LSTAloc: A Driver-Oriented Incentive Mechanism for Mobility-on-Demand Vehicular Crowdsensing Market

Chaocan Xiang, Wenhui Cheng, Chi Lin, Xinglin Zhang, Daibo Liu, Xiao Zheng*, Zhenhua Li

Abstract—With the popularity of Mobility-on-Demand (MOD) vehicles, a new market called MOD-Vehicular-Crowdsensing (MOVE-CS) was introduced for drivers to earn more by collecting road data. Unfortunately, MOVE-CS failed after two years of operation. To identify the root cause, we survey 581 drivers and reveal its simple incentive model based on blindly competitive rewards. This model brings most drivers few yields, resulting in their withdrawals. In contrast, a similar market termed MOD-Human-Crowdsensing (MOMAN-CS) remains successful thanks to a complex model based on exclusively customized rewards. Hence, we wonder whether MOVE-CS can be resurrected by learning from MOMAN-CS. Despite considerable similarity, we can hardly apply the incentive model of MOMAN-CS to MOVE-CS, since MOD drivers are also concerned with passenger missions that dominate their earnings. To this end, we analyze a large-scale dataset of 12,493 MOD vehicles, finding that drivers have explicit preference for short-term, immediate gains as well as implicit rationality in pursuit of long-term, stable profits. Therefore, we design a novel driver-oriented incentive mechanism for MOVE-CS, called *LSTAloc*, at the heart of which lies a spatial-temporal differentiation-aware task allocation scheme empowered by submodular optimization. Applied to the dataset, our design would essentially benefit both the drivers and platform to incentivize MOD vehicular crowdsensing efficiently, thus possessing the potential to resurrect MOVE-CS.

Index Terms—Incentive Mechanism, Vehicular Crowdsensing, Task Allocation, Submodular Optimization.

1 Introduction

Recent years have witnessed the prosperity of the Mobility-on-Demand (MOD) vehicle market, led by Uber, Lyft, DiDi, and so forth [2]. As of December 2020, Uber and Lyft had each incorporated over one million drivers in the U.S. [3], and the global market is reaching $228 billion by 2022 [4]. However, we notice that many MOD drivers have suffered from shrinking earnings year by year from 2013 to 2020 [5], probably owing to more competition among them; the circumstance has been further aggravated in the past nearly two years due to the recent COVID-19 pandemic [6]. As a result, a new market termed MOD-Vehicular-Crowdsensing (MOVE-CS) was introduced in 2017, pioneered by the Payver platform [7]. Payver pays the drivers to collect road data on the move, mainly according to the road length and the specific segments, typically at $0.01-0.05 per mile.

After receiving the collected road data from the drivers, Payver usually sold them to demanding companies such as digital map construction corporations (*e.g.,* Google Maps [8] and lvl5 [9]). Thereby, the platform and the drivers seemed to have achieved a win-win situation. After only three months of operation, Payver had taken in nearly 2000 Uber and Lyft drivers, collecting the data of more than 500K-mile roads and boosting their earnings by 5% to 15% [10]. Unfortunately, after two years of operation, there remained few participant drivers, and thus Payver had to bankrupt itself in April 2019 [11].

To figure out the root cause of the aforementioned adversity, we surveyed 581 MOD drivers (clarified in Sec. 2.1) via Amazon Mechanical Turk, a well-known crowdsourcing platform [12]. They comprise 41.2% of women and 58.8% of men, aged from 20 to 60; 43.6%, 77.3%, and 90.2% of them drive at least once every day, week, and month, respectively. The survey results unveil that MOVE-CS drivers' withdrawals are highly related to the simple incentive model adopted by Payver based on blindly competitive rewards. Specifically, because each driver collects data for certain road segments without the knowledge of others, they often end up with low-novelty collected data for repetitive road segments. Hence, this model leads most drivers into few or even negative yields (*e.g.,* when the sensing task is performed while the vehicle is vacant), triggering their opt-outs from the MOVE-CS market.

Contrary to the MOVE-CS market, we spot that a similar market named MOD-Human-Crowdsensing (MOMAN-CS), led by Gigwalk [13], preserves its success since 2010. It hires people to collect merchandise data (*e.g.,* the location, price, and sales) for specific vendors, and has incorporated 1.7 million participants by 2021 [13]. Behind the success of

- * *Corresponding authors*
- *Chaocan Xiang and Wenhui Cheng are with the College of Computer Science, Chongqing University, Chongqing, China. E-mail: xiangchaocan@cqu.edu.cn.*
- *Chi Lin is with School of Software Technology, Dalian University of Technology; Key Laboratory for Ubiquitous Network and Service Software of Liaoning Province, Dalian, China. E-mail: c.lin@dlut.edu.cn.*
- *Xinglin Zhang is with the School of Computer Science and Engineering, South China University of Technology, China. E-mail: zhxlinse@gmail.com.*
- *Daibo Liu is with the College of Computer Science and Electronic Engineering, Hunan University, Changsha, China. E-mail: dbliu@hnu.edu.cn.*
- *Xiao Zheng is with the School of Computer Science and Technology, Anhui University of Technology, Maanshan, China and also with the Institute of Artificial Intelligence, Hefei Comprehensive National Science Center, Hefei, China. E-mail: xzheng@ahut.edu.cn.*
- *Zhenhua Li is with the School of Software, Tsinghua University, Beijing, China. E-mail: lizhenhua1983@gmail.com.*

- *This work is partially supported by NSF China under Grants No. 62172063. A preliminary version of this work appeared in IEEE INFOCOM 2022 [1].*

Gigwalk, we find a complex incentive model with exclusively customized rewards. Specifically, for a task, Gigwalk posts an initial reward and only allows one person to accept it; if no one takes it on for a long time, the reward will be increased. Now that the incentive model of MOMAN-CS is capable of incentivizing humans effectively, and humans steer the vehicles, we wonder if this model can be applied to driver incentivization in hopes of resurrecting the MOVE-CS market.

Since there is considerable similarity between the two markets, most incentive mechanisms in MOMAN-CS can be borrowed to improve MOVE-CS. For example, road data collection in MOVE-CS can be divided into exclusive sensing tasks for drivers to choose. Also, those untrodden road segments can be assigned with more rewards. Nevertheless, we find a key obstacle during the applying process, *i.e.*, the MOD drivers are also concerned with passenger missions which typically dominate their earnings. Therefore, the task allocation strategy in MOVE-CS should differ significantly from that in MOMAN-CS.

To practically address this obstacle, we analyze a large-scale dataset[1] of 12,493 MOD vehicles' service records during a whole month (03/01/2017–03/31/2017) in a 4,400 km^2 metropolitan area with 10.3 million residents. In detail, it includes the pick-up/drop-off locations, time-variant occupied/vacant statuses, and fine-grained vehicle trajectories for each passenger mission (explicated in Sec. 2.2). The analyses reveal that:

(1) On a daily basis, we observe that the majority (88.2%) of drivers move from low-yield zones to high-yield zones for picking up passengers, showcasing their *explicit preference* for *short-term, immediate gains*.
(2) On a monthly basis, however, we note that a considerable portion (30%) of drivers still drive from high-yield zones to low-yield zones for picking up passengers with a high occurrence rate of 21.1%. Surprisingly perhaps, we find their hourly earnings to be 17.5% more than the average level ($126.6 monthly raise), uncovering their *implicit rationality* in pursuit of *long-term, stable profits*.

Motivated by these findings, we present Long-Short-Term Profit-combined Task Allocation (*LSTAloc*), a novel driver-oriented incentive mechanism for MOVE-CS, whose primary goal is to satisfy both drivers' explicit preference for short-term gains and their implicit need of long-term profits. To this end, *LSTAloc* actively allocates tasks to the drivers with balanced intelligence, in order to not only quickly attract more participants, but also bring sufficient profits to regular drivers. Meanwhile, *LSTAloc* should also take the platform's profit into account. In practice, in some cases the interests of drivers and the platform are in correspondence, *e.g.*, when the platform allocates a task enabling a driver to go from a low-yield to a high-yield zone, this driver is very likely to accept it even with a relatively low reward. In other cases, their interests might be in conflict, *e.g.*, for an untrodden road segment whose information is however valuable to the platform, the platform has to offer a relatively high reward to motivate drivers.

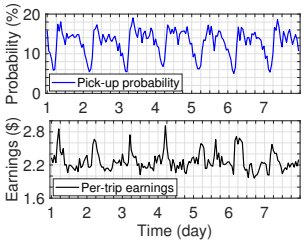
To address the challenges mentioned above, we design a spatial-temporal differentiation-aware task allocation scheme empowered by submodular optimization. In specific, based on the historical MOD vehicle dataset, we first construct a two-dimensional pick-up profit heatmap and predict the evolution of the profit heatmap by exploiting Recurrent Neural Networks (RNN). Based on this heatmap, we present the differentiation-aware sensing reward design. Then, we estimate each driver's acceptance probability by leveraging their mobility model. With the above information, we formulate a task allocation problem considering both new and regular drivers' concerns, as well as the platform's profit. Unfortunately, it is NP-hard to find the optimal solution to the problem (the computation cost increases exponentially with the number of drivers). To resolve this, following the methodology of submodular optimization, we devise a near-optimal algorithm, leveraging greedy local-search to achieve a guaranteed near-optimal solution.

To summarize, we make three key contributions:

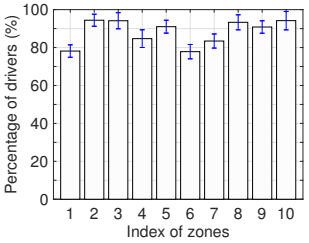
(1) *Novel driver-oriented incentive mechanism design for MOVE-CS*: By surveying 581 drivers and analyzing a large-scale dataset of 12,493 MOD vehicles, we find that drivers have *explicit* preference for short-term, immediate gains as well as *implicit* rationality in pursuit of long-term, stable profits. Inspired by the findings, we propose *LSTAloc*, a novel driver-oriented incentive mechanism, which not only satisfies the MOD drivers' explicit and implicit needs but also takes the platform's profit into account.
(2) *Guaranteed near-optimal task allocation*: We present a spatial-temporal differentiation-aware task allocation scheme empowered by submodular optimization, including the pick-up heatmap prediction based on RNN, the differentiation-aware sensing reward design, the mobility-model-driven acceptance probability estimation, and the submodularity-based task allocation algorithm. A series of theorectic analyses prove that it can achieve an acceptable approximation ratio $(1 - e^{-2})/2$ with polynomial time complexity.
(3) *Extensive emulations based on a large-scale vehicular dataset*: Using the aforementioned MOD vehicle dataset, we emulate the operation process of the original MOVE-CS model and *LSTAloc* respectively on a common commodity server. Results show that with *LSTAloc*, 87.3% of the allocated tasks cater for drivers' explicit preference of short-term gains; meanwhile, all the drivers are expected to make positive earnings and 50% of them make 3.2× more earnings (than with the original MOVE-CS model), serving their implicit need of long-term profits. Besides benefiting the drivers, *LSTAloc* brings 34.3% more profit to the platform.

The rest of this paper is organized as follows. We first introduce the motivation based on user studies and large-scale data analysis in Sec. 2. Then, we provide the novel incentive model called *LSTAloc* and present the crucial research problem in Sec. 3, followed by designing its key algorithms in Sec. 4. In Sec. 5, we conduct extensive evaluations based on the large-scale MOD vehicle dataset. Moreover, we make

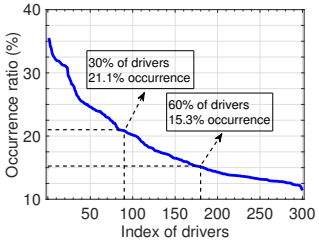
1. We collected all the data (excluding user-sensitive information) under a well-organized IRB with informed consent of involved drivers and passengers.



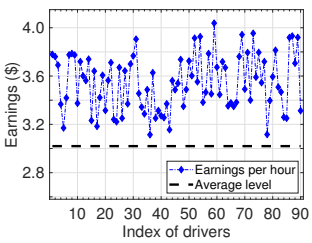
(a) Temporal differentiation analysis



(a) Drivers' short-term behavior



(b) Drivers' occurrence ratios



(c) Drivers' earnings per work hour

Fig. 1: Variations of pick-up probability (top) and per-trip earnings (bottom) in different time periods for the same zone.

Fig. 2: In-depth analysis of drivers' behavior patterns via 2D slicing. (a) Drivers' short-term preference on a daily basis, *i.e.*, the percentage of drivers moving from low-yield zone to high-yield zone. (b-c) Drivers' long-term pursuit on an individual basis, including their occurrence ratios of driving from high-yield zone to low-yield zone, and their average earnings in a month.

discussions in Sec. 6 and review the related works in Sec. 7. Finally, we conclude this paper in Sec. 8.

2 Motivation

In this section, we investigate the reasons behind the downturn of the MOVE-CS market via user studies and large-scale data analysis. Then, the results are fed back to explore potential methods to incentivizing MOD vehicular crowdsensing efficiently for resurrecting the MOVE-CS market.

2.1 Crowdsourcing-based User Studies

Methodology. To investigate why the above two markets faced completely different fates, we conduct user studies [14] with 581 MOD drivers via Amazon Mechanical Turk. The respondent pool is restricted to qualified drivers. The participants comprise 41.2% of women and 58.8% of men, including North Americans (34.4%), Europeans (12.7%), Asians (38.6%), and others (14.3%, such as Australians and Africans), aging from 20 to 60; 43.6%, 77.3%, and 90.2% of them drive at least once every day, every week, and every month, respectively.

We adopt the USE questionnaire methodology [15] and use a 5-point Likert scale (ranging from Strongly Disagree to Strongly Agree) to assess the participants' perceptions. Results are classified into two groups, *i.e.*, 4 and 5 for agreement; 1, 2, and 3 for disagreement. The queries are designed to get to the bottom of two key questions, *i.e.*, why does the MOVE-CS model fail to encourage MOD drivers? why is the MOMAN-CS model effective to incentivize users?

Results. We first investigate whether the MOD drivers are willing to perform sensing tasks. Survey results indicate *92.6% of participants are willing to perform sensing tasks on the move*. Digging deeper, it seems related to the fact that the majority (63.8%) of drivers take on sensing tasks with expectations of extra earnings, conforming to common sense.

Further survey on the two models shows that the blind competition model adopted by Payver is not welcomed by 63.3% of participants; 94.3% regard the repeated data collection—which may cause a lower or even negative profit—as a major drawback. Therefore, it is reasonable to deduce that the blind competition model introduces uncertainty in drivers' profits, which severely impacts their enthusiasm for task participation. Contrarily, 95.2% of participants prefer MOMAN-CS, because it not only has transparent rewards (70.2% agreement), but also gives them more choices of tasks (81.3% agreement). To sum up, *MOVE-CS' downfall was likely a result of the employed blind competition model failing to offer drivers stable profits*, while *MOMAN-CS motivates participants successfully with its exclusive task selection and transparent reward*.

2.2 Large-scale Dataset Collection and Analysis

Dataset collection. Cooperating with a MOD company, we acquire a large-scale MOD driver dataset; all the user-sensitive information is removed according to the local IRB protocols. This dataset comprises 92 GB service records of 12,493 MOD vehicles for one month (03/01/2017–03/31/2017) in a 4,400 km^2 metropolitan area with 10.3 million residents. Each record contains an anonymized vehicle ID, the trajectory time series with an interval of 15 seconds, and an occupied/vacant state indicator. Moreover, with the trajectory series, we calculate the pick-up profits according to the existing policies on MOD vehicle fares [16].

Pick-up profit analysis. The pick-up profit denotes the average profit of MOD drivers from picking up passengers in a zone during a time period (*e.g.*, 1 hour). It highly depends on this zone's pick-up probability and per-trip earnings. The pick-up probability refers to the proportion of vehicles that pick up passengers out of all the vacant vehicles in the zone within a time period. The per-trip earnings represent the average income earned by vehicles in the zone during a time period. Hence, in the following, we randomly select an area (about 256 km^2) of the city, and divide it into 14×18 uniform zones. Then, we analyze the spatial-temporal differences of pick-up profits in each zone and time period in terms of the pick-up probability and the per-trip earnings.

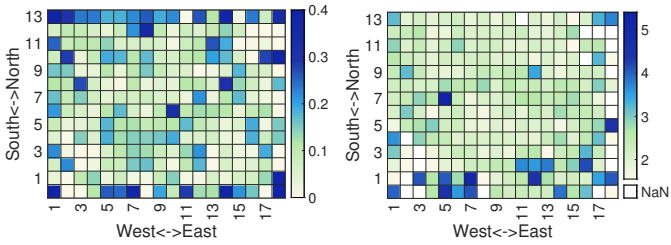
We initially analyze the temporary diversity of pick-up probability and per-trip earnings during different time periods in a randomly selected zone. As demonstrated in Fig. 1a, both the pick-up probability (top) and the per-trip earnings (bottom) vary significantly with time. In particular, the pick-up probability and per-trip earnings is distributed between 5.0% and 19.1%, \$1.96 and \$2.91, respectively. Moreover, both pick-up probability and per-trip earnings roughly follow periodic patterns, *e.g.*, Fig. 1a illustrates that

the pick-up probability from midnight to 6 a.m. is always smaller than that of the other periods in a day, as most citizens are asleep. Next, we analyze their spatial diversities in different zones during a time period (*e.g.,* 6 p.m. to 7 p.m.). As demonstrated in Figs. 3a and 3b, similarly, both the pick-up probability and the per-trip earnings are found to vary with zones in the same period. In particular, the pick-up probability and per-trip earnings widely fluctuate between 0 and 39%, \$1.55 and \$5.41, respectively. In summary, *the pick-up profits of MOD drivers have huge spatial-temporal differences in different zones and time periods.*

MOD drivers' behavior analysis. MOD drivers have behavior patterns with a common goal (making money) but diverse individual preferences (such as how to make more money) based on driving experience. To fully grasp their behavior patterns, we conduct a comprehensive analysis of the large-scale dataset by slicing it — in the aspects of both short-term and long-term profits — on a daily basis and an individual basis, respectively.

First, we slice the dataset on a daily basis to study drivers' short-term preference in each day. Then, we select ten low-yield zones randomly. Targeting each zone, we calculate the corresponding percentage of drivers, who move directly (from this low-yield zone) into a high-yield zone for picking up passengers. As shown in Fig. 2a, the average percentage of one month in all selected zones is 88.2%. It indicates *most drivers in low-yield zones have a tendency of moving out (towards higher-yield zones), which is compelling evidence of drivers' explicit preference for immediate gains*.

Second, to understand drivers' long-term pursuit, we slice the dataset on an individual basis, each slice with the entire one-month driving records of a driver. Then, we randomly select 300 drivers. Focusing on each driver's behavior pattern, we calculate the occurrence of her/his moving, in the entire month, from a high-yield zone into a low-yield zone for passenger pick-ups. After ranking drivers, as shown in Fig. 2b, we find that 30% of them (90 drivers) have more than 21.1% occurrence, which appears to be weird at first glance. By comparing the hourly pick-up profits of these 90 drivers against the average level of all 12,493 drivers, surprisingly, we find that these drivers make 17.5% more pick-up profits per work hour than the average level (about \$126.6 monthly raise considering the 8-hour work day), as shown in Fig. 2c. After a thorough analysis of these findings, the mystery finally uncovers its veil: *regular drivers possess the ability of dynamic profit prediction to some degree, and rationally choose where to go based on this knowledge in pursuit of long-term, stable profits, rather than blindly seek the immediate gains*.



(a) Pick-up probability **(b)** Per-trip earnings

Fig. 3: Pick-up profits in different zones during the same time period, in terms of (a) the pick-up probability and (b) the per-trip earnings (\$).

3 *LSTAloc* Design for MOVE-CS

Motivated by the findings in Sec. 2, we design a new driver-oriented incentive mechanism model called *LSTAloc* for resurrecting the MOVE-CS market, and advance the crucial research problem.

3.1 Incentive Model Design

Logic behind the design. *LSTAloc* leverages the *active task allocation of the platform* to simultaneously satisfy MOD drivers' explicit and implicit needs for short-term and long-term profits, respectively. The logic behind this model design is as follows:

(1) The dataset analysis in Sec. 2.2 shows that MOD drivers have an explicit preference for short-term, immediate gains as well as implicit rationality in pursuit of long-term, stable profits. Therefore, the model design should respond to drivers' demands to encourage their participation.

(2) However, the short-term and long-term profits can only be predicted with global knowledge of the pick-up profits at any time and place, which is barely possible even for regular drivers, let alone new registrants. Therefore, it is unrealistic to let drivers actively select from all the tasks in a short response time, while gaining acceptable profits.

(3) Hence, we deploy the task allocation scheme instead of task selection by drivers like the MOMAN-CS model. Specifically, the professional platform with enough spatial-temporal knowledge predicts the pick-up profits and the acceptance probabilities on drivers' behalf, simultaneously actively allocating tasks by comprehensively considering their short-term and long-term profits.

To sum up, *the LSTAloc model saves drivers from the extremely complicated computation of profit prediction, reducing the response time of each driver, while they are still left enough wiggle room for options. Simultaneously, the platform can also pursue its own interest in this process*.

***LSTAloc* model design.** To begin with, our *LSTAloc* model consists of three major steps as follows:

(1) *Task publishing*: The new MOVE-CS platform discretizes the required road data collection into J exclusive sensing tasks, according to the topology and length of the roads as well as the specified applications. We denote the set of these published tasks by $\mathcal{J}$, *i.e.,* $\mathcal{J} = \{1, \ldots, j, \ldots, J\}$. The zone of each task j ($j \in \mathcal{J}$) and the platform profit from it are represented by z_j and u_j, respectively.

(2) *Task requesting and allocation*: There are large numbers of MOD drivers delivering passengers in the city, willing to opt in MOVE-CS. Let $\mathcal{K}$ denote the set of these MOD drivers, *i.e.,* $\mathcal{K} = \{1, \ldots, k, \ldots K\}$. Each driver k ($k \in \mathcal{K}$) reports her/his current zone z_k. Then, the platform allocates one task for each driver, along with its location&reward and the driver's expected profit. Let x_{kj} denote whether task j is allocated to driver k, *i.e.,* $x_{kj} = 1$ if yes, and $x_{kj} = 0$ otherwise. The allocation set is then denoted by $\mathbf{x} = [x_{kj}]_{K \times J}$.

(3) *Task acceptance and execution*: Once a driver k is allocated task j, she/he has a probability ρ_{kj} of accepting and

performing it, called the *acceptance probability*. Each driver's acceptance probability depends on not only their preferences on the tasks, but also their tendencies to the zones of the tasks [17], [18]. To assure a high task execution probability, a task may be allocated to multiple drivers. Hence, the execution probability of each task j can be calculated by $1-\prod_{k=1}^{K}(1-\rho_{kj})^{x_{kj}}$. Similar to references [19], [20], the total platform profit, *i.e.,* the expected profits of all the performed sensing tasks for the platform, is represented by

$$U(\mathbf{x})=\sum_{j=1}^{J}u_j\big(1-\prod_{k=1}^{K}(1-\rho_{kj})^{x_{kj}}\big). \tag{1}$$

Finally, after driver k accepts and completes the allocated sensing task j (*e.g.,* uploading his/her collected road data to the MOVE-CS platform), she/he will receive a reward from the platform, represented by c_{jk}; the rewards of all drivers then form a set $\mathbf{c}=[c_{kj}]_{K\times J}$. Hence, the driver's total earnings equal to the sum of the reward by performing a sensing task and the pick-up profit by transporting passengers. As illustrated in Table 1, we list the frequently used notations of this paper.

3.2 Research Problem and Challenge Analysis

The crucial problem in the *LSTAloc* model design is how to allocate tasks to drivers alongside proper rewards, by study and prediction of the spatial-temporal differences of pick-up profit and the driver's acceptance probability, which is presented in the following.

Long-short-term profit-aware optimal task allocation problem (*LSTO*): *Given the historical MOD vehicle dataset, how to allocate each task j to a MOD driver k with the sensing reward $[c_{kj}]_{K\times J}$, so as to maximize the total platform profit $U(\mathbf{x})$ under the constraint of budget B, while satisfying both drivers' explicit preference for short-term gains and their implicit needs of long-term profits.*

$$\underset{[x_{ij}],[c_{kj}],[\rho_{kj}]}{\text{Maximize}}\quad U(\mathbf{x})=\sum_{j=1}^{J}u_j\big(1-\prod_{k=1}^{K}(1-\rho_{kj})^{x_{kj}}\big), \tag{2}$$

$$\text{s.t.}\quad \sum_{j=1}^{J}x_{kj}\le 1,\forall k\in\{1,\dots,K\}, \tag{3}$$

$$\sum_{k=1}^{K}\sum_{j=1}^{J}x_{kj}\rho_{kj}c_{kj}\le B. \tag{4}$$

Note that, Eq. (3) restricts each driver is allocated at most one task in each allocation period, similar to references [17], [21]. Eq. (4) indicates that the sensing rewards of all the drivers are not more than the platform's budget B, so as to maximize the platform profit within the limited budget.

In addressing the above-mentioned problem, there exist three main challenges as follows:

(1) *It is difficult to predict the global distribution of the pick-up profits, due to their spatial-temporal dynamics*. The pick-up profits exhibit spatial-temporal dynamics, as demonstrated in Figs. 1a, 3a, and 3b. Furthermore, the highly complicated movement of both passengers and MOD vehicles between different zones and time complicates such dynamics, hence rendering the accurate prediction on the global distribution of pick-up profits particularly difficult.

(2) *It is challenging to satisfy the demands of both drivers and the platform, which are aligned in some cases but conflicted in others*. In some circumstances, a task may require sensing in a high-yield zone where drivers are eager to move towards, so that they will probably accept it with a relatively low reward, in alignment with the platform's interest. In other conditions, a task valuable to the platform is perhaps related to an unpopular zone, where drivers are reluctant to go. The two sides do not share mutual benefits so that drivers will only undertake the task if the reward is high enough to reach their expectations, thereby increasing the platform's cost.

TABLE 1: Frequently Used Notations

Notations	Descriptions
j, J, $\mathcal{J}$	sensing task j, its total number, task set.
k, K, $\mathcal{K}$	MOD driver k, its total number, driver set.
z_i, Z, $\mathcal{Z}$	zone i, its total number, zone set.
t, T, $\mathcal{T}$	period t, its total number, period set.
x_{kj}	whether task j is allocated to driver k.
ρ_{kj}	the acceptance probability of driver k for task j.
p_i^t, r_i^t	pick-up probability and per-trip income in z_i at t.
$I_{kj}(t)$	pick-up profit difference of driver k driving to z_j at t.
$c_{kj}(t)$	sensing reward for executing task j by driver k at t.
$b_{kj}(t)$	driver k's expected income for pick-up from z_k to z_j.
$\mathbf{x}$, $\mathbf{c}$	$\mathbf{x}=[x_{kj}]_{K\times J}$, $\mathbf{c}=[c_{kj}]_{K\times J}$.
$\mathcal{A}$, $\mathcal{V}$	allocated vehicle-task pairs set, ground set.
u_j, B	utility of task j, budget constraint of task excution.
$U(\cdot)$	sensing utility function.
$\mathbf{H}^T$, $\mathbf{h^t}$	heatmap within T periods, t-th heatmap frame.

(3) *It is intricate to achieve the optimal task allocation due to the NP-hardness and the unknown acceptance probability*. Given the acceptance probability and the sensing reward, the optimal task allocation problem is an NP-hard problem, according to Theorem 1. As a result, it is extremely challenging to achieve the optimal allocation with computational efficiency, especially for the large-scale MOVE-CS market with massive drivers (such as 12,493 drivers in our dataset). Even worse, the acceptance probability of each driver is previously unknown. Moreover, it is difficult to estimate the acceptance probability, since it depends on not only the task content (such as the task category), but also the context (such as the locations of drivers and tasks).

Theorem 1. *Given the sensing reward and the acceptance probability, the optimal task allocation problem is NP-hard.*

Proof. Consider a special case of this problem, where the platform's budget B is large enough for recruiting the drivers to execute all the tasks. Hence, constraint (4) can be relaxed. In this case, given the sensing reward $[c_{kj}]_{K\times J}$ and the acceptance probability $[\rho_{kj}]_{K\times J}$, the optimal task allocation problem can be reduced from the classical NP-hard *Target-based Weapon Target Assignment* problem [22] as follows: given J targets (*i.e.,* tasks) each with profit u_j ($\forall j\in\mathcal{J}$), and K weapons (*i.e.,* drivers) each of which (*i.e.,* weapon k) has a probability ρ_{kj} of destroying (executing) each target j (*i.e.,* task j), assign (allocate) each weapon (driver) k to exactly one target (task) j so as to maximize the expected destroying probability (the total expected profits) $\sum_{j=1}^{J}u_j(1-\prod_{k=1}^{K}(1-\rho_{kj})^{x_{kj}})$ for all targets (tasks). Therefore, Theorem 1 is proved. □

4 Key Algorithm Design for *LSTAloc*

To address the above three challenges, we propose a spatial-temporal differentiation-aware task allocation scheme em-

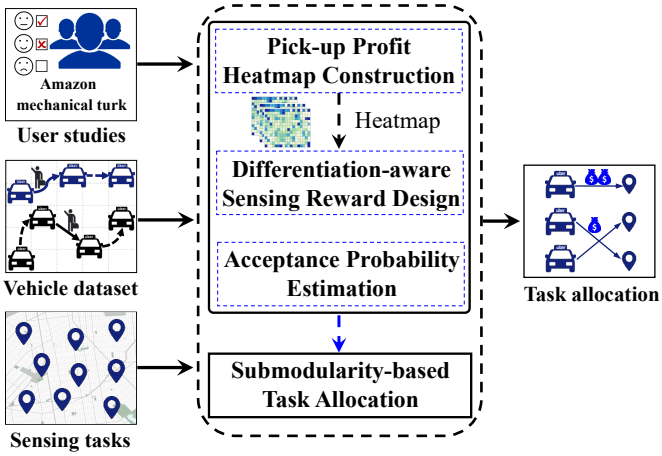


Fig. 4: Overview of our algorithm.

powered by submodular optimization. As illustrated in Fig. 4, it mainly consists of four components:

(1) *Pick-up profit heatmap construction* (Sec. 4.1): Utilizing the historical MOD vehicle dataset, we first construct the two-dimensional pick-up profit heatmaps, which are then used to predict the future heatmaps by exploiting dual-attention-based RNN.

(2) *Differentiation-aware sensing reward design* (Sec. 4.2): Based on the global knowledge of the pick-up profit heatmaps, we learn the spatial-temporal dynamics of pick-up profits, which is fed back to devise the sensing rewards for satisfying the driver's explicit and implicit needs of the short-term and long-term profits, respectively.

(3) *Acceptance probability estimation leveraging drivers' mobility model* (Sec. 4.3): Inspired by the predictable mobility of MOD drivers, we leverage the drivers' mobility model to estimate the movement probability of drivers from one zone to another. The results are then fed back to estimate the acceptance probability of drivers from the aspects of their content preference and context preference.

(4) *Submodularity-based task allocation* (Sec. 4.4): Given the sensing reward design and the acceptance probability estimation, we first analyze the properties of the optimal task allocation problem by reformulation. Guided by the analysis results, we present an approximation algorithm to address this NP-hard problem, following the methodology of submodular optimization.

4.1 Pick-up Profit Heatmap Construction

Two-dimensional pick-up profit heatmap model. The pick-up profit is highly dependent on the pick-up probability and the per-trip earnings in each zone. Furthermore, the dataset analysis in Sec. 2.2 indicates that both the pick-up probability and the per-trip earnings have the spatial-temporal dynamics in different zones and time periods. As a result, we use the two-dimensional heatmaps, called *pick-up profit heatmaps*, to represent the dynamic spatial-temporal pick-up profits.

In particular, we divide the map of an entire city into Z non-overlapping zones, according to the shape of the area and the specified spatial granularity. Let z_i and $\mathcal{Z}$ denote each zone i and the set of zones, respectively, such that $z_i \in \mathcal{Z}$. Similarly, the time is evenly divided into T time slots, and the set of time slots is denoted as $\mathcal{T}$. Each time slot t is also named period t. Let p_i^t and r_i^t denote the pick-up probability and the per-trip earnings in zone i at period t. Hence, the pick-up profit heatmaps $\mathbf{H}^T$ during the periods $[1, T]$ are represented as

$$\mathbf{H}^T = \{\mathbf{h}^t | 1 \le t \le T\}, \tag{5}$$

$$\mathbf{h}^t = \{(p_i^t, r_i^t) | \forall i \in \mathcal{Z}\}, \tag{6}$$

where $\mathbf{h}^t$ denotes the t-th frame of the heatmaps, representing the pick-up profits of all the zones at period t. Moreover, each pixel of the heatmap frame (*i.e.*, $(p_i^t, r_i^t) \in \mathbf{h}^t$) represents the pick-up probability and the per-trip earnings in zone i at period t. Intuitively, in the heatmaps, the warmer the color, the more the pick-up profits that drivers are expected to get (with higher pick-up probability and more per-trip earnings), as illustrated in Figs. 3a and 3b.

Heatmap construction based on MOD vehicle dataset. We construct the pick-up profit heatmap by using the historical MOD vehicle dataset, including their trajectories, occupied/vacant statuses, and pick-up earnings of each trip, as illustrated in Fig. 5a. Specifically, as illustrated in Fig. 5b, based on the trajectories of MOD vehicles and their occupied/vacant status information, we compute the ratio of the vehicles picking up passengers to all the vacant vehicles in zone i within period t as the pick-up probability p_i^t. Moreover, we compute the per-trip earnings r_i^t by using the average income of all the vehicles that pick up passengers from zone i at period t. Thus, we use the historical MOD vehicle dataset during the periods $[1, T]$ to construct the corresponding pick-up profit heatmaps, *i.e.*, $\mathbf{H}^T = \{(p_i^t, r_i^t) | \forall i \in \mathcal{Z}, 1 \le t \le T\}$.

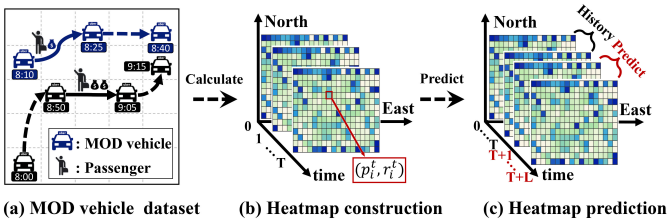
Heatmap prediction based on RNN. As shown in Fig. 5c, we utilize the pick-up profit heatmaps $\mathbf{H}^T$ of periods $[1, T]$ to accurately predict the future L periods. Note that, the prediction length L is dependent on the time interval of each task allocation in Sec. 4.2. In particular, the dual-attention based RNN [23], [24] is exploited to accurately predict the next profit heatmaps based on the historical ones. Its key idea is to leverage an LSTM-based encoder-decoder architecture with dual-attention mechanisms, including spatial attention and temporal attention. The spatial attention is exploited to capture the complex spatial correlations across different zones, while the temporal attention is used to learn the time-varying correlations between different time periods. In summary, we use the MOD vehicle dataset to construct and predict the global pick-up profit heatmaps of the periods $[1, T+L]$, *i.e.*, $\mathbf{H}^{T+L} = \{(p_i^t, r_i^t) | \forall i \in \mathcal{Z}, 1 \le t \le T+L\}$.

4.2 Differentiation-aware Sensing Reward Design

Based on the global pick-up profit heatmaps, we first compute the spatial-temporal differences of pick-up profits, which are used to design the sensing rewards of drivers.

The pick-up profit difference represents the expected pick-up profit increase for an unoccupied vehicle to drive from one zone to another for performing the sensing task. Formally, we let $I_{kj}(t_0)$ represent the pick-up profit difference for driver k, when moving from her/his original zone z_k to the task's zone z_j at t_0. Thus, $I_{kj}(t_0)$ is given by

$$I_{kj}(t_0) = \mathbb{E}_{t_0}[p_j^t \cdot r_j^t] - \mathbb{E}_{t_0}[p_k^t \cdot r_k^t], \tag{7}$$



(a) MOD vehicle dataset (b) Heatmap construction (c) Heatmap prediction

Fig. 5: Illustration of pick-up profit heatmap construction and prediction based on the MOD vehicle dataset using dual-attention-based RNN.

where $\mathbb{E}_{t_0}[\cdot]$ denotes the mathematical expectation with respect to t_0. $\mathbb{E}_{t_0}[p_k^t \cdot r_k^t]$ denotes the expected pick-up profit of driver k at her/his original zone z_k; $\mathbb{E}_{t_0}[p_j^t \cdot r_j^t]$ represents that of driver k at the new zone z_j. Both of them are dependent on the probability p_k^t (p_j^t) that drivers pick up passengers in z_k (z_j) at period t, and the probability $\bar{p}_k^t$ ($\bar{p}_j^t$) that drivers cannot do it in z_k (z_j) before t. Hence, we have

$$\mathbb{E}_{t_0}[p_k^t \cdot r_k^t] = \sum_{t=t_0}^{t_0+L_j} \bar{p}_k^t p_k^t r_k^t, \tag{8}$$

$$\mathbb{E}_{t_0}[p_j^t \cdot r_j^t] = \sum_{t=t_0+t_{kj}}^{t_0+t_{kj}+L_j} \bar{p}_j^t p_j^t r_j^t, \tag{9}$$

where $\bar{p}_k^t = 1 + \mathrm{sgn}(t-1)\big(\prod_{m=1}^{t-1}(1-p_k^{t-m}) - 1\big)$. L_j represents the number of time periods for task j's allocation interval, which is set according to specific applications. t_{kj} represents the spent time periods driving from z_k to z_j at t_0, which can be accurately estimated by existing effective algorithms [25] and real-time applications [8].

Learning each driver's pick-up profit difference, we design a sensing reward model, by subtracting it from her/his expected income. In specific, let $b_{kj}(t_0)$ denote driver k's expected income from passenger missions, when driving from z_k to z_j. Thus, according to Eq. (7), the sensing reward $c_{kj}(t_0)$ for executing task j by driver k at t_0 is represented as

$$c_{kj}(t_0) = b_{kj}(t_0) - (\mathbb{E}_{t_0}[p_j^t \cdot r_j^t] - \mathbb{E}_{t_0}[p_k^t \cdot r_k^t]). \tag{10}$$

Note that $b_{kj}(t_0)$ can be computed based on the driving time and distance [26], [27], according to the pricing policies of MOD vehicles [16].

Referring to the actual hourly wage of drivers, the platform offers each of them a reward, according to their expected profit, so that drivers are willing to spend time on the sensing tasks. A driver's expected profit is a combination of her/his explicit reward $c_{kj}(t_0)$ directly given by the platform, and implicit reward $\mathbb{E}_{t_0}[p_j^t \cdot r_j^t] - \mathbb{E}_{t_0}[p_k^t \cdot r_k^t]$ subtly obtained by relocation to a higher-yield zone for task j. Thus, as shown in Eq. (10), if the implicit reward is adequate, the platform can lower the explicit reward; if it is insufficient, the platform should offer a higher explicit reward for compensation. As a result, all drivers are granted more profits than the rewards from only passenger missions, named positive profits. In sum, *the sensing reward design based on the pick-up profit differentiation learning can ensure positive profits of all the drivers by balancing the explicit and implicit rewards.*

4.3 Acceptance Probability Estimation Leveraging Drivers' Mobility Model

The acceptance probability depends on two main factors (including the *content preference* and the *context preference*) for the following reasons. First, the task's content (such as its category, description, and time cost) is an important factor influencing the driver's decision on the task's acceptance [17]. Second, the task's context, namely the zone in which the task is located, also plays a crucial role in the driver's decision on the task's acceptance [28]. For instance, if a task is located in a commercial area, the driver may prefer to accept it due to more opportunities for picking up passengers [28]. As illustrated in Fig. 6a, when the drivers are located in the 1-st zone, they have a higher movement probability towards the 2-nd zone (*i.e.*, a commercial area) than the 3-rd zone (*i.e.*, a residential area). Moreover, the higher the movement probability towards the 2-nd zone, the more willingly drivers will accept the tasks of this zone. In the following, we first leverage the driver's mobility model to estimate the acceptance probability from the aspects of both the content preference and the context preference. And then, we use the vehicular dataset to validate the mobility model.

Acceptance probability estimation based on mobility model. We utilize γ_j to represent drivers' content preference on task j, *i.e.*, $\gamma_j \in (0,1)$. Similar to references [17], [18], the content preference γ_j can be previously set according to the task category, the task description, *etc.* The *context preference*, as mentioned above, highly depending on the drivers' movement, can be characterized by the *movement probability* φ_{kj} that driver k moves from his/her localized zone k to the task's zone j at least once during the time period. Note that the time period is related to task j's allocation interval, which is determined by the specific task types. As a result, the acceptance probability can be modeled as

$$\rho_{kj} = \varphi_{kj}\gamma_j. \tag{11}$$

Assuming that the movement of MOD drivers from one zone to another follows a Poisson process [29], the movement probability that driver k moves from his/her localized zone k to zone j for n times during the time period can be represented as

$$\varphi_{kj}(n) = \Big(\frac{1}{\lambda_{kj}}\Big)^n \frac{e^{-\frac{1}{\lambda_{kj}}}}{n!}, \tag{12}$$

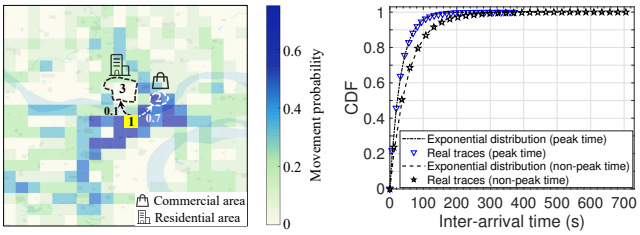
where λ_{kj} denotes the Poisson parameter, which is estimated based on the historical MOD vehicular dataset. In specific, for each driving case from zone k to zone j, we compute the time interval between two successive arrivals of zone j, called the *inter-arrival time*. We use the average inter-arrival time for all the driving cases from zone k to zone j as the λ_{kj}'s estimation.

Since the context preference φ_{kj} is quantified by the movement probability of driver k from zone k to zone j at least once, according to Eq. (12), we have

$$\varphi_{kj} = \sum_{n=1}^{+\infty} \varphi_{kj}(n) = 1 - e^{-\frac{1}{\lambda_{kj}}}. \tag{13}$$

Thus, according to Eqs. (11)(13), we can estimate the acceptance probability of driver k for task j as

$$\rho_{kj} = (1 - e^{-\frac{1}{\lambda_{kj}}})\gamma_j. \tag{14}$$



(a) Movement probability **(b)** CDF

Fig. 6: (a) Illustration of movement probability of MOD driver from one zone to another zone. (b) CDF of drivers' inter-arrival time in comparison with the Exponential distribution.

Dataset-based mobility model validation. We use the MOD vehicular dataset to validate the model assumption that the mobility model of MOD drivers follows a Poisson distribution. In other words, we verify whether the inter-arrival time of MOD drivers follows the Exponential distribution or not [20]. Specifically, as illustrated in Fig. 6b, we randomly select two zones and plot the CDF of the inter-arrival time of all driving cases from one zone to another during the peak time (7:00 am–9:00 am and 5:00 pm–8:00 pm) and the non-peak time (0:00 am–6:00 am), respectively. The results show that the average inter-arrival time is 31.6 and 52.6 seconds at peak and non-peak times, respectively. Moreover, Fig. 6b illustrates that the inter-arrival time distribution obtained from the real traces is well-fitted by the Exponential distribution. Furthermore, we use the Chi-square test method [30] to verify this model assumption. The test results are 20.7 and 30.8 for the peak and non-peak times, respectively. Given the significance level 0.05, both test results are less than the critical value (*i.e.*, $\chi^2_{28}(0.05) = 41.3$). As a result, the above results prove that *the mobility of MOD drivers from one zone to another follows a Poisson distribution* with 95% confidence probability.

4.4 Submodularity-based Task Allocation Algorithm

Problem analysis based on equivalent transformation. Firstly, given the sensing reward design $[c_{jk}]_{K\times J}$ and the acceptance probability $[\rho_{kj}]_{K\times J}$, we can equivalently transform the LSTO problem into a set function optimization problem for facilitating the problem analysis. In specific, we define the ground set $\mathcal{V} := \{v = (k,j) | \forall k \in \mathcal{K}, \forall j \in \mathcal{J}\}$. Let $\mathcal{A}$ denote the set of the allocated driver-task pairs, *i.e.*, $\mathcal{A} := \{v = (k,j) | x_{kj} = 1, \forall k \in \mathcal{K}, \forall j \in \mathcal{J}\}$, and $\mathcal{A} \subseteq \mathcal{V}$. Moreover, $\forall \mathcal{A} \subseteq \mathcal{V}$, $U(\mathcal{A}) := \{U(\mathbf{x}) | \forall (k,j) \in \mathcal{A}, x_{kj} = 1; \forall (k,j) \notin \mathcal{A}, x_{kj} = 0\}$. It is noted that $U(\mathcal{A})$ and $U(\mathbf{x})$ are different functions, but we adopt the same symbol $U(\cdot)$ for simplification. Based on the above definitions, the LSTO problem can be equivalently transformed as

$$\max_{\mathcal{A}\subseteq\mathcal{V}} \; U(\mathcal{A}) = \sum_{j:(k,j)\in\mathcal{A}} u_j \big(1 - \prod_{k:(k,j)\in\mathcal{A}} (1-\rho_{kj})\big), \tag{15}$$

$$\text{s.t.} \sum_{j:(k,j)\in\mathcal{A}} \mathbb{1}_{(k,j)\in\mathcal{A}} \le 1, \forall k \in \mathcal{K}, \tag{16}$$

$$\sum_{(k,j)\in\mathcal{A}} c_{kj}\rho_{kj} \le B, \tag{17}$$

where $\mathbb{1}$ denotes the indicator function, which maps the elements of the subset to 1, and the other elements to 0. Based on the set function optimization problem, we theoretically analyze the properties of its objective function and constraints as follows.

Definition 1. (**Non-negativity**, **Monotonicity**, **Submodularity** [31]): A set function $U(\cdot) : 2^{\mathcal{V}} \to \mathbb{R}$ is i) *non-negative*, if $\forall \mathcal{A} \subseteq \mathcal{V}, U(\mathcal{A}) \ge 0$ and $U(\emptyset) = 0$; ii) *monotone*, if $\forall \mathcal{A} \subseteq \mathcal{V}$ and $\forall v_1 \in \mathcal{V} \setminus \mathcal{A}$, $U(\mathcal{A} \cup \{v_1\}) \ge U(\mathcal{A})$; iii) *submodular*, if $\forall \mathcal{A} \subseteq \forall \mathcal{B} \subseteq \mathcal{V}$ and $\forall v_1 \in \mathcal{V} \setminus \mathcal{B}$, $U(\mathcal{A} \cup \{v_1\}) - U(\mathcal{A}) \ge U(\mathcal{B} \cup \{v_1\}) - U(\mathcal{B})$.

Lemma 1. The objective function $U(\mathcal{A})$ in Eq. (15) is *non-negative*, *monotone*, and *submodular*.

Proof. According to Def. 1, we prove that it is non-negative, monotone, and submodular one by one.

i) *Non-negativity*. If $\mathcal{A} = \emptyset$, $\forall k \in \mathcal{K}, \forall j \in \mathcal{J}, x_{kj} = 0$. Thus, according to Eq. (2), we have $U(\emptyset) = 0$. Moreover, since $U(\mathcal{A})$ represents the task utility that is non-negative, we have $\forall \mathcal{A} \subseteq \mathcal{V}, U(\mathcal{A}) \ge 0$. Thus, the objective function is non-negative.

ii) *Monotonicity*. For $\forall \mathcal{A} \subseteq \mathcal{V}$, $\forall v_1 = (k_1, j_1) \in \mathcal{V} \backslash \mathcal{A}$, according to Eq. (15), we have $U(\mathcal{A} \cup \{v_1\}) - U(\mathcal{A}) = u_{j_1}\rho_{k_1 j_1} \prod_{k:(k,j_1)\in\mathcal{A}} (1-\rho_{kj_1}) \ge 0$. Hence, we can obtain $U(\mathcal{A})$ is monotone.

iii) *Submodularity*. $\forall \mathcal{A} \subseteq \forall \mathcal{B} \subseteq \mathcal{V}$, $\forall v_1 = (k_1, j_1) \in \mathcal{V} \backslash \mathcal{B}$, according to Eq. (15), we can obtain

$$U(\mathcal{A} \cup \{v_1\}) - U(\mathcal{A}) = u_{j_1}\rho_{k_1 j_1} \prod_{k:(k,j_1)\in\mathcal{A}} (1-\rho_{kj_1}), \tag{18}$$

$$U(\mathcal{B} \cup \{v_1\}) - U(\mathcal{B}) = u_{j_1}\rho_{k_1 j_1} \prod_{k:(k,j_1)\in\mathcal{B}} (1-\rho_{kj_1}). \tag{19}$$

Since $\mathcal{A} \subseteq \mathcal{B}$, we have

$$\prod_{k:(k,j_1)\in\mathcal{B}} (1-\rho_{kj_1}) = \prod_{k:(k,j_1)\in\mathcal{A}} (1-\rho_{kj_1}) \times \prod_{k:(k,j_1)\in\mathcal{B}\backslash\mathcal{A}} (1-\rho_{kj_1}). \tag{20}$$

As $\forall k \in \mathcal{K}, 0 \le \rho_{kj_1} \le 1$, we have $\prod_{k:(k,j_1)\in\mathcal{B}\backslash\mathcal{A}} (1-\rho_{kj_1}) \le 1$. Hence, according to Eq. (20),

$$\prod_{k:(k,j_1)\in\mathcal{A}} (1-\rho_{kj_1}) \ge \prod_{k:(k,j_1)\in\mathcal{B}} (1-\rho_{kj_1}). \tag{21}$$

According to Eqs. (18), (19), and (21), we have

$$U(\mathcal{A} \cup \{v_1\}) - U(\mathcal{A}) \ge U(\mathcal{B} \cup \{v_1\}) - U(\mathcal{B}). \tag{22}$$

As a result, according to Def. 1, the objective function $U(\mathcal{A})$ is submodular. Thus, Lemma 1 is proved. □

Definition 2. (**Matroid** [32]): Consider a finite ground set $\mathcal{V}$ and a non-empty collection of subsets of $\mathcal{V}$ which is represented as $\mathcal{I}$. The pair $(\mathcal{V}, \mathcal{I})$ is called a *matroid*, if and only if the following three conditions hold: i) $\emptyset \in \mathcal{I}$; ii) If $\forall \mathcal{A} \subseteq \forall \mathcal{B} \in \mathcal{I}$, $\mathcal{A} \in \mathcal{I}$; iii) If $\forall \mathcal{A} \in \mathcal{I}, \forall \mathcal{B} \in \mathcal{I}$ and $|\mathcal{A}| < |\mathcal{B}|$, $\exists v_1 \in \mathcal{B}$ satisfies $\mathcal{A} \cup \{v_1\} \in \mathcal{I}$.

Definition 3. (**Knapsack constraint** [33]): Given a set of weights $\{w_k\}$ for the ground set $\mathcal{K} = \{1, ..., k, ..., K\}$ and a knapsack of capacity B, the associated constraint is called *knapsack constraint* if the sum of weights of elements in the solution $\mathcal{A} \subseteq \mathcal{K}$ satisfies $\sum_{k\in\mathcal{A}} w_k \le B$.

Lemma 2. Constraints (16) and (17) are a *matroid* constraint and a *knapsack* constraint, respectively.

Proof. We first prove that constraint (16) is a matroid constraint, according to Def. 2. Specifically, let $\mathcal{I}$ denotes the collection of all the solutions that satisfy constraint (16). In

the following, we prove that the pair $(\mathcal{V}, \mathcal{I})$ constructed by constraint (16) satisfies the three conditions of matroid.

i) $\emptyset \in \mathcal{V}$, and $\emptyset$ satisfies constraint (16). Hence, we have $\emptyset \in \mathcal{I}$, and the first condition is true.

ii) Since $\mathcal{B} \in \mathcal{I}$, according to Eq. (16), for $\forall k_1 : (k_1, j) \in \mathcal{B}$, we have

$$\sum_{j:(k_1,j)\in\mathcal{B}} \mathbb{1}_{(k_1,j)\in\mathcal{B}} \leq 1. \tag{23}$$

Moreover, since $\mathcal{A} \subseteq \mathcal{B}$, for $\forall k_2 : (k_2, j) \in \mathcal{A}$, we can get

$$\sum_{j:(k_2,j)\in\mathcal{A}} \mathbb{1}_{(k_2,j)\in\mathcal{A}} \leq 1. \tag{24}$$

Thus, based on Eq.(24), we have $\mathcal{A} \in \mathcal{I}$, and the second condition is true.

iii) Since $\mathcal{A} \in \mathcal{I}$, $\mathcal{B} \in \mathcal{I}$, and $|\mathcal{A}| < |\mathcal{B}|$, $\exists k_1 : (k_1, j_1) \in \mathcal{B}\backslash\mathcal{A}$ which satisfies the following inequation:

$$\sum_{j:(k_1,j)\in\mathcal{A}} \mathbb{1}_{(k_1,j)\in\mathcal{A}} < \sum_{j:(k_1,j)\in\mathcal{B}} \mathbb{1}_{(k_1,j)\in\mathcal{B}} = 1. \tag{25}$$

Thus, based on Eqs.(16)(25), we can get

$$\sum_{j:(k_1,j)\in\mathcal{A}} \mathbb{1}_{(k_1,j)\in\mathcal{A}} = 1. \tag{26}$$

Moreover, since $\mathcal{A} \in \mathcal{I}$, according to Eqs. (16)(25)(26), for $\forall k_2 : (k_2, j) \in \mathcal{A} \cup \{(k_1, j_1)\}$, the following inequation holds:

$$\sum_{k_2:(k_2,j)\in\mathcal{A}\cup\{(k_1,j_1)\}} \mathbb{1}_{(k_2,j)\in\mathcal{A}\cup\{(k_1,j_1)\}} \leq 1. \tag{27}$$

As a result, we can obtain $\mathcal{A} \cup \{(k_1, j_1)\} \in \mathcal{I}$, and the third condition is true. As constraint (16) satisfies all the three conditions, it is a matroid constraint.

Moreover, we prove that constraint (17) is a knapsack constraint. Consider that each element $(k, j) \in \mathcal{V}$ has the weight $c_{kj}\rho_{kj}$; the sum of weights of elements in the solution $\mathcal{A} \subseteq \mathcal{V}$ is constrained by the knapsack capacity B. Thus, Lemma 2 is proved. □

Algorithm design. According to Lemmas 1 and 2, the problem is proved to be maximizing a non-negative, monotone, and submodular function with a knapsack constraint and a matroid constraint. Thus, following the methodology of submodular optimization [34], as shown in Alg. 1, we propose *a greedy local search-based near-optimal task allocation algorithm*. Three key ideas behind this algorithm design are as follows:

(1) *Creating feasible solutions via the swapping operations*. Thanks to the exchange property [32] of the matroid constraint, we exploit the *swapping operation* to create feasible solutions which satisfy both the matroid constraint (16) and the knapsack constraint (17). In specific, let (v_+, v_-) and $\mathcal{V}_\mathrm{s}$ denotes a swap and the swap set, respectively, where v_+ represents a non-allocated pair, *i.e.,* $v_+ = (k_+, j_+) \in \mathcal{V}\backslash\mathcal{A}$; v_- denotes an allocated one, *i.e.,* $v_- = (k_-, j_-) \in \mathcal{A} \cup \{\emptyset\}$; $\mathcal{V}_\mathrm{s} = \{(v_+, v_-)\}$. It is worthy noting that $\emptyset$ represents a dummy element; the swapping operation in line 10 is equivalent to adding v_+^* to $\mathcal{A}$ directly, when v_-^* is $\emptyset$.

(2) *Greedy search based on the largest marginal profit-cost ratio*. Since this problem is maximizing a monotone and submodular objective function with a knapsack constraint, in line 7, we greedily search the swap (v_+^*, v_-^*) which achieves the largest marginal profit-cost ratio, *i.e.,* $(v_+^*, v_-^*) = \arg\max \pi(v_+, v_-)$, where

$$\pi(v_+, v_-) = \frac{U(\mathcal{A}\backslash\{v_-\} \cup \{v_+\}) - U(\mathcal{A})}{c_{k_+j_+}}. \tag{28}$$

If the new swap (v_+^*, v_-^*) can satisfy constraints (16)(17) and improve the platform profit by at least $\frac{\epsilon}{K^2J^2}$, it uses this swap to updates the solution.

(3) *Achieving guaranteed near-optimal solution based on two-level iteration*. The performance of the greedy search in the inner loop (*i.e.,* lines 6-11) fluctuates with the varied initialization s. Furthermore, it is still an open issue to obtain the optimal initialization [34]. Thus, we use the outer loop (*i.e.,* lines 3-13) to iteratively search the the best result within an initialization set, so as to achieve a near-optimal solution. Specifically, in line 2, we first create the initialization set $\mathcal{S}$, which includes all the sets of any two different elements in $\mathcal{V}$. Afterward, in lines 3-13, it iteratively searches each locally optimal solution based on each initialization by using the inner loop and selects the best one as the final result $\mathcal{A}^*$.

Algorithm 1: Greedy Local Search-based Near-optimal Task Allocation Algorithm.

1 **Input:** Task set $\mathcal{J}$; MOD driver set $\mathcal{K}$; Budget B; Sensing reward set $[c_{kj}]_{K\times J}$; Tasks' profit set $\{u_j\}$; Drivers' acceptance probability set $[\rho_{kj}]_{K\times J}$;
Output: Task allocation $[x_{kj}]_{K\times J}$; Platform profit U.
2 Initialize $\mathcal{S} \leftarrow \{(v_0, v_1)|\forall v_0 \in \mathcal{V}, \forall v_1 \in \mathcal{V}, v_0 \neq v_1\}$;
3 **for** s in $\mathcal{S}$ **do**
4 $\mathcal{A} \leftarrow s$;
5 $\mathcal{V}_s = \{(v_+, v_-)|\forall v_+ \in \mathcal{V}\backslash\mathcal{A}, \forall v_- \in \mathcal{A} \cup \{\emptyset\}\}$;
6 **while** $\mathcal{V}_s \neq \emptyset$ **do**
7 $(v_+^*, v_-^*) = \arg\max_{(v_+,v_-)\in\mathcal{V}_s} \frac{U(\mathcal{A}\backslash\{v_-\}\cup\{v_+\})-U(\mathcal{A})}{c_{k_+j_+}}$;
8 $\mathcal{V}_s \leftarrow \mathcal{V}_s\backslash\{(v_+^*, v_-^*)\}$;
9 **if** $\mathcal{A} \cup \{v_+^*\}\backslash\{v_-^*\}$ satisfies constraints (16)(17) and $U(\mathcal{A} \cup \{v_+^*\}\backslash\{v_-^*\}) \geq (1 + \frac{\epsilon}{K^2J^2})U(\mathcal{A})$ **then**
10 $\mathcal{A} \leftarrow \mathcal{A} \cup \{v_+^*\}\backslash\{v_-^*\}$;
11 $\mathcal{V}_s = \{(v_+, v_-)|\forall v_+ \in \mathcal{V}\backslash\mathcal{A}, \forall v_- \in \mathcal{A} \cup \emptyset\}$;
12 **if** $U(\mathcal{A}) > U(\mathcal{A}^*)$ **then**
13 $\mathcal{A}^* \leftarrow \mathcal{A}$;
14 Set $\mathbf{x} \leftarrow \{x_{kj} = 1|\forall k, \forall j, (k, j) \in \mathcal{A}^*\}$;
15 Compute $U(\mathbf{x})$ based on $\mathbf{x}$, $\{u_j\}$, and $[\rho_{kj}]_{K\times J}$, according to Eq. (15);
16 **return** $\mathbf{x}$ and $U(\mathbf{x})$.

Theorem 2. Alg. 1 achieves near-optimal solutions with a $(1 - e^{-2})/2$-approximation ratio in polynomial time $\mathcal{O}(K^6J^6\log(KJ))$, where K and J denote the numbers of drivers and tasks, respectively.

Proof. According to Lemmas 1 and 2, the optimal task allocation problem is maximizing a monotone, submodular objective function with a matroid constraint and a knapsack constraint. As a result, referring to [34], Alg. 1 which uses a greedy local-search strategy can achieve a $(1 - e^{-2})/2$-approximation ratio.

Since Alg. 1 has at most K^2J^2 initializations, there are no more than K^2J^2 outer loops, where K and J represent

This article has been accepted for publication in IEEE Transactions on Mobile Computing. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TMC.2023.3271671

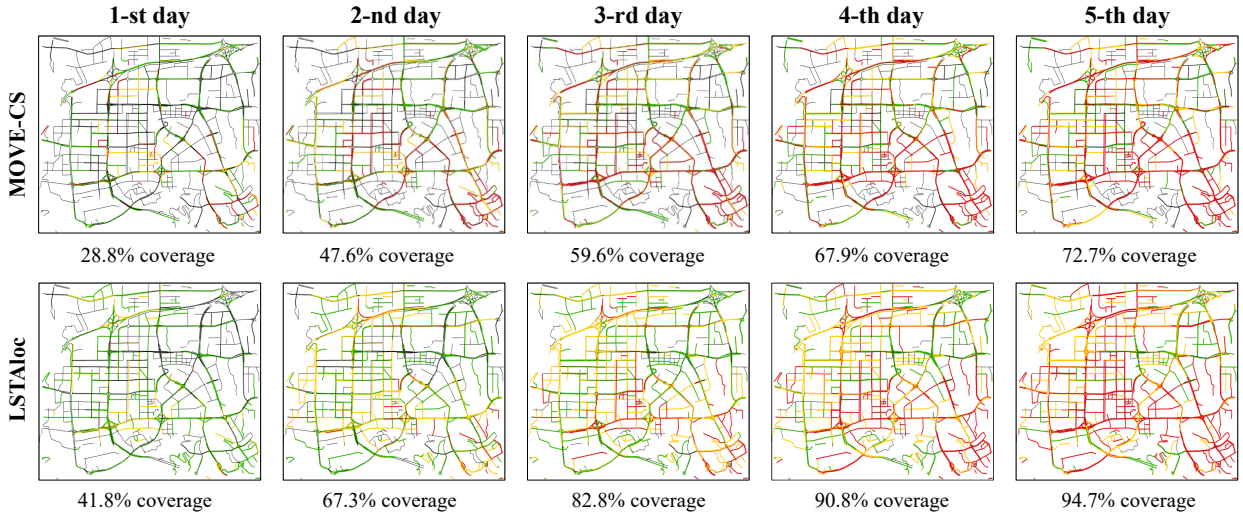
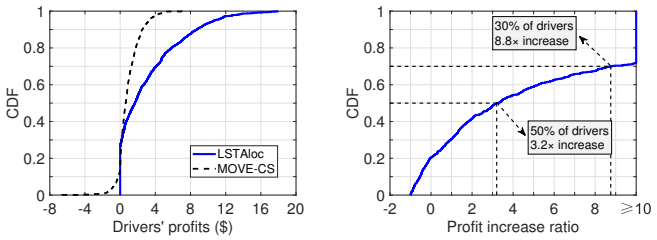


Fig. 7: Road coverage heatmaps in the *LSTAloc* and MOVE-CS models; black, green, yellow, and red lines represent collected times of 0, 1, 2, and 3, respectively.



(a) Drivers' profits **(b)** Profit increase ratio

Fig. 8: (a) Drivers' profits in the *LSTAloc* and MOVE-CS models, and (b) the profit increase ratio of *LSTAloc* compared with the original MOVE-CS model.

the numbers of drivers and tasks, respectively. In each outer loop, since Alg. 1 improves the profit by more than $\frac{\epsilon}{K^2J^2}$, it has at most $\frac{K^2J^2}{\epsilon}\log(KJ)$ swap operations. Furthermore, the time complexity of each swap operation is $\mathcal{O}(K^2J^2)$, since $\mathcal{V}$ has at most KJ elements. As a result, the time complexity of Alg. 1 is $\mathcal{O}(K^6J^6\log(KJ))$. To sum up, Theorem 2 is proved. □

5 Evaluation

We use the large-scale MOD vehicle dataset to emulate the operation process of the original MOVE-CS model and the *LSTAloc* model. Furthermore, we comprehensively compare the performance of our algorithm with five baseline algorithms, followed by evaluating the performance of the acceptance probability estimation component.

5.1 Emulation Methodology and Settings

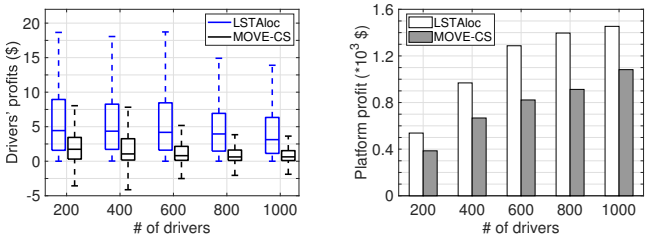
The emulation of the proposed *LSTAloc* model and the original MOVE-CS model is based on the large-scale MOD vehicle dataset (specified in Sec. 2.2) as follows. First, the MOVE-CS platform requires road data of 878 road segments with a total length of 191.1 miles in a 32 km^2 area. Then, for higher accuracy, each road is required to be sensed k times with decreasing profit u to the platform ($k = 3$, $u = \$2.5, 1.5$, and 0.5 per mile for the three times respectively). There are 1000 MOD drivers, randomly selected as participants willing to collect the road data for the MOVE-CS market. Next, we run the emulation for five days, which may end in advance if the budget is exhausted.

For the MOVE-CS model, drivers collect road data on the move at any time they want during their work hours. Drivers averagely spend \$0.06 per mile on fuels [35]; their data collection costs are only induced in the unoccupied state during extra trips for tasks. After the data are uploaded, each driver gets a reward. The original settings Payver adopted (\$0.01-0.05 per mile) are so unreasonable that most participants can only get few or even negative profits. To make a fair comparison, in the emulation, in contrast, we let a portion (1/a) of the platform profit u be the reward, *e.g.*, k=3, $1/a$=0.2, and the rewards are $\$0.5, 0.3, 0.1$ per mile respectively, according to the economic theory [36]. For the *LSTAloc* model, there are N rounds of task allocations. In each round, as explicated in Sec. 3.1, the platform publishes sensing tasks; each corresponds to one collection of a road segment. The platform then predicts drivers' pick-up profits and their acceptance probabilities, based on which it allocates sensing tasks to these drivers. Each driver accepts and accomplishes the tasks. Once the task is accomplished, she/he receives the reward given by the proposed algorithm. We implement the emulation on a commodity server with 3.00GHz dual-core Intel Core Xeon Gold 6561 CPU and 192GB RAM.

5.2 Results of Model Evaluation

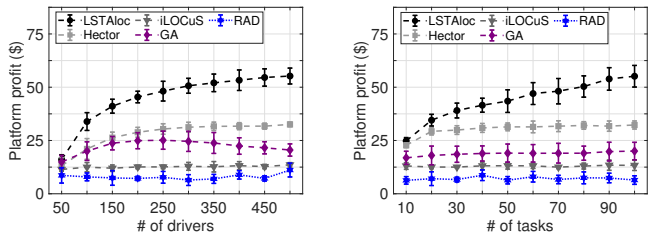
Drivers' profits. We first evaluate the two models on drivers' profits. As illustrated in Fig. 8a, for the MOVE-CS model, 14.5% of drivers have negative profits from sensing tasks, because they might spend a lot on collecting repeated road data, resulting in rewards far less than the driving cost. In contrast, all the drivers in the *LSTAloc* model make positive profits, thanks to the sensing reward design based on the spatial-temporal differentiation of pick-up profits. Moreover, we analyze all the task allocation

This article has been accepted for publication in IEEE Transactions on Mobile Computing. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TMC.2023.3271671



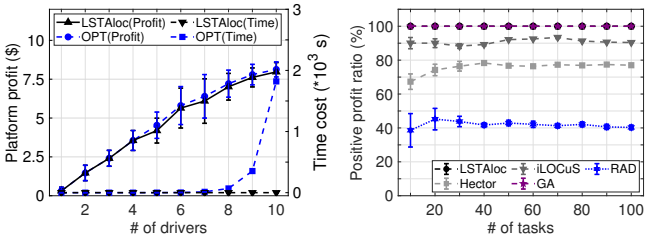
(a) Drivers' profits **(b)** Platform profit

Fig. 9: Impacts of different numbers of drivers on the performance of two models, in terms of drivers' profits and platform profit.



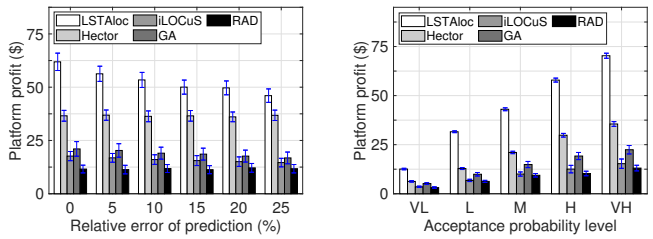
(a) Drivers' number **(b)** Tasks' number

Fig. 10: Comparison of the *LSTAloc* algorithm and four baselines in terms of the platform profit in different numbers of drivers and tasks.



(a) Near-optimality **(b)** Drivers' profits

Fig. 11: Comparison of the *LSTAloc* algorithm and four baselines in terms of the near-optimality and the drivers' profits.



(a) Error of pick-up profit heatmap **(b)** Drivers' acceptance probability

Fig. 12: Influences of different heatmap errors and acceptance probabilities on the algorithm's performance.

results in *LSTAloc*. Results show that, 87.3% of the allocated tasks enable drivers from low-yield to high-yield zones, consistent with their desires for immediate gains.

Furthermore, we compare the drivers' profits in the two models by calculating the drivers' profit increase ratios in *LSTAloc* to those in MOVE-CS. As demonstrated in Fig. 8b. we find in our model, 50% of drivers increase profits by 320%, and 30% have an increase ratio of 880%, compared with MOVE-CS. Further analysis indicates its effectiveness is anchored in the active task allocation scheme, *i.e.*, incentivizing drivers to complete the tasks suitable for them. Besides, Fig. 8b shows that 20% of drivers suffer decreased profits (than in the MOVE-CS model), due to no tasks allocated to them, which can be solved by prior allocation to these drivers in the next round.

Platform's profit. We evaluate the platform profit in the two models. We first visualize the coverage heatmap of collected road segments throughout the five days. As shown in Fig. 7, our coverage ratio of collected roads in each day is consistently higher than that in MOVE-CS. Our coverage ratio increases day by day, and ends up 94.7% in the last day, 22.0% higher than that in MOVE-CS. Meanwhile, our platform profit increases by 34.3%, also attributed to the active task allocation scheme, *i.e.*, encouraging drivers to unpopular roads, increasing the road coverage ratio as well as the platform profit.

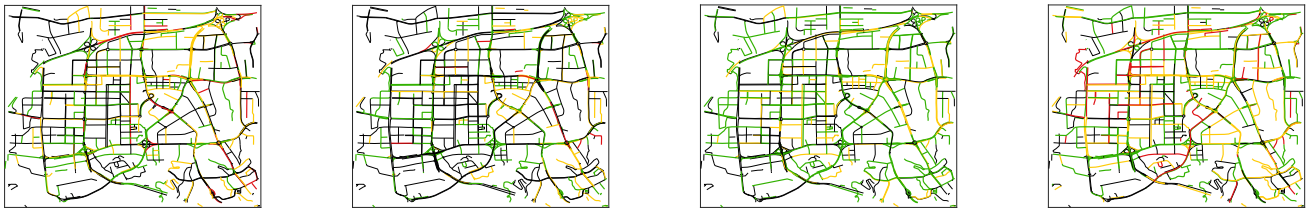
Impacts of parameters. We evaluate the impacts of the number of drivers on the model performance, in the aspects of drivers' profits and the platform profit. As shown in Fig. 9a, we illustrate the box-plot of the drivers' profits in the two models. The results show that *LSTAloc* can guarantee all the drivers' positive profits, while 14.6% of drivers have negative profits in MOVE-CS model. Moreover, the drivers' profits of the two models decrease with the number of drivers, since more opt-in drivers lead to more fierce competition for earnings. However, the decrease ratio in MOVE-CS is averagely 32.2% higher than that in *LSTAloc*. Moreover, Fig. 9b demonstrates that the platform profits of the two models improve with the number of drivers. The platform profit of *LSTAloc* outperforms that of MOVE-CS by 45.8% on average. Other parameters (*e.g.*, budget) show similar effects on results, so we do not show them due to the page limit.

5.3 Results of Algorithm Evaluation

Baseline algorithms. To comprehensively evaluate the performance of the key algorithm of *LSTAloc*, we exploit five baselines as follows: (1) *Hector* [27] greedily allocates sensing tasks with maximal marginal profit-cost efficiency to the drivers, while using their basic driving costs as the rewards. (2) *GA* [37] exploits the Genetic algorithm to maximize the platform profit with the assumption that the sensing rewards are already given. (3) *iLOCuS* [26] allocates the sensing tasks greedily to minimize the task distribution divergence, while utilizing the high pick-up probability of the task's zone as the hidden incentives. (4) *RAD* randomly allocates tasks with a uniform distribution of pick-up profits. (5) *OPT* uses the brutal-force search method to achieve the optimal solution with exponential time cost. In the remaining, we call the proposed algorithm *LSTAloc* as well for simplification.

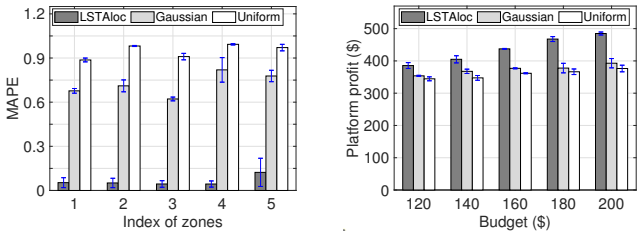
Platform profit. We first evaluate the platform profit of the *LSTAloc* algorithm in different numbers of drivers

This article has been accepted for publication in IEEE Transactions on Mobile Computing. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TMC.2023.3271671



(a) w/o ape (probability=25%) **(b)** w/o ape (probability=75%) **(c)** w/o ape (probability=100%) **(d)** *LSTAloc*

Fig. 13: Ablation studies by comparing *LSTAloc* with its variant without the acceptance probability estimation (*i.e.,* w/o ape), regarding the coverage heatmap of collected road segments. Note that, we set the acceptance probability of *LSTAloc* w/o ape 25%, 75%, and 100% in (a), (b), and (c), respectively.



(a) Estimation error **(b)** Platform profit

Fig. 14: Impacts of different mobility models on the algorithm's performance, including the estimation error (*i.e.,* MAPE) and the platform profit.

and tasks, compared to four baselines. As demonstrated in Figs. 10a and 10b, the platform profit of *LSTAloc* exceeds those of RAD, GA, Hector, and iLOCus by 466.8%, 103.2%, 61.7%, and 257.1% in different numbers of drivers, respectively, and by 516.5%, 132.7%, 44.4%, and 237.8% in different numbers of tasks, respectively. Fig. 10a illustrates that the platform profit of *LSTAloc* increases with the number of drivers. However, its growth rate tends to decrease with the number of drivers. It is because more participating drivers can provide the platform with more options for task allocation, thus enhancing the platform's profit. However, owing to the limited budget, the growth rate of platform profit decreases with the number of drivers. In addition, Figs. 10a and 10b show that the performances of Hector, GA, and iLOCus are close to that of *LSTAloc* when the numbers of drivers and tasks are very small. The reasons are that the number of the candidate solutions for task allocation is very limited owing to few drivers and tasks, hence leading to similar performances for these methods.

Drivers' profits and near-optimality. We evaluate the near-optimality and drivers' profits of *LSTAloc* by comparing it with Hector, iLOCus, RAD, and OPT. As illustrated in Fig. 11a, we compare the platform profit and the time cost of *LSTAloc* with those of OPT in a small-scale scenario (*i.e.,* $M = 10, S = 6$). The results illustrate that *LSTAloc* can averagely achieve 97.2% of the optimal platform profit with only 0.004% of OPT's time cost in different numbers of drivers. As for drivers' profits, there exists a large gap between *LSTAloc* and baselines. As shown in Fig. 11b, *LSTAloc* guarantees a 100% positive profit ratio (*i.e.,* the percentage of MOD drivers with positive profits after opting in MOVE-CS), outperforming Hector, iLOCus, and RAD by 24.2%, 9.2%, and 58.1%. Hence, differing from the three baselines, *LSTAloc* ensures a positive profit for every driver, thanks to consideration of long-term and short-term profits of drivers. In particular, since GA adopts the same reward design scheme as *LSTAloc*, it also achieves a positive profit ratio of 100% like *LSTAloc*.

Algorithm's robustness. To investigate the robustness of the *LSTAloc* algorithm, we evaluate the influences of different heatmaps' errors and acceptance probabilities on the algorithm's performance. First, we evaluate the impacts of prediction error of heatmaps on the performance of the *LSTAloc* algorithm. Particularly, we add the White Gaussian noise into the pick-up profit heatmaps, where the noise deviation is set according to the relative prediction error ranging from 0% to 25%. As demonstrated in Fig. 12a, the platform profits of *LSTAloc* and iLOCuS decrease with the prediction error, since both of them are based on the predicted heatmaps. Moreover, the platform profit of *LSTAloc* is always much higher than those of other baselines with different prediction errors. In the case of 25% prediction error, *LSTAloc* even outperforms RAD, GA, Hector, and iLOCus by 307.9%, 173.9%, 27.6%, and 214.9%, respectively. Moreover, we let the acceptance probability of drivers follow the stochastic uniform distribution $\mathcal{U}$ and vary it in five levels, *i.e.,* $\mathcal{U}(0, 0.2)$, $\mathcal{U}(0.2, 0.4)$, $\mathcal{U}(0.4, 0.6)$, $\mathcal{U}(0.6, 0.8)$, and $\mathcal{U}(0.8, 1)$, called Very Low (VL), Low (L), Moderate (M), High (H), and Very High (VH), respectively. As demonstrated in Fig. 12b, the higher the acceptance probability, the more the platform profits for all methods. *LSTAloc* outperforms RAD, GA, Hector, and iLOCus in the platform profit averagely by 410.9%, 200.8%, 104.2%, and 346.9%, respectively.

5.4 Acceptance Probability Estimation Component Evaluation

To evaluate the performance of the acceptance probability estimation component, we emulate the *LSTAloc* model based on the large-scale MOD vehicle dataset (specified in Sec. 5.1). Acquiring ground-truth information about whether these drivers accept the allocated tasks is challenging, so we leverage the real-world MOD vehicle dataset to construct an approximate alternative. Specifically, drivers often have schedules for picking up passengers and would

not usually change them [17]. Moreover, when a MOD driver passes through a road segment, his/her vehicle can automatically collect road data without nearly extra cost. Therefore, when an allocated task is located in a zone the driver will pass through, this driver will be more likely to accept it. To this end, we assume that if the driver travels through the road segment of the allocated task at least once during the period, she/he will accept the task. Otherwise, they will not accept it [20]. Since the key to the acceptance probability estimation is the movement probability estimation based on the mobility model, we set the content preference a constant (*i.e.,* 100%) so as to evaluate the impact of this key model particularly. In the following, we first evaluate the component's performance by comparing it with two mobility models, then conduct ablation studies.

Impacts of different mobility models. We evaluate the influence of the mobility model on the *LSTAloc* algorithm's performance by comparing it with Gaussian and Uniform models. As shown in Fig. 14a, we select five zones randomly and compare the Mean Absolute Percentage Error (MAPE) of the movement probability estimation for these zones by using different mobility models. The results indicate that *LSTAloc*—which uses the Poisson model—achieves the best accuracy; its accuracy averagely outperforms the Gaussian and Uniform models by 65.8% and 88.5%, respectively. Furthermore, Fig. 14b illustrates that *LSTAloc* achieves much more platform profits than the Gaussian and Uniform models in different budgets. The platform profit in the *LSTAloc* averagely outperforms those in the Gaussian and Uniform models by 16.7% and 21.4%, respectively.

Ablation studies. We conduct an ablation study to evaluate the effectiveness of the acceptance probability estimation component. In specific, we utilize a baseline without the acceptance probability estimation component called *LSTAloc* w/o ape, which assumes that the acceptance probability of each driver is unchanged and previously known. As demonstrated in Figs. 13a, 13b, and 13c, we visualize the coverage heatmap of collected road segments with the limited budget for *LSTAloc* w/o ape when setting the acceptance probability 25%, 75%, and 100%, respectively. The results indicate that the lower acceptance probability settings may reduce the coverage of collected road segment, since most tasks are allocated repeatedly to multiple drivers, hence decreasing the coverage ratio. Moreover, Fig. 13d shows the coverage ratio in *LSTAloc* is up to 70.4%, which is 17.9% higher than those in *LSTAloc* w/o ape. The reason is that, without the acceptance probability estimation component, most allocated tasks in *LSTAloc* w/o ape may not meet the drivers' willingness and will not be accepted by these drivers, thus resulting in the dramatic decrease of the performance.

6 Discussions & Future Work

Multi-task allocation for each driver. In *LSTAloc* model, we restricts that each driver is allocated at most one task in each allocation period, so as to alleviate the negative influence of executing sensing tasks on the chief job of MOD drivers, *i.e.,* picking up passengers. If we intend to offer drivers more freedom of task selection, we can extend our work to the multi-task allocation model that each driver is allocated more than one task. In this new model, besides task allocation optimization, the driving trajectories should be wisely planned so that each driver can pass by all of their allocated tasks with minimal driving distance/cost, *i.e.,* a classical Travelling salesman problem (TSP) [38]. More challengingly, these two problems are tightly coupled with each other. For example, different task allocation results may affect their driving trajectory plans. Thus, in the future, we will address the task allocation problem and TSP jointly.

Concerning spatio-temporal coverage of tasks. Accounting for the spatio-temporal coverage of executed tasks is significant for many practical applications, such as city-wide air quality monitoring [26], [37]. Our work aims at maximizing the total platform profit (*i.e.,* the total utility of executed tasks), which may weaken the spatio-temporal coverage of tasks. To improve their spatio-temporal coverage, we can extend our work by improving the utilities for the tasks which are always not allocated by the platform (or not accepted by drivers). Furthermore, we can add a coverage constraint that the expected execution probability of each task (*i.e.,* $1-\prod_{k=1}^{K}(1-\rho_{kj})^{x_{kj}}$) is no less than a predefined threshold. Since this constraint is non-linear, it makes the problem more challenging. In the future, we will study how to solve the problem of maximizing a submodular function with a matroid constraint, a knapsack constraint, and a non-linear constraint.

Privacy protection for both drivers and passengers. In MOD vehicular crowdsensing, a large amount of data about drivers and passengers is shared to analyze the drivers' behaviors, potentially incurring their privacy leakage [39], [40]. On the one hand, *LSTAloc* makes a slight influence on the passengers' privacy with the following two reasons. First, we mainly use the MOD vehicles to execute the sensing tasks during their unoccupied time without passengers. Moreover, only the trajectories and the occupied/vacant state indicators of MOD vehicles are utilized to construct the pick-up profit heatmap and estimate the acceptance probability. Therefore, little private information about each passenger is shared, and our work causes subtle risks on the privacy of passengers. On the other hand, *LSTAloc* may affect the drivers' privacy, as it uses their driving trajectories with the anonymized vehicle ID. Two feasible ways can be used to protect the drivers' privacy. First, since the MOD platform (*e.g.,* DiDi, Uber, *etc.*) is trusted for drivers, we can use this platform to allocate MOVE-CS tasks for each driver, thereby protecting their privacy from being stolen by untrusted third-party platforms. Second, many good privacy protection algorithms can be utilized, such as the differential privacy [41] and personalized privacy protection [42]. In the future, we will study the trade-off between the driver's privacy protection and platform profit.

Advices for MOD vehicular crowdsensing: Based on the findings presented in this paper, there are some suggestions for promoting the market of MOVE-CS. First, leveraging MOD vehicles is a promising way to collect sensing data cheaply. However, the incentives for MOD drivers should be wisely considered to encourage enough drivers. Second, surveying drivers' perceptions and analyzing their behavior is critical to designing an efficient incentive mechanism. Finally, accounting for drivers' long-short-term profits is

TABLE 2: Comparisons between existing works and *LSTAloc* regarding the system model and algorithm performance.

	Incentivizing non-MOD vehicles					**Incentivizing MOD vehicles**					
	[37]	[43]	[44]	[45]	[46]	[47]	[48]	[49]	[50]	[26]	**Ours**
Picking up passengers	×	×	×	×	×	✓	✓	✓	✓	✓	✓
Executing sensing tasks	✓	✓	✓	✓	✓	×	×	×	✓	✓	✓
Drivers' long-short term profits	×	×	×	×	×	×	×	×	×	×	✓
Guaranteed performance	✓	✓	×	✓	✓	×	✓	×	×	×	✓

greatly important to incentivize their participation when allocating sensing tasks.

7 Related Work

Recently, there have been considerable studies of incentivized mobile crowdsensing [46], [51]–[55], most of which focus on human mobility [52]–[54], [56] without considering the special impacts of vehicle mobility. Compared with human mobility, vehicle mobility is often constrained by road networks, but exhibits more dynamics, thus leading to new opportunities and challenges in mobile crowdsensing [46], [55], [57]. Since this paper belongs to the category of incentivizing vehicular crowdsensing [45], we review its related studies in terms of non-MOD vehicles (such as private cars) and MOD vehicles.

Incentivizing non-MOD vehicular crowdsensing. As one of the earliest works about incentivizing vehicular crowdsensing, He *et al.* [37] design a participant recruitment strategy, jointly leveraging both the current location and the predictable mobility pattern of vehicles. Since reference [37] does not consider the uncertainty of vehicle mobility, Wang *et al.* [43] study both the deterministic and probabilistic trajectory models and propose two efficient vehicle recruitment algorithms. Zhu *et al.* [44] use RNN to predict the future vehicle mobility, which is used to select vehicles to maximize their coverage with limited budget. Moreover, regarding the selfish drivers who may strategically misreport their costs, Xiao *et al.* [45] present a reverse-auction-based truthful incentive mechanism design for non-deterministic vehicular crowdsensing. Fan *et al.* [27] propose Hector, a novel joint scheduling and incentive mechanism of vehicular crowdsensing, which is proved truthful, individual-rational, computation-efficient, and close-to-optimal. Further, accounting for the uncertain task delay, Chen *et al.* [46] propose a truthful budget utility maximization auction scheme. To sum up, *the above works focus on common vehicles without concerning the special tasks of MOD vehicles*, *i.e., picking up passengers*.

Incentivizing MOD vehicular crowdsensing. The incentive mechanism design for MOD vehicular crowdsensing differs significantly from that for non-MOD vehicular crowdsensing, since the MOD drivers are also concerned with passenger missions which typically dominate their earnings. For example, He *et al.* [47] propose a new adaptive pricing scheme called CAPRICE, which uses the proactive pricing signals to incentivize MOD drivers to balance the supply-demand in picking up passengers. Xu *et al.* [48] design an effective order dispatching algorithm which considers both the immediate passenger satisfaction and the expected future income of drivers. Ke *et al.* [49] propose a passenger-driver matching scheme empowered by convex combinatorial optimization and deep reinfocement learning to help reduce the average pick-up time of drivers. Nevertheless, both of them only account for the pick-up profit of MOD drivers and neglect their extra incomes of executing crowdsensing tasks. In contrast, a recent work called iLOCuS [26], highly related to this paper, proposes a hybrid incentive mechanism, which combines the monetary rewards from executing sensing tasks and the non-monetary hidden incentives (*i.e.,* the passenger's requests at the task's zone) from picking up passengers. Xu *et al.* [50] also combined both monetary and non-monetary rewards as incentives to motivate drivers to achieve high-quality sensing coverage in a target sensing area. Nevertheless, both of them neglect the in-depth demands of MOD drivers for short-term and long-term profits, inefficient in encouraging the drivers. Xiang *et al.* [58] propose a sensing task allocation scheme based on the deep reinforcement learning, achieving a near optimal solution with a factor which depends on the maximal and minimal costs of all the sensing tasks. Contrarily, this work uses the greedy local search to achieve a $(1 - e^{-2})/2$-approximation ratio, thereby having more robustness in the real applications with different settings. Additionally, in the conference version of this paper [1], we present an operation model to resurrect the market of MOVE-CS, based on the assumption that the acceptance probability of each driver is unchanged and previously known. However, in real applications, each driver's acceptance probability varies with the tasks and the context. To this end, this paper develops an effective acceptance probability estimation model to estimate the acceptance probability of MOD drivers by leveraging their predictable mobility.

Summary. As illustrated in Table 2, on the one hand, existing works focus on incentivizing non-MOD vehicles and neglect the special pick-up jobs of MOD drivers, thus discouraging the MOD drivers from participating in crowdsensing. On the other hand, parts of works take into account both the pick-up profits and the sensing task's incomes in the incentive mechanism design. Nevertheless, they fail to consider MOD drivers' explicit and implicit needs. *Distinguished from them, based on user studies and dataset-based in-depth analysis, we uncover that MOD drivers have an explicit preference for short-term, immediate gains and implicit rationality in pursuit of long-term, stable profits. Moreover, motivated by the findings, we propose a novel driver-oriented incentive mechanism that satisfies the MOD drivers' explicit and implicit needs and considers the platform's profit*.

8 Conclusion

In this paper, motivated by findings in drivers' survey and the dataset-based MOD driver behavior analysis, we

propose *LSTAloc*, a new driver-oriented incentive mechanism for MOVE-CS to satisfy drivers' explicit and implicit needs and consider the platform's profit. Behind it lies a spatial-temporal differentiation-aware task allocation scheme empowered by submodular optimization. It involves pick-up heatmap prediction based on RNN, the differentiation-aware sensing reward design, the acceptance probability estimation based on drivers' mobility model, and the submodularity-based task allocation algorithm. The emulation reveals that *LSTAloc* guarantees not only positive profits for drivers, but also a near-optimal profit for the platform, hence incentivizing MOD drivers effectively to resurrect MOVE-CS.

REFERENCES

[1] C. Xiang, Y. Li, Y. Zhou, S. He, Y. Qu, Z. Li, L. Gong, and C. Chen, "A comparative approach to resurrecting the market of mod vehicular crowdsensing," in *Proc. IEEE INFOCOM*, 2022, pp. 1479–1488.

[2] X. Tang, Z. Qin, F. Zhang, Z. Wang, Z. Xu, Y. Ma, H. Zhu, and J. Ye, "A deep value-network based approach for multi-driver order dispatching," in *Proc. ACM KDD*, Anchorage, AK, USA, 2019, pp. 1780–1790.

[3] "How many uber drivers are there?" 2021. [Online]. Available: https://therideshareguy.com/how-many-uber-drivers-are-there/.

[4] "Global mobility on demand market forecast & opportunities by 2022," 2017. [Online]. Available: https://www.techsciresearch.com/report/global-mobility-on-demand-market/1254.html.

[5] "Drivers for uber, lyft are earning less than half of what they did four years ago, study finds," 2019. [Online]. Available: https://www.marketwatch.com/story/drivers-for-uber-lyft-are-earning-less-than-half-of-what-they-/did-four-years-ago-study-finds-2018-09-24.

[6] "Uber and lyft have a driver shortage problem, and it's costing them a lot of money," 2021. [Online]. Available: https://www.theverge.com/2021/4/7/22371850/uber-lyft-driver-shortage-covid-bonus-stimulus.

[7] "Payver," 2019. [Online]. Available: https://www.lazymoneyguy.com/payver/.

[8] "Google maps," 2021. [Online]. Available: https://www.google.com/maps.

[9] "Lvl5," 2020. [Online]. Available: https://www.crunchbase.com/organization/lvl5.

[10] "Thanks to a dashcam, crafty uber drivers are boosting their pay," 2017. [Online]. Available: http://money.cnn.com/2017/07/19/technology/business/rideshare-drivers-camera/index.html.

[11] "Doordash acquires autonomous driving startup scotty labs," 2019. [Online]. Available: https://techcrunch.com/2019/08/20/doordash-acquires-autonomous-driving-startup-scotty-labs/.

[12] "Amazon mechanical turk," 2021. [Online]. Available: https://www.mturk.com/.

[13] "Gigwalk," 2021. [Online]. Available: https://www.gigwalk.com/.

[14] "User study form," 2021. [Online]. Available: https://forms.gle/f6rSLextsfaXy8JS9.

[15] A. M. Lund, "Measuring usability with the USE questionnaire," *Usability interface*, vol. 8, no. 2, pp. 3–6, 2001.

[16] "How much does a ride with the uber app cost?" 2020. [Online]. Available: https://www.uber.com/global/en/price-estimate/.

[17] Z. Wang, J. Zhao, J. Hu, T. Zhu, Q. Wang, J. Ren, and C. Li, "Towards personalized task-oriented worker recruitment in mobile crowdsensing," *IEEE Transactions on Mobile Computing*, vol. 20, no. 5, pp. 2080–2093, 2020.

[18] F. Wu, S. Yang, Z. Zheng, S. Tang, and G. Chen, "Fine-grained user profiling for personalized task matching in mobile crowdsensing," *IEEE Transactions on Mobile Computing*, vol. 20, no. 10, pp. 2961–2976, 2020.

[19] X. Zhang, G. Xue, R. Yu, D. Yang, and J. Tang, "Truthful incentive mechanisms for crowdsourcing," in *Proc. IEEE INFOCOM*, HongKong, China, 2015, pp. 2830–2838.

[20] D. Zhang, H. Xiong, L. Wang, and G. Chen, "Crowdrecruiter: selecting participants for piggyback crowdsensing under probabilistic coverage constraint," in *Proc. ACM Ubicomp*, Seattle, WA, USA, 2014, pp. 703–714.

[21] Z. Wang, J. Hu, R. Lv, J. Wei, Q. Wang, D. Yang, and H. Qi, "Personalized privacy-preserving task allocation for mobile crowdsensing," *IEEE Transactions on Mobile Computing*, vol. 18, no. 6, pp. 1330–1341, 2019.

[22] R. A. Murphey, "Target-based weapon target assignment problems," in *Nonlinear assignment problems*. Springer, 2000, pp. 39–53.

[23] X. Fan, C. Xiang, C. Chen, P. Yang, L. Gong, X. Song, P. Nanda, and X. He, "Buildsensys: reusing building sensing data for traffic prediction with cross-domain learning," *IEEE Transactions on Mobile Computing*, vol. 20, no. 6, pp. 2154–2171, 2021.

[24] J. Zhang, S. Liang, Z. Deng, and J. Shao, "Spatial-temporal attention network for temporal knowledge graph completion," in *Proc. DASFAA*, 2021, pp. 207–223.

[25] Y. Duan, L. Yisheng, and F.-Y. Wang, "Travel time prediction with lstm neural network," in *Proc. IEEE ITSC*, Rio de Janeiro, Brazil, 2016, pp. 1053–1058.

[26] S. Xu, X. Chen, X. Pi, C. Joe-Wong, P. Zhang, and H. Y. Noh, "iLOCuS: incentivizing vehicle mobility to optimize sensing distribution in crowd sensing," *IEEE Transactions on Mobile Computing*, vol. 19, no. 8, pp. 1831–1847, 2020.

[27] G. Fan, H. Jin, Q. Liu, W. Qin, X. Gan, H. Long, L. Fu, and X. Wang, "Joint scheduling and incentive mechanism for spatio-temporal vehicular crowd sensing," *IEEE Transactions on Mobile Computing*, vol. 20, no. 4, pp. 1449–1464, 2021.

[28] E. Mitsopoulou, I. Litou, and V. Kalogeraki, "Multi-objective online task allocation in spatial crowdsourcing systems," in *Proc. IEEE ICDCS*, 2020, pp. 1123–1133.

[29] M. Xing, J. He, and L. Cai, "Utility maximization for multimedia data dissemination in large-scale vanets," *IEEE Transactions on Mobile Computing*, vol. 16, no. 4, pp. 1188–1198, 2016.

[30] M. L. McHugh, "The chi-square test of independence," *Biochemia medica*, vol. 23, no. 2, pp. 143–149, 2013.

[31] G. L. Nemhauser, L. A. Wolsey, and M. L. Fisher, "An analysis of approximations for maximizing submodular set functions-I," *Mathematical Programming*, vol. 14, no. 1, pp. 265–294, 1978.

[32] J. G. Oxley, *Matroid theory*. Oxford University Press, 1992.

[33] J. Lee, V. S. Mirrokni, V. Nagarajan, and M. Sviridenko, "Non-monotone submodular maximization under matroid and knapsack constraints," in *Proc. ACM STOC*, 2009, pp. 323–332.

[34] K. Sarpatwar, B. Schieber, and H. Shachnai, "Constrained submodular maximization via greedy local search," *Operations Research Letters*, vol. 47, no. 1, pp. 1–6, 2019.

[35] "Driving cost of mod vehicles," 2019. [Online]. Available: https://www.greencarreports.com/news/1123364_nyc-taxi-mpg-requirements-also-cut-pollution-study-confirms.

[36] G. S. Becker, M. Grossman, and R. T. Michael, *Economic theory*. Routledge, 2017.

[37] Z. He, W. Wu, and D. Qi, "High quality participant recruitment in vehicle-based crowdsourcing using predictable mobility," in *Proc. IEEE INFOCOM*, Hong Kong, China, 2015, pp. 2542–2550.

[38] T. Wu, P. Yang, H. Dai, C. Xiang, and W. Xu, "Optimal charging oriented sensor placement and flexible scheduling in rechargeable wsns," *ACM Transactions on Sensor Networks*, vol. 18, no. 3, pp. 1–27, 2022.

[39] X. Zheng and Z. Cai, "Privacy-preserved data sharing towards multiple parties in industrial iots," *IEEE Journal on Selected Areas in Communications*, vol. 38, no. 5, pp. 968–979, 2020.

[40] Z. Cai, X. Zheng, J. Wang, and Z. He, "Private data trading towards range counting queries in internet of things," *IEEE Transactions on Mobile Computing*, pp. 1–17, 2022, (early access).

[41] H. Zhao, M. Xiao, J. Wu, Y. Xu, H. Huang, and S. Zhang, "Differentially private unknown worker recruitment for mobile crowdsensing using multi-armed bandits," *IEEE Transactions on Mobile Computing*, vol. 20, no. 9, pp. 2779–2794, 2021.

[42] P. Sun, Z. Wang, L. Wu, Y. Feng, X. Pang, H. Qi, and Z. Wang, "Towards personalized privacy-preserving incentive for truth discovery in mobile crowdsensing systems," *IEEE Transactions on Mobile Computing*, vol. 21, no. 1, pp. 352–365, 2022.

[43] X. Wang, W. Wu, and D. Qi, "Mobility-aware participant recruitment for vehicle-based mobile crowdsensing," *IEEE Transactions on Vehicular Technology*, vol. 67, no. 5, pp. 4415–4426, 2018.

[44] X. Zhu, Y. Luo, A. Liu, W. Tang, and M. Z. A. Bhuiyan, "A deep learning-based mobile crowdsensing scheme by predicting vehicle

mobility," *IEEE Transactions on Intelligent Transportation Systems*, vol. 22, no. 7, pp. 4648–4659, 2021.
[45] M. Xiao, G. Gao, J. Wu, S. Zhang, and L. Huang, "Privacy-preserving user recruitment protocol for mobile crowdsensing," *IEEE/ACM Transactions on Networking*, vol. 28, no. 2, pp. 519–532, 2020.
[46] X. Chen, L. Zhang, Y. Pang, B. Lin, and Y. Fang, "Timeliness-aware incentive mechanism for vehicular crowdsourcing in smart cities," *IEEE Transactions on Mobile Computing*, vol. 21, no. 9, pp. 3373–3387, 2022.
[47] S. He and K. G. Shin, "Spatio-temporal adaptive pricing for balancing mobility-on-demand networks," *ACM Transactions on Intelligent Systems and Technology*, vol. 10, no. 4, pp. 1–28, 2019.
[48] Z. Xu, Z. Li, Q. Guan, D. Zhang, Q. Li, J. Nan, C. Liu, W. Bian, and J. Ye, "Large-scale order dispatch in on-demand ride-hailing platforms: a learning and planning approach," in *Proc. ACM KDD*, London, UK, 2018, pp. 905–913.
[49] J. Ke, F. Xiao, H. Yang, and J. Ye, "Learning to delay in ride-sourcing systems: A multi-agent deep reinforcement learning framework," *IEEE Transactions on Knowledge and Data Engineering*, vol. 34, no. 5, pp. 2280–2292, 2022.
[50] S. Xu, X. Chen, X. Pi, C. Joe-Wong, P. Zhang, and H. Y. Noh, "Vehicle dispatching for sensing coverage optimization in mobile crowdsensing systems," in *Proc. ACM IPSN*. New York, NY, USA: ACM, 2019, pp. 311–312.
[51] Z. Cai, Z. Duan, and W. Li, "Exploiting multi-dimensional task diversity in distributed auctions for mobile crowdsensing," *IEEE Transactions on Mobile Computing*, vol. 20, no. 8, pp. 2576–2591, 2021.
[52] J. Liu, Y. Yang, D. Li, X. Deng, S. Huang, and H. Liu, "An incentive mechanism based on behavioural economics in location-based crowdsensing considering an uneven distribution of participants," *IEEE Transactions on Mobile Computing*, vol. 21, no. 1, pp. 44–62, 2022.
[53] C. Zhang, L. Zhu, C. Xu, J. Ni, C. Huang, and X. Shen, "Location privacy-preserving task recommendation with geometric range query in mobile crowdsensing," *IEEE Transactions on Mobile Computing*, vol. 21, no. 12, pp. 4410–4425, 2022.
[54] C. Huang, H. Yu, R. A. Berry, and J. Huang, "Using truth detection to incentivize workers in mobile crowdsourcing," *IEEE Transactions on Mobile Computing*, vol. 21, no. 6, pp. 2257–2270, 2022.
[55] C. H. Liu, Z. Dai, Y. Zhao, J. Crowcroft, D. Wu, and K. K. Leung, "Distributed and energy-efficient mobile crowdsensing with charging stations by deep reinforcement learning," *IEEE Transactions on Mobile Computing*, vol. 20, no. 1, pp. 130–146, 2021.
[56] W. Liu, E. Wang, Y. Yang, and J. Wu, "Worker selection towards data completion for online sparse crowdsensing," in *Proc. INFOCOM*, 2022, pp. 1509–1518.
[57] Z. Dai, C. H. Liu, Y. Ye, R. Han, Y. Yuan, G. Wang, and J. Tang, "Aoi-minimal uav crowdsensing by model-based graph convolutional reinforcement learning," in *Proc. IEEE INFOCOM*, 2022, pp. 1029–1038.
[58] C. Xiang, Y. Li, L. Feng, C. Chen, S. Guo, and P. Yang, "Near-optimal vehicular crowdsensing task allocation empowered by deep reinforcement learning," *Chinese Journal of Computers*, vol. 45, no. 5, pp. 918–934, 2022.

Chaocan Xiang is an associate professor at the College of Computer Science, Chongqing University, Chongqing, China. He received the BS and Ph.D. degrees in computer science and engineering from the Nanjing Institute of Communication Engineering, China, in 2009 and 2014, respectively. He studied at the Real-Time Computing Lab, the University of Michigan, Ann Arbor in 2017. His current research interests include crowd-sensing networks and IoT. He has published more than 20 research papers in important conferences and journals, such as ACM UbiComp, IEEE INFOCOM, IEEE TMC, IEEE TPDS, IEEE T-ITS, IEEE TNSE, and ACM TOSN.

Wenhui Cheng received his B.E. degree in Mechanical and Electronic Engineering from East China Jiaotong University, China, in 2020. He is currently pursuing a Ph.D. degree at Chongqing University, with research interests in drones, urban computing, and crowd-sensing networks.

Chi Lin received B.E. and Ph.D. degrees from Dalian University of Technology, Dalian, China, in 2008 and 2013, respectively. He has been an Assistant Professor with the School of Software, Dalian University of Technology, since 2014. Since 2017, he has been an Associate Professor with the School of Software, Dalian University of Technology. He has authored more than 70 scientific papers in several journals and conferences including Mobicom, INFOCOM, ICNP, ICDCS, SECON, ICPP, IEEE/ACM TON, TMC, ACM TECS, IEEE TVT, and special issue of *SCIENCE*. His research interests include pervasive computing, cyber-physical systems, and wireless sensor networks. In 2015, he was the recipient of the ACM Academic Rising Star. He is a senior member of the IEEE and China Computer Federation (CCF).

Xinglin Zhang received a B.E. degree in School of Software from Sun Yat-sen University in 2010 and a Ph.D. degree in the Department of Computer Science and Engineering from Hong Kong University of Science and Technology in 2014. He is currently an associate professor at South China University of Technology. His research interests include mobile crowdsensing, edge computing, and mobile computing. He is a member of the IEEE and the ACM.

Daibo Liu received the Ph. D. degree in computer science and engineering from the University of Electronic Science and Technology of China, Chengdu, China, in 2018. He was a Visiting Researcher with School of Software, Tsinghua University from 2014-2016, and Department of Electrical and Computer Engineering, University of WisconsinMadison from 2016-2017. He is currently an associate professor with the College of Computer Science and Electronic Engineering, Hunan University, Changsha, China. His research interests include low power wireless transmission, mobile and pervasive computing, and IoT security. He is a member of the IEEE and ACM.

This article has been accepted for publication in IEEE Transactions on Mobile Computing. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TMC.2023.3271671

Xiao Zheng is currently a Professor with the School of Computer Science and Technology, Anhui University of Technology, Anhui, China. He received the B.S. degree from Anhui University, China, in 1997, the M.S. degree from Zhejiang University of Science and Technology, China, in 2003, and the Ph.D. degree in computer science and technology from Southeast University, China, in 2014. He He is also a Guest Scientist with the Institute of Artificial Intelligence, Hefei Comprehensive National Science Center, China. His research interests include service computing, mobile cloud computing and privacy protection. He has been a Guest Editor of IEICE Transactions on Communications. He is a senior member of CCF, and a member of IEEE and ACM.

Zhenhua Li received the BSc and MSc degrees from Nanjing University, in 2005 and 2008 respectively, and the PhD degree from Peking University, in 2013, all in computer science and technology. He is currently an Associate Professor with the School of Software, Tsinghua University. His research areas cover network measurement, mobile networking/emulation, and cloud computing/storage.