Break the Data Barriers While Keeping Privacy: A Graph Differential Privacy Method

Yijing Li, *Student Member, IEEE*, Xiaofeng Tao, *Senior Member, IEEE*, Xuefei Zhang, *Member, IEEE*, Mingsi Wang, and Shuo Wang, *Member, IEEE*

***Abstract*—The booming development of Internet of Vehicles (IoV) has brought new vitality to the construction of intelligent transportation systems (ITS). At the same time, a huge amount of data has been generated due to the gradual development of IoV toward large scale, complex, and diversified. These data are owned by the companies that vehicles belonging to or service providers, such as taxi companies own taxi data. Due to interest and privacy considerations, data owners are not willing to share data, thus a serious data isolated island problem is created, which is detrimental to the development of ITS. Therefore, this article focuses on how to prevent privacy disclosure of vehicles while sharing vehicle data to improve the service. Considering the amount of interactive data and privacy disclosure during data release, vehicle data are abstracted from text form into a graph-structured data form. At the same time, graph differential privacy (DP) together with anonymity protection is proposed innovatively to firmly protect vehicle privacy. Moreover, to solve the high complexity of big data graph-structure transformation, an accelerated nodes and edges combined graph DP (ACGDP) algorithm is proposed. Based on the simulations of real-world data that combine electric and nonelectric taxies, it is verified that our proposed scheme has a tradeoff between information availability and privacy protection. With the graph DP processed data, our proposed scheme reduces the average wasted mileage for charging by 3.87% and achieves a 44.28% increase in drivers' income. Drivers' satisfaction of receiving orders and charging preference reaches 68% after the graph-structured data reuse.**

***Index Terms*—Data ownership management, data sharing, graph differential privacy (DP), Internet of Vehicles (IoV), privacy.**

I. Introduction

INTERNET of Vehicles (IoV) has attracted a lot of interest in the 5G and beyond 5G era since the vehicular network is one of the important Internet of Things (IoT) application scenarios [1], [2]. As the number of vehicles continues to grow and vehicular business expands, IoV is developed toward intelligence, large scale, and heterogeneous and has brought new vitality to the construction of intelligent transportation systems (ITS) [3]. As a result, IoV generates a large amount of data that contains a wealth of vehicle information.

On the one hand, the great amount of data has been a great challenge for storage, sharing, and data reliability [4]. On the other hand, these data are owned by the companies that vehicles belonging to or service providers, such as taxi companies own taxi data. Due to interest and privacy considerations, data owners are not willing to share data, thus a serious data isolated island problem is created [5], [6], which is detrimental to the development of ITS.

However, the development of ITS urgently calls for the sharing and efficient use of big data [3]. Due to the limitations of vehicle observation data, the improvement of single-vehicle intelligence will soon encounter the ceiling, which is difficult to meet the needs of future IoV with low delay and high security. The strong demand to break down data barriers has driven the industry to look for more secure and efficient data sharing solutions [7]. There are mainly three top demands in finding the solutions: 1) improve storage efficiency; 2) reduce sharing pressure; and 3) improve privacy protection.

First, when considering the purpose of improving storage efficiency and reducing sharing pressure, it is necessary to transfer the data structure [8]–[10]. Given the complexity and high redundancy of vehicle data (e.g., routes), storing and releasing data in a text form can take up too much storage space and put pressure on transfers [11]. Moreover, when taking privacy issues into account, storing and releasing vehicle data as full paths expose too much privacy and discard multidimensional parameters [12]. Therefore, the current mainstream practice is to abstract the plain text into a graph form, where the participating subjects are regarded as nodes and multidimensional information is characterized as the edge [13].

The graph data reuse application is generally adopted in social networks [14], [15], where users subscribe to their social media content on the social platforms. The service provider thus collects and maintains a huge amount of users' information. For some purposes such as gaining more interest, the service provider may anonymize the data and release data to adversary or third party users while erasing different characteristics to keep privacy according to different needs [16].

Manuscript received 14 October 2021; revised 27 December 2021; accepted 7 February 2022. Date of publication 15 February 2022; date of current version 20 February 2023. This work was supported in part by the National Natural Science Foundation of China under Grant 61932005 and Grant 61941105; in part by the 111 Project of China under Grant B16006; and in part by the National Youth Top-Notch Talent Support Program. *(Corresponding author: Xiaofeng Tao.)*

Yijing Li, Xiaofeng Tao, and Xuefei Zhang are with the National Engineering Laboratory for Mobile Network Technologies, Beijing University of Posts and Telecommunications, Beijing 100876, China (e-mail: liyijing@bupt.edu.cn; taoxf@bupt.edu.cn; zhangxuefei@bupt.edu.cn).

Mingsi Wang is with the School of Information and Communication Engineering, Beijing University of Posts and Telecommunications, Beijing 100876, China (e-mail: mingsiwang@bupt.edu.cn).

Shuo Wang is with the National Engineering Laboratory of Electric Vehicle and Collaborative Innovation Center for Electric Vehicle, Beijing Institute of Technology, Beijing 100081, China (e-mail: shuo.wang@bit.edu.cn).

Digital Object Identifier 10.1109/JIOT.2022.3151348

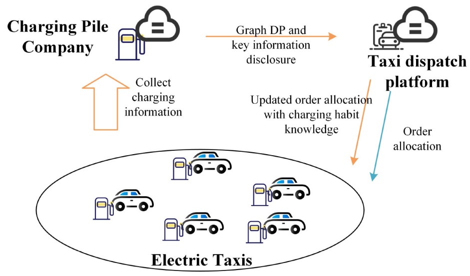


Fig. 1. Background of graph DP-based privacy-preserved data reuse: data sharing from the charging pile company to taxi dispatch platform.

For different uses, traditional anonymity methods may hide some labels, and the traditional way can achieve complete node anonymity at most, but in fact, the structure of the graph still exposes a lot of information. Assuming that the data reuse party has nonzero knowledge of any part of the graph, which is obviously a reasonable assumption, then the party may reverse deduce others' privacy according to the existing knowledge. Even if it is completely anonymous, the node with unique connection degrees can be easily identified, so the associated nodes can be easily inferred, let alone the information leaked by the unlabeled social network.

Therefore, it is particularly important to find a way to guarantee privacy, in addition, to anonymize when data are released, where differential privacy (DP) turns up in our view. DP is a mechanism proposed by Dwork [17] and is perfected in 2012 by Hardt *et al.* [18]. This mechanism ensures the probability of any output (i.e., anonymized data) equally likely from all nearly identical input data sets (i.e., original data), and thus, guarantees that all outputs are insensitive to any individual's data [17]. It is widely used in privacy protection since 2006, especially in query and data release field [19], [20]. Moreover, with the development of IoV, the DP adoption in the vehicular networks has also become a hot issue in recent years [21], [22].

However, the studies about DP in vehicular networks mainly pay attention to the local DP to protect the spontaneous information privacy in releasing [21], [22] while the passive privacy disclosure of a vehicle company or a service provider is ignored. Moreover, most of the studies ignore the fact that traditional vehicle data released in text or trace image forms bring a great challenge in data storage and data release costs.

Therefore, in this article, we focus on a scenario in which a charging pile company collects the charging information of electric cabs, does the graph DP and key information disclosure for data privacy protection and the data availability, and deliver it to the taxi dispatch platform for updating the order allocation with charging habit knowledge (Fig. 1). Our main contributions can be concluded as follows.

1) To the best of our knowledge, this article is the first study that focuses on graph DP mechanism-based privacy disclosure preserving mechanism for vehicle data. Different from the existed vehicle privacy protection studies, we not only provide high-dimensional privacy protection for vehicles but also reduce the stress of data storage and release costs.
2) Considering the complex vehicle data and the low-latency demands in IoV, an accelerated nodes and edges combined graph DP (ACGDP) algorithm is proposed to solve the high complexity brought by large-scale graph DP with both nodes and edges.
3) Simulation results verify that the proposed scheme has a tradeoff between information availability and privacy protection and provides guidance for practical application. In graph data reuse applications, the proposed scheme reduces the average distance for charging by 3.87% and achieves a 44.28% increase in drivers' income. Drivers' satisfaction of receiving orders and charging preference reaches 68% after the graph-structured data reuse.
4) Macroscopically speaking, this study provides guidance on how to efficiently protect vehicle privacy while achieving cross-platform and cross-service data distribution and sharing utilization.

II. Related Work

This article focuses on a scenario in which a charging pile company collects the charging information of electric taxis and delivers it to the taxi dispatch platform with the purpose of enabling more efficient vehicle dispatch while protecting privacy. In this research, DP is the main technique to achieve privacy preservation.

DP is a mechanism proposed by Dwork [17]. This mechanism ensures the probability of any output (i.e., anonymized data) equally likely from all nearly identical input data sets (i.e., original data), and thus, guarantees that all outputs are insensitive to any individual's data. By 2012, this mechanism was refined and proved to have the following four properties [18]: 1) sequential synthesis; 2) parallel synthesis; 3) transformation invariance; and 4) medium convexity. Therefore, it is recognized as an effective and flexible method to address privacy concerns, especially in the query and data release fields, and is widely used in privacy protection.

Graph DP is a special part of DP that adds noise to the graph-structure data [23], [24]. It is mainly divided into edge-differential scheme and node differential scheme, and the two are usually independent in previous studies. For example, Zhou and Pei [25] proposes an anonymization social approach to satisfy the k-anonymity requirement through graph networks, in which all nodes are clustered into distinct sets containing super points with special properties. Zhou and Pei [26] improved this approach by using k-anonymity and l-diversity DP to combat neighborhood attacks. Casas-Roma *et al.* [27] proposed a k-anonymity algorithm to improve the availability of data by retaining the most important edges. The anonymization method achieves privacy hiding of the original graph by clustering. However, macro features of the original graph structure can still be studied, which means that the original data are still at risk of being compromised by malicious attackers.

Recently, based on the previous work on graph DP protection privacy, more attention is paid to the balance of privacy protection and data utility. For example, research [28] aims at collecting and generating attribute social graphs in a decentralized manner to provide local DP (LDP) for the collected data. AsgLDP is also proposed as a new technique to generate privacy-preserving attribute graph data while satisfying LDP. The asgLDP preserves various graph attributes by carefully designing the joint distribution of injected noise and estimated attribute data. A random matrix approach for OSN data publishing is proposed in [29] to achieve storage and computational efficiency by reducing the dimensionality of the adjacency matrix. The authors validate the proposed approach using publicly available OSN graphs from Facebook, LiveJournal, and Pokec and evaluate the published data in node clustering, node ranking, and node classification in three different applications. Study [30] proposes a method for generating degree difference privacy graphs with field theory, and the correspondence between gravitational fields and field theory models in physics is established. Extensive experiments show that the method can retain more real social relationships than previous methods and does not produce loss of structural features, such as degree distributions and clustering coefficients on the data set.

With the development of IoV, the DP adoption in the vehicular networks has also become a hot issue in recent years [21], [22]. Research [31] proposes a novel privacy-preserving navigation solution, PiSim, which supports similar queries in navigation services, transforming the typical navigation approach into a traffic congestion query. The proposed PiSim scheme protects location privacy and route privacy and defends against multiple requests, false reports, and collusive attacks by malicious drivers. A new framework is proposed in [32] to sanitize fine-grained vehicle trajectories with DP (VTDP), which can provide strict privacy protection to adversaries with arbitrary background knowledge. Shi *et al.* [33] proposed a deep-reinforcement-learning-based route scheduling method that applies a DP-based geographic differentiation scheme to preserve the sensitive location information uploaded by passengers. In study [34], considering the location privacy issues in IoV architectures, the authors propose a novel framework for location privacy-preserving service usage based on the deployment of edge nodes, aiming to provide a tunable privacy-preserving scheme to balance utility and privacy.

However, the studies about DP in vehicular networks mainly pay attention to the local DP to protect the spontaneous information privacy in releasing while the passive privacy disclosure of a vehicle company or a service provider, such as taxi company. Moreover, most of the studies ignore the fact that traditional vehicle data released in text or trace image forms brings a great challenge in data storage and data release costs.

Therefore, we focus on a scenario in which a charging pile company collects the charging information of electric taxis, does the graph DP and key information disclosure for data privacy protection and the data availability, and delivers it to the taxi dispatch platform for updating the order allocation with charging habit knowledge.

III. System Model

As is introduced above, we consider privacy-preserved data sharing between the charging pile company and taxi dispatch platform, as is shown in Fig. 1. There are mainly three parts in the system: 1) electric taxis; 2) charging pile company; and 3) taxi dispatch platform.

1) *Electric Taxis:* Electric taxis, as the most basic part of the system, provide the charging data required by the system through the act of dispatching and receiving orders and charging in the city.
2) *Charging Pile Company:* Charging pile company, which provides charging services for electric taxies through charging piles throughout the city and collects relevant charging data, including charging locations, charging time, etc.
3) *Taxi Dispatch Platform:* Taxi dispatch platform provides taxi dispatch services to users in cities by vehicle location and order start and end locations. When the platform obtains the graph of electric taxi drivers' charging habits from the charging pile company, it abstracts new dimensional features to improve the efficiency of the dispatch service.

As is shown in Fig. 2, the whole process of our graph DP-based privacy-preserved data reuse is provided, which could help better understand our research. There are mainly two full data sets, one for multidimensional charging information extraction and graph-structure virtualization, another for order information extraction and order dispatching platform formulation. Based on graph DP methods, we add noises to the charging information graph for privacy preserving, then deliver the processed graph to the vehicle dispatching platform. The final purpose is to achieve privacy-preserved data sharing cross-industries while keeping data applicability.

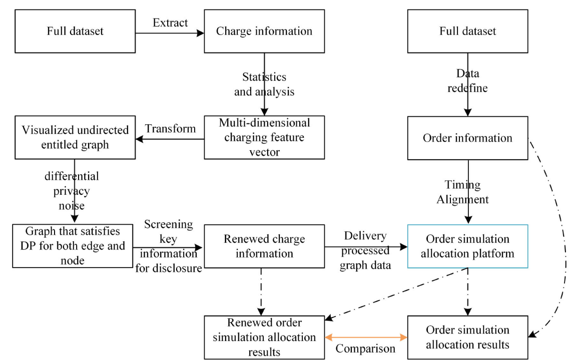


Fig. 2. Working flow of the graph DP-based privacy-preserved data reuse.

IV. Graph Differential Privacy in Charging Data Form Conversion

Fig. 2 presents the working flow of our research, in which our study can be divided into two parts: 1) the graph DP in charging data form the conversion process and 2) the dispatching platform update with the charging habit knowledge process. Here, we explain the former part in the following.

A. Problem Formulation

As shown in Fig. 4(a) and (b), we extract the information of each charging from the full data set of electric vehicles, including the start and end time of charging, the location of charging, and the change of electric quantity. We take the vehicle as the point, define the edge between vehicles with overlapping charging locations in the month, and form a graph as in Fig. 4(c).

The graph is defined as $G = (V, E)$, where $V = \{1, 2, \ldots, N\}$ are the set of nodes, which is also vehicles in the original graph G. $E = \{w_{i,j} \times \alpha_{i,j}\}, i, j \in \{1, 2, \ldots, N\}, i \neq j$ is regarded as the set of edges, where $w_{i,j} \in [0, 1]$ is the weight of edge between nodes i and j, and $\alpha_{i,j} \in \{0, 1\}$ defines the existence of edge between nodes i and j.

As for the calculation of weight $w_{i,j}$, we introduce a statistical vector of the charging preference vector to show the charging preference of vehicles. We count the charging times of each vehicle in the month as CT_i and divide the charging process into four states, the long-time charging LC_i, short-time charging SC_i, high quantity of electric charging HC_i, and low quantity of electric charging LoC_i, respectively. These four states and the charging times of each vehicle CT_i together form the charging preference vector

$$\vec{CP}_i = \left(\frac{\mathrm{LC}_i}{\mathrm{CT}_i}, \frac{\mathrm{SC}_i}{\mathrm{CT}_i}, \frac{\mathrm{HC}_i}{\mathrm{CT}_i}, \frac{\mathrm{LoC}_i}{\mathrm{CT}_i}\right), \quad i \in \{1, 2, \ldots, N\}. \tag{1}$$

The vector will further be used to determine the charging preference similarity between two vehicles. Since vehicles are regarded as nodes in the graph, the similarity is measured as edge weight between two nodes

$$w_{i,j} = \vec{CP}_i \times \vec{CP}_j\,. \tag{2}$$

A large weight indicates that the high similarity between the charging preference of the two vehicles.

Our main goal is to find a graph DP algorithm A and an adjacent graph $G' = (V', E')$ that satisfies the DP definition of

$$\frac{Pr[A(G) = O]}{Pr[A(G') = O]} \leq e^{\varepsilon} \tag{3}$$

which ensures the privacy of vehicles while keeping information availability.

Here, information availability is represented as the difference between the original graph G and the adjacent graph G', which can also be regarded as the information loss

$$|V \oplus V'| + |E \oplus E'| = \mathrm{diff}\big(G\&G'\big). \tag{4}$$

Moreover, we define the local sensitivity of node i as follows:

$$\sum_{j\in\{1,2,\ldots,N\}, i\neq j} |E_{i,j} \oplus E'_{i,j}| = LS(A)_i. \tag{5}$$

The global sensitivity is the maximum local sensitivity of the nodes in graph

$$GS(A) = \max_i LS(A)_i. \tag{6}$$

The objective function is to find a graph DP algorithm A and an adjacent graph $G' = (V', E')$ that satisfies (3) and the following equation through multiple attempts:

$$\min_A GS(A). \tag{7}$$

Algorithm 1 Algorithm of Simply Combined Nodes and Edges DP

Input: Original graph G, exponential noise parameter ε_1 and ε_2, laplace noise parameter ε_3
Output: Perturbation graphG'
1. Initialize Original graph $G = (V, E)$;
2. Add exponential noise with parameter ε_1 to $V = \{1, 2, \ldots, N\}$ and gain $(V1, E1)$;
3. Add exponential noise with parameter ε_2 to all $\alpha_{i,j}$ in $E1$ set and gain $(V1, E2)$;
4. Add laplace noise with parameter ε_3 to all $w_{i,j}$ in $E2$ set;
5. **repeat** Step 2 to 4 for K time to gain a graph G' satisfying equation (3) and (7);
return Perturbation graph G';

B. Algorithm

First, we will introduce the simple definition of exponential DP [35] and Laplace DP [36]. Since the basic schemes are already well studied in the former researches [17]–[24], [37], we only list the main information here for better understanding.

The exponential scheme is to realize DP under a nonstatistical query. The sensitivity of exponential DP is defined as follows:

$$\Delta q = \max_{D,D'} ||q(D, R_i) - q\big(D', R_i\big)||_1 \tag{8}$$

where $q(D, R_i)$ is the scoring function for gaining output R_i with data set D. The exponential mechanism outputs the result R_i with a probability of

$$\Pr[R_i] = \frac{\exp\left(\frac{\varepsilon q(D,R_i)}{2\Delta q}\right)}{\sum \exp\left(\frac{\varepsilon q(D,R_i)}{2\Delta q}\right)}. \tag{9}$$

Different from the exponential scheme, the Laplace scheme is a numerical mechanism. The definition of the Laplace mechanism can be described as follows:

$$M(D) = f(D) + Y \tag{10}$$

where $f(D)$ indicates the query function, Y indicates the Laplace random noise, and $M(D)$ indicates the final output. We define a Laplace noise $Y \sim L(0, (\Delta f/\varepsilon))$ satisfies $(\varepsilon, 0)$ DP, where Δf indicates sensitivity and ε indicates sensitivity budget. The sensitivity in the Laplace scheme indicates the maximum variation range of a query function $f(\cdot)$ for two adjacent data sets D and D'.

Due to the complexity of graph difference privacy with both nodes and edges, when looking for graphs that meet the epsilon and sensitivity settings, simply combining the exponential noise DP for points with the Laplace DP for edges will lead to long latency. Therefore, we propose an ACGDP algorithm method.

Fig. 3 shows the simply combined nodes and edges graph DP algorithm (Algorithm 1) and the accelerated nodes and edges graph DP algorithm (Algorithm 2). The scheme of the simply combined method is to separate nodes and edges for numerical and nonnumerical DP, respectively. In the case of

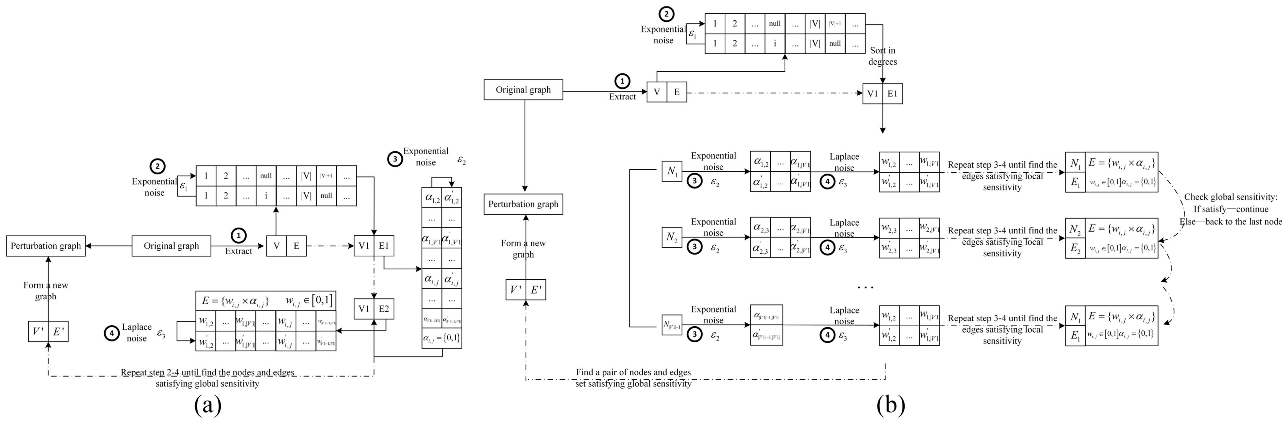


Fig. 3. Algorithm comparison of graph DP with both nodes and edges: (a) simply combined method and (b) ACGDP algorithm method.

Algorithm 2 ACGDP Algorithm

Input: Original graph G, exponential noise parameter ε_1 and ε_2, laplace noise parameter ε_3, local sensitivity threshold $ls(A)$, global sensitivity threshold $gs(A)$

Output: Perturbation graphG'

1. Initialize Original graph $G = (V, E)$;

2. Add exponential noise with parameter ε_1 to $V = \{1, 2, \ldots, N\}$, sort the nodes by degrees and gain $(V1, E1)$;

for nodes=1,2...$|V| - 1$ **do**

 3. Add exponential noise with parameter ε_2 to $\alpha_{i,j}$ of all edges that connected to node i;

 4. Add laplace noise with parameter ε_3 to $w_{i,j}$ of all edges that connected to node i;

 5. Check the local sensitivity of node i and the global sensitivity including nodes from 1 to i;

 if Satisfy $ls(A)$ and $gs(A)$ **then**

  turns to the next node $i + 1$;

 else

  turns to Step 3;

 end if

 until i equals $|V1|$

end for

return Perturbation graph G';

---

strict sensitivity constraints, this method is difficult and takes a long time to find a graph that meets the conditions.

Therefore, we propose an ACGDP algorithm. Both exponential DP and Laplace DP are adopted in this method, where exponential noise is used for nonnumerical nodes and edges existence and Laplace noise is used for numerical edge weights.

We first extract the node set and the edge set from the original graph and add exponential noise to the node set with a privacy budget ε_1, which includes the node delete and creation. The fictitious nodes will inherit the edge link relations of replaced nodes. Then, we sort the new node set with degrees and form a new node set and an edge set. In the next step, we add exponential noise with privacy budget ε_2 for edge link existence and Laplace noise with privacy budget ε_3 to edge weight from the node with most degrees. We have precalculated the mean local privacy sensitivity threshold $ls(A)$ for each node and checked the local sensitivity satisfactory. Only when a node satisfies the local sensitivity constraints, will the add noise process takes a step to the next node. When all nodes are finished for steps 3 and 4, we check whether the new graph satisfies the global sensitivity threshold. We repeat the whole process for times to gain a perturbation graph with the least global sensitivity under preset threshold $gs(A)$.

Assume a graph with N nodes, a node could connect to other nodes in 2^{N-1} different ways. That means a graph with N nodes could have 2^N neighbor graphs. When takes the edges difference into consideration, the whole graph has

$$2^{N-1} * 2^{N-2} * \cdots * 2^1 = 2^{\frac{N(N-1)}{2}} \tag{11}$$

possibilities of forms. Furthermore, when we consider a more complex situation when edges with weight $w_{i,j}$, which in our assumption has a range limited from 0 to 1, the possible forms are uncountable.

Due to the number of noise and the complexity of the graph structure, Algorithm 1 may take at most

$$T_{\text{step2}} \cdot T_{\text{step3}} \cdot T_{\text{step4}} = N \cdot 2^{\frac{N(N-1)}{2}} \cdot 2^{\frac{N(N-1)}{2}} Y^* \tag{12}$$

times to find a satisfied graph, where Y^* is possibility of classification according to different granularity in Laplace noise. And it takes at most

$$T_{\text{step2}} \cdot T_{\text{step3\&4}} = N \cdot 2^{\frac{(N+2)(N+1)}{2} \cdot Y^*} \tag{13}$$

times for Algorithm 2 to find such a satisfied graph. The advantage in complexity is caused by the periodic inspection of local sensitivity.

Due to the randomness of noise, add noise to each point and each edge complete the noise adding process of the whole graph and then make a limited comparison, it is normal to cause noise overflow (exceeding the global sensitivity budget) when the elements in the graph are not completely traversed. However, we set the average local limit in advance and check the local limit after adding noise to each point or edge, so as to avoid unnecessary noise adding attempts after noise overflow, so as to achieve the purpose of acceleration.

Moreover, as we have introduced in Section II, DP is proved to have the following four properties [18]: 1) sequential synthesis; 2) parallel synthesis; 3) transformation invariance; and 4) medium convexity. With these properties, we analyze the sensitivity budget of algorithm one as

$$\varepsilon[A_1] = \varepsilon\{V_1\} + \varepsilon\left\{\underbrace{\varepsilon\{E_1\}}_{E_2}\right\} = \max\{\varepsilon_1, (\varepsilon_2 + \varepsilon_3)\}. \quad (14)$$

The sensitivity budget of Algorithm 2 is analyzed as follows:

$$\begin{aligned}\varepsilon[A_2] &= \varepsilon_{N_1}\left\{\underbrace{\varepsilon\{V_1\}}_{G_1}\right\} + \varepsilon_{N_2}\left\{\underbrace{\varepsilon\{V_1\}}_{G_1}\right\} \\ &\quad + \cdots + \varepsilon_{N_{|V_1|}}\left\{\underbrace{\varepsilon\{V_1\}}_{G_1}\right\} \\ &= \varepsilon_1\varepsilon_2\varepsilon_3 \le \max\{\varepsilon_1, (\varepsilon_2 + \varepsilon_3)\} \\ &\quad \text{s.t. } \varepsilon_i \in [0, 1]. \end{aligned} \quad (15)$$

Therefore, we conclude that Algorithm 2 achieves advantages both in time complexity and sensitivity budget.

V. Dispatch Platform Update With Charging Preference Knowledge

A. Problem Formulation

In this section, we introduce the next step of our research, the dispatching platform update process with the charging habit knowledge. We will define some necessary parameters and statistics in the following. The graph-structure data recovery process is presented in Section V-B in the following.

First, we divide the time of the whole month into T^* time slots, and each time slot lasts M seconds. We sort the orders extracted from the full data set and define the order lists of time slot T as OL^T. Assume there are X orders in OL^T, the xth order has the attributes of order start location and order end location of $\mathrm{OL}^T_{x,\mathrm{start}}$ and $\mathrm{OL}^T_{x,\mathrm{end}}$, respectively. For vehicle i in N, the real-time location of vehicle is L^t_i. The distance for picking up customers of order x can be indicated as

$$\mathrm{Dis}_{\mathrm{pickup}} = ||L^t_i - \mathrm{OL}^{T(t)}_{x,\mathrm{start}}|| \quad (16)$$

and the order income as

$$\mathrm{Order}_{\mathrm{income}} = ||L^{T(t)}_{\mathrm{end}} - \mathrm{OL}^{T(t)}_{x,\mathrm{start}}|| \quad (17)$$

where $||\cdot||$ indicates the distance calculated by longitude and latitude and measured in the cross road. Assume vehicles run in a stable velocity v on the road, then the corresponding time consume can be calculated as

$$t_{\mathrm{pickup}} = ||L^t_i - \mathrm{OL}^{T(t)}_{x,\mathrm{start}}|| \quad (18)$$

$$t_{\mathrm{order}} = ||L^{T(t)}_{\mathrm{end}} - \mathrm{OL}^{T(t)}_{x,\mathrm{start}}||. \quad (19)$$

The real-time electric quantity is $\mathrm{EQ}^t_i \in [0, 100]$ and with a stable charging speed θ in every charging process. Since each vehicle has its own charging preference $CP_i^{\rightarrow} = ([\mathrm{LC}_i/\mathrm{CT}_i], [\mathrm{SC}_i/\mathrm{CT}_i], [\mathrm{HC}_i/\mathrm{CT}_i], [\mathrm{LoC}_i/\mathrm{CT}_i]), i \in \{1, 2, \ldots, N\}$, we assume that after finishing an order, vehicles will choose whether to charge or not based on the real-time electric quantity and the third and fourth element in the charging preference vector.

For charging actions, the charging distance can be indicated as

$$Dis_{\mathrm{charging}} = ||L^t_{\mathrm{charging}} - \mathrm{OL}^{T(t)}_{x,\mathrm{end}}|| \quad (20)$$

where L^t_{charging} is the location of the nearest charging pile. Once the vehicle makes the decision to charging, the charging time is depended on the first and second element in the charging preference vector, which are long-time charging and short-time charging, respectively.

After the dispatching process update for a month (in our simulation system), we gain the simulated statistical records of both orders receiving and charging and compare the simulated results with the original statistics to see the information availability of the graph DP process. Our goal is to maximize the drivers' income as well as maximize the charging preference satisfaction of drivers $|\vec{CP_i} \cdot \vec{CP_i}^{\mathrm{simu}}|$.

Algorithm 3 Graph Data Recovery Algorithm

Input: perturbation graph $G' = \left(V', E'\right)$, key node number adjustment parameter δ

Output: key node set S_{KN}, recovery graph G''

1. Classify the degree of each node of V', regard nodes with the same degree as a set;
2. Exclude the set with only one node to reduce the degree of privacy disclosure;
3. Find the potential key node candidates in Set_{D_k} (nodes of degree k);

for node i in Set_{D_k} **do**

 4. Calculate the shortest distance from each node $Path_{i,j}$ and the sum $SumP_i$;
 5. Note the reachable nodes of node i as set Re_i;

end for

6. Find the key node set S_{KN} that satisfies $\bigcup_{i \in S_{KN}} \mathrm{Re}_i = V'$ with the maximum $\sum_{i \in S_{KN}} SumP_i$;
7. Gain the real $\vec{CP_i}$ of S_{KN};
8. Recovery the best matched edge weight according to $CP_i^{\rightarrow}$ of S_{KN} and E';

return key node set S_{KN}, recovery graph G'';

B. Algorithm

In this section, we present the graph-structure data recovery process (Algorithm 3) and the updated vehicle dispatching algorithm with charging information (Algorithm 4) and simply introduce the pickup distance minimization-oriented comparison algorithm.

The process of graph data reuse can be summarized as follows: excluding sensitive nodes (in this case, the degree is unique) on the premise that the node with the highest recovery efficiency is selected as the key node for reuse, the real charging preference vector of the node can be obtained to infer the charging preference of connected nodes. At the same

Algorithm 4 Updated Vehicle Dispatching Algorithm With Charging Information

Input: Order list OL^T, location L_i^t

Output: Simulated charging preference vector $\vec{CP_i}^{simu}$, ave-dis-for-pickup, ave-dis-for-charging, sum-income

1. Initialize time t;
for vehicle=1,2...N **do**
  2. Calculate slot $T = \left[\frac{t}{M}\right] + 1$;
  for time slot=1,2...T^* **do**
    for Order=1,2...$|X|$ **do**
      3. Claculate Dis_{pickup} and the distance between the end point of the order and the nearest charging pile for each order;
    end for
    if One order has less than χ of the distance between the end point of the order and the nearest charging pile for each order **then**
      4. Choose this order and charging according to $\vec{CP_i}$ after order;
      5. Update t, T after charging;
    else
      6. Choose the order with minimized Dis_{pickup};
      7. Calculate $Order_{income}$, t_{pickup} and t_{order};
      8. Update t, T, and check EQ_i^t;
      9. Make the decision of whether charging or not according to $\vec{CP_i}$;
      if choose to charge **then**
        10. charging according to $\vec{CP_i}$;
        11. Update t, T after charging;
      else
        12. turns to Step 2;
      end if
    end if
  end for
end for
return Simulated charging preference vector $\vec{CP_i}^{simu}$, ave-dis-for-pickup, ave-dis-for-charging, sum-income;

time, it should be noted that the node here is not completely anonymous, but there is a one-to-one mapping based on a pseudonym, so as not to expose the vehicle ID. It is enough to optimize the dispatch one by one. The specific key node selection and the data recovery process are shown in the following Algorithm 3.

For the updated vehicle dispatching algorithm with charging information, we compare the distance between the endpoint of the order and the nearest charging pile at the endpoint. Due to the data set constraints and to visually view the comparison with-and-without charging preference knowledge, we have simplified the dispatch algorithm, where we consider a 24-h dispatch form and assume there are enough taxis in the system to reduce the popularity complexity. In our assumption, if it is less than the average mileage χ consumed to find the charging pile in the city, then select this order, otherwise select the order with the minimized pickup distance. After each order receiving, make a decision on whether to charging or not according to the current power and the driver's charging preference. If make the decision of charge, then vehicle charging according to the preference and the other order receiving methods remain unchanged.

As the name of the pickup distance minimization-oriented comparison algorithm indicates, the objective of the comparison algorithm is to minimize the distance between pick up customers each time in order receiving, and the vehicle only looks for a charging pile to charging when the power is less than 10. Since it is just a simple comparison algorithm, we do not show the algorithm in detail.

VI. Performance Analysis

To examine the credibility of the graph DP algorithm, the effect of privacy protection, and the amount of data information after application, we combine electric taxi driving data and nonelectric taxi driving data to simulate and analyze the data. The driving data of nonelectric taxis include the operation state, trigger event, longitude and latitude positioning, driving direction, and speed of about 200 vehicles in Beijing in November 2012. The operation states include loaded state, parking state, stop running state, and other states, from where we analyzed the order information of this month through the state transition record. The driving data of electric taxis include the complete driving path of 114 electric taxis in Beijing in June 2018. Run state, charging state, mileage, velocity, longitude, and latitude are recorded every 30 ms. Charging information is extracted in this data and the charging habits are analyzed based on the charging information.

These charging-related data in 2018 are still bound to the vehicle entity and have not been transferred, and the location in the 2012 data set is also in Beijing, where there is no order generation process from scratch. Combining the two data sets, we compare the order allocation results with-and-without charging habit to see the differences in drivers' satisfaction and order incomes. Referring to study [38], we consider the time gap will not affect the correctness of our conclusion.

A. Graph Differential Privacy in Charging Data Form Conversion

First, we present and analyze the simulation results in graph DP for the charging data privacy-preserve process. As in Section V-A, we define the four elements of the charging preference vector. By analyzing the charging distribution of charging times and electric quantities before charging, we divide the four states as follows.

We define the case that vehicle stays at the charging location for at least 10 min after fully charged as long-time charging, the case that difference of vehicle power before and after charging is less than 30 as short-time charging, the case that vehicle decides to charge when electric quantity is higher than 70 as a high quantity of electric charging, and the case that vehicle decides to charge when electric quantity is less the 10 as low quantity of electric charging.

Fig. 4 shows the examples of electric taxi data set full data and extracted charging information. For visibility, we only take the first 20 vehicles for visual analysis. First, we make statistics on the charging locations of 20 vehicles in the month.

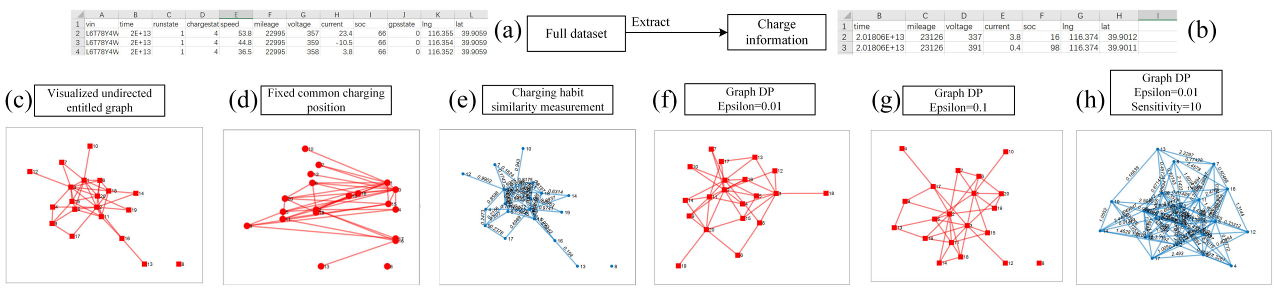


Fig. 4. Results of graph DP in charging data form conversion. (a) Full data set of driving log. (b) Charging information extracted from a full data set. (c) Visualized undirected graph of charging location preferences of 20 vehicles. (d) Visualized undirected graph of charging location preferences of 20 vehicles with fixed common charging location. (e) Charging habits similarity measurements of the 20 vehicles. (f) Graph DP of exponential noise with epsilon = 0.01. (g) Graph DP of exponential noise with epsilon = 0.1. (h) Graph DP of both exponential noise and Laplace noise with epsilon = 0.01 and a global sensitivity limitation of 10.

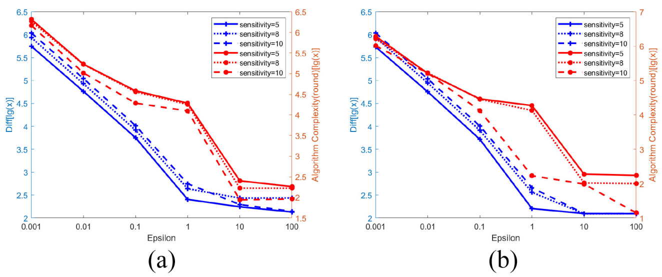


Fig. 5. Graph information loss and algorithm complexity under different sensitivity limitation. (a) Exponential noise only. (b) Both exponential noise and Laplace noise.

Vehicles are presented as nodes and two vehicles charging at the same charging pile are regarded as a social relationship, which we connect with the edge in the figure as in Fig. 4(c). Then, we fix each vehicle, that is, each node, at the most commonly used charging pile location to get Fig. 4(d). Charging habits are divided into four dimensions: long-time charging and short-time charging preferences, and high-power charging and low-power charging preferences. Each vehicle forms a vector with a length of 4 and calculates the vector distance between each other to measure the similarity of charging habits as in Fig. 4(e). Fig. 4(f) presents the graph DP of exponential noise with epsilon = 0.01, while Fig. 4(g) presents the graph DP of exponential noise with epsilon = 0.1. In Fig. 4(h), the results of graph DP of both exponential noise and Laplace noise with epsilon = 0.01 and a global sensitivity limitation of 10 are shown.

It can be seen from the figure that after adding DP, when epsilon is equal to 0.1, it is quite different from the original figure, indicating that data availability is reduced, but privacy protection is strengthened. When epsilon is equal to 1, it is the opposite. This result is consistent with the nature of DP itself, which verifies the correctness of our method. In fact, there is a tradeoff between data availability and privacy protection. The specific settings in the actual scenario depend on the final application requirements.

Then, we simulate the data difference comparison between the original graph and the graph after graph DP processing, that is, the comparison of data availability, and the algorithm complexity of finding a qualified adjacency graph under the constraints of different sensitivity in Fig. 5. For fairness, the algorithm complexity here is expressed by the number of algorithm cycles. Fig. 5(a) presents the graph differences and algorithm under graph DP with exponential noise only, while Fig. 5(b) presents the performance under graph DP with both exponential noise and Laplace noise.

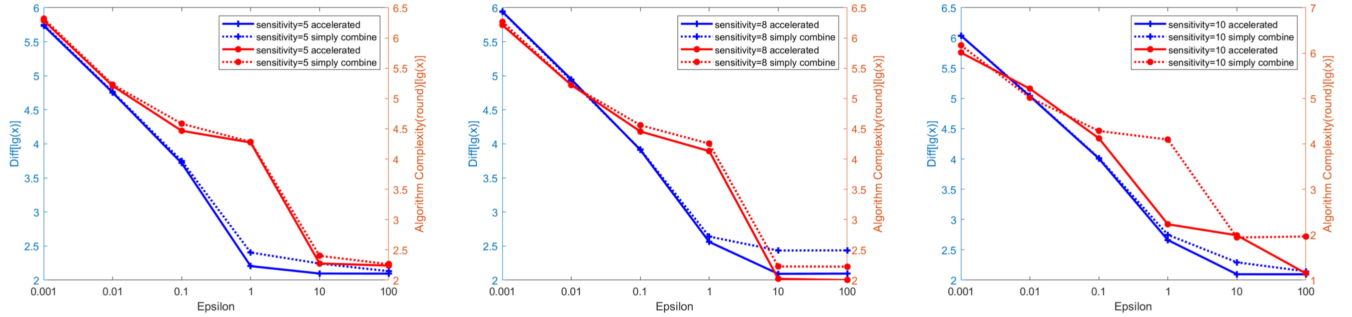
It can be seen that with the increase of sensitivity, that is, the relaxation of data difference, it becomes easier to find algorithms that meet the requirements and are more suitable for situations with high experimental requirements, but at the same time, the data availability will also be reduced, resulting in the loss of information. This is also a tradeoff problem that should be adjusted according to application requirements.

Then we compare the simply combined graph DP with both nodes and edges with the accelerated algorithm (ACGDP) and compare the data availability and algorithm complexity under sensitivities 5, 8, and 10. It can be seen that the ACGDP algorithm always maintains the approximate information loss with the simply combined algorithm and can effectively accelerate the graph DP process, especially when the epsilon is large. This is mainly because we do not start from global sensitivity to find the adjacency graph that meets the requirements, but after sorting the degrees of points, we gradually try to build the adjacency graph from the perspective of local sensitivity, avoiding many unnecessary graph attempts. The following is the result of comparing the performance and running speed of the two algorithms (Fig. 6).

B. Dispatching Platform Update With Charging Habit Knowledge

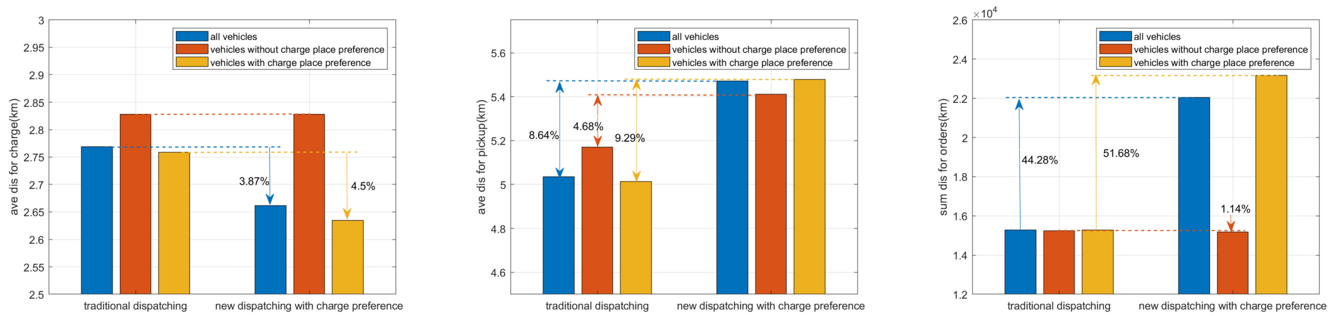
What we have discussed above are the phenomena when the charging pile company performs graph difference processing on the data. In the following, we will discuss some comparisons of the vehicle dispatching platform between the situation before and after the vehicle dispatching platform obtains the vehicle charging graph data.

First, we divide the vehicles into two categories, one is vehicles with obvious charging location preference, and the other is vehicles without obvious location preference. According to the statistics, the proportion is about 22:3. We simulate the mileage consumed for charging and picking up customers before and after knowing the vehicle charging information as shown in Fig. 7. Here, the parameter epsilon is set as 0.1.



(a) (b) (c)

Fig. 6. Graph information loss and algorithm complexity comparison of simply combined graph DP and accelerated graph DP under different sensitivity limitation. (a) Results under limitation of sensitivity 5. (b) Results under limitation of sensitivity 8. (c) Results under limitation of sensitivity 10.



(a) (b) (c)

Fig. 7. Dispatching results of different vehicles under the traditional dispatch platform and the new dispatch platform with charging preference information. (a) Comparison of the average mileage consumed by vehicle charging in the month. (b) Average mileage consumed for picking up customers in the month. (c) Comparison of the total mileage of orders (income) in the month.

It can be seen in Fig. 7(a) that after learning the charging information, the mileage consumed by vehicles for charging decreased by 3.87% on average. This parameter does not increase among vehicles without location preference but decreases by 4.5% among vehicles with preference.

On the other hand, since our comparison algorithm is to minimize the current order receiving distance, it is inevitable that the average mileage consumption for picking up customers will be greater after learning the charging information, increasing by about 8.64% [Fig. 7(b)]. However, from the absolute value perspective, the average increase is less than 500 m, which we think is still acceptable.

Then, we simulate the total mileage of receiving orders as in Fig. 7(c). Here, the total mileage of receiving orders refers to the remaining mileage excluding the consumption of picking up customers and charging, which can be understood as the driver's income in this month. From the simulation results, it can be seen that the driver's income can be increased by 44.28%, which is more obvious in the case of vehicles with charging place preference, reaching 51.68%. The reason is that with the increase of charging times, the order receiving rate of long-distance orders becomes higher and more benefits are obtained. This result shows the feasibility of our proposed algorithm in maintaining the availability of information.

Taking drivers' income as an example, if we assume that the taxi dispatch platform has all detailed vehicle charging data, the income of all vehicles will be increased by about 51.03% compared with traditional dispatching. Here, we do not distinguish whether the vehicle has obvious charging location preference, because the comparison algorithm may produce virtual vehicle nodes, and the comparison is not accurate enough, so we do not characterize this comparison in the figure. It is obvious that through graph DP, we can effectively protect vehicle privacy and achieve almost the same data reuse effect.

Finally, we simulate the driver's satisfaction with the charging habit preference as in Fig. 8. We calculated the distance between the statistical charging preference vector and the vector obtained in the actual simulation. The large value of vector distance indicates the close relationship between the statistical charging preference and the rewind charging preference, which is, the high satisfaction of the drivers'. It can be seen that drivers' satisfaction of receiving orders and charging preference reaches 68% after the graph-structured data reuse. The high satisfaction is more obvious for drivers with charging location preference, where the proportion of small vector distance value is low, and the proportion of 0.25–0.75 is relatively large. Since it is difficult to define the drivers' satisfaction with receiving orders and charging without considering their preferences, therefore, no comparison is made here.

With the simulation device (a laptop with a processor i-8550U CPU 1.80 GHz 1.99 GHz) and around 120 vehicles, we take around 20 min to form a charging location-related graph on average, where the charging preference is characterized as

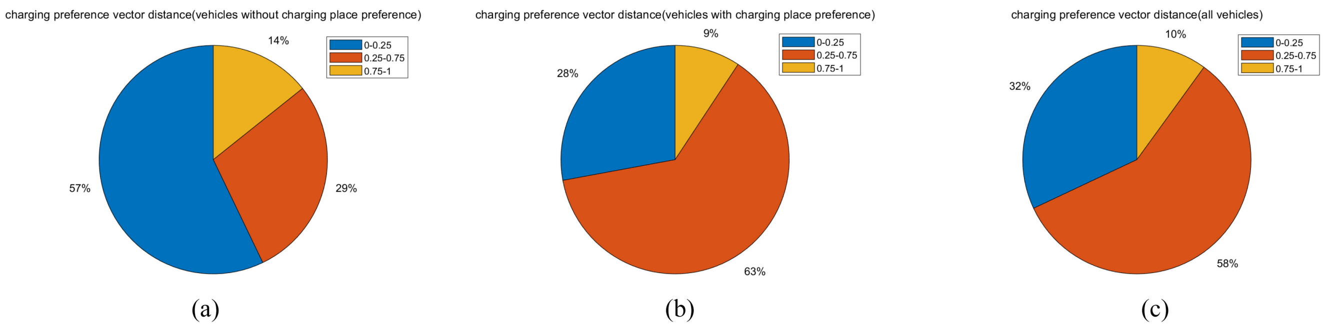


Fig. 8. Drivers' charge satisfaction under the new dispatch platform with charging preference information. (a) Vehicles without charging place preference. (b) Vehicles with obvious charging place preference. (c) All vehicles.

edge weight. And it takes us around 30 min on average to realize the data recovery and realize further updated vehicle dispatch with charging preference.

It seems a long latency to realize the whole graph DP in such a scenario. However, the scheme is realized from center (charging pile company) to center (taxi dispatch platform). Although the data amount of real application is thousands of times of our simulation data, the computing resource of those companies is also increased over great times to realize the same algorithm.

Moreover, the vehicle dispatch update is based on the charging preference of drivers, which is not instantaneous value but cumulative statistics. This means we do not have to collect the charging information and apply it to vehicle dispatch in real time. It requests a regular time period to finish the update, such as a week or a month.

All in all, the latency of our graph DP-based algorithm is quite reasonable in this scenario. The algorithm is feasible to be applied in real-world dispatch.

VII. Conclusion

This study is designed to solve the data isolated island problem in IoV. The presented study tries to apply graph DP with both nodes and edges for privacy preserving in the data sharing between the charging pile company and taxi dispatching platform. Innovatively, an ACGDP algorithm is proposed to solve the high complexity brought by large-scale graph DP with both nodes and edges. Simulation results based on real-world vehicle data show that our proposed scheme reduces the average distance for charging by 3.87% and achieves a 44.28% increase in drivers' income. In the next step of our research, more intelligent graph DP and vehicle dispatching algorithms will be studied, as better guidance for realistic application.

References

[1] W. Xu *et al.*, "Internet of Vehicles in big data era," *IEEE/CAA J. Autom. Sinica*, vol. 5, no. 1, pp. 19–35, Jan. 2018.

[2] F. Yang, S. Wang, J. Li, Z. Liu, and Q. Sun, "An overview of Internet of Vehicles," *China Commun.*, vol. 11, no. 10, pp. 1–15, Oct. 2014.

[3] Q. Cui *et al.*, "Big data analytics and network calculus enabling intelligent management of autonomous vehicles in a smart city," *IEEE Internet Things J.*, vol. 6, no. 2, pp. 2021–2034, Apr. 2019.

[4] L. Zhu, F. R. Yu, Y. Wang, B. Ning, and T. Tang, "Big data analytics in intelligent transportation systems: A survey," *IEEE Trans. Intell. Transp. Syst.*, vol. 20, no. 1, pp. 383–398, Jan. 2019.

[5] Y. Li, X. Tao, X. Zhang, J. Liu, and J. Xu, "Privacy-preserved federated learning for autonomous driving," *IEEE Trans. Intell. Transp. Syst.*, early access, Jun. 16, 2021, doi: 10.1109/TITS.2021.3081560.

[6] Y. Lu, X. Huang, Y. Dai, S. Maharjan, and Y. Zhang, "Blockchain and federated learning for privacy-preserved data sharing in Industrial IoT," *IEEE Trans. Ind. Informat.*, vol. 16, no. 6, pp. 4177–4186, Jun. 2020.

[7] X. Zheng *et al.*, "Big data for social transportation," *IEEE Trans. Intell. Transp. Syst.*, vol. 17, no. 3, pp. 620–630, Mar. 2016.

[8] E. Mehmood and T. Anees, "Challenges and solutions for processing real-time big data stream: A systematic literature review," *IEEE Access*, vol. 8, pp. 119123–119143, 2020.

[9] J. He, W. Chen, and Z. Tang, "STrans: A comprehensive framework for structure transformation," *Tsinghua Sci. Technol.*, vol. 21, no. 2, pp. 231–240, Apr. 2016.

[10] J. Lin and H. Zhang, "Data structure visualization on the Web," in *Proc. IEEE Int. Conf. Big Data (Big Data)*, Atlanta, GA, USA, 2020, pp. 3272–3279.

[11] C. Prehofer and S. Mehmood, "Big data architectures for vehicle data analysis," in *Proc. IEEE Int. Conf. Big Data (Big Data)*, Atlanta, GA, USA, 2020, pp. 3404–3412.

[12] D. Liu, "Big data analytics architecture for Internet-of-Vehicles based on the spark," in *Proc. Int. Conf. Intell. Transp. Big Data Smart City (ICITBS)*, Xiamen, China, 2018, pp. 13–16.

[13] P. W. Battaglia *et al.*, "Relational inductive biases, deep learning, and graph networks," 2018, *arXiv:1806.01261*.

[14] S. P. Singh and C. K. Leung, "A theoretical approach for discovery of friends from directed social graphs," in *Proc. IEEE/ACM Int. Conf. Adv. Soc. Netw. Anal. Min. (ASONAM)*, Hague, The Netherlands, 2020, pp. 697–701.

[15] M. Kashani, S. Gorgin, and S. V. Shojaedini, "Improvement of the recognition of relationships in social networks using complementary graph coloring based on cellular automata," in *Proc. 5th Conf. Knowl. Based Eng. Innov. (KBEI)*, Tehran, Iran, 2019, pp. 13–17.

[16] M. Rahman and R. Karim, "Comparative study of different methods of social network analysis and visualization," in *Proc. Int. Conf. Netw. Syst. Security (NSysS)*, Dhaka, Bangladesh, 2016, pp. 1–7.

[17] C. Dwork, "Differential privacy: A survey of results," in *Proc. Int. Conf. Theory Appl. Models Comput.*, 2008, pp. 1–19.

[18] M. Hardt, K. Ligett, and F. McSherry, "A simple and practical algorithm for differentially private data release," in *Advances in Neural Information Processing Systems*, vol. 25. Red Hook, NY, USA: Curran, 2012.

[19] M. Bi, Y. Wang, Z. Cai, and X. Tong, "A privacy-preserving mechanism based on local differential privacy in edge computing," *China Commun.*, vol. 17, no. 9, pp. 50–65, Sep. 2020.

[20] J. Soria-Comas, J. Domingo-Ferrer, D. Sánchez, and D. Megías, "Individual differential privacy: A utility-preserving formulation of differential privacy guarantees," *IEEE Trans. Inf. Forensics Security*, vol. 12, pp. 1418–1429, 2017.

[21] X. Zhang, J. Wang, H. Zhang, L. Li, M. Pan, and Z. Han, "Data-driven transportation network company vehicle scheduling with users' location differential privacy preservation," *IEEE Trans. Mobile Comput.*, early access, Jun. 22, 2021, doi: 10.1109/TMC.2021.3091148.

[22] X. Li *et al.*, "PAPU: Pseudonym swap with provable unlinkability based on differential privacy in VANETs," *IEEE Internet Things J.*, vol. 7, no. 12, pp. 11789–11802, Dec. 2020.

[23] S. Costea, M. Barbu, and R. Rughinis, "Qualitative analysis of differential privacy applied over graph structures," in *Proc. 11th RoEduNet Int. Conf.*, Sinaia, Romania, 2013, pp. 1–4.

[24] Q. Ye, H. Hu, M. H. Au, X. Meng, and X. Xiao, "LF-GDPR: A framework for estimating graph metrics with local differential privacy," *IEEE Trans. Knowl. Data Eng.*, early access, Dec. 24, 2020, doi: 10.1109/TKDE.2020.3047124.

[25] B. Zhou and J. Pei, "Preserving privacy in social networks against neighborhood attacks," in *Proc. IEEE 24th Int. Conf. Data Eng.*, Cancun, Mexico, 2008, pp. 506–515.

[26] B. Zhou and J. Pei, "The *k*-anonymity and *l*-diversity approaches for privacy preservation in social networks against neighborhood attacks," *Knowl. Inf. Syst.*, vol. 28, no. 1, pp. 47–77, 2011.

[27] J. Casas-Roma, J. Herrera-Joancomartí, and V. Torra, "*k*-degree anonymity and edge selection: Improving data utility in large networks," *Knowl. Inf. Syst.*, vol. 50, no. 2, pp. 447–474, 2017.

[28] C. Wei, S. Ji, C. Liu, W. Chen, and T. Wang, "AsgLDP: Collecting and generating decentralized attributed graphs with local differential privacy," *IEEE Trans. Inf. Forensics Security*, vol. 15, pp. 3239–3254, 2020.

[29] F. Ahmed, A. X. Liu, and R. Jin, "Publishing social network graph Eigenspectrum with privacy guarantees," *IEEE Trans. Netw. Sci. Eng.*, vol. 7, no. 2, pp. 892–906, Apr.–Jun. 2020.

[30] H. Zhu, X. Zuo, and M. Xie, "DP-FT: A differential privacy graph generation with field theory for social network data release," *IEEE Access*, vol. 7, pp. 164304–164319, 2019.

[31] M. Li, Y. Chen, S. Zheng, D. Hu, C. Lal, and M. Conti, "Privacy-preserving navigation supporting similar queries in vehicular networks," *IEEE Trans. Dependable Secure Comput.*, early access, Aug. 18, 2020, doi: 10.1109/TDSC.2020.3017534.

[32] B. Liu, S. Xie, H. Wang, Y. Hong, X. Ban, and M. Mohammady, "VTDP: Privately sanitizing fine-grained vehicle trajectory data with boosted utility," *IEEE Trans. Dependable Secure Comput.*, vol. 18, no. 6, pp. 2643–2657, Nov./Dec. 2021.

[33] D. Shi *et al.*, "Deep *Q* -network-based route scheduling for TNC vehicles with passengers' location differential privacy," *IEEE Internet Things J.*, vol. 6, no. 5, pp. 7681–7692, Oct. 2019.

[34] L. Zhou, L. Yu, S. Du, H. Zhu, and C. Chen, "Achieving differentially private location privacy in edge-assistant connected vehicles," *IEEE Internet Things J.*, vol. 6, no. 3, pp. 4472–4481, Jun. 2019.

[35] Z. Huang and S. Kannan, "The exponential mechanism for social welfare: Private, truthful, and nearly optimal," in *Proc. IEEE 53rd Annu. Symp. Found. Comput. Sci.*, New Brunswick, NJ, USA, 2012, pp. 140–149.

[36] Z. Jorgensen, T. Yu, and G. Cormode, "Conservative or liberal? personalized differential privacy," in *Proc. IEEE 31st Int. Conf. Data Eng.*, Seoul, South Korea, 2015, pp. 1023–1034.

[37] C. Dwork and A. Roth, "The algorithmic foundations of differential privacy," *Found. Trends Theor. Comput. Sci.*, vol. 9, nos. 3–4, pp. 211–407, 2014.

[38] Y. Xu, S. Çolak, E. C. Kara, S. J. Moura, and M. C. González, "Planning for electric vehicle needs by coupling charging profiles with urban mobility," *Nat. Energy*, vol. 3, no. 6, pp. 484–493, 2018.

Yijing Li (Student Member, IEEE) received the B.S. degree from Mathematics School, Shandong University, Jinan, China, in 2017. She is currently pursuing the Ph.D. degree in communications and information systems with Beijing University of Posts and Telecommunications, Beijing, China.

Her research interests are in the area of mobile-edge computing network, blockchain, machine-learning techniques, and intelligent transportation system.

Xiaofeng Tao (Senior Member, IEEE) received the bachelor's degree in electrical engineering from Xi'an Jiaotong University, Xi'an, China, in 1993, and the master's and Ph.D. degrees in telecommunication engineering from Beijing University of Posts and Telecommunications (BUPT), Beijing, China, in 1999 and 2002, respectively.

He is a Professor with BUPT. He has authored or coauthored over 200 papers and three books in wireless communication areas. He focuses on 5G/B5G research.

Prof. Tao is the Chair of the IEEE ComSoc Beijing Chapter and a Fellow of the Institution of Engineering and Technology.

Xuefei Zhang (Member, IEEE) received the B.S. and Ph.D. degrees in telecommunications engineering from Beijing University of Posts and Telecommunications (BUPT), Beijing, China, in 2010 and 2015, respectively.

From September 2013 to August 2014, she was visiting the School of Electrical and Information Engineering, University of Sydney, Sydney, NSW, Australia. She is currently with the National Engineering Laboratory, BUPT. Her research areas include mobile-edge computing, blockchain, reinforcement learning, and intelligent transportation system.

Mingsi Wang is currently pursuing the undergraduate degree in communication engineering with the School of Information and Communication Engineering, Beijing University of Posts and Telecommunications, Beijing, China.

Shuo Wang (Member, IEEE) received the B.S. degree in electrical engineering from the Shandong University of Science and Technology, Qingdao, China, in 2009, the M.S. degree in electrical engineering from China University of Mining and Technology, Beijing, China, in 2012, and the Ph.D. degree in engineering from the Faculty of Engineering and Information Technology, University of Technology Sydney, Ultimo NSW, Australia, in 2017.

He is an Assistant Professor with Beijing Institute of Technology, Beijing, where he was a Postdoctoral Research Fellow from 2017 to 2021. His research interests include wireless charging, electric vehicle, and its big data analysis.