without risk, trust will not evolve, be relevant, or be required [33, 37, 118, 148, 93, 36, 111, 97, 35, 96]. Risk has been defined in different ways. One view defines the risk as the degree of uncertainty that a decision will lead to potentially significant or undesired outcomes for a trustor [165, 155, 148, 43, 127, 36, 132, 34, 35]. This definition treats risk and uncertainty as synonyms [148]. Another view defines the risk as a compound concept that embodies both perceived probability and magnitude (impact) of adverse outcomes [118, 148, 36]. Thus, according to this view, trust is required in situations where there is both the possibility of an undesired outcome and a considerable impact of it [148].

Mayer et al. [118] highlight the distinction between trust and risk-taking such that trust is the *willingness to be vulnerable* whereas risk-taking is actually becoming vulnerable [30]. There are propositions that focus on the effect of risk-taking on trust. For example, Stasiulis et al. [172] showed that risk-taking is a type of work that fosters trust among mental health care service teams, such as breaking hospital rules by clinic managers. There is another side of the coin where trust affects risk-taking behaviors or risk perceptions. Mayer et al. [118] assume that trust fosters risk-taking, i.e. intention to accept vulnerability results in taking actions to be vulnerable. Besides fostering risk-taking behaviors, trust helps to reduce [36, 132] or overcome [122] perceptions of risk and uncertainty in trust relations, such as that of consumers towards online vendors [122] and partners in a strategic alliance [36].

It is important to note the difference between risk (objective) and perceived risk (subjective). The prior is an inherent risk that can be objectively calculable, such as for lottery and card games whereas the latter is the belief about [132] or the estimation of objective risk by an individual [36]. Trust does not affect on objective risk inherent in situations but reduces the subjective/perceived risk [36]. It is also important to distinguish uncertainty from the risk [148]. As we discussed earlier, risk is associated with the “known” probability of adverse events [92, 148]. For uncertainty, adverse events are still possible but without knowing probabilities [92, 148, 65]. People try to reduce uncertainty to calculable risk [65]. As people usually do not have the time or ability to calculate expected probabilities and as trust provides short-cut probability calculations [111], people will resort to trust in uncertain situations [111, 65] to be able to continue cooperation and economic transactions [65]. For example, not having detailed knowledge about the skills of and motivations for actions by others results in uncertainty [37] and the need for trusting others.

3.11 Trust and Distrust

Trust and distrust are part of relationships in all social systems [103]. For stable social structures, they should both exist in a healthy dose [103]. Distrust is a concept that is distinct from trust [103, 121]. It is crucial to know whom to distrust as much as whom to trust [6]. Distrust is seen as even more important than trust prediction by some researchers because predicting distrust helps to mitigate the risk that is involved in trusting someone [118, 6]. Both trust and distrust are mechanisms for individuals to contain and manage social complexity and uncertainty in relationships [111]. Yet, they do this in different ways. Trust eliminates the consideration of an individual that another party will have undesirable conduct and makes desirable actions to be viewed as certain [111]. Conversely, distrust simplifies decision making and reduce complexity by allowing undesired conduct to be seen as more likely [111].

Although trust has been investigated in different domains, distrust has gained attention recently [6]. Most of the early trust models/trust prediction approaches either ignored distrust or simply accepted it as the absence of trust or being neutral [194]. However, some researchers ([28, 69]) argue that in the real world, distrusting is neither being neutral nor the mere absence of trust [6]. What distinguish it from the absence of trust or neutrality are characteristics that are specific to distrust. These characteristics include fear, unease, and pessimism and require precautions to be taken [103]. It is also crucial to note the difference between distrust and *violated trust*. As Sitkin and Roth [166] argue, violated trust is about violations by a trusted party specific to a particular task or context. If a violation spans across different contexts, i.e. violates fundamental values and undermines trust in different contexts, then it is regarded as distrust.

Some researchers view trust and distrust as opposite to each other and mutually exclusive [103]. According to this view, trust is the confidence about desired behaviors of a partner [38], whereas distrust is the confidence that a partner will display undesired behaviors, based on the knowledge about the capabilities and intentions of the partner [38]. In other words, distrust is about the expectation that an individual is not capable and will not behave responsibly [13]. In addition to expecting that a partner will not act in best interests, distrust may be associated with the expectation that a partner will engage in potentially harmful behaviors [62]. Yet, Lewicki and McAllister [103] argue that trust and distrust are not at the opposite end of a single continuum. Trust is not the opposite of distrust, and low trust is not the same as high distrust and vice versa [103]. They are distinct mechanisms to manage social complexity in relationships, but they can coexist [103]. As such, individuals can both trust and distrust each other for different experiences in their relationships [103]. Different elements affect the growth and decline of trust and distrust, but these elements develop together in the experiences of an individual [103].

3.12 Implications for IoT Networks

In this section, I discuss how trust-related concepts/propositions that we identified from our literature review could apply to IoT networks. Specifically, I group our discussions under the following topics: trust life cycle, factors affecting trust, and trust and risk relationships.

Next, in light of these, we propose a conceptual trust framework that could be referred to by network administrative domains as a guideline.

3.12.1 Trust Lifecycle

Just like trust evolution during a trusting relationship is a crucial concept in social sciences, it is also significant for IoT networks. We term trust evolution in IoT networks as *trust lifecycle*, where trust is initialized when the network is first deployed or a device joins the network for the first time. It increases, decreases, or stays constant based on the actions by IoT devices and interactions between them. The trust repair concept also applies to the trust lifecycle. If the trust of an IoT device decreases and a human intervenes, the trust gets repaired or reset. In the following sections, I discuss ideas in this direction.

3.12.1.1 Initial Trust Formation

Initial trust formation is an important challenge in trust relationships, as I discuss in Section 3.5. The same challenge applies to IoT networks. *How much should we trust a device just joining a network?*

In social sciences, one view is that interaction is a prerequisite for trust formation [101]. This may lead to the conclusion that initially without interaction, there is no trust or trust information. Based on this argument, we may consider setting the initial trust of an IoT device to zero if it is a single value. An opposite strategy could be setting the trust of a device on another device to the highest possible value if they are encountering the first time and only once. This idea reflects the findings of social science studies in which high levels of initial trust were observed for temporary teams [74]. This was found for teams that have not worked before and do not expect to do so in near future for a complex task [95]. We note that this strategy would be better to apply in computer networks that are not deployed for highly critical systems, such as military or healthcare. Otherwise, assigning high trust values by default could lead to high-risk scenarios.

Another view suggests depending on trust propensity (i.e., dispositional trust) for initial trust formation when there is a lack of interactions, as discussed in Section 3.1.4. We believe that dispositional trust could also have a significant role in IoT trust computations. We base our argument on discussions about human-automation interactions by Hoff and Bashir [73]. They discuss that individuals have varying levels of a tendency to trust automation (i.e., dispositional trust). Also, they argue that a human trustor is two-degree separated from an automation system as a trustee because the human trustor considers the human designer of a system when putting trust on it (two degrees of separation: human-automation → automation-human). Combining these discussions, we can think that there is a three-degree separation for device-device interactions in an IoT network (human(trustor)-device(used by trustor) → device(used by trustor)-device(used/designed by a trustee) → device(used/designed by a trustee)-human(trustee)). Thus, we may think of setting dispositional trust of a device towards other devices directly based on the dispositional trust level of its owner towards other people or technology. This maybe the same level for all devices in a network if they are all owned by the same trustor. Mindset type (fixed or growth) is also proposed as a dispositional factor for trust formation ([143]). Mindset relates to personality and is relatively a stable factor. It is not affected by interactions in a relationship, so it can be used for initial trust formation. This concept can be utilized for initial trust formation in IoT networks. When there is not enough evidence for assigning trust values in a network, we may use a predetermined mindset of devices and associated trust values for initialization. The determination of mindset for devices could be based on end-user preference or some network-based measures, such as node centrality, in/out-degree, etc.

We suggest another set of related strategies for assigning initial trust values for IoT networks. Bolton et al. [17] found that if the overall trust in an online market is positive, a buyer having a bad experience could tolerate it owing to the high general market trust. This can be utilized to assign initial trust values when two networks join for collaborative purposes. More precisely, we may consider a network as a marketplace and an IoT device as a buyer. For example, a trustor device in one of the networks could take the average of trust values of the nodes in another network and use it as the initial trust value for any device for the first time interaction. We note that the two networks do not have to share the trust values directly, as it may be a privacy concern. It can be done by a trusted third-party service such as one deployed in the cloud, assuming these networks are merged under a contract.

Alternatively, they can exchange trust-related data to be used to infer trust values, such as environmental and technical data suggested in [5] for an IoT trust framework. Bolton et al. [17] also found that some buyers decreased their trust level towards all sellers for future interactions after their trust has been compromised. Other sellers, whose trust was rewarded, continued to trust sellers in their future interactions We can utilize a similar strategy for the integration of multiple network administrative domains. We can consider two types of devices in a domain as pessimistic and optimistic, and we can assign initial trust values for devices in other domain according to their trust view, e.g., zero for pessimistic devices and a value higher than a medium value in a trust scale for optimistic devices (such as 0.7 in a trust scale of $[0, 1]$).

3.12.1.2 Trust Development after Initialization

This phase covers the time after a network is deployed and initial trust scores are assigned to devices. As discussed previously, trust is a dynamic process and evolves, i.e., it increases, decreases, or stays stable sometime, based on different factors. With the trust development concept, we refer to trust evolution. In this section, we speculate on ideas concerning it. We organize them concerning significant concepts for trust development after initialization.

Trust Transfer: I discussed trust development concept earlier in this chapter, in the context of social sciences. Trust transfer is also crucial for trust computations in IoT. For example, trust propagated along paths in a network is known as referral trust in Evidence-Based Subjective Logic and trust transference is termed trust propagation [5] (see Chapter 4). Stewart [173] proposed a trust transfer model for online business settings. They found that the perceived interaction and perceived similarity between two websites increase the trust in a less-known website if the better-known website is trusted. This idea may be applied to trust transfer in IoT. As such, the similarity of two devices (e.g. contextual and security-related features) and their interactions (pattern of interactions with other devices in the network) could be used to transfer trust. This could be done both for transferring trust from well-known trusted devices and distrust from well-known distrusted devices to less known devices. Another implication for trust transfer in IoT could be considered based on Reusen and Stouthuysen's [147] trust transfer study. They showed that third-party information, having two levels as neutral and favorable information towards a trustee, affect competence and goodwill trust differently. We may think about applying the two levels of third-party information for computing trust transfer in computer networks. Neutral information could be only knowing the existence of a link between two devices or the number of times they interact with each other. Favorable information could be more detailed,such as the result of their interaction as feedback (satisfied or not) or as the trust of a trustor node is honored or cheated. Also, again based on Reusen and Stouthuysen's [147] study, grouping trust-related attributes as competence- and goodwill-related may be worth exploring for IoT. For example, the type of the communication link used by devices could be a competence-related attribute as it affects the reliable transfer of data, whereas the freshness of the cryptographic keys used to encrypt communicated data may relate to goodwill as an old key may lead to attacks to the device and imply a compromised device. This grouping may be useful also because the effectiveness of trust repair efforts is affected by the type of trust violated as discussed in Section 3.6.

Combining Real-time and Historical Trust Scores: When we talk about trust changes over time, an important problem to be addressed by trust frameworks for IoT is *how to combine, i.e., how to set weights for historical and real-time trust values.* Inspired by the findings of studies and propositions discussed in the articles we reviewed, we speculate on the ideas as potential solutions to address this problem. First, in their study to investigate the effect of reputation information flow on trust development in electronic reputation mechanisms, Bolton et al. [17] found that buyers weigh recent observations higher than older ones for trust formation. Based on this finding, we can think that recent trust IoT network. For example, the exponentially weighted average method is used for combining historical and real-time trust values [5]. Instead of setting coefficients of 0.5 for both types of trust scores, the coefficient for the real-time trust score could be set to a value higher than 0.5. Second, people tend to weigh negative information more than positive information [182]. For example, when the economy is going bad, people may give more importance to it to discount the trust of the government [72]. Also, Bolton et al. [17] found that some groups of buyers tend to give more importance to their negative experiences. Hoff and Bashir [73] also argue that positive and negative past experiences may have different influence on trust in automation. We may consider these propositions and findings for weighting trust scores based on the outcome of interactions between IoT devices For example, if a device exploited the trust of another device in one of the past experiences, that record could be weighted more than other positive records. This decision of weighing negative or positive experiences more can be made by the user of a trust framework.

Trust Repair: In connection with the discussion of historical and real-time trust values above, we can consider setting trust values based on trust repair actions. For example, we can consider reset to a device by a company, firmware upgrade, or any kind of maintenance as trust repair actions and reset the historical trust value. As trust repair is needed after trust violations and is related to distrust towards violator(s), focusing on distrust in computer networks is also crucial. A potential distrust mitigation mechanism in a relationship after the trust fails is the *transformation of distrust into trust.* Lewicki and McAllister [103] argue that distrust relationships can be effectively managed if they are transformed into trust relationships that allow functional interactions within constraints. They explain this with a practical example from the White House. Leon Panetta, as the Chief of Staff, turned his distrusting relationship with Dick Morris, who was the political adviser to President Clinton, into a trusting relationship by setting out clearly specified parameters and rules such that Morris could work within these constraints as a trusted and invaluable partner. We can consider the transformation of distrust relationships into trusting ones by constraining them for computer network settings. For example, when a node demonstrates untrustworthy behaviors and is distrusted, we may want to limit its actions in the network (instead of shutting it down or kicking it out), such as prohibiting the use of certain communication protocols and downloading files, or enforcing software patches, refreshing keys, etc.

3.12.2 Factors Affecting Trust in IoT

In this section, I discuss factors that may affect trust in IoT drawing upon our review of trust in social sciences. Recalling that the perceived social presence of a website is positively associated to trust [88, 101] and interactivity is the main component of social presence [180], we may think about social presence as a factor in trust development in IoT networks. The social presence or absence of devices can be considered as their being active or inactive. Hence, interactive devices can be rewarded with higher trust. The trust of devices that join the network and stay inactive for a long time after a series of interactions can be degraded (see Chapter 6 for more details about it).

Hendriks et al. [71] investigated whether the trustworthiness of a researcher and the credibility of a study by the researcher is affected by who replicated the original study and if the replication was successful or not. We may consider an analogy for IoT (see Chapter 6 for more details). For example, we can think that an IoT device and a measurement reported by it are analogous to an author and a scientific study, respectively. We can consider replication study, in IoT context, as a *measurement replication* that corresponds to multiple reporting of a measure, such as temperature. It may come from the device itself in a predefined short time window or from other devices, so replication author here will be *replicating device*, i.e., device itself or other devices. *Replication success* could be thought of as the deviation from other reported measures within the predefined time window. For example, if it is below a threshold, we can say that it is successful, if it is above a threshold, it is a failure. Hence, for IoT networks, we may think that the *trustworthiness of a device* and the *credibility of a reported measure* are different concepts, and they may be affected by the replicating device and replication success[6].

Within a similar direction, Alarcon et al. [7] showed that the source of a code repair, i.e., a human programmer vs. an automated code repair software, has an effect on the trust towards the code repair. We may think about similar concepts for trust in IoT. More precisely, we may think that different sources of trust repair in a network would affect trust values differently. For example, we may think of the source of repair as the type of repair, such as hardware-based trust repair actions (e.g. increasing the processing or storage capability of a node) or software-based trust repair actions (e.g. rebooting the node or getting over the air updates, etc.). Then we may model trust changes based on the type of repair action.

[6] Device trustworthiness and measurement credibility are different concepts because, for example, a device may be hacked by an attacker (untrustworthy device) but still may report correct measurements (credible measurement) for hiding its malicious behavior until an attack

3.12.3 Trust and Risk Relationships

As discussed in Section 3.10, the relationship between trust and risk is an important topic in social sciences literature. There are arguments for the two sides of this relationship. One group claims that trust positively impacts risk-taking behavior, whereas the other group argues risk-taking behaviors positively impact trust. We contend that these arguments may apply to IoT networks. For example, if we consider that the devices within a network trust each other, then they share sensitive data as a risk-taking behavior. Another view that focuses on the other direction claims that risk-taking behaviors promote trust. When we consider this view for IoT networks, we can think that risk-taking behaviors by devices , such as using stronger cryptographic measures (risk here could be faster battery consumption for IoT devices), could lead to higher trust towards them .

Another crucial concept is the calculative view to trust. It considers trust as the calculation of risk probabilities, gains, and losses from an encounter with a trustee, as discussed in Section 3.4 and 3.10. According to Coleman [29] and other economics researchers, trust is the expected payoff of a trustor from a trust relationship. That is, it is the combination of the perceived probability of an undesired outcome (perceived probability of trustee being trustworthy or untrustworthy) and the impact of that outcome (gain and loss from trustworthy and untrustworthy behaviors) (Equation (1)). This view may have an implication for computing trust and making trust decisions in IoT. To our knowledge, trust has been approached typically as a probability with different scales in networks (e.g., values between [0,1], [-1,1],[0,100], or [-100,100]). For example, Subjective Logic (SL) [77] is one such trust model. It considers probabilities for trust, distrust, and uncertainty to represent the *trust opinion triplet* of a trustor node in a trust network (see Chapter 4 for more details about SL). Considering only probabilities may not be enough for establishing trust relationships in IoT networks. Hence, this implies an enhancement or a complementary step on the SL (or any probability-based trust computation in networks). The SL opinion triplet of a trustor node can be plugged into Equation 1 as p. We can consider the gain and loss of the trustor node (G and L in equation (1)) from its relationship with a trustee node and calculate the expectation of the trustor node ($E(R)$ in Equation (1)).

3.12.4 A Conceptual Trust Framework

In this section, I present the conceptual trust framework we proposed for multiple IoT network administrative-domains, based on our literature review in [3] and prior discussions in this section (Section 3.12). Fig 3 displays the proposed framework. We propose *a hierarchical relationship* between the duties for managing trust relationships among administrative domains. At the lowest level, there are *individual administrative domains* that are responsible for their trust management. Above individual domains is the *cross-domain layer* for inter-operation between two administrative domains. This layer is about tasks for trust management and shared responsibility in a pairwise matching of individual domains. At the top, there are trust management activities that concern all the domains. Next, I elaborate on these three layers and their areas of responsibility concerning the trust.

3.12.4.1 Areas of Responsibility of Individual Administrative Domains

We propose four main areas of trust-related responsibilities for an *individual administrative domain.* These are *trust objectives, trust management, trust computation,* and *trust and risk interplay.*

We consider two *trust objectives* for IoT networks (inspired by discussions in [172] about objectives of health clinics governing bodies). One objective is *trust in operations* or *trust in data.* Another objective is *trust in trust framework*, which corresponds to better or more accurate trust metrics. Concerning this, we can consider the criteria proposed by Akilal et al. [6], for evaluating trust/distrust prediction solutions in social networks. They emphasize that a trust/distrust prediction solution should be *accurate, fast,* and *robust.* The *accuracy* of a solution requires predicting trust values as close as possible to the decision by a user in social networks. The *speed* is about time that is required for the prediction of trust by an algorithm, which should be very quick because users' actions depend on trust predictions. We argue that this condition may be even more critical for some IoT networks, such as those deployed in healthcare scenarios, that require almost continuous operation of devices with minimal interruption. The *robustness* means that the accuracy of trust processing/prediction should not be affected by the sparsity of a network, i.e., missing most of the links between

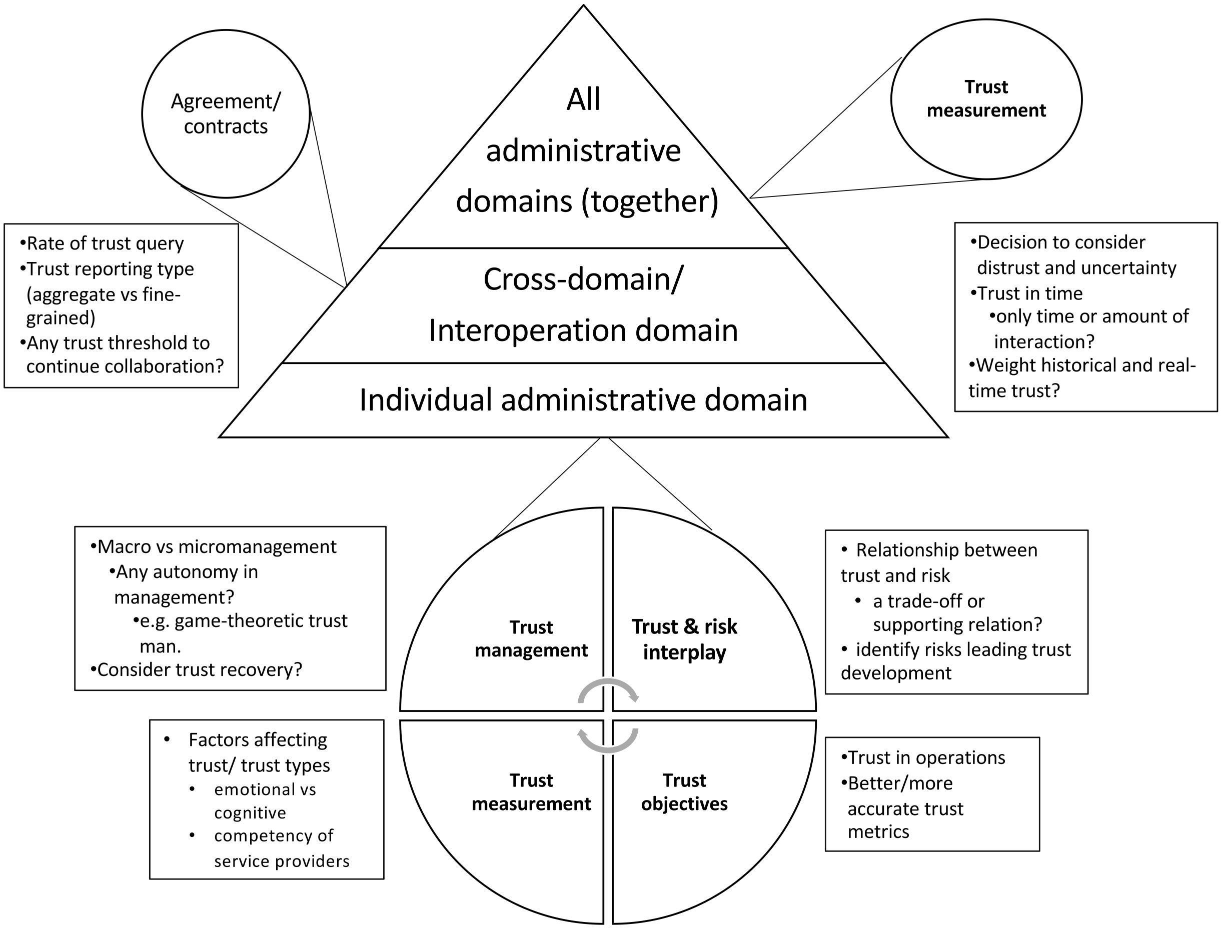

Agreement/ contracts
All administrative domains (together)
Trust measurement
•Rate of trust query
•Trust reporting type (aggregate vs fine-grained)
•Any trust threshold to continue collaboration?
Cross-domain/ Interoperation domain
•Decision to consider distrust and uncertainty
•Trust in time
•only time or amount of interaction?
•Weight historical and real-time trust?
Individual administrative domain
•Macro vs micromanagement
•Any autonomy in management?
•e.g. game-theoretic trust man.
•Consider trust recovery?
Trust management
Trust & risk interplay
• Relationship between trust and risk
• a trade-off or supporting relation?
• identify risks leading trust development
• Factors affecting trust/ trust types
• emotional vs cognitive
• competency of service providers
Trust measurement
Trust objectives
•Trust in operations
•Better/more accurate trust metrics

Figure 3: A conceptual trust framework for multiple IoT network administrative domains

users in social networks context.

We argue that *trust management* could be at two levels as *micro* vs. *macro* (inspired by discussions in [172] about health clinic manager characteristics). For instance, querying trust levels or trust-related attributes of IoT nodes very frequently could be micromanaging trust (Low query rates correspond to macro-management of trust, in this case). This might have a negative impact on energy efficiency for nodes as they need to communicate frequently. Another view to micro-management of trust could be looking at immediate neighbors of a target node for reputation information that will be used in trust computations, as proposed in [6] for social networks. Yet, this approach may have a drawback such that trust values are biased to reputation information provided by/inferred from neighbors of a node. If neighbors are not benevolent, the trust of a benevolent target node may be reduced intentionally, or vice versa for a dishonest target node. Another issue regarding trust management in IoT is whether autonomy could be considered, such as game-theoretic trust management similar to trust games in economics studies as in [21, 10, 16] [7]. Also, we argue that the decision of considering the recovery of trust in devices in a network is a type of management-related activity (see example trust repair actions discussed earlier in Section 3.12.1).

In regards to *trust computation*, we consider factors affecting trust and trust types, such as emotional and cognitive trust, which are cited frequently as a basis for personal trust in social sciences literature [88, 124, 172, 59, 73, 100]. Earlier in Section 3.12.1 and 3.12.2, we discussed factors that may impact trust values and trust measurement in computer networks, as implied from social science disciplines. Here we add that service providers/device vendors may affect the trust values of a device due to their competency (or benevolence and integrity (ABI model in Section 3.1.4). Change of trust in devices over time, i.e., trust life cycle, is also in the scope of trust computations. Activities that can be taken by an individual administrative domain related to trust lifecycle have been discussed earlier in Section 3.12.1. As a reminder, the trust life cycle consists of initial trust formation and trust development afterward, which includes trust transfer/transitivity , combining historical and recent trust values, and trust repair activities.

We have already acknowledged the significance of risk and its relationship to trust in

[7] More detailed game-theoretic trust discussions are presented in our review paper.

IoT in Section 3.12.3. For the *trust and risk interplay* area of responsibility, administrative domains should specify possible risks and if there is a contradictory or supporting relationship between those risks and the trust. In other words, they should consider if a risky state/action fosters or prevents trust development. For example, using stronger cryptographic measures could include the risk of higher power consumption, but may lead to increased trust.

3.12.4.2 Areas of Responsibility of Cross-domains and All Domains

Considering that multiple network administrative domains collaborate to deliver some services, we assume they do this through formal contracts, informal agreements, or interactions. For cross-domain areas of trust responsibilities, we imagine that two domains should mutually agree about issues such as how the trust of devices will be reported, i.e., an aggregate or fine-grained measure. This can be decided based on the type of collaboration between domains and their privacy concerns, i.e., if they would like to share less or more trust-related information. We remind that they do not have to share the trust values of their nodes directly. It can be done by a trusted third-party service such as one deployed in the cloud and that service can share trust values after applying privacy protection algorithms. A domain can exchange trust-related data that can be utilized by the other domain to calculate trust values of devices, such as environmental and technical attributes. The frequency of query and sharing of these data, i.e., trust values or trust-related data, should also be jointly decided because it may be an issue for service availability for the domain that provides these data. Finally, deciding if there will be any trust threshold to continue collaboration is another cross-domain-related issue. It may be the case that devices in an administrative domain have low trust, but the other domain has no other alternative to get services and still want to collaborate.

For areas of trust responsibilities that concern all domains in a collaboration, we argue that the main issues will be about trust computations. A common representation of trust among domains could be a potential candidate for decisions to be made by domains. For instance, if distrust and/or uncertainty should be concerned (as in [6, 77]) or just trust levels will be reported. Another example could be to decide if historical records of trust will be stored and shared by domains. If so, if and how historical and real-time values could be combined (see Section 3.12.1 for ideas).

4.0 Automated Trust Computation in IoT: Multiple Attributes and Logic

In this chapter, we propose a trust measurement framework for a single IoT network administrative domain that automatically computes the trust of “things”, which we call *MADM-EBSL Framework* (Multi-Attribute Decision Making (MADM) - Evidence-Based Subjective Logic (EBSL) Framework). In the sequel, I present the challenges of trust computation in IoT environments and the motivation for the proposed framework, followed by the background information for the MADM and EBSL. Next, I present the proposed trust measurement framework and discuss evaluation results.

4.1 Motivation and Background

Trust mechanisms are fundamental for people to overcome perceptions of uncertainty and risk in using IoT services and applications [49]. This uncertainty and risk in IoT environments arise from the potential for malicious activity, such as man-in-the-middle of a link, Distributed Denial of Service (DDoS), physical node compromise, etc. Even when things behave benignly, they may have inherent environmental limitations and variations in behavior (inexpensive vs. robust to wear and tear) that impact trust in the reported information. Thus, the *trust* in the devices, the data they report, and the actions they take should be measured through a trust measurement framework.

There are several challenges in trust computations in IoT. First of all, there is a vast increase in the number of IoT nodes and communication among them [64, 40, 11, 5]. Many transactions occur primarily among things with support from computation and storage in the cloud [5]. Also, given that human intervention is expensive, trust computations need to be *automatically* performed without human intervention. In addition, the trust in the data delivered by “things” depends on several factors, such as contextual/environmental conditions and technical/protocol-dependent features (key freshness, communication link characteristics). If the trust in reported measurements is a single number, it needs to capture various conditions as such. Thus, there is a need for a trust measurement method that considers the communication among for "things", environmental and security-related factors, and the network topology.

There are *inherent* trust issues in the things sensing the environment due to several limitations. For instance, some sensors may operate correctly under some environmental conditions, but the trust in the data they sense may degrade under others. The characteristics of nodes may not, however, be easily quantifiable since the trust may not *monotonically* change with the characteristic. There are technical aspects of nodes that influence their trust. A thing whose keys used for cryptographic protocols have remained unchanged for a long time is more vulnerable to compromise than a thing that has its keys refreshed frequently. We consider both of these types as impacting what we call the *functional trust* (FT) of a thing. Trust needs to take into account the *beliefs* of things on each other (especially neighbors) and the belief of the cloud infrastructure on things based on topology, the delivered data, and even potential external data sources. We call this *referral trust* (RT). For example, a thing may reduce its trust in a neighbor if both sense the same physical phenomenon, e.g. the temperature, and there are significant differences. The cloud may adjust its trust on a node sensing temperature if the values are very dissimilar to the coarse-grained values reported by an external trusted source.

Based on the discussions above, in our research we attempt to answer the following research question:

> ***RQ.*** *How can we determine a temporal trust in things based on remote knowledge of the network topology, inherent limitations of sensors, generated data, and potential misuse of things or their connections, without continual human intervention?*

To address these challenges, we propose the *MADM-EBSL Framework* that automatically computes the trust of "things". We use Multi-Attribute Decision Making (MADM) to measure functional and referral trust of an IoT node. We adopted the Evidence-Based Subjective Logic (EBSL) [167] to account for uncertainty in trust values and trust transitivity in a trust network of "things". Referring back to the perspectives that I frame the existing trust measurement solutions in Chapter 2, the MADM-EBSL considers all key characteristics of trust, i.e., multi-dimensionality, context-specificity, dynamism, and uncertainty. Regarding other perspectives, it is a central, transitive-global, and fully-automated trust measurement scheme. In what follows, I present the preliminaries that the MADM-EBSL is built upon.

4.1.1 Multi-Attribute Decision Making (MADM)

Decision making (DM) is the process of modeling a decision maker's preference on a set of competing *alternatives (a.k.a options)* concerning a set of *criteria*, which are usually in conflict. When alternatives are evaluated based on more than one criterion, the problem is referred to as Multi-Criteria Decision Making (MCDM) [80]. There are two classes of MCDM methods: MADM and Multi-Objective Decision Making (MODM) [75]. The former deals with a finite set of alternatives, while the latter is suitable for continuous DM problems where there is an infinite number of alternatives [80]. In our trust framework, we use the MADM approach since we have a finite set of alternatives and well-defined *attributes*. We present a detailed definition of our DM problem for trust computation in Section 4.2.4.1. Note that there is no clear distinction between *criteria* and *attributes* in MCDM literature, so we use them interchangeably.

Attributes in MADM can be grouped under three categories based on the utility gained with respect to the attribute value [190]:

- *Benefit:* Benefit attributes provide increasing monotonic utility, meaning that the higher attribute values are preferable.
- *Cost:* Cost attributes provide decreasing monotonic utility, so lower attribute values are preferable.
- *Non-monotonic:* Non-monotonic attributes do not have a monotonic utility change with respect to their values. The most preferred value of a non-monotonic attribute falls into a point between the range of the attribute values.

Examples of these attributes are efficiency, cost, and room temperature, respectively [80]. Attributes can also be categorized into two classes as ordinal and cardinal attributes [80]. Ordinal attributes do not have a numerical scale such as ratio or interval, so their values are compared by their rank orders such as very high, average, low, etc. Cardinal attributes can have an interval or ratio scale and can be assigned with numerical values.

Concerning attribute information processing, MADM approaches are classified as compensatory and non-compensatory methods [76]. We adopt a compensatory method in our DM problem, which, unlike the non-compensatory methods, allows a trade-off between attributes [190]. In particular, we utilize the Simple Additive Weighting (SAW) as a widely accepted compensatory scoring model [190] in the proposed trust framework, which ranks alternatives and selects the ones with the highest score. It is one of the simplest models but produces results that are very close to more sophisticated DM methods [76, 131].

4.1.2 Subjective Logic (SL)

We use Evidence-based Subjective Logic (EBSL) [167] to compute the trust opinion for a measurement provided by an IoT device in a TN. As the EBSL is built on the pillars of Subjective Logic (SL) [77], I first overview SL in this section. I present the background about EBSL in Section 4.1.3.

SL [77] is a framework that is used for logical reasoning with uncertain propositions, i.e., it combines concepts of standard binary logic and probability calculus. SL also uses elements from the Dempster-Shafer belief theory [77]. Thus, SL accounts for the uncertainty in an *opinion* towards a *proposition.* In SL, an opinion is represented as a triplet, —*opinion triplet*, $\omega = \{b, d, u\}$, where b, d, and u respectively represent the belief, disbelief, and uncertainty of opinion and $\forall b, d, u \in [0, 1]$, $b + d + u = 1$.

SL provides two fundamental sets of operators to combine opinions, which are *logical* and *evidential* operators. As our MADM-EBSL framework is based on the EBSL that adapts evidential operators of SL, the MADM-EBSL uses evidential operators for combining opinions. Logical operators are for combining opinions of an agent about multiple propositions. They include *propositional conjunction*, *propositional disjunction* and *negation*, which are the applications of standard binary logic operators AND, OR, and NOT on SL opinions. Evidential operators are for combining opinions of multiple agents about a single proposition. They include *consensus* and *discounting* operators. Consensus reflects combining evidence from multiple sources to form an opinion and is calculated using Equation 2. Discounting reflects the transfer of evidence among entities to form an opinion and is calculated using

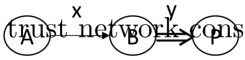


Figure 4: A simple trust network consisting of two nodes

Equation 3. The MADM-EBSL utilizes consensus and discounting for trust propagation and aggregation in a trust network of "things", following [167].

$$x \oplus y = \frac{(x_u y_b + y_u x_b, x_u y_d + y_u x_d, x_u y_u)}{x_u + y_u - x_u y_u} \tag{2}$$

$$x \otimes y = (x_b y_b, x_b y_d, x_d + x_u + x_b y_u) \tag{3}$$

where x, y are two opinion triplets, $x_b, x_d, x_u, y_b, y_d, y_u$ denote belief, disbelief, and uncertainty components of opinion x and y, respectively.

4.1.3 Evidence-based Subjective Logic (EBSL)

As discussed earlier, the EBSL adapts evidential operators of SL for combining opinions. Also, it combines SL with a graph model to compute the trust of nodes in a trust network (TN). For explaining the basic concepts, we present a toy example of a TN in Figure 4. In this TN, A and B represent two IoT devices: *thing* A and *thing* B. P is a proposition about which B forms an opinion. In our case, P is *"a data item d_t that is collected at time t has the value $v(d_t)$"*. In the figure, x represents the opinion that A has about the trustworthiness of B in providing recommendations, which is known as *referral trust (RT)* in EBSL. y represents the opinion that B has about the trustworthiness of the data items it measures, which corresponds to *functional trust (FT)* in EBSL.

Opinions on the path from a node to another node form a *trust chain.* In a valid trust chain, there must be an FT on the last edge in addition to RTs on previous edges [167]. In a TN, the trust of a target node is computed by aggregating trust opinions of paths (trust chains) between the target node and a source node. This operation is referred to as *aggregation of trust* and performed using *consensus operator* ($\oplus$). Note that the consensus operator of the EBSL is the same as the original consensus operator of SL, and calculated by Equation (2). The trust opinion of a chain is updated every time each node is visited along the path. This update operation is referred to as the *propagation of referral trust*

Notation	**Meaning**
MT	$= \{mt_1, mt_2, ..., mt_p\}$: a set of measurement types
IM_{mt}, IM_{AB}	An initial matrix constructed for mt and for the edge between node A and B
D	A decision matrix obtained from IM_{mt}
$A_i =< ID, d_t^{ID}) >$	An alternative in D, which is a pair of ID of a "thing" and a data item d_t^{ID} collected by it at time t
X_j, w_j	An attribute in D and its weight
x_{ij}	Value of attribute j for alternative i
r_{ij}	Normalized value of x_{ij}
TN	Trust network
n_s, n_d	Source and destination nodes in TN
tc	A trust chain in TN
$< n_A, n_B >$	An edge between node A and B
T_{tc}	The trust opinion on a tc
OT_{dest}	Overall trust opinion for n_d
$RFT(d_t^{ID})$	Real-time FT score of d_t^{ID}
$RFD(d_t^{ID})$	Real-time functional distrust score for d_t^{ID}
DI_{ID}	Distrust interval of a device with given ID
RFT_t, HFT_t, HFT_{t-1}	Real-time and historical FT values of a device at time t and t-1, respectively
ω^{ID}_{P,RF_t}	Opinion triplet of a device with given ID on proposition P, representing RFT_t
ω^{ID}_{P,HF_t}, $\omega^{ID}_{P,HF_{t-1}}$	Opinion triplet of a device with given ID on proposition P, representing HFT_t and HFT_{t-1}, respectively
RRT_t, HRT_t, HRT_{t-1}	Real-time and historical RT value of an edge $< n_A, n_B >$ at time t and t-1, respectively
ω^{A}_{B,RR_t}	Opinion triplet of node A on node B, representing RRT_t
ω^{A}_{B,HR_t}, $\omega^{A}_{B,HR_{t-1}}$	Opinion triplet of node A on node B, representing HRT_t and HRT_{t-1}, respectively

Table 2: Notations

and performed using the *discounting operator* ($\otimes$). *Opinion discounting* represents the flow of information from one node to another in a TN. Note that the discounting operator of the EBSL is not the same as the original consensus operator of SL. The EBSL extends SL with a new *opinion discounting* operator ($\boxtimes$) as Škorić et al. [167] identify a number of basic problems of the discounting operator in SL. Namely, it has a *double-counting* problem which requires converting a TN into a "canonical form". Adopting the discounting operator proposed in the EBSL results in a consistent SL algebra as it resolves the double-counting problem of the original operator. The discounting operator of EBSL ($\boxtimes$) is expressed as follows:

$$x \boxtimes y = g(x).y = \frac{(g(x)y_b, g(x)y_d, y_u)}{(y_b + y_d)g(x) + y_u} \tag{4}$$

where x, y are two opinion triplets, y_b, y_d, y_u are belief, disbelief, and uncertainty components of opinion y, and $g(x) \geq 0$ is a scalar which denotes the proportion of evidence that is transferred from a node to another node. $g(x)$ can be selected arbitrarily, and suggested options are x_b or $\sqrt{x_b}$ where x_b is the belief for opinion x [167]. We use $g(x) = x_b$ for our computations.

4.2 MADM-EBSL Trust Measurement Framework

In this section, we present our proposed trust computation mechanism. Table 2 provides a list of notations and abbreviations we use. We first explain the overall process of trust computation (Figure 6). Next, we present an algorithm for computing the OT opinion for a measurement reported by a target "thing". In the sequel, we explain how we obtain FT and RT components of the OT. Finally, we explain how these two are combined using the EBSL.

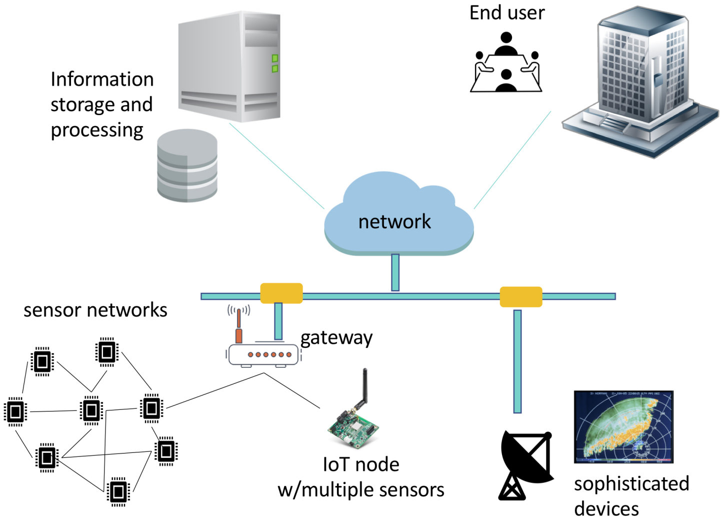

Information storage and processing
End user
network
sensor networks
gateway
IoT node w/multiple sensors
sophisticated devices

Figure 5: System model of the proposed trust computation mechanism

4.2.1 System Model

The main components of our proposed trust measurement framework are illustrated in Figure 5 with resource-constrained "things" and more sophisticated devices such as a gateway. These components form a TN. The Gateway collects measurements from "things" and sends them to the cloud server quasi-periodically. The transferred data include measurements of interest, such as humidity, wind, water velocity, etc., and the data used for trust computations. We elaborate on the data used for trust computation in oncoming sections. The transmission of the data between IoT devices and from IoT devices to the cloud server can be realized by using different wireless technologies (see Table 8). Note that secure protocols are used in the proposed system model, so the security of communicated data depends on the deployed protocols and infrastructure.

When the cloud server receives data from the gateway, it records the data items and trust-related measures into the database together with a randomly assigned (but known) ID for each IoT device. Note that IDs are generated by the cloud, so an IoT device cannot imitate this number during the data collection process. From these periodic measurements, the cloud server computes a *real-time functional trust (RFT)* score for a node (referred to as destination node) and a *real-time referral trust (RRT)* score for each pair of nodes (or each edge) in the TN. These real-time scores are then converted into SL opinions by applying a transformation that we explain in Section 4.2.4.3. The cloud server keeps historical records for each IoT device and fuses the RFT of a device with its historical FT (HFT). The result of this fusion is an SL opinion triplet that we use to represent FT (opinion y in Figure 4). Similarly, the RRT of each edge is fused with its historical RT (HRT) to obtain an RT opinion triplet. The *Overall trust (OT)* for a measurement reported by a target IoT device (destination node) is then computed by combining FT and RT values on the path from a source device to the target device in the TN [1].

4.2.2 The Overall Process of Trust Computation

I explain the process of computing OT_{dest} at a high-level in this section. I present details in Sections 4.2.4 and 4.2.5. Figure 6 shows the overall process. The data items and trust-related attributes are collected to measure both FT (sensors embedded to thing A) and RT (between thing A and B). We use these attributes and data as input to the MADM method and obtain trust scores and a distrust interval DI_{ID} from them. Steps of MADM include the construction of a decision matrix D, normalization of the values in it, computing weights that reflect the relative importance of attributes, and applying SAW on the normalized matrix using the weights. Then, we convert scores from MADM into a distrust interval DI_{ID} that is then used to get opinion triplets that represent RFT and RRT. We combine RFTs (or RRTs) using an exponentially weighted moving average (EWMA) method to obtain the HFT of a measurement as an opinion triplet (or HRT opinion for referral trust). Finally, we combine the opinion triplets on the paths from a source node to a destination node for obtaining OT_{dest} in a given TN.

Note that an IoT device can be embedded with multiple sensors, each measuring different *measurement types (mt)*, such as humidity, water velocity, or precipitation. A *measurement, data item,* or *observation*, refers to a single measurement collected by a single sensor of an IoT device. In our trust computation algorithm, we assume that the OT of an IoT device in a TN is queried for a single mt. Hence, we compute OT_{dest} as an opinion triplet for a single IoT device and a single mt. If we would like to let the user query the *trust of an IoT*

[1] The trust computation module may be accessed by end-users through a web application interface to query the trust of IoT devices in the TN.

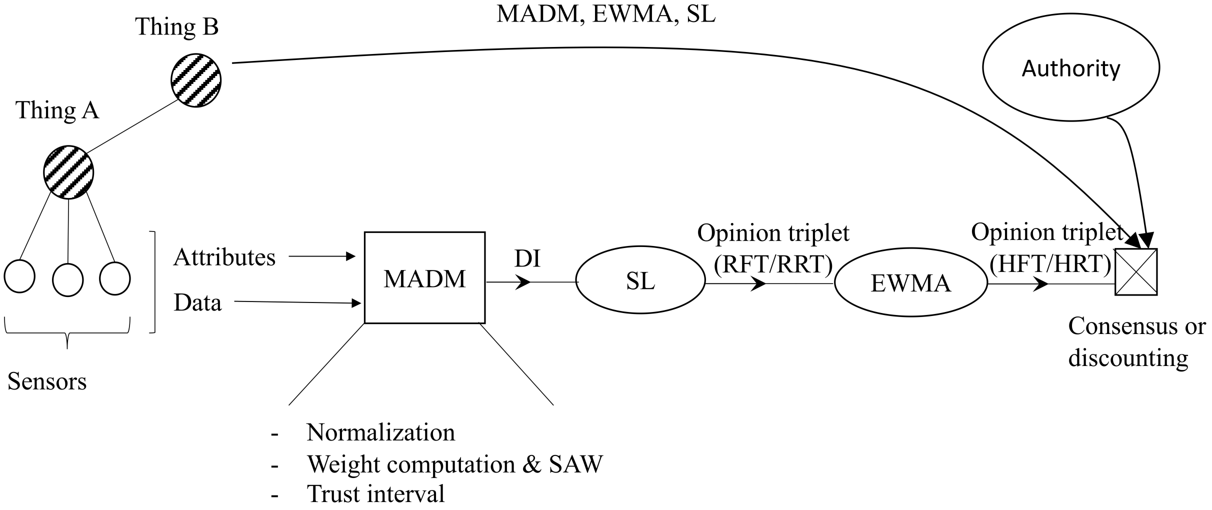


Figure 6: Overall trust computation process

device for all the mts, we can combine each opinion triplet for each mt using the *consensus* operator of SL [77]. Similarly, there may be multiple IoT devices measuring the same mt. In this case, if a user wants to query the *trust of an mt*, we can combine opinion triplets from the devices providing observations for the same mt.

4.2.3 Trust Computation Algorithm

We present Algorithm 1 to compute OT_{dest} for a destination node n_d. The first step is the initialization of $RFT(d_t^{ID})$. Next, the algorithm constructs a table for real-time data items collected by all the "things" (bigTable in line 2). Then, it finds and groups together d_t^{ID}'s collected for mt by all "things" measuring mt (line 3) to obtain an initial matrix (IM_{mt}) as a representation of the decision making problem (DMP). The mt of interest is given as an input to the algorithm, amongst the ones provided by the destination node n_d. Next, the algorithm constructs D from IM_{mt} (line 4), normalizes attributes in D (lines 5-6), and computes an RFT score for each d_t^{ID} in D (lines 7-10). After this, a DI_{ID} is computed from the list of $RFT(d_t^{ID})$'s for the n_d represented by the ID (line 11). DI_{ID} is then converted to an SL opinion triplet (ω^{ID}_{P,RF_t}) that represents the opinion of n_d on the trustworthiness of its measurements (line 12). Next, real-time and historical opinions for FT are combined using the EWMA method (line 13).

Algorithm 1: Compute Trust

```
   Input  : $TN$, $n_s$, $n_d$, $mt$
   Output: $OT_{dest}$
1  Initialize: $RFT(d_t^{ID}) = 0$ ;
2  $bigTable \leftarrow$ generateBigTable();
3  $IM_{mt} \leftarrow$ groupDataItems($bigTable$, $mt$);
4  $D \leftarrow$ constructD($IM_{mt}$);
5  for $X_j \in D$ do
6      normalize($X_j$);
7  for $d_t^{ID} \in D$ do
8      for $X_j \in D$ do
9          $RFT(d_t^{ID}) \leftarrow RFT(d_t^{ID}) + w_j \times r_{ij}$;
10     $RFTList(ID) \leftarrow$ add $RFT(d_t^{ID})$;
11 $DI_{ID} \leftarrow$ computeDI($RFTList(ID)$);
12 $\omega_{P,RF_t}^{ID} \leftarrow$ convertToOpinion($DI_{ID}$);
13 $\omega_{P,HF_t}^{ID} \leftarrow \alpha(\omega_{P,RF_t}^{ID}) + (1-\alpha)(\omega_{P,HF_{t-1}}^{ID})$;
14 for $tc$ $in$ $TN$ do
15     for $< n_A, n_B >$ $on$ $tc$ do
16         $\omega_{B,RR_t}^{A} \leftarrow$ computeRT($n_A, n_B$);
17         $\omega_{B,HR_t}^{A} \leftarrow \alpha(\omega_{B,RR_t}^{A}) + (1-\alpha)(\omega_{B,HR_{t-1}}^{A})$;
18         $T_{tc} \leftarrow T_{tc} \boxtimes (\omega_{B,HR_t}^{A})$;
19         if $n_B = n_d$ then
20             $T_{tc} = T_{tc} \boxtimes (\omega_{P,HF_t}^{ID})$;
21     $OT_{dest} \leftarrow OT_{dest} \oplus T_{tc}$;
22 return $OT_{dest}$;
```

Lines 14-21 are for TN-related operations. First, every edge on every trust chain from n_s to n_d is assigned with an opinion triplet ω_{B,RR_t}^{A} (line 16). Then, similar to RFT, the EWMA method is used for combining the real-time and historical RT opinion for each edge (line 17). The next step is the propagation of referral trust (i.e., HRT_t) using the discounting operator proposed in [167] and updating T_{tc} (line 18). Note that when n_d is reached, a last discounting operation is performed using the FT opinion of n_d (lines 19-20). Finally, T_{tc} of every trust chain is aggregated using the consensus operator[77] (line 21) and OT_{dest} is returned (line 22).

4.2.4 Functional Trust

We compute the RFT_t of an IoT device based on the MADM approach. We take all $RFT(d_t^{ID})$ scores of a device, generate a DI_{ID} from them, convert the DI_{ID} into an opinion

ω^{ID}_{P,RF_t}, and obtain the opinion reflecting HFT (i.e., ω^{ID}_{P,HF_t}) by combining ω^{ID}_{P,RF_t} and $\omega^{ID}_{P,HF_{t-1}}$. We use ω^{ID}_{P,HF_t} as the FT label on the last edge of the TN. Next, we describe how MADM is used by considering a "water level sensor" as an example scenario. Note that this is a representative example to illustrate our approach and *not* the actual evaluation.

4.2.4.1 MADM Problem Definition for Trust Computations

We model the derivation of trust of the data collected by IoT devices as a DMP. In our framework, an alternative A_i is a single data item d^{ID}_t collected by a "thing". The set of alternatives is all of the d^{ID}_ts collected by all "things" measuring a specific type of measurement in a predetermined time window, such as d^{ID}_ts from all "water level sensors" for "water level" mt in 2 hours. Next, we describe the criteria we use in MADM formulation.
Selected Criteria– Table 3 shows the criteria we use in MADM for trust computations. We use the first two of them for computing RFT scores and the last four for RRT scores. We explain the first two FT-related criteria here and the remaining RT-related criteria in Section 4.2.5. Key freshness is inversely proportional to the time elapsed since the last time a cryptographic key was established. As mentioned earlier, trust in sensed data may be affected by several environmental factors. We select temperature among these factors as an example criterion in the MADM formulation of RFT. The indicator is the ambient temperature under which the data item is collected. Here, we assume that temperature is a non-monotonic attribute, which means that its utility function does not show a regular increase or decrease [2].
Initial Representation of the Decision Problem– In Table 4, we present IM_{mt}, a simplified, yet carefully selected example for a water level sensor as a data source for which we want to compute the RFT. As shown in the table, the A_i's are from a single IoT device (ID=1). There may of course be more than one water level sensor collecting the water level measure, so Table 4 and the corresponding D may include alternatives from multiple sensors. However, a single IoT device collecting a specific measure is a special case concerning normalization as we explain below. For this example, we suppose there are five data items

[2] For an ultrasonic sensor such as a water level sensor in our scenario, the measurement accuracy shows a non-linear behavior with the temperature[2]

Criteria	Indicator	Unit	Type	Utility type
Time of key establishment	Key freshness[a]	Days	Quantitative	Benefit
Temperature	Ambient temperature of the "thing"	Fahrenheit	Quantitative	Non-monotonic
Time of data collection	Data freshness[a]	Seconds	Quantitative	Benefit
Measurement deviation	Absolute measurement deviation [b]	unit of mt	Quantitative	Cost
Link type	Data reliability due to link type	-	Qualitative	Benefit
Location	Distance between two nodes	Meters	Quantitative	Cost

[a] Formula respectively for key freshness and data freshness are: $1/(T_{key\_est}+1)$ and $1/(T_{data\_col}+1)$, where T_{key_est} and T_{data_col} are time elapsed since last time a cryptographic key was established and a data item was collected, respectively.

[b] Formula for absolute measurement deviation is: $|d_t(A)-d_t(B)|$, $mt_j \in MT$.

Table 3: MADM Criteria Used in MADM-EBSL for Trust Computations

collected in five-minute intervals. The key was established a day before data is collected (yesterday in the table) and it has been refreshed again (today in the table). The ambient temperature of the sensor remains stable.

Construction of the Decision Matrix– We present the D corresponding to the IM_{mt} in Table 5. The task in this step is to transform raw data, i.e., criteria values in IM_{mt}, into trust indicator values in D. The difference between D and IM_{mt} is that D has *trust indicators* of the selected criteria (see Table 3) as column labels and indicator values in cells, rather than raw data. The columns labeled as "O" are of interest to this step. They represent the original values of attributes transformed from IM_{mt}. To construct D, we assume that we are looking at the data in Table 4 today at 12 pm.

Normalization of the Decision Matrix– The next step after constructing D is normal-

Alternative	Time of key establishment	Time of data collection	Temperature
ID=1 d_1	Yesterday	09:05 am	75
ID=1 d_2	Yesterday	09:10 am	75
ID=1 d_3	Yesterday	09:15 am	75
ID=1 d_4	Now	09:20 am	74
ID=1 d_5	Today	09:25 am	74

Table 4: IM_{mt} for Water Level Measure

Alternative	**Key freshness**		**Temperature**		**RFT**
	O	N	O	N	
ID=1 d_1	0.5000	0.5000	75	0.9460	0.6115
ID=1 d_2	0.5000	0.5000	75	0.9460	0.6115
ID=1 d_3	0.5000	0.5000	75	0.9460	0.6115
ID=1 d_4	1.0000	1.0000	74	0.9651	0.9913
ID=1 d_5	1.0000	1.0000	74	0.9651	0.9913
Glob_min		0		0	
Glob_max		1		100	
LPV_j		-		32	
UPV_j		-		70	

O: original value (x_{ij}), N: normalized value (r_{ij})

Table 5: Decision Matrix for RFT Score Computation

izing the values in it. The normalization method is important to get a single trust value and depends on the utility type [163] (see Table 3). We present the normalization methods that we use in Table 6. We first experimented with normalization methods presented in seminal works of MADM [76, 190]. We discovered that they may lead to several issues in the trust computation – *zero denominator, zero normalized value, identical normalized value,* and *zero variance*. These usually occur when D includes identical values for d_t^{ID}s from a single IoT device[3]. A zero denominator leads to division by zero problems. Zero variance is a similar issue as the variance of attribute values is a denominator of the non-monotonic normalization formula in [76, 190]. A zero normalized value would cause discarding an attribute from MADM formulation. Identical normalized attribute values occur when the values of d_t^{ID}'s are identical in two different D's regardless of the actual attribute values x_{ij}.

The *zero denominator*, *zero normalized value*, and *identical normalized value* are related to linear normalization of monotonic attributes. To address them, we propose to adjust one of the linear normalization formulae proposed in [76, 190] with the *global maximum* and *global minimum* values for an attribute. The *zero variance* issue is related to the normalization of non-monotonic attributes. We address this by proposing a new normalization formula in Table 6 that adjusts the non-monotonic normalization formula in [76, 190]. The non-monotonic normalization formula in [76, 190] considers a single most favorable value for an attribute. We argue that using a *preferred interval* is a better choice as attributes may

[3] For a detailed explanation of how each issue is relevant to which normalization method and for sample scenarios that they occur, please refer to: https://github.com/nbaltaci/trust-framework-for-IoT

Utility Type	**Attribute type**	**Normalization formula** [a, b]	
Monotonic	Benefit	$r_{ij} = \frac{x_{ij}}{x_j^*}$	$x_j^* = glob\_max(x_j)$
	Cost	$r_{ij} = 1 - \frac{x_{ij}}{x_j^*}$	$x_j^* = glob\_max(x_j)$
Non-monotonic	Benefit	$r_{ij} = e^{(-2z^2)}$	$z = \frac{LPV_j - x_{ij}}{LPV_j - glob\_min(x_j)}$; $x_{ij} < LPV_j$
	Cost		$z = \frac{x_{ij} - UPV_j}{glob\_max(x_j) - UPV_j}$; $x_{ij} > UPV_j$
	Preferred values		$z = 0$; $LPV_j \leq x_{ij} \leq UPV_j$

[a] $glob\_max(x_j), glob\_min(x_j)$: Functions returning predefined global maximum and global minimum values of the jth attribute

[b] LPV_j, UPV_j : Lower and upper preference values of the preference interval PI of the jth attribute

Table 6: Normalization Methods Used to Normalize Criteria

not always be represented by a single most favorable value. For example, the ambient temperature attribute may have a preferred interval for a water level sensor where the accuracy is best for sensed data. We can consider the compensated temperature range (see [14]) of a water level sensor as the preferred interval for the temperature attribute. We represent the *preferred interval* of attribute j as $PI_j = [LPV_j, UPV_j]$, where LPV_j and UPV_j are lower and upper preference value of PI_j, respectively. We treat attribute values less than LPV_j as benefit attributes and attribute values greater than UPV_j as cost attributes (like the utility types in Table 3). Attribute values within PI_j are taken as the most favorable value and their normalized value is 1. This means that a non-monotonic attribute value yields the highest utility within PI_j. Its utility decreases as it moves away from PI_j.

For this example, we apply our normalization methods on the original values of D given in Table 5 and present normalized values in the same table under columns labeled with "N". We note the global maximum, global minimum, and preference values that we use for normalization in the table. We set $PI_j = [32, 70]$ for temperature attribute, which is an acceptable range according to [60, 51].

4.2.4.2 Weight Elicitation and Overall Score Calculation

As noted in Section 4.1.1, we use SAW to calculate the overall trust score of each item for our scenario. Weights assigned to attributes reflect their relative importance and are used to calculate the overall score for alternatives in D. Identifying the weights is known as the weight elicitation process. There are two ways to obtain attribute weights: recruiting human decision-makers who can assign a weight for each attribute or using a weight approximation method. A weight approximation method uses a ranked list of the attributes based on their relative importance and finds the surrogate weights, which are expected to be close to the "true" weights to be assigned by decision-makers [149]. In this paper, we adopt a weight approximation approach. There are three widely used methods, which are Rank Order Centroid (ROC), Rank Sum (RS), and Rank Reciprocal (RR) [174]. We use ROC in our trust computations as it has been shown to be superior to other methods (in terms of deviation from actual weights assigned by real decision-makers) [149]. The formula for calculating a ROC weight of an attribute X_j is: $w_j = 1/n \sum_{k=R_j}^{n} 1/k$, where n and R_j are the total number of attributes and the rank of X_j (the rank of an attribute indicates its importance), respectively[4].

4.2.4.3 Converting Trust Scores into Opinion Triplets

We next convert the scores ($RFT(d_t^{ID})$) obtained from MADM into opinion triplets that reflect FT (ω_{P,RF_t}^{ID}). This is performed in two steps. First, for each IoT device, we convert trust scores into distrust scores and compute a distrust interval (DI_{ID}). Next, we convert DI_{ID} into ω_{P,RF_t}^{ID}. We adopt methods presented in [191] for these conversions. In [191], local distrust values of a data report is converted into a distrust interval. Similarly, we convert *real-time functional distrust* scores ($RFD(d_t^{ID})$) into a $DI_{ID} = [LDB, UDB]$, where LDB and UDB respectively correspond to a lower distrust bound and upper distrust bound and $RFD(d_t^{ID}) = 1 - RFT(d_t^{ID})$. Then, computations are performed as follows:

$$LDB = max\{0, \overline{RFD(d_t^{ID})} - SD_{RFD(d_t^{ID})}/2\} \tag{5}$$

$$UDB = min\{1, \overline{RFD(d_t^{ID})} + SD_{RFD(d_t^{ID})}/2\} \tag{6}$$

[4] We compute weights by applying all the three methods and present them in supplemental material: https://github.com/nbaltaci/trust-framework-for-IoT

where $\overline{RFD(d_t^{ID})}$ and $SD_{RFD(d_t^{ID})}$ represent the mean and the standard deviation of distrust scores, respectively. Next, we compute an opinion triplet from DI_{ID} as follows:

$$\begin{aligned}\omega_{P,RF_t}^{ID} = (1 - \frac{LDB + UDB}{2} - \frac{UDB - LDB}{2}, \\ \frac{LDB + UDB}{2}, \frac{UDB - LDB}{2})\end{aligned} \tag{7}$$

Note that the proposed method in [191] converts a **distrust interval** into an opinion triplet for **data reliability**. Similarly here, we convert a distrust interval into an opinion that reflects the RFT of an IoT device (ω_{P,RF_t}^{ID}). Using RFT scores in Table 5 and (5) and (6), we compute $DI_{ID} = [0.1326, 0.3406]$. For this DI_{ID} value, we obtain $\omega_{P,RF_t}^{ID} = (0.6594, 0.2366, 0.1040)$ using (7).

4.2.4.4 Historic Reputation of an IoT Device

We combine the RFT of an IoT device with its HFT to take into account the *history* of a device (ω_{P,HF_t}^{ID}) in OT computation. This is performed over opinion triplets that represent RFT (ω_{P,RF_t}^{ID}) and HFT ($\omega_{P,HF_{t-1}}^{ID}$). $\omega_{P,HF_{t-1}}^{ID}$ is a cumulative value of opinion triplets generated from DIs of a device at different time points in the past. We use the EWMA method to combine RFT and HFT. This leads to a case where weights of observations decrease exponentially from the recent observations to earlier ones. As a result, recent trust values have more impact on ω_{P,HF_t}^{ID}. Also, as presented in Algorithm 1, reputation is computed by using the equation below, where $\alpha \in [0, 1]$. Note that $\omega_{P,HF_{t=0}}^{ID} = 0$, so **for** $t = 1$, $\omega_{P,HF_{t=1}}^{ID} = \omega_{P,RF_{t=1}}^{ID}$.

$$\omega_{P,HF_t}^{ID} = \alpha\omega_{P,RF_t}^{ID} + (1 - \alpha)\omega_{P,HF_{t-1}}^{ID} \tag{8}$$

We can tune α to weigh historical values differently.

4.2.5 Referral Trust

In this section, we explain how we obtain RT (ω_{B,HR_t}^{A}), as labels on the edges of the TN. Like the RFT scores, we compute RRT scores using MADM with the same steps. We

	Measurement deviation			**Link type**			**Location**			**Data freshness**					
										IM					
Alternative	IM	O	N	IM	O	N	IM	O	N	PD [a]	DD [a]	TE [a]	O[a]	N	**RRT**
ID=B $d1$	0.000	0.000	1.00	BT	Avg	0.556	(x,y)	100	0.8	33.33	881	914.33	0.0011	0.0011	0.7879
ID=B $d2$	0.000	0.000	1.00	BT	Avg	0.556	(x,y)	100	0.8	33.33	709	742.33	0.0013	0.0013	0.7879
ID=B $d3$	0.002	0.002	0.96	WiFi	Low	0.333	(x,y)	100	0.8	33.33	218	251.33	0.0040	0.0040	0.7069
ID=B $d4$	0.002	0.002	0.96	WiFi	Low	0.333	(x,y)	100	0.8	33.33	54	87.33	0.0113	0.0113	0.7069
ID=B $d5$	0.002	0.002	0.96	WiFi	Low	0.333	(x,y)	100	0.8	33.33	796	829.33	0.0012	0.0012	0.7069
Glob_min	0			1			0			0					
Glob_max	0.05			9			500			1					

IM:value in IM_{AB}, O: value in decision matrix (x_{ij}), N: normalized value (r_{ij})
PD: propagation delay, DD: data delay, TE: time elapsed since the data item was sent
[a] In units of $10^{-8}s$, [b] In units of $10^{8}s^{-1}$

Table 7: Decision Matrix for RT Computation

convert RRT scores into DI, obtain opinion triplets from DI, and use historical values with EWMA. We do not delve into details of these steps again here, but explain points specific to RRT and present the results of those steps below.

First, the initial matrix IM is constructed for a pair of "things", e.g., thing A and thing B, for RRT computation in the cloud server. An edge in a TN is directed between one-hop neighbor nodes from a source node to a destination node (see Figure 4). We assume that thing A is the source and the trust it has on thing B is to be computed, so we represent the matrix as IM_{AB}. We show the initial representation in Table 7 under the columns labeled as "IM". We include *measurement deviation*, *link type*, *location*, and *data freshness* as criteria (see Table 3) in the MADM formulation of RRT. Here, measurement deviation corresponds to the deviation between the values of data items reported by two nodes for a specific mt and a time interval t. If two nodes do not report data items with the same mt, then the deviation is set to zero. In our sample scenario, we consider that thing A and thing B both report water level measurements. Hence, the *measurement deviation* refers to the deviation in water level reported by the two "things" with a measurement unit of meters. In the example given in Table 7, measurement deviation changes from 0 to 0.0020 meters after two data exchanges. The *location* criterion corresponds to the geographical coordinates of an IoT device. In Table 7, we arbitrarily represent the location of thing B as (x,y) and assume no change in it. *Link type* refers to the communication protocol that can be used to send data between devices in the IoT network. It covers all types of links among devices,

Link type	**Typical Coverage**	**Data Reliability due to Link Type**	**Attribute value**
Zigbee	10-1000 m	Very low	1
WiFi	30-250 m	Low	3
Bluetooth	1-100 m	Average	5
RFID	10 m	High	7
NFC	10 cm	Very High	9

Table 8: Link Types

between a pair of IoT devices, IoT device and a gateway or a gateway and a cloud server. The most commonly used communication protocols in IoT are presented in Table 8 (such as in a flood early warning system [2] or for device pairing in general [46]). We use the list in Table 8 as a set of possible values for link type criterion in our RRT computations. *Data freshness* is the recency of the data sent by an IoT device. The data freshness is a benefit attribute, since we trust information when data is more recent/fresh and trust decreases as data become outdated. It is inversely proportional to the time elapsed after a data item has been collected by a device and until it reaches a destination point. The elapsed time is the sum of propagation, processing, transmission delay, etc.[5]

Second, we construct D from IM_{AB}. We show the values of D in Table 7 under the column labeled as "O". D is constructed by converting the raw data of IM using the indicators of attributes presented in Table 3. The measurement deviation attribute in D is $|d_t^B - d_t^A|$. For the location[6] criterion, we use the distance between the IoT nodes. Even when the distance between two nodes does not change, it is still a distinguishing factor for RT computation. The link type criterion is represented by a qualitative indicator: *data reliability due to the link type* as seen in Table 3. We assume that this indicator is an ordinal variable and takes fuzzy values in Table 8: very high, high, average, low, and very low. This criterion reflects how reliable the link is in carrying the data. To convert the link type to *data reliability due to link type*, we assume that data reliability is affected by the typical coverage of a link. For example, in Table 8, NFC has the shortest range so it has very high

[5] The propagation delay is the time for the data to travel from a device to another device, so it is *distance/propagation speed*. For our example, propagation delay is $33.3333 \times 10^{-8}s$ as the propagation speed for wireless communication is $3 \times 10^8 m/s$.

[6] We note that it is unlikely that the location of a sensor will change unless it is embedded in a mobile device.

reliability. For data freshness criterion, we use a quantitative indicator that is the inverse proportion of the elapsed time, as we explained earlier. It should be emphasized that other factors can be used or included in RT calculations in a similar manner.

Third, we need to normalize D. We present the normalized values in Table 7 under the column "N". As seen in Table 3, measurement deviation, distance, and data freshness are monotonic quantitative attributes. Hence, these attributes are normalized by using the formula presented in Table 6. On the other hand, the link type attribute has a qualitative indicator. We map the values of a qualitative indicator to numerical values using a 10-point Bipolar scale [76]. Since data reliability due to link type is a benefit attribute as seen in Table 3, we map the value of very high to a numerical value of 9. Table 8 shows the mapping of remaining values for all link types under the column *Attribute value*. Then, we normalize attribute values using the linear normalization formula for monotonic benefit attributes in Table 6. Note that we present the global maximum and minimum values used in normalization formulae at the bottom of Table 7.

Finally, we use SAW to obtain overall trust scores. Like the FT computation, we compute approximated weights by using the ROC method. Overall scores are presented in Table 7, under the column "RRT". In weight computation, we assume the attributes are ranked from the left to the right in Table 7 concerning the order of importance in decision making. At the next step, we apply (5) and (6) on these scores and obtain $DI = [0.2385, 0.2829]$. We obtain an opinion triplet $\omega^A_{B,RR_t} = (0.7171, 0.2607, 0.0222)$ from the DI by applying (7). Finally, we combine ω^A_{B,RR_t} with $\omega^A_{B,HR_{t-1}}$ using the EWMA method to obtain ω^A_{B,HR_t}. In the next section, we explain how to combine opinions ω^A_{B,HF_t} and ω^A_{B,HR_t} to compute OT_{dest}.

4.2.6 Overall Trust Opinion

As we explain in Section 4.2.3, the last step of trust computation for finding OT_{dest} is traversing through the trust chains (tc's) between n_s and n_d in a TN. The trust opinion of a chain (T_{tc}) is updated as each node along the chain is visited. This update operation is performed using the discounting operator ($\boxtimes$) of EBSL – see Equation (4) in Section 4.1.3. After the visit of each tc is completed, OT_{dest} is updated with the recently computed T_{tc}.

This update operation is performed using the consensus operator ($\oplus$) of SL – see Equation (2) in Section 4.1.2. In our framework, we replace x and y in (2) and (4) with relevant opinion triplets in a TN of "things". For example, we replace x with ω^{A}_{B,HR_t} and y with ω^{B}_{P,HF_t} for applying $\boxtimes$ on the sample TN in Figure 4. In the supplemental material[4], we present sample trust computations using a sample trust network.

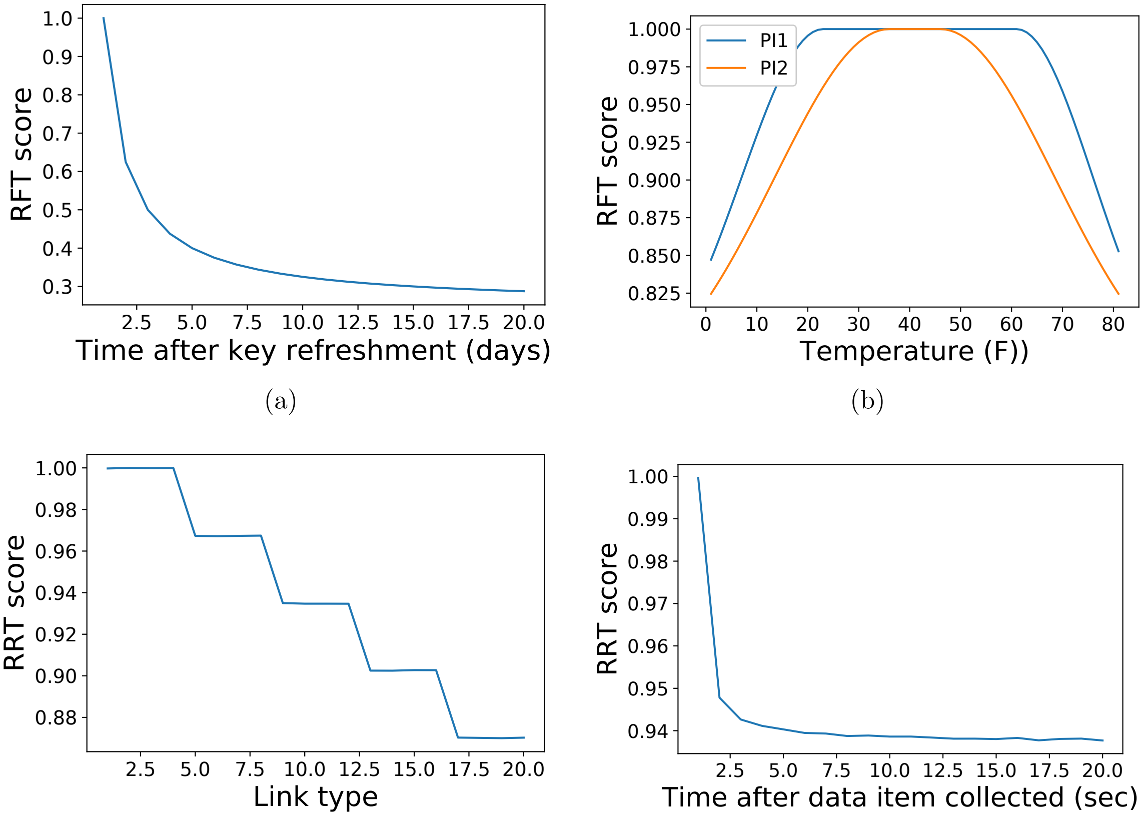
4.3 Evaluation

In this section, we first examine how MADM attributes affect trust scores. Next, we validate our proposed trust computation method using real data.

4.3.1 Effect of Attribute Values on Trust Scores

We assume that there are 20 data items measured by a "thing" and *synthetically* generate attribute values for each. We change the value of "attribute of interest" while keeping other attributes constant at their most favorable values. These are global maximum, global minimum, and preference interval values in Table 5 and 7, respectively for a benefit, cost, and non-monotonic attribute. We assume the importance of the attributes is as per their order in Table 5 and 7. We use the normalization methods in Table 6 and use ROC weights to compute RFT and RRT scores.

Figure 7 shows the effect of change in individual MADM attributes on RFT and RRT scores for key freshness, data freshness, link type, and temperature. Key and data freshness attributes fall as (1/time elapsed) – the computed scores show a rapid decrease when the elapsed time increases. RFT scores for the link type attribute is a step-wise function. The reason is that we keep the link type constant for each group of 4 data items and then change it for the next block of 4 data items. The preference interval for temperature impacts the RFT scores in Figure 7b. In this figure, $PI1 = [32, 70]$ and $PI2 = [45, 55]$. A narrower PI has a smaller plateau – a shorter range for preferred temperature values that yield the highest RFT score.



(a)

(b)

(c)

(d)

Figure 7: Effect of changing MADM attributes on trust scores. Effect of: (a) key freshness (b) temperature preference interval (c) link type (d) data freshness

4.3.2 Validation of the Proposed Method

We conducted two sets of experiments using synthetic data sampled from real datasets to validate our trust computation framework. Below, we present datasets we use in the evaluation, the method of evaluation, and the results we obtain.

4.3.2.1 Dataset and simulation setup

We use two different datasets in our experiments. The first, by Dataport [141], contains weather data collected from 3 cities in Texas, the USA from June 2012 to November 2012. We assume that these data represent weather-related data collected and reported by IoT nodes and call it a *local sensor dataset*. The second dataset is provided by the National

Centers for Environmental Information (NCEI) [130], which contains local climatological data for different cities. We assume that NCEI is an authority such that their reported data can be used as a coarse-grained check of the reliability of the data collected by IoT nodes and call it the *authority dataset*. We match the authority data to the local sensor data reported within the same time frame. We consider *temperature* as the *measurement type* mt of the measurement reported by sensors.

We assume an IoT network with nodes distributed according to a Poisson point process in a rectangular area of 10000 m^2. The Poisson distribution has parameter $\lambda = 0.001$ (intensity)[7]; hence, there are 16 nodes in the network. We consider Dijkstra's shortest path algorithm [39] for routing. We generated 300 data items for each node (explained below). We randomly selected 10% of nodes to be faulty (corresponds to 2 nodes in our random network) – they report incorrect observations (temperature values) during the entire simulation. We also consider possible benign nodes being compromised modeled as follows. A randomly selected node is assumed to be compromised after the first 200 observations. The last 100 observations are drawn from a distribution that has a different mean (smaller mean value in the simulation) compared to the mean value of the benign observations.

To apply MADM for FT computation, we used the attributes discussed in Section 4.2.4, i.e. *key freshness and temperature* in that order of importance. We generated key freshness values randomly between 0 and 1. For the temperature attribute, we used statistical distributions fitting best to the real temperature values in the local sensor and the authority datasets. Using the parameters of fitted distributions, we generated temperature values as measurements by each local sensor and the authority. The attributes we use to apply MADM for RT computation are the same as the attributes we discussed in Section 4.2.5, i.e., *measurement deviation, distance, link type, and data freshness.* Note, measurement deviation corresponds to the absolute difference of the temperature value measured by two sensors. We use the Euclidean distance between nodes. The link type attribute was decided based on the distance[7]. For data freshness, we computed propagation time between each pair of nodes based on their distance as explained earlier and added random delay values from a

[7] This value is obtained by experimenting with different values, the goal being able to have different types of links between sensors (nodes within a distance less than 10 cm are connected by NFC and more than 46 meters are connected by Zigbee.

Gaussian distribution $N(5, 1)$ (sec) to model possible processing delays. We computed data freshness as the inverse of the sum of the delays.

4.3.2.2 Evaluation Method

For the first set of experiments, we set each node as a destination node in the random TN we generated and compute trust scores and opinions for each. To do so, we implemented the steps presented in Algorithm 1. We set initial $\omega^{ID}_{P,RF_{t-1}} = (0, 0, 1)$, $\omega^{A}_{B,HR_{t-1}} = (0, 0, 1)$, and $\alpha = 0.5$. We compute distrust scores for each of 300 data items by applying MADM for both FT and RT. Then, we assign a DI to each row in the dataset that takes into account distrust scores starting from the first row until the row of interest. We then convert the DI for each row into a real-time trust opinion (both for FT and RT) and combine the real-time opinion with historical opinions. We obtain a series of RFT scores, RFT and HFT opinions, RRT and HRT opinions for each node in the TN.

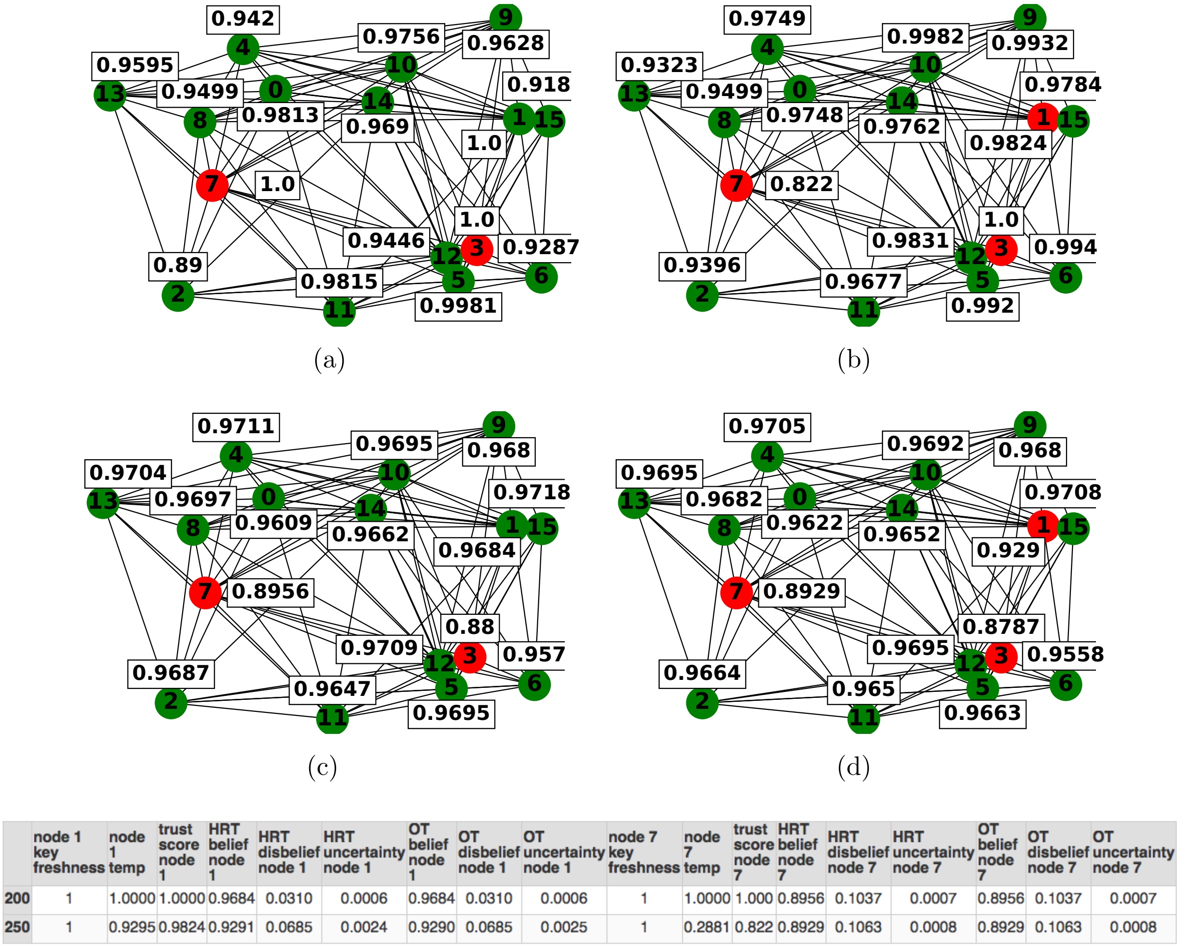
For the second set of experiments, we use a randomly selected destination node in the TN and compute trust scores and opinions for this destination node. In these experiments, we compare trust values generated by our approach for the destination node to the trust that the cloud generates from the authority data. We use Algorithm 1 to compute trust scores and opinions for the destination node. To find the trust based on authority data, we use a measure that is the absolute difference of the temperature values reported by the destination node and the authority for the same time instances (we recall that these temperature values were generated for nodes and the authority based on the best-fit distributions to the real data for the same time interval – so they will be different). We normalize this measure and use it as a real-time distrust score for each reported temperature value by the destination node. The remaining steps are the same as explained above, i.e., converting scores into a DI, then from DI's into real-time and historical opinions. In the following sections, I present the results of our experiments.

4.3.2.3 Effect of Reported Temperature on Trust Scores and Opinions

Recall that one of the nodes was randomly selected to model node compromise. Here, node 1 in Figure 8 is a compromised node. It reports temperature values sampled from the best-fit distribution until node compromise (200th data item) and then temperature values from a distribution that has *significantly lower* mean temperature after node compromise; we gradually decrease the mean temperature, with the lowest being 60 less than the "honest" mean. Nodes 3 and 7 in Figure 8 are assumed to be faulty from the beginning of the simulation.

We computed RFT/RRT scores, RFT/HFT and RRT/HRT opinion triplets for each node by keeping all attributes constant at their best normalized values except for the reported temperature. In other words, key freshness (FT attribute), distance, link type, data freshness (RT attributes) were set to 1. Note that the measurement deviation (RT attribute) was random as it is computed as the difference in reported temperature values between each pair of nodes. We then took two snapshots of the network and visualize RFT scores and OT opinions for every node: one network snapshot before the node compromise (200th data item) and one snapshot afterward (250th data item). Figure 8 shows the two network snapshots for RFT scores and OT opinions. The table in the figure shows the RFT scores, HRT and OT opinion values of nodes 1 and 7 for the 200 and 250 data items. As shown in Figures 8a and 8b, node 1 has a lower RFT score after being compromised and is marked by red. Node 7 also has a lower RFT score for data item=250 whereas node 3 has the same RFT score in both snapshots. The reason for these two faulty nodes to have different patterns in RFT scores is that their reported temperature values show some fluctuations around a mean value and do not necessarily decrease/increase constantly. Recall that the ambient temperature impacts the measurement accuracy, which is reflected in the functional trust. Here, node 7 has a lower normalized value w.r.t. temperature in the second snapshot, so its RFT score decreases whereas node 3 has the same normalized temperature value, so its RFT score stays the same. The same observation applies to the other healthy nodes, i.e., some of them have higher and some have lower RFT scores depending on the ambient temperature.

Figures 8c and 8d show the overall trust (OT) beliefs in the two snapshots. As expected,



(a)

(b)

(c)

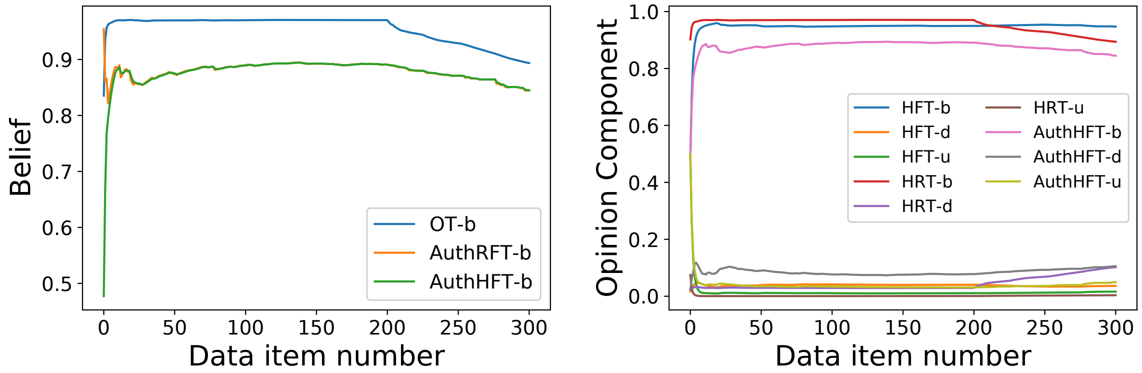
(d)

	node 1 key freshness	node 1 temp	trust score node 1	HRT belief node 1	HRT disbelief node 1	HRT uncertainty node 1	OT belief node 1	OT disbelief node 1	OT uncertainty node 1	node 7 key freshness	node 7 temp	trust score node 7	HRT belief node 7	HRT disbelief node 7	HRT uncertainty node 7	OT belief node 7	OT disbelief node 7	OT uncertainty node 7
200	1	1.0000	1.0000	0.9684	0.0310	0.0006	0.9684	0.0310	0.0006	1	1.0000	1.000	0.8956	0.1037	0.0007	0.8956	0.1037	0.0007
250	1	0.9295	0.9824	0.9291	0.0685	0.0024	0.9290	0.0685	0.0025	1	0.2881	0.822	0.8929	0.1063	0.0008	0.8929	0.1063	0.0008

(e)

Figure 8: Effect of temperature values on trust scores and opinions. Network snapshots showing: (a) FT scores before node compromise (b) FT scores after node compromise (c) OT beliefs before node compromise (d) OT beliefs after node compromise

there is a decrease in OT (belief) for the compromised node, node 1. Note that faulty nodes (3 and 7) have lower OT beliefs in Figure 8d as they continue to report false data. It should be emphasized that neighboring nodes are affected if one of their neighbors is faulty or compromised, as the referral trust will also be impacted (in this evaluation, this is reflected in the measure deviation attribute). We note this in the slight decrease in the benign nodes' belief in the second snapshot (Figure 8d).



(a)

(b)

Figure 9: Comparison of our trust computation method to an authority data-based method: (a) beliefs (b) components of historical opinions for the destination node.

4.3.2.4 Comparing Proposed Solution to Authority Data Approach

In these experiments, we compare the trust computation with a baseline method that computes trust scores and opinions using the authority dataset as explained in Section 4.3.2.2. Note that the authority data-based approach only uses a single attribute, which is the measurement deviation, whereas our proposed method considers multiple attributes discussed earlier. We randomly select a destination node and model node compromise for this destination node as in the previous experiment.

We compare overall trust opinion for the destination node generated by our trust computation method to the opinion based on authority data in Figure 9a. In this experiment, we keep all FT and RT attributes fixed at their maximum value except temperature and measurement deviation attributes. We focus on belief components in opinions. *OT-b* refers to the belief component of OT opinion obtained by applying our trust computation. *AuthRFT-b* and *AuthHFT-b* are the belief components of the opinion for functional trust, generated by considering the authority data. Our method has an immediate and dramatic decrease in its OT belief on the destination node after node compromise whereas with only the authority data, this happens, but with a smaller decrease. It is also observed that historical belief values align with the real-time belief values after a short period. In Figure 9b we present a more detailed comparison. Here we compare our method to the authority data-based method concerning all components of historical opinions. All FT and RT attributes are fixed except temperature and measurement deviation attributes. We observe that our method has a decreasing HRT belief for the destination node as does the authority data-based approach. We also see that decrease in HFT belief is not as apparent as the decrease in HRT belief. This is because pairwise measurement deviation between the destination node and all other nodes in the TN magnifies the effect of node compromise in RT compared to FT.

4.4 Summary and Discussion

In this chapter, we propose an automated trust computation framework for IoT. We present a method for computing nodes' trust, which is based on EBSL and MADM. Our approach combines FT and RT components within a trust network (TN) taking into account uncertainty in trust evaluations. A significant contribution of our framework is that we provide a method to quantify trust scores and convert them into opinions to be used in a TN. The trust scores can account for multiple contextual and technical attributes affecting both FT and RT using the MADM approach. We validate the proposed trust evaluation mechanism using sampling from real data. The results show that the trust framework can effectively capture nodes' behavior and limitations, where faulty and compromised nodes can be identified. It is also observed that the trust of nodes would be impacted by their neighbors in the network.

We note that the attributes we include in MADM are *design decisions that can change depending on the system and/or application.* Thus, the investigation of additional attributes in MADM formulation for trust computation in application-specific IoT networks could be a candidate future research direction. Yet, our proposed approach can be taken as a guideline example as it addresses various attribute types. Finally, we used temperature data in our experiments, which is one of the data in the dataset we obtained. There were other weather-related data, such as humidity, wind, etc. There was no specific reason to choose temperature. It can be replaced with any other phenomenon measured by nodes.

5.0 Trust Transfer in IoT: Network Connectivity and Trust Problem Size

5.1 Motivation and Background

In social sciences trust literature, trust transfer is described as building trust impressions based on the information provided by third parties instead of relying on own prior experience with a trustee [147]. In network science, trust transfer, a.k.a *trust transitivity* [65, 61, 6], is described such that if a node u trusts node v, which trusts node w, then node u may trust node w indirectly without having to have direct prior experience. Trust prediction solutions for networks may rely on the trust transitivity concept such that trust is *propagated* on paths (a sequence of nodes linked together) in a network and *aggregated* from multiple paths [6]. Chapters 2 and 3 present more detailed information about trust transitivity.

Trust transitivity is essential for trust assessments because, in the absence of direct experience with a trustee, a trustor can utilize experience from other parties [145]. Networks with less connectivity may confront the inaccurate trust prediction issue as trust will not be transferred due to missing node connections. Despite its significance, trust transfer in sparse networks is an under-researched problem [3]. For this research task, I explore how network connectivity and trust problem size affect trust scores in a transitive trust computation framework for IoT, which is our previously proposed MADM-EBSL framework [5].

In the following, I first elaborate on network connectivity and trust problem size concepts. Next, I discuss attack types that apply to the MADM-EBSL trust computation framework. Following this, I present an experimental evaluation of MADM-EBSL concerning network connectivity and trust problem size. Finally, I summarize the contributions of the research.

5.1.1 Network Connectivity and Trust Problem Size

I have discussed network sparsity earlier in Section 2.3 of Chapter 2. For the completeness of explanation here, *network connectivity* is about how dense the links among nodes in a network are. Trust may not emerge, or trust computations may not be as accurate in sparse networks as in highly connected networks. For instance, Wu and Li [188] observed that trust prediction accuracy drops in sparse multi-domain RFID networks. Thus, it is still an open question for further investigation in IoT trust management. *Trust problem size* has been defined in social sciences literature for the relationships among humans and other entities [3]. Specifically, in experimental economics [48], trust problem size is explained as the probability of a trustee being opportunistic, i.e., abusing trust. More precisely, if there is a high probability of a trustee being opportunistic, it is a *large trust problem size*. An example of a large trust problem size could be asking for a loan to do business in an industry with unreliable entrepreneurs [48]. Conversely, if the probability of a trustee being opportunistic is low, i.e., the probability of a trustee honoring trust is high (a friendly trustee), it will be a *small trust problem size*. An example of a small trust problem size could be selling ordinary consumer products in a brick and mortar market [47].

To the best of my knowledge, trust problem size has not been discussed in IoT trust research literature. For this research task, I consider an analogy between trust problem size definition in social sciences and a possible definition for IoT environments. I argue that the *ratio of malicious nodes* in a network and *attack model/attack type* are the factors that affect trust problem size in IoT environments. Conducting experiments with the ratio of malicious nodes as an experimental parameter is also encouraged by the existing work ([25, 1, 188]) to evaluate the performance of a trust computation solution. Also, there is a gap in the IoT trust literature for evaluating trust computation solutions against different attack types [1]. Hence, it is crucial to assess a trust computation solution against several applicable attacks.

5.1.2 Attack Model

I have already presented an overview of attacks on trust management systems for IoT in Chapter 2. There, I grouped them under two categories as *performance-related* and *recommendation/reputation-related* attacks. From these attacks, only a subset of performance-related attacks applies to our proposed MADM-EBSL trust framework. The reason is that our proposed solution does not employ ratings/service feedback by IoT nodes or recommendation sharing among them. Thus, it provides an inherent defense against reputation-based trust attacks.

Among performance-related attacks, the following three apply to our proposed trust framework: *individual malicious nodes, opportunistic service attack*, and *on-and-off attack*. I discuss how I address these attacks in Section 4.3.2.2. The rest of the performance-related attacks discussed in Section 2.5 of Chapter 2 do not apply to our proposed MADM-EBSL framework, which are malicious pre-trusted nodes, whitewashing, and Sybil attacks. Malicious pre-trusted nodes do not apply to the MADM-EBSL because it does not employ pre-trusted nodes for trust computations. Also, the MADM-EBSL provides defense against whitewashing and Sybil attacks because nodes are uniquely identified by a central trust management/measurement module in the cloud using an identifier (e.g., cryptographic keys can be used for identification of IoT nodes and to provide defense against white-washing attacks [1]). Trust scores of IoT nodes and their reported data are stored with their unique ID, known to the trust module in the cloud. Thus, an IoT device cannot imitate this number during data collection [5].

5.2 Experimental Evaluation

I conducted experiments using synthetic data sampled from real datasets to investigate the effect of network connectivity and trust problem size on trust scores computed by the MADM-EBSL. In the following, I describe datasets used in experiments and the evaluation method. Finally, I discuss the results of the experimental evaluation.

5.2.1 Dataset and Simulation Setup

I use a dataset provided by Dataport [141] for experimental evaluations. It contains weather data collected from three cities in Texas, USA, from June 2012 to November 2012. We assume that these data represent weather-related data collected and reported by IoT nodes. We consider *temperature* as the *measurement type mt* of measurements reported by sensors. Similar to the simulation settings in Section 4.3.2.1 of Chapter 4, we assume an IoT network with nodes distributed according to a Poisson point process in a rectangular area and the Poisson distribution parameter $\lambda = 0.001$[1]. Differently, the rectangular area defined in simulations here is wider with a value of $100,000\ m^2$. Thus, there are 100 nodes in the network. We, again, consider Dijkstra's shortest path algorithm [39] for routing.

To model different *network connectivity levels*, I generate multiple versions of the network described above. The versions differ by *the number of nearest neighbors* (*nn*), or node degree, each node connects. In simulations, we have six different network connectivity levels $nn = 1, 3, 5, 7, 9, 10$. To model different *trust problem sizes*, I generate multiple versions of a network with a given connectivity level. The versions differ in terms of *malicious node percentage* (*mnp*) and the *attack type* performed by malicious nodes. We have six different levels of malicious node percentages, $mnp = 10\%, 30\%, 50\%, 70\%, 90\%, 100\%$. Also, I investigate three different attack types I discussed earlier; *individual malicious nodes, opportunistic service*, and *on-and-off* attacks. I present a detailed explanation of how I address these attacks and simulate the malicious behaviors related to them in Section 4.3.2.2.

Similar to the experiments in Chapter 4, I generate 300 data items for each node. To generate data items for each node, I consider the MADM attributes mentioned earlier in Chapter 4, i.e., *key freshness and temperature* as FT attributes, and *measurement deviation, distance, link type, and data freshness* as RT attributes. Interested readers are referred to Section 4.3.2.1 of Chapter 4 for details of generating attribute values (data items) for each node. Noteworthy mentioning is the ranking of attributes in simulations in this chapter. I rank the attribute of interest first and keep the remaining in the same order of importance given in the lists above. For example, when I want to investigate the effect of temperature, I rank the temperature among FT attributes and the measurement deviation among RT attributes first (because the measurement of interest is temperature). The central trust management module can change these based on the situation.

[1] This value is obtained by experimenting with different values, the goal being able to have different types of links between sensors (nodes within a distance of less than 10 cm are connected by NFC, and more than 46 meters are connected by Zigbee).

5.2.2 Evaluation Method

In this section, I first elaborate on the overall methodology of experimental evaluation for the research task in this chapter. Next, I present the details about how I model node behaviors (trust problem size configurations in experiments), i.e., benign node behaviors, faulty node behaviors, and malicious behaviors for different attack types. Finally, I discuss alternative options for representing/including historical trust in overall trust computations.

5.2.2.1 Overall Method

In the experiments in this chapter, I compare patterns in trust opinions generated by our approach for a destination node in different network settings with varying network connectivity and trust problem size levels. For these experiments, I randomly select a malicious node in the TN (randomly generated network) as the destination node and compute trust scores and opinions for this destination node. Note that the malicious destination node is the same across each network (with different malicious node percentages and connectivity levels). We set it fixed for allowing trust opinion comparison across networks. If we select a different destination node for each network, we obtain unexpected patterns in results and cannot compare their trust scores to understand the effect of nn and mnp.

I implemented the steps of Algorithm 1 to compute trust scores and opinions for the destination node. Similar to the experiments in Chapter 4, I set the initial $\omega^{ID}_{P,RF_{t-1}} = (0,0,1)$, $\omega^{A}_{B,HR_{t-1}} = (0,0,1)$, and $\alpha = 0.5$. Similarly, I compute distrust scores for each of the 300 data items by applying MADM for both FT and RT. Then, I assign a DI to each row (a single data item) in the dataset. DI takes into account distrust scores starting from the first row until the row of interest. I then convert the DI for each row into a real-time trust opinion (both for FT and RT) and combine the real-time opinion with historical opinions. As a result, we obtain a series of RFT scores, RFT and HFT opinions, RRT and HRT opinions for each node in the TN.

5.2.2.2 Trust Problem Size/Attack Model Configurations

As mentioned earlier in the chapter, trust problem size relates to malicious node percentage and attack model/type. I already discussed malicious node percentage settings used for simulations. In this section, I focus on attack models and how I simulate them as malicious behaviors. I evaluate the MADM-EBSL against the *individual malicious node, opportunistic service*, and *on-and-off* attacks (See Section 2.5 for a discussion of the reasons for selecting these attacks). In addition to malicious node behaviors, I simulate benign and faulty node behaviors. Below, I discuss the details of how I simulate these node behaviors.

- *Benign nodes*: As benign nodes always behave honestly, i.e., report measurements correctly, I model their behavior by finding the best-fitting distribution to the real data, and sampling values with "honest" mean from the best-fit distribution. After assigning malicious and behavior nodes, the remaining nodes are assigned to the benign category.
- *Faulty nodes*: They report incorrect data during the whole simulation, i.e., values sampled with a single standard deviation distance from the mean of best-fit distribution. After assigning malicious nodes, 10% of the remaining nodes are assigned to the faulty category.
- *Individual malicious nodes*: These nodes always act maliciously. In the context of our proposed MADM-EBSL, an individual malicious node always provides poor service, i.e., reports measurements significantly deviating[2] from the readings reported by benign nodes during the whole simulation.
- *Opportunistic service attack*: In this attack, a node performs benign actions and maintains high trust to stay in the network longer. Then, utilizing its high trust score, it starts demonstrating malicious behaviors. In simulations for this attack, the randomly selected destination node starts its malicious behaviors after the 200th transaction. After this point until the end of the simulation (the last 100 observations), the malicious

[2] A value drawn from a distribution with a significantly larger or smaller mean value than the mean value of best-fitting distribution for the attribute of interest, such as temperature. More precisely, malicious behavior is simulated as reporting measures sampled from a distribution with multitudes of σ distance from the mean of best-fit distribution. Note that I decide whether to add or subtract multitudes of σ to/from the mean value of the best-fitting distribution by randomly assigning a value to a binary flag variable.

node always reports measurements significantly deviating[2] from those reported by benign nodes.

- *On-and-off attack*: In this attack, a node alternates its benign and malicious behaviors in order not to be recognized by the trust management system. An on-off attack can be of two types — *patterned* and —*random* [23]. A patterned attack consists of an exact number of *good* behaviors (G) followed by an exact number of *bad behaviors*(B), represented as $nG - mB$ such as $4G - 1B$ [23]. A random attack does not follow such a regular pattern. The attacking node randomly demonstrates Bs and Gs with a certain percentage, such as 20% Bs and 80% Gs [23]. The higher the G/B ratio, the harder it is for a trust management system to detect malicious behaviors [23]. In simulations for this attack, I experiment with both types of on-off attacks with a $4G - 1B$ (and 20% $B-$ 80% G) ratio.

5.2.2.3 Trust History Configurations

We have proposed using all the historical trust records for trust computations in the MADM-EBSL. We found that a *sliding window* approach yields better results during our preliminary analysis for the research task in this chapter. Fig 10 compares trust scores (i.e., the belief component of overall trust opinion) produced using the *whole history*, *no history*, and *sliding window* (*windows size (ws) = 10*) for opportunistic service attack. We generated similar plots for other attacks which we investigated the effects on the MADM-EBSL. As they lead to the same results, we only discuss the opportunistic service attack scenario here. Fig 10 shows that the sliding window approach is favorable because the whole history case reflects the effect of an attack on trust scores in the longer term. Also, fluctuations in the no history case make it difficult to compare trust scores among different network configurations. Furthermore, using the whole history of trust records may not be feasible for some IoT applications [23], such as wireless sensor networks, that include devices with limited storage and processing speed. As a result, we selected the *ws=10* configuration for the rest of the experiments. Again, the central trust management entity can choose variation of these approaches depending on the situation.

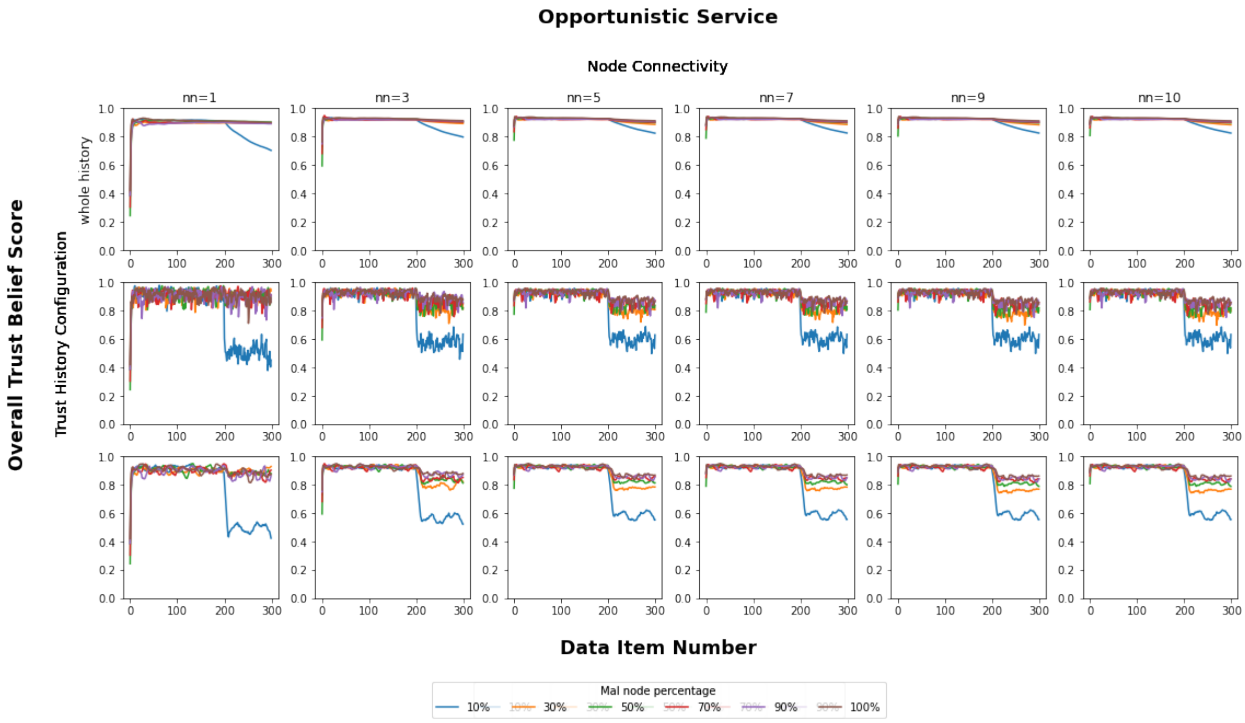
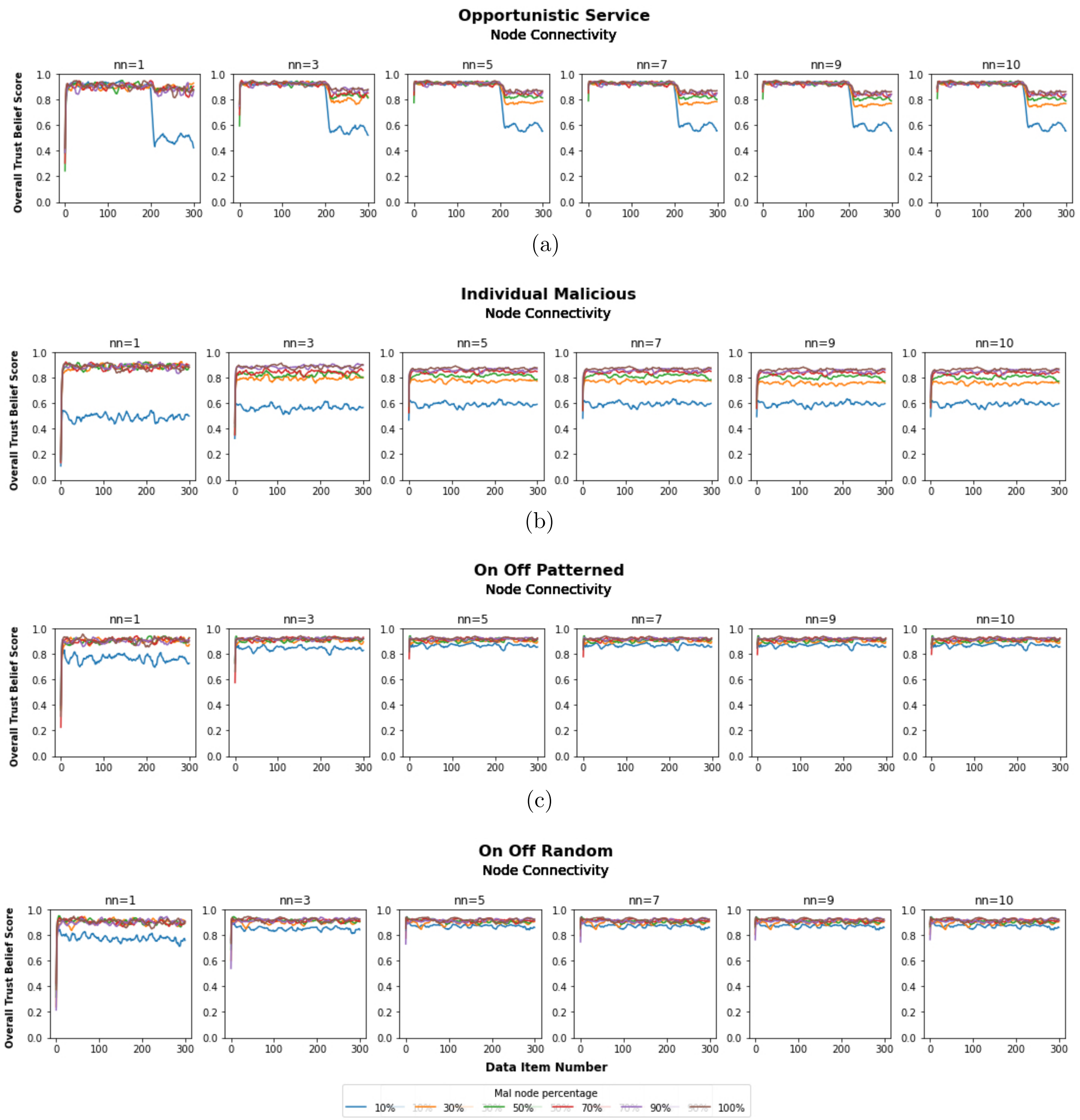


Figure 10: Comparing trust history configurations

The decision to utilize full historical data or modify the window size, whether by reducing or increasing it, can have significant implications. Understanding when each approach may be useful is crucial for making informed choices. For example, in scenarios involving a long off-short on attack, opting for an instantaneous approach may yield better outcomes. However, in other contexts where a comprehensive understanding of historical patterns is necessary, employing the full history or adjusting the window size accordingly becomes imperative. By considering the specific requirements of the situation at hand, trust management authorities can make well-informed decisions regarding the appropriate utilization of historical data and window size adjustments.

5.2.3 Results

I present the results of the experimental evaluation in Fig 11. I compare the OT belief scores for the destination malicious node concerning varying levels of network connectivity and trust problem size. In the sequel, I discuss findings on the effect of network connectivity



(a)

(b)

(c)

(d)

Figure 11: Comparison of *OT* belief across varying network/node connectivity and malicious node percentage levels for the following attack scenarios: (a) opportunistic service (b) individual malicious node (c) on-off patterned (d) on-off random.

and trust problem size on trust scores.

5.2.3.1 Effect of Network Connectivity on Trust Scores

The effect of network connectivity on trust scores of the malicious node is not nuanced by only looking at a given trust problem size, i.e., attack type and *mnp*. To better sort out the effect, we need to consider the network connectivity and trust problem size together. For instance, if we look at the trust scores of a node performing an opportunistic service attack in Fig 11a and focus on $mnp = 10\%$, we do not observe a significant difference in trust scores from $nn = 1$ across $nn = 10$ except for a slight shift (increase) in average trust belief. This sounds counter-intuitive. Overall, we would expect a malicious node to have a lower trust score if it is connected to more nodes in the network because its incorrect report of readings will be revealed by more nodes. Yet, if we also look at trust scores for other *mnp* values, we notice that higher network connectivity levels make the drop in trust scores more pronounced after the attack starts. Thus, we can interpret these observations for the opportunistic service attack as follows: *"Larger network connectivity levels become critical to detect opportunistic service attack for larger trust problem sizes, i.e., malicious node percentages in a network."*. Figure 12 shows the relationship between node connectivity levels and difference in average trust scores from a hypothetical trust level (0) for opportunistic service attack after attack starts (after 200th data point). As seen in the figure, for $mnp = 10\%$, the optimal node connectivity level seems to be 1 because for larger *nn* values, difference increases. For $mnp = 30\%$, $mnp = 50\%$, and $mnp = 70\%$, the optimal node connectivity level seems to be 3 as seen from the knee points on the lines. For $mnp = 90\%$, and $mnp = 100\%$, difference in average trust scores increases first and then starts to decrease. We may say that $nn = 5$ is the optimal ode connectivity level because for larger *nn* values, the decrease in the difference is not significant.

If we look at Fig 11b, we see no significant difference in average trust scores among different network connectivity levels when $mnp = \%10$ for an individual malicious node that always reports incorrect readings. There is only a slight increase in trust belief (from $nn = 1$ to $nn = 3$ only) and a decrease in the variation of trust scores. For higher *mnp* values, increasing network connectivity leads to a lower average and a variance in trust scores. Again, we can interpret these results for the individual malicious node attack similar to those

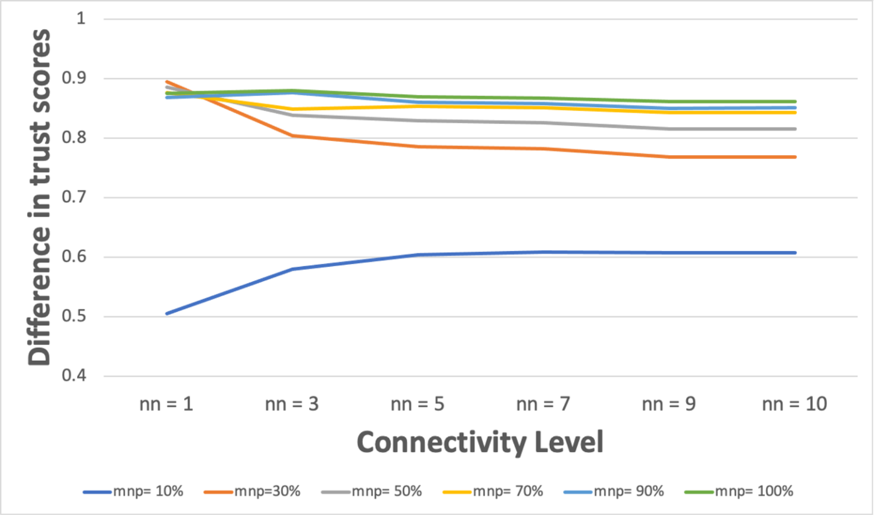


Figure 12: Network connectivity level vs. difference in avg. trust scores from a hypothetical ground truth value

for the opportunistic service attack. *"Larger network connectivity levels become critical to detect individual malicious node attack for larger trust problem sizes, i.e., malicious node percentages in a network."*.

The trust scores of a node performing a patterned on-off attack in Fig 11c have patterns similar to those in opportunistic service and individual malicious node attacks for $mnp = 10\%$. In other words, average trust scores increase, and their variance decreases when network connectivity increases. Yet, for higher mnp values, we do not observe any change for changing network connectivity. Trust scores of the malicious node stay high, and including trust opinions of more nodes does not affect this. This finding confirms the observation in the network trust literature [23, 108, 107]; *On-off attacks are challenging to detect by trust management schemes as malicious nodes alternate their good and bad behaviors not to get recognized.*

Finally, looking at the trust scores of a node performing a random on-off attack in Fig 11d, results are pretty much similar to those of an on-off patterned attack. The only difference seems to be the lower variance in trust scores, especially observed more clearly for $mnp = \%10$.

Putting all these discussions together, we can summarize the effect of network connectivity on trust scores as follows; *Network connectivity has an interaction effect on trust scores.*

More precisely, it affects trust scores differently for low and high mnp values. For higher mnp values, increasing network connectivity leads to lower trust scores, especially for individual malicious node and opportunistic service attacks. For smaller mnp values, increasing network connectivity leads to higher and/or less varied trust scores. As a result, higher network connectivity levels are needed for higher mnp values. Lower network connectivity can be a better choice for smaller mnp values because nodes cannot conceal their malicious behaviors when network connectivity is low, given the lower trust scores for lower mnp values.

5.2.3.2 Effect of Trust Problem Size on Trust Scores

Interpreting the effect of trust problem size is less challenging than the effect of the network connectivity. The plots in Fig 11 point to the obvious finding; *Networks with a larger trust problem size lead to higher trust scores than networks with a smaller trust problem size.*

To remind, we consider that the trust problem size comprises two factors; malicious node percentage and attack type, based on our definition earlier in the chapter. A higher malicious node percentage means a larger trust problem size (or a more difficult trust problem for a network). Thus, as also observed in plots in Fig 11, higher malicious node percentages lead to higher trust scores, which means less drop in scores after/during attacks and more concealing of malicious behaviors. On the other hand, it is nontrivial to hypothesize a monotonic relationship between trust attack types and trust problem size. Looking at plots in Fig 11, we see that in none of the patterned and random on-off attacks, trust scores do not drop as much as in opportunistic service or individual malicious nodes. Thus, we can reason that on-off attacks should be closer to the larger trust problem size. Yet, we cannot rank individual malicious nodes and opportunistic service attacks easily. Similarly, the nuance is not apparent between the on-off patterned and random attacks.

5.3 Summary and Discussion

In this chapter, I explored the effect of network connectivity and trust problem size on trust scores in our previously proposed MADM-EBSL trust computation framework for IoT. As the MADM-EBSL is a transitive framework, the results of experiments shed light on our initial question; *How trust scores in an IoT network are affected by trust transitivity through varying network connectivity and trust problem size?* Results show that higher network connectivity levels lead to a more apparent decrease in trust scores of a malicious node for larger trust problem sizes (Larger trust problem sizes correspond to higher malicious node percentages in a network and attacks more challenging to detect, such as on-off attacks.). The results also show that a larger trust problem size leads to higher trust scores for a malicious node.

6.0 An Extension to the MADM-EBSL Trust Framework for IoT Networks

In this chapter, I propose a trust computation framework for IoT by extending our previously proposed MADM-EBSL framework in [5]. The extension is based on trust concepts we derived in our review of trust paradigm in the social sciences literature. More precisely, I include additional attributes in MADM process for trust computations based on the factors that may affect trust towards IoT devices and their reported measurements.

In the following, I present background on and motivation for the factors I selected among those that affect trust in social relationships and reflected in trust computations in the extended framework. Next, I discuss the details of the extended trust computation framework. Following this, I present experimental evaluations to compare the extended trust framework to the MADM-EBSL framework. Finally, I summarize the contributions and discuss the limitations of/potential future work for the research in this chapter.

6.1 Background and Motivation

For this task, I extend the MADM-EBSL framework by including additional factors that affect trust computations in IoT networks. These factors correspond to trust-related information that captures *multiple device vendors,* the *social presence of IoT devices,* and *the source of a replicated measurement.* In this section, I present a high level overview of them and motivation behind including them in the proposed framework. I present more details about their incorporation into the MADM-EBSL framework and discuss some exceptional cases in Section 6.2.3.

I argue that the *device vendor* could affect the trust towards a measurement reported by an IoT device. For instance, the perception about the trustworthiness of a device vendor within a community of an IoT network domain users/admins or the reputation of a device vendor within the larger community could impact the trust of IoT devices manufactured by the vendor. Previous research has also explored similar factors as attributes for computing trust of IoT devices, such as device type (e.g., printer, camera, air conditioner, etc.), device manufacturer, and device model [170]. Another reason to include vendor trust may be digital sovereignty issues [142, 175], i.e., is a device from a competitor or nation that is adversarial? I consider including device vendor as one of the MADM attributes for computing the *functional trust* of a measurement reported by an IoT device.

For the inclusion of the *social presence of IoT devices* as a factor in IoT trust computations, I was inspired by the Social Presence Theory (SPT) [164] that is widely explored for trust formation in the online commerce field [3]. This is one of the implications we derived for IoT networks from our review of trust in social sciences literature in Chapter 3. SPT was originally proposed to assess the degree to which a computer-mediated communication medium enables perception about personal, sensitive, and sociable human contact as a communication partner, i.e., an experience for communication partners as being psychologically present [54]. SPT was adopted to reflect the social presence of online business websites in the e-commerce discipline. A positive association between social presence and trust towards a website has been observed by several researchers (e.g., [88, 101]). Also, the interactivity of a website, i.e., users being responsive and interact with each other, is the main component that affects the social presence of an online platform [101]. As a result, a more interactive user community of an e-commerce website would result in a higher social presence perception, which in turn leads to higher trust by users. I propose that we may think about social presence as a factor in trust development in IoT networks. As such, the social presence/absence of an IoT device can be measured by the level to which it is being active/inactive. Hence, interactive devices can be rewarded with higher trust. The trust of devices that join the network and stay inactive for a long time after a series of interactions can be degraded. Using social presence as a trust attribute may also help provide defense against the white-washing attack (see Section 2.5) as it will punish inactivity for a long period[1].

The idea for including the *source of a replicated measure* and *the success of replication* as trust attributes comes from our survey of trust in social sciences ([3]), as well. Hendriks et al. [71] investigated if the trustworthiness of a researcher and the credibility of a study by the researcher is affected by who replicated the original study, —*replication source*—,

[1] For details and exceptional cases, see Section 6.2.3

and if the replication was successful or not, —*replication success*. They found that both the credibility of a research study and the trustworthiness of its author(s) are higher for successful replications, regardless of the replication source (i.e., the authors themselves or other researchers). I argue that the *replication source* and *replication success* should be included in a trust framework for IoT, as well.[2] For IoT networks, we can consider an IoT device and a measurement reported by it as analogous to a researcher and a scientific study, respectively. In the IoT context, the counterpart of a replication study can be a *measurement replication* that corresponds to multiple reporting of a measure, such as temperature readings coming from the device itself or from other devices in a predefined short time window. Replication is needed for measurements reported by IoT nodes to provide improved service availability. Thus, the *replication source* in the IoT scenario will be *replicating IoT devices*, i.e., the device itself or other devices. *Replication success* could be thought of as the deviation of a reported measure from other reported measures within the predefined time window. In other words, the less deviation in a reported measurement compared to measurements reported close in time by the device itself or close in proximity by other devices, the more successful, i.e., accurate, it is.

The reason for including replication success is obvious; to spot inaccurate/deviating measurements reported by a node and decide if a node is malicious or malfunctioning. We captured the notion of replication success in our previously proposed MADM-EBSL framework as *measurement deviation* attribute in referral trust computations (see Chapter 4). Thus, the trust framework I propose in this section captures replication success because it adds to the existing set of attributes in the MADM-EBSL. As opposed to the findings by Hendriks et al. [71] that the replication source does not impact the trustworthiness of a researcher or the credibility of a study, I propose the replication source may affect the credibility of a reported measure by an IoT device. It is essential to include the source of a replicated measurement for trust computations in IoT as there may be a single device reporting a mea-

[2] For IoT networks, we may think that the *trustworthiness of a device* and the *credibility of a reported measure* are different concepts, and they may be affected by the replicating device and replication success, similar to that the trustworthiness of a researcher and the credibility of a research study are different concepts. Device trustworthiness and measurement credibility are different concepts because, for example, a device may be hacked by an attacker (untrustworthy device) but still may report correct measurements (credible measurement) for hiding its malicious behavior until an attack, i.e., opportunistic service or camouflage attack (see Section 2.5 for attacks to trust management).

surement type. In this case, the MADM-EBSL sets measurement deviation attribute to zero so discards it from computations. This may lead to an increase in opportunistic service or camouflage attacks as the framework cannot capture inconsistent service quality by nodes that know they can report highly deviating measurements as their measurements will not be compared to any other reference values. As a result, I propose to capture the replication source by including measurement deviation attribute both in functional and referral trust computations.

Finally, it is noteworthy to mention that I do not consider including node recommendations and feedback into my proposed solution, though these are widely adopted mechanisms used in previously proposed IoT trust computation solutions. The reason is that a service should be provided in an automated and programmable way to be delivered timely and practically [83], including trust measurement/management service in IoT. As I also discuss in Section 2.4, when node recommendations or transaction feedback is included in a trust measurement solution, it is likely that human intervention is needed. Thus, these mechanisms could render an automated trust computation solution ineffective.

6.2 Extended IoT Trust Framework

6.2.1 Overview of the Framework

As the proposed IoT trust framework in this chapter extends our previously proposed MADM-EBSL framework, it is based on the same system model. The overall process of trust computations and the algorithm used for computing trust opinions are the same. The difference between the two is the set of trust attributes used in the MADM process for functional trust score computations.

6.2.2 MADM Problem Definition for Trust Computations

The MADM problem definition for trust computations in the extended framework is mostly the same with that for the MADM-EBSL model, — both for FT and RT computations. Here, I do not repeat all the details of this process. Yet to summarize, the steps are; *i*)initial representation of the decision matrix (IM_{mt}) using a select set of decision criteria (for FT and RFT separately), *ii*) normalizing the decision matrix D, and *iii*) weight elicitation and overall score calculation. As aforementioned, the difference lies in the set of selected decision criteria (trust attributes) for FT. In the following, I shed light on the FT attributes included in the extended framework.

6.2.3 Selected MADM Criteria for FT Computations

In the MADM-EBSL framework, we included the time of key establishment and temperature as criteria in MADM definition for FT computations. We used the key freshness and the ambient temperature of a node as indicators (a.k.a. FT attributes) to represent these factors. In addition to these, the extended framework includes *device vendor*, *social presence*, and *replication source* as the decision making criteria in the MADM process for FT computations. Table 9 shows all the MADM criteria used in the extended framework for trust computations.

As Table 9 shows, I propose to measure device vendor trust attribute using *data reliability due to device vendor*, which is a qualitative monotonic benefit indicator, similar to link type attribute in the MADM-EBSL (see Chapter 4 for attribute types). As it is a qualitative monotonic attribute, we can map its non-numeric values to numeric values using a 10-point Bipolar scale [76]. Figure 13 represents this mapping. Although it is beyond the scope of this dissertation, using the common vulnerabilities database [32] may be possible to rank the vendor trust.

For representing the *social presence* of an IoT device, we need an indicator (attribute) that addresses a variety of node types. Note that there maybe some IoT nodes that operate in receive-only mode or passive by their nature, such as due to resource constraints. For these nodes, assessing their social presence using a universal interaction level would be misleading. Thus, the baseline level could be set concerning the expectations from devices, such as by referring to their specifications. I consider *device-based duty-cycle* as the indicator. The concept comes from the RFID (Radio Frequency Identification) Systems. It represents

Criteria	Indicator	Unit	Type	Utility type
Time of key establishment	Key freshness[a]	Days	Quantitative	Benefit
Temperature	Ambient temperature of the "thing"	Fahrenheit	Quantitative	Non-monotonic
Device vendor	Data reliability due to vendor	-	Qualitative	Benefit
Social presence	Device-based duty cycle	-	Quantitative	Non-monotonic
Measurement self-deviation	Absolute measurement self-deviation [b]	unit of mt	Quantitative	Cost
Time of data collection	Data freshness[a]	Seconds	Quantitative	Benefit
Measurement deviation	Absolute measurement deviation [c]	unit of mt	Quantitative	Cost
Link type	Data reliability due to link type	-	Qualitative	Benefit
Location	Distance between two nodes	Meters	Quantitative	Cost

[a] Formula respectively for key freshness and data freshness are: $1/(T_{key\_est} + 1)$ and $1/(T_{data\_col} + 1)$, where T_{key_est} and T_{data_col} are time elapsed since last time a cryptographic key was established and a data item was collected, respectively.

[b] Formula for absolute measurement self-deviation is: $|d_t(A) - d_{t-1}(A)|$, $mt_j \in MT$.

[c] Formula for absolute measurement deviation is: $|d_t(A) - d_t(B)|$, $mt_j \in MT$.

Table 9: Criteria Used in the Extended Framework for Trust Computations

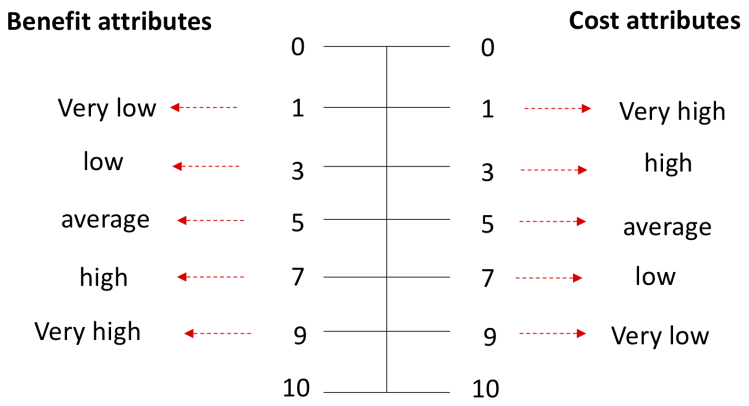


Figure 13: Bipolar scale for quantification of qualitative indicators

the percentage of the time a reader is transmitting a signal/emitting energy/ communicating/being "on" [82, 139]. If we generalize this concept to other protocols, we can consider *duty cycle* as the proportion of the "on" time to "off" time of an IoT device. As a result of my research for modeling duty-cycle indicator, I have found two possibilities which I call *protocol-based* and *device-based* duty-cycle values. Protocol-based duty-cycle values correspond to the worst case limits imposed by regulations/standards. Device-based duty-cycle values come from best practices and most preferred duty-cycle limits. In my framework, I consider *device-based duty-cycle* because regulations are not always imposed on duty cycle values. It is challenging to use a preset value/interval for duty cycle based on protocol requirements. There are countries or communication protocols which do not have any regulation/limitation on duty cycle. So, manufacturers identify limits based on the need of device/applications. Also, for many cases, it not uncommon to let the developers/deployers/admins to set the desired duty cycle limits. Following best practices/ the most commonly used duty cycle limits is another option to keep up with regulations and for energy saving purposes. For example in [4], 1% is mentioned as the most preferred value of the maximum duty cycle for Lorawan in different regions. I also argue that neither extremely low nor high interactivity/social presence is desired for a trustworthy IoT device, as both could signal malicious or faulty behaviors. Thus, this attribute can be considered as a non-monotonic attribute that is used for functional trust computations utilizing the MADM (see Chapter 4 for attribute types). In simulations for this research, I consider a water level sensor by following the sample scenario in our MADM-EBSL paper [5]. Based on a sample water level sensor specification [161], I set the $LPV = 65\%$ and $UPV = 85\%$[3]

I propose to represent *replication source* factor by including measurement deviation attribute both in functional and referral trust computations. That is, the measurement deviation attribute in the functional trust score (herein *measurement self-deviation*) will reflect the replicated measurements by a node itself, i.e., how close measurements are reported by an IoT node within a short time interval. The measurement deviation attribute in the referral trust score will reflect the replicated measurements by neighboring nodes, i.e., how close measurements are reported by neighboring nodes of an IoT node within a short time interval. As is the case for the measurement deviation attribute of referral trust in the MADM-EBSL, measurement self-deviation attribute of functional trust is a quantitative cost attribute (see Chapter 4 for attribute types).

6.3 Experimental Evaluation

Experimental evaluation for this research task aims at investigating the effect of including additional trust attributes into the MADM-EBSL framework. For this purpose, I used synthetic data sampled from real datasets and compared trust opinions computed by the MADM-EBSL framework to those computed by the extended framework. Next, I describe datasets used in experiments and the evaluation method. Finally, I present the evaluation results.

6.3.1 Dataset and Simulation Setup

In this experimental evaluation, I utilize the same dataset and similar setup I do with MADM-EBSL in chapters 4 and 5. Readers interested in the details can refer to Section 4.3.2.1 in Chapter 4 and Section 5.2.1 in Chapter 5. To recap, the dataset [141] includes weather data collected in USA 2012. Nodes are assumed to report temperature as the

[3] because according to the specification, duty cycle of the water level sensor in water is $75\% \pm 10\%$.

measurement type. They are distributed in a rectangular area through Poisson point process with $\lambda = 0.001$ yielding 22 nodes. Similar to the experiments in Chapter 5, I generate multiple versions of the same network, which have different levels of network connectivity (nn) and malicious node percentage (mnp).

I followed a similar approach for generating 300 data items for each node. To generate these data, I consider the MADM attributes mentioned earlier in Chapter 4 and additional attributes discussed in this chapter. Thus, FT attributes include *key freshness, temperature, data reliability due to vendor, device-based duty cycle,* and *measurement self-deviation.* RT attributes comprise *measurement deviation, distance, link type, and data freshness.* Interested readers are referred to Section 4.3.2.1 of Chapter 4 for details of generating values for attributes in the original MADM-EBSL. I generated data reliability due to vendor values randomly using Uniform distribution $U(1, 9)$[4]. I draw random values from a Gaussian distribution N(75, 10) to model duty cycle (remember the best practice value explained in Section 6.2.3. For measurement self-deviation, I used the generated node temperature values. I calculated measurement self-deviation values by the absolute difference of temperature values in two consecutive observations.

Also I investigate three different attack types as in Chapter 5; individual malicious nodes, opportunistic service, and on-and-off attacks. I consider these attacks for modeling malicious behaviors as well as benign and faulty node behaviors and generating node reported data (see Section 4.3.2.2 for details).

6.3.2 Evaluation Method

In the experiments in this chapter, I compare patterns in trust opinions generated by MADM-EBSL for a destination node to those generated by the extended framework with additional trust attributes. Similar to experiments in Chapter 5, I randomly select a malicious node in a TN (randomly generated network) as the destination node and compute trust scores and opinions for this destination node. I implemented the steps of Algorithm 1 to compute trust scores and opinions for the destination node. Likewise as in Chapter 5,

[4] We do not specifically assign a vendor to a node, or assign such vendors a reliability or trust.

comparisons are made for different network settings with varying network connectivity and trust problem size levels. The malicious destination node is the same across each network sample. Noteworthy mentioning is trust history configurations used in evaluations. As I found that a sliding window approach is superior to approach that considers the whole trust history in Chapter 5, I followed the same method for trust computations here.

6.3.3 Results

In this section, I discuss the findings of experimental evaluation. I compare trust opinions generated by the MADM-EBSL framework to the ones by the extended framework to figure out how including additional trust attributes affect trust opinions. There are two comparisons discussed in the sequel; OT and HFT belief scores. All comparisons for the destination malicious node are made concerning varying levels of network connectivity and trust problem size. In the sequel, I discuss findings on the effect of adding trust attributes on trust scores.

6.3.3.1 Effect of Additional Trust Attributes on Overall Trust

I have explored several cases for investigating the effect of including additional trust attributes in the MADM-EBSL on the overall trust. Specifically, I compared the MADM-EBSL (herein base model) to the extended model where *i*) only temperature (FT attribute) and measurement deviation (RT attribute) are random and all other RT attributes and additional FT attributes are fixed at their max value, *ii*) temperature and newly added FT attributes are all random and RT attributes are fixed at their max value except the measurement deviation, *iii*) the same configuration with *ii*, but the ranking of the FT attributes (so, attribute weights) changes[5]. All three cases yielded very similar results, so I only report the results of the first case.

Fig 14 display the overall trust belief scores for the base and extended model for $nn \in \{1, 10\}$ and varying mnp levels. For better readability purposes, I only presented results for

[5] Thus, the attribute of interest (and the one which is ranked first among other FT attributes) is not only temperature, but I also changed rankings such that one of the newly added FT attributes is ranked first in each case

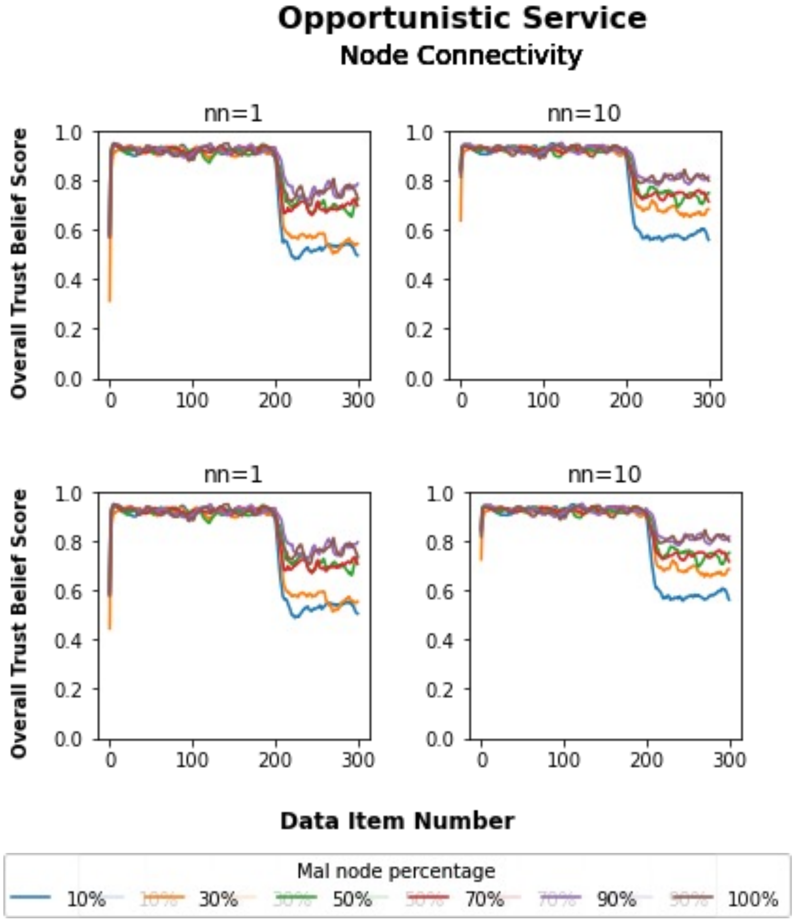


Figure 14: Overall trust belief scores for a malicious node performing opportunistic service attack when $nn \in \{1, 10\}$ and varying mnp levels computed by base (top) and extended (bottom) models

two nn levels for only the opportunistic service attack. The results for all node connectivity levels $nn \in \{1, 3, 5, 7, 9, 10\}$ are shown together for all attack types in Appendix B in Figs **??**, **??**, **??**, and **??**. The figures show that there is no significant difference in overall trust belief scores between the base and the extended model for none of the nn and mnp levels, and attack types. As scores look almost the same between the base and extended model, I wanted to take a deeper look at them by visualizing these in a different format as in Fig 15[6]

[6] The results plotted in this figure are generated by the case where only temperature (FT attribute) and measurement deviation (RT attribute) are kept random, and all other FT and RT attributes are fixed at their max value, including the additional FT attributes in the extended model. In these computations, temperature and measurement deviation are ranked first, so they have the highest importance and the weight in MADM computations.

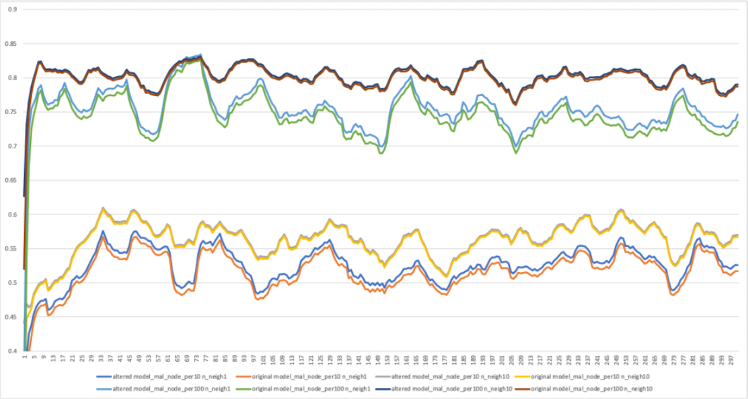


Figure 15: Overall trust belief score comparison between the base and the extended model for individual malicious node attack and select *nn* and *mnp* levels

I randomly picked the individual malicious node attack as a case here because the results were very similar for other attack types, as well. As observed in the figure, overall trust belief scores generated by the baseline and the extended model are not exactly same, but there is a slight difference between the two. This signaled a clue that additional attributes in deed may have a greater impact, but are just shaded when FT is molded into the overall trust with RT. This led the additional analysis of FT scores in the next section.

6.3.3.2 Effect of Additional Trust Attributes on Functional Trust

In this section, I present the results of simulations for investigating the effect of additional trust attributes on HFT belief scores concerning the three cases I described in Section 6.3.3.1.

Fig 16 shows the case number i). There are two common patterns observed for different attack types;

- Including additional trust attributes with fixed max values in functional trust component leads to a reduce in the variance of FT scores.
- Including additional trust attributes with fixed max values in functional trust component leads to an increase on FT scores on average.

The leftmost column in Fig 17 displays the results for the case number ii). The same patterns above apply to the FT scores here; reduced variance and increased average of FT trust scores obtained as a result of additional attributes in the FT component, which are

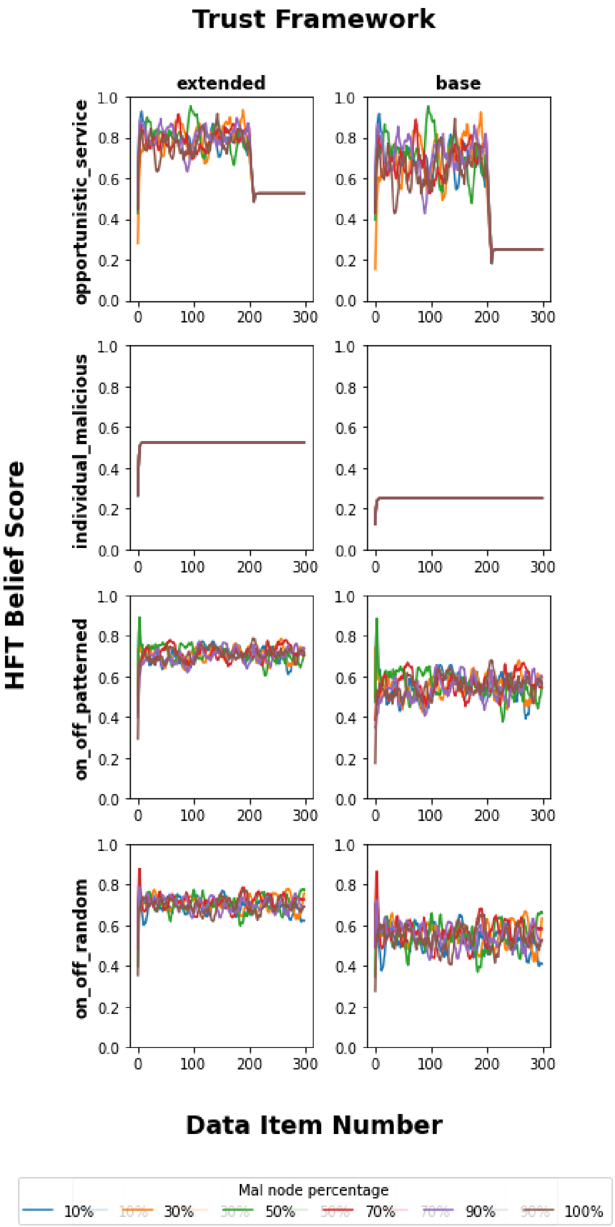


Figure 16: Historical functional trust belief comparison between the base and extended model for all attack types and *mnp* levels (*nn* is fixed to 1 as other *nn* levels have similar results) where only temperature have random values among other FT attributes.

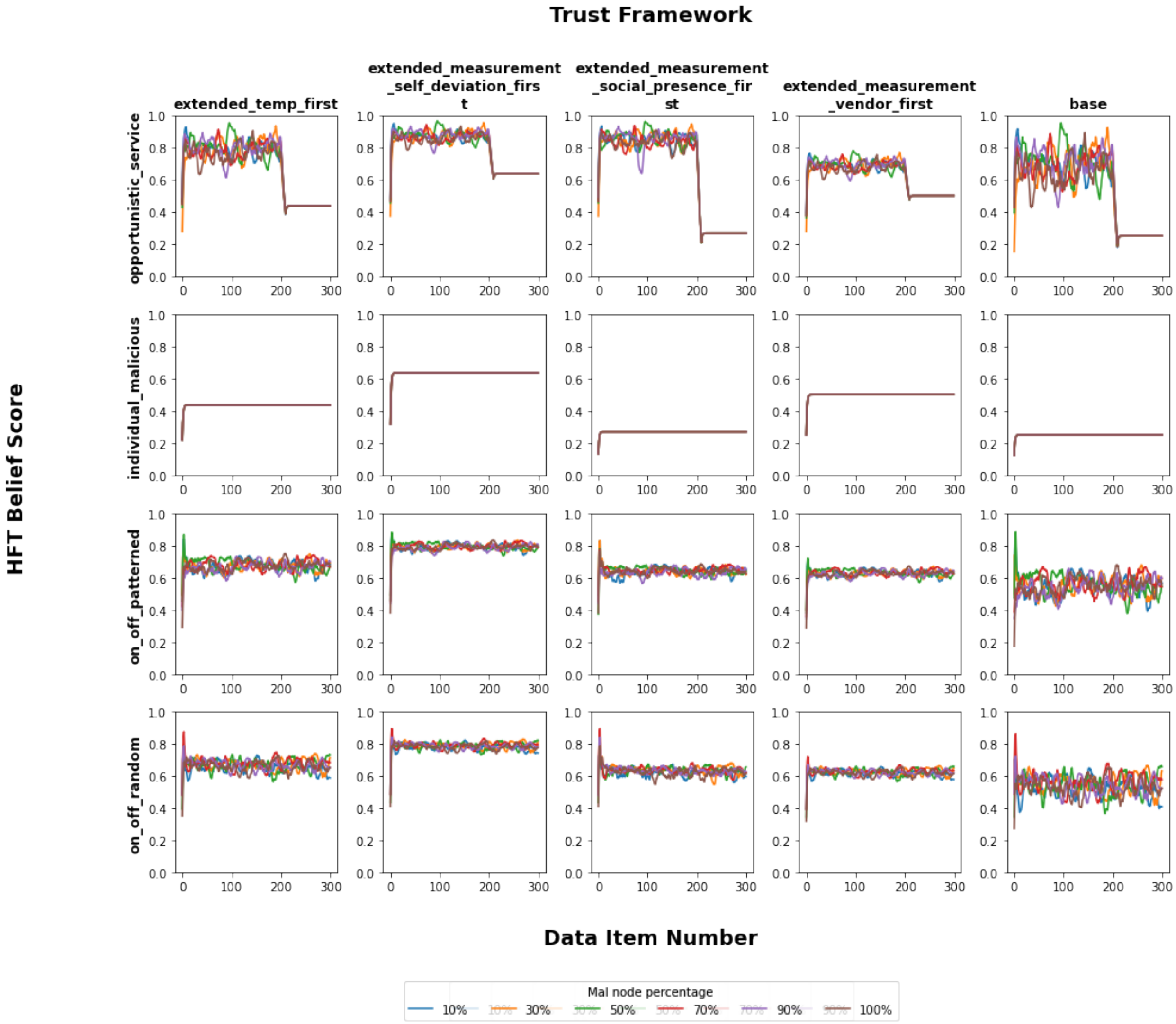


Figure 17: Historical functional trust belief comparison between the base and extended model for all attack types and *mnp* levels (*nn* is fixed to 1 as other *nn* levels have similar results) where temperature and newly added FT attributes have random values.

kept random.

The remaining four columns in Fig 17 compare the base model to the extended model by changing the ranking of FT attributes and putting one of the newly added attributes to the first place at a time. This corresponds to the case number *iii*) and changes attribute weights in the MADM process. When we look at the results within each attack type, we do see a difference in HFT score patterns concerning the attribute rankings/weights. I present average HFT scores produced under different attack types in Table 10. For instance for opportunistic attack, HFT scores are averaged around ([before attack, after attack]);

- $[0.7, 0.25]$ for the base model where only key freshness and temperature are included as FT attributes
- $[0.8, 0.4]$ when temperature is ranked first/have the highest weight among other FT attributes in the extended framework
- $[0.9, 0.6]$ when measurement self-deviation is ranked first/have the highest weight among other FT attributes in the extended framework
- $[0.9, 0.25]$ when social presence (i.e., device-based duty cycle) is ranked first/have the highest weight among other FT attributes in the extended framework
- $[0.7, 0.5]$ when device vendor (i.e., data reliability due to vendor) is ranked first/have the highest weight among other FT attributes in the extended framework

There are similar patterns for all attack types. For instance, ranking the social presence attribute first yields similar results to the base model in terms of average HFT scores. The reason for this is social presence is a non-monotonic attribute like temperature attribute, so the data are generated for them using the same procedure in the simulations. This leads to results where it seems adding the social presence attribute does not change HFT scores. Yet, in the real settings, when social presence attribute takes different series of values than those generated in simulations for this chapter[7], results can be different. Also, ranking the measurement self-deviation (in reported temperature readings) first leads to higher average HFT scores compared to ranking the temperature first. This could be because while the measurement self-deviation of the malicious node could be small (i.e., measurement reported

[7] In my simulations, social presence is represented through device-based duty-cycle, which I generated using a truncated Gaussian distribution with $\psi = (75, 10, 0, 100)$.

Attack	**Extended framework FT attribute order**				**Base model**
	T first*	MSD first	SP first	V first	
Opportunistic service (before attack, after attack)	0.8, 0.4	0.9, 0.6	0.25, 0.9	0.5, 0.7	0.25, 0.7
Individual malicious (after attack)	0.4	0.6	0.25	0.5	0.25
On-off patterned (after attack)	0.7	0.8	0.65	0.6	0.5
On-off random (after attack)	0.7	0.8	0.65	0.6	0.5

T: Temperature, MSD: Measurement self-deviation, SP: social presence, V: vendor

Table 10: Average HFT scores produced by the extended vs. base model for different attacks

by the node in a short time-frame are close to each other and do not deviate drastically) and does not drop its FT score, temperature attribute may cause to degrade its FT scores because the reported measures could be far different from the expected values (i.e., LPV and UPV of temperature as a non-monotonic attribute).

Finally, similar to the findings of experimental evaluation in Chapter 5, both patterned and random on-off attacks seem to be harder to detect compared to opportunistic service and individual malicious node attacks as average FT scores are higher under on-off attacks.

6.4 Summary and Discussion

In this chapter, I explored the effect of including additional trust attributes on trust scores computed by our previously proposed MADM-EBSL [5] framework. I proposed to include multiple device vendors, the social presence of devices, and the replication success and source of a reported measurement as additional functional trust attributes in trust computations. I compared the outcomes of this extended framework to that of the MADM-EBSL concerning overall and historical functional trust scores. While an additional set of functional trust attributes did not yield significant difference in the overall trust beliefs towards a selected node, it did so for HFT belief scores.

I tried various scenarios in terms of the randomness/stability of attribute values and the ranking (weights) of included FT attributes. Results showed that including additional FT attributes with only fixed values at their max leads to an increase in the average and a decrease in the variance of HFT scores. When these attributes are set random, their rankings (weights) affect the computed HFT scores differently. For instance, ranking the device vendor first leads to higher HFT scores on average compared to ranking the social presence attribute first, for opportunistic service and individual malicious node attacks.

7.0 Trust Repair Strategies and a Trust Model for IoT

In this chapter, I discuss the importance of trust repair and trust repair strategies for IoT drawing upon our review of trust in social sciences in [3]. Then, I discuss trust repair processes as a part of a trust management framework for IoT networks. I propose a trust repair model that models trust value changes of IoT devices after trust repair actions. Finally, I evaluate the effectiveness of the proposed trust repair model through simulations.

7.1 Motivation

When a trustor puts trust in a trustee, it makes itself vulnerable because a trustee may not fulfill its duties due to a lack of skills or motivation [3]. A *trust violation* occurs if a trustee exploits the vulnerability of a trustor. *Trust repair* is restoring the broken trust partially or completely after a trust violation [3]. Trust is easy to break and challenging to repair [3]. If violations repeat in a trust relationship, trust repair becomes even more challenging [102].

Trust repair is indicated as one of the under-investigated and least theorized research areas in different fields, including organizational science, business, and psychology [3]. It is also an untouched topic for IoT networks. To the best of my knowledge, trust repair has not been explored before in the IoT trust literature. A preliminary literature search on GoogleScholar has yielded no relevant result[1]. Although some papers discuss legal issues and propose conceptual frameworks for trust repair in cloud computing (e.g., [114, 113]), there is a gap in the literature for a computational model of trust repair in IoT networks.

[1] I used keywords *"trust repair" iot* in *anywhere in the article* to find a relevant paper.

7.2 Trust Repair in IoT

Given the lack of existing research for trust repair in IoT networks, I present terminology for it and speculate about IoT trust repair strategies/trust repair actions in this dissertation, drawing upon our review of trust in social sciences in [3]. *Trust repair in IoT environments* corresponds to restoring the trust of an IoT device after it commits trust violations, such as due to malfunctioning or being compromised. This restoration is typical to be realized by human intervention, so trust gets repaired or reset. In other words, the historical trust score (see Chapter 4) of an IoT device can be reset to a level after a trust repair action. For example, we can consider resetting a device by a company, firmware upgrade, or any maintenance as trust repair actions. Periodic resets may become a feature in IoT if trust metrics are widely adopted.

In social sciences literature, it has been argued that the effectiveness of trust repair efforts is determined by *trust violation type* that could be *competence-based*, *benevolence-based*, or *integrity-based* trust violations [58], i.e., whether violation occurs due to concerns of competence, benevolence, or integrity. These three dimensions are characteristics ascribed to a trustee by a trustor to decide about its trustworthiness, according to the widely cited ABI (Ability, Benevolence, Integrity) model [118] in social sciences literature. Competence refers to perceptions that a trustee has enough skills and knowledge to perform a task. Benevolence means a trustee cares for and acts in the interests of the trustor. Integrity is the perception that a trustee is honest, credible, and keeps promises. Sometimes benevolence and integrity are considered together as a combined trust construct *goodwill*, as both reflect ethical traits [3]. Tomlinson and Mayer [178] argue that integrity is the most stable among the three dimensions. Hence, if a trustor decides that low integrity is the cause of a trust violation, it is the hardest to repair the trust. I argue that *trust violations in IoT* can also be grouped as *competence-based* and *goodwill-based.* That is, if an IoT device violates trust due to the lack of capabilities or technical issues, such as network connection or transmission errors, it will be a competence-based trust violation. If an IoT device violates trust due to intentional malicious actions, it will be a goodwill-based trust violation (see Section 7.3 for more detail).

Different actions should be taken to repair different dimensions of trust [58]. To this end, researchers have focused on the grouping of trust repair strategies/actions as verbal and non-verbal strategies in social sciences literature [178]. In other words, verbal and non-verbal trust repair actions have been shown to impact/restore different dimensions of trust or have different effectiveness [3]. Similarly, I propose that trust repair actions in IoT can be grouped as *software-based* and *hardware-based/physical* actions, i.e., the *source* or *type of trust repair*. As software-based trust repair actions, we may think of rebooting an IoT node remotely using a watchdog software in case of a functionality issue (failures in network connection or processes) or getting over-the-air updates. Another example could be utilizing challenge-response protocols for trust repair such that if a node successfully responds to a challenge, then it may be deemed as trustworthy and its trust gets increased[2]. As for hardware-based trust repair, we may consider any action that supports an IoT node physically or with additional hardware resources through human intervention. For instance, a human may intervene by restarting an IoT device in the case of connection or process problems. Alternatively, a human operator may place guards to physically secure an IoT node or install hardware to support its processing or storage capability. I argue that different types of trust repair may affect trust scores differently. More precisely, a hardware/physical trust repair action may provide better guarantees than software-based repair actions, as the former includes a direct intervention of a human with stronger/more reliable solutions. Thus, we can model trust score changes after a trust violation based on the type of trust repair actions.

To sum up, I propose that trust repair processes in IoT and changes in trust scores may be affected by the type of trust violation, —*competence-based* and *goodwill-based*, and the type of trust repair, — *software-based* and *hardware-based.*

[2] The node may involve in a trust violation, but it maybe due to a failure (i.e., competence-based violation) rather than a malicious action, so its trust could be increased in case of a successful response.

7.3 The System Model of the Proposed Solution for Trust Repair in IoT

In this section, based on concepts discussed in the previous section, I discuss how trust repair can be incorporated into a trust framework for IoT. Figure 18 shows an initial system model. A central, powerful IoT node (a device with better processing and storage capabilities, such as a smart TV or a home assistant/hub as suggested in [1]) is decided as a trust manager responsible for trust management. Using a trust model, such as the one provided by the MADM-EBSL framework [5], the trust manager computes trust scores of nodes in the network and stores them. It also monitors trust scores to observe trust violations. Trust violations can be detected using an anomaly detection method or a simple heuristic, such as a trust score declines for recent n observations[3]. The system also monitors node interaction behaviors using a watchdog agent/software installed on each IoT node. Each node observes the following three types of interaction behaviors of its immediate neighbors using the watchdog agent, inspired from [188]: discarding data/orders, tampering with data/orders, and replaying data/orders. These interaction behaviors can be at three levels, —*normal*, *malfunctioning*, and *malicious* as decided by watchdog records. More precisely, if there is no discarding, tampering, or replaying/forging of data/orders, then the node behavior is recorded as normal. If there is a discarding, tampering, or replaying/forging of data/orders and if it happens due to network connection or transmission errors, then node behavior is marked as malfunctioning. If there is no obvious reason for a risky node behavior detected by the watchdog agent, then the behavior is marked as malicious.

When the trust manager detects a trust violation by an IoT node, it decides the type of violation by looking at interaction behavior records for this node. If an interaction is marked as malfunctioning, then the type of violation is decided as *competence*. In case it is malicious, then the trust violation type will be *goodwill*. If no trust repair action is taken after the detection of a violation, the trust score of the violator node will stay at the level as computed by the underlying trust measurement model[4]. Yet, if a trust repair action is taken, the trust score of the violator node will be adjusted by the trust manager using the

[3] This is out of the scope of this dissertation and a future research direction.

[4] The underlying trust measurement model should reflect trust violation as a drop in the trust score, so there should be no need for extra intervention to drop the score.

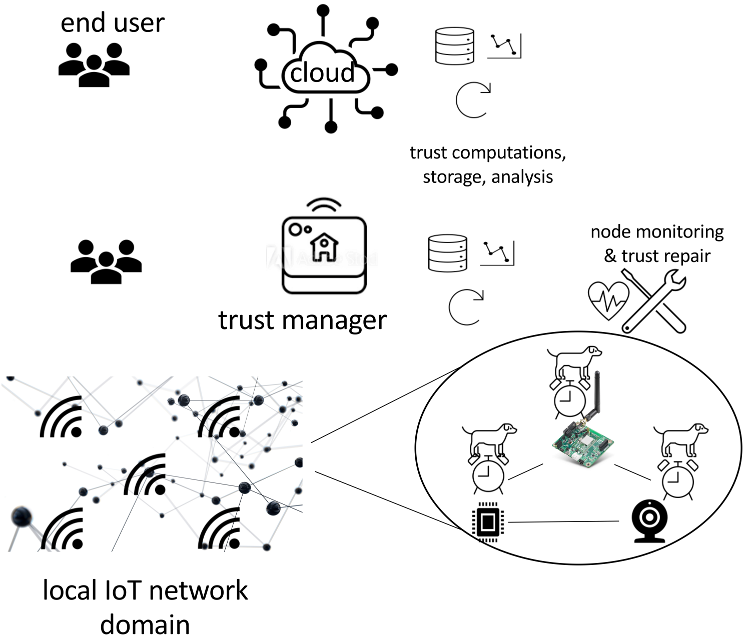

end user
cloud
trust computations, storage, analysis
trust manager
node monitoring & trust repair
local IoT network domain

Figure 18: System Model

violation type and trust repair action type (software- vs. hardware-based).

In the next section, I elaborate on the trust repair model to be used by the trust manager for reflecting trust violations and trust repair actions on trust measurements.

7.4 A Trust Repair Model for IoT Environments

Figure 19 presents a schematic representation of the overall trust repair process. As I discussed earlier, the result of trust repair actions can be reflected in historical trust scores. For the trust repair model proposed here, the MADM-EBSL[5] is the underlying trust framework used for trust computations. If we consider that an IoT device violates trust after a trust opinion is calculated for it at time t and a trust repair action is taken afterward, its overall trust opinion (OT_{dest}) will be updated. OT_{dest} reflects the historical trust of an IoT device as it incorporates historical functional trust opinion and historical referral trust opinions from other nodes in a network.

The update in $OT_{dest} = (b_{dest}, d_{dest}, u_{dest})$ depends on the type of trust violation by a

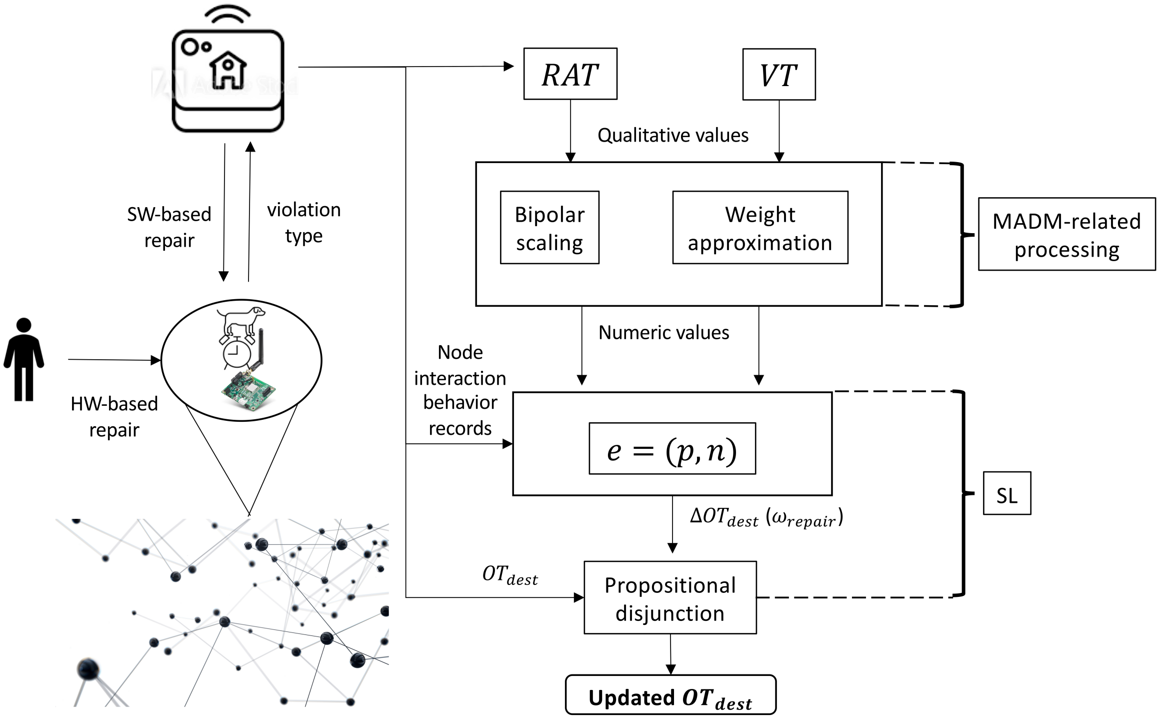


Figure 19: Trust repair process

node—VT, the type of trust repair action taken by network administrator—RAT, and the number of times a node commits a trust violation—n_V. Also, the update in trust should be reflected as an increase in the *belief* component of overall trust opinion after a trust repair action. As a result, an increase in the overall belief towards a node—Δb_{dest} will be directly proportional to numerical values that represent VT and RAT and inversely proportional to the number of violations. In other words,

$$\Delta b_{dest} \propto \frac{VT, RAT}{n_V} \tag{9}$$

There are several questions to consider to convert the relationships in (9) into a measurable form and integrate them with OT_{dest}.

1. How can we convert qualitative values of VT and RAT, i.e., $\{competence, goodwill\}$ and $\{software - based, hardware - based\}$ into numerical values?
2. How can we formulate Δb_{dest} or ΔOT_{dest} more generally?
3. What kind of mathematical/logical operator can we use to integrate ΔOT_{dest} and OT_{dest}?

Rank	2[a]	3	4	5	6	7	8	9	10
1	0.7500	0.6111	0.5208	0.4567	0.4083	0.3704	0.3397	0.3143	0.2929
2	0.2500	0.2778	0.2708	0.2567	0.2417	0.2276	0.2147	0.2032	0.1929
3		0.1111	0.1458	0.1567	0.1583	0.1561	0.1522	0.1477	0.1429
4			0.0625	0.0900	0.1028	0.1085	0.1106	0.1106	0.1096

[a] $w_j = 1/n \sum_{k=R_j}^{n} 1/k$, * R_j: rank of the attribute

Table 11: Attribute Weights Assigned by the ROC Weight Elicitation Method

To answer the first question, we can think in the same direction with the method we used in MADM-EBSL for the quantification of qualitative attributes. In MADM-EBSL, to transform a qualitative attribute (data reliability due to link type and device vendor, with values from very low to very high) into an interval scale, we utilized a 10-point Bipolar scale (see also Figure 13 in Chapter 6). We can assign a score for trust improvement for each VT and RAT value considering as if they are cost and benefit attributes used in the MADM, respectively. Yet, there are only two possible qualitative values to map. Thus, instead of using a 10-point Bipolar scale with equal intervals as in Figure 13, we can consider attribute weight approximation approaches for quantifying VT and RT (see ROC, RS, RR in Chapter 4). That is, we can rank VT and RAT values concerning their impact on trust improvement for an IoT device. Then, we can use, e.g., ROC to find the proportional importance of different values and convert them to integer values. More precisely, if we look at Table 11 for attribute weights with two attributes, we see that ROC respectively assigns 0.75 and 0.25 to the first and second attributes, concerning their rank of importance in a decision-making problem. RAT and VT can be considered analogous to a benefit and cost attribute with two possible values. The two possible values of RAT are hardware-based and software-based, respectively concerning their effect on improving the overall trust of an IoT node. The two possible values of VT are goodwill and competence, in order based on their importance for affecting the overall trust of an IoT node. Thus, we can consider that the first and second values of RAT and VT (i.e., concerning their importance) will be numbers proportional to 0.75 and 0.25.

To answer the second question, we revisit the definition of an opinion triplet in [77, 167]. Opinions are formed based on evidence. Evidence is denoted as a pair $e = (p, n)$, where p and n are positive finite numbers that respectively represent the amount of evidence supporting and contradicting a *proposition*. There is a one-to-one mapping between an opinion $x = (x_b, x_d, x_u) \in \Omega$ (opinion space) and its evidence $e = (p, n)$ as follows:

$$(x_b, x_d, x_u) = \frac{p, n, 2}{p + n + 2} \tag{10}$$

I formulate ΔOT_{dest}, i.e., change in the overall trust opinion for an IoT node after a trust repair action, using (10). The variables p and n correspond to evidence provided by RAT and VT that respectively supports and contradicts the following proposition: *A trust repair action act* $\in ACT$ *with rat* $\in RAT$ *after a trust violation with vt* $\in VT$ *by a node dest improves its overall trust* OT_{dest}. In connection with the solution I proposed to answer the first question, I argue that p can be assigned an integer value reflecting the proportion between the values of RAT, i.e., *hardware-based* $\propto 0.75$ and *software-based* $\propto 0.25$. Similarly, n can be assigned an integer value reflecting the proportion between the values of VT, i.e., *goodwill* $\propto 0.75$ and *competence* $\propto 0.25$.

To answer the third question, we should consider opinion combining operators of SL [77] as candidates because OT_{dest} is in the form of an SL opinion triplet. SL provides two fundamental sets of operations, namely *logical* operators to combine opinions of an agent about multiple propositions and *evidential* operators to combine opinions of multiple agents about a single proposition. Logical operators include *propositional conjunction*, *propositional disjunction*, and *negation*, which are the applications of standard binary logic operators AND, OR, and NOT on SL opinions. Evidential operators are *consensus* and *discounting*, which we have utilized in MADM-EBSL for trust propagation and aggregation in a trust network. A logical operator suits the needs of the research problem here, i.e., to reflect the improvement on a trust opinion towards an IoT node after a trust repair action, better than an evidential operator. The reason is that, in trust repair scenario, there is a single source of evidence, which is the *trust manager*, and multiple propositions about which it forms an opinion after a node involves in a trust violation. As explained in Section 7.3, the trust manager keeps track of trust opinions for nodes. In other words, it records OT_{dest} and knows

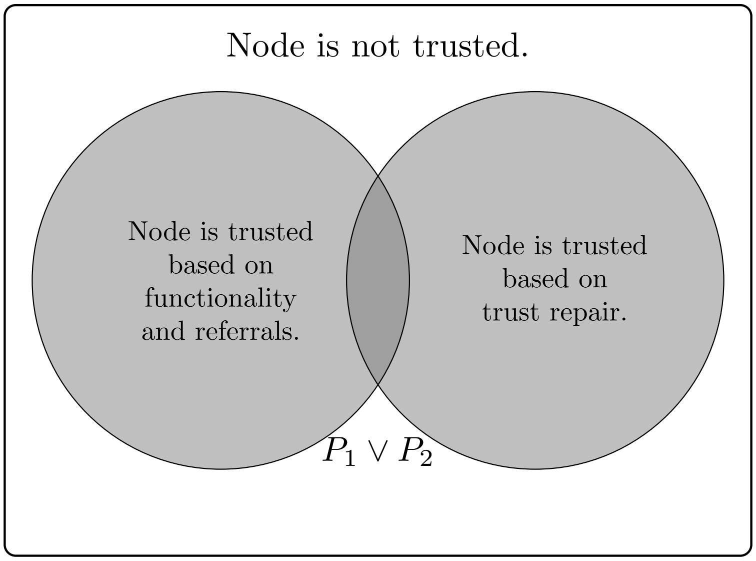

Node is not trusted.
Node is trusted based on functionality and referrals.
Node is trusted based on trust repair.
$P_1 \vee P_2$

Figure 20: Propositions to form a trust opinion towards an IoT node by a trust manager

of trust violations and trust repair actions as a central trust authority. It uses the trust repair model to adjust the trust score of an IoT node. Thus, as a single agent, the trust manager will be combining its opinion about the following two propositions: *i*) Node *dest* is trustworthy based on its functionality and referrals from other nodes in the network, and *ii*) The trust repair action taken after node *dest* violated trust improves its overall trust.

Among the logical operators, the proposed model uses the *propositional disjunction* operator (see Equation 11) because a disjunction returns true if at least one of the disjuncts is true, i.e., unless both of them are false. In a trust repair scenario, the trust of an IoT node can be denoted as the disjunction of two aforementioned propositions, and the existence of trust from either case will be enough to compute the trust of a node. Figure 20 illustrates these trust propositions and their relationship to form an opinion.

$$x \vee y = (x_b y_b, \quad x_d + y_d - x_d y_d, \quad x_b y_u + x_u y_b + x_u y_u) \tag{11}$$

7.5 Experimental Evaluation

In this section, I present the results of the experimental evaluation of the proposed trust repair model. The purpose is two folds; *i*) comparing the trust scores of a node computed originally by the MADM-EBSL model to those computed by the trust repair model to see how trust scores are affected in case of an intervention rather than leaving them on their own course, *ii*) investigating the impact of RAT and VT on trust repair. In the following, I discuss how I sampled synthetic data from real datasets to use in experiments. Next, I elaborate on the evaluation method. Finally, I present the results and discuss findings.

7.5.1 Dataset and Simulation Setup

For the experiments in this chapter, I utilized the same dataset and similar setup as in chapters 4, 5, and 6. More information on this can be found in Section 4.3.2.1 in Chapter 4, Section 5.2.1 in Chapter 5 and Section 6.3.1 in Chapter 6. To reiterate, the data consists of temperature measurements reported by 22 nodes distributed in a rectangular area using a Poisson point process with a λ of 0.001. These reported measurements are sampled from a dataset of weather data from the USA in 2012 [141].

For consistency with the original MADM-EBSL evaluations, I used a single network with a fixed level of malicious node percentage (10%) and node connectivity (1) levels. To focus on the effect of trust repair on trust scores, I used a single attack type for the single network, unlike in Chapters 5 and 6. Specifically, I used opportunistic service attack to model malicious behaviors, while also modeling benign and faulty node behaviors to generate node reported data (see Section 4.3.2.2 for details). Also, I simulated an additional behavior where a node behaves benignly and commits a trust violation only once (considering this as a random 299G-1B on-off attack).

I generated 300 data items for each node using an approach similar to that used in the original MADM-EBSL in Chapter 4, with FT attributes including *key freshness* and *temperature*, and RT attributes comprising *measurement deviation, distance, link type*, and *data freshness.* For further details on generating values for attributes in the original MADM-

EBSL, interested readers can refer to Section 4.3.2.1 of Chapter 4.

7.5.2 Evaluation Method

The objective of conducting experimental evaluation is to compare the trust opinions generated by the original MADM-EBSL with the proposed trust repair model outlined in this chapter for a designated destination node. To accomplish this, I randomly selected a node from each type, namely malicious, faulty, and benign, to act as the destination in a network with $mnp = 10\%$ and nn=1. I computed trust opinions for the destination node using Algorithm 1 in the original MADM-EBSL framework. Subsequently, I applied trust repair logic to these opinions as described in Section 7.4, using Equations 10 and 11. As for trust history configurations, I considered both using the whole history of trust scores and a sliding window approach to see the effect of trust repair both in short and long terms (see Chapter 5 for details on trust history configurations).

7.5.3 Results

In this section, I present the findings of experimental evaluation. I discuss the effect of trust repair on overall trust opinions (specifically, beliefs) through two main comparisons for the three aforementioned node types; *i*) comparisons of OT beliefs computed using the original MADM-EBSL to trust repair model for *the effect of trust repair action type (RAT)*, *ii*) comparisons of OT beliefs computed using the original MADM-EBSL to trust repair model for *the effect of trust violation type (VT).* Also, I discuss the effect of trust repair on OT beliefs concerning different trust history configurations.

I present the results of the experimental evaluation for all effects together in plots in Figures 21, 22, and 23 for a malicious, faulty, and benign node, respectively. The following sections present plots for those effects separately and discuss implications. From Figures 21, 22, and 23, we can observe that the effects of RAT and VT on OT depend on the node type (malicious, faulty, and benign), node behavior (opportunistic service attack vs. single bad behavior), and trust history configuration (using whole history vs. sliding window of trust scores).

First, different lines (OT beliefs for different VT and RAT combinations) may overlap for a malicious and benign node whereas they do not for a faulty node. For instance, for a malicious node, the line for *hardware-based repair + goodwill-type violation* combination overlaps with that for *software-based repair + competence-type violation.* Also, trust repair seems to affect OT towards a faulty node slower than towards a malicious node. Yet, beliefs towards a faulty node continue to improve over time, whereas for a malicious node, they stay more stable once they increase. Finally, as seen in Figure 23, all different combinations of RAT and VT lead to overlapping trust scores. Thus, regardless of the type of response action and trust violation, trust will recover to the fullest extent possible for a benign node. These results reflect the behaviors modeled differently for malicious, faulty, and benign nodes. As a malicious node behaves benign and accumulate trust until some point in the simulations, a trust repair action improves its trust scores faster than a faulty node, which also reports incorrect measurements. Although the measurements reported by a faulty node deviate from benign measurements less than those by a malicious node, it consistently reports incorrect measures. That is the reason why its trust scores improve slower than a malicious node. Yet, a faulty node has its scores recover gradually after a trust repair action, whereas a malicious node has its trust scores stable in the short term and decreasing in the long term. If we modeled different attack types as in Chapters 5 and 6, we could observe different trust repair effects. For instance, for an individual malicious node, trust repair could work slower than a faulty node.

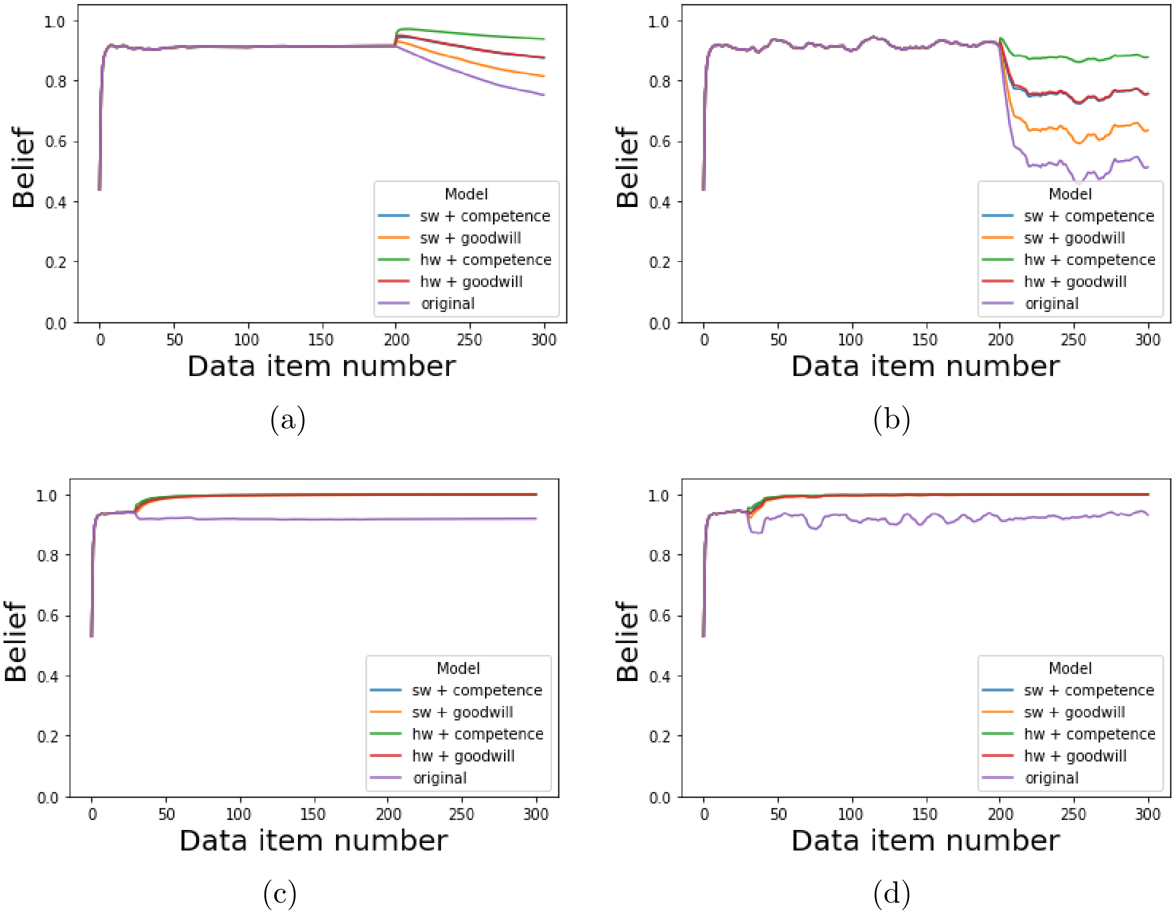
Second, for opportunistic service and single bad behavior, trust repair affects scores differently. If we compare the plots at the top of Figures 21 and 22 to the ones at the bottom, we observe that the type of trust repair action and trust violation does not affect trust scores for a single bad behavior as opposed to opportunistic service attack scenario. This may tell that in case of a single random failure, independent from whether the failure is a result of competence or goodwill of the node, it does not matter whether we take a software-based or hardware-based repair action. As a result, the less costly or more convenient option could be chosen by the network administrator.

Third, when we compare the whole trust history to the sliding window approach to trust computations, we observe two noticeable patterns. One observation is that OT belief seems to decrease for the whole history configuration for the opportunistic service attack scenario (Figures 21a and 22a. On the contrary, it seems not to decrease (stable after attack and repair start point in Figure 21b and increasing after attack and repair start point in Figure 22b) for the sliding window configuration for the opportunistic scenario. This means we need to consider the long-term effect of trust repair to make decisions. Trust may seem to improve in the short-term, i.e., when we consider only a recent window of trust records in historical trust computations, but it may not improved to a desired level when all the previous trust history is included in trust computations. The second observation concerning trust history configuration is that for a single bad behavior, there is no noticeable difference in OT beliefs generated by the trust model concerning different RAT and VT combinations between the whole history and sliding window approaches.

7.5.3.1 Effect of Trust Repair Action Type on Overall Trust

Figure 24 shows the effect of trust repair action type (RAT) on OT beliefs for a malicious node. Looking at the top row for opportunistic service attack, we see that hardware-based trust repair yields higher trust scores than software-based trust repair for both trust violation types, competence and goodwill. This holds for the two different trust history configurations, whole history and sliding window approach. When we look at the bottom row for the single bad behavior scenario, we do not observe a difference between software- and hardware-based trust repair for none of trust violation types and trust history configurations. As a result, we can draw the conclusion that the type of trust repair action affects the outcome of trust repair when a malicious node displays bad behaviors repeatedly.

Figure 25 shows the effect of trust repair action type (RAT) on OT beliefs for a faulty node. Similar to a malicious node, hardware-based trust repair yields higher trust scores than software-based trust repair for opportunistic service attack, for both competence- and goodwill-type trust violation, and for both trust history configurations; whole history and sliding window. Different than a malicious node, for the single bad behavior scenario, hardware-based repair yields higher trust scores than software-based repair in all combinations of trust violation type and trust history configurations (Figures 25e, 25f, 25g, and



(a)

(b)

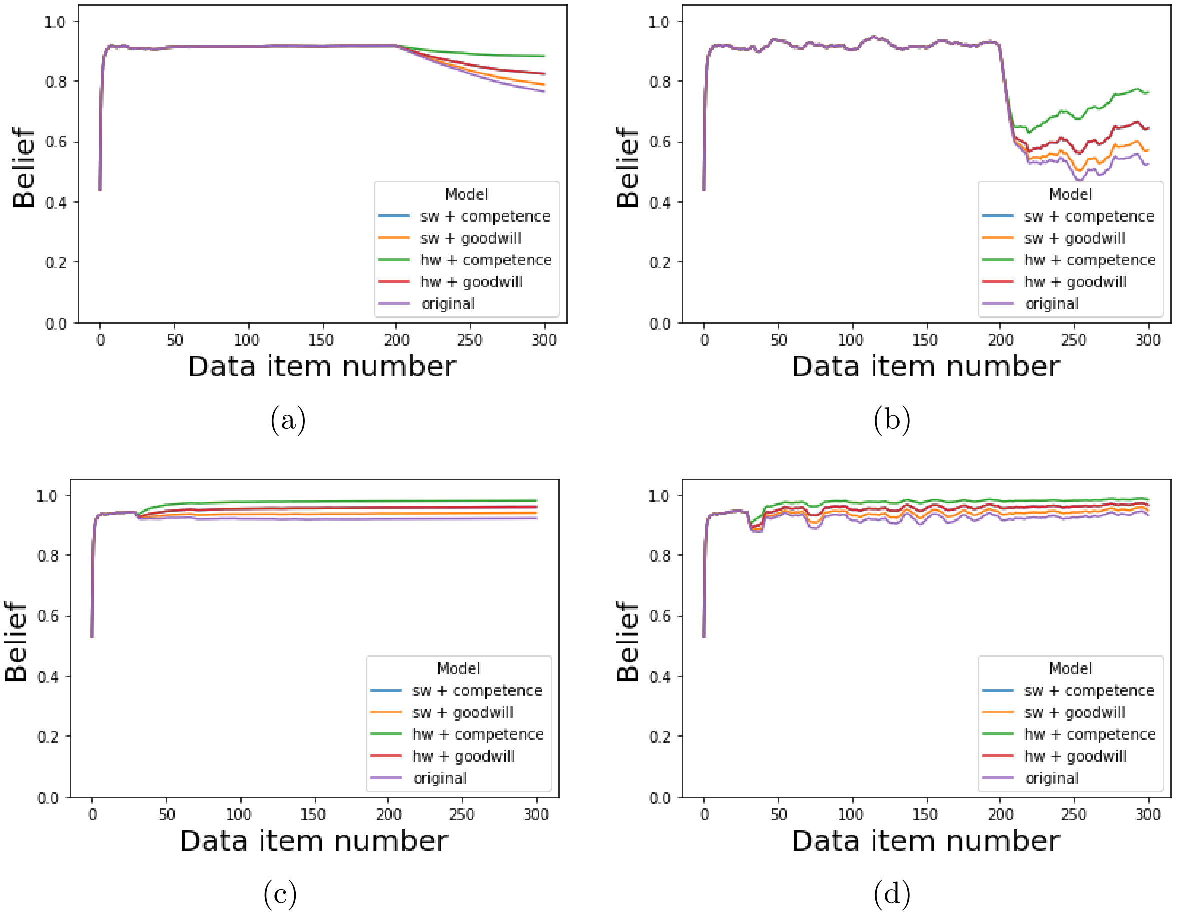
(c)

(d)

Figure 21: Effect of trust repair for a malicious node: (a) opportunistic service + whole history (b) opportunistic service + sliding window (c) single bad behavior + whole history (d) single bad behavior + sliding window

25h).

Finally, as Figure 26 shows, the type of trust repair action does not have an effect on trust scores of a benign node for neither of the opportunistic service attack and single bad behavior cases. Note that it is not the benign node itself displaying bad behavior and committing the trust violation, but malicious nodes in the network. Thus, the *OT* belief of a benign node drops even though it behaves benignly. Yet, as the figure shows, its trust recovers to the maximum possible value regardless of the type of trust repair action and trust violation, and trust history configuration.



(a)

(b)

(c)

(d)

Figure 22: Effect of trust repair for a faulty node: (a) opportunistic service + whole history (b) opportunistic service + sliding window (c) single bad behavior + whole history (d) single bad behavior + sliding window

7.5.3.2 Effect of Trust Violation Type on Overall Trust

Figures 27,28, and 29 display the effect of VT (trust violation type) on OT belief for a malicious, faulty, and benign node, respectively. Patterns of OT belief concerning the effect of trust violation type are very similar to those concerning the effect of trust repair action type. More precisely, for a malicious node and opportunistic service attack case (top row at Figure 27), goodwill-type trust violation leads to a less improvement in trust scores when a trust repair action is taken, regardless of the trust history configuration. On the contrary, trust violation type does not affect trust repair outcome when malicious nodes commit to a trust violation for a single time (bottom row at Figure 27). Similarly, a faulty

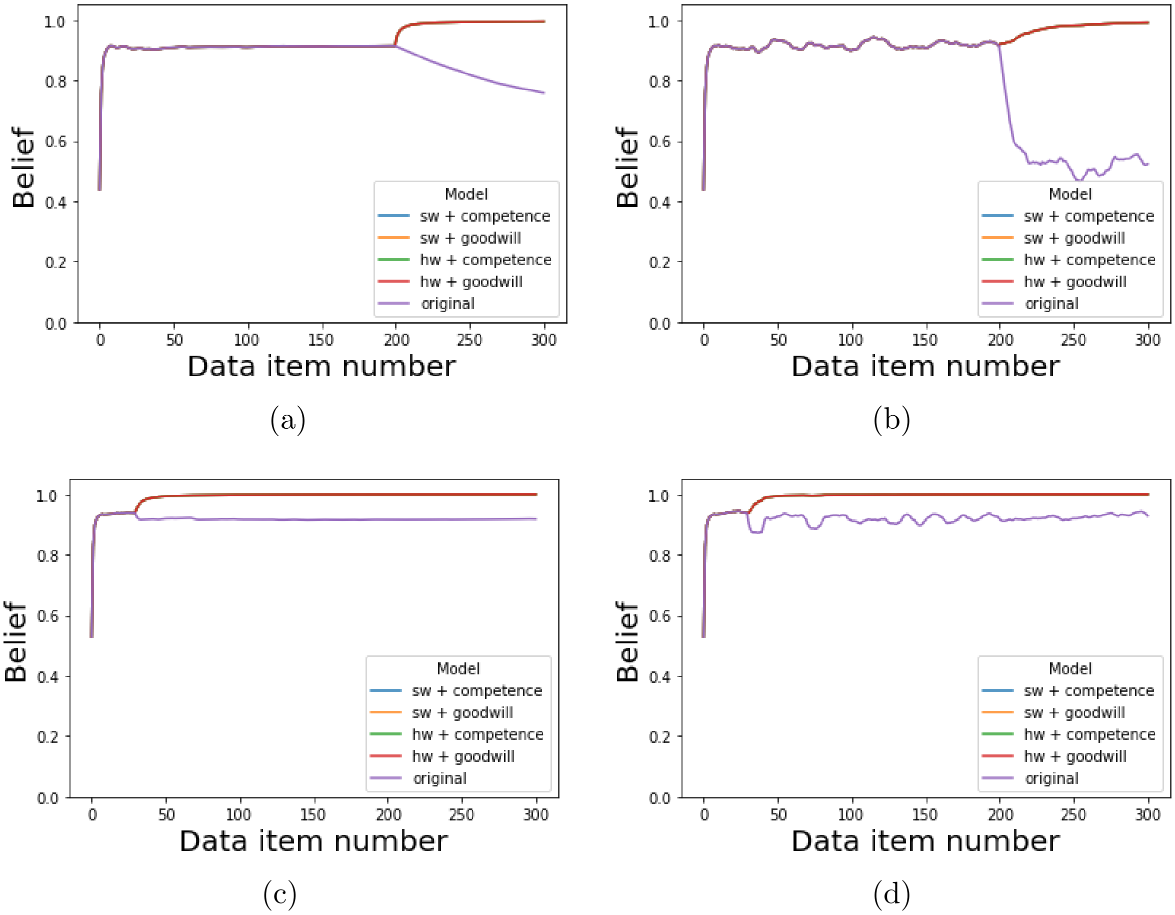
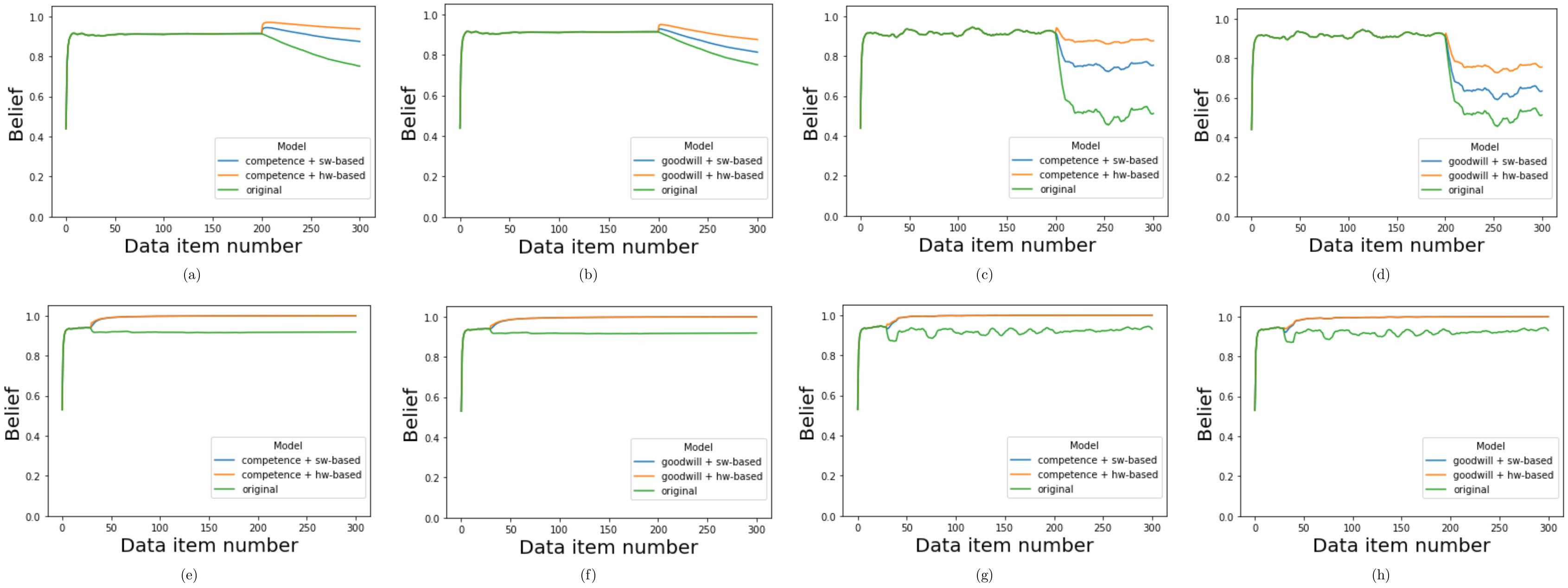


Figure 23: Effect of trust repair for a benign node: (a) opportunistic service + whole history (b) opportunistic service + sliding window (c) single bad behavior + whole history (d) single bad behavior + sliding window

node has higher trust scores as a result of trust repair in case of a competence-based trust violation, regardless of trust repair action type and trust history configuration, for both the opportunistic service attack case (top row at Figure 28) and single bad behavior case (bottom row at Figure 28). Finally, for a benign node, VT does not have an effect on trust repair for none of the combinations of node behavior, RAT, and trust history configuration, as seen in Figure 29.



(a)

(b)

(c)

(d)

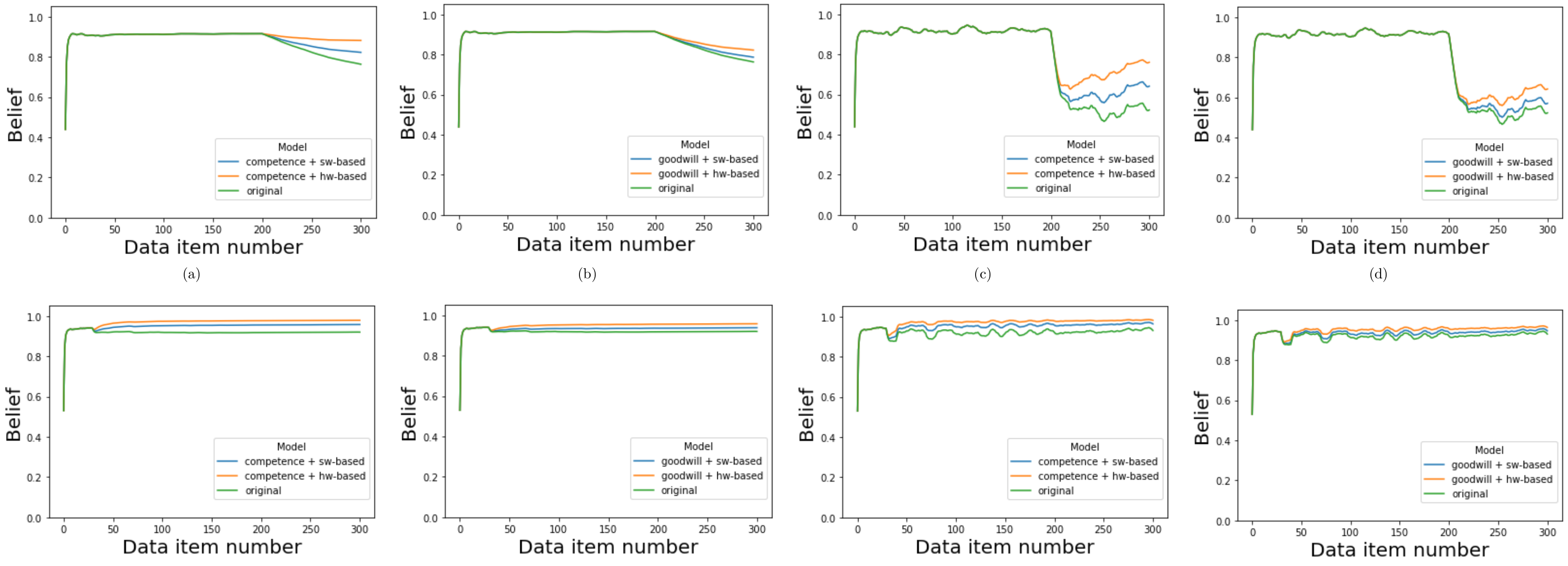
(e)

(f)

(g)

(h)

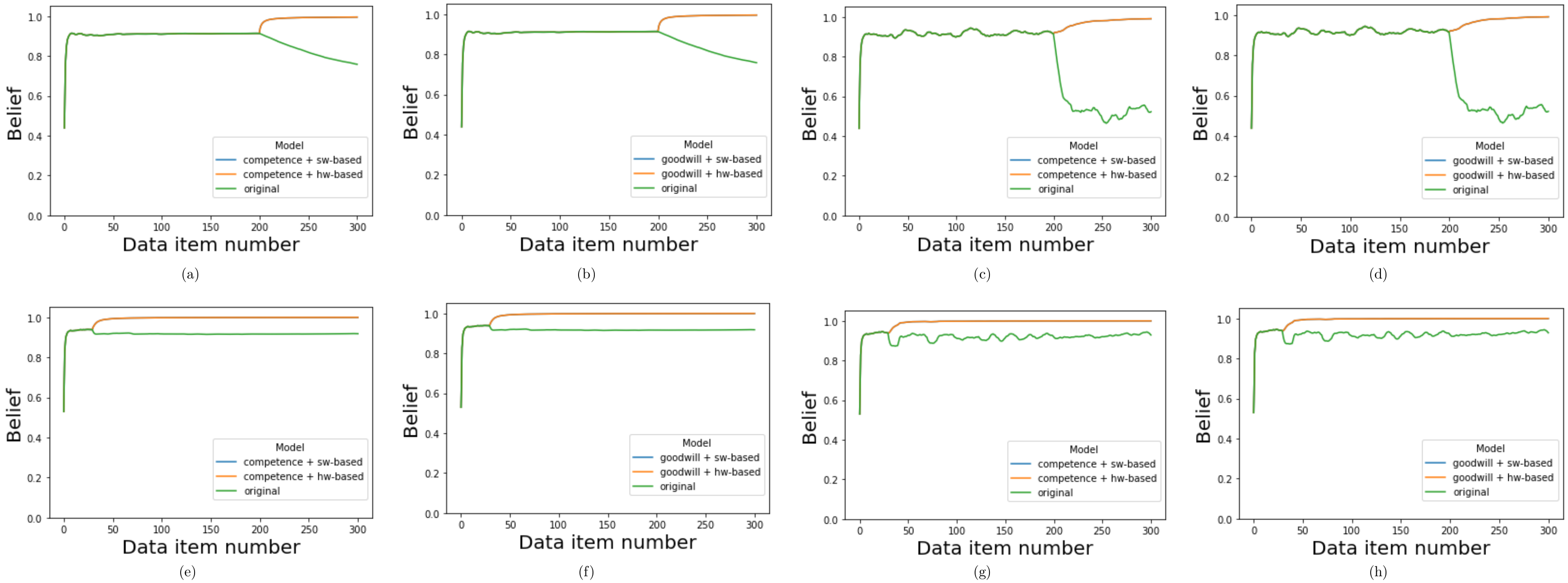
Figure 24: Effect of *RAT* on *OT* for a **malicious node:** (a) opportunistic service + competence + whole history (b) opportunistic service + goodwill + whole history (c) opportunistic service + competence + sliding window (d) opportunistic service + goodwill + sliding window (e) single bad behavior + competence + whole history (f) single bad behavior + goodwill + whole history (g) single bad behavior + competence + sliding window (h) single bad behavior + goodwill + sliding window



(a) (b) (c) (d)

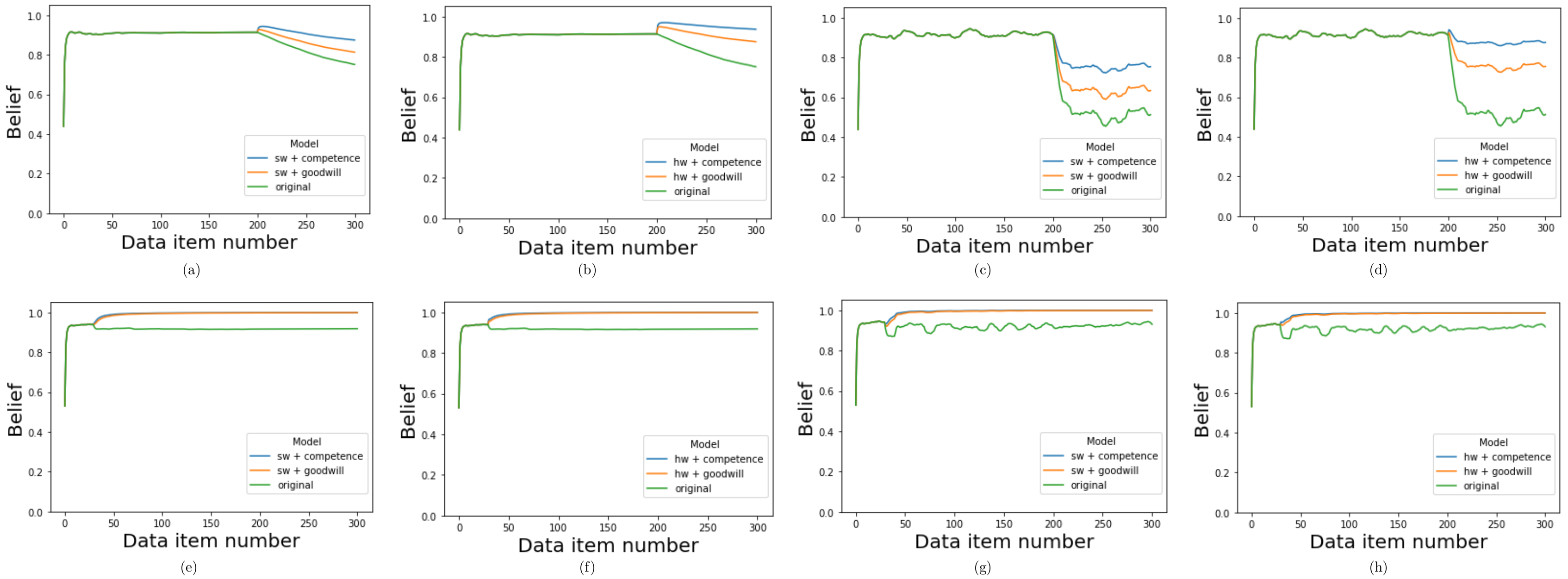
(e) (f) (g) (h)

Figure 25: Effect of *RAT* on *OT* for a **faulty node:** (a) opportunistic service + competence + whole history (b) opportunistic service + goodwill + whole history (c) opportunistic service + competence + sliding window (d) opportunistic service + goodwill + sliding window (e) single bad behavior + competence + whole history (f) single bad behavior + goodwill + whole history (g) single bad behavior + competence + sliding window (h) single bad behavior + goodwill + sliding window



(a) (b) (c) (d) (e) (f) (g) (h)

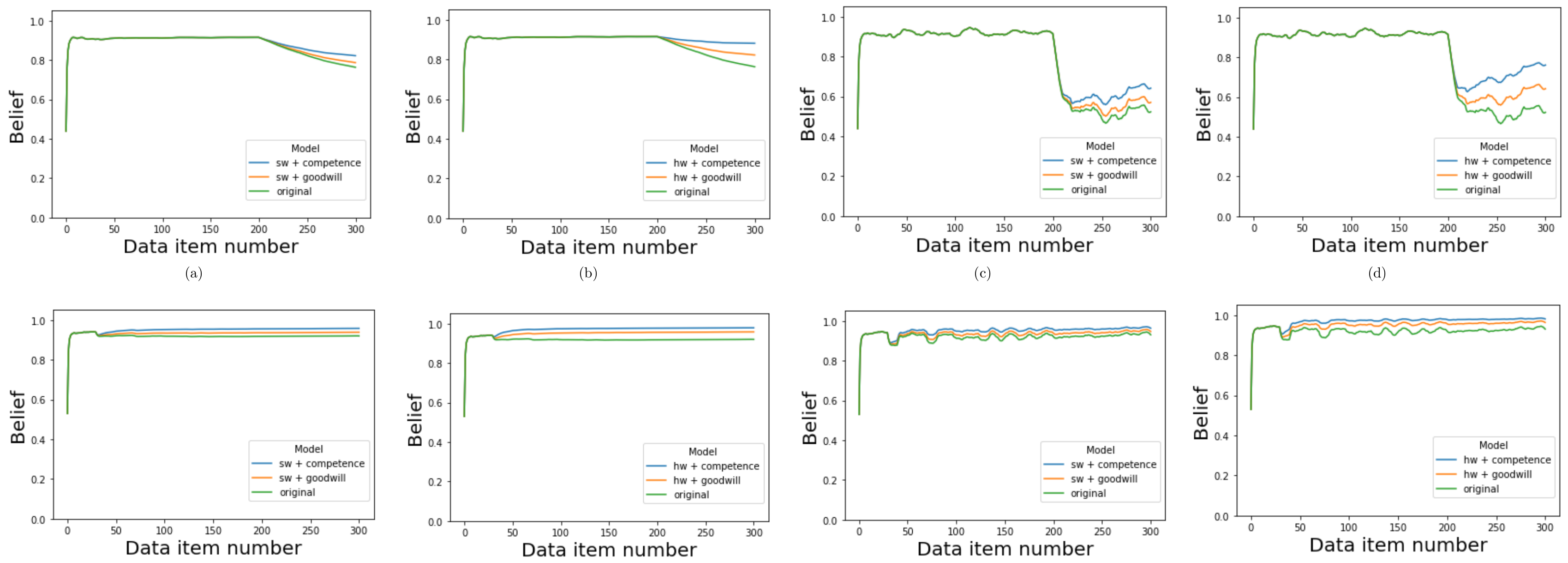
Figure 26: Effect of *RAT* on *OT* for a **benign node:** (a) opportunistic service + competence + whole history (b) opportunistic service + goodwill + whole history (c) opportunistic service + competence + sliding window (d) opportunistic service + goodwill + sliding window (e) single bad behavior + competence + whole history (f) single bad behavior + goodwill + whole history (g) single bad behavior + competence + sliding window (h) single bad behavior + goodwill + sliding window



(a) (b) (c) (d)

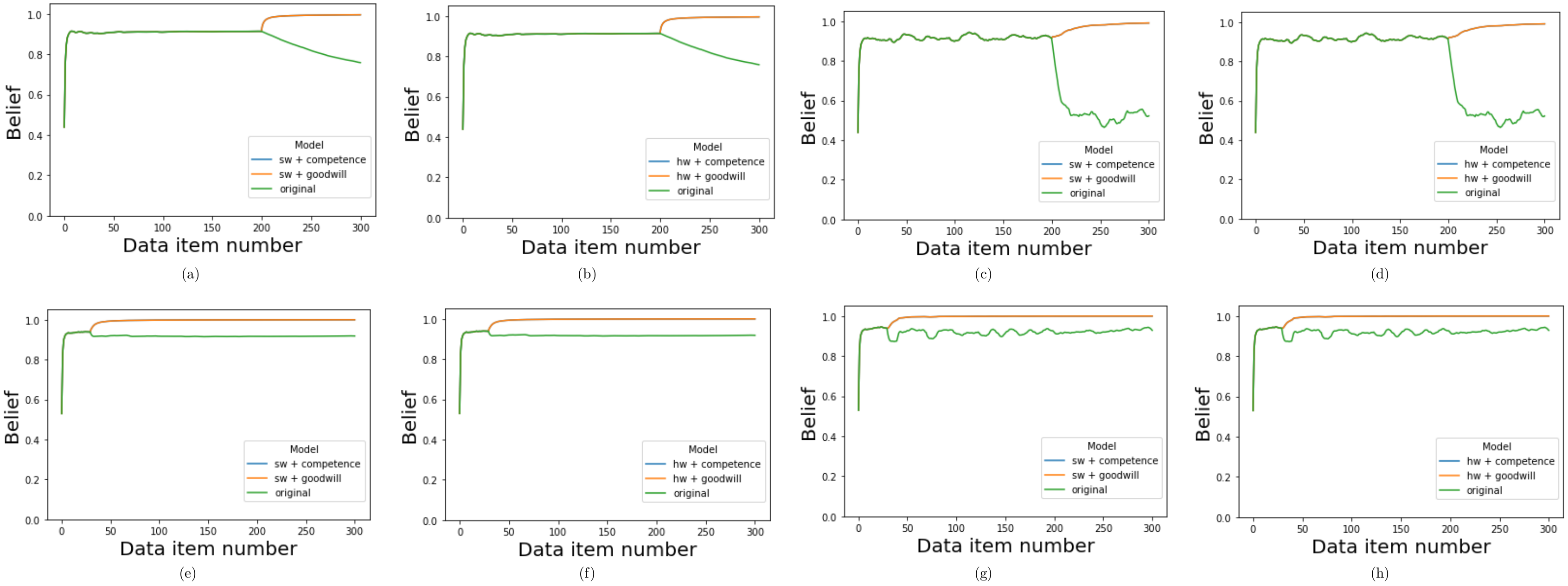
(e) (f) (g) (h)

Figure 27: Effect of VT on OT for a **malicious node:** (a) opportunistic service + sw-based repair + whole history (b) opportunistic service + hw-based repair + whole history (c) opportunistic service + sw-based repair + sliding window (d) opportunistic service + hw-based repair + sliding window (e) single bad behavior + sw-based repair + whole history (f) single bad behavior + hw-based repair + whole history (g) single bad behavior + sw-based repair + sliding window (h) single bad behavior + hw-based repair + sliding window



(a) (b) (c) (d) (e) (f) (g) (h)

Figure 28: Effect of VT on OT for a **faulty node:** (a) opportunistic service + sw-based repair + whole history (b) opportunistic service + hw-based repair + whole history (c) opportunistic service + sw-based repair + sliding window (d) opportunistic service + hw-based repair + sliding window (e) single bad behavior + sw-based repair + whole history (f) single bad behavior + hw-based repair + whole history (g) single bad behavior + sw-based repair + sliding window (h) single bad behavior + hw-based repair + sliding window



(a) (b) (c) (d) (e) (f) (g) (h)

Figure 29: Effect of VT on OT for a **benign node:** (a) opportunistic service + sw-based repair + whole history (b) opportunistic service + hw-based repair + whole history (c) opportunistic service + sw-based repair + sliding window (d) opportunistic service + hw-based repair + sliding window (e) single bad behavior + sw-based repair + whole history (f) single bad behavior + hw-based repair + whole history (g) single bad behavior + sw-based repair + sliding window (h) single bad behavior + hw-based repair + sliding window

7.6 Summary and Discussion

In this chapter, I discuss trust repair for IoT and propose a trust repair model inspired by trust repair concepts in social sciences. The proposed model includes trust violation type (VT) and trust repair action type (RAT) as factors impacting the outcome of the trust repair process. I conducted experiments to shed light on the effect of these factors on trust repair. I compared OT beliefs computed by the original MADM-EBSL framework to those by the proposed trust repair model for different combinations of VT, RAT, trust history configuration (whole history vs. sliding window), and node behavior (behaviors of malicious nodes in the network, specifically opportunistic service and single bad behavior) for malicious, faulty, and benign nodes.

There are multiple indications of the results of the experimental evaluation. Some of them worth highlighting includes that VT and RAT affect trust repair differently concerning the type of node that the trust of which is to be repaired (malicious, faulty, and benign), simulated behavior of malicious nodes in the network (opportunistic service attack and single bad behavior), and trust history configuration. In addition, a hardware-based trust repair action yields higher trust scores than a software-based trust repair action in opportunistic service attack case, in general. Moreover, trust repair is more effective when VT is competence compared to goodwill-type violation for most of the cases. Also, for single bad behavior case, VT and RAT do not affect trust repair results for malicious and benign nodes. Yet another important observation was that trust scores of a benign node is recovered to its fullest extent regardless of VT, RAT, malicious nodes' behavior, and trust history configuration.

Noteworthy is emphasizing the cases where a hardware-based repair may be needed given that it is rather costly to intervene with a human. Referring to the results in Section 7.5.3.1 and Figures 24, 25, and 26, we notice that one factor is the type of node we want to repair trust of. For example, if it is a benign node, RAT does not have an effect on trust scores (see Figure 26), so a software-based repair action could be chosen as the cheaper option. Another factor is the type of violation that a node commits. For example, if we want to recover the trust of a faulty node, we may want to consider a hardware-based repair when VT is decided as competence by the central trust manager. As seen in Figures 25c and

25d, a hardware-based repair provides a more pronounced improvement in trust scores than a software-based repair for a competence-type violation for an opportunistic service attack. On the other hand, for a single bad behavior case, the difference in improvement provided by the two types of repair is not significant. Therefore again, software-based repair may be preferred. Finally, the type of the IoT application or the node could be a factor for making decisions. If it is a critical application, such as a medical IoT network, then even slight differences make a difference. In that case, hardware-based repair could be an option even it is much more costly.

8.0 Conclusion and Future Research

8.1 Conclusion

Trust management is vital in IoT, as trust mechanisms play a vital role in mitigating uncertainty and perceived risks associated with the usage of IoT services and applications. The absence of trust can result in severe consequences for the IoT ecosystem. For instance, a malicious node could potentially disrupt the smooth operation of an IoT network, thereby elevating the reputation of malicious nodes and degrading that of benign ones [1]. Additionally, a lack of trust could lead to disruptions in the entire data management process, thereby impeding the proper functioning of IoT devices [4].

Cryptographic algorithms used for safeguarding IoT systems against external attacks do not provide protection against insider attacks [154], as insider nodes are already part of the network and have secure communication established through exchanged cryptographic keys [11]. Also, strong security measures cannot be implemented on constrained IoT nodes [1]. As a result, trust management solutions are critical for detecting legitimate nodes that exhibit malicious behavior within IoT environments [9, 154, 188]. Although trust management solutions have been proposed to address these challenges, they have limitations. For example, trust management schemes based on the Social IoT paradigm, feedback, or service recommendation could require human intervention [12] and may not be suitable for automated trust computation.

In this dissertation, I have reviewed existing research that proposes trust management solutions for IoT and discussed their limitations. The main focus is key trust properties as we have distilled from our review of trust in social sciences and automated trust computation. More precisely, I highlight multi-dimensionality, context-specificity, dynamism, uncertainty, and transitivity properties of trust with automation capability.

The central pieces of the automated trust computation solutions I have proposed are Multi-Attribute Decision Making (MADM) and Evidence-Based Subjective Logic (EBSL). The key contributions of this dissertation are as follows:

- I have presented the work [3] that reviews trust in social sciences in Chapter 3. We have overviewed trust measures used in the social science literature. We have proposed a trust taxonomy based on our review of trust types identified by social science researchers. Moreover, we have discussed several aspects of trust and challenges in assuring it, including trust development over time, trust repair, trust transfer and reputation, the relationship of trust with context, distrust, risk, and uncertainty. Finally, we have discussed the implications for IoT networks and presented a conceptual trust management framework for IoT.
- We have developed an automated trust computation framework, MADM-EBSL (Multi-Attribute Decision Making - Evidence-Based Subjective Logic Framework) for IoT, in Chapter 4. A significant contribution of the MADM-EBSL is that it addresses trust management challenges in IoT, including the need for automatic trust computations without human intervention, the complexity of trust measurement due to various contextual/environmental and technical/protocol-dependent factors, and the difficulty of quantifying the trustworthiness of nodes. The trust scores can account for multiple contextual and technical attributes affecting both functional and referral trust of an IoT node using the MADM approach. We adopted the Evidence-Based Subjective Logic (EBSL) [167] to account for uncertainty in trust values and trust transitivity in a trust network of "things". We implemented the MADM-EBSL framework and evaluated the performance of it using synthetic data and sampling from real datasets. The results show that the trust framework can effectively capture node behaviors and limitations and can identify faulty and compromised nodes. We also observed that the trust of nodes are impacted by their neighbors in the network.
- I have explored the impact of network connectivity and trust problem size on trust scores in the MADM-EBSL framework in Chapter 5. The results indicate that higher network connectivity levels lead to a more significant reduction in trust scores for a malicious node in larger trust problem sizes. Larger trust problem sizes correspond to higher percentages of malicious nodes in a network, making attacks more challenging to detect, such as on-off attacks. Furthermore, the results suggest that a larger trust problem size results in higher trust scores for a malicious node.

- Based on the concepts distilled from our review of trust in social sciences, I include additional functional trust attributes in the MADM-EBSL in Chapter 6. These attributes are multiple device vendors, the social presence of devices, and the replication success and source of a reported measurement. I have investigated the effect of extra trust attributes on trust scores through experimental evaluations. The results showed that including additional functional trust attributes did not significantly affect the overall trust beliefs towards a selected node but did affect the HFT belief scores.
- I have described a trust repair model for IoT in Chapter 7, inspired by trust repair concepts in social sciences. The model takes into account trust violation type (VT, competence vs. goodwill) and trust repair action type (RAT, hardware-based vs. software-based) as factors that impact the outcome of the trust repair process. I have investigated the effect of these factors on trust repair through experiments. To do so, I compared the OT beliefs of malicious, faulty, and benign nodes after repair to those computed by the original MADM-EBSL framework. The results indicated that trust repair is impacted differently by VT, RAT, and node type (malicious, faulty, or benign). In general, hardware-based repair yields higher trust scores than software-based repair in opportunistic service attacks. Trust repair is more effective for competence-type violations. For single bad behavior cases, VT and RAT do not affect trust repair for malicious and benign nodes. The results also showed that trust scores for a benign node fully recovered regardless of VT and RAT.

8.2 Future Research

Below are some possible future research directions for the research in this dissertation:

- We have explored the Subjective Logic [77] to account for uncertainty in trust opinions and evidence from multiple nodes in an IoT network. Alternatively, we could consider a Bayesian approach to estimate trust scores while accounting for uncertainty. Bayesian approaches are based on Bayes' theorem. They rely on assigning prior probabilities to model initial beliefs and updating these beliefs using observed data through the likelihood function. Both approaches have their strengths and limitations, so it is not easy to provide a preference for one over another. Yet, it may be worth exploring a Bayesian approach to IoT trust computations and comparing the results to our framework that is based on Subjective Logic.

- Potential future work as the continuation of attempts in Chapter 5 includes additional experimental evaluations that consider malicious nodes performing different attack types simultaneously. Simulations in Chapter 5 model a network having all malicious nodes perform the same attack. Although I looked at the effect of various attack models on trust scores, I investigated them separately. A mix of different malicious behaviors in a network may affect trust scores differently than homogeneous behaviors.
- Experiments in all chapters used the same methodology to generate a random network with no specific topology, distributing nodes in a rectangular area using the Poisson point process. Also worth exploring is the effect of network topology on trust scores, which may include trees, complete networks, a hierarchical structure, federated systems, etc.
- Another future research direction is related to the evaluation of the MADM-EBSL framework, the framework proposed in Chapter 6, the trust repair model in Chapter 7, and any trust computation framework/model in general. Although we successfully devised frameworks to quantify the trust of IoT nodes considering various contextual and technical factors and the amount of trust to repair after a trust violation, we have not found a trust threshold as a baseline value to compare our trust scores and compute the model accuracy. We have not come across a research work in the IoT trust domain that proposes a solid methodology to calculate a trust threshold value to base the output of trust computations. Despite some researchers using trust thresholds in their experimental evaluations, such as in [25, 1, 145, 23, 20], there is no foundation for their preference for trust thresholds. The lack of a solid methodology for setting a trust threshold value is a challenge to be solved by future work.
- One potential future research direction for the research in Chapter 6 could be to investigate the effectiveness of the extended MADM-EBSL framework for detecting more complex attacks, such as coordinated attacks or attacks that combine multiple attacks.
- As the continuation of the research in Chapter 7, it may be worthwhile to investigate

the use of machine learning techniques to predict the most effective trust repair action for a given trust violation type and node behavior. Also, investigating additional factors affecting trust repair could be another future research direction. The proposed model considers trust violation type and response action type as two key factors impacting the outcome of the trust repair process. However, other factors can also play a significant role in trust repair, such as the severity of the trust violation.

Appendix A Frequently Cited Trust Measures in Social Sciences

Table 12: Trust measures cited frequently in social science disciplines

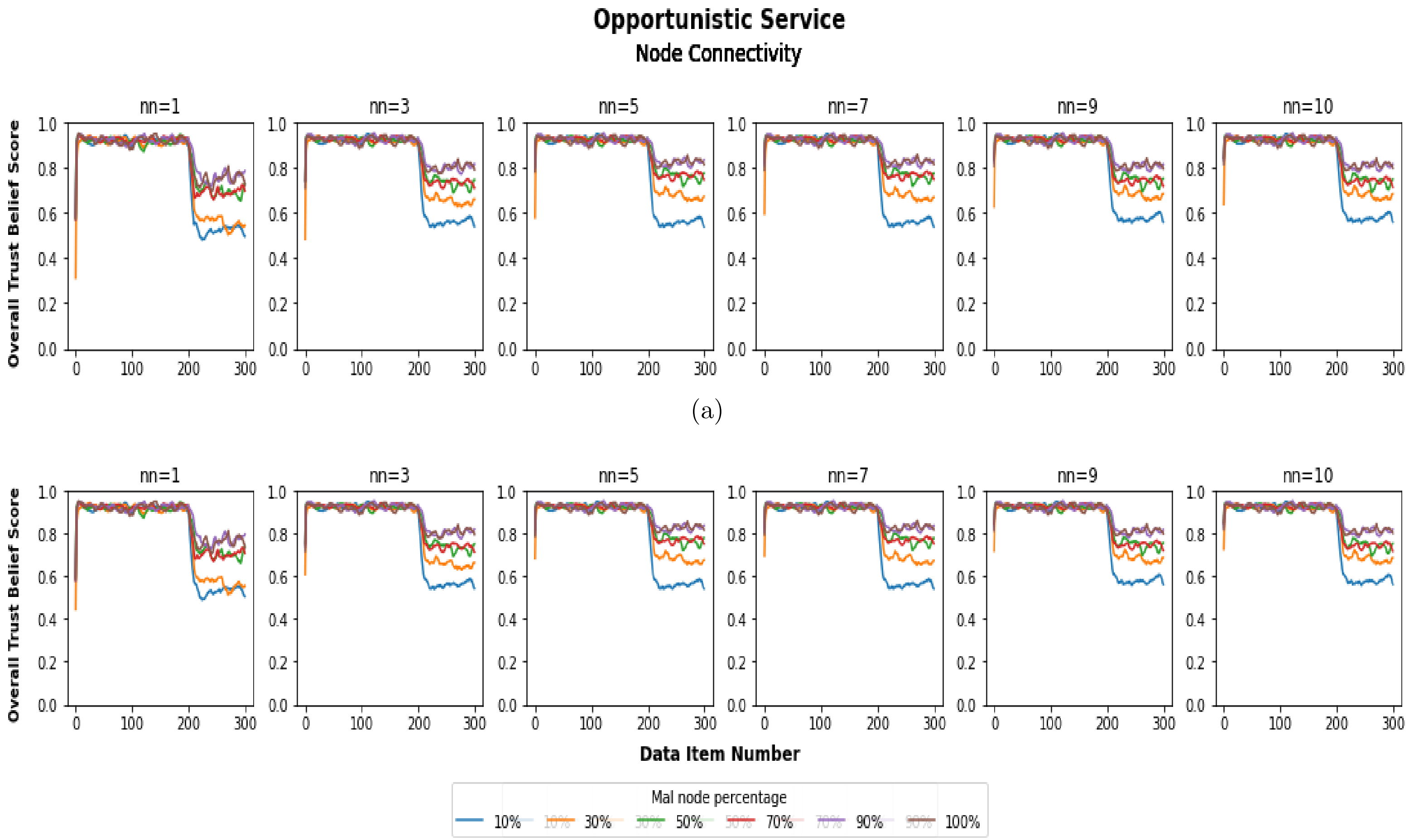
Construct	Variable	Study Using Items	Relevant Survey Items
Trust		Mayer and Davis (1999)[1]	If I had my way, I wouldn't let top management have any influence over issues that are important to me. I would be willing to let top management have complete control over my future in this company. I really wish I had a good way to keep an eye on top management. I would be comfortable giving top management a task or problem which was critical to me, even if I could not monitor their actions.
		Kim and Peterson (2013)	This s-commerce firm is trustworthy. I trust that this s-commerce firm keeps my best interests in mind. This s-commerce firm will keep its promises. I believe in the information that this s-commerce firm provides. This s-commerce firm wants to be known as a company that keeps its promises and commitments.
		Gefen et al. [53]	I feel that this online vendor is honest. I feel that this online vendor is trustworthy. I feel that this online vendor cares about customers. I feel that this online vendor would provide me with good service.

Trustworthiness (Mayer and Davis [117]), Trusting beliefs (McKnight et al. (2002)) Trust (Gefen and Straub [54])	Ability (Mayer and Davis [117], Gefen and Straub [54]),Competence (McKnight et al. (2002))	Mayer and Davis (1999)	Top management is very capable of performing its job. Top management is known to be successful at the things it tries to do. Top management has much knowledge about the work that needs done. I feel very confident about top management's skills. Top management has specialized capabilities that can increase our performance. Top management is well qualified.
		McKnight et al. (2002)	1. The trustee is competent and effective in providing legal advice. 2. The trustee performs its role of giving legal advice very well. 3. Overall, the trustee is a capable and proficient Internet legal advice provider. 4. In general, the trustee is very knowledgeable about the law.
		Gefen and Straub [54]	1. The trustee is competent. 2. The trustee knows about books. 3. The trustee knows how to provide excellent service.
	Benevolence	Mayer and Davis [117]	Top management is very concerned about my welfare. My needs and desires are very important to top management. Top management would not knowingly do anything to hurt me. Top management really looks out for what is important to me. Top management will go out of its way to help me.
		McKnight et al. [122]	1. I believe that the trustee would act in my best interest. 2. If I required help, the trustee would do its best to help me. 3. The trustee is interested in my well-being, not just its own.

		Gefen and Straub [54]	1. I expect I can count on the trustee to consider how its actions affect me. 2. I expect that trustee's intentions are benevolent. 3. I expect that the trustee puts trustors' interests before their own. 4. I expect that the trustee is well meaning.
	Integrity	Mayer and Davis [117]	Top management has a strong sense of justice. I never have to wonder whether top management will stick to its word. Top management tries hard to be fair in dealings with others. Top management's actions and behaviors are not very consistent. I like top management's values. Sound principles seem to guide top management's behavior.
		McKnight et al. [122]	1. The trustee is truthful in its dealings with me. 2. I would characterize the trustee as honest. 3. The trustee would keep its commitments. 4. The trustee is sincere and genuine.
		Gefen and Straub [54]	1. Promises made by the trustee are likely to be reliable. 2. I do not doubt the honesty of the trustee. 3. I expect that the trustee will keep promises they make.
Trust propensity (Mayer and Davis [117]) Disposition to trust (McKnight et al. [122]) Trust disposition (Gefen and Straub [54] Propensity to trust (Pavlou and Gefen [140]))	Trust propensity	Mayer and Davis (1999)	One should be very cautious with strangers. Most experts tell the truth about the limits of their knowledge. Most people can be counted on to do what they say they will do. These days, you must be alert or someone is likely to take advantage of you. Most sales people are honest in describing their products. Most repair people will not overcharge people who are ignorant of their specialty. Most people answer public opinion polls honestly. Most adults are competent at their jobs.

	Trusting stance	McKnight et al. (2002)	1. I usually trust people until they give me a reason not to trust them. 2. I generally give people the benefit of the doubt when I first meet them. 3. My typical approach is to trust new acquaintances until they prove I should not trust them.
	Trusting disposition	Gefen and Straub [54]	1. I generally trust other people. 2. I tend to count upon other people. 3. I generally have faith in humanity. 4. I feel that people are generally well meaning. 5. I feel that people are generally trustworthy. 6. I feel that people are generally reliable.
	Propensity to trust	Pavlou and Gefen [140]	1. Most people are reliable. 2. Most people keep promises and commitments. 3. Most people are honest.

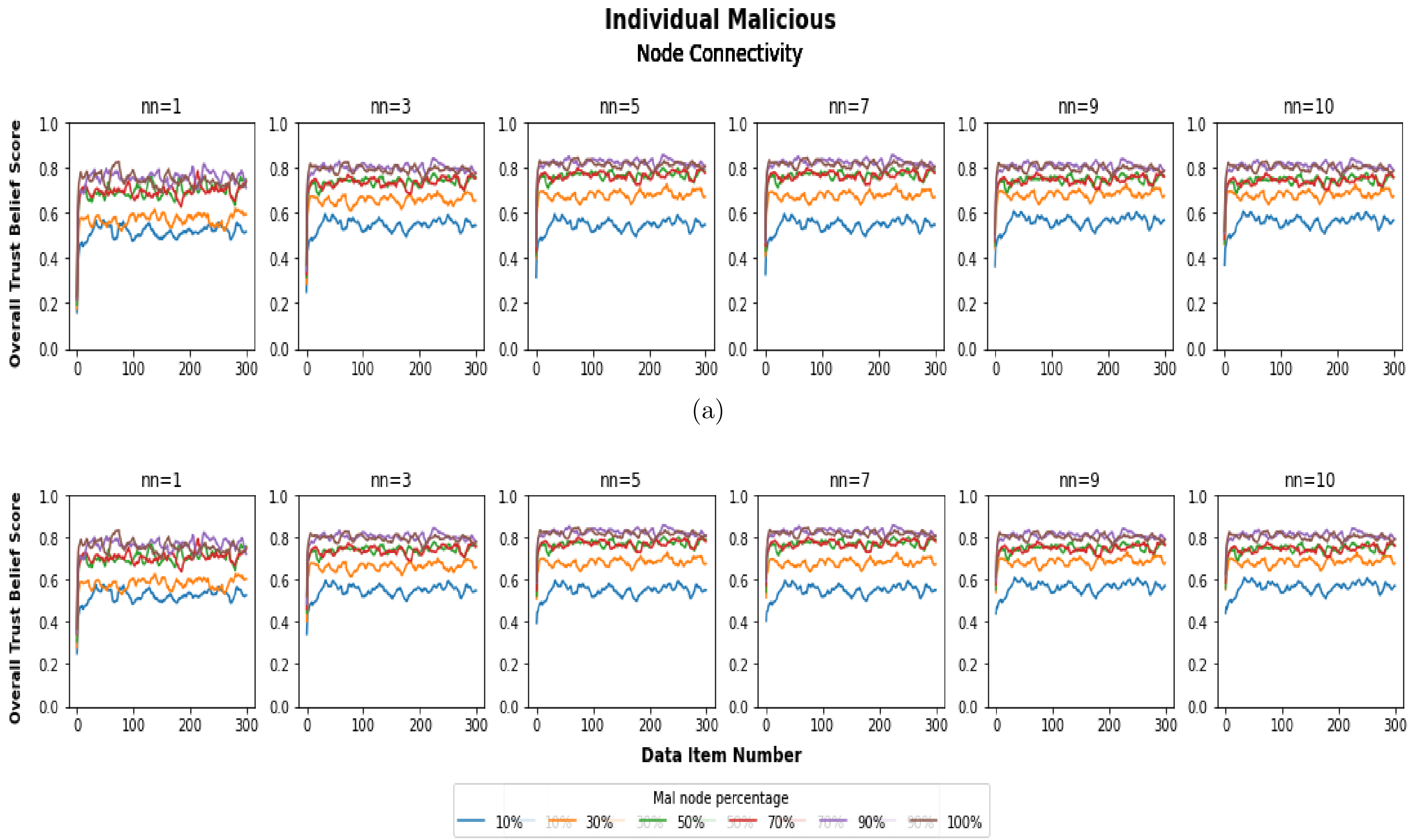
Appendix B Overall Trust Comparison: Original vs. Extended Framework



(a)

(b)

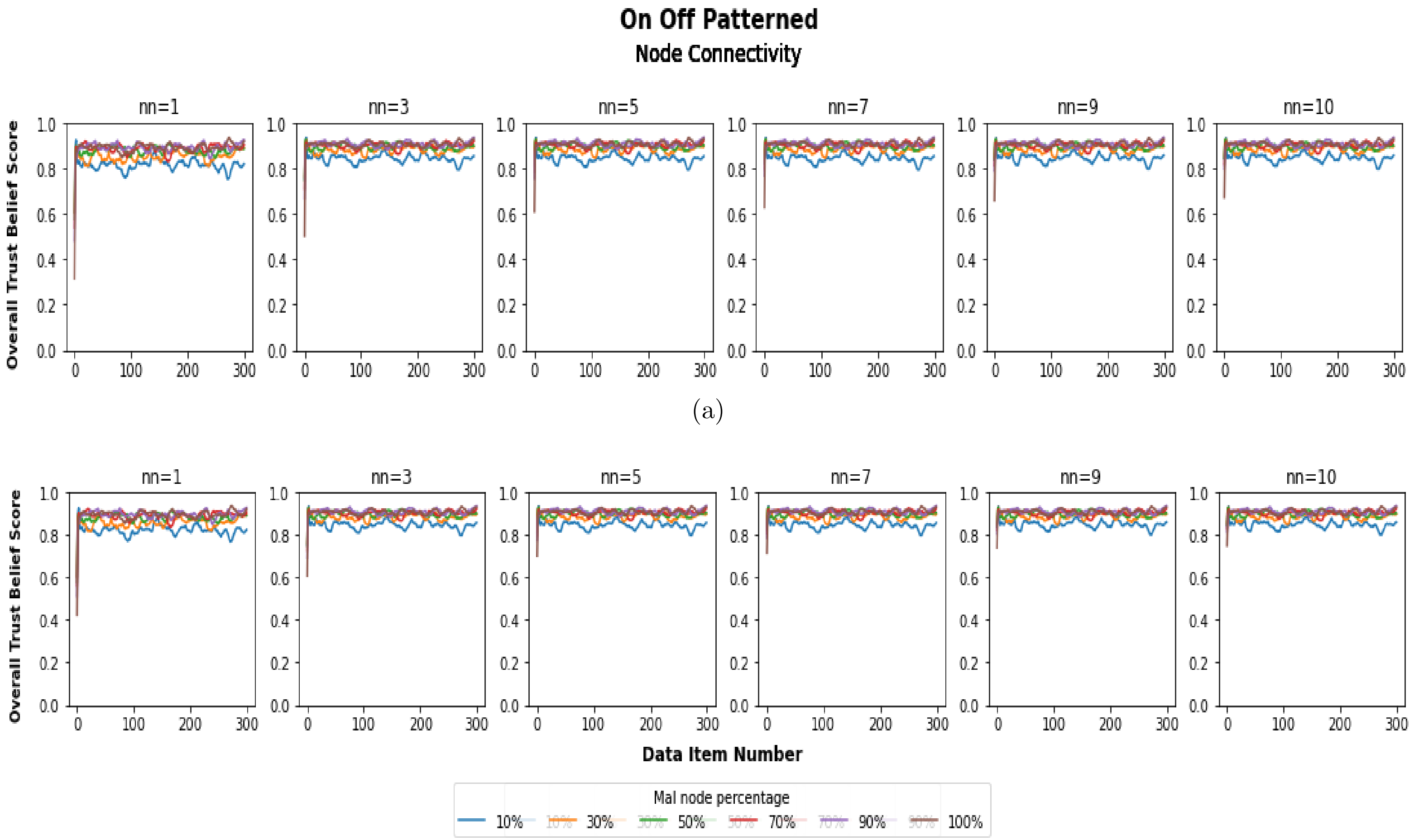
Figure 30: Overall trust belief scores for a malicious node performing opportunistic service attack concerning varying *nn* and *mnp* levels computed by: (*a*) base (*b*) extended



(a)

(b)

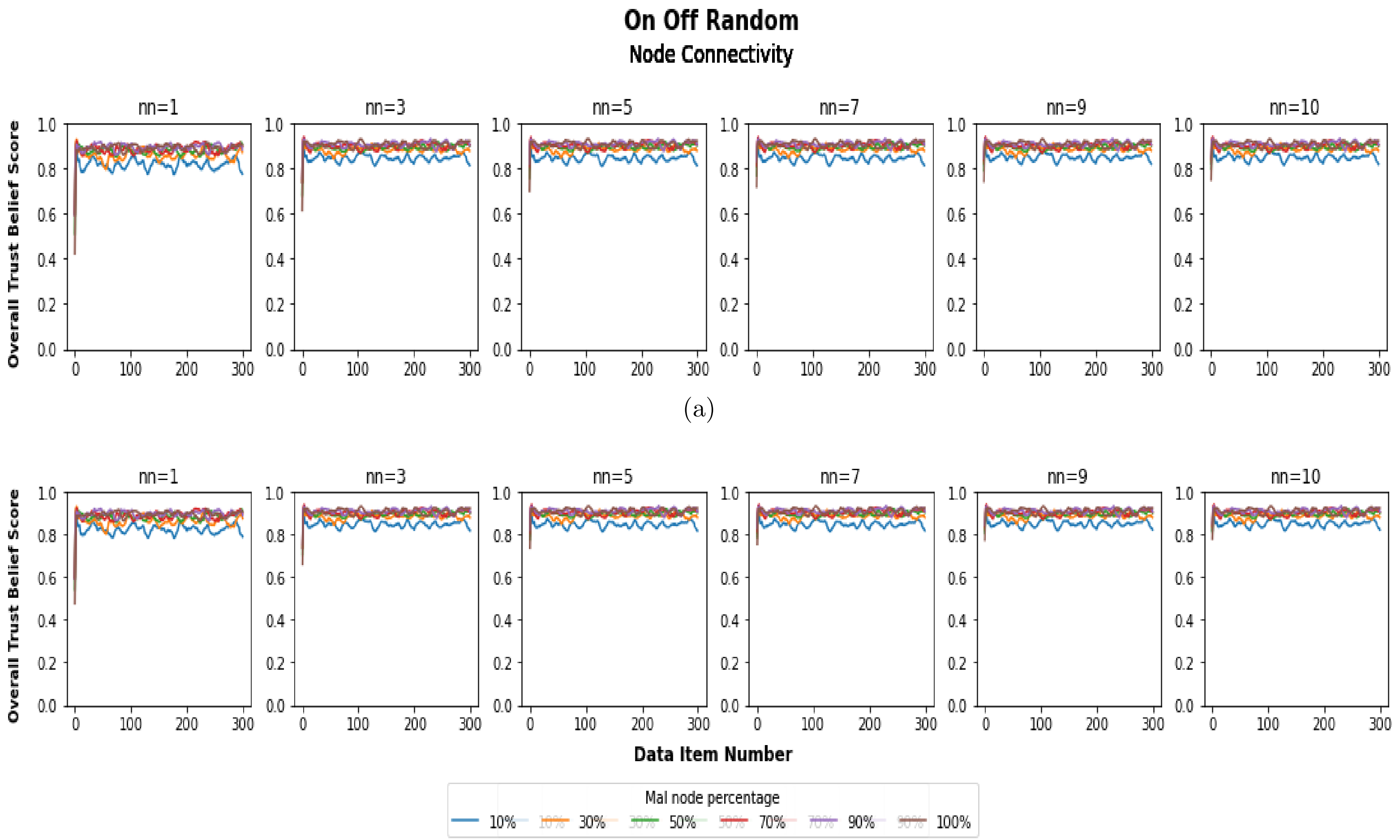
Figure 31: Overall trust belief scores for a malicious node performing individual malicious node attack concerning varying *nn* and *mnp* levels computed by: (*a*) base (*b*) extended



(a)

(b)

Figure 32: Overall trust belief scores for a malicious node performing patterned ($4G1B$) on-off attack concerning varying nn and mnp levels computed by: (a) base (b) extended



(a)

(b)

Figure 33: Overall trust belief scores for a malicious node performing random on-off attack concerning varying nn and mnp levels computed by: (a) base (b) extended

Bibliography

[1] Soroush Aalibagi, Hamidreza Mahyar, Ali Movaghar, and Harry Eugene Stanley. A matrix factorization model for hellinger-based trust management in social internet of things. *IEEE Transactions on Dependable and Secure Computing*, 2021.

[2] Melisa Acosta-Coll, Francisco Ballester-Merelo, Marcos Martinez-Peiró, De la Hoz-Franco, et al. Real-time early warning system design for pluvial flash floods—a review. *Sensors*, 18(7):2255, 2018.

[3] Nuray Baltaci Akhuseyinoglu, Mai Abdelhakim, and Prashant Krishnamurthy. A multi-disciplinary survey of trust in social sciences: Key concepts, methods, measures, and implications for computer networks. Unpublished, University of Pittsburgh, N.D.

[4] Nuray Baltaci Akhuseyinoglu and James Joshi. Access control approaches for smart cities. In Fadi Al-Turjman and Muhammad Imran, editors, *IoT Technologies in Smart Cities: From sensors to big data, security and trust*. Institution of Engineering and Technology, London, 2020.

[5] Nuray Baltaci Akhuseyinoglu, Maryam Karimi, Mai Abdelhakim, and Prashant Krishnamurthy. On automated trust computation in iot with multiple attributes and subjective logic. In *2020 IEEE 45th Conference on Local Computer Networks (LCN)*, pages 267–278. IEEE, 2020.

[6] Karim Akilal, Hachem Slimani, and Mawloud Omar. A very fast and robust trust inference algorithm in weighted signed social networks using controversy, eclecticism, and reciprocity. *Computers & Security*, 83:68–78, 2019.

[7] Gene M Alarcon, Charles Walter, Anthony M Gibson, Rose F Gamble, August Capiola, Sarah A Jessup, and Tyler J Ryan. Would you fix this code for me? effects of repair source and commenting on trust in code repair. *Systems*, 8(1):8, 2020.

[8] Mohammad Dahman Alshehri and Farookh Khadeer Hussain. A comparative analysis of scalable and context-aware trust management approaches for internet of things. In *International conference on neural information processing*, pages 596–605. Springer, 2015.

[9] Mohammad Dahman Alshehri and Farookh Khadeer Hussain. A centralized trust management mechanism for the internet of things (ctm-iot). In *International Confer-*

ence on Broadband and Wireless Computing, Communication and Applications, pages 533–543. Springer, 2017.

[10] Nava Ashraf, Iris Bohnet, and Nikita Piankov. Decomposing trust and trustworthiness. *Experimental economics*, 9(3):193–208, 2006.

[11] Sarah Asiri and Ali Miri. An iot trust and reputation model based on recommender systems. In *2016 14th Annual Conference on Privacy, Security and Trust (PST)*, pages 561–568. IEEE, 2016.

[12] Luigi Atzori, Antonio Iera, Giacomo Morabito, and Michele Nitti. The social internet of things (siot)–when social networks meet the internet of things: Concept, architecture and network characterization. *Computer networks*, 56(16):3594–3608, 2012.

[13] Bernard Barber. The logic and limits of trust. 1983.

[14] Michèle Beyer. What does temperature compensation or compensated temperature range for pressure sensors mean? `https://blog.wika.com/knowhow/temperature-compensation-pressure-sensors/`. Accessed: 2019-02-11.

[15] Gregory A Bigley and Jone L Pearce. Straining for shared meaning in organization science: Problems of trust and distrust. *Academy of management review*, 23(3):405–421, 1998.

[16] Iris Bohnet and Steffen Huck. Repetition and reputation: Implications for trust and trustworthiness when institutions change. *American economic review*, 94(2):362–366, 2004.

[17] Gary E Bolton, Elena Katok, and Axel Ockenfels. How effective are electronic reputation mechanisms? an experimental investigation. *Management science*, 50(11):1587–1602, 2004.

[18] Branko Božič, Sabina Siebert, and Graeme Martin. A grounded theory study of factors and conditions associated with customer trust recovery in a retailer. *Journal of Business Research*, 109:440–448, 2020.

[19] Ludwig Bstieler. Trust formation in collaborative new product development. *Journal of Product Innovation Management*, 23(1):56–72, 2006.

[20] Sonja Buchegger and Jean-Yves Le Boudec. A robust reputation system for mobile ad-hoc networks. Technical report, EPFL, 2003.

[21] Vincent Buskens and Werner Raub. Embedded trust: Control and learning. *Advances in group processes*, 19(2002):167–202, 2002.

[22] John K Butler Jr. Toward understanding and measuring conditions of trust: Evolution of a conditions of trust inventory. *Journal of management*, 17(3):643–663, 1991.

[23] Younghun Chae, Lisa Cingiser DiPippo, and Yan Lindsay Sun. Trust management for defending on-off attacks. *IEEE Transactions on Parallel and Distributed Systems*, 26(4):1178–1191, 2014.

[24] Arjun Chaudhuri and Morris B Holbrook. The chain of effects from brand trust and brand affect to brand performance: the role of brand loyalty. *Journal of marketing*, 65(2):81–93, 2001.

[25] Ray Chen, Fenye Bao, and Jia Guo. Trust-based service management for social internet of things systems. *IEEE transactions on dependable and secure computing*, 13(6):684–696, 2015.

[26] Ray Chen, Jia Guo, and Fenye Bao. Trust management for soa-based iot and its application to service composition. *IEEE Transactions on Services Computing*, 9(3):482–495, 2016.

[27] Yue Chen, Xiangbin Yan, Weiguo Fan, and Michael Gordon. The joint moderating role of trust propensity and gender on consumers' online shopping behavior. *Computers in Human Behavior*, 43:272–283, 2015.

[28] Jinsook Cho. The mechanism of trust and distrust formation and their relational outcomes. *Journal of retailing*, 82(1):25–35, 2006.

[29] James S Coleman. *Foundations of social theory*. Harvard university press, 1994. p. 4.

[30] Jason A Colquitt, Brent A Scott, and Jeffery A LePine. Trust, trustworthiness, and trust propensity: A meta-analytic test of their unique relationships with risk taking and job performance. *Journal of applied psychology*, 92(4):909, 2007.

[31] Juliet Corbin and Anselm Strauss. *Basics of qualitative research: Techniques and procedures for developing grounded theory.* Sage publications, 2014.

[32] MITRE Corporation. Common Vulnerabilities Database. `https://cve.mitre.org`, 1999. Accessed: 2023-03-22.

[33] Cynthia L Corritore, Beverly Kracher, and Susan Wiedenbeck. On-line trust: concepts, evolving themes, a model. *International journal of human-computer studies*, 58(6):737–758, 2003.

[34] Steven C Currall and Andrew C Inkpen. A multilevel approach to trust in joint ventures. *Journal of international business studies*, 33(3):479–495, 2002.

[35] Steven C Currall and Timothy A Judge. Measuring trust between organizational boundary role persons. *Organizational behavior and Human Decision processes*, 64(2):151–170, 1995.

[36] Tushar Kanti Das and Bing-Sheng Teng. Trust, control, and risk in strategic alliances: An integrated framework. *Organization studies*, 22(2):251–283, 2001.

[37] Morton Deutsch. Trust and suspicion. *Journal of conflict resolution*, 2(4):265–279, 1958.

[38] Morton Deutsch. The effect of motivational orientation upon trust and suspicion. *Human relations*, 13(2):123–139, 1960.

[39] Edsger W Dijkstra et al. A note on two problems in connexion with graphs. *Numerische mathematik*, 1(1):269–271, 1959.

[40] Ikram Ud Din, Mohsen Guizani, Byung-Seo Kim, Suhaidi Hassan, and Muhammad Khurram Khan. Trust management techniques for the internet of things: A survey. *IEEE Access*, 7:29763–29787, 2018.

[41] Kurt T Dirks and Donald L Ferrin. Trust in leadership: Meta-analytic findings and implications for research and practice. *Journal of applied psychology*, 87(4):611, 2002.

[42] Kurt T Dirks, Peter H Kim, Donald L Ferrin, and Cecily D Cooper. Understanding the effects of substantive responses on trust following a transgression. *Organizational Behavior and Human Decision Processes*, 114(2):87–103, 2011.

[43] Patricia M Doney and Joseph P Cannon. An examination of the nature of trust in buyer–seller relationships. *Journal of marketing*, 61(2):35–51, 1997.

[44] Xinxin Fan, Ling Liu, Mingchu Li, and Zhiyuan Su. Grouptrust: Dependable trust management. *IEEE Transactions on Parallel and Distributed Systems*, 28(4):1076–1090, 2016.

[45] Fernando Flores and Robert C Solomon. Creating trust. *Business Ethics Quarterly*, pages 205–232, 1998.

[46] Mikhail Fomichev, Flor Álvarez, Daniel Steinmetzer, Paul Gardner-Stephen, and Matthias Hollick. Survey and systematization of secure device pairing. *IEEE Communications Surveys & Tutorials*, 20(1):517–550, 2017.

[47] Vincenz Frey. Boosting trust by facilitating communication: A model of trustee investments in information sharing. *Rationality and Society*, 29(4):471–503, 2017.

[48] Vincenz Frey, Vincent Buskens, and Rense Corten. Investments in and returns on network embeddedness: An experiment with trust games. *Social Networks*, 56:81–92, 2019.

[49] Mario Frustaci, Pasquale Pace, Gianluca Aloi, and Giancarlo Fortino. Evaluating critical security issues of the iot world: present and future challenges. *IEEE Internet of Things Journal*, 5(4):2483–2495, 2018.

[50] Francis Fukuyama. *Trust: The social virtues and the creation of prosperity*, volume 99. Free press New York, 1995.

[51] Janice M Fulford. Guidance for selecting and conducting evaluations of hydrologic instruments and equipment at the usgs hydrologic instrumentation facility. `https://water.usgs.gov/hif/services/evaluations/HIFEvaluationGuidance.pdf`. Accessed: 2019-02-11.

[52] C Ashley Fulmer and Michele J Gelfand. At what level (and in whom) we trust: Trust across multiple organizational levels. *Journal of management*, 38(4):1167–1230, 2012.

[53] David Gefen, Elena Karahanna, and Detmar W Straub. Trust and tam in online shopping: An integrated model. *MIS quarterly*, 27(1):51–90, 2003.

[54] David Gefen and Detmar W Straub. Consumer trust in b2c e-commerce and the importance of social presence: experiments in e-products and e-services. *Omega*, 32(6):407–424, 2004.

[55] Dimitrios Georgakopoulos, Prem Prakash Jayaraman, Maria Fazia, Massimo Villari, and Rajiv Ranjan. Internet of things and edge cloud computing roadmap for manufacturing. *IEEE Cloud Computing*, 3(4):66–73, 2016.

[56] Ammar Gharaibeh, Mohammad A Salahuddin, Sayed Jahed Hussini, Abdallah Khreishah, Issa Khalil, Mohsen Guizani, and Ala Al-Fuqaha. Smart cities: A survey on data management, security, and enabling technologies. *IEEE Communications Surveys & Tutorials*, 19(4):2456–2501, 2017.

[57] Kim Giffin. The contribution of studies of source credibility to a theory of interpersonal trust in the communication process. *Psychological bulletin*, 68(2):104, 1967.

[58] Nicole Gillespie and Graham Dietz. Trust repair after an organization-level failure. *Academy of management review*, 34(1):127–145, 2009.

[59] Lucy Gilson. Trust and the development of health care as a social institution. *Social science & medicine*, 56(7):1453–1468, 2003.

[60] Inc. Global Water Instrumentation. Water level sensor user manual. `http://www.globalw.com/downloads/wl400/wl400manual.pdf`. Accessed: 2019-02-11.

[61] Jennifer Golbeck. Personalizing applications through integration of inferred trust values in semantic web-based social networks. In *Semantic network analysis workshop at the 4th international semantic web conference*, volume 16, page 30. Publishing, 2005.

[62] Trudy Govier. Is it a jungle out there? trust, distrust and the construction of social reality. *Dialogue: Canadian Philosophical Review/Revue canadienne de philosophie*, 33(2):237–252, 1994.

[63] Mark S Granovetter. The strength of weak ties. In *Social networks*, pages 347–367. Elsevier, 1977.

[64] Jayavardhana Gubbi, Rajkumar Buyya, Slaven Marusic, and Marimuthu Palaniswami. Internet of things (iot): A vision, architectural elements, and future directions. *Future generation computer systems*, 29(7):1645–1660, 2013.

[65] Alya Guseva and Akos Rona-Tas. Uncertainty, risk, and trust: Russian and american credit card markets compared. *American sociological review*, pages 623–646, 2001.

[66] Mahmood Hajli. A research framework for social commerce adoption. *Information Management & Computer Security*, 21(3):144–154, 2013.

[67] Michael P Haselhuhn, Maurice E Schweitzer, and Alison M Wood. How implicit beliefs influence trust recovery. *Psychological Science*, 21(5):645–648, 2010.

[68] Khaled Hassanein and Milena Head. Manipulating perceived social presence through the web interface and its impact on attitude towards online shopping. *International Journal of Human-Computer Studies*, 65(8):689–708, 2007.

[69] Katherine Hawley. Trust, distrust and commitment. *Noûs*, 48(1):1–20, 2014.

[70] Daojing He, Chun Chen, Sammy Chan, Jiajun Bu, and Athanasios V Vasilakos. Retrust: Attack-resistant and lightweight trust management for medical sensor networks. *IEEE transactions on information technology in biomedicine*, 16(4):623–632, 2012.

[71] Friederike Hendriks, Dorothe Kienhues, and Rainer Bromme. Replication crisis= trust crisis? the effect of successful vs failed replications on laypeople's trust in researchers and research. *Public Understanding of Science*, 29(3):270–288, 2020.

[72] Marc J Hetherington and Thomas J Rudolph. Priming, performance, and the dynamics of political trust. *The Journal of Politics*, 70(2):498–512, 2008.

[73] Kevin Anthony Hoff and Masooda Bashir. Trust in automation: Integrating empirical evidence on factors that influence trust. *Human factors*, 57(3):407–434, 2015.

[74] Jess Hohenstein and Malte Jung. Ai as a moral crumple zone: The effects of ai-mediated communication on attribution and trust. *Computers in Human Behavior*, 106:106190, 2020.

[75] C-L Hwang and Abu Syed Md Masud. *Multiple Objective Decision Making–Methods and Applications: A State-of-the-Art Survey*, volume 164. Springer Science & Business Media, 2012.

[76] Ching-Lai Hwang and Kwangsun Yoon. Methods for multiple attribute decision making. In *Multiple attribute decision making*, pages 58–191. Springer, 1981.

[77] Audun Jøsang. A logic for uncertain probabilities. *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, 9(03):279–311, 2001.

[78] 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.

[79] Tiina Kähkönen. Employee trust repair after organizational change. *Journal of Organizational Change Management*, 2020.

[80] Pradip P Kalbar, Subhankar Karmakar, and Shyam R Asolekar. Selection of an appropriate wastewater treatment technology: A scenario-based multiple-attribute decision-making approach. *Journal of environmental management*, 113:158–169, 2012.

[81] 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*, pages 640–651, 2003.

[82] Tom Karygiannis, Bernard Eydt, Greg Barber, Lynn Bunn, and Ted Phillips. Guidelines for securing radio frequency identification (rfid) systems. *NIST Special publication*, 80:1–154, 2007.

[83] Kostas Katsalis, Navid Nikaein, and Andy Edmonds. Multi-domain orchestration for nfv: Challenges and research directions. In *2016 15th International Conference on Ubiquitous Computing and Communications and 2016 International Symposium on Cyberspace and Security (IUCC-CSS)*, pages 189–195. IEEE, 2016.

[84] Herbert W Kee and Robert E Knox. Conceptual and methodological considerations in the study of trust and suspicion. *Journal of conflict resolution*, 14(3):357–366, 1970.

[85] Srinivasan Keshav. How to read a paper. *ACM SIGCOMM Computer Communication Review*, 37(3):83–84, 2007.

[86] Peter H Kim, Kurt T Dirks, Cecily D Cooper, and Donald L Ferrin. When more blame is better than less: The implications of internal vs. external attributions for the repair of trust after a competence-vs. integrity-based trust violation. *Organizational behavior and human decision processes*, 99(1):49–65, 2006.

[87] Peter H Kim, Donald L Ferrin, Cecily D Cooper, and Kurt T Dirks. Removing the shadow of suspicion: the effects of apology versus denial for repairing competence-versus integrity-based trust violations. *Journal of applied psychology*, 89(1):104, 2004.

[88] Sanghyun Kim and Hyunsun Park. Effects of various characteristics of social commerce (s-commerce) on consumers' trust and trust performance. *International Journal of Information Management*, 33(2):318–332, 2013.

[89] Yeolib Kim and Robert A Peterson. A meta-analysis of online trust relationships in e-commerce. *Journal of Interactive Marketing*, 38:44–54, 2017.

[90] Elad Klein and Joshua Robison. Like, post, and distrust? how social media use affects trust in government. *Political Communication*, 37(1):46–64, 2020.

[91] George Klir and Bo Yuan. *Fuzzy sets and fuzzy logic*, volume 4. Prentice hall New Jersey, 1995.

[92] Frank Hyneman Knight. *Risk, uncertainty and profit*, volume 31. Houghton Mifflin, 1921.

[93] Peter Kollock. The emergence of exchange structures: An experimental study of uncertainty, commitment, and trust. *American Journal of sociology*, 100(2):313–345, 1994.

[94] Shantanu Konwar, Amrita Bose Paul, Sukumar Nandi, and Santosh Biswas. Mcdm based trust model for secure routing in wireless mesh networks. In *2011 World Congress on Information and Communication Technologies*, pages 910–915. IEEE, 2011.

[95] Roderick M Kramer. Trust and distrust in organizations: Emerging perspectives, enduring questions. *Annual review of psychology*, 50(1):569–598, 1999.

[96] Albertus Laan, Niels Noorderhaven, Hans Voordijk, and Geert Dewulf. Building trust in construction partnering projects: An exploratory case-study. *Journal of purchasing and supply management*, 17(2):98–108, 2011.

[97] Kim Langfield-Smith. The relations between transactional characteristics, trust and risk in the start-up phase of a collaborative alliance. *Management Accounting Research*, 19(4):344–364, 2008.

[98] Richard Laugharne and Stefan Priebe. Trust, choice and power in mental health. *Social psychiatry and psychiatric epidemiology*, 41(11):843–852, 2006.

[99] Paul J Lavrakas. *Encyclopedia of survey research methods*. Sage Publications, 2008.

[100] John D Lee and Katrina A See. Trust in automation: Designing for appropriate reliance. *Human factors*, 46(1):50–80, 2004.

[101] Lai-Ying Leong, Teck-Soon Hew, Keng-Boon Ooi, and Alain Yee-Loong Chong. Predicting the antecedents of trust in social commerce–a hybrid structural equation modeling with neural network approach. *Journal of Business Research*, 110:24–40, 2020.

[102] Roy J Lewicki, Barbara B Bunker, et al. Developing and maintaining trust in work relationships. *Trust in organizations: Frontiers of theory and research*, 114:139, 1996.

[103] Roy J Lewicki, Daniel J McAllister, and Robert J Bies. Trust and distrust: New relationships and realities. *Academy of management Review*, 23(3):438–458, 1998.

[104] Roy J Lewicki, Edward C Tomlinson, and Nicole Gillespie. Models of interpersonal trust development: Theoretical approaches, empirical evidence, and future directions. *Journal of management*, 32(6):991–1022, 2006.

[105] J David Lewis and Andrew Weigert. Trust as a social reality. *Social forces*, 63(4):967–985, 1985.

[106] Jinghui Li, Bifei Mao, Zhizhang Liang, Zeqi Zhang, Qiushi Lin, and Xin Yao. Trust and trustworthiness: What they are and how to achieve them. In *2021 IEEE International Conference on Pervasive Computing and Communications Workshops and other Affiliated Events (PerCom Workshops)*, pages 711–717. IEEE, 2021.

[107] Nan Li, Vijay Varadharajan, and Surya Nepal. Context-aware trust management system for iot applications with multiple domains. In *2019 IEEE 39th International Conference on Distributed Computing Systems (ICDCS)*, pages 1138–1148. IEEE, 2019.

[108] Wenjia Li, Houbing Song, and Feng Zeng. Policy-based secure and trustworthy sensing for internet of things in smart cities. *IEEE Internet of Things Journal*, 5(2):716–723, 2017.

[109] Zhiting Lin and Liang Dong. Clarifying trust in social internet of things. *IEEE Transactions on Knowledge and Data Engineering*, 30(2):234–248, 2017.

[110] Baozhou Lu, Weiguo Fan, and Mi Zhou. Social presence, trust, and social commerce purchase intention: An empirical research. *Computers in Human Behavior*, 56:225–237, 2016.

[111] Niklas Luhmann. *Trust and power*. London:Wiley, 1979.

[112] Nguyen Cong Luong, Dinh Thai Hoang, Ping Wang, Dusit Niyato, Dong In Kim, and Zhu Han. Data collection and wireless communication in internet of things (iot) using economic analysis and pricing models: A survey. *IEEE Communications Surveys & Tutorials*, 18(4):2546–2590, 2016.

[113] Theo Lynn. Dear cloud, i think we have trust issues: Cloud computing contracts and trust. In *Data Privacy and Trust in Cloud Computing*, pages 21–42. Palgrave Macmillan, Cham, 2020.

[114] Theo Lynn, Lisa van der Werff, and Grace Fox. Understanding trust and cloud computing: An integrated framework for assurance and accountability in the cloud. In *Data Privacy and Trust in Cloud Computing*, pages 1–20. Palgrave Macmillan, Cham, 2020.

[115] Zhenxing Eddie Mao, Margie F Jones, Mimi Li, Wei Wei, and Jiaying Lyu. Sleeping in a stranger's home: A trust formation model for airbnb. *Journal of Hospitality and Tourism Management*, 42:67–76, 2020.

[116] Venkata Marella, Bikesh Upreti, Jani Merikivi, and Virpi Kristiina Tuunainen. Understanding the creation of trust in cryptocurrencies: the case of bitcoin. *Electronic Markets*, pages 1–13, 2020.

[117] Roger C Mayer and James H Davis. The effect of the performance appraisal system on trust for management: A field quasi-experiment. *Journal of applied psychology*, 84(1):123, 1999.

[118] Roger C Mayer, James H Davis, and F David Schoorman. An integrative model of organizational trust. *Academy of management review*, 20(3):709–734, 1995. pg. 712.

[119] Daniel J McAllister. Affect-and cognition-based trust as foundations for interpersonal cooperation in organizations. *Academy of management journal*, 38(1):24–59, 1995.

[120] D Harrison McKnight and Norman L Chervany. What is trust? a conceptual analysis and an interdisciplinary model. *AMCIS 2000 proceedings*, page 382, 2000.

[121] D Harrison McKnight and Norman L Chervany. What trust means in e-commerce customer relationships: An interdisciplinary conceptual typology. *International journal of electronic commerce*, 6(2):35–59, 2001.

[122] D Harrison McKnight, Vivek Choudhury, and Charles Kacmar. Developing and validating trust measures for e-commerce: An integrative typology. *Information systems research*, 13(3):334–359, 2002.

[123] D Harrison McKnight, Larry L Cummings, and Norman L Chervany. Initial trust formation in new organizational relationships. *Academy of Management review*, 23(3):473–490, 1998.

[124] David Mechanic and Sharon Meyer. Concepts of trust among patients with serious illness. *Social science & medicine*, 51(5):657–668, 2000.

[125] Carolina VL Mendoza and João H Kleinschmidt. Mitigating on-off attacks in the internet of things using a distributed trust management scheme. *International Journal of Distributed Sensor Networks*, 11(11):859731, 2015.

[126] Carolina VL Mendoza and João H Kleinschmidt. Defense for selective attacks in the iot with a distributed trust management scheme. In *2016 IEEE International Symposium on Consumer Electronics (ISCE)*, pages 53–54. IEEE, 2016.

[127] Christine Moorman, Gerald Zaltman, and Rohit Deshpande. Relationships between providers and users of market research: The dynamics of trust within and between organizations. *Journal of marketing research*, 29(3):314–328, 1992.

[128] Robert M Morgan and Shelby D Hunt. The commitment-trust theory of relationship marketing. *Journal of marketing*, 58(3):20–38, 1994.

[129] Suneth Namal, Hasindu Gamaarachchi, Gyu MyoungLee, and Tai-Won Um. Autonomic trust management in cloud-based and highly dynamic iot applications. In *2015 ITU Kaleidoscope: Trust in the Information Society (K-2015)*, pages 1–8. IEEE, 2015.

[130] National Centers for Environmental Information. Climate data online. `https://www.ncdc.noaa.gov/cdo-web/datasets#LCD`. Accessed: 2019-03-08.

[131] Felix Naumann. Data fusion and data quality. In *Proc. of the New Techniques and Technologies for Statistics Seminar, Sorrento, Italy, May 1998*. Humboldt-Universität zu Berlin, Mathematisch-Naturwissenschaftliche Fakultät II, 1998.

[132] Andreas I Nicolaou and D Harrison McKnight. Perceived information quality in data exchanges: Effects on risk, trust, and intention to use. *Information systems research*, 17(4):332–351, 2006.

[133] Navya Sri Nizamkari. A graph-based trust-enhanced recommender system for service selection in iot. In *2017 International Conference on Inventive Systems and Control (ICISC)*, pages 1–5. IEEE, 2017.

[134] Talal H Noor, Quan Z Sheng, Lina Yao, Schahram Dustdar, and Anne HH Ngu. Cloudarmor: Supporting reputation-based trust management for cloud services. *IEEE transactions on parallel and distributed systems*, 27(2):367–380, 2015.

[135] Bart Nooteboom. Trust, opportunism and governance: A process and control model. *Organization studies*, 17(6):985–1010, 1996.

[136] Bart Nooteboom. *Trust: Forms, foundations, functions, failures and figures*. Edward Elgar Publishing, 2002.

[137] Pippa Norris. The conceptual framework of political support. In *Handbook on political trust*. Edward Elgar Publishing, 2017.

[138] Sachiko Ozawa and Pooja Sripad. How do you measure trust in the health system? a systematic review of the literature. *Social science & medicine*, 91:10–14, 2013.

[139] Kishan Patel. Duty cycle: What is it and how is it used? `https://www.atlasrfidstore.com/rfid-insider/duty-cycle-what-is-it-and-how-is-it-used`, 2018. Accessed: 2023-01-31.

[140] Paul A Pavlou and David Gefen. Building effective online marketplaces with institution-based trust. *Information systems research*, 15(1):37–59, 2004.

[141] Pecan Street Inc. Dataport. Weather data. `https://dataport.cloud`. Accessed: 2019-02-18.

[142] Special Competitive Studies Project. Mid-decade challenges to national competitiveness, 2022.

[143] Rogelio Puente-Diaz and Judith Cavazos-Arroyo. 7–0? that is awful! should i trust my national team again?: The role of mindsets in team trust. *International Journal of Psychology*, 55(2):315–322, 2020.

[144] Robert D Putnam et al. *Bowling alone: The collapse and revival of American community.* Simon and schuster, 2000.

[145] Sherif Emad Abdel Rafey, Ayman Abdel-Hamid, and Mohamad Abou El-Nasr. Cbstm-iot: Context-based social trust model for the internet of things. In *2016 International Conference on Selected Topics in Mobile & Wireless Networking (MoWNeT)*, pages 1–8. IEEE, 2016.

[146] Glenn D Reeder and Marilynn B Brewer. A schematic model of dispositional attribution in interpersonal perception. *Psychological Review*, 86(1):61, 1979.

[147] Evelien Reusen and Kristof Stouthuysen. Trust transfer and partner selection in interfirm relationships. *Accounting, Organizations and Society*, 81:101081, 2020.

[148] Jens Riegelsberger, M Angela Sasse, and John D McCarthy. The mechanics of trust: A framework for research and design. *International Journal of Human-Computer Studies*, 62(3):381–422, 2005.

[149] Ron Roberts and Paul Goodwin. Weight approximations in multi-attribute decision models. *Journal of Multi-Criteria Decision Analysis*, 11(6):291–303, 2002.

[150] William Ross and Jessica LaCroix. Multiple meanings of trust in negotiation theory and research: A literature review and integrative model. *International Journal of Conflict Management*, 1996.

[151] Julian B Rotter. A new scale for the measurement of interpersonal trust. *Journal of personality*, 1967.

[152] Julian B Rotter. Generalized expectancies for interpersonal trust. *American psychologist*, 26(5):443, 1971.

[153] Denise M Rousseau, Sim B Sitkin, Ronald S Burt, and Colin Camerer. Not so different after all: A cross-discipline view of trust. *Academy of management review*, 23(3):393–404, 1998.

[154] Yosra Ben Saied, Alexis Olivereau, Djamal Zeghlache, and Maryline Laurent. Trust management system design for the internet of things: A context-aware and multi-service approach. *Computers & Security*, 39:351–365, 2013.

[155] Tetsuya Sato, Yusuke Yamani, Molly Liechty, and Eric T Chancey. Automation trust increases under high-workload multitasking scenarios involving risk. *Cognition, Technology & Work*, 22(2):399–407, 2020.

[156] Oliver Schilke and Karen S Cook. A cross-level process theory of trust development in interorganizational relationships. *Strategic Organization*, 11(3):281–303, 2013.

[157] F David Schoorman, Roger C Mayer, and James H Davis. An integrative model of organizational trust: Past, present, and future, 2007.

[158] Paul H Schurr and Julie L Ozanne. Influences on exchange processes: Buyers' preconceptions of a seller's trustworthiness and bargaining toughness. *Journal of consumer research*, 11(4):939–953, 1985.

[159] Donald P Schwab. Construct validity in organizational behavior. *Res Organ Behav*, 2:3–43, 1980.

[160] Maurice E Schweitzer, John C Hershey, and Eric T Bradlow. Promises and lies: Restoring violated trust. *Organizational behavior and human decision processes*, 101(1):1–19, 2006.

[161] SST Sensing. Data sheet liquid level switches optomax digital series. `https://sstsensing.com/wp-content/uploads/2015/10/DS0032rev14_LLDigital.pdf`, 2017. Accessed: 2023-02-01.

[162] Glenn Shafer. *A mathematical theory of evidence*, volume 42. Princeton university press, 1976.

[163] Hsu-Shih Shih, Huan-Jyh Shyur, and E Stanley Lee. An extension of topsis for group decision making. *Mathematical and Computer Modelling*, 45(7-8):801–813, 2007.

[164] John Short, Ederyn Williams, and Bruce Christie. *The social psychology of telecommunications*. Toronto; London; New York: Wiley, 1976.

[165] Sim B Sitkin and Amy L Pablo. Reconceptualizing the determinants of risk behavior. *Academy of management review*, 17(1):9–38, 1992.

[166] Sim B Sitkin and Nancy L Roth. Explaining the limited effectiveness of legalistic "remedies" for trust/distrust. *Organization science*, 4(3):367–392, 1993.

[167] Boris Škorić, Sebastiaan JA de Hoogh, and Nicola Zannone. Flow-based reputation with uncertainty: evidence-based subjective logic. *International Journal of Information Security*, 15(4):381–402, 2016.

[168] Paul Slovic. Perceived risk, trust, and democracy. *Risk analysis*, 13(6):675–682, 1993.

[169] Mark Snyder and Arthur A Stukas Jr. Interpersonal processes: The interplay of cognitive, motivational, and behavioral activities in social interaction. *Annual review of psychology*, 50(1):273–303, 1999.

[170] Heesuk Son, Namyong Kang, Bumjin Gwak, and Dongman Lee. An adaptive iot trust estimation scheme combining interaction history and stereotypical reputation. In *2017 14th IEEE Annual Consumer Communications & Networking Conference (CCNC)*, pages 349–352. IEEE, 2017.

[171] LC Stack. Trust. *Dimensions of personality*, 1978.

[172] Elaine Stasiulis, Barbara E Gibson, Fiona Webster, and Katherine M Boydell. Resisting governance and the production of trust in early psychosis intervention. *Social Science & Medicine*, page 112948, 2020.

[173] Katherine J Stewart. Trust transfer on the world wide web. *Organization science*, 14(1):5–17, 2003.

[174] William G Stillwell, David A Seaver, and Ward Edwards. A comparison of weight approximation techniques in multiattribute utility decision making. *Organizational behavior and human performance*, 28(1):62–77, 1981.

[175] David Tipper and Prashant Krishnamurthy. Digital sovereignty and resilience. *Available at SSRN 4178621*, 2022.

[176] Santtu Toivonen, Gabriele Lenzini, and Ilkka Uusitalo. Context-aware trust evaluation functions for dynamic reconfigurable systems. *MTW*, 190, 2006.

[177] Edward C Tomlinson, Brian R Dineen, and Roy J Lewicki. The road to reconciliation: Antecedents of victim willingness to reconcile following a broken promise. *Journal of management*, 30(2):165–187, 2004.

[178] Edward C Tomlinson and Roger C Mayer. The role of causal attribution dimensions in trust repair. *Academy of Management Review*, 34(1):85–104, 2009.

[179] Nguyen B Truong, Tai-Won Um, and Gyu Myoung Lee. A reputation and knowledge based trust service platform for trustworthy social internet of things. *Innovations in clouds, internet and networks (ICIN), Paris, France*, 2016.

[180] Chih-Hsiung Tu. On-line learning migration: from social learning theory to social presence theory in a cmc environment. *Journal of network and computer applications*, 23(1):27–37, 2000.

[181] Wen Tu and Yuanyuan Xu. The evolution of interorganizational trust in cross-sector collaborations: Two comparative cases from china. *Nonprofit Management and Leadership*, 2020.

[182] Amos Tversky and Daniel Kahneman. Judgment under uncertainty: Heuristics and biases. *science*, 185(4157):1124–1131, 1974.

[183] Bart S Vanneste, Phanish Puranam, and Tobias Kretschmer. Trust over time in exchange relationships: Meta-analysis and theory. *Strategic Management Journal*, 35(12):1891–1902, 2014.

[184] Maria L Vélez, José M Sánchez, and Concha Álvarez-Dardet. Management control systems as inter-organizational trust builders in evolving relationships: Evidence from a longitudinal case study. *Accounting, Organizations and Society*, 33(7-8):968–994, 2008.

[185] Yating Wang, Yen-Cheng Lu, Ing-Ray Chen, Jin-Hee Cho, Ananthram Swami, and Chang-Tien Lu. Logittrust: A logit regression-based trust model for mobile ad hoc networks. In *6th ASE International Conference on Privacy, Security, Risk and Trust, Boston, MA*, pages 1–10, 2014.

[186] Anne Marie Warren, Ainin Sulaiman, and Noor Ismawati Jaafar. Social media effects on fostering online civic engagement and building citizen trust and trust in institutions. *Government Information Quarterly*, 31(2):291–301, 2014.

[187] Oliver E Williamson. Calculativeness, trust, and economic organization. *The journal of law and economics*, 36(1, Part 2):453–486, 1993.

[188] Xu Wu and Feng Li. A multi-domain trust management model for supporting rfid applications of iot. *PloS one*, 12(7):e0181124, 2017.

[189] Zheng Yan, Peng Zhang, and Athanasios V Vasilakos. A survey on trust management for internet of things. *Journal of network and computer applications*, 42:120–134, 2014.

[190] K Paul Yoon and Ching-Lai Hwang. *Multiple attribute decision making: an introduction*, volume 104. Sage publications, 1995.

[191] Vladimir Zadorozhny, Prashant Krishnamurthy, Mai Abdelhakim, Kostantinos Pelechrinis, and Jiawei Xu. Data credence in iot: Vision and challenges. *Open Journal of Internet of Things (OJIOT), v. 3, N. 1, 114-126, 2017. Special Issue: Proceedings of the International Workshop on Very Large Internet of Things (VLIoT 2017) in conjunction with the VLDB 2017 Conference.*, 3(1):114–126, 2017.

[192] Andrea Zanella, Nicola Bui, Angelo Castellani, Lorenzo Vangelista, and Michele Zorzi. Internet of things for smart cities. *IEEE Internet of Things journal*, 1(1):22–32, 2014.

[193] Kuan Zhang, Jianbing Ni, Kan Yang, Xiaohui Liang, Ju Ren, and Xuemin Sherman Shen. Security and privacy in smart city applications: Challenges and solutions. *IEEE Communications Magazine*, 55(1):122–129, 2017.

[194] Cai-Nicolas Ziegler and Jennifer Golbeck. Models for trust inference in social networks. In *Propagation phenomena in real world networks*, pages 53–89. Springer, 2015.