Collaboration Between Social Internet of Things and Mobile Users for Accuracy-Aware Detection

Kang-Yen Chen[‡], Chih-Hang Wang[†], Sheng-Hao Chiang[†], De-Nian Yang[†], Wen-Tsuen Chen[§], and Jang-Ping Sheu[§]
[‡]Dept. of Communication Engineering, National Tsing Hua University, Hsinchu, Taiwan
[†]Institute of Information Science, Academia Sinica, Taipei, Taiwan
[§]Dept. of Computer Science, National Tsing Hua University, Hsinchu, Taiwan
E-mail: s106064529@m106.nthu.edu.tw, {superwch7805, jiang555, dnyang}@iis.sinica.edu.tw, and {wtchen, sheujp}@cs.nthu.edu.tw

***Abstract*—Social Internet of Things (SIoT) has become an emerging network paradigm, where IoT devices with Artificial Intelligence (AI) and social relations can automatically establish a collaborative group to identify events locally. On the other hand, mobile users can act as ubiquitous and versatile sensors to improve the accuracy of SIoT event detection. In this paper, we explore the SIoT Collaboration with Crowdsourcing (SCC) problem to jointly select SIoT devices and hire users to monitor events and locations with accuracy requirements, while minimizing the total SIoT communication and computation costs and the user hiring cost. We prove that SCC is NP-hard and cannot be approximated by any factor unless P = NP. Then, we propose a new algorithm, Accuracy- and Social-aware SIoT and User Selection (ASSUS), with the idea of Collaborative Tree (CT) and Accuracy Profit (AP), where CT exploits users' social relations to properly choose intermediate SIoTs. Simulation results manifest that ASSUS can effectively reduce more than 50% of the total cost compared with state-of-the-art algorithms.**

I. Introduction

Social Internet of Things (SIoT) has emerged to cooperatively process data and identify local event by a set of IoT devices with social relations [1], [2].[1] First, SIoTs produced by the same manufacturer can share *parental object relation*, whereas those possessed by the same owner can share *ownership object relation*. Next, SIoTs can build *co-location object relation* and *co-work object relation* if they are neighbors and designed to manage similar events, respectively. For example, the SIoTs with co-location and co-work object relations can collaboratively identify the traffic congestion and crowd gathering regions to figure out *why* and *how* that happens [2]. Moreover, researchers have recently exploited the social relations between mobile devices [3] and SIoTs [4] jointly to trace and spot undocumented patients and infectious places for containing COVID-19. To precisely identify events, a monitored location is usually covered by multiple SIoTs to enhance the fault tolerance and reliability of SIoTs. However, nearby SIoTs may not always meet the accuracy requirement for monitoring each location due to the noise and uncertainty of measurement.

Online social networks (OSNs) provide plentiful information to capture society dynamics and enable people to act as ubiquitous and versatile sensors to report (post) their observations [5], [6]. To improve SIoT detection, a promising way is to hire users online via OSNs (so-called crowdsourcing [7]) to observe desired locations. Since performing sensing tasks requires users to devote their time and mobile device resources (e.g., battery), it is desirable to motivate participants by rewarding them with a certain quantity of payment [8].[2] However, malicious users may spread fake information on OSNs, affecting the correctness of event identification. To this end, Daniel et al. [9] explored the truth discovery under users' reliability, report credibility, and historical behaviors to avoid misinformation spread. Jun et al. [10] proposed a peer prediction-based trustworthy service rating to identify malicious and unreliable users. Kardelen et al. [11] extracted the features of OSNs to estimate the accuracy of event detection from users. Nevertheless, the above works did not consider the accuracy of SIoTs and users jointly to select SIoTs and hire users for ensuring the accuracy requirement of event detection and location monitoring. For example, the coastal monitoring system in the Poetto beach in Italy needs users' feedback to improve the quality of SIoT detection [12].

Different from previous works focusing on only SIoTs [1], [2] or OSNs [5], [6], this paper exploits crowdsourcing via OSNs to help SIoTs search and monitor events to minimize the total costs, including SIoT communication and computation costs and the user hiring cost (i.e., payment for hiring users [8]). However, the problem is challenging due to the following research issues. 1) *Tradeoff in cost and accuracy*. An SIoT of a user (i.e., ownership object relation) may be able to be employed by her friends when the user agrees [13]. However, a trustful user possessing SIoTs may require a larger hiring cost due to a long distance to the monitored location.[3] Moreover, when a user has fewer SIoTs, it may be necessary to involve more users to exploit more SIoTs (with a large total cost) for satisfying the accuracy requirement. 2) *Collaborative group construction for SIoTs*. To identify an event, it needs to build connected SIoTs to ensure that SIoTs can communicate with each other. However, some closeby SIoTs cannot be adopted when they belong to untrusted users, and distant SIoTs of

[1]For ease of presentation, we use (S)IoTs to represent (S)IoT devices.

[2]Crowdsourcing platforms, like Uber (https://www.uber.com) and Amazon Mechanic Turk (https://www.mturk.com/), reward users for their participation.

[3]The hiring cost is proportional to the moving distance (i.e., the effort that the user makes to reach the location) [14].

trusted friends with larger communication costs are required in this case. 3) *Tradeoff in SIoT connectivity and OSN social relations*. A user with good social centrality is able to employ the SIoTs of more friends. However, these SIoTs may be located in various places and thereby are difficult to ensure the SIoT connectivity.

To effectively address the above issues, this paper formulates a new optimization problem, named SIoT Collaboration with Crowdsourcing (SCC), to select SIoTs and users (i.e., hire users to help monitor an event or a location) for minimizing the total SIoT communication and computation costs, as well as the hiring cost of friends and crowdsourcing. We prove that SCC is NP-hard and inapproximable within any factor unless $P = NP$. Then, we design an algorithm, called Accuracy- and Social-aware SIoT and User Selection (ASSUS), to construct a *Collaborative Tree (CT)* of SIoTs and choose users to jointly detect events and monitor locations for improving SIoT detection accuracy. For the first challenge, ASSUS introduces the notion of *Accuracy Profit (AP)* to examine the accuracy increment per unit cost for each SIoT and user. For the second challenge, AP carefully evaluates the computation cost of each SIoT and the communication cost of the corresponding path to attach to the CT, as well as the hiring cost of users possessing those SIoTs. ASSUS then extracts SIoTs and users based on AP to ensure the accuracy requirements and minimize the total cost. Afterward, ASSUS tailors CT by replacing some SIoTs and their paths connected to CT with those inducing smaller costs. For the third challenge, ASSUS removes the users whose *activated* SIoTs[4] are no longer required by examining their social relations. Afterward, ASSUS trims the CT to minimize the communication cost and pairwisely swaps users' monitored locations to reduce the hiring cost.

The rest of this paper is organized as follows. Section II formulates SCC and presents the hardness result. Section III proposes ASSUS. Section IV summarizes the simulation results, and Section V concludes this paper.

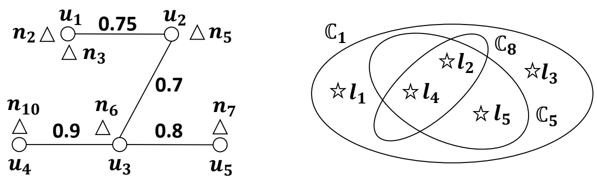
II. Problem

A. System Model

The system model includes SIoTs and OSNs. For SIoTs, we denote by $G^{SIoT} = (V^{SIoT}, E^{SIoT})$ a network with a set of SIoTs V^{SIoT} and a set of social links E^{SIoT}. Each SIoT $n \in V^{SIoT}$ includes a computation cost β_n. A link $e_{n,m} \in E^{SIoT}$ with communication cost $\alpha_{n,m}$ exists if SIoTs $n, m \in V^{SIoT}$ have social relations and can communicate with each other [1], [2]. Due to the *ownership object relations* [1], [2], some SIoTs are private (belong to some users) and the others are public. If an SIoT is private, it requires to be *activated* (detailed later) to employ it. Let $\mathbb{L}$ be the set of monitored locations, and each SIoT $n \in V^{SIoT}$ is associated with coverage $\mathbb{C}_n \subseteq \mathbb{L}$ and accuracy a_l^n for monitoring location $l \in \mathbb{C}_n$.

For OSNs, we denote by $G^{OSN} = (V^{OSN}, E^{OSN})$ an OSN with a set of users V^{OSN} and a set of social links E^{OSN}

[4]An SIoT owned by a user is *activated* if she or her friends with sufficiently strong social relations are selected (detailed later). An activated SIoT can be employed to monitor locations or act as a relay to connect to other SIoTs.



(a) The user's OSN (b) The coverage of SIoTs

Fig. 1. An example of OSN and SIoT coverage.

TABLE I
An Example of CM

$a_{l_1}^{n_1}$		identified result			
		θ_1	θ_2	θ_3	θ_4
true event	θ_1	410	10	30	50
	θ_2	10	250	30	10
	θ_3	34	25	700	41
	θ_4	10	70	80	240

topology, where a link $e_{u,v} \in E^{OSN}$ with weight (i.e., the strength of social relation) $\phi_{u,v}$ exists if users $u, v \in V^{OSN}$ are friends. According to the trust transitivity property [15], [16], the social relation between two users can be aggregated (e.g., if A trusts B and B trusts C, then A trusts C to some extent). The social relation $r_{u,v}$ of u and v is expressed as $r_{u,v} = \max_{e_{u,u'} \in E^{OSN}} \phi_{u,u'} \cdot r_{u',v}$ [16]. Each user u includes an accuracy a_l^u for monitoring location $l \in \mathbb{L}$[5] and owns a set $\mathbb{D}_u \subseteq V^{SIoT}$ of SIoTs (i.e., *ownership object relations*). For example, $r_{u_1,u_3} = 0.75 \times 0.7 = 0.525$ and $\mathbb{D}_1 = \{n_2, n_3\}$ since u_1 owns $\{n_2, n_3\}$ in Fig. 1(a).[6]

We exploit the confusion matrix (CM) [17] to represent the accuracy of SIoTs [18] and users [19] according to the historical records. The accuracy a_l^n (a_l^u) of SIoT n (user u) monitoring location l is defined as the proportion of correct identifications (classifications) to the total identified instances,[7] $a_l^n = \frac{\sum_{i=1}^{N} CM_l^n(i,i)}{\sum_{i=1}^{N}\sum_{j=1}^{N} CM_l^n(i,j)}$. $CM_l^n = [M_{i,j}]_{N\times N}$ represents the identification results of SIoT n monitoring location l. $M_{i,j}$ is the identification result that the SIoT identifies event i as event j, and N is the total number of identifications. Each row in CM indicates the instances in a true event, while each column represents the instances in an identified result. Fig. 1(b) shows an example, where each circle and star represent the SIoT coverage and monitored location, respectively. SIoT n_1 covers $(l_1, l_2, l_3, l_4, l_5)$, and it can identify 4 events (i.e., $\theta_1, \theta_2, \theta_3, \theta_4$) for l_1, with the CM shown in Table I. For true event θ_1, n_1 correctly identifies it as θ_1 410 times ($\{10, 30, 50\}$ for $\{\theta_2, \theta_3, \theta_4\}$) in the overall 500 times (i.e., $410 + 10 + 30 + 50 = 500$). Hence, the accuracy of $a_{l_1}^{n_1}$ is $\frac{410+250+700+240}{500+300+800+400} = 0.8$.

B. Problem Formulation

Equipped with the SIoT, OSN and accuracy model, we formulate SCC as follows. The objective is to minimize the

[5]The users' opinions on a specific event can be collected on the OSNs (e.g., Facebook users' interactions on user walls) [11], and the credibility assessment system can evaluate the accuracy of users' judgment [9].

[6]In Fig. 1(a), each circle and triangle represent user and SIoT, respectively.

[7]For simplicity, we only describe the accuracy of SIoTs and omit users since they are associated with the same formula.

total SIoT communication, computation, and the hiring cost. Let $x_{n,m}$ and y_n denote if edge $e_{n,m} \in E^{SIoT}$ is selected for communication and SIoT n is chosen for monitoring locations,[8] respectively. The total communication and computation costs of SIoTs is $C_{SIoT} = \sum_{e_{n,m} \in E^{SIoT}} x_{n,m} \cdot \alpha_{n,m} + \sum_{n \in V^{SIoT}} y_n \cdot \beta_n$. For crowdsourcing, let binary variables z_u and $w_{u,l}$ denote if user u is chosen and if user u is assigned to monitor location l, respectively.[9] The total hiring cost of users is $C_{OSN} = \sum_{u \in V^{OSN}} z_u \cdot \gamma_u + \rho \sum_{u \in V^{OSN}, l \in \mathbb{L}} w_{u,l} \cdot s_{u,l}$, where γ_u is the basic payment (cost) for choosing user u for monitoring or using u's SIoTs [14], ρ is the unit distance cost, and $s_{u,l}$ is the distance between user u and monitored location l. SCC minimizes the total cost $C_{SIoT} + C_{OSN}$.

SCC includes the following constraints. 1) *SIoT connectivity constraint*. The selected SIoTs forming a collaborative group are required to be connected on G^{SIoT} to ensure that they can communicate with each other [2]. 2) *Accuracy constraint*. The average accuracy of selected SIoTs and users that cover each monitored location $l \in \mathbb{L}$ with the accuracy exceeding a threshold λ. Following [11], [18], for each monitored location l, the average accuracy $a(l)$ is

$$a(l) = \frac{\sum_{n \in V^{SIoT}} a_l^n \cdot y_n + \sum_{u \in V^{OSN}} a_l^u \cdot w_{u,l}}{\sum_{n \in V^{SIoT}} |l \cap \mathbb{C}_n| \cdot y_n + \sum_{u \in V^{OSN}} w_{u,l}} \geq \lambda. \quad (1)$$

3) *Social trust constraint*. To preserve privacy [13], the SIoT set $\mathbb{D}_v$ owned by user v (i.e., *ownership object relations*) can be *activated* by choosing v or u if and only if social relation $r_{u,v}$ of users u and v exceeds a threshold δ [15].

Definition 1 (SCC)**.** Given 1) a set of monitored locations $\mathbb{L}$ with accuracy requirement λ, 2) an SIoT network $G^{SIoT} = (V^{SIoT}, E^{SIoT})$ with computation cost β_n, accuracy a_l^n and coverage $\mathbb{C}_n$ for each SIoT $n \in V^{SIoT}$, and weight $\alpha_{n,m}$ for each link $e_{n,m} \in E^{SIoT}$, and 3) an OSN $G^{OSN} = (V^{OSN}, E^{OSN})$ with possessed SIoTs $\mathbb{D}_u$, basic cost γ_u, distance $s_{u,l}$ and accuracy a_l^u for each user u monitoring location $l \in \mathbb{L}$, weight $\phi_{u,v}$ for each link $e_{u,v} \in E^{OSN}$ and social relation threshold δ, SCC aims to select a subset of V^{SIoT} and V^{OSN} to minimize $C_{SIoT} + C_{OSN}$, whereas the SIoT connectivity, accuracy, and social trust constraints hold.

Theorem 1. *AASUS is NP-hard and cannot be approximated by any factor unless $P = NP$.*

Proof. Due to the space constraint, the detailed proofs of NP-hardness and inapproximability are presented in [20]. □

III. Algorithm Design

To solve SCC, an intuitive approach is to assign each user to the nearest monitored location and then iteratively select the SIoT with maximum coverage until the accuracy requirement of each monitored location is met [21]. However, a user with a low distance cost may be associated with a large hiring cost, and an SIoT with good coverage may induce unacceptable accuracy and large computation and communication costs. Moreover, the approach ignores the OSN social relations and SIoT connectivity. To address these issues, we propose ASSUS with the following phases: 1) Collaborative SIoT and User Selection (CSUS), 2) SIoT Replacement (SR), and 3) CT Pruning and User Swapping (CTPUS). CSUS first introduces *Accuracy Profit (AP)* to estimate the accuracy increment per unit cost for each SIoT and user. It then jointly selects SIoTs and users with the maximum AP to meet the accuracy requirements, where the selected SIoTs are connected via minimum-cost paths to construct a *Collaborative Tree (CT)* to maintain SIoT connectivity. Next, SR replaces some SIoTs and their paths connected to CT with those incurring smaller costs, where the social relations of the corresponding users are carefully examined to ensure trust. Finally, CTPUS evaluates *ownership object relations* to trim the CT for reducing the communication cost and pairwisely swaps users' monitored locations to lower the hiring cost. The time complexity of ASSUS is $O(|V^{SIoT}|^2 \log |V^{SIoT}| + |V^{OSN}|^2 |\mathbb{L}|)$. Due to the space constraint, the detailed complexity analysis and pseudocode are provided in [20].

1) *Collaborative SIoT and User Selection (CSUS)*: CSUS first constructs a collaborative tree (CT) for SIoTs to ensure the SIoT connectivity. The *accuracy profit* (AP) of SIoTs and users are evaluated jointly to ensure the accuracy requirements and minimize the total cost. Specifically, the AP of each SIoT and user is defined as the ratio of its accuracy increment to its induced cost for satisfying the accuracy requirements with the minimum cost. CSUS iteratively extracts the SIoT (or user) with the maximum AP and finds the minimum-cost path connected to the CT to ensure the SIoT connectivity. In the following, we first define the AP of each SIoT and user.

For an SIoT n, the costs include the communication and computation costs, as well as the hiring cost of a user that is involved for choosing n due to the *ownership object relations*. Let $a(l)$ be the accuracy of monitored location l, and $CT(n)$ denotes the SIoT on the CT nearest to n. We denote by $P^{SIoT}_{n,CT(n)}$ the minimum-cost path between n and $CT(n)$ on G^{SIoT}, and the induced communication cost thereby is the total cost of links, $c(P^{SIoT}_{n,CT(n)}) = \sum_{e_{n',m'} \in P^{SIoT}_{n,CT(n)}} \alpha_{n',m'}$.[10] The induced hiring cost for selecting SIoTs on the path $P^{SIoT}_{n,CT(n)}$ is $\sum_{u \in \mathcal{U}_{n,CT(n)} \setminus \mathbb{U}} \gamma_u$, where $\mathcal{U}_{n,CT(n)}$ is the set of users owning the SIoTs on $P^{SIoT}_{n,CT(n)} \setminus \{CT(n)\}$, and $\mathbb{U}$ is the set of chosen users to 1) monitor locations or 2) activate their SIoTs to be employed. The AP of SIoT n is

$$AP(n) = \frac{\sum_{l \in \mathbb{C}_n} (a_l^n - a(l))}{c(P^{SIoT}_{n,CT(n)}) + \beta_n + \sum_{u \in \mathcal{U}_{n,CT(n)} \setminus \mathbb{U}} \gamma_u}. \quad (2)$$

The AP of user u includes two cases since u can be chosen for activating SIoTs or monitoring a location l. Specifically,

[8] An SIoT may be chosen as a relay only for ensuring the SIoT connectivity, and its computation cost in this case is ignored.

[9] A user may be chosen only for *activating* her or her friends' SIoTs, but she is not assigned to monitor any location. Accordingly, two binary variables are required to differentiate them.

[10] If n is on CT or the first selected SIoT (i.e., the root), $P^{SIoT}_{n,CT(n)} = n$ and $c(P^{SIoT}_{n,CT(n)}) = 0$.

TABLE II
AN EXAMPLE OF SIOT ACCURACY

	l_1	l_2	l_3	l_4	l_5		l_1	l_2	l_3	l_4	l_5
n_1	0.8	0.78	0.76	0.77	0.82	n_6	0.82	0.8	0.84		0.76
n_2	0.78		0.76	0.77	0.85	n_7	0.72		0.8		
n_3		0.7		0.85		n_8		0.65		0.88	
n_4			0.65	0.88		n_9	0.81	0.82			0.8
n_5		0.8		0.82	0.78	n_{10}	0.77			0.81	

TABLE III
AN EXAMPLE OF γ_u, a_l^u AND $s_{u,l}$

	γ_u	$a_{l_1}^u$	$a_{l_2}^u$	$a_{l_3}^u$	$a_{l_4}^u$	$a_{l_5}^u$	$s_{u,1}$	$s_{u,2}$	$s_{u,3}$	$s_{u,4}$	$s_{u,5}$
u_1	1	0.3	0.5	0.4	0.6	0.5	10	9	11	8	10
u_2	1	0.6	0.6	0.5	0.81	0.81	50	60	55	3	3
u_3	4	0.6	0.5	0.6	0.84	0.85	66	65	60	2	2
u_4	2	0.5	0.4	0.5	0.82	0.83	50	60	60	3	4
u_5	2	0.92	0.9	0.8	0.9	0.9	110	95	110	95	95

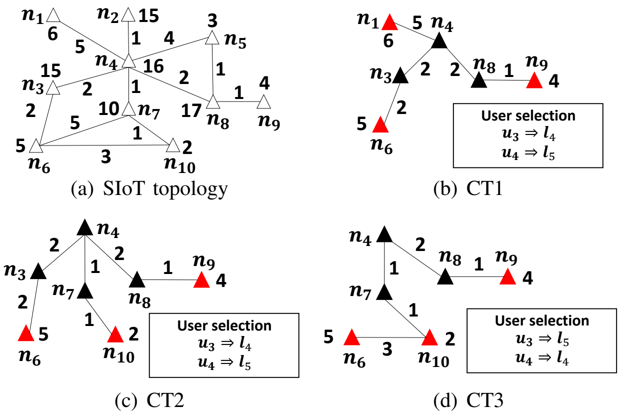
the AP of user u for monitoring location l is

$$AP(u,l) = \begin{cases} \frac{a_l^u - a(l)}{\rho \cdot s_{u,l}}, \forall u \in \mathbb{U} \setminus \mathbb{W}, \\ \frac{a_l^u - a(l)}{\gamma_u + \rho \cdot s_{u,l}}, \forall u \in V^{OSN} \setminus \mathbb{U}, \end{cases} \tag{3}$$

where $\mathbb{W} \subseteq \mathbb{U}$ is the set of users chosen for monitoring locations. If user u has been chosen for activating her SIoTs (i.e., the basic cost γ_u has been added), only the distance cost $\rho \cdot s_{u,l}$ of u moving to monitored location l is considered; otherwise, both the basic cost and distance cost are necessary to be examined.

CSUS iteratively chooses the SIoT (or user) with the maximum AP to ensure the accuracy requirements with the minimum cost and construct a CT accordingly. More specifically, if SIoT n with the maximum $AP(n)$ is chosen for monitoring,[11] CSUS also selects the SIoTs (and their corresponding owners) on $P_{n,CT(n)}^{SIoT}$ as relays to ensure the SIoT connectivity, i.e., the selected SIoT is connected to CT via $P_{n,CT(n)}^{SIoT}$. On the other hand, if the user u with the maximum $AP(u,l)$ is chosen, CSUS assigns user u to monitor location l. If there are more than one user with the same maximum AP, CSUS extracts the one that activates more SIoTs. Specifically, we define $Q_u = \mathbb{D}_u \cup \bigcup_{v|r_{u,v} \geq \delta} \mathbb{D}_v, \forall u \in V^{OSN}$ as the set of activated SIoTs when u is chosen, and $Q_{\mathbb{U}} = \bigcup_{u \in \mathbb{U}} Q_u$. CSUS chooses the user with the maximum $|Q_u \setminus Q_{\mathbb{U}}|$ to activate more SIoTs. Afterward, CSUS updates the accuracy $a(l)$ of each location $l \in \mathbb{C}_n$ considered above according to (1). CSUS stops when the accuracy requirement of each monitored location is met.

Example 1. Figs. 1(a) and 2(a) present an illustrative example. The weight of the triangle is the SIoT computation cost, and the weight of the edge is the communication cost between SIoTs. Parameters δ, λ and ρ are set to 0.8, 0.8 and 1, respectively, and other parameters are summarized in Table II and Table III. Since $AP(n_1) = \frac{0.8+0.78+0.76+0.77+0.82}{0+6+0} = 0.655$ is the largest, CSUS first selects n_1 as the root of CT. Similarly, it assigns u_3 to monitor l_4 since it generates the maximum AP, and $\mathbb{U} = \{u_3\}$ and $\mathbb{W} = \{u_3\}$. Afterward, CSUS selects n_6 and connects it to CT via path $P_{n_6,CT(n_6)}^{SIoT} = n_6 \rightarrow n_3 \rightarrow n_4 \rightarrow n_1$ and $AP(n_6) = \frac{(0.82-0.8)+(0.8-0.78)+(0.84-0.76)+(0.76-0.82)}{9+5+1} = 0.004$, where $c(P_{n_6,CT(n_6)}^{SIoT}) = 2+2+5 = 9$ and $\beta_{n_6} = 5$. Since $n_6 \in \mathbb{D}_3$ and $n_3 \in \mathbb{D}_1$ on $P_{n_6,CT(n_6)}^{SIoT} \setminus \{CT(n_6)\}$, it is required to select $\mathcal{U}_{n_6,CT(n_6)} \setminus \mathbb{U} = \{u_1\}$. With the same process, the final results of CT and user selection (to monitor locations) are shown in Fig. 2(b). The result of $\mathbb{U} = \{u_1, u_3, u_4\}$, $\mathbb{W} = \{u_3, u_4\}$ and the set of SIoTs for monitoring and relaying are $\{n_1, n_6, n_9\}$ and $\{n_3, n_4, n_8\}$, respectively. The total cost $C_{SIoT} + C_{OSN} = \big((5+2+2+2+1)+(6+5+4)\big) + \big((4+2+1)+(4+2)\big) = 40$ and the accuracy of monitored locations $(l_1, l_2, l_3, l_4, l_5)$ are $(0.81, 0.8, 0.8, 0.805, 0.8025)$.



(a) SIoT topology (b) CT1 (c) CT2 (d) CT3

Fig. 2. An illustrative example of SCC.

2) *SIoT Replacement (SR)*: SR replaces the selected SIoTs with others that generate a smaller total cost and then lowers the hiring cost by removing some users and their SIoTs. Let $CT = (V^{CT}, E^{CT})$ denote the CT with the selected SIoTs V^{CT} (including relays) and communication links E^{CT}, and $\mathbb{CT}_L$, $\mathbb{CT}_B$ and $\mathbb{CT}_M$ represent the set of leaves, branch nodes (i.e., the nodes with at least three incident edges) and the monitoring nodes (i.e., the SIoTs selected for monitoring locations) on CT, respectively. Note that each leaf node $n \in \mathbb{CT}_L$ must be the SIoT selected for monitoring (i.e., $\mathbb{CT}_L \subseteq \mathbb{CT}_M$), since CSUS constructs CT by iteratively connecting a monitoring SIoT via a path. Our idea is to iteratively evaluate each leaf n and replace the corresponding path $P_{n,B(n)}^{CT}$ by an improved one $P_{m,CT(m)}^{SIoT}$ with a smaller cost, where $m \notin \mathbb{CT}_M$, $B(n)$ is the nearest upstream monitoring node or branch node of n on CT,[12] and $P_{n,B(n)}^{CT}$ is the minimum-cost path from n to $B(n)$ on CT.

More specifically, SR first prioritizes leaf nodes $\mathbb{CT}_L$ in descending order of their reduced costs if they are replaced. The reduced cost $RC(n)$ of a leaf node $n \in \mathbb{CT}_L$ includes the computation cost β_n, communication cost from n to $B(n)$ on CT, and hiring costs of users that can be removed if their SIoTs are no longer required after removing $P_{n,B(n)}^{CT}$. That is, $RC(n) = c(P_{n,B(n)}^{CT}) + \beta_n + \sum_{u \in \mathbb{X}} \gamma_u$, where $\mathbb{X}$ is the set of users u that can be removed from $\mathbb{U}$ if u is not chosen for

[11]Note that if n has been selected as a relay in the previous iteration, $\sum_{u \in \mathcal{U}_{n,CT(n)} \setminus \mathbb{U}} \gamma_u = 0$ since $\mathcal{U}_{n,CT(n)} = \emptyset$.

[12]To ensure the SIoT connectivity, the branch node cannot be removed directly even if it is chosen as a relay.

monitoring, and its activated SIoTs Q_u are no longer involved as the monitoring nodes or relays.

SR then iteratively replaces each path $P^{CT}_{n,B(n)}$ by an improved one with a smaller total cost. More specifically, since the accuracy requirement of some locations may not be satisfied after removing $P^{CT}_{n,B(n)}$, SR iteratively chooses the SIoT $m \in V^{SIoT} \setminus \mathbb{CT}_M$ that leads to the most accuracy increment for those unsatisfied locations $l \in \mathbb{C}_n$, whereas the accuracy requirements of other locations are still met.[13] In each iteration, SR finds the increasing cost $IC(m) = c(P^{SIoT}_{m,CT(m)}) + \beta_m + \sum_{u \in \mathcal{U}_{m,CT(m)} \setminus \mathbb{U}} \gamma_u$ until the accuracy requirement of each location is satisfied or $\sum IC(m) \geq RC(n)$. If $\sum IC(m) \geq RC(n)$, $P^{CT}_{n,B(n)}$ will not be replaced, and SR examines the next leaf node. Otherwise, it connects $P^{SIoT}_{m,CT(m)}$ to CT and updates $\mathbb{CT}_M = \mathbb{CT}_M \cup \{m\} \setminus \{n\}$ and $\mathbb{CT}_L = \mathbb{CT}_L \setminus \{n\}$.[14] The above process stops until all the leaf nodes are examined.

Example 2. Fig. 2(b) presents an illustrative example. $RC(n_1) = c(P^{CT}_{n_1,B(n_1)}) + \beta_{n_1} = 5 + 6 = 11$. $RC(n_6) = c(P^{CT}_{n_6,B(n_6)}) + \beta_{n_6} + \sum_{u \in \mathbb{X}} \gamma_u = 4 + 5 + 1 = 10$, where $\mathbb{X} = \{u_1\}$ because u_1 is not chosen for monitoring, and her activated SIoTs $\{n_2, n_3\}$ are no longer required after removing $P^{CT}_{n_6,B(n_6)}$. Since $RC(n_1)$ is the largest one, SR removes $P^{CT}_{n_1,B(n_1)}$, but $a(l_5)$ becomes unsatisfied. Hence, SR selects n_{10} to complement $a(l_5)$ with increasing cost $IC(n_{10}) = 2+2+2 = 6$ since $IC(n_{10}) = 6 < RC(n_1) = 11$. Fig. 2(c) shows the final CT and selected users, where $\mathbb{U} = \{u_1, u_3, u_4, u_5\}$, $\mathbb{W} = \{u_3, u_4\}$, $\mathbb{CT}_L = \{n_6, n_9, n_{10}\}$, $\mathbb{CT}_B = \{n_4\}$ and $\mathbb{CT}_M = \{n_6, n_9, n_{10}\}$. The total cost is reduced from 40 to 35.

3) *CT Pruning and User Swapping (CTPUS)*: After SR selects users to activate more SIoTs to improve the solution, CTPUS first examines those SIoTs with ownership object relations to trim the CT, and it then minimizes the hiring cost by swapping users' monitored locations. Specifically, CTPUS first carefully examines the graph $\bar{G}^{SIoT} = G^{SIoT} \setminus \{Q_{V^{OSN}} \setminus Q_{\mathbb{U}}\}$ to avoid incurring additional hiring costs (i.e., $\bar{G}^{SIoT}$ consists of only the public SIoTs or the activated SIoTs) when pruning the CT.[15] CTPUS sequentially removes the path $P^{CT}_{n,B(n)}$ and reconnects n to the nearest node m on CT via the minimum-cost path if $c(P_{n,m}) < c(P^{CT}_{n,B(n)})$, to reduce the SIoT communication costs, where $P_{n,m}$ is the minimum-cost path between n and m on $\bar{G}^{SIoT}$. CTPUS then minimizes the hiring cost by pairwisely swapping users' monitored locations (to reduce their moving distances) and removing some users that are no longer required. Specifically, let $l(u)$ be the location monitored by user u. CTPUS iteratively swaps users u and v if 1) $s_{u,l(u)} + s_{v,l(v)} > s_{u,l(v)} + s_{v,l(u)}$ and 2) the accuracy requirements of $l(u)$ and $l(v)$ are satisfied after swapping, until every user is examined. Afterward, CTPUS removes the users that are not involved for monitoring locations and activating SIoTs to lower the hiring cost.

[13]Note that an SIoT may deteriorate the accuracy of some locations due to the longer monitoring distance or noises.

[14]To avoid the ping-pong effect, the removed nodes will not be considered in the following iterations.

[15]Note that CT is on $\bar{G}^{SIoT}$ since each node $n \in CT$ is activated.

Example 3. Following Example 2, in Fig. 2(c), $\bar{G}^{SIoT} = G^{SIoT} \setminus \{Q_{V^{OSN}} \setminus Q_{\mathbb{U}}\}$, $c(P^{CT}_{n_6,B(n_6)}) = 4$, and $c(P_{n_6,n_{10}}) = 3$, where $P^{CT}_{n_6,B(n_6)} = n_6 \to n_3 \to n_4$ and $P_{n_6,n_{10}} = n_6 \to n_{10}$. CTPUS removes $P^{CT}_{n_6,B(n_6)}$ and u_1 because $n_3 \in \mathbb{D}_{u_1}$, and it connects n_6 to CT since $c(P_{n_6,n_{10}}) = 3 < c(P^{CT}_{n_6,B(n_6)}) = 4$. CTPUS then swaps the monitored locations of u_3 ($l_4 \Rightarrow l_5$) and u_4 ($l_5 \Rightarrow l_4$) because $s_{u_3,l(u_4)} + s_{u_4,l(u_3)} = 5 < s_{u_3,l(u_3)} + s_{u_4,l(u_4)} = 6$ and $a(l_4)$ and $a(l_5)$ are satisfied after swapping. Moreover, $Q_{u_3} = \{n_6, n_7, n_{10}\}$ since $\gamma_{u_3,u_4} = 0.9 \geq 0.8$ and $\gamma_{u_3,u_5} = 0.8 \geq 0.8$ such that u_5 (activate $\mathbb{D}_{u_5} = \{n_7\}$) can be removed to reduce the redundant hiring cost. The final results of CT and user selections are shown in Fig. 2(d), where $\mathbb{U} = \{u_3, u_4\}$, $\mathbb{W} = \{u_3, u_4\}$, $\mathbb{CT}_L = \{n_6, n_9\}$ and $\mathbb{CT}_M = \{n_6, n_9, n_{10}\}$. Finally, the total cost is reduced to 30.

IV. Simulation

A. Simulation Setup

We first uniformly distribute users and monitored locations into a $400m \times 400m$ square area according to [18]. To ensure the accuracy requirement of each monitored location, we then deploy some SIoTs with sufficient accuracy to cover it, while the other SIoTs are evenly distributed over the whole area [18]. The numbers of SIoT, users, monitored locations are set to 1200, 100 and 1000 in default, respectively. The accuracy requirement λ is set to 0.8 [22]. The social relations of SIoTs are established based on the co-location and ownership object relations [16], and the communication costs are assigned according to their communication distances [23], while the computation costs are set according to the SIoT identification ability [24]. The OSN topology is generated by the Barabasi Albert approach [25]. The weights of social links are generated from 0 to 1, and the average number of SIoTs owned by a user is set to 5 in default. The social relation threshold δ is assigned to 0.8 [15]. Since there is no related work exploring the interplay between SIoTs and OSNs, we compare ASSUS with state-of-the-art SIoT and user selection schemes, Nearest-first (NA) [21], Simple-Greedy (SG) [15], and BadZak-1 [26]. To evaluate ASSUS, we change the following parameters: 1) the number of SIoTs, 2) OSN degree, and 3) social relation threshold δ. We measure the following performance metrics: 1) total cost, 2) communication and computation costs, 3) hiring cost, 4) moving distance, and 5) SIoT coverage. Each result is averaged over 200 samples. Due to the space constraint, more simulation results are presented in [20].

B. Simulation Result

In Figs. 3(a) and 3(b), ASSUS significantly outperforms the baselines (i.e., BadZak-1, NA and SG) regarding the total cost and hiring cost, because it carefully examines the AP of each SIoT and user to minimize the total cost, while the accuracy requirements are satisfied. Moreover, CSUS replaces the paths in the CT to reduce SIoT communication costs, and CTPUS swaps users' monitored locations to lower their

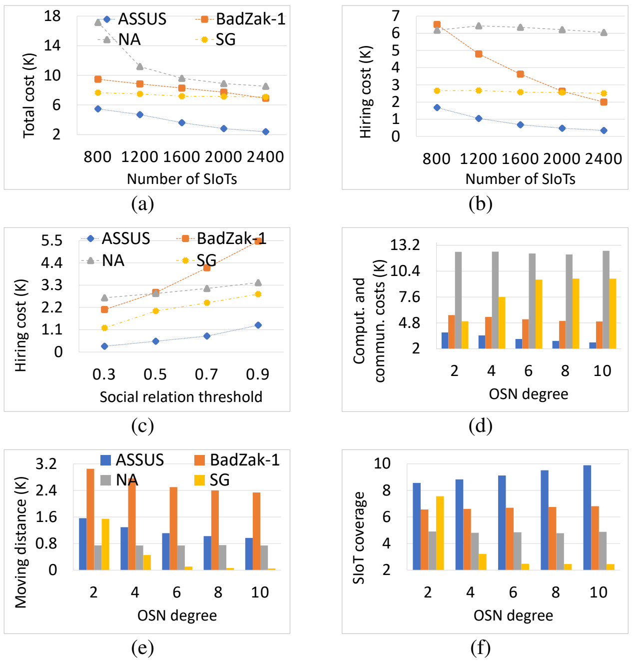


Fig. 3. Simulation results. (a) Total cost vs the number of SIoTs. (b) Hiring cost vs the number of SIoTs. (c) Hiring cost vs social relation threshold. (d) Computation and communication costs vs OSN degree. (e) Moving distance vs OSN degree. (f) SIoT coverage vs OSN degree.

moving distances and minimize hiring costs. In contrast, the baselines do not jointly consider the accuracy of SIoTs and users, inducing a much higher cost to ensure the accuracy. In Fig. 3(c), ASSUS has more opportunities to select users with better accuracy to satisfy the accuracy requirements, and these users with good social centrality can activate more SIoTs, when social relation constraint is looser. In Fig. 3(d), ASSUS induces much smaller computation and communication costs as OSN degree increases, since it can exploit friends' SIoTs with better AP to construct a CT. In contrast, the baselines ignore OSN social relations to activate SIoTs. In Figs. 3(f) and 3(e), ASSUS achieves a good balance between SIoTs and OSNs to minimize the total cost (see Fig. 3(a)). It exploits more activated SIoTs with better accuracy when OSN degree grows. Although NA and SG induce smaller moving distances by assigning users to monitor their nearest locations, they ignore the accuracy of SIoT and lead to worse SIoT coverage.

V. Conclusions

To the best of our knowledge, this paper makes the first attempt to explore the collaboration between SIoTs and OSN users for accuracy-aware detection and monitoring. We first formulate SCC to minimize the total communication and computation costs of SIoTs and the total hiring cost of OSN users. We prove that SCC is NP-hard and cannot be approximated by any ratio unless $P = NP$. Then, we design ASSUS with the idea of CT and AP to ensure the SIoT connectivity for communications and accuracy requirements, where CT also considers users' social relations to activate possessed SIoTs. Simulation results manifest that ASSUS can effectively reduce the total cost by more than 50%.

References

[1] M. Lippi, M. Mamei, S. Mariani, and F. Zambonelli, "An argumentation-based perspective over the social iot," *IEEE IoT J.*, vol. 5, no. 4, pp. 2537–2547, Aug 2018.
[2] C. Wang, J. Kuo, D. Yang, and W. Chen, "Collaborative social internet of things in mobile edge networks," *IEEE IoT J.*, pp. 1–1, 2020.
[3] B. Wang, Y. Sun, T. Q. Duong, L. D. Nguyen, and L. Hanzo, "Risk-aware identification of highly suspected covid-19 cases in social iot: A joint graph theory and reinforcement learning approach," *IEEE Access*, vol. 8, pp. 115 655–115 661, 2020.
[4] B. Benreguia, H. Moumen, and M. A. Merzoug, "Tracking covid-19 by tracking infectious trajectories," *IEEE Access*, vol. 8, pp. 145 242–145 255, 2020.
[5] S. Zhao, Y. Gao, G. Ding, and T. Chua, "Real-time multimedia social event detection in microblog," *IEEE Trans. Cybern.*, vol. 48, no. 11, pp. 3218–3231, Nov 2018.
[6] L. Shi *et al.*, "Human-centric cyber social computing model for hot-event detection and propagation," *IEEE Trans. Comput. Social Syst.*, vol. 6, no. 5, pp. 1042–1050, Oct 2019.
[7] W. Gong, B. Zhang, and C. Li, "Location-based online task assignment and path planning for mobile crowdsensing," *IEEE Trans. Veh. Technol.*, vol. 68, no. 2, pp. 1772–1783, Feb 2019.
[8] D. Zhao, X. Li, and H. Ma, "How to crowdsource tasks truthfully without sacrificing utility: Online incentive mechanisms with budget constraint," in *Proc. IEEE INFOCOM*, 2014.
[9] D. Zhang, D. Wang, N. Vance, Y. Zhang, and S. Mike, "On scalable and robust truth discovery in big data social media sensing applications," *IEEE Trans. Big Data.*, vol. 5, no. 2, pp. 195–208, June 2019.
[10] J. Du *et al.*, "Peer prediction-based trustworthiness evaluation and trustworthy service rating in social networks," *IEEE Trans. Inf. Forensics Secur.*, vol. 14, no. 6, pp. 1582–1594, 2019.
[11] K. Cepni, M. Ozger, and O. B. Akan, "Event estimation accuracy of social sensing with facebook for social internet of vehicles," *IEEE IoT J.*, vol. 5, no. 4, pp. 2449–2456, Aug 2018.
[12] R. Girau *et al.*, "Coastal monitoring system based on social internet of things platform," *IEEE IoT J.*, vol. 7, no. 2, pp. 1260–1272, 2020.
[13] I. Chen, F. Bao, and J. Guo, "Trust-based service management for social internet of things systems," *IEEE Trans. Depend. Secur. Comput.*, vol. 13, no. 6, pp. 684–696, Nov 2016.
[14] W. Wang *et al.*, "Strategic social team crowdsourcing: Forming a team of truthful workers for crowdsourcing in social networks," *IEEE Trans. Mob. Comput.*, vol. 18, no. 6, pp. 1419–1432, June 2019.
[15] J. Tang, X. Tang, and J. Yuan, "Profit maximization for viral marketing in online social networks: Algorithms and analysis," *IEEE Trans. Knowl. Data Eng.*, vol. 30, no. 6, pp. 1095–1108, June 2018.
[16] Z. Lin and L. Dong, "Clarifying trust in social internet of things," *IEEE Trans. Knowl. Data Eng.*, vol. 30, no. 2, pp. 234–248, Feb 2018.
[17] M. Janidarmian, A. Roshan Fekr, K. Radecka, and Z. Zilic, "Multi-objective hierarchical classification using wearable sensors in a health application," *IEEE Sens. J.*, vol. 17, no. 5, pp. 1421–1433, March 2017.
[18] C.-H. Yang, C.-H. Wang, D.-N. Yang, and W.-T. Chen, "Accuracy and precision-aware iot device selection in mobile edge networks," in *Proc. IEEE WCNC*, 2018.
[19] M. Alrubaian, M. Al-Qurishi, M. M. Hassan, and A. Alamri, "A credibility analysis system for assessing information on twitter," *IEEE Trans. Depend. Secur. Comput.*, vol. 15, no. 4, pp. 661–674, July 2018.
[20] K.-Y. Chen *et al.*, "Collaboration between social internet of things and mobile users for accuracy-aware detection (full version)," Oct. 2020, [Online]. Available: http://mnet.cs.nthu.edu.tw/NTHU-TechRep-KY.pdf.
[21] X. Tao and W. Song, "Location-dependent task allocation for mobile crowdsensing with clustering effect," *IEEE IoT J.*, vol. 6, no. 1, pp. 1029–1045, 2019.
[22] Y. Wang and M. S. Kankanhalli, "Tweeting cameras for event detection," in *Proc. Int. Conf. World Wide Web.*, 2015.
[23] H. Lin and W. Chen, "An approximation algorithm for the maximum-lifetime data aggregation tree problem in wireless sensor networks," *IEEE Trans. Wireless Commun.*, vol. 16, no. 6, pp. 3787–3798, 2017.
[24] Z. Liu *et al.*, "A 1.8mw perception chip with near-sensor processing scheme for low-power aiot applications," in *Proc. IEEE ISVLSI.*, 2019.
[25] W. Ren, J. Wu, X. Zhang, R. Lai, and L. Chen, "A stochastic model of cascading failure dynamics in communication networks," *IEEE Trans. Circuits Syst. II Express Briefs.*, vol. 65, no. 5, pp. 632–636, 2018.
[26] A. Rachedi and H. Badis, "Badzak: An hybrid architecture based on virtual backbone and software defined network for internet of vehicles," in *Proc IEEE ICC*, 2018.