IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received March 1, 2020, accepted March 16, 2020, date of publication March 24, 2020, date of current version April 7, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.2983092

A Misbehaving-Proof Game Theoretical Selection Approach for Mobile Crowd Sourcing

MENATALLA ABOUOUF[1], (Member, IEEE), HADI OTROK[1], (Senior Member, IEEE), SHAKTI SINGH[1], (Member, IEEE), RABEB MIZOUNI[1], AND ANIS OUALI[2]

[1]Center on Cyber-Physical Systems, EECS Department, Khalifa University, Abu Dhabi, United Arab Emirates
[2]Emirates ICT Innovation Center (EBTIC), Abu Dhabi, United Arab Emirates

Corresponding author: Menatalla Abououf (menatalla.abououf@ku.ac.ae)

This work was supported by the Center on Cyber Physical System as part of Khalifa University Internal Research Fund under Grant 8474000137.

ABSTRACT With the tremendous advances in ubiquitous computing, mobile crowd sourcing (MCS) has become an appealing part of the Internet of Things (IoT). In MCS systems, workers collect data with a certain quality, and get incentivized in return. However, MCS systems are vulnerable to misbehaving acts such as workers submitting multiple false or fake reports using multiple devices to affect the majority vote of the task. In addition, workers may try to maximize their profit by submitting multiple truthful data, a behavior that may prevent other potential workers to participate. The selection of such workers for a task has a negative impact on the decision making or the payoff of the task. Most of the current approaches aim to maximize the completion of tasks based on the reputation or the credibility of workers, but without consideration of tasks' payoff and the threat of misbehavior. In literature, a misbehaving act, where workers impersonate multiple identities using multiple devices to maliciously change the majority votes or selfishly increase their payment, has not been addressed during MCS recruitment. In this paper, a two-layer selection approach is proposed based on game-theory in which the payoff of the tasks is maximized based on the individual contributions of the workers. In addition, the proposed model detects and eliminates the misbehaving act where workers submit multiple reports using multiple devices, during the recruitment phase. Simulations using real-life datasets show that the proposed approach succeeds in detecting and eliminating misbehaving devices, and outperforms the benchmarks in terms of the payoff of the tasks.

INDEX TERMS Mobile crowd sourcing, misbehaving acts, identity management, trust management, game theory, Gale-Shapley.

I. INTRODUCTION

Mobile Crowd Sourcing (MCS) is a paradigm that utilizes the ubiquity, mobility, and distribution of mobile devices in order to collect large amount of data efficiently, in terms of time and cost [1]. Typically, there are three main entities in MCS: task requester, management platform, and mobile users or workers. The management platform recruits workers, or service providers, that ensure the quality of the available tasks' execution. The recruited workers communicate with the management platform upon task's completion, where the submitted data is evaluated and the workers are paid accordingly [2]. In order to enhance the efficiency of MCS in terms of number of completed tasks and their completion time, recent works have introduced multitasking, where workers are selected to perform multiple tasks, especially in a large solution space, where the solution becomes overly complex.

The data collected by the workers depends on the nature of the task. If the task is objective, such as environmental monitoring including noise pollution and nuclear source localization [3], the data is in the form of sensory readings. On the other hand, if the task is subjective, the crowd's opinions regarding a certain phenomenon are requested [4], [5]. An example of subjective tasks could be evaluating and documenting the severity of a fire, crash, or damages after a natural disaster. The most common approach to reach a consensus is the use of majority-voting, where the most frequent answer is considered as the closest answer to the ground truth. However, it becomes challenging to formulate the right decision in the presence of misbehaving workers,

The associate editor coordinating the review of this manuscript and approving it for publication was Yassine Maleh.

which can mislead the requester by submitting multiple false reports. Recently, a similar situation was reported in [6], where using 99 phones and a handcart, a participant was able to trick the Google Maps App and create a virtual traffic jam. This demonstrates the need for a more reliant MCS system that is able to detect and prevent such instances.

MCS is vulnerable to external adversaries' attacks, such as spoofing and jamming, which aim to degrade or crash the service. These can be handled using network's security measures [7]. MCS is also vulnerable to internal misbehaving acts by malicious or selfish workers aiming to degrade the Quality of Service (QoS) of the tasks. These kinds of maliciousness need to be addressed in the workers' recruitment phase. An internal misbehaving act, known as Sybil attack, was first introduced in MCS and IoT by a Microsoft researcher, where multiple identities are used in participatory sensing to contribute a huge amount of false data [8]. This was generally solved by authorization tokens where a device cannot use multiple tokens simultaneously. However, current solutions do not consider the presence of misbehaving workers that accomplish a task numerous times using multiple devices, and their effect on MCS. These workers take advantage of the anonymity, implemented in MCS for privacy protection, where submitted reports cannot be linked to the worker and therefore it cannot be deduced that two or more reports are submitted by the same worker [9]. In addition, they register to the MCS platform using different verified email addresses and personal information, and behave normally, which makes their detection extremely difficult [10].

Additionally, guaranteeing high payoff for the tasks within the constraints and the requirements of the task requester is another critical aspect of MCS. The payoff of a task is defined as the benefit gained in contrast to the costs endured by that task. The benefit can be in the form of the QoS achieved by the task, which is defined differently based on the application. The QoS during selection commonly describes the confidence in the selected workers to truthfully perform the task. The existing allocation approaches, such as [2], [11], and [12] optimize the QoS of the tasks without assessing the contribution that each individual worker adds to the QoS. As a result, the task requesters tend to overpay for the selected workers using these selection models, thus compromising the payoff of the tasks.

To overcome the aforementioned challenges, a novel two-layer selection approach, Misbehavior-Proof Workers Selection (MPWS), is proposed. MPWS aims to select workers that maximize the payoff of the tasks while detecting and eliminating misbehaving devices. In the first layer, groups of workers are allocated to a cluster of tasks using genetic algorithm such that the QoS of each task is maximized. In the second layer, a game-theoretical approach based on Gale-Shapley is deployed to distribute the workers in the group amongst the tasks in the cluster such that the payoff of each task is maximized. The second layer also addresses the issue of the misbehaving act, where a worker uses multiple devices to impersonate multiple identities and maliciously change the majority-voting decision of a task [13], or selfishly maximize their profit. This layer is designed to detect and eliminate such misbehaving devices without invading the privacy of the participating workers. Hence, the main contributions of this work can be summarized as follows:

- Propose a novel selection mechanism that allocates workers with high QoS-to-cost ratio, thus maximizing the payoff of the tasks.
- Introduce and study a misbehaving act where workers use multiple devices to impersonate multiple identities in an MCS system.
- Develop a stand-alone mechanism that detects and eliminates misbehaving devices during the selection process, while keeping the owners of these devices anonymous.

The proposed approach is simulated using real-life datasets under the conditions of both truthful and untruthful environments. In an untruthful environment, workers impersonate multiple identities using multiple devices, whereas in a truthful environment, each device is owned by a unique worker. Two benchmarks are used to assess the performance of MPWS: Group-based Multi-task Workers Selection (GMWS) [11] and Gale-Shapley Matching game Selection (GSMS) [12].

II. BACKGROUND AND RELATED WORK

A. BACKGROUND

The Gale-Shapley matching game is a game theoretical approach that was first introduced by Gale and Shapley for the stable marriage and college admission problems [14]. The stable marriage problem is a one-to-one stable matching between a set of men and a set of women, where the sets are donated by $U = \{m_1, \ldots, m_{n_1}\}$ and $W = \{w_1, \ldots, w_{n_2}\}$, respectively. In addition, there is a set $E \subseteq U \times W$ of acceptable men-women pairs. Each man $m_i \in U$ has an acceptable set of women $A(m_i)$, where $A(m_i) = \{w_j \in W : (m_i, W_j) \in E\}$. Similarly, each woman has an acceptable set of men $A(w_j)$, where $A(w_j) = \{m_i \in U : (m_i, W_j) \in E\}$.

Each player in the game, from U and W, has ordinal preferences where they rank their $A(m_i)$ and $A(w_j)$ in a strict order in their respective preference lists [15]. In the classical stable marriage problem, the set of men is equal to the set of women, i.e. $n_1 = n_2 = n$, and all men are in every woman's acceptable list, and vice versa, i.e $E = U \times M$. The matching is considered unstable if there exists a couple that would rather be paired together over their assigned match [16]. However, if the sets of men and women are not equal, $n_1 \neq n_2$, as is the case in this work, n_1 and n_2 are made equal by simply adding the appropriate number of men or women with empty preference list. A blocking pair, which is a pair that blocks a matching M, is then defined as $(m_i, w_j) \in E \setminus M$, where m_i is unpaired or prefers w_j to its current matching, or w_j is unpaired or prefers m_i to its current pair. In this scenario, the game achieves equilibrium by reaching a stable matching if it encounters no blocking pair [14].

On the other hand, the college admission problem, also known as hospital/residents problem, is a one-to-many stable assignment of students to different colleges based on two-sided preference, given that each college has a limited admission capacity [14]. In this work, one-to-one matching is used, which makes the college admission problem not applicable for this work.

B. RELATED WORK

This section summarizes some of the work done related to the allocation of workers to multiple tasks in MCS and the evaluation of the trustworthiness of the data and the workers.

1) MULTITASK ALLOCATION

Workers allocation is one of the main challenging aspects in MCS since different tasks have different requirements, such as acceptable QoS, reputation, and execution time [3], [17]. In addition, workers in an MCS system have different capabilities such as the devices they are carrying, acceptable traveling distance [18], and reputation [19], [20]. The recruitment of workers is commonly done by two main approaches, group-based selection (GRS) and individual-based selection (IRS). In the former, a group of workers is evaluated and considered for a task, or set of tasks, based on their collective capabilities. In the latter, the workers are assessed individually and are recruited if they meet the tasks' requirements and constraints. GRS has shown superiority when compared with IRS in both single-task and multiple-tasks assignments [11], [21].

Recent research is directed towards multi-worker multitask allocation due to its efficiency in terms of workers utilization [18], especially in subjective tasks. In [11], Group-based Multi-task Workers Selection (GMWS), which is a cluster-based multitasking approach, is proposed. In GMWS, a group of workers is allocated using genetic algorithm to perform all tasks in the cluster, such that the collective QoS of each task is maximized. In [12], a game-theoretical approach based on the Gale-Shapley game is proposed to match multiple workers and tasks while satisfying both sides' preferences. The preferences of the workers depend on their proximity to the tasks, whereas the preferences of the tasks depend on the expected QoS. The QoS for both approaches depend on the willingness of the selected workers to perform the task. However, these approaches do not consider the payment of the worker, and subsequently, do not consider the payoff of the tasks.

Another multi-worker multitasking approach is proposed in [17], where particle swarm optimization is employed to ensure high payoff for tasks while minimizing the response time of the tasks. The approach aims to maximize the number of accomplished tasks by allowing the allocated workers to recommend other workers from their social network to perform the task, if they are unable to. If the task still failed to be accomplished, the allocated workers are penalized by reducing their reputation scores. However, this approach does not consider the contribution of each selected worker to the achieved QoS of the tasks.

There are two modes of workers selection in MCS, opportunistic and participatory. In opportunistic sensing, the workers are allocated to tasks as part of their daily routines, whereas in participatory sensing, the workers are required to travel to the tasks in order to perform them.

In [22], an opportunistic multi-task allocation framework, MTasker, is proposed using a descent greedy approach to maximize the overall utility of each task, while considering a minimum quality constraint for a task, the sensor availability, and the maximum allowed task allocation for a worker. The task's utility is defined by temporal-spatial coverage of the workers. While this approach considers the contribution of allocated worker, it only considers opportunistic workers, where their contribution can be predicted by their mobility pattern. A similar goal of maximizing the spatial-temporal covering is presented in [23]. This is achieved by social-network assisted system for worker recruitment in mobile crowd sensing where influence maximization is used to select seed users.

The work in [24] proposes a framework that optimizes the integration of these two modes by selecting a group of opportunistic workers offline, based on their daily routines, and complement them by online selection of participatory workers, under a constrained budget. The proposed framework, HyTasker, uses a nested-loop greedy process for the offline selection while simultaneously predicting the allocation for the participatory workers.

The aforementioned works mainly focus on optimizing the workers recruitment in MCS. However, the trustworthiness of these workers is not considered.

2) DATA AND WORKERS TRUSTWORTHINESS

Another critical aspect of MCS is examining the trustworthiness of the data. Trustworthiness is associated with the data that the user collects and it is in a direct relationship with the user's reputation [20]. The injection of faulty data has a negative impact on the overall system because they could be misdeemed as correct data.

A common mechanism to assess the validity of submitted reports is by reputation-based systems. In [20], an approach to validate and quantify the data truthfulness is proposed. The approach collaborates statistical and vote-based user reputation scores. Statistical or centralized reputation is calculated by the management platform, whereas vote-based or decentralized reputation is evaluated based on the votes of the participants.

Another approach is estimating the credibility of the submitted data based on calculating the reliability of the worker [25]. This is evaluated based on four factors, (i) similarities between a worker's submitted report in comparison with others submitted at the same time frame, (ii) reliability of the user based on the plausibility of their event reports, (iii) frequency of submitted reports, and (iv) the feedback reported from other users evaluating the data being reported.

Another common approach to checking the trustworthiness of the data is by truth discovery of the submitted data. In [26]–[28], the unreliable sensory data collected by workers is encrypted and sent to the cloud along with the encrypted reliability of the workers. The truth discovery is conducted based on the reliability of the workers on the encrypted data in the cloud to discover the ground truth which is later sent to the task requester for decryption. While in these works the privacy of the workers is considered, however, workers that maliciously submit false data to disrupt the systems are not considered. In [29], privacy-preserved truth discovery mechanism is proposed while considering malicious workers who deliberately aim to disrupt the system. This work updates the reliability and the ground truth and filters out the false data before sensing them to the cloud while keeping the computational costs and communication overheads minimal.

Moreover, cross validating the submitted data is also another common approach to discover the ground truth of a given task. In [30], validating workers are recruited to assess the data collected by sensors and contributing crowd in timely fashion. The collected data from both parties are then consolidated to have a better representation of the ground truth, which is later used to re-evaluate the contribution and the incentives of the participating workers. In [31], data is collected from there different sources depending on their availability, crowdsensed data (S-report), crowdsourced data (U-report) and authoritative data (E-report). If E-reports are available, they are considered the ground truth, otherwise, the average of S-reports ad U-reports are used to discover the ground truth. In [32] A truth discovery algorithm and corresponding reward distribution methods are proposed. The ground truth is estimated based on the aggregated crowdsensed data and partial control data which are obtained from infrastructure IoT sensors. The truth discovery considers both vulnerabilities of the workers and bias of their devices, but does not consider false injected data.

The aforementioned approaches assess the credibility of the workers after the tasks are assumed to be accomplished, and not during the selection phase. In addition, they do not consider misbehaving workers that may manipulate the system by using multiple identities.

In efforts towards suppressing Sybil attacks without improvising the privacy of the participating workers, [9] proposed authorization tokens with lifetime for authenticated devices. These devices receive encrypted credentials, known as pseudonym, to ensure the integrity of submitted reports. A device cannot use more than one pseudonym simultaneously, which prevents Sybil attacks. Moreover, in [8], a trust model is proposed based on a cloud-based service management framework that detects Sybil attack assuming the following: a) abnormal traffic of connected devices is tracked and can be detected, b) each worker owns a unique ID, and c) worker posing Sybil attack submits data using a single radio frequency. However, these approaches do not consider workers who carry multiple devices and the pseudo identities can be manipulated. They can receive multiple tokens for every device and exploit different radio frequency.

III. PROPOSED APPROACH

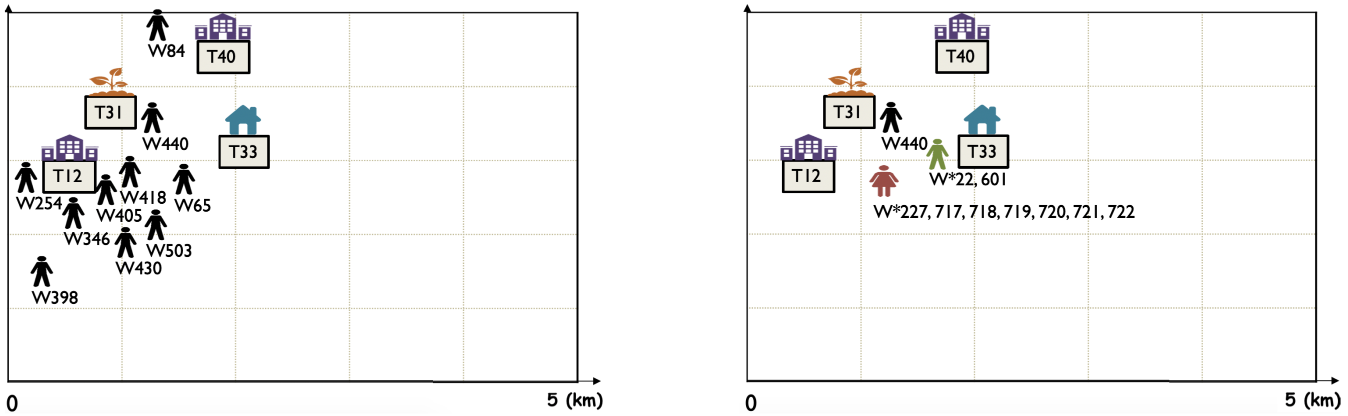
The problem of misbehaving workers carrying multiple devices to maximize their profit and jeopardizing the achieved QoS of the tasks is demonstrated in Section III-A as a motivation for the work. Following, Section III-B proposes a two-layer workers selection approach, MPWS, to detect and eliminate those untruthful workers whilst achieving high payoff for the tasks. The first layer uses a meta-heuristic approach to select a group of workers that guarantees the highest expected QoS for a cluster of tasks. The second layer aims to distribute the group of workers among the cluster's tasks such that untruthful workers are excluded and the payoff for the individual tasks is maximized.

A. MOTIVATIONAL SCENARIO

In this section, the allocation problem simulated in an environment where all participants are truthful versus an environment where some misbehave, is illustrated. The simulation was performed for 40 tasks and 600 available workers, where Group-based Multi-task Workers Selection (GMWS) [11] is used for the selection. GMWS is a multitasking approach where selected workers are requested to perform multiple tasks, such that the QoS of each task is maximized. This selection approach clusters the geographically-close tasks and allocates a group of workers to each cluster to perform the tasks in that cluster. In this example, a snapshot of the final selection is illustrated, for a cluster of 4 spatial-temporal tasks and 10 workers that are selected to perform them.

In both environments, the same dataset of independent devices is used. For the untruthful environment 10% of random workers are duplicated to simulate impersonating multiple identities. Fig. 1a shows 10 different individuals selected in a truthful environment, each owning exactly one device. On the other hand, Fig. 1b shows the selection of 10 devices belonging to only 3 different owners, where the first worker uses exactly one device the second carries two selected devices and the last carries seven selected devices. Hence, in this scenario two workers are misbehaving by using multiple devices.

In GMWS, workers that maximize the QoS of the tasks are selected and the tasks are scheduled such that it minimizes the traveling distance. In this example, the collective QoS achieved by the tasks depends on their initial reputation before attempting the task, the willingness of the worker in attempting the task and the completion time, these parameters are further explained in section III-B. It can be seen from Fig 1 that except $W440$, none of the other behaving workers that were selected in the truthful environment (Fig 1a) are selected in the untruthful environment (Fig 1b). This is because the presence of misbehaving devices in an untruthful environment may prevent other potential workers from being selected, if the expected QoS of misbehaving devices is higher than that of behaving devices.



(a) Truthful environment.

(b) Untruthful environment.

FIGURE 1. Illustrative example for a simplified MCS allocation problem in truthful and untruthful environments.

TABLE 1. Expected vs. Achieved QoS by each task in truthful and untruthful environments.

	Truthful environment	Untruthful environment	
Task	Achieved QoS	Expected QoS	Achieved QoS
T12	0.995	0.996	0.809
T31	0.992	0.993	0.770
T33	0.996	0.998	0.871
T40	0.997	0.998	0.846

TABLE 2. Total payments in truthful and untruthful environments.

Truthful environment	Untruthful environment	
Payment	Truthful payment	Untruthful payment
112.46	58.44	61.86

TABLE 3. List of parameters and symbols used and definitions.

Symbols	Definition
L_j^T	Location of task j
S_j^T	Set of sensors required by task j
Q_j^T	Minimum QoS requirement of task j
TC_j	The time constraint for task j
R_j^T	Set of acceptable reputation thresholds j
P_j^T	Set of acceptable maximum payments requests corresponding to each R_j^T
L_i^W	Location of worker i
SA_i^W	Set of available sensors on the device of worker i
R_i^W	Reputation of worker i based on previously assigned tasks
C_i^W	Confidence in worker i to complete task j based on their their historical mobility
D_i^W	Maximum traveling distance constraint set by worker i
P_i^W	Payment requested by worker i per traveled kilometer
R_i^{CT}	Reputation of worker i relative to a culser CT
QoS_i^{CT}	QoS achieved by worker i relative to cluster CT
T_i^j	Time needed for worker i to reach task j
QoS^{CT}	Collective QoS achieved by group g relative to cluster CT
k	Final assigned workers performing a task, where $k \subseteq g$
QoS^j	Collective QoS achieved by assigned workers k relative to task j
TP^j	Total payment by task j to assigned workers k
d_i^j	Euclidean distance between worker i and task j

The QoS of each task in a truthful environment is compared against that in an untruthful environment in Table 1. The expected QoS is evaluated based on the selected workers without the detection of the duplicate misbehaving devices, on the other hand the achieved QoS is computed while considering only unique entities. As evident from Table 1, the achieved QoS based on the selection is up to $\sim$ 22% less than the expected QoS by the tasks in the untruthful environment.

As the tasks request multiple workers, only three unique reports are submitted in the untruthful environment in this example, which degrades the achieved QoS of the tasks. In addition, since the misbehaving workers are not detected, they are getting paid for performing the task. Table 2 shows that in the case of the untruthful environment, $\sim$ 51% of the total cost of the tasks is given to untruthful devices.

Hence, as shown by this example, a novel approach is required to detect and eliminate untruthful devices to realize the achieved QoS, and to prevent misbehaving workers from maximizing their profit and reducing the chances of behaving devices to participate. In addition, false reports submitted by multiple-identity workers have graver implications on the final decision making of the task requester. It is worth noting that this scenario is viable in any selection mechanism where tasks must be performed by multiple workers, which is mostly the case for quality assurance or for collecting different opinions. If a mobile user initiating multiple identities is a good candidate for selection, then most probably their other identities are also good candidates. This is because the other devices share similar profiles and same location.

B. MODEL DESCRIPTION

In this model, each task publisher requests multiple workers to perform the task. Furthermore, every worker can perform multiple tasks, even for those posted by different publishers. The tasks are characterized by $T =< L_j^T, S_j^T, Q_j^T, TC_j^T, R_j^T, P_j^T >$, and the workers are characterized by $W =< L_i^W, SA_i^W, R_i^W, C_i^W, D_i^W, P_i^W >$, where the different parameters are defined in Table 3.

C. FIRST LAYER: GROUP SELECTION FOR A CLUSTER OF TASKS

This layer maximizes the QoS of the tasks while minimizing their completion time. This is achieved by - 1) clustering the tasks based on their geographic locations using k-medoids algorithm, 2) selecting a group of workers that maximizes the expected QoS of the tasks in a cluster using genetic algorithm, and 3) scheduling the tasks for the workers such that the traveling distance is minimized using tabu search algorithm [11].

For the clustering of tasks, the Silhouette evaluation [33] is used to determine the number of clusters CT such that the distances between the individual tasks in a cluster is minimized. The number of clusters that achieves the highest separation score, is used by the k-medoids clustering algorithm [34] to cluster the tasks in a given Area of Interest (AoI). K-mediods is a partitioning algorithm that minimizes the distance among nodes in a cluster. This gives it the advantage over other partitioning agorithms, such as k-means, which minimizes the distance between the nodes and a cluster center [34], or the distance-dependent Chinese restaurant process, which minimizes the distance between any two nodes in a cluster [35], [36].

Subsequently, for every CT, a group of workers is assigned such that every member added to the group must have an added value and must not violate any of the constraints of the tasks in the corresponding cluster. The groups of workers are formed using genetic algorithm [21], starting with a group of one member, and gradually increasing until the maximum number of required workers is reached. The group that achieved the highest expected QoS for the cluster is selected. The expected QoS of a worker relative to a cluster QoS_i^{CT} is evaluated as:

$$QoS_i^{CT} = R_i^{CT} \times C_i^W \times \tau_i^{CT} \tag{1}$$

where R_i^{CT} is the reputation of the worker relative to the minimum reputation required by the tasks in the cluster, i.e. if $R_i^W \geq max(R_j^T)\ \forall j \in CT$, then $R_i^{CT} = R_i^W$, otherwise, $R_i^{CT} = 0$. τ_i^{CT} is a decreasing function with the increase in the worker's traveling time to the cluster, defined as [11]:

$$\tau_i^{CT} = \min_{j \in CT}[1 - \max(0, \min[\log_{TC_j^T}(T_i^j), 1])] \tag{2}$$

where T_i^j is the time taken in seconds for worker i to reach task j. The collective QoS of a group of workers g relative to a cluster CT, $QoS^{CT}(g)$, is evaluated as the probability of receiving at least one successful reading following the binomial distribution as:

$$QoS^{CT}(g) = 1 - \prod_{i=1}^{g}[1 - QoS_i^{CT}] \tag{3}$$

Finally, the tasks in a cluster are scheduled using tabu search algorithm [37] for each member, such that the traveling distance for the worker is minimized in order to minimize the total payment. In this layer, all the group members are expected to perform all the tasks in the cluster they are allocated to. While every member adds a value to their assigned cluster, the added value to each individual task varies.

It may be considered a loss to some tasks in a cluster to employ a member of the assigned group if their added value is minimal compared to their cost, as this reduces the payoff of the task. In addition, since the tasks are scheduled for every $i \in g$ based on the shortest distance and the spatial location of the tasks, if there exists an untruthful worker in the group holding multiple devices, i.e. a subgroup within g is one individual, this worker will have the same schedule for all their devices and will not be detected. This in turn contributes to: 1) misbehaving devices selfishly increasing their profit, while preventing potential behaving devices from being selected, 2) misbehaving devices maliciously steering the decision making of the task requester by reporting the same information multiple times, and 3) false calculation of the actual achieved QoS for the tasks in the cluster, as demonstrated in Section III-A.

D. SECOND LAYER: GALE-SHAPLEY MATCHING GAME SELECTION

To overcome the challenges presented by the first layer, a second layer based on Gale-Shapley matching game is proposed. Gale-Shapley is game-theoretical bipartite matching problem that considers two-sided preferences with complexity of $O(n^2)$. Game-theory is a strategic interaction among rational parties. In this work, we assume that all the players are rational by trying to increase their payoff. In the original one-to-one Gale-Shapley [14], also known as the stable marriage problem, a set of men and a set of women are matched in pairs based on their preset preferences, such that there is no unmatched pair that would rather be matched together. For this work, the Gale-Shapley matching is considered since other matching algorithms, such as the Hungarian maximum matching algorithm [38], do not consider two-sided preferences and/or have higher computational complexity. Hence, playing this game will ensure that the workers and the tasks will list their preferences appropriately, otherwise they will not be matched.

In this work, the game is adapted to many one-to-one matching to allow multitasking, where the selection is divided into multiple rounds of one-to-one matching in which misbehaving workers are detected. A round is defined as a time slot in which at least one task-worker assignment is made. Hence, the assigned workers should finish the given tasks within the round time, considering that the task's completion time depends on the traveling time of the worker to the task. The members in the selected group from the first layer g and the tasks in cluster CT are matched using the adapted Gale-Shapley.

Each worker, $i \in g$, creates a *preference list* by ranking the tasks $j \in CT$ as the resulting order of the tabu search algorithm, since this is the shortest path a worker can perform the tasks in. On the other hand, each $j \in CT$ creates a *preference list* by ranking the workers $i \in g$ in the descending order of their Shapley value [39] relative to the task. The Shapley value is proportional to the contribution in QoS of each worker to each task. The created *preference lists* include all group members and all the tasks in the cluster, since the group formed by the first layer conforms with all the tasks constraints and vice versa. Using the *preference lists* by both workers and tasks, the stable assignments are made in each round, where a worker can be assigned to at most one task, and a task can be allocated to only one worker. Using this technique, if there exists some misbehaving workers carrying multiple devices, even though all their devices will have the same *preference lists*, they cannot co-exist performing the same task at the same time, making it impossible for such behavior to go undetected. In other words, workers multitask but in a specific order at a specific time period. If a worker is misbehaving using multiple devices, then they most probably will receive the same order for all their devices, since their devices will share the same location. By design, in the proposed approach, the devices will receive the order of the tasks with a certain delay, defined by the time period of the round, which will make it necessary to be at multiple locations at the same time.

As a result, misbehaving workers will not be able to complete all the assigned tasks using all their devices, and consequently the reputation of their devices will drop. The reputation here is evaluated as:

$$R_i^W = \alpha R_i^W + (1-\alpha)\frac{Completed\ tasks}{Total\ assigned\ tasks} \tag{4}$$

where the updated R_i^W of the worker depends on the weighted sum of the previous reputation and the ratio of completed to assigned tasks from that cluster. Equation (4) is also used within the rounds, where the number of completed and assigned tasks are updated to anticipate the final reputation. If the round finished and an allocated task was not completed, the corresponding device's reputation drops. Since each task has a set of acceptable payments (P_j^T) corresponding to minimum reputation thresholds (R_j^T), the device whose reputation drops below $max(R_j^T)$ for $j \in CT$, or becomes low compared to the requested payment, is eliminated within the rounds. This also gives chances to truthful devices to get assigned.

In addition, since the matching is based on preferences for both tasks and workers, if $i \in g$ are all truthful independent devices, the workers selected for each task will be the workers contributing the most to the QoS of that task, i.e. have the highest Shapley value for that task. This means that it is possible that a worker $i \in g$ is assigned to some or all the tasks $j \in CT$. However, it cannot happen that a worker is not assigned to any task, or a task is not assigned any worker. This is because each member in the group formed by the first layer has a high contribution to the QoS of at least one of the tasks $j \in CT$. Subsequently, the payoff of the tasks will increase as the drop in the cost for the task will be much greater than the drop in the QoS. The payoff of a task is defined as the QoS achieved by the task over the cost of assigned workers, and is computed for a task j as:

TABLE 4. Final assignments for tasks and workers in different rounds.

W	r1	r2	r3	r4	r5	r6	r7	r8	r9	r10
601	40	-	-	-	-	-	-	33	31	12
22	-	40	-	-	-	-	-	-	33	31
440	33	-	-	-	-	-	-	40	-	-
227	31	33	40	12	-	-	-	-	-	-
717	-	31	33	40	12	-	-	-	-	-
718	-	-	31	-	-	-	-	-	-	-
719	-	-	-	33	-	-	-	-	-	-
720	-	-	-	-	33	40	31	12	-	-
721	-	-	-	-	-	33	40	31	12	-
722	-	-	-	-	-	-	33	-	-	-

$$Payoff^j = \frac{QoS^j}{TP^j} \tag{5}$$

where QoS^j is computed using (3) for final assigned members k, where $k \subseteq g$. TP^j is the total payment by task j to assigned workers and is computed as:

$$TP^j = \sum_{i \in k} d_i^j P_i^W \tag{6}$$

where d_i^j is the distance, in kilometer, between worker's current location L_i^W and task's location L_j^T.

Revisiting the example of the untruthful environment in Section III-A, a group of workers are selected by the first layer as shown in Figure 1b. Taking the 7 untruthful devices $W227, 717, 718 - 722$ as an example, their *preference lists* are $T < 33, 40, 31, 12 >$. For illustration purposes, it is assumed that all these devices start with the same reputation value 0.663 and the tasks' required minimum reputation is set to 0.6.

Table 4 shows the final assignments using the second layer, many one-to-one Gale-Shapley, divided into 10 rounds.

It can be seen from this example that $W227$ and $W717$ got the same order of assignments but starting from different rounds, and the same would have been true for the remaining misbehaving devices of this owner. However, due to the inability of the owner to be at 3 different task locations at the same time in $r3$, device $W718$ could not complete the task, hence its reputation dropped below 0.6 and got eliminated from further assignments. The same applies to devices $W719 - 722$. It is worth noting that by using this technique misbehaving devices are prevented from performing all the tasks.

E. OVERALL APPROACH

The overall proposed algorithm is presented in Fig. 2, and is detailed as follows:

Step 1) - **Line 1:** The dataset of the workers population and the tasks are used as input. This will be later discussed in Section IV.

Step 2) **Lines 2 to 3:** The number of clusters CT that minimizes the separation scores between the tasks in an AoI is evaluated using the Silhouette evaluation. K-medoid

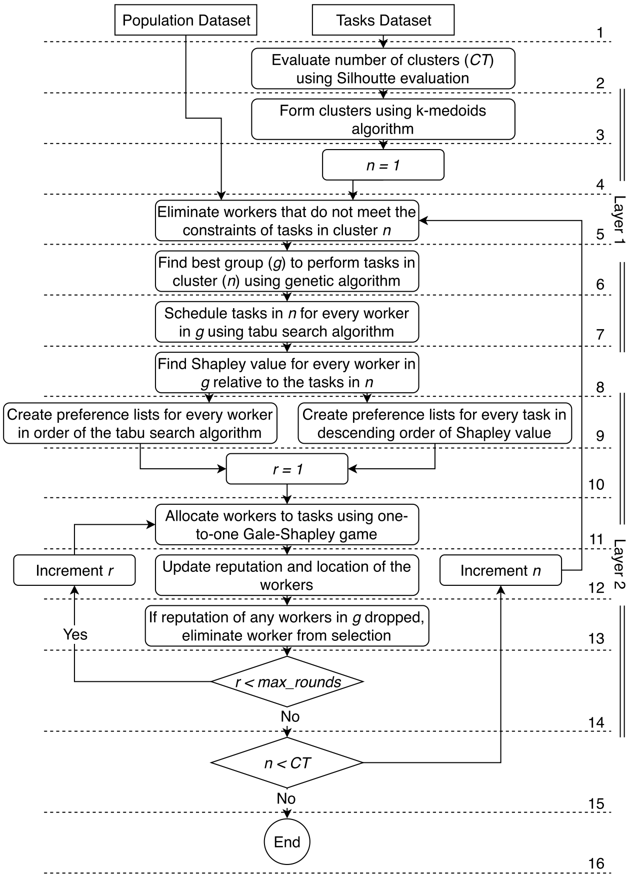

Population Dataset
Tasks Dataset
Evaluate number of clusters (CT) using Silhoutte evaluation
Form clusters using k-medoids algorithm
n = 1
Eliminate workers that do not meet the constraints of tasks in cluster n
Find best group (g) to perform tasks in cluster (n) using genetic algorithm
Schedule tasks in n for every worker in g using tabu search algorithm
Find Shapley value for every worker in g relative to the tasks in n
Create preference lists for every worker in order of the tabu search algorithm
Create preference lists for every task in descending order of Shapley value
r = 1
Allocate workers to tasks using one-to-one Gale-Shapley game
Increment r
Update reputation and location of the workers
Increment n
If reputation of any workers in g dropped, eliminate worker from selection
Yes
r < max_rounds
No
n < CT
No
End
Layer 1
Layer 2
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16

FIGURE 2. Overall approach for the proposed two-layer selection model.

clustering algorithm is then used to cluster the tasks into CT clusters.

Step 3) **Lines 4 to 7:** For every cluster of tasks, the population dataset is filtered for qualified workers, where genetic algorithm is employed to find the best group $g \in W$ of workers that maximizes the expected QoS for the tasks in cluster $n \in CT$. Tabu search algorithm is then used to find the optimal path for every worker in $g \in W$ that minimizes the traveling time for the worker and the completion time of the tasks.

Step 4) **Lines 8 to 9:** Every task creates a *preference list* of the workers based on the descending order of their contributions assessed by the Shapley value. In addition, every worker in $g \in W$ creates a *preference list* of the tasks in the order of the shortest path discovered in step 3.

Step 5) **Lines 10 to 14:** Based on the *preference lists* created in step 4, the adapted Gale-Shapley game is used to match each worker in $g \in W$ to tasks in $n \in CT$ in different rounds. At each round, if the worker performs the matched task, their reputation increases, otherwise it decreases. If their reputation decreases below the minimum reputation requirement, that worker is considered ineligible for further selection. This process is repeated until the maximum number of rounds is reached. The maximum number of rounds is set to the number of workers in the group, this is to ensure that every worker gets the chance to be assigned even in the presence of misbehaving workers.

Step 6) The QoS of each task in $n \in CT$ is calculated using (3), and the final reputation of the participating workers are updated using (4). In addition, the payment for each worker for their completed tasks is calculated using (6). Finally, steps 2-6 are repeated for all clusters $n \in CT$.

IV. SIMULATION RESULTS

In this section, the proposed two-layer misbehavior-proof workers selection (MPWS) approach is simulated and evaluated in truthful and untruthful environments, to prove its robustness. The former is the case where all participants are assumed to be truthful, i.e. every participant owns one device, while the latter is simulated by randomly selecting 10% of the workers to be misbehaving and duplicating their entry 1-9 times, i.e. every misbehaving participant owns 1-9 extra devices. The proposed mechanism is also compared with two multi-worker multi-task approaches, i.e. group- based Group-based Multi-task Workers Selection (GMWS) [11] and Gale-Shapley matching game selection (GSMS) [12]. In all simulations, a Dell Intel Xeon workstation equipped with 256 GB RAM and 300 GB hard disk is used.

The effect of varying the number of available tasks while fixing the number and locations of 600 participants is evaluated. The different number of tasks increases gradually from set of 10 to 100 in an AoI of size 5 × 5 *km*, by randomly generating 10 tasks and adding them to the existing dataset. For all approaches, the maximum number of workers is set to 10, and a number of 10 simulations for each set of tasks is taken to average out the results. In the case of truthful environment, the 10 simulations are independent of each other; a different set of tasks are generated, and the dataset of the population is reset. To demonstrate detection and elimination in the untruthful environment (10% injected misbehaving devices), the same set of tasks are repeated for all 10 simulations, while resetting the locations of the participants in the population. For a fair comparison, all three approaches use the same dataset for population and tasks, and the same equations (1) - (6).

The *ID* and the *latitude-longitude* coordinates of the participants in the population are obtained from Sarwat Foursquare dataset, which is a social networking application containing data about users, their social connections, check-ins and venues [40], [41]. The data is filtered for those who are within the studied AoI. *Reputation* is obtained from a real-life dataset, the Stack Exchange Data Dump.[1] The remaining attributes, which include: *Sensor Availability*, *Distance Constraint*, *Confidence*, and *Payment Requested* are generated following a uniform distribution, where *Payment Requested* is between 1 to 12 *unit*/*km*. The tasks' dataset consists of: *ID*, *Longitude-Latitude* coordinates, *Required Sensors*, *Minimum QoS*, and *Time Constraint*. For the tasks' minimum reputation thresholds: $R_j^T \geq 0.75$, $0.5 \leq R_j^T < 0.75$, and

[1] https://archive.org/details/stackexchange

$R_j^T < 0.5$, the corresponding maximum allowed payments are set to 12, 7, and 0, respectively.

A. BENCHMARK

1) GROUP-BASED MULTI-TASK MULTI-WORKER SELECTION (GMWS)

In this approach, tasks are clustered based on the geographic location and a group of workers is allocated for every cluster, where every worker is requested to complete all the tasks in the cluster. The tasks are scheduled for every worker in the group in order to minimize the total distance traveled. Since the aim of the approach is to minimize the completion time and maximize the QoS achieved by the tasks, there is a higher emphasis on the traveling time in the QoS equation. To fairly compare with the proposed approach, the selection mechanism is adapted to use the same equations used in Section III. In addition, the notion of workers payment is added to the approach with the same mechanism to not select workers that request payments higher than their reputation, as proposed.

2) GALE-SHAPLEY MATCHING GAME SELECTION (GSMS)

In this approach, Gale-Shapley game is used to match tasks with workers to maximize the user satisfaction. This is done by allocating workers to their most preferred tasks while maximizing the tasks' QoS. The approach defines confidence as the degree at which the workers' preferences are met:

$$C_{i,r}^{j} = 1/rank_{i,r}(t_j) \tag{7}$$

where $rank_{i,r}$ is the rank task j in worker i's *preference list* at round r. This is adapted to the definition of confidence in the proposed approach, which is the willingness of the worker in attempting the task, given their historical mobility. In addition, GSMS proposes multi-task multi-worker assignment by performing the many one-to-many Gale-Shapley game. This is also adapted to the one-to-one game many times as is the case of the proposed approach. Finally, the notion of payment is also added.

B. EVALUATION

1) TRUTHFUL ENVIRONMENT

Fig. 3 shows comparison of the proposed approach (MPWS), GMWS, and GSMS for the total QoS, in a truthful environment. The total QoS is computed by summing the QoS achieved by all available tasks.

It can be seen that GMWS and GSMS outperform MPWS in terms of total QoS by a maximum of 1.2% and 1.8%, respectively. This is expected since in GMWS, all members of the group are requested to perform all the tasks in the assigned cluster, where each member contribute to the QoS of each task even if this contribution is minimal. Whereas, GSMS aims to allocate the maximum number of workers requested based on the preferences of both workers and tasks. On the other hand, MPWS distribute these members of the group among the tasks in the cluster according to the contribution of the members to the tasks and the workers' preferences. Thus, members with relatively low contributions to a specific task are discarded from this task's allocated group.

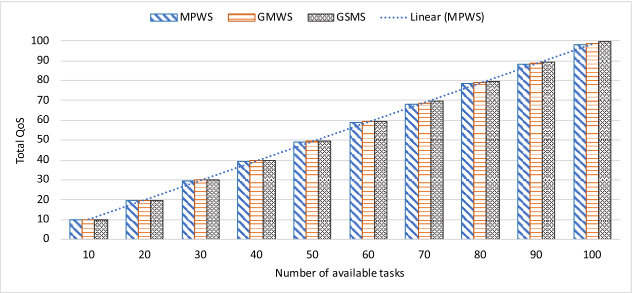


FIGURE 3. The total QoS achieved by all tasks for varying number of tasks.

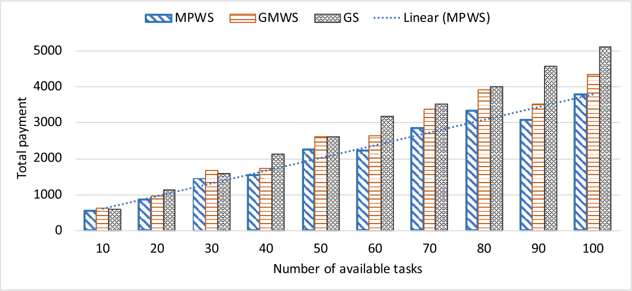


FIGURE 4. The total payment by all tasks for varying number of available tasks.

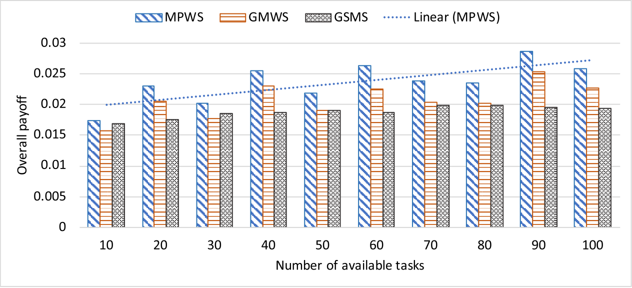


FIGURE 5. The overall payoff of all tasks for varying number of available tasks.

In spite of the minimal drop in QoS by MPWS, it has the advantage when it comes to the cost endured by tasks in workers selection. Fig. 4 shows the comparison of the total payment by the tasks for all three approaches. It is evident that MPWS decreased total workers payment by up to 15% and 32% when compared with GMWS and GSMS, respectively. It also shows that as the number of available tasks increases, the difference between the approaches becomes more significant. This shows that MPWS performs better even as the workers selection process gets more complicated.

The significance of reducing the cost for the tasks is further illustrated by comparing the overall payoff of the tasks, which is defined as the total QoS achieved by tasks over the total payment by the tasks. According to Fig. 5, the overall payoff of the tasks using MPWS outperforms GMWS and GSMS by

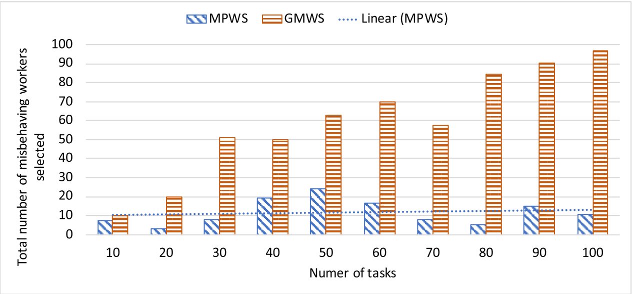


FIGURE 6. The total misbehaving workers selected for varying number of tasks.

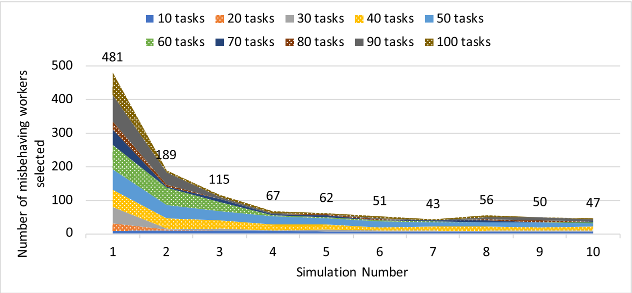


FIGURE 7. Total number of misbehaving workers selected vs. the simulation number. The graph is a stack of the varying number of tasks, i.e. 10-100 tasks, for each of the 10 simulations. The width of each line represents the total for the corresponding number of tasks.

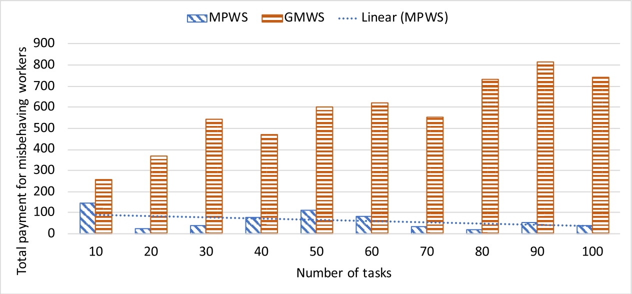


FIGURE 8. The total payment for misbehaving devices for varying number of tasks.

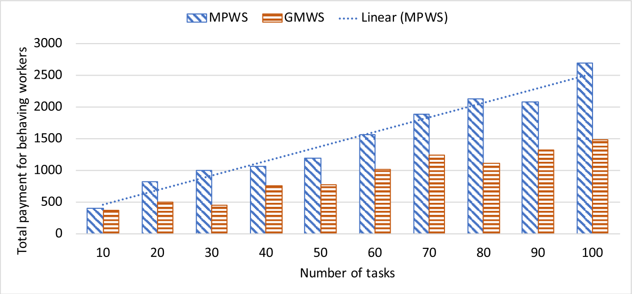


FIGURE 9. The total payment for behaving devices for varying number of tasks.

up to 17% and 46%, respectively, thus showing the efficacy of the proposed approach.

2) UNTRUTHFUL ENVIRONMENT

In this section the proposed approach is compared against GMWS in the case of untruthful environment. Whilst the duplicate devices for the same participant have different ID numbers, they share the same location and start with the same reputation score. The reputation of each individual device is then updated independently, depending on the completion of the task using that device.

To illustrate how the proposed approach detects and eliminates misbehaving workers, Fig. 6 shows the total number of misbehaving devices selected in the groups. This is computed as the sum of all misbehaving devices in the selected groups averaged out in 10 simulations for each set of tasks.

The proposed approach outperforms GMWS by up to 94% in terms of not selecting misbehaving workers. This is because throughout the simulations, MPWS detects the misbehaving devices and reduces their reputation until the reputation of the device is no longer eligible to be selected. On the other hand, the GMWS approach is unable to detect these devices, thus including it in every selection and increasing their reputation. Fig. 7 illustrates what happens in between 10 sequential simulations for the total number of misbehaving devices selected for varying number of tasks. The tasks are increasing bottom-up from 10-100 tasks, and the numbers demonstrate the total devices selected for the stack at every simulation. It is evident that the number of misbehaving devices drops significantly ($\sim$ 61%) just between the first (481) and second simulation (189), and then keep decreasing as simulations progress, thus demonstrating fast recovery of MPWS after detecting misbehaving devices.

Furthermore, in MPWS, if a misbehaving device is selected and fails to perform the task, that device does not get paid. Fig. 8 shows a comparison of the total payments given to misbehaving workers for MPWS and GMWS. MPWS decreases total payment for untruthful devices by 95% when compared with GMWS. This proves that even though untruthful devices are initially selected, MPWS makes it challenging for participants to submit duplicate reports using multiple devices, hence increasing the payoff of the tasks and decreasing the profit for the misbehaving workers. On the contrary, in GMWS, when duplicate reports are submitted by multiple devices owner, each device gets paid, and unrealistic QoS is achieved.

Fig. 9 shows a comparison of payment given to truthful workers. It can be seen that MPWS demonstrate higher payment for behaving workers by up to 119%, since MPWS eliminates misbehaving devices from the selection process throughout iterations, which increases the inclusion of behaving workers. On the other hand, the re-selection of misbehaving devices in GMWS prevents some of the potential behaving devices from getting the chance to participate in the MCS system, thus demonstrating lower payment for behaving workers.

Finally, Fig. 10 shows a stacked graph of the percentage paid to untruthful devices out of the total payments by the tasks as simulations progress. The maximum untruthful

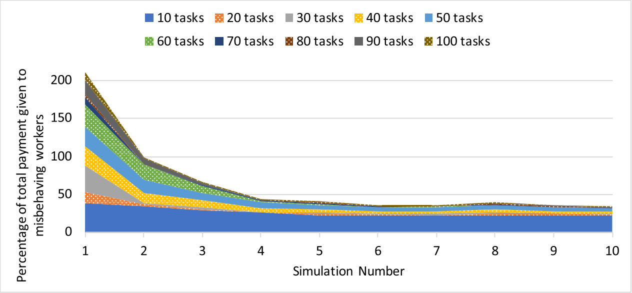


FIGURE 10. The percentage of total payment given to misbehaving workers vs. the simulation number. The graph is a stack of the varying number of tasks, i.e. 10-100 tasks, for each of the 10 simulations. The width of each line represents the percentage for the corresponding number of tasks.

paid percentage reaches up to 40% at simulation number 1, and it reduces to 0% from simulation number 3. Moreover, the proposed approach performs even better when the number of tasks is higher, since this decreases the probability of untruthful devices to complete tasks, thus making it easier to detect and eliminate them.

V. CONCLUSION

In this work, a two-layer misbehavior-proof selection approach (MPWS) is proposed. In the first layer, a group of workers is selected for a cluster of tasks to maximize the QoS of the tasks. Whereas in the second layer, a game-theoretical approach is proposed to enhance the payoff of the task requesters and to overcome the misbehaving act of impersonating multiple identities using multiple devices. It is worth noting that the second layer is independent of the first layer, and can be used as an extension to any multi-worker multitasking selection approach. The proposed approach was evaluated and compared with two benchmarks, GMWS and GSMS. The simulations results demonstrate that MPWS decreases the total payment for tasks by up to 15% and 32% when compared with GMWS and GSMS, respectively. In addition, the payoff for the tasks is increased by 17% and 46% when compared with GMWS and GSMS, respectively. The results also show that MPWS decreased selection of misbehaving devices by up to 94% and their payment by up to 95% when compared with GMWS, thus showing the efficacy of the proposed approach. For future work, the problem of multiple identities using multiple devices will be addressed in opportunistic crowd sensing scheme. In opportunistic sensing real-world mobility datasets will be used to predict the availability of the workers in the locations of the allocated tasks to further enhance the confidence of these workers in performing the tasks.

REFERENCES

[1] Z. Rahman, S. P. Mattingly, R. Kawadgave, D. Nostikasari, N. Roeglin, C. Casey, and T. Johnson, "Using crowd sourcing to locate and characterize conflicts for vulnerable modes," *Accident Anal. Prevention*, vol. 128, pp. 32–39, 2019.

[2] L. Pu, X. Chen, J. Xu, and X. Fu, "Crowd foraging: A QoS-oriented self-organized mobile crowdsourcing framework over opportunistic networks," *IEEE J. Sel. Areas Commun.*, vol. 35, no. 4, pp. 848–862, Apr. 2017.

[3] A. Alagha, S. Singh, R. Mizouni, A. Ouali, and H. Otrok, "Data-driven dynamic active node selection for event localization in IoT applications—A case study of radiation localization," *IEEE Access*, vol. 7, pp. 16168–16183, 2019.

[4] Z. Xu, L. Mei, K.-K.-R. Choo, Z. Lv, C. Hu, X. Luo, and Y. Liu, "Mobile crowd sensing of human-like intelligence using social sensors: A survey," *Neurocomputing*, vol. 279, pp. 3–10, Mar. 2018.

[5] S. Castano, A. Ferrara, L. Genta, and S. Montanelli, "Combining crowd consensus and user trustworthiness for managing collective tasks," *Future Gener. Comput. Syst.*, vol. 54, pp. 378–388, Jan. 2016.

[6] *Man Uses 99 Phones and a Handcart to Create a 'Virtual Traffic Jam' on Google Maps*. Accessed: Feb. 25, 2019. [Online]. Available: https://interestingengineering.com/man-uses-99-phones-and-a-handcart-to-create-a-virtual-traffic-jam-on-google-maps

[7] L. Xiao, D. Jiang, D. Xu, and N. An, "Secure mobile crowdsensing with deep learning," 2018, *arXiv:1801.07379*. [Online]. Available: http://arxiv.org/abs/1801.07379

[8] S.-H. Chang and Z.-R. Chen, "Protecting mobile crowd sensing against Sybil attacks using cloud based trust management system," *Mobile Inf. Syst.*, vol. 2016, May 2016, Art. no. 6506341.

[9] S. Gisdakis, T. Giannetsos, and P. Papadimitratos, "Security, privacy, and incentive provision for mobile crowd sensing systems," *IEEE Internet Things J.*, vol. 3, no. 5, pp. 839–853, Oct. 2016.

[10] K. Zhang, X. Liang, R. Lu, and X. Shen, "Sybil attacks and their defenses in the Internet of Things," *IEEE Internet Things J.*, vol. 1, no. 5, pp. 372–383, Oct. 2014.

[11] M. Abououf, R. Mizouni, S. Singh, H. Otrok, and A. Ouali, "Multi-worker multi-task selection framework in mobile crowd sourcing," *J. Netw. Comput. Appl.*, vol. 130, pp. 52–62, Mar. 2019.

[12] M. Abououf, S. Singh, H. Otrok, R. Mizouni, and A. Ouali, "Gale-shapley matching game selection—A framework for user satisfaction," *IEEE Access*, vol. 7, pp. 3694–3703, 2019.

[13] L. Xiao, D. Jiang, D. Xu, W. Su, N. An, and D. Wang, "Secure mobile crowdsensing based on deep learning," *China Commun.*, vol. 15, no. 10, pp. 1–11, Oct. 2018.

[14] D. F. Manlove, *Algorithmics of Matching Under Preferences*, vol. 2. Singapore: World Scientific, 2013.

[15] D. Gale and L. S. Shapley, "College admissions and the stability of marriage," *Amer. Math. Monthly*, vol. 69, no. 1, pp. 9–15, 1962.

[16] D. Gusfield and R. W. Irving, *The Stable Marriage Problem: Structure and Algorithms*. Cambridge, MA, USA: MIT Press, 1989.

[17] R. Estrada, R. Mizouni, H. Otrok, A. Ouali, and J. Bentahar, "A crowd-sensing framework for allocation of time-constrained and location-based tasks," *IEEE Trans. Services Comput.*, early access, Jul. 11, 2017, doi: 10.1109/TSC.2017.2725835.

[18] B. Guo, Y. Liu, W. Wu, Z. Yu, and Q. Han, "ActiveCrowd: A framework for optimized multitask allocation in mobile crowdsensing systems," *IEEE Trans. Human-Mach. Syst.*, vol. 47, no. 3, pp. 392–403, Jun. 2017.

[19] R. Azzam, R. Mizouni, H. Otrok, S. Singh, and A. Ouali, "A stability-based group recruitment system for continuous mobile crowd sensing," *Comput. Commun.*, vol. 119, pp. 1–14, Apr. 2018.

[20] M. Pouryazdan, B. Kantarci, T. Soyata, L. Foschini, and H. Song, "Quantifying user reputation scores, data trustworthiness, and user incentives in mobile crowd-sensing," *IEEE Access*, vol. 5, pp. 1382–1397, 2017.

[21] R. Azzam, R. Mizouni, H. Otrok, A. Ouali, and S. Singh, "GRS: A group-based recruitment system for mobile crowd sensing," *J. Netw. Comput. Appl.*, vol. 72, pp. 38–50, Sep. 2016.

[22] J. Wang, Y. Wang, D. Zhang, F. Wang, H. Xiong, C. Chen, Q. Lv, and Z. Qiu, "Multi-task allocation in mobile crowd sensing with individual task quality assurance," *IEEE Trans. Mobile Comput.*, vol. 17, no. 9, pp. 2101–2113, Sep. 2018.

[23] J. Wang, F. Wang, Y. Wang, D. Zhang, L. Wang, and Z. Qiu, "Social-network-assisted worker recruitment in mobile crowd sensing," *IEEE Trans. Mobile Comput.*, vol. 18, no. 7, pp. 1661–1673, Jul. 2018.

[24] J. Wang, F. Wang, Y. Wang, L. Wang, Z. Qiu, D. Zhang, B. Guo, and Q. Lv, "HyTasker: Hybrid task allocation in mobile crowd sensing," *IEEE Trans. Mobile Comput.*, vol. 19, no. 3, pp. 598–611, Mar. 2019.

[25] A. Salamanis, A. Drosou, D. Michalopoulos, D. Kehagias, and D. Tzovaras, "A probabilistic framework for the reliability assessment of crowd sourcing urban traffic reports," *Transp. Res. Procedia*, vol. 14, pp. 4552–4561, Jan. 2016.

[26] Y. Zheng, H. Duan, X. Yuan, and C. Wang, "Privacy-aware and efficient mobile crowdsensing with truth discovery," *IEEE Trans. Dependable Secure Comput.*, vol. 17, no. 1, pp. 121–133, Jan. 2017.

[27] C. Miao, L. Su, W. Jiang, Y. Li, and M. Tian, "A lightweight privacy-preserving truth discovery framework for mobile crowd sensing systems," in *Proc. IEEE INFOCOM Conf. Comput. Commun.*, May 2017, pp. 1–9.

[28] Y. Zheng, H. Duan, and C. Wang, "Learning the truth privately and confidently: Encrypted confidence-aware truth discovery in mobile crowdsensing," *IEEE Trans. Inf. Forensics Security*, vol. 13, no. 10, pp. 2475–2489, Oct. 2018.

[29] C. Zhang, L. Zhu, C. Xu, X. Liu, and K. Sharif, "Reliable and privacy-preserving truth discovery for mobile crowdsensing systems," *IEEE Trans. Dependable Secure Comput.*, early access, May 28, 2019, doi: 10.1109/TDSC.2019.2919517.

[30] T. Luo, J. Huang, S. S. Kanhere, J. Zhang, and S. K. Das, "Improving IoT data quality in mobile crowd sensing: A cross validation approach," *IEEE Internet Things J.*, vol. 6, no. 3, pp. 5651–5664, Jun. 2019.

[31] C. Prandi, S. Ferretti, S. Mirri, and P. Salomoni, "Trustworthiness in crowd- sensed and sourced georeferenced data," in *Proc. IEEE Int. Conf. Pervasive Comput. Commun. Workshops (PerCom Workshops)*, Mar. 2015, pp. 402–407.

[32] F. Shi, Z. Qin, D. Wu, and J. A. McCann, "Effective truth discovery and fair reward distribution for mobile crowdsensing," *Pervasive Mobile Comput.*, vol. 51, pp. 88–103, Dec. 2018.

[33] P. J. Rousseeuw, "Silhouettes: A graphical aid to the interpretation and validation of cluster analysis," *J. Comput. Appl. Math.*, vol. 20, pp. 53–65, Nov. 1987.

[34] T. S. Madhulatha, "Comparison between k-means and k-medoids clustering algorithms," in *Advances in Computing and Information Technology*. Berlin, Germany: Springer, 2011, pp. 472–481.

[35] D. M. Blei and P. I. Frazier, "Distance dependent chinese restaurant processes," *J. Mach. Learn. Res.*, vol. 12, pp. 2461–2488, Aug. 2011.

[36] E. E. Tsiropoulou, G. Mitsis, and S. Papavassiliou, "Interest-aware energy collection & resource management in machine to machine communications," *Ad Hoc Netw.*, vol. 68, pp. 48–57, Jan. 2018.

[37] V. Sels, J. Coelho, A. Manuel Dias, and M. Vanhoucke, "Hybrid tabu search and a truncated branch-and-bound for the unrelated parallel machine scheduling problem," *Comput. Oper. Res.*, vol. 53, pp. 107–117, Jan. 2015.

[38] D. Bruff, "The assignment problem and the hungarian method," *Notes Math*, vol. 20, nos. 29–47, p. 5, 2005.

[39] L. S. Shapley, "A value for *n*-person games," *Contrib. Theory Games*, vol. 2, no. 28, pp. 307–317, 1953.

[40] J. J. Levandoski, M. Sarwat, A. Eldawy, and M. F. Mokbel, "LARS: A location-aware recommender system," in *Proc. IEEE 28th Int. Conf. Data Eng. (ICDE)*, Apr. 2012, pp. 450–461.

[41] M. Sarwat, J. J. Levandoski, A. Eldawy, and M. F. Mokbel, "LARS*: An efficient and scalable location-aware recommender system," *IEEE Trans. Knowl. Data Eng.*, vol. 26, no. 6, pp. 1384–1399, Jun. 2014.

MENATALLA ABOUOUF (Member, IEEE) received the B.Sc. degree in electrical and electronic engineering and the M.Sc. degree in electrical and computer engineering from the Khalifa University of Science and Technology, Abu Dhabi, United Arab Emirates. She is currently working as a Research Engineer with the Center on Cyber-Physical Systems, Khalifa University of Science and Technology, Abu Dhabi. Her research interests include crowd sourcing, artificial intelligence, blockchain, and the IoT.

HADI OTROK (Senior Member, IEEE) received the Ph.D. degree in ECE from Concordia University. He holds an Associate Professor position with the Department of ECE, Khalifa University of Science and Technology, an Affiliate Associate Professor with the Concordia Institute for Information Systems Engineering, Concordia University, Montreal, Canada, and an Affiliate Associate Professor with the Electrical Department, École de Technologie Supérieure (ETS), Montreal. His research interests include the domain of computer and network security, web services, ad hoc networks, application of game theory, and cloud security. He has co-chaired several committees at various IEEE conferences. He is an Associate Editor of *Ad-Hoc Networks* (Elsevier), the IEEE Communications Letters, *Wireless Communications*, and *Mobile Computing* (Wiley).

SHAKTI SINGH (Member, IEEE) received the B.Sc., M.Sc., and Ph.D. degrees in electrical and computer engineering from Purdue University, West Lafayette, IN, USA. He is currently an Assistant Professor with the Electrical and Computer Engineering Department, Khalifa University of Science and Technology, Abu Dhabi, United Arab Emirates. His research interests include semiconductor devices and integrated circuits, sensors, sensing technologies, crowd sourcing, crowd sensing, and the development of IoT and wireless sensor networks.

RABEB MIZOUNI received the M.Sc. and Ph.D. degrees in electrical and computer engineering from Concordia University, Montreal, Canada, in 2002 and 2007, respectively. She is an Associate Professor in electrical and computer engineering with the Khalifa University of Science and Technology. She is currently interested in the deployment of context aware mobile applications, crowd sensing, software product line, and cloud computing.

ANIS OUALI received the B.Sc. degree in computer engineering from L'Ecole Nationale des Sciences de l'Informatique (ENSI), Tunisia, in 2000, the M.Sc. degree in computer science from Université du Québec à Montréal (UQAM), Canada, in 2004, and the Ph.D. degree from the Electrical and Computer Engineering Department, Concordia University, Montreal, Canada. He joined EBTIC in 2010 and is currently working in the network optimization team which focuses on solving network design related problem. His research interests include P2P networks for video streaming, distributed computing, and content adaptation.