

Data-Driven Order Assignment for Last Mile Delivery

Sheng Liu

Department of Industrial Engineering and Operations Research
University of California, Berkeley
lius10@berkeley.edu

Long He

NUS Business School, National University of Singapore
longhe@nus.edu.sg

Zuo-Jun Max Shen

Department of Industrial Engineering and Operations Research and Department of Civil and Environmental Engineering
University of California, Berkeley
maxshen@berkeley.edu

We study how delivery data can be applied to improve on-time performance in last mile delivery service. Motivated by a food delivery service provider, we discuss a data-driven framework to model the delivery performance and optimize the decision on how to assign orders to drivers. From the analytics for the service provider, we decompose driver's total delivery time into two components: *uncertain service time* at customer locations and *predictable travel time* on road. Leveraging classical results in routing literature and machine learning approaches, we propose a prediction model for the delivery tour length, which captures driver's routing behavior in practice as well as provides satisfactory prediction for the travelling salesman problem in theory. We then demonstrate an application of the proposed delivery tour prediction model in the order assignment problem. An optimization model is first developed to minimize the expected total delay of all routes, which can be solved via sample average approximation scheme using the historical data. Furthermore, to deal with the inadequacy of samples at some locations, we develop a distributionally robust optimization model using limited distributional information on service time. Utilizing the independence in service times verified from data analytics, we overcome the difficulty in obtaining the robust solution under piecewise linear objective, and derive a mixed-integer second order conic program formulation that is computationally tractable and scalable. A branch-and-price algorithm is then proposed to solve both models efficiently. In the numerical study, we show the benefits of data-driven order assignment models integrated with delivery tour prediction, compared to the models based on classical vehicle routing problems. Our results indicate the importance of learning from the operational data to reflect the practical aspect—driver's routing behavior. We also discuss several practical and managerial issues, such as the impact of sample size and staffing levels, from the numerical experiments.

Key words: last mile delivery, data-driven modeling, prediction, robust optimization, branch-and-price.

History: This draft: 15 May, 2018

1. Introduction

As ecommerce booms and customers expect faster delivery, food shopping has recently been shaped to a case in point. Fast-growing online platforms enable convenient food ordering and delivery ser-

VICES for customers. In food delivery service provided by platforms such as Grubhub and UberEATS, food is prepared and packaged by restaurants and the platform is only responsible for food pickup and delivery. By contrast, food service providers such as SpoonRocket and Domino's Pizza prepare and deliver their own food boxes. A key challenge faced by both types of service providers is how to assign prepared orders to available carriers or drivers for fast and efficient delivery.

In this paper, we consider the order assignment problem motivated by a food service provider that prepares and delivers food to its customers in Shanghai, China. The provider operates a central kitchen, which is referred to as "the depot" thereafter, and serves customers within a certain radius from the depot. Customers place orders before each cutoff time, e.g., 10:30 am, are promised to receive the orders within a time window, e.g., by 11:45 am. However, the provider found that delays seemed to be inevitable if the orders were not well allocated to drivers, as a delay at one location will propagate to subsequent visits by the same driver.

The order assignment problem faced by food service providers is complicated, due to the following practical challenges. First, the complex road condition and practical constraints make it difficult for drivers to follow suggested routes or delivery sequences. For example, the considered provider allows drivers to have freedom in deciding their own routes to deliver the assigned orders. It is thus common to see drivers – riding electric bikes in Chinese cities – flexibly adjust their routes based on their experience and realtime road conditions. Consequently, the actual delivery routes usually deviate from the recommended routes and an accurate estimation of actual delivery tour for serving a set of customer locations is difficult. Second, the time a driver spends at a customer location, which we term as the "service time", is highly uncertain. As the customer locations are often high rise buildings in metropolitans, drivers usually need to find parking spaces, navigate to the right floor and meet customers in person. The service time varies and depends on the customer location and the order size, which are also random from day to day. Generally, it takes longer service time to navigate in a taller building and to deliver larger orders at one location. Compared to the service time, the travel time on road has much less uncertainty, as indicated by the service provider for reasons, including that the drivers riding electric bikes can take bicycle lanes to avoid traffic congestion. Therefore, the travel time is more predictable once the delivery tour is determined.

To tackle the challenges described above, in this paper, we propose a data-driven framework to model the delivery performance and optimize order assignment decisions. We use data analytics to develop a delivery tour length prediction function from delivery operational data, to incorporate driver's routing behavior in the subsequent order assignment optimization models. The resulting data analytics also highlight the importance of dealing with service time uncertainty, in meeting the on-time target. Using historical delivery data in the scheme of sample average approximation

(SAA), we propose a data-driven order assignment (DOA) model as a mixed-integer linear program (MILP) that assigns orders to available drivers to minimize the expected total delay of all routes. To deal with the inadequacy of observations at several locations, we further develop a DOA model as a mixed-integer second order conic program (MISOCP), using the distributionally robust optimization (DRO) framework, where limited distributional information can be obtained from the data. We then propose a branch-and-price algorithm for both DOA models, aiming to improve the computational performance for real applications.

In the case study using the delivery data set, we employ a data-driven approach to evaluate the out-of-sample performance of the proposed DOA models with delivery tour length prediction function, in benchmark with the models based on classical vehicle routing problem (VRP). We also discuss several practical and managerial issues, such as the impact of sample size for SAA formulations and the provider's staffing considerations. We summarize our contributions as follows.

1. *Data-driven modeling:* We decompose the uncertainty of total delivery time into the uncertainty service time at the customer locations and a more predictable travel time on road. That is, from a graph perspective, we aggregate all uncertainties to the nodes and leaving the edges predictable. Such treatment allows us to model the travel time and integrate uncertainties in the optimization models. Our delivery tour length prediction function learned from real data reveals higher prediction accuracy compared to the solution from classical traveling salesman problem in the practical application. Besides, it predicts the tour length without explicit optimization on the delivery sequence and renders computational tractability in the subsequent order assignment optimization.

2. *Data-driven optimization:* We employ the SAA and DRO frameworks to utilize real data for both abundant-data and limited-information contexts. Utilizing the independence in service time, which is verified from data analytics, we overcome the difficulty in solving the robust problem under piecewise linear objective and derive an equivalent MISOCP formulation, which is computationally tractable and scalable. Furthermore, by exploiting the problem structure, the branch-and-price algorithm delivers superior performance in both solution time and solution quality, compared to solving the standard MILP and MISOCP formulations directly using commercial solvers.

3. *Managerial insights:* From the out-of-sample evaluation in the case study, we find that the DOA models significantly outperform the VRP-based models, which ignore driver's routing behavior. The results numerically quantify the benefit of data-driven modeling. We also observe that the performance of the DOA model using SAA improves with larger sample size, at the cost of longer computational time. However, the VRP-based model using SAA does not necessarily benefit from larger sample size, due to its bias in delivery tour length estimation. Moreover, applying the VRP-based model may lead to unnecessary overstaffing to target on-time performance. Finally, the

performance gap between the DOA and VRP-based models is larger under more stringent delivery requirements — with less drivers and tighter delivery time window.

The remainder of the paper is organized as follows. In Section 3 and 4, we analyze the food delivery data from the industry partner that lead to practical modeling of uncertainties in delivery route and the development of the tour prediction function. Based on the results of data analytics, we discuss data-driven order assignment models in Section 5. Next, we develop solution algorithms in Section 6 and conduct a case study based on the delivery data in Section 7. Finally, we conclude our paper in Section 8.

2. Literature Review

The order assignment problem in last mile delivery has been studied extensively in the form of vehicle routing problems (VRP) in the transportation and operations research literature (see, e.g., Solomon 1987, Laporte 2007). VRPs determine the visiting sequence of a number of vehicles to minimize the travel cost. There are various extensions of VRPs developed in both deterministic and stochastic contexts. Some classical instances include VRP with stochastic demand and VRP with stochastic travel time, and with or without time window constraints (see Laporte et al. 1992, Gendreau et al. 1996, Campbell and Thomas 2008, Erera et al. 2010 for details). More recently, Jaillet et al. (2016) propose a new decision criterion to measure the risk associated with violations of time window constraints when travel times and demands are uncertain. Meanwhile, in a dynamic manner, Bertsimas and Van Ryzin (1991, 1993) analyze VRPs with stochastic wait times and derive several policies under light and heavy traffic conditions using queuing theory. Compared to the stochastic VRP with time window, our data-driven order assignment models optimize the order assignment decisions using a delivery tour length prediction function learned from data, which does not require explicitly determining the delivery sequence.

For many strategic planning decisions, such as service region partitioning, VRP tours (i.e., travel distances) are usually approximated by analytical functions. For example, the well-known Beardwood-Halton-Hammersley (BHH) Theorem (Beardwood et al. 1959) allows the length of an optimal traveling salesman problem (TSP) tour to be expressed in the probability density function of demand points, following the law of large numbers. Utilizing the asymptotic result in BHH theorem, Carlsson (2012) considers the problem of partitioning a service region to balance the workload among vehicles, where demand locations are independent and identically distributed samples from a given probability density function. Moreover, continuum approximation (CA) models are widely used in the design of supply chain network to avoid the combinatorial nature of such decisions and yield tractable analytical solutions. Its applications include terminal design problem (Ouyang and Daganzo 2006), inventory routing problem (Shen and Qi 2007), dynamic facility location (Wang

et al. 2016), supply chain distribution network design (Lim et al. 2016), and analysis of online retail grocery (Belavina et al. 2016). Alternatively, approximation functions to TSP tours can also be obtained through simulation. In a study of last-mile transportation system for passengers, Wang and Odoni (2014) conduct extensive simulation experiments to obtain tractable approximations for TSP tour and customer waiting time in queue.

A significant difference between our paper and the above literature is that we employ a data-driven approach to estimate the actual delivery tour length, which reflects driver’s routing behavior. Inspired by analytical approximations of TSP tour (e.g., Beardwood et al. (1959)), we construct relevant features and predict the delivery tour length based on spatial information of orders using machine learning, which enables us to capture the travel patterns of drivers from the delivery data. Moreover, the delivery tour length prediction function also performs well in estimating the TSP tour, and thus can be used as a closed-form approximation of TSP tour that preserves computational tractability in optimization.

In the context of last mile delivery, assigning orders to vehicles is also referred to as dispatching. For example, Klapp et al. (2016) study the dynamic dispatch problem where orders arrive dynamically through a day. Their research involves the decision on whether to dispatch a single vehicle in each epoch, considering future arrivals. By contrast, in our context, orders are batched together by their cutoff times and our research focus is on assignment within a batch of orders by taking into account the delivery tour and service time. Our problem is also related to the order batching problem in warehouse. For instance, Gademann and Velde (2005) study the order picking strategy in a parallel-aisle warehouse to minimize the total travel time, where the extraction time — the time spent at the pick location, is omitted in their optimization model, due to the assumption of constant total extraction time. In our paper, however, we identify a major reason for delayed deliveries is the uncertainty in service time — the time spent at the customer location. We explicitly consider such uncertainty in our models to minimize the expected total delay.

As data become richer and more accessible in operational contexts, more researchers are calibrating their modeling assumptions and corresponding analytical functions with real data. Our paper is thus closely related to the following papers that integrate predictive models with optimization. Kong et al. (2010) provide a simple model for the benefit of potential region in organ allocation that can be estimated from data and maintains tractable in optimization. Collaborating with a large gas utility, Angalakudati et al. (2014) address the resource allocation problem to minimize maintenance crew overtime in performing standard jobs as well as meeting emergency jobs. Ferreira et al. (2015) apply machine learning techniques such as regression trees to estimate lost sales, predict demand for new products and incorporate the demand prediction model into the pricing optimization problem. Hekimoğlu et al. (2016) perform data analysis to derive the functional form

of the price evolution and build the pricing and selection model for wines based on the empirical findings. Ahipařaoglu et al. (2015, 2016) develop optimization models based on cross-moment and marginal distribution information to estimate and analyze traffic equilibrium of transportation systems. Bertsimas et al. (2016) combine machine learning and optimization techniques to guide the design of chemotherapy regimens. For optimization problems with unknown cost parameters in the objective functions, Elmachtoub and Grigas (2017) propose a new predictive framework to incorporate the structure of the optimization problem, which enables robust performance against model misspecification. Efforts in integrating prediction with optimization can also be found in Ban and Rudin (2018), where feature-based machine learning algorithms are developed for the newsvendor problem. In addition, recent research in revenue management has generated nonparametric choice modeling approaches that can be used to improve the prediction accuracy and assortment decision (Farias et al. 2013, Bertsimas and Miřic 2015). Our paper follows a similar journey in data-driven modeling, where the key features in the research problem are identified and formulated based on data analytics. Nevertheless, our idea of decomposition of total delivery time allow us to model the uncertainty in service time as well as develop a prediction function for the travel time, which is novel and practical for delivery operations.

The closest to our paper is Zheng et al. (2016) where least squares linear and quadratic estimators are proposed to approximate the distribution of project completion using related persistency problem. Specifically, the linear estimator is an affine function of the random durations of individual activities. The authors solve for the least squares normal approximation to obtain the best parameters that consist of two parts: the intercept and coefficients associated with individual activities. Similar in spirit, we consider the total delivery time in two parts: the travel time and the uncertain service time at customer locations, in analogy to the intercept and random activity durations in Zheng et al. (2016). Because the travel time has less uncertainty compared to the service time in our application, we further develop a prediction model for the travel time by utilizing the spatial information of customer locations as features.

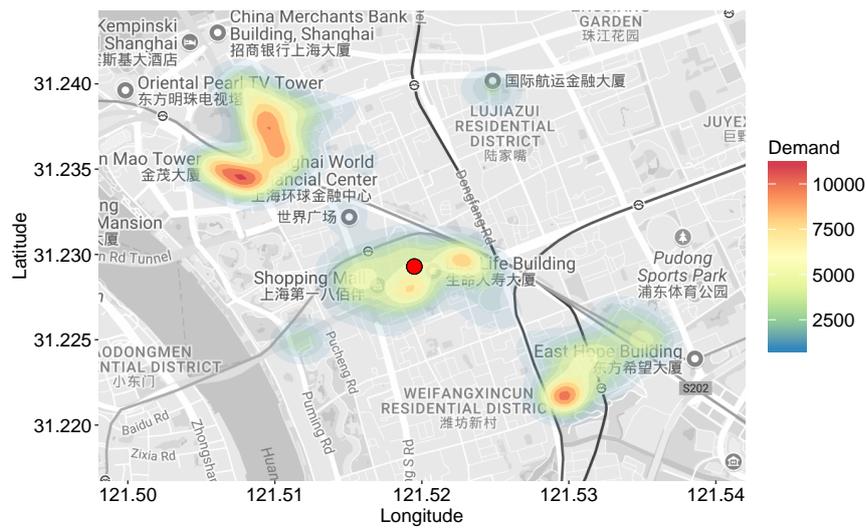
When analytics is applied to develop data-driven models, the resulting optimization formulations often call for data-driven solution approaches, without assumptions of specific probability distributions for underlying uncertainties. A popular choice of such approaches is the sample average approximation (SAA) scheme that captures the stochastic nature of the uncertainties, e.g., the service time in our problem, without parametric assumptions. Furthermore, distributionally robust optimization (DRO) framework is also widely used, especially when the empirical data are not sufficient to calibrate full distributional information. Our paper proposes both SAA and DRO models for the order assignment problem, under the mentioned practical considerations. When

customer locations are unknown at the time of partitioning the territory for multiple vehicle routing, Carlsson and Delage (2013) deploy the DRO framework to optimize the worst-case workload under limited distributional information, such as the first- and second-order moments information, of demand spatial distribution. With the recent development of DRO using Wasserstein distance (e.g., Esfahani and Kuhn 2015, Gao and Kleywegt 2016), Carlsson et al. (2017) develop robust partitioning for TSP under the worst-case spatial distribution of demand, which is specified by the Wasserstein distance between the continuous distributional density function and observed discrete sample points. In our study of order assignment problem, however, the customer locations are realized as the orders have been received and are waiting to be delivered. Therefore, our focus is the uncertainty in service time aiming to reduce expected delays, instead of the uncertainty in spatial distribution of demand to balance the workload of vehicles. Our problem is also related to the robust bin packing problem studied in Zhang et al. (2016), because the target delivery time and uncertain service time in the order assignment problem are analogous to the fixed capacity and uncertain item weight in the bin packing problem respectively. While Zhang et al. (2016) use chance constraints with the probability of exceeding bin's capacity, we consider the worst-case expected total delay in the objective function of our DRO model. To address the difficulty in dealing with resulting piecewise linear objective under DRO (Ardestani-Jaafari and Delage 2016), we decompose the DRO model by the independence in service time and utilize the general projection property in Popescu (2007) to derive a tractable optimization formulation.

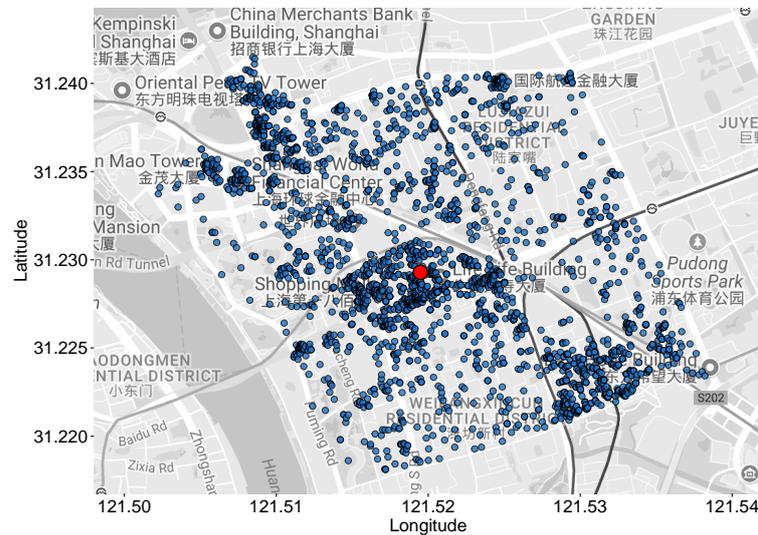
3. Analytics for A Food Delivery Service

We acquire an operational data set that contains detailed ordering and delivery information for a 2-month period in 2015, from a food delivery service provider operating in Shanghai, China. The data set records the following information: 1) order time: the time when the order is received by the provider; 2) quantity: the number of items ordered; 3) time window: the guaranteed delivery time; 3) pick-up time: the time when the order is collected by a driver; 4) delivery time: the actual time when the order is delivered to the customer; 5) longitude and latitude: the customer location; 6) cutoff time: it identifies the order batch and all orders with the same cutoff time will be dispatched together. Cutoff time is 75 minutes before the guaranteed delivery time. The provider has determined a sequence of cutoff times $\{t_1, t_2, \dots\}$ and all orders placed in $(t_k, t_{k+1}]$ will be guaranteed to be delivered by $t_{k+1} + 75$ minutes. For instance, orders placed between 10:15 am and 10:30 am are promised to be delivered by 11:45 am.

There are 2,846 customer locations identified in the data set. Figure 1 shows the spatial distribution of customer locations and the historical demand density in the study period. We observe that most customer locations are within a square area around the depot and many customer locations are geographically adjacent. The demand density is not uniform across the service region.



(a) Demand density heatmap



(b) Customer locations

Figure 1 Historical demand density and customer locations (the red dot represents the depot).

Among all customer locations, there are 1,184 points that only generate 1 order during the 2-month period. Moreover, 21.55% of the total orders were not delivered on time. We summarize the number of orders and delayed time in Table 1. In particular, the variability in the number of orders is huge among customer locations, and the average delay per order is 9.26 minutes, which is not satisfactory.

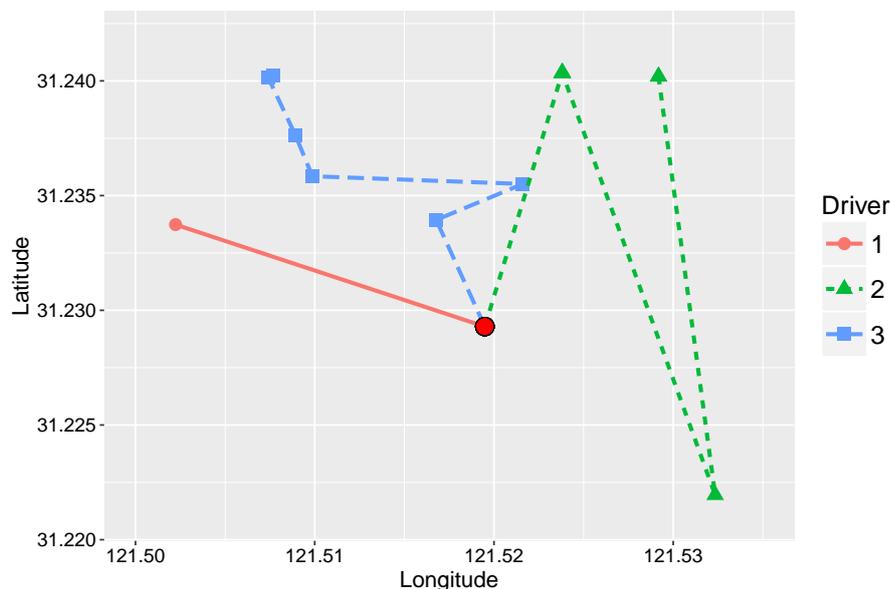
3.1. Order Assignment Practice

Potential causes for the delay in delivery involve slow food preparation at the depot and poor order assignment policy leading to inefficient delivery routes. Due to the limited production capacity at

Table 1 Statistics of demand and delay (in minutes) during the 2-month period

| | Min | Median | Max | Mean |
|-------------------------------|-----|--------|-----|-------|
| Number of orders per location | 1 | 2 | 612 | 10.36 |
| Delay per order | 0 | 0 | 197 | 9.26 |

the depot and unexpected disruptions, food preparation may take longer time than planned, which suppresses the time window for driver's delivery. Resolving this issue often requires expanding capacity and optimization of food preparation operations, e.g., job scheduling, which is out of scope of this paper and we leave for future exploration. When delivery time window is predictable, e.g., after food preparation process, the delivery performance heavily depends on the order assignment policy. In its current practice, the provider assigns orders manually. Figure 2 demonstrates a set of orders that share the same cutoff time and similar pickup times (belong to one dispatch), and how they were delivered in sequence by different drivers (we only show 3 drivers for clarity purpose). We observe that driver 1 only carried one order and visited one customer location while driver 3 made deliveries to more than 5 locations. In addition, driver 2 was assigned to locations at opposite directions and consequently took long detours. Such assignment decision could cause inefficient utilization of drivers and undesired delivery performance. In this dispatch, driver 3 failed to deliver on time. Furthermore, one may notice that driver 2 did not follow the shortest path by traveling back and forth. It is because the provider only assigns orders to drivers without specifying the delivery sequence, allowing them to design their own routes.

**Figure 2** Delivery routes by 3 drivers (the red dot represents the depot).

It is therefore necessary to develop an efficient order assignment policy that delivers better on-time performance. In this paper, the on-time performance of a route is measured by its delay in visiting its last customer, since all customers in the same batch share the same promised delivery time. That is, whenever the last visit is on-time, all visits in the same route are on-time. Since the drivers have freedom in customizing their routes, the specific visiting sequence is unpredictable. Therefore, focusing on the delay of all routes is more sensible than focusing on the delay of all orders, which requires to know (or solve for) the visit sequence at the stage of order assignment that the drivers may not follow in practice.

In the discussion below, we consider the total delivery time of a route consists of two components: the uncertain service times at customer locations and the predictable travel time of the route. It allows us to model the dependence of total delivery time on the order assignment, e.g., number of orders and their locations, as well as the uncertainty at customer locations. From a graph perspective, our primary idea is to introduce the uncertainty at the nodes while keep the edges predictable.

3.2. Uncertainty in Service Time

Food orders are usually delivered to customers in person. As many customer locations are high rise buildings in urban areas, drivers often need to find parking spaces, navigate to the right floor and meet customers to handover the orders. To differentiate with travel time in delivery tour, we use the term “service time” to denote the time a driver spends between the arrival and departure at a customer location. Since the service time is not measured explicitly in the data set, we estimate the service time spent at each customer location as follows. We first measure the travel distance between two consecutively visited customer locations using the shortest path on Google Maps, and calculate the travel time assuming an average electric bike speed as 15km/hr in Shanghai (Cherry and Cervero 2007). Such assumption is based on the fact that electric bikes are allowed to take bicycle lanes to avoid traffic congestions. The service time is therefore estimated as the difference between the delivery time observed from the data and the estimated arrival time, i.e., the departure time and the estimated travel time from the previous customer location. We note that by assuming a different speed, the estimated values of service time may differ. Nevertheless, in this section, we intend to investigate the existence of uncertainty in service time, as highlighted by practitioners in delivery service. In theory, an accurate measure of the service time can be obtained, if the provider tracks more detailed vehicle location information.

The distribution of service time estimated across all customer locations is presented in Figure 3 (a), where its mean is 4.11 minutes. We observe that the variability of service time is large system-wide. We also notice that the service time is heterogeneous across different locations. The

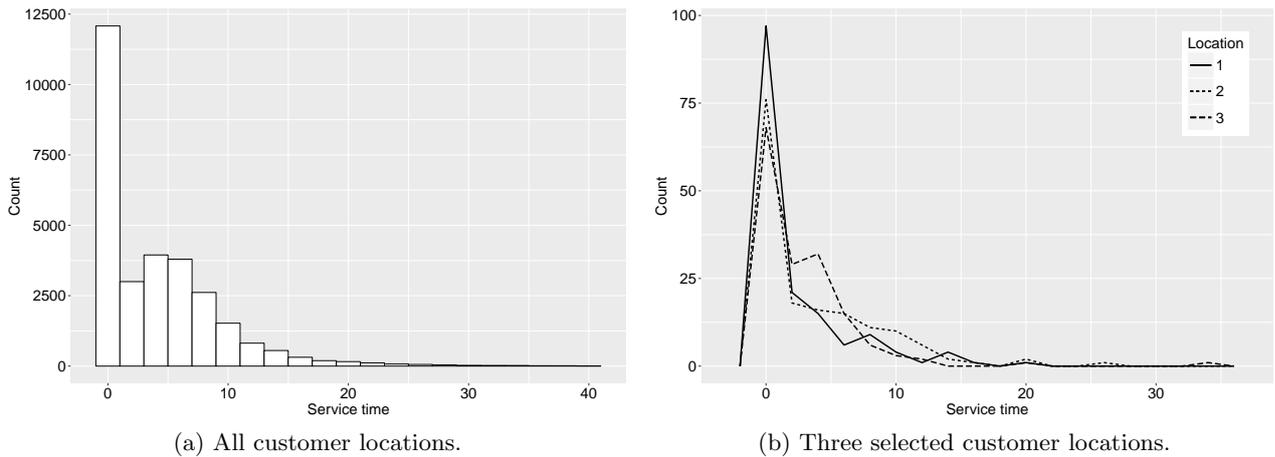


Figure 3 Distribution of service time (in minutes).

mean service time of a customer location varies from 0 minute to 38.69 minutes. Furthermore, the distributions of service time at various locations are different. Figure 3 (b) displays histograms of service time at three different customer locations with similar number of orders. We can see that the service times at those locations follow different distributions, e.g., the probability of having service time longer than 5 minutes is higher at location 3 than the rest.

Furthermore, the service time across different customer locations are mostly independent. We apply Hoeffding’s independence test (Hoeffding 1948) to pairs of locations that share enough number of visits in the data set. The result shows that only 3% of them reject the null hypothesis of independence with a significance level of 0.05. This observation facilitates the analysis of the worst-case expected total delays in Section 5.2.

3.3. Driver’s Routing Behavior

In the practice of the studied service provider, drivers have freedom in planning their own routes. It is because the service provider believes that the drivers are familiar with the local area, of which the information may not be represented in its database, e.g., real time road condition and hidden paths not shown on maps. While some providers suggested routes for their drivers, they also allow drivers some flexibility for adjustment. Given assigned orders, the route planning for a single driver can be solved as a traveling salesman problem (TSP) in theory. This subsection examines driver’s routing behavior in comparison to the theoretical TSP routes.

Based on the routes reconstructed from data, we obtain the actual delivery tour — the travel distance from the depot to the last visited customer by each driver. To both the provider and drivers, the key objective is to minimize delays: the provider receives complaints from customers and drivers receive penalties on delayed orders. Since the final segment from the last visited customer location to the depot does not influence the on-time performance, drivers tend to minimize one-way

delivery tour, starting from the depot and ending at the last customer location. Using a standard TSP formulation with dummy node (see details in the Appendix), one can find the shortest possible one-way delivery tour to visit all assigned customers in a single route by a driver. However, the theoretical TSP solution may not represent the reality on how drivers actually plan their routes and deliver orders. To compare the TSP routes with the actual ones, we calculate the lengths of both routes using the distances between customer locations calibrated from the Distance Matrix function of Google Maps. Figure 4 presents the histogram of the differences between the actual delivery tour length and that from TSP solution for all observed routes. We find that the actual delivery tour length is consistently greater or equal to the delivery tour length from the TSP solution. The histogram has a long tail that highlights the instances where actual delivery tour deviates from the shortest route by a large amount. The difference between the actual delivery tour length and the TSP solution is 0.59 km on average and can be as large as 6.67 km.

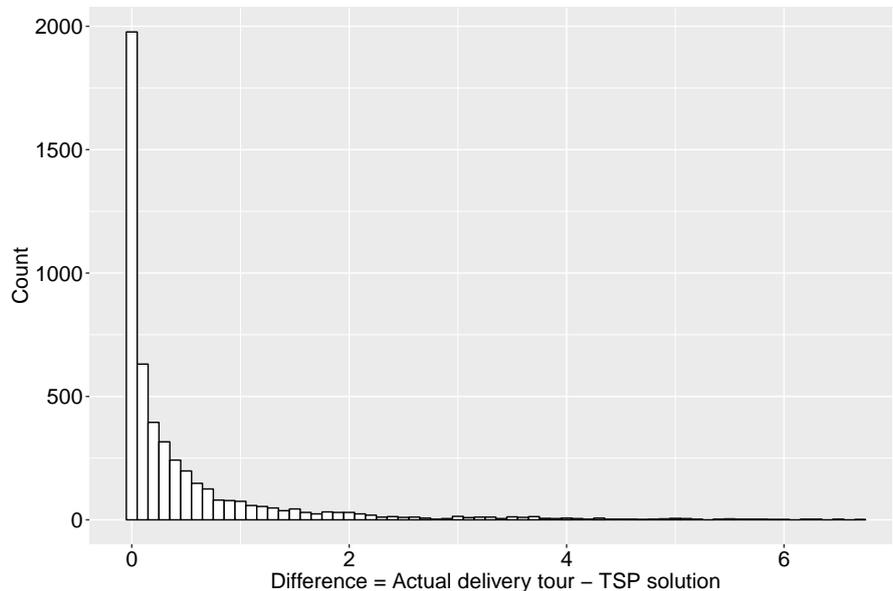


Figure 4 The difference between the actual delivery tour length and TSP solution (in kilometers).

The above result implies that drivers often deviate from the theoretical shortest routes. This phenomenon is also confirmed in empirical studies on driver’s routing behavior (e.g., Lima et al. 2016). There are several possible explanations. First, the TSP formulation does not consider many practical constraints that were transforming the behavior of drivers. For example, some road intersections have limited left turn flows and drivers may choose to avoid these intersections. Second, drivers may prefer some travel patterns over others. For instance, zigzagging routes are found to be undesirable because of the increased possibility of accidents on busy streets (Holland et al.

2017). Third, delivery drivers may adjust the routes based on practical road conditions, weather conditions and customers' updated locations. Modeling all practical constraints and behavior considerations is difficult and incorporating them as numerous constraints into a complicated VRP model is almost intractable. To overcome this difficulty, we utilize machine learning techniques to predict the actual delivery tour length for visiting a set of customer locations without specifying the visiting sequence, which is detailed in the next section.

4. Delivery Tour Prediction

Existing literature has proposed approximate formulas for TSP and VRP with asymptotic results under various scenarios. In particular, when planning a delivery route for a single driver, a VRP reduces to a TSP. Assuming demand locations are independently and uniformly distributed in a square area of area A , the delivery tour length TSP^* of the optimal TSP route satisfies (Beardwood et al. 1959):

$$\lim_{n \rightarrow \infty} \frac{TSP^*}{\sqrt{n}} = \varphi \sqrt{A}, \quad (1)$$

where n is the number of points and φ is a constant. Similar results can be also derived for VRP (see Daganzo 2005 for details). However, the approximate formulas require strong stochastic assumptions and can only yield good results when n is large (Shen and Qi 2007). In food delivery practice, each driver often visits less than 10 different customer locations, and thus makes the approximate formulas inappropriate. Furthermore, the existing result ignores driver's routing behavior discussed in Section 3.3 and lacks the fitting test on real data.

In the following, we develop a data analytics approach to predict the delivery tour length — how far a driver travels in practice. The first step is feature engineering, i.e. we need to come up with features that can capture the main factors influencing the delivery tour. Note that the one-way delivery tour includes 1) the travel distance from the depot to the first visited customer and 2) the travel distance from the first customer to the subsequent customers. For the first part, we take the distance from the depot to the nearest customer location, d , as a proxy. For the second part, we consider the following candidate features motivated by the asymptotic results of TSP, e.g., from Equation (1), and empirical observations:

- n : the number of distinct customer locations;
- a : the maximum latitudinal distance between a pair of realized customer locations;
- b : the maximum longitudinal distance between a pair of realized customer locations;
- $a\sqrt{n-1}$, $b\sqrt{n-1}$, $a(n-1)$, $b(n-1)$ and $\sqrt{ab(n-1)}$;
- s_a : the average inter-customer latitudinal distance;
- s_b : the average inter-customer longitudinal distance.

In total, we have 11 features, including the nearest distance feature d . Measuring a and b in latitudinal and longitudinal distances aligns with the road network in the studied area in Shanghai. For cities with different road orientations, one can apply rotation transformations for proper orientation in the measurement of a and b .

After extracting the above features, the second step is to select the best prediction model based on data. We test a wide range of machine learning models including least absolute shrinkage and selection operator (LASSO), ridge regression, support vector regression and random forest. We make a brief introduction to these four models (for more details, please refer to Friedman et al. 2001). Let β denotes the coefficient vector corresponding to the 11 features. LASSO extends the least squares regression by introducing $p\|\beta\|_1$, a ℓ_1 -norm penalty term of the coefficients β , into the loss minimization problem. The shrinkage parameter $p > 0$ is used to control the sparsity of the model. The greater the value of p , the sparser the resulted coefficients will be. By contrast, ridge regression adds $p\|\beta\|_2^2$, a squared ℓ_2 -norm penalty term of the coefficients β , into the objective function in the least squares regression. Both methods are able to reduce undesired over-fitting by regularization. Support vector regression (SVR) stems from support vector machine (SVM) for classification and uses a different loss function than least squares. It maps data to higher dimension with a kernel function, which facilitates capturing complicated nonlinear relationship. Different from the first three models, random forest runs a number of decision trees that split the feature space into subregions, which then predict the same value for each subregion. To enhance the prediction accuracy, different trees are generated by sampling training data and feature space.

We conduct 5-fold cross-validation to select the best hyper-parameters for the four models (e.g. shrinkage parameters in LASSO and ridge regression, and the number of trees grown in random forest). For SVR, we choose the commonly used radial basis function (RBF) kernel. All training and validation procedures are implemented in R. In Table 2, we report the average cross-validation mean squared errors (MSE)—the average of $\frac{\sum_{s \in \mathcal{S}} (l_s - \hat{l}_s)^2}{|\mathcal{S}|}$, where l_s denotes the actual delivery tour length and \hat{l}_s denotes the predicted delivery tour length for test sample s in each testing fold \mathcal{S} . We also include the MSE of TSP solution as a reference in Table 2.

Table 2 Performance Evaluation of Machine Learning Methods and TSP Solution

| Method | Average MSE |
|------------------|-------------|
| LASSO | 0.314 |
| Ridge regression | 0.317 |
| SVR | 0.295 |
| Random forest | 0.304 |
| TSP solution | 1.002 |

Based on the cross-validation results, we choose LASSO to derive the delivery tour prediction function, for the reasons detailed below.

1. *Accuracy*: LASSO achieves significantly higher prediction accuracy than TSP solution with 68.66% reduction in MSE. It is satisfactory compared to more sophisticated non-linear models, e.g., SVR.

2. *Interpretability*: Due to its simple form — linear in features, LASSO is more interpretable than SVR and random forest, which are often viewed as “black box”. The delivery tour prediction function resulting from LASSO implies the importance of each feature on the tour length.

3. *Tractability*: Its linear functional form also makes LASSO favorable in developing a tractable optimization model for order assignment. The non-linear nature of SVR and random forest will result in excessively complicated optimization formulations, if not intractable. While ridge regression is also a linear model, LASSO has slightly better accuracy and the treatment for the prediction function from LASSO can also be applied to that from ridge regression.

With the tuned shrinkage parameter, LASSO selects 6 features. Figure 5 presents how the cross-validation average MSE of LASSO varies with respect to different values of shrinkage parameter p as well as the resulted different number of features, in which we observe that the best performance is attained with only 6 features summarized below. The delivery tour prediction function from LASSO is (for $n \geq 1$):

$$\beta_0 d + \beta_1 a + \beta_2 b + \beta_3 a \sqrt{n-1} + \beta_4 b \sqrt{n-1} + \beta_5 n, \quad (2)$$

where $\beta_0 = 1.006$, $\beta_1 = 27.69$, $\beta_2 = 79.64$, $\beta_3 = 54.27$, $\beta_4 = 56.51$, $\beta_5 = 0.004$ and the adjusted $R^2 = 0.970$ for the full dataset. Note that the reported value of β is calibrated from the delivery data set. For different applications, one shall estimate β using their corresponding data sets.

4.1. Performance of the Prediction Function

To further examine the prediction performance of LASSO, we discuss its application in predicting the actual delivery tour in practice as well as approximating the TSP tour in theory. The first part shows its practicality in capturing driver’s routing behavior, and the later suggests its potential as a closed-form approximation to TSP tour without solving an optimization program.

Figure 6 presents the mean absolute percentage error (MAPE) of LASSO and the TSP solution in predicting the actual delivery tour lengths with different number of customer locations. MAPE, defined as $\frac{\sum_{s \in \mathcal{S}} |l_s - \hat{l}_s|}{|\mathcal{S}|} \times 100\%$, is unit free and measures the relative prediction errors. We can see that LASSO achieves significantly smaller errors than the TSP solution, especially when there are more customer locations to visit. For the cases with fewer customer locations, e.g., only 2 customer locations, the performances of TSP and LASSO are similar. It is worthwhile to note that the

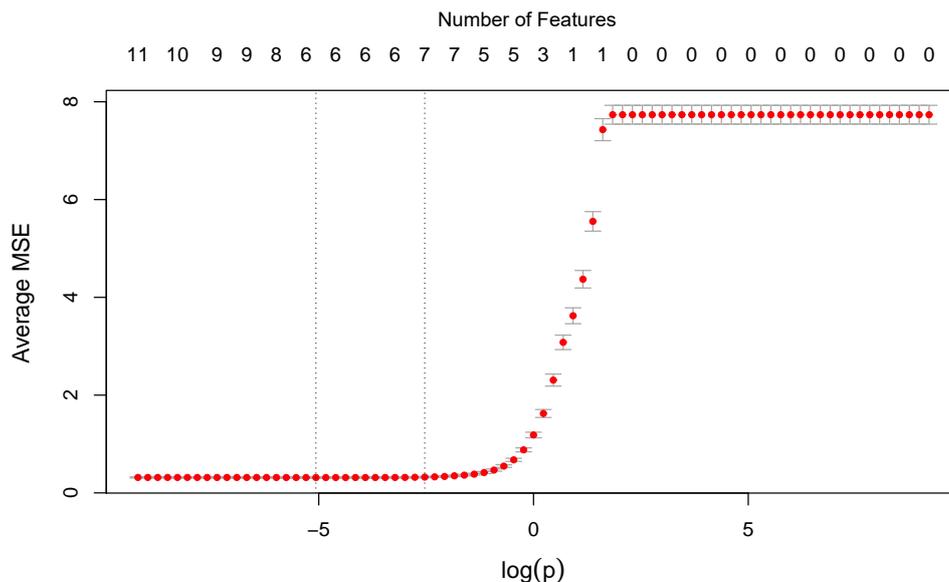


Figure 5 The average MSE of LASSO with different number of features by varying shrinkage parameter p .

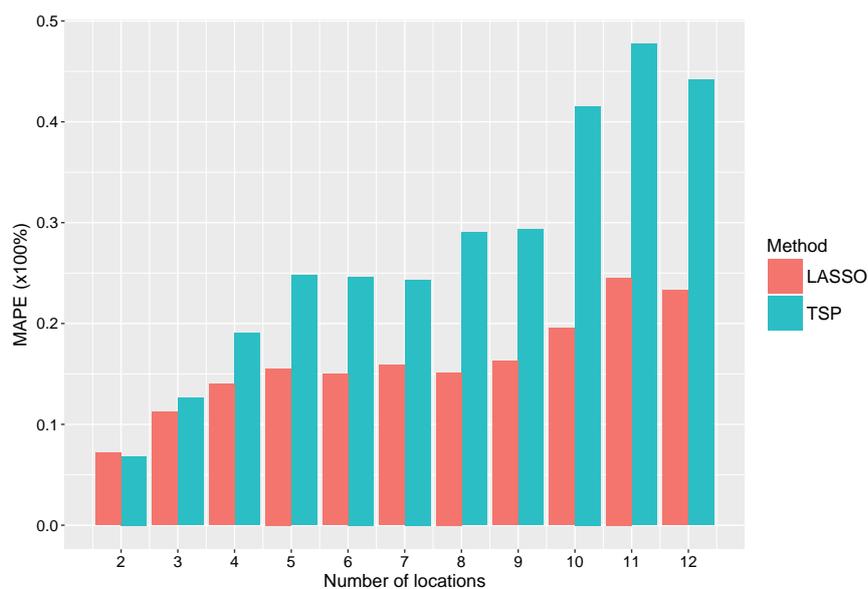


Figure 6 MAPE of LASSO prediction function in (2) versus TSP solution.

MAPE of LASSO is less than 25% and grows mildly with the number of customer locations, while the MAPE of the TSP solution can be close to 50%.

To establish the connection between the prediction function and transportation literature on routing problems, we discuss its approximation performance for TSP tours. We use the same order assignment realizations observed in the delivery data and calibrate their TSP tours. That is, we eliminate driver's routing behavior by assuming the drivers follow TSP tours exactly, e.g., as if

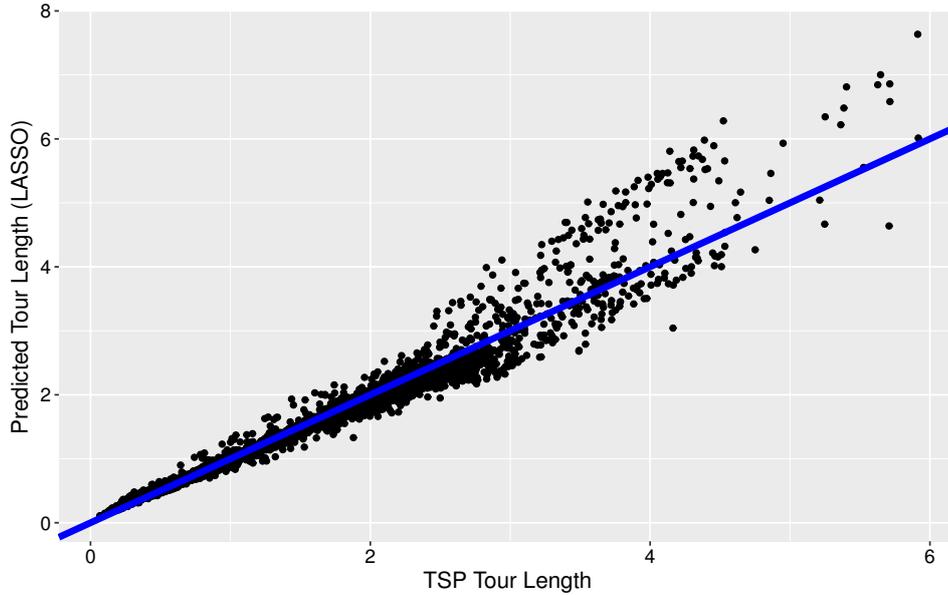


Figure 7 Distances (in km) from LASSO prediction function versus TSP tours

they were autonomous vehicles. Thus, the calibrated dataset shares the same set of customer points and order history with the original delivery dataset. According to the 5-fold cross-validation results, LASSO still selects the same 6 features as in (2), but with a different coefficient vector β resulting from the fitting for TSP tours. Notably, the average MSE drops to 0.037 when fitting LASSO for theoretical TSP tours. Figure 7 shows that the prediction function (2) serves as a good approximation to the TSP tour length, especially for trips within 5km.

The observations above suggest the potential of using the closed-form tour prediction function in various applications involving TSP tours. For example, when a driver's visit sequence is not required, as in the case of order assignment, it greatly simplifies the procedures of obtaining a TSP tour length, without solving a mathematical program. Furthermore, an advantage of the prediction function (2) is that it only requires estimation of 6 parameters, e.g., the coefficients in β . In contrast, a TSP formulation requires the estimation of travel distance (or time) matrix for all pairs of locations, e.g., 11×11 size matrix for 10 customer locations and a depot. Consequently, more estimation errors may be introduced to TSP formulation.

To conclude this section, we remark several further observations. First, as LASSO employs the l_1 -norm regularization, it results in a sparse coefficient vector and requires less features (automatic feature selection), which makes it easy to interpret and integrate with optimization models. Second, unlike the TSP solution that consistently underestimates the delivery tour length, LASSO prediction is much less biased. Third, LASSO is appropriate to estimate the actual delivery tour length even when $n = 1$: it predicts $\beta_0 d + \beta_5 \approx d$ with $\beta_0 \approx 1$ and β_5 is very small. Finally, the

function (2) indeed predicts the average delivery tour length. The actual tour length may vary depending on other spatial characteristic of the realized customer locations, whose deviation from the function (2) can be viewed as the residuals. We treat such prediction as accurate in the basic models and discuss how uncertainty in travel time can be incorporated in Section 5.3. Due to its simple structure and outstanding prediction power, we will adopt the delivery tour prediction function in (2) from LASSO for order assignment optimization in the next section.

5. Order Assignment Models

We consider a firm that operates a single central depot, e.g., kitchen in food service, and assigns its orders to drivers, aiming to minimize the total delay in delivery. For the ease of discussion and implementation, we assume that orders from the same location, e.g., a building or block, are delivered by the same driver. Subsequently, the firm's decision is to assign the customer locations with orders, denoted by \mathcal{I} , to a group of available drivers \mathcal{K} . The drivers freely design their routes to visit the assigned customer locations.

According to our data analytics, a critical part of the uncertainty lies in the service time at each customer location. We model the uncertain service time at location $i \in \mathcal{I}$ using a random variable \tilde{t}_i and denote the joint distribution of \tilde{t}_i among all locations in \mathcal{I} as \mathbb{P} . Let $y_{ik} \in \{0, 1\}$ be the binary decision variable denoting whether customer location i is served by driver k : 1 if i is served by driver k and 0 otherwise. Subsequently, the vector $\mathbf{y}_k = (y_{ik}, \forall i \in \mathcal{I})$ defines the set of locations to be visited by driver k . From the prediction function (2), the delivery tour length by driver k is predicted as:

$$l(\mathbf{y}_k) = \beta_0 d(\mathbf{y}_k) + \beta_1 a(\mathbf{y}_k) + \beta_2 b(\mathbf{y}_k) + \beta_3 a(\mathbf{y}_k) \sqrt{\sum_{i \in \mathcal{I}} y_{ik} - 1} + \beta_4 b(\mathbf{y}_k) \sqrt{\sum_{i \in \mathcal{I}} y_{ik} - 1} + \beta_5 \sum_{i \in \mathcal{I}} y_{ik}, \quad (3)$$

where the terms $d(\mathbf{y}_k)$, $a(\mathbf{y}_k)$ and $b(\mathbf{y}_k)$ are the distance from the depot to the closest customer, the maximum latitudinal and longitudinal distances between a pair of customer locations, respectively, given the customer locations \mathbf{y}_k served by driver k . Since real-time traffic and weather information is usually revealed at the time of order assignment, we assume a constant speed (or velocity) v and predict the travel time by driver k as $\frac{l(\mathbf{y}_k)}{v}$ in the models presented below. Our model can be extended to consider other factors affecting actual travel time, e.g., heterogeneous speed v_k for a specific driver k . We further discuss introducing uncertainty in travel time in Section 5.3.

Given the target delivery time window τ , the order assignment problem is then formulated as the following stochastic program that minimizes the expected total delay of all routes:

$$\min_{y_{ik}} \sum_{k \in \mathcal{K}} \mathbb{E}_{\mathbb{P}} \left[\sum_{i \in \mathcal{I}} \tilde{t}_i y_{ik} + \frac{l_k}{v} - \tau \right]^+$$

$$\begin{aligned}
\text{s.t. } & \sum_{k \in \mathcal{K}} y_{ik} = 1, \quad \forall i \in \mathcal{I}, \\
& l_k = l(\mathbf{y}_k), \quad \forall k \in \mathcal{K}, \\
& y_{ik} \in \{0, 1\}, \forall i \in \mathcal{I}, \quad \forall k \in \mathcal{K},
\end{aligned}$$

where the first set of constraints ensure that each customer location i is served by some driver.

Since evaluating the objective function in the above stochastic program requires full knowledge of the joint service time distribution as well as integration, we deploy the sample average approximation (SAA) scheme. Given a set of historical samples \mathcal{S} , let t_i^s be the service time at location i in sample $s \in \mathcal{S}$. The SAA formulation for order assignment is provided below:

$$\begin{aligned}
\min_{y_{ik}} & \sum_{s \in \mathcal{S}} \sum_{k \in \mathcal{K}} \left[\sum_{i \in \mathcal{I}} t_i^s y_{ik} + \frac{l_k}{v} - \tau \right]^+ \\
\text{s.t. } & \sum_{k \in \mathcal{K}} y_{ik} = 1, \quad \forall i \in \mathcal{I}, \\
& l_k = l(\mathbf{y}_k), \quad \forall k \in \mathcal{K}, \\
& y_{ik} \in \{0, 1\}, \quad \forall i \in \mathcal{I}, k \in \mathcal{K}.
\end{aligned}$$

In this formulation, we drop the constant multiplier $1/|\mathcal{S}|$ for the sample average in the objective. The nonlinear objective function can be linearized by introducing nonnegative auxiliary variables, e.g., ω_k^s with constraints $\omega_k^s \geq \sum_{i \in \mathcal{I}} t_i^s y_{ik} + \frac{l_k}{v} - \tau$ and $\omega_k^s \geq 0$. Thus, the objective function can be rewritten as $\sum_{s \in \mathcal{S}} \sum_{k \in \mathcal{K}} \omega_k^s$. The challenge in this optimization problem is to deal with $l(\mathbf{y}_k)$ for delivery tour prediction.

5.1. Reformulation of $l(\mathbf{y}_k)$

We introduce auxiliary variables a_k , b_k , d_k and n_k to denote $a(\mathbf{y}_k)$, $b(\mathbf{y}_k)$, $d(\mathbf{y}_k)$ and $\sum_{i \in \mathcal{I}} y_{ik}$ respectively, using the following constraints:

$$l_k = \beta_0 d_k + \beta_1 a_k + \beta_2 b_k + \beta_3 a_k \sqrt{n_k - 1} + \beta_4 b_k \sqrt{n_k - 1} + \beta_5 n_k, \quad \forall k \in \mathcal{K}, \quad (4)$$

$$n_k \geq \max \left\{ \sum_i y_{ik}, 1 \right\}, \quad \forall k \in \mathcal{K}, \quad (5)$$

$$a_k \geq |\text{lat}_i - \text{lat}_{i'}| (y_{ik} + y_{i'k} - 1), \quad \forall i, i' \in \mathcal{I}, k \in \mathcal{K}, \quad (6)$$

$$b_k \geq |\text{long}_i - \text{long}_{i'}| (y_{ik} + y_{i'k} - 1), \quad \forall i, i' \in \mathcal{I}, k \in \mathcal{K}, \quad (7)$$

where lat_i and long_i are the latitude and longitude of customer location i .

Note that constraints (6) and (7) are required for all distinct pairs of customer locations. Thus, the number of these constraints $O(I^2 \cdot K)$ can be large, when there are many customer locations. Alternatively, we propose the following equivalent $O(I \cdot K)$ number of constraints:

$$a_k = \bar{a}_k + \underline{a}_k \geq 0, \quad \forall k \in \mathcal{K}, \quad (8)$$

$$b_k = \bar{b}_k + \underline{b}_k \geq 0, \quad \forall k \in \mathcal{K}, \quad (9)$$

$$\bar{a}_k \geq \text{lat}_i \cdot y_{ik}, \quad \forall i \in \mathcal{I}, k \in \mathcal{K}, \quad (10)$$

$$\underline{a}_k \geq -\text{lat}_i + M(y_{ik} - 1), \quad \forall i \in \mathcal{I}, k \in \mathcal{K}, \quad (11)$$

$$\bar{b}_k \geq \text{long}_i \cdot y_{ik}, \quad \forall i \in \mathcal{I}, k \in \mathcal{K} \quad (12)$$

$$\underline{b}_k \geq -\text{long}_i + M(y_{ik} - 1), \quad \forall i \in \mathcal{I}, k \in \mathcal{K}. \quad (13)$$

With the above set of constraints, the nonlinearity remains in Equation (4), e.g., the square root term $\sqrt{n_k - 1}$. Since n_k is an integer with a limited range, we can linearize this nonlinear (and non-convex) term using binary variables. In practice, the number of orders assigned to driver k , i.e., n_k , can not be arbitrarily large. Therefore, we can safely restrict n_k to be within a certain range $\{0, 1, \dots, N_k\}$. One can interpret N_k as the maximum number of orders for driver k , and can be determined by the practitioner or estimated from data. Based on our data set, we observe that the maximum number of orders assigned to a driver in one delivery route is 20 and most routes contain less than 15 orders. In the extreme case, N_k could be the total number of orders to be delivered. We can thus express n_k as a piecewise function with a set of auxiliary binary variables:

$$n_k = \sum_{j=0}^{N_k} j \cdot u_{kj}, \quad (14)$$

$$\sum_{j=0}^{N_k} u_{kj} = 1, \quad (15)$$

$$u_{kj} \in \{0, 1\}, \quad \forall j \in \{0, 1, \dots, N_k\}. \quad (16)$$

Consequently, l_k can be expressed as:

$$l_k = \sum_{j=0}^{N_k} f_{kj},$$

where $f_{k0} = 0$ and

$$f_{kj} \geq \beta_0 d_k + \beta_1 a_k + \beta_2 b_k + \beta_3 a_k \sqrt{j-1} + \beta_4 b_k \sqrt{j-1} + \beta_5 n_k + M(u_{kj} - 1), \quad \forall j \in \{1, \dots, N_k\} \quad (17)$$

and all f_{kj} 's are nonnegative. Alternatively, we can choose the following set of constraints introduced by Glover (1975):

$$D_{kj}^+ u_{kj} \geq f_{kj} \geq D_{kj}^- u_{kj}, \quad (18)$$

$$\beta_0 d_k + \beta_1 a_k + \beta_2 b_k + \beta_3 a_k \sqrt{j-1} + \beta_4 b_k \sqrt{j-1} + \beta_5 n_k - D_{kj}^- (1 - u_{kj}) \geq f_{kj}, \quad (19)$$

$$f_{kj} \geq \beta_0 d_k + \beta_1 a_k + \beta_2 b_k + \beta_3 a_k \sqrt{j-1} + \beta_4 b_k \sqrt{j-1} + \beta_5 n_k - D_{kj}^+ (1 - u_{kj}), \quad (20)$$

where D_{kj}^+ and D_{kj}^- are the upper and lower bounds on $\beta_0 d_k + \beta_1 a_k + \beta_2 b_k + \beta_3 a_k \sqrt{j-1} + \beta_4 b_k \sqrt{j-1} + \beta_5 n_k$, which can be selected based on the data. It is clear that constraints (18)-(20)

are stronger than constraints (17), despite that more constraints are required. By replacing l_k with the above linear functions, we obtain a mixed integer linear program (MILP), which can be readily solvable by commercial optimization solvers. The resulting MILP has $O(K \cdot I + N \cdot K)$ binary variables and $O(K \cdot I \cdot S)$ number of constraints.

We summarize the data-driven order assignment using SAA scheme as the following mixed-integer linear program:

$$\min_Y \sum_{s \in \mathcal{S}} \sum_{k \in \mathcal{K}} \omega_k^s \quad (\text{DOA-SAA})$$

$$\text{s.t. } \omega_k^s \geq \sum_{i \in \mathcal{I}} t_i^s y_{ik} + \frac{l_k}{v} - \tau, \quad \forall k \in \mathcal{K}, s \in \mathcal{S},$$

$$\omega_k^s \geq 0, \quad \forall k \in \mathcal{K}, s \in \mathcal{S},$$

$$\sum_{k \in \mathcal{K}} y_{ik} = 1, \quad \forall i \in \mathcal{I},$$

$$l_k = \sum_{j=0}^{N_k} f_{kj}, \quad \forall k \in \mathcal{K},$$

$$d_k = \sum_{i \in \mathcal{I}} \hat{d}_i x_{ik}, \quad \forall k \in \mathcal{K}, \quad (21)$$

$$\sum_{i' \in \mathcal{I}} x_{i'k} \geq y_{ik}, \quad \forall i \in \mathcal{I}, k \in \mathcal{K}, \quad (22)$$

$$x_{ik} \leq y_{ik}, \quad \forall i \in \mathcal{I}, k \in \mathcal{K}, \quad (23)$$

Constraints (8) – (13),

Constraints (14) – (16), $\forall k \in \mathcal{K}$,

Constraints (18) – (20), $\forall k \in \mathcal{K}, j \in \{1, \dots, N_k\}$,

$y_{ik} \in \{0, 1\}$, $\forall i \in \mathcal{I}, k \in \mathcal{K}$,

where \hat{d}_i is the distance from the depot to customer location i . Constraints (21)-(23) ensure that d_k is the distance from the depot to the nearest customer location in driver k 's route.

5.2. Robust Order Assignment

For the DOA-SAA model to perform well, it requires a large number of samples. However, the observations of service time at many customer locations are sparse in our data. Moreover, its computation time grows if more samples are generated, e.g., via bootstrapping, for the SAA model. To deal with such challenges, we develop a distributionally robust optimization model that utilizes limited distributional information and is independent of sample size in computation.

Suppose the joint distribution \mathbb{P} of service time $\tilde{t}_i, \forall i \in \mathcal{I}$ lies in an ambiguity set \mathbb{F} , such that \mathbb{P} contains limited distributional information, i.e., the marginal mean and variance for each customer location. The support of \tilde{t}_i is set to \mathbb{R} , to allow potentially (although rarely) negative service time.

For instance, a negative service time may denote the case when a customer collects an order without asking a driver to visit the customer location, e.g., with the use of food delivery lockers at more convenient locations to drivers. In particular, we construct the moment ambiguity set \mathbb{F} as below by imposing zero correlation between different customer locations (supported by our independence test described in subsection 3.2) :

$$\mathbb{F} = \left\{ \mathbb{P} \in \mathcal{P}_0(\mathbb{R}^{|\mathcal{I}|}) \left| \begin{array}{l} \mathbb{E}_{\mathbb{P}}(\tilde{t}_i) = \mu_i, \quad \forall i \in \mathcal{I} \\ \mathbb{E}_{\mathbb{P}} \left[(\tilde{t}_i - \mu_i)^2 \right] = \sigma_i^2, \quad \forall i \in \mathcal{I} \\ \mathbb{E}_{\mathbb{P}} \left[(\tilde{t}_i - \mu_i) (\tilde{t}_j - \mu_j) \right] = 0, \quad \forall i \neq j \in \mathcal{I} \end{array} \right. \right\}.$$

To simplify our discussion, we introduce the term $h_k = \frac{l_k}{v} - \tau$ to denote the slack for service time, after considering the routing time in a delivery tour. The distributionally robust optimization (DRO) model that minimizes the worst-case expected total delay is given by:

$$\begin{aligned} \min_Y \max_{\mathbb{P} \in \mathbb{F}} \mathbb{E}_{\mathbb{P}} \sum_{k \in \mathcal{K}} \left[\sum_{i \in \mathcal{I}} \tilde{t}_i y_{ik} + h_k \right]^+ & \quad (24) \\ \text{s.t. } h_k = \frac{l_k}{v} - \tau, \quad \forall k \in \mathcal{K}, & \\ \text{Constraints in DOA-SAA.} & \end{aligned}$$

While the DRO model in (24) involves the joint distribution of all service time \tilde{t}_i , we show in Proposition 1 below that it can be reduced to have a DRO model with an objective of a sum of K separable worst-case expected delays.

PROPOSITION 1. *Let $\tilde{T}_k = \sum_{i \in \mathcal{I}_k} \tilde{t}_i$, where $\mathcal{I}_k = \{i \in \mathcal{I} : y_{ik} = 1\}$, be the projected random variable, i.e., the total service time of driver k . The DRO model in (24) is equivalent to the DRO model as below:*

$$\begin{aligned} \min_Y \sum_{k \in \mathcal{K}} \max_{\mathbb{Q}_k \in \mathbb{G}_k} \mathbb{E}_{\mathbb{Q}_k} \left[\tilde{T}_k + h_k \right]^+ & \quad (25) \\ \text{s.t. } h_k = \frac{l_k}{v} - \tau, \quad \forall k \in \mathcal{K}, & \\ \text{Constraints in DOA-SAA.} & \end{aligned}$$

where \mathbb{Q}_k is in the ambiguity set \mathbb{G}_k , defined by

$$\mathbb{G}_k = \left\{ \mathbb{Q}_k \in \mathcal{P}_0(\mathbb{R}) \left| \begin{array}{l} \mathbb{E}_{\mathbb{Q}_k}(\tilde{T}_k) = \sum_{i \in \mathcal{I}_k} \mu_i \\ \mathbb{E}_{\mathbb{Q}_k} \left[\left(\tilde{T}_k - \sum_{i \in \mathcal{I}_k} \mu_i \right)^2 \right] = \sum_{i \in \mathcal{I}_k} \sigma_i^2 \end{array} \right. \right\}.$$

Proof We first show that the inner maximization problem for the worst-case expected total delay can be separable. Let \mathbb{P}_k be the joint distribution of the service time at locations assigned

to driver k , i.e., $\{\tilde{t}_i\}_{\mathcal{I}_k}$ where $\mathcal{I}_k = \{i \in \mathcal{I} : y_{ik} = 1\}$. Correspondingly, we consider the ambiguity set \mathbb{F}_k as the projection of \mathbb{F} on \mathbb{P}_k . That is, \mathbb{F}_k is specified as

$$\mathbb{F}_k = \left\{ \mathbb{P}_k \in \mathcal{P}_0(\mathbb{R}^{|\mathcal{I}_k|}) \left| \begin{array}{ll} \mathbb{E}_{\mathbb{P}_k}(\tilde{t}_i) = \mu_i, & \forall i \in \mathcal{I}_k \\ \mathbb{E}_{\mathbb{P}_k}[(\tilde{t}_i - \mu_i)^2] = \sigma_i^2, & \forall i \in \mathcal{I}_k \\ \mathbb{E}_{\mathbb{P}_k}[(\tilde{t}_i - \mu_i)(\tilde{t}_j - \mu_j)] = 0, & \forall i \neq j \in \mathcal{I}_k \end{array} \right. \right\}.$$

Note that each location is assigned to a single driver. The collection of $\mathcal{I}_k, \forall k \in \mathcal{K}$ is a partition of \mathcal{I} , i.e., $\cup_{k \in \mathcal{K}} \mathcal{I}_k = \mathcal{I}$ and $\mathcal{I}_k \cap \mathcal{I}_{k'} = \emptyset$ for $k \neq k' \in \mathcal{K}$. Therefore, for any given $\mathbb{P}_k \in \mathbb{F}_k$ for all $k \in \mathcal{K}$, we can construct the joint distribution of service times at all locations as $\mathbb{P} = \prod_{k \in \mathcal{K}} \mathbb{P}_k$. It is straightforward to see that the constructed \mathbb{P} belongs to the ambiguity set \mathbb{F} . Hence, the inner maximization problem in (24) can be reformulated in the following steps:

$$\begin{aligned} & \max_{\mathbb{P} \in \mathbb{F}} \mathbb{E}_{\mathbb{P}} \sum_{k \in \mathcal{K}} \left[\sum_{i \in \mathcal{I}} \tilde{t}_i y_{ik} + h_k \right]^+ \\ &= \max_{\mathbb{P} \in \mathbb{F}} \sum_{k \in \mathcal{K}} \mathbb{E}_{\mathbb{P}} \left[\sum_{i \in \mathcal{I}_k} \tilde{t}_i + h_k \right]^+ \\ &= \max_{\mathbb{P}_k \in \mathbb{F}_k, \forall k \in \mathcal{K}} \sum_{k \in \mathcal{K}} \mathbb{E}_{\mathbb{P}_k} \left[\sum_{i \in \mathcal{I}_k} \tilde{t}_i + h_k \right]^+ \\ &= \sum_{k \in \mathcal{K}} \max_{\mathbb{P}_k \in \mathbb{F}_k} \mathbb{E}_{\mathbb{P}_k} \left[\sum_{i \in \mathcal{I}_k} \tilde{t}_i + h_k \right]^+. \end{aligned}$$

Here, the first equality follows from the linearity of expectation and the definition of $\mathcal{I}_k = \{i \in \mathcal{I} : y_{ik} = 1\}$. The second equality is established based on the previous argument—for any given $\mathbb{P}_k \in \mathbb{F}_k$, we can construct $\mathbb{P} = \prod_{k \in \mathcal{K}} \mathbb{P}_k \in \mathbb{F}$, while for any given $\mathbb{P} \in \mathbb{F}$, we can construct $\mathbb{P}_k \in \mathbb{F}_k$ by projection. The third equality is based on separability of the objective function by the decision variables \mathbb{P}_k .

Let $\tilde{T}_k = \sum_{i \in \mathcal{I}_k} \tilde{t}_i$, be the projected random variable. From Proposition 1 in Popescu (2007), the inner maximization problem can be further reformulated as

$$\max_{\mathbb{P}_k \in \mathbb{F}_k} \mathbb{E}_{\mathbb{P}_k} \left[\sum_{i \in \mathcal{I}_k} \tilde{t}_i + h_k \right]^+ = \max_{\mathbb{Q}_k \in \mathbb{G}_k} \mathbb{E}_{\mathbb{Q}_k} \left[\tilde{T}_k + h_k \right]^+,$$

where

$$\mathbb{G}_k = \left\{ \mathbb{Q}_k \in \mathcal{P}_0(\mathbb{R}) \left| \begin{array}{l} \mathbb{E}_{\mathbb{Q}_k}(\tilde{T}_k) = \sum_{i \in \mathcal{I}_k} \mu_i \\ \mathbb{E}_{\mathbb{Q}_k} \left[\left(\tilde{T}_k - \sum_{i \in \mathcal{I}_k} \mu_i \right)^2 \right] = \sum_{i \in \mathcal{I}_k} \sigma_i^2 \end{array} \right. \right\}.$$

Therefore, for each inner maximization problem, we can instead solve the corresponding “projected” problem over the class of univariate distributions in \mathbb{G}_k . It then leads to the DRO model provided in (25). \square

To derive a tractable formulation to problem (25), we deal with $\max_{\mathbb{P}_k \in \mathbb{F}_k} \mathbb{E}_{\mathbb{P}_k} \left[\tilde{T}_k + h_k \right]^+$ in the following lemma.

LEMMA 1. *The inner problem $\max_{\mathbb{P}_k \in \mathbb{F}_k} \mathbb{E}_{\mathbb{P}_k} \left[\tilde{T}_k + h_k \right]^+$ can be solved by the following equivalent optimization problem with second-order conic constraints:*

$$\begin{aligned}
& \min_{\lambda_k, \eta_k, \theta_k} \quad \lambda_k + \eta_k \sum_{i \in \mathcal{I}} \mu_i y_{ik} + \theta_k \sum_{i \in \mathcal{I}} \sigma_i^2 y_{ik} \\
& \text{s.t.} \quad \lambda_k + (\eta_k - 1) \sum_{i \in \mathcal{I}} \mu_i y_{ik} - h_k + \theta_k \geq \left\| \left(\lambda_k + (\eta_k - 1) \sum_{i \in \mathcal{I}} \mu_i y_{ik} - h_k - \theta_k \right) \right\|_2 \\
& \quad \lambda_k + (\eta_k - 1) \sum_{i \in \mathcal{I}} \mu_i y_{ik} - h_k \geq 0 \\
& \quad \lambda_k + \eta_k \sum_{i \in \mathcal{I}} \mu_i y_{ik} + \theta_k \geq \left\| \left(\lambda_k + \eta_k \sum_{i \in \mathcal{I}} \mu_i y_{ik} - \theta_k \right) \right\|_2 \\
& \quad \left(\lambda_k + \eta_k \sum_{i \in \mathcal{I}} \mu_i y_{ik} + \theta_k \right)^2 \geq \left(\lambda_k + \eta_k \sum_{i \in \mathcal{I}} \mu_i y_{ik} - \theta_k \right)^2 + \eta_k^2, \\
& \quad \theta_k \geq 0.
\end{aligned}$$

Based on Lemma 1, we can formulate the distributionally robust order assignment as a tractable second order conic program (the proof is available in the Appendix):

PROPOSITION 2. *The distributionally robust order assignment policy can be solved by the following mixed-integer second order conic program (MISOCP):*

$$\begin{aligned}
& \min_{\mathbf{Y}, \boldsymbol{\rho}} \quad \sum_{k \in \mathcal{K}} \left[\rho_k + \sum_{i \in \mathcal{I}} \mu_i y_{ik} + h_k \right] & \text{(DOA-DRO)} \\
& \text{s.t.} \quad h_k = \frac{l_k}{v} - \tau, \quad \forall k \in \mathcal{K}, \\
& \quad \rho_k \geq \left\| \begin{pmatrix} \sigma_1 y_{1k} \\ \vdots \\ \sigma_I y_{Ik} \\ \sum_{i \in \mathcal{I}} \mu_i y_{ik} + h_k \end{pmatrix} \right\|_2, \quad \forall k \in \mathcal{K}, \\
& \quad \text{Constraints in DOA-SAA.}
\end{aligned}$$

The problem DOA-DRO has K second order conic constraints. Since our problem has natural independence in service times among the drivers, we can decompose the objective function as well as the ambiguity set, and derive an efficient model. For the general distributionally robust optimization model without imposing zero correlation, we can still derive a decomposable formulation but with more conic constraints.

5.3. Uncertain Travel Time

In the analytics and models above, we consider the primary uncertainty in service times at customer locations and develop prediction function for travel distance, which can be translated into travel time with a known speed. We note that the function (2) predicts the average travel distance by a driver. Therefore, it may be necessary to introduce uncertainty in travel time, when the traffic information is less reliable and driver's routing behavior is heterogeneous.

Here we discuss a simple remedy to deal with uncertain travel time. Since the uncertainty in travel time may come from the uncertainty in travel distance or that in travel speed, we denote the uncertain travel time by driver k as \tilde{r}_k . Its deviation from the predicted travel time $\frac{l_k}{v}$ is thus defined by $\varepsilon_k = \tilde{r}_k - \frac{l_k}{v}$. Suppose the provider has knowledge about the realized travel time \tilde{r}_k in its delivery data, one can obtain an empirical distribution of ε_k for the DOA-SAA model and estimate the moment information of ε_k for the DOA-DRO model. We can replace the terms $\frac{l_k}{v}$ with $\frac{l_k}{v} + \varepsilon_k$ in both DOA models and deal with ε_k as an additional random variable with the same treatment of uncertain service time \tilde{t}_i . Therefore, both DOA models can handle the uncertain travel time, while keeping the same forms as presented above.

6. Branch-and-Price Algorithm

The DOA models formulated in Section 5 can be solved in commercial mixed integer solvers such as Gurobi and CPLEX. However, as the number of locations (I) and the number of drivers (K) grow, the assignment problem becomes increasingly hard to solve. Furthermore, order assignment decision for food service is usually required be made quickly (e.g., 20 minutes) in practice. Thus, the standard MILP/MISOCP formulation may fail to deliver good quality solutions in time. To overcome this issue, we utilize the structure of the order assignment problem and formulate it as a set partitioning problem, which allows us to develop an efficient branch-and-price algorithm.

6.1. Set Partitioning Formulation

Let the subset of locations assigned to a driver be associated with a cost, which is the expected delay of delivering to this subset of locations. The order assignment problem is to find a partitioning of locations to subsets such that the total cost (expected total delay) is minimized. As a result, the order assignment problem can be stated as the set partitioning master problem (MP):

$$\min_{\mathbf{z}} \sum_{j \in \mathcal{J}} c_j z_j \quad (\text{MP})$$

$$\text{s.t.} \quad \sum_{j \in \mathcal{J}} \delta_{ij} z_j = 1, \quad \forall i \in \mathcal{I}, \quad (26)$$

$$\sum_{j \in \mathcal{J}} z_j = K, \quad (27)$$

$$z_j \in \{0, 1\}, \quad \forall j \in \mathcal{J},$$

where \mathcal{J} is the set of all possible subsets satisfying cardinality constraints. δ_{ij} is 1 if location i belongs to subset j , and 0 otherwise. Constraints (26) ensure that every location is covered by a subset and constraint (27) ensures the number of selected subsets equal to K . For SAA, the subset cost c_j is the sample average delay of subset j . For the robust counterpart, c_j is the worst-case expected delay of subset j given the first and second moment information.

MP involves an exponential number of variables (columns) since the number of possible subsets grows exponentially in the number of locations. Instead of enumerating all possible subsets and solve the entire MP, column generation provides a way to generate candidate subsets and promises to solve the LP relaxation of MP when a stop routine is invoked. Specifically, column generation solves the following pricing subproblem at each iteration:

$$\begin{aligned} \min_{\bar{\mathbf{y}}} \quad & -\pi_o - \sum_{i \in \mathcal{I}} \pi_i \bar{y}_i + c(\bar{\mathbf{y}}) & (\text{SP}) \\ \text{s.t.} \quad & \text{Constraints in DOA-SAA or DOA-DRO,} \end{aligned}$$

where π_o is the dual value for constraint (27) and π_i 's are the dual values for constraints (26) in MP. The constraints in SP are similar to those in DOA-SAA and DOA-DRO except that the subscript k is dropped. Note that in the case of DOA-DRO, the subproblem is also a MISOCP but with less binary variables and only one conic constraint. Thus, the subproblem can be solved efficiently. If the optimal objective of SP is negative, we add to MP the subset corresponding to $\bar{\mathbf{y}}^*$, i.e., $\{i \in \mathcal{I} : \bar{y}_i^* = 1\}$, and then solve the new MP. We repeat this process until the optimal objective of SP is nonnegative, implying that the LP relaxation of MP is solved. For more details about set partitioning and column generation, we refer readers to Barnhart et al. (1998)

6.2. Algorithm Overview

We start the algorithm from an initial pool of subsets that is a feasible partitioning of customer locations. The initial partitioning is found by solving the order assignment problem by replacing the delivery tour length with its lower bound $(a + b + d)$, i.e., the minimum delivery tour length required to cover all customer locations. This simpler problem removes many hard constraints and allows us to find a good feasible partitioning quickly.

The optimal solution from solving the LP relaxation of MP with column generation is not necessarily integral and applying a branching rule is required. However, a standard branching rule that sets a fractional variable to $\{0, 1\}$ can fail, since preventing a subset j from reappearing in SP (corresponding to setting $z_j = 0$) is difficult. To resolve this issue, we adopt the branching rule introduced by Ryan and Foster (1981) that can be added to SP in branches. Basically, this

branching rule detects a pair of subsets S_1, S_2 with fractional values in the solution, and two locations i, j with one in $S_1 \cap S_2$ and the other in S_1/S_2 . In the two branches, the left branch adds constraints such that i and j can only be in the same subset while the right branch adds constraints such that i and j can not belong to the same subset.

Following the suggestions of Mehrotra et al. (1998), we use a depth-first-search (DFS) to select the next node to solve. Also, for all nodes except the root node, we do not solve the LP relaxation to optimality. Instead, we stop column generation once the objective value of the LP relaxation is less than the objective value at the root node.

6.3. Comparison to MILP Formulation

To demonstrate the efficiency of branch-and-price algorithm, we generate instances from real data by varying parameter settings. We run experiments in Python 3.6 using Gurobi 7, on a 3.50 GHz Xeon CPU. We choose the termination criterion to be (i) optimality gap is below 1%, or (ii) CPU time exceeds 20 minutes.

Table 3 compares the performance of solving DOA-SAA using the standard MILP formulation and the branch-and-price algorithm for a range of instances with 30 samples. The t column presents solution times and Z_{MILP} , Z_{BP} columns report the objective values derived with MILP and branch-and-price algorithm at the termination, respectively. Unlike the standard MILP formulation that fails to solve 4 out of 6 instances, branch-and-price algorithm scales well and solves all 6 instances within 600 seconds. Table 4 reports the computational results of solving DOA-DRO with branch-and-price algorithm and a standard MISOCP formulation. The last column of Table 4 also reports the optimality gap of branch-and-price algorithm if computational time limit is hit. Still, branch-and-price algorithm delivers superior performance in terms of both solution time and solution quality. Branch-and-price algorithm can find a good quality solution (optimality gap $\leq 3\%$) for the robust model within 20 minutes while the standard MISOCP formulation fails in all instances.

Table 3 Solution time (in CPU seconds) and objective values of MILP and branch-and-price for DOA-SAA (* indicates computation stopped at 20-min time limit)

| Locations | Drivers | Time Window (in minutes) | MILP | | Branch-and-Price | |
|-----------|---------|-----------------------------|-------|-------------------|------------------|-----------------|
| | | | t | Z_{MILP} | t | Z_{BP} |
| 29 | 7 | 80 | 2 | 0.00 | 9 | 0.00 |
| 29 | 7 | 60 | 1200* | 61.31 | 151 | 42.78 |
| 37 | 8 | 80 | 1200* | 90.20 | 86 | 40.40 |
| 37 | 8 | 60 | 1200* | 873.23 | 126 | 571.26 |
| 42 | 9 | 80 | 37 | 0.00 | 29 | 0.00 |
| 42 | 9 | 60 | 1200* | 471.04 | 521 | 284.31 |

Table 4 Solution time (in CPU Seconds) and objective values of MISOCP and branch-and-price for DOA-DRO (* indicates computation stopped at 20-min time limit)

| Locations | Drivers | Time Window (in minutes) | MISOCP | | Branch-and-Price | |
|-----------|---------|-----------------------------|--------|-------------------|------------------|-----------------------|
| | | | t | Z_{MILP} | t | Z_{BP} (Gap) |
| 29 | 7 | 80 | 1200* | 8.45 | 365 | 6.98 |
| 29 | 7 | 60 | 1200* | 61.31 | 151 | 42.78 |
| 37 | 8 | 80 | 1200* | 37.81 | 1200* | 18.43 (3%) |
| 37 | 8 | 60 | 1200* | 64.80 | 1200* | 45.93 (3%) |
| 42 | 9 | 80 | 1200* | 15.82 | 1013 | 11.36 |
| 42 | 9 | 60 | 1200* | 48.02 | 1200* | 27.31 (3%) |

7. Numerical Study

In this section we first discuss the setup of our numerical study using a delivery data set from a food service provider in Shanghai. We then employ a machine learning approach to evaluate the out-of-sample performance of the proposed DOA models with benchmarks. We discuss the detailed experiment setup as follows.

Data: We consider the batch of orders placed between 10 : 15 am and 10 : 30 am, and were promised to be delivered by 11 : 45 am, which corresponds to the typical busy period during a day. Across 35 sample days, we have 7,043 orders and 1,048 different customer locations observed in the study period. The batch of orders on each sample day is used in each experiment instance, where we apply our models and benchmarks.

Preprocessing: Based on the customer locations, we calculate the ℓ_1 -norm distance between each pair of them as the proximity measure. Among the resulting 1,048 customer locations, some locations belong to the same building but at different floors. Therefore, these locations can be considered as one single customer location representing a building (or a plaza). Also, in the densely populated area of Shanghai, it is often economical to assign orders coming from nearby buildings to one driver instead of multiple drivers. Motivated by these observations, we apply a clustering method—the minimax linkage hierarchical clustering, to cluster customer locations and meanwhile manage the problem size. Minimax hierarchical clustering is a type of agglomerative clustering algorithms that build trees in a bottom-up approach. It starts with singletons and gradually merges two closest clusters stage by stage, until only one cluster remains. The resulting binary tree is commonly displayed as a dendrogram, as shown in Figure 8. Each leaf node represents a data point and is placed at the bottom with height 0. Each interior node indicates a merging and the corresponding height is equal to the distance between the clusters merged at that node. A key choice for such clustering algorithms is the distance measure between two clusters. Common measures include complete, single, average, and centroid linkage (see Friedman et al. 2001 for example). The minimax hierarchical clustering uses a different linkage:

$$d(G, H) = \min_{x \in G \cup H} d_{\max}(x, G \cup H),$$

where G and H are clusters and $d_{\max}(x, G \cup H)$ is the maximum distance between the point x and a point in cluster $G \cup H$. This linkage measures the minimax radius of the resulting merged cluster. The most central point of the merged cluster (in terms of minimizing d_{\max}) is called prototype. Minimax hierarchical clustering is proved to have many desirable properties including interpretability and robustness (see Bien and Tibshirani 2011 for more details). In our applications, we can restrict the minimax radius of the obtained clusters by cutting the dendrogram at a certain height. Thus, each resulting cluster contains only locations that are within a certain distance from each other.

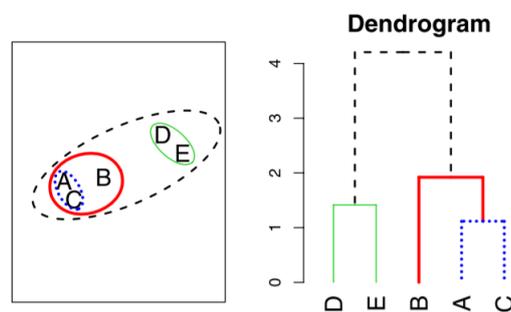


Figure 8 A dendrogram with 5 locations (Bien and Tibshirani 2011)

We choose to cut the dendrogram at height 0.15, which implies that the radius of the resulting clusters can not exceed 0.15 km. It corresponds to the average street block size in the inner ring of Shanghai (Pan and Cao 2015). Therefore, we obtain a total of 97 clustered locations, with each cluster roughly represents a block. On a typical day, orders are received from 40 out of these 97 clusters in the busy period.

Sample Generation and Estimation : An important input to SAA models is the set of samples of service times at different locations. We generate samples by bootstrapping from observed set of service times at each (clustered) location. As the distribution of service time does not exhibit normal behavior, bootstrapping is helpful in approximating the distribution without analytical functional assumptions. In the case of robust model, sample mean and variance are estimated from observations to construct the ambiguity set.

7.1. Model Evaluation

We compare the performance of the assignment policy produced by the DOA models with two benchmark models that assume drivers follow the shortest-distance routes. With the shortest route

assumption, the order assignment model essentially becomes a stochastic vehicle routing problem with stochastic service times (SVRP). Similar to the DOA models, we can formulate a SAA model of SVRP and its distributionally robust counterpart to minimize the worst-case expected delays. We refer to the benchmarks as VRP-based models — VRP-SAA for a SAA model and VRP-DRO for a DRO model. Note that our calculation of delay does not account for the return trip from the last visited location to the depot. As discussed in Section 3.3, we introduce a dummy node into the location network, which helps drop the return trip from the route. We employ the commonly used illegal subtour elimination scheme in solving SVRP (see Laporte et al. 1992 for more details). Both the benchmark VRP-SAA and VRP-DRO models are solved with the branch-and-price algorithm, which has been shown to be an efficient exact method of solving VRPs (Fukasawa et al. 2006). Please refer to the detailed formulations of VRP-SAA and VRP-DRO in the Appendix.

To assess the out-of-sample performance of order assignment decisions, we construct validation sets by drawing independent samples from historical observations. We do not estimate the probability density functions as the number of observations at some locations are very limited, which makes many parametric and nonparametric estimation methods inappropriate. For each problem instance in the numerical test, we generate 1,000 validation sets.

In the evaluation of delays from implementing proposed solutions in the validation sets, we need to estimate the actual delivery tour length for a set of locations, which we may have not observed in history. Since driver’s behavior is hard to simulate, we rely on machine learning models to accomplish this task. Based on the cross validation result in Table 2 of Section 3, support vector regression is the leading model that gives the lowest error. Hence, we apply support vector regression to estimate the actual delivery tour in various scenarios.

7.2. Results and Discussion

In this subsection, we report the performance of the DOA models versus the VRP-based models on various instances. Generated instances vary by the number of customer locations and the number of available drivers, with each combination represents a specific day in the study period. We first apply a relatively small sample size of 30 for the SAA approach, such that those instances can be solved within 20 minutes for both DOA-SAA and VRP-SAA, and then discuss the impact of the sample size on their solution quality. We also test the performance of all models with different delivery time windows. A tighter delivery time window corresponds to the case when the provider takes urgent orders, or when food preparation delay occurs and less time is left for last mile delivery.

7.2.1. Performance Comparison. Table 5 and Table 6 present the average total delivery delay produced by different order assignment models with a delivery time window of 80 minutes and 60 minutes, respectively. For the instances with 80-minute delivery time window, DOA-DRO

gives the lowest average delay among all models. In particular, it achieves smaller average delays than VRP-DRO in all instances, with improvement varying from 33% to 70%. Similarly, DOA-SAA outperforms VRP-SAA substantially, reducing the average delay in most of the instances by 10% to 74%. Such performance gap is also observed in Table 6 for a relatively shorter delivery time window. This implies the importance of incorporating driver’s routing behavior, which is ignored in the VRP-based models. Note that for instances with a 60-minute delivery time window, DOA-DRO still achieves the best performance overall but does not always dominate DOA-SAA. As SAA models only consider a small number of samples in the experiments, their performance can get better if the sample size becomes larger to capture more distributional information of the service time. Table 7 summarizes the performance gap between DOA-SAA and VRP-SAA, and between DOA-DRO and VRP-DRO.

Table 5 Average Performance of Different Models (80-minute Delivery Time Window)

| Locations | Drivers | Average Total Delay | | | |
|-----------|---------|---------------------|---------|---------|---------|
| | | DOA-SAA | VRP-SAA | DOA-DRO | VRP-DRO |
| 26 | 6 | 3.063 | 11.899 | 2.115 | 3.152 |
| 29 | 7 | 3.782 | 2.824 | 0.951 | 2.610 |
| 32 | 7 | 4.538 | 12.881 | 3.469 | 7.059 |
| 33 | 7 | 3.875 | 4.321 | 2.617 | 4.463 |
| 36 | 8 | 2.090 | 7.129 | 1.371 | 3.189 |
| 37 | 8 | 1.291 | 2.158 | 0.231 | 0.762 |
| 38 | 8 | 9.725 | 13.067 | 5.276 | 8.220 |
| 40 | 8 | 3.679 | 5.576 | 1.604 | 4.479 |
| 42 | 9 | 6.527 | 8.163 | 1.652 | 4.093 |

Table 6 Average Performance of Different Models (60-minute Delivery Time Window)

| Locations | Drivers | Average Total Delay | | | |
|-----------|---------|---------------------|---------|---------|---------|
| | | DOA-SAA | VRP-SAA | DOA-DRO | VRP-DRO |
| 26 | 6 | 21.327 | 32.659 | 20.114 | 27.363 |
| 29 | 7 | 9.036 | 22.743 | 8.535 | 15.486 |
| 32 | 7 | 20.364 | 32.677 | 25.123 | 27.341 |
| 33 | 7 | 19.985 | 30.555 | 16.976 | 23.037 |
| 36 | 8 | 14.031 | 20.726 | 12.897 | 22.975 |
| 37 | 8 | 8.860 | 9.012 | 7.014 | 9.628 |
| 38 | 8 | 36.340 | 53.495 | 34.582 | 54.515 |
| 40 | 8 | 22.817 | 43.635 | 30.643 | 30.643 |
| 42 | 9 | 21.832 | 37.731 | 20.650 | 28.211 |

To understand why VRP-based models fail to deliver satisfactory results, we compare the assignment decisions produced by DOA-SAA and VRP-SAA for the 26-location instance. The evaluation

Table 7 Average Performance Gap Between Data-driven and VRP-Based Models

| Time Window | SAA | | DRO | |
|-------------|--------|--------|--------|--------|
| | Mean | Max | Mean | Max |
| 80-minute | 33.99% | 74.26% | 52.78% | 69.72% |
| 60-minute | 35.91% | 60.27% | 26.69% | 44.89% |

on the validation set shows that the average total delay is 3.06 minutes for DOA-SAA and 11.90 minutes for VRP-SAA. The main contributors to the delay of VRP-SAA are two drivers, namely A and B, who experience average delay of 5.60 minutes and 4.83 minutes, respectively. Based on the validation result, driver A actually travels 7.115 km (28.46 minutes) and driver B actually travels 7.057 km (28.23 minutes) to visit their assigned customers. However, under the modeling assumption of VRP-SAA, the two drivers would only travel 3.619 km (14.48 minutes) and 4.127 km (16.51 minutes). Therefore, VRP-SAA underestimates their travel time and misleads the assignment decision. By contrast, DOA-SAA foresees the longer actual delivery tours and assigns the two drivers differently from VRP-SAA to avoid severe delays. As a result, the average delay of driver A and B following the assignment decision from DOA-SAA is reduced to only 0.08 minutes and 0.02 minutes, respectively.

7.2.2. The Impact of Sample Size. The performance of SAA depends on the number of samples used to approximate the expected objective function value. To obtain a good quality solution, the number of samples is suggested to be growing logarithmically on the size of the feasible set (Kleywegt et al. 2002). However, growing number of samples may make the problem computationally inefficient for our application. Furthermore, since the real delivery tour is unknown, our approximated objective function may not approach the true expected objective even when the number of samples is sufficiently large. To understand the impact of sample size on the performance of SAA models, we calculate the average total delay of SAA models on the validation sets by varying sample sizes, as shown in Figure 9. For the two illustrated instances, DOA-SAA achieves lower average delay than DOA-DRO, which serves as a reference and is independent of the sample size, when the number of samples gets as large as 180. However, the performance of VRP-SAA does not necessarily improve as the sample size grows. For the second instance in Figure 9, the average delay of VRP-SAA becomes larger as the sample size increases from 30 to 120. One possible reason is that the objective value of VRP-SAA becomes more biased when the number of samples becomes larger, again due to the ignorance of driver’s routing behavior. Consequently, the solution to VRP-SAA can “overfit” to the samples and deviate from the true optimal solution. In contrast, the objective value of DOA-SAA is less biased and is closer to the true objective value. Thus, its solution is able to approach the true optimal solution for large sample size.

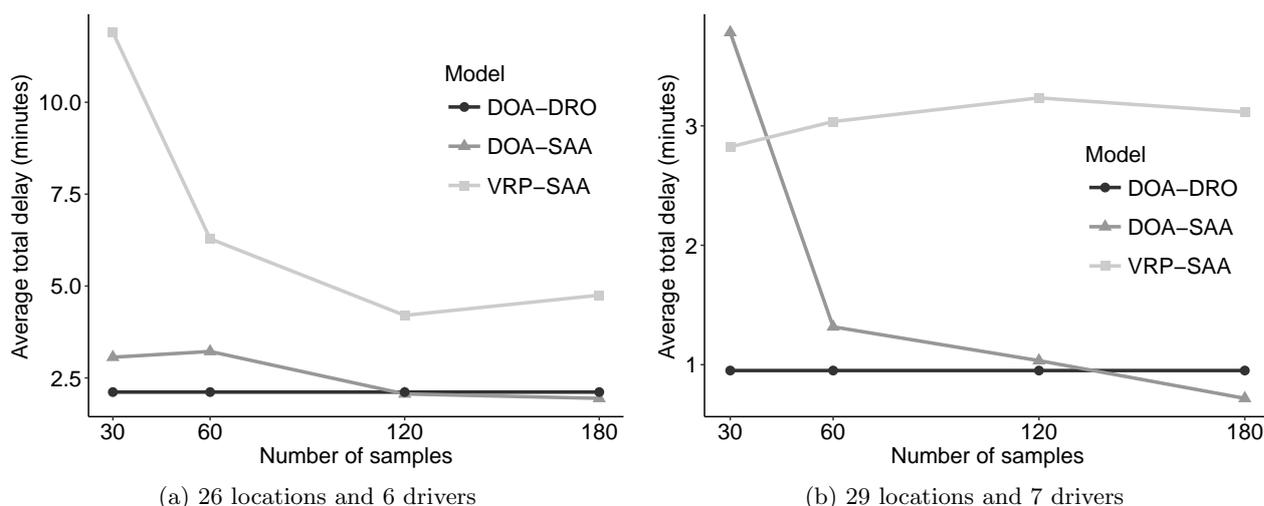


Figure 9 Impact of sample size on SAA models (delivery time is 80 minutes)

7.2.3. Staffing Considerations. One way to reduce the delivery delay is to expand the fleet size by hiring more drivers. To determine the optimal staffing level, the company needs to weigh the benefit from reduced delivery delays with increased wage payment to drivers. Based on the DOA models, we can evaluate the delivery performance with different staffing levels. Figure 10 shows how the average total delay changes with the number of dispatched drivers in two SAA models: DOA-SAA and VRP-SAA with sample size of 180. As more drivers are dispatched, the order assignment solution from DOA-SAA yields lower average delivery delay. When the delivery time window is 80 minutes, DOA-SAA's solution achieves an average total delay that is less than 5 minutes with only 6 drivers, for both instances of 29 and 26 locations. When the allowable time window reduces to 60 minutes, DOA-SAA needs 8 drivers to ensure the average delay does not exceed 5 minutes. However, applying VRP-SAA can cause unnecessary overstaffing for achieving the same service goal. As shown in Figure 10b and 10d, VRP-SAA requires 10 and 9 drivers to reduce the average total delay to less than 5 minutes for both instances, which are higher than the corresponding staffing levels according to DOA-SAA. In addition, increasing the staffing level does not always lead to improved delivery performance for VRP-SAA. As observed in Figure 10a and 10b, the performance of VRP-SAA is even worsen off as a result of adding more drivers. This result seems counterintuitive, but as we argue before, optimizing a biased objective function can give very poor solutions. Recall that the performance is evaluated out-of-sample. It is likely that the benefit from adding a driver is offset by the loss from a worse order assignment decision from a biased model. Consequently, it is indeed necessary to employ DOA-SAA, instead of VRP-SAA, to capture driver's routing behavior in the order assignment and staffing decisions. Finally, by comparing Figure 10a and 10b (and also Figure 10c and 10d), the performance gap between DOA

and VRP-based models widens under more stringent delivery requirement— with less drivers and shorter delivery time window.

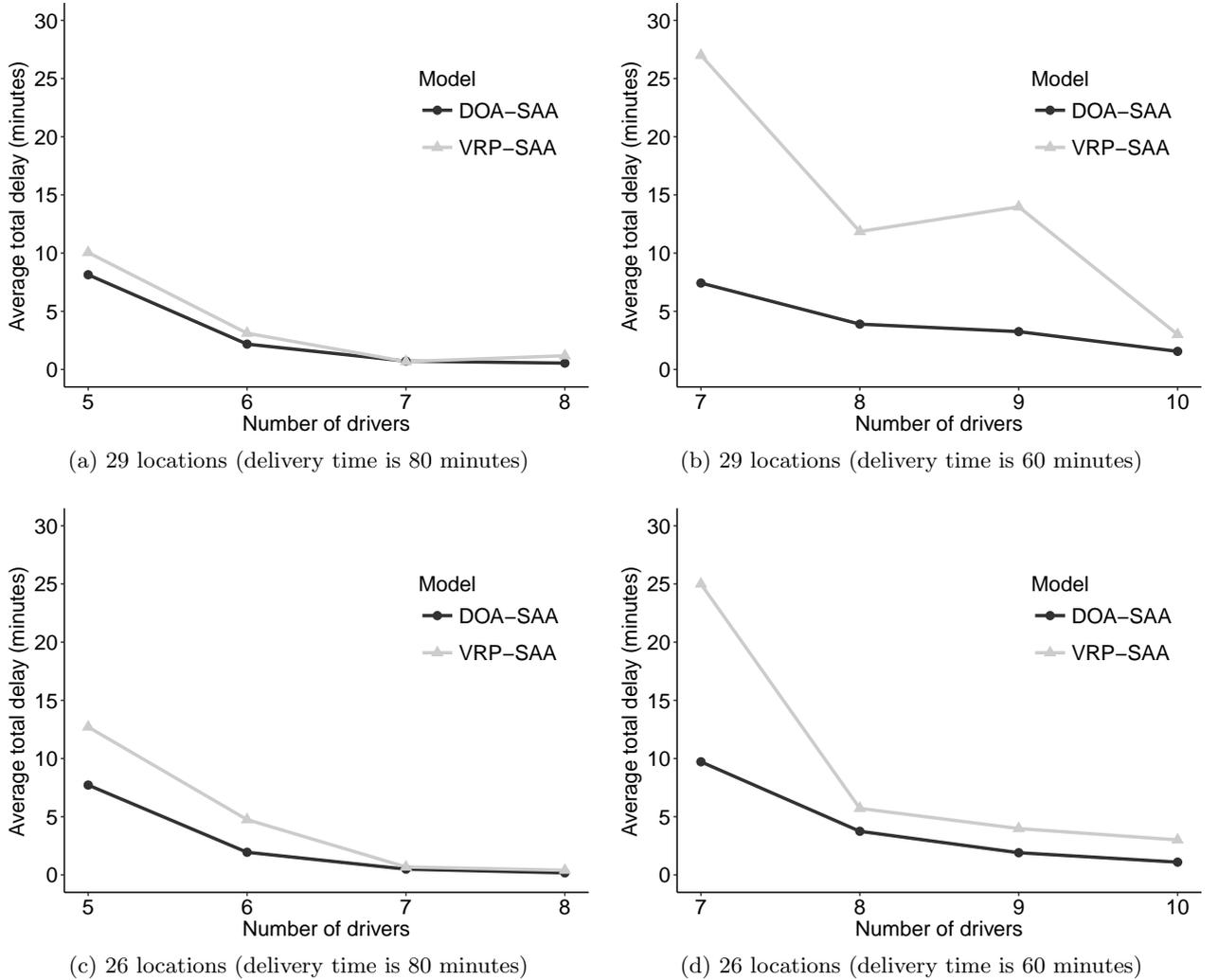


Figure 10 Average total delay with different staffing levels (DOA-SAA and VRP-SAA with sample size = 180)

8. Conclusion

In this study, we proposed a data-driven approach to model and solve the order assignment problem faced by last mile delivery service providers. Using the delivery data set from a food service provider in China, we identified driver’s routing behavior that deviate from theoretical shortest-distance tour and constructed a delivery tour length prediction function based on the historical data.

In view of uncertain service time, we incorporated the delivery tour prediction function into a stochastic optimization model for order assignment, whose SAA counterpart can be solved as a mixed-integer linear program via reformulation. We also developed a distributionally robust

optimization model to account for the scarcity of observations in historical data, and further derived a computationally tractable SOCP formulation based on the independence in service time. To solve both models efficiently, a branch-and-price algorithm is proposed by utilizing the problem structure in order assignment.

The performance of our proposed DOA models were benchmarked with the corresponding VRP-SAA and VRP-DRO models that use the classical VRP formulation. Our results showed that DOA-DRO outperformed all other models with the lowest average delay when the delivery time window was 80-minute and both DOA models performed better than their VRP-based counterparts under tighter delivery time window, e.g., 60 minutes. We also analyzed the impact of sample size and found that DOA-SAA might achieve lower average delay than DOA-DRO with a large number of samples, however, at the cost of increasing computation time. VRP-SAA model was less satisfactory and the performance of VRP-SAA did not necessarily improve as the sample size grew, potentially due to its misalignment with driver's actual routing behavior. Finally, we observed that DOA-SAA model captured the evolution of delay with varying staffing levels more accurately, while the solution from VRP-SAA might give misleading results. The accurate estimation of delay with variable number of drivers facilitates the firm's planning its staffing level.

This paper considered the firm that dispatches orders to the drivers after orders have been prepared. While such order assignment is responsive to the demand realization and improves the utilization of drivers, it requires extra efforts in operations and technology from the service provider, compared to the static order assignment strategy: first divide the service region and dedicatedly allocate drivers to their own subregions, e.g., delivery zoning and territory partitioning (e.g., Carlsson 2012, Carlsson and Delage 2013). The practical advantage of static order assignment is that drivers become more familiar and effective in visiting their assigned customer locations, which leads to a smoother delivery experience. Therefore, a natural extension is to develop a data-driven static order assignment model. A key challenge arises from the static order assignment is the uncertainty in customer locations to visit in each batch, in addition to the uncertainty in service time. While our SAA model can be directly extended using indicators for customer location realizations in all samples, it would be interesting to study a DRO model for static order assignment with both layers of uncertainties in customer location and service time.

References

- Ahipaşaoğlu, Selin Damla, Uğur Arıkan, Karthik Natarajan. 2016. On the flexibility of using marginal distribution choice models in traffic equilibrium. *Transportation Research Part B: Methodological* **91** 130–158.

- Ahipaşaoglu, Selin Damla, Rudabeh Meskarian, Thomas L Magnanti, Karthik Natarajan. 2015. Beyond normality: A cross moment-stochastic user equilibrium model. *Transportation Research Part B: Methodological* **81** 333–354.
- Angalakudati, Mallik, Siddharth Balwani, Jorge Calzada, Bikram Chatterjee, Georgia Perakis, Nicolas Raad, Joline Uichanco. 2014. Business analytics for flexible resource allocation under random emergencies. *Management Science* **60**(6) 1552–1573.
- Ardestani-Jaafari, Amir, Erick Delage. 2016. Robust optimization of sums of piecewise linear functions with application to inventory problems. *Operations research* **64**(2) 474–494.
- Ban, Gah-Yi, Cynthia Rudin. 2018. The big data newsvendor: Practical insights from machine learning. *Operations Research* **Forthcoming**.
- Barnhart, Cynthia, Ellis L Johnson, George L Nemhauser, Martin WP Savelsbergh, Pamela H Vance. 1998. Branch-and-price: Column generation for solving huge integer programs. *Operations research* **46**(3) 316–329.
- Beardwood, Jillian, John H Halton, John Michael Hammersley. 1959. The shortest path through many points. *Mathematical Proceedings of the Cambridge Philosophical Society*, vol. 55. Cambridge University Press, 299–327.
- Belavina, Elena, Karan Girotra, Ashish Kabra. 2016. Online grocery retail: Revenue models and environmental impact. *Management Science* **63**(6) 1781–1799.
- Bertsimas, Dimitris, Velibor V Mišić. 2015. Data-driven assortment optimization. *working paper* .
- Bertsimas, Dimitris, Allison OHair, Stephen Relyea, John Silberholz. 2016. An analytics approach to designing combination chemotherapy regimens for cancer. *Management Science* **62**(5) 1511–1531.
- Bertsimas, Dimitris J, Garrett Van Ryzin. 1991. A stochastic and dynamic vehicle routing problem in the euclidean plane. *Operations Research* **39**(4) 601–615.
- Bertsimas, Dimitris J, Garrett Van Ryzin. 1993. Stochastic and dynamic vehicle routing in the euclidean plane with multiple capacitated vehicles. *Operations Research* **41**(1) 60–76.
- Bien, Jacob, Robert Tibshirani. 2011. Hierarchical clustering with prototypes via minimax linkage. *Journal of the American Statistical Association* **106**(495) 1075–1084.
- Campbell, Ann M, Barrett W Thomas. 2008. Probabilistic traveling salesman problem with deadlines. *Transportation Science* **42**(1) 1–21.
- Carlsson, John Gunnar. 2012. Dividing a territory among several vehicles. *INFORMS Journal on Computing* **24**(4) 565–577.
- Carlsson, John Gunnar, Mehdi Behroozi, Kresimir Mihic. 2017. Wasserstein distance and the distributionally robust tsp .
- Carlsson, John Gunnar, Erick Delage. 2013. Robust partitioning for stochastic multivehicle routing. *Operations research* **61**(3) 727–744.

- Cherry, Christopher, Robert Cervero. 2007. Use characteristics and mode choice behavior of electric bike users in china. *Transport Policy* **14**(3) 247–257.
- Daganzo, Carlos F. 2005. *Logistics systems analysis*. Springer Science & Business Media.
- Elmachtoub, Adam N, Paul Grigas. 2017. Smart” predict, then optimize”. *arXiv preprint arXiv:1710.08005* .
- Erera, Alan L, Juan C Morales, Martin Savelsbergh. 2010. The vehicle routing problem with stochastic demand and duration constraints. *Transportation Science* **44**(4) 474–492.
- Esfahani, Peyman Mohajerin, Daniel Kuhn. 2015. Data-driven distributionally robust optimization using the wasserstein metric: Performance guarantees and tractable reformulations. *Mathematical Programming* 1–52.
- Farias, Vivek F, Srikanth Jagabathula, Devavrat Shah. 2013. A nonparametric approach to modeling choice with limited data. *Management Science* **59**(2) 305–322.
- Ferreira, Kris Johnson, Bin Hong Alex Lee, David Simchi-Levi. 2015. Analytics for an online retailer: Demand forecasting and price optimization. *Manufacturing & Service Operations Management* **18**(1) 69–88.
- Friedman, Jerome, Trevor Hastie, Robert Tibshirani. 2001. *The elements of statistical learning*, vol. 1. Springer series in statistics Springer, Berlin.
- Fukasawa, Ricardo, Humberto Longo, Jens Lysgaard, Marcus Poggi de Aragão, Marcelo Reis, Eduardo Uchoa, Renato F Werneck. 2006. Robust branch-and-cut-and-price for the capacitated vehicle routing problem. *Mathematical programming* **106**(3) 491–511.
- Gademann, Noud, Steef Velde. 2005. Order batching to minimize total travel time in a parallel-aisle warehouse. *IIE transactions* **37**(1) 63–75.
- Gao, Rui, Anton J Kleywegt. 2016. Distributionally robust stochastic optimization with wasserstein distance. *arXiv preprint arXiv:1604.02199* .
- Gendreau, Michel, Gilbert Laporte, René Séguin. 1996. Stochastic vehicle routing. *European Journal of Operational Research* **88**(1) 3–12.
- Glover, Fred. 1975. Improved linear integer programming formulations of nonlinear integer problems. *Management Science* **22**(4) 455–460.
- Hekimoğlu, Mert Hakan, Burak Kazaz, Scott Webster. 2016. Wine analytics: Fine wine pricing and selection under weather and market uncertainty. *Manufacturing & Service Operations Management* **19**(2) 202–215.
- Hoeffding, Wassily. 1948. A non-parametric test of independence. *The annals of mathematical statistics* 546–557.
- Holland, Chuck, Jack Levis, Ranganath Nugehalli, Bob Santilli, Jeff Winters. 2017. Ups optimizes delivery routes. *Interfaces* **47**(1) 8–23.

- Jaillet, Patrick, Jin Qi, Melvyn Sim. 2016. Routing optimization under uncertainty. *Operations research* **64**(1) 186–200.
- Klapp, Mathias A, Alan L Erera, Alejandro Toriello. 2016. The one-dimensional dynamic dispatch waves problem. *Transportation Science* **52**(2) 402–415.
- Kleywegt, Anton J, Alexander Shapiro, Tito Homem-de Mello. 2002. The sample average approximation method for stochastic discrete optimization. *SIAM Journal on Optimization* **12**(2) 479–502.
- Kong, Nan, Andrew J Schaefer, Brady Hunsaker, Mark S Roberts. 2010. Maximizing the efficiency of the us liver allocation system through region design. *Management Science* **56**(12) 2111–2122.
- Laporte, Gilbert. 2007. What you should know about the vehicle routing problem. *Naval Research Logistics (NRL)* **54**(8) 811–819.
- Laporte, Gilbert, Francois Louveaux, H el ene Mercure. 1992. The vehicle routing problem with stochastic travel times. *Transportation science* **26**(3) 161–170.
- Lim, Michael K, Ho-Yin Mak, Zuo-Jun Max Shen. 2016. Agility and proximity considerations in supply chain design. *Management Science* **63**(4) 1026–1041.
- Lima, Antonio, Rade Stanojevic, Dina Papagiannaki, Pablo Rodriguez, Marta C Gonz alez. 2016. Understanding individual routing behaviour. *Journal of The Royal Society Interface* **13**(116) 20160021.
- Mehrotra, Anuj, Ellis L Johnson, George L Nemhauser. 1998. An optimization based heuristic for political districting. *Management Science* **44**(8) 1100–1114.
- Ouyang, Yanfeng, Carlos F Daganzo. 2006. Discretization and validation of the continuum approximation scheme for terminal system design. *Transportation Science* **40**(1) 89–98.
- Pan, Qisheng, Jason Cao. 2015. *Recent Developments in Chinese Urban Planning*. Springer.
- Popescu, Ioana. 2007. Robust mean-covariance solutions for stochastic optimization. *Operations Research* **55**(1) 98–112.
- Shen, Zuo-Jun Max, Lian Qi. 2007. Incorporating inventory and routing costs in strategic location models. *European journal of operational research* **179**(2) 372–389.
- Solomon, Marius M. 1987. Algorithms for the vehicle routing and scheduling problems with time window constraints. *Operations research* **35**(2) 254–265.
- Wang, Hai, Amedeo Odoni. 2014. Approximating the performance of a last mile transportation system. *Transportation Science* **50**(2) 659–675.
- Wang, Xin, Michael K Lim, Yanfeng Ouyang. 2016. A continuum approximation approach to the dynamic facility location problem in a growing market. *Transportation Science* **51**(1) 343–357.
- Zhang, Yiling, Ruiwei Jiang, Siqian Shen. 2016. Ambiguous chance-constrained bin packing under mean-covariance information. *arXiv preprint arXiv:1610.00035* .
- Zheng, Zhichao, Karthik Natarajan, Chung-Piaw Teo. 2016. Least squares approximation to the distribution of project completion times with gaussian uncertainty. *Operations Research* **64**(6) 1406–1421.

Appendix. Proofs and Detailed Formulation

Proofs of Lemma 1 and Proposition 2

Proof of Lemma 1 By strong duality of $\max_{\mathbb{P}_k \in \mathbb{F}_k} \mathbb{E}_{\mathbb{P}_k} [\tilde{T}_k + h_k]^+$, we have

$$\begin{aligned} \min_{\lambda_k, \eta_k, \theta_k} \quad & \lambda_k + \eta_k \sum_{i \in \mathcal{I}} \mu_i y_{ik} + \theta_k \sum_{i \in \mathcal{I}} \sigma_i^2 y_{ik} \\ \text{s.t.} \quad & \lambda_k + \eta_k \tilde{T}_k + \theta_k \left(\tilde{T}_k - \sum_{i \in \mathcal{I}} \mu_i y_{ik} \right)^2 \geq [\tilde{T}_k + h_k]^+, \quad \forall \tilde{T}_k \in \mathbb{R}. \end{aligned}$$

It is equivalent to

$$\begin{aligned} \min_{\lambda_k, \eta_k, \theta_k} \quad & \lambda_k + \eta_k \sum_{i \in \mathcal{I}} \mu_i y_{ik} + \theta_k \sum_{i \in \mathcal{I}} \sigma_i^2 y_{ik} \\ \text{s.t.} \quad & \lambda_k + \eta_k \tilde{T}_k + \theta_k \left(\tilde{T}_k - \sum_{i \in \mathcal{I}} \mu_i y_{ik} \right)^2 \geq \tilde{T}_k + h_k, \quad \forall \tilde{T}_k \in \mathbb{R}, \\ & \lambda_k + \eta_k \tilde{T}_k + \theta_k \left(\tilde{T}_k - \sum_{i \in \mathcal{I}} \mu_i y_{ik} \right)^2 \geq 0, \quad \forall \tilde{T}_k \in \mathbb{R}. \end{aligned}$$

From the constraints, we observe that θ must be nonnegative. Otherwise, the quadratic constraints will be violated by large values of \tilde{T}_k . We rewrite the first constraint as:

$$\min_{\tilde{T}_k \in \mathbb{R}} \quad \lambda_k + (\eta_k - 1) \tilde{T}_k + \theta_k \left(\tilde{T}_k - \sum_{i \in \mathcal{I}} \mu_i y_{ik} \right)^2 \geq h_k.$$

Since the left-hand-side can be solved, it becomes:

$$\begin{aligned} & \lambda_k + (\eta_k - 1) \sum_{i \in \mathcal{I}} \mu_i y_{ik} - \frac{(\eta_k - 1)^2}{4\theta_k} \geq h_k \\ \equiv & \begin{cases} (\lambda_k + (\eta_k - 1) \sum_{i \in \mathcal{I}} \mu_i y_{ik} - h_k + \theta_k)^2 \geq (\lambda_k + (\eta_k - 1) \sum_{i \in \mathcal{I}} \mu_i y_{ik} - h_k - \theta_k)^2 + (\eta_k - 1)^2 \\ \lambda_k + (\eta_k - 1) \sum_{i \in \mathcal{I}} \mu_i y_{ik} - h_k \geq 0. \end{cases} \end{aligned}$$

Similarly, the second constraint is equivalent to

$$\begin{aligned} & \lambda_k + \eta_k \sum_{i \in \mathcal{I}} \mu_i y_{ik} - \frac{\eta_k^2}{4\theta_k} \geq 0 \\ \equiv & \begin{cases} (\lambda_k + \eta_k \sum_{i \in \mathcal{I}} \mu_i y_{ik} + \theta_k)^2 \geq (\lambda_k + \eta_k \sum_{i \in \mathcal{I}} \mu_i y_{ik} - \theta_k)^2 + \eta_k^2 \\ \lambda_k + \eta_k \sum_{i \in \mathcal{I}} \mu_i y_{ik} \geq 0. \end{cases} \end{aligned}$$

Thus, the inner problem is equivalent to the optimization problem with second-order cone constraints, as provided in the proposition. \square

Proof of Proposition 2 From the results in Lemma 1, the distributionally robust optimization problem (25) can be reformulated as

$$\begin{aligned} \min_{\mathbf{Y}, \lambda, \eta, \theta} \quad & \sum_{k \in \mathcal{K}} \left[\lambda_k + \eta_k \sum_{i \in \mathcal{I}} \mu_i y_{ik} + \theta_k \sum_{i \in \mathcal{I}} \sigma_i^2 y_{ik} \right] \\ \text{s.t.} \quad & h_k = \frac{l_k}{v} - \tau, \forall k \in \mathcal{K}, \\ & \left(\lambda_k + (\eta_k - 1) \sum_{i \in \mathcal{I}} \mu_i y_{ik} - h_k + \theta_k \right)^2 \geq \left(\lambda_k + (\eta_k - 1) \sum_{i \in \mathcal{I}} \mu_i y_{ik} - h_k - \theta_k \right)^2 + (\eta_k - 1)^2, \forall k \in \mathcal{K}, \end{aligned}$$

$$\begin{aligned}
& \lambda_k + (\eta_k - 1) \sum_{i \in \mathcal{I}} \mu_i y_{ik} - h_k \geq 0, \forall k \in \mathcal{K}, \\
& \left(\lambda_k + \eta_k \sum_{i \in \mathcal{I}} \mu_i y_{ik} + \theta_k \right)^2 \geq \left(\lambda_k + \eta_k \sum_{i \in \mathcal{I}} \mu_i y_{ik} - \theta_k \right)^2 + \eta_k^2, \forall k \in \mathcal{K}, \\
& \lambda_k + \eta_k \sum_{i \in \mathcal{I}} \mu_i y_{ik} \geq 0, \forall k \in \mathcal{K}, \\
& \theta \geq 0,
\end{aligned}$$

Constraints in DOA-SAA.

We replace $\lambda_k + \eta_k \sum_{i \in \mathcal{I}} \mu_i y_{ik} = \varrho_k$ and simplify the above program as:

$$\begin{aligned}
\min_{\mathbf{Y}, \eta, \theta, \varrho} \quad & \sum_{k \in \mathcal{K}} \left[\varrho_k + \theta_k \sum_{i \in \mathcal{I}} \sigma_i^2 y_{ik} \right] \\
\text{s.t.} \quad & h_k = \frac{l_k}{v} - \tau, \forall k \in \mathcal{K}, \\
& \left(\varrho_k - \sum_{i \in \mathcal{I}} \mu_i y_{ik} - h_k + \theta_k \right)^2 \geq \left(\varrho_k - \sum_{i \in \mathcal{I}} \mu_i y_{ik} - h_k - \theta_k \right)^2 + (\eta_k - 1)^2, \forall k \in \mathcal{K}, \quad (28) \\
& \varrho_k - \sum_{i \in \mathcal{I}} \mu_i y_{ik} - h_k \geq 0, \forall k \in \mathcal{K}, \quad (29) \\
& (\varrho_k + \theta_k)^2 \geq (\varrho_k - \theta_k)^2 + \eta_k^2, \forall k \in \mathcal{K}, \quad (30) \\
& \varrho_k, \theta_k \geq 0, \forall k \in \mathcal{K}, \quad (31)
\end{aligned}$$

Constraints in DOA-SAA.

Now we consider the above minimization problem with fixed \mathbf{Y} . After fixing \mathbf{Y} , the resulting minimization problem boils down to K independent subproblems, which corresponds to K drivers. Let the KKT multipliers be $\alpha_1, \alpha_2, \dots, \alpha_5 \geq 0$ (the subscript k is dropped here for brevity), corresponding to constraints (28)-(31). The stationarity conditions can be written as follows:

$$1 = 4\theta_k \alpha_1 + \alpha_2 + 4\theta_k \alpha_3 + \alpha_4, \quad (32)$$

$$0 = -2(\eta_k - 1)\alpha_1 - 2\eta_k \alpha_3, \quad (33)$$

$$\sum_{i \in \mathcal{I}} \sigma_i^2 y_{ik} = 4(\varrho_k - \sum_{i \in \mathcal{I}} \mu_i y_{ik} - h) \alpha_1 + 4\varrho_k \alpha_3 + \alpha_5. \quad (34)$$

First, we show that $\varrho_k > 0$ and $\theta_k > 0$: 1) If $\varrho_k = 0$, then from constraints (30) we have $\eta_k = 0$. It follows that $\alpha_1 = 0$ based on the stationarity condition (33). Then from (34), $\alpha_5 = \sum_{i \in \mathcal{I}} \sigma_i^2 y_{ik} > 0$. Hence $\theta_k = 0$ by the complementary slackness, which indicates that $\eta_k = 1$ from constraints (28). Thus we get a contradiction. 2) If $\theta_k = 0$, constraints (28) imply that $\eta_k = 1$ while constraints (30) imply $\eta_k = 0$, which generates a contradiction. As a result, ϱ_k and θ_k must be both positive, and $\alpha_4 = \alpha_5 = 0$.

Second, we prove that $\varrho_k > \sum_{i \in \mathcal{I}} \mu_i y_{ik} + h_k$ by contradiction. If $\varrho_k = \sum_{i \in \mathcal{I}} \mu_i y_{ik} + h_k$, $\eta_k = 1$ as implied by constraints (28). Then from the stationarity condition (33), we have $\alpha_3 = 0$. It follows that the RHS of constraint (34) is zero while the LHS is strictly positive. Hence $\varrho_k > \sum_{i \in \mathcal{I}} \mu_i y_{ik} + h_k$, and $\alpha_2 = 0$ by the complementary slackness.

We can now rewrite the stationarity conditions as:

$$1 = 4\theta_k(\alpha_1 + \alpha_3), \quad (35)$$

$$2\alpha_1 = 2\eta_k(\alpha_1 + \alpha_3), \quad (36)$$

$$\sum_{i \in I} \sigma_i^2 y_{ik} = 4(\varrho_k - \sum_{i \in I} \mu_i y_{ik} - h) \alpha_1 + 4\varrho_k \alpha_3. \quad (37)$$

Next we can show that $\alpha_1, \alpha_3 > 0$ by contradiction: 1) If $\alpha_1 = 0$, then we have $\theta_k, \alpha_3 > 0$ from (35) and $\eta_k = 0$ from (36). However, by the complementary slackness, $\alpha_3 = 0$ indicates $(\varrho_k + \theta_k)^2 = (\varrho_k - \theta_k)^2$, which is impossible as both ϱ_k and θ_k are positive. 2) If $\alpha_3 = 0$, stationarity conditions (35) and (36) imply that $\alpha_1 = 1/2\theta_k > 0$ and $\eta_k = 1$. By the complementary slackness, constraints (28) must be satisfied at the equality, so $\rho_k = \sum_{i \in I} \mu_i y_{ik} + h_k$ (recall $\theta_k > 0$). Then the RHS of (37) is zero while the LHS is strictly positive, which leads to a contradiction. Consequently, we have both α_1 and α_3 are positive. Then the complementary slackness implies:

$$4(\varrho_k - \sum_{i \in I} \mu_i y_{ik} - h)\theta_k = (\eta_k - 1)^2, \quad (38)$$

$$4\varrho_k \theta_k = \eta_k^2. \quad (39)$$

Then we can solve for $\varrho_k, \theta_k, \eta_k$ and α_1, α_3 using the above five equations (35)-(39):

$$\varrho_k = \frac{\left[\sum_{i \in I} \mu_i y_{ik} + h_k + \sqrt{(\sum_{i \in I} \mu_i y_{ik} + h_k)^2 + \sum_{i \in I} \sigma_i^2 y_{ik}} \right]^2}{4\sqrt{(\sum_{i \in I} \mu_i y_{ik} + h_k)^2 + \sum_{i \in I} \sigma_i^2 y_{ik}}}, \quad (40)$$

$$\eta_k = \frac{\sum_{i \in I} \mu_i y_{ik} + h_k + \sqrt{(\sum_{i \in I} \mu_i y_{ik} + h_k)^2 + \sum_{i \in I} \sigma_i^2 y_{ik}}}{2\sqrt{(\sum_{i \in I} \mu_i y_{ik} + h_k)^2 + \sum_{i \in I} \sigma_i^2 y_{ik}}}, \quad (41)$$

$$\theta_k = \frac{\varrho_k - (\sum_{i \in I} \mu_i y_{ik} + h_k)\eta_k}{\sum_{i \in I} \sigma_i^2 y_{ik}}. \quad (42)$$

As a result, we have

$$\varrho_k + \theta_k \sum_{i \in I} \sigma_i^2 y_{ik} = \sum_{i \in I} \mu_i y_{ik} + h_k + \sqrt{(\sum_{i \in I} \mu_i y_{ik} + h_k)^2 + \sum_{i \in I} \sigma_i^2 y_{ik}}. \quad (43)$$

So the original robust optimization formulation equals to:

$$\begin{aligned} \min_{\mathbf{Y}, \rho} \quad & \sum_{k \in \mathcal{K}} \left[\rho_k + \sum_{i \in \mathcal{I}} \mu_i y_{ik} + h_k \right] \\ \text{s.t.} \quad & h_k = \frac{l_k}{v} - \tau, \forall k \in \mathcal{K}, \\ & \rho_k^2 \geq \sum_{i \in \mathcal{I}} \sigma_i^2 y_{ik} + (\sum_{i \in \mathcal{I}} \mu_i y_{ik} + h_k)^2, \forall k \in \mathcal{K}, \\ & \text{Constraints in DOA-SAA.} \end{aligned}$$

The second constraints can be transformed to

$$\rho_k^2 \geq \sum_{i \in \mathcal{I}} \sigma_i^2 y_{ik}^2 + (\sum_{i \in \mathcal{I}} \mu_i y_{ik} + h_k)^2, \forall k \in \mathcal{K},$$

by utilizing the fact that y_{ik} is binary. Thus, the resulting optimization model is a MISOCP. \square

The Formulation of One-Way Traveling Salesman Problem

Given the set of realized customer locations $\mathcal{V} \subset \mathcal{I}$ and the depot node 0, we also introduce a dummy node $0'$ to facilitate the one-way travel distance calculation. Define a complete arc set $\mathcal{A}_{\mathcal{V}}$ on the node set $\mathcal{V}' = \mathcal{V} \cup \{0, 0'\}$ and each arc $(i, j) \in \mathcal{A}_{\mathcal{V}}$ is associated with a distance d_{ij} . The distance between the dummy node and all other nodes are 0, i.e. $d_{0'i} = d_{i0'} = 0$ for $i \in \mathcal{V} \cup \{0\}$. The decision variables are binary variables ζ_{ij} that indicate whether the driver travels arc $(i, j) \in \mathcal{A}_{\mathcal{V}}$. The formulation for the one-way traveling salesman problem is

$$\min \sum_{(i,j) \in \mathcal{A}} \zeta_{ij} d_{ij}, \quad (44)$$

$$\sum_{j \in \mathcal{I}'} \zeta_{ij} = 1, \quad \forall i \in \mathcal{I}, \quad (45)$$

$$\sum_{j \in \mathcal{I}'} \zeta_{ji} = 1, \quad \forall i \in \mathcal{I}, \quad (46)$$

$$\sum_{i \in \mathcal{I}} \zeta_{0i} = 1, \quad (47)$$

$$\sum_{i \in \mathcal{I}} \zeta_{i0'} = 1, \quad (48)$$

$$\zeta_{0'0} = 1, \quad (49)$$

$$\sum_{i,j \in \mathcal{S}} \zeta_{ij} \leq |\mathcal{S}| - 1, \quad \forall \mathcal{S} \subset \mathcal{I}', 2 \leq |\mathcal{S}| \leq I. \quad (50)$$

Constraints (45) and (46) are degree (flow) constraints for customer nodes. Constraint (47) ensure that the route begins from the depot. Constraint (48) specifies that the dummy node is entered once from the customer nodes and constraint (49) requires the route returns to the depot through the dummy node so no return trip cost is incurred. Constraints (50) are subtour elimination constraints that prevents the formation of illegal subtours. We implement the subtour elimination constraints as lazy constraints in Gurobi.

The Detailed Description of SP in the Branch-and-Price Algorithm

Constraints of DOA-SAA The complete set of constraints in the pricing subproblem of DOA-SAA is:

$$\sum_{i \in \mathcal{I}} \bar{y}_i \leq N \quad (51)$$

$$c(\bar{\mathbf{y}}) = \sum_{s \in \mathcal{S}} w^s, \quad (52)$$

$$\omega^s \geq \sum_{i \in \mathcal{I}} t_i^s \bar{y}_i + \frac{l}{v} - \tau, \quad \forall s \in \mathcal{S} \quad (53)$$

$$\omega^s \geq 0, \quad \forall s \in \mathcal{S}, \quad (54)$$

$$l = \sum_{j=0}^N f_j, \quad (55)$$

$$D_j^+ u_j \geq f_j \geq D_j^- u_j, \quad \forall j \in \{1, \dots, N\}, \quad (56)$$

$$\beta_0 d + \beta_1 a + \beta_2 b + \beta_3 a \sqrt{j-1} + \beta_4 b \sqrt{j-1} + \beta_5 n - D_j^- (1 - u_j) \geq f_j, \quad \forall j \in \{1, \dots, N\}, \quad (57)$$

$$f_j \geq \beta_0 d + \beta_1 a + \beta_2 b + \beta_3 a \sqrt{j-1} + \beta_4 b \sqrt{j-1} + \beta_5 n - D_j^+ (1 - u_j), \quad \forall j \in \{1, \dots, N\}, \quad (58)$$

$$d = \sum_{i \in \mathcal{I}} \hat{d}_i \bar{x}_i, \quad \forall k \in \mathcal{K}, \quad (59)$$

$$\sum_{i \in \mathcal{I}} \bar{x}_i \geq \bar{y}_i, \quad \forall i \in \mathcal{I}, \quad (60)$$

$$\bar{x}_i \leq \bar{y}_i, \quad \forall i \in \mathcal{I}, \quad (61)$$

$$n = \sum_{j=0}^N j \cdot u_j, \quad (62)$$

$$\sum_{j=0}^N u_j = 1, \quad (63)$$

$$a = \bar{a} + \underline{a} \geq 0, \quad (64)$$

$$b = \bar{b} + \underline{b} \geq 0, \quad (65)$$

$$\bar{a} \geq \text{lat}_i \cdot \bar{y}_i, \quad \forall i \in \mathcal{I}, \quad (66)$$

$$\underline{a} \geq -\text{lat}_i + M(\bar{y}_i - 1), \quad \forall i \in \mathcal{I}, \quad (67)$$

$$\bar{b} \geq \text{long}_i \cdot \bar{y}_i, \quad \forall i \in \mathcal{I}, \quad (68)$$

$$\underline{b} \geq -\text{long}_i + M(\bar{y}_i - 1), \quad \forall i \in \mathcal{I}, \quad (69)$$

$$u_j \in \{0, 1\}, \quad \forall j \in \{0, 1, \dots, N\}, \quad (70)$$

$$\bar{y}_i \in \{0, 1\}, \quad \forall i \in \mathcal{I}. \quad (71)$$

The constraints have similar meanings to those in the original DOA-SAA except that the zone subscript is removed.

Constraints of DOA-DRO In DOA-DRO, the delay cost $c(\bar{\mathbf{y}})$ has a different form than DOA-SAA but the constraints defining the travel distance l remains the same. So the complete set of constraints in the SP of DOA-DRO can be stated as

$$\sum_{i \in \mathcal{I}} \bar{y}_i \leq N, \quad (72)$$

$$c(\bar{\mathbf{y}}) = \rho + h + \sum_{i \in \mathcal{I}} \mu_i \bar{y}_i, \quad (73)$$

$$h = \frac{l}{v} - \tau, \quad (74)$$

$$\rho^2 \geq \sum_{i \in \mathcal{I}} \sigma^2 y_i^2 + \left(\sum_{i \in \mathcal{I}} \mu_i y_i + h \right)^2, \quad (75)$$

$$\text{Constraints (55)-(71)}. \quad (76)$$

Constraints of VRP-SAA In addition to the set of customer locations \mathcal{I} and the depot node 0, a dummy node $0'$ is added as in the one-way TSP formulation. Similarly, we define a complete arc set \mathcal{A} on the node set $\mathcal{I}' = \mathcal{I} \cup \{0, 0'\}$ and each arc $(i, j) \in \mathcal{A}$ is associated with a distance d_{ij} . The distance between the dummy node and all other nodes are 0, i.e. $d_{0'i} = d_{i0'} = 0$ for $i \in \mathcal{I} \cup \{0\}$. In the pricing subproblem of VRP-SAA, the decision variables are binary variables ζ_{ij} that indicate whether the driver travels arc $(i, j) \in \mathcal{A}$, as well as the variables \bar{y}_i that indicate whether customer i is covered in this route. The detailed formulation is given as follows:

$$\sum_{i \in \mathcal{I}} \bar{y}_i \leq N, \quad (77)$$

$$c(\bar{\mathbf{y}}) = \sum_{s \in \mathcal{S}} w^s, \quad (78)$$

$$\omega^s \geq \sum_{i \in \mathcal{I}} t_i^s \bar{y}_i + \frac{l}{v} - \tau, \quad \forall s \in \mathcal{S}, \quad (79)$$

$$l = \sum_{(i,j) \in \mathcal{A}} \zeta_{ij} d_{ij}, \quad (80)$$

$$\sum_{j \in \mathcal{I}'} \zeta_{ij} = \bar{y}_i, \quad \forall i \in \mathcal{I}, \quad (81)$$

$$\sum_{j \in \mathcal{I}'} \zeta_{ji} = \bar{y}_i, \quad \forall i \in \mathcal{I}, \quad (82)$$

$$\sum_{i \in \mathcal{I}} \zeta_{0i} = 1, \quad (83)$$

$$\sum_{i \in \mathcal{I}} \zeta_{i0'} = 1, \quad (84)$$

$$\zeta_{0'0} = 1, \quad (85)$$

$$\sum_{i,j \in \mathcal{S}} \zeta_{ij} \leq |\mathcal{S}| - 1, \quad \forall \mathcal{S} \subset \mathcal{I}', \quad 2 \leq |\mathcal{S}| \leq I. \quad (86)$$

Constraints (81)-(86) play the same roles as in the one-way TSP formulation.

Constraints of VRP-DRO The constraints in the pricing subproblem of VRP-DRO can be obtained by combining the constraints from DOA-DRO and VRP-SAA: (72)-(75) and (80)-(86).