Privacy-preserving Cooperative Online Matching over Spatial Crowdsourcing Platforms

Yi Yang
Beijing Institute of Technology
Beijing, China
yangyi@bit.edu.cn

Yurong Cheng
Beijing Institute of Technology
Beijing, China
yrcheng@bit.edu.cn

Ye Yuan
Beijing Institute of Technology
Beijing, China
yuan-ye@bit.edu.cn

Guoren Wang
Beijing Institute of Technology
Beijing, China
wanggr@bit.edu.cn

Lei Chen
The Hong Kong University of Science and Technology
Hong Kong SAR, China
leichen@cse.ust.hk

Yongjiao Sun
Northeastern University
Shenyang, China
sunyongjiao@mail.neu.edu.cn

ABSTRACT

With the continuous development of spatial crowdsourcing platform, online task assignment problem has been widely studied as a typical problem in spatial crowdsourcing. Most of the existing studies are based on a single-platform task assignment to maximize the platform's revenue. Recently, cross online task assignment has been proposed, aiming at increasing the mutual benefit through cooperations. However, existing methods fail to consider the data privacy protection in the process of cooperation and cause the leakage of sensitive data such as the location of a request and the historical data of cooperative platforms. In this paper, we propose Privacy-preserving Cooperative Online Matching (PCOM), which protects the privacy of the users and workers on their respective platforms. We design a PCOM framework and provide theoretical proof that the framework satisfies the differential privacy property. We then propose two PCOM algorithms based on two different privacy-preserving strategies. Extensive experiments on real and synthetic datasets confirm the effectiveness and efficiency of our algorithms.

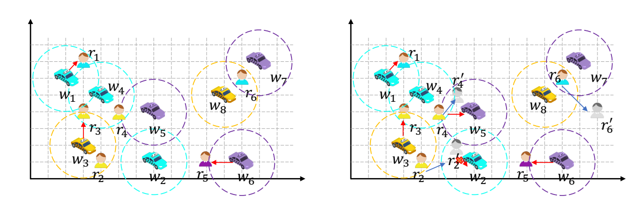
PVLDB Reference Format:
Yi Yang, Yurong Cheng, Ye Yuan, Guoren Wang, Lei Chen, and Yongjiao Sun. Privacy-preserving Cooperative Online Matching over Spatial Crowdsourcing Platforms. PVLDB, 16(1): 51 - 63, 2022.
doi:10.14778/3561261.3561266

PVLDB Artifact Availability:
The source code, data, and/or other artifacts have been made available at https://github.com/Yi107/Privacy-preserving-COM.git.

1 INTRODUCTION

In recent years, with the development of mobile Internet and sharing economy technology, people's daily life has gradually become inseparable from spatial crowdsourcing applications, such as online taxi-calling service (e.g., DiDi [2] and Uber [4]) and food delivery service (e.g., Eleme [5] and Meituan [3]). The main task of the spatial crowdsourcing platform is to arrange suitable workers to complete the spatiotemporal requests on time, and to maximize the platform's revenue or maximize the total matching number [29].

In order to enable the spatial crowdsourcing platforms to allocate requests reasonably, the existing studies design reasonable task matching algorithms to achieve different goals [12, 24, 26]. These existing studies mainly focus on designing matching algorithms for single platform. However, in the real life, the distribution of workers and requests within a single platform greatly affects the throughput of the system. Take the situation in Figure 1 as an example.



(a) Optimal Result of Single Platform Online Matching (b) Optimal result of PCOM

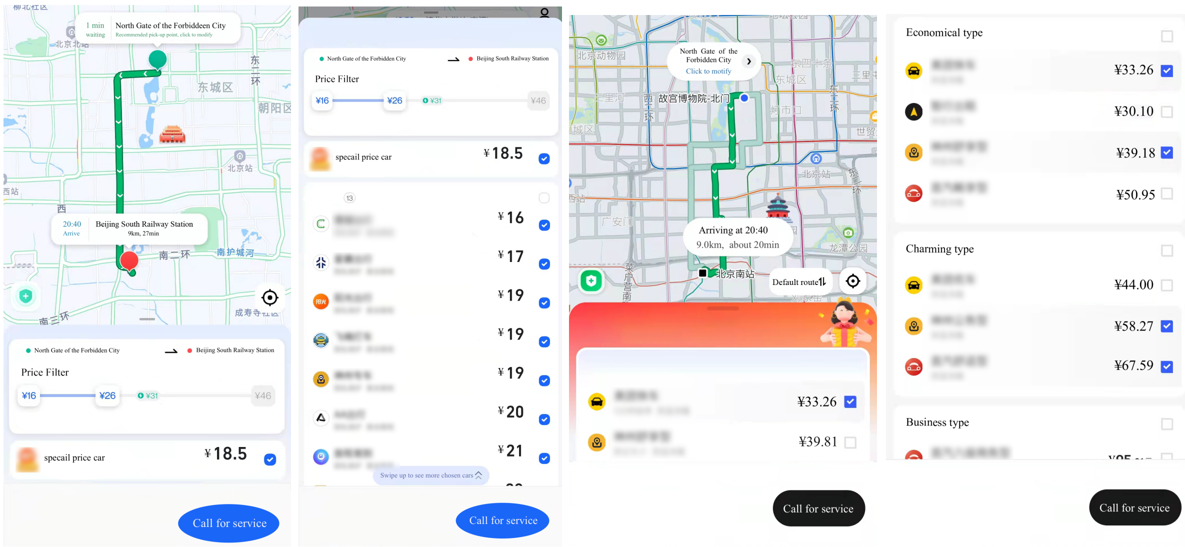
Figure 1: The matching results in different Spatial Crowdsourcing Problems

Example 1: Figure 1 shows the locations of users and cars in different grids. The dashed circle centered on the car represents the service area of the car. Cars and users of the same color belong to the same crowdsourcing platform. We denote the platform in yellow as platform A, the platform in blue as platform B, and the platform in purple as platform C. Table 1 shows the arrival time of cars and users. Figure 1(a) shows the optimal result of single platform online matching. The matching result is (w_1, r_1), (w_3, r_3), (w_6, r_5), leaving r_2 and r_4 unmatched. In this case, the platform usually gives unmatched requests two choices. One is to increase the price of the request, and the platform dispatches workers from farther away (e.g. w_8) to complete the request. The other one is to let users wait until there are available workers nearby. However, these two cases will increase the waiting time of users which will cause a reduction in user satisfaction. The platform will lose users when their satisfaction is too low.

Proceedings of the VLDB Endowment, Vol. 16, No. 1 ISSN 2150-8097.
doi:10.14778/3561261.3561266

Table 1: The Arrival Order of Workers and Requests

Time	8:00	8:00	8:01	8:02	8:04	8:05	8:06	8:07	8:08	8:10	8:11	8:12	8:13	8:13
Arrival Order	w_1	w_2	r_1	r_2	w_3	w_4	r_3	w_5	r_4	w_6	w_7	r_5	w_8	r_6



(a) The route of request in AN (b) Cooperative platforms in AN (c) The route of request in Meituan (d) Cooperative platforms in Meituan

Figure 2: The cooperative process in AutoNavi and Meituan

The existing spatial crowdsourcing platforms, such as AN (AutoNavi) and Meituan choose multi-platform cooperation to solve the above problem. As shown in Figure 2, the request that cannot be completed by the local platform is sent to other platforms. Workers on other platforms finish cooperation through bidding. However, only determining cooperation through worker bidding is not conducive to improving total revenue. It does not consider the global matching situation. In addition, workers have to focus on rushing for requests for a long time, which increases the risk of driving. Research [10] proposes a better way called Cross Online Matching (COM) for cooperation. It enables the platform determines whether to cooperate and assign tasks reasonably. COM improves the efficiency of cooperation while reducing the risk of work.

The above two cooperation methods are still impractical for widely used in real-world, due to the ignorance of data privacy protection. On one hand, the cooperation process of AutoNavi directly exposes the user's location information to all platforms, causing the user's dissatisfaction. On the other hand, the cooperation process of COM requires the historical request information of cooperative platforms to calculate a reasonable revenue. Sending historical data to other platforms may lead to the disclosure of sensitive information, such as the location of workers' frequent visits, the pricing method of the platform, etc. Malicious platforms can use it to infer other sensitive information, such as the preference of the workers and the geographical distribution of requests. Based on the additional information, the malicious company can adjust its operating methods to seize the market. The leakage of worker preferences will lead to malicious platforms to snatch workers from other platforms by adjusting pricing method, and the leakage of geographic distribution of requests will lead to malicious platforms to snatch requests from other platforms by adjusting the distribution of workers.

In this paper, to address the challenges above, we design solutions to perturb the location of requests based on geo-indistinguishable technology and propose two exponential mechanisms to protect the privacy of historical data during the process of evaluating cooperative requests. Most importantly, we provide theoretical proof of our proposed framework that satisfies the privacy requirement.

In summary, the main contributions of this paper are as follows:

- We formulate the Privacy-Preserving Cooperative Online Matching (PCOM) and propose PCOM framework, which considers the privacy protection of sensitive data in the process of multi-platform cooperation and maximizes the revenue of each platform.
- We theoretically prove that the proposed PCOM framework satisfies the differential privacy property. Two algorithms are designed to solve the PCOM problem based on the theoretical proof. Algorithm 1 uses platform-based historical data to directly price the revenue of the request, and Algorithm 2 comprehensively considers revenue and the probability of the request being accepted for pricing.
- Extensive experiments on both real and synthetic datasets verify the effectiveness and efficiency of our algorithms.

2 PROBLEM STATEMENT

This section formally defines the privacy-preserving cooperative online matching problem and the evaluation criteria related to it.

2.1 Problem Definition

Before formally defining our PCOM problem, we first introduce some basic definitions.

Definition 2.1 (Request): A request r in spatial crowdsourcing platform is denoted as $r =< t_r, l_r, d_r, v_r >$, where t_r is the appearance time of r, l_r is the appearance location in 2D space of r, d_r is the travel distance required to complete r, and v_r is the value to be paid to complete r.

Definition 2.2 (Worker): A worker w in spatial crowdsourcing platform is a triplet $w =< t_w, l_w, rad_w >$, where t_w is the appearance time of w, l_w is the appearance location in 2D space of w, and rad_w is the service radius of w in 2D space.

Definition 2.3 (Local/Cooperative Platform): The platform is denoted as $p =< R, W >$, where R is the set of requests on p and W is the set of workers registered on p. Given a request r, the platform where request r appears is denoted as the *local platform* (i.e. p_{loc}) of r. On the contrary, a *cooperative platform* (i.e. p_{cop}) of r is the platform in the cooperation except the local platform. The workers in the local platform called *local worker*, denoted as w_{loc}, and the workers in the cooperative platform called *cooperative worker*, denoted as w_{cop}. The requests assigned to the cooperative platform are *cooperative requests*.

In *Example 1*, for request r_1, platform B is its local platform, while other platforms are the cooperative platforms. Worker w_1, w_2, w_4 are the local workers of r_1, while others are the cooperative workers. If r_4 is assigned to w_5, r_4 is cooperative requests.

Definition 2.4 (Outer Payment [10]): When a cooperative worker is needed to serve a request r, s/he would like to obtain a payment $v_r' \in (0, v_r]$. In this case, v_r' is called the outer payment of r.

We assume that the platform will not complete the request to make itself lose money (i.e. $v_r' < v_r$).

Definition 2.5 (Revenue [10]): We consider the revenue of each platform in two cases. In the first case, r is served by its local worker. The platform would receive v_r. In the second case, r is served by a cooperative worker. The platform would receive $v_{r_i} - v_{r_i}'$. Assume that M is a feasible matching. Let M_{loc} be the matching that satisfies the first case, and M_{cop} be the matching that satisfies the second case. The total revenue of the platform is calculated by:

$$Rev = \sum_{i=1}^{|M_{loc}|} v_{r_i} + \sum_{i=1}^{|M_{cop}|} (v_{r_i} - v_{r_i}') \tag{1}$$

Now we can define our *Privacy-preserving Cooperative Online Matching (PCOM)* Problem as follows.

Definition 2.6 (Privacy-preserving Cooperative Online Matching Problem): Given a set of spatial crowdsourcing platforms P willing to participate in cooperation, each platform $p \in P$ contains the requests and the workers. The workers and requests appear sequentially. The PCOM problem aims to find a cooperative matching result M with a maximum revenue *Rev* for every platform in the cooperation, under the following constraints:

- Time constraint: the requests can only be completed by the workers who appear before it.
- 1-By-1 constraint: one request can only be served by one worker at a time. Vise versa.
- Invariable constraint: once a worker is assigned to a request, the assignment cannot be changed or revoked.
- Range constraint: a worker can only serve the requests whose location is in the service radius of this worker.
- Privacy constraint: it contains two parts:
  - Real-time data privacy: the precise location of the cooperative request is the most important sensitive real-time data, so we consider its privacy as real-time data privacy. When a request needs to be served by cooperative workers, the precise location of the request should not be exposed to cooperative platforms.
  - Historical data privacy: the sensitive historical data of workers contains the appearance location, appearance time, ending location, ending time, and value of his/her completed requests. These sensitive data should not be revealed to cooperative platforms.

In the case of a worker serving multiple requests at the same time, the PCOM problem can be regarded as multiple workers appearing in the same location at the same time. The worker will appear on the platform again after s/he finishes the service.

PCOM problem mainly focuses on the privacy-preserving process in cooperative online matching. Thus, the privacy-preserved matching algorithm should ensure the privacy constraint.

2.2 Evaluation Criteria

Differential privacy is widely used in the fields of database query and geographic perturbation as an evaluation criterion of privacy. Therefore, we also use the same evaluation criterion to measure the degree of privacy protection.

Definition 2.7 (Differential Privacy [13]): Assume that D and D' are two neighboring datasets which differ on at most one element. Given a randomized algorithm A, Range(A) is the set of all possible outputs of A in D and D'. A is ε-differentially private if for any arbitrary output $O \in$ Range(A):

$$\Pr[A(D) = O] \leq \exp(\varepsilon) * \Pr[A(D') = O] \tag{2}$$

The privacy budget ε is used to control the ratio of the probability that A outputs a value in the same range under different inputs. It is obvious that the smaller ε is, the greater the similarity of the output probability distribution of A under different inputs is, that is to say, A guarantees a better privacy-preserving level. Normally, $\varepsilon \in (0, 1]$. Using this mechanism can effectively protect any piece of data of any worker from leaking to other crowdsourcing platforms.

Definition 2.8 ((ε, r)-Geo-indistinguishability [7]): Based on differential privacy, a notion of location privacy has been proposed, called Geo-indistinguishability(Geo-I). Assume that $\mathcal{X}$ is a set of exact locations. Given a randomized algorithm A, Range(A) is the set of possible reported outputs. A is (ε, r)-Geo-indistinguishability if for all $x, x' \in \mathcal{X}$, $z \in$ Range(A), which $d(x, x') \leq r$:

$$\Pr[A(x) = z] \leq e^{\varepsilon d(x,x')} \Pr[A(x') = z] \tag{3}$$

Similarly, ε represents the privacy budget of A, r represents the radius that satisfies the privacy constraint. Geo-I makes two input locations with the same output indistinguishable. Hence, the precise location of the task will not be leaked to other platforms.

3 THE PCOM PROBLEM FRAMEWORK

In this section, we first illustrate the framework of the PCOM problem. As discussed in the introduction, different from the existing AutoNavi and COM methods, our PCOM framework can protect the privacy of both real-time data and historical data. We also theoretically prove that the proposed PCOM framework satisfies the differential privacy property.

3.1 The Framework of PCOM Problem

The PCOM framework contains two cases. When a request r appears on the local platform p_{loc}, p_{loc} determines whether there are available local workers based on the location of r. If the available local worker set is not empty, p_{loc} will assign r to a suitable local worker using the task assignment methods such as TOTA [26] and FTOA[28]. If there is no available local worker, p_{loc} will send r to the cooperative platforms and performs the cooperative process.

Figure 3 shows the cooperative process of the PCOM framework. It consists of five steps:

- To ensure privacy, the local platform (platform A) perturbs the location of r with a privacy level (ϵ_1, r)-Geo-I according to the Geo-I mechanism and sends the perturbed request $r^{'}$ to all the cooperative platforms (platform B and C).
- Each cooperative platform finds available workers according to $l_r^{'}$ and the constraints in Definition 2.6. The platforms which have available workers use a differential privacy mechanism with $\epsilon_2 - DP$ privacy level to estimate the outer payment $v_r^{'}$ of r and send $v_r^{'}$ to the local platform.
- After receiving the outer payments returned by all cooperative platforms, the local platform selects one with the smallest $v_r^{'}$ for cooperation (platform B).
- The selected cooperative platform determines whether and which available cooperative worker to serve r according to the acceptance probability of him/her. The acceptance probability can be calculated by the definition in [10]. We assume that the cooperative platform will tell its workers that r belongs to other platforms, and the cooperative workers have the right to refuse to serve r.
- Finally, once there is a cooperative worker willing to serve r, the local platform releases r with a precise location. The cooperative worker can then verify whether r satisfies the range constraint in Definition 2.6. If the range constraint is met, r can be served; otherwise, r is rejected. If r is successfully served, the cooperative platform will receive $v_r^{'}$ and the local platform will receive $v_r - v_r^{'}$.

Step 1 shows the cooperative platforms only obtain the request with a perturbed geographic location. Step 2 shows that in the PCOM framework, the outer payment is estimated by the cooperative platform itself. In this case, the historical request data always stays in its owner's platform and never be sent out, which avoids the leakage of historical data. This pricing method may lead to the cooperative platform maliciously increasing the outer price in order to earn more revenue. However, considering that the local platform will choose a platform with smaller outer payment to cooperate, the cooperative platform will return a reasonable price.

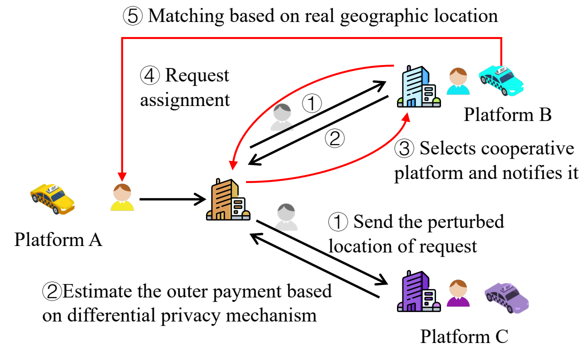


Figure 3: The Cooperative Process in PCOM Framework

According to the existing study [23], we assume that the workers can obtain the exact locations of the cooperative requests via an extra privacy channel after the assignment. Therefore, cooperations among all the platforms can be completed successfully.

Example 2: Figure 1(b) shows the optimal result of PCOM. For platform A, when r_3 appears on the platform, the platform performs a local matching process which is the same as the COM framework. r_3 is assigned to w_3. When r_4 appears on the platform, the platform performs cooperative process. The location of r_4 (l_{r_4}) is perturbed to $l_{r_4}^{'}$. Then the platform sends $r_4^{'}$ to platform B and platform C. These two platforms estimate the outer payment based on differential privacy mechanism. Suppose $v_{r_4} = 10$, the outer payment in platform B is 7 while that in platform C is 6. Platform A chooses platform C for cooperation. Platform C decides to send worker w_5 out for cooperation. Finally, the precise location of r_4 is released to w_5. r_4 satisfies the range constrain, thus r_4 can be served by w_5.

It is obvious that the privacy protection mechanism will affect the matching results. Therefore, it is necessary to analyze the privacy of the PCOM framework. We theoretically prove that the PCOM framework provides $((\epsilon_1 + \epsilon_2) * \max_{p \in P} |p_W|)$-differential privacy, where $|p_W|$ is the number of workers in cooperative platform p.

3.2 Analysis of the Privacy of PCOM

In this section, we provide the privacy analysis of the PCOM framework. We prove that the framework is ε-differentially private.

PCOM returns multi-platform matching results and their revenue. Therefore, for any request, its matching result directly affects the final result of PCOM. For any request, its geographic location privacy and the privacy of platform historical data are considered in the PCOM framework. Obviously, the privacy of one request does not affect the privacy of another. Thus, the privacy of each request in the matching process is independent, so the impact of the privacy on the PCOM framework can be reduced to its impact on the matching results of a single request. Based on this feature, we focus on the privacy of a single request matching process.

LEMMA 1. *Assume that r_a is a cooperative request, W_B is the set of all cooperative workers of r_a, $w_{B_i} \in W_B$, and Match is a matching* *mechanism satisfies PCOM framework. The probability that r_a is served by the cooperating workers can be calculated by*

$$\begin{aligned} &Pr[Match(r_a, W_B) = 1] \\ &= 1 - \prod_{i=0}^{||W_B||} [1 - Pr[Match(r_a, w_{B_i}) = 1]] \end{aligned} \tag{4}$$

It is obvious that lemma 1 holds since r_a can be served only if there is one acceptable cooperating worker. Therefore, the following privacy analysis will be based on the matching of r_a and a single cooperative worker w_{B_i}.

For a given r_a and w_{B_i}, two conditions must be met for r_a to be served by w_{B_i}. First, the range constrain, denoted as $dis(l_{r_a}, l_{w_{B_i}}) \leq rad_{w_{B_i}}$, where $dis(a, b)$ represents the Euclidean distance between two locations. Second, the outer payment v'_{r_a} should be less than the value of r_a itself, denoted as $v'_{r_a} \leq v_{r_a}$.

The PCOM framework adds privacy in two parts: 1. The platform perturbs the location of r with a privacy mechanism K that satisfies (ϵ_1, r)-Geo-I; 2. The cooperative platform uses a privacy mechanism $Estprice$ that satisfies ϵ_2-DP to calculate outer payment. The historical data of cooperative workers that needs to be used in $Estprice$ denoted as his_{W_B}. The conditions above are changed into $dis(K(l_{r_a}), l_{w_{B_i}}) \leq rad_{w_{B_i}}$ and $Estprice(r_a, his_{W_B}) \leq v_{r_a}$. These conditions are independent to each other, thus the equation $Pr[Match(r_a, w_{B_i}) = 1]$ can be calculated by

$$\begin{aligned} &Pr[Match(r_a, w_{B_i}) = 1] \\ &= Pr[dis(K(l_{r_a}), l_{w_{B_i}}) \leq rad_{w_{B_i}}] * Pr[Estprice(r_a, his_{W_B}) \leq v_{r_a}] \end{aligned} \tag{5}$$

We discuss the two parts of equation 5 separately.

Lemma 2. *Given any privacy mechanism K satisfies ϵ-GEO-I,*

$$Pr[dis(K(l_{r_a}), l_{w_{B_i}}) = r] \leq e^{\epsilon d(l_{r_a}, l')} Pr[dis(K(l'), l_{w_{B_i}}) = r] \tag{6}$$

where l' is another location on the map.

Proof. $Pr[dis(K(l_{r_a}), l_{w_{B_i}}) = r]$ is equal to the probability that $l_{l_{r_a}}$ is mapped to a point on a circle with $l_{w_{B_i}}$ as the center and r as the radius. Since K satisfies ϵ-GEO-I [7], equation 6 holds. □

Theorem 1. *Given any privacy mechanism K satisfies ϵ-GEO-I,*

$$\begin{aligned} &Pr[dis(K(l_{r_a}), l_{w_{B_i}}) \leq rad_{w_{B_i}}] \\ &\leq e^{\epsilon d(l_{r_a}, l')} Pr[dis(K(l'), l_{w_{B_i}}) \leq rad_{w_{B_i}}] \end{aligned} \tag{7}$$

Proof. From lemma 2, since function dis is a continuous function, the CDF of it can be expressed as the integral of its probability density function.

$$\begin{aligned} &Pr[dis(K(l_{r_a}), l_{w_{B_i}}) \leq rad_{w_{B_i}}] \\ &= \int_0^{rad_{w_{B_i}}} Pr[dis(K(l_{r_a}), l_{w_{B_i}}) = r] dr \\ &\leq \int_0^{rad_{w_{B_i}}} e^{\epsilon d(l_{r_a}, l')} Pr[dis(K(l'), l_{w_{B_i}}) = r] dr \\ &= e^{\epsilon d(l_{r_a}, l')} Pr[dis(K(l'), l_{w_{B_i}}) \leq rad_{w_{B_i}}] \end{aligned} \tag{8}$$

□

Theorem 1 shows the privacy properties of the first part in equation 5. Given the privacy mechanism K satisfies ϵ-GEO-I, the difference between the probabilities of two locations in the graph satisfying the worker's service range is limited to $e^{\epsilon d(l_{r_a}, l')}$.

Lemma 3. *Given any privacy mechanism Estprice satisfies ϵ-DP, $Pr[Estprice(r_a, his_{W_B}) = p] \leq e^{\epsilon} Pr[Estprice(r_a, his'_{W_B}) = p]$ where his_{W_B} and his'_{W_B} differ in a single historical request data.*

Lemma 3 is a special property of differential privacy. When the privacy mechanism satisfies ϵ-DP, lemma3 holds [14].

Theorem 2. *Given any privacy mechanism Estprice satisfies ϵ-DP,*

$$Pr[Estprice(r_a, his_{W_B}) \leq v_{r_a}] \leq e^{\epsilon} Pr[Estprice(r_a, his'_{W_B}) \leq v_{r_a}] \tag{9}$$

Proof. From lemma 3, since function $Estprice$ is a discrete function, the CDF of it can be expressed as the sum of its probability density function.

$$\begin{aligned} &Pr[Estprice(r_a, his_{W_B}) \leq v_{r_a}] \\ &= \sum_{p_i=0}^{v_{r_a}} Pr[Estprice(r_a, his_{W_B}) = p_i] \\ &\leq \sum_{p_i=0}^{v_{r_a}} e^{\epsilon} Pr[Estprice(r_a, his'_{W_B}) = p_i] \\ &= e^{\epsilon} Pr[Estprice(r_a, his'_{W_B}) \leq v_{r_a}] \end{aligned} \tag{10}$$

□

Theorem 2 shows the privacy properties of the second part in equation 5. Given the privacy mechanism $Estprice$ satisfies ϵ-DP, the difference between the probabilities of the outer payment calculated on two adjacent historical datasets smaller than the value of request is limited to e^{ϵ}.

Corollary 1. *The privacy of a successful single request matching process satisfies $(\epsilon_1 d(l_{r_a}, l') + \epsilon_2)$-DP, where ϵ_1 is the privacy level of K in PCOM, ϵ_2 is the privacy level of Estprice in PCOM, $d(l_{r_a}, l')$ is the range of geographical protection.*

Proof. Based on theorem 1 and 2, equation 5 satisfies

$$\begin{aligned} &Pr[Match(r_a, w_{B_i}, his_{W_B}) = 1] \\ &= Pr[dis(K(l_{r_a}), l_{w_{B_i}}) \leq rad_{w_{B_i}}] * Pr[Estprice(r_a, his_{W_B}) \leq v_{r_a}] \\ &\leq e^{\epsilon_1 d(l_{r_a}, l')} Pr[dis(K(l'), l_{w_{B_i}}) \leq rad_{w_{B_i}}] * \\ &e^{\epsilon_2} Pr[Estprice(r_a, his'_{W_B}) \leq v_{r_a}] \\ &= e^{(\epsilon_1 d(l_{r_a}, l') + \epsilon_2)} Pr[Match(r'_a, w_{B_i}, his'_{W_B}) = 1] \end{aligned} \tag{11}$$

□

Considering that request is not successfully served. The privacy of this result also satisfies $(\epsilon_1 d(l_{r_a}, l') + \epsilon_2)$-DP. The proof is the same as the above process, in order to avoid redundancy, we will not describe it in detail here.

After finishing the privacy analysis based on the matching of r_a and a single cooperative worker w_{B_i}, we extend it to multiple cooperative workers.

Corollary 2. *The privacy of a PCOM framework provides at least* $((\epsilon_1 d(l_{r_a}, l') + \epsilon_2) * \max_{p \in P} |p_W|)$*-differential privacy, where* ϵ_1 *is the privacy level of* K *in PCOM,* ϵ_2 *is the privacy level of* $Estprice$ *in PCOM,* $d(l_{r_a}, l')$ *is the range of geographical protection,* P *is the platform set consists of cooperative platforms.*

Proof. Based on corollary 1, we first discuss the request which cannot be served based on PCOM framework. To simplify the formula, we denote $Pr[Match(r_a, w_{B_i}, his_{W_B}) = 1]$ as P_{aBi}, and $Pr[Match(r'_a, w_{B_i}, his'_{W_B}) = 1]$ as P'_{aBi}.

$$\begin{aligned} Pr[Match(r_a, W_B) = 0] &= \prod_{i=0}^{||W_B||} (1 - P_{aBi}) \\ &\leq \prod_{i=0}^{||W_B||} e^{(\epsilon_1 d(l_{r_a}, l') + \epsilon_2)} (1 - P'_{aBi}) \\ &= e^{(\epsilon_1 d(l_{r_a}, l') + \epsilon_2)||W_B||} Pr[Match(r'_a, W'_B) = 0] \end{aligned} \tag{12}$$

We then discuss the request which can be served. A request can be served means that at least one worker can successfully serve it, so the following formula is satisfied.

$$\begin{aligned} &Pr[Match(r_a, W_B) = 1] \\ &= \prod_{i=0}^{||W_B||} P_{aBi} + C^i_{||W_B||} (P_{aBi})^i * (1 - P_{aBi})^{||W_B|| - i} \\ &\leq e^{(\epsilon_1 d(l_{r_a}, l') + \epsilon_2)||W_B||} Pr[Match(r'_a, W'_B) = 1] \end{aligned} \tag{13}$$

□

In practical applications, the perturbation range of precise geographical location is generally 1 km. So when range of geographical protection is set to 1 km, PCOM framework provides $((\epsilon_1 + \epsilon_2) * \max_{p \in P} |p_W|)$-differential privacy.

4 DIRECT PCOM ALGORITHM

In this section, we propose a Direct Privacy-preserving Cross Online Matching (*D-PCOM*) algorithm to solve the PCOM problem in a greedy form. To maximize the benefits of the local platform, D-PCOM prioritizes requests to local workers.

4.1 Overview

We first introduce the matching process of D-PCOM. When a request r appears to a platform, the platform first finds the available local worker w_{loc} to serve r. If no local worker is available, D-PCOM will consider cooperating with the cooperative platform. The local platform first perturbs the location of r based on the Geo-I mechanism in [7] and send r' to all cooperative platforms. The cooperative platform returns the outer payment v'_r based on all historical request information with a similar travel distance through a privacy mechanism. The following cooperative matching process follows

Algorithm 1: The D-PCOM Algorithm

Input: R, W, P_{cop}, ϵ_1, ϵ_2
Output: The matching M, total revenue Rev

1 $M = \emptyset$, $Rev = 0$, $V'_r = \emptyset$
2 **foreach** *new arrival* $r \in R$ **do**
3   Find the available local worker $w_{loc} \in W$ for r
4   **if** $w_{loc} \neq NULL$ **then**
5   Assign r to w_{loc} and put $< r, w_{loc} >$ into M
6     $Rev += v_r$
7   **else**
8   Perturb the location of r with privacy level ϵ_1 and send r' to the cooperative platforms in P_{cop}
9     $V'_r \leftarrow \texttt{Algorithm2}(r', P_{cop}, \epsilon_2)$;
10   **if** $\min V'_r < v_r$ **then**
11       $v'_r, p_{cop} \leftarrow \min V'_r$
12   p_{cop} determines available w_{cop} to serve r'
13   **if** $w_{cop} \neq NULL$ **then**
14   r send real location to w_{cop} in a special way
15   **if** $dis(l_{w_{cop}}, l_r) \leq rad_{w_{cop}}$ **then**
16   Assign r to w_{cop} and put $< r, w_{cop} >$ into M
17           $Rev += v_r - v'_r$
18   **else**
19   Reject r
20   **else**
21   Reject r
22   **else**
23   Reject r
24 **return** M, Rev

steps 3 to 5 in the PCOM framework. D-PCOM finally returns the matching result and the total revenue.

4.2 Algorithm Details

Algorithm 1 shows the procedure of D-PCOM. Since the matching process of each platform participating in cooperation is the same, we describe D-PCOM algorithm from the perspective of one platform.

The input of D-PCOM is the request set R, the local worker set W, the cooperative platform set P_{cop}, the privacy level to perturb location ϵ_1, and the privacy level ϵ_2 to protect sensitive data. The output of D-PCOM is a matching result M and the revenue Rev of platform. Initially, let $M = \emptyset$, $Rev = 0$, and the outer payment set $V'_r = \emptyset$ (Line 1). For each new arrival request r, the platform finds the available worker satisfying all constraints to serve it. If the available worker w_{loc} exists, the platform assigns r to w_{loc} and the total revenue is updated by $Rev = Rev + v_r$ (Line 3-6). If no available local worker is found, the platform perturbs the location of r based on the Geo-I mechanism in [7] with privacy level ϵ_1 and performs the cooperative process (Line 8). In the cooperative process, D-PCOM calls Algorithm 2 to calculate the outer payment set (Line 9). The elements in the outer payment set include the calculated outer payment and its corresponding platform. The details of Algorithm

Algorithm 2: Direct Pricing Algorithm

Input: $r^{'}$, P_{cop}, ϵ_2

Output: The outer payment set $V_r^{'}$

1 $V_r^{'} = \emptyset$, $d_{min} = \lfloor d_{r'} \rfloor$, $d_{max} = \lceil d_{r'} \rceil$
2 **foreach** *cooperative platform* $p_{cop} \in P_{cop}$ **do**
3   Query the unit price distribution set U of historical requests which satisfies $d_r \in [d_{min}, d_{max}]$
4   **foreach** (u_i,num_i) $\in U$ **do**
5   calculate $Pr[u = u_i] = \frac{e^{\epsilon_2 * \ln num_i / 2 \ln 2}}{\sum_{j=0}^{|U|} e^{\epsilon_2 * \ln num_j / 2 \ln 2}}$
6   select $u_i^{'}$ based on the probability distribution of u
7   insert $(u_i^{'} * d_{r'}, p_{cop})$ into $V_r^{'}$
8 **return** $V_r^{'}$

2 will be illustrated later. After calculating the outer payment, the platform selects one cooperative platform p_{cop} with the smallest outer payment $v_r^{'}$ which is smaller than v_r (Line 11). Then p_{cop} determines available cooperative worker based on the constraints and the acceptance probabilities (Line 12). Once there is an available cooperative worker, the platform release the real location to him/her (Line 14). If the real location still satisfies the range constrain, the platform assigns r to w_{cop} and updates the revenue (Line 16-17). The request will be rejected in other cases (Line 19 and 21).

Algorithm 2 shows a pricing algorithm with a privacy mechanism. We choose the exponential mechanism in differential privacy since it has better performance at returning discrete results. We first introduce the pricing mechanism and then prove its privacy.

The value of a request on a platform is calculated by the unit price and its distance. For given cooperative request $r^{'}$, the platform first queries the unit price distribution of historical requests which satisfies $d_r \in [\lfloor d_{r'} \rfloor, \lceil d_{r'} \rceil]$ (Line 3). When privacy protection is not considered, we select the unit price with the most occurrences and calculate outer payment based on the selected unit price and $d_{r'}$.

According to the idea above, we design a privacy mechanism that satisfies ϵ_2-differential privacy. Based on the definition in exponential mechanism [19], we set the utility function $F(p_{cop}, u_i)$ that returns the logarithm of occurrences of u_i. The sensitivity of F is $\ln 2$. Then we calculate the probability distribution of u_i which is proportional to $e^{\epsilon_2 * \ln num_i / 2 \ln 2}$ (Line 5). We select $u_i^{'}$ based on the probability distribution of u_i ($\{p_0, ... p_n\}$) by randomly generate a decimal number $a \in [0, 1]$. For $x \in [0, n]$, if $a \in [\sum_0^x p_j, \sum_0^{x-1} p_j)$, we return the xth unit price as $u_i^{'}$ (Line 6). We finally insert cooperative platform with its outer payment in $V_r^{'}$ (Line 7).

Theorem 3. *The sensitive of the utility function in the exponential mechanism proposed above is* $\ln 2$.

Proof. $\max ||F(p_{cop}, u_i) - F(p_{cop}^{'}, u_i)||_1 = \ln(x+1) - \ln x \leq \ln 2$ where p_{cop} and $p_{cop}^{'}$ are adjacent datasets and x is the number of u_i in p_{cop}. ◻

Example 3: Considering the situation in Figure 1(b). For request r_1, platform B finds an available worker w_1. Thus, D-PCOM performs local matching and assigns w_1 to serve it. For request r_4($d_r = 2.5, v_r = 10$), platform A fails to find any available worker for it, it perturbs the location to $r_4^{'}$ and sends to platform B and C. Assume that the privacy level $\epsilon_2 = 0.5$. Take platform C as a pricing example. Assume that the unit price distribution in platform C is $\{(2.8, 5), (3, 2), (3.1, 4), (3.5, 1)\}$, where $(2.8, 5)$ means there are 5 historical requests in platform C which satisfies $d_r \in [2, 3]$ whose unit price is 2.8. Then $Pr(u = 2.8) = \frac{e^{0.5 * \ln 5 / 2 \ln 2}}{\sum_{j=0}^{j=4} e^{0.5 * \ln num_j / 2 \ln 2}} = 0.312$. The probability distribution of u_i is $(0.312, 0.224, 0.289, 0.175)$. Assume that the algorithm selects 3.1 as unit price, then $v_{r_4}^{'} = 3.1 * 2.5 = 7.75$. $(7.75, p_C)$ is inserted to $V_r^{'}$. Assume that $V_r^{'} = \{(10.5, p_B), (7.75, p_C)\}$. Since $7.75 < 10$ platform A selects platform C for cooperation. Assume that w_5 is willing to serve $r_4^{'}$ and the real location still satisfies the range constrain, r_4 is assigned to w_5. Platform C receives 7.75 and platform A receives $10 - 7.75 = 2.25$.

4.3 Algorithm Analysis

D-PCOM algorithm satisfies the privacy requirements in the PCOM framework and performs matching strictly following the steps in the framework. Based on the property of the framework, D-PCOM algorithm provides $((\epsilon_1 + \epsilon_2) * \max_{p \in P} |p_W|)$-differential privacy.

The calculational complexity of Algorithm 2 is $O(|P_{cop}| \max |U|)$, where $|P_{cop}|$ is the number of cooperative platforms and $\max |U|$ is the largest number of unit prices obtained from historical requests. In D-PCOM, the complexity of local matching process is $O(|W|)$ which is formed by finding available local workers. The complexity of cooperative process is $O((|P_{cop}| \max |U|) + \max |W_{cop}|)$, where $\max |W_{cop}|$ is the maximum size of worker set in cooperative platform. The complexity of D-PCOM algorithm is $O(|R| * (|W| + (|P_{cop}| \max |U|) + \max |W_{cop}|))$. The space complexity is $O(|W| + |R|)$, where $|W|$ (resp. $|R|$) is the size of workers (resp. requests).

4.4 Shortcomings of D-PCOM

Similar to the shortcomings in DemCOM [10], the D-PCOM algorithm has two shortcomings. First, in the process of matching, D-PCOM may assign too many local workers to perform the requests with small value. This will lead to the loss of requests with larger prices caused by the leakage of workers, reducing the revenue of the platform. Therefore, the platform should allocate more high-priced requests to local workers. Second, in the process of calculating the outer payment, D-PCOM considers the historical requests of all workers on the platform, which ignores the preference of available workers of the request. Therefore, for the globally calculated $v_r^{'}$, the acceptance probability in $v_r^{'}$ of available workers is small in some areas with higher unit price, resulting in a higher probability of r being rejected. Similarly, in some areas with lower unit price, $v_r^{'}$ may be greater than v_r, resulting in r being directly rejected. On the other hand, since the cooperative platforms perform the calculation of the outer payment, they should consider the willingness of cooperation in local platform. To overcome these shortcomings, we propose a selectable algorithm.

5 SELECTABLE PCOM ALGORITHM

In this section, we propose a Selectable Privacy-preserving Cross Online Matching (*S-PCOM*) algorithm to solve the PCOM problem which overcomes the shortcomings in D-PCOM.

5.1 Overview

To solve the first shortcoming, we filter requests by computing a random threshold. Requests with value larger than the threshold are assigned based on the local matching process, while others are assigned based on the cooperative matching process. To solve the second shortcoming, in the cooperative matching process, we first find the available cooperative workers according to the disturbed geographic location, and calculate the outer payment based on the acceptance probability of the available cooperative workers. Due to the competitive relationship between cooperative platforms, the higher the outer payment, the lower the probability of the cooperative platform being selected by the local platform. Therefore, considering the willingness of cooperation in local platform, the expectation of outer payment for worker w is calculated as

$$\mathbb{E}(v_i, w) = v_i * (1 - Pr(v_i, w)) \tag{14}$$

where $Pr(v_i, w)$ is the probability that w would like to serve the request with outer payment v_i. Equation 14 estimates the expectation of the outer payment of the local platform through the acceptance probability of the available cooperative workers. The cooperative platform calculates the outer payment $v_r^{'}$ based on the value of the expectation of outer payment through a privacy mechanism.

5.2 Algorithm Details

Algorithm 3 shows the procedure of S-PCOM. The input and output of S-PCOM are the same as those in D-PCOM. Initially, let $\theta = \lceil \ln(\max(v_r) + 1) \rceil$, where $\max(v_r)$ is the largest value of requests in R (Line 1). k controls the threshold of value which is mentioned in section 5.1. An integer from $[1, \theta]$ is chosen according to the probability $\frac{1}{\theta}$ and given to k (Line 2). For each new arrival request r, if $v_r \leq e^k$, the algorithm performs local matching process (Line 5). Otherwise, the platform perturbs the location of r with privacy level ϵ_1 and performs the cooperative process (Line 7). In the cooperative process, S-PCOM calls Algorithm 4 to calculate the outer payment set (Line 8). The details of Algorithm 4 will be illustrated later. After calculating the outer payment, the platform continues the following cooperation process (Line 9).

Algorithm 4 shows a pricing algorithm with a privacy mechanism. Considering the preference of available workers and the willingness of cooperation in local platform, the cooperative platform should return an outer payment with a larger expectation. Thus, we set the utility function $F(p_{cop}, v_i)$ that returns the expectation of v_i. We do not consider the based measure of p_i since its uniform. The sensitivity of F is $(\max v * \frac{1}{\max |His_w|})$, where $\max v$ is the maximum value of historical request and $\max |His_w|$ is the maximum size of historical request for a single worker.

The cooperative platform first finds the available worker set (W_{ava}) of $r^{'}$ based on the location of $l_{r'}$ (Line 4-6). For each available worker, the platform queries the historical data whose travel distance is in $[d_{min}, d_{max}]$ and formulate the historical request set R_{his} and historical value set V_{his} (Line 7-11). Based on the historical value set, the platform calculates the expectation of each historical value v_i for all available workers and set the maximum expectation as the expectation of v_i (Line 12-15). Then we calculate the probability distribution of v_i which is proportional to $e^{\epsilon_2 * \mathbb{E}(v_i)/2(\max v/\max |His_w|)}$ (Line 17). We do not consider the based measure of v_i since its uniform. We finally select $v_i^{'}$ based on the probability distribution of v_i with the same method in Algorithm 2 and insert cooperative platform with its outer payment in the set of outer payment (Line 18-19).

Algorithm 3: The S-PCOM Algorithm

Input: R, W, P_{cop}, ϵ_1, ϵ_2
Output: The matching M, total revenue Rev

```
1  θ = ⌈ln(max(v_r) + 1)⌉, M = ∅, Rev = 0, V'_r = ∅;
2  k ← randomly choosing an integer from [1, θ] with
     probability 1/θ
3  foreach new arrival r ∈ R do
4      if v_r ≥ e^k then
5          Call Line 3-6 of Algorithm 1 to perform local
             matching process
6      else
7          Perturb the location of r with privacy level ε_1 and
             send r' to the cooperative platforms in P_cop
8          V'_r ← Algorithm4(r', P_cop, ε_2);
9          Call Line 11-21 of Algorithm 1 to perform
             cooperative matching process
10 return M, Rev
```

Theorem 4. *The sensitive of the utility function in the exponential mechanism proposed above is* $(\max v/\min |His_w|)$

Proof.

$$\max ||F(p_{cop}, v_i) - F(p_{cop}^{'}, v_i)||_1 = v_i * \max ||\Delta Pr[v_i, w]||_1$$
$$= v_i * \max\{|\frac{n}{N+1} - \frac{n}{N}|, |\frac{n+1}{N+1} - \frac{n}{N}|\} \leq \max v/\min |His_w| \tag{15}$$

where n is the number of historical data with $v_r < v_i$ and N is the total number of historical data of any worker. ◻

Example 4: Considering the situation in Figure 1(b). Assume that v_{r_1}=15,v_{r_4}=6,ϵ_2=0.5. Suppose $k = 2$. When r_1 arrives, $v_{r_1} > e^2$, S-PCOM performs local matching and assign w_1 to serve it. When r_4 arrives, $v_{r_4} < e^2$, S-PCOM performs cooperative matching. Assume there is only one available worker in platform C and $V_{his} = \{4.0, 4.5, 5.2, 6.7, 7\}$. His/her acceptance probability of the historical value set is {(4.0, 0.56), (4.5, 0.6), (5.2, 0.7), (6.7, 0.8), (7, 0.9)}. Then the expectation of outer payment in platform C is calculated based on line 14-15 in Algorithm 4, {(4.0, 1.76), (4.5, 1.8), (5.2, 1.56), (6.7, 1.34), (7, 0.7)}. Assume that $|His_w| = 10$. Then $Pr[v = 4.0] = \frac{e^{0.5*1.76/2*(7/10)}}{\sum_{j=0}^{j=5} e^{0.5*\mathbb{E}(v_j)/2(7/10)}} = 0.223$. The total probability distribution of v_i is {(0.223, 0.226, 0.207, 0.192, 0.152)}. Assume that 5.2 is selected as outer price, then platform A receives $6 - 5.2 = 0.8$.

5.3 Algorithm Analysis

Same as D-PCOM, S-PCOM also provides $((\epsilon_1 + \epsilon_2) * \max_{p \in P} |p_W|)$-differential privacy. The calculational complexity of Algorithm 4 is $O(|P_{cop}| * |W_{p_{cop}}| * |V_{his}|)$, where $|P_{cop}|$ is the number of cooperative platforms, $|W_{p_{cop}}|$ is the number of workers in cooperative platform, and $|V_{his}|$ is the number of historical value. In Algorithm

Algorithm 4: Optimal Pricing Algorithm

Input: $r^{'}$, P_{cop}, ϵ_2

Output: The outer payment set $V_r^{'}$

$V_r^{'} = \emptyset$, $d_{min} = \lfloor d_{r'} \rfloor$, $d_{max} = \lceil d_{r'} \rceil$

foreach *cooperative platform* $p_{cop} \in P_{cop}$ **do**

  $W_{ava} = \emptyset$, $R_{his} = \emptyset$, $V_{his} = \emptyset$

  foreach $w \in W_{p_{cop}}$ **do**

    if $dis(l_w, l_{r'}) \le rad_w$ **then**

      Insert w into W_{ava}

  foreach $w \in W_{ava}$ **do**

    foreach $r \in R_w$ *and* $d_r \in [d_{min}, d_{max}]$ **do**

      Insert r into R_{his}

      if $v_r \notin V_{his}$ **then**

        Insert v_r into V_{his}

  foreach $v_i \in V_{his}$ **do**

    foreach $w \in W_{ava}$ **do**

      Calculate $\mathbb{E}(v_i, w) = v_i * (1 - Pr(v_i, w))$

    $\mathbb{E}(v_i) = \max \mathbb{E}(v_i, w)$

  foreach $v_i \in V_{his}$ **do**

    Calculate

    $$Pr[v = v_i] = \frac{e^{\epsilon_2 * \mathbb{E}(v_i)/2(\max v/\min |His_w|)}}{\sum_{j=0}^{||V_{his}||} e^{\epsilon_2 * \mathbb{E}(v_j)/2(\max v/\min |His_w|)}}$$

  select $v_i^{'}$ based on the probability distribution of v

  insert $(v_i^{'}, p_{cop})$ into $V_r^{'}$

return $V_r^{'}$

3, the complexity of local matching process is $O(|W|)$ which is the same as D-PCOM. The complexity of cooperative matching process is $O((|P_{cop}| * |W_{p_{cop}}| * |V_{his}|) + \max |W_{cop}|)$, where $\max |W_{cop}|$ is the largest size of worker set in cooperative platform. The complexity of S-PCOM algorithm is $O(|R| * (|W| + (|P_{cop}| * |W_{p_{cop}}| * |V_{his}|) + \max |W_{cop}|))$. The space complexity is $O(|W| + |R|)$, where $|W|$ (resp. $|R|$) is the size of workers (resp. requests).

6 EXPERIMENTAL EVALUATION

This section presents the effectiveness, efficiency and scalability of our proposed PCOM algorithms by conducting a series of experiments on both real and synthetic datasets.

6.1 Experiment Setup

Real Datasets. The real dataset is collected by DiDi, Shenzhou and Yueche [1]. The real dataset contains the trajectory and the revenue of every request per day. We use the request trajectory data provided by the platform to calculate the data required by workers and requests. We choose 3 real datasets for experiments: the trajectory of Chengdu on 1st Nov. 2016 (denoted as DCN01 for DiDi, YCN01 for Yueche, and SCN01 for Shenzhou), those on 15th Nov. 2016 (denoted as DCN15, YCN15, and SCN15), and those of Xian on 30th Nov. 2016 (denoted as DXN30, YXN30, and SXN30). Table 2 shows the details of the real datasets where R is the number of requests, W is the number of workers and rad is the service radius of the workers.

Table 2: Real Datasets

	DiDi			Yueche			Shenzhou		
	DCN01	DCN15	DXN30	YCN01	YCN15	YXN30	SCN01	SCN15	SXN30
R	91321	100973	57611	90589	100448	57638	82331	89312	42134
W	9145	11199	2441	7038	9333	2686	5231	5481	1840
rad	1	1	1	1	1	1	1	1	1

Table 3: Synthetic Datasets

Factors	Setting
$\|R\|$	500,1000,**2500**,5k,10k,20k,50k
$\|W\|$	100, 200, 500, **1000**, 2500, 5k, 10k
Geo-I privacy level ϵ_1	0.1, 0.4, **0.7**, 1
Exponential mechanism privacy level ϵ_2	0.1, 0.4, **0.7**, 1

Synthetic Datasets. For the synthetic dataset, we randomly pick up the requests and workers from other dates in real datasets. We randomly select 500-50k requests and 100-10k workers, which forms a set consisting of 1000-100k requests and 200-20k workers in total. The location and arriving time are consistent with the real dataset. We also vary two privacy levels. We assume that the number of requests and the number of workers between different platforms are the same. Table 3 shows the settings of these synthetic datasets. We set the default value following the setting of existing work [10, 20], which is $|R|$ = 2500, $|W|$ = 500, ϵ_1=0.7, and ϵ_2=0.7.

Compared Algorithms. We compared our D-PCOM and S-PCOM with the state-of-art cross online matching algorithm [10], denoted as *RamCOM*. We also compared our algorithms with the online matching algorithm without cooperation [26], denoted as *TOTA*. 1 km is the service radius of most research at present [10, 24, 25]. To make a reasonable comparison with the existing research, we also set the service radius to 1 km. At the same time, 1 km is a reasonable setting in real life which workers can quickly arrive at this distance. We perform experiments on the threshold of S-PCOM in advance, and find that the algorithm works best when k=3. So the threshold is set to 3.

We compare the performance of the algorithms to show the effectiveness of our algorithms in terms of four metrics: (1) the total revenue of each platform, denoted as Rev_D, Rev_Y, and Rev_S; (2) the total number of served requests of each platform, denoted as $|M_D|$, $|M_Y|$, $|M_S|$; (3) the total number of completed cooperative requests, denoted as Cop_R; (4) The accepted ratio of the cooperative requests, denoted as AR. The average response time and memory cost show the efficiency of our algorithms. We test the scalability of our algorithms in terms of the total revenue, response time, and memory cost w.r.t. $|R|$, $|W|$, ϵ_1, ϵ_2 respectively. Our experiments are conducted on a machine with 16GB Memory, Intel(R) Core(TM) i7-9700 CPU @ 3.00GHz, with Windows 10 system, using C++ and its standard template library (STL).

6.2 Effectiveness

We compare our algorithms with the existing studies in terms of effectiveness. The experiments are conducted based on the real datasets, and the results are shown in Table 4-6. We evaluate our algorithms in terms of four metrics: Total Revenue, Total number

Table 4: The result on DCN01, YCN01 and SCN01

Methods	$Rev_D(\times10^6)$	$Rev_Y(\times10^6)$	$Rev_S(\times10^6)$	$\|M_D\|$	$\|M_D\|$	$\|M_S\|$	Cop_R	AR	Response Time (ms)	Memory (MB)
TOTA	1.632	1.28	1.031	64,812	61,236	50,368	-	-	0.63	48.7
RamCOM	1.661	1.52	1.231	68,672	65,235	52,349	72,054	0.667	1.21	48.7
D-PCOM	1.695	1.393	1.183	75,217	64,492	54,961	19,132	0.404	2.31	51.9
S-PCOM	1.658	1.48	1.195	66,707	69,210	50,209	71,158	0.613	12.02	52.4

Table 5: The result on DCN15, YCN15 and SCN15

Methods	$Rev_D(\times10^6)$	$Rev_Y(\times10^6)$	$Rev_S(\times10^6)$	$\|M_D\|$	$\|M_D\|$	$\|M_S\|$	Cop_R	AR	Response Time (ms)	Memory (MB)
TOTA	1.783	1.731	1.059	69,511	72,151	50,431	-	-	0.76	53
RamCOM	1.891	1.873	1.191	71,831	75,651	53,693	75,571	0.75	1.44	53
D-PCOM	1.888	1.824	1.162	78,841	75,266	53,171	24,741	0.449	2.39	54.2
S-PCOM	1.886	1.86	1.216	70,872	74,666	53,914	73,951	0.6	14.34	57.9

Table 6: The result on DXN30, YXN30 and SXN30

Methods	$Rev_D(\times10^6)$	$Rev_Y(\times10^6)$	$Rev_S(\times10^6)$	$\|M_D\|$	$\|M_D\|$	$\|M_S\|$	Cop_R	AR	Response Time (ms)	Memory (MB)
TOTA	0.512	0.509	0.421	22,420	22,134	16,453	-	-	0.45	19.6
RamCOM	0.587	0.661	0.513	24,391	24,097	18,104	15,562	0.351	0.95	19.6
D-PCOM	0.557	0.638	0.454	25,551	25,751	17,881	10,657	0.262	1.43	21
S-PCOM	0.613	0.658	0.475	23,766	24,337	17,950	14,840	0.321	2.15	24.3

of completed requests, Total number of Cooperative requests and Acceptance Ratio. Since the real dataset does not contain the privacy level, we set ϵ_1=0.7, ϵ_2=0.7 for two PCOM algorithms.

6.2.1 *Effectiveness w.r.t Total Revenue.*

The results show that our PCOM algorithms still maintain the effectiveness of cooperative matching under the condition of privacy protection. Take the result in Table 4 as an example. Comparing to *TOTA*, *D-PCOM* increases the total revenue of each platform by an average of \$17347 per day, while *S-PCOM* increases \$20501. Compared to *RanCOM* which does not consider data privacy, the PCOM algorithms only decrease \$4152 which is 1.8% of the total revenue. The effectiveness (w.r.t. total revenue) of PCOM algorithms has been proven.

6.2.2 *Effectiveness w.r.t Total Number of Served Requests.*

Compared to *TOTA*, PCOM algorithms both increase the total number of served requests. Since *D-PCOM* assigns every possible request to the local worker, the number of served requests is the largest. In most cases, the number of served requests of *S-PCOM* is smaller than that of *D-PCOM*. However, the total revenue of *S-PCOM* is larger than that of *D-PCOM* which means *S-PCOM* serves more high-value requests. The number of served requests in *S-PCOM* is smaller than that in *RamCOM* due to the perturbation of location. Some requests cannot be served based on their real location.

6.2.3 *Effectiveness w.r.t Total number of Cooperative Requests.*

Since *TOTA* algorithm solves the single-platform matching problem, it does not have cooperative requests. The total number of cooperative requests of *D-PCOM* is smaller than the algorithms with threshold since it considers the local matching process first. The number of cooperative requests in *S-PCOM* is smaller than that in *RamCOM* due to the perturbation of location.

6.2.4 *Effectiveness w.r.t Acceptance Ratio.*

TOTA algorithm solves the single-platform matching problem, it does not have acceptance ratio. The acceptance ratio of *D-PCOM* is smaller than *S-PCOM*. It means that considering the preference of available workers and the willingness of cooperation in the pricing process is useful.

6.3 Efficiency

We demonstrate the efficiency of the algorithm based on its response time and memory cost. The experimental results are shown in 4-6.

6.3.1 *Efficiency w.r.t Response Time.*

The response time of PCOM algorithms is larger than that of *TOTA* and *RamCOM*. Because the unit of response time is milliseconds, it can be tolerated in the real world. Therefore, the PCOM algorithms remain highly efficient.

6.3.2 *Efficiency w.r.t Memory Cost.*

The memory costs of these algorithms are almost the same. Since in the pricing process, the PCOM algorithms select the outer payment based on the historical data, they cost more memory. Based on the size of the given real dataset, such a small memory cost shows that the PCOM algorithms are cost-efficient.

6.4 Scalability

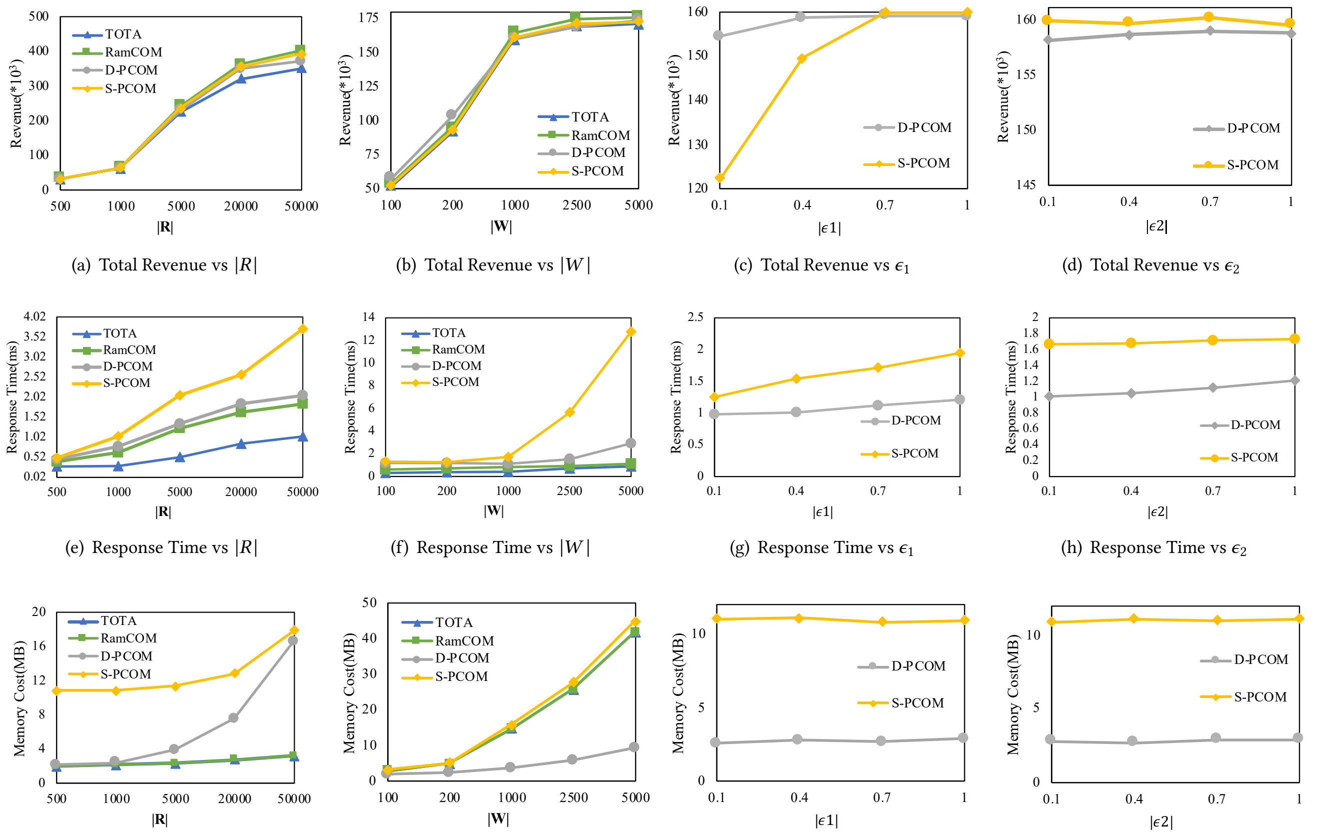
We test the scalability of our algorithms on synthetic datasets.

6.4.1 *Total revenue w.r.t $|R|$.*

Figure 4(a) shows the results of total revenue w.r.t the total number of requests $|R|$. For all algorithms, the total revenue increases with the increase of $|R|$ due to the increase of the completed requests. However, the magnitude of growth is not linear. Because as the number of requests grows, the number of workers will be insufficient.

6.4.2 *Total revenue w.r.t $|W|$.*

Figure 4(b) shows the results of total revenue w.r.t the total number of workers $|W|$. When $|W| < 1000$,



(a) Total Revenue vs $|R|$ (b) Total Revenue vs $|W|$ (c) Total Revenue vs ϵ_1 (d) Total Revenue vs ϵ_2

(e) Response Time vs $|R|$ (f) Response Time vs $|W|$ (g) Response Time vs ϵ_1 (h) Response Time vs ϵ_2

(i) Memory cost vs $|R|$ (j) Memory cost vs $|W|$ (k) Memory cost vs ϵ_1 (l) Memory cost vs ϵ_2

Figure 4: Scalability of Privacy-preserving Cooperative Online Matching Algorithms

the total revenue increases rapidly, since there are sufficient available workers. However, since the default number of requests is 2500, when $|W| > 1000$, all the requests can be served by local worker.

6.4.3 Total revenue w.r.t ϵ_1. Figure 4(c) shows the results of total revenue w.r.t the privacy level ϵ_1. The result shows that when $\epsilon_1 < 0.7$, the privacy mechanism has a significant impact on total revenue. Many cooperation requests are rejected in the fifth step of PCOM because of excessive perturbation of the geographic location. When $\epsilon_1 > 0.7$, the matching results gradually converge. *S-PCOM* performs a more cooperative process, so the perturbation of geographical location has a much greater impact than *D-PCOM*.

6.4.4 Total revenue w.r.t ϵ_2. Figure 4(d) shows the results of total revenue w.r.t the privacy level ϵ_2. The result shows ϵ_2 has little effect on the total revenue. That is because the result of the utility function in the privacy mechanism is similar. Although the privacy level has little effect on the results, it still guarantees that the data will not be attacked.

6.4.5 Response Time w.r.t $|R|$. Figure 4(e) shows the results of response time w.r.t the total number of requests $|R|$. The response time increases with the increase of $|R|$, since more requests should be assigned. The largest response time is smaller than 5ms and the increase of response time is almost linear with the exponential increase of $|R|$, which ensures the scalability of the algorithms.

6.4.6 Response Time w.r.t $|W|$. Figure 4(f) shows the results of response time w.r.t the total number of workers $|W|$. The result shows that the response time of *S-PCOM* is more sensitive since its complexity is related to $|W|$. The largest response time is smaller than 20ms, which ensures the scalability of the algorithms.

6.4.7 Response Time w.r.t ϵ_1 and ϵ_2. Figure 4(g) and 4(h) show the response time w.r.t two privacy level in PCOM algorithms. The result shows they both have little effect on the result. That is because the privacy level does not affect the complexity of the algorithm.

6.4.8 Memory cost w.r.t $|R|$, $|W|$, ϵ_1 and ϵ_2. Figure 4(i), 4(j), 4(k) and 4(l) show the memory costs of algorithms w.r.t. $|R|$, $|W|$, ϵ_1 and ϵ_2 respectively. It can be seen that as the number of workers and requests increases, the memory costs also increase. This is because the increased data needs more memory to store. However, privacy levels do not affect memory costs as they do not increase data.

6.5 Summary of Result

The experimental results show that the privacy level of location has a greater impact on the two PCOM algorithms because it directly affects the accuracy and completeness of matching. The privacy level of pricing has little effect on the two algorithms because the utility function produces similar results. The experimental results also show that when the degree of privacy is reasonably chosen (i.e. ϵ=0.7), S-PCOM performs similarly to the optimal cooperative matching algorithm that does not consider privacy.

7 RELATED WORK

Spatial Crowdsourcing Matching. The Spatial Crowdsourcing Matching problem is based on the task assignment. The main purpose of task assignment is to arrange workers and requests under specific objectives, while satisfying some spatial-temporal constrains. According to the arrival scenario, the task assignment approaches can be categorized into two types: static scenario and dynamic scenario. We mainly focus on the second scenario. In dynamic scenario, the algorithms are classified into two types, one-sided matching and two-sided matching. In one-side matching, only the information of one-side is unknown, while in two-sided matching, the information of both side is unknown. Approaches have been proposed to maximize the number of assignments [15–17, 28, 30]. In one-side matching, Jaillet et al. [16] apply linear program and obtain the best-known competitive ratio of 0.706. In two-side matching, Wang et al. [30] extend the charging-based framework to obtain a better ratio of 0.526. Tong et al. [28] apply the offline-guide-online method to obtain a ratio of 0.47. Approaches have also been proposed to maximize the total revenue [6, 21]. In one-side matching, Aggarwal et al. [6] propose a perturbed Greedy algorithm under adversarial order which achieves a competitive ratio of $1 - \frac{1}{e}$. In two-side matching, Ting et al. [21] propose a randomized algorithm Greedy-RT under adversarial order.

The task assignment algorithms all focus on different objectives on one single platform. When the distribution of requests and workers is non-uniform, some requests cannot be accepted. This situation will reduce the revenue of platform the satisfaction of users. In order to solve this problem, Cheng et al. [10] propose two Cross Online Matching algorithms which enable cooperation between platforms. Both of these algorithms can increase the platform's revenue and solve the problem of non-uniform distribution of requests and workers. However, these two algorithms do not consider the issue of data privacy between platforms.

The Incentive Mechanism Problem. The incentive mechanism problem determines the rewards to workers in order to motivate more workers to serve the request. The researches on this problem is generally divided into two models. First is supply-and-demand-aware model. In this model, the platform decides the reward according to the dynamic supply and demand in space and time. Chen et al. [9] use Markov decision process to determine the reward to workers. Tong et al. [27] aim to find the optimal pricing strategy based on bipartite graphs. The second model is auction-based model. In this model, workers can submit their expected reward, and the platform makes decision afterward. [8] applies first-price auction scheme in incentive mechanism for ride-sharing.

The studies above focus on the incentive mechanism on a single platform. These studies are not applicable due to the requirements of cross online matching problem. The incentive mechanisms in [10] effectively solve this problem. However, these mechanisms are calculated based on the historical request data of the other platforms, and the data of other platforms is completely exposed. The mechanisms in our work is privacy-preserved which can reasonably price the requests without exposing the data.

Privacy-Preserving Task Assignment in Spatial Crowdsourcing. In recent years, privacy plays an important role in spatial crowdsourcing. Most researches focus on the privacy of the location information of tasks and workers. Kazemi et al. [18] design a voting mechanism based on the spatial cloak of the workers. Recently, Differential Privacy [13] is widely used in privacy-preserving task assignment. To et al. [22] adopt the Private Spatial Decomposition approach [11] and devise a privacy mechanism to protect the count of workers in regions. However, these mechanisms cannot set the privacy on individual location. To solve this problem, Geo-Indistinguishability [7] is proposed as a formal notion of location privacy. Tong et al. [20] design a privacy mechanism based on Hierarchically Well-Separated Trees for online task assignment, and prove the competitive ratio of $O(\frac{1}{\epsilon^4} \log N \log^2 k)$ where ϵ is the privacy budget. The mechanisms above guarantee the location privacy between the platform and the worker (request). Using similar ideas, the mechanisms in our work guarantee the location privacy of the tasks between platform and platform.

8 CONCLUSION

In this paper, we propose Privacy-preserving Cooperative Online Matching, which protects sensitive data during the cooperative matching process. We also design a PCOM framework, and theoretically prove that it provides $((\epsilon_1 + \epsilon_2) * \max_{p \in P} |p_W|)$-differential privacy. Based on the framework, we propose two algorithms D-PCOM and S-PCOM with two privacy-preserving pricing mechanisms. Extensive experimental results over real and synthetic datasets show the effectiveness and efficiency of our algorithms.

In future work, researchers can further discuss more privacy mechanisms based on the PCOM framework. Similar to single-platform task matching, the cooperative matching algorithm that maximizes the benefits of each platform still needs to be studied.

ACKNOWLEDGMENTS

Yurong Cheng is supposed by the NSFC (Grant Nos.61902023, U18 11262, U21B2007). Ye Yuan is supported by the NSFC (Grant Nos. 61932004, 62225203, U21A20516). Guoren Wang is supported by the NSFC (Grant Nos. 61732003, U2001211). Yongjiao Sun is supported by the NSFC (Grant No. 61972077). Lei Chen's work is partially supported by National Key Research and Development Program of China Grant No. 2018AAA0101100, the Hong Kong RGC GRF Project 16202218, CRF Project C6030-18G, C1031-18G, C5026-18G, CRF C2004-21GF, AOE Project AoE/E603/18, RI Project R6020-19, Theme-based project TRS T41-603/20R, China NSFC No. 61729201, Guangdong Basic and Applied Basic Research Foundation 2019B151530001, Hong Kong ITC ITF grants ITS/044/18FX and ITS/470/18FX, Microsoft Research Asia Collaborative Research Grant, HKUST-NAVER/LINE AI Lab, Didi-HKUST joint research lab, HKUST-Webank joint research lab grants and HKUST Global Strategic Partnership Fund (2021 SJTU-HKUST). Yurong Cheng is the corresponding author.

REFERENCES

[1] 2022. DataSet. https://github.com/Yi107/Dataset-for-PCOM.git.
[2] 2022. DiDi. https://www.didiglobal.com/.
[3] 2022. Meituan. https://waimai.meituan.com.
[4] 2022. Uber. https://www.uber.com/.
[5] 2022. Uber Eats. https://www.ele.me/.
[6] Gagan Aggarwal, Gagan Goel, Chinmay Karande, and Aranyak Mehta. 2011. Online Vertex-Weighted Bipartite Matching and Single-bid Budgeted Allocations. In *Proceedings of the Twenty-Second Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2011, San Francisco, California, USA, January 23-25, 2011*, Dana Randall (Ed.). SIAM, 1253–1264. https://doi.org/10.1137/1.9781611973082.95
[7] Miguel E. Andrés, Nicolás Emilio Bordenabe, Konstantinos Chatzikokolakis, and Catuscia Palamidessi. 2013. Geo-indistinguishability: differential privacy for location-based systems. In *2013 ACM SIGSAC Conference on Computer and Communications Security, CCS'13, Berlin, Germany, November 4-8, 2013*, Ahmad-Reza Sadeghi, Virgil D. Gligor, and Moti Yung (Eds.). ACM, 901–914. https://doi.org/10.1145/2508859.2516735
[8] Mohammad Asghari, Dingxiong Deng, Cyrus Shahabi, Ugur Demiryurek, and Yaguang Li. 2016. Price-aware real-time ride-sharing at scale: an auction-based approach. In *Proceedings of the 24th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems, GIS 2016, Burlingame, California, USA, October 31 - November 3, 2016*, Siva Ravada, Mohammed Eunus Ali, Shawn D. Newsam, Matthias Renz, and Goce Trajcevski (Eds.). ACM, 3:1–3:10. https://doi.org/10.1145/2996913.2996974
[9] Mengjing Chen, Weiran Shen, Pingzhong Tang, and Song Zuo. 2018. Optimal Vehicle Dispatching for Ride-sharing Platforms via Dynamic Pricing. In *Companion of the The Web Conference 2018 on The Web Conference 2018, WWW 2018, Lyon , France, April 23-27, 2018*, Pierre-Antoine Champin, Fabien L. Gandon, Mounia Lalmas, and Panagiotis G. Ipeirotis (Eds.). ACM, 51–52. https://doi.org/10.1145/3184558.3186924
[10] Yurong Cheng, Boyang Li, Xiangmin Zhou, Ye Yuan, Guoren Wang, and Lei Chen. 2020. Real-Time Cross Online Matching in Spatial Crowdsourcing. In *36th IEEE International Conference on Data Engineering, ICDE 2020, Dallas, TX, USA, April 20-24, 2020*. IEEE, 1–12. https://doi.org/10.1109/ICDE48307.2020.00008
[11] Graham Cormode, Cecilia M. Procopiuc, Divesh Srivastava, Entong Shen, and Ting Yu. 2012. Differentially Private Spatial Decompositions. In *IEEE 28th International Conference on Data Engineering (ICDE 2012), Washington, DC, USA (Arlington, Virginia), 1-5 April, 2012*, Anastasios Kementsietsidis and Marcos Antonio Vaz Salles (Eds.). IEEE Computer Society, 20–31. https://doi.org/10.1109/ICDE.2012.16
[12] John P. Dickerson, Karthik Abinav Sankararaman, Aravind Srinivasan, and Pan Xu. 2018. Assigning Tasks to Workers based on Historical Data: Online Task Assignment with Two-sided Arrivals. In *Proceedings of the 17th International Conference on Autonomous Agents and MultiAgent Systems, AAMAS 2018, Stockholm, Sweden, July 10-15, 2018*, Elisabeth André, Sven Koenig, Mehdi Dastani, and Gita Sukthankar (Eds.). International Foundation for Autonomous Agents and Multiagent Systems Richland, SC, USA / ACM, 318–326. http://dl.acm.org/citation.cfm?id=3237435
[13] Cynthia Dwork. 2006. Differential Privacy. In *Automata, Languages and Programming, 33rd International Colloquium, ICALP 2006, Venice, Italy, July 10-14, 2006, Proceedings, Part II (Lecture Notes in Computer Science)*, Michele Bugliesi, Bart Preneel, Vladimiro Sassone, and Ingo Wegener (Eds.), Vol. 4052. Springer, 1–12. https://doi.org/10.1007/11787006_1
[14] Cynthia Dwork. 2008. Differential Privacy: A Survey of Results. In *Theory and Applications of Models of Computation, 5th International Conference, TAMC 2008, Xi'an, China, April 25-29, 2008. Proceedings (Lecture Notes in Computer Science)*, Manindra Agrawal, Ding-Zhu Du, Zhenhua Duan, and Angsheng Li (Eds.), Vol. 4978. Springer, 1–19. https://doi.org/10.1007/978-3-540-79228-4_1
[15] Jon Feldman, Aranyak Mehta, Vahab S. Mirrokni, and S. Muthukrishnan. 2009. Online Stochastic Matching: Beating 1-1/e. In *50th Annual IEEE Symposium on Foundations of Computer Science, FOCS 2009, October 25-27, 2009, Atlanta, Georgia, USA*. IEEE Computer Society, 117–126. https://doi.org/10.1109/FOCS.2009.72
[16] Patrick Jaillet and Xin Lu. 2014. Online Stochastic Matching: New Algorithms with Better Bounds. *Math. Oper. Res.* 39, 3 (2014), 624–646. https://doi.org/10.1287/moor.2013.0621
[17] Richard M. Karp, Umesh V. Vazirani, and Vijay V. Vazirani. 1990. An Optimal Algorithm for On-line Bipartite Matching. In *Proceedings of the 22nd Annual ACM Symposium on Theory of Computing, May 13-17, 1990, Baltimore, Maryland, USA*, Harriet Ortiz (Ed.). ACM, 352–358. https://doi.org/10.1145/100216.100262
[18] Leyla Kazemi and Cyrus Shahabi. 2011. A privacy-aware framework for participatory sensing. *SIGKDD Explor.* 13, 1 (2011), 43–51. https://doi.org/10.1145/2031331.2031337
[19] Frank McSherry and Kunal Talwar. 2007. Mechanism Design via Differential Privacy. In *48th Annual IEEE Symposium on Foundations of Computer Science (FOCS 2007), October 20-23, 2007, Providence, RI, USA, Proceedings*. IEEE Computer Society, 94–103. https://doi.org/10.1109/FOCS.2007.41
[20] Qian Tao, Yongxin Tong, Zimu Zhou, Yexuan Shi, Lei Chen, and Ke Xu. 2020. Differentially Private Online Task Assignment in Spatial Crowdsourcing: A Tree-based Approach. In *36th IEEE International Conference on Data Engineering, ICDE 2020, Dallas, TX, USA, April 20-24, 2020*. IEEE, 517–528. https://doi.org/10.1109/ICDE48307.2020.00051
[21] Hing-Fung Ting and Xiangzhong Xiang. 2015. Near optimal algorithms for online maximum edge-weighted b-matching and two-sided vertex-weighted b-matching. *Theor. Comput. Sci.* 607 (2015), 247–256. https://doi.org/10.1016/j.tcs.2015.05.032
[22] Hien To, Gabriel Ghinita, and Cyrus Shahabi. 2014. A Framework for Protecting Worker Location Privacy in Spatial Crowdsourcing. *Proc. VLDB Endow.* 7, 10 (2014), 919–930. https://doi.org/10.14778/2732951.2732966
[23] Hien To, Cyrus Shahabi, and Li Xiong. 2018. Privacy-Preserving Online Task Assignment in Spatial Crowdsourcing with Untrusted Server. In *34th IEEE International Conference on Data Engineering, ICDE 2018, Paris, France, April 16-19, 2018*. IEEE Computer Society, 833–844. https://doi.org/10.1109/ICDE.2018.00080
[24] Yongxin Tong, Lei Chen, Zimu Zhou, H. V. Jagadish, Lidan Shou, and Weifeng Lv. 2019. SLADE: A Smart Large-Scale Task Decomposer in Crowdsourcing. In *35th IEEE International Conference on Data Engineering, ICDE 2019, Macao, China, April 8-11, 2019*. IEEE, 2133–2134. https://doi.org/10.1109/ICDE.2019.00261
[25] Yongxin Tong, Jieying She, Bolin Ding, Lei Chen, Tianyu Wo, and Ke Xu. 2016. Online Minimum Matching in Real-Time Spatial Data: Experiments and Analysis. *Proc. VLDB Endow.* 9, 12 (2016), 1053–1064. https://doi.org/10.14778/2994509.2994523
[26] Yongxin Tong, Jieying She, Bolin Ding, Libin Wang, and Lei Chen. 2016. Online mobile Micro-Task Allocation in spatial crowdsourcing. In *32nd IEEE International Conference on Data Engineering, ICDE 2016, Helsinki, Finland, May 16-20, 2016*. IEEE Computer Society, 49–60. https://doi.org/10.1109/ICDE.2016.7498228
[27] Yongxin Tong, Libin Wang, Zimu Zhou, Lei Chen, Bowen Du, and Jieping Ye. 2018. Dynamic Pricing in Spatial Crowdsourcing: A Matching-Based Approach. In *Proceedings of the 2018 International Conference on Management of Data, SIGMOD Conference 2018, Houston, TX, USA, June 10-15, 2018*, Gautam Das, Christopher M. Jermaine, and Philip A. Bernstein (Eds.). ACM, 773–788. https://doi.org/10.1145/3183713.3196929
[28] Yongxin Tong, Libin Wang, Zimu Zhou, Bolin Ding, Lei Chen, Jieping Ye, and Ke Xu. 2017. Flexible Online Task Assignment in Real-Time Spatial Data. *Proc. VLDB Endow.* 10, 11 (2017), 1334–1345. https://doi.org/10.14778/3137628.3137643
[29] Yongxin Tong, Zimu Zhou, Yuxiang Zeng, Lei Chen, and Cyrus Shahabi. 2020. Spatial crowdsourcing: a survey. *VLDB J.* 29, 1 (2020), 217–250. https://doi.org/10.1007/s00778-019-00568-7
[30] Yajun Wang and Sam Chiu-wai Wong. 2015. Two-sided Online Bipartite Matching and Vertex Cover: Beating the Greedy Algorithm. In *Automata, Languages, and Programming - 42nd International Colloquium, ICALP 2015, Kyoto, Japan, July 6-10, 2015, Proceedings, Part I (Lecture Notes in Computer Science)*, Magnús M. Halldórsson, Kazuo Iwama, Naoki Kobayashi, and Bettina Speckmann (Eds.), Vol. 9134. Springer, 1070–1081. https://doi.org/10.1007/978-3-662-47672-7_87