Future Generation Computer Systems 92 (2019) 758–776

Contents lists available at ScienceDirect

Future Generation Computer Systems

journal homepage: www.elsevier.com/locate/fgcs

Cooperative trust relaying and privacy preservation via edge-crowdsourcing in social Internet of Things

Vishal Sharma [a], Ilsun You [a,*], Dushantha Nalin K. Jayakody [b], Mohammed Atiquzzaman [c]

[a] Department of Information Security Engineering, Soonchunhyang University, Asan-si 31538, Republic of Korea
[b] Department of Software Engineering, Engineering School of Information Technologies and Robotics, National Research Tomsk Polytechnic University, Russia
[c] School of Computer Science, University of Oklahoma, Norman, OK, United States

HIGHLIGHTS

- Novel architecture for S-IoT by using edge-crowdsourcing.
- Formation of a light-weight distributed query system by using "Fission Computing".
- Low complex solutions for trust relaying and privacy preservation by using entropy modeling.
- Simulative and theoretical validation for trust and privacy preservation.
- Application case study for detecting false news sources in S-IoT.

ARTICLE INFO

Article history:
Received 15 June 2017
Received in revised form 31 October 2017
Accepted 24 December 2017
Available online 28 December 2017

Keywords:
IoT
S-IoT
Trust
Privacy
Edge-computing
Crowdsourcing

ABSTRACT

Social-Internet of Things (S-IoT) fortifies the relationship between computing entities for efficient utilization of resources. S-IoT accentuates on building convivial circles between the devices, which sanction sharing of information without compromising own performance. Trust and privacy are the two factors responsible for the maintenance of these social circles. Trust establishes faith between the entities and privacy supports abstraction of device information. There exist a plethora of approaches, which fixate on trust and privacy in social networks. However, the subsisting solutions rely on a trust scoring system and utilize a single centralized server for these calculations. Such scoring systems are manipulable by eavesdroppers, which engender erroneous reputation leading to high trust values. Considering the desideratum of an efficient trust and privacy-preserving solution, this paper proposes a novel solution in the form of fission computing. The proposed approach relies on the edge-crowd integration for maintenance of trust and preservation of privacy rules in S-IoT. The proposed solution uses crowdsources as mini-edge servers and entropy modeling for defining trust between the entities. Fission managers are responsible for maintaining the privacy rules, which operate for every S-IoT application. The proposed approach is analyzed through numerical simulations by using a safe network as a performance benchmark. Further, the article presents a case study on the detection of fake news sources in S-IoT as an application of the proposed approach.

1. Introduction

Ever since the evolution of "Internet of Things" (IoT), the focus has been on the connectivity between objects irrespective of the types of applications running on them. IoT is observing an all time increase in the number of entities contending for connectivity. Such an increase has raised the serious need for resource management and coordination between these devices. IoT supports daily life issues by providing a variety of applications for specific tasks and forms a crucial part of industrial development [1,2].

Another aspect of IoT is the grouping of devices for efficient resource management. Such grouping of IoT devices is studied under Social-IoT (S-IoT) [3,4]. Every object in S-IoT forms a group by establishing a social circle with the available devices. These circles use social relationships for resolving complex tasks that require mutual support, such as, location marking apps, equipment branding, data relaying networks, smart home management, smart grid systems, autonomous networking, traffic management, etc. [5,6].

The high probability of a relationship between the entities is the key to success in S-IoT. Trust and privacy are the two factors that

* Corresponding author.
E-mail addresses: vishal_sharma2012@hotmail.com (V. Sharma), isyou@sch.ac.kr (I. You), nalin.jayakody@ieee.org (D.N.K. Jayakody), atiq@ou.edu (M. Atiquzzaman).

https://doi.org/10.1016/j.future.2017.12.039
0167-739X/

govern these relationships. Efficient handling of these factors allows smooth operations in S-IoT [7,8]. Trust refers to the faith that an entity exhibits on the other entity for its operations, whereas privacy is the seclusion of the information during regular operations of devices. Privacy and trust are dependent on application frameworks as well as the configuration of devices participating in S-IoT.

Over the last few years, many researchers have focused on these two factors, such as TRM-IoT by Chen et al. [9], reputation systems by Truong et al. [10], mobility-based trust by Kantarci and Mouftah [11], etc. Most of them use a centralized server-based trust management system and application-driven privacy preservation. The approaches that use distributed solutions rely on a scoring system which is manipulable by eavesdroppers. One such solution includes a credential management system for trust maintenance [12]. This approach focuses on the level of assurance in IoT. Trust-based services are another example of such solutions. These services use fuzzy modeling for trust management [9]. Adaptive trust management is another variant for trust scoring in S-IoT [13]. Trust scoring by subjective and objective evaluations is a single server-based solution for trust management in S-IoT [14]. However, these solutions can fall prey to different types of known or unknown attacks in S-IoT.

The existing approaches which resolve trust issues do not focus on privacy conditions during data exchanges and vice versa. One such solution is privacy preserving framework for IoT [15]. This approach resolves privacy issues such as confidentiality, authentication, and intervenability, but there is less focus on trust management. The subsisting solutions, despite being novel and reliable, lack practical applications and case studies for highlighting their actual operations. To the best of authors' knowledge, there is a lack of an efficient solution which can support both the requirements of trust and privacy in S-IoT.

The types of architecture and infrastructure also play a vital role in trust management and privacy preservation [16,17]. Nowadays, the focus has been on the use of edge computing for the realization of connections between the devices involved in S-IoT. Edge Computing shifts the resources near a user-site for resolving communication overheads associated with multiple relationships in S-IoT [18,19]. Edge computing expands the current cloud architecture by providing a new layer of computational resources near a device making computing easier and efficient. But, deploying more edge servers near user-site increases the cost and complexity of the network. These issues are resolvable by using available IoT devices as mini-edge servers.

Crowdsourcing can be one of the efficient solutions for using IoT devices as min-edge servers. Crowdsourcing shifts tasks to the IoT devices while operating these devices as specialized servers [11]. Motivated by this, the proposed approach uses the properties of edge computing and crowdsourcing for cooperative trust relaying and privacy preservation in S-IoT.

1.1. Problem statement

S-IoT focuses on building social relationships between the IoT devices. S-IoT supports interaction on the basis of mutual coordination and social structuring between the devices. The user applications in S-IoT depend on the level of interaction and large-scale data analysis. The data analysis involves inspection of the activities of an individual who participates in the formation of S-IoT. The level of interaction indicates the strength of connections between the two entities in S-IoT. Both these are manageable by using efficient query processors on a centralized server. But, capturing the centralized node or coordinating server may illude a user.

There are different mechanisms for trust management, such as identity-based systems, social networking, and cooperative solutions. These provide trust by building a reputation system for IoT devices. These solutions, although efficient, yet do not meet the need of the privacy rules in S-IoT. Also, excessive dependency on centralized server leads to a high risk of attacks, such as Denial of Service (DoS), Distributed-DoS (DDoS), Sybil and man-in-middle attacks [20]. Such attacks are preventable by the maintenance of trust and preservation of privacy via distributed mechanisms. However, the distributed mechanisms suffer from the lack of inferential control, low-complex processing, scalability and various performance issues such as computational overheads, community formations, privacy preservation and trust modeling for non-reputation systems. All these factors raise a desideratum of an efficient solution, which is capable of managing trust between the IoT devices without compromising their privacy rules.

1.2. Our contributions and highlights

The work presented in this article aims at resolving issues related to the formation of trust and preservation of privacy rules in S-IoT. The proposed work eliminates the dependency on the centralized servers by forming a scalable and distributed system which uses end-user devices as mini-edge servers. This allows the formation of edge-enabled crowdsourcing for handling a large number of user queries and management of user policies. The proposed model overcomes the key issues of distributed trust management, trust-relaying, privacy preservation, and light-weight query system for a huge set of users in S-IoT. The highlights of the proposed work are:

- Novel architecture for S-IoT via edge-crowdsourcing.
- Formation of a light-weight distributed query system by exploiting the limited resources of mini-edge servers.
- Inclusion of "Fission Computing" as a new paradigm for distributed trust management.
- Low complex solution for trust relaying and privacy preservation by using entropy modeling.
- Simulative and theoretical validation for trust and privacy preservation.
- Application case study for detecting false news sources in S-IoT.

The rest of the paper is structured as follows: Section 2 presents a detailed related work. Section 3 presents environment modeling with details of network and system setup. Section 4 gives details of the proposed solution. It also presents the new paradigm, "Fission Computing", for trust and privacy preservation in S-IoT. Section 5 conducts performance evaluation by focusing on the theoretical analysis and the numerical simulations. Section 6 presents a case study on the detection of fake news sources by using the proposed approach. Finally, Section 7 concludes the paper.

2. Related works

The work in this paper focuses on trust and privacy by integrating crowdsourcing and edge computing for S-IoT. There are several types of research in the past that have focused on these aspects. This section presents some of these key works by dividing them into four main categories as shown in Fig. 1.

2.1. Trust in S-IoT

There are no strict parameters and no formal definition for computing trust of an entity in S-IoT. Trust is usually observed as a score that is mutually evaluated by different units active on the same platform [21].

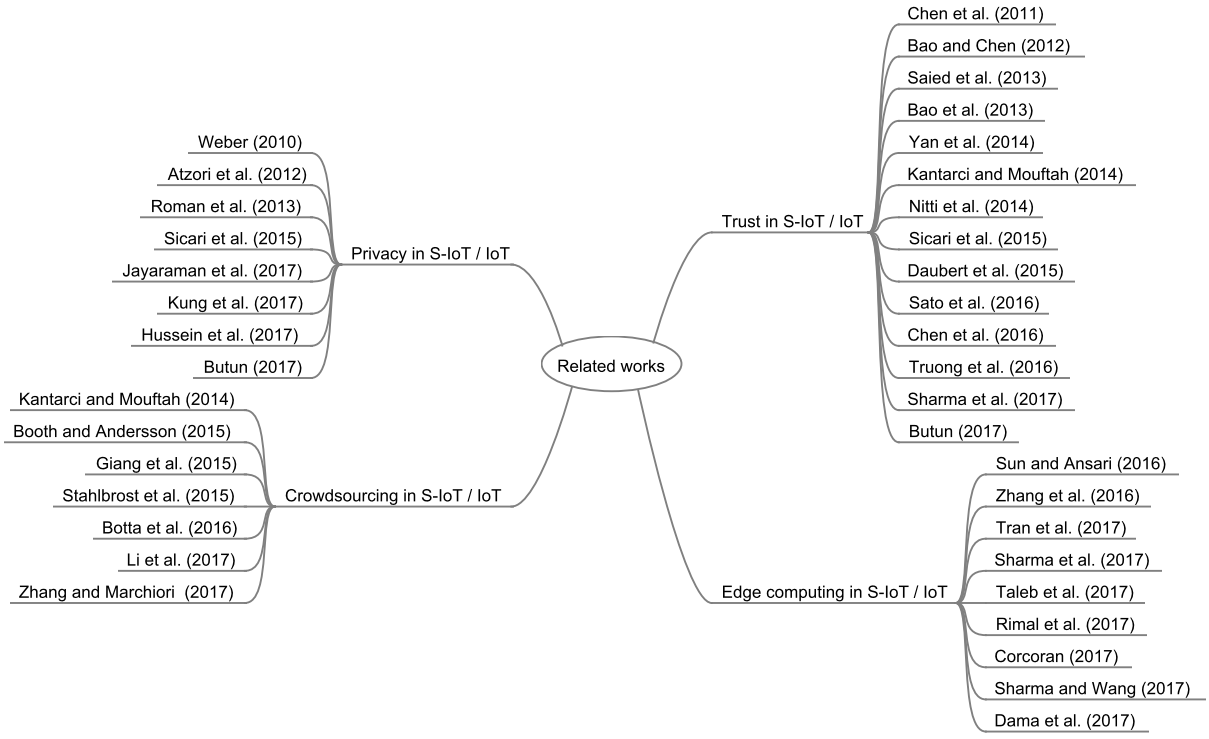

Fig. 1. An illustration of the related works studied in this paper.

Trust-based services can perform better than the non-trust-based services. Thus, provisioning of trust becomes utmost important for improving the performance as well as coordination between the devices in S-IoT [1,12]. There exist a plethora of solutions which have considered trust in all different forms but with the same goal.

Most of the trust-based solutions aim at balancing the connectivity between the service seekers and the service providers, which is of considerable significance for maintaining user control while defining the trust rules [22,23]. Dong et al. [9] developed a fuzzy inference system for defining the trust in IoT. The authors proposed a system which calculates trust reputation and conducted simulation studies for its validation. This is a route-based approach which is improvable by focusing more on the user side solutions rather than global evaluations.

Some important surveys, such as Sicari et al. [7], Yan et al. [8], provides in-depth details of trust issues in S-IoT. These surveys present four major trust solutions, namely, social networking based trust, fuzzy reputation systems, cooperative trust management, and identity-based solutions. Even though this classification is reasonable, various anomaly detection and behavior monitoring systems are also applicable. These systems can help in maintaining trust by identifying malicious nodes in the network [21,24].

Community-based trust formation is one of the integral parts of S-IoT. Such systems check the role of an individual amongst different entities. Bao et al. [25] used community-interest amongst S-IoT for maintaining trust. The authors provided the best possible settings for trust protocol that correlates the inter- and intra-community of interests. To increase its scalability, the authors extended their work in Chen et al. [13,26], which uses the trust for service management in S-IoT. Distributing the operations among the connected nodes can reduce the computational overheads of service-management. Such solutions can be observed in trust-based relaying in delay tolerant networks [27]. These solutions can be extended for enhancing the trust activities of the node depending on the conditions of the network.

2.2. Privacy in S-IoT

S-IoT is all about the communal connectivity between the IoT devices. A network with trust is mandatory for sustaining applications. But, trust alone cannot guarantee protected environment for all its users. The trustworthiness of S-IoT requires strict privacy-preserving mechanisms that can prevent the damage to social reputation and personal data of an individual [3,28].

Most of the applications in S-IoT depend on the mutual information provided by the users. The details of an individual play a supporting role in these networks and should be secretly handled. Access to core information, including the location, place, and time of an individual, should be limited [29–31].

Privacy and trust go hand in hand. A solution which aims at trust relationship should be secure enough to protect the user and device information [10]. Sicari et al. [7] developed a prototypical system for privacy preservation and trust management for IoT. The authors suggested a filtering mechanism for services that are demanded by the users. Since information sharing is more important in their work, it is mandatory to use authentication mechanisms and data-exchange protocols for its actual implementation.

Kung et al. [15] developed a privacy maintaining framework for IoT. The authors proposed an engineering aspect of IoT and focused on data controllers, data processors, and associated integrators for privacy-preservation. Although such inference resolves issues related to privacy-preservation, yet there is a need for control on social interaction and computational burdens. Jayaraman et al. [32] focused on privacy issues of IoT. The authors proposed a privacy-preserving IoT architecture. Their solution is efficient enough for maintaining privacy in large-scale IoT networks. But, its application to social setups while cross-platform validations is still an open issue and need further extensions. Similar to trust in distributed systems, approaches for vehicular networks can be considered for enhancing privacy in S-IoT [33].

Privacy is always an issue for the applications which depend much on the user involvement. Formation of abstracted systems

can provide a simple layer of security for S-IoT. However, there are limited works on abstraction systems for privacy-preservation, and the proposed work inclines to exploit such requisites for preserving privacy between social interactions of IoT users.

2.3. Crowdsourcing in S-IoT

Crowdsourcing enables the use of resources from the people by forming a user ad hoc network. The resources from the community help in resolving various computational tasks with low complexity [34,35]. Crowdsourcing is seen as a massive opportunity for IoT to provide social strength to its devices. There are various crowdsourcing solutions which provide a vast range of applications in IoT, but the majority of them have not focused on the social aspects of these devices. Although crowdsourcing is itself a social aspect involved in IoT, yet it requires separate evaluations [36].

Trust and privacy management in crowdsourcing are the two major issues which have a huge impact on S-IoT. Kantarci and Mouftah [11,37] considered the mobility-based trustworthiness amongst crowdsources in IoT networks. The authors developed a framework by exploiting "Sensing as a Service" in cloud-based IoT. This is a reputation-based system and may fall prey to the Sybil and the DDoS attacks [38].

Zhang and Marchiori [39] emphasized on reducing the dependency of IoT networks on the service gateways. The authors applied the concept of crowdsourcing for simplifying the IoT deployment and reducing the cost of the system. The authors demonstrated the utility of crowdsourcing in futuristic IoT networks.

Giang et al. [40] utilized crowdsourcing for exploiting the sensors and other equipment for setting up IoT network. The authors emphasized on a simpler extension of Internet-enabled wireless sensor networks to form IoT. However, trust evaluations and privacy in social aspects of sensors are still open issues with these solutions.

2.4. Edge computing in S-IoT

Solutions like fogging and edge computing provide a support for extending IoT as crowdsourcing networks [41,42]. Edge Computing aims at shifting the resources near a user-site for resolving communication overheads associated with the IoT devices. Edge computing expands the current cloud architecture by providing a new layer of data analysis near a device, which allows easier and efficient computing. Ranging from surveillance by remote objects to data collection, edge computing can efficiently handle the demands of data analyses as well as decision making [18]. Operations like search, tracking, and information retrieval from near user networks can be performed by edge servers with low latency [43,44].

Reduction in processing-latency and formation of smarter environments are achievable via traffic offloading by the means of edge servers [45,46]. Improving the quality of experience is one of the primary tasks in such scenarios. The available solutions exploit the features of edge computing for IoT networks but do not consider the trust formations and privacy-preservation in S-IoT.

Mobile edge computing can be adopted as an intelligent solution for integrating a large set of devices by defining governing rules for their connectivity [47]. However, the majority of industrial applications are hesitant about their full implementation especially targeting social aspects because of low-inferential support from the subsisting solutions.

Sharma and Wang [48] presented an approach for live computational assessment of IoT devices via edge computing. This solution lays a strong background in the formation of traceable IoT networks. However, the satisfaction of the performance issues regarding the social interpretation of IoT devices is yet to be covered in their solution. Dama et al. [49] suggested the futuristic cellular IoT networks. The authors emphasized the role of IoT in dense cellular networks. The solutions presented by the authors give a strong evidence for the use of edge-cloud in dense IoT-enabled cellular networks. Although edge computing offers a wide range of applicability and common solution to a vast set of problems targeting IoT, to the best of our knowledge, their involvement for social trust and privacy preservation via crowdsourcing is limited.

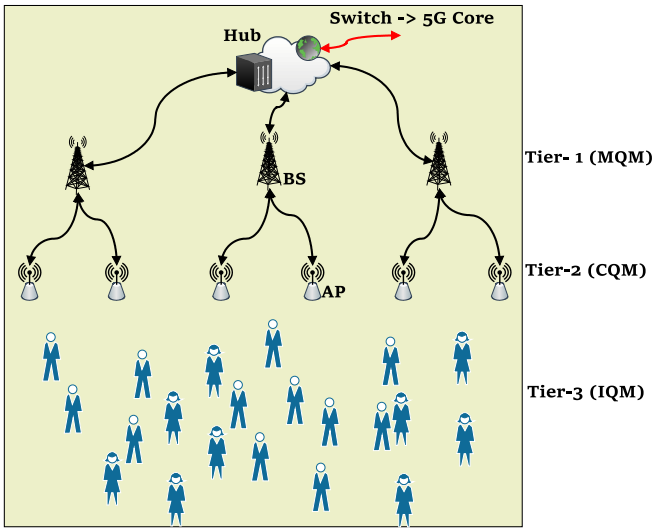

Fig. 2. An illustration of the network model comprising three major tiers of BS, AP, and users.

It is observed that there are a plethora of approaches, which focus on providing trustworthiness and privacy in cellular-IoT and S-IoT. Most of the approaches use a centralized server to calculate the trust on the basis of given set of parameters. These approaches also address the aspects of distributed trust management, but only to a limited level. To the best of our knowledge, none of the existing solutions has used the fission computing paradigm for trust and privacy preservation via edge-crowdsourcing. Considering this, a state-of-the-art comparison is presented in Table 1. The table compares the existing solutions with the proposed approach on the basis of ideology, parameters used, trust management, privacy preservation, focus, use of crowdsourcing and application aspects.

3. Environment model

The proposed approach uses an edge computing environment which is formed by utilizing crowd as mini-edge servers. The proposed approach can be treated similar to distributed solutions, such as trust and privacy in delay-tolerant networks, but with different methodology, application and concept. Each person active in the crowd is assumed to possess a handheld device (mini-servers) that can forward the traffic as well as perform computations along with its normal operations. These mini-servers allow the formation of an edge-crowdsourcing network. This section provides the details of the network model along with its system model, which is used by the proposed approach for trust relaying and privacy-preservation.

3.1. Network model

Edge-crowdsourcing is used for maintaining trust and privacy amongst the IoT devices. The network model considered in this paper is inspired by the switch–hub architecture as shown in Fig. 2. The network comprises sets N, B, and Z representing users, Base

Table 1
Comparison of the proposed approach with state-of-the-art solutions.

Approach	Author	Ideology	Parameters	Focus	Trust	Privacy	Crowdsourcing	Applications
Establishing trust in emerging IoT	Sato et al. [12]	Level of assurance in IoT	Confidence, Identity proofing, credential management	IoT	Yes	–	No	–
Trm-IoT	Chen et al. [9]	Fuzzy trust and reputation modeling	End to end packet forwarding, Convergence speed, Detection probability	IoT	Yes	–	No	–
Trust-based service management	Chenet al. [13]	Adaptive trust management	Trust values, trustworthiness score	S-IoT	Yes	–	No	–
Reputation and knowledge based trust	Truong et al. [10]	Fuzzy reputation system and functional architecture	Recommendation, Reputation, Knowledge scores	S-IoT	Yes	–	No	–
Privacy engineering framework	Kung et al. [15]	Privacy policies and objectives	Confidentiality, Intervenability, Authentication	IoT	–	Yes	No	Organization normative framework
Mobility-aware trust	Kantarci and Mouftah [11]	Cloud centric-IoT	Average utility, platform utility, total payment to users	IoT	Yes	–	Yes	–
Trust management in IoT	Saied et al. [23]	Cognitive trust management	Trust level, Level of interaction	IoT	Yes	–	No	–
Trust in S-IoT	Nitti et al. [14]	Subjective/Objective trustworthiness	Success rate, Transaction number, Trustworthiness	S-IoT	Yes	–	No	–
Trust evaluation in S-IoT	Truong et al. [50]	Modeling by using trust indicators (TIs) Reputation, Experience and Knowledge	Reputation ranking, Experience	S-IoT	Yes	–	No	User recruitment model
Proposed Approach	Sharma et al.	Trust and privacy via edge-crowdsourcing (Fission Computing)	Entropy, Integration Cost, Availability	S-IoT	Yes	Yes	Yes	Fake news detection

Stations (BSs) and the Access Points (APs), respectively. The network uses multiple query manager systems, namely, Main Query Manager (MQM), Crowd Query Manager (CQM), and Individual Query Manager (IQM) for handling the traffic and information flow across the network. MQM is operable at BS, CQM is operable at AP, and each handheld device is assumed to possess an IQM. The distribution of the query management system helps in managing the operations of the edge computing.[1]

The proposed network model does not rely on a central reputation system as such systems are hazardous to trust violations and can be manipulated by Sybil attacks. The centralized reputation system operates over a particular set of users, and on the basis of defined parameters, it finds the trust score, whereas the proposed model is capable of overcoming the issues related to centralized reputation system by providing on-demand trust evaluations. The trust is calculated whenever a new link is active or when some new applications request for the intended information.

3.2. IQM–CQM–MQM interaction

The interaction between the key components (IQM–CQM–MQM) is maintained on the basis of availability of the users and the crowd size C_s. A threshold is fixed for each of these entities for on-demand operations without depending on any triggering mechanism.

This can be understood with the help of a simple parameter, θ, which is defined as the population size required by each query manager. Let θ be split into $\theta_1, \theta_2, \theta_3$ for IQM, CQM, MQM, respectively. Here, θ can be calculated as the product of the number of trust requests per user r and the total number of users n in each zone, such that:

$$\theta = r \times n, \ \sum_{A} (n) = |N|. \tag{1}$$

where A is the total area, and θ decides whether the entire network will be used as a centralized solution or as a distributed solution allowing multiple possibilities for decision in the case of threats or trust-violations. In general, for edge-crowdsourcing, $\theta_1 < \theta_2 < \theta_3$, and $\theta_1 = \theta_2 = \theta_3$ converts the entire system into a centralized solution allowing every computation to be performed at MQM. Such scenarios may be useful in the case of sparse networks as the central node can alone manage the entire load. For edge computing, without utilizing the user resources, $\theta_1 = \theta_2 < \theta_3$.

Currently, C_s is governed only by the population size and the range of a query management system. Also, the proposed approach mainly relies on the number of users and the requests shared between the entities in the edge-crowdsourcing model. However, C_s can be varied depending on the number of parameters which are subject to the role of user and active operations in the network. The details of properties and parameters that can be used for calculating C_s are shown in Table 2.

Table 2
C_s parameters.

Parameter	Value
Number of trust requests	$r, \sum r = R$
Number of users	n
Number of queries	q
Number of connections per user	C_p
Population size	$\Theta = n \times \mathrm{r}$

[1] IQM, CQM, MQM are also referred as Individual Fission Manager, Crowd Fission Manager, and Main Fission Manager, respectively (used in later parts).

The choice of C_s depends on the level of complexity for evaluating trust in S-IoT. In the proposed model, population size is considered as the main component of C_s and the other metrics are used for finding other properties that are explained in following subsections.

3.3. System model

Formation of a trustworthy network in S-IoT is based on various aspects, such as the crowd model, edge model, user movement, trust model and the privacy validation. This section defines all these models individually and formulates the entire setup into multiple optimization problems.

3.3.1. User movement

The user model is the key to success for efficient trust relaying in S-IoT. Since the integration of crowdsourcing and edge computing is possible only by the accurate identification of user movement, a location-based social mobility model is selected for it. The problem for user movement is derived from the periodic and social mobility model of Cho et al. [51]. This model is operated towards the number of times a user is active and the difference in the number of connections supported by it.

This model helps to identify the probability for the existence of a link between any two entities $P[u_1, u_2]$ in the network [51], which is expressed as

$$P[u_1, u_2] = \sum_{R} |\tau_{u_1,u_2}|^{-\beta_1} . \|G_r\|^{-\beta_2}, \tag{2}$$

where u_1 and u_2 are the users, τ_{u_1,u_2} is the time difference between the two users since activation, G_r is the gap between the requested links and the available links, and R is the total requests generated by the u_1 in a particular time stamp. The model will be reversed if the u_2 makes a request to u_1.

In actual model [51], β_1 and β_2 are parameters for the training the fitting model, but here, both these denote the characteristic constants for time difference and the resource difference. The value of each is considered on the basis of probabilistic range for the existence of a connection between the nodes prior to a request for new connection. The sum of these entities is equal to 1 if there is a connection as well as a data exchange between the users prior to the new request, and 0.05 if there is a connection but no exchange of information. Currently, these values are assumed as per the network model, but these can be fixed by determining the other properties of the network.

Now, assuming that the users move according to some pattern, the availability of the user in a network can be given as a rate of depletion of resources (links) or moving out of the user from the network. Believing that a user can sustain connectivity up to k meters, the overall availability of a user u_1 w.r.t. u_2 is given as:

$$A_v = \frac{A_u}{d_r}, \tag{3}$$

where d_r is the depletion rate of the links [52] given by:

$$d_r[u_1, u_2] = r_0 e^{-\lambda|\tau_{u_1,u_2}|}. \tag{4}$$

Here, r_0 is the initial links active between the users, and $\frac{1}{\lambda}$ is the lifetime which is calculated as mean resource consumption per connection. Also,

$$A_u = P[u_1, u_2]\left(\sqrt{D^2 + k^2}\right)^{-\frac{1}{2}(\beta_1+\beta_2)}, \tag{5}$$

where D is the distance between the user and the nearest AP. The rate at which the users move across the network can be obtained by the above-mentioned formulations. It is to be noted that individual velocity model is not used for user movement, rather relative velocity is used as it is of much importance in crowdsourcing.

3.3.2. Urban crowd model

The crowd model is derived in continuation of the user movement, which provides A_v for every user. The initial steps involve relating availability of every user with each other, i.e. Eq. (3) provides two values for every link between the nodes of the network. These two values for each link may or may not be same. In general cases, a mean value can be used to determine a single value for each link, but for more realistic mapping, the max–min approach is used to derive a single weight. The weight for each link between the users is given as:

$$W = \begin{cases} mean\,(A_v(u_1), A_v(u_2)) & t_s = 1 \\ \max\,(\min\,(A_v(u_1)[i])\,,\, \min\,(A_v(u_2)[i])) & t_s > 1 \\ A_{v,0} & otherwise \end{cases} \tag{6}$$

where t_s is the time stamp equivalent to τ and it denotes the number of entries available for A_v. $A_{v,0}$ is the initial value for availability and i varies from 1 to t_s.

The unification of weight helps in generating a simpler graph for the entire set of users. Since the crowd in urban setup follows a topology governed by the pedestrian ways, roads, etc., the similar concept is used while modeling the urban crowd for S-IoT. The urban crowd model helps to identify the number of subgroups formed in the entire network. It also helps in determining the leaving and joining rate of a user. The motion of each user is governed by the Hughes [53] model, according to which, the flow of users f_p across an area with a unit width is given as:

$$f_p = \rho \times \mathbb{S}, \tag{7}$$

where ρ is the density of users, $\mathbb{S}$ is the average speed given as

$$\mathbb{S} = \eta_1 - \rho\,(\eta_2)\,. \tag{8}$$

Here, η_1 and η_2 are the controlling constants for speed and density, respectively. Now, considering a unit variation of places for all the users in the network, the governing equation [53] for the users in urban crowd is given as:

$$-\frac{\delta\rho}{\delta t} + \frac{\delta}{\delta x}\left(\rho\,\mathbb{S}^2\,\frac{\delta\omega}{\delta x}\right) + \frac{\delta}{\delta y}\left(\rho\,\mathbb{S}^2\,\frac{\delta\omega}{\delta y}\right) = 0, \tag{9}$$

where $\frac{\delta\omega}{\delta x}$ and $\frac{\delta\omega}{\delta y}$ denote the change of position ω w.r.t x and y in a standard frame of reference, respectively.

The crowd model is subject to the formation of multiple subgroups that operate with a different value for the governing equation. Users with $\mathbb{S} = 0$ are stationary, while the others may have a high or low speed. Also, this variation may be a result of a difference in the density of each subgroup.

Considering the given area and density of users, the desired system can be managed for the flow of data and selection of controller by the application of mixture distribution [54,55], which aims at the selection of random variable for multiple subgroups. Each of the selected group has different properties, but their cumulative distribution is considered normal. This helps to replicate the exact scenario of crowd movement in S-IoT.

Now, the mixture distribution is formulated by selecting the weights for each user by following Eq. (6). Let m be the number of subgroups with different values for governing equation. Now, the density function for the overall network can be given as:

$$f_m = \sum_{i=1}^{m} W_i P_i\,(\rho)\,, \tag{10}$$

where $P_i\,(\rho)$ is the probability density function. Also, the crowd size can be represented as the mixture density distribution, i.e. $C_s = f_m$. The average weight W_i for each subgroup with a given density is normalized on scale of 0 to 1 by following the mean weight for each link. Now, n users correspond to $f_m\,(u)$, where u

denotes the random user to be selected as the controller for the given mixture density distribution. The value for u is selected for all the subgroups by identifying a user with availability higher than the mean availability of the network/subgroup and minimum deviation from its peak value. Considering γ_j and W_j as the mean and weight for jth user, respectively, the overall mean is expressed as [54,55],

$$\gamma = \sum_{j=1}^{n} W_j \gamma_j, \tag{11}$$

and total variance $\mathbb{V}_t^2$ is expressed as [54]:

$$\mathbb{V}_t^2 = \sum_{j=1}^{n} W_j \left(\left(\gamma_j - \gamma\right)^2 + \mathbb{V}_j^2 \right), \tag{12}$$

where

$$\mathbb{V}_j^2 = \frac{1}{t_s} \sum_{k=1}^{t_s} \left(W_{i,k} - \overline{W}\right)^2. \tag{13}$$

Here, $\overline{W}$ is the average weight of the network during the entire session. Following up with the conditions given earlier, a user which obeys following rule is selected as the controller:

$$mean(A_v) \geq \frac{1}{n} (\gamma), \tag{14}$$

and

$$\mathbb{V} \leq \frac{1}{n} (\mathbb{V}_t). \tag{15}$$

If there exists a case, that two or more nodes follow similar trends for mean and variance, then a node with lower value for d_r is selected as the controller.

3.3.3. Edge-Crowd integration for trust modeling

From the above models, there exists a controller in every zone which is a part of MQM–CQM–IQM network formation allowing on-demand trust relaying in S-IoT. From Eqs. (14) and (15), let U_m (=u) be the user that is selected as a controller in a particular zone. Now, the edge-crowd integration model for trust relaying is derived w.r.t. MQM–CQM-U_m, which can be extended to the entire zones of other selected controllers.

The edge-crowd integration is performed by utilizing a fitness value that derives an integration cost I_C. The initial ranking of all the individuals, which serve as IQM controller, is obtained from I_C. These help in the direct application of the proposed solution to different set of applications, such as, location updates for S-IoT users, feedback system and even in the detection of the fake news and its sources.

I_C is defined by the number of parameters and is initially evaluated through MQM–CQM-U_m. Every individual's I_C is predicted by the U_m whenever an individual tries to fetch the results for some query via U_m. Also, the model allows sub-query system formation by generating I_C for all the links. Alternatively, U_m disseminates trust to every link it is connected. For example, if a U_m covers a zone with x number of links between its individuals, and the integration cost for it in combination with MQM and CQM is $I_{C,i}$, then, this cost is disseminated to every link by a trivial policy of weight assignment such that $\sum_{i=1}^{x} W \times I_{C,i} = I_{C,i}$. The values for weight are obtained from the previously generated values for every link in Eq. (6).

Memory and energy are considered as two major driving factors for edge-crowd integration. Currently, issues like channel interference, spectrum management, and realistic scheduling are not considered in the proposed solution. Let $\mathbb{M}$ and $\mathbb{E}$ be the memory and energy utilization, respectively, each having a threshold given by $\mathbb{M}^{TH}$ and $\mathbb{E}^{TH}$. The memory consumption for an individual in the network is given as:

$$\mathbb{M} = \sum_{i=1}^{g} \sum_{k=1}^{h} m_{ij}, \tag{16}$$

where h is the number of applications active on links x. Every controller can calculate the total memory consumption for every device on the basis of allocated connections and keep a check on the availability of a device in its zone. The energy consumption is calculated as:

$$\mathbb{E} = \mathbb{P} \times T_s, \tag{17}$$

where

$$\mathbb{P} = \mathbb{P}_c + \mathbb{P}_m. \tag{18}$$

Here, $\mathbb{P}_c$ is the power consumption for each CPU cycle and $\mathbb{P}_m$ is the power consumption for each memory operation. Now,

$$I_c = I_{c,1} + I_{c,2}, \tag{19}$$

where

$$I_{c,1} = \alpha_1 \|\mathbb{M}\| + \alpha_2 \|\mathbb{E}\|, \tag{20}$$

and

$$I_{c,2} = \frac{\alpha_3}{\|A_v\|}. \tag{21}$$

Here, α is the speed constant which is divided into α_1, α_2, α_3 depending on the order of preference for the defined parameters such that, by using [53],

$$\alpha_1 = \mathbb{R}_s \left(\frac{\mathbb{E}_L}{\mathbb{E} - \mathbb{E}^L} \right) \frac{\left(\mathbb{E} - \mathbb{E}^{TH}\right)}{\mathbb{E}}, \tag{22}$$

$$\alpha_2 = \mathbb{R}_s \left(\frac{\mathbb{M}_L}{\mathbb{M} - \mathbb{M}^L} \right) \frac{\left(\mathbb{M} - \mathbb{M}^{TH}\right)}{\mathbb{M}}, \tag{23}$$

$$\alpha_3 = \mathbb{R}_s \left(\frac{A_{v,L}}{A_v - A_{v,L}} \right) \frac{\left(A_v - A_v^{TH}\right)}{A_v}, \tag{24}$$

where $\mathbb{R}_s$ is the relative speed of an individual w.r.t. CQM, $\mathbb{E}_L$, $\mathbb{M}_L$, $A_{v,L}$ are the limiting values for energy, memory and availability, respectively, below which the system cannot perform. The equations are altered for avoiding the formation of a complex system, which otherwise require conversion into polar coordinates to generate actual values. Usually, for maintaining a stable network, threshold and limiting values are kept as close as possible to avoid the network from entering the dead state. The network compliances for the convergence towards a stable state such that Eq. (19) is minimized, i.e. $I_c = \min\left(I_{c,1} + I_{c,2}\right)$.

3.4. Privacy-rules

S-IoT aims at supporting different IoT applications by providing a mechanism for information sharing as well as relaying. The network model considered in this paper also focuses on the similar pattern of data provisioning, but relies heavily on the crowd sources. The rules for privacy preservation [56], which forms the basis of the proposed work, are as follows:

- User profile protection: The approach for trust and privacy should ensure that information about the individuals involved in relaying should not be disclosed under any circumstances. From the considered network point of view, U_m should ensure that no identity of an individual as well as

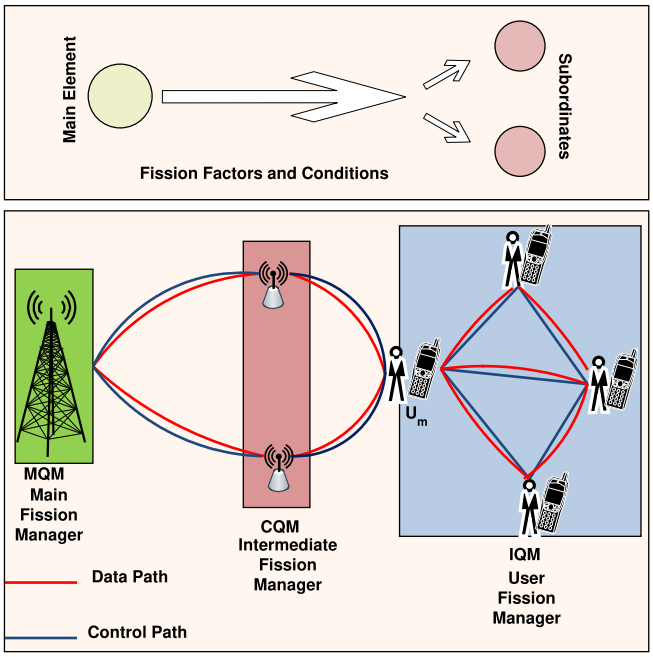

Fig. 3. An illustration of fission process and fission computing with its application in the proposed network. The fission managers are installed on MQM, CQM, and IQM which form a part of light-weight query system for trust relaying and privacy preservation.

the information shared with them be revealed to any unauthorized individual. Further, the communication between the MQM–CQM and the U_m should be private and the node which is selected as the controller should not disclose its identity. This can be simplified as no user in the crowd is aware of the controller, not even the controller itself.

- Data flow: Since, the role of the controller is of much importance, the amount of data to be shared without its knowledge should be limited. This prevents exploitation of resources as well as leakage of information of the controller. This can be attained by forming more subgroups with more controllers, as this further allows load balancing.
- Device confidence and authorization: With IoT being turned into a social network, it becomes utmost important for the applications to gain device confidence. This allows different applications in S-IoT with access to device information, such as memory, energy, and CPU utilization, for trust relaying. The applications should be light and should not cause excessive burden on the regular operations of the device. In the considered network model, it is assumed that each device has the proposed application loaded with access to all the required features. However, in practical scenarios, the choice of enabling such access is up to an individual and his/her confidence in a particular application will play a vital role.
- Device authentication: All the above rules of privacy are bundled together via device authentication. The network model used in this paper relays between the MQM, CQM, and IQM. Thus, the authentication policies should be strong enough to ensure the privacy of the users during edge-crowdsourcing.

The threat model in the defined network focuses on preventing the information from going into a wrong device and avoiding any eavesdropping. It is to be noted that with a large number of connections and flow of data, intruders get multiple points of entry to acquire information. Such scenarios are needed to be taken care of while deploying the proposed model.

4. Proposed solution

The proposed approach is capable of relaying trustfully between the users in S-IoT. The proposed approach also aims at privacy preservation for preventing any eavesdropping on user profile and information. The proposed approach uses crowd-resources instead of causing all the applications to maintain trust via the centralized server and it is an on-demand trust relaying system which is based on the maximum availability, minimum depletion rate, minimum integration cost and high probability of connectivity between the users. As discusses earlier, all these procedures are driven by the selection of a temporary controller.

The proposed approach provides a light-weight query processing system and uses fission computing as a new paradigm for interaction between MQM, CQM, and IQM as shown in Fig. 3. The concept of fission computing allows trust relaying via crowdsourcing as well as help modeling the authentication system for privacy preservation over the crowd model.

4.1. Fission computing

Fission computing is inspired by the nuclear physics [57], which refers to the splitting the core into multiple subparts keeping intact the properties of all the parts. Similar to this property, the proposed approach divides the queries into multiple subsets allowing the entire network to handle trust as a load balancing solution. The definition, concept, factors, and approach for fission-based trust relaying are explained below:

4.1.1. Definition

Let $\mathbb{F}$ be the fission set comprising elements that form the basis of the network. Fission set is divided into multiple subgroups, which helps in easy management of the system as well as its resources. In the proposed approach, $\mathbb{F}$ is equal to the set consisting of MQM, CQM and IQM, i.e. $\mathbb{F} = \{N \cup B \cup Z\}$. The division of fission set into subsets is carried such that the sum of resources across all the newly generated sets is same and does not change on further division. Here, resources are derived from C_s and are equally divided into subsets.

4.1.2. Concept

Fission computing is proposed as a new paradigm for distributed trust management and relaying. The concept of fission computing is provided below:

- The first step is the identification of entity which can subdivide its operations into two or more subordinates.
- The second step is the identification of properties on the basis of which the resources can be divided onto its subordinates. It is to be noted that steps 1 and 2 can be performed interchangeably depending on the applications. In some scenarios, properties can be identified at first and then subordinates with these properties can be selected, while in other scenarios, properties can be selected after identifying the subordinates.
- The third step is the setting of fission factors for managing the fission computing. These factors are the decision rules that coordinate with each other for better management. In the proposed approach, the fission factors form the basis of fission manager which drives the light-weight query system for interaction between the divided components (N, B, Z).

4.1.3. factors

Fission factors can vary depending on the choice and need of an application. The proposed approach uses entropy-based fission

rules for defining the concept of fission computing. Every entity and parameter set in the environmental modeling are subjected to entropy modeling [58,59], which provides a strong base for optimization while defining trust over the links between the crowdsources. The fission factors in the proposed approach are as follows:

$$E_0 = -\sum_{g} P[u_1, u_2] \log P[u_1, u_2], \tag{25}$$

$$E_1 = -\sum_{n} A_v \log A_v, \tag{26}$$

$$E_2 = -\sum_{A} \frac{1}{f_p} \log \frac{1}{f_p}, \tag{27}$$

$$E_3 = -\sum_{A} C_s \log C_s, \tag{28}$$

$$E_4 = -\sum_{n} \frac{1}{d_r} \log \frac{1}{d_r}, \tag{29}$$

$$E_5 = -\sum_{g} \frac{1}{I_c} \log \frac{1}{I_c}. \tag{30}$$

The combined optimization conditions for the fission factors are as follows:

$$E_{c,1} = \min\left(E_0 + E_5\right), \tag{31}$$

$$E_{c,2} = \min\left(E_1 + E_4\right), \tag{32}$$

$$E_{c,3} = \min\left(E_2 + E_3\right). \tag{33}$$

Now, for optimal solution, the conditional entropy for all the fission factors should also be minimized [59]. These entropies are defined by considering the alternative factors that cause huge impact over the other factor. These include

$$C_{e,1}\left(E_{c,2}|E_5\right) = \min\left(\sum_{g,\,A} P\left(E_5, E_{c,2}\right) \log \frac{E_5}{P\left(E_5, E_{c,2}\right)}\right) \tag{34}$$

and

$$C_{e,2}\left(E_{c,3}|E_5\right) = \min\left(\sum_{g,\,A} P\left(E_5, E_{c,3}\right) \log \frac{E_5}{P\left(E_5, E_{c,3}\right)}\right). \tag{35}$$

4.2. Fission-manager for light-weight query system

The proposed approach utilizes the concept of dividing query managers by relying on fission computing. The fission manager is the background application of query managers. The fission manager forms the light-weight query system that allows easy updates and information flow across the network entities. The fission manager operates on the basis of threshold values fixed at the MQM and shared with the CQM and the IQM. These threshold values can be updated at any instance depending on the complexity of network and type of application. Also, these values can be set differently for every application. However, the link-based applications use the same set of values. An illustration of the functional view of the proposed fission manager that helps attaining distributed trust management via fission computing is presented in Fig. 4. The components and their utilities in fission manager are explained below:

- MQM_ID: This is the ID of the main BS which controls the entire zone comprising multiple APs and users. This field is unique for every BS.
- MQM_RANGE: This field stores the range of BS which provides information about the number of subordinates controlled by it.
- MQM_ZONE: This field provides the information of current application zone. This information is used to store the subareas which are shared with other BSs.
- CQM_ID: This is the ID of the AP which interacts with the BS.
- C_s ID: This is the ID allocated to each workgroup represented as crowd size.
- CQM_LIST: This is the list of all the APs which are available under a particular MQM. This list stores the actual IDs and new IDs for each CQM that are installed on AP. Further, it also stores the fission factors that maintain the trust by keeping the record of entropy for every component by following the Eqs. (25)–(33). It is to be noted that at this point of contact, combined entropies are not evaluated.
- IQM_LIST: This is the list of every individual active in the zone of CQM who wants some information to be used for their corresponding applications. The IQM_LIST for CQM operates similarly to the MQM whereas IQM_LIST for IQM contains combined entropies which are used for identifying the minimum entropy paths between the requesting entity and the crowd servers.
- TRUST: This field stores the information whether the available entities are trustworthy or not. This is done on the choice of minimum entropy by matching it against the threshold values that are set by the MQM.
- CONTROLLER: This field stores the controller_ID. The controller_ID is known only to the CQM, but no other node in the network is aware of the actual ID of the controller and corresponds to it only as a normal network entity. However, the ID-based communication is subjected to different metadata depending on the application properties.
- ACTIVE: This field suggests whether the trust full entities are active or not.
- INFO_EXCH_MODULE: This part forms the lightweight information exchange module which is installed on all the fission managers. This is helpful in corresponding with every manager via its parameters by passing them as an argument in the communication functions.

An illustration of information exchange procedures with trust formation and privacy preservation by using fission principles is presented in Fig. 5. These procedures show the interaction rules between the network entities after the initialization of values in the fission manager. These steps determine the trusted parties which can be selected for trust-relaying and privacy maintenance. The explanations of the steps in the presented diagram are as follows:

- The initial step starts with registration between the IQM, CQM, and MQM. This step also follows allocation of new IDs to the subordinates allowing protection of actual IDs. All the procedures in these processes are carried through new IDs.
- Next step starts with the initialization of the application, which requires resources and information that is available with the crowdsources. This is followed by setting of the metadata. The metadata is shared with the controller via secured path allowing the controller to know the requirements of application active on the requesting IQM.
- The above step is followed by the peer-request to the controller from the source IQM. The controller checks with the similar procedures throughout the crowd and makes a list of available crowdsources. Then this list is shared with the requesting IQM.
- After this, requesting IQM identifies other IQMs, which have the same application installed on them and marks them in its fission manager. At the same time, the controller registers the application metadata with the CQM.
- Following the above steps, fission-entropy factors are evaluated at every available entity in the network. During these

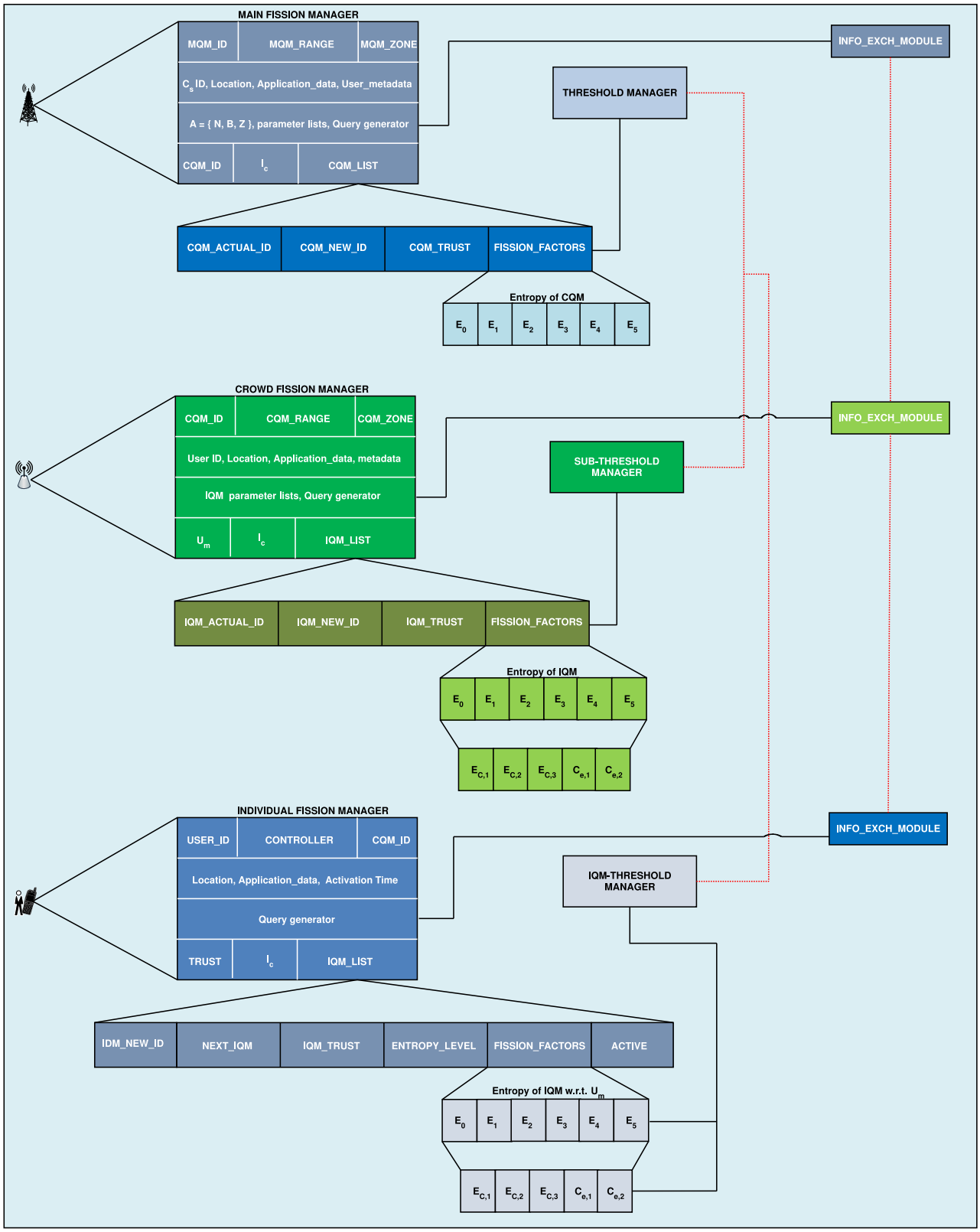

Fig. 4. An illustration of the functional view of fission manager for light-weight query system in the proposed approach for efficient trust-relaying.

steps, the threshold values are also distributed across the crowd entities as well as the intermediate subordinates.

- After calculating fission factors, calculated values for entropy are evaluated against the threshold limits and a decision is taken on the trust of each entity.
- Next, the trusted parties are selected for determining privacy rules and this involves the application of distributed fission managers for privacy preservation in S-IoT. The privacy rules are ensured between the controllers, CQM and MQM.

- Finally, the IDs and metadata are validated for every entity which is labeled as a trusted party before beginning the procedures for trust relaying.

Algorithm 1 Fission Algorithm

```
Input: N, B, Z
Output: Fission components
set I_c conditions
select controller
i=1
while i ≤ |𝔽| do
    Select entities with I_c=min
    Update controller
    i=i+1
end while
```

Algorithm 2 Validation procedure at source application

```
Input: N, B, Z, U_m
Output: Validation
mark the controller
i=1
while i ≤ |𝔽| do
    receive ID, metadata from IQM
    receive ID, metadata from U_m
    if (Info_match == true) then
        Validate
    else
        Reset and revalidate
    end if
    i=i+1
end while
```

Algorithm 3 Path selection and trust relaying

```
Input: N, B, Z, U_m
Output: Trust Path
mark the controller
i=1
receive thresholds
while i ≤ |𝔽| do
    Calculate E_0 to E_5, E_c,1 to E_c,3
    Calculate C_e,1 and C_e,2
    if (Threshold conditions are satisfied) then
        mark i=trust
    else
        unmark i
    end if
    path=marked(i)
    select shortest path=min(E)
    i=i+1
end while
```

The above-given steps are general and followed for every application aiming at S-IoT networks via edge-crowdsourcing. The fission manager can be customized on the basis of the requirement of the application. The parameter choice and access control can be included with the default parameters. However, both these factors affect the trust properties and the privacy rules of the network and require customization to match the scenario.

4.3. Trust relaying and privacy decisions with fission computing

After defining the fission manager, factors, and sharing procedures, the proposed approach uses trust relaying and privacy decisions for information exchange between the crowd entities. The trust relaying and privacy decision for edge-crowdsourcing are taken on a trivial rule of minimum entropy. However, depending on the level of complexity and affordable computations, the approaches can exploit fission computing by using decision systems, optimization theory or nature-inspired solutions. An illustration of trust relaying and privacy preservation for application support to a particular node via controller is shown in Fig. 6. The figure shows the procedure followed for relaying path selection on the basis of minimum entropy ensuring that trust is maintained throughout the operations. The proposed approach operates in three phases:

- Phase - I: Step 1 is initiated as request_info depending on the application which is activated on the requesting node. Next, step 2 is started, which includes calculation of self-entropies. These steps are repeated for all the active applications and continued until every entity performs self-evaluation for entropies. Also, the controller evaluates the entropy of every link which is active between it and the other IQMs. The application user evaluates the entropy conditions for its direct peers and waits for the updates from the controller.
- Phase-II: This phase includes steps 3 and 4. Step 3 is based on the update_info. This sends information to the application requesting node from the controller about the number of available devices and resources. Also, in response, the controller sets the thresholds as per the information from CQM and MQM. This forms the step 4 of the phase-II. Next, the application node sends its requirements to the controller and checks for the trusted path. Every user evaluates its direct contacts for entropy and calculates I_c. The controller also finds the combined cost for every entropy by considering every active link.
- Phase-III: Step 5 forms an integral part of Phase-III. This step identifies the minimum entropy routes in relation to the defined system model for every component that interacts with the controller. Next, the route information is shared with the application requesting node. Following this, the application node starts privacy preservation by matching the metadata for every entity. Finally, the minimum entropy path is selected for utilizing the resources of the edge-crowd model. It is to be noted that application which is specific to the requirement of a particular data, does not account minimum entropy, rather it obtains data via any trust-worthy path irrespective of the distance.

These steps are iterated every time there are new queries or there is a violation of trust and privacy rules. Once initiated, metadata and application information are updated at regular intervals for every active link in the network. The steps for fission computing, entity validation for application entity, and trust relaying are given in Algorithms 1, 2, and 3, respectively. Algorithm 1 identifies the entities which will participate in trust relaying on the basis of integration cost for their links with the controller. This is the simpler algorithm that is run in the initial phase with its complexity depending on the number of entities involved in the network. Algorithm 2 provides the validation procedures for ensuring that every entity which participates in the trust relaying obeys the privacy rules throughout the connectivity. This algorithm relies on the metadata received from the IQM for every connected entity. The validation is performed by matching the metadata with the available information of every device. At any instance, if the validation fails, metadata and information gathering procedures are repeated until a final decision is not taken on the success or failure of validation. Finally, Algorithm 3 gives the details of path evaluation for selecting minimum entropy-based trust relaying between the requested application and the available crowdsource. This algorithm is based on the threshold

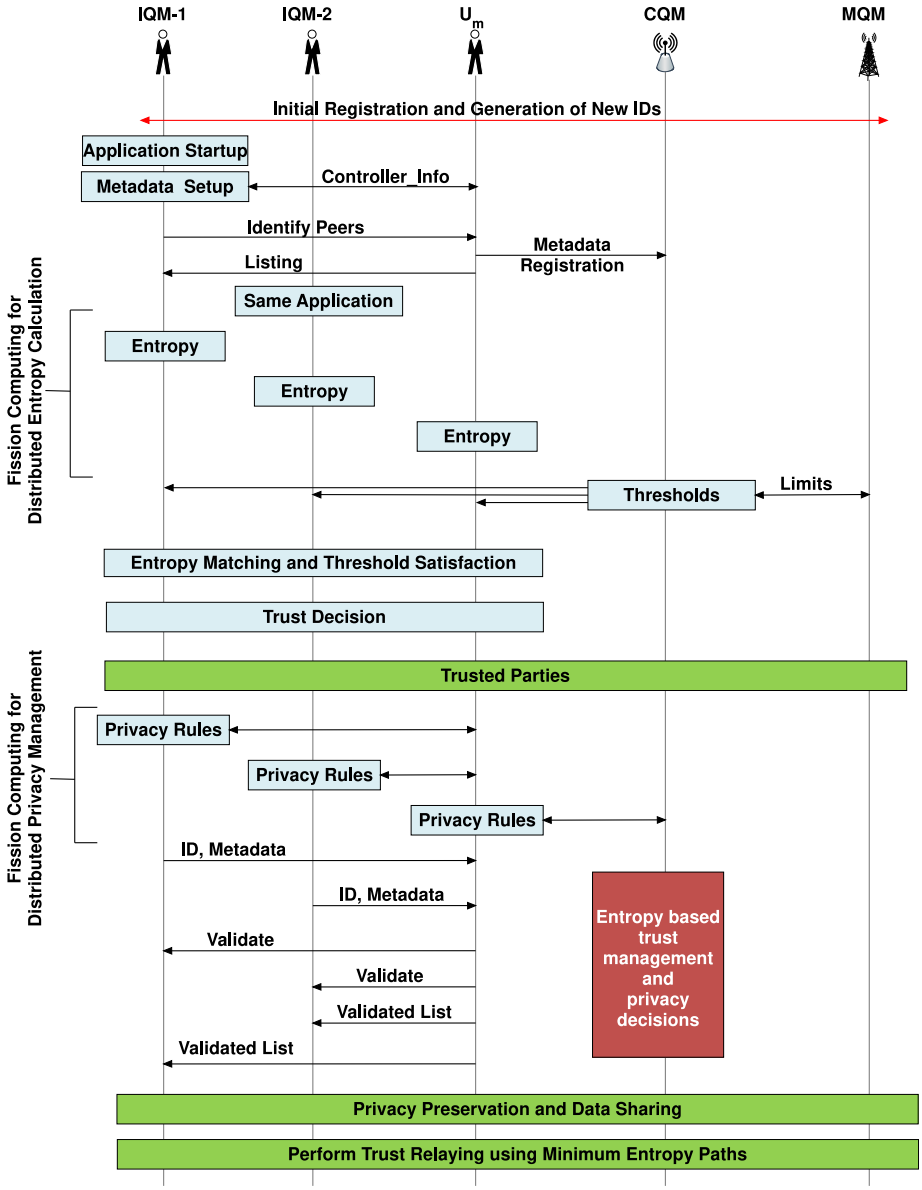

Fig. 5. An illustration of information exchange procedures with trust formation and privacy preservation between multiple users as a part of crowd model by using fission principles.

conditions and uses entropies as the decisive metrics for selecting the safe and appropriate path. These algorithms are iterative and operate whenever a new application request some support from the crowdsources.

5. Performance evaluation

The proposed approach relies on crowdsources for the majority of its operations, except for obtaining the thresholds and ID-based privacy. The proposed approach is evaluated from multiple aspects. Firstly, the theoretical analysis is presented for the environment model. Secondly, the model is verified for its privacy operations and metadata authentication. Thirdly, numerical simulations are performed to analyze its performance.

5.1. Theoretical analysis

This section presents the theoretical aspects of the proposed approach for privacy preservation and flow equations:

Lemma 1. *The privacy of application data and requesting node is maintained w.r.t. a controller as well as other crowdsources.*

Proof. The proposed approach relies on thresholds which are generated by the MQM (BS) and shared with the CQM (AP), which is responsible for disseminating it with the controllers and other entities by using different IDs as agreed during the initialization of the network. The application running on every device is unaware of the actual ID used for corresponding as all the exchanges are handled by the fission manager. The fission manager is abstracted from the application with a support for limited access to its fields. This allows application data as well as node information to remain private throughout the communication. Also, the controller is selected from the crowdsources which manages the flow between the IQMs, but the device acting as the controller is unaware of its status and operates as if it is a normal IQM. This allows privacy by non-disclosure of real identity even to the running applications. However, this success depends on the level of abstraction of the proposed fission manager. □

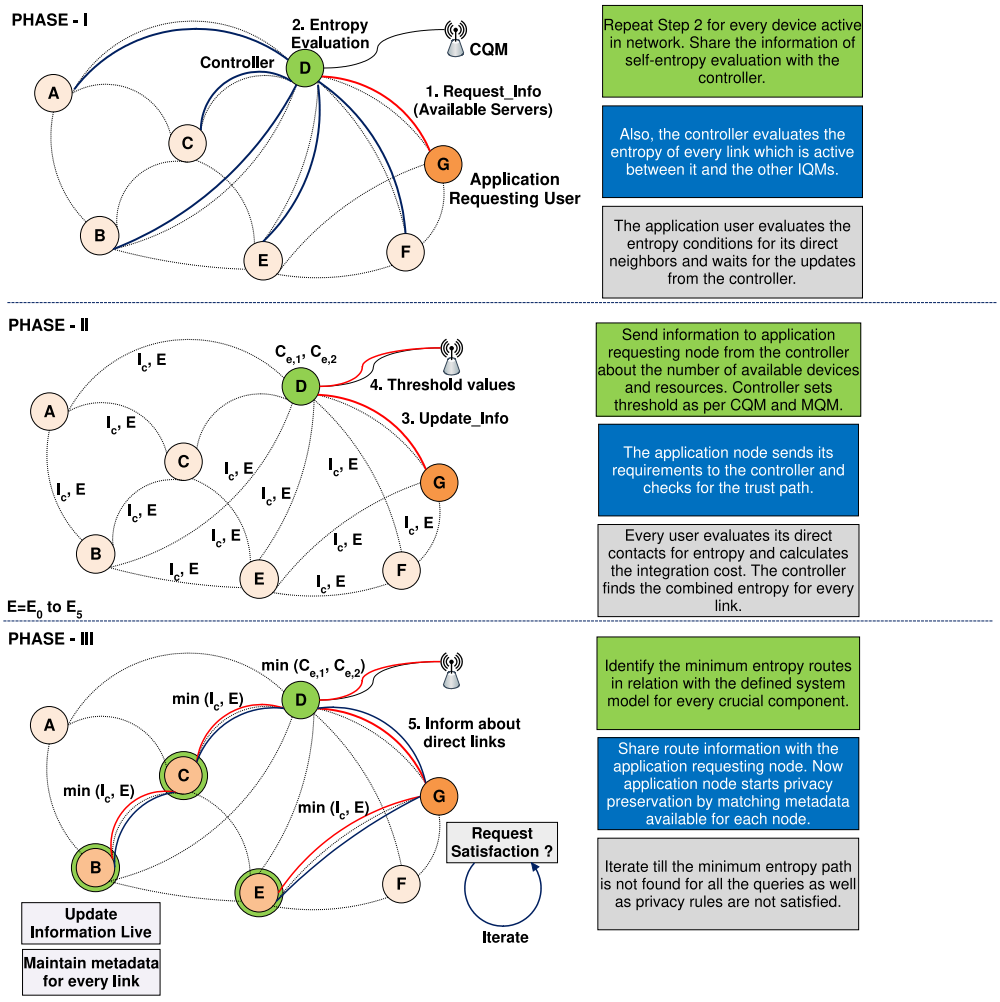

Fig. 6. An illustration of trust relaying and privacy preservation for application support to a particular node via the controller. The figure shows the procedure followed for relaying path selection on the basis of minimum entropy ensuring that trust is maintained throughout the operations.

Lemma 2. *The proposed approach protects against eavesdropping and flow control over resources. It also helps in identification of sources if provided full access to fission managers.*

Proof. The proposed approach relies on MQM–CQM-U_m for its controlled operations. Since the actual ID is never floated in the network, an eavesdropper cannot identify the source of information seeker. Also, the initialization of network begins with registration policies which index the fission manager of every device with the CQM allowing generation of new IDs in lieu of access to control information of a device. CQM can alter the ID at any instance to the controller (which itself is abstracted), allowing protection against eavesdropper. This is further enhanced by dividing the flow of packets across the crowdsources. Since the type of traffic and requirements will be different for every crowd link, no two devices follow similar context, making it difficult to interpret the actual requirements of an application. This helps in non-disclosure of the type of application requesting resource support in edge-crowd integration. □

Lemma 3. *Trust values can be secured and crowdsources can be prevented from knowing the entropy outcomes and thresholds.*

Proof. The proposed approach uses multiple aspects of entropy modeling to decide on the trust between any two active links. The capturing of entropy thresholds can allow Sybil attacks which can cause an intruder to gain false reputation. This can be prevented by allowing fission managers the authority to validate at the CQM or the controller only. This refers to non-sharing of thresholds with the requesting node and taking a decision on a particular entity which is assumed to be safe. However, such scenario may affect the distributed nature of the proposed solution. □

Lemma 4. *Constraints on user flow are also applicable to traffic and flow governing equations can regulate the traffic in edge-crowd integration.*

Proof. Eqs. (7) to (9) can be modeled to regulate the flow of traffic. From these equations, the packet flow can be given as [53]:

$$f_\delta = \varphi - \zeta\ \varrho_\delta, \tag{36}$$

where φ is the arrival rate defined as a number of requests per second, ζ is the number of links formed per second, and ϱ_δ is the number of requests per link. Now, considering that common discomfort function for the traffic flow is unity irrespective of the traffic conditions, the traffic governing equation is given as [53]:

$$-\frac{\delta \varrho_\delta}{\delta t} + \frac{\delta}{\delta x}\left(\varrho_\delta f_\delta^2 \frac{\delta \omega}{\delta x}\right) + \frac{\delta}{\delta y}\left(\varrho_\delta f_\delta^2 \frac{\delta \omega}{\delta y}\right) = 0, \tag{37}$$

and the driving factor is given as:

$$f_\delta\ \varrho_\delta \geq 1. \tag{38}$$

Table 3
Parameter configurations.

Parameter	Value	Description
A	10 000 × 10 000 sq. m.	Simulation area
$\|N\|$	1000–10 000	Users (n)
$\|B\|$	1	Base stations
$\|Z\|$	5	APs
r	5	Number of trust requests
T_s	3600 s	Running time
q	$2 \times n$	Number of queries
β_1	0.03	Characteristic constants
β_2	0.02	Characteristic constants
D	10–50 m	Distance between users
k	20–100 m	Distance from AP
g	10	Maximum links per node
h	2	Number of fission applications
$\frac{1}{\lambda}$	r/g	Mean life time
ρ	0.5 m^{-2}	Density of users
η_1	1.4 m/s	Speed constant [53]
η_2	0.25 m^3/s	Density constant [53]
$\mathbb{S}$	0.5–1.2 m/s	Speed
m	1024 bytes	Memory per link
$\mathbb{P}$	200 W	Power consumption
$\mathbb{E}^{TH}$	2000 J	Energy thresholds
$\mathbb{E}_L$	1500 J	Energy limiting value
$\mathbb{M}^{TH}$	512 bytes	Memory thresholds
$\mathbb{M}_L$	128 bytes	Memory limiting value
A_v^{TH}	max(A_v)-mean(A_v)	A_v thresholds
$A_{v,L}$	mean(A_v)	A_v limiting value

The flow conditions are applicable to the mobility of users, and thus, the constraints on one can be the controlling factor for the other. □

Lemma 5. *At a constant rate of users and traffic, the performance of the system is unaffected.*

Proof. From Eqs. (9) and (37), if the mobility of user is 0, i.e. $S = 0$, there will be no flow of users. However, this can also happen in a scenario where the leaving and joining rate of users in a zone of CQM is equal. This makes the relative speed 0 because of joining and leaving the same group. In such scenarios, speed modeling gets affected, but traffic flow remains usual, and the governing equation for crowd deduces to:

$$\frac{\delta \rho}{\delta t} = 0, \tag{39}$$

whereas the traffic model is unaffected. Thus, it can be noticed that density of users has much impact on the performance of the system, and traffic conditions are limited to the impact of relative speed. This proves that the system which is static from outside is still performing at the similar rate as observed for varying densities. □

5.2. Numerical simulations

This section evaluates the performance of the proposed approach over the synthetic dataset by following numerical simulations. The synthetic dataset comprised 10 000 users each following a pedestrian route with a speed varying between 0.5 m/s and 1.2 m/s. These users are served by a single BS and five APs. It is assumed that each user is having a processing device equipped with the proposed fission manager with configurations given in Table 3. The baseline is considered by performing trust relaying for all the users active and supportive of fission computing.

The results are obtained for every parameter and selected outcomes are presented by varying the users between 1000 and 10 000. The entropy thresholds are set at the average of all the values available for active links between the crowdsources. An illustration of the dataset with a sample size of 50 withdrawn from the synthetic dataset is shown in Fig. 7. The figure illustrates the probability of connectivity for a single link with its integration cost and the relative speed with the other crowdsources. The users with no resources or extremely high speed possess 0 value for every parameter and are not used in the evaluation.

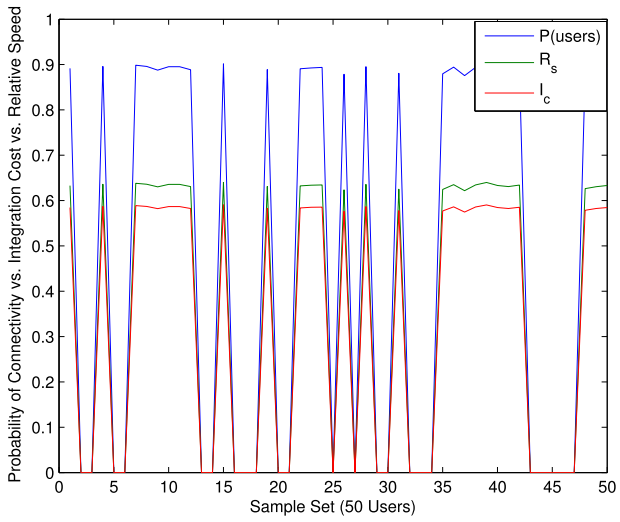

Fig. 7. Probability of connectivity vs. Integration Cost vs. Relative Speed for a sample set of 50 users drawn from an area with 10 000 users.

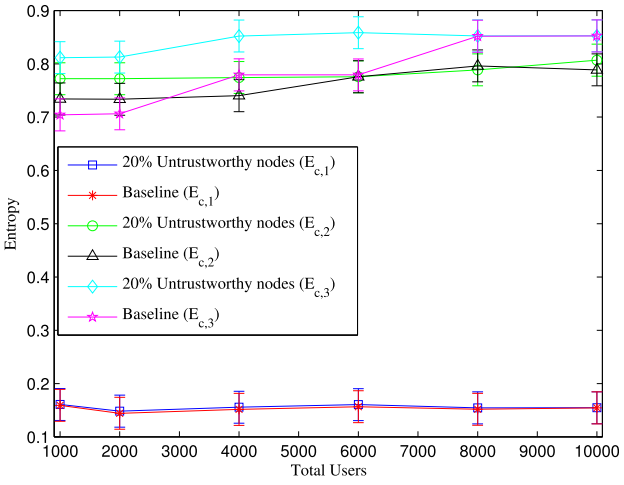

Fig. 8. Entropy ($E_{c,1}$, $E_{c,2}$, $E_{c,3}$) variation vs. total users for baseline and 20% untrustworthy users.

Firstly, considering the role of entropy in trust relaying, values of optimization conditions in Eqs. (31) to (33) are analyzed for a scenario with 20% untrustworthy users in comparison with the baseline. For these optimization conditions, the number of untrustworthy users does affect the performance of the system by increasing the entropy of every link. However, the increase in these values is negligible and varies between 1.0% and 2.5%, as shown in Fig. 8. Further, with a limited increase in these values, it can be identified that the proposed approach can operate successfully despite the presence of non-supportive nodes in the crowdsources. This small variation in entropy is the result of crowd-side evaluations, which otherwise would have been higher if each value is calculated for links with the CQM or MQM.

Similar to entropy optimization equations, the other driving factors in the proposed approach are the combined entropies. The combined entropies are defined w.r.t. I_c which decides whether a

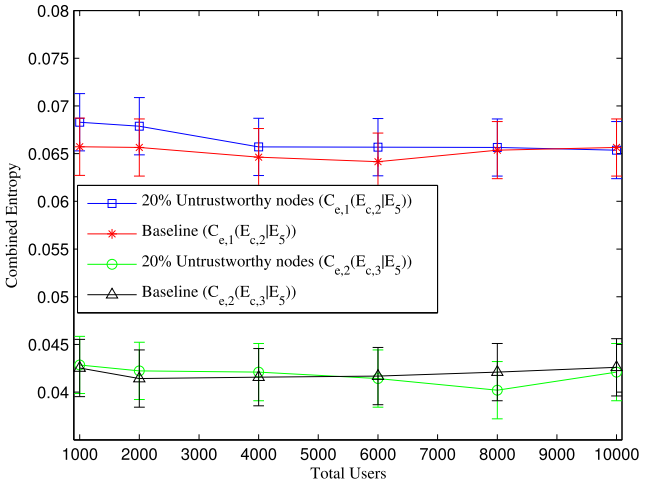

Fig. 9. Combined entropy (Ce, 1, $C_{e,2}$) variation vs. total users for baseline and 20% untrustworthy users.

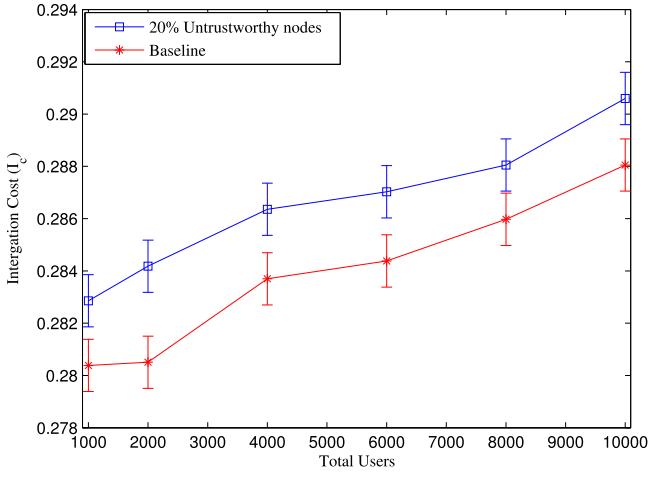

Fig. 10. Integration Cost (I_c) vs. total users for baseline and 20% untrustworthy users.

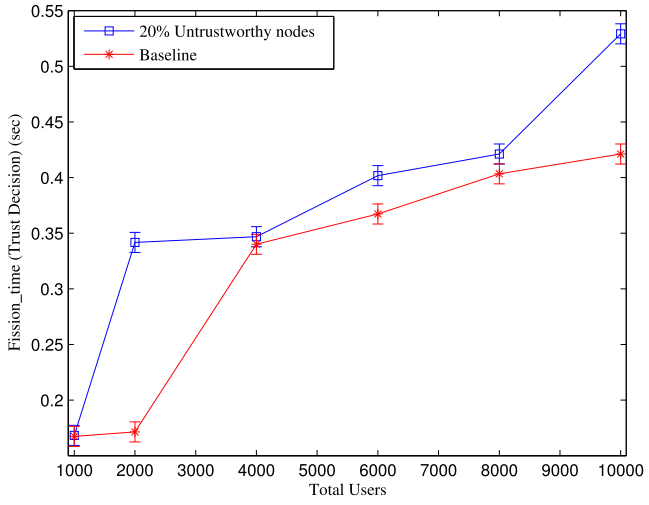

Fig. 11. Fission time (sec) vs. total users for baseline and 20% untrustworthy users.

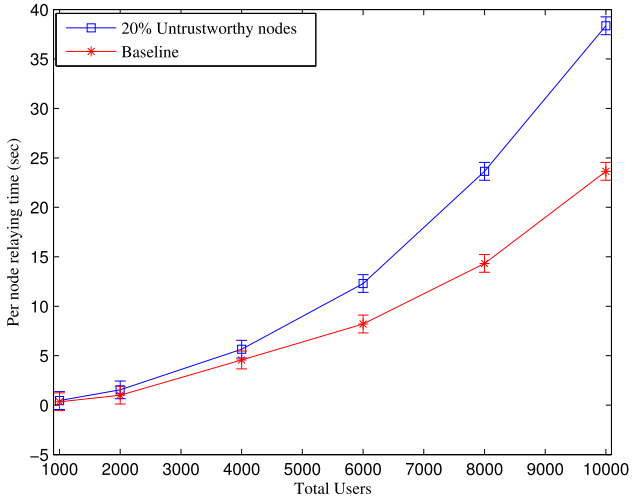

Fig. 12. Per node relaying time (sec) vs. total users for baseline and 20% untrustworthy users.

path is trustworthy or not. Also, it helps in identification of trustworthy nodes by considering other metrics such as the probability of connectivity, number of resources, relative speed, decay rate and flow of users. The results show that the proposed approach with 20% untrustworthy users varied from baseline for combined entropy in the range between 0.4% and 1.8% as shown in Fig. 9. This variation is outcome of the negligible change in the per-link entropy for every parameter modeled in Eqs. (25) to (30).

I_c is the most crucial factor for smooth operation of the proposed approach. It decides whether a route is feasible by forming the base for entropy conditions. A network with a lower value performs better and can lead to multiple halts if its value reaches the critical level. The critical level can be defined as the value of I_c above which the network is no use to a particular application. The minimum variation of I_c is 0.8% whereas, in extreme conditions, this variation goes up to 10% as shown in Fig. 10. Lower variation in I_c shows better performance, whereas an extreme increase may result in higher delays, which may affect the resource dependency of crowdsources.

The proposed approach relies on crowdsources for supporting applications by forming an edge-crowd network which relies on three major entities, namely, MQM, CQM, and IQM. As described in proposed part, these entities are driven by a fission manager, which is installed on every component of the network. The values of every parameter vary from application to application. In order to numerically analyze the fission time, which includes sharing of control information and settings of the device, standard values are used for untrustworthy as well as baseline scenarios. The results in Fig. 11 illustrate that the proposed approach consumes 15.3% more fission time when operated in 20% untrustworthy scenario compared to baseline.

Fission time influences the per-node relaying time, which is the time taken by a node to fix its values once encountered with an untrustworthy crowdsource. It includes reauthentication as well as revalidation of active links with the controller and the CQM. Results in Fig. 12 show that the proposed approach consumes 36.4% more relaying time in untrustworthy scenario compared to baseline. However, despite the excess time, the proposed approach is capable of maintaining trust as well as preserves the privacy rules, thus, making it an effective solution for edge-crowd based trust relaying.

6. Case study: prevention of fake news in S-IoT

The proposed approach relies on updating its fission manager whenever the context, as well as the metadata, is mismatched

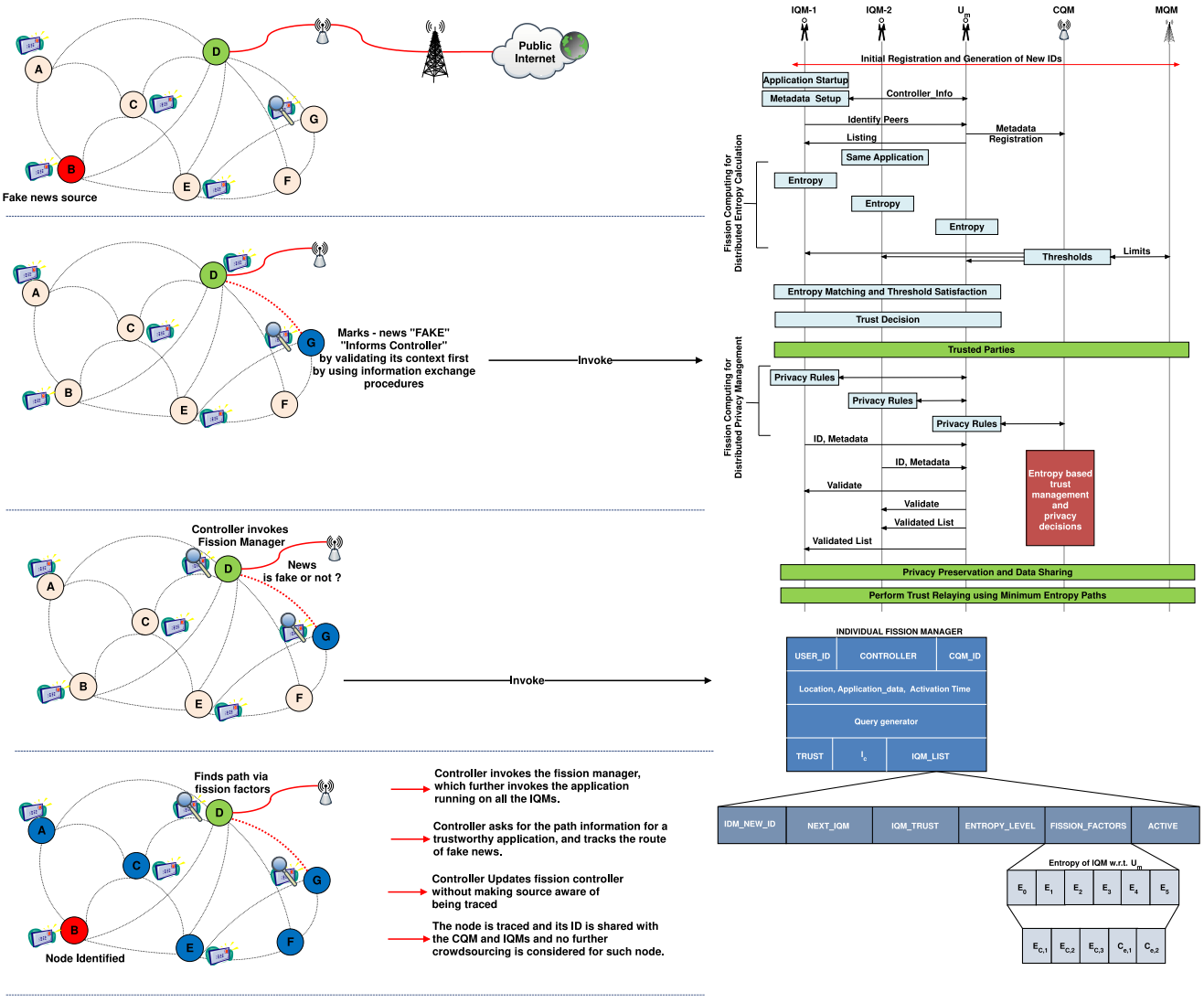

Fig. 13. An illustration of the fake news prevention in S-IoT by using the proposed fission computing paradigm for trust and privacy preservation via edge-crowdsourcing.

between any two entities. The proposed approach regulates and monitors the performance and resources of the edge-crowdsources via U_m, which is a temporary controller. As described in the system modeling, a network zone can have one or more controller depending on the requirement of application and complexity of the network. Here, complexity refers to the size and resources which are to be managed during the fission process. The proposed fission manager supports trust-relaying which allows backtracking the sources that circulate wrong and misleading information in the network. One such application of the proposed approach is the prevention of fake news once it is identified by any crowdsource. The proposed approach not only privately informs the other IQMs but also identifies the IQM responsible for such information. The proposed approach can be used to detect any misleading node in the network once it is marked suspicious by any user in the network. Such systems can help preventing false reviews for location, prevention of misleading content for tourists as well as the spread of old and unwanted news materials.

A representative illustration of fake news detection in S-IoT is shown in Fig. 13. The figure shows the tracking of the path by the controller once a node shares information regarding misleading content in the network. The controller first verifies the news from its public sources and ensures that there is no eavesdropping. Once ensured, the controller invokes the fission manager of every IQM in the network and carries revalidation procedures through the similar application on which the news is spread. The fission manager for that particular application reassesses the content and uses information exchange procedures, which help the controller in backtracking the path of originated news. Once these procedures are finalized, the fission manager marks the source as a misleading IQM and spreads ID across different IQMs using the same application. Also, to ensure trust in the network, the controller reacquires the threshold values from the CQM, which also disseminates it across other IQMs. Finally, the procedure of crowdsourcing continues by following the initial guidelines as already explained in the proposed section.

The case study is conducted by using the similar setup as used for the numerical verification and results are recorded after first response by the identifying node and the revert back time. The performance is affected by the time after which the fake news is identified by one of the peers (IQM) in the S-IoT. After identification, extra time is consumed by the controller for checking the authenticity of fake news marking. However, for evaluation, it is believed that the controller trusts the identifying node and immediately starts retracing the source of fake news, thus, reducing the overall tracking time. The fission time in the detection of sources for fake news is presented in Fig. 14. The results show that a variation in a number of sources affects the performance of fission

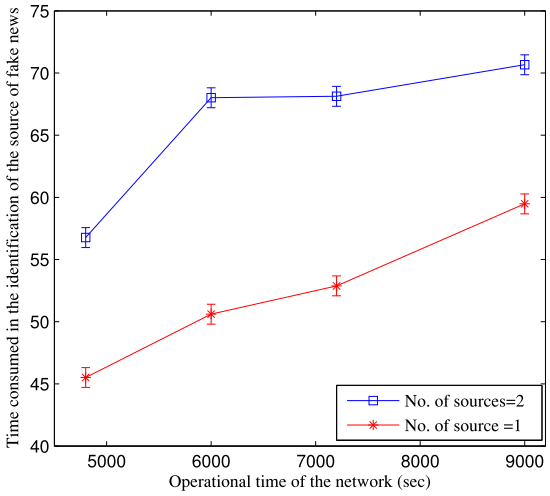

Fig. 14. Time consumed in the identification of the source of fake news vs. operational time of the network.

manager, and for two sources it is affected by 20.9% compared to a scenario with a single source. These results are efficient, but in real-world scenarios, time for calculating the equations in the defined system model, network latency, and node-authentication may cause further delays. Irrespective of these, the proposed approach can be extended to different network setups provided that all the configurations of fission managers are aligned correctly.

7. Conclusion

Trust and privacy in S-IoT are of paramount importance and should be fortified for the smooth run of S-IoT devices. In this paper, a novel solution was proposed in the form of fission computing, which used edge-crowd integration for maintenance of trust and preservation of privacy rules in S-IoT. The proposed solution used crowdsources as mini-edge servers and entropy modeling for maintenance of trust. The proposed cooperative trust relaying and privacy-preserving solution was evaluated by utilizing numerical simulations. The results were obtained by defining the number of untrustworthy users across the total users compared to the baseline scenario. A theoretical analysis was provided to study the impact and multifariousness of the proposed solution. Further, a case study was presented on the detection of fake news sources in S-IoT. Results and case study show that the proposed approach was capable of providing trust without sanctioning an eavesdropper to enter S-IoT network. Further, it was ascertained that privacy rules were always maintained by utilizing the proposed abstracted fission manager.

Acknowledgments

This work was supported by Institute for Information & communications Technology Promotion (IITP) grant funded by the Korea government (MSIT) (No. 2017-0-00664, Rule Specification-based Misbehavior Detection for IoT-Embedded Cyber-Physical Systems) as well as the Soonchunhyang University Research Fund.

References

[1] F. Bao, I.-R. Chen, Dynamic trust management for internet of things applications, in: Proceedings of the 2012 International Workshop on Self-Aware Internet of Things, ACM, 2012, pp. 1–6.
[2] I. Yaqoob, E. Ahmed, I.A.T. Hashem, A.I.A. Ahmed, A. Gani, M. Imran, M. Guizani, Internet of things architecture: Recent advances, taxonomy, requirements, and open challenges, IEEE Wirel. Commun. 24 (3) (2017) 10–16.
[3] L. Atzori, A. Iera, G. Morabito, M. Nitti, The social internet of things (siot)–when social networks meet the internet of things: Concept, architecture and network characterization, Comput. Netw. 56 (16) (2012) 3594–3608.
[4] E. Ahmed, M.H. Rehmani, Introduction to the special section on social collaborative internet of things, 2017, https://doi.org/10.1016/j.compeleceng.2017.04.023.
[5] N.K. Giang, J. Im, D. Kim, M. Jung, W. Kastner, Integrating the epcis and building automation system into the internet of things: a lightweight and interoperable approach, J. Wirel. Mob. Netw. Ubiquitous Comput. Dependable Appl. (JoWUA) 6 (1) (2015) 56–73.
[6] T. Robles, R. Alcarria, D. Martn, M. Navarro, R. Calero, S. Iglesias, M. Lpez, An iot based reference architecture for smart water management processes, J. Wirel. Mob. Netw. Ubiquitous Comput. Dependable Appl. (JoWUA) 6 (1) (2015) 4–23.
[7] S. Sicari, A. Rizzardi, L.A. Grieco, A. Coen-Porisini, Security, privacy and trust in internet of things: The road ahead, Comput. Netw. 76 (2015) 146–164.
[8] Z. Yan, P. Zhang, A.V. Vasilakos, A survey on trust management for internet of things, J. Netw. Comput. Appl. 42 (2014) 120–134.
[9] D. Chen, G. Chang, D. Sun, J. Li, J. Jia, X. Wang, Trm-iot: A trust management model based on fuzzy reputation for internet of things, Comput. Sci. Inf. Syst. 8 (4) (2011) 1207–1228.
[10] N.B. Truong, T.-W. Um, G.M. Lee, A reputation and knowledge based trust service platform for trustworthy social internet of things, in: Innovations in Clouds, Internet and Networks (ICIN), Paris, France, 2016.
[11] B. Kantarci, H.T. Mouftah, Mobility-aware trustworthy crowdsourcing in cloud-centric internet of things, in: 2014 IEEE Symposium on Computers and Communication, (ISCC), IEEE, 2014, pp. 1–6.
[12] H. Sato, A. Kanai, S. Tanimoto, T. Kobayashi, Establishing trust in the emerging era of iot, in: 2016 IEEE Symposium on Service-Oriented System Engineering, (SOSE), IEEE, 2016, pp. 398–406.
[13] R. Chen, F. Bao, J. Guo, Trust-based service management for social internet of things systems, IEEE Trans. Dependable Secure Comput. 13 (6) (2016) 684–696.
[14] M. Nitti, R. Girau, L. Atzori, Trustworthiness management in the social internet of things, IEEE Trans. Knowl. Data Eng. 26 (5) (2014) 1253–1266.
[15] A. Kung, F. Kargl, S. Suppan, J. Cuellar, H.C. Pöhls, A. Kapovits, N.N. McDonnell, Y.S. Martin, A privacy engineering framework for the internet of things, in: Data Protection and Privacy:(In) Visibilities and Infrastructures, Springer, 2017, pp. 163–202.
[16] P. Li, J. Li, Z. Huang, T. Li, C.-Z. Gao, S.-M. Yiu, K. Chen, Multi-key privacy-preserving deep learning in cloud computing, Future Gener. Comput. Syst. 74 (2017) 76–85.
[17] Z. Huang, S. Liu, X. Mao, K. Chen, J. Li, Insight of the protection for data security under selective opening attacks, Inf. Sci. 412–413 (2017) 223–241.
[18] X. Sun, N. Ansari, Edgeiot: Mobile edge computing for the internet of things, IEEE Commun. Mag. 54 (12) (2016) 22–29.
[19] A. Ahmed, E. Ahmed, A survey on mobile edge computing, in: 2016 10th International Conference on Intelligent Systems and Control, (ISCO), IEEE, 2016, pp. 1–8.
[20] K. Zhang, X. Liang, R. Lu, X. Shen, Sybil attacks and their defenses in the internet of things, IEEE Internet Things J. 1 (5) (2014) 372–383.
[21] V. Sharma, I. You, R. Kumar, P. Kim, Computational offloading for efficient trust management in pervasive online social networks using osmotic computing, IEEE Access (2017).
[22] J. Daubert, A. Wiesmaier, P. Kikiras, A view on privacy & trust in iot, in: 2015 IEEE International Conference on Communication Workshop, (ICCW), IEEE, 2015, pp. 2665–2670.
[23] Y.B. Saied, A. Olivereau, D. Zeghlache, M. Laurent, Trust management system design for the internet of things: A context-aware and multi-service approach, Comput. Secur. 39 (2013) 351–365.
[24] V. Sharma, I. You, R. Kumar, Isma: Intelligent sensing model for anomalies detection in cross platform osns with a case study on iot, IEEE Access 5 (2017) 3284–3301.
[25] F. Bao, R. Chen, J. Guo, Scalable, adaptive and survivable trust management for community of interest based internet of things systems, in: 2013 IEEE Eleventh International Symposium on Autonomous Decentralized Systems, (ISADS), IEEE, 2013, pp. 1–7.
[26] R. Chen, J. Guo, F. Bao, Trust management for soa-based iot and its application to service composition, IEEE Trans. Serv. Comput. 9 (3) (2016) 482–495.
[27] D. Reina, R.-I. Ciobanu, S. Toral, C. Dobre, A multi-objective optimization of data dissemination in delay tolerant networks, Expert Syst. Appl. 57 (2016) 178–191.
[28] R.H. Weber, Internet of things–new security and privacy challenges, Comput. Law Secur. Rev. 26 (1) (2010) 23–30.
[29] D. Hussein, S.N. Han, G.M. Lee, N. Crespi, Social cloud-based cognitive reasoning for task-oriented recommendation in the social internet of things, IEEE Cloud Comput. (2017).
[30] I. Butun, Privacy and trust relations in internet of things from the user point of view, in: Computing and Communication Workshop and Conference (CCWC), 2017 IEEE 7th Annual, IEEE, 2017, pp. 1–5.

[31] R. Roman, J. Zhou, J. Lopez, On the features and challenges of security and privacy in distributed internet of things, Comput. Netw. 57 (10) (2013) 2266–2279.

[32] P.P. Jayaraman, X. Yang, A. Yavari, D. Georgakopoulos, X. Yi, Privacy preserving internet of things: From privacy techniques to a blueprint architecture and efficient implementation, Future Gener. Comput. Syst. (2017).

[33] J. Sánchez-García, J.M. García-Campos, D. Reina, S. Toral, F. Barrero, Onsitedriverid: A secure authentication scheme based on Spanish eid cards for vehicular ad hoc networks, Future Gener. Comput. Syst. 64 (2016) 50–60.

[34] A. Ståhlbröst, C.M. Angelopoulos, O. Evangelatos, S. Krco, S. Nikoletseas, T. Raptis, S. Ziegler, Understanding modes of crowdsourcing and related crowd motivators, in: ISPIM Conference Proceedings, The International Society for Professional Innovation Management (ISPIM), 2015, p. 1.

[35] A. Botta, W. De Donato, V. Persico, A. Pescapé, Integration of cloud computing and internet of things: a survey, Future Gener. Comput. Syst. 56 (2016) 684–700.

[36] W. Li, I. Santos, F.C. Delicato, P.F. Pires, L. Pirmez, W. Wei, H. Song, A. Zomaya, S. Khan, System modelling and performance evaluation of a three-tier cloud of things, Future Gener. Comput. Syst. 70 (2017) 104–125.

[37] B. Kantarci, H.T. Mouftah, Trustworthy crowdsourcing via mobile social networks, in: Global Communications Conference (GLOBECOM), 2014 IEEE, IEEE, 2014, pp. 2905–2910.

[38] T. Booth, K. Andersson, Network security of internet services: Eliminate DDoS reflection amplification attacks, J. Internet Serv. Inf. Secur. (JISIS) 5 (3) (2015) 58–79.

[39] K. Zhang, A. Marchiori, Crowdsourcing low-power wide-area iot networks, in: 2017 IEEE International Conference on Pervasive Computing and Communications, (PerCom), IEEE, 2017, pp. 41–49.

[40] N.K. Giang, M. Blackstock, R. Lea, V. Leung, Distributed data flow: A programming model for the crowdsourced internet of things, in: Proceedings of the Doctoral Symposium of the 16th International Middleware Conference, ACM, 2015, p. 4.

[41] B. Zhang, W. Li, B.B. Zhou, A.Y. Zomaya, Home fog server: Taking back control from the cloud, in: Globecom Workshops (GC Wkshps), 2016 IEEE, IEEE, 2016, pp. 1–6.

[42] N. Zhang, X. Yang, M. Zhang, Y. Sun, Crowd-funding: A new resource cooperation mode for mobile cloud computing, PLoS One 11 (12) (2016) e0167657.

[43] T.X. Tran, A. Hajisami, P. Pandey, D. Pompili, Collaborative mobile edge computing in 5g networks: New paradigms, scenarios, and challenges, IEEE Commun. Mag. 55 (4) (2017) 54–61.

[44] V. Sharma, J.D. Lim, J.N. Kim, I. You, Saca: Self-aware communication architecture for iot using mobile fog servers, Mobile Inf. Syst. 2017 (2017).

[45] T. Taleb, S. Dutta, A. Ksentini, M. Iqbal, H. Flinck, Mobile edge computing potential in making cities smarter, IEEE Commun. Mag. 55 (3) (2017) 38–43.

[46] B.P. Rimal, D.P. Van, M. Maier, Mobile edge computing empowered fiber-wireless access networks in the 5g era, IEEE Commun. Mag. 55 (2) (2017) 192–200.

[47] P. Corcoran, Mobile-edge computing and internet of things for consumers: Part II: Energy efficiency, connectivity, and economic development, IEEE Consum. Electron. Mag. 6 (1) (2017) 51–52.

[48] S.K. Sharma, X. Wang, Live data analytics with collaborative edge and cloud processing in wireless iot networks, IEEE Access 5 (2017) 4621–4635.

[49] S. Dama, V. Sathya, K. Kuchi, T.V. Pasca, A feasible cellular internet of things: Enabling edge computing and the iot in dense futuristic cellular networks, IEEE Consum. Electron. Mag. 6 (1) (2017) 66–72.

[50] N.B. Truong, H. Lee, B. Askwith, G.M. Lee, Toward a trust evaluation mechanism in the social internet of things, Sensors 17 (6) (2017) 1346.

[51] E. Cho, S.A. Myers, J. Leskovec, Friendship and mobility: user movement in location-based social networks, in: Proceedings of the 17th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, ACM, 2011, pp. 1082–1090.

[52] V. Sharma, H.-C. Chen, R. Kumar, Driver behaviour detection and vehicle rating using multi-UAV coordinated vehicular networks, J. Comput. System Sci. 86 (2017) 3–32.

[53] R.L. Hughes, A continuum theory for the flow of pedestrians, Transp. Res. B 36 (6) (2002) 507–535.

[54] J.S. Marron, M.P. Wand, Exact mean integrated squared error, Ann. Statist. (1992) 712–736.

[55] S. Ray, B.G. Lindsay, The topography of multivariate normal mixtures, Ann. Statist. (2005) 2042–2065.

[56] Privacy in iot. http://www.businessinsider.com/internet-of-things-security-privacy-2016-8 (Last Accessed: 30.05.17).

[57] D.F. Handbook, Nuclear Physics and Reactor Theory, Department of Energy, Washington DC, 1993.

[58] S.V. Nagaraj, Entropy-based spectrum sensing in cognitive radio, Signal Process. 89 (2) (2009) 174–180.

[59] T.M. Cover, J.A. Thomas, Elements of Information Theory, John Wiley & Sons, 2012.

Vishal Sharma received the Ph.D. and B.Tech. degrees in computer science and engineering from Thapar University (2016) and Punjab Technical University (2012), respectively. He worked at Thapar University as a Lecturer from Apr'16–Oct'16. Now, he is a post-doctoral researcher in MobiSec Lab. at Department of Information Security Engineering, Soonchunhyang University, Republic of Korea. Dr. Sharma has received the best paper award from the IEEE International Conference on Communication, Management and Information Technology (ICCMIT), Warsaw, Poland in April 2017. He is member of various professional bodies and past Chair of ACM Student Chapter-Patiala. His areas of research and interests are 5G networks, UAVs, estimation theory, and artificial intelligence.

Ilsun You (SM'13) received the M.S. and Ph.D. degrees in computer science from Dankook University, Seoul, Korea, in 1997 and 2002, respectively. He received the second Ph.D. degree from Kyushu University, Japan, in 2012. From 1997 to 2004, he was at the THINmultimedia Inc., Internet Security Co., Ltd. and Hanjo Engineering, Co., Ltd. as a research engineer. Now, he is an associate professor at Department of Information Security Engineering, Soonchunhyang University. He has served or is currently serving as a main organizer of international conferences and workshops such as MObiWorld, MIST, SeCIHD, AsiaARES, and so forth. Dr. YOU is the EiC of Journal of Wireless Mobile Networks, Ubiquitous Computing, and Dependable Applications (JoWUA). He is in the Editorial Board for Information Sciences (INS), Journal of Network Computer Applications (JNCA), International Journal of Ad Hoc and Ubiquitous Computing (IJAHUC), Computing and Informatics (CAI), Journal of High Speed Networks (JHSN), Intelligent Automation & Soft Computing (AutoSoft), and Security and Communication Networks (SCN). His main research interests include internet security, authentication, access control, and formal security analysis. He is a Fellow of the IET and a senior member of the IEEE.

Dushantha Nalin K. Jayakody (M'14 received the B.Eng. degree (with first-class honors) from Pakistan and was ranked as the merit position holder of the University under SAARC Scholarship). He received his M.Sc. degree in Electronics and Communications Engineering from Eastern Mediterranean University, Cyprus (under the University full graduate scholarship) and ranked as the first merit position holder of the department. He received the Ph.D. degree in Electronics and Communications Engineering from the University College Dublin, Ireland. From 2014–2016, he has held a Postdoc position at the University of Tartu, Estonia and University of Bergen, Norway. From 2016, he is an Assoc. Professor at the Institute of Cybernetics, National Research Tomsk Polytechnic University, Russia where he also serves as the Director of Tomsk Infocomm Lab. Dr. Jayakody has received the best paper award from the IEEE International Conference on Communication, Management and Information Technology (ICCMIT), Warsaw, Poland in April 2017. Dr. Jayakody is a Member of IEEE and he has served as session chair or technical program committee member for various international conferences, such as IEEE PIMRC 2013/2014, IEEE WCNC 2014/2016, and IEEE VTC 2015 etc. He currently serves as a lead guest editor for the Elsevier **Physical Communications** Journal and MDPI **Information** journal. Also, he serves as a reviewer for various IEEE Transactions and other journals.

Mohammed Atiquzzaman obtained his M.S. and Ph.D. in Electrical Engineering and Electronics from the University of Manchester (UK). He is currently holds the Edith Kinney Gaylord Presidential professorship in the School of Computer Science at the University of Oklahoma and is a senior member of IEEE. Dr. Atiquzzaman is the Editor-in-Chief of Journal of Networks and Computer Applications and the founding Editor-in-Chief of Vehicular Communications and has served/serving on the editorial boards of IEEE Communications Magazine, International Journal on Wireless and Optical Communications, Real Time Imaging Journal, Journal of Communication Systems, Communication Networks and Distributed Systems, and Journal of Sensor Networks. He also guest edited 12 special issues in various journals. He has served as co-chair of IEEE High Performance Switching and Routing Symposium (2011 and 2003) and has served as symposium co-chairs for IEEE Globecom (2006, 2007, and 2014) and IEEE ICC (2007, 2009, 2011, and 2012) conferences. He co-chaired ChinaComm (2008) and SPIE Next-Generation Communication and Sensor Networks (2006) and the SPIE Quality of Service over Next Generation Data Networks conferences (2001, 2002, 2003,

and 2005). He was the panels co-chair of INFOCOM'05 and is/has been in the program committee of numerous conferences such as INFOCOM, ICCCN, and Local Computer Networks. He serves on the review panels of funding agencies such as the National Science Foundation and National Research Council (Canada) and Australian Research Council (Australia). In recognition of his contribution to NASA research, he received the NASA Group Achievement Award for "outstanding work to further NASA Glenn Research Center's effort in the area of Advanced Communications/Air Traffic Management's Fiber Optic Signal Distribution for Aeronautical Communications" project. He is the co-author of the book " Performance of TCP/IP over ATM networks" and has over 270 refereed publications, which are accessible at www.cs.ou.edu/~atiq. His research interests are in communications switching , transport protocols, wireless and mobile networks, ad hoc networks, satellite networks, quality of service, and optical communications. His research has been funded by National Science Foundation (NSF), National Aeronautics and Space Administration (NASA), Us Air Force, Cisco, Honeywell, Oklahoma Department of Transportation, Oklahoma Highway Safety Office through grants totaling over $7M.