A Classification of Trust Computation Models for Service-Oriented Internet of Things Systems

Jia Guo and Ing-Ray Chen
Department of Computer Science
Virginia Tech
{jiaguo, irchen}@vt.edu

Abstract— **Future Internet of Things (IoT) systems will connect the physical world into cyberspace everywhere and everything via billions of smart objects and are expected to have a high economic impact. To date there is little work on trust computation in IoT environments for security enhancement, especially for dealing with misbehaving owners of IoT devices that provide services to other IoT devices in the system. In this paper we classify trust computation models to-date for IoT systems. Our approach is to classify existing trust computation models based on five design dimensions: trust composition, trust propagation, trust aggregation, trust update, and trust formation. We summarize advantages and drawbacks of each dimension's options, and highlight the effectiveness of defense mechanisms against malicious attacks. We also summarize the most and least studied trust computation techniques in the literature and provide insight on the effectiveness of trust computation techniques as applying to IoT systems. Finally, we identify gaps in IoT trust computation research and suggest future research directions.**

Keywords— trust, security, Internet of things (IoT), social IoT.

1. INTRODUCTION

It is envisioned that a future service–oriented Internet of Things (IoT) system will connect a great amount of smart objects in the physical world, including radio frequency identification (RFID) tags, sensors, actuators, PDAs, and smartphones, as well as virtual objects in cyberspace such as data and virtual desktops on the cloud [19, 21, 22]. The emerging paradigm of the Internet of Things (IoT) has attracted a wide variety of applications running on top of it, including e-health [27, 15], smart-home, smart-city, and smart-community [25].

A service-oriented IoT system can be viewed as a P2P owner-centric community with devices (owned by humans) requesting and providing services on behalf of the owners, and with devices establishing social relationships autonomously with other devices based on social rules set by their owners, as well as interacting with each other opportunistically as they come into contact. To best satisfy the service requester and maximize application performance, it is crucial to evaluate the trustworthiness of service providers in IoT environments. The motivation of providing a trust management system for IoT systems is clear: There are misbehaving owners and consequently misbehaving devices that may perform discriminatory attacks based on their social relationships with others for their own gain at the expense of other IoT devices which provide similar services. Further, misbehaving nodes with close social ties may collude and monopoly a class of services. Since trust provisioning in this environment inherently is fully integrated with service provisioning, the notion of trust-based service management is of paramount importance.

To date there is limited work on trust computation in IoT environments for security enhancement, especially for dealing with misbehaving owners of IoT devices that provide services to other IoT devices in the system. In this paper we classify state-of-the-art trust computation models for IoT systems. Our approach is to classify existing trust computation models based on five design dimensions: trust composition, trust propagation, trust aggregation, trust update, and trust formation. We summarize advantages and drawbacks of each dimension's options, and highlight the effectiveness of defense mechanisms against malicious attacks. We also summarize the most and least studied trust computation techniques in the literature and provide insight on the effectiveness of trust computation techniques as applying to service-oriented IoT systems. Finally, we identify gaps in IoT trust computation research and suggest future research areas.

The rest of the paper is organized as follows: Section 2 develops a classification tree for organizing existing trust computation techniques for IoT systems and explains the dimensions used. Section 3 develops a threat model and presents defense mechanisms developed in the literature against malicious attacks. Section 4 classifies existing IoT trust computation techniques following the classification tree developed. In Section 5, we summarize the most and least studied IoT trust computation techniques in the literature. We provide insight on the effectiveness of trust computation techniques as applying to IoT systems and identify research gaps that are worthy of further research efforts. Section 6 presents our conclusion and suggests future research directions.

2. CLASSIFICATION TREE

In this section, we develop a classification tree for classifying trust computation techniques. The intent is to identify research gaps in IoT trust computation research. Figure 1 shows our classification tree based on five design dimensions: trust composition, trust propagation, trust aggregation, trust update, and trust formation. It is color coded with red indicating the most visited, yellow for least visited and blue for little visited. Below we discuss each classification design dimension in detail.

2.1 Trust Composition

Trust composition refers to what components to consider in trust computation. Trust components include quality of service (QoS) trust and social trust.

2.1.1 QoS Trust

QoS trust refers to the belief that an IoT device is able to provide quality service in response to a service request. QoS trust in general refers to performance and is measured by competence,

cooperativeness, reliability, task completion capability, etc. [18] used transaction performance to measure QoS trust. [3] used end-to-end packet forwarding ratio, energy consumption, and packet delivery ratio to measure QoS trust.

2.1.2 Social Trust

Social trust derives from social relationship between owners of IoT devices and is measured by intimacy, honesty, privacy, centrality, and connectivity. [7] made use of friendship, social contact, and community of interest (CoI) to rate a rater. [8] measured social trust by connectivity, intimacy, honesty and unselfishness. Social trust is especially prevalent in social IoT systems where IoT devices must be evaluated not only based on QoS trust, i.e., a device's capability to execute a service request, but also based on social trust, i.e., s device's commitment and good will to perform a service request. Moreover, when taking in a recommendation, an IoT device may trust its socially connected devices (of their owners) over unrelated devices.

2.2 Trust Propagation

Trust propagation refers to how to propagate trust evidence to peers. In general, there are two trust propagation schemes – that is, distributed and centralized.

2.2.1 Distributed

Distributed trust propagation refers to IoT devices autonomously propagating trust observations to other IoT devices they encounter or interact with without the use of a centralized entity. This is particularly the case in which it is difficult to setup or access a centralized entity in IoT environments mimicking a mobile ad hoc network (MANET) and/or a wireless sensor network (WSN). [7] proposed a distributed trust propagation scheme for social IoT systems. [18] proposed a distributed trust system by which IoT devices each store trust values of other IoT devices in the network. In [3], each node in the network maintains a data forwarding information table by overhearing activities of its neighboring nodes.

2.2.2 Centralized

Centralized trust propagation requires the presence of a centralized entity, either a physical cloud or a virtual trust service implemented by participating IoT devices. [18] proposed a DHT (Distributed Hash Table) structure to store node trust feedbacks and answer queries for node trust. [21] proposed a centralized trust manager keeping trust information of IoT entities and selecting capable IoT devices for answering a service request.

2.3 Trust Aggregation

Trust aggregation refers to aggregating trust evidence collected through either self- observations or feedbacks from peers. Major trust aggregation techniques investigated in the literature [14] include weighted sum, belief theory, Bayesian inference (with belief discounting), fuzzy logic, and regression analysis.

2.3.1 Weighted Sum

Weighted sum is a popular technique to aggregate evidence. Many reputation systems aggregate ratings or feedbacks using weighted sum such that raters with a higher reputation or transaction relevance have a higher weight. [18] uses *credibility* (derived from QoS and social trust) as the weight associated with the recommendation or feedback provided by a rater for indirect trust aggregation. [7] also uses similarity (derived from social trust) as the weight for indirect trust aggregation. Weighted sum can also be used to aggregate direct trust (through self-observations) with indirect trust (through feedbacks or recommendations) for the same trust property (e.g., service quality). There is a further classification of whether the weights assigned to direct trust and indirect trust can be dynamically adjusted or just static at design time.

2.3.2 Belief Theory

Belief theory, also known as evidence theory or Dempster–Shafer theory (DST), is a general framework for reasoning with uncertainty, with connections to other frameworks such as probability, possibility and imprecise probability theories. [13] proposed a subjective logic which operates on subjective beliefs about the world, and used opinion metric to denote the representation of a subjective belief. [27] adopted Dempster-Shafer Theory as the underlying trust computational model to compute trust of agents in autonomous systems. The basic idea is model trust by belief, disbelief and uncertainty. A node's opinion in another node is denoted by (b, d, u, a) [14] where b, d, and u represent belief, disbelief, and uncertainty, respectively, with b+d+u=1, and a is the base rate probability in the absence of evidence. The average trust is therefore the probability expectation value computed as b+au. Subjective logic operators such as the discount and consensus operators can be used to combine opinions (self-observations or recommendations) [13].

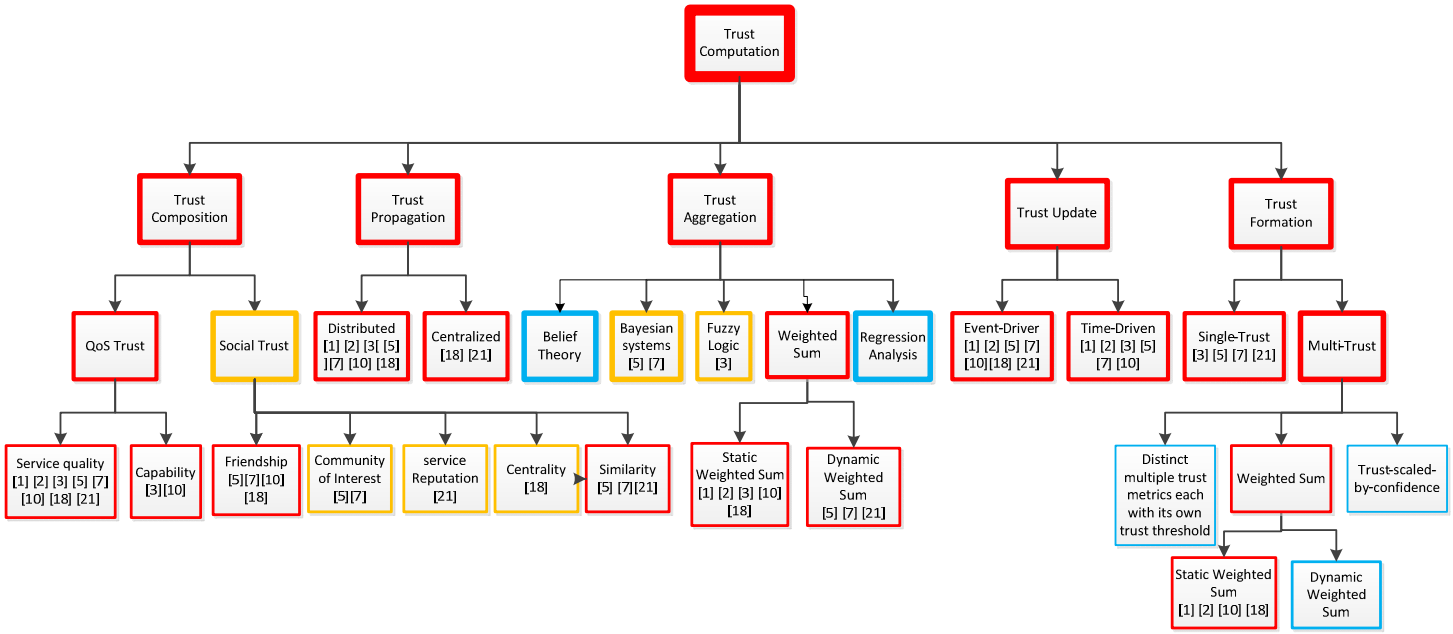

Figure 1: Classification Tree.

2.3.3 Bayesian inference with Belief Discounting

Bayesian inference treats trust as a random variable following a probability distribution with its model parameters being updated upon new observations. It is a popular trust computational model because of its simplicity and sound statistical basis. [15] proposed a Beta reputation system based on Bayesian inference with the trust value modeled as a random variable in the range of [0, 1] following Beta distribution; the amounts of positive and negative experiences are mapped to the (α, β) parameters in Beta distribution so that the average trust is computed as $\frac{\alpha}{\alpha+\beta}$. [12] applied Bayesian inference for a reputation system in a WSN, taking binary positive and negative ratings as input, and computing sensor node reputation scores. Belief discounting is applied to defend against bad-mouthing attacks (saying a good node as a bad node) and ballot-stuffing attacks (saying a bad node as a good node).

2.3.4 Fuzzy Logic

Fuzzy logic is a form of many-value logic; it deals with reasoning that is approximate rather than fixed and exact. Compared to traditional binary sets, fuzzy logic variables may have a truth value that ranges in degree between 0 and 1. Fuzzy logic has been extended to handle the concept of partial truth, where the truth value may range between completely true and completely false. Furthermore, when linguistic variables are used, these degrees may be managed by specific membership functions. Reputation or trust is represented as a fuzzy measure with membership functions describing the degrees of trust, e.g., a trust value in the range (−1.25, 1.25) denotes very low trust, (0, 2.5) low trust, (1.25, 3.75) medium trust, (2.5, 5) high trust, (3.75, 6.25) high trust, and so on. Hence, a node with a trust value of 0.25 is 75% very low trust (a membership function) and 25% low trust (another membership function). Fuzzy logic provides rules for reasoning with fuzzy measures.

2.3.5 Regression Analysis

Regression analysis is a statistical process for estimating the relationships among variables. It can be applied to estimate the relationships between trust and a set of variables characterizing the behavior of a node. [24] applied logit regression to learn the relation between cumulative evidence gathered by node i toward node j and the corresponding environmental context variables including energy-sensitivity, capability-limitation, and cost-awareness. This behavior pattern is learned dynamically and can be used to predict node j's trust behavior, i.e., whether node j, from the perspective of node i, can provide good service when called for given an environment setting characterized by a set of context variable values as input.

2.4 Trust Update

Trust update concerns when trust is updated. In general, there are two schemes - event-driven scheme and time-driven scheme.

2.4.1 Event-driven

In the event-driven scheme, all trust data in a node get updated after a transaction or event is made. This can be when a service is rendered and therefore a feedback regarding service quality is sent to the trust manager in the cloud, or recorded in each node cache for trust aggregation. A recommendation can also be sent upon request in encounter-based environments [4, 9] where nodes run into each other and request for recommendations for other nodes.

2.4.2 Time-driven

In the time-driven scheme, evidence (self-observations or recommendations) is collected periodically and trust is updated by applying a trust aggregation technique. In case no evidence is collected, trust decay over time is frequently applied because one should trust recent information more than past information. The exponential decay function with a parameter adjusting the rate of trust decay over time can be used depending on the specific application needs [4,9].

2.5 Trust Formation

Trust formation refers to how to form the overall trust out of multiple trust properties. In the literature, trust formation is considered from the aspect of single-trust or multi-trust.

2.5.1 Single-trust

Single-trust refers to the fact that only one trust property is considered in a trust protocol. For example, service quality is deemed the single most important metric in service-oriented IoT systems [7]. Therefore, an IoT device is being evaluated on its ability to produce quality service when called for. In a social IoT system, the service quality may be affected by the relationship between the service requester and the service provider, so inherently trust in service quality in a social IoT system is pair-wise. In other words, a node is more concerned with one-to-one trust toward another node based on their relationship, rather than the general reputation of the node derived from the public belief.

2.5.2 Multi-trust

Multi-trust implements the common belief that trust is multidimensional, so multiple trust properties should be considered for trust formation. [8] considered multiple trust properties including intimacy, honesty, unselfishness and competence, to assess the overall trust of a MANET node. There are multiple ways to do trust formation:

- One can just use individual trust properties without combining them together but define a minimum threshold for each trust property depending on the application requirements. For example, honesty is important, so a high threshold is used but competence may not be very critical so a low threshold is set.
- One can use *weighted sum* to combine individual trust properties together into an overall trust metric. The weight assigned can reflect the application requirements. [7] proposed to readjust the weights of direct and indirect trust to maximize the average user satisfaction experiences during the most recent time period. [21] proposed to change the weight associated with positive recommendations dynamically to derive the overall trust value. [16] proposed to use a penalty coefficient weight based on the authentication history to update trust.
- One can use a *trust-scaled-by-confidence* technique for trust formation. The idea is to scale the most important trust property with less important trust properties which serve as confidence. [25] considered competence and integrity as two trust properties for rating a node with competence being the more important trust metric. It considered two scaling schemes: (a) competence trust drops to zero if integrity trust falls below a threshold; (b) competence trust scales up (to 1 maximum) or down (to 0 minimum), depending on whether integrity trust is higher or lower than the threshold.

3. THREAT MODEL

Trust as a soft security measure is particularly applicable to service-oriented IoT systems because IoT devices owned by

Table 1: Classification of Existing Works according to the Classification Tree and Summary of Defense Mechanisms used against Trust-Related Attacks.

Classification	Work	SPA	BMA	BSA	OSA	OOA
QoS + Social / Distributed / Bayesian inference + Dynamic weighted sum / Event + Time-driven / Single-trust (Section 4.1)	2014 Chen, et al. [5][7]	Direct service quality trust assessment and feedback propagation	Social similarity to rate a recommender	Social similarity to rate a recommender	Adaptive filtering to adjust the weights of direct and indirect service quality trust dynamically	NA
QoS + Social / Distributed + Centralized / Static weighted sum / Event-driven / Multi-trust with static weighted sum (Section 4.2)	2014 Nitti, et al. [18]	Direct service quality trust assessment and feedback propagation	Credibility to rate a recommender	Credibility to rate a recommender	Long-term and short-term direct service quality trust assessment	NA
QoS / Centralized / dynamic weighted sum / Event-driven / Single-trust (Section 4.3)	2014 Saied, et al. [21]	Direct service quality trust assessment and feedback propagation	Recommender trust to rate a recommender	Recommender trust to rate a recommender	NA	NA
QoS / Distributed / Fuzzy logic + Static-weighted sum / Time-driven / Single-trust with static weighted sum (Section 4.4)	2011 Chen, et al. [3]	Direct service quality trust assessment and feedback propagation	NA	NA	NA	NA
Class 5: QoS + Social / Distributed / Static weighted sum / Event + Time-driven / Multi-trust with static weighted sum (Section 4.5)	2012 Bao, et al. [1][2]	Honesty trust assessment and feedback propagation	Honesty trust assessment and feedback propagation	Honesty trust assessment and feedback propagation	NA	NA
	2015 Chen, et al. [10]	Direct service quality trust assessment and feedback propagation	Recommender trust to rate a recommender	Recommender trust to rate a recommender	NA	NA

human beings inherently can be malicious for their own gain. Malicious users can also collude to dominate the service provider market. In this section, we enumerate possible threats to IoT systems. Our intent is to survey how existing IoT trust management protocols in the literature deal with malicious attacks and identify gaps for future research.

In an IoT system, every IoT device can be a service provider (SP) or a service requester (SR) itself. Therefore every IoT device wants to be selected to provide service for profit when it is a SP and wants to find the best SPs for best service available when it is a SR. A malicious SP node acts for its own benefit and would like to be selected for service even if the service provides is inferior. In the context of IoT, we are concerned with trust-related attacks that can disrupt the trust system. Bad-mouthing and ballot-stuffing attacks are the most common forms of reputation attacks. Self-promoting and opportunistic service attacks are the most common forms of attacks based on self-interest [7]. On-off attacks are often used by malicious nodes to evade detection. Thus, a malicious IoT device (because its owner is malicious) can perform the following trust-related attacks:

1. Self-promotion attacks (SPA): a malicious node it can promote its importance (by providing good recommendations for itself) so as to be selected as a SP, but then can provide bad or malfunctioned service.
2. Bad-mouthing attacks (BMA): a malicious node can ruin the trust of a well-behaved node (by providing bad recommendations against it) so as to decrease the chance of that node being selected for service. This is a form of collusion recommendation attack, i.e., a malicious node can collaborate with other malicious nodes to ruin the trust of a good node.
3. Ballot-stuffing attacks (BSA): a malicious node can boost the trust of a malicious node (by providing good recommendations) so as to increase the chance of that malicious node being selected as a SP. This is another form of collusion recommendation attacks, i.e., it can collaborate with other malicious nodes to boost the trust of each other.
4. Opportunistic service attacks (OSA): a malicious node can provide good service to gain high reputation opportunistically especially when it senses its reputation is dropping because of providing bad service. With good reputation, it can effectively collude with other bad node to perform bad-mouthing and ballot-stuffing attacks.
5. On-off attacks (OOA): instead of always performing best service, a malicious node can perform bad service. With on-off attacks, a malicious node performs bad service on and off (or randomly) so as to avoid being labeled as a low trust node and risk itself not being selected as a SP, as well as not being able to effectively perform bad-mouthing and ballot-stuffing attacks. One can view on-off attacks as random attacks.

4. STATE OF THE ART

Trust management for IoT is still in its infancy with limited work reported in the literature to date, possibly due to limited experiences with IoT platforms and experimentations. We found [1], [2], [3], [5], [7], [10], [18], [21] and [23] to date. In this section, we follow the classification tree to classify these existing works. We adopt the notation *a/b/c/d/e* where *a* = trust

composition, *b* = trust propagation, *c* = trust aggregation, *d* = trust update and *e* = trust formation. We also discuss defense mechanisms used (if any) in these works to defend against malicious attacks discussed in Section 3. Based on the classification tree, there are 6 classes of works as summarized in Table 1. Below we discuss in detail each classification as well as existing works that fall under.

4.1 Class 1: QoS + Social / Distributed / Bayesian inference + Dynamic weighted sum / Event + Time-driven / Single-trust

Among all works listed in Table 1, only [5],[7] fall into this classification with [5] being the preliminary work of [7]. [5],[7] use service quality (a QoS trust metric) to rate a SP, and social similarity (a social trust metric) to rate a recommender based on the concept of collaborative filtering to select feedbacks using similarity rating of friendship, social contact, and community of interest relationships as the filter.

In trust propagation, every node acts autonomously to collect evidence (through self-observations or recommendations) and also serves as a recommender upon

request. Hence it is based on distributed trust propagation. A node first collects evidence of the service quality trust and social similarity trust of adjacent nodes. Then it collects recommendations from qualified adjacent nodes about other nodes in the system.

In trust aggregation, [5],[7] use Bayesian inference to aggregate self-observations into direct trust. Social similarity-weighted sum is used to aggregate recommendations into indirect trust. A novel adaptive filtering technique is proposed to adjust weights associated with direct trust and indirect trust dynamically to minimize trust bias and maximize application performance.

In trust update, both event-driven and time-driven are considered. The direct trust is updated upon each service interaction while the indirect trust is updated periodically using peer recommendations collected during the period. The work also considers and analyzes the effect of trust decay on trust convergence rate.

In trust formation, only a single service quality trust is considered, so it falls into the single-trust category. However, [5],[7] use several social similarity metrics, i.e., friendship, social contact, and community-of-interest, and apply the weighted sum technique to combine these social similarity metrics into one to rate a recommender. The best weighting scheme to combine the three metrics into one is identified for a service-oriented IoT application.

Entry 1 of Table 1 summarizes the defense mechanisms used by [5],[7] to defend against malicious attacks. SPA is detected in the protocol design and the feedback is propagated through trust propagation. BMA and BSA are tolerated by using social similarity to rate a recommender. OSA is resolved by adaptive filtering which dynamically adjust the weights associated with direct and indirect trust to capture the opportunistic service attack behavior. However, OOA is not considered.

4.2 Class 2: QoS + Social / Distributed + Centralized / Static weighted sum / Event-driven / Multi-trust with static weighted sum

[18] falls into this classification. In trust composition, [18] considers both QoS trust and social trust. QoS trust properties considered include transaction service quality and computational capability. Social trust properties considered include centrality, relationship factor (such as ownership, co-location, co-work, social and co-brand), and credibility. In particular, credibility is used to rate a recommender who provides indirect evidence and is computed by a weighted sum of the direct trust toward the recommender and the relative centrality of the recommender.

In trust propagation, [18] is rather unique in that it considers both distributed and centralized models. In the distributed model, each node computes its own *subjective trustworthiness* toward another node. Transaction service quality trust is assessed by individual nodes after a transaction is completed, and feedbacks are propagated as indirect evidence from one node to another upon request. In the centralized model, transaction service quality feedbacks are propagated to a centralized entity making use of a Dynamic Hash Table (DHT) structure on the network to maintain the *objective trustworthiness* status (global reputation) of a node. A node can query the DHT to receive the trust value of other nodes in the network, and the DHT returns the objective trustworthiness scores after searching the database.

In trust aggregation, [18] applies *static weighted sum* to compute centrality trust, direct service quality trust, and indirect service quality trust separately. In particular, for direct service quality trust assessment, transaction relevance (along with relationship factor and computational capability to a lesser extent) is used as the weight. For indirect service quality trust assessment, credibility is used as the weight. The difference between subjective trustworthiness and objective trustworthiness is how evidence is collected. For subjective trustworthiness, a node uses the relative centrality for centrality trust assessment, self-observations for direct service quality trust assessment, and feedbacks provided to its peers for indirect service quality trust assessment of another node. For objective trustworthiness, the network centrality is used for centrality trust assessment, and all feedbacks are used for both direct and indirect service quality trust assessment.

In trust update, the event-driven scheme is taken. For the distributed model, a service receiver rates the service quality of a transaction at the end of the transaction, stores the rating its local storage, and provides a feedback to its peers upon request. For the centralized model, the DHT collects the feedback and updates its trust database after the end of each transaction.

In trust formation, [18] is considered a multi-trust scheme by applying *static weighted sum* to combine centrality trust (which is a social trust property) with service quality trust (which is a QoS trust property obtained from direct service quality trust and indirect service quality trust) into an overall trust value.

For defending against attacks, [18] uses direct service quality trust assessment and feedback propagation for SPA, credibility rating for BMA and BSA, and differentiating long-term and short-term direct service quality trust assessment to change credibility to defend against OSA. However OOA is not considered.

4.3 Class 3: QoS / Centralized / dynamic weighted sum / Event-driven / Single-trust

[21] proposes a reputation system for a service-oriented IoT system and falls into this classification. In trust composition, it considers service quality as the sole trust metric but uses context information such as service type and node capability (e.g., energy status) to associate a service quality rating. In trust propagation, it uses a centralized manager to store all reputation reports (with context information) sent by individual SRs, after service is rendered. Upon receiving a new service request, the centralized manager selects SP candidates based on the service context for servicing the request. It then uses only evaluation reports with similar service context for reputation assessment of each SP candidate. In trust aggregation, the service context similarity between a stored report and the target service is computed by a global contextual distance function. A higher weight is used if a higher service context similarity is found. The reputation score is

then computed by *dynamic weighted sum* with the weight associated with a report corresponding to the *recommendation trust* of the recommender who supplies the report. The recommendation trust is updated dynamically based on if the recommender's report agrees or deviates from the majority of reports with a similar service context. Trust update is performed by the centralized manager via a learning process which presumably occurs whenever new reports are received, so it is based on event-driven. In trust formation, only service quality is considered.

[21] deals with SPA by detection of service quality and feedback propagation. The centralized manager rates a recommender dynamically based on the degree to which the recommender's report deviates from the majority reports. This can effectively defend against BMA and BSA, if the majority recommenders are not malicious. OSA and OOA are not considered.

4.4 Class 4: QoS / Distributed / Fuzzy logic + Static-weighted sum / Time-driven / Single-trust with static weighted sum

[3] falls into this classification. In trust composition, it considers direct QoS trust metrics such as end-to-end packet forwarding ratio (EPFR), energy consumption (AEC), and packet delivery ratio (PDR). In trust propagation, trust is propagated in a distributed manner. Each node in the network maintains a data forwarding information table of other nodes.

In trust aggregation, the overall trust of a node toward another node is aggregated using a static weighted sum of direct trust based on direct interaction experiences, and indirect trust based on recommendations. The direct trust is computed by aggregating EPFR, AEC and PDR direct interaction evidence using a static-weighted sum. If the aggregated trust value passes a threshold, the experience is a positive experience; otherwise, it is a negative experience. A fuzzy membership function taking into consideration of the number of positive and negative experiences together with uncertainty is used to compute the direct trust. The indirect trust on the other hand is computed by the product of the trustor's direct trust toward the recommender with the recommender's recommendation trust toward the trustee. The recommender's recommendation trust is computed in the same way as direct trust except that positive and negative recommendation experiences, together with uncertainty, are used in the fuzzy membership function definition. Since direct trust and recommendation trust are each defined by a fuzzy membership function, the overall trust can be aggregated by applying fuzzy logic "add" (for adding direct trust with indirect trust based on static weighted sum) and "multiply" (for multiplying direct trust with recommendation trust to obtain indirect trust) operators. In trust update, the local trust is updated periodically, so it is time-driven. In trust formation, only single-trust (service quality trust) is considered. No social trust is considered in this work.

SPA is detected in the protocol design and the feedback is propagated to the central manager through trust propagation. BMA, BSA, OSA and OOA are not considered.

4.5 Class 5: QoS + Social / Distributed / Static weighted sum / Event + Time-driven / Multi-trust

[1] [2] fall into this classification. In trust composition, it considers separate trust properties, including QoS trust properties such as honesty and cooperativeness, and social trust such as community-interest. In trust propagation, [1] [2] follow the distributed scheme where each node maintains its own trust assessment towards other nodes and propagates its recommendation trust toward other nodes.

In trust aggregation, a trustor node aggregates its current trust toward the trustee node with new evidence based on weighted sum. The new evidence can be either direct evidence if the trustor node directly encounters and interacts with the trustee node, or indirect evidence if it does not encounter the trustee node but receives a recommendation trust toward the trustee node from a recommender it encounters. In the latter case, the recommendation trust received is discounted by the trustor's direct trust toward the recommender. A novelty is that the weights associated with the past experience and the current evidence can be dynamically adjusted to tradeoff the trust convergence rate and trust fluctuation rate. Although the effect of weight parameters is analyzed, there is no discussion of how the weight parameters can be dynamically adjusted. Therefore they can be classified as static weighted sum at most.

In trust update, the trust management protocol is encounter-based as well as activity based, meaning that the trust value is updated upon an encounter event or an interaction activity. Two nodes encountering each other or involved in a direct interaction activity can directly observe each other and update their trust assessment. They also exchange their trust evaluation results toward other nodes as recommendations.

In trust formation, this paper considers multiple trust properties: honesty, cooperativeness and community-interest trust. However the issue of how to form the overall trust out of these separate trust properties is not discussed.

[1] [2] use honesty trust assessment and feedback propagation for defending against SPA, BMA and BSA. However, OSA and OOA are not considered.

[10] also falls into this classification. In trust composition, it considers both QoS trust (i.e., quality reputation and energy status) and social trust (i.e., social relationship using factors considered in [7, 18]). In trust propagation, this paper adopts a distributed scheme where each node maintains its own trust assessment towards other nodes. Evidence for each trust component is propagated separately. In trust aggregation, each trust component is assessed separately based on static weighted sum. In particular, quality reputation is a static weighted sum of direct trust and indirect trust. The indirect trust is also a feedback-trust-weighted sum of feedbacks received from all recommenders with each recommender's feedback trust being updated based on how the recommender's feedback deviates from the average. The social relationship trust and the energy trust are also each assessed by a static weight sum function. In trust update, it is event-driven with the trust value being computed based on transaction completion and timer events. In trust formation, it falls into the multi-trust category with the overall trust being formed from the three trust components, quality reputation, energy, and social relationship, based on static weighted sum.

5. GAP

In this section, we identify research gaps in trust computation for IoT systems as summarized in Table 2. We identify the most visited, least visited, and little visited trust computation methods.

A modern IoT system is inherently socially oriented since IoT devices are owned by humans which are connected by social networking. We see from Table 2 that most works indeed consider both QoS trust and social trust for trust composition. We take the view that a valid trust model for IoT must consider both QoS trust and social trust. Among social trust properties, similarity and friendship are the most visited among all. There is a need to explore other social metrics such as centrality, and community of interest, especially for rating a recommender.

Table 2: Most, Least and Little Visited IoT Trust Computation Models in the Literature.

Most visited	Trust composition	QoS trust [1][2][3][5][7][10][18][21]
		Social trust [1][2][5][7][10][18]
	Trust propagation	Distributed [1][2][3][5][7][10][18]
		Centralized [18][21]
	Trust aggregation	Static weighted sum [1][2][3][10][18]
		Bayesian inference[5][7]
	Trust update	Event-driven [1][2][5][7][10][18][21]
		Time-driven [1][2][3][5][7][10]
	Trust formation	Single-trust [3][5][7][21]
		Multi-trust with static weighted sum [1][2][10][18]
Least visited	Trust aggregation	Fuzzy logic [3]
		Dynamic weighted sum [5][7][21]
Little visited	Trust aggregation	Belief theory
		Regression analysis
	Trust formation	Multi-trust with dynamic weighted sum
		Multi-trust each with its own minimum threshold
		Multi-trust with trust scaled by confidence

Both centralized and distributed trust propagation methods are well studied. In general distributed trust propagation is for subjective trust computation without relying on a centralized entity. This argument is justified in an IoT system populated with millions or even billions of heterogeneous IoT devices. Centralize trust computation on the other hand is for objective reputation computation (common belief of the public) in IoT environments where access to a cloud is possible. We consider both distributed and centralized trust propagation methods valid.

In trust aggregation, static weighted sum and Bayesian inference are the two most visited methods, dynamic weighted sum and fuzzy logic are least visited, while belief theory and regression analysis have not been investigated in the literature. From Table 2, we can see some gaps in the least visited area and little visited area.

In trust update, both event-driven and time-driven are well studies. Event driven trust update is frequently performed when a service or transaction is completed, or when a node encounters another node, while time-driven trust update occurs periodically to preserve energy. Both approaches have their need.

In trust formation, there is a big gap from most visited methods in single-trust and multi-trust with static weighted sum, to little visited methods in multi-trust with dynamic weighted sum, multi-trust each with its own minimum threshold, or multi-trust with trust scaled by confidence. Multi- trust refers to the trust protocol that considers more than one trust properties, each being assessed separately and a trust formation method is applied to form the overall trust out of these multiple trust properties.

6. RESEARCH DIRECTION

There are several research directions for trust computation in service-oriented IoT systems.

The first and foremost is explore untouched trust aggregation techniques based on belief theory or regression analysis. Regression analysis especially is applicable when IoT nodes can access a centralized trust manager, say, located on a cloud because of the limited computing power of IoT devices to do efficient statistical analysis. Feedback data including service context information such as capability and energy of the SP, the traffic congestion condition of the network, and the service quality feedback itself can be propagated to the cloud for complex statistical analysis to better connect context information with service quality, and thus provide a more accuracy estimate of service quality trust of a SP in question.

The second research direction is to further explore innovative social trust metrics and the best way to combine them for IoT trust computation. In the literature, using social similarity to rate a trustee or a recommender is emerging [7]. However, how to combine several social metrics such as friendship, social contact, and community-of-interest, into social similarity is still an open problem. Further, other than similarity, there are also many IoT social properties such as centrality, selfishness, and cooperativeness and honesty that need to be further explored.

The third research direction is to devise and validate a trust computation model that can defend against all attacks. Existing works have considered ways to defend against self-promotion, bad-mouthing and ballot-stuffing attacks effectively but not opportunistic service and on-off attacks. Properly leveraging social trust may be an effective way to defend against bad-mouthing and ballot-stuffing attacks but this needs to be verified with real service-oriented social IoT systems.

There is a big gap in the area of trust formation when there are several distinct trust metrics and one wants to combine several trust metrics into one overall trust metric. The literature is thin in this area. Only [1][2][10][18] considered the use of static weighted sum for trust formation. The fourth research direction is to investigate the use of more effective trust formation methods including dynamic weighted sum, setting a minimum threshold for each trust property without combining multiple properties into one, and using one trust property as the main one which is scaled by other trust properties serving as confidence. In particular, dynamic weighted sum potentially can improve application performance when the weights associated multiple trust properties can be dynamically adjusted based on environment context information available at runtime. A potential technique to use is regression analysis [24] to link context information with trust accuracy and/or application performance so as to determine the best weight assignment. Another potential technique is adaptive filtering [7] to follow the principle that a node should have high trust toward IoT devices who have more positive user satisfaction experiences and, conversely, low trust toward those with more negative user satisfaction experiences.

The fifth research direction which is an open problem is to design a trust computation method that can scale. A modern IoT system comprises not just thousands but millions or even billions of heterogeneous IoT devices. For distributed trust propagation, a push-based periodic trust propagation method will not scale. A more scalable pull-based trust propagation method is called for. A possible technique is encounter-based trust propagation [8] such that trust information is exchanged only when IoT devices encounter with each other, so there is no extra traffic created or energy spent for forwarding recommendation packets. Also many IoT devices will be tiny so it is impractical for each IoT device to store the trust values of all other IoT devices in the system just for trust-based decision making. Therefore a scalable storage scheme based on heuristic design principles is called for. One possibility is to store trust information for nodes with the highest trust values and nodes recently interacted or encountered, as these nodes are most likely to share common interests [5]. Performance analysis of such scalable trust propagation and trust storage methods is of paramount importance.

The sixth and the final research direction is to integrate cloud service with trust management service, aka, trust as a service, for centralized trust management of an IoT community. The IoT community for example can be an e-health group paying particular attention to air pollution for the welfare of a group of users who may suffer from polluted air quality, an intelligent your-ride-on-demand IoT group (like uber), or a smart city group consisting of visitors, merchants, restaurants, and entertainment business entities, etc. Trust as a service would be a perfect service provided by the cloud to members in each of these groups. Service reputation feedbacks along with service context information can be fed into the cloud for a complex yet complete statistical analysis. Users requesting a service or a composite service (i.e., several services bundled together via service composition and binding) can be assured of high-quality service as a result of "trust as a service" being applied to such a service-oriented IoT group.

ACKNOWLEDGMENT

This material is based upon work supported in part by the U. S. military Research Laboratory and the U. S. military Research Office under contract number W911NF-12-1-0445.

REFERENCES

[1]. F. Bao and I.R. Chen, "Dynamic Trust Management for the Internet of Things Applications," *International Workshop on Self-Aware Internet of Things*, San Jose, USA, Sept., 2012.

[2]. F. Bao and I.R. Chen, "Trust Management for the Internet of Things and Its Application to Service Composition", *IEEE International Symposium on a World of Wireless, Mobile and Multimedia Networks*, San Francisco, USA, June 2012, pp. 1-6.

[3]. D. Chen, G. Chang, D. Sun, J. Li, J. Jia, and X. Wang "TRM-IoT: a trust management model based on fuzzy reputation for Internet of Things." *Computer Science and Information Systems*, vol. 8, no. 4, 2011, pp. 1207-1228.

[4] I.R. Chen, et al., "Dynamic Trust Management for Delay Tolerant Networks and its Aapplication to Secure Routing," *IEEE Trans. Parallel and Distributed Systems*, vol. 25, no. 5, 2014, pp. 1200-1210.

[5]. F. Bao, I.R. Chen and J. Guo, "Scalable, Adaptive and Survivable Trust Management for Community of Interest Based Internet of Things Systems", *11th International Symposium on Autonomous Decentralized System*, Mexico City, Mexico, March 2013

[6]. I.R. Chen, J. Guo, "Hierarchical Trust Management of Community of Interest Groups in Mobile Ad Hoc Networks", *Ad hoc Networks,* 2015.

[7]. I.R. Chen, J. Guo, F. Bao, "Trust Management for SOA-based IoT and Its Application to Service Composition", *IEEE Transactions on Service Computing*, 2015.

[8]. I.R. Chen, J. Guo, "Dynamic Hierarchical Trust Management of Mobile Groups and Its Application to Misbehaving Node Detection", *28th IEEE International Conference on. Advanced Information Networking and Applications*, Victoria, Canada, May 2014, pp. 1-6.

[9] I.R. Chen, F. Bao, M. Chang, J.H. Cho, "Trust management for encounter-based routing in delay tolerant networks," *Global Telecommunications Conference*, Miami, USA, 2010, pp. 1-6.

[10] Z. Chen, R. Ling, C.M. Huang and X. Zhu, "A scheme of access service recommendation for the Social Internet of Things," *International Journal of Communication Systems*, Feb. 2015.

[11] L.C. Freeman, Centrality on social networks, *Social Networks*, vol. 1, 1979, pp. 215– 239.

[12] S. Ganeriwal, L.K. Balzano, and M.B. Srivastava, "Reputation-Based Framework for High Integrity Sensor Networks," *ACM Trans. Sensor Networks*, vol. 4, no. 3, 2008, pp. 1-37.

[13] A. Jøsang, "A logic for uncertain probabilities", I*nternational Journal of Uncertainty, Fuzziness and Knowledge-Based System*s, vol. 9, no. 3, June 2001, pp. 279– 311.

[14]. A. Jøsang, R. Ismail, C. Boyd, "A survey of trust and reputation systems for online service provision", *Decision Support Systems*, 2007

[15] A. Jøsang et al., "The Beta Reputation System," *Proc. 15th Bled Electronic Commerce Conf.*, 2002, pp. 1-14.

[16] Liu Y, Chen Z, Xia F, Lv X, Bu F. "A trust model based on service classification in mobile services", *IEEE/ACM international conference on cyber, physical and social computing*, 2010, pp. 572–577.

[17] D.W. Manchala, "Trust metrics, models and protocols for electronic commerce transactions", *18th IEEE International Conference on Distributed Computing Systems*, 1998.

[18] M. Nitti, R. Girau, and L. Atzori, "Trustworthiness Management in the Social Internet of Things," *IEEE Transactions on Knowledge and Data Management*, vol. 26, no. 5, 2014, pp. 1253-1266.

[19] R. Roman, J. Zhou, and J. Lopez, "On the features and challenges of security and privacy in distributed Internet of things," Computer Networks, vol. 57, 2013, pp. 2266-2279.

[20] J. Sabater, C. Sierra, "REGRET: a reputation model for gregarious societies", *4th Int. Workshop on Deception, Fraud and Trust in Agent Societies*, Montreal, Canada, 2001, pp. 61– 70.

[21] Y.B. Saied, A. Olivereau, D. Zeghlache, and M. Laurent, "Trust management system design for the Internet of Things: A context-aware and multi-service approach," *Computers and Security*, 2014.

[22] S. Sicaria, A. Rizzardia, L.A. Griecob, and A. Coen-Porisini, "Security, privacy and trust in Internet of Things: The road ahead," Computer Networks, vol. 76, Jan. 2015, pp. 146-164.

[23] K. Wan, and V. Alagar, "Integrating Context-Awareness and Trustworthiness in IoT Descriptions," *IEEE International Conference on Internet of Things*, Aug. 2013, Peking, China, pp. 1168-1174.

[24] Y. Wang, Y.C. Lu, I.R. Chen, J.H. Cho, and A. Swami, "LogitTrust: A Logit Regression-based Trust Model for Mobile Ad Hoc Networks," 6th *ASE International Conference on Privacy, Security, Risk and Trust*, Boston, MA, Dec. 2014.

[25] Y. Wang, I.R. Chen, J.H. Cho, K.S. Chan and A. Swami, "Trust-based Service Composition and Binding for Tactical Networks with Multiple Objectives," *32th IEEE Military Communications Conference*, San Diego, CA, Nov. 2013.

[26] Z. Yan, P. Zhang, A.V. Vasilakos, "A survey on trust management for Internet of Things," *Journal of Network and Computer Applications*, vol. 42, june 2014, pp. 120-134.

[27] B. Yu, M.P. Singh, "An evidential model of distributed reputation management", *1st ACM Int. Joint Conference on Autonomous Agents and Multiagent Systems*, July 2002.