Boosting Service Provisioning in SIoT by Exploiting Trust and Capability Levels of Social Objects

Giancarlo Sciddurlo, Antonio Petrosino, Domenico Striccoli,
Giuseppe Piro, Luigi Alfredo Grieco, and Gennaro Boggia
Dept. of Electrical and Information Engineering - Politecnico di Bari, Bari, Italy
Email: {*name.surname*}@*poliba.it*
CNIT, Consorzio Nazionale Interuniversitario per le Telecomunicazioni

***Abstract*—The Social Internet of Things is gaining momentum thanks to its unique capability of i) autonomously building social relationships among smart objects and ii) supporting the novel services within the Social Network. During the service provisioning, the Trust Management System is in charge of selecting suitable objects able to accomplish the requested services. In this context, available solutions present two relevant issues to be solved. First, they generally assume to select social objects that only achieve higher trustworthiness without considering their actual computing capabilities. Indeed, it would significantly compromise the Quality of Experience in the Social Network of objects. Second, they assume to implement this task directly within constrained devices, becoming unpractical considering their limited computational and storage capabilities. To improve and speed up service provisioning, this paper proposes a novel Trust Management System scheme that fairly distributes service requests by jointly considering the trustworthiness and resource capabilities of available objects. This Trust Management System has been designed to exploit fog computing to efficiently handle the whole process of service provisioning in real-world deployments while relieving constrained devices from all processing and storage efforts. Its behavior is investigated through computer simulations. Obtained results demonstrated that the conceived approach outperforms baseline solutions in terms of latency, fairness in services distribution, and responsiveness in malicious nodes' detection.**

***Index Terms*—Social Internet of Things, Service Provisioning, Resource Capability-aware Scheme.**

I. Introduction

The promising integration of Social Networks in the Internet of Things (IoT) domain promoted the birth of the Social Internet of Things (SIoT) [1]. By leveraging autonomous interactions, smart objects can build social relationships composing a Social Network without human intervention. Thus, the transition from smart objects to social objects introduces additional opportunities to enhance network resource visibility and service discovery [2]. Reproducing the digital counterpart of the physical IoT devices strongly favors the network navigability and opens the opportunity to explore several new application scenarios (e.g., healthcare applications [3], and Vehicular Social Networks [4]). It requires the development of effective methodologies for service provisioning, where the trustworthiness of service providers must be guaranteed [5]. In this context, the Trust Management System (TMS) represents the key element for the evaluation of the behavior of social objects and their selection as a service provider. It facilitates trusted interactions between social objects by computing their trust level, thus penalizing nodes that adopt malicious or incorrect behaviors [6].

The research in this field explored many methodologies devoted to calculate and manage the trust levels of the service providers in SIoT environments (see [7]–[12] and the review of the state-of-the-art in Section II). However, to the best of the authors' knowledge, specific strategies accounting for the computing capabilities of social objects in the TMS to speed up and improve service provisioning are still an unexplored issue. Furthermore, available solutions are not fully applicable to real-world scenarios due to their computational complexity, being not easily manageable by most of the resource-constrained IoT devices.

In order to extend the scientific literature in this direction, this work proposes a novel resource capability-aware scheme for the TMS. Specifically, the additional functionalities introduced in this contribution jointly consider the trustworthiness, resource availability, and the computational capabilities of the objects registered in the Social Network to speed up the trusted service provisioning. Furthermore, differently from other works, the proposed strategy exploits fog computing to implement all the TMS functionalities. It relieves the processing and storage efforts of the IoT nodes for the overall TMS procedure, including the construction of social relationships, thus making the strategy suitable for realistic scenarios.

Simulation results show the effectiveness of the proposed scheme in terms of latency in service provisioning (reduced up to 67% with respect to the baseline solutions) while guaranteeing fairness in the distribution of available resources among service providers. Furthermore, the new features of the TMS improve the responsiveness of the identification of malicious users, promptly excluding them from the process of service provisioning.

The remainder of this paper is organized as reported below. Section II reviews the state-of-the-art addressing the most recent proposal on TMSs. Section III illustrates the structure

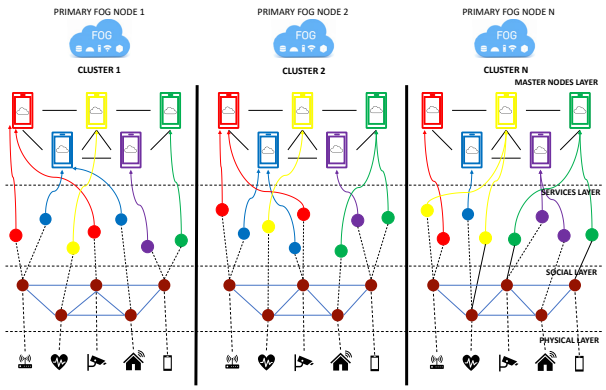


Fig. 1. The proposed layered architecture.

of the proposed environment, including the system architecture and the details about the implemented TMS procedure. Section V evaluates the performance of the proposed approach through computer simulations. Section VI concludes the paper and draws future research activities.

II. Related works

At the time of this writing and to the best of the authors' knowledge, actual contributions only consider the design and the analysis of a TMS from a trustworthiness perspective, thus leaving some limitations on the implemented procedures concerning responsiveness, resource capability, efficiency, and scalability.

For example, the studies in [7] and [8] face the selection of the most suitable service provider in a TMS considering the energy constraints to evaluate the provider resources. The former proposes device trust dimension referred to the current energy status remaining unaware of device computational capability. The latter aims to increase the network lifetime, but it ignores the management of service requests and storage-saving procedures.

The paper [9] develops two algorithms for an efficient resource selection available through information sharing between social objects. Nevertheless, the described strategies do not implement a trust metric that jointly considers trustworthiness and resource efficiency aspects. The contribution in [10] proposes a distributed architecture based on a Blockchain for the secure provision of trust in an IoT system. It adopts a stochastic approach to detect and prevent malicious behaviors within a lightweight implementation. The results are obtained only for typical deployments operating with a limited number of nodes. So, the approach is not suitable for real-time systems, since it can only handle information generated at a quite low speed.

The authors in [11] investigate a TMS for the evaluation of service providers' past experiences and quality of service recommendations. The described self-adapted model dynamically fits changes in network context or type of demanded service. However, the model presented is entirely borne by the social objects. Consequently, it has the main limitation of the applicability to real-world scenarios with limited computation and storage capabilities of IoT devices.

Another work presented in [12] suggests a hybrid method to overcome both the weakness of centralized and decentralized approaches for trust management. Despite considering the evaluation of available resources, it mainly focuses on the user trust classification. In particular, it detects possible trust attacks via Machine Learning methods, without investigating the opportunity to distribute service requests to the most suitable service providers from the capability point of view.

Differently from the previously discussed studies, this paper aims to solve the typical open issues regarding a TMS in SIoT environment, overcoming the limitations of the current state-of-the-art. The strategy presented herein, in fact, investigates a novel resource capability-aware scheme that embraces several challenging aspects, such as responsiveness, resource capability, efficiency, and scalability in a SIoT environment to achieve trusted interactions in the service provisioning, supplying efficient resource management and quality of experience between users.

III. The overall system architecture

This work considers a reference environment made up of several IoT nodes grouped in clusters (based on the geographical location) and managed in a distributed manner. Fig.1 shows the resulting multi-layered SIoT architecture.

The lower layer is the Physical Layer, where the set of nodes is represented by the physical IoT devices. Each IoT device can act as a service requester or service provider. For the sake of generality, this work considers three different classes of IoT devices [13]:

- Class 0: devices are very constrained in terms of resources (i.e., sensors with tens or hundreds of kilobytes of RAM);
- Class 1: devices are constrained in terms of resources, but with some processing capabilities (i.e., Arduino, smart-cameras);
- Class 2: devices have enough resources and a lot of RAM to perform heavy computations (i.e., smartphones).

The second layer is the Social Layer. In this layer, IoT devices become social objects and, through their abstraction, they can represent the digital counterpart of the physical ones gaining the opportunity to expose their attributes useful to generate social relationships. In particular, a social object is identified by an ID, an owner and manufacturer identifier, and all the attributes specifying its performance abilities, such as power level and clock speed.

The third layer is the Service Layer. Here, each social object specifies the list of services it can provide. In such a virtualized service layer, a social object can join communities based on the same application context, facilitating network navigability.

Besides, the overall architecture embraces two levels leveraging fog computing technology. The first, namely the Master Node level, is formed by fog nodes with high computational capabilities to handle and distribute service requests. It hosts the TMS, which recommends the service provider to be

TABLE I
FRIENDSHIP TIES RATES.

Type of relationship	OOR	POR	C-LOR	C-WOR
S_{ij}	0.7	0.65	0.6	0.55

selected during the service provisioning process. Each Master Node manages one or more service communities. Moreover, to support the decision for the most suitable service provider, it stores all the information related to past experiences and the whole set of attributes of registered social objects useful to generate a social-based virtual topology for each service community. The upper level, namely the Primary Fog level, is constituted by Primary Fog Nodes with high storage capacity that handles the set of information of a cluster of social objects. Specifically, a Primary Fog Node allows the proper synchronization between the distributed clusters structure through the interaction between other Primary Nodes.

IV. DETAILS ON THE CONCEIVED METHODOLOGY

By joining the network, a social object expresses its availability to provide a service. In this phase, through proper API, it communicates its attributes to Master Nodes, useful to reconstruct a virtual topology of the established social relationships. The social relationships calculation and storage are fully delegated to Master Nodes, enabling the applicability of the proposed strategy to a real-world scenario by offloading the social objects from this computational effort.

Whenever a node needs to retrieve a service, a request is sent to the Master Node that hosts the community providing that service. This approach simplifies the service discovery, thus limiting the selection to a subset of providers based on social relationships.

By handling a service request, each Master Node runs the TMS functionalities aiming to compute the most suitable service provider. Again, social objects are relieved of any computational load. This is a further advantage since the computation of the most suitable service provider could significantly impact storage and resources employed, being impractical for several IoT devices.

The two procedures implemented by the TMS for the management of service requests are summarized in the following steps.

Step 1 - Trust List Evaluation. For the incoming request, the Master Node verifies in its database the attendance of social objects having a social relationship with the requester. Consequently, it produces a Trust list of potential service providers. For each provider in the Trust list, the TMS evaluates the Trust value, which quantifies the level of trustworthiness of the service provider. It is computed as follows.

Considering the i-th social object requesting a service and the j-th social object as a possible provider, the Trust value $T_{i,j}$ is calculated through two main factors. Firstly, the Sociality factor $S_{i,j}$ expresses the friendship ties between social objects. Table I describes the rates of the established relationship, classified in order of relevance (i.e., the Ownership Object Relationship (OOR) referred to the same owner is the stronger friendship tie, followed by Parental Object Relationship (POR), Co-Location Object Relationship (C-LOR), and Co-Work Object Relationship (C-WOR) referred to the same manufacturer, location, and working goal, respectively). Secondly, the Reputation factor $R_{i,j}$ is defined based on the history of the previous behavior of social objects, expressed through past received feedback. The Reputation factor is evaluated as follows:

$$R_{i,j} = \alpha\Delta_{i,j} + \beta\Theta_{i,j} + \gamma\Pi_{i,j}, \tag{1}$$

where:

- $\Delta_{i,j}$ represents the direct reputation and is calculated as the sum of positive feedback values divided by the total number of the feedbacks given by the i-th requester to the j-th provider;
- $\Theta_{i,j}$ represents the friend indirect reputation and is calculated as the sum of positive feedback values divided by the total number of the feedbacks given by friends of the i-th requester to the j-th provider;
- $\Pi_{i,j}$ represents the overall indirect reputation and is calculated as the sum of positive feedback values divided by the total number of the feedbacks given by the other non-friends of the i-th requester to the j-th providers;
- α, β, and γ are weights determining the relevance of each factor.

Finally, the Trust value is computed as reported in Eq.(2) (for details on the Trust model please refer to [14]):

$$T_{i,j} = S_{i,j} \cdot R_{i,j}. \tag{2}$$

Once the Trust value has been assigned to all potential providers in the Trust list, the procedure immediately discards any service provider with a Trust value lower than an empirically selected threshold. This feature prevents the possibility for misbehaved nodes to clean up their reputation and return to acting maliciously later.

Step 2 - Resource Capability Management. In order to increase the quality of the service provisioning experienced, the composed Trust list is double-sorted. Specifically, the first sorting parameter is the device class, whereas the second is the computed Trust value. This methodology represents a novel key aspect of the conceived scheme: this is a new ordering method aiming to ensure proper resource utilization, enabling the opportunity to satisfy delay-sensitive tasks demanding stringent performance and quality. The benefit of the trusted provider selection offering the right resources for the execution of the services will be motivated by the numerical results in the next Section.

Through the aforementioned strategy, the Master Node obtains a ranking based on social (Step 1) and performance (Step 2) perspectives. Besides, the proposed TMS considers a further investigation, addressing the resource availability of

TABLE II
DEVICE PARAMETERS.

Social Object Class	Power Level	Clock Speed (Clk) [Megacyles/s]	QoE Class
Smartphone	0.8	2000	2
Smart Gateway	0.6	1000	1
Smart cam	0.4	1000	1
Sensors	0.2	40	0

TABLE III
SERVICES REQUIREMENTS.

Service ID	1	2	3	4	5	6
Resource Consumption	0.3	0.2	0.1	0.2	0.1	0.3
Information Size (B) [Mbit]	1.4	1.0	0.6	1.0	0.6	1.4

suitable providers. Indeed, the resource management functionality monitors the resources of the recommended provider and verifies its availability to perform the service. If this check fails, the recommended provider is temporarily removed from the Trust list (it will be reconsidered in the list whenever it will have enough available resources). The system runs the same investigation on the updated ranking until an eligible provider is found.

The most suitable provider in the ranking, entertaining the needed resources, is selected for the service execution. Finally, the requester provides feedback about the executed service. The feedback communicates if the service accomplishment correctly matches the request. The received feedback is then stored in the proper Master Nodes, useful for Trust value updates in case of upcoming service requests.

V. PERFORMANCE EVALUATION

In this Section, the performance of the proposed methodology is evaluated through computer simulations. To this end, a C++ object-oriented and event-driven simulator has been developed from scratch to estimate the proper Key Performance Indicators (KPIs) of a heterogeneous Social Network, evaluating latency, fairness, and responsiveness offered by the involved providers.

A. Simulation parameters

Without loss of generality, the proposed scenario considers a cluster of social objects under the control of a single Primary Node. In this cluster, the TMS is performed by 5 Master Nodes. The number of social objects belonging to the cluster ranges from 100 to 300 (i.e., 100, 150, 200, and 300).

According to what is described in Section III, each social object is characterized by a unique object ID, owner ID, manufacturer ID, geographical location, processor clock speed (expressed in megacycles/s), power level, and a list of offered services. Several types of devices (i.e., smartphones, smart gateway, smart cameras, and sensors) are generated within a uniform distribution across the cluster, whose computing capabilities are reported in Table II. A social object can be classified into 3 different Quality of Experience (QoE) classes (also reported in Table II).

As expected for a common SIoT deployment, the conceived scheme assigns each generated social object to a Master Node. Then, the defined attributes, such as owner, manufacturer, and geographical position, allow to define social relationships. Specifically, this work considers OOR for objects owned by the same user, POR for devices produced by the same manufacturer, and C-LOR for devices in the same location. Furthermore, the proposed scenario considers 6 different types of services, each one defined by a unique ID, a parameter addressing the requested resources for task completion (ranging from 0.1 to 0.3), and the bit size of the information to be processed as denoted in Table III. According to this, the j-th social object can offer the service S by reserving part of its computing capabilities for an amount of time $t(j, S)$ equal to:

$$t(j, S) = \frac{X \cdot B(S)}{Clk(j)}, \tag{3}$$

where X is the number of CPU cycles needed to process a single bit, $B(S)$ describes the total number of bits to process to accomplish the service S, and $Clk(j)$ denotes the clock speed of the j-th social object in charge to process the service S expressed in cycles/s. According to [15], X has been set equal to 1000 $\frac{cycles}{bit}$.

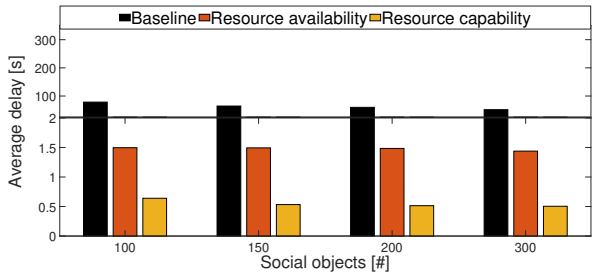
To evaluate the network performance considering different traffic loads, service requests are generated accordingly to a Poisson distribution with an average rate λ ranging from 4 to 10 request/s and randomly selected from those described in Table III. Moreover, for each considered scenario, results are obtained over 20 different seeds to account for several network topologies, services, and social relationship distributions.

Finally, the conceived scheme is compared with two other approaches. The first, which is the baseline approach, does not consider any knowledge of the quantity and the quality of the available resources in the network. The second, namely the resource availability aware approach (presented in [14]), takes into account resource availability but does not consider any resource capability management scheme for the selection of service providers in TMS.

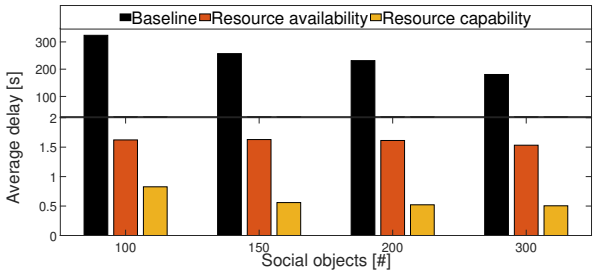
B. Average delay

The first KPI used to compare the conceived approach against the others is the average delay experienced in the service provisioning. It represents the average time taken by each requested service from its generation until the end of its execution. Results are reported in Fig. 2. More in detail, the baseline approach bases the provider selection only on the trustworthiness parameter while neglecting the availability of resources. As a result, it fosters the provider selection through the same small subset of nodes with high trustworthiness by overloading them with service requests. Therefore, even with low λ values, this approach experiences the highest average delay.

On the contrary, the resource availability approach experiences a lower average delay than the baseline thanks to its ability to spread service requests over the social objects owning enough free resources to accomplish it. This strategy



(a) λ = 4 requests/s.



(b) λ = 10 requests/s.

Fig. 2. Average delay.

significantly reduces the waiting time for an available provider. Delay results do not substantially vary even though the request rate increases.

Finally, the proposed resource capability-aware scheme always registers the lowest experienced delay. Indeed, considering providers with adequate resources and a higher QoE class during the provider selection boosts the performance for the fulfillment of the required tasks and drastically reduces latency in service provisioning. The efficiency of the proposed approach to accomplish requests by finding the best provider is confirmed in a large-scale scenario, ensuring better responsiveness and scalability for the network if compared with the others. In fact, in the case of 300 social objects and λ equal to 10 requests/s, the average delay reduces up to 67% with respect to the resource availability approach.

C. Processing Time

Table IV shows the processing time of the conceived scheme to estimate the suitable service provider. The simulator runs on a computer equipped with a CPU i7-7700 and 16 GB of RAM. The processing time denotes the average time needed by a Master Node to perform the overall procedure executed in the TMS and described in Section IV. Each scheduled request is processed in a time range from 1.2 to 4.5 ms, which is very negligible if compared to the overall delay experienced in the service provisioning. Thus, the magnitude of the obtained processing times demonstrates the computational lightness and the scalability of the proposed scheme for different traffic loads and Social Network sizes.

D. QoE Fairness Index

To strengthen the obtained results, the well-known QoE Fairness Index (presented in [16]) is evaluated. The index estimation quantifies the fairness in services distribution by considering the QoE perceived by social objects. It is calculated as:

$$F = 1 - \frac{2\sigma}{H - L}, \tag{4}$$

where σ is the standard deviation providing a measure of the dispersion of QoE among social objects, while H and L are the upper and lower device classes, respectively. Fig. 3 depicts

TABLE IV
PROCESSING TIME.

Scenario		Processing time [ms]
Social Objects	**λ [requests/s]**	
100	4	4.5
	10	3.6
150	4	1.3
	10	1.2
200	4	2.0
	10	1.7
300	4	4.2
	10	4.0

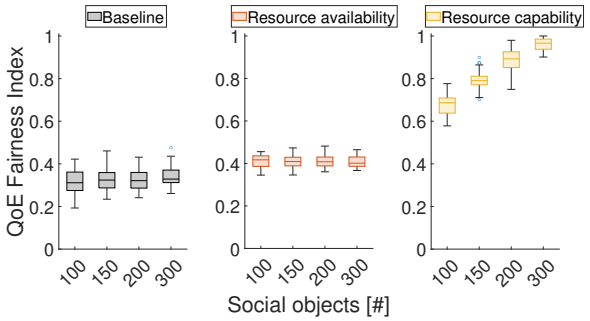


Fig. 3. QoE Fairness Index.

the QoE Fairness Index. On each box, the central mark denotes the median. The bottom and top edges, instead, indicate the 25^{th} and 75^{th} percentiles, respectively. Considering 100 social objects, the proposed approach experiences a QoE Fairness Index equal to 0.6, doubling if compared to the others. The efficiency also increases considering a larger Social Network size. In fact, the boost of the QoE Fairness Index triples in a large-scale scenario of 300 social objects, confirming the capability of the proposed approach to be scalable and fair in services distribution.

E. Responsiveness in malicious nodes identification

This Section proposes further valuable considerations about the effectiveness of the conceived strategy in identifying malicious nodes. Fig.4 depicts the temporal evolution of the direct

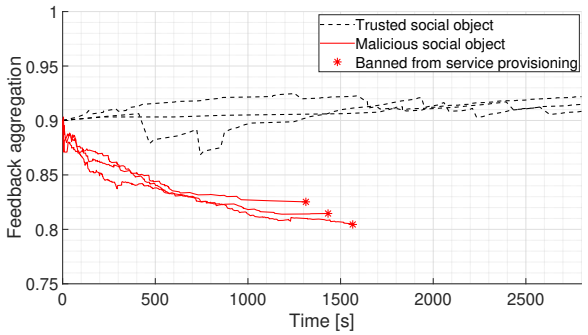


Fig. 4. Temporal evolution of the aggregated feedback.

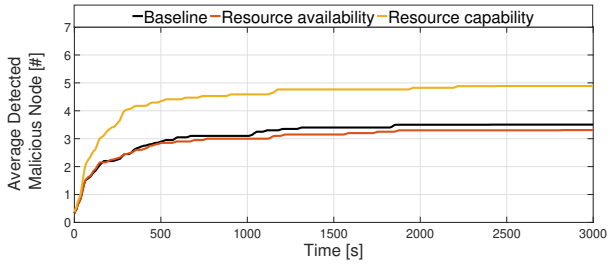


Fig. 5. Responsiveness in malicious nodes identification.

feedback received by the same provider averaged on the total number of feedbacks, called aggregated feedback. This aspect is evaluated in a Master Node for six social objects provisioning the same service. In this scenario, three selected nodes can act maliciously, providing poor services more frequently than others. Consequently, the assigned negative feedback impacts the overall reputation of the misbehaved provider. The obtained results show the ability of the proposed TMS to identify malicious nodes. Indeed, the misbehaved social objects are isolated from the service provisioning process and will never be contacted in the future, as testified by the red curves, truncated after about 1300 s. This result testifies that no further feedback will be provided anymore for the three malicious nodes that are banned from service provisioning. To provide further insight, Fig. 5 depicts the number of malicious nodes recognized by the TMS over time and averaged over 20 seeds for the three schemes chosen for comparison. The considered scenario includes 100 social objects, and ten among them have a higher probability of acting maliciously. In this scenario, already after 500 seconds, the TMS designed in this work can identify a higher number of nodes to be excluded from the network if compared to the other solutions, testifying excellent responsiveness of the proposed approach to detect ambiguous behaviors during the service provisioning process.

VI. Conclusions

This paper proposes a novel resource capability-aware TMS scheme in the service provisioning for a SIoT environment. The proposed strategy aims to grant trusted services with a high Quality of Experience, overcoming some fundamental limitations in this research field, such as responsiveness, resource capability, efficiency, and scalability. Computer simulations have been performed by comparing the proposed approach with baseline solutions. Obtained results highlight that the proposed scheme can process service requests in real-time, experiencing low latency within a fair resource distribution and relieving the IoT devices from any computational load. Furthermore, it allows a responsive identification of malicious nodes, preventing them from acting as providers for forthcoming service requests. Future research activities will further examine the multi-clustered structure by investigating the synchronization procedure between Primary Nodes. Furthermore, a security service will be set for real-time propagation of the malicious nodes identification along the clusters.

References

[1] L. Atzori, A. Iera, G. Morabito, and M. Nitti, "The social internet of things (siot) – when social networks meet the internet of things: Concept, architecture and network characterization," *Computer Networks*, vol. 56, 11 2012.

[2] L. Atzori, A. Iera, and G. Morabito, "From "smart objects" to "social objects": The next evolutionary step of the internet of things," *IEEE Communications Magazine*, vol. 52, no. 1, pp. 97–105, 2014.

[3] A. M. Esfahani, A. M. Rahmani, and A. Khademzadeh, "Msiot: Mobile social internet of things, a new paradigm," in *2020 10th International Symposium onTelecommunications (IST)*, 2020, pp. 187–193.

[4] Z. Yang, K. Yang, L. Lei, K. Zheng, and V. C. M. Leung, "Blockchain-based decentralized trust management in vehicular networks," *IEEE Internet of Things Journal*, vol. 6, no. 2, pp. 1495–1505, 2019.

[5] H. Zhang, L. Zhu, H. Du, L. Zhang, K. Zhang, Y. Yan, and C. Wang, "Structural balance of social internet of things networks with ambiguous relationships," *Wireless Communications and Mobile Computing*, vol. 2021, p. 7964409, Aug 2021.

[6] A. Tewari and B. Gupta, "Security, privacy and trust of different layers in internet-of-things (iots) framework," *Future Generation Computer Systems*, vol. 108, pp. 909–920, 2020.

[7] Z. Chen, R. Ling, C.-M. Huang, and X. Zhu, "A scheme of access service recommendation for the social internet of things," *International Journal of Communication Systems*, vol. 29, no. 4, pp. 694–706, 2016.

[8] A. Zannou, A. Boulaalam, and E. H. Nfaoui, "Siot: A new strategy to improve the network lifetime with an efficient search process," *Future Internet*, vol. 13, no. 1, 2021.

[9] A. Metrouh, "Social internet of things: a novel selection approach for dynamic resources substitution," *Evolutionary Intelligence*, 06 2021.

[10] B. Bordel and R. Alcarria, "Distributed trust and reputation services in pervasive internet-of-things deployments," in *International Symposium on Mobile Internet Security*. Springer, 2021, pp. 16–29.

[11] R. Abidi and N. B. Azzouna, "Self-adaptive trust management model for social iot services," in *2021 International Symposium on Networks, Computers and Communications (ISNCC)*, 2021, pp. 1–7.

[12] W. Abdelghani, I. Amous, C. A. Zayani, F. Sèdes, and G. Roman-Jimenez, "Dynamic and scalable multi-level trust management model for social internet of things," *The Journal of Supercomputing*, Jan 2022.

[13] M. O. Ojo, S. Giordano, G. Procissi, and I. N. Seitanidis, "A review of low-end, middle-end, and high-end iot devices," *IEEE Access*, vol. 6, pp. 70 528–70 554, 2018.

[14] G. Sciddurlo, I. Huso, D. Striccoli, G. Piro, and G. Boggia, "A multi-tiered social iot architecture for scalable and trusted service provisioning," in *2021 IEEE Global Communications Conference (GLOBECOM)*, 2021, pp. 1–6.

[15] G. Zhang, F. Shen, Y. Zhang, R. Yang, Y. Yang, and E. A. Jorswieck, "Delay minimized task scheduling in fog-enabled iot networks," in *2018 10th International Conference on Wireless Communications and Signal Processing (WCSP)*, 2018, pp. 1–6.

[16] T. Hoßfeld, L. Skorin-Kapov, P. E. Heegaard, and M. Varela, "Definition of qoe fairness in shared systems," *IEEE Communications Letters*, vol. 21, no. 1, pp. 184–187, 2016.