Stable Worker–Task Assignment in Mobile Crowdsensing Applications

Fatih Yucel, Murat Yuksel, and Eyuphan Bulut

1 Introduction

Mobile crowdsensing (MCS) aims to complete spatio-temporal sensing tasks, which usually require massive expenses and execution times when performed individually, using the help from mobile participants (workers). This happens through recruitment of mobile users and leveraging the sensing capabilities (e.g., microphone, camera, and GPS) on their mobile devices. In an MCS system, there are mainly four entities; namely, the *platform*, *requesters*, *tasks*, and *workers*. Requesters define the tasks and post them to the platform with the requirements of their tasks, e.g., deadline, reward, and budget. Workers are the mobile users that register to the system with a set of their capabilities and limitations, e.g., a regional service area. The platform, knowing the tasks requested and the works eligible for each task in the system, makes the assignment of tasks to the requesters. This assignment can be made through a predefined logic with some goal, e.g., maximum tasks matched with minimum cost to requesters. Moreover, the platform, instead of performing the matching itself, can let the workers and requesters communicate and agree on an assignment in a distributed way. In this case, the platform acts as a mediator between the workers and task requesters.

One of the key and most studied problems in MCS systems is the assignment of sensing tasks to workers. The challenging part of this problem is there are many parameters that can be considered and objectives can vary depending on

F. Yucel · E. Bulut (✉)
Department of Computer Science, Virginia Commonwealth University, Richmond, VA, USA
e-mail: yucelf@vcu.edu; ebulut@vcu.edu

M. Yuksel
Department of Electrical and Computer Engineering, University of Central Florida, Orlando, FL, USA
e-mail: murat.yuksel@ucf.edu

J. Wu, E. Wang (eds.), *Mobile Crowdsourcing*, Wireless Networks,
https://doi.org/10.1007/978-3-031-32397-3_6

the system design. On the one side, there are task requesters who want their tasks to be completed in the best way (e.g., with the minimum cost to them and with the minimum delay), and, on the other side, there are workers who would like to make the best profit from the rewards they obtain once the tasks are completed and after their costs are taken out. There is also the platform that might be getting some registration cost from each user (i.e., task requester and worker) or some fee from each task completed and thus may aim to match as many tasks as possible to the eligible workers or maximize the total quality of service (QoS) received by the task requesters [1, 2]. In most of the existing studies in the literature, however, the objective during the task assignment process is defined in the favor of either one side (workers or task requesters) or for the system/platform itself. However, such assignments that do not take into account the individual needs and preferences of different entities may result in dissatisfied users and impair their future participation. This is because users in practice may not want to sacrifice their individual convenience for the sake of system utility or the other side's benefit.

In this chapter, we study the task assignment problem in MCS systems considering the preferences of entities involved in an MCS scenario. Preference-aware or stable matching has indeed been extensively studied in general bipartite matching problems especially in the economics literature [3]. However, these studies do not consider the features that are specific to MCS systems such as budget of task requesters, uncertainty in matching opportunities due to unknown worker trajectories, and time constraints of tasks. The stable matching problem for task assignment in MCS systems can indeed be defined in many different ways because of the varying settings of MCS scenarios, and, in each, the solution can be based on different approaches. Thus, we overview the different stable matching definitions studied recently in the MCS domain for the task assignment problem and provide a summary of proposed solutions. We also refer the readers to the actual studies for the details of the solutions. We hope this will highlight the spectrum of different stability definitions considered for MCS systems and summarize the differences.

The remainder of the chapter is organized as follows: in Sect. 2, we first start with a background on worker–task assignment problem and proposed solutions in the MCS literature, as well as with a background on stable matching theory and its applications in several domains. We then provide a motivation for using stable matching in MCS in Sect. 3. In Sect. 4, we provide with the classification of the MCS scenarios studied while considering preference awareness in the task assignments. We provide the blocking or unhappy pair definitions considered in different MCS settings and discuss how the stability is defined in each. We also summarize the algorithms proposed to find the stable matchings in such settings. Finally, we discuss the open problems that need to be studied in the MCS systems while considering the preference awareness and conclude the chapter in Sect. 5.

2 Background

In this section, we first provide an overview of worker–task assignment solutions in different MCS settings. Then, we look at the matching problems and solutions studied considering the user preferences in different domains. During these overviews as well as through the rest of this chapter, we base our discussion considering the three main categories of MCS scenarios illustrated in Fig. 1. In *participatory* sensing, workers can interrupt their daily schedule to carry out the assigned tasks (e.g., measuring air quality at a specific location) at the expense of additional cost, e.g., traveling distance. In *opportunistic* sensing, workers do not alter their schedules and perform the assigned tasks (e.g., traffic monitoring) only when they happen to be in the task regions, thus without an additional cost but with less likelihood of visiting task regions. Finally, in *hybrid* sensing, workers provide some flexibility through a set of alternative paths they can follow and let the platform decide which one to use to increase the utility from the matching.

2.1 *Worker–Task Assignment in Mobile Crowdsensing*

The overall performance of an MCS system and the satisfaction of its users are highly dependent on the efficiency of the assignments; thus, there have been many task assignment solutions proposed in different studies. These studies have considered various objectives in the task assignment process such as maximizing the number of completed tasks [5], minimizing the completion times of tasks [6], minimizing the incentives provided to the users [7], assuring the task or sensing quality [2] under some constraints on traveling distance [8], energy consumption [1], and expenses of task requesters [9]. Beyond these works, the issues of security [10], privacy [11], and truthfulness [12] have also been considered in the worker recruitment process.

Fig. 1 Three different MCS scenarios considered [4]. Solid line is the regular path that the user follows. Dashed blue lines are alternative similar paths to the user's regular path. Red line is the path the user is forced to follow to complete the tasks

In participatory MCS, since workers need to travel between the task regions to perform the assigned tasks, a key factor that shapes the task assignment process is the travel costs of the workers. In [13], the authors investigate the problem of minimizing the total travel costs of the workers while maximizing the number of completed tasks and keeping the rewards to be paid to the workers as low as possible. In [8], the authors study the task assignment problem in an online setting and aim to maximize the total task quality while ensuring that the travel costs of the workers do not exceed their individual travel budgets. In [14], the authors adopt a system model in which each worker has a maximum traveling distance that needs to be considered in the assignment process, and the objective is to maximize the profit of the platform. The authors propose a deep reinforcement learning-based scheme that significantly outperforms the other heuristic algorithms. In [15], the goal is also to minimize the travel distance of workers, and however, differently from the aforementioned studies, the authors consider the issue of user privacy and present a mechanism that finds the task assignments without exposing any private information about workers or task requesters. Lastly, in [16], the authors study the destination-aware task assignment problem in participatory crowdsourcing systems.

On the other hand, in opportunistic MCS, the main objectives are to maximize the coverage and to minimize the completion times of the tasks due to the uncontrolled mobility, i.e., a task can only be performed if its region resides on the trajectory of a worker. In [17], the authors study the maximum coverage task assignment problem in opportunistic MCS with worker trajectories that are known beforehand. It is assumed that each task needs to be performed at a certain point of interest and has a weight that indicates how important its completion is to the platform, which has a fixed budget and can hence recruit only so many workers. The objective of the platform is to select a set of workers within the budget constraints, which maximizes the weighted coverage over the set of tasks according to the given trajectories of workers. The authors develop a $(1 - 1/e)$-approximate algorithm with a time complexity of $O(n^5)$, where n is the number of workers in the system. [18] studies the same problem and proposes a greedy algorithm that, despite not having a theoretical guarantee, outperforms the algorithm proposed in [17] in terms of achieved coverage in certain settings and runs in $O(n^2)$ time.

Adding the delivery probability of the sensed data to the goals notably changes the problem being studied as shown in [19]. In this design, after carrying out a task, a worker should either deliver the sensed data to the server through one of the collection points (i.e., Wi-Fi APs) on his trajectory or transmit it to another user who will deliver it for him. Thus, here, not only does the platform need to estimate whether and when workers would visit task regions and collection points, but it is also crucial to obtain and utilize the encounter frequencies of workers to improve the delivery probability of the sensed data. The authors present different approximation algorithms for the systems with deterministic and uncertain worker trajectories and evaluate their performance on real datasets. The data delivery aspect of the problem in [19] has also been studied in [20] and [21]. They both utilize Nash Bargaining Theory to decide on whether or not selfish data collectors and mobile (relay) users who only take part in delivery of sensed data would like to cooperate with each other according to their utility in either scenario. However, in [21], the authors consider a more complete mobile social network model and present an enhanced data collection mechanism.

Another aspect to the MCS system design is the uncertainty of workers' trajectories. In [22], the problem of maximizing spatio-temporal coverage in vehicular MCS with uncertain but predictable vehicle (i.e., worker) trajectories is investigated. The authors first prove that the problem is NP-hard when there is a budget constraint and then propose a greedy approximation algorithm and a genetic algorithm. In [23], the authors also assume predictable worker trajectories. However, they focus on the task assignment problem in a mobile social network where task assignments and delivery of sensed data are realized in an online manner when task requesters and workers encounter with each other. They aim to minimize the task completion times and propose different approximation algorithms to optimize both worst-case and average-case performance. For predictions of worker trajectories, [22] uses spatio-temporal trajectory matrices, while [23] assumes that user inter-meeting times follow an exponential distribution, which is used widely in mobile social networks [24–26] literature.

Recently, there are studies [27, 28] that look at the task assignment problem in a hybrid system model to simultaneously leverage the advantages of participatory and opportunistic MCS. In [27], the authors propose a two-phased task allocation process, where opportunistic task assignment is followed by participatory task assignment. The objective behind this design is to maximize the number of tasks that are performed in an opportunistic manner, which is much less costly compared to participatory MCS, and then to ensure that the tasks that cannot be completed by opportunistic workers are assigned to workers that are willing to perform tasks in a participatory manner to alleviate the coverage problem in opportunistic MCS. On the other hand, in [28], the workers carry out the sensing tasks only in opportunistic mode, but they provide the matching platform with multiple paths that they would take if requested, instead of a single path as in classic opportunistic MCS. This enables the platform to find a matching with a high task coverage. However, none of the studies considers the stability in the assignment based on the preferences of the workers and task requesters.

2.2 Matching Under Preferences

Stable matching problem is introduced in the seminal paper of Gale and Shapley [29] and can simply be defined as the problem of finding a matching between two groups of objects such that no pair of objects favor each other over their partners in the matching. Gale and Shapley have also introduced what is called the deferred acceptance procedure that finds stable matchings in both one-to-one matching scenarios (e.g., stable marriages) and many-to-one matching scenarios with capacity constraints (e.g., stable college admissions) in $O(mn)$ time, where m and n are the size of the sets being matched. Since its introduction in [29], the concept of stability has been utilized in different problems including hospital resident assignment [30], resource allocation in device-to-device communications [31], SDN controller assignment in data center networks [32], supplier and demander matching in electric vehicle charging [33–35], and peer-to-peer energy sharing among mobile devices [36–38].

Some matching problems allow or require nodes in one or both sides to be matched with multiple nodes, i.e., many-to-one and many-to-many matching problems. A few studies investigate the issue of stability in such matching problems. For instance, [39] and [40] study the many-to-one stable matching of students–colleges and doctors–hospitals, respectively. In [39], all colleges define a utility and a wage value for students and aim to hire the best set of students (i.e., with the highest total utility) within their budget constraints. Each student also forms a preference list over colleges. The authors prove that there may not exist a stable matching in this setting and even checking the existence is NP-hard. However, they provide a polynomial time algorithm that finds pairwise stable matchings in the so-called typed weighted model where students are categorized into groups (e.g., Master's and PhD students) and colleges are restricted to define a set of possible wages for each group, i.e., they cannot define a particular wage for each student. [40] studies the same problem and proposes two different fully polynomial time approximation algorithms with some performance guarantee in terms of coalitional stability for general and proportional (i.e., the wage of doctors are proportional to their utility for hospitals) settings. However, the study does not provide an experimental analysis of the algorithms or discuss their actual/expected performance in these settings. Moreover, the proposed solutions can only be applied to a limited set of scenarios.

There are some studies that look at the stable matching problem in settings with incomplete information on user preferences or dynamic user arrivals/departures. [41] and [42] both study the dynamic stable taxi dispatching problem considering passenger and taxi preferences. However, the objective adopted in [42] is to find locally optimal stable assignments for a given time-point, whereas that in [41] is to minimize the number of unhappy taxi–passenger pairs globally. The authors in [43] investigate the stable matching problem in the presence of uncertainty in user preferences. [44] looks at the problem of minimizing the number of partner changes that need to be made in a stable matching to maintain stability when preference profiles of some users change in time. Lastly, [45] studies an interesting combination of famous stable marriage and secretary (hiring) problems.

The concept of stability is studied in multi-dimensional matching problems as well. In [46], the authors introduce the three-dimensional stable matching problem. In this problem, there are three sets of different types, each individual from a set has a preference list over all pairs from the other two sets, and the goal is to form stable/satisfactory families of three, where each individual in a family is a member of a different set. Wu [47] investigates a different version of this problem, where each individual has a one-dimensional preference list over the individuals from the other two sets instead of over all pairs of individuals as in [46]. In [48], the authors extend the stable roommates problem [49] to a three-dimensional setting, where a set of individuals are assigned into groups of three instead of two based on their preferences. Lastly, in [50], the authors study the problem of matching data sources, servers, and users in a stable manner in video streaming services under restricted preference settings.

In a typical MCS system, the objectives of workers and task requesters can be defined as to maximize their profits and to have their tasks completed with the highest quality possible, respectively. Thus, they are likely to have preferences over possible assignments they can get, and the task assignment in MCS can be consequently characterized as a matching problem under preferences. Apart from the studies that we will present in the next section, there are only a few studies that consider user preferences in mobile crowdsensing (or in mobile crowdsourcing). In [51], the authors study the budget-constrained many-to-many stable task assignment problem, which they prove to be NP-hard, and propose efficient approximation algorithms. Similarly, in [52], stability in many-to-many assignments has also been studied considering a competition congestion metric. In [53], the authors study the same problem, but in a system model with capacity constraints. On the other hand, [54] considers a many-to-one matching setting and introduce additional constraints (e.g., minimum task quality requirements) that are taken into account in the matching process, along with user preferences. Lastly, in [55], the authors consider a budget-constrained MCS system where the quality of a worker is identical for all tasks and present an exponential-time algorithm to find weakly stable many-to-one matchings. Note that there are also studies (e.g., [56]) that use auctions for fairness in crowdsensing systems, but these are out of the scope of this work.

3 Why Should We Care About Stability in MCS?

The answer of this question is rather obvious in MCS campaigns with no central authority, where the task assignments are made in a distributed manner or by a piece of software that runs on the cloud and is managed jointly by the users. This is because aside from malicious intent, there is no reason for the users of such systems to adopt a task assignment mechanism that would favor certain individuals, and thus the long-term functioning of such systems can only be made possible by considering user preferences in a fair way and producing stable assignments where no user has an incentive to deviate from their assigned partners.

However, in MCS campaigns with a central service provider (SP) which aims to maximize its profits, the question set forth above is more of a business question than an engineering question. The main motivation for an SP to consider the user preferences and provide a stable worker–task assignment would be to ensure the continuous participation of the existing users and to promote their willingness to perform the assigned tasks. However, this may not align perfectly with the SP's goal of maximizing its own utility. So, here the SP faces a critical business decision: it should either choose to maximize its own utility by disregarding user preferences, but potentially suffer from the consequences of doing so (e.g., unhappy users abandoning the platform), or should prioritize user preferences to keep the users actively participating in the MCS campaign at the cost of its own utility.

In order to demonstrate this trade-off between the utility of SP and user happiness, we have performed a series of experiments and shown that [57] a task assignment solely maximizing the utility of the SP without considering user preferences may make the majority of users unhappy with their assignments. In such cases, if the users that get such dissatisfying assignments do not obey the task assignment results and not perform the task, as they can be selfish and can consider their own benefit, the SP will face a significant utility loss. Furthermore, if this dissatisfaction causes even a certain part of the users to abandon the system in each assignment cycle, over time, the numbers of workers and task requesters participating in the campaign will decrease and this will result in an exponential utility loss for the SP. Even though these results show the value of considering user preferences, they do not necessarily indicate that the SP should always only care about user preferences and ignore its own utility. In fact, in some scenarios, the SP may benefit from producing task assignments [57], which maximize its own utility while keeping the conflicts with user preferences as minimal as possible (i.e., the system utility and user preferences as primary and secondary objectives, respectively).

4 Stable Task Assignments in Different MCS Applications

In this section, we present the stability definitions considered in three different MCS scenarios (i.e., participatory, opportunistic, and hybrid). Throughout the section, we also use the terms *uniform* and *proportional* MCS systems, where the former refers to the MCS scenarios when the QoS provided by each worker is the same for all tasks, and the latter refers to the MCS scenarios where the rewards that are offered to the workers are proportional to the QoS they provide. Note that these are exclusive to the three aforementioned categories (i.e., participatory, opportunistic, and hybrid).

Throughout the chapter, we assume a system model with a set of workers $\mathcal{W} = \{w_1, w_2, \ldots, w_n\}$ and a set of sensing tasks $\mathcal{T} = \{t_1, t_2, \ldots, t_m\}$. We also define $c_t(w)$ as the cost of performing task t for worker w and use $r_t(w)$ for the reward that worker w is offered to carry out task t.

4.1 Participatory MCS

An MCS system is called participatory if the available workers considered in the task assignment process are actively waiting for new tasks to be assigned and are willing to go to the locations of the assigned tasks immediately or whenever requested by the task requesters. In other words, the task assignment mechanisms in these systems can be assumed to have the ability to control the mobility of workers, generally within some constraints specified by workers. This, of course, creates additional cost to the task requesters, as workers need to travel to task locations and ask for more rewards to cover this cost.

In MCS systems where each task can recruit multiple workers within their budget constraints, the stability can be defined in two different ways: pairwise and coalitional. Due to the many-to-one nature of task assignments and budget constraints, the conditions of both pairwise and coalitional stability differ from the classic stability conditions specified in [29], and thus existing stable matching solutions cannot be used to find pairwise or coalitional stable matchings in such systems. Moreover, depending on the relation (i.e., proportional or not) between the QoS provided by workers and the reward they gain, the hardness of the problem and the corresponding solution approach completely change.

Since a rational worker will aim to maximize their profit and will not accept to perform the tasks that cost higher than the corresponding rewards that will be paid, we can define the preference list of worker w as

$$P_w = t_{i_1}, t_{i_2}, \ldots, t_{i_k}, \tag{1}$$

where $P_w \subseteq \mathcal{T}, \forall t \in P_w, r_t(w) > c_t(w)$, and $\forall t' = t_{i_j}, t'' = t_{i_{j+1}}, r_{t'}(w) - c_{t'}(w) > r_{t''}(w) - c_{t''}(w)$. We denote the jth task (t_{i_j}) in P_w by $P_w(j)$ and utilize $t' \succ_w t''$ notation to express that t' precedes t'' in P_w.

On the other hand, a rational task requester will try to maximize the total quality of service (QoS) that can be obtained from the recruited workers considering his/her budget constraint. Let $q_t(w)$ denote the QoS that worker w can provide for task t and b_t denote the budget of task t. Then, we can define the preference list of task t as

$$P_t = S_1, S_2, \ldots, S_k, \tag{2}$$

where $\forall S \in P_t, S \subseteq \mathcal{W}$ and $\sum_{w \in S} r_t(w) \leq b_t$, and $\forall S_i, S_{i+1} \in P_t, \sum_{w \in S_i} q_t(w) \geq \sum_{w \in S_{i+1}} q_t(w)$.

Here, given a matching $\mathcal{M}$ and $w \in \mathcal{W}, t \in \mathcal{T}$, the partner[1] of worker w is denoted by $\mathcal{M}(w)$ and the partner set of task t is denoted by $\mathcal{M}(t)$. If $\mathcal{M}(u) = \emptyset$ for user $u \in \mathcal{W} \cup \mathcal{T}$, it means user u is unmatched in $\mathcal{M}$. Note that the last set in the preference list of each task t is $\emptyset$, so we have $S \succ_t \emptyset, \forall S \in (P_t \setminus \emptyset)$. Also, even though the preference lists of workers do not include $\emptyset$, since we assume that the workers in our system are rational, we have $t \succ_w \emptyset$, for all $w \in \mathcal{W}$ and $t \in P_w$. We denote the remaining budget of task t in $\mathcal{M}$ by $b_t^{\mathcal{M}} = b_t - r_t(\mathcal{M}(t))$.

[1] The partner of a worker refers to the task that the worker is assigned to perform, while the partner set of a task refers to the set of all workers assigned with the task.

Table 1 Mobile crowdsensing scenarios (i.e., uniform (U) and proportional (P)) and corresponding applicable algorithms ($*$ indicates the algorithm is applicable but has a very poor performance since it is not specifically designed for that scenario)

MCS Type	UTA [58]	PSTA [58]	Heuristic [58]	SJA [55]	ϕ-STA [59]	θ-STA [59]
P. & U.	✓	✓	✓	✓	✓	*
P. & N.U.		✓	✓		✓	*
N.P. & U.	✓	✓	✓			✓
N.P. & N.U.		✓	✓			✓

Definition 1 (Unhappy Pair) Given a matching $\mathcal{M}$, a worker w, and a task t form an unhappy (blocking) pair $\langle w, t\rangle$ if $t \succ_w \mathcal{M}(w)$, and there is a subset $S \subseteq \mathcal{M}(t)$ such that $\{w\} \succ_t S$ and $r_t(w) \leq b_t^{\mathcal{M}} + r_t(S)$.

Then, a matching $\mathcal{M}$ is said to be **pairwise stable** if it does not admit any unhappy pairs.

Definition 2 (Unhappy Coalition) Given a matching $\mathcal{M}$, a subset of workers $S \subseteq \mathcal{W}$ and a task t form an unhappy (blocking) coalition $\langle S, t\rangle$ if $\forall w \in S$, $t \succ_w \mathcal{M}(w)$ and there is a subset $S' \subseteq \mathcal{M}(t)$ such that $S \succ_t S'$ and $r_t(S) \leq b_t^{\mathcal{M}} + r_t(S')$.

Similarly, a matching $\mathcal{M}$ is said to be **coalitionally stable** if it does not admit any unhappy coalition.

Following these different unhappy pair and stability definitions and using a classification of MCS systems based on the variability in the QoS provided by the workers for different tasks (uniform/non-uniform) and the relationship between the QoS provided by the workers and the rewards they are offered (proportional/non-proportional), three different stable task assignment algorithms, namely UTA, PSTA, and Heuristic, have been provided for different MCS classes and scenarios in [58]. These algorithms are summarized in Table 1. In [58], we prove that UTA and PSTA algorithms always produce pairwise stable task assignments in uniform and proportional MCS systems, respectively. With simulation results, we also show that our algorithms significantly outperform the state-of-the-art stable task assignment algorithms in most scenarios. Specifically, PSTA and Heuristic algorithms usually achieve the highest outward and overall user happiness, respectively.

In participatory MCS systems where the number of workers is scarce, it is also possible that some workers are assigned to multiple tasks to complete all the tasks, while still assigning each task to a single worker. While this provides workers with an opportunity to earn more rewards, the consideration of preferences and stability can be different. Note that in such systems, the task assignments can be performed either instantly or in a predetermined way. In the former, workers are assigned one task at a time, and they are assigned a new one only after completing their currently assigned task. However, this creates an uncertainty for the worker and also the assignments made become not optimal. On the other hand, if the task assignments for all workers could be planned in a foreseeable future (e.g., the next hour or day), such issues can be avoided. This is studied in [60] and a task assignment algorithm that not only considers the scheduling of tasks for workers but also respects user preferences so that no user will have a desire to deviate from their assignment (i.e., stable) is proposed.

4.2 Opportunistic MCS

An MCS system is called opportunistic if the mobility of workers is not controllable and the tasks can be completed by workers only when they happen to visit the task locations in their regular mobility patterns. Thus, the task assignments should be made considering the likelihood that the workers will visit the task regions within an acceptable time frame. As workers are not directed to a certain location, there is also no travel cost associated with task assignments, and however, it may take longer for workers to visit task regions in an opportunistic manner (compared to participatory sensing).

A particularly important objective in the opportunistic MCS systems is to maximize the sensing coverage over a set of points of interest (POIs), which has recently been studied in [9, 18, 61, 62]. However, these studies either do not consider budget constraints of task requesters or assume that there is only a single task requester (i.e., a single budget constraint) in the system. This may not be a practical assumption as there can be multiple task requesters with a unique set of goals and an individual budget constraint. Moreover, some task requesters may prefer to allocate a separate budget for different sets of POIs. Thus, in such systems, stable assignment that considers each requester's preferences would benefit all. When the utility functions are additive (i.e., the total utility of a set of workers for a task is equal to the sum of their individual utilities). However, when the utility functions are additive (...), the stability can be handled easily. On the other hand, the coverage of workers over a set of POIs is usually non-additive because of the commonly covered POIs by different workers. In order to handle such scenarios, in [63], we propose the following definition for unhappy coalitions.

Definition 3 (Unhappy Coalition with Non-additive Worker Utilities) Given a matching $\mathcal{M}$, a task t and a subset S of workers form an unhappy coalition (denoted by $\langle S, t \rangle$) if the following conditions hold for a subset S' of the workers assigned to task t in $\mathcal{M}$:

- Task t would be better off with S than with S', i.e.,

$$U_t(S \cup (\mathcal{M}(t) \setminus S')) > U_t(\mathcal{M}(t)), \tag{3}$$

  where $U_t(S)$ defines the non-additive utility of set S of workers [63],
- Task t can replace S' with S without violating her budget constraint, i.e.,

$$r_t(S) - r_t(S') \leq b_t^{\mathcal{M}}, \tag{4}$$

where $b_t^{\mathcal{M}}$ is the remaining budget of task t in $\mathcal{M}$ (i.e., $b_t^{\mathcal{M}} = b_t - \sum_{w \in \mathcal{M}(t)} r_t(w)$),

- Every worker w in S prefers task t to task t' to whom he is currently assigned in $\mathcal{M}$, i.e.,

$$\forall w \in S,\ g_t(w) > g_{t'}(w), \tag{5}$$

where $g_t(w) = r_t(w) - c_t(w)$ is the net profit of performing task t for worker w and $g_{t'}(w) = 0$ if worker w is currently unmatched (i.e., $\mathcal{M}(w) = t' = \emptyset$).

Given this definition, a matching $\mathcal{M}$ is considered **coalitionally stable** if it does not contain any unhappy coalitions. However, we show that in some MCS instances it may not be possible to find a stable matching under this definition. Thus, we propose α-stability if the matching obtained achieves not more than α dissatisfaction for each worker. This α-stability is studied in different scenarios (e.g., proportional rewards) and corresponding algorithms that guarantee certain α values are provided. These algorithms are adapted from the well-known online budgeted maximum coverage (OBMC) problem [64].

In an opportunistic MCS setting, it is also possible that worker trajectories can be uncertain and hence not known in advance. Thus, existing solutions fail to produce an effective task assignment. Moreover, the uncertainty in worker trajectories requires a different stability definition. In [65], we study this problem with the following unhappy pair definition.

Definition 4 (Decision–time Unhappy Pair) A worker–task pair (w_i, t_j) is said to be a decision–time unhappy pair if the following conditions hold for any time-step s in $[t_j.b, t_j.d]$ (i.e., beginning and deadline of task j):

- Worker w_i has a positive remaining capacity.
- Task t_j is unassigned.
- Worker w_i is in region $t_j.r$.
- Either (i) SP matches worker w_i to task t_j, but at least one of them would be better off otherwise, i.e.,

$$\mathbf{W}_i(s) > \mathbf{W}'_{i,j}(s) \ \text{ or } \ \mathbf{T}_j(s) > \mathbf{T}'_{j,i}(s) \tag{6}$$

- Or (ii) SP does not match worker w_i to task t_j, but they both would be better off otherwise, i.e.,

$$\mathbf{W}'_{i,j}(s) > \mathbf{W}_i(s) \ \text{ and } \ \mathbf{T}'_{j,i}(s) > \mathbf{T}_j(s). \tag{7}$$

Here, $\mathbf{W}_i(s)$ and $\mathbf{W}'_{i,j}(s)$ refer to the expected total reward worker w_i would get in time frame $[s, T]$ if he was not assigned to task t_j at time-step s, and otherwise, respectively. Similarly, $\mathbf{T}_j(s)$ and $\mathbf{T}'_{j,i}(s)$ refer to the expected sensing quality to be received by task t_j if it is not assigned to worker w_i at time-step s and otherwise, respectively.

Thus, a matching $\mathcal{M}$ is called an **online stable matching** if it does not admit any decision–time unhappy pairs. While it is straightforward to see that the optimal matching strategy toward such stable matching would match a worker–task pair if (7) holds, the challenge comes from the computation of the values of $\mathbf{W}_i(s)$, $\mathbf{W}'_{i,j}(s)$, and $\mathbf{T}_j(s)$ because $\mathcal{A}_s$ (which is defined as the set of all possible worker visit scenarios for the time frame $[s, T]$ given the visit probabilities of the workers for all task regions [65]) grows exponentially with the number of users and the length of the assignment period (T). In [65], we compute these values efficiently without actually forming the set $\mathcal{A}_s$ and develop an efficient solution that always finds online stable matchings.

4.3 Hybrid MCS

Besides a purely participatory or a purely opportunistic MCS system, it is also possible to have a hybrid MCS system to take advantage of both systems while avoiding the issues in each. The key issue in the participatory mode is that the paths assigned to workers are likely to disturb their daily schedules and introduce significant additional travel costs, whereas the opportunistic mode mainly suffers from the issue of poor coverage, as a task cannot be carried out if its region will not be visited in time by any worker in the system during their self-defined trips.

A hybrid (or semi-opportunistic) sensing mode can address these issues and finds a middle ground between the participatory and opportunistic modes [28]. In the hybrid mode, the workers provide the matching platform with alternative paths (e.g., dashed lines in Fig. 1) they would be willing to take within their comfort zones in addition to the path they would normally take. This yields a wider range of task assignment options for both workers and tasks and hence not only improves the task coverage but also expands the set of tasks that workers can carry out, allowing them to increase their profits by performing more tasks. However, existing studies [27, 28] do not consider the stability in the assignment based on the preferences of the workers and task requesters; thus, the resulting assignment may impair their long-term participation in the MCS campaign.

The three-dimensional version of stable matching was indeed introduced by Knuth [66] by considering three sets of agents (e.g., woman, man, dogs) and their preferences on the others. Moreover, several variants that consider cyclic preference relations [67] as well as one-dimensional preference lists over all individuals from the other two sets [47] have also been studied. Such three-dimensional stable matching has also been considered in several other application domains such as server-data source–user matching [50] in video streaming services.

In some recent studies, three-dimensional stability is also considered in spatial-crowdsourcing context. However, these studies have a limited understanding of user preferences and stability. For example, [68] only considers the preferences of users on the potential places (that the tasks will be completed) based on their proximity, while workers and task requesters do not have preferences over each other. On the other hand, there are also studies [69, 70] that consider trichromatic matching (i.e., matching of three items such as tasks, workers, and workplaces/PoIs) with some stability definitions. However, these studies mainly focus on task scheduling within a deadline without considering the matching stability based on user preferences and aim to maximize the number of matched items.

In this section, we present a totally different scenario that is studied in [4] where only the nodes in two (i.e., workers, tasks) of the three sets have preferences over each other depending on the features of the nodes in the third set (i.e., acceptable paths of workers).

Each worker w_i provides the service provider *(SP)* with a set of paths $\mathcal{P}_i = \{p_{i,1}, p_{i,2}, \ldots, p_{i,a_i}\}$ that he finds acceptable from his current location to his destination. In each assignment period, it is the responsibility of SP to find a satisfactory assignment between workers and tasks by matching workers to one of their acceptable paths and assigning a subset of tasks on their selected paths. Each path $p_{i,j}$ has a capacity $c_{i,j}$ associated with it, which indicates the maximum number of tasks that worker w_i is willing to perform if he is assigned to path $p_{i,j}$. Given these constraints, then we define the following to base our stable solution on.

Definition 5 (Unhappy Triad) Given a matching $\mathcal{M}$, worker w_i, path $p_{i,j}$, and a set S of tasks form an unhappy triad denoted by $\langle w_i, p_{i,j}, S\rangle$ if

- S is an acceptable assignment for w_i, i.e.,

$$1 \le |S| \le c_{i,j},\ S \subseteq L_i^w,\ \text{and } S \subseteq \mathcal{T}_{i,j}. \tag{8}$$

- w_i is an acceptable assignment for each $t_k \in S$, i.e.,

$$w_i \in L_k^t \text{ and } t_k \in \mathcal{T}_{i,j}. \tag{9}$$

- Each task $t_k \in S$ either prefers worker w_i to their current assignment w_h in $\mathcal{M}$, i.e.,

$$q_{i,k} > q_{h,k} \textit{ where } q_{h,k} = 0 \text{ if } w_h = \emptyset, \tag{10}$$

  or is already assigned to worker w_i, i.e., $\mathcal{M}_u(t_k) = w_i$.
- Worker w_i prefers the task set S to his current assignment in $\mathcal{M}$, i.e.,

$$\sum_{t_h \in S} r_{h,i} > \sum_{t_k \in \mathcal{M}_u(w_i)} r_{k,i}. \tag{11}$$

Thus, given an unhappy triad $\langle w_i, p_{i,j}, S\rangle$, we see from the first two conditions that it is possible to assign the tasks in the set S to worker w_i through path $p_{i,j}$ without violating any feasibility constraints and see from the last two conditions that this would make at least one task in S and worker w_i strictly better off without making any task in S worse off.

A matching is said to be **3D-stable** if it does not contain any unhappy triads. In [4], we provide two different algorithms for different MCS systems. In the first algorithm, we provide a solution that always finds the stable matching in uniform MCS systems. In the second algorithm, we consider the general MCS instances where stable matchings may not exist and propose an approximation algorithm that finds near-optimal matchings in terms of stability.

5 Conclusion and Open Problems

In this chapter, we studied the worker–task assignment problem in MCS systems while considering their preferences or aiming to obtain a stable matching solution. Due to the various kinds of MCS scenarios possible and many parameters (e.g., budget of task requesters, capacity of workers) that can be considered in their design, obtaining a stable matching is very challenging, and existing solutions cannot be applied directly. Thus, we provided an overview of the recent studies that considered stability in their design while assigning the tasks in the system to the eligible workers. Considering the three main categories of MCS systems, namely participatory, opportunistic, and hybrid, we have provided the core blocking or unhappy pair definitions considered to define the stability in each scenario and discussed in what conditions the proposed algorithms can find exact stable or approximate stable solutions.

Besides the studies considered in this chapter, as the stability has only been studied in a few recent studies in MCS literature, there are many potential interesting problems that stability can be studied during the task assignment process. For example, in MCS systems that are defined within a mobile social network (MSN) using the local communication technologies (e.g., device to device (D2D), Bluetooth, and Wi-Fi) between the users, the decision of online task assignments while considering the stability of the decisions made is a challenging and not studied problem. Note that in such MCS systems, not only the task assignments happen in a distributed fashion and when the workers and task requesters meet each other, but also the delivery of tasks completed happens opportunistically and when the same parties meet each other again. Thus, two problems should be considered together. While the short-distance communication helps reduce the overhead on cellular networks and allows for local user recruitment and sensed data collection even if the cellular network coverage is poor [19, 71], the uncertainty increases in the system, making the stability management much harder (which can get more challenging with correlation among the mobility patterns of users [72]). Having budget constraints of task requesters and capacity constraints of workers make the problem even further challenging. Some recent studies have looked at the task assignment and worker recruitment problem [73–76]) in such MSN-based opportunistic crowdsensing scenarios. While these studies leverage the opportunistic encounters of nodes for task assignment and communication between nodes, they do not consider stability in the assignment. Thus, this problem is still an open problem.

Another key aspect that has been overlooked so far is the benefit of cooperation between workers. In MCS systems with non-trivial tasks, it may be the case that two workers who cannot carry out a certain task individually can do so if they are both assigned to the task and work in a cooperative manner. Therefore, their total utility for the task would be larger than the sum of their individual utilities. Additional costs, however, may need to be incurred to make them work cooperatively, which need to be considered in the assignment process, along with the potential benefits to be reaped. This is similar to the assignment problem with non-additive utility functions studied in [63], but a major difference is that the total (coverage-based) utility of two (or more) workers for a task in the model considered there cannot be larger than the sum of their individual utilities.

In this chapter, we assumed a system model with rational and reliable participants. However, there may be, for example, workers who are trying to spread misinformation by submitting fabricated data. When the possibility of having such malicious users are taken into consideration, user preferences may become uncertain. We have also assumed that the sets of workers and tasks were known to the matching platform before the sensing campaign actually starts. Yet for many real-world applications, a more realistic model would allow users to join and leave the system and allow task requesters to publish new tasks and withdraw some of their existing tasks in real time during the campaign. Lastly, it is possible to improve the long-term efficiency of the proposed algorithms by forming the task assignments for an assignment period by modifying the assignments in the previous task assignment period(s) instead of creating a new task assignment from scratch in each assignment period. This has the potential to largely reduce the total running time of the proposed algorithms, especially in MCS applications, where user preferences do not change significantly between consecutive assignment periods. Thus, this is another interesting problem that can be explored with the stability in mind.

References

1. J. Wang, J. Tang, G. Xue, D. Yang, Towards energy-efficient task scheduling on smartphones in mobile crowd sensing systems. Comput. Netw. **115**, 100–109 (2017)
2. T. Hu, M. Xiao, C. Hu, G. Gao, B. Wang, A QoS-sensitive task assignment algorithm for mobile crowdsensing. Pervasive Mob. Comput. **41**, 333–342 (2017)
3. L. Tesfatsion, in *Gale-Shapley Matching in an Evolutionary Trade Network Game* (Iowa State University, Ames, 1998)
4. F. Yucel, E. Bulut, Three-dimensional stable task assignment in semi-opportunistic mobile crowdsensing, in *IEEE 23rd International Symposium on a World of Wireless, Mobile and Multimedia Networks (WoWMoM)* (2022)
5. L. Kazemi, C. Shahabi, Geocrowd: enabling query answering with spatial crowdsourcing, in *Proceedings of the 20th International Conference on Advances in Geographic Information Systems* (2012), pp. 189–198

6. M. Abououf, R. Mizouni, S. Singh, H. Otrok, A. Ouali, Multi-worker multi-task selection framework in mobile crowd sourcing. J. Netw. Comput. Appl. **130**, 52–62 (2019)
7. T. Liu, Y. Zhu, L. Huang, TGBA: a two-phase group buying based auction mechanism for recruiting workers in mobile crowd sensing. Comput. Netw. **149**, 56–75 (2019)
8. W. Gong, B. Zhang, C. Li, Location-based online task assignment and path planning for mobile crowdsensing. IEEE Trans. Veh. Technol. **68**, 1772–1783 (2018)
9. Z. Zheng, F.Wu, X. Gao, H. Zhu, S. Tang, G. Chen, A budget feasible incentive mechanism for weighted coverage maximization in mobile crowdsensing. IEEE Trans. Mob. Comput. **16**, 2392–2407 (2016)
10. M. Xiao, J. Wu, S. Zhang, J. Yu, Secret-sharing-based secure user recruitment protocol for mobile crowdsensing, in *IEEE Conference on Computer Communications (INFOCOM)* (2017), pp. 1–9
11. L. Wang, D. Yang, X. Han, T. Wang, D. Zhang, X. Ma, Location privacy-preserving task allocation for mobile crowdsensing with differential geo-obfuscation, in *Proceedings of the 26th International Conference on World Wide Web* (2017), pp. 627–636
12. J. Li, Z. Cai, J. Wang, M. Han, Y. Li, Truthful incentive mechanisms for geographical position conflicting mobile crowdsensing systems. IEEE Trans. Comput. Soc. Syst. **5**, 324–334 (2018)
13. Y. Liu, B. Guo, Y. Wang, W. Wu, Z. Yu, D. Zhang, TaskMe: multi-task allocation in mobile crowd sensing, in *Proceedings of the 2016 ACM International Joint Conference on Pervasive and Ubiquitous Computing* (2016), pp. 403–414
14. X. Tao, W. Song, Task allocation for mobile crowdsensing with deep reinforcement learning, in *2020 IEEE Wireless Communications and Networking Conference (WCNC)* (2020), pp. 1–7
15. B. Zhao, S. Tang, X. Liu, X. Zhang, W. Chen, iTAM: bilateral privacy-preserving task assignment for mobile crowdsensing. IEEE Trans. Mob. Comput. **20** , 3351–3366 (2020)
16. Y. Zhao, K. Zheng, Y. Li, H. Su, J. Liu, X. Zhou, Destination-aware task assignment in spatial crowdsourcing: a worker decomposition approach. IEEE Trans. Knowl. Data Eng. **32**, 2336–2350 (2019)
17. M. Zhang, P. Yang, C. Tian, S. Tang, X. Gao, B. Wang, F. Xiao, Quality-aware sensing coverage in budget-constrained mobile crowdsensing networks. IEEE Trans. Veh. Technol. **65**, 7698–7707 (2015)
18. J. Chen, J. Yang, Maximizing coverage quality with budget constrained in mobile crowdsensing network for environmental monitoring applications. Sensors **19**, 2399 (2019a)
19. M. Karaliopoulos, O. Telelis, I. Koutsopoulos, User recruitment for mobile crowdsensing over opportunistic networks, in *2015 IEEE Conference on Computer Communications (INFOCOM)* (2015), pp. 2254–2262
20. Y. Zhan, Y. Xia, Y. Liu, F. Li, Y. Wang, Incentive-aware time-sensitive data collection in mobile opportunistic crowdsensing. IEEE Trans. Veh. Technol. **66**, 7849–7861 (2017)
21. Y. Zhan, Y. Xia, J. Zhang, Y. Wang, Incentive mechanism design in mobile opportunistic data collection with time sensitivity. IEEE Internet Things J. **5**, 246–256 (2017)
22. Z. He, J. Cao, X. Liu, High quality participant recruitment in vehicle-based crowdsourcing using predictable mobility, in *IEEE Conference on Computer Communications (INFOCOM)* (2015), pp. 2542–2550
23. M. Xiao, J. Wu, L. Huang, R. Cheng, Y. Wang, Online task assignment for crowdsensing in predictable mobile social networks. IEEE Trans. Mob. Comput. **16**, 2306–2320 (2016)
24. J. Wu, M. Xiao, L. Huang, Homing spread: community home-based multi-copy routing in mobile social networks, in *IEEE Conference on Computer Communications (INFOCOM)* (2013), pp. 2319–2327
25. A. Dhungana, E. Bulut, Energy sharing based content delivery in mobile social networks, in *20th IEEE International Symposium on "A World of Wireless, Mobile and Multimedia Networks"(WoWMoM)* (2019), pp. 1–9
26. Y. Li, D. Jin, P. Hui, S. Chen, Contact-aware data replication in roadside unit aided vehicular delay tolerant networks. IEEE Trans. Mob. Comput. **15**, 306–321 (2015)
27. J. Wang, F. Wang, Y. Wang, L. Wang, Z. Qiu, D. Zhang, B. Guo, Q. Lv, HyTasker: hybrid task allocation in mobile crowd sensing. IEEE Trans. Mob. Comput. **19**, 598–611 (2019)

28. W. Gong, B. Zhang, C. Li, Z. Yao, Task allocation in semi-opportunistic mobile crowdsensing: paradigm and algorithms. Mob. Netw. Appl. **25**, 772–782 (2020)
29. D. Gale, L.S. Shapley, College admissions and the stability of marriage. Am. Math. Mon. **69**(1), 9–15 (1962)
30. National Resident Matching Program (2021). https://www.nrmp.org/matching-algorithm/
31. Z. Zhou, C. Gao, C. Xu, T. Chen, D. Zhang, S. Mumtaz, Energy-efficient stable matching for resource allocation in energy harvesting-based device-to-device communications. IEEE Access. **5**, 15184–15196 (2017)
32. T. Wang, F. Liu, J. Guo, H. Xu, Dynamic SDN controller assignment in data center networks: stable matching with transfers, in *The 35th Annual IEEE International Conference on Computer Communications (INFOCOM)* (2016), pp. 1–9
33. R. Zhang, X. Cheng, L. Yang, Flexible energy management protocol for cooperative EV-to-EV charging. IEEE Trans. Intell. Transp. Syst. **20**, 172–184 (2018)
34. F. Yucel, K. Akkaya, E. Bulut, Efficient and privacy preserving supplier matching for electric vehicle charging. Ad Hoc Netw. **90**, 101730 (2019)
35. E. Bulut, M. Kisacikoglu, K. Akkaya, Spatio-temporal non-intrusive direct V2V charge sharing coordination. IEEE Trans. Veh. Technol. **68**, 9385–9398 (2019)
36. E. Bulut, B. Szymanski, Mobile energy sharing through power buddies, in *IEEE Wireless Communications and Networking Conference (WCNC), San Francisco, CA, USA, March 19–22, 2017* (2017). pp. 1–6
37. A. Dhungana, T. Arodz, E. Bulut, Exploiting peer-to-peer wireless energy sharing for mobile charging relief. Ad Hoc Netw. **91**, 01882 (2019)
38. E. Bulut, S. Hernandez, A. Dhungana, B. Szymanski, Is crowdcharging possible?, in*IEEE 27th International conference on Computer Communication and Networks, ICCCN 2018, Hangzhou, China, July 30 - August 2, 2018* (2018), pp. 1–9
39. A. Ismaili, N. Hamada, Y. Zhang, T. Suzuki, M. Yokoo, Weighted matching markets with budget constraints. J. Artif. Intell. Res. **65**, 393–421 (2019)
40. Y. Kawase, A. Iwasaki, Approximately stable matchings with budget constraints, in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 32 (2018)
41. B. Zhao, P. Xu, Y. Shi, Y. Tong, Z. Zhou, Y. Zeng, Preference-aware task assignment in on-demand taxi dispatching: an online stable matching approach, in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 33 (2019), pp. 2245–2252
42. H. Zheng, J. Wu, Online to offline business: urban taxi dispatching with passenger-driver matching stability, in *2017 IEEE 37th International Conference on Distributed Computing Systems (ICDCS)* (2017), pp. 816–825
43. H. Aziz, P. Biró, S. Gaspers, R. Haan, N. Mattei, B. Rastegari, Stable matching with uncertain linear preferences. Algorithmica. **82**, 1410–1433 (2020)
44. R. Bredereck, J. Chen, D. Knop, J. Luo, R. Niedermeier, Adapting stable matchings to evolving preferences, in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 34 (2020), 1830–1837
45. K. Eriksson, J. Sjöstrand, P. Strimling, Optimal expected rank in a two-sided secretary problem. Oper. Res. **55**, 921–931 (2007)
46. D. Knuth, N. De Bruijn, in *Stable Marriage and Its Relation to Other Combinatorial problems: An Introduction to the Mathematical Analysis of Algorithms*. (American Mathematical Society, Providence, 1997)
47. J. Wu, Stable matching beyond bipartite graphs, in *2016 IEEE International Parallel and Distributed Processing Symposium Workshops (IPDPSW)* (2016), pp. 480–488
48. K. Iwama, S. Miyazaki, K. Okamoto, Stable roommates problem with triple rooms, in *Proceeding of the 10th KOREA–JAPAN Joint Workshop on Algorithms and Computation (WAAC)* (2007), pp. 105–112
49. R. Irving, An efficient algorithm for the "stable roommates" problem. J. Algorithms. **6**, 577–595 (1985)
50. L. Cui, W. Jia, Cyclic stable matching for three-sided networking services. Comput. Netw. **57**, 351–363 (2013)

51. C. Dai, X. Wang, K. Liu, D. Qi, W. Lin, P. Zhou, Stable task assignment for mobile crowdsensing with budget constraint. IEEE Trans. Mob. Comput. **20**, 3439–3452 (2020)
52. G. Yang, B. Wang, X. He, J. Wang, H. Pervaiz, Competition-congestion-aware stable worker-task matching in mobile crowd sensing. IEEE Trans. Netw. Serv. Manag. **18**, 3719–3732 (2021)
53. M. Abououf, S. Singh, H. Otrok, R. Mizouni, A. Ouali, Gale-shapley matching game selection—A framework for user satisfaction. IEEE Access. **7**, 3694–3703 (2018)
54. X. Yin, Y. Chen, C. Xu, S. Yu, B. Li, Matchmaker: stable task assignment with bounded constraints for crowdsourcing platforms. IEEE Internet Things J. **8**, 1599–1610 (2020)
55. Y. Chen, X. Yin, Stable job assignment for crowdsourcing, in *IEEE Global Communications Conference (GLOBECOM)* (2017), pp. 1–6
56. B. Holzbauer, B. Szymanski, E. Bulut, Socially-aware market mechanism for participatory sensing, in *Proceedings of ACM International Workshop on Mission-Oriented Wireless Sensor Networking (MiSeNet) at Mobicom 2012, Istanbul, Turkey, August 26, 2012* (2012), pp. 9–14
57. F. Yucel, E. Bulut, User satisfaction aware maximum utility task assignment in mobile crowdsensing. Comput. Netw. **172**, 107156 (2020)
58. F. Yucel, M. Yuksel, E. Bulut, QoS-based budget constrained stable task assignment in mobile crowdsensing. IEEE Trans. Mob. Comput. **20**, 3194–3210 (2020)
59. Y. Kawase, A. Iwasaki, Approximately stable matchings with budget constraints, in *Thirty-Second AAAI Conference on Artificial Intelligence* (2018), pp. 1113–1120
60. F. Yucel, E. Bulut, Time-dependent stable task assignment in participatory mobile crowdsensing, in *IEEE 45th Conference on Local Computer Networks (LCN)* (2020), pp. 433–436
61. M. Zhang, P. Yang, C. Tian, S. Tang, X. Gao, B. Wang, F. Xiao, Quality-aware sensing coverage in budget-constrained mobile crowdsensing networks. IEEE Trans. Veh. Technol. **65**, 7698–7707 (2016)
62. Z. Feng, Y. Zhu, Q. Zhang, L. Ni, A. Vasilakos, TRAC: truthful auction for location-aware collaborative sensing in mobile crowdsourcing, in *IEEE Conference on Computer Communications (INFOCOM), Toronto, Canada* (2014), pp. 1231–1239
63. F. Yucel, M. Yuksel, E. Bulut, Coverage-aware stable task assignment in opportunistic mobile crowdsensing. IEEE Trans. Veh. Technol. **70**, 3831–3845 (2021)
64. D. Rawitz, A. Rosén, Online budgeted maximum coverage, in *24th Annual European Symposium on Algorithms (ESA)* (2016)
65. F. Yucel, E. Bulut, Online stable task assignment in opportunistic mobile crowdsensing with uncertain trajectories. IEEE Internet Things J. **9**, 9086–9101 (2021)
66. D. Knuth, in *Mariages Stables* (Les Presses De L'Universit De Montral, Montréal, 1976)
67. C. Ng, D. Hirschberg, Three-dimensional stable matching problems. SIAM J. Discret. Math. **4**, 245–252 (1991)
68. B. Li, Y. Cheng, Y. Yuan, G. Wang, L. Chen, Three-dimensional stable matching problem for spatial crowdsourcing platforms, in *25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining* (2019), pp. 1643–1653
69. T. Song, Y. Tong, L. Wang, J. She, B. Yao, L. Chen, K. Xu, Trichromatic online matching in real-time spatial crowdsourcing, in *IEEE 33rd International Conference on Data Engineering (ICDE)* (2017), pp. 1009–1020
70. B. Zheng, C. Huang, C. Jensen, L. Chen, N. Hung, G. Liu, G. Li, K. Zheng, Online trichromatic pickup and delivery scheduling in spatial crowdsourcing, in *IEEE 36th International Conference on Data Engineering (ICDE)* (2020), pp. 973–984
71. Y. Wang, H. Li, T. Li, Participant selection for data collection through device-to-device communications in mobile sensing. Pers. Ubiquit. Comput. **21**, 31–41 (2017)
72. E. Bulut, S. Geyik, B. Szymanski, Utilizing correlated node mobility for efficient DTN routing. Pervasive Mob. Comput. **13**, 150–163 (2014)
73. M. Xiao, J. Wu, L. Huang, Y. Wang, C. Liu, Multi-task assignment for crowdsensing in mobile social networks, in *IEEE Conference on Computer Communications (INFOCOM)* (2015), pp. 2227–2235
74. M. Xiao, J. Wu, L. Huang, R. Cheng, Y. Wang, Online task assignment for crowdsensing in predictable mobile social networks. IEEE Trans. Mob. Comput. **16**, 2306–2320 (2017)

75. Y. Han, H. Wu, Minimum-cost crowdsourcing with coverage guarantee in mobile opportunistic D2D networks. IEEE Trans. Mob. Comput. **16**, 2806–2818 (2017)
76. F. Yucel, E. Bulut, Location-dependent task assignment for opportunistic mobile crowdsensing, in *IEEE 17th Annual Consumer Communications & Networking Conference (CCNC)* (2020), pp. 1–6