Transportation Research Part C 152 (2023) 104143

Contents lists available at ScienceDirect

Transportation Research Part C

journal homepage: www.elsevier.com/locate/trc

Online-to-offline on the railway: Optimization of on-demand meal ordering on high-speed railway[☆]

Chunling Luo[a], Lei Xu[b,*]

[a] Department of Data Science and Management, Alibaba Business School, Hangzhou Normal University, China
[b] Georgia Tech Shenzhen Institute, Tianjing University, China

ARTICLE INFO

Keywords:
On-demand meal service
Railway catering service
Pre-ordering policy
Distributionally robust optimization
Cutting-plane algorithm

ABSTRACT

On-demand meal ordering on high-speed railway is an emerging on-demand service that is becoming increasingly popular. One key challenge about this service is how to design the ordering system to maximize profits while guaranteeing timely delivery. However, few related studies have been done for this new business. In this paper, we first develop a mathematical model to optimize the existing ordering procedure, which sets a deadline for order placement. Furthermore, we propose a pre-ordering policy which cancels the order deadline but ensures timely delivery via stocking up. Considering the ambiguity in demand distributions and uncertainty in delivery time, we develop distributionally robust stochastic optimization models for both the existing and the pre-ordering procedures. Cutting-plane algorithms are developed to solve the proposed models. Computational studies on a real-world high-speed railway line in China show that our proposed new policy could simultaneously increase service availability, improve the profit, and reduce delay rate.

☆ This article belongs to the Virtual Special Issue on "On-Demand Transportation".

* Corresponding author.
E-mail addresses: iseluoc@gmail.com (C. Luo), xulei@gtsi.edu.cn (L. Xu).

https://doi.org/10.1016/j.trc.2023.104143
Received 18 May 2022; Received in revised form 6 April 2023; Accepted 9 April 2023
Available online 2 May 2023

Nomenclature

Main notation introduced in this paper.

Sets

$[R]$	Set of restaurants, indexed by i

Parameters

T	Train departure time, the length from 0 to time T is split into equalized subintervals, indexed by t
$\tilde{d}_{i,t}$	Demand for on-demand meals for restaurant i at time t, $\underline{d}_{i,t} \leq \tilde{d}_{i,t} \leq \bar{d}_{i,t}$
$\tilde{v}_i$	Random delivery time from restaurant i to station platform
$V_{i,t}$	Quantity of meals that restaurant i can prepare during time interval $(t-1,t)$
p_i^c	Commission fee charged from restaurant i per ordered meal
β_i	Delay penalty cost for restaurant i per ordered meal
p_i^s	Selling price per meal
c_i	Pre-ordering fee per meal that the platform pay to the restaurant i
$P_{i,s}$	The probability of scenario s of delivery time $\tilde{v}_i$

Decision variables

$x_{i,t} \in \{0,1\}$	Binary decision variable: 1 if ordering at time t is accepted by restaurant i and otherwise 0
$y_{i,t} \in \{0,1\}$	Binary decision variable: 1 if ordering at time t is accepted as real-time orders by restaurant i and otherwise 0
$h_{i,t}$	Accumulated quantity of orders waiting to be prepared by restaurant i at time t
e_i	Quantity of pre-ordered meals the platform books from restaurant i with unit pre-ordering fee c_i
q_i	Quantity of successfully executed ordered meals from restaurant i
δ_i	Quantity of delayed meal of restaurant i
$z_{i,t}$	Auxiliary decision variable to represent $\lvert\tilde{d}_{i,t} - \mu_{i,t}\rvert$ by linear constraints

Dual decision variables

λ	Dual variable corresponding to constraint that the sum of the probabilities in a probability distribution always equal 1
$\theta_{i,t}$	Dual variable corresponding to lower bound of the expected value of $\tilde{d}_{i,t}$
$\gamma_{i,t}$	Dual variable corresponding to upper bound of the expected value of $\tilde{d}_{i,t}$
$\pi_{i,t}$	Dual variable corresponding to bound of average absolute deviation of $\tilde{d}_{i,t}$

1. Introduction

High-speed railway (HSR), as the fastest ground-based commercial transportation mode, could provide flexible services to customers and society with the characteristics of safety, velocity, capacity and sustainability. Therefore, it is often described as the "transport mode of the future" and its network is rapidly expanding around the world, especially in China (Espinosa-Aranda et al., 2015; Fu et al., 2015). The annual ridership of China's HSR reached 2.005 billion in 2018 and 2.29 billion in 2019. The increasingly huge number of ridership makes catering services for HSR a significant concern of railway operators for higher-quality passenger service. It is reported that railway provided 171 million meals in 2014 (Wu et al., 2017b) and the annual catering sales volume at HSR reaches 4 billion RMB in China. The meal demands are still growing with the climbing number of annual ridership. Railway corporation has been making continuous efforts to improve catering service. Traditionally, HSR mainly provides ambient meals and cold chain meals (Wu et al., 2017a), which will be delivered from various distribution centers to trains at specific stations (Wu et al., 2019) and offered to passengers from the train's dining car after heating. However, these food has been widely criticized by the public due to its lack of variety, lack of flavor, poor quality and high prices.

In response to these public complaints, in recent years, on-demand meal ordering has been introduced into the existing catering service system for HSR in different countries. On-demand meal ordering on HSR allows passengers to order meals offered not only by the train but also other qualified vendors that offer hot and fresh take-out meals and/or special local specialties via online ordering platform or mobile app. India introduced this service in 2014. It covered 1516 trains and 45 stations spreading all across the country in 2015 already and the quantity of e-catering orders reached as high as 20,000 per day during the pre-COVID period. China introduced this service in 2017, with 27 HSR stations covered initially and extended to another 11 stations in 2018. Ever since its introduction, on-demand meal ordering on HSR is viewed as an innovation of railway catering services. This service is welcomed due to its convenience and abundant choices of cuisines. The order quantity has been increasing. It was reported that during the spring festival of 2020, HSR on-demand meal orders at Nanchang East station (China) in 2020 increased by 118% compared with the same period in 2019 and the quantity of daily order at this single station was between 1500 to 1700.

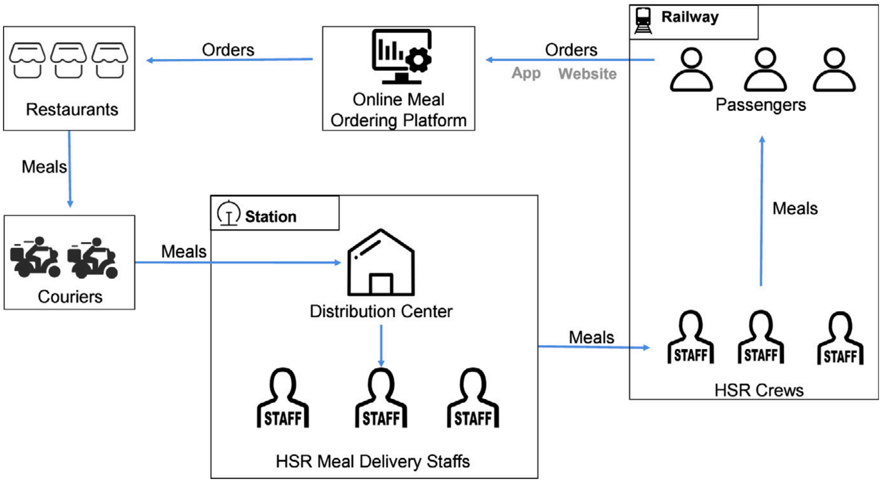


Fig. 1. On-demand meal ordering and delivery on high-speed railway.

On-demand meal ordering on HSR is as convenient as other on-demand meal ordering services for customers, which allow them to place orders online and wait meals to be delivered to them without the trouble of going to restaurants in person. However, different from off-board cases, delivery of meals ordered online on HSR needs joint cooperation of multiple participants (the procedure is presented in Fig. 1). Ordered meals will first be delivered by couriers/riders from the restaurants to the distribution center located at the station, and then HSR meal delivery staff will pass these meals to railway crew, who finally send meals to passengers' seats. Moreover, HSR follows a fixed transport plan, including line plan that specifies route, stopping pattern and frequency of the line (Fu et al., 2015) and train timetable that specifies the arrival and departure time at each station along the line (Barrena et al., 2014; Yue et al., 2016; Khadilkar, 2017). Usually HSR only stops for a few minutes at each station. Hence, meals must be delivered to the railway within the tight time window between the arrival time and the departure time specified in the timetable (Jiang et al., 2020). Unlike other on-demand meal ordering systems like Ele.me and UberEats that allow late delivery, on-demand meal orders on HSR fail immediately without any remedy if meals are not transferred to the station on time.

To deal with this issue, currently on-demand meal ordering platforms for HSR set a deadline for order placement to guarantee timely delivery. In China, passengers were requested to place orders at least two hours before the train departs from the station in 2017. Later in early 2018 it was shortened to one hour. In India, the deadline ranges from 45 to 60 minutes before train departure time. However, whether the current deadline setting for HSR on-demand meal ordering is the best still needs validation and verification. If the deadline is set too late, then there will exist high risk of delivery delays and order failures, resulting in not only cost of delayed meals, but also poor experiences of the customers and lower public confidence in this service. On the contrary, if the deadline is set too early, then potential orders after the deadline that could have been satisfied will be lost, resulting in high opportunity cost and reducing the profit of the system. In addition, earlier deadline implies lower level of service availability and flexibility. To guarantee high level of profit and service quality simultaneously, advanced quantitative management approaches are needed to better operate this new online-to-offline (O2O) practice, especially on issues related to order management, such as order deadline setting, ordering procedures and quantity management, etc. However, few researches have been done on these problems (Jiang et al., 2020). In this paper, we will attempt to study the abovementioned issues in detail.

Order management strategies and policies for on-demand meal ordering system depend heavily on meal demands, which are uncertain in nature (Wu et al., 2018). Meal demands on HSR vary with the number of passengers on the train, which varies across yearly, weekly and even daily (Niu et al., 2015; Qi et al., 2018; Sun et al., 2018a). Time of the day also influences meal demands. For example, demands during peak dining periods will be higher. Although historical sales data during the last several years (i.e., all data available ever since the introduction of on-demand meal ordering on HSR) could provide some probabilistic information of the demands, it is still very hard to estimate the exact probability distributions because available data for a station at a specific time is still scarce, especially for new introduced stations and lines. Moreover, this new emerging market is still growing with the popularity of this service gradually increases over time. Hence, distributions estimated from the historical data are usually not accurate. Considering the lack of complete probabilistic information, we adopt distributionally robust optimization approach to deal with uncertainty in demands in this work. Distributionally robust optimization is a cutting-edge optimization approach that is designated for decision problems with partial knowledge of distributions, namely, ambiguity in probability distribution (Rahimian and Mehrotra, 2019). This approach hedges against ambiguity by optimizing the worst-case of an objective function over the ambiguity set, i.e., a family of probability distributions consistent with the prior knowledge. The robust property of this approach can help guarantee the system performance, especially timely delivery, across all possible distributions, which is of importance in on-demand meal delivery service.

Besides uncertainty in demands, we also consider uncertainty in delivery time of meals. The delivery time of meals is affected by multiple factors, such as weather, traffic condition, unexpected accidents, and so on. As delivery time directly affects the arrival time of meals, it is necessary to take account of this factor in the management of on-demand meal ordering. Thanks to rich availability of map and traffic data, it is usually not difficult to obtain the distribution of delivery time.

In this paper, we will study how to improve the order management of HSR on-demand meal ordering system with uncertainty considered. The main contributions of our work can be stated as follows:

- First, we develop a mathematical optimization model to optimize the existing ordering procedure by identifying the best order deadline, with an objective to improve the system revenue but penalize order delays. In our model, we consider two key uncertain factors: demand and delivery time. Ambiguity in demands is characterized by range of expected values and average absolute deviation while uncertainty in delivery time is characterized by known probability distribution due to rich available information. We develop a distributionally robust stochastic programming model to find the best decision that optimizes the worst-case expected profit of the platform. To the best knowledge of authors, this is the first attempt to construct quantitative optimization tool for this new business under uncertainty.
- Considering the limitations of exiting ordering procedure, we further propose a novel pre-ordering policy for HSR on-demand meal ordering service. Under this new policy, the system will stock up or pre-order meals to cancel the order deadline, that is, passengers are allowed to place orders at any time. Received orders are either assigned to the restaurants as a normal real-time order or satisfied by the stock-up meals when there is no enough time for real-time processing. This policy could simultaneously improve the system revenue, service availability and guarantee high level of timely delivery. Advanced modeling is applied to achieve these goals considering uncertainties in key factors of this problem. In particular, distributionally robust stochastic optimization models are developed to help determine the best quantity of pre-ordered meals and order assignment policy.
- The advanced mathematical optimization models we develop to optimize the existing and new-proposed ordering procedures are both very complex due to the difficulty in tackling ambiguity and uncertainty in this problem. As a result, it is impossible to solve these models with off-the-shelf algorithms and tools. To address this issue, we reformulate these models into tractable formulations based on duality theory and develop cutting-plane algorithms to solve the reformulated models efficiently.

The remainder of this paper is organized as follows. Section 2 reviews research works related to optimization of HSR on-demand meal ordering service, including traditional meal delivery services, catering services on the railway system and related theoretical methodologies. Section 3 describes in general the problem of interest. Section 4 introduces the order deadline optimization model and solution approach for solving the distributionally robust optimization framework. Section 5 further develops a pre-ordering policy and also presents its distributionally robust optimization model which is solved by a cutting plane algorithm. Section 6 conducts a thorough case study on a HSR line in China. Finally, Section 7 draws the conclusion.

2. Literature review

In this paper, we consider a HSR meal delivery problem where the revenue-maximization, order- delay-averse operating platform makes order acceptance deadline decision and optimal pre-ordering decision, considering multiple restaurants preparing meal orders with fixed yield, demand uncertainty and also stochastic delivery time.

The HSR meal delivery service includes traditional dining car service and online booking service. The traditional dining car service has been early introduced since trains became the main transportation mode. A series of researches have been done on the management and operation of this traditional HSR catering service. Wu et al. (2017b) proposed a two-phase robust fuzzy quality function deployment process to identify and evaluate potential service stations before the design phase of the distribution system. Wu et al. (2017a) further studied the design of the distribution system for high-quality perishable food products to HSR. They developed a three-echelon location routing problem with tight time windows and time deadlines to determine the optimal locations, capacities of the distribution center and the number of meals delivered to stations. Later (Wu et al., 2018) formulated arc-flow based and path flow-based system-optimized optimization models to identify the order quantity and flow pattern in a food supply chain network for HSR catering services, considering uncertainties in demand and time constraints. Wu et al. (2019) further optimized the distribution plan for HSR catering service via developing a non-linear mixed integer programming problem for identifying the locations for food production and controlling of food inventory level on trains during their trips, taking account of time-varying demands and capacity restrictions. One main characteristic of the supply system for railway catering is that this system has strict time constraints. Theoretical studies closely related to this type of problem mainly includes delivery and pick up problem with time window, e.g., Dumas et al. (1991), Ropke and Pisinger (2006), Baldacci et al. (2011) and Sun et al. (2018b), and vehicle routing problem with time window, e.g., Desrochers et al. (1992), Bräysy and Gendreau (2005a,b), Pan et al. (2021) and Zhang et al. (2021). Interested readers may refer to the abovementioned literature for a detailed study on these topics. Flight catering service is also an important research topic in transportation field. For example, Goto et al. (2004) developed a finite horizon Markov decision process model for the airline meal provisioning activity to identify policies for the number of meals to upload. Ho and Leung (2010) studied a manpower scheduling problem with job time windows for flight catering. However, the literature for *online on-demand* meal service in railway or airline systems is scarce. This is probably because that the on demand meal booking service was only seen in recent years in HSR system as an emerging internet-driven application. Therefore, studies on quantitative management of this new service in transportation system are needed.

HSR on-demand meal ordering services are an extension of the ordinary meal delivery services that has been emerging worldwide recently. For example, on-demand meal ordering and delivery market in China increased from 0.15 billion CNY in 2010 to 44.24 billion CNY in 2015 (He et al., 2019). Two main research directions in meal delivery service are the pickup/delivery optimization and the economic analysis of the meal/food delivery platform. In the former research stream, Reyes et al. (2018a) defined meal delivery problem as a complex pickup and delivery system with multiple pick-up points (restaurants), dynamic order arrivals, delivery capacity distributed throughout the day in the form of courier shifts, and the possibility to pick up multiple orders simultaneously among others. Yildiz and Savelsbergh (2019) developed a novel formulation to minimize total courier compensation/average click-to-door time/click-to-door overage. The formulation relies on a core concept of *work package* characterized by the bundle served, the start location, the end location, the start time and the end time of the package, and the assumption of a clairvoyant decision maker with perfect information about order arrivals (i.e., rather than dynamic arrivals). Reyes et al. (2018b) proposed a rolling-horizon approach that incorporates postponement and bundling, which were also considered in Ulmer et al. (2021). Ulmer et al. (2021) demonstrated that, although postponement and bundling are valuable in deriving good solutions, the buffer is the most important element of their solution approach. Some works (e.g., Pinto et al. (2019) and Liu (2019)) studied how to deliver online meals via drones driven by the research heat in drone routing, and others examined the optimal workforce capacity and optimal meal order allocations (e.g., Dai and Liu (2020)). Mao et al. (2019), on the other hand, performed an empirical study to investigate the impact of delivery performance on future customer orders for an on-demand meal delivery service platform. In the second research stream which arises recently, Bahrami et al. (2021) considered a three-sided market (i.e., customers, suppliers and crowdsourced drivers) for on-demand food and grocery delivery services and investigated the commissions paid by customers and suppliers, and the wage offered to drivers. Several managerial insights were derived from their analysis. Liu and Li (2023) studied the economic equilibrium for a food delivery market and also a ridehailing market. The two markets are interdependent as the two platforms share the same pool of for-hire drivers. On-demand meal ordering on HSR has features different from traditional HSR and flight catering services as well as normal on-demand meal ordering service. Hence, specific studies are needed. However, few studies have been done on this new innovative on-demand service on HSR. Recently Jiang et al. (2020) developed a departure time and route selection model to coordinate the delivery time of order and departure time of the train. However, topics related to order management have not been studied elaborately, which is the focus of this work.

The methodology of our work belongs to distributionally robust optimization (DRO), an important and eye-catching field of optimization approaches that deals with decision-making problems under uncertainty. DRO aims to address challenges of classical stochastic and dynamic programming, including ambiguity, big data, the optimizers' curse and the curse of dimensionality (Delage et al., 2018). DRO maximizes the worst-case expected reward over an ambiguity set which can be generally divided into two classes: sets based on moment information (e.g., Natarajan et al. (2008), Delage and Ye (2010) and Xu et al. (2021)) or sets based on 'distance' from a reference distribution (e.g., Ben-Tal et al. (2013), Gao and Kleywegt (2016), Esfahani and Kuhn (2018) and Liu et al. (2022)). There exist a number of literature reviews on DRO, we omit detailed review for length limitation. Interested readers may refer to Delage et al. (2018) and Rahimian and Mehrotra (2019) for a comprehensive review. Transportation researchers have popularized the application of DRO by using it to address the inherent uncertainties and randomness in transportation systems, e.g., passenger demand, travel time or stochastic arrivals. To name a few, Shehadeh et al. (2021) employed DRO to investigate the fleet sizing and allocation problem for on-demand last-mile service when the distribution of the demand is unknown. Zhang et al. (2021) applied DRO with Wasserstein distance-based ambiguity set to study vehicle routing problem with time windows. Our work follows the stream of DRO employing ambiguity set based on moment information, which has been used in various practical problems (e.g., Nakao et al. (2017), Zhao et al. (2019) and Xiong et al. (2016)). To the best of our knowledge, this work is (among) the first to solve meal delivery service in transportation field via DRO approach.

3. Problem description

In this paper, we study how to optimize the ordering procedure for HSR on-demand meal ordering system. We consider one HSR line that provides on-demand meal ordering service at a certain station with R restaurants available on the platform, each of which indexed by $i \in [R]$. Suppose the train departs from the station at a fixed departure time T. We consider a known period of time, spanning from $t = 1$ to $t = T$ given equalized time interval. Generally, $t = 1$ can be set to be the moment when the first on-demand meal order can be placed or alternatively the moment when restaurants begin to prepare meals. In the latter case, demands before $t = 1$ could be accumulated and described by $h_{i,0}$, which denotes accumulate quantity of orders waiting to be prepared initially. In addition, $t = T$ can be set to be the train departure time of the station that provides this service. In addition, the delivery time from restaurant $i \in [R]$ to the station is uncertain and denoted by $\tilde{v}_i$. Due to rich availability of map and traffic data, we suppose that the delivery time $\tilde{v}_i$ follows a known discrete probability distribution with probability $P_{i,s}$, where $s = 1, 2, \ldots, S_i$ denotes all possible scenarios of $\tilde{v}_i$.

Let $\tilde{d}_{i,t}$ represent on-demand meal demand occurring at time t for restaurant i, with t restricted to $0 \le t \le T$. This setting follows the fact that for each station, orders to be served at this station are placed strictly no later than train departure time. Considering the uncertainty of demands, $\tilde{d}_{i,t}$ is a random variable. Unfortunately, due to the lack data and information, exact form of its probability distribution is very difficult to obtain in this new business. On the other hand, information on marginal moment of random meal demands can be relatively easier to capture. In this paper, we take into account these issues and adopt the approach of distributionally robust optimization. We define an ambiguity set based on moments information, which consists of all possible probability distributions of the demands. Let $\tilde{\boldsymbol{d}} \in \mathbb{R}^{R\times T}$ denote the vectors of demands (we use letters in boldface to denote the vector of variables hereafter). We define the ambiguity set of the joint probability distribution of $\tilde{\boldsymbol{d}}$, denoted by $\mathcal{F}$, as follows:

$$\mathcal{F}(\underline{\boldsymbol{\mu}}, \bar{\boldsymbol{\mu}}, \bar{\boldsymbol{\eta}}) = \left\{ \mathbb{P} : \quad \begin{array}{l} \int_{\boldsymbol{W}} d\mathbb{P}(\tilde{\boldsymbol{d}}) = 1 \\ \underline{\mu}_{i,t} \le \int_{\boldsymbol{W}} \tilde{d}_{i,t} d\mathbb{P}(\tilde{\boldsymbol{d}}) \le \bar{\mu}_{i,t}\ \forall i \in [R], t \le T \\ \int_{\boldsymbol{W}} |\tilde{d}_{i,t} - \mu_{i,t}| d\mathbb{P}(\tilde{\boldsymbol{d}}) \le \bar{\eta}_{i,t},\ \forall i \in [R], t \le T \end{array} \right\} \tag{1}$$

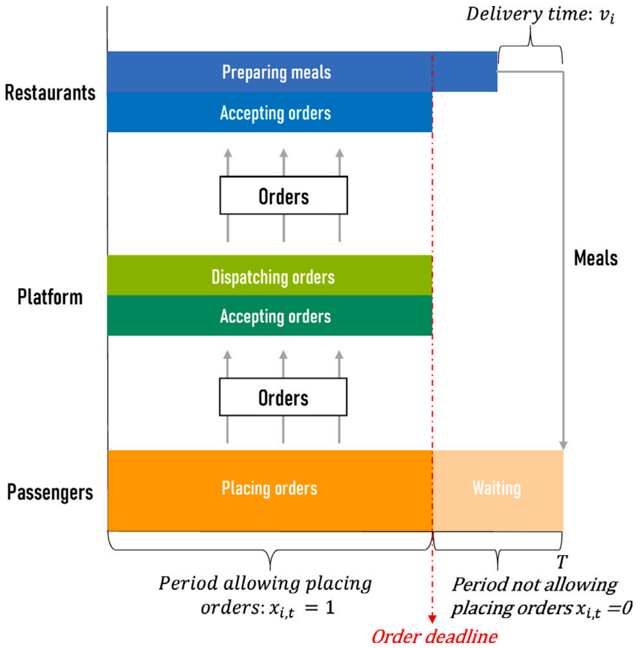


Fig. 2. Existing ordering process.

Here, $\boldsymbol{W}$ is the support, with vectors $\boldsymbol{W} := \{\tilde{\boldsymbol{d}}_i : \underline{d}_{i,t} \leq \tilde{d}_{i,t} \leq \bar{d}_{i,t}, \forall t \leq T, \forall i \in [R]\}$. The first line in (1) ensures that only valid distributions over the support set of $\tilde{\boldsymbol{d}}$ is contained. The second line indicates the marginal means are within a range, and the last line restricts the average absolute deviation from a nominal value $\mu_{i,t}$ is upper bounded. Later we will show that the measure of this average absolute deviation can be reformulated linearly, which is computationally attractive.

4. Optimizing order deadline for the existing ordering procedure

In this section, we focuses on how to optimize the existing ordering procedure for HSR on-demand meal ordering system. In particular, we will study how to optimize the most important policy of the existing ordering procedure: deadline setting for order placement.

4.1. Existing ordering procedure and order deadline optimization model

In the existing ordering procedure (illustrated in Fig. 2), the platform sets a deadline for order placement. Suppose the order deadline is $T - L_i$, where L_i is the length of period when HSR on-demand meal ordering is unavailable for restaurant i. Let binary decision variable $x_{i,t} \in \{0, 1\}$ denote whether on-demand meal orders to be served by restaurant i are allowed to be placed at time t. Currently, when $t \leq T - L_i$, passengers can place on-demand meal orders (i.e., $x_{i,t} = 1$); when $t > T - L_i$, orders to be served at station i will no longer be accepted (i.e., $x_{i,t} = 0$). All placed orders will be assigned to restaurants by the platform and meals will be prepared and delivered to the station before the train departure time T. In China, currently the platform sets the same order deadline (one hour) for most stations and restaurants. Actually, such a setting is only based on safe approximation, probably at the cost of losing potential orders. In addition, different restaurants may have different delivery time, meal demands and meal preparation speed. Hence, quantitative approach are needed to optimize the setting for distinct restaurants in different stations. The following effort is to investigate how to identify the best order deadline considering the uncertainty in demands $\tilde{\boldsymbol{d}}$ and delivery time $\tilde{v}_i$. In particular, we develop distributionally robust stochastic optimization models to optimize the order deadline with the objective to optimize the worst-case expected profit of the system.

To model the problem, we first define the following parameters and variables that are important. As we presented above, the demand for the on-demand service is described by $\tilde{d}_{i,t}$. In on-demand service, besides delivery time $\tilde{v}_i$, the meal preparation speed also affects the time of delivery because meals could only be delivered after it is finished by the restaurants. We let meal preparation capacity $V_{i,t}$ to describe the production speed of meal. To be specific, $V_{i,t}$ denotes the maximal number of meals restaurant i could prepare during time interval $(t-1, t)$. Ordered meals accepted but not yet finished will accumulate and we let $h_{i,t}$ denote the accumulated quantity of orders at restaurant i waiting to be prepared at time t. All accepted orders that are still not ready for delivery at time $T - \tilde{v}_i$ (the latest time for timely delivery) will be delayed, that is, $h_{i,T-\tilde{v}_i}$ is the quantity of delayed orders at station i. If accepted orders are successful delivered to customers, then the platform will takes a commission on the value of the orders. We use p_i^c to denotes the commission fee charged from restaurant i per successfully delivered ordered meal. If the accepted orders are delayed, then the platform will receive no commission and suffer penalty cost due to compensation of undelivered orders as well as potential loses caused by unreliable service. We use β_i to denote penalty cost the platform bears per delayed meal supplied by restaurant i.

The key decision variables in the following developed model (2) are binary decision variables $x_{i,t} \in \{0,1\}$, which denote whether restaurant i still accept orders at time t and characterize the deadline of order placement. The objective function (2a) of the developed optimization model is to maximize the worst-case expected profit of the on-demand service platform by finding optimal $x_{i,t}$. The profit of the platform equals the total commission fee (i.e., $\sum_{i\in[R]} p_i^c(\sum_{t=0}^T \tilde{d}_{i,t}x_{i,t} - h_{i,T-\tilde{v}_i})$) minus total delayed penalty cost (i.e., $\sum_{i\in[R]} \beta_i h_{i,T-\tilde{v}_i}$). To be specific, given the value of $x_{i,t}$, the quantity of successful orders for restaurants i equals $\sum_{t=0}^T \tilde{d}_{i,t}x_{i,t} - h_{i,T-\tilde{v}_i}$, which equals the quantity of total orders accepted $\sum_{t=0}^T \tilde{d}_{i,t}x_{i,t}$ minus the quantity of delayed orders $h_{i,T-\tilde{v}_i}$. Other operational costs (e.g., online platform construction and maintenance, HSR meal delivery staffs and crews pay) are almost fixed and omitted in the optimization model.

We consider uncertainty in delivery time $\tilde{\boldsymbol{v}}$. As the probability distribution of $\tilde{\boldsymbol{v}}$ is known, the expectation over $\tilde{\boldsymbol{v}}$ can be calculated based on definition of expected value. Uncertainty in demands $\tilde{\boldsymbol{d}}$ is also considered, but its distribution is not exactly known but characterized by ambiguity set $\mathcal{F}(\underline{\boldsymbol{\mu}}, \bar{\boldsymbol{\mu}}, \bar{\boldsymbol{\eta}})$. We aims at maximizing the worst-case expected profit of the platform across all possible probability distributions of $\tilde{\boldsymbol{d}}$ in the set $\mathcal{F}(\underline{\boldsymbol{\mu}}, \bar{\boldsymbol{\mu}}, \bar{\boldsymbol{\eta}})$. To find the worst-case expected profit, we need to solve a minimization problem with respect to demand probabilities $\mathbb{P}$ subject to constraints in (1). This is the inner problem of (2a).

We impose Constraints (2b) to (2e) to depict the on-demand meal service process. Constraint (2b) is the order deadline constraints. Note that the latest t satisfies $x_{i,t} = 1$ corresponds to the order deadline of restaurant i. Constraint (2b) guarantees that all orders occur before the deadline is accepted while all orders occur after the deadline is not. Constraint (2c) defines the relationship between quantity of accepted orders and accumulated unprepared meals. Specifically, the quantity of unprepared orders for restaurant i at time t equals the maximum of zero and $h_{i,t-1} + \tilde{d}_{i,t}x_{i,t} - V_{i,t}$, which refers to the quantity of unprepared orders at time $t-1$ plus the number of orders accepted at time t minus the quantity of orders the restaurant prepared during time period $(t-1,t)$. Constraint (2d) imposes the quantity unprepared orders at time 0. Constraint (2e) indicates variable domain.

$$\max_{\boldsymbol{x}} \min_{\mathbb{P}\in\mathcal{F}(\underline{\boldsymbol{\mu}},\bar{\boldsymbol{\mu}},\bar{\boldsymbol{\eta}})} \mathbb{E}\left[\sum_{i\in[R]}\left(p_i^c(\sum_{t=0}^T \tilde{d}_{i,t}x_{i,t} - h_{i,T-\tilde{v}_i}) - \beta_i h_{i,T-\tilde{v}_i}\right)\right] \tag{2a}$$

$$s.t.\ x_{i,t} \geq x_{i,t+1}, \forall i\in[R], t\leq T-1 \tag{2b}$$

$$h_{i,t} = \max\left\{h_{i,t-1} + \tilde{d}_{i,t}x_{i,t} - V_{i,t}, 0\right\}, \forall i\in[R], 1\leq t\leq T \tag{2c}$$

$$h_{i,0} = \tilde{d}_{i,0}, \forall i\in[R] \tag{2d}$$

$$\boldsymbol{x}\in\{0,1\} \tag{2e}$$

Model (2) considers multiple restaurants together with their demands characterized by joint probability distribution in ambiguity set $\mathcal{F}(\underline{\boldsymbol{\mu}}, \bar{\boldsymbol{\mu}}, \bar{\boldsymbol{\eta}})$. If the demands of different restaurants are independent, then model (2) can be solved for each restaurants separately.

The proposed model (2) is a distributionally robust stochastic optimization model that optimizes the order deadline of the HSR on-demand meal ordering system. It is a two-stage mixed integer programming (MIP), which cannot be solved directly by off-the-shelf algorithms and solvers. In the following, we will reformulate the problem and propose a cutting-plane algorithm for solving this problem efficiently.

4.2. Reformulation of order deadline optimization model

In this section, we work on transforming Problem (2) into a tractable form. First, we focus on its inner expectation minimization problem of objective function (2a). There are two types of uncertain factors in model (2): demand and delivery time. The distribution of demand is ambiguous while the distribution of the delivery time is known. We first deal with the uncertainty in delivery time. Note that the delivery time $\tilde{\boldsymbol{v}}$ only affects $h_{i,T-\tilde{v}_i}$ in (2a) and the probability distribution of delivery time $\tilde{v}_i$ is known and denoted by $P_{i,s}, s=1,\ldots,S_i$. Based the definition of expected value, the expectation of inner problem in objective function (2a) with respect to $\tilde{\boldsymbol{v}}$ can be transformed into the following equivalent form:

$$\begin{aligned}&\min_{\mathbb{P}\in\mathcal{F}(\underline{\boldsymbol{\mu}},\bar{\boldsymbol{\mu}},\bar{\boldsymbol{\eta}})} \mathbb{E}\left[\sum_{i\in[R]}\left(p_i^c(\sum_{t=0}^T \tilde{d}_{i,t}x_{i,t} - h_{i,T-\tilde{v}_i}) - \beta_i h_{i,T-\tilde{v}_i}\right)\right]\\ =&\min_{\mathbb{P}\in\mathcal{F}(\underline{\boldsymbol{\mu}},\bar{\boldsymbol{\mu}},\bar{\boldsymbol{\eta}})} \mathbb{E}_{\mathbb{P}}\left[\sum_{i\in[R]}\left(p_i^c\sum_{t=0}^T \tilde{d}_{i,t}x_{i,t} - (p_i^c+\beta_i)\sum_{s=1}^{S_i} P_{i,s}h_{i,T-v_{i,s}}\right)\right].\end{aligned} \tag{3}$$

Note that the outer problem of (2a) is a maximization problem. To make this problem tractable, we transform the inner problem (3) to a maximization problem by duality theory as follows.

Problem (3) aims to identify the minimal expectation across all probability distribution $\mathbb{P}$ in ambiguity set $\mathcal{F}(\underline{\boldsymbol{\mu}}, \bar{\boldsymbol{\mu}}, \bar{\boldsymbol{\eta}})$. In particular, it is a minimization problem with decision variables $\mathbb{P}$ (the joint probabilities of demands) subject to constraints given in Eq. (1). Note that expected value is a linear function in terms of $\mathbb{P}$ and all of the three types of constraints in Eq. (1) could be expressed as linear constraints with respect to $\mathbb{P}$ (see details in Proof of Proposition 1 in Appendix). Hence, Problem (3) is a linear programming and it satisfies strong duality as long as it is feasible. Let $\lambda, \boldsymbol{\gamma}, \boldsymbol{\theta}, \boldsymbol{\pi}$ be dual variables that correspond to the constraints on valid probability distribution, lower and upper marginal expectation bound, and average absolute deviation bound respectively. The dual of Problem (3) is presented in the following Proposition.

Proposition 1. *Given x, Problem (3) can be reformulated as follows:*

$$\max_{\lambda,\gamma,\theta,\pi} -\lambda + \sum_{i\in[R]}\left(\sum_{t=0}^{T}\left(-\gamma_{i,t}\bar{\mu}_{i,t} + \theta_{i,t}\underline{\mu}_{i,t} - \pi_{i,t}\bar{\eta}_{i,t}\right)\right) \tag{4a}$$

$$s.t.\ \min_{\tilde{d}\in W} \sum_{i\in[R]}\left(\sum_{t=0}^{T}\left(p_i^c\tilde{d}_{i,t}x_{i,t} + (\gamma_{i,t}-\theta_{i,t})\tilde{d}_{i,t} + \pi_{i,t}|\tilde{d}_{i,t}-\mu_{i,t}|\right) - (p_i^c+\beta_i)\sum_{s=1}^{S_i}P_{i,s}h_{i,T-v_{i,s}}\right) \geq -\lambda \tag{4b}$$

$$\gamma,\theta,\pi \geq 0 \tag{4c}$$

where W is the support for $\tilde{d}$ and $\lambda,\gamma,\theta,\pi$ are dual variables.

In this step, we successfully convert Problem (3) into a maximization problem with a main constraint in (4b). However, the left-hand-side (LHS) of Constraint (4b) is a minimization problem with respect to $\tilde{d}$, which impedes solving the optimization model directly. Moreover, note that $h_{i,T-v_i}$ is a function of $\tilde{d}_{i,t}$ given by Constraints (2c) and (2d), which are not linear. These difficulties are resolved in Proposition 2.

Proposition 2. *The LHS of the robust constraint (4b) can be reformulated as follows:*

$$G(x,\gamma,\theta,\pi) = \min_{d,h,I,z} \sum_{i\in[R]}\left(\sum_{t=0}^{T}\left(p_i^c\tilde{d}_{i,t}x_{i,t} + (\gamma_{i,t}-\theta_{i,t})\tilde{d}_{i,t} + \pi_{i,t}z_{i,t}\right) - (p_i^c+\beta_i)\sum_{s=1}^{S_i}P_{i,s}h_{i,T-v_{i,s}}\right) \tag{5a}$$

$$s.t.\ \underline{d}_{i,t} \leq \tilde{d}_{i,t} \leq \bar{d}_{i,t}, \forall i\in[R], t\leq T \tag{5b}$$

$$z_{i,t} \geq \tilde{d}_{i,t} - \mu_{i,t}, \forall i\in[R], t\leq T \tag{5c}$$

$$z_{i,t} \geq -\tilde{d}_{i,t} + \mu_{i,t}, \forall i\in[R], t\leq T \tag{5d}$$

$$h_{i,t} \geq h_{i,t-1} + \tilde{d}_{i,t}x_{i,t} - V_{i,t}, \forall i\in[R], 1\leq t\leq T \tag{5e}$$

$$h_{i,t} \leq \sum_{t'=0}^{t}\bar{d}_{i,t'}I_{i,t}, \forall i\in[R], 1\leq t\leq T \tag{5f}$$

$$h_{i,t} \leq h_{i,t-1} + \tilde{d}_{i,t}x_{i,t} - V_{i,t} + \sum_{t'=0}^{t}\bar{d}_{i,t'}(1-I_{i,t}), \forall i\in[R], 1\leq t\leq T \tag{5g}$$

$$h_{i,0} = \tilde{d}_{i,0}, \forall i\in[R] \tag{5h}$$

$$d,h,z \geq 0, I\in\{0,1\} \tag{5i}$$

In Problem (5), x,γ,θ,π are given constants while d,h,I,z are decision variables. Constraint (5b) states the bound of demand $\tilde{d}_{i,t}$. Continuous decision variables $z_{i,t}$ are introduced to transform absolute value $|\tilde{d}_{i,t}-\mu_{i,t}|$ into linear constraints (5c) and (5d). In addition, binary decision variables $I_{i,t}$ are introduced to transform the relationship $h_{i,t} = \max\{h_{i,t-1} + \tilde{d}_{i,t}x_{i,t} - V_{i,t}, 0\}$ into linear integer constraints (5e) to (5g).

Proposition 2 transforms the LHS of Constraint (4b) into a tractable mixed integer linear programming (MILP) Problem (5). Based on Propositions 1 and 2, we transform the distributionally robust order deadline optimization model (2) as follows.

Theorem 1. *Problem (2) can be transformed by the following reformulation:*

$$\max_{x,\lambda,\gamma,\theta,\pi} -\lambda + \sum_{i\in[R]}\left(\sum_{t=0}^{T}\left(-\gamma_{i,t}\bar{\mu}_{i,t} + \theta_{i,t}\underline{\mu}_{i,t} - \pi_{i,t}\bar{\eta}_{i,t}\right)\right) \tag{6a}$$

$$s.t.\ G(x,\gamma,\theta,\pi) \geq -\lambda \tag{6b}$$

$$x_{i,t} \geq x_{i,t+1}, \forall i\in[R], t\leq T-1 \tag{6c}$$

$$x\in\{0,1\},\ \gamma,\theta,\pi\geq 0 \tag{6d}$$

Theorem 1 transforms Problem (2) into a one-stage maximization optimization problem (6). The key challenge in solving Problem (6) is that Constraint (6b) requires $G(x,\gamma,\theta,\pi)$ to be greater than $-\lambda$, where $G(x,\gamma,\theta,\pi)$ is a minimization MILP problem that is a function of (x,γ,θ,π). If we solve problem (6) directly by expressing Constraint (6b) as originally (4b), we need to involve all possible combinations of scenarios of demand $\tilde{d}_{i,t}, \forall i\in[R], t\leq T$. The size of the system of constraints equivalent to Constraint (4b) will be $\prod_{i\in[R]}\prod_{t=0}^{T}|\tilde{d}_{i,t}|$, which is intractably large. As a result, we have to find a feasible and tractable way to solve the model. Note that given a solution $(x,\lambda,\gamma,\theta,\pi)$, the value of $G(x,\gamma,\theta,\pi)$ could be calculated by off-the-shelf tools relatively easy and whether Constraint (6b) is satisfied can be checked. Hence, we propose a cutting-plane algorithm to solve Problem (6) based on this observation. In this algorithm, we first relax Constraint (6b) with some tractable constraints and find a candidate solution by solving the relaxed problem. Then we check whether the candidate solution satisfies Constraint (6b). If yes, then the obtained solution is the optimal solution. Otherwise, new cuts will be introduced to the relaxed problem until the optimal solution is found.

4.3. Cutting-plane algorithm for solving order deadline optimization model

Note that $G(\boldsymbol{x},\boldsymbol{\gamma},\boldsymbol{\theta},\boldsymbol{\pi})$ is an optimization problem with respect to $\tilde{\boldsymbol{d}}$ and $\boldsymbol{h}$ but in essence it merely depends on value of $\tilde{\boldsymbol{d}}$ because value of $\boldsymbol{h}$ is determined given $\tilde{\boldsymbol{d}}$. Given a known solution $d_{i,t}^k$ and $h_{i,t}^k$ of $G(\boldsymbol{x},\boldsymbol{\gamma},\boldsymbol{\theta},\boldsymbol{\pi})$, if Constraint (6b) is satisfied, then the following relationship always holds:

$$\sum_{i\in[R]}\left(\sum_{t=0}^{T}\left(p_i^c d_{i,t}^k x_{i,t}+(\gamma_{i,t}-\theta_{i,t})d_{i,t}^k+\pi_{i,t}|d_{i,t}^k-\mu_{i,t}|\right)-(p_i^c+\beta_i)\sum_{s=1}^{S_i}P_{i,s}h_{i,T-v_{i,s}}^k\right)\geq-\lambda. \tag{7}$$

Hence, Constraint (7) is a necessary condition of Constraint (6b) and Problem (6) can be relaxed based on this condition. The relaxed problem could provide a candidate optimal solution of Problem (6) which has better or equivalent performance than all feasible solutions of Problem (6) but may not satisfy Constraint (6b). If the found candidate solution is proved to satisfy all constraints, especially Constraint (6b) in Problem (6), then this solution is the optimal solution to the original problem. Otherwise, new cuts should be added to refine the relaxed problem until the optimal solution is found.

Based on the above result, we develop a cutting-plane algorithm to solve the reformulated order deadline optimization Problem (6). In this algorithm, the master problem is a relaxed problem that relaxes Constraint (6b) based on multiple known scenarios of demands $d_{i,t}^k$ and their corresponding $h_{i,t}^k$. The master problem at Kth iteration, denoted as $\mathbf{MASTER}_E$ is presented in the following:

$$\max_{\boldsymbol{x},\boldsymbol{\lambda},\boldsymbol{\gamma},\boldsymbol{\theta},\boldsymbol{\pi},\boldsymbol{h}}-\lambda+\sum_{i\in[R]}\left(\sum_{t=0}^{T}\left(-\gamma_{i,t}\bar{\mu}_{i,t}+\theta_{i,t}\underline{\mu}_{i,t}-\pi_{i,t}\bar{\eta}_{i,t}\right)\right) \tag{8a}$$

$$\text{s.t. } x_{i,t}\geq x_{i,t+1},\forall i\in[R],t\leq T-1 \tag{8b}$$

$$\sum_{i\in[R]}\left(\sum_{t=0}^{T}\left(p_i^c d_{i,t}^k x_{i,t}+(\gamma_{i,t}-\theta_{i,t})d_{i,t}^k+\pi_{i,t}|d_{i,t}^k-\mu_{i,t}|\right)-(p_i^c+\beta_i)\sum_{s=1}^{S_i}P_{i,s}h_{i,T-v_{i,s}}^k\right)$$
$$\geq-\lambda,k=0,\dots,K-1 \tag{8c}$$

$$h_{i,t}^k\geq h_{i,t-1}^k+d_{i,t}^k x_{i,t}-V_{i,t},\forall i\in[R],1\leq t\leq T,k=0,\dots,K-1 \tag{8d}$$

$$h_{i,0}^k=d_{i,0}^k,\forall i\in[R],k=0,\dots,K-1 \tag{8e}$$

$$\boldsymbol{x}\in\{0,1\},\ \ \boldsymbol{h},\boldsymbol{\gamma},\boldsymbol{\theta},\boldsymbol{\pi}\geq 0 \tag{8f}$$

In each iteration of this algorithm, a new solution $(\boldsymbol{x},\lambda,\boldsymbol{\gamma},\boldsymbol{\theta},\boldsymbol{\pi},\boldsymbol{h})$ is obtained by solving the master problem $\mathbf{MASTER}_E$. This solution is passed to the minimization problem $G(\boldsymbol{x},\boldsymbol{\gamma},\boldsymbol{\theta},\boldsymbol{\pi})$ to check whether Constraint (6b), which is relaxed in the master problem, is satisfied. If yes, then the obtained solution is the optimal solution of Problem (6). Otherwise, the system of Constraints (8c) to (8e) corresponding to the optimal solution of $G(\boldsymbol{x},\boldsymbol{\gamma},\boldsymbol{\theta},\boldsymbol{\pi})$ in this iteration will be introduced to the master problem to further refine the feasible set. The detailed steps of the cutting-plane algorithm for solving Problem (6) are presented in Algorithm 1

Algorithm 1 A cutting-plane algorithm for solving the order deadline optimization problem

Step 1: **Initialize**. Set $K=0$. Select an initialized value of $d_{i,t}^0$ for each $i\in[R],t\leq T$. A pre-defined non-negative tolerance value ϵ.

Step 2: Solve $\mathbf{MASTER}_E$ problem (8) and let $(\boldsymbol{x}^{\boldsymbol{K}},\lambda^{\boldsymbol{K}},\boldsymbol{\gamma}^{\boldsymbol{K}},\boldsymbol{\theta}^{\boldsymbol{K}},\boldsymbol{\pi}^{\boldsymbol{K}})$ denote the obtained optimal solution.

Step 3: solve $G(\boldsymbol{x}^{\boldsymbol{K}},\boldsymbol{\gamma}^{\boldsymbol{K}},\boldsymbol{\theta}^{\boldsymbol{K}},\boldsymbol{\pi}^{\boldsymbol{K}})$ defined in (5) and let $d_{i,t}^K$ the optimal solution of $G(\boldsymbol{x}^{\boldsymbol{K}},\boldsymbol{\gamma}^{\boldsymbol{K}},\boldsymbol{\theta}^{\boldsymbol{K}},\boldsymbol{\pi}^{\boldsymbol{K}})$.

- If $G(\boldsymbol{x}^{\boldsymbol{K}},\boldsymbol{\gamma}^{\boldsymbol{K}},\boldsymbol{\theta}^{\boldsymbol{K}},\boldsymbol{\pi}^{\boldsymbol{K}})<-\lambda^K-\epsilon$, then add the following cuts corresponding to $d_{i,t}^K$ to the $\mathbf{MASTER}_E$ problem and go back to Step 2.

$$\sum_{i\in[R]}\left(\sum_{t=0}^{T}\left(p_i^c d_{i,t}^K x_{i,t}+(\gamma_{i,t}-\theta_{i,t})d_{i,t}^K+\pi_{i,t}|d_{i,t}^K-\mu_{i,t}|\right)-(p_i^c+\beta_i)\sum_{s=1}^{S_i}P_{i,s}h_{i,T_i-v_{i,s}}^K\right)\geq-\lambda$$
$$h_{i,t}^K\geq h_{i,t-1}^K+d_{i,t}^K x_{i,t}-V_{i,t},\forall i\in[R],1\leq t\leq T$$
$$h_{i,t}^K\geq 0,\forall i\in[R],t\leq T$$
$$h_{i,0}^K=d_{i,0}^K,\forall i\in[R]$$

- Otherwise, terminate and return solution $(\boldsymbol{x}^{\boldsymbol{K}},\lambda^{\boldsymbol{K}},\boldsymbol{\gamma}^{\boldsymbol{K}},\boldsymbol{\theta}^{\boldsymbol{K}},\boldsymbol{\phi}^{\boldsymbol{K}})$

Step 4: Set $K=K+1$ and go back to step 2.

Step 5: Repeat Step 2, 3 and 4 until the algorithm terminates when the termination condition in Step 3 is satisfied.

In each iteration, new decision variables $h_{i,t}^K$ and constraints correspond to the solution of $G(\boldsymbol{x},\boldsymbol{\gamma},\boldsymbol{\theta},\boldsymbol{\pi})$ are introduced into the master problem. The number of constraints of $\mathbf{MASTER}_E$ at Kth iteration is $(T-1)R+K(1+R+TR)$, which is linear to number of

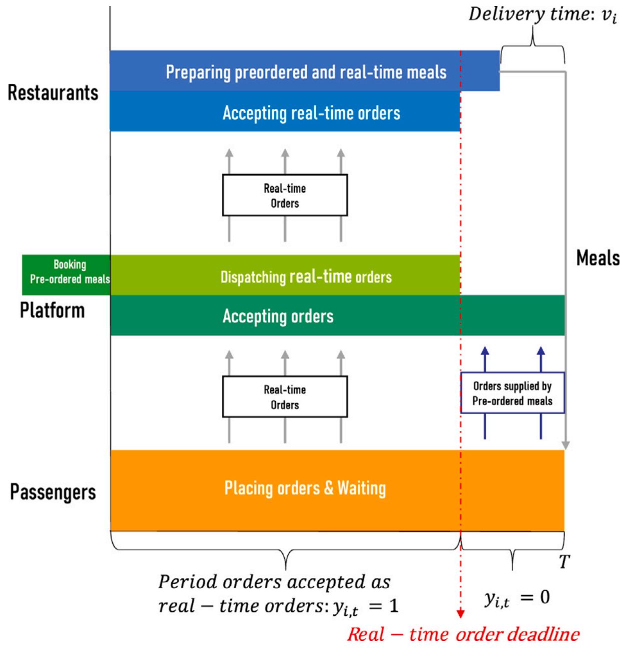


Fig. 3. Illustration of pre-ordering policy model.

iteration. The total number of decision variables is $4TR + KTR + 1$ with number of binary decision variables always equal TR. The unchanged number of binary decision variables makes complexity of solving the master problem grows very slowly. On the other hand, the size of $G(\boldsymbol{x}, \boldsymbol{\gamma}, \boldsymbol{\theta}, \boldsymbol{\pi})$ remains unchanged as number of iteration grows and our computational studies show that it can be solved very quickly. To be specific, $G(\boldsymbol{x}, \boldsymbol{\gamma}, \boldsymbol{\theta}, \boldsymbol{\pi})$ has $6TR + R$ constraints and $4TR$ decision variables among which TR are binary.

5. New order management policy and its optimization

The existing HSR on-demand meal ordering procedure sets a deadline for the acceptance of orders to guarantee timely delivery. However, this policy reduces the service availability and also loses potential demands arising after the deadline. In this section, we propose a novel pre-ordering policy to address these issues by stocking up in advance, or called as pre-ordering. This new policy allows customers to place orders at any time. To cope with the requirement that all ordered meals must be delivered to the station before train departure time, extra meals will be booked in advance to satisfy orders placed after the latest delivery time.

5.1. Pre-ordering policy

Here we introduce how the novel pre-ordering policy is designed in detail (demonstrated in Fig. 3). The system consists of three parts: passengers, the platform and restaurants. Under the new policy, passengers are allowed to place an order at any time before the train departure time of the station with on-demand meal delivery service. However, these orders are divided into two categories according to order placement time. A deadline will be set by the platform. All orders placed before the deadline will be regarded as real-time orders and these orders will be accepted by the platform, assigned to restaurants to be prepared. All orders placed after the deadline will be satisfied by meals the platform booked in advance from the restaurants. To be specific, the platform pre-orders extra meals to satisfy orders arising after the real-time order deadline. These pre-ordered meals will be prepared and delivered to the train before the train departure time with a predetermined schedule. These meals can also be used to satisfy delayed real-time orders when available. However, if the pre-ordered meals are not sold out in the end, these meals will be wasted.

The platform will charge p_i^c commission fee per meal successfully executed (i.e., ordered by the customers and delivered to them on time). However, profits and risks of pre-ordered meals will be shared by the platform and restaurants. In particular, the platform will pay c_i pre-ordering fee per pre-ordered meal to restaurant $i \in [R]$. If one pre-ordered meal is sold in the end, then the platform will gain $p_i^c - c_i$ and the restaurant will gain $p_i^s + c_i - p_i^c$, where p_i^s denotes the total profit per meal. On the other hand, if one pre-ordered meal is not sold, then the platform will bear loss of c_i while the restaurant will bear loss of $c_i - c_i^s$, where c_i^s denotes the unit cost of providing one meal.

The pre-ordering policy could be regarded as a combination of “Make-to-Order” and “Make-to-Stock” in on-demand HSR meal ordering service. This policy takes “Make-to-Order” mode before the pre-determined deadline and prepare and deliver meals only when orders are confirmed. After the deadline, “Make-to-Stock” mode is taken, that is, orders preparing and delivery are based on anticipated demands before the order comes. Technically, there is no difficulty in realizing this policy. From a profit perspective, as long as the setting of deadline and quantity of pre-ordered meals are delicately set such that the profit of both the platform and restaurants can be improved, this policy will be welcomed by involved parties. In this following of section, we will explore on the feasibility of our new proposed pre-ordering and aim at identifying the best pre-ordering policy such that the profit of the platform is maximized.

This problem includes the following decision variables. We introduce binary decision variables $y_{i,t}$ to represent whether orders for meals to be served by restaurant i placed at time t are accepted as real-time orders: $y_{i,t} = 1$ if so and $y_{i,t} = 0$ otherwise. Decision variable e_i refers to the quantity of pre-ordered meals booked in advance from restaurant i with unit pre-ordering fee c_i. Decision variable q_i denotes the quantity of orders successfully executed, that is, meals ordered and successfully delivered to passengers by restaurant i, and decision variable δ_i indicates the quantity of delayed meals, that is, meals ordered by passengers to served by restaurant i but not delivered to them successfully. Similar to the order deadline optimization model we proposed above, we consider real-time order deadline constraint (9b) and accumulated unprepared real-time order constraints (15d) and (15e). The best pre-ordering policy is derived from the following distributionally robust optimization model:

$$\max_{\boldsymbol{e},\boldsymbol{y},\boldsymbol{h}} \left\{ \sum_{i\in[R]} -c_i e_i + \min_{\mathbb{P}\in\mathcal{F}} \mathbb{E}[\sum_{i\in[R]} g_i(\boldsymbol{e},\boldsymbol{y},\tilde{\boldsymbol{d}},\boldsymbol{h},\tilde{v}_i)] \right\} \tag{9a}$$

$$s.t\ y_{i,t} \geq y_{i,t+1}, \forall i \in [R], t \leq T-1 \tag{9b}$$

$$h_{i,t} = \max\left\{ h_{i,t-1} + \tilde{d}_{i,t} y_{i,t} - V_{i,t}, 0 \right\}, \forall i \in [R], 1 \leq t \leq T \tag{9c}$$

$$h_{i,0} = \tilde{d}_{i,0}, \forall i \in [R] \tag{9d}$$

$$\boldsymbol{y} \in \{0,1\}, \boldsymbol{e},\boldsymbol{h} \geq \boldsymbol{0} \tag{9e}$$

where $g_i(\boldsymbol{e},\boldsymbol{y},\tilde{\boldsymbol{d}},\boldsymbol{h},\tilde{v}_i) = p_i^c q_i - \beta_i \delta_i$ is the total revenue gained by commission from restaurants minus delayed penalty caused by restaurant i. Here, q_i denotes the quantity of successfully executed orders defined to be the minimal value of total demand and meals successfully delivered to the station (i.e., supply), mathematically:

$$q_i = \min\{\sum_{t\leq T} \tilde{d}_{i,t},\ \sum_{t\leq T} \tilde{d}_{i,t} y_{i,t} - h_{i,T-\tilde{v}_i} + e_i\}.$$

And δ_i refers to the quantity of delayed orders, which equals maximal of the delayed real-time orders plus insufficiency (or excess) of pre-orders and zero, mathematically given by:

$$\delta_i = \max\{h_{i,T-\tilde{v}_i} + \sum_{t\leq T} \tilde{d}_{i,t}(1-y_{i,t}) - e_i,\ 0\},$$

where $\sum_{t\leq T} \tilde{d}_{i,t}(1-y_{i,t}) - e_i$ is quantity of orders arising after the real-time order deadline minus the quantity of pre-ordered meals, i.e., insufficiency (or excess) of pre-order amount.

Observe that the proposed model (9) is a two-stage MILP with complex constraints which are hard to solve. With similar procedures in Section 4, we reformulate Problem (9) and propose a cutting-plane algorithm to solve it. For conciseness, we present relevant propositions in the Appendix and we only present main results in the main body of this work.

Theorem 2. *Problem (9) could be reformulated as follows:*

$$\max_{\boldsymbol{e},\boldsymbol{y},\lambda,\boldsymbol{\gamma},\boldsymbol{\theta},\boldsymbol{\pi}} \left\{ \sum_{i\in[R]} -c_i e_i - \lambda + \sum_{i\in[R]} \sum_{t=0}^{T} \left(-\bar{u}_{i,t}\gamma_{i,t} + \underline{\mu}_{i,t}\theta_{i,t} - \eta_{i,t}\pi_{i,t} \right) \right\} \tag{10a}$$

$$s.t\ y_{i,t} \geq y_{i,t+1}, \forall i \in [R], t \leq T-1 \tag{10b}$$

$$G'(\boldsymbol{y},\boldsymbol{e},\boldsymbol{\gamma},\boldsymbol{\theta},\boldsymbol{\pi}) \geq -\lambda \tag{10c}$$

$$\boldsymbol{y} \in \{0,1\} \tag{10d}$$

$$\boldsymbol{e},\boldsymbol{\gamma},\boldsymbol{\theta},\boldsymbol{\pi} \geq 0 \tag{10e}$$

with $G'(\boldsymbol{y},\boldsymbol{e},\boldsymbol{\gamma},\boldsymbol{\theta},\boldsymbol{\pi})$ defined to be the following optimization problem:

$$\begin{aligned}
\min_{\tilde{\boldsymbol{d}},\boldsymbol{q},\boldsymbol{\delta},\boldsymbol{h},\boldsymbol{I},\boldsymbol{z}} & \left\{ \sum_{i\in[R]} \sum_{s=1}^{S_i} P_{i,s} \left(p_i^c q_{i,s} - \beta_i \delta_{i,s} \right) + \sum_{i\in[R]} \sum_{t=0}^{T} \left((\gamma_{i,t} - \theta_{i,t})\tilde{d}_{i,t} + \pi_{i,t} z_{i,t} \right) \right\} \\
s.t.\ & \underline{d}_{i,t} \leq \tilde{d}_{i,t} \leq \bar{d}_{i,t}, \forall i \in [R], t \leq T \\
& z_{i,t} \geq \tilde{d}_{i,t} - \mu_{i,t}, \forall i \in [R], t \leq T \\
& z_{i,t} \geq -\tilde{d}_{i,t} + \mu_{i,t}, \forall i \in [R], t \leq T \\
& h_{i,t} \geq h_{i,t-1} + \tilde{d}_{i,t} x_{i,t} - V_{i,t}, \forall i \in [R], 1 \leq t \leq T \\
& h_{i,t} \leq \sum_{t'=0}^{t} \bar{d}_{i,t'} I_{i,t}, \forall i \in [R], t \leq T
\end{aligned}$$

$$h_{i,t} \leq h_{i,t-1} + \tilde{d}_{i,t}x_{i,t} - V_{i,t} + \sum_{t'=0}^{t} \bar{d}_{i,t'}(1 - I_{i,t}), \forall i \in [R], 1 \leq t \leq T$$
$$h_{i,0} = \tilde{d}_{i,0}, \forall i \in [R]$$
$$q_{i,s} \leq \sum_{t \leq T} \tilde{d}_{i,t}, \forall i \in [R], s = 1, \ldots, S_i$$
$$q_{i,s} \leq \sum_{t \leq T} \tilde{d}_{i,t}y_{i,t} - h_{i,T-v_{i,s}} + e_i, \forall i \in [R], s = 1, \ldots, S_i$$
$$q_{i,s} \geq \sum_{t \leq T} \tilde{d}_{i,t} - \sum_{t=0}^{T} \bar{d}_{i,t}I_{i,q,s}, \forall i \in [R], s = 1, \ldots, S_i$$
$$q_{i,s} \geq \sum_{t \leq T} \tilde{d}_{i,t}y_{i,t} - h_{i,T-v_{i,s}} + e_i - \sum_{t=0}^{T} \bar{d}_{i,t}(1 - I_{i,q,s}), \forall i \in [R], s = 1, \ldots, S_i$$
$$\delta_{i,s} \geq h_{i,T-v_{i,s}} + \sum_{t \leq T} \tilde{d}_{i,t}(1 - y_{i,t}) - e_i, \forall i \in [R], s = 1, \ldots, S_i$$
$$\delta_{i,s} \leq \sum_{t=0}^{T} \bar{d}_{i,t}I_{i,\delta,s}, \forall i \in [R], s = 1, \ldots, S_i$$
$$\delta_{i,s} \leq h_{i,T-v_{i,s}} + \sum_{t \leq T} \tilde{d}_{i,t}(1 - y_{i,t}) - e_i + \sum_{t=0}^{T} \bar{d}_{i,t}(1 - I_{i,\delta,s}), \forall i \in [R], s = 1, \ldots, S_i$$
$$\tilde{\boldsymbol{d}}, \boldsymbol{h}, \boldsymbol{q}, \boldsymbol{\delta}, \boldsymbol{z} \geq 0, \boldsymbol{I} \in \{0, 1\}.$$

Theorem 2 transforms Problem (9) into a maximization problem with a complex Constraint (10c), which contain a minimization problem $G'(\boldsymbol{y}, \boldsymbol{e}, \boldsymbol{\gamma}, \boldsymbol{\theta}, \boldsymbol{\pi})$. However, Constraint (10c) still makes it impossible to solve the problem directly. To address this issue, we also develop a cutting-plane algorithm presented in the following section.

5.2. Cutting-plane algorithm for solving the pre-ordering policy optimization model

Algorithm 2 A cutting-plane algorithm for solving the pre-ordering policy optimization model

Step 1: **Initialize**. Set $K = 0$. Set an initialized value of $d^0_{i,t}$ for each $i \in [R], t \leq T$. Define a predefined non-negative tolerance value ϵ.

Step 2: Solve $\mathbf{MASTER}_P$ (12) and let $(\boldsymbol{y^K}, \boldsymbol{e^K}, \boldsymbol{\lambda^K}, \boldsymbol{\gamma^K}, \boldsymbol{\theta^K}, \boldsymbol{\pi^K})$ denote the obtained optimal solution.

Step 3: Solve $G'(\boldsymbol{y^K}, \boldsymbol{e^K}, \boldsymbol{\gamma^K}, \boldsymbol{\theta^K}, \boldsymbol{\pi^K})$ defined in (16) and let $d^K_{i,t}$ denote the optimal solution of $G'(\boldsymbol{y^K}, \boldsymbol{e^K}, \boldsymbol{\gamma^K}, \boldsymbol{\theta^K}, \boldsymbol{\pi^K})$.

- If $G'(\boldsymbol{y^K}, \boldsymbol{e^K}, \boldsymbol{\gamma^K}, \boldsymbol{\theta^K}, \boldsymbol{\pi^K}) < -\lambda^K - \epsilon$, then add the following constraints to $\mathbf{MASTER}_P$ problem:

$$\sum_{i \in [R]} \sum_{s=1}^{S_i} P_{i,s}\left(p^c_i q^K_{i,s} - \beta_i \delta^K_{i,s}\right) + \sum_{i \in [R]} \sum_{t=0}^{T} \left((\gamma_{i,t} - \theta_{i,t})d^K_{i,t} + (|d^K_{i,t} - \mu_{i,t}|)\pi_{i,t}\right) \geq -\lambda$$
$$h^K_{i,t} \geq h^K_{i,t-1} + d^K_{i,t}y_{i,t} - V_{i,t}, \ \ \forall i \in [R], 1 \leq t \leq T$$
$$h^K_{i,0} = d^K_{i,0}, \forall i \in [R]$$
$$q^K_{i,s} \leq \sum_{t \leq T} d^K_{i,t}, \forall i \in [R], s = 1, \ldots, S_i$$
$$q^K_{i,s} \leq \sum_{t \leq T} d^K_{i,t}y_{i,t} - h^K_{i,T_i-v_{i,s}} + e_i, \forall i \in [R], s = 1, \ldots, S_i$$
$$\delta^K_{i,s} \geq \sum_{t \leq T} d^K_{i,t}(1 - y_{i,t}) + h^K_{i,T_i-v_{i,s}} - e_i, \forall i \in [R], s = 1, \ldots, S_i$$
$$\boldsymbol{h^K}, \boldsymbol{q^K}, \boldsymbol{\delta^K} \geq 0$$

- Otherwise, terminate and return solution $(\boldsymbol{y^K}, \boldsymbol{e^K}, \boldsymbol{\lambda^K}, \boldsymbol{\gamma^K}, \boldsymbol{\theta^K}, \boldsymbol{\pi^K})$

Step 4: Set $K = K + 1$ and go back to step 2.

Step 5: Repeat Step 2, 3 and 4 until the algorithm terminates when the termination condition in Step 3 is satisfied.

The idea of the cutting-plane algorithm for solving Problem (10) is similar to Algorithm 1. Given the values of $(\boldsymbol{y}, \boldsymbol{e}, \boldsymbol{\gamma}, \boldsymbol{\theta}, \boldsymbol{\pi})$, $G'(\boldsymbol{y}, \boldsymbol{e}, \boldsymbol{\gamma}, \boldsymbol{\theta}, \boldsymbol{\pi})$ in Problem (10) is a minimization problem in essence with respect to the values of $\tilde{\boldsymbol{d}}$. With similar derivation procedure, we could relax Constraint (10c) by a system of linear constraints based on known scenarios of demands $\tilde{\boldsymbol{d}}$. Let $\boldsymbol{d}^k, k = 0, \dots, K-1$ denote the optimal solution of $G'(\boldsymbol{y}, \boldsymbol{e}, \boldsymbol{\gamma}, \boldsymbol{\theta}, \boldsymbol{\pi})$ in the kth iteration of the algorithm. The master problem of at Kth iteration, denoted as $\mathbf{MASTER}_P$, is as follows:

$$\max_{\boldsymbol{e},\boldsymbol{y},\boldsymbol{\lambda},\boldsymbol{\gamma},\boldsymbol{\theta},\boldsymbol{\pi},\boldsymbol{h},\boldsymbol{q},\boldsymbol{\delta}} \left\{ \sum_{i\in[R]} -c_i e_i - \lambda + \sum_{i\in[R]} \sum_{t=0}^{T} \left(-\bar{u}_{i,t}\gamma_{i,t} + \underline{\mu}_{i,t}\theta_{i,t} - \eta_{i,t}\pi_{i,t} \right) \right\} \tag{12a}$$

$$s.t.\ y_{i,t} \geq y_{i,t+1}, \forall i \in [R], t \leq T-1 \tag{12b}$$

$$\sum_{i\in[R]} \sum_{s=1}^{S_i} P_{i,s} \left(p_i^c q_{i,s}^k - \beta_i \delta_{i,s}^k \right) + \sum_{i\in[R]} \sum_{t=0}^{T} \left((\gamma_{i,t} - \theta_{i,t}) d_{i,t}^k + (|d_{i,t}^k - \mu_{i,t}|)\pi_{i,t} \right) \geq -\lambda, k = 0, 1, \dots, K-1 \tag{12c}$$

$$h_{i,t}^k \geq h_{i,t-1}^k + d_{i,t}^k y_{i,t} - V_{i,t},\ \forall i \in [R], 1 \leq t \leq T, k = 0, 1, \dots, K-1 \tag{12d}$$

$$h_{i,0}^k = d_{i,0}^k, \forall i \in [R], k = 0, 1, \dots, K-1 \tag{12e}$$

$$q_{i,s}^k \leq \sum_{t=0}^{T} d_{i,t}^k, \forall i \in [R], s = 1, \dots, S_i, k = 0, 1, \dots, K-1 \tag{12f}$$

$$q_{i,s}^k \leq \sum_{t=0}^{T} d_{i,t}^k y_{i,t} - h_{i,T_i - v_{i,s}}^k + e_i, \forall i \in [R], s = 1, \dots, S_i, k = 0, 1, \dots, K-1 \tag{12g}$$

$$\delta_{i,s}^k \geq \sum_{t=0}^{T} d_{i,t}^k (1 - y_{i,t}) + h_{i,T_i - v_{i,s}}^k - e_i, \forall i \in [R], s = 1, \dots, S_i, k = 0, 1, \dots, K-1 \tag{12h}$$

$$\boldsymbol{y} \in \{0,1\}, \boldsymbol{e}, \boldsymbol{\gamma}, \boldsymbol{\theta}, \boldsymbol{\pi}, \boldsymbol{h}, \boldsymbol{q}, \boldsymbol{\delta} \geq 0 \tag{12i}$$

The master problem $\mathbf{MASTER}_P$ is a relaxation of Problem (10). Constraints (12c)–(12h) is the relaxation of Constraint (10c) corresponding to $d_{i,t}^k, k = 0, \dots, K-1$.

The details of the cutting-plane algorithm for solving the subproblem of Problem (10) are presented in Algorithm 2. In each iteration of this algorithm, we solve master problem $\mathbf{MASTER}_P$ and obtain a candidate solution whose objective value is an upper bound of Problem (10). Then we check whether the obtained solution satisfies Constraint (10c). If Constraint (10c) is satisfied, then the obtained solution is the optimal solution of Problem (10). Otherwise, new cuts will be introduced into the master problem to update the solution. The procedure is repeated until an optimal solution is found.

Similar to Algorithm 1, the number of constraints and decision variables in master problem $\mathbf{MASTER}_P$ grows linearly as iteration grows while the number of binary decision variables remains unchanged. In particular, the number of constraints and decision variable of $\mathbf{MASTER}_P$ at Kth iteration are $(T-1)R + K(1 + R + TR + 3S_i R)$ and $4TR + KTR + KS_i R + 1$ respectively, with TR binary decision variables. Moreover, the size of $G'(\boldsymbol{y}, \boldsymbol{e}, \boldsymbol{\gamma}, \boldsymbol{\theta}, \boldsymbol{\pi})$ also remains unchanged as number of iteration grows.

6. Computational study

In this section, we demonstrate the proposed optimization models based on the real-world HSR meal delivery system in China. The idea of on-demand railway meal delivery service system was borrowed from the O2O meal delivery that has been popularized worldwide since around 2013. The introduction of railway meal delivery service not only improves literally passengers' travel experience and can be seen as part of smart transportation. In China, the railway meal delivery system was established in 2017. It is operated on both a mobile application and on the official website of Chinese HSR. At its initial launch, a total of 27 HSR stations including for example, Beijing, Shanghai, Hangzhou, Hefei, Changchun were covered as a first trial. Most of the stations are located in capital cities or transportation junctions where long standing time and large volume of passengers usually happen. One year later, another 11 stations were included in the meal delivery system. Fig. 4 shows the 38 HSR stations and the corresponding rail lines where on-demand meal delivery services can be accessed. Currently, to book a on-demand meal order on HSR, passengers are required to do so at least one hour before train departure time of the station where meals will be served.

In this computational study, we consider one HSR line that provides on-demand meal delivery service at a certain station with 5 restaurants available. We divide time into blocks of 5 minutes. The on-demand HSR meal demand is estimated based on passenger capacity of high-speed trains (e.g., 16-car version of Fuxing train can carry about 1200 people). For each restaurant, the nominal value, lower bound and upper bound of demand per unit time block is randomly generated from uniform distribution $U(4,5)$, $U(5,6)$ and $U(6,7)$ respectively for peak hours while $U(2,3)$, $U(3,4)$ and $U(4,5)$ for non-peak hours respectively. Demand is set to be within $(3,9)$ per unit time during peak hours and $(0,7)$ during non-peak hours. The average absolute deviation is reasonably given as 6. In the generated sample, the expected value of total demand of all restaurants is in the range of $(537,664)$. The delivery time of the five restaurants follow different discrete distributions with known probabilities. The quantities of meals that the five restaurants can prepare per unit time are set to be $15, 10, 8, 5, 3$ respectively.

To facilitate result analysis, we assume that the price for meals provided by all restaurants are 50 and the commission rate is 20% (close to the common practices in on-demand meal services). Hence, the commission fee per meal is set to be 10. Different values of the delay penalty (from zero to 20) as well as pre-ordering cost (from 1 to 10) are tested in this study to explore their effects. One thing to note is that scaling the values of meal price, commission fee and delay cost simultaneously will not change the solution and performance of the model.

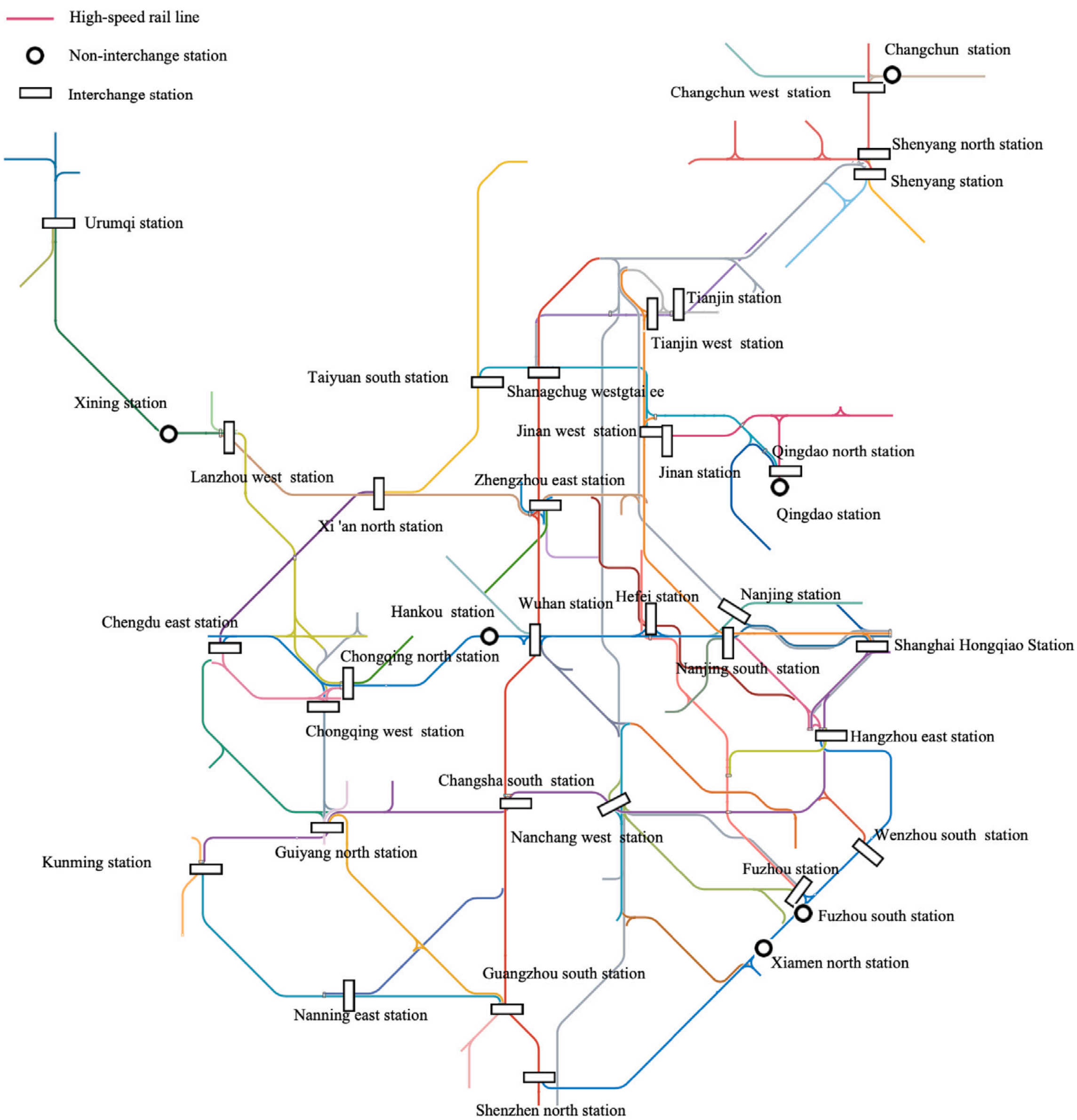


Fig. 4. The 38 high-speed railway stations that provide on-demand meal delivery services in China.
Source: Data from Chinese railway official website https://www.12306.cn/index/.

6.1. Results for system's performance

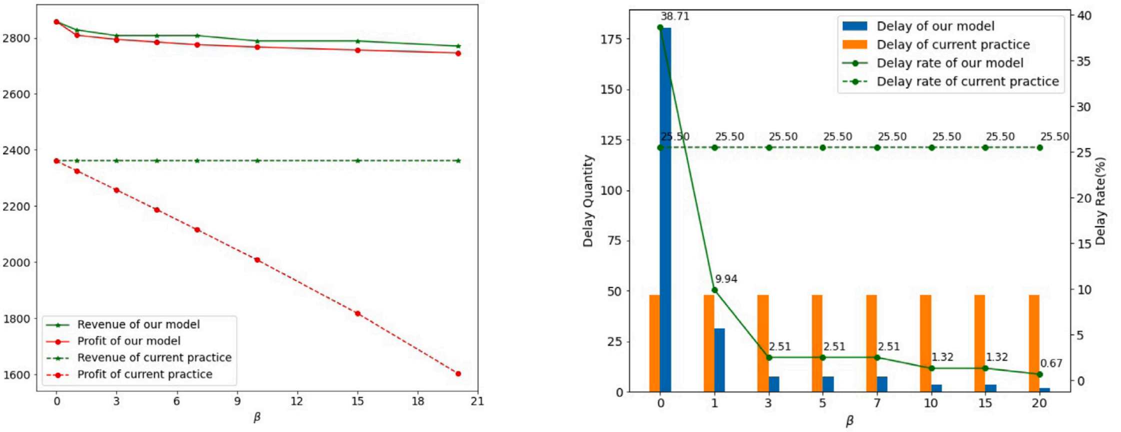
We first run experiments for the order deadline optimization model (2) and solve the model via Algorithm 1. In our setting, we set commission fee p_i^c as 10. In model (2), delay penalty β_i measures the weights of cost of delayed orders. This parameter influences the optimal deadline setting and also the performance of the system. Hence, we solve the model given different values of β_i and show the results in Table 1.

Table 1 shows that the optimal order deadline depends on value of β_i. When β_i is small, the decision maker weighs less on cost of delay and the optimal order deadline is set to be shorter to avoid demand lost. Specifically, when no delay penalty cost is considered (i.e., $\beta_i = 0$), the optimal order deadline corresponds to the shortest delivery time for all restaurants (i.e., 5,10,15,10,20 minutes before train departure time for the five restaurants respectively). As the value of β_i increases, the optimal order deadline becomes earlier to reduce delay rate of orders. Consequently, the worst-case expected quantity of delayed orders as well as delay rate decrease at the cost of smaller worst-case expected profit, revenue (i.e., commission), and quantity of fulfilled orders. One thing to note is that as long as delay cost is considered, even very small, the order deadline becomes substantially earlier for restaurants with relatively small meal preparation capacity. For example, when $\beta_i = 1$, the optimal deadlines of the last three restaurants whose

Table 1
Results of order deadline optimization model.

p^c	β	Obj	Revenue	Delay	Delay rate (%)	Completed orders	Deadline (min)
10	0	2857.8	2857.8	180.5	38.71	285.8	(5,10,15,10,20)
10	1	2808.5	2827.6	31.2	9.94	282.8	(5,10,60,65,80)
10	3	2793.6	2807.3	7.2	2.51	280.7	(5,25,60,70,85)
10	5	2784.5	2807.3	7.2	2.51	280.7	(5,25,60,70,85)
10	7	2775.3	2807.3	7.2	2.51	280.7	(5,25,60,70,85)
10	10	2766.9	2788.3	3.7	1.32	278.8	(5,25,65,70,85)
10	15	2756.1	2788.3	3.7	1.32	278.8	(5,25,65,70,85)
10	20	2745.8	2770.3	1.9	0.67	277.0	(5,30,65,70,85)

In this table, 'Obj' refers to the optimal objective value of the model, which represents the worst-case expected profit of the platform; 'Revenue' refers to the worst-case expected total commission fee the platform received; 'Delay' refers to the worst-case expected number of delayed orders; 'Delay rate' is the estimated worst-case expected delay rate among all accepted orders; 'Completed orders' refers to the worst-case expected quantity of ordered meals that are successfully delivered to customers; 'Deadline' presents the optimal order deadlines of the 5 restaurants, i.e., how many minutes before train departure time to stop accepting orders.



(a) The worst-case expected profits and revenue (b) The worst-case expected delayed demand and delay rate

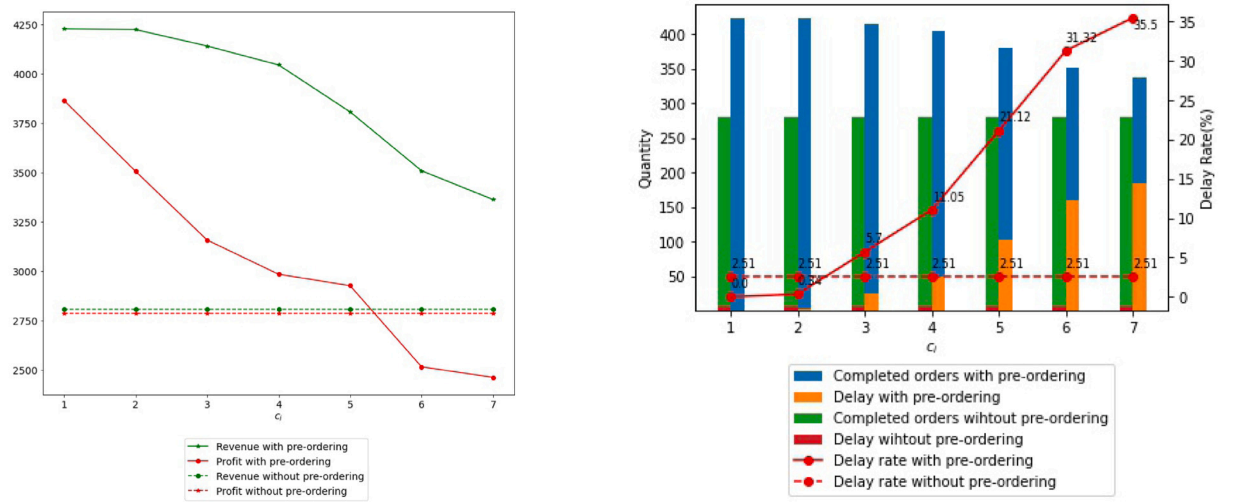
Fig. 5. Performance of order deadline optimization model.

meal preparation capacity is relatively small (8,5,3 per unit time respectively as we set before) becomes 60, 65 and 80 minutes before train departure time, about 1 h earlier than the case where no delayed cost is considered. Moreover, for restaurant with relatively large meal preparation capacity, its order deadline changes slightly as delay cost increases. For example, for the first restaurant whose meal preparation capacity is 15 meals per unit time, which is larger than the largest upper bound of demand (9 in our setting), the optimal deadline always remains to be 5 minutes before train departure time, the same as the shortest possible delivery time.

Table 1 also shows that different restaurants with different meal preparation capacity, delivery time and demands will have different optimal order deadline. The optimal deadline for some restaurants are far always from the current setting used in practice, i.e., 1 h before train departure. Comparing the performances of settings obtained by our model and the current practice (see Figs. 5(a) and 5(b)), our approach can dramatically improve the profit of the platform as well as delay rate when delay cost is considered. To be specific, under the current practice, the estimated delay rate is 25.5% and the worst-case expected value of total commission fee the platform gained is about 2360.7. In our approach, the delay rate could be reduced with the profit and total commission fee improved at the same time. For example, when delay cost β_i is set to be 3, the worst-case expected profit, revenue and delay rate are 2793.6, 2807.3 and 2.51%, improved by 23.8%, 18.9% and 984% respectively comparing to the current practices.

Next, we proceed to solve and analyze the optimization model for our proposed pre-ordering policy. Commission fee p_i^c is still set to be 10. In the new model, two other parameters are involved: delay penalty β_i and pre-ordering fee c_i. We solve the model with different values of β_i and c_i selected. The results are shown in Table 2 and Figs. 6(a) and 6(b).

The result shows that comparing to ordering policy without pre-ordering, the profit and revenue of the platform improves when pre-ordering cost c_i is not too large relatively to commission fee. For example, when unit commission charge $p_i^c = 10$ and delay cost $\beta_i = 5$, the worst-case expected net profit of the system is greater than that for policy without pre-ordering for all $c_i \leq 4$. In addition, when c_i is set to be small (e.g., smaller than 2), the delay rate is also improved. Moreover, the total quantity of completed orders under the pre-ordering policy also improved largely, leading to higher revenue for the restaurants and platform at the same time. This indicates that the pre-ordering policy could improve revenue, delay rate and service availability (e.g., more demand satisfied and freedom of order placement time) for both the platform and restaurants at the same time.



(a) The worst-case expected profits and revenue with and without pre-ordering policy

(b) The worst-case expected delayed demand and delay rate with and without pre-ordering policy

Fig. 6. Performance of pre-ordering policy.

Next, we discuss on the effects of values of β_i and c_i. We observe that given a fixed value of delay cost β_i, when the value of pre-order cost c_i increases, the quantity of pre-ordered meals and the worst-case expected profit of the profit (i.e., the objective value of the model) decreases while the worst-case expected delayed quantity and delay rate increases (see Figs. 6(a) and 6(b)). This indicates that as the pre-ordering cost increases, the platform becomes conservative in pre-ordering to reduce possible loss caused by oversupply. Moreover, we observe that the optimal deadlines for real-time orders are consistently corresponding to the possible shortest delivery time across all settings we consider here. In other words, under pre-ordering setting, the system tends to accept real-time orders as many as possible to maximize its own profit. We further study the influences of β_i with c_i fixed. We observe that as the delay cost β_i grows up, generally the optimal quantity of pre-ordered meals e_i increases, which reduces the worst-case expected delayed orders and delay rate.

Table 3 shows the computational time and the number of iterations needed to find the optimal solutions for 50 randomly generated cases with different number of restaurants R for Algorithms 1 and 2 respectively. All the algorithms were coded in Python and executed by commercial solver Gurobi version 9.5. A Windows 10 computer with 11th Gen Intel(R) Core(TM) i7-11700K @ 3.60 GHz processor and 64 GB RAM was used to run the experiments.

It can be observed that both Algorithms 1 and 2 can be implemented relatively fast. In particular, when the number of restaurants is 10, which is larger than many practical cases, the computational time running on a personal computer is about 2 h for Algorithm 1 and 1 h for Algorithm 2. The computational time is acceptable in practice because decisions are made offline in daily operation. We note that the proposed model can not only address problem settings with joint demands (i.e., demands are correlated among restaurants) which is of interest in this paper, the model can also be generalized to the scenario where the demands are independent. By implementing the model with one restaurant considered each time, we can solve a single-restaurant problem efficiently. On the other hand, if we purely consider a single-restaurant problem where the demands are assumed to be independent, it cannot be extended to incorporate demand correlations due to the violation of the assumption.

6.2. Comparison with deterministic and stochastic models

In this section, we compare the proposed pre-ordering distributionally robust optimization (DRO) model with two commonly used approaches in the literature: stochastic programming model and deterministic optimization model. Stochastic programming model assumes that demands are random with the full knowledge of probability distributions known. Deterministic optimization model, on the other hand, treats demands as deterministic.

In all approaches, the commission fee per order p_i^c is set as 10. The unit cost of a pre-ordered meal c_i is 5. We use the delay penalty per order $\beta = 10, 20, 30$ to generate three problem instances. The preparation capacity is set as 2 per unit time. To emphasize comparison of different models, in this part, we consider one restaurant in total. We randomly generate demand samples as available data for all of these three approaches. Specifically, we assume the true demand follows normal distribution whose mean is between $[3, 4]$ for non-peak hours and mean is between $[7, 9]$ in peak hours. The standard deviation is set as 3. Based on these assumptions, we randomly generate a demand sample of 50 data points as input data. The sample size is relatively small because we consider situations where we lack complete demand distributional information.

Table 2
Results of pre-ordering policy optimization model.

c	β	Obj	Revenue	Pre-ordering cost	Delay	Delay rate (%)	Completed orders	Real-time order deadline	Pre-ordering quantity
1	5	3862.3	4226.0	363.0	0.0	0.00	422.6	(5,10,15,10,20)	(7,38,85,104,129)
2	5	3502.0	4222.6	718.0	1.4	0.34	422.3	(5,10,15,10,20)	(7,38,85,100,129)
3	5	3156.7	4139.5	936.0	25.0	5.70	414.0	(5,10,15,10,20)	(7,29,73,82,121)
4	5	2872.0	4043.6	1060.0	50.2	11.05	404.4	(5,10,15,10,20)	(7,14,61,74,109)
5	5	2649.4	3805.9	880.0	101.9	21.12	380.6	(5,10,15,10,20)	(7,14,31,51,73)
6	5	2514.5	3507.8	504.0	160.0	31.32	350.8	(5,10,15,10,20)	(7,5,12,25,35)
7	5	2461.6	3361.8	315.0	185.0	35.50	336.2	(5,10,15,10,20)	(3,5,7,5,25)
1	7	3861.9	4226.0	363.0	0.0	0.00	422.6	(5,10,15,10,20)	(7,38,85,104,129)
3	7	3145.4	5591.4	2429.0	6.8	1.20	559.1	(5,10,15,10,20)	(7,31,82,99,128)
6	7	2348.3	3829.9	1104.0	97.1	20.22	383.0	(5,10,15,10,20)	(7,14,37,54,72)
1	10	3861.8	4226.0	363.0	0.0	0.00	422.6	(5,10,15,10,20)	(7,38,85,104,129)
3	10	3140.7	4218.3	1068.0	2.6	0.62	421.8	(5,10,15,10,20)	(7,38,82,100,129)
6	10	2214.9	4003.0	1494.0	59.2	12.89	400.3	(5,10,15,10,20)	(7,14,59,75,94)
1	15	3860.8	4226.0	363.0	0.0	0.00	422.6	(5,10,15,10,20)	(7,38,85,104,129)
3	15	3138.0	4222.6	1077.0	1.4	0.34	422.3	(5,10,15,10,20)	(7,38,85,100,129)
6	15	2115.4	4123.7	1824.0	29.0	6.57	412.4	(5,10,15,10,20)	(7,20,73,85,119)
10	15	1097.0	2156.9	186.0	95.7	30.74	215.7	(5,10,15,10,20)	(7,9,33,58,79)
1	20	3860.5	4226.9	365.0	0.0	0.00	422.7	(5,10,15,10,20)	(7,38,87,104,129)
3	20	3135.6	3504.0	360.0	1.1	0.32	350.4	(5,10,15,10,20)	(7,38,85,101,129)
6	20	2075.3	2510.0	334.0	13.5	5.09	251.0	(5,10,15,10,20)	(7,31,79,92,125)
10	20	861.3	4040.5	2650.0	50.2	11.06	404.0	(5,10,15,10,20)	(7,14,63,80,101)

In this table, 'Obj' refers to the optimal objective value of the pre-ordering optimization model; 'Revenue' refers to the worst-case expected total commission fee the platform received; 'Pre-ordering cost' is the total cost the platform pays for the pre-ordering orders; 'Delay' and 'Delay rate' refer to the worst-case expected number of delayed orders and estimated worst-case delay rate; 'Real-time order deadline' refers to the deadline (how many minutes before train departure time) for the real-time orders; 'Completed orders' refer to the quantity of orders placed by the customers and are successfully delivered to them; 'Pre-ordering quantity' refers to the quantity of meals the platform pre-orders from the five restaurants respectively.

Table 3
Computational efficiency.

R	Algorithm 1						Algorithm 2					
	Iteration			Time (min)			Iteration			Time (min)		
	Min	Mean	Max	Min	Mean	Max	Min	Mean	Max	Min	Mean	Max
1	26	28.80	32	0.01	0.01	0.02	27	29.14	32	0.02	0.02	0.04
3	187	238.26	308	0.64	1.33	2.48	159	205.36	291	0.79	1.29	2.31
5	513	617.84	743	31.41	77.82	128.05	405	505.34	641	15.62	32.21	62.28
10	529	771.04	985	34.98	126.84	343.35	494	563.78	656	28.17	53.22	81.27

Table 4
Parameters used in each approach variant.

	DRO	SAA	Deterministic
Parameter	$\underline{d}_{i,t}, \bar{d}_{i,t}, N_{i,t}$ $\underline{\mu}_{i,t}, \bar{\mu}_{i,t}, \mu_{i,t}, \eta_{i,t}$	Demands generated from normal distribution with means $\bar{x}_{i,t}$ and std $s_{i,t}$	$d_{i,t} = \bar{x}_{i,t}$

All of the three optimization approaches use the sample of 50 data points to estimate parameters they required (see Table 4). For deterministic model, demands are set to equal the mean of the sample. For our DRO model the lower and upper bounds of corresponding parameters (mean and average absolute deviation) are computed from 95% confidence interval and the nominal demand is set as the average of the sample. For stochastic programming, we estimate the mean of normal distribution based on the average value of the sample data. We employ the Sample Average Approximation (SAA) to formulate approximately the Stochastic Programming model. In particular, let Ω denote the number of scenarios, indexed by ω. Let $\boldsymbol{d}_\omega$ denote the vector of demand realization under scenario ω. Let $\bar{x}_{i,t}$ denote the sample means and $s_{i,t}$ the standard deviation (std) from the sample. We can formulate the SAA model as follows:

$$\max \left\{ -c_i e_i + \frac{1}{\Omega} \sum_{\omega=1}^{\Omega} g_i(\boldsymbol{e}, \boldsymbol{x}, \boldsymbol{d}_{\omega}) \right\}, \tag{13}$$

which is now a MILP and can be solved by off-the-shelf solver.

For all of these three approaches, we obtain solution based on the sample of 50 data points and analyze the in-sample performance of these solutions based on this sample. We further evaluate the out-of-sample performances of these solutions based on another sample of 500 data points which are randomly generated based on the same setting of data used for in-sample experiments.

Table 5
In-sample and out-of-sample performances for DRO, SAA and Deter approaches.

In-sample performance										
Model		$\beta = 10$			$\beta = 20$			$\beta = 30$		
		Ave	Min	Max	Ave	Min	Max	Ave	Min	Max
DRO	obj	1299.864	1280.600	1320.600	1300.380	1280.600	1320.600	1300.380	1280.600	1320.600
	com	1300.122	1280.600	1320.600	1300.380	1280.600	1320.600	1300.380	1280.600	1320.600
	delay	0.026	0.000	0.134	0.000	0.000	0.000	0.000	0.000	0.000
	rate	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
SAA	obj	1259.552	1243.960	1270.600	1291.848	1276.040	1309.960	1300.380	1280.600	1320.600
	com	1279.966	1267.460	1295.200	1297.536	1279.080	1316.660	1300.380	1280.600	1320.600
	delay	2.041	1.314	2.792	0.284	0.040	0.624	0.000	0.000	0.000
	rate	0.009	0.006	0.012	0.001	0.000	0.002	0.000	0.000	0.000
Deter	obj	1191.596	1175.160	1205.160	1137.204	1104.040	1162.740	1082.812	1032.920	1120.320
	com	1245.988	1236.780	1256.220	1245.988	1236.780	1256.220	1245.988	1236.780	1256.220
	delay	5.439	4.158	7.112	5.439	4.158	7.112	5.439	4.158	7.112
	rate	0.025	0.019	0.032	0.025	0.019	0.032	0.025	0.019	0.032
Out-of-sample performance										
		$\beta = 10$			$\beta = 20$			$\beta = 30$		
		Ave	Min	Max	Ave	Min	Max	Ave	Min	Max
DRO	obj	1307.164	1283.900	1341.600	1307.680	1283.900	1341.700	1307.680	1283.900	1341.700
	com	1307.422	1283.900	1341.650	1307.680	1283.900	1341.700	1307.680	1283.900	1341.700
	delay	0.026	0.000	0.078	0.000	0.000	0.000	0.000	0.000	0.000
	rate	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
SAA	obj	1260.842	1241.720	1283.920	1297.255	1276.370	1325.740	1307.680	1283.900	1341.700
	com	1284.261	1265.960	1310.590	1304.205	1283.120	1336.380	1307.680	1283.900	1341.700
	delay	2.342	1.514	3.111	0.348	0.078	0.572	0.000	0.000	0.000
	rate	0.010	0.006	0.013	0.001	0.000	0.002	0.000	0.000	0.000
Deter	obj	1190.148	1165.620	1212.100	1131.382	1089.280	1161.050	1072.616	1012.940	1110.000
	com	1248.914	1236.870	1267.290	1248.914	1236.870	1267.290	1248.914	1236.870	1267.290
	delay	5.877	4.572	7.634	5.877	4.572	7.634	5.877	4.572	7.634
	rate	0.027	0.021	0.035	0.027	0.021	0.035	0.027	0.021	0.035

In this table, 'DRO' is short for our proposed distributionally robust optimization model. 'SAA' refers to the stochastic programming model based on Normal distribution. 'Deter' is short for deterministic model. For the performance criteria, 'obj', 'com', 'delay' and 'rate' respectively represent the expected objective value, the expected total commission the platform received, the expected number of delayed orders and the expected rate of delay over total orders received given the optimal solutions of each model.

The evaluations are repeated 100 times. We present the results of in-sample and out-of-sample performances (the average (Ave), minimal (Min) and maximal (Max) values) in Table 5 for each of the three problem instances given $\beta = 10, 20, 30$ respectively. The performance DRO model ranks the best, followed by the SAA approach for most in-sample and out-of-sample runs, with some exceptions where the DRO and SAA perform the same for all criteria (e.g., In-sample run given $\beta = 30$). For the problem instance $\beta = 30$, the expected worst-case delays for the DRO and SAA models are all zero. This can be probably explained by the capacity of both models to avoid high penalties for delays which cannot be achieved by the Deter model. We also note that the DRO model not only outperforms other approaches, but also exhibits the highest level of robustness against parameter variations for both the in-sample and out-of-sample experiments.

7. Discussion and conclusion

On-demand meal ordering service on HSR is an emerging O2O business on railway. This service has become increasingly popular and its potential market is huge. The most distinct feature of this O2O business is that it has strict time constraint on delivery. Hence, it is important to design ordering procedure elaborately to simultaneously guarantee revenue and service quality. However, few studies have been done on this problem.

In this paper, we specifically study order management for HSR on-demand meal ordering service. We first propose a distributionally robust model to optimize the existing ordering procedure, which sets an order deadline to guarantee timely delivery. Our approach considers uncertainty in demands and delivery time and identifies the best order deadline that maximizes the total worst-case expected profit across all possible distributions of demands, with profit defined to be total revenue from commission fee minus cost of delayed orders.

The existing procedure always requires an order deadline for meal processing and delivery, which leads to lost demands and also service unavailability. Considering this limitation, we propose a pre-ordering policy to improve the current practice. The new policy cancels the order deadline to allow passengers to place orders anytime. To guarantee timely delivery, the system pre-orders meals to satisfy late orders. We also develop distributionally robust optimization models to optimize the new policy. Computational experiments show that the new strategies could simultaneously increase service availability, improve the profit, as well as reduce the number of delayed orders.

Our proposed distributionally robust optimization models for both the existing and new strategies are computationally complex. To tackle computational challenges, we reformulate our models into treatable one-stage optimization problem with one complex constraint based on duality theory. We further propose cutting-plane algorithms to solve the reformulated models by means of MILP. Moreover, computational experiment results show that our proposed distributionally robust models can outperform stochastic programming and deterministic optimization approaches in terms of both in-sample and out-of-sample performance.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

We sincerely thank the editors and three reviewers for their insightful comments that have considerably improved the paper. This work is supported by Natural Science Foundation of China (72101070), Zhejiang Provincial Natural Science Foundation of China (LY23G010001), Hangzhou Normal University (2021QDL004) and Start-up funding by Georgia Tech Shenzhen Institute, Tianjin University.

Appendix

Proof for Proposition 1. Let $\tilde{\boldsymbol{d}} = (\tilde{\boldsymbol{d}}_1, \tilde{\boldsymbol{d}}_2, \ldots, \tilde{\boldsymbol{d}}_T)$ and $\mathbb{P}$ denote the joint probability distribution of $\tilde{\boldsymbol{d}}_i$. Suppose $\tilde{\boldsymbol{d}}$ has $|W|$ scenarios denoted by $\boldsymbol{d}^1, \ldots, \boldsymbol{d}^{|W|}$ with probability $P^1, \ldots, P^{|W|}$. Problem (3) can be rewritten as follows:

$$
\begin{aligned}
\min_{\mathbb{P}\in\mathcal{F}(\underline{\mu},\bar{\mu},\bar{\eta})} & \sum_{k=1}^{|W|}\left[\sum_{i\in[R]}\left(p_i^c\sum_{t=0}^{T}\tilde{d}_{i,t}x_{i,t}-(p_i^c+\beta_i)\sum_{s=1}^{S_i}P_{i,s}h_{i,T-v_{i,s}}\right)\right]\\
s.t. & \sum_{k=1}^{|W|}P^k=1\\
& \sum_{k=1}^{|W|}P^k d_{i,t}^k\le\bar{\mu}_{i,t},\forall i\in[R],t\le T\\
& \sum_{k=1}^{|W|}P^k d_{i,t}^k\ge\underline{\mu}_{i,t},\forall i\in[R],t\le T\\
& \sum_{k=1}^{|W|}P^k|d_{i,t}^k-\mu_{i,t}|\le\bar{\eta}_{i,t},\forall i\in[R],t\le T\\
& P^k\ge 0
\end{aligned}
$$

Note that the above problem is a linear programming with decision variables $P^1, \ldots, P^{|W|}$. Obviously, strong duality holds and we take the dual of this problem. Let $\lambda, \gamma_{i,t}, \theta_{i,t}, \pi_{i,t}$ be the dual variables of the above problem. The dual of the above problem is:

$$
\max_{\lambda,\gamma,\theta,\pi} -\lambda+\sum_{i\in[R]}\left(\sum_{t=0}^{T}\left(-\gamma_{i,t}\bar{\mu}_{i,t}+\theta_{i,t}\underline{\mu}_{i,t}-\pi_{i,t}\bar{\eta}_{i,t}\right)\right) \tag{14a}
$$

$$
\text{s.t.} \sum_{i\in[R]}\left(\sum_{t=0}^{T}\left(p_i^c\tilde{d}_{i,t}x_{i,t}+(\gamma_{i,t}-\theta_{i,t})\tilde{d}_{i,t}+\pi_{i,t}|\tilde{d}_{i,t}-\mu_{i,t}|\right)-(p_i^c+\beta_i)\sum_{s=1}^{S_i}P_{i,s}h_{i,T-v_{i,s}}\right)\ge-\lambda,\quad \forall d_{i,t}^k, k=1,2,\ldots|W| \tag{14b}
$$

$$
\boldsymbol{\gamma},\boldsymbol{\theta},\boldsymbol{\pi}\ge 0 \tag{14c}
$$

Constraint (14b) is equivalent to:

$$
\min_{\tilde{\boldsymbol{d}}\in\boldsymbol{W}}\sum_{i\in[R]}\left(\sum_{t=0}^{T}\left(p_i^c\tilde{d}_{i,t}x_{i,t}+(\gamma_{i,t}-\theta_{i,t})\tilde{d}_{i,t}+\pi_{i,t}|\tilde{d}_{i,t}-\mu_{i,t}|\right)-(p_i^c+\beta_i)\sum_{s=1}^{S_i}P_{i,s}h_{i,T-v_{i,s}}\right)\ge-\lambda
$$

This completes the proof. □

Proof for Proposition 2. The LHS of the robust constraint (4b) minimizes the function $\sum_{i\in[R]}\left(\sum_{t=0}^{T}\left(p_i^c\tilde{d}_{i,t}x_{i,t}+(\gamma_{i,t}-\theta_{i,t})\tilde{d}_{i,t}+\pi_{i,t}|\tilde{d}_{i,t}-\mu_{i,t}|\right)-(p_i^c+\beta_i)\sum_{s=1}^{S_i}P_{i,s}h_{i,T-v_{i,s}}\right)$ over $\tilde{\boldsymbol{d}}$ with support $\boldsymbol{W}$. In the reformulated formulation of Proposition 2, Constraint (5b) ensures that $\tilde{\boldsymbol{d}}$ takes values within its support. Constraint (5c) and (5d) transforms the absolute value $|\tilde{d}_{i,t}-\mu_{i,t}|$ into linear constraints with linear decision variable $z_{i,t}$. As $h_{i,t} = \max\left\{ h_{i,t-1} + \tilde{d}_{i,t} y_{i,t} - V_{i,t}, 0 \right\}, \forall 1 \le t \le T, i \in [R]$, Constraint (5e) to (5g) transform this relationship into linear constraints with binary decision variable $I_{i,t}$. Constraint (5e) to (5g) ensures that if $h_{i,t-1} + \tilde{d}_{i,t} y_{i,t} - V_{i,t} \ge 0$, then $I_{i,t} = 1$ and $h_{i,t} = h_{i,t-1} + \tilde{d}_{i,t} y_{i,t} - V_{i,t}$. Otherwise, $I_{i,t} = 0$ and $h_{i,t} = 0$. This completes the proof.

Proof for Theorem 1. The proof of Theorem 1 follows from Propositions 1 and 2. □

To prove Theorem 2, we present the following two propositions. First, we reformulate the expectation minimization problem $\min_{\mathbb{P}\in\mathcal{F}} \mathbb{E}_{\mathbb{P}}[g_i(\boldsymbol{e}, \boldsymbol{y}, \tilde{\boldsymbol{d}})]$ in Problem (9) into a tractable form based on duality theory in the following Proposition.

Proposition 3. *Problem (9) can be reformulated as:*

$$\max_{\boldsymbol{e},\boldsymbol{y},\lambda,\boldsymbol{\gamma},\boldsymbol{\theta},\boldsymbol{\pi}} \left\{ \sum_{i\in[R]} -c_i e_i - \lambda + \sum_{i\in[R]} \sum_{t=0}^{T} \left(-\bar{u}_{i,t}\gamma_{i,t} + \underline{\mu}_{i,t}\theta_{i,t} - \eta_{i,t}\pi_{i,t} \right) \right\} \tag{15a}$$

$$s.t\ y_{i,t} \ge y_{i,t+1}, \forall i \in [L], t \le T-1 \tag{15b}$$

$$\min_{\tilde{\boldsymbol{d}}\in\boldsymbol{W}} \left\{ \sum_{i\in[R]} \sum_{s=1}^{S_i} P_{i,s} g_i(\boldsymbol{e}, \boldsymbol{y}, \tilde{\boldsymbol{d}}, \boldsymbol{h}, v_{i,s}) + \sum_{i\in[R]} \sum_{t=0}^{T} \left((\gamma_{i,t} - \theta_{i,t})\tilde{d}_{i,t} + (|\tilde{d}_{i,t} - \mu_{i,t}|)\pi_{i,t} \right) \right\} \ge -\lambda \tag{15c}$$

$$h_{i,t} = \max\left\{ h_{i,t-1} + \tilde{d}_{i,t} y_{i,t} - V_{i,t}, 0 \right\}, \forall 1 \le t \le T, i \in [R] \tag{15d}$$

$$h_{i,0} = \tilde{d}_{i,0}, \forall i \in [R] \tag{15e}$$

$$\boldsymbol{y} \in \{0, 1\} \tag{15f}$$

$$\boldsymbol{e}, \boldsymbol{\gamma}, \boldsymbol{\theta}, \boldsymbol{\pi} \ge 0 \tag{15g}$$

Proof. The proof of Proposition 3 is similar to that of Proposition 1. □

Proposition 3 transforms Problem (9) into a maximization optimization problem with complex Constraints (15c). The LHS of Constraints (15c) is a minimization problem whose inner problem $g_i(\boldsymbol{e}, \boldsymbol{y}, \tilde{\boldsymbol{d}})$ is a maximizing problem subject to a series of constraints. We further reformulate this problem as follows.

Proposition 4. *The LHS of Constraint (15c) can be reformulated as follows:*

$$G'(\boldsymbol{y}, \boldsymbol{e}, \boldsymbol{\gamma}, \boldsymbol{\theta}, \boldsymbol{\pi}) = \min_{\tilde{d},q,\delta,h,I,z} \left\{ \sum_{i\in[R]} \sum_{s=1}^{S_i} P_{i,s} \left(p_i^c q_{i,s} - \beta_i \delta_{i,s} \right) + \sum_{i\in[R]} \sum_{t=0}^{T} \left((\gamma_{i,t} - \theta_{i,t})\tilde{d}_{i,t} + \pi_{i,t} z_{i,t} \right) \right\} \tag{16a}$$

$$s.t.\ \underline{d}_{i,t} \le \tilde{d}_{i,t} \le \bar{d}_{i,t}, \forall i \in [R], t \le T \tag{16b}$$

$$z_{i,t} \ge \tilde{d}_{i,t} - \mu_{i,t}, \forall i \in [R], t \le T \tag{16c}$$

$$z_{i,t} \ge -\tilde{d}_{i,t} + \mu_{i,t}, \forall i \in [R], t \le T \tag{16d}$$

$$h_{i,t} \ge h_{i,t-1} + \tilde{d}_{i,t} x_{i,t} - V_{i,t}, \forall i \in [R], 1 \le t \le T \tag{16e}$$

$$h_{i,t} \le \sum_{t'=0}^{t} \bar{d}_{i,t'} I_{i,t}, \forall i \in [R], t \le T \tag{16f}$$

$$h_{i,t} \le h_{i,t-1} + \tilde{d}_{i,t} x_{i,t} - V_{i,t} + \sum_{t'=0}^{t} \bar{d}_{i,t'}(1 - I_{i,t}), \forall i \in [R], 1 \le t \le T \tag{16g}$$

$$h_{i,0} = \tilde{d}_{i,0}, \forall i \in [R] \tag{16h}$$

$$q_{i,s} \le \sum_{t\le T} \tilde{d}_{i,t}, \forall i \in [R], s = 1, \dots, S_i \tag{16i}$$

$$q_{i,s} \le \sum_{t\le T} \tilde{d}_{i,t} y_{i,t} - h_{i,T-v_{i,s}} + e_i, \forall i \in [R], s = 1, \dots, S_i \tag{16j}$$

$$q_{i,s} \ge \sum_{t\le T} \tilde{d}_{i,t} - \sum_{t=0}^{T} \bar{d}_{i,t} I_{i,q,s}, \forall i \in [R], s = 1, \dots, S_i \tag{16k}$$

$$q_{i,s} \ge \sum_{t\le T} \tilde{d}_{i,t} y_{i,t} - h_{i,T-v_{i,s}} + e_i - \sum_{t=0}^{T} \bar{d}_{i,t}(1 - I_{i,q,s}), \forall i \in [R], s = 1, \dots, S_i \tag{16l}$$

$$\delta_{i,s} \ge h_{i,T-v_{i,s}} + \sum_{t\le T} \tilde{d}_{i,t}(1 - y_{i,t}) - e_i, \forall i \in [R], s = 1, \dots, S_i \tag{16m}$$

$$\delta_{i,s} \le \sum_{t=0}^{T} \bar{d}_{i,t} I_{i,\delta,s}, \forall i \in [R], s = 1, \dots, S_i \tag{16n}$$

$$\delta_{i,s} \leq h_{i,T-v_{i,s}} + \sum_{t \leq T} \tilde{d}_{i,t}(1 - y_{i,t}) - e_i + \sum_{t=0}^{T} \bar{d}_{i,t}(1 - I_{i,\delta,s}), \forall i \in [R], s = 1, \ldots, S_i \tag{16o}$$

$$\tilde{\boldsymbol{d}}, \boldsymbol{h}, \boldsymbol{q}, \boldsymbol{\delta}, \boldsymbol{z} \geq 0, \boldsymbol{I} \in \{0, 1\}. \tag{16p}$$

Proof. The proof for Proposition 4 is similar to that for Proposition 2. In particular, Constraint (16b) ensures that $\tilde{\boldsymbol{d}}$ takes value in its support $\boldsymbol{W}$. Constraint (16c) and (16d) transform absolute value $|\tilde{d}_{i,t} - \mu_{i,t}|$ into linear constraints with decision variable $z_{i,t}$. Constraint (16e) to (16g) express $h_{i,t}$ by linear constraints with binary decision variable $I_{i,t}$. Similar transformation is done for $q_{i,s}$ and $\delta_{i,s}$ with Constraint (16i) to (16l) and Constraint (16m) to (16o) respectively. □

Proof for Theorem 2. The proof of Theorem 2 follows from Proposition 3 and 4. □

References

Bahrami, S., Nourinejad, M., Yin, Y., Wang, H., 2021. The three-sided market of on-demand delivery. Available at SSRN 3944559.
Baldacci, R., Bartolini, E., Mingozzi, A., 2011. An exact algorithm for the pickup and delivery problem with time windows. Oper. Res. 59 (2), 414–426.
Barrena, E., Canca, D., Coelho, L.C., Laporte, G., 2014. Exact formulations and algorithm for the train timetabling problem with dynamic demand. Comput. Oper. Res. 44, 66–74.
Ben-Tal, A., Den Hertog, D., De Waegenaere, A., Melenberg, B., Rennen, G., 2013. Robust solutions of optimization problems affected by uncertain probabilities. Manage. Sci. 59 (2), 341–357.
Bräysy, O., Gendreau, M., 2005a. Vehicle routing problem with time windows, Part I: Route construction and local search algorithms. Transp. Sci. 39 (1), 104–118.
Bräysy, O., Gendreau, M., 2005b. Vehicle routing problem with time windows, Part II: Metaheuristics. Transp. Sci. 39 (1), 119–139.
Dai, H., Liu, P., 2020. Workforce planning for O2O delivery systems with crowdsourced drivers. Ann. Oper. Res. 291 (1), 219–245.
Delage, E., Kuhn, D., Natarajan, K., Wiesemann, W., 2018. Distributionally Robust Optimization. Technical Report.
Delage, E., Ye, Y., 2010. Distributionally robust optimization under moment uncertainty with application to data-driven problems. Oper. Res. 58 (3), 595–612.
Desrochers, M., Desrosiers, J., Solomon, M., 1992. A new optimization algorithm for the vehicle routing problem with time windows. Oper. Res. 40 (2), 342–354.
Dumas, Y., Desrosiers, J., Soumis, F., 1991. The pickup and delivery problem with time windows. European J. Oper. Res. 54 (1), 7–22.
Esfahani, P.M., Kuhn, D., 2018. Data-driven distributionally robust optimization using the Wasserstein metric: Performance guarantees and tractable reformulations. Math. Program. 171 (1), 115–166.
Espinosa-Aranda, J.L., García-Ródenas, R., del Carmen Ramírez-Flores, M., López-García, M.L., Angulo, E., 2015. High-speed railway scheduling based on user preferences. European J. Oper. Res. 246 (3), 772–786.
Fu, H., Nie, L., Meng, L., Sperry, B.R., He, Z., 2015. A hierarchical line planning approach for a large-scale high speed rail network: The China case. Transp. Res. A 75, 61–83.
Gao, R., Kleywegt, A.J., 2016. Distributionally Robust Stochastic Optimization with Wasserstein Distance. Working Paper.
Goto, J.H., Lewis, M.E., Puterman, M.L., 2004. Coffee, tea, or…?: a Markov decision process model for airline meal provisioning. Transp. Sci. 38 (1), 107–118.
He, Z., Han, G., Cheng, T., Fan, B., Dong, J., 2019. Evolutionary food quality and location strategies for restaurants in competitive online-to-offline food ordering and delivery markets: An agent-based approach. Int. J. Prod. Econ. 215, 61–72.
Ho, S.C., Leung, J.M., 2010. Solving a manpower scheduling problem for airline catering using metaheuristics. European J. Oper. Res. 202 (3), 903–921.
Jiang, Y., Wu, X., Chen, Y., 2020. Designing a flexible catering system for high-speed railway considering departure time selection and time deadline constraints. IEEE Access 8, 44300–44317.
Khadilkar, H., 2017. Data-enabled stochastic modeling for evaluating schedule robustness of railway networks. Transp. Sci. 51 (4), 1161–1176.
Liu, Y., 2019. An optimization-driven dynamic vehicle routing algorithm for on-demand meal delivery using drones. Comput. Oper. Res. 111, 1–20.
Liu, F., Chen, Z., Wang, S., 2022. Globalized distributionally robust counterpart. Available at SSRN 4221952.
Liu, Y., Li, S., 2023. An economic analysis of on-demand food delivery platforms: Impacts of regulations and integration with ride-sourcing platforms. Transp. Res. E 171, 103019.
Mao, W., Ming, L., Rong, Y., Tang, C.S., Zheng, H., 2019. Faster Deliveries and Smarter Order Assignments for an On-Demand Meal Delivery Platform. Working Paper.
Nakao, H., Shen, S., Chen, Z., 2017. Network design in scarce data environment using moment-based distributionally robust optimization. Comput. Oper. Res. 88, 44–57.
Natarajan, K., Pachamanova, D., Sim, M., 2008. Incorporating asymmetric distributional information in robust value-at-risk optimization. Manage. Sci. 54 (3), 573–585.
Niu, H., Zhou, X., Gao, R., 2015. Train scheduling for minimizing passenger waiting time with time-dependent demand and skip-stop patterns: Nonlinear integer programming models with linear constraints. Transp. Res. B 76, 117–135.
Pan, B., Zhang, Z., Lim, A., 2021. Multi-trip time-dependent vehicle routing problem with time windows. European J. Oper. Res. 291 (1), 218–231.
Pinto, R., Zambetti, M., Lagorio, A., Pirola, F., 2019. A network design model for a meal delivery service using drones. Int. J. Logist. Res. Appl. 1–21.
Qi, J., Cacchiani, V., Yang, L., 2018. Robust train timetabling and stop planning with uncertain passenger demand. Electron. Notes Discrete Math. 69, 213–220.
Rahimian, H., Mehrotra, S., 2019. Distributionally Robust Optimization: A Review. Working Paper.
Reyes, D., Erera, A.L., Savelsbergh, M.W., 2018b. Complexity of routing problems with release dates and deadlines. European J. Oper. Res. 266 (1), 29–34.
Reyes, D., Erera, A., Savelsbergh, M., Sahasrabudhe, S., O'Neil, R., 2018a. The meal delivery routing problem. Optim. Online 6571.
Ropke, S., Pisinger, D., 2006. An adaptive large neighborhood search heuristic for the pickup and delivery problem with time windows. Transp. Sci. 40 (4), 455–472.
Shehadeh, K.S., Wang, H., Zhang, P., 2021. Fleet sizing and allocation for on-demand last-mile transportation systems. Transp. Res. C 132, 103387.
Sun, P., Veelenturf, L.P., Hewitt, M., Van Woensel, T., 2018b. The time-dependent pickup and delivery problem with time windows. Transp. Res. B 116, 1–24.
Sun, H., Wu, J., Ma, H., Yang, X., Gao, Z., 2018a. A bi-objective timetable optimization model for urban rail transit based on the time-dependent passenger volume. IEEE Trans. Intell. Transp. Syst. 20 (2), 604–615.
Ulmer, M.W., Thomas, B.W., Campbell, A.M., Woyak, N., 2021. The restaurant meal delivery problem: Dynamic pickup and delivery with deadlines and random ready times. Transp. Sci. 55 (1), 75–100.
Wu, X., Nie, L., Xu, M., 2017a. Designing an integrated distribution system for catering services for high-speed railways: A three-echelon location routing model with tight time windows and time deadlines. Transp. Res. C 74, 212–244.
Wu, X., Nie, L., Xu, M., 2017b. Robust fuzzy quality function deployment based on the mean-end-chain concept: Service station evaluation problem for rail catering services. European J. Oper. Res. 263 (3), 974–995.

Wu, X., Nie, L., Xu, M., Yan, F., 2018. A perishable food supply chain problem considering demand uncertainty and time deadline constraints: Modeling and application to a high-speed railway catering service. Transp. Res. E 111, 186–209.

Wu, X., Nie, L., Xu, M., Zhao, L., 2019. Distribution planning problem for a high-speed rail catering service considering time-varying demands and pedestrian congestion: A lot-sizing-based model and decomposition algorithm. Transp. Res. E 123, 61–89.

Xiong, P., Jirutitijaroen, P., Singh, C., 2016. A distributionally robust optimization model for unit commitment considering uncertain wind power generation. IEEE Trans. Power Syst. 32 (1), 39–49.

Xu, L., Ng, T.S., Costa, A., 2021. Optimizing disruption tolerance for rail transit networks under uncertainty. Transp. Sci. 55 (5), 1206–1225.

Yildiz, B., Savelsbergh, M., 2019. Provably high-quality solutions for the meal delivery routing problem. Transp. Sci. 53 (5), 1372–1388.

Yue, Y., Wang, S., Zhou, L., Tong, L., Saat, M.R., 2016. Optimizing train stopping patterns and schedules for high-speed passenger rail corridors. Transp. Res. C 63, 126–146.

Zhang, Y., Zhang, Z., Lim, A., Sim, M., 2021. Robust data-driven vehicle routing with time windows. Oper. Res. 69 (2), 469–485.

Zhao, P., Gu, C., Huo, D., Shen, Y., Hernando-Gil, I., 2019. Two-stage distributionally robust optimization for energy hub systems. IEEE Trans. Ind. Inform. 16 (5), 3460–3469.