

# Measuring Maximum Urban Capacity of Taxi-based Logistics

Yueyue Chen, Deke Guo, *Senior Member, IEEE*, Ming Xu, *Member, IEEE*, Guoming Tang, and Geyao Cheng

**Abstract**—City-wide package delivery becomes popular due to the dramatic rise of online shopping. In order to speed up the package delivery process without increasing the delivery cost, a promising system has been proposed, which leverages the crowdsourced taxis. Many efforts have been done on this novel system in recent literature. However, a fundamental problem still remains open, i.e., measuring the maximum capacity of taxi-based logistics at the urban scale. In this paper, we first propose an accurate and efficient measurement mechanism to tackle this problem in the Non-stop package delivery method. The basic idea is to construct a spatial-temporal graph according to the passenger demands and calculate the maximum urban capacity by combining the results of several carefully designed max-flow problems. Then, we expand our measurement mechanism to be used in other taxi-based package delivery methods after a few adaptations, including the One-hop method and the Stop-and-wait method. At last, we evaluate our measurement mechanism and compare the maximum urban capacity of various package delivery methods with a real-world dataset from an online taxi-taking platform.

**Index Terms**—Maximum capacity, logistics, crowdsourced taxis, urban mobility.

## I. INTRODUCTION

A promising alternative to the traditional logistics approaches has attracted a lot of attention, which leverages the crowdsourced taxis. By having packages take hitchhiking rides with existing taxis that are transporting passengers on the street, the delivery cost of packages can be reduced significantly since only small additional efforts and time are needed from the involved taxi drivers. Moreover, it does not induce extra air noise and pollutions and is also helpful to reduce traffic congestion. Thanks to the blossom of the online taxi-taking platforms, such as DiDi [1] and Uber [2], it becomes possible to globally schedule and optimize the taxi-based logistics without interrupting the passenger itineraries. Recently, many methods for such taxi-based logistics have been proposed, including the One-hop method [3], Non-stop method [4], and the Stop-and-wait method [5]. The details of these methods will be introduced in the next section.

However, a fundamental problem still remains open: measuring the maximum capacity of taxi-based logistics in a city.

Corresponding author: Deke Guo

Y. Chen is with the unit 61660 of PLA, M. Xu is with the college of computer, National University of Defense Technology, Changsha, China (e-mail: yueyuechen@nudt.edu.cn; xuming@nudt.edu.cn). D. Guo, G. Tang and G. Cheng are with Science and Technology on Information Systems Engineering Laboratory, National University of Defense Technology, Changsha, China; D. Guo is also with Tianjin Key Laboratory of Advanced Networking (TANK), School of Computer Science and Technology, College of Intelligence and Computing, Tianjin University, Tianjin, China (e-mail: dekeguo@nudt.edu.cn; gmtang@nudt.edu.cn; chenggeyao13@nudt.edu.cn).

Although various methods have been proposed for taxi-based logistics, the maximum capacity of such a promising logistics system has not been discussed; while previous literature only focuses on transportation modes and package route plannings. In order to tackle this problem, it is necessary to design efficient and accurate measurement mechanisms and algorithms to evaluate crowdsourced urban mobility. The former efficiency property is mandated by the need to cope with hundreds of thousands of passenger orders routinely occurring in a large city. The latter accuracy property determines the relevance of the model results to the real world. The importance of tackling this fundamental problem is in three folds: i) assess the advantage of the taxi-based logistics on traditional logistics approaches, based on which the financial, environmental, and other benefits can be further quantified; ii) fairly compare the different package delivery methods proposed for taxi-based logistics, and the results may be useful to the policymakers of a city; iii) help the logistics companies to distribute packages more reasonably according to the maximum capacity at a different time in a day.

In this paper, we solve the measurement problem of the maximum urban capacity of taxi-based logistics. To the best of our knowledge, this is the first effort to address this essential measuring problem at the urban scale for package delivery via crowdsourced taxis. Nevertheless, this problem is far more complicated than imagined to be solved. First, there is no unified definition for the maximum urban capacity of taxi-based logistics. The package delivery always spans a long time period, which brings confusion to the capacity definition and measurement. Second, a general measurement mechanism is expected for the maximum urban capacity, with which the upper bound of every proposed package delivery method can be calculated. In this way, fair comparisons between different taxi-based logistics methods can be conducted. Third, none of the existing taxi-based logistics guarantee the successful delivery of all packages, yet only the successfully delivered packages should be counted when measuring the maximum urban capacity. By tackling these challenges, the contributions of this paper are as follows:

- We propose the maximum urban capacity problem of taxi-based logistics. We define the maximum urban capacity as the maximum number of on-road packages among the city within a unit time<sup>1</sup>.

<sup>1</sup>We only consider the taxis (or private cars) registered on the online taxi-taking platforms when calculating the maximum urban capacity since no package delivery method has been proposed for the non-platformed taxis till now.

- We propose an efficient and accurate measurement mechanism for the maximum capacity problem based on the Non-stop package delivery method. Wherein, we first demonstrate the passenger demands through a spatial-temporal graph, then calculate the maximum urban capacity by combining the results of several carefully designed max-flow problems.
- We further expand our measurement mechanism to be used in other package delivery methods after a few adaptations, including the One-hop method and the Stop-and-wait method.
- We conduct evaluations based on a real-world dataset and compare the maximum urban capacity of different package delivery methods. We find that the capacity gap between the Non-stop method and the Stop-and-wait method is small, yet the One-hop method is far less than the other two methods.

The remainder of this paper is organized as follows. We introduce the background and related work in Section II. In Section III, we formally state our problem and propose our maximum capacity measurement mechanism based on the Non-stop package delivery method. We expand our measurement mechanism to the other two representative package delivery methods with a few adaptations in Section IV. We discuss evaluation results in Section V and conclude the paper in Section VI.

## II. BACKGROUND AND RELATED WORK

### A. Taxi-based logistics

Smart logistics has attracted much attention recently [6], [7], [8]. Taxi-based logistics [9] means the delivery of packages via crowdsourced taxis in a city. Unlike the traditional package delivery approaches, which rely on dedicated couriers, this promising alternative allows the passengers and the packages with similar itineraries and time schedules to share one vehicle. It takes the full advantage of ridesharing between the passengers and packages to reduce transportation congestions and air pollutions. Note that, in the taxi-based logistics, the package delivery is in a hitchhiking way, only the taxis with passengers inside can be used to transport packages.

In order to achieve the taxi-based package delivery in the real world, some practical assumptions are made in this paper and related papers. First, the information of passengers is not known to the platform until they make taxi-taking requests on the platform; the taxi drivers are requested to accept the passenger orders assigned by the platform. In fact, this assumption is already achieved on the online taxi-booking platform, such as DiDi [1] and Uber [2]. Second, taxi drivers are requested to deliver packages if they are selected. There is always spare space for the delivered package in the selected taxi, and the package delivery does not bring inconvenience to passengers. **In the taxi-based logistics, the delivery cost of a package can be reduced dramatically compared with traditional delivery methods [3].** Thus the delivery payments from customers can be spared to design proper incentive mechanisms for taxi drivers [10], [11]. Third, the traces of taxis are all trackable to make sure the security of the delivered packages.

### B. Three main package delivery methods in taxi-based logistics

In order to achieve taxi-based logistics and improve delivery efficiency, three main package delivery methods have been proposed in recent literature, i.e., the One-hop method, the Non-stop method, and the Stop-and-wait method.

In the One-hop package delivery method, it attempts to deliver a package in one shot [12], [3], [13]. If the delivery information, including the origin, destination, departure time, of a package and a passenger is similar to each other, they can share one taxi ride. For example, Walmart proposes to make use of its in-store customers to deliver goods to its online customers on their way home from the store [12]. Wang et al. [3] encourage a private car to change its regular route to a similar one which passes through the package pick up and drop off locations. An illustration of the One-hop package delivery method is shown in Fig. 1 (a). The dotted lines denote the paths of passengers, and the solid line denotes the path of a package. Such one-hop ridesharing systems provide little chance to be utilized in the city-wide package delivery since the similar itineraries with likely origin and destination are quite limited.

In the Non-stop package delivery method [4], a package is assigned to a selected taxi, which is requested to deliver the package all along. Meanwhile, the taxi is also able to transport one or more passengers successively until the package reaches its destination. During the whole process of package delivery, the package stays in the taxi without leaving, i.e., the package is non-stopped. An illustration of the Non-stop package delivery is shown in Fig. 1 (b). By strategically transporting the passenger orders (A, B, and C in the figure), the taxi successfully delivers the package to its destination. Thus, the drop off location and time of passenger A (B) are close to the pickup location and time of passenger B (C), the destination of passenger C is close to the package destination.

In the Stop-and-wait package delivery method [5], a package is delivered via multiple passenger orders (in another word, via multiple taxis). Thus, the origin of the package should be similar to the origin of the first passenger order, and the destination of the package should be similar to the destination of the last passenger order. In order to achieve this scheme and deliver packages successfully, many consignment warehouses are distributed among the target area. A taxi could transport a package if there is a passenger order between two warehouses. A package could stop in a warehouse and wait for another proper taxi before approaching the destination. Fig. 1 (c) illustrates an example of the Stop-and-wait package delivery, the package starts with the passenger order A, stops two times during the delivery process waiting to take passenger order B and C, at last, arrives at the destination.

## III. MAXIMUM URBAN CAPACITY MEASUREMENT FOR THE NON-STOP PACKAGE DELIVERY METHOD

In this section, we first state the maximum urban capacity problem of taxi-based logistics, then propose our measurement mechanism for this problem under the Non-stop package delivery method.



Fig. 1: Three main package delivery methods via crowdsourced taxis.

### A. Problem statement

The goal of this paper is to measure the maximum urban capacity of taxi-based logistics. However, how to define the maximum capacity has not been discussed in previous literature. The definition should be at the urban scale and reflect the temporal changes during the package delivery process. We refer to the maximum capacity problem in road network [14], [15], [16], which is usually defined as the maximum number of vehicles passing through the network in a unit time. Similarly, we define the maximum capacity of taxi-based logistics in this paper as the number of on-road delivering packages in a unit time, with the constraint that the packages will be delivered successfully.

Thus our maximum capacity problem is formulated as: *Measure the maximum number of on-road delivering packages, denoted as  $C$ , within a unit time through crowdsourced taxis, based on passenger demands and package requests. The constraint is the longest transportation time  $LTT$ , which indicates that the packages should be successfully delivered within  $LTT$  slots from the departure time.* The passenger demands include the passenger origin, destination, departure time, and arrival time of each passenger order. The package requests include a set of origins and destinations (abbr. ODs).

In order to calculate the maximum urban capacity of taxi-based logistics, we first construct the spatial-temporal graph for the given passenger demands, then present the measurement mechanism for the maximum capacity problem.

### B. Network model of passenger orders

In the taxi-based logistics, the package delivery highly rely on the passenger demands, since only the taxis with passengers inside are used to translate packages. Therefore, the passenger demands should be depicted as the basement to calculate the maximum capacity of taxi-based logistics. In order to depict the given passenger demands, we divide the target area (such as a city) into  $M$  blocks and split the 24 hours in a day into  $N$  slots (for example, 10 minutes a slot). For each block, we assign a representative location from which all other locations in this block can be reached in a given time, and let the location represents the block. To simplify the parameters, we denote  $b_i$  as the representative location of the  $i^{th}$  block in our following modeling and analysis. We define  $\delta(b_i, b_j|t)$  as the travel time departure from  $b_i$  at  $t^{th}$  slot and arrive at  $b_j$  with the shortest path, where  $1 \leq i, j \leq M$ ,  $1 \leq t \leq N$ . Note that, the slot length

is usually very short in our model, thus the travel time of different passenger orders from  $b_i$  to  $b_j$  departure from one slot is assumed to be the same.

Based on the above pre-processing, we construct the spatial-temporal graph  $G(V, E)$  based on the passenger demands, as shown in Fig. 2 (a). Each column represents the blocks in the same slot, each row represents the same block across different slots. An edge between two vertices  $n(b_i, t_k) \in V$  and  $n(b_j, t_g) \in V$  is constructed if  $t_g - t_k = \delta(b_i, b_j|t_k)$ , and the edge weight is the number of passenger orders departure from  $b_i$  to  $b_j$  starting at slot  $t_k$ , which is denoted as  $\Omega(n(b_i, t_k), n(b_j, t_g))$ . The edges are marked as solid lines in Fig. 2 (a).

Note that, we only consider the origin and destination instead of the whole trajectory of a passenger order. The reason is that the package delivery is processed through a hitchhiking way, i.e., passenger transportation cannot be interrupted. If we consider all the blocks in a whole trajectory of a passenger order and reflect them in the spatial-temporal graph, the passenger order may be interrupted to deliver two or more sequential packages. In the following content, we can see that the construction of the spatial-temporal graph makes it possible to transform our problem into multiple classical max-flow problems.

### C. The maximum capacity calculation with the same package departure time

As introduced above, we define the maximum urban capacity as the maximum number of on-road packages in the target unit time, while the counted packages will be delivered successfully within the given  $LTT$ . Suppose the target unit time is the  $k^{th}$  slot, to calculate the maximum urban capacity (denoted as  $C(k)$ ), we should count every package flowing through the target slot and make sure that the package can be delivered successfully in the future slots. In order to present our calculation process clearly, we first discuss a special problem, i.e., the maximum capacity calculation with the same package departure time.

Suppose all the packages have the same package departure time (denoted as  $PDT$  in following content). An example can be seen in Fig. 2 (b), the  $PDT$  is  $t_1$ , the package origins include  $b_1$  and  $b_2$ , the destinations include  $b_2$  and  $b_3$ , and the longest transportation time ( $LTT$ ) is 2 slots. By constructing the spatial-temporal graph of orders, the capacity measurement

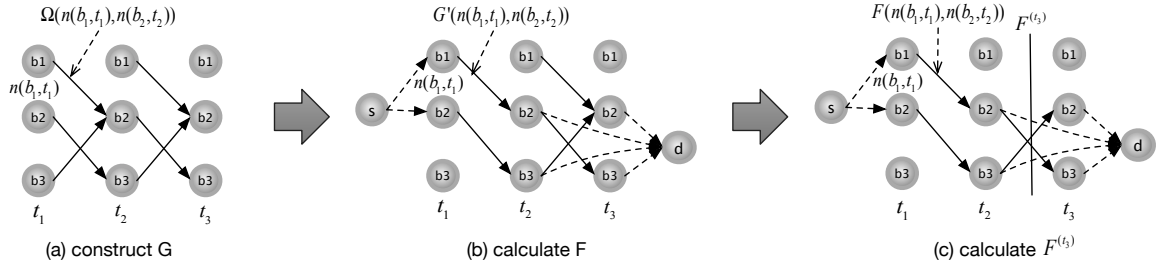


Fig. 2: One iteration of the capacity calculation for the Non-stop package delivery method

problem with a given *PDT* becomes the traditional max-flow problem. More specifically, it is a multi-source multi-sink max-flow problem. In Fig. 2 (b), the multiple sources include  $n(b_1, t_1)$  and  $n(b_2, t_1)$ , the multiple sinks include  $n(b_2, t_2)$ ,  $n(b_2, t_3)$ ,  $n(b_3, t_2)$ , and  $n(b_3, t_3)$ . It is worth noticing that, the sinks include all the destination blocks across the *LTT* away from the *PDT*. As shown in Fig. 2 (b), there are 4 sinks totally, although the destinations only include 2 blocks.

According to the analysis in [17], a multi-source multi-sink max-flow problem can be transformed into a single-source single-sink max-flow problem by adding a virtual source vertex linking all the sources and a virtual sink vertex linking all the sinks, and weight the added edges positive infinity. As shown in Fig. 2 (b), the added virtual source vertex is denoted as  $s$ , the virtual sink vertex is denoted as  $d$ , and the added edges are marked as dotted lines. A single-source single-sink max-flow problem can be solved efficiently by the Push-relabel algorithm [18], which is an improvement version of the Ford-Fulkerson algorithm.

#### D. The maximum capacity calculation with different package departure time

In the last subsection, we introduced the maximum capacity calculation with the same package departure time, which can be formulated as a max-flow problem. In the real-world, the package departure time can be very different. In order to calculate the urban capacity of taxi-based logistics, we need to count every possible package departure time. Thus, we propose to split the urban capacity calculation problem into multiple sub-problems. In each sub-problem, one possible package departure time is considered, and the resulted maximum capacity is calculated through solving a max-flow problem. At last, the needed urban capacity will be the conditional addition of the several sub-problem results. The addition process is introduced in detail in the following content.

In our formulation, all the packages must obey the time constraint and be delivered in *LTT* slots. Based on the constructed spatial-temporal graph and the given *LTT*, the possible departure slots of packages can be calculated easily, which are  $c(k-LTT)$ ,  $c(k-LTT+1)$ , ...,  $c(k-1)$ . Note that, the time takes circular values from hour 0 to 24, and then repeat. Thus, we define a function  $c(x \pm y)$  to handle this repeat.  $c(x+y) = (x+y) \bmod(N)$ ,  $c(x-y) = \begin{cases} x-y, & x-y \geq 0 \\ N-|x-y|, & x-y < 0 \end{cases}$ . Thus, we divide the maximum capacity problem into *LTT* sub-problems according to different departure times. In each

sub-problem, we count all the possible packages departure from the same slot and will flow through the  $k^{th}$  slot and be successfully delivered.

As analyzed above, the number of packages with a certain *PDT* can be calculated through the Push-relabel algorithm. Suppose the constructed spatial-temporal graph is denoted as  $G_t$  when the *PDT* is  $t$ , the resulted max-flow network is denoted as  $F_t$ , and the value of the max-flow is denoted as  $|F_t|$ . However,  $F_t$  cannot be directly deduced from  $G_t$  by using the Push-relabel algorithm. The reason is that, when calculating the max-flow under a given *PDT*, the spatial-temporal graph should be adjusted since some of the passenger orders have been utilized in the earlier slots. In this paper, the ridesharing between packages is not considered, thus, the passenger orders which have been utilized once before should not be utilized for another time. In order to tackle this problem, we construct another spatial-temporal graph for each *PDT*, and mark it as  $G'_t$  when *PDT*= $t$ . The calculation of  $G'_t$  excludes the utilized passenger orders in earlier slots by subtracting them in the related edge value  $\Omega$ . Thus, the  $G'_t$  can be constructed as  $G'_t = G_t - F_{c(t-1)} \cdots - F_{c(t-LTT)}$ .

Nevertheless, the above formula can make it fall into an infinite loop because the time is endless. The calculation of  $F_{c(t_1)}$  is based on  $G'_{c(t-1)}$ , and the calculation of  $G'_{c(t-1)}$  needs rollback another *LTT* slots. In this paper, we initialize it by calculating the first max-flow network based on an original spatial-temporal graph, denoted as  $G_{init}$ , instead of rolling back. In other words,  $G'_{init} = G_{init}$ . The *init* can be determined according to the actual situation of the applications, the farther the *init* away from the target slot, the more accurate the answer. We must admit that it would introduce some errors, we will explore this problem in our future work. Therefore, the calculation of  $G'_t$  can be rewritten as  $G'_t = G_t - F_{c(t-1)} \cdots - F_{\min\{c(t-LTT), init\}}$ . Thus,  $F_t$  is deduced through  $G'_t$  by adding virtual source and sink vertices and processing the Push-relabel algorithm.

Let  $|F_t^{(k)}|$  denotes the number of on-road packages which are generated at the  $t^{th}$  (i.e., *PDT*= $t$ ) slot and flow through the  $k^{th}$ . In order to calculate  $|F_t^{(k)}|$ , we need to analysis the max-flow network  $F_t$ . According to the Max-flow Min-cut theorem [17], the max-flow value  $|F_t|$  is equal to a random cut of the max-flow network  $F_t$ . A cut divides the vertices of the network  $F_t$  into two sets  $V_t^{(1)}$  and  $V_t^{(2)}$ , where  $s \in V_t^{(1)}$  and  $d \in V_t^{(2)}$ . In our problem, we request the set  $V_t^{(1)}$  includes the vertices

---

**Algorithm 1:** Maximum capacity calculation for Non-stop package delivery method

---

**Input:**  $LTT$ , passenger demands, package requests (ODs)

**Output:**  $C(k)$

```

1  $C(k) = 0$ ;
2 for  $t = (k - LTT) : 1 : (k - 1)$  do
3   Construct  $G_t$  based on the passenger demands;
4    $G'_t = G_t - F_{c(t-1)} \cdots - F_{\min\{c(t-LTT), \text{init}\}}$ ;
5   Add virtual source and sink vertices based on  $G'_t$ ;
6   Process the Push-relabel algorithm and get the  $F_t$ ;
7    $|F_t^{(k)}| = |F_t| - \sum_n (\Omega(n, d) |n \in V_t^{(1)} \& n \in D_t|)$ ;
8    $C(k) = C(k) + |F_t^{(k)}|$ ;
9 end

```

---

$\{s, \dots, n(b_1, k), \dots, n(b_M, k)\}$ , the set  $V_t^{(2)}$  includes the vertices  $\{n(b_1, c(k+1)), \dots, n(b_M, c(k+1)), \dots, d\}$ . Suppose the set of source vertices is denoted as  $S_t$ , and the set of sink vertices is denoted as  $D_t$ . Thus,  $|F_t^{(k)}|$  can be calculated as  $|F_t^{(k)}| = |F_t| - \sum_n (\Omega(n, d) |n \in V_t^{(1)} \& n \in D_t|)$ . An example is illustrated in Fig. 2 (c). Suppose the graph is the resulted max-flow graph, the cut (dedicated as the straight solid line) value  $|F_{t_1}|$  is the sum of  $F_{t_1}(n(b_2, t_2), n(b_3, t_3))$ ,  $F_{t_1}(n(b_2, t_2), d)$ ,  $F_{t_1}(n(b_3, t_2), n(b_2, t_3))$ , and  $F_{t_1}(n(b_3, t_2), d)$ , yet the value of  $|F_{t_1}^{(t_3)}|$  is the sum of  $F_{t_1}(n(b_2, t_2), n(b_3, t_3))$  and  $F_{t_1}(n(b_3, t_2), n(b_2, t_3))$ , since the dotted lines indicate the virtual edges.

**Theorem 1.** Suppose  $C(k)$  denotes the maximum number of on-road packages in the  $k^{\text{th}}$  slot with the constraint that the packages should be delivered successfully within the  $LTT$  slots. The  $C(k)$  can be calculated as:  $C(k) = |F_{c(k-1)}^{(k)}| + |F_{c(k-2)}^{(k)}| + \cdots + |F_{c(k-LTT)}^{(k)}|$ .

*Proof.* We prove it by reducing to absurdity. Suppose there is a package  $p$  to go through the  $k^{\text{th}}$  slot and then be delivered successfully, yet  $p \notin F_{c(k-i)} (1 \leq i \leq LTT)$ . The departure time, departure block and destination of the package are denoted as  $t_p, s_p, d_p$ , respectively.

If  $t_p < c(k-LTT)$ , it means the package is not delivered successfully within deadline, thus will not be counted. If  $t_p \geq k$ , it means the package is produced at or after the  $k^{\text{th}}$  slot, and it can not give pressure to the transportation at  $k^{\text{th}}$  slot. Both situations are contradicted with our assumption. Thus,  $c(k-LTT) \leq t_p \leq c(k-1)$ .

If the package is not included by  $F_{c(t_p)}$ , this means there is an extra package other than the max-flow departure at  $t_p$ . It indicates that the Push-relabel algorithm fails to get the max-flow, which is contradicted with the analysis in [17].

Therefore, it concludes that the package  $p$  is generated within  $c(k-LTT)$  to  $c(k-1)$ , and is included by  $F_{c(t_p)}$ . It is contradicted with the assumption  $p \notin F_{c(k-i)} (1 \leq i \leq LTT)$ . Thus theorem is proved.  $\square$

The pseudocode of our measurement mechanism can be seen in Algorithm 1. More specifically, we use an adjacency matrix to represent a graph. In order to

TABLE I: Meanings of the main Parameters

Parameter	Meaning
$LTT$	longest transportation time.
$PDT$	package departure time.
$G_t$	spatial-temporal graph of passenger orders while the PDT is $t$ .
$G'_t$	the residual graph of $G_t$ after subtracting historical max-flow.
$F_t$	the max-flow of $G'_t$ .
$ F_t^{(k)} $	the max value flowing through $k^{\text{th}}$ slot in $G'_t$ .
$C(k)$	the max capacity at $k^{\text{th}}$ slot.

characterize the matrixes, we set the vertices number of the graphs include  $G_{PDT}$ ,  $G'_{PDT}$ ,  $F_{PDT}$ , as  $M \times N \times LTT$ , starting from the given  $PDT$ . As a result, the time series between two adjacent matrixes are not aligned. However, in our calculation, the operating between two matrixes should be time aligned. Thus, the time series between matrixes should be treated carefully. For example,  $G_{i+j} - F_i \triangleq G_{i+j}[1:M \times c(LTT-j), 1:M \times c(LTT-j)] - F_i[M \times j:M \times LTT, M \times j:M \times LTT]$ . The computational complexity of our measurement mechanism for maximum urban capacity can be calculated and proofed as follows.

**Theorem 2.** The computational complexity of Algorithm 1 is  $O(M^2 \times N^2 \times LTT^3 \times N_{edge})$ .

*Proof.* The computational complexity of the Algorithm 1 is determined by the loop number and the Push-relabel algorithm. The complexity of the line 4 is  $O(1)$  if we calculate the max-flow of each  $PDT$  and store it for future use. According to [17], the complexity of the Push-relabel algorithm (line 6) is  $O(N_{vertex}^2 \times N_{edge})$ , where  $N_{vertex}$  is the number of vertices and  $N_{edge}$  is the number of edges. In our formulation,  $N_{vertex} = M \times N \times LTT$ . The complexity of line 7 is  $O(n)$ , where  $n \in V_t^{(1)} \& n \in D_t$ . The loop number is  $LTT$ . Therefore, the complexity of the whole measurement mechanism is  $O(LTT \times N_{vertex}^2 \times N_{edge})$ . Note that, the constructed spatial-temporal graph is very sparse in our problem.  $\square$

#### IV. MAXIMUM URBAN CAPACITY MEASUREMENT FOR THE ONE-HOP AND THE STOP-AND-WAIT PACKAGE DELIVERY METHOD

As introduced in Section II, there are three kinds of package delivery methods via the crowdsourced taxis within a city, i.e., the One-hop method, the Non-stop method, and the Stop-and-wait method. The goal of this paper is measuring the maximum urban capacity of taxi-based logistics, thus a general measuring mechanism is needed to suit all of the proposed package delivery methods. In the last section, we discuss the capacity measurement for the Non-stop package delivery. In this section, we expand our capacity measurement mechanism to different package delivery methods with a few adaptations.

For the One-hop package delivery, there is no need to trace back to the  $c(k-LTT)$  slot when we aim to calculate the  $C(k)$ . The calculation of the defined maximum urban capacity for the One-hop method is just counting the passenger orders which should satisfy three conditions: 1) going through the target slot, 2) satisfy package requests (i.e., from given origins

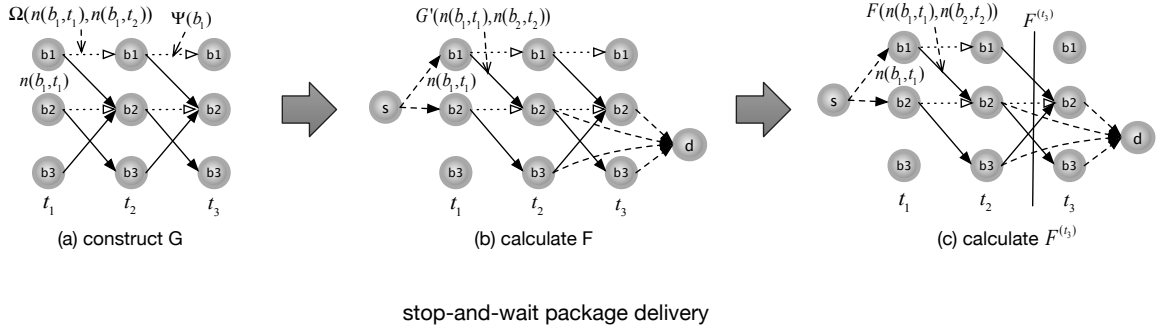


Fig. 3: One iteration of the capacity measurement for the Stop-and-wait package delivery method

to given destinations), 3) the travel time is less than the constraint  $LTT$ .

For the Stop-and-wait package delivery, the capacity calculation is similar to the Non-stop method. The major difference is that a package is allowed to wait in a warehouse for one or more slots in the Stop-and-wait method. In order to model this difference, an edge is added in the graph  $G$  between two vertices, which are the same block yet in different slots and there is a warehouse inside the block. The edge weight is the volume of the warehouse<sup>2</sup>. For instance, in Fig. 3 (a),  $b_1$  and  $b_2$  are chosen as the warehouse locations, the edge weight of  $G(n(b_1, t_2), n(b_1, t_3))$  is redefined as  $\Psi(b_1)$ , where  $\Psi(b_1)$  is the volume of the warehouse located in the  $b_1$  block. The added edges representing warehouses are denoted as dotted lines in Fig. 3 (a).

However, the adding of warehouse edges bring new problems in some special situations. For example, if a block is an origin, a destination, and also has a warehouse, our measurement mechanism will introduce redundancy. As illustrated in Fig. 3 (c), the block  $b_2$  is a multi-identified block. The flow  $f_{(s, n(b_2, t_1), n(b_2, t_2), n(b_2, t_3), d)}$  is obviously a theoretically legal flow in our constructed spatial-temporal graph, this will be included in the max-flow results  $F_{t_1}$  and  $|F_{t_1}^{t_3}|$ . However, this flow is illegal in reality, since the corresponding packages do not move at all. Therefore, this flow should be excluded from our computation results. The volume of the warehouse is  $\Psi(b_2)$ , so the maximum number of this kind of illegal flow is  $\Psi(b_2)$ . Therefore, when calculating the  $|F|$ ,  $\sum_i \Psi(b_i)$  should be subtracted from the final result. We find that in our formulation, it is unfair and unreasonable if a block is an origin and also has a warehouse, since a package can wait at the origin as long as possible, which can also be achieved in the packed place. Thus, we request that in our formulation, an origin block cannot be a warehouse block. This assumption can be achieved by refining the area division.

In summary, the capacity measurement for the One-hop package delivery method can be achieved through a selection process; the capacity measurement for the Stop-and-wait method can be achieved by adding the warehouse edges in the spatial-temporal graph and utilizing the same calculation process with the Non-stop method. The computation complex-

ity of the One-hop method is the number of the passenger orders  $N_{psg}$ , the computation complexity of the Stop-and-wait method is similar with that of the Non-stop method, the difference is in  $N_{edge}$ .

## V. PERFORMANCE EVALUATION

In this section, we first introduce the real-world dataset and parameter settings in our experiments, then evaluate the efficiency and effectiveness of our measurement mechanism for the maximum urban capacity.

### A. Experimental settings

1) *Dataset*: We use a real-world dataset<sup>3</sup> collected and published by DiDi Chuxing, which is a popular online taxi-taking platform in China. The dataset includes passenger order data and taxi trajectory data from 2016.11.1 to 2016.11.30 in the city of Chengdu, China. The information of each passenger order consists of the order ID, the departure time, the origin, the destination, and the arrival time; the information of each taxi consists of the taxi ID, the order ID, the time stamps and locations with a sampling rate of 2~4 seconds. We target at the area with the most dense passenger orders, the longitude from 104 to 104.12 and the latitude from 30.6 to 30.72, which is the most central area in Chengdu. In our experiments, the target area is divided into  $10 \times 10$  blocks, the size of each block is around  $1.56 \text{ km}^2$ , and one day is split into 144 slots, the length of a time slot is 10 minutes. In our experiments, we first randomly take the data in 2016.11.2 as an example to evaluate the one-day maximum urban capacity, then estimate the capacity changes in one month.

2) *Package requests*: We only consider the in-city package delivery, which means the origin and destination of a package are all within the city. Since the datasets do not contain information about package delivery, we generate the package information manually. It's difficult to track the package flows within a city, since it's the privacy of presidents. However, we could estimate the package requests through other kinds of datasets. We crawl the stores marked in the Baidu map<sup>4</sup> as the package origins, the residential area as the destinations. The experimental origins and destinations are chosen randomly

<sup>2</sup>We assume that the block division is fine enough that there are no passenger orders in one block (i.e., origin and destination within the same block).

<sup>3</sup><https://gaia.didichuxing.com>

<sup>4</sup><http://map.baidu.com>

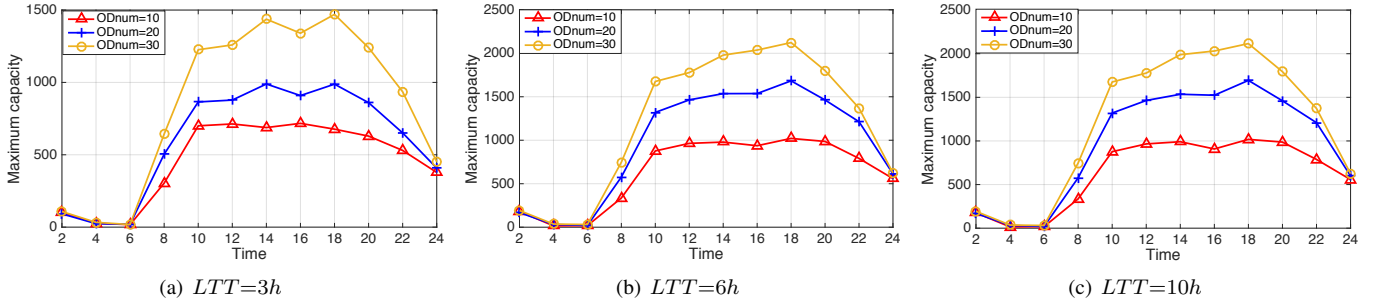


Fig. 4: Maximum urban capacity under different  $LTT$ ,  $ODnum$ , and time in the Non-stop method.

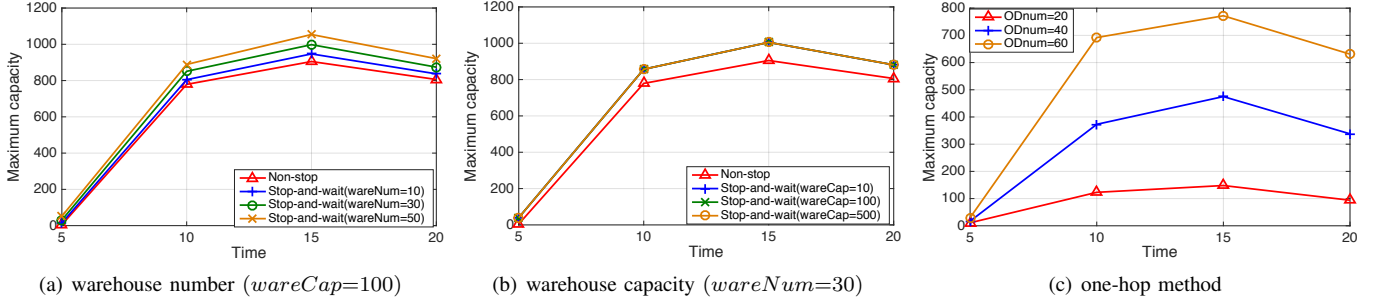


Fig. 5: Maximum urban capacity of different package delivery methods.

from the crawled sets in the following evaluation results. Note that, one block in our parameter settings could contain both an origin and a destination.

3) *Evaluation environment*: All the evaluations in this paper are run in Matlab R2015b on an Intel Core i5-7400 PC with 16-GB RAM and Windows 7 operation system.

## B. Performance results

The following questions are of our interests:

- How does the maximum urban capacity change with different parameter settings, including the  $LTT$ , the  $OD$  number, and the time?
- How different are the maximum urban capacities under various package delivery methods?
- What's the capacity pattern during a week or a month?
- How does the maximum urban capacity change with different package requests?
- How many computation resources are needed to calculate the maximum urban capacity?

1) *Maximum capacity under different  $LTT$ ,  $OD$  number, and time*: We evaluate the maximum urban capacity of the Chengdu city under different parameter settings, including the  $LTT$ , the  $ODnum$  and the evaluation time. The results are shown in Fig. 4. The X axis is the different evaluation time within a day. The Y axis is the corresponding maximum urban capacity. The  $ODnum=10$  in the labels means that there are 10 origins and 10 destinations in this evaluation. The origins and destinations are selected randomly from the crawled sets. Note that, different origins (or destinations) in the same block are not distinguished in our mechanism. Thus, we assume the number of origins (or destinations) is the same as the number of blocks in our experiments. The origins and

destinations in the three subfigures of Fig. 4 are the same when the number is set as the same one. The  $LTT$  is the longest transportation time of a package, which means a package needs to be delivered within the deadline. From the evaluation results, we can see several phenomenons.

First, the maximum urban capacity varies with time in a day. In Fig. 4, we can see that the maximum urban capacity first increase with the time before 18 o'clock, then decrease after 18 o'clock in a day. The root cause is that the passenger demands vary with the time, and the package delivery capacity of the city depends on the passenger order situations. More specifically, the maximum urban capacity before 6 o'clock is almost 0 in Fig. 4 (a) no matter what the  $ODnum$  is. The reason is that the number of passenger orders between 3 o'clock and 6 o'clock is the least. This phenomenon provides suggestions for the logistics companies about the package departure time. It's vain to send packages earlier than the 6 o'clock in the morning.

Second, the maximum urban capacity varies with the number of origins and destinations. In Fig. 4, we can see that the more origins and destinations, the bigger the maximum urban capacity is. The reason is that the passenger demands can be made more full use when there are more  $OD$ s. When the  $LTT$  is 3 hours, the maximum urban capacity of the Chengdu city is at most 650 delivered packages when the  $ODnum$  is 10, while the result when the  $ODnum$  is 30 can reach 1500. Nevertheless, the increase of the urban capacity is not linear with the growth of the  $ODnum$  in our experiments, because the origins and destinations are selected randomly. The distributions of origins and destinations also influence the urban capacity, which will be discussed later.

Third, the maximum urban capacity varies with the  $LTT$ s.

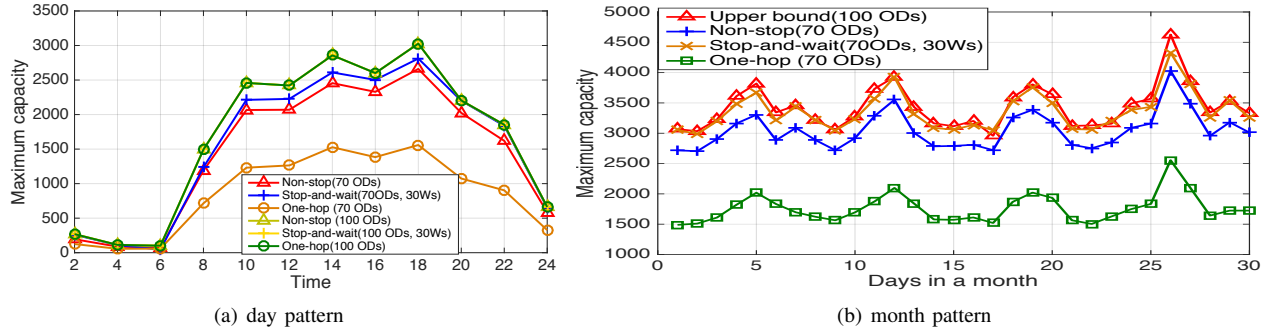


Fig. 6: Patterns of the maximum urban capacity in a day and a month.

In our measurement mechanism, only the successfully delivered packages are counted when calculating the maximum urban capacity. We can see from Fig. 4 that the maximum urban capacity is at most 1500 when the  $LTT$  is 3 hours and the  $ODnum$  is 30, and the corresponding result when the  $LTT$  is 6 hours can reach 2050 since longer  $LTT$  makes more packages being delivered successfully. However, the results when the  $LTT$  is 10 hours and 6 hours are almost the same. The reason is that the number of the passenger orders in the target slot is limited, 6 hours is enough to make full use of the limited passenger orders since we count all the packages from every possible departure slots in our mechanism. Thus, to evaluate the maximum urban capacity when the  $LTT$  as 24 hours, there is no need to count all the departure slots from 24 hours ago, we can estimate it with the result of a smaller  $LTT$ , such as 6 hours.

2) *Maximum capacity compare between different package delivery methods:* In this paper, we provide a general mechanism for the maximum urban capacity measurement in taxi-based logistics, the mechanism can be used for all of the three kinds of taxi-based package delivery methods. In the above experiment, we evaluate the maximum capacity for the Non-stop method. In this experiment, we evaluate the maximum capacity for the other two methods, the Stop-and-wait method in Fig. 5(a), Fig. 5(b), and the One-hop method in Fig. 5(c). In the Stop-and-wait method, many warehouses are needed as the relays of package transportations. Note that, the locations of the warehouses are selected randomly in our evaluation. The Non-stop method can be seen as a special situation of the Stop-and-wait method, where both the number and capacities of the warehouses are 0.

In Fig. 5(a), we evaluate the relationship between the maximum urban capacity and the warehouse numbers, while setting the warehouse capacity as 100. In this evaluation, we select 10 origins and 10 destinations randomly. We can see that, the more warehouses, the larger the maximum urban capacity. The reason is that, more warehouses could make more use of the passenger orders, which are not fully utilized in the Non-stop method. However, the capacity increasing brought by the warehouses is limited, only 17% when up to half of all the blocks contain warehouses.

In Fig. 5(b), the relationship between the maximum urban capacity and the warehouse capacities are evaluated, the warehouse number is 30. We can see that, the maximum

urban capacity first increase with the growth of the warehouse capacity, then remain unchanged no matter how large the warehouse capacity is. The maximum urban capacity under 10 (warehouse capacity) is the same with that under 100 and 500. In our experiment results, we found that for each warehouse number, the urban capacity has an upper bound. Before reaching the upper bound, it is useful to increase the warehouse capacity, yet after reaching the upper bound, improving the warehouse capacity is in vain. This phenomenon can provide some reference for practical setting about the warehouse capacities.

In Fig. 5(c), we evaluate the relationship between the maximum urban capacity and the  $ODnum$  in the One-hop method. The larger the  $ODnum$ , the larger the urban capacity. The relationship is the same with the Non-stop method.

3) *Patterns of the maximum capacity in a day and a month:* In Fig. 6(a), we compare the capacity results in a day of the discussed three package delivery methods under the same parameter settings. We first set the number of origins and destinations as 70 and the number of warehouses in the Stop-and-wait method as 30 (abbr. as 70ODs, 30Ws in the figure). The  $LTT$  is set as 3 hours. We can see that the capacity gap between the Non-stop method and the Stop-and-wait method is small. The latter one is about 1.5% on average higher than the former one. Yet the capacity of the One-hop method is far less than the multi-hop methods. The reason is that the one-hop passenger orders between origins and destination going through the target slot is limited, the multi-hop package delivery methods can further utilize the passenger orders between blocks which are not origins or destinations.

Then we set the number of origins and destinations to 100, and evaluate the maximum urban capacity of the three kinds of package delivery methods. We can see that the urban capacities of the One-hop method, the Non-stop method, and the Stop-and-wait method are the same. The reason is that all the blocks are origins and destinations, all the passenger orders are fully utilized. Both the Non-stop method and the Stop-and-wait method degenerate to the One-hop method. Note that, when all the blocks contain both an origin and a destination, the maximum urban capacity is almost the same with the number of passengers at the same time. However, it is a special situation and not realistic in the real world due to the refined area division. Thus, it is not accurate to estimate the maximum urban capacity through the number of passenger orders. We



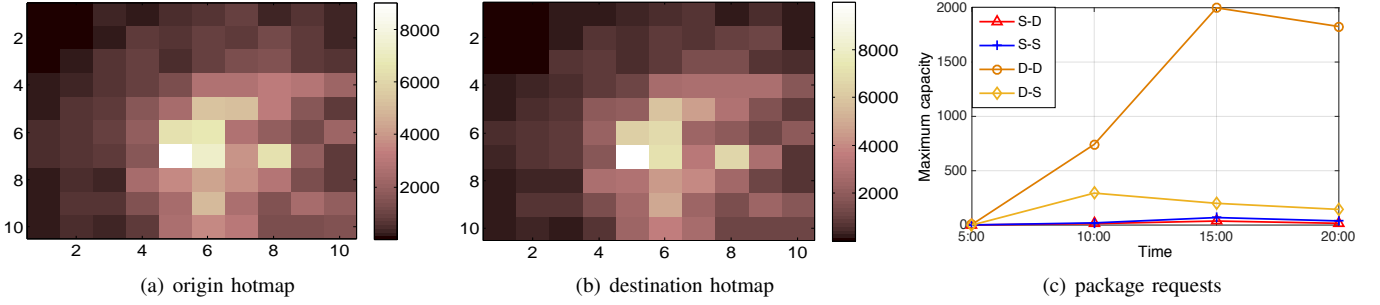


Fig. 7: Maximum urban capacity under different package requests.

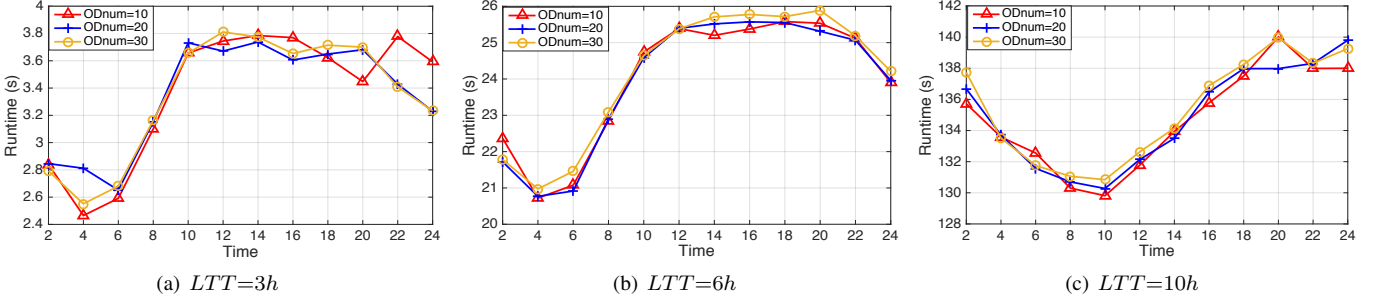


Fig. 8: The runtime of our maximum capacity calculation mechanism.

emphasize that our proposed measurement mechanism for the maximum urban capacity is accurate and efficient based on given package requests and passenger mobility.

We further evaluate the maximum urban capacity in a month, and the results are shown in Fig. 6(b). The evaluation time is from 2016.11.01 to 2016.11.30. In this experiment, the  $LTT$  is set as 6 hours, and the evaluation slot in each day is 15:00. We calculate the maximum urban capacity under the three kinds of package delivery methods and the upper bound where all the blocks contain both an origin and a destination. We can find from the results that all the patterns show periodic changes week by week. The 2016.11.05, 2016.11.12, 2016.11.19 and 2016.11.26 are the four Saturdays in this month, and the maximum urban capacity of all the four kinds of patterns reach the maximum. The reason is that the taxi-taking demands are the max on Saturday in Chengdu city. The maximum urban capacity of the multi-hop package delivery method is around 3000 when there are 70 ODs and 30 warehouses, while that of the one-hop method is around 1800. The maximum capacity of the Non-stop method and the Stop-and-wait method is close to each other, yet that of the One-hop method is far less than the multi-hop methods.

4) *Maximum capacity under different package requests:* In Fig. 4, the maximum urban capacity varies with the number of ODs, yet the relationship between them is not linear, because the OD distribution other than the number also influences the calculation result. The OD distribution reflects the package requests. In order to estimate the maximum urban capacity under various package requests, we divide the target blocks into four kinds, i.e., dense origin, sparse origin, dense destination, and sparse destination. The dense (sparse) origin indicates lots of (few) passengers departure from this block, and the dense

(sparse) destination indicates that lots of (few) passengers arrive in this block. The threshold between the density and sparsity can be made manually according to the different situations. The hot-maps of the origins and destinations of passenger orders are plotted by Fig. 7(a) and Fig. 7(b). We can see that the two hot-maps are similar to each other, most of the hot-origins are also the hot-destinations, which is in line with common sense.

We estimate the maximum urban capacity with different package requests, including dense origin to sparse destination (D-S), sparse origin to dense destination (S-D), dense origin to dense destination (D-D), and sparse origin to sparse destination (S-S). The results can be seen in Fig. 7(c). Note that, the  $ODnum$  is set as 5 in this evaluation. We can see that the maximum urban capacity varies obviously under different package requests. The capacity difference between the S-D, the S-S, and the D-S is small, yet the result of the D-D is far more than the other three kinds of package requests. That phenomenon is in line with our cognition. As we select the ODs randomly in Fig. 4, the result may fall in the gap between the four package requests. For example, it is reasonable for the maximum urban capacity at 15:00 varies from 20 to 2000 if the 5 ODs are selected randomly.

5) *Computation complexity:* We evaluate the computation complexity of our maximum capacity measurement mechanism. In Section III, we have analyzed the complexity of our mechanism theoretically, the block number  $M$ , the slot number  $N$ , the longest transportation time  $LTT$ , and the number of edges in the constructed spatial-temporal graph are the key factors. In our evaluation, the  $M$  and  $N$  is not changed, the influence of the  $LTT$  is demonstrated in Fig. 8. The runtimes under different  $LTT$  settings including 3 hours, 6 hours, 10

hours, are demonstrated in Fig. 8 (a), (b), (c), respectively.

We can see that, for a given  $LTT$ , the runtime of our mechanism with different  $ODnums$  are similar. The reason is that the  $ODnum$  influences the number of edges in  $G'$  by changing the number of added virtual edges, but this effect is small due to the large basement. With the growth of  $LTT$ , the runtime of our mechanism increases significantly. This phenomenon is determined by our theoretical analysis, where the  $LTT$  setting plays a key role. However, as analyzed in Fig. 4, there is no need to set  $LTT$  so long since the result under 6 hours and 10 or more hours are almost the same. When the  $LTT$  is set as less than 6 hours, the runtime of our measurement mechanism is in seconds. These experiment result proof the efficiency of our measurement mechanism.

## VI. CONCLUSION AND DISCUSSIONS

In this paper, we address the maximum urban capacity problem in taxi-based package delivery. We first define the maximum urban capacity as the number of on-road packages in unit time. Then we propose a measurement mechanism for the Non-stop package delivery method. At last, we expand our measurement mechanism to other package delivery methods with a few adaptations. We evaluate the maximum urban capacity of different methods with real-world datasets and find that the capacity gap between the Non-stop method and Stop-and-wait method is small, yet the capacity of the One-hop method is far less than the multi-hop methods.

We propose a mechanism for the maximum urban capacity measurement of the taxi-based logistics in this paper. We emphasize that this mechanism is just a basic mechanism and many more realistic settings can be achieved based on this mechanism. For example, the packages produced in each origin is assumed to be as many as needed in this model. In more realistic scenes, the edge weight between the virtual origin and each actual origin can be set as the production capacity of each origin instead of the infinity used in this model. Moreover, the  $LTT$  is assumed as the same for all packages and the ODs departure from the same time are also assumed as the same. In a more complicated scenario, the group of the  $LTT$ ,  $OD$ , and departure time settings can be more flexible and it does not change our mechanism.

## ACKNOWLEDGMENT

This work is partially supported by the National Natural Science Foundation of China under Grant No.61772544, 61872372, U19B2024, and 61672195, the NUDT Research Grants No. ZK19-38.

## REFERENCES

- [1] "Didi chuxing," <http://www.didichuxing.com>, 2018.
- [2] "Uber," <http://www.uber.com>, 2018.
- [3] F. Wang, Y. Zhu, F. Wang, and J. Liu, "Ridesharing as a service: Exploring crowdsourced connected vehicle information for intelligent package delivery," *IWQoS*, 2018.
- [4] Y. Chen, D. Guo, M. Xu, G. Tang, T. Zhou, and B. Ren, "Pptaxi: Non-stop package delivery via multi-hop ridesharing," *IEEE Transactions on Mobile Computing*, 2019.

- [5] C. Chen, D. Zhang, X. Ma, B. Guo, L. Wang, Y. Wang, and E. Sha, "CROWDELIVER: Planning city-wide package delivery paths leveraging the crowd of taxis," *IEEE Transactions on Intelligent Transportation Systems*, vol. 18, no. 6, pp. 1478–1496, 2017.
- [6] R. Lowe and M. Rigby, "The last mile: Exploring the online purchasing and delivery journey," *Barclays*, September, 2014.
- [7] R. Masson, A. Trentini, F. Lehuédé, N. Malhéné, O. Péton, and H. Tlahig, "Optimization of a city logistics transportation system with mixed passengers and goods," *EURO Journal on Transportation and Logistics*, vol. 6, no. 1, pp. 81–109, 2017.
- [8] G. Cheng, D. Guo, J. Shi, and Y. Qin, "Smart city-wide package distribution using crowdsourced public transportation systems," *IEEE Internet of Things Journal*, 2019.
- [9] J.-F. Rougès and B. Montreuil, "Crowdsourcing delivery: New interconnected business models to reinvent delivery," *1st international physical internet conference*, pp. 1–19, 2014.
- [10] H. Gao, C. H. Liu, W. Wang, J. Zhao, Z. Song, X. Su, J. Crowcroft, and K. K. Leung, "A survey of incentive mechanisms for participatory sensing," *IEEE Communications Surveys & Tutorials*, vol. 17, no. 2, pp. 918–943, 2015.
- [11] X. Zhang, Z. Yang, W. Sun, Y. Liu, S. Tang, K. Xing, and X. Mao, "Incentives for mobile crowd sensing: A survey," *IEEE Communications Surveys & Tutorials*, vol. 18, no. 1, pp. 54–67, 2016.
- [12] A. Arslan, N. Agatz, L. G. Kroon, and R. Zuidwijk, "Crowdsourced Delivery – A Pickup and Delivery Problem with Ad-Hoc Drivers," *SSRN Electronic Journal*, pp. 1–29, 2016.
- [13] Y. Liu, B. Guo, S. Member, C. Chen, H. Du, Z. Yu, S. Member, D. Zhang, and H. Ma, "FoodNet: Toward an Optimized Food Delivery Network based on Spatial Crowdsourcing," *IEEE Transactions on Mobile Computing*, vol. 1233, no. c, pp. 1–1, 2018.
- [14] Y. Iida, "Methodology for maximum capacity of road network," *Transaction of Japan Society of Civil Engineers*, vol. 205, pp. 147–150, 1972.
- [15] A. Chen, H. Yang, H. K. Lo, and W. H. Tang, "Capacity reliability of a road network: an assessment methodology and numerical results," *Transportation Research Part B: Methodological*, vol. 36, no. 3, pp. 225–252, 2002.
- [16] B. Bevrani, R. L. Burdett, A. Bhaskar, and P. K. Yarlagadda, "A multi commodity flow model incorporating flow reduction functions," *Flexible Services and Manufacturing Journal*, pp. 1–31, 2019.
- [17] T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein, *Introduction to algorithms*. MIT press, 2009.
- [18] L. R. Ford and D. R. Fulkerson, "Maximal flow through a network," in *Classic papers in combinatorics*. Springer, 2009, pp. 243–248.



**Yueyue Chen** received her B.S., M.S., and Ph.D. degrees in Computer Science and Technology from National University of Defense Technology (NUDT), Changsha, China in 2013, 2015 and 2019, respectively. She is currently an engineer in unit 61660 of PLA. Her main research interests include wireless networks, mobile computing, and crowd-sensing.



**Deke Guo** received the B.S. degree in industry engineering from the Beijing University of Aeronautics and Astronautics, Beijing, China, in 2001, and the Ph.D. degree in management science and engineering from the National University of Defense Technology, Changsha, China, in 2008. He is currently a Professor with the College of System Engineering, National University of Defense Technology, and is also with the College of Intelligence and Computing, Tianjin University. His research interests include distributed systems, software-defined networking, data center networking, wireless and mobile systems, and interconnection networks. He is a senior member of the IEEE and a member of the ACM.



**Ming Xu** is a professor in Department of Network Engineering, College of Computer, National University of Defense Technology, Changsha, China. He is an editor of the Journal of Communications and the International Journal of Pervasive Computing, and a technical program member of several international conferences such as IEEE PerCom. His current research interests include ad-hoc networks, vehicular networks, and wireless mesh networks. He is a member of the IEEE.



**Guoming Tang** received the B.S. and M.S. degrees from the National University of Defense Technology, China, in 2010 and 2012, respectively, and the Ph.D. degree in Computer Science from the University of Victoria, Canada, in 2017. He joined the College of System Engineering at the National University of Defense Technology in 2017 where he is currently an Assistant Professor. Aided by machine learning and optimization technologies, his research mainly focuses on computational sustainability issues and targets at distributed/networking systems.



**Geyao Cheng** received the B.S. and M.S. degree in management science and engineering from National University of Defense Technology, Changsha, China, in 2017 and 2019, respectively. She is currently a Ph.D in College of System Engineering, National University of Defense Technology, Changsha, China. Her research interests include mobile computing and crowdsensing.