Contents lists available at GrowingScience

International Journal of Industrial Engineering Computations

homepage: www.GrowingScience.com/ijiec

An integrated optimization for minimizing the operation cost of home delivery services in O2O retail

Xu Wang[a,b*] and Jian Zhong[a,b]

[a]*Mechanical and Engineering School, Chongqing University, Chongqing, 400044, China*
[b]*Chongqing Key Laboratory of Logistics, Chongqing University, Chongqing, 400044, China*

C H R O N I C L E

Article history:
Received October 2 2022
Accepted December 25 2022
Available online
December, 25 2022

Keywords:
O2O retail
Home delivery services
Vehicle routing
Driver sizing
Driver scheduling

A B S T R A C T

During the spread of the epidemic, the home delivery service (HDS) has been quickly introduced by retailers which helps customers avoid the risk of viral infection while shopping at offline stores. However, the operation cost of HDS is a huge investment for O2O retailers. How to minimize the operating costs of HDS is an urgent issue for the industry. To solve this problem, we outline those management decisions of HDS that have an impact on operating costs, including dynamic vehicle routing, driver sizing and scheduling, and propose an integrated optimization model by comprehensively considering these management decisions. Moreover, the dynamic feature of online orders and the heterogeneous workforces are also considered in this model. To solve this model, an efficient adaptive large neighborhood search (ALNS) and branch-and-cut algorithms are developed. In the case study, we collected real data from a leading O2O retailer in China to assess the effectiveness of our proposed model and algorithms. Experimental results show that our approach can effectively reduce the operating costs of HDS. Furthermore, a comprehensive analysis is conducted to reveal the changing patterns in operating costs, and some valuable management insights are provided for O2O retailers. The theoretical and numerical results would shed light on the management of HDS for O2O retailers.

1. Introduction

During the spread of the epidemic, the online-to-offline (O2O) retail mode is currently popular in the retail industry. The O2O retail mode implements both offline sales and online sales in an offline store. Customers can place orders online and receive their goods through the home delivery service (HDS) offered by retailers, while traditional offline customers pick up the goods by themselves and check out at cashier desks (Chen, Fan, Gu, & Pan, 2022). As HDS helps customers avoid the risk of viral infection while shopping at offline stores, these retailers configured with HDS have grown rapidly (He et al., 2021). In practice, there are two specific modes to help retailers to configure HDS: self-building and platform mode. For the self-building mode, the retailer developed an app and built a fleet to provide HDS to consumers, such as Hema Fresh in China and Costco in the U.S. For the platform mode, retailers sell on an independent Online-to-offline (O2O) platform and the HDS is provided by the O2O platform, such as Meituan in China and Instacart, Postmates, and DoorDash in the U.S. In this paper, our research is arising from an O2O retailer with the self-building mode. To express more concisely, the O2O retailer that appears later refers to the self-building mode. A typical O2O retailer was used to demonstrate how it works. Hema Fresh (Hema, 2021), which was established in 2017, is a pioneering O2O retailer in China, has more than 200 brick-and-mortar stores, and offers more than 7,000 high-quality items to 25 million customers. Local customers within a 3km radius of the offline shop could place orders online and choose the specific time windows of HDS to receive their goods. The length of

* Corresponding author
E-mail: wx921@163.com (X. Wang)
ISSN 1923-2934 (Online) - ISSN 1923-2926 (Print)
2023 Growing Science Ltd.
doi: 10.5267/j.ijiec.2022.12.005

time windows is fixed and determined by the retailer, which we term the committed delivery time. Such as, Hema Fresh promises the HDS will be fulfilled within 30 minutes. As a result, the business hours of Hema Fresh are divided into multiple delivery periods based on the committed delivery time of HDS. Customers can choose which period to receive their goods. For the O2O retailers, HDS plays a key role, which directly affects customer satisfaction. However, the operation cost of HDS is a huge investment for O2O retailers. How to minimize the operation costs of HDS is an urgent issue for O2O retailers. Our research is focused on this question. To solve this problem, we divided the operation cost into transportation costs and employment costs by analyzing the operating process of HDS, and outlined those management decisions that have an impact on operating costs, including dynamic vehicle routing, driver sizing and scheduling. Furthermore, our proposed an integrated optimization method by comprehensively considering the impact of these management decisions on operation cost. To the best of our knowledge, few studies have integrated these issues to comprehensively analyze the impact of these multiple management decisions on HDS' operation costs (see the literature review in Section 2).

In addition, these unique characteristics of the O2O retail industry set our study apart from others, such as the dynamic features of online orders and the heterogeneous workforces. Firstly, the volume of online orders fluctuates dramatically both on an hourly and daily basis. In hourly terms, the volume of online orders has two distinct peak periods: lunchtime and dinner time. In daily terms, the volume of online orders also fluctuates greatly on holidays, weekends, and weekdays. Secondly, there are three types of workforce, namely, in-house, outsourcing, and crowdsourcing drivers, with different work shifts, shift lengths, serviceability, salary structures, and rest breaks. These unique characteristics of the O2O retail industry are also considered in our optimization model. Therefore, we developed an integrated optimization model to formulate the multiple management decisions, which also considers the dynamic features of online orders and the heterogeneous workforces. In this model, the multiple management decisions are formulated as three sub-models, which are multi-period vehicle routing problems with the deadline model (MPVRPD), the daily heterogeneous driver sizing and scheduling model (DHDSSM), and the monthly heterogeneous driver sizing and scheduling model (MHDSSM). Multiple interlinked constraints were incorporated into the three sub-models, which enables the integrated model to greatly reproduce the decision-making process in real scenarios. To obtain the optimal solution, the adaptive large neighborhood search (ALNS) algorithm and branch-and-cut algorithm are applied to solve the integrated model. A case study of an O2O retailer in China is explored to assess the effectiveness of our proposed model. In addition, comprehensive analyses are conducted to reveal the pattern of HDS efficiency changes, and several valuable management insights are provided. The contributions of this research are summarized as follows.

(1) We proposed a novel integrated optimization approach for O2O retailers to minimize the operation cost of HDS by integrating multiple decision-making processes, including vehicle routing, drive sizing, and scheduling.
(2) An integrated optimization model embedded with a scenario-based sample method is developed to formulate the problem, which considers the dynamic features of online orders and the heterogeneous workforces in the O2O retail industry.
(3) The adaptive large neighborhood search and branch-and-cut algorithms are applied to solve the integrated optimization model, and the effectiveness of our proposed model and algorithms is verified by a real case study of a leading Chinese O2O retailer.
(4) A comprehensive analysis is conducted to uncover patterns of change in HDS operating costs, and several valuable management insights are provided. The theoretical and numerical results would shed light on the management of HDS for O2O retailers.

The remainder of this paper is organized as follows. Section 2 presents the related literature. The integrated optimization model and algorithms are introduced in Section 3. Section 4 is the case study. The discussion is shown in Section 5. Section 6 is the conclusion and directions for future research.

2. Related works

The multiple decision-making processes in HDS consisting of vehicle routing, drive sizing, and scheduling, are embedded in our integrated model. Therefore, we focus on compiling the literature on vehicle routing, driver sizing, and scheduling in O2O retail and other service sectors due to some similarities with O2O retail.

2.1 Dynamic vehicle routing problem

The operation cost of HDS includes transportation costs and employment costs. Numerous studies have shown that management decisions on vehicle routing have an impact on transportation costs (Soeffker, Ulmer, & Mattfeld, 2022). As a result, literature on vehicle routing issues has been compiled. Considering the dynamic feature of online orders, the vehicle routing problem arising in O2O retail mode can be classified as the dynamic vehicle routing problem (DVRP). The literature on DVRP emerged in the late 1970s, and developed substantially, particularly since the beginning of the century (Ojeda Rios et al., 2021). Pillac, Gendreau, Guéret, and Medaglia (2013) provided a classification of DVRPs into four categories. Based on the operation process of O2O retail mode, the DVRP in O2O retail mode can be classified as static and deterministic problems (one of the four categories). In static and deterministic problems, all input is known beforehand and vehicle routes do not change once they are in execution. Multi-period static programming is an effective strategy to solve this type of DVRP, which transformed the problem into multiple static vehicle routing problems (Ojeda Rios et al., 2021). Therefore, we formulate the DVRP in O2O retail mode as the static multi-period capacity vehicle routing problem (MPCVRP). Considering the time windows feature of online orders in HDS, the MPCVRP arising in O2O retail mode can be further classified as MPVRPTW. Some literature has adopted this model to formulate the DVRP in O2O retail mode. Lespay and Suchan (2022) introduced the territory design for MPCVRPTW arising in a food company's distribution center and proposed a mixed-integer linear program (MILP) and heuristic algorithm to solve the problem. Wang et al. (2021) addressed the issue of collaboration and transportation resource sharing in retail, by formulating and solving a two-echelon collaborative multi-depot MPCVRP. And they proposed a multi-objective integer programming model and a heuristic algorithm to solve the problem. Formulating the dynamic routing plan problem as MPCVRPTW is feasible for most scenarios, but in this scenario of our research, a more efficient model can be used.

In our research scenarios, the time window of online orders is fixed based on the typical retailer's presentation in Section 1. In addition, the time window of these orders in a period is the same. Hence, for the DVRP arising in O2O retail, we can adopt a more relaxed formulation and define it as a multi-period capacity vehicle routing problem with a deadline (MPVRPD). Compared to the MPCVRPTW model, which is widely used in this field, MPVRPD is not only equally applicable to our research, but it is also more efficient to solve. The algorithm used to solve MPVRPTW also can be used to solve MPCVRPD. These algorithms can be divided into exact algorithms and heuristic algorithms. An overview of exact algorithms can be seen in these papers (Dayarian, Crainic, Gendreau, & Rei, 2015; Desaulniers, Pecin, & Contardo, 2019; Lam, Desaulniers, & Stuckey, 2022). However, the MDCVRPD is an NP-hard problem and its scale is much larger in real-world applications. Metaheuristics are often more suitable for practical applications. Therefore, we propose an improved ALNS algorithm to solve the CVRPD that is obtained by decomposing MDCVRPD.

ALNS is a relatively new metaheuristic that was first introduced by Stefan Ropke and David Pisinger in their seminal works (Pisinger & Ropke, 2007; S. Ropke & D. Pisinger, 2006; Stefan Ropke & David Pisinger, 2006) as an extension of the large neighborhood search (Shaw, 1998). ALNS framework offers several benefits for the designers of optimization algorithms (Turkeš, Sörensen, & Hvattum, 2021). The framework of ALNS allows us to utilize multiple neighborhoods within the same searching process in an adaptive way, where this adaptiveness is attained by recording the performance of each neighborhood and dynamically adjusting the selection of methods according to this record (Keskin, Laporte, & Çatay, 2019). In addition, the adaptive selection of neighborhoods provides some extra freedom for the designers to incorporate more destroy and repair operators as the dynamic selection mechanism will limit the execution of ineffective operators(Guastaroba, Côté, & Coelho, 2021). An overview of the ALNS can be found in these reviews (Ahuja, Ergun, Orlin, & Punnen, 2002; Elshaer & Awad, 2020; Windras Mara, Norcahyo, Jodiawan, Lusiantoro, & Rifai, 2022).

2.2 Heterogeneous driver sizing and scheduling

The literature on workforce scheduling is quite vast, but we mainly focus on the literature about workforce scheduling in the O2O retail and other service sectors due to some similarities with retail. Several review papers were published covering the workforce scheduling problems for nurse rostering, physician scheduling, and railway crew scheduling (Defraeye & Van Nieuwenhuyse, 2016; Erhard, Schoenfelder, Fügener, & Brunner, 2018; Lee & Loong, 2019; Maria Gonzalez-Neira, Montoya-Torresc, & Barrera, 2017; Ojstersek, Brezocnik, & Buchmeister, 2020). And several authors contributed to workforce scheduling in the traditional retail sector include (Álvarez, Ferrer, Muñoz, & Henao, 2020; Bürgy, Michon-Lacaze, & Desaulniers, 2019; Chen et al., 2022; Mani, Kesavan, & Swaminathan, 2015; Mou & Robb, 2019). To the best of our knowledge, no review paper on driver scheduling problems in the O2O retail context has been published. DHDSSM and MHDSSM arising in O2O retail can be seen as an extension of workforce scheduling problems. Baker Kenneth (1976) classified workforce scheduling problems into three types based on the scheduling cycle; (1) shift scheduling: which refers to the assignment of daily work shift; (2) day-off scheduling: which refers to the assignment of days-off into in weekly schedule; and (3) weekly tour scheduling: it refers to the combination of shift scheduling and day-off scheduling. Considering that the employment cycle of drivers in O2O e-commerce is measured in months, we combine shift scheduling, day-off scheduling, and weekly tour scheduling, to extend the scheduling problem as DHDSSM and MHDSSM. Furthermore, each month, the volume of online orders also fluctuates greatly on holidays, weekends, and weekdays. Hence, a scenario-based sample method is applied to predict daily online orders (Liu, Sadowska, & De Schutter, 2022; Qian, Guo, Sun, & Wang, 2022), which makes the driver's scheduling scheme more accurate.

In this paper, our proposed problem has unique features. Ernst, Jiang, Krishnamoorthy, and Sier (2004) classified the scheduling problems into three categories based on the demand: (1) task-based: it refers to the demand of employees obtained for the individual task; e.g., transport scheduling, bus scheduling; (2) shift-based: it refers to the number of workers required to be on duty during the different shift; e.g., nurse scheduling, ambulance scheduling, and crew scheduling; and (3) flexible demand: it refers to varying demand in the smallest period of the planning horizon. e.g., retail workforce scheduling, and call center scheduling. Based on the categories, our proposed DHDSSM is a mixed scheduling problem, due to the scheduling of crowdsourcing drivers in O2O retail mode being flexible demand, and the in-house and outsourcing drivers being shift-based. In addition, another feature that differs from other studies is the heterogeneity of drivers considered in our research. Xu et al. (2022) considered two types of individuals for the vehicle routing problem in disaster scenarios. Another heterogeneity of workers, such as learning ability and worker skills, was also considered in these studies (Fang, Guan, Yue, Zhang, & Wang, 2022; Fichera, Costa, & Cappadonna, 2017; Toth & Kulcsar, 2021). In the O2O retail industry, there are three different types of drivers that can be hired to deliver goods. For each type of driver, their salary structure, serviceability, work shift, shift length, and rest break are different. In our study, we comprehensively considered these heterogeneities. In addition, the DHDSSM and MHDSSM are integrated with the dynamic vehicle routing problem in this paper, which is more difficult to solve. Zamorano and Stolletz (2017) devised a branch-and-price algorithm to solve the technician routing and scheduling problem with up to about 66 customers. Pereira, Alves, and Moreira (2020) proposed a constructive algorithm and heuristic algorithms based on ant colony optimization to solve the multiperiod workforce scheduling and routing problem with up to about 20 customers and 60 tasks. Punyakum, Sethanan, Nitisiri, Pitakaso, and Gen (2022) proposed a heuristic algorithm to tackle a multi-visit and multi-period workforce scheduling and routing problem with up to about 100 customers. In this scenario of our search, the number of online orders during peak periods is more than 200. Existing studies have not dealt with such large-scale workforce and routing problems. Therefore, it is necessary to design new models and algorithms to solve large-scale workforce and routing problems.

Based on the literature review and the characteristics of our research questions, a more relaxed formulation for DVRP arising in O2O retail is proposed. In addition, compared with the literature on workforce size and scheduling problem, our proposed DHDSSM and MHDSSM has more unique features. Furthermore, these existing approaches to the integrated problem consisting of vehicle routing and workforce scheduling are not applicable to large-scale instances. In this way, our study contributes to filling the above gaps.

3. Proposed method

3.1. Problem statement

The research problem in this paper arises from a pioneering O2O retailer in China that has self-built HDS. The O2O retailer remains open from 8:30 to 22:00, seven days a week. Each day, the business hours are divided into multiple delivery periods based on the committed delivery time. For this retailer, the committed delivery time is set to 30 minutes, hence the business hour is divided into 27 delivery periods. Customers within a 3km radius of the offline shop can place orders online and choose a suitable delivery period to receive their goods. The volume of online orders fluctuates dramatically both on an hourly and daily basis. In hourly terms, the volume of online orders has two distinct peak periods: lunchtime and dinner time. In daily terms, the volume of online orders also fluctuates greatly on holidays, weekends, and weekdays. In O2O retail, there are three types of drivers available (in-house, outsourcing, and crowdsourcing drivers), which have different work shifts, shift lengths, service quality, salary construct, and rest breaks. For each driver, all depart from the offline shop. After completing the delivery, in-house and outsourcing drivers need to return to the offline shop and wait for the next task. Crowdsourcing drivers don't need to go back to the shop. O2O retailers have strict requirements for HDS service.

In this scenario, the operation cost of HDS is a huge investment for O2O retailers. How to maintain the service level of HDS within the acceptable range of customers with minimum operation cost is a new challenge for O2O retailers.? We decompose the complex problem into three sub-problems, as follows.

(1) How to minimize transportation costs for each period?
(2) How to minimize employment costs for each day?
(3) How to minimize the driver sizing for one month?

Meanwhile, the requirements of retailers, customers, and drivers need to be satisfied. In these problems, we make the following assumptions which are common in the literature on O2O retail. For each type of driver, the average speed obtained from historical orders is adopted to represent their delivery speed. For each type of driver, the average service score obtained from the customer's rating in historical orders is adopted to represent their service quality. If the customer's rating is higher, its service quality is better.

The transportation and employment costs of HDS have an impact on management decisions regarding vehicle routing, driver sizing, and scheduling. Therefore, this paper proposed a systematic methodology to solve these problems, which will be introduced in the next part. To make the methodology easier to understand, the mathematical notation and description in this paper are presented in Table 1.

Table 1
Definition of symbols.

Set	Description
V,E	Sequences of vertices V and edges E in graph G, $V=\{1,\dots v\dots n\}$, $E=\{1,\dots e\dots n\}$, $v \in V$, $e \in E$.
M	Sequences of days in a month, $M=\{1,\dots m\dots 28\}$, $m \in M$.
M'	Sequences of weekends in a month, $M'=\{6,\dots m\dots 28\}$, $m \in M'$.
T	Sequences of periods in a day, $T=\{1,\dots t\dots n\}$, $t \in T$.

O^t	Sequences of online orders in the t^{th} period, $O^t=\{1,\ldots i\ldots n\}$, $i \in O^t$, $t \in T$.
S^t	Sequences of vehicle demands in the t^{th} period, $S^t=\{1,\ldots s\ldots n\}$, $s \in S^t$, $t \in T$.
W	Driver's types of in-house, outsourcing, and crowdsourcing are indicated by serial numbers 0, 1, and 2, $W=\{0,1,2\}$, $w \in W$.
H_w	Order of work shifts for w^{th} type drivers, $H_w=\{1,\ldots h,\ldots,n\}$, $h \in H_w$, $w \in W$.
Parameters	
Q_w	Number of available drivers with the w^{th} type, $q \in Q_w$, $w \in W$.
p_w^h	Number of peak periods covered by the h^{th} work shift of the w^{th} type drivers, $w \in W$, $h \in H_w$.
l_w	Shift length of the w^{th} type drivers, $w \in W$.
r_w	Service quality of the w^{th} type drivers, $w \in W$.
rs_i	Customer's rating score of the i^{th} order, $i \in O$.
tg_i	The generation time of the i^{th} order; $i \in O$.
tc_i	Completion time of the i^{th} order; $i \in O$.
te_i	Earliest arrival time of the i^{th} order; $i \in O$.
tl_i	Latest arrival time of the i^{th} order; $i \in O$.
lp_i	Pick-up location of the i^{th} order; $i \in O$.
ld_i	Drop-off location of the i^{th} order; $i \in O$.
d_{ij}	Distance between the i^{th} order and j^{th} order, $i,j \in O$.
fc_w	Fixed cost per day for the w^{th} type drivers, $w \in W$.
vc_w	Variable cost per order for the w^{th} type drivers, $w \in W$.
a	Minimum working day in a month.
b	Maximum working day in a month.
$\boldsymbol{c}$	Vehicle transportation costs per kilometer.
ẟ	Average number of times per month that drivers serve during peak hours.
Ʊ	Average speed of the driver in meters per second.
Ɛ	Probability of crowdsourcing drivers accepting orders.
uſ	Maximal number of days off scheduled on weekends per month.
$\boldsymbol{\beta}$	Maximum capacity of vehicles.
$\mathscr{R}$	Average service time for the customer, such as parking time, upstairs time, etc.
Θ	Minimum service score required by retailers.
Decision Variables	
u_i	Arriving time of the i^{th} order, $i,j \in O^t$.
v_{ij}^s	If s^{th} vehicle contains the i^{th} order and j^{th} order, $v_{ij}^s = 1$;otherwise, 0. $i,j \in O^t$, $s \in S^t$.
x_{sw}^t	In the t^{th} period, if the s^{th} vehicle is assigned to w^{th} type drivers, $x_{sw}^t = 1$; otherwise, 0. $t \in T$, $s \in S^t$, $w \in W$.
y_w^h	Number of the w^{th} type drivers required for the h^{th} work shift, $h \in H_w$, $w \in W$.
z_{qw}^{mh}	If the q^{th} of the w^{th} type driver works on the h^{th} work shift of the m^{th} day, $z_{qw}^{mh} = 1$; otherwise, 0. $m \in M$, $q \in Q_w$, $h \in H_w$, $w \in W$.
λ_q^w	If the q^{th} of the w^{th} type driver is employed, $\lambda_q^w = 1$; otherwise, 0. $w \in W$, $q \in Q_w$.

3.2 Methodology

A three-stage optimization methodology is proposed to minimize the transportation and employment costs of HDS. The framework of the three-stage optimization methodology is shown in Fig. 1. At first, a scenario-based sample method is applied to formulate the hourly and daily fluctuations of online order volumes. Historical data of online orders in these scenarios including peak periods, weekdays, weekends, and festivals are used to predict the order volumes. Machine learning is a powerful tool for scenario prediction and makes predictions more accurate (Hendry & Pretis, 2022; Liu et al., 2022; Qian et al., 2022). Based on the forecast data, we can further optimize the vehicle routing, driver sizing, and scheduling decisions.

In this methodology, three interconnected sub-models are constructed based on the multiple management decisions in HDS, which are MPVRPD, DHDSSM, and MHDSSM. For each period, MPVRPD is applied to minimize transportation costs TC_t^m and obtain the low bound of drivers' demand numbers. After the low bound of drivers' demand number in each period is obtained by the MPVRPD, which will be used as a constraint in DHDSSM. DHDSSM is developed to obtain the minimum demand number for different types of drivers in a day. The daily employment costs are denoted as EC^m. Until the daily demand numbers for different types of drivers are obtained, MHDSSM is applied to obtain the monthly minimal driver sizing.

Using a three-stage model to describe the problem can improve computational efficiency and facilitates solving larger-scale cases. Since MPVRPD is an NP-hard problem, if we integrate this problem with the driver sizing and scheduling problems into a single model, the computation time of this model will be unacceptable when dealing with large-scale cases. Therefore, we developed the three-stage model. Furthermore, multiple interlinked constraints were incorporated into the three sub-models, which enables the model to greatly reproduce the decision-making process of HDS in real scenarios. Each sub-model will be described in detail in the next section.

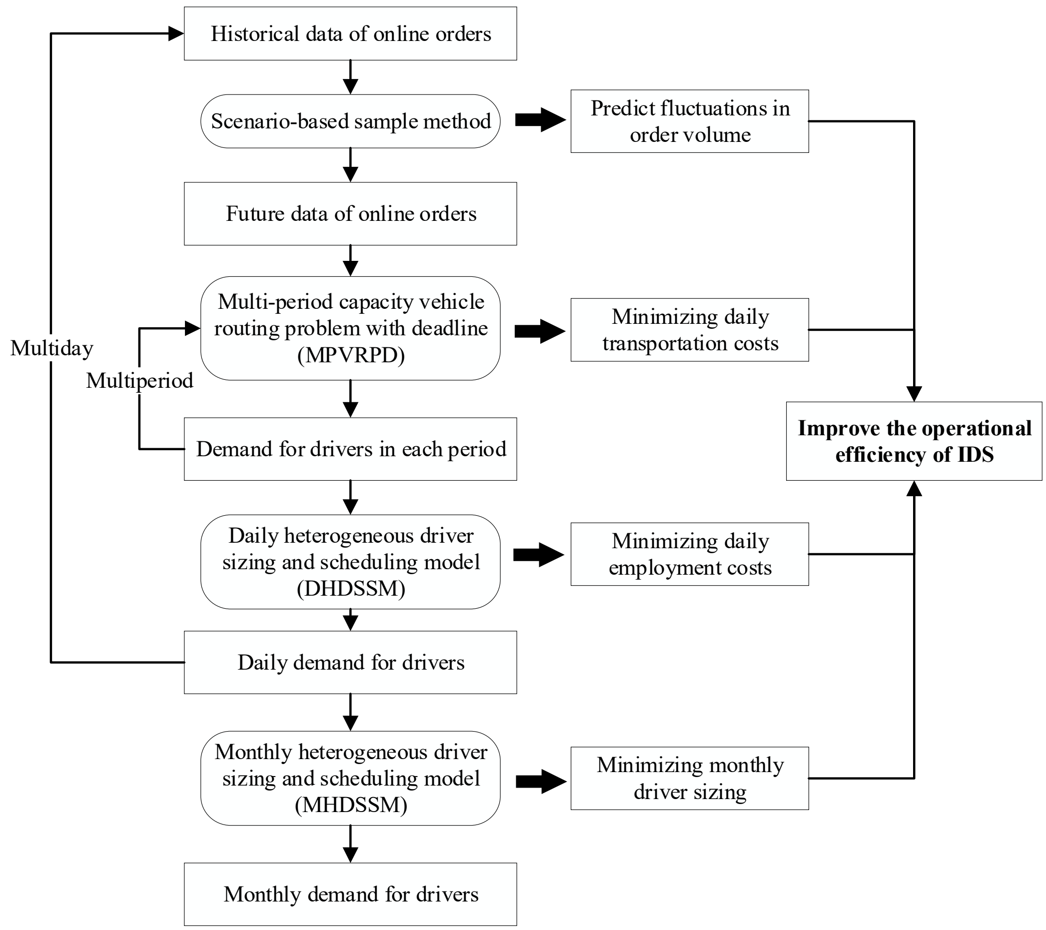


Fig. 1. Framework of the three-stage optimization methodology.

3.3 Models

3.3.1 Model of MPVRPD

The first-stage model is MPCVRPD. Classical models of MPVRPD can be found in these papers (Dayarian et al., 2015; Larrain, Coelho, Archetti, & Speranza, 2019; Lespay & Suchan, 2022). Considering the characteristics of the O2O retail, we modified the classical model as follows.

(1) Objective function

For each period, the objective function aims to minimize the transportation cost TC_t^m. The transportation cost is mainly related to the travel distance, so the objective function is shown below.

$$TC_t^m = \text{Min} \sum\nolimits_{k \in S} \sum\nolimits_{(i,j) \in O_t} d_{ij} v_{ij}^s * \text{c} \qquad \forall t \in T, m \in M \qquad (2)$$

(2) Constraints

Constraints (3) and (4) indicate that vehicles start and end at the offline shop.

$\sum_{(0,j)\in O^t} v_{0j}^s = 1 \qquad \forall s \in S^t \qquad (3)$

$\sum_{(i,n+1)\in O^t} v_{i,n+1}^s = 1 \qquad \forall s \in S^t \qquad (4)$

Constraint (5) represents that each order must be delivered.

$\sum_{s\in S^t} \sum_{(i,j)\in O^t} v_{i,j}^s = 1 \qquad \forall i,j \in O^t \qquad (5)$

Constraint (6) indicates the volume of orders assigned to vehicles could not exceed the capacity of the vehicle.

$\sum_{(i,j)\in O^t} v_{i,j}^s \le \beta \qquad \forall s \in S^t \qquad (6)$

Constraints (7) and (8) show that the arrival time of orders must be less than the deadline.

$u_i + d_{ij}/\mho + \mathcal{R} - M(1 - v_{ij}) \le u_j \qquad \forall i,j \in O^t, s \in S^t \qquad (7)$

$u_i \le t_i^l \qquad \forall i \in O^t \qquad (8)$

Constraint (9) is the flow balance constraint.

$\sum_{(i,j)\in O^t} v_{ij}^s - \sum_{(j,i)\in O^t} v_{ji}^s = 0 \qquad \forall s \in S^t \qquad (9)$

Constraints (10) and (11) denote the range of decision variables.

$v_{ij}^s \in \{0,1\} \qquad \forall i,j \in O^t,\ s \in S^t \qquad (10)$

$u_i \ge 0 \qquad \forall i \in O^t \qquad (11)$

3.3.2 Model of DHDSSM

The second-stage model is DHDSSM. An integer linear programming model is devised to formulate the DHDSSM, which aims to minimize daily employment costs. In this model, the heterogeneity of drivers with different work shifts, shift lengths, serviceability, and salary structures are considered. Different work shifts H_w mean that each driver's work shift can start at a different time. And the shift length l_w of different types of drivers is different. If the retailer's business hours are from 8:30 to 22:00, the relationship between H_w and l_w can be expressed as follows, taking in-house drivers as an example. Set the shift length of l_0 to 8 hours, and work shifts of in-house drivers are obtained by dividing business hours by the shift length l_0. The order of these work shifts is stored in H_0, noted as $H_0 = \{1, \dots, 11\}$, $h \in H_0$. If an in-house driver is assigned to work in the first work shift, $h = 1$, it means the in-house driver's working hours are from 8:30 to 16:30. If an in-house driver is assigned to work in the last work shift, $h = 11$, it means the in-house driver's working hour is from 14:00 to 22:00.

(1) Objective function

For each day, the objective function EC^m aims to minimize the daily employment costs, which are related to the number of orders assigned to the employee and the fixed employment cost. Therefore, the objective function is shown below.

$$EC^m = \text{Min} \sum_{t\in T} \sum_{s\in S^t} \sum_{w\in W} x_{sw}^t * vc_w * |s| + \sum_{w\in W} \sum_{h\in H_w} y_w^h * fc_w \qquad (12)$$

(2) Constraints

Constraints (13) indicates that each vehicle can be assigned to one driver.

$$\sum_{w\in W} x_{sw}^t = 1 \qquad \forall s \in S^t, t \in T \qquad (13)$$

In constraints (14), the number of vehicles required in the t^{th} period is denoted as $|S^t|$, which is obtained from the output of MPVRPD. Based on MPVRPD, we can obtain the number of vehicles required in each period. Constraints (14) represents that the number of drivers working in the t^{th} period must be greater than the low bound of vehicles' required numbers $|S^t|$ in the same period.

$$\sum_{w\in W} \sum_{t\subset h, h\in H_w} y_w^h \ge |S^t| \qquad \forall t \in T \qquad (14)$$

Constraints (15) emphasizes that the number of drivers working in the t^{th} period must be greater than the number of vehicles required in the t^{th} period.

$$\sum_{s \in S^t} x_{sw}^t \leq \sum_{t \subset h, h \in H_w} y_w^h \qquad \forall t \in T, w \in W \tag{15}$$

Since the working hours of crowdsourcing drivers are flexible, we apply probabilities to estimate the number of crowdsourcing drivers required in the *t^th^* period. In addition, the working time of crowdsourcing drivers is limited during the peak time T'. Constraints (16) indicates the number of crowdsourcing drivers required in the *t^th^* period.

$$\sum_{s \in S^t} x_{sw}^t \leq \mathcal{E} * |S^t| \qquad w=2, \forall t \in T' \tag{16}$$

Constraint (17) ensures that the service quality meets the requirements of O2O retailers.

$$\sum_{t \in T} \sum_{s \in S^t} \sum_{w \in W} x_{sw}^t * r_w * |s| \geq \Theta \tag{17}$$

Constraints (18) and (19) denote the range of decision variables.

$$s_{sw}^t = \{0,1\} \qquad \forall t \in T, s \in S^t, w \in W \tag{18}$$

$$y_w^h \geq 0, y_w^h \in Z \qquad \forall h \in H_w, w \in W \tag{19}$$

(3) Valid inequalities

We introduce two valid inequalities to strengthen the constraint (19), the inequalities are expressed as follows.

$$y_w^h \leq Max\{S^t\} \qquad w \in \{0,1\}, S^t \subset S \tag{20}$$

$$y_2^h \leq Max\{S^t\} * E \qquad w = 2, S^t \subset S \tag{21}$$

3.3.3 Model of MHDSSM

In the third stage, the integer nonlinear programming model is devised to formulate the MHDSSM, which aims to minimize the monthly drive sizing. In this paper, we assume that each month consists of four weeks. And the model is shown below.

(1) Objective function

Since crowdsourcing drivers have free working hours and they decide on their own what time to start working, we just consider in-house and outsourcing drivers in MHDSSM. The driver's types of in-house, outsourcing are indicated by serial numbers 0 and 1, $W'=\{0,1\}$, $w \in W'$.

$$Min \sum_{w \in W'} \sum_{q \in Q_w} \lambda_q^w \tag{22}$$

(2) Constraints

Constraint (23) limits that each driver can only choose one work shift in a day.

$$\sum_{h \in H_w} z_{qw}^{mh} \leq 1 \qquad \forall w \in W', q \in Q_w, m \in M \tag{23}$$

Constraints (24) and (25) represent the driver's workday constraints.

$$\sum_{m \in M} \sum_{h \in H_w} z_{qw}^{mh} \leq b \qquad \forall w \in W', q \in Q_w \tag{24}$$

$$if\ \lambda_q^w = 1, \text{then} \sum_{m \in M} \sum_{h \in H_w} z_{qw}^{mh} \geq a \qquad \forall w \in W', q \in Q_w \tag{25}$$

In constraint (26), y_w^h is the output of DHDSSM, which be used as a parameter in MHDSSM. Based on DHDSSM, we can obtain the number of drivers required for each day, denoted as y_w^{mh}. Constraint (26) indicates that the number of working drivers on the *m^th^* day must be greater than the number of drivers required on the same day.

$$\sum_{q\in Q_w} z_{qw}^{mh} \geq y_w^{mh} \qquad \forall w \in W', m \in M, h \in H_w \tag{26}$$

Constraint (27) shows that the driver is hired as long as he or she is scheduled for a work shift.

$$\lambda_q^w = \begin{cases} 0, & \sum_{m\in M}\sum_{h\in H_w} z_{qw}^{mh} \leq 0 \\ 1, & \sum_{m\in M}\sum_{h\in H_w} z_{qw}^{mh} \geq 1 \end{cases} \qquad \forall w \in W', q \in Q_w \tag{27}$$

The constraint (28) indicates that there is a limitation on the number of days off scheduled in weekends per month. Scheduling a day off on a weekday is not the same as a weekend, and more drivers want their day off to be scheduled on a weekend. That way they can spend the weekend with their families.

$$if\ \lambda_q^w = 1, \text{then}, \sum_{m\in M'}\sum_{h\in H_w} z_{qw}^{mh} \leq \text{ʉ} \qquad \forall w \in W', q \in Q_w \tag{28}$$

The constraint (29) indicates that the higher the service quality of the driver, the more frequently the driver works during peak periods.

$$if\ \lambda_q^w = 1, \text{then} \sum_{m\in M}\sum_{h\in H_w} z_{qw}^{mh} * p_w^h \geq \text{ẟ} + \lceil r_w - \theta \rceil \qquad \forall w \in W', q \in Q_w \tag{29}$$

Constraints (30) and (31) denote the range of decision variables.

$$z_{qw}^{mh} \in \{0,1\} \qquad \forall m \in M, q \in Q_w, h \in H_w, w \in W \tag{30}$$

$$\lambda_q^w \in \{0,1\} \qquad \forall w \in W, q \in Q_w \tag{31}$$

(3) Valid inequalities

The constraint (27) is relaxed to the formulation (32), which can simplify the model and improve the calculation efficiency.

$$\sum_{m\in M}\sum_{h\in H_w} z_{qw}^{mh} \leq \lambda_q^w * |M| \qquad \forall w \in W', q \in Q_w \tag{32}$$

3.4 Solving algorithm

Since the three sub-models have different computational scales in the application, we apply heuristic and exact algorithms to solve them respectively. In the scenario we studied, the number of online orders during peak periods exceeded 200, and the exact-based algorithm could not solve the MPVRPD within an acceptable time. Therefore, an adaptive large neighborhood search algorithm (ALNS) is developed to solve the MPVRPD. Besides, the branch-and-cut algorithm is applied to solve DHDSSM and MHDSSM.

The ALNS developed in this paper is an extension of the route minimization heuristic algorithm proposed by Nagata and Bräysy (2009) and the slack induction by string removals heuristic algorithm proposed by Christiaens and Vanden Berghe (2020). The main feature of our approach is that several novel destroy operators are devised and embedded in the ALNS. Moreover, a simple adaptive local search mechanism is developed to improve the quality of solutions. The next subsections will discuss the main building blocks of ALNS, starting with an overview of ALNS.

Fig. 2 gives an overview of ALNS where the *deleteTime*, *repairTime*, and *destoryTime* are the user-defined parameters. Our proposed ALNS is based on a two-stage approach where the number of routes and the travel distance is independently minimized. In the first stage (lines1-8), the aim is to minimize the number of routes, and the aim of the second stage (lines 9-15) is to minimize the travel distance. ALNS starts with an initial solution (line 1) generated by the function called INITIAL_SOLUTIONS(***N_{cus}***), where ***N_{cus}*** is the number of customers. In the initial solution, each customer is served individually by a separate route. And then the procedure (lines 2-8) is repeatedly applied to reduce the number of routes one by one until the total computation time reaches a given limit.

Algorithm 1 (Adaptive large neighborhood search)

```
begain
1:   s := INITIAL_SOLUTIONS(N_cus);
2:   While time < deleteTime do
3:       s', EP := DELETE_ROUTE(s)
4:       While EP ≠∅ or time < repairTime do
5:           s", EP = REPAIR_ALGORITHM(s', EP)
6:       end while
7:       If EP = = ∅ then s := s"
8:   end while
9:   While time < destoryTime do
10:      s', EP = DESTORY_ALGORITHM(s)
11:      While EP ≠∅  and time < reapairTime do
12:          s", EP = REPAIR_ALGORITHM(s', EP)
13:      end while
14:      If EP = = ∅ then s := s"
15:  end while
end
```

Fig. 2. Framework of the adaptive large neighborhood search.

To minimize the number of routes, the function DELETE_ROUTE(s) is started by selecting and removing a route randomly from the current solution *s* (line 3). Thus, the output *s'* of the algorithm is a feasible partial solution, and the customer concluded in the removed route is stored to the ejection pool *EP*. The main idea of the ejection pool is to hold the set of unserved (ejected) customers. And then, the function REPAIR_ALGORITHM(s', EP) is devised to insert the unserved customer in the partial feasible solution *s'* one by one until the total computation time reaches a given limit or the *EP* is empty. After this loop, we check whether *EP* is empty. If *EP* is empty, it means that we find a new feasible complete solution. We can be based on the new solution and start the next iteration.

In the second stage (lines 9–15), ALNS aims to minimize the travel distance until the computation time reaches a given time limit. For each iteration, the first function DESTORY_ALGORITHM(s) is designed to delete the bad customer nodes from the solution *s*. The classical destroy operators such as random removal (Stefan Ropke & David Pisinger, 2006), worst removal (S. Ropke & D. Pisinger, 2006), and shaw removal (Christiaens & Vanden Berghe, 2020) are applied in these functions. After that, the function REPAIR_ALGORITHM(s', EP) is applied again to insert these bad customer nodes in the partial feasible solution *s'* one by one until the total computation time reaches a given limit or the *EP* is empty. After this loop, we check whether *EP* is empty. If *EP* is empty, we base on the new solution to start the next iteration.

Algorithm 2 (Repair algorithm)

```
Input   s', EP
1:   While EP ≠∅  and time < reapairTime do
2:       Remove a customer v form EP with LIFO strategy
3:       s" := INSERT_ALGORITHM(v, s')
4:       If s" is infeasible then
5:           s" := ADAPTIVE_LOCAL_SEARCH(s")
6:       If s" is infeasible then
7:           s", EP' = EJECTION_ALGORITHM(v, s")
8:           EP = EP U EP'
9:       s' := s"
10:  End while
11:  If EP ≠ ∅ then s" := s'
Output  s", EP
```

Fig. 3. The framework of the repair algorithm.

Fig. 3 gives an overview of the repair algorithm. This algorithm starts with the removed customer *v* which is selected from the EP according to the last in first out strategy (LIFO). First, the INSERT_ALGORITHM(*v, s'*) is applied to insert the customer *v* in the position with minimal distance. If there is not a feasible solution, the ADAPTIVE_LOCAL_SEARCH(*s''*) is applied to search for a feasible solution in the neighboring area. If there is not a feasible solution in the neighboring area, the EJECTION_ALGORITHM(*v, s''*) is used to randomly eject customers until the solution *s''* is feasible. the ejection customer is added to the EP (line 8). Finally, the feasible solution *s''* are copied to *s'*, start the next iteration. After this loop, we check whether the EP is empty. If EP is not empty, we restore the solution s'' to the input state.

Fig. 4 gives an overview of the adaptive local search algorithm. The classical relocate, exchange, 2opt, 2opt* operators are used to search for a better feasible solution. A roulette wheel is applied as the adaptive mechanism. The preference value (PV) is initialized as a dictionary, the key is the operator's name, and the value is initialized to 1 (line 1). The function RWSELSECTION(PV) based on the preference value PV can output the operator which has higher probabilities to improve the solution s. If the chosen operator finds a better feasible solution, the preference value PV of this operator is increased by 1, otherwise, 0.5.

Algorithm 3 (Adaptive local search algorithm)

```
input   s
1:      PV = {relocate:1,2opt*:1,exchange:1,2opt:1}
2:      While time < localsearchTime do
3:          operator_name = RWSELSECTION(PV)
4:          if operator_name =='relocate' then
5:              s', save_cost = RELOCATE(s)
6:          If operator_name ==' 2opt*' then
7:              s', save_cost = 2OPT* (s)
8:          If operator_name ==' exchange' then
9:              s', save_cost = EXCHANGE(s)
10:         If operator_name ==' 2opt' then
11:             s', save_cost = 2OPT(s)
12:         If s' is feasible and save_cost > 0 then
13:             PV[operator_name]=PV[operator_name]+1
14:             s := s'
15:         else:
16:             PV[operator_name] = 0.5
17:     End while
Output  s
```

Fig. 4. The framework of the adaptive local search algorithm.

The branch-and-cut algorithm is an efficient method to solve the integer programming model. Considering that the DHDSSM and MHDSSM model is very compact and the size of instances in the application is small, we just use the standard process of the Branch-and-cut algorithm to deal with this problem. The pseudo-code of the algorithm is shown in this article (Hopcroft 1973), so we do not make redundant claims here.

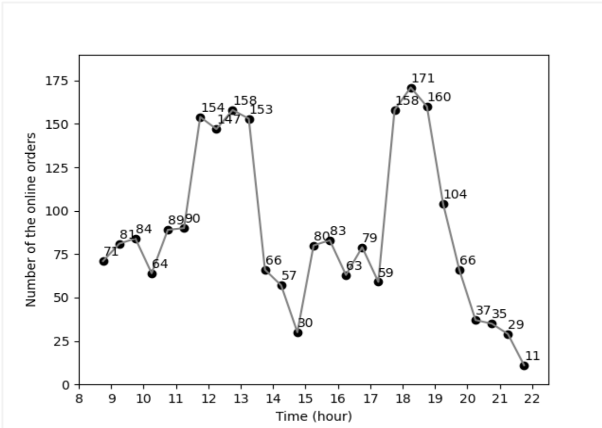
4. Experimental results

4.1 Data collected

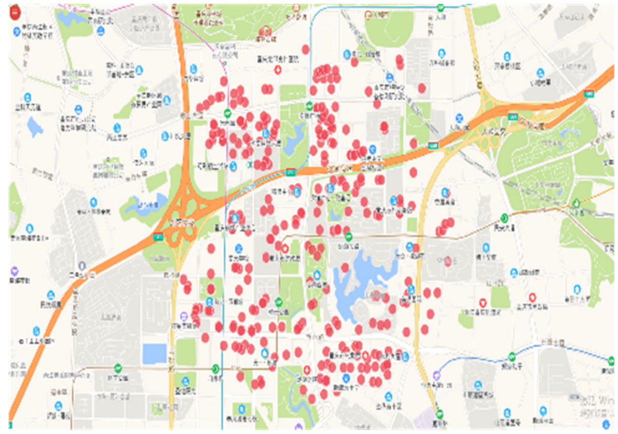
The data is collected from a leading O2O retailer in China which has more than 200 offline stores in China and over 400,000 online orders per day. We took one of these stores as the subject of our study. The offline store located in the Yubei district of Chongqing, China. Local customers can place orders online and choose an appropriate time to receive their goods by the HDS. The O2O retailer remains open from 8:30 to 22:00, seven days a week. During each day, the business hours are divided into 27 time periods with a length of 30 minutes. We chose orders from a typical weekday as experimental data to validate our approach. The experiment data contains 2380 online orders. The critical information on online orders is shown in Table 2. Complete data supporting the findings of this study are publicly available at https://github.com/0BigMax0/Data-BWSO. The temporal and spatial distributions of these online orders are shown in Figure 5. It's obvious that online orders fluctuate more sharply during peak periods: like lunchtime and dinner time. Online orders are distributed within 3 km of offline stores.

Table 2
Partial information of historical orders on a weekday

Order	tg_i	tc_i	te_i	tl_i	lp_i	ld_i	W	rs_i
1	8:12	8:52	8:30	9:00	[106.539375,29.592201]	[106.531681, 29.576249]	1	100
2	8:08	8:56	8:30	9:00	[106.539375,29.592201]	[106.531811, 29.584772]	2	95
3	8:08	8:58	8:30	9:00	[106.539375,29.592201]	[106.532845, 29.589659]	1	100
…	…	…	…	…	…	…	…	…
2378	21:19	21:51	21:30	22:00	[106.539375,29.592201]	[106.533968, 29.589057]	3	80
2379	21:26	21:52	21:30	22:00	[106.539375,29.592201]	[106.537291, 29.589327]	1	95
2380	21:27	21:43	21:30	22:00	[106.539375,29.592201]	[106.534673, 29.580077]	1	100



(a) Temporal distributions of orders



(b) Spatial distributions of orders

Fig. 5. Temporal and Spatial distributions of orders (a)-(b).

4.2 Parameter setting

To verify the effectiveness of our proposed integrated optimization methodology, the ALNS is coded in Python 3.9, and runs in Windows 10 with Intel Core i5-8250U CPU, with 1.80 GHz and 8.00 GB RAM. The branch-and-cut algorithm is called from the Gurobi solver in version 9.1.2. In Gurobi, the gap of the optimal solution is set to 5%, and the maximum computation time of Gurobi is set to 10 minutes. In ALNS, the parameters of *deleteTime, destoryTime,* and *reapairTime* are set to 40, 20, and 5 in seconds respectively. In addition, other parameters are obtained by drawing the experience of the manager with a deep background in the O2O retail industry. All parameters used in the experiment are shown in Table 3.

Table 3
Setting model's parameters.

Parameters	Unit	Value
W	--	[0,1,2]
M	--	[1,2,3,…,28]
T	--	[1,2,3,,..,27}
H_0	Hour	[8:30,9:00,…,14:00]
H_1	Hour	[8:30,9:00,…,14:00]
H_2	Hour	[8:30,9:00,…,21:30]
r_w	Score	[95,85,80]
fc_w	CNY	[35,150,0]
vc_w	CNY	[2,0,4]
l_w	Hour	[8,8,0.5]
a	Day	12
b	Day	20
c	CNY	0.4
Ʊ	m/s	11
ẟ	--	20
Ɛ	--	20 %
ư	Day	6
β	Orders	8
$\mathscr{R}$	Minute	3
Θ	Score	90

4.3 Computation results in the real case

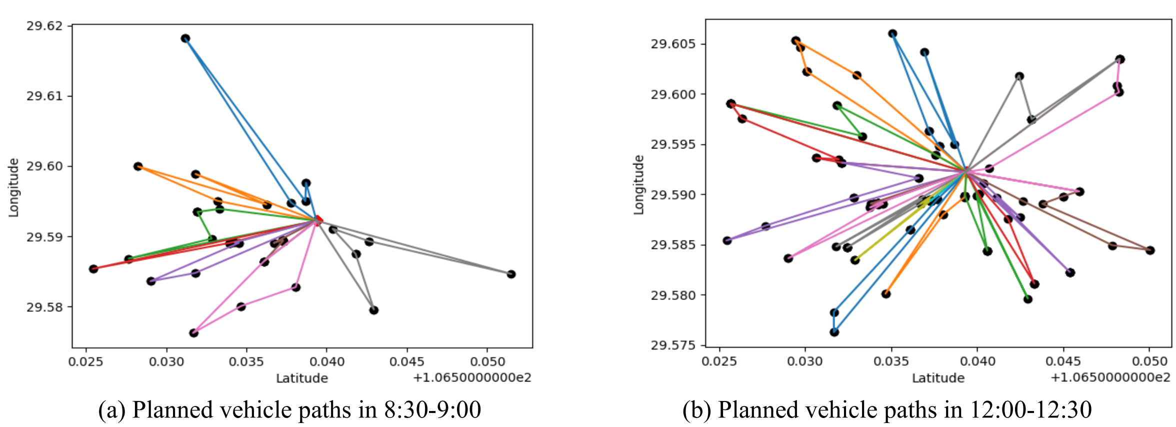
4.3.1 Result of MPVRPD in each period

To compare the efficiency of our proposed ALNS, the efficient memetic algorithm (MA) proposed by Nagata, Bräysy, and Dullaert (2010) is applied to solve the MPVRPD. Due to the very short committed delivery time, the calculation time for solving MPVRPD should ideally be less than 1 minute. In this paper, the calculation time for the ALNS and MA are limited to 1 minute. The experimental results are shown in Table 4. In Table 4, the first column represents the sequence of periods. The second and third columns show the start time and end time of the period. The fourth column represents the number of online orders in that period. The fifth column represents the results of the MA, which includes the minimum number of drivers (*Min.D*) and minimum transportation cost (*Min.C*) in CNY. The sixth column records the results of our proposed algorithm, which also includes the same indexes. The last two rows statistics the cumulative number of drivers and the total transportation costs.

Table 4
Results of MPVRPD on a typical day

Seq	t^e	t^l	*Num*	MA (1 min)		ALNS (1min)	
				Min.D	*Min.C*	*Min.D*	*Min.C*
1	8:30	9:00	71	10	12.07	10	10.95
2	9:00	9:30	81	12	11.54	11	10.45
3	9:30	10:00	84	13	13.81	11	12.76
4	10:00	10:30	64	11	9.04	9	8.54
5	10:30	11:00	89	13	13.40	12	11.80
6	11:00	11:30	90	13	13.17	12	12.21
7	11:30	12:00	154	21	19.12	20	17.88
8	12:00	12:30	147	20	17.23	19	16.79
9	12:30	13:00	158	21	20.77	20	20.60
10	13:00	13:30	153	20	20.71	20	18.91
11	13:30	14:00	66	10	7.73	9	7.28
12	14:00	14:30	57	9	6.88	8	6.10
13	14:30	15:00	30	4	6.00	4	6.21
14	15:00	15:30	80	11	12.04	11	11.87
15	15:30	16:00	83	11	13.49	11	12.70
16	16:00	16:30	63	9	8.00	8	7.39
17	16:30	17:00	79	12	11.53	10	10.45
18	17:00	17:30	59	8	8.05	8	7.32
19	17:30	18:00	158	21	18.51	20	19.09
20	18:00	18:30	171	22	22.95	22	20.70
21	18:30	19:00	160	23	21.72	21	19.82
22	19:00	19:30	104	16	15.13	14	15.39
23	19:30	20:00	66	11	11.85	9	11.00
24	20:00	20:30	37	5	5.12	5	5.27
25	20:30	21:00	35	5	7.35	5	7.47
26	21:00	21:30	29	5	5.20	4	4.86
27	21:30	22:00	11	2	2.26	2	2.26
	Total			338	334.67	315	316.08

The experiment results show that the ALNS is highly applicable to solving MPVRPD. The cumulative number of required drivers obtained by ALNS was 315, less than the 338 obtained by MA. In addition, ALNS obtained a total transportation cost of RMB 316.08. This is also better than the RMB 334.67 obtained by the MA. Therefore, our proposed ALNS is more effective for solving the dynamical vehicle routing problem of O2O home delivery services. Figure 6 shows part of the experimental results obtained by ALNS, which depicts the planned delivery paths in the periods 8:30-9:00 and 12:00-12:30. In this figure, the black point represents the location of online orders, and a colorful cycle represents a planned delivery path. These planned delivery paths have no significant redundancy and ensure that O2O retailers receive lower transportation costs.



(a) Planned vehicle paths in 8:30-9:00 (b) Planned vehicle paths in 12:00-12:30

Fig. 6. Planned delivery paths in partial time intervals (a)-(b).

4.3.2 Result of DHDSSM in each day

Based on the experimental result in MPVRPD, the minimum number of drivers required in each period is obtained. Then, we will solve the driver sizing and scheduling problem, which is formulated as DHDSSM. In this paper, the branch-and-cut algorithm is applied to solve the DHDSSM. The experiment results are shown below.

The minimum daily employment cost obtained by the Gurobi solver is 4327, and the decision information is shown in Table 5. In Table 5, the first column represents the sequence of periods. The second column represents the number of online orders. The third column shows the required number of drivers in each period, which are obtained by the MPVRPD. The last three columns show the decision information of the optimal scheduling plan, which records the number of drivers who start working at that time. The last row statistic is the total number of each column.

Table 5
The optimal solution of DHDSSM.

Seq	*Orders*	*Drivers*	*In-house*	*Outsourcing*	*Crowdsourcing*
1	71	10	1	10	0
2	81	11	0	0	0
3	84	11	0	0	0
4	64	9	0	0	0
5	89	12	1	0	0
6	90	12	4	1	0
7	154	20	1	4	0
8	147	19	0	3	0
9	158	20	0	1	0
10	153	20	1	0	0
11	66	9	1	1	0
12	57	8	0	2	0
13	30	4	0	0	0
14	80	11	0	0	0
15	83	11	0	0	0
16	63	8	0	0	0
17	79	10	0	0	0
18	59	8	0	0	0
19	158	20	0	0	0
20	171	22	0	0	2
21	160	21	0	0	2
22	104	14	0	0	0
23	66	9	0	0	0
24	37	5	0	0	0
25	35	5	0	0	0
26	29	4	0	0	0
27	11	2	0	0	0
Total	2380	315	9	22	4

The experimental results show that 31 drivers are needed to fulfill the whole day's online orders, and 4 planned delivery paths are needed for crowdsourcing. To more visually demonstrate the relationship between the driver's needs and the scheduling plan, we plotted Fig. 7, where the red line indicates the number of drivers required and the bar graph indicates the number of drivers supplied. In this figure, the number of drivers working in each period meet the number of drivers required. In addition, the number of idle drivers is higher in the middle of the day, so these idle drivers can have enough time to take turns for meals and rest. To make it easier for managers to use, the best decision information is presented in the form of a Gantt chart in Fig. 8. Each driver can access his work shift in the Gantt chart.

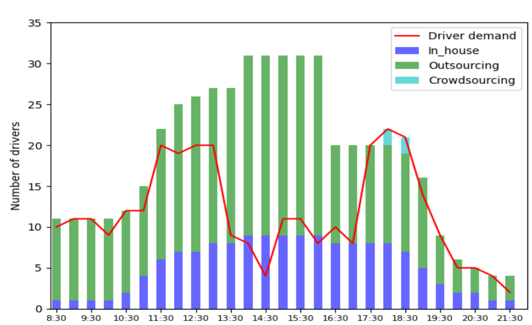


Fig. 7. Relationship between drivers' demand and the scheduling of drivers

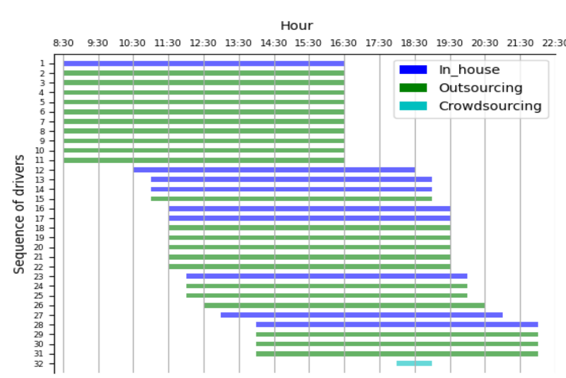


Fig. 8. Gantt chart for drivers' work shifts

4.3.3 Result of MHDSSM in a month

Based on the DHDSSM, the number of drivers required in a day is obtained. To obtain the minimum number of drivers required in a month, we have to know the number of drivers required each day. Since the data of daily online orders is necessary to generate the daily number of drivers required, we applied the scenario-based sample method to predict future online orders. The distribution of online orders in the same scenario has a certain correlation, hence in the context of the O2O retail industry, we can roughly divide the scenario into weekdays and weekends. The detailed prediction method can be found in this paper (Qian et al., 2022). However, due to the lack of complete online orders in each scenario, we assumed that the number of drivers required on the weekend day is more than 20% of that workday. After that, MHDSSM is applied to obtain the minimum number of drivers required in a month. The minimum driver sizing for a month obtained by the Gurobi solver is 45, and the decision information is shown in Table 6. In Table 6, the first column represents the sequence of drivers. The second column represents the type of driver. The third and fourth columns indicate the number of working days (WD) and days off (DO) in a month for each driver. The fifth column shows the number of days off scheduled on weekends (DOW). The last row indicates the number of peak times (*NPT*) served by the drivers. To make it easier for managers to use, the best decision information is presented in the form of a Gantt chart in Fig. 9. Each driver has a clear view of his workday in the Gantt chart.

Table 6
The optimal solution of MHDSSM

Seq	*Type*	*WD*	*DO*	*DOW*	*NPT*
1	in-house	19	9	5	37
2	in-house	20	8	6	38
3	in-house	20	8	6	38
4	in-house	18	10	5	36
5	in-house	20	8	6	36
6	in-house	19	9	5	37
7	in-house	20	8	6	39
8	in-house	20	8	6	36
…	…	…	…	…	…
33	outsourcing	19	9	6	27
34	outsourcing	20	8	6	31
35	outsourcing	20	8	6	31
36	outsourcing	19	9	6	28
37	outsourcing	19	9	6	30
38	outsourcing	20	8	6	31
39	outsourcing	19	9	6	27
40	outsourcing	20	8	6	30
41	outsourcing	20	8	6	28
42	outsourcing	20	8	6	31
43	outsourcing	20	8	6	34
44	outsourcing	20	8	6	28
45	outsourcing	19	9	6	19

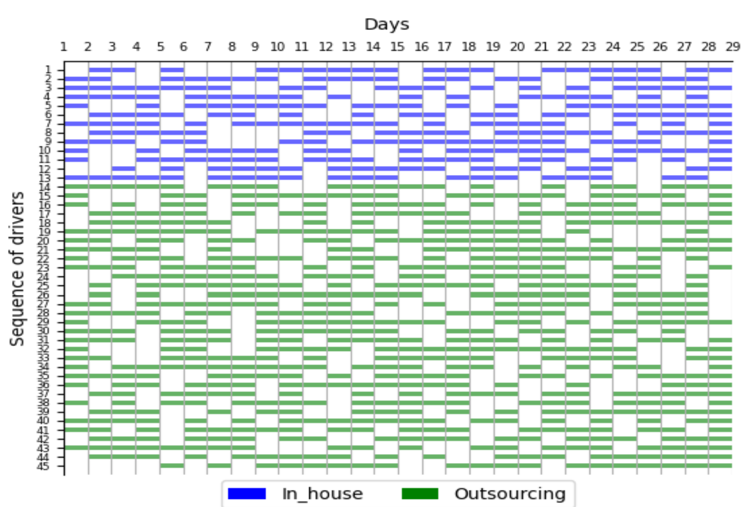


Fig. 9. Gantt chart for drivers' workday.

5. Discussion

5.1 Impact of the relaxation of rest break

To analyze the impact of rest breaks on the monthly driver sizing, we focus on these important parameters *b*, and *uſ*. The other parameters are set the same as those set in Section 5.3.3. The experimental results are recorded in Table 9.

In Table 7, the first column records the maximum working day in a month, which is set to 20, 22, and 24 days. The second column records the maximum number of days worked on weekends per month. The last column records the number of drivers required monthly obtained by MHDSSM.

Table 7
Impact of the relaxation of rest breaks

b	u'	Driver required in the month		
		In-house	*Outsourcing*	*Total*
20	2	36	96	132
	4	18	48	66
	6	13	32	45
	8	13	32	45
22	2	36	96	132
	4	18	48	66
	6	12	32	44
	8	11	27	38
24	2	36	96	132
	4	18	48	66
	6	12	32	45
	8	11	27	38

As the parameter u' increases, the number of drivers decreases significantly. If the maximum working day in a month does not exceed 20 days, and regardless of whether days off are scheduled on weekends, O2O retailers just need to employ 45 drivers. If the maximum number of working days per month is set to 22 days, and regardless of whether days off are scheduled on weekends, O2O retailers just need 38 drivers. In this case, if drivers agree to set the maximum number of working days per month at 22 days, and regardless of whether days off are scheduled on weekends, the O2O retailer can hire the minimum number of drivers to fulfill online orders.

5.2 Impact of the relaxation of shift length

To analyze the impact of shift length on daily employment costs, we first set the salary composition for different types of employees during overtime work. Outsourcing divers earn 20 CNY per hour during overtime. In-house drivers earn 4 CNY for delivering an order during overtime. After that, we analyze the impact of shift length on the daily employment costs by solving the DHDSSM and MHDSSM. The experimental results are recorded in Table 8. In Table 8, the shift length of in-house and outsourcing drivers in hours are set in the first and second columns respectively. The third columns are the minimum employment costs in CNY obtained by DHDSSM. The fourth and fifth columns represent the decision information of the optimal solution. The decision information contains the number of in-house drivers, outsourcing drivers, and the total number of drivers.

Table 8
Impact of the relaxation of shift length.

l_0	l_1	*EC*	Daily demand for drivers			Monthly demand for drivers		
			In-house	*Outsourcing*	*Total*	*In-house*	*Outsourcing*	*Total*
8	8	4327	9	22	31	13	32	45
	9	4448	13	17	30	19	25	44
	10	4227	11	16	27	16	23	39
9	8	4307	8	22	30	12	34	46
	9	4448	13	17	30	20	25	45
	10	4227	11	16	27	16	23	39
10	8	4253	6	22	28	9	32	41
	9	4328	10	17	27	15	25	40
	10	4227	11	16	27	16	23	39

In this case, if O2O retailers maintain in-house drivers' shift lengths at 8 hours and extend the shift length of outsourcing drivers to 10 hours, they can effectively reduce employment costs by 100 CNY. If O2O retailers maintain outsourcing drivers' shift lengths at 8 hours and extend the shift length of in-house drivers to 10 hours, they can reduce employment costs by 74 CNY. Therefore, extending the shift length of outsourcing drivers to 10 hours can effectively reduce employment costs for O2O retailers.

5.3 Impact of the relaxation of committed delivery time

(1) Committed delivery time's impact on the distribution of online orders

In this test, the committed delivery time is set to [15, 20, 25, 30, 35, 40, 45] in minutes. And then, we divide the 2380 online orders into the corresponding time intervals based on the completed time of these online orders.

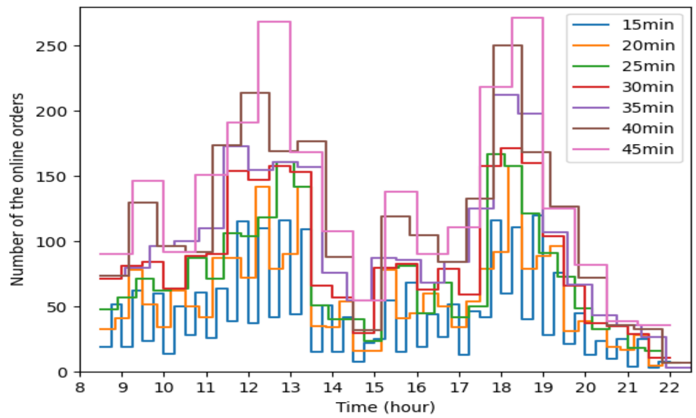


Fig. 10. The distribution character of online orders with the different committed delivery time.

As shown in Fig. 10, the line graph shows the number of online orders in each time interval. It is obvious that the committed delivery time is larger, and the number of online orders fluctuates more sharply during lunch and dinner time. When the committed time is set to 45 minutes, online orders exceed 250 in the peak period and below 50 in the low period. However, when the committed delivery time is set to 15 minutes, online orders are 120 in the peak period and below 50 in the low period.

(2) Committed delivery time's impact on daily transportation and employment costs

To analyze the impact of the committed delivery time on the daily transportation cost, the ALNS is applied to solve the MPVRPD model with different committed delivery times. The experimental results are shown in Fig. 11.

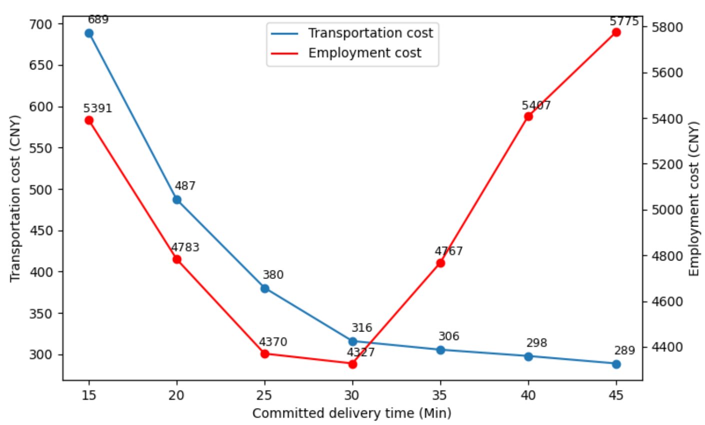


Fig. 11. The impact of the committed delivery time on transportation and employment costs.

For the transportation cost, we would draw the conclusion that the longer the committed delivery time set by the O2O retailer, the lower the transportation costs. When the committed delivery time is set to 45 minutes, the transportation cost is only about 289 CNY, which is just half of the transportation cost when the committed delivery time is set to 15 minutes. However, when we revisit the issue from the customer's perspective, customers' behaviors would change when setting a longer committed delivery time. Therefore, increasing the committed delivery time could reduce transportation costs, but there may be a risk of losing customers.

For the employment cost, when the committed delivery time increases from 15 to 30 minutes, the employment cost decreases significantly. When the committed delivery time exceeds 30 minutes, the employment cost rises dramatically. The reason is that when the promised delivery time is shorter, the number of orders delivered by the driver is smaller to avoid overruns. If the committed delivery time is set too long, the number of online orders will be too huge. Although the capacity of delivery tools would be utilized sufficiently, a large number of drivers are still needed to fulfill these huge orders. In this case, if the committed delivery time is set to 30 minutes, which could obtain the minimum employment cost with 4327 CNY.

(3) Committed delivery time's impact on daily total costs

Fig. 12 illustrates the impact of the committed delivery time on the total cost. It is obvious that the share of transportation costs is low in the total cost, and that transportation costs decrease as the committed delivery time increases. Employment cost is a major component of total cost approaching 90%. In this case, if the committed delivery time is set to 30 minutes, the minimum total cost is 4643 CNY.

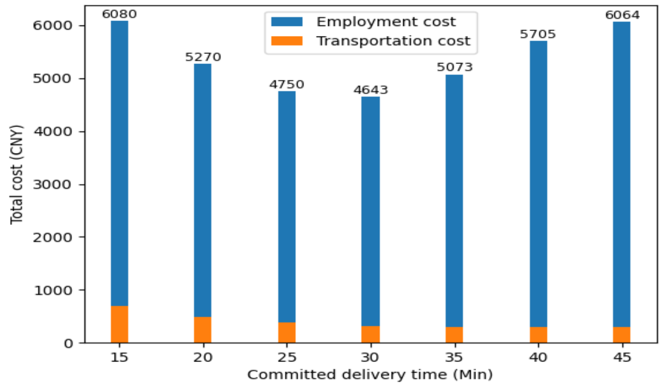


Fig. 12. The impact of the committed delivery time on the daily total costs.

6. Conclusion

In this paper, we propose an integrated optimization methodology to minimize transportation costs and employment costs for O2O home delivery services. We propose an innovative integrated model that integrates multiple decision processes in HDS, including vehicle routing, drive sizing, and scheduling, and considers the dynamic characteristics of online orders and the heterogeneity of the workforce in O2O retail. To solve the integrated model, we developed an efficient adaptive large neighborhood search (ALNS) and branch-and-cut algorithms. A case study of an O2O retailer in China is explored to assess the effectiveness of our proposed model and algorithm.

More importantly, some valuable managerial implications are proposed in the sensitivity analysis, which we summarize below.

(1) To minimize the monthly driver sizing, it is an effective measure to not ensure that drivers' days off fall on weekends when making scheduling plans. In addition, it is also effective to extend the maximum number of working days per month, but it is not certain that every driver will agree.

(2) To minimize the employment costs, it is a valuable attempt to extend the shift length of drivers. Extending the shift length of drivers contributes to reducing the number of drivers required each day. In the case study, even though we offer higher wages during overtime, it still helps us reduce employment costs. It has to be stressed that the effectiveness of this method depends on the salary structure of the driver.

(3) To minimize transportation costs, it is an effective measure to extend the committed delivery time. However, when the committed delivery time is extended beyond 30 minutes, the employment cost rises dramatically. How to set an appropriate committed delivery time is the key.

(4) To minimize the total costs, setting an appropriate committed delivery time is an effective measure. Based on our proposed method can help O2O retailers find the appropriate committed delivery time for their company. In this case, if the committed delivery time is set to 30 minutes, the minimum total costs 4643 CNY.

Although this paper proposes an effective integrated optimization model to help O2O retailers minimize the total costs of HDS, there are still some limitations. More varieties of salary structures should be considered in future studies, which have a stronger impact on employment costs. Although the transportation cost is a relatively small part of the total cost, developing more efficient heuristic algorithms to obtain low-cost path planning solutions is still an effective way to reduce transportation costs. More heterogeneous features affecting the quality of delivery services should be considered.

References

Ahuja, R. K., Ergun, Ö., Orlin, J. B., & Punnen, A. P. (2002). A survey of very large-scale neighborhood search techniques. *Discrete Applied Mathematics, 123*(1), 75-102. doi:https://doi.org/10.1016/S0166-218X(01)00338-9
Álvarez, E., Ferrer, J.-C., Muñoz, J. C., & Henao, C. A. (2020). Efficient shift scheduling with multiple breaks for full-time employees: A retail industry case. *Computers & Industrial Engineering, 150*, 106884.

doi:https://doi.org/10.1016/j.cie.2020.106884

Baker Kenneth, R. (1976). Workforce Allocation in Cyclical Scheduling Problems: A Survey. *Journal of the Operational Research Society, 27*(1), 155-167.

Bürgy, R., Michon-Lacaze, H., & Desaulniers, G. (2019). Employee scheduling with short demand perturbations and extensible shifts. *Omega, 89*, 177-192. doi:https://doi.org/10.1016/j.omega.2018.10.009

Chen, J., Fan, T., Gu, Q., & Pan, F. (2022). Emerging technology-based online scheduling for instant delivery in the O2O retail era. *Electronic Commerce Research and Applications, 51*, 101115. doi:https://doi.org/10.1016/j.elerap.2021.101115

Christiaens, J., & Vanden Berghe, G. (2020). Slack Induction by String Removals for Vehicle Routing Problems. *Transportation Science, 54*(2), 417-433. doi:10.1287/trsc.2019.0914

Dayarian, I., Crainic, T. G., Gendreau, M., & Rei, W. (2015). A branch-and-price approach for a multi-period vehicle routing problem. *Computers & Operations Research, 55*, 167-184. doi:https://doi.org/10.1016/j.cor.2014.06.004

Defraeye, M., & Van Nieuwenhuyse, I. (2016). Staffing and scheduling under nonstationary demand for service: A literature review. *Omega, 58*, 4-25. doi:https://doi.org/10.1016/j.omega.2015.04.002

Desaulniers, G., Pecin, D., & Contardo, C. (2019). Selective pricing in branch-price-and-cut algorithms for vehicle routing. *EURO Journal on Transportation and Logistics, 8*(2), 147-168. doi:https://doi.org/10.1007/s13676-017-0112-9

Elshaer, R., & Awad, H. (2020). A taxonomic review of metaheuristic algorithms for solving the vehicle routing problem and its variants. *Computers & Industrial Engineering, 140*, 106242. doi:https://doi.org/10.1016/j.cie.2019.106242

Erhard, M., Schoenfelder, J., Fügener, A., & Brunner, J. O. (2018). State of the art in physician scheduling. *European Journal of Operational Research, 265*(1), 1-18. doi:https://doi.org/10.1016/j.ejor.2017.06.037

Ernst, A. T., Jiang, H., Krishnamoorthy, M., & Sier, D. (2004). Staff scheduling and rostering: A review of applications, methods and models. *European Journal of Operational Research, 153*(1), 3-27. doi:https://doi.org/10.1016/S0377-2217(03)00095-X

Fang, W., Guan, Z., Yue, L., Zhang, Z., & Wang, H. (2022). Heterogeneous-vehicle distribution logistics planning for assembly line station materials with multiple time windows and multiple visits. *International Journal of Industrial Engineering Computations, 13*(4), 473-490. doi:10.5267/j.ijiec.2022.8.002

Fichera, S., Costa, A., & Cappadonna, F. A. (2017). Heterogeneous workers with learning ability assignment in a cellular manufacturing system. *International Journal of Industrial Engineering Computations, 8*(4), 427-440. doi:10.5267/j.ijiec.2017.3.005

Guastaroba, G., Côté, J. F., & Coelho, L. C. (2021). The Multi-Period Workforce Scheduling and Routing Problem. *Omega (United Kingdom), 102*. doi:10.1016/j.omega.2020.102302

He, B., Mirchandani, P., Shen, Q., & Yang, G. (2021). How should local Brick-and-Mortar retailers offer delivery service in a pandemic World? Self-building Vs. O2O platform. *Transportation Research Part E: Logistics and Transportation Review, 154*, 102457. doi:https://doi.org/10.1016/j.tre.2021.102457

Hema. (2021). www.freshhema.com.

Hendry, D. F., & Pretis, F. (2022). Analysing differences between scenarios. *International Journal of Forecasting*. doi:https://doi.org/10.1016/j.ijforecast.2022.02.004

Keskin, M., Laporte, G., & Çatay, B. (2019). Electric Vehicle Routing Problem with Time-Dependent Waiting Times at Recharging Stations. *Computers and Operations Research, 107*, 77-94. doi:10.1016/j.cor.2019.02.014

Lam, E., Desaulniers, G., & Stuckey, P. J. (2022). Branch-and-cut-and-price for the Electric Vehicle Routing Problem with Time Windows, Piecewise-Linear Recharging and Capacitated Recharging Stations. *Computers & Operations Research*, 105870. doi:https://doi.org/10.1016/j.cor.2022.105870

Larrain, H., Coelho, L. C., Archetti, C., & Speranza, M. G. (2019). Exact solution methods for the multi-period vehicle routing problem with due dates. *Computers & Operations Research, 110*, 148-158. doi:https://doi.org/10.1016/j.cor.2019.05.026

Lee, T.-S., & Loong, Y.-T. (2019). A review of scheduling problem and resolution methods in flexible flow shop. *International Journal of Industrial Engineering Computations, 10*(1), 67-88. doi:10.5267/j.ijiec.2018.4.001

Lespay, H., & Suchan, K. (2022). Territory Design for the Multi-Period Vehicle Routing Problem with Time Windows. *Computers & Operations Research, 145*, 105866. doi:https://doi.org/10.1016/j.cor.2022.105866

Liu, S., Sadowska, A., & De Schutter, B. (2022). A scenario-based distributed model predictive control approach for freeway networks. *Transportation Research Part C: Emerging Technologies, 136*, 103261. doi:https://doi.org/10.1016/j.trc.2021.103261

Mani, V., Kesavan, S., & Swaminathan, J. M. (2015). Estimating the Impact of Understaffing on Sales and Profitability in Retail Stores. *Production and Operations Management, 24*(2), 201-218. doi:https://doi.org/10.1111/poms.12237

Maria Gonzalez-Neira, E., Montoya-Torresc, J. R., & Barrera, D. (2017). Flow-shop scheduling problem under uncertainties: Review and trends. *INTERNATIONAL JOURNAL OF INDUSTRIAL ENGINEERING COMPUTATIONS, 8*(4), 399-426. doi:10.5267/j.ijiec.2017.2.001

Mou, S., & Robb, D. J. (2019). Real-Time Labour Allocation in grocery stores: A simulation-based approach. *Decision Support Systems, 124*, 113095. doi:https://doi.org/10.1016/j.dss.2019.113095

Nagata, Y., & Bräysy, O. (2009). A powerful route minimization heuristic for the vehicle routing problem with time windows. *Operations Research Letters, 37*(5), 333-338. doi:10.1016/j.orl.2009.04.006

Nagata, Y., Bräysy, O., & Dullaert, W. (2010). A penalty-based edge assembly memetic algorithm for the vehicle routing problem with time windows. *Computers & Operations Research, 37*(4), 724-737. doi:10.1016/j.cor.2009.06.022

Ojeda Rios, B. H., Xavier, E. C., Miyazawa, F. K., Amorim, P., Curcio, E., & Santos, M. J. (2021). Recent dynamic vehicle routing problems: A survey. *Computers & Industrial Engineering, 160*, 107604. doi:https://doi.org/10.1016/j.cie.2021.107604

Ojstersek, R., Brezocnik, M., & Buchmeister, B. (2020). Multi-objective optimization of production scheduling with evolutionary computation: A review. *International Journal of Industrial Engineering Computations, 11*(3), 359-376. doi:10.5267/j.ijiec.2020.1.003

Pereira, D. L., Alves, J. C., & Moreira, M. C. d. O. (2020). A multiperiod workforce scheduling and routing problem with dependent tasks. *Computers & Operations Research, 118*, 104930. doi:https://doi.org/10.1016/j.cor.2020.104930

Pillac, V., Gendreau, M., Guéret, C., & Medaglia, A. L. (2013). A review of dynamic vehicle routing problems. *European Journal of Operational Research, 225*(1), 1-11. doi:https://doi.org/10.1016/j.ejor.2012.08.015

Pisinger, D., & Ropke, S. (2007). A general heuristic for vehicle routing problems. *Computers & Operations Research, 34*(8), 2403-2435. doi:https://doi.org/10.1016/j.cor.2005.09.012

Punyakum, V., Sethanan, K., Nitisiri, K., Pitakaso, R., & Gen, M. (2022). Hybrid differential evolution and particle swarm optimization for Multi-visit and Multi-period workforce scheduling and routing problems. *Computers and Electronics in Agriculture, 197*, 106929. doi:https://doi.org/10.1016/j.compag.2022.106929

Qian, H., Guo, H., Sun, B., & Wang, Y. (2022). Integrated inventory and transportation management with stochastic demands: A scenario-based economic model predictive control approach. *Expert Systems with Applications, 202*, 117156. doi:https://doi.org/10.1016/j.eswa.2022.117156

Ropke, S., & Pisinger, D. (2006). An adaptive large neighborhood search heuristic for the pickup and delivery problem with time windows. *Transportation Science, 40*(4), 455-472. doi:10.1287/trsc.1050.0135

Ropke, S., & Pisinger, D. (2006). A unified heuristic for a large class of Vehicle Routing Problems with Backhauls. *European Journal of Operational Research, 171*(3), 750-775. doi:https://doi.org/10.1016/j.ejor.2004.09.004

Shaw, P. (1998). Using Constraint Programming and Local Search Methods to Solve Vehicle Routing Problems. .*Lecture Notes in Computer Science, 1520*, 417–430.

Soeffker, N., Ulmer, M. W., & Mattfeld, D. C. (2022). Stochastic dynamic vehicle routing in the light of prescriptive analytics: A review. *European Journal of Operational Research, 298*(3), 801-820. doi:https://doi.org/10.1016/j.ejor.2021.07.014

Toth, N., & Kulcsar, G. (2021). New models and algorithms to solve integrated problems of production planning and control taking into account worker skills in flexible manufacturing systems. *International Journal of Industrial Engineering Computations, 12*(4), 381-400. doi:10.5267/j.ijiec.2021.5.004

Turkeš, R., Sörensen, K., & Hvattum, L. M. (2021). Meta-analysis of metaheuristics: Quantifying the effect of adaptiveness in adaptive large neighborhood search. *European Journal of Operational Research, 292*(2), 423-442. doi:10.1016/j.ejor.2020.10.045

Wang, Y., Li, Q., Guan, X., Xu, M., Liu, Y., & Wang, H. (2021). Two-echelon collaborative multi-depot multi-period vehicle routing problem. *Expert Systems with Applications, 167*, 114201. doi:https://doi.org/10.1016/j.eswa.2020.114201

Windras Mara, S. T., Norcahyo, R., Jodiawan, P., Lusiantoro, L., & Rifai, A. P. (2022). A survey of adaptive large neighborhood search algorithms and applications. *Computers & Operations Research, 146*, 105903. doi:https://doi.org/10.1016/j.cor.2022.105903

Xu, L. N., Wang, Z. Y., Chen, X. D., & Lin, Z. W. (2022). Multi-Parking Lot and Shelter Heterogeneous Vehicle Routing Problem With Split Pickup Under Emergencies. *Ieee Access, 10*, 36073-36090. doi:10.1109/ACCESS.2022.3163715

Zamorano, E., & Stolletz, R. (2017). Branch-and-price approaches for the Multiperiod Technician Routing and Scheduling Problem. *European Journal of Operational Research, 257*(1), 55-68. doi:https://doi.org/10.1016/j.ejor.2016.06.058