Integrated Scheduling of Zone Picking and Vehicle Routing with Time Windows in the Front Warehouse Mode

CHEN Wei, ZHANG Yu-lin

(School of Economics and Management, Southeast University, Nanjing, Jiangsu, 211189, China)

Abstract: Recently, joint scheduling of order picking and delivery decisions has been valued in many businesses. Such coordination plays a critical role in the Front Warehouse Mode (FWM) where customers are able to order the ready-to-use goods to be delivered to their home in a short time. It is challenging to achieve timely order fulfillment economically due to the fragmented requirements. To handle the challenge, an integrated scheduling approach to link order picking and delivery closely is required. Hence, in this study, we focus on the integrated scheduling of zone picking and vehicle routing with time windows in the FWM, where the order picking is performed with a zone picking policy and key decisions on order batching, order sequencing, and vehicle routing are made. Besides, the factors such as fulfillment deadlines, convey time between the picking zones, and variable delivery speeds are considered to better plan the integrated problem. To help retailers optimally fulfill customers' orders, we first present a mixed integer programming formulation and then develop a Two-stage Iterated Search (TIS) algorithm to minimize the operation cost and the overdue penalty cost. With extensive numerical experiments, we demonstrate the effectiveness and efficiency of the proposed approach by comparing its performance with a leading commercial solver and a two-stage heuristic algorithm (H-2). To practically validate the application of our framework, we compare the results produced by TIS with that of the Traditional Sequential Scheduling (TSS) approach and analyze several key parameters. The results show that the TIS algorithm performs better than the traditional approach in different evaluation indicators, and the time allocation in two stages as well as the caution intensity do affect the fulfillment expense.

Keywords: the front warehouse; integrated scheduling; zone picking; vehicle routing; two-stage iterated search algorithm

Corresponding author.

E-mail addresses:zhangyl@seu.edu.cn (Y.L. Zhang),eliwchen@foxmail.com (W. Chen)

1. Introduction

In recent years, China's online retail sales have experienced a rapid expansion and the proportion of online retail sales in the total social retail sales is increasing year by year. The e-commerce sales have reached CNY￥4.816 trillion, accounting for 24.7% of total social retail sales in the first half of 2019 while the proportion was 23.6% in 2018(IIMediaResearch, 2019). Offline retail sales still make up about 75 percent of total retail sales in China, despite the rapid shift toward online shopping. Hence, many physical stores tend to add online sales channels to drive sales(Alawneh & Zhang, 2018). More and more small shops are being turned into order-and-delivery stations for e-commerce, and internet giants like Alibaba and Tencent have worked aggressively to bring physical retail into the digital commerce space(CGAP, 2019). For consumers, with the ubiquity of smartphones and the trend of omnichannel commerce, ordering ready-to-use products online and retrieving goods with instant delivery services are woven into the fabric of their daily lives. The transaction size of China's instant delivery market reached CNY￥79.52 billion in the third quarter of 2019, and the growth rate of instant delivery users decreased from 3.9% in 2018 to 0.9%(BigData-Research, 2019), implying that the market competition is intensifying. This requires retailers to be able to offer a reliable, consistent and timely response to demands to gain competitive advantages.

In such a context, a business model called the front warehouse mode (FWM) is currently popular with middle-class consumers, such as Alibaba’s freshhema (Hema, 2019), missfresh (Missfresh, 2019) and JD’s 7fresh (JD, 2019). They all provide a free and timely delivery service for online customers around order fulfillment centers. Online orders are picked in the front warehouse and delivered by vehicles to customers within a limited time. The order picking process and instant delivery should be taken into account together when fulfilling orders. Compared with B2C e-commerce, the order fulfillment center of the FWM is closer to consumers. Consequently, the FWM can significantly not only reduce logistics operation costs but also shorten order fulfillment time and improve the consumer experience. Nevertheless, it is inevitable to pour significant human and material resources into the FWM. On top of all that the difficulty and costs of order fulfillment have increased owing to the small lot-size, high frequency, time-sensitive, fragmented orders as well as customers' further attention to service quality such as timeliness and reliability. It is pointless to provide an timely delivery service if it causes higher costs due to excessive use of warehouse and distribution resources. Thus, how to perform the order fulfillment at the lowest cost within the promised time is a severe challenge for retailers if they want to remain competitive.

Traditional Sequential Scheduling (TSS) approach is commonly used when performing the order fulfillment where order picking operation is conducted first without considering the delivery decision, and then the vehicle routes planning. It is difficult to optimize the overall cost and fulfillment time when adopting TSS which only achieves the best optimization of each stage.

Integrated scheduling of order picking and delivery can generally optimize the target value by 5% ~ 20%(Moons, Ramaekers, Caris, & Arda, 2017). Thus, it is quite necessary to research order fulfillment problems from an integration perspective. Some researchers studied the integrated order picking and delivery problem (Moons, Ramaekers, Caris, & Arda, 2018) but the order picking is similar to single or parallel machine scheduling which does not suit the FWM. It is likely to adopt complex picking strategies such as batch picking, synchronized or progressive zoning with multiple pickers, etc. For delivery methods, individual and immediate delivery (idd) method(Zhang, Liu, Tang, & Li, 2019a) and shipping with fixed delivery departure dates (fdep) method (van Gils, Caris, Ramaekers, & Braekers, 2019) are often studied. Being a practical method, the batch delivery with routing (routing) method where vehicle routing is a part of the decision does not gain much attention.

In this study, we address the integrated scheduling of zone picking and vehicle routing with time windows (ISZPVRTW) where the S-shape picking route policy and sort-while-pick policy (Parikh & Meller, 2008) are adopted and order picking is performed in a progressive zoning system. Moreover, the convey time between the picking zones, setup time and order packing time are considered to better plan the integrated problem. These factors are often ignored in most situations while they have great impacts on the order fulfillment performance in the FWM. Our research efforts concern not only the economic aspects of the integrated scheduling of order picking and delivery but also the service level. There is a certain penalty cost for overdue orders, and the cost of the overdue penalty is included in the objective function. We aim to minimize overdue penalty costs such as the violations of the imposed delivery time windows. Furthermore, security problems have risen to the second largest issue of instant delivery services, and 42.7% of users are worried about delivery security issues such as loss or damage in goods(IIMediaResearch, 2019b). Hence, the evaluation criteria of service should be not only the timeliness but also safety issues. For this reason, we assume that the driving speed will change with delivery statuses (departure, in-transit, and return), instead of remaining constant to reduce the risk of cargo damage and improve customer service.

In this paper, we formulate the ISZPVRTW problem as a mixed-integer programming model to minimize the order fulfillment cost including order picking costs, delivery costs, and overdue penalty costs. The following decisions need to be made. (1) the batching of orders; (2) the picking sequence of batches; (3) the assignment of orders to vehicles; (4) the time when vehicles should start to deliver and the visiting sequence of customers. The remainder of the paper is organized as follows. We first review some related literature on the order picking, and the integrated scheduling of order picking and delivery in Section 2 to derive the research gap and emphasizes the main contributions of this paper. The ISZPVRTW problem is formulated and we analyze the model in Section 3. The two-stage iterated search (TIS) algorithm is designed in Section 4. A series of experiments are conducted and the results are discussed in Section 5. The conclusion and directions for future research are presented in Section 6.

2. Literature review

2.1. Order picking planning

Order picking is the process of retrieving products from storage (or buffer areas) in response to a specific customer request (De KosterLe-Duc & Roodbergen, 2007). The cost of order picking is estimated to be as much as 55% of the total warehouse operating expense(M, 2012)， whcih is the most labor-intensive operation in warehouses with manual systems. For these reasons, order picking planning should be taken into account as the highestpriority problem for productivity improvements. The layout of the warehouse, the storage strategy, the routing policy, the zoning method, and the batching policy are commonly adopted in the optimization of order picking (Yu & de Koster, 2009). Batching and zoning are two important factors that influence the order picking efficiency when given other strategies.

Order batching is the process of grouping orders together and jointly releasing them for picking to reduce the travel time or distance. It is proved that offline order batching is an NP-hard problem when the number of batches is more than 2 (GADEMANN & VELDE, 2005). Hence, large-scale order batching problems are difficult to be solved efficiently with exact algorithms. Various heuristic algorithms are used currently. The first type is simple heuristic algorithms, including the seed algorithm(Ho & Tseng, 2006) and the saving algorithm (De KosterVan der Poort & Wolters, 1999). There are two phases in the seed algorithm that are order selection and order consolidation. The saving algorithm is essentially a special seed algorithm, and its order selection refers to the picking time or distance saved after the two orders are combined. The second type is intelligent heuristic algorithms emerging with the development of computer techniques. (PanShih & Wu, 2015) develop a novel order batching method with the group genetic algorithm (GGA) to balance the workload of each picking zone and minimize the number of batches in pick-and-pass order picking systems. A hybrid algorithm combining particle swarm optimization (PSO) and ant colony optimization (ACO) was proposed to solve the joint planning problem of order batching and picking routes (Cheng, Chen, Chen, & Jung-Woon Yoo, 2015). The adaptive large-scale neighborhood search (ALNS) algorithm and the tabu search (TS) algorithm are jointly used to solve the large-scale order batching problem and the effectiveness and robustness of the hybrid algorithm are verified by numerical experiments (ŽuljKramer & Schneider, 2018).

There are some products with a high repurchase rate and low volumes, such as food, medicine and household daily necessities stored in the front warehouse. The policy to improve the efficiency of picking such goods is not only batching but also zone picking (Pan et al., 2015). Zoning is an approach that is dividing the whole picking area into several smaller areas (zones) and assigning order pickers to pick requested items within the zone (Yu et al., 2009). It can not only improve the familiarity of storage locations for pickers but also reduce the congestion of picking aisles, thereby improving the efficiency of order picking. The analysis of zoning is classified into synchronized zoning, where all zone pickers work on the same batch of orders at the same time, and progressive zoning, where each batch of orders is processed at zone at a time (Yu et al., 2009). The zoning policy is widely studied through mathematical modeling, simulation, etc (de Koster, 1994; MelaciniPerotti & Tumino, 2011; Petersen, 2002). However, the scheduling of order picking in zoning systems is valued in practice, which is formulated as production scheduling problems in flow shop or flexible job-shop (FJS) systems. (Liou & Hsieh, 2015) study the multi-stage flow shop scheduling problem with sequential dependence of transportation and machine setup time. Analogously, the multi-stage product assembly problem with transportation and material delivery time in a flow shop system is studied (Sheikh, Komaki, Kayvanfar, & Teymourian, 2019). These researches above are only about the design and application of a policy that is batching or zoning. Whereas, the multi-strategy and multi-stage collaborative optimization has won many concerns in recent years. (van Gils, Ramaekers, Braekers, Depaire, & Caris, 2018) verify the effectiveness of multi-strategy collaboration on the efficiency optimization in order picking systems with simulation experiments. (Yu & De Koster, 2008) establish a queueing network model for pick-and-pass systems with a progressive zone picking policy and the effect of order batching strategy on picking efficiency is analyzed. The integrated scheduling problem of order picking, order assignment and order sequencing is also studied (ZhangWang & Huang, 2016a)(Henn, 2015).

2.2. Integrated order picking and delivery

The integrated order picking and delivery scheduling(IOPDS) problem is quite similar to the integrated production and delivery scheduling(IPDS) problem. The researches related to the IPDS problem have been very rich(Chen, 2010; ChenHsueh & Chang, 2009; Moons et al., 2017)， but less attention is paid to the IOPDS problem. (Moons et al., 2018) compare the IOPDS problem with IPDS, and tell the similarities and differences between them. The IOPDS problem is more complex and the reasons can be listed as follows. First, the order processing time in IOPDS is usually not constant and it is influenced by operational decisions such as the storage strategy, the routing policy, the zoning method, etc. Also, the order picking speed will be affected by human factors like the fatigue of the pickers and the external environment such as the external sound and light. Furthermore, online shopping customers have stronger requirements on the timeliness of order fulfillment. As a result, it is more intractable to cope with the IOPDS problem.

The research to date has tended to focus on the IOPDS problem in the B2C e-commerce environment. (ZhangWang & Huang, 2016b) study the integrated on-line scheduling of order batching and delivery problems when the delivery operation is outsourced to a third party, which means the departure time is fixed given by the third party. In addition, the IOPDS problem with multiple zones and limited vehicle capacity is studied where the direct shipping method is used (ZhangWang & Huang, 2018). (Li, Li, Aneja, Guo, & Tian, 2019) focus on the IOPDS problem where an on-line retailer has multiple distribution centers(DC), and the integrated order allocation to DC and vehicle routing decisions are made. These scholars generally consider the order batching in the model while abstracting the DC as a picker, which is essentially a traditional single-machine scheduling problem with batching. Thus, to practically solve the more realistic problem, (Moons, Braekers, Ramaekers, Caris, & Arda, 2019) extend the IOPDS model with one picker to multiple pickers who have different working efficiency, and the centralized decision is made on order assignment, sequencing, and vehicle routing optimization. Similarly,(van Gils et al., 2019) establish an IOPDS model in pick-and-sort order picking systems where the fdep delivery method is adopted, and order batching, assignment and picking routes decisions are made together.

The studies above are mainly about the IOPDS problem with a single-picker picking method or a multi-picker parallel picking method, while the integrated scheduling of zone picking and vehicle routing is less studied. (Kaminsky, 2003) and (SoukhalOulamara & Martineau, 2005) research the complexity of the IPDP problem in a flow shop, which is similar to the IOPDS problem with a zone picking policy. The research on the IOPDS problem in the FWM is even more scarce. There is a similar business model called online-to-offline(O2O). The IOPDS problem with multi-pickers who have various learning effects in an O2O supermarket is studied(Zhang, Liu, Tang, & Li, 2019b), but the order picking operation in the model is quite complicated so delivery constraints are loose where the simple method like idd is used. In general, the researches on IPDP and IOPDS in distribution centers are comprehensive. However, they are mainly about simple integrated problems with single or parallel machines, and for the delivery characteristics, the idd, fdept and direct shipping methods are commonly used. Less attention is paid to the integrated scheduling problems in the FWM that considers zone picking policy, completion deadlines, and vehicle routing decisions in practice.

In our study, the order picking method is batch picking with the sort-while-pick policy. The related orders will be grouped into a batch and assigned to pickers. There are several bags for storing goods of orders in a batch temporarily. Pickers need to pick goods in each zone while sorting them into different bags. The capacitated vehicles, the convey time between the picking zones, setup time and order packing time are considered to better plan the integrated problem. The operation cost-based function includes the order picking cost and delivery cost which consists of driving costs and start-up costs of vehicles. The service-based function is the cost of the penalty for overdue orders. On the whole, we address the specific IOPDS problem in the FWM which differs from previous studies in the IPDP or IOPDS in DC. First, we use the batching policy to improve efficiency. Second, the picking area in the front warehouse is divided into several zones and each one is equipped with a picker. Third, when the picker starts to work, the S-shape picking routes and sort-while-pick methods are used. Fourth, vehicle routing decisions and factors like the convey time between the picking zones, setup time and order packing time, having an impact on operation performance, are taken into account and goods should be delivered to multiple customers in limited time. Finally, our research efforts concern not only the economic aspects of the IOPDS problem but also the service level. The objective is to minimize operation costs and overdue penalty costs.

3. Problem formulation

3.1. Problem definition

We consider an order picking system in the FWM that can process online orders from different customers within a three-kilometer radius. At the beginning of the planning horizon, each customer places exactly one order including various items from smartphones. After receiving the orders, the retailer is responsible for organizing the order picking and distribution processes quickly. Customers can receive their groceries within 30 minutes. However, the total scheduling cost will increase because timely delivery requires more distribution resources. Thus, how to minimize the fulfillment cost while providing a reliable and timely delivery service has been bothering retailers. We integrate the order picking and delivery scheduling and use the framework to help retailers optimize decisions. The order fulfillment process is illustrated in Fig 1, and the layout of the picking area is presented in Fig 2.

The picking area of the front warehouse is divided into multiple zones containing different products. Pickers are assigned to different zones and only required to pick items in the assigned zone. There is an order processing desk in the bottom right corner of each zone for processing information and making preparations for picking tasks. Each order contains several items and orders will be organized as several batches. The sort-while-pick policy and S-shape picking routes method are used in a zone picking system where the batch of orders is passed from one zone to the next. Zone picking systems are widely used in practice. The picker in each zone sets out from the order processing desk at first. Next, he/she roams around the zone with S-shape picking routes, filling bags with online orders, then place them on a conveyor belt to the packing desk. Finally he/she returns to the order processing desk preparing the next task. In this process, convey time, packing time, and setup time are considered. Each batch should be processed by one of the pickers, and each picker can deal with only one batch at a time. Thus, it is important to determine the picking sequence of batches.

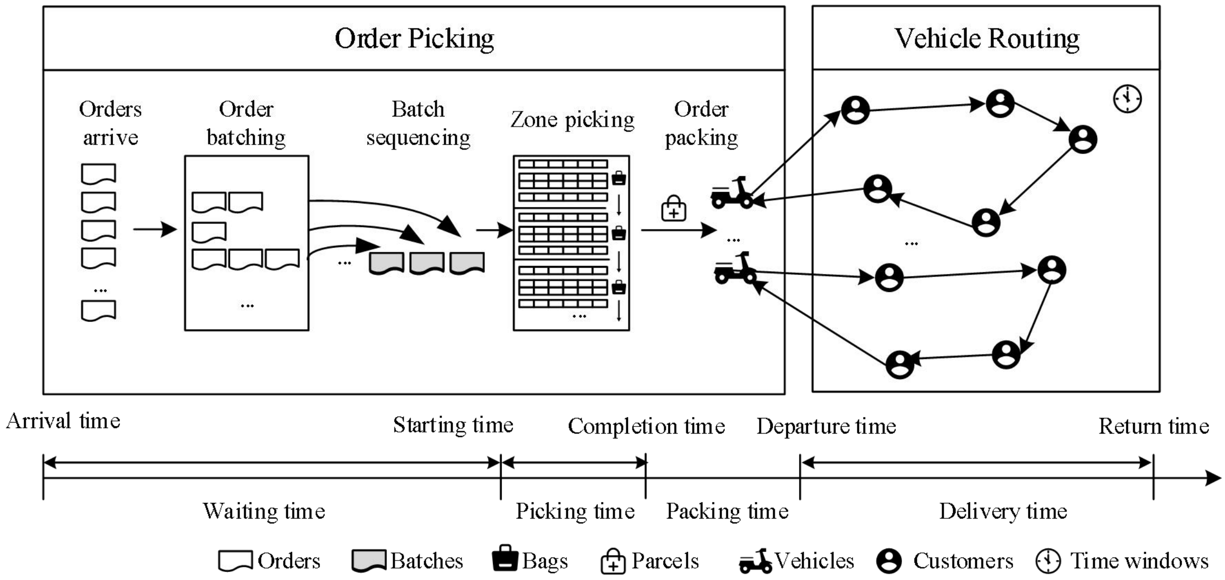


Fig.1 The integrated zone picking and delivery scheduling problem

The front warehouse is equipped with a sufficient number of delivery staff and homogeneous vehicles. Each vehicle is initially stationed in the front warehouse, and after serving the specified route, returns to the front warehouse. The driving speed varies with the delivery statuses (departure, in-transit, and return) and different statuses are linked with different caution intensity ensuring to reduce the risk of cargo damage. A soft time window constraint is imposed in the distribution phase. If the order is overdue, the penalty costs will occur. Service time is required when visiting each distribution point. Each vehicle is responsible for one batch at a time. Each order is inseparable, which means each customer cannot be visited multiple times. The delivery cost is composed of the start-up and driving costs. Our objective is to minimize order fulfillment costs, including the order picking cost, the delivery cost, and the overdue penalty.

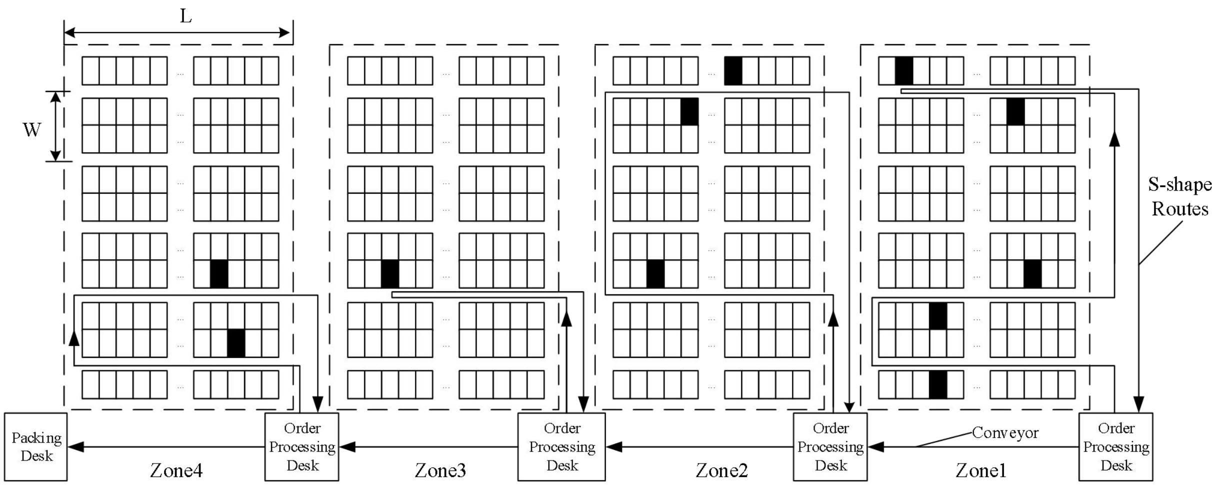


Fig.2 Layout of the order picking area

The ISZPVRTW problem studied in this paper can be described by the five-field notation (Chen, 2010), which is described in Table 1. Moreover, to balance practicality and tractability, we introduce some assumptions in the problem setting we consider: (1) Retailers have gathered information regarding each customer order before making decisions. (2) The online orders from different customers within a three-kilometer radius are accepted. In other words, the order beyond the range is not in our discussion. (3) One picker can only process one picking task, and a batch can only be handled by one picker at a time. (4) All batches need to flow through all zones; (5) S-shape picking routes policy and sort-while-pick policy(Parikh et al., 2008) are used; (6) There are sufficient capacitated vehicles which are homogeneous, and the driving speed varies with delivery statuses; (7) The order cannot be split and should be delivered to the customer completely; (8) The picking task cannot be interrupted, which means the picker should execute the next picking task immediately after completing the current picking task.

Table 1 Five-field notation

Notations	Explanations
α	F_m : progressive systems with m zones; sort-while-pick: the sorting and picking operations are conducted almost simultaneously
β	batch: orders are processed by batches; S-shape: the shape of picking routes is S
π	$V(\infty,Q)$: the number of vehicles is large enough and each vehicle can deliver at most Q items; CVRPTW: capacitated vehicle routing problem with time windows
δ	n : the number of customers is n, each customer can exactly place one order.
γ	TC : order fulfillment cost(order picking cost+delivery cost+ overdue penalty cost)

$$F_m, sort\text{-}while\text{-}pick | batch, S\text{-}shape | V(\infty,Q), CVRPTW | n | TC$$

3.2. Model setting

3.2.1. Notations

In this section, the notations used in the discussion of the ISZPVRTW problem are introduced as follows:

Sets

N : The set of orders, where $i \in N = \{1,\ldots,\overline{N}\}$

N_0 : The set of customers and the front warehouse, where $i \in N_0 = \{0\} \cup N$

B : The set of batches, where $b \in B = \{1,\ldots,\overline{B}\}$

M : The set of zones, where $m \in M = \{1,\ldots,\overline{M}\}$

V : The set of vehicles, where $v \in V = \{1,\ldots,\overline{V}\}$

Π : The set of positions, where $b \in \Pi = \{1,\ldots,\overline{B}\}$

Parameters

W : The distance between two adjacent aisles in the picking area

L : The length of an aisle

lm : The deadline for order fulfillment

q_i : The number of items of the order i

v^{travel} : The travel speed of pickers

v^{pick} : The picking speed of pickers

t^{pack} : The packing time per item

t^{setup} : The setup time per batch

t^{convey} : The convey time between two adjacent zones

v^{drive} : The driving speed of vehicles

$t^{service}$: The service time of the customer

Q : The maximal number of items in one batch

λ_i : The caution intensity of the delivery status i

f : The fixed start-up cost of each vehicle

β : Unit cost of delivery mileage

c^{op} : Unit cost of order picking

c^{od} : Unit cost of the overdue penalty

Decision variables

A_{bm} : The number of aisles traveled when picking batch b in zone m

A_{bm}^{n} : The index of the aisle nearest to initial point when picking batch b in zone m

A_{bm}^{f} : The index of the aisle farthest to initial point when picking batch b in zone m

D_{bm}^{f} : The distance from the initial point to the position of the farthest item.

d_{mb}^{pick} : The picking distance of batch b in zone m

t_{mb}^{pick} : The picking time of batch b in zone m

t_{km}^{comp} : The completion time of picking of position k in zone m

t_{v}^{depart} : The departure time of the vehicle v

d_{ig}^{trans} : The distance between the customer i and g

T_{ig} : The driving time between the customer i and g

a_i : The time of arriving at the customer i

x_{ib} : The binary variable equals 1 if the order i is assigned to batch b, and 0 otherwise

τ_b : The binary variable is equal to 1 if the batch b is not empty, and 0 otherwise

y_{bk} : The binary variable equals 1 if the batch b is picked in position k, and 0 otherwise

z_{iv} : The binary variable equals 1 if the order i is assigned to the vehicle v, and 0 otherwise

w_{igv} : The binary variable equals 1 if the vehicle v travels from the customer i to g, and 0 otherwise

3.2.2. Mathematical model

$$\text{Min} \quad \beta \sum_{v \in V} \sum_{i \in N_0} \sum_{g \in N_0} w_{igv} d_{ig}^{trans} + f \sum_{g \in N} w_{0gv} + c^{op} \sum_{v \in V} t_{v}^{depart} + c^{od} \sum_{i \in N} \max\{0, a_i - lm\} \tag{1}$$

s.t.

$$\sum_{b\in B} x_{ib} = 1, \quad \forall i \in N \tag{2}$$

$$\sum_{i\in N} x_{ib} q_i \le Q, \quad \forall b \in B \tag{3}$$

$$0 \le \sum_{b\in B} \tau_b \le \overline{B} \tag{4}$$

$$\sum_{k\in \pi} y_{bk} = 1, \quad \forall b \in B \tag{5}$$

$$\sum_{b\in B} y_{bk} = 1, \quad \forall k \in \Pi \tag{6}$$

$$t_{1,1}^{comp} \ge \sum_{b\in B} y_{b1} \cdot t_{1,b}^{pick} + t^{setup} \tag{7}$$

$$t_{1,m}^{comp} \ge t_{1,m-1}^{comp} + t^{convey} + \sum_{b\in B} y_{b1} \cdot t_{mb}^{pick}, \quad \forall m \in \left\{2,...,\overline{M}\right\} \tag{8}$$

$$t_{k,1}^{comp} \ge t_{k-1,1}^{comp} + \sum_{b\in B} y_{b\mathrm{k}} \cdot t_{1,b}^{pick}, \quad \forall k \in \left\{2,...,\overline{B}\right\} \tag{9}$$

$$t_{km}^{comp} = \max\left\{t_{k\text{-}1,m}^{comp}, t_{k,m-1}^{comp} + t^{convey}\right\} + \sum_{b\in B} y_{bk} \cdot t_{mb}^{pick}, \quad \forall k \in \left\{2,...,\overline{B}\right\}, m \in \left\{2,...,\overline{M}\right\} \tag{10}$$

$$d_{mb}^{pick} = \begin{cases} \left(A_{bm} - 1\right)\cdot W + 2D_{bm}^{f} + \left(A_{bm}^{f} - 1\right)\cdot W, & A_{bm} = 1 \\ \left(A_{bm}^{n} - 1\right)\cdot W + A_{bm}L + \left(A_{bm} - 1\right)\cdot W + \left(A_{bm}^{f} - 1\right)\cdot W, & A_{bm} = even \\ \left(A_{bm}^{n} - 1\right)\cdot W + \left(A_{bm} - 1\right)L + 2D_{bm}^{f} + \left(A_{bm} - 1\right)\cdot W + \left(A_{bm}^{f} - 1\right)\cdot W, & A_{bm} = odd \end{cases} \tag{11}$$

$$t_{mb}^{pick} = \frac{d_{mb}^{pick}}{v^{travel}} + \sum_{i\in N} \frac{x_{ib} \cdot q_i}{v^{pick}}, \quad \forall b \in B, m \in M \tag{12}$$

$$t_{km}^{comp} \ge 0, \quad \forall m \in M, k \in B \tag{13}$$

$$t_{v}^{depart} = \max_{k\in\pi, i\in N, m\in M} \left\{ x_{ib} z_{iv} y_{bk} \left(t_{km}^{comp} + t^{convey} + t^{pack} \cdot \sum_{i\in N} x_{ib} \cdot q_i \right) \right\}, \quad \forall v \in V \tag{14}$$

$$T_{ig} = \begin{cases} dis_{og} / v^{drive}\left(1 - \lambda_1\right), & \forall g \in N \\ dis_{ig} / v^{drive}\left(1 - \lambda_2\right), & \forall i, g \in N \\ dis_{i0} / v^{drive}, & \forall i \in N \end{cases} \tag{15}$$

$$t_{v}^{depart} + T_{0g} - a_g \le G \cdot \left(1 - \sum_{v\in V} w_{0gv}\right), \quad \forall g \in N \tag{16}$$

$$a_i + t^{service} + T_{ig} - a_g \le G \cdot \left(1 - \sum_{v\in V} w_{igv}\right), \quad \forall i, g \in N \tag{17}$$

$$\sum_{g \in N_0} \sum_{v \in V} w_{igv} = 1, \quad \forall i \in N \tag{18}$$

$$\sum_{i \in N_0} \sum_{v \in V} w_{igv} = 1, \quad \forall g \in N \tag{19}$$

$$\sum_{i \in N_0^-(g)} w_{igv} - \sum_{i \in N_0^-(g)} w_{giv} = 0, \quad \forall g \in N, v \in V \tag{20}$$

$$\sum_{g \in N_0} w_{igv} \le z_{iv}, \quad \forall i \in N, v \in V \tag{21}$$

$$\sum_{g \in N} w_{0gv} \le 1, \quad \forall v \in V \tag{22}$$

$$\sum_{i \in N} w_{i0v} \le 1, \quad \forall v \in V \tag{23}$$

$$\sum_{i \in N} z_{iv} q_i \le Q, \quad \forall v \in V \tag{24}$$

$$x_{ib}, y_{bk}, z_{iv}, w_{i'gv} \in \{0,1\}, \quad \forall i \in N, i' \in N_0, g \in N_0, b \in B, k \in B, v \in V \tag{25}$$

The objective (1) is to minimize the order fulfillment cost, including order picking cost, delivery cost and overdue penalty cost. The delivery cost is composed of the fixed cost of start-up and driving costs. Constraints (2) ~ (12) are constraints in the order picking stage. Constraints (2) ~ (4) are batching constraints. Equation (2) guarantees that each operation of each customer order must only be assigned to one batch, and inequality (3) restricts the maximum number of items per batch. Specifically, Constraint (4) indicates the number of batches cannot exceed the maximum. Constraints (5) ~ (13) are batch sequencing constraints. Equation (5) and (6) ensure that each batch appears in the set of positions π only once. Inequality (7) defines the expressions of the completion time of the batch which is picked firstly in zone 1. Inequality (8) denotes the completion time of the batch which is picked firstly in other zones. Inequality (9) calculates the completion time of other batches which are not processed firstly in zone 1. Equation (10) restricts operations such that one batch can only be processed in one zone at a time and one zone can only process one batch at a time. Equation (11) gives the expression of the picking distance of batch b in the zone m. Equation (12) calculates the picking time of batch b in the zone m, which is determined by the picking distance and picking volume, and picking distance is calculated by the S-shape routes policy. Constraint (13) indicates that the completion time of each batch is non-negative.

Constraints (14)~(24) indicate the constraints of the delivery part. Constraints (14)~(17) are the constraints about time. Equation (14) calculates the departure time of the vehicle v that is responsible for the batch b. Equation (15) is the expression of the driving time between the customer i and g, and the driving speed is separated into three grades according to the delivery statuses (departure, in-transit, return). Different grades correspond to different levels of caution intensity. Constraint (16) calculates the arrival time of the vehicle v from the front warehouse to the next customer, and G is a very large number. Constraint (17) specifies the arrival time of the vehicle v from the customer i to g. Equations (18) ~ (20) are the flow conservation constraints at each customer, which indicates that each customer can only be visited once; the vehicle v must leave after serving the customer g; the vehicle v only serves the customer whose order is assigned to it. Constraints (22) ~ (23) are closed-loop constraints, which means that vehicles must eventually return to the front warehouse. Constraint (24) guarantees the capacity of each vehicle is not exceeded by the total size of the orders. Constraint (25) restricts the range of variables.

3.2.3. Model analysis

In the model established in our paper, some decision variables are integers, and the objective function and some constraints are nonlinear. Therefore, the model is a mixed-integer nonlinear programming model (MINLP). Although it is easy to linearize the objective function and constraint (10), the constraint (13) is difficult to linearize, so it cannot be solved directly by a solver like Gurobi or Cplex. Besides, the number of 0-1 decision variables is $\overline{B}*\overline{N}+\overline{B}+\overline{B}*\overline{B}+\overline{N}*\overline{V}+\overline{N}*\overline{N}*\overline{V}$, which grows significantly with the data size. The calculation time is generally within an unacceptable range when the problem is on a massive scale. Hence, it is considered to decompose the model and reduce the complexity of the model to obtain a satisfactory approximate solution in reasonable calculation time.

The order fulfillment process can be subdivided into two stages: order picking and vehicle routing schedule. The optimization goal of our model is to reduce the cost of order fulfillment. The cost of overdue penalties is an important part of the order fulfillment cost. Therefore, it is critical to coordinate the two stages to reduce the overdue time for achieving our goal. There are many constraints in the delivery stage and the delivery time is the main part of the order fulfillment time, so it is difficult to optimize the overall performance if the delivery stage is not optimized firstly. In this paper, the reverse scheduling(RS) idea is proposed. We solve the capacitated vehicle routing problem with time windows (CVRPTW) in the second stage deciding the values of z_{iv} and w_{igv}. Then, the zone picking problem with due times（ZPDT） in first stage is solved. In this stage, firstly, we group the orders in the same delivery route into a batch deciding the x_{ib}, which is to unify the delivery batch and the picking batch. Secondly, we solve the batch sequencing problem with due times (BSPDT) deciding the value of y_{bk} based on the latest departure time of each vehicle obtained in the second stage. The RS process is demonstrated in Fig.3.

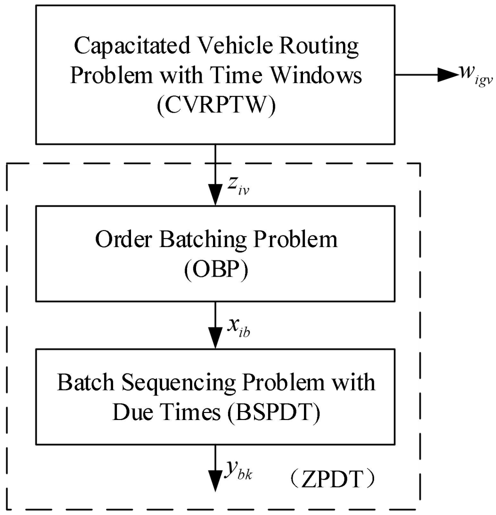


Fig.3 Reverse scheduling process

Since the results of the second stage can be used as the input of the first stage, the number of simultaneous decision variables in the model are greatly reduced, and the computing time can be shortened effectively. Thus, based on the RS idea we decompose the ISZPVRTW problem into two sub-problems: ZPDT and CVRPTW in this paper. Several properties can be found.

Property 1 The ISZPVRTW problem is NP-hard.

Proof. When time windows and capacity constraints are not considered and the number of routes generated is limited to 1, CVRPTW degenerates into the traveling salesman problem (TSP) that has proven to be an NP-hard problem(Papadimitriou, 1977). Based on the computational hierarchical relations of scheduling problems(Pinedo & Hadavi, 2016), CVRPTW also belongs to NP-hard problems. Besides, $F_2 \| \sum C_j$ has proved to be NP-hard(GareyJohnson & Sethi, 1976). Since the ISZPVRTW problem contains $F_2 \| \sum C_j$ and CVRPTW as a special case, it must be NP-hard, too.

Property 2 If the ISZPVRTW problem has an optimal solution, the optimal scheduling must meet the following conditions.

(1) Batches picked in the same zone should be processed continuously, and there must be no idle time between adjacent batches.

(2) All orders that belong to the same delivery batch need to be processed in the same picking batch.

(3) The batches must be delivered immediately when the picking operation is completed.

Proof.

(1) If there is the idle time between the batch b and the next batch b' picked in the same zone , then batch b' can always be moved forward, eliminating idle time without increasing the objective value TC.

(2) If any delivery batch v contains orders which come from different picking batches $b_1,...,b_{j-1},b_j$, then the departure time of the vehicle v can be calculated as at least $\sum\left(\sum t_{b_1m}^{pick},...,\sum t_{b_{j-1}m}^{pick},\sum t_{b_jm}^{pick}\right)+\overline{M}\cdot j\cdot t^{convey}+t^{pack}\cdot\sum q_i$. If we assign the orders in the same delivery batch to the same picking batch, the departure time of the vehicle v can be reduced to be $\max\left\{\sum t_{b_1m}^{pick},...,\sum t_{b_{j-1}m}^{pick},\sum t_{b_jm}^{pick}\right\}+\overline{M}\cdot t^{convey}+t^{pack}\cdot\sum q_i$ while not increasing the objective value TC.

(3) If the batches are not delivered immediately when the picking operation is completed, we can always let the delivery staff start working immediately, making vehicle departure time equal to the completion time of order picking without increasing the objective value TC.

4. Algorithm design

The integrated scheduling problem is usually NP-hard, which is mainly solved by meta-heuristic algorithms like tabu search (ArmentanoShiguemoto & Løkketangen, 2011), genetic algorithm(GA) (Low, Chang, Li, & Huang, 2014), etc., and it is verified to be feasible and effective. GA is an intelligent optimization algorithm based on population evolution. It has very strong hidden parallelism, global optimization capabilities, and robustness to complex problems. For these reasons, we develop a two-stage iterated search (TIS) algorithm based on GA to solve the problem in this paper. The general idea of the TIS can be explained as follows. We solve the CVRPTW model first and then the ZPDT model is solved. The results of ZPDT are summited to the fitness function, and the solution quality will be comprehensively evaluated through the fitness function value. A new round of iteration will be stared if the termination condition is not reached, otherwise, the algorithm ends. To solve the optimization problem with TIS, the following steps are required.

4.1. Chromosome coding and population initialization

We use permutation coding to code chromosomes. Each chromosome represents a vehicle scheduling scheme, which is composed of various vehicle routes. Each vehicle route can be coded as a non-repeated and out-of-order arrangement $1\sim\overline{N}$, with a leading and a trailing value of 0 indicating that each vehicle needs to depart from and return to the front warehouse. For example, "0-1-3-5-0-2-4-0" means that the current vehicle scheduling plan contains two routes. Orders 1, 3, and 5 are delivered by vehicle 1 in the order "1-3-5". Orders 2 and 4 are delivered by vehicle 2 in the order "2-4". To make the population quality follow a Pareto distribution, 4 kinds of chromosome generation approaches were used to initialize the population. Approach 1 and approach 2 are illustrated in Fig. 4.

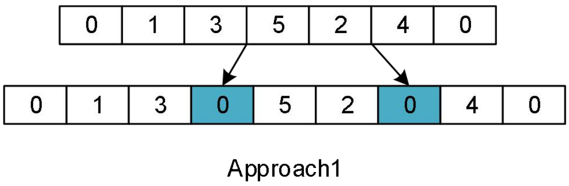


Fig.4 Approaches of chromosome generation

The first approach produces poor quality chromosomes. First, non-repeating natural numbers ranged from 1 to N are generated randomly, and number 0 is randomly inserted to non-adjacent positions to split routes. Approach 2 produces chromosomes of medium quality. Similarly, non-repeating natural numbers ranged from 1 to N are arranged randomly. Orders from left to right are assigned to the same vehicle one after another until the capacity limit is met, and number 0 will be inserted at this position. Then the assignment will be continued until all orders are allocated. Approach 3 produces better-quality chromosomes where the c-w algorithm(Clarke & Wright, 1964)for route planning is used. Approach 4 generates the best quality chromosomes where the pywrapcp and routing_enums_pb2 packages in OR-Tools are adopted to solve CVRPTW. During population initialization, the chromosome generation approaches from 1 to 4 will be selected with a probability of 25%, 45%, 15%, and 15% until the number of chromosomes generated reaches the size of the population. The solution obtained will be added to the population as the initial solution.

4.2. Fitness function design

The fitness function is also called the evaluation function. It is a criterion for distinguishing the good and bad of individuals in a group according to the objective function. We use the penalty function method to deal with model constraints. That is to say, we relax the constraints in the model, and impose penalties on chromosomes that violate the constraints. The penalty will be added to the objective function finally.

Step1: The chromosome is broken down into several routes, and one vehicle will be responsible for one batch. Then the total distance of each route, the load of each route, the delivery time of each order are calculated. Relax the delivery constraints, add the penalty term to the objective function, and calculate the objective function of CVRPTW. The vehicle departure time t_v^{depart} is initialized to 0, and the delivery cost is shown in equation (26), where d is the maximum delivery time of each batch, α_1 is the overload penalty factor, and α_2 is the delivery overdue penalty factor.

$$C_d = f\sum_{i\in N} w_{0iv} + \beta \sum_{i\in N_0}\sum_{g\in N_0}\sum_{v\in V} w_{igv} + \alpha_1 \sum_{v\in V} \max\left\{\sum_{i\in N} q_i z_{iv} - Q, 0\right\} + \alpha_2 \sum_{v\in V} \max\{a_i - d, 0\} \quad (26)$$

Step2: Group orders in each route into a batch and count the total number of items of each batch. Calculate the picking time of batch b in zone m according to equation (12). For any batch, its completion deadline for order picking can be formulated $d_b^{op} = \min\{lm - x_{ib}a_i\}$.

Step3: Batches are picked and sorted by the SPT or LDT rules. If the SPT scheduling result is better, the batch with the shortest total picking time will be picked first; otherwise, according to the LDT rules, the batch with the longest delivery time will be picked first. After determining the picking sequence of batches and the picking time of each batch, the completion time of order picking of each batch can be calculated according to the formulas (8) ~ (14), which is equal to the departure time t_v^{depart}. Penalties are imposed on batches that exceed d_b^{op}, and the penalty cost is added to the objective function updated as shown in equation (27). Besides, the fitness function is demanded to be maximum while the objective function is required to be minimum so that it is necessary to establish a mapping relation between them. The fitness function can be established as equation (28).

$$TC = C_d + c^{op} \sum_{v\in V} t_v^{depart} + c^{od} \sum_{v\in V} \max\{t_v^{depart} - z_{iv}x_{ib}d_b^{op}, 0\} \quad (27)$$

$$Fitness = 1/TC \quad (28)$$

4.3. Genetic operator design

The elite retention strategy is adopted to directly retain the most adaptable parents to the offspring, ensuring that during the evolution of the population, the best individuals that have appeared so far will not be lost and destroyed by selection, crossover and mutation operations. Individuals with high fitness are retained to the next generation population through the tournament selection operator. In terms of crossover operators, a post-crossover operator is designed for recombination between parent chromosomes. The recombination process is described in Fig. 5.

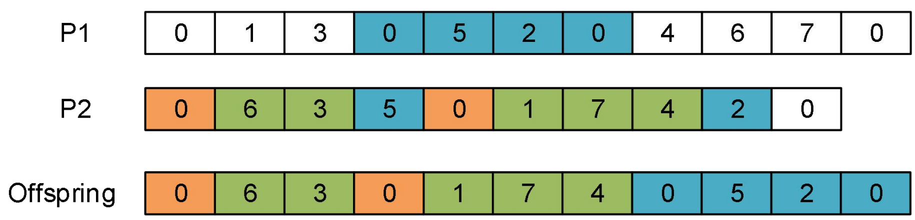


Fig.5 Post-crossover

First, select a route randomly from the parent chromosome1 (the blue nodes in Fig. 5) and place them at the end of the offspring chromosome. Then place the unvisited nodes from parent chromosome 2 (the green nodes in Fig.5) to the front of the offspring chromosome in turn. According to the position of node 0 of the parent chromosome2 (the orange nodes in Fig. 5), the node 0 is inserted into the offspring chromosome. To ensure the diversity of crossover methods and to avoid inadequate search, the two parental chromosomes will be selected randomly from the post-crossover, partially matched crossover, and order crossover based on the probability of 20%，40%，40%. In terms of mutation operators, two-point cross mutation and 2-OPT(Croes, 1958)mutation are randomly selected according to the probability of 80% and 20% respectively.

4.4. The terminal condition

The iteration limit method is used where the algorithm terminates when the genetic generation reaches the maximum number of iterations.

5. Numerical experiments

5.1. Parameters setting

To maintain the variety and practical relevance of testing data, the layout values of the shelves are set referred to parameters of the front warehouse. The layout of the front warehouse is similar to the one used in practice and literature (Zhang et al., 2019b), presented in Fig. 2. Twenty-five customer orders within 3 kilometers of the front warehouse are randomly generated. The order data is presented in Table 2. The size of items is ignored, and each item occupies one separate storage location.

The picking area is divided into 4 zones based on product attributes, containing 1200 items. There are 5 picking aisles in each picking zone, and each aisle in the zone has 60 items including the left and right sides in the aisle. The aisle length L is 15 m, and the distance between two adjacent aisles W is 2 m. The distance between zones is negligible. Four order pickers are employed in the front warehouse and each one is responsible for one zone. The order picker in one zone starts from the order processing desk, picks items while traversing the zone, and ends at the order processing desk. The picker's traveling speed v^{travel} is 80 m / min, and the picking rate v^{pick} is 15 items/min. The convey time between the adjacent two zones t^{convey} is 0.8 min, the packing time of one item t^{pack} is 0.05 min, and the picking preparation time (setup time) t^{setup} is 0.15 min. The batch and vehicle capacity Q is 12, the driving speed of the vehicle v^{drive} is 500 m / min, the delivery cost β is 5 yuan/km, and the vehicle start-up cost f is 3 yuan/vehicle.

Table2 Data of customer orders

ID	Coordinates	Demands	Items	ID	Coordinates	Demands	Items
0	(5, 5)	NA	NA	13	(4, 7)	4	208, 1110, 48, 774
1	(2, 4)	5	77, 655, 756, 1193, 299	14	(10, 1)	3	53, 533, 140
2	(7, 7)	2	796, 1169	15	(1, 2)	3	1156, 449, 872
3	(3, 8)	2	168, 338	16	(9, 8)	3	226, 687, 872
4	(8, 2)	1	1189	17	(4, 4)	5	351, 26, 1185, 504, 266
5	(2, 0)	2	1170, 369	18	(6, 9)	3	1104, 780, 1149
6	(5, 1)	1	893	19	(2, 10)	2	932, 885
7	(1, 6)	2	96, 303	20	(7, 10)	1	1089
8	(0, 10)	5	921, 459, 592, 1108, 695	21	(4, 10)	1	720
9	(10, 6)	1	96	22	(9, 1)	4	673, 495, 990, 851
10	(6, 3)	2	552, 946	23	(4, 2)	1	836
11	(4, 9)	3	199, 1157, 881	24	(3, 5)	3	287, 1100, 1132
12	(9, 4)	3	317, 1049, 39	25	(4, 7)	4	1176, 326, 509, 617

The customer coordinates (x,y) are randomly generated between $[0,10]$ and each cell is $300m \times 300m$. The front warehouse locates in (5,5), and manhattan distance is used to calculate the distance between two customers where $d_{ig}^{trans} = |x_1 - x_2| + |y_1 - y_2|$. The service time of each order λ is 1 min, the caution intensity λ_1 and λ_2 are set to 0.3 and 0.15, respectively. Owing to the small size orders, we assume that the number of items per order q_i is randomly sampled from U(1,5). The order delivery time limit d is 20 min, the order completion time limit lm is 30 min, and the overdue penalty cost in order picking stage c^{od} is 2 yuan/min. The order picking cost c^{op} is 1.5 yuan/min, the overload penalty factor α_1 is 5 and the delivery overdue penalty factor α_2 is 8. The experiments are conducted on an Intel Core i5 processor and 8.0 GB RAM. The algorithm is implemented with Pycharm.

5.2. Results analysis

The scheduling results are demonstrated in Table 3, Fig. 6 and Fig.7. In the delivery stage, there are 6 routes generated and each route contains 4 to 5 orders. A load of each vehicle is about 9 to 12, and the average cargo load factor is 91.7% which indicates the higher vehicle utilization.

Table3 Result of IZPDS

Batch	Vehicle routes	Load	Starting time	Departure time	Return time	Order fulfillment cost
5	0-2-16-18-20-0	9	0.0	6.2	19.6	42.4
3	0-1-15-5-6-0	11	0.6	7.5	26.1	44.3
6	0-24-17-23-10-0	11	1.3	8.5	23.3	42.8
1	0-3-19-8-7-0	11	2.1	9.2	24.5	34.9
2	0-9-12-22-14-4-0	12	2.8	10.4	29.2	45.6
4	0-13-11-21-25-0	12	3.7	11.3	25.1	38.0
	Sum	66	10.5	53.1	147.8	248.0
	Max	12	3.7	11.3	29.2	45.6
	Average	11	1.8	8.9	24.6	41.3
Average value per order		2.6	0.4	2.1	5.9	9.9

In the order picking stage, the starting time of order picking is the same as the waiting time of orders. The average waiting time of orders is 1.8 min, and the completion time of order picking (departure time of vehicles) is about 6 to 11 min. The proportion of effective order picking time is 79.3% revealing the high utilization of resources. Order fulfillment time is between 19 and 29 min. The total order fulfillment cost is 248 yuan, and the average order fulfillment cost per order is controlled within 10 yuan. All orders can be delivered within the promised time, and the on-time fulfillment rate is 100%, realizing cost optimization based on guaranteed service levels.

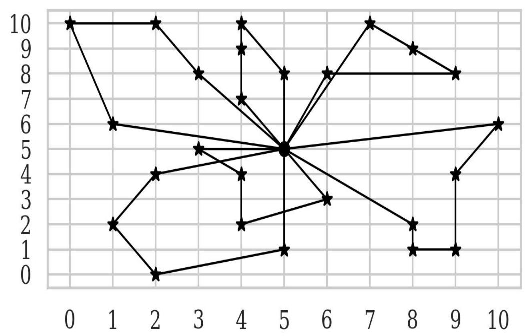


Fig.6 Vehicle routing diagram with TIS

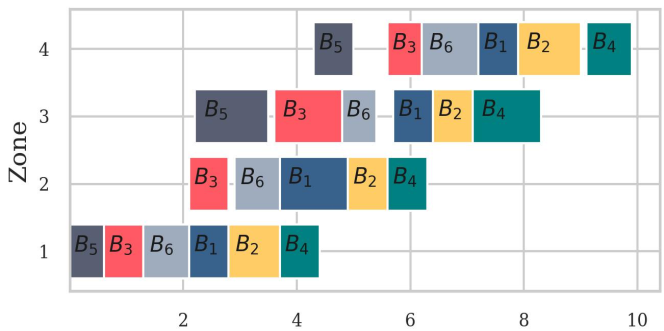


Fig.7 Order picking diagram with TIS

5.3. Results comparison

To verify the effectiveness of the TIS algorithm designed based on the RS idea, in this section we will compare the scheduling results produced by the TIS algorithm and the results based on the traditional sequential scheduling (TSS) approach from four indicators: order waiting time, vehicle waiting time, order fulfillment time, order picking cost, delivery cost and order fulfillment cost. The process of TSS is presented in Fig.8.

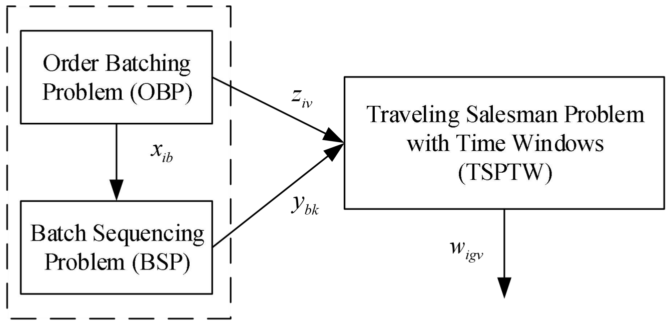


Fig.8 Traditional sequential scheduling

The order batching problem (OBP) is solved first, and then the batch sequencing problem (BSP) is solved. The entire model degenerates into the traveling salesman problem with time windows(TSPTW) after working out the OBP and BSP (deciding the x_{ib}、 y_{bk} and z_{iv}). It is only needed to determine the delivery sequence of orders when dealing with the TSPTW. The solution process is as follows. First, without considering the impact of delivery, we use the aisle similarity clustering method (Zhang, Wang, Chan, & Ruan, 2017) to realize the order batching operation to minimize order picking costs. Second, the SPT rule is applied to arrange the order where batches are arranged in ascending order according to the picking time. Batches with shorter picking times are prioritized. Eventually, combined with the latest completion time of the batch, Gurobi is used to solve the TSPTW to generate a delivery scheduling plan. The results based on TSS are exhibited in Fig. 9 and Fig.10, and the comparison of the results between TSS and TIS is reported in Table 4.

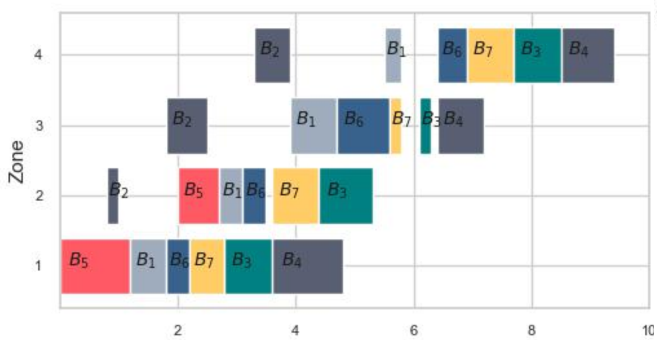


Fig.9 Order picking diagram with TSS

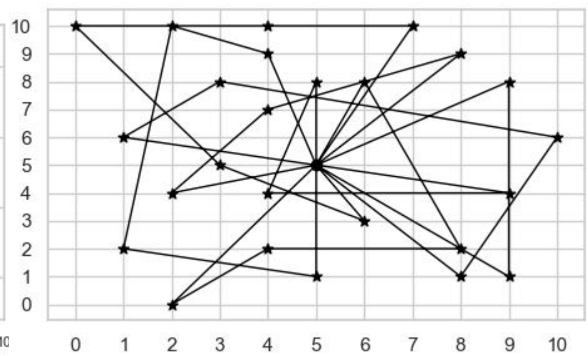


Fig.10 Vehicle routing diagram with TSS

As illustrated in Fig. 10, the geographical location of customers within the same batch differs greatly, and the vehicle routing diagram is in a mess so that there is greater room for optimization in the delivery stage. From Table 4, TIS is better than TSS in terms of time and cost, which effectively improves the efficiency and reduces the order fulfillment cost. The improvement of order waiting time and vehicle waiting time are 9.48% and 4.15% respectively, which improved the utilization rate of warehouse and vehicle resources.

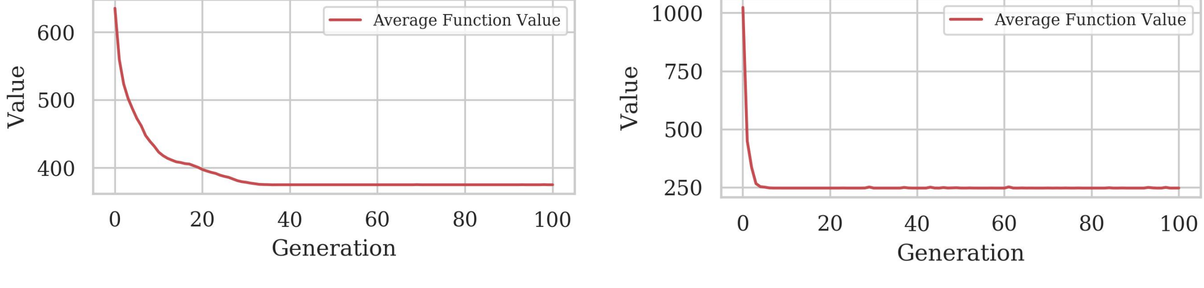
Table4 Results comparison between TSS and TIS

Indicators	TSS	TIS	Improvement rate(%)
Order waiting time	11.6	10.5	9.48
Vehicle waiting time	55.4	53.1	4.15
Order fulfillment time	179.83	147.8	17.81
Order picking cost	83	80	3.61
Delivery cost	288	168	41.67
Order fulfillment coat	371.1	247.95	33.19

From the perspective of improvement, TIS can significantly optimize the efficiency and cost of the delivery. The improvement rate of the delivery cost is the largest, reaching 41.67%, the overall improvement rate of order fulfillment cost is as high as 33.19%, and the optimization rate of the order fulfillment time is 17.81%. Such an improvement indicates that TIS approach can bring a significant added value to the consumers, while keeping the solution as economic as possible. The difference of results between TIS and TSS approach may be explained as follows. The assignment of orders is determined by OBP without considering the locations of customers. Therefore, orders belonging to the same vehicle may be scattered everywhere, resulting in increased delivery costs and overdue penalty costs. TIS can coordinate the two phases and make full use of the information in the delivery phase to optimize the order fulfillment process.

5.4. Performance of TIS

The convergence process of the TIS algorithm is shown in Fig.11. (Rudolph, 1994)used the finite Markov chain theory to prove that the canonical genetic algorithm (CGA) using only three genetic operators of crossover, mutation and selection (proportional selection method) cannot converge to the global optimal value. While the canonical genetic algorithm with elite retention is globally convergent(Rudolph, 1996). Therefore, we use the elite retention strategy based on CGA to ensure that the best individuals that have appeared so far will not be lost and destroyed by selection, crossover and mutation operations. The iteration result of the ordinary genetic algorithm where only chromosome generation approach 2 and a simple crossover operator are used is exhibited in Fig. 11 (a). The iteration result of the TIS algorithm is shown in Fig. 11 (b).



(a) (b)

Fig.11 Convergence diagram of algorithm TIS

Both algorithms are designed with an elite retention strategy, and the final objective value remains basically unchanged from Fig. 11 (a) and (b), indicating algorithms have converged. As can be seen from Fig.11, the TIS algorithm designed is superior to the ordinary genetic algorithm in convergence speed and solution quality because the optimal value of ordinary genetic algorithm is 346.15 while the one of TIS is 248.

To further test the performance of TIS, we select several examples of different scales. Each example is tested 5 times and the averaged result is used. To compare the results of the TIS algorithm with the optimal solution obtained by Gurobi, we set the maximum solution time of the CVRPTW model to be 600 seconds, run the solver twice, and take the averaged result. In addition, the comparison between TIS and H-2 algorithm is also conducted to further validate the performance of TIS. H-2 algorithm is similar to TIS, but there is no interaction between the order picking stage and delivery stage in the H-2 algorithm. The performance comparison is reported in Table 5.

Table5 Performance of the developed algorithm

订单数	CPU Time			TC			
	TIS	H-2	Gurobi	TIS	H-2	Gurobi	GAP(%)
15	24.4	14.0	622.1	165.6	168.1	165.4	0.14
20	27.5	15.5	643.4	208.4	214.7	206.8	0.80
25	30.9	16.5	716.4	248.0	255.4	247.1	0.36
30	35.4	21.2	805.1	332.1	337.5	316.1	5.07
35	40.8	19.7	866.6	374.0	378.2	360.5	3.72
40	42.2	21.7	892.0	419.3	421.6	397.5	5.49
45	45.2	21.9	1071.4	455.9	454.7	434.1	5.03
50	48.2	23.7	1121.7	497.5	492.0	490.1	1.51
55	51.5	23.6	1143.2	536.0	535.8	527.6	1.59
60	58.1	24.8	1161.6	643.3	637.7	618.2	4.07

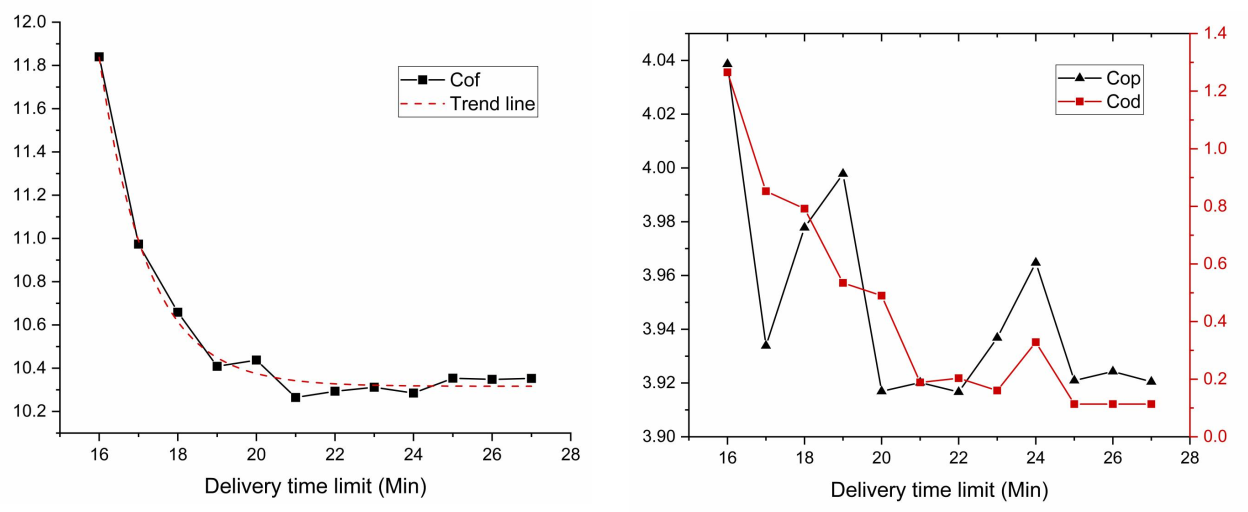
The gap between TIS and Gurobi is calculated as follows: $GAP = \left(TC^{\mathrm{TIS}} - TC^{\mathrm{Gurobi}}\right) \cdot 100\% / TC^{\mathrm{Gurobi}}$

From Table 5, it can be seen that when the scale of the example gradually increases to 60, for small-scale problems with less than 25 orders, the result of TIS is absolutely close to the optimal solution. When the problem size increases to 60, the gap rate increases but it is basically controlled within 5%. Therefore, TIS can mostly generate near-optimal solutions. Compared with the H-2 algorithm, on average, the TIS algorithm performs better with small-sized and medium-sized problems. However, when the problem size become relatively large, the gap between TIS and H-2 algorithm is not large because as the scale increases, the iterative search space of the TIS algorithm is enlarged, which reduces its performance. The solution time and order fulfillment cost with the TIS algorithm reveals a linear rather than an exponential trend, indicating that the high potential of the proposed TIS algorithm in achieving better solutions in acceptable times.

5.5. Parameters analysis

5.5.1. Time allocation

In terms of time allocation, Hema consumes about 33% of the total time to order picking, and the time allocation proportion between order picking and delivery in Missfresh is roughly 3.6:2. It can be seen that different platforms have different allocations of order fulfillment time. Different proportions of time allocation under different problem scales may result in cost fluctuations so that it is worthwhile to explore the impact of time allocation on order fulfillment costs and service quality and help decision-makers to determine the appropriate time allocation proportion. Simulation experiments for different instances with different time allocation proportions are conducted. Each example is run 5 times, and we take the averaged results. The simulation results are presented in Fig.12.



(a) (b)

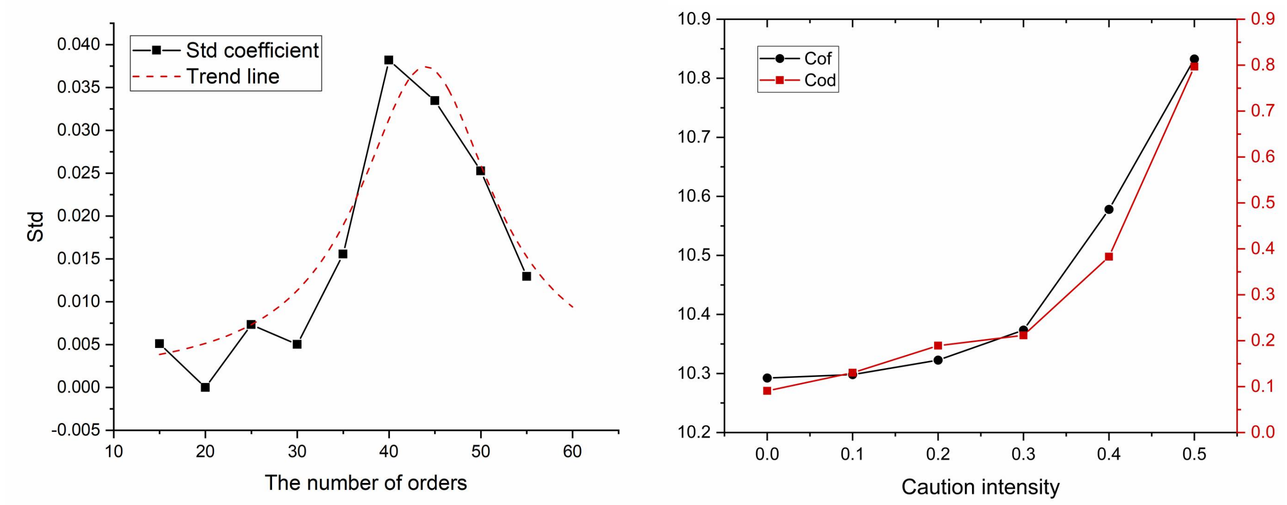
Fig.12 Analysis of time allocation

The change of the average fulfillment cost per order is illustrated in Fig. 12 (a). As a whole, the order fulfillment cost (Cof) in Fig. 12 (a) shows a downward trend with the increase of delivery time limit. There is a significant decrease when the delivery time limit is quite short, and the cost fluctuates slightly when the delivery time limit reaches 21 min. Fig. 12 (b) shows the fluctuation of the average order picking cost (Cop) and overdue penalty cost (Cod) of a single order with the increase of delivery time limit. As can be seen from the figure, as the delivery time limit increases, Cod generally decreases, while the Cop fluctuates to some extent.

When the delivery time limit is 21 ~ 23 min, the Cof, Cop and Cod remain a stable low level. When the delivery time limit exceeds 24 min, there is a small increase in Cof, and Cop and Cod first increase and then fall quickly before maintaining stability. These facts can be explained as follows. When the delivery time limit is quite short (less than 21 min in this experiment), it will cause large delivery overdue costs. When the delivery time limit is longer (from 23 to 24 min in this experiment), the time reserved for the order picking is relatively short, which easily causes the cost of picking overdue. When the delivery time limit is long enough (greater than 25 min in this experiment) to exceed a certain threshold, the order is destined to be overdue in the order picking stage, and there is hardly any optimization space in costs. From the perspective of the time allocation adjustment, compared with the appropriately reducing and greatly expanding, appropriately extending the delivery time limit is conducive to obtaining optimal results. Hence, for the managers, when carrying out the integrated scheduling of order picking and delivery, it is necessary to pay attention to adjusting the time allocation of each stage and appropriately extend the time limit of delivery.

5.5.2. Caution intensity

In this paper, we assume that the delivery person will adjust the caution intensity according to the delivery statuses to minimize the risk of cargo damage. Therefore, different intensity of caution may cause different cost fluctuations. It is necessary to analyze the intensity of caution.We construct different caution intensity combinations [λ , 0.75 λ], and take λ as [0, 0.1, 0.2, 0.3, 0.4, 0.5].Each example will be run 5 times, and the averaged results will be used. The results are revealed in Fig.13.



(a) (b)

Fig.13 Analysis of caution intensity

Fig. 13 (a) shows the change of the standard deviation coefficient of order fulfillment costs for different problem sizes. The larger the standard deviation coefficient, the greater the fluctuation of order fulfillment costs at the scale of the problem, and the more sensitive the model is to the caution intensity. It can be seen from the figure that the standard deviation coefficient shows a parabolic trend. The small-scale and large-scale problems have small standard deviation coefficients, while the medium-scale standard deviation coefficients are large. Fig. 13 (b) is a line chart of order fulfillment costs and overdue penalties costs. It can be seen from the figure that, in general, both types of costs show an upward trend as the caution intensity increases. When the caution intensity is between 0 and 0.3, the increase in costs is relatively flat, while between 0.3 and 0.5, both types of costs have increased significantly. Therefore, for managers, it is possible to provide a strong caution intensity when the scale is small to increase service levels at a lower cost. For large-size instances, the caution intensity should be controlled to avoid significant cost increases.

6. Conclusions

To achieve the global optimal order fulfillment performance, it is of vital importance to integrate the order picking and delivery decisions into a single optimization problem. We present a mixed-integer model for managing integrated scheduling of zone picking and vehicle routing with time windows (ISZPVRTW) in the FWM and consider the real-world factors impactive in planning order fulfillment process, such as the fulfillment deadline, convey time between the picking zones, setup time and the variable driving speed. The decisions on order batching, picking sequencing, and vehicle routing are made together. To solve the model efficiently, we analyze the complexity of the model and the optimal solution properties. An efficient two-stage iterated search (TIS) algorithm is proposed, and it is validated by a series of computational experiments. For instances within 60 orders, the gap of results produced by TIS and Gurobi is approximately not beyond 5%. In general, the TIS algorithm performs better than CGA and H-2.

Furthermore, the value of integration was examined by comparing the TIS and traditional sequential scheduling (TSS) approach. There are several benefits for retailers to adopt an integration policy. First, the integration policy can significantly achieve a cost saving of order fulfillment which is even up to 33.19%. Second, from the perspective of service level, the waiting time of orders and vehicles as well as the order fulfillment time are all shortened by integrating picking and delivery operations, which means retailers are able to arrange the warehouse and distribution resources to meet demands efficiently. Thus, the integration leads to the positive impacts on both the economic and service performance indicators, offering retailers added strength in this challenging environment. From a managerial point of view, we demonstrate the influence of time allocation and caution intensity. It is more beneficial to appropriately extend the delivery time limit when allocating the time to two stages. For small-sized problems, it is recommended to provide a strong caution intensity for it can enhance service levels with lower cost growth, but the caution intensity should be controlled to avoid significant cost growth when the problem scale is getting large.

There are various opportunities to extend the problem. In this paper, we assume that only one picker is assigned to each zone, which may cause long waiting times and uneven workload in the zone when the order quantity is large. Besides, online orders usually arrive dynamically over time. Thus, our research directions could be the on-line flexible picking scheduling problem with multiple zones and pickers.

Acknowledgments

This research has been supported by the National Natural Science Foundation of China (Nos. 71671036, 71171046), Major Projects of Philosophy and Social Sciences in Universities of Jiangsu Province (No. 2018SJZDA005).

References

Alawneh, F. & Zhang, G. 2018. Dual-channel warehouse and inventory management with stochastic demand. Transportation Research Part E: Logistics and Transportation Review, 112: 84-106.

Armentano, V. A. & Shiguemoto, A. L. & Løkketangen, A. 2011. Tabu search with path relinking for an integrated production–distribution problem. Computers & Operations Research, 38(8): 1199-1209.

BigData-Research. 2019. 2019 H3 China Instant Delivery Industry Development Report.

CGAP. 2019. The New Retail Revolution.

Chen, H. & Hsueh, C. & Chang, M. 2009. Production scheduling and vehicle routing with time windows for perishable food products. Computers & Operations Research, 36(7): 2311-2319.

Chen, Z. 2010. Integrated production and outbound distribution scheduling: review and extensions. Operations Research, 58(1): 130-148.

Cheng, C., Chen, Y., Chen, T., & Jung-Woon Yoo, J. 2015. Using a hybrid approach based on the particle swarm optimization and ant colony optimization to solve a joint order batching and picker routing problem. International Journal of Production Economics, 170: 805-814.

Clarke, G. & Wright, J. W. 1964. Scheduling of Vehicles from a Central Depot to a Number of Delivery Points. Operations Research, 12(4): 568-581.

Croes, G. A. 1958. A method for solving Traveling-Salesman problems. Operations Research, 6(6): 791-812.

De Koster, M. B. M. & Van der Poort, E. S. & Wolters, M. 1999. Efficient order batching methods in warehouses. International Journal of Production Research, 37(7): 1479-1504.

De Koster, R. 1994. Performance approximation of pick-to-belt orderpicking systems. European Journal of Operational Research, 72(3): 558-573.

De Koster, R. & Le-Duc, T. & Roodbergen, K. J. 2007. Design and control of warehouse order picking: A literature review. European Journal of Operational Research, 182(2): 481-501.

GADEMANN, N. & VELDE, S. 2005. Order batching to minimize total travel time in a parallel-aisle warehouse. IIE Transactions, 37(1): 63-75.

Garey, M. R. & Johnson, D. S. & Sethi, R. 1976. The complexity of flowshop and jobshop scheduling. Mathematics of Operations Research, 1(2): 117-129.

Hema. 2019.

Henn, S. 2015. Order batching and sequencing for the minimization of the total tardiness in picker-to-part warehouses. Flexible Services & Manufacturing Journal, 27(1): 86-114.

Ho, Y. C. & Tseng, Y. Y. 2006. A study on order-batching methods of order-picking in a distribution centre with two cross-aisles. International Journal of Production Research, 44(17): 3391-3417.

IIMediaResearch. 2019a. 2019H1 China E-Commerce Development Overview Report.

IIMediaResearch. 2019b. 2019H1 China Immediate Delivery Market Research Report.

JD. 2019.

Kaminsky, P. 2003. The effectiveness of the longest delivery time rule for the flow shop delivery time problem. Naval Research Logistics (NRL), 50(3): 257-272.

Li, X., Li, J., Aneja, Y. P., Guo, Z., & Tian, P. 2019. Integrated order allocation and order routing problem for e-order fulfillment. IISE Transactions, 51(10): 1128-1150.

Liou, C. & Hsieh, Y. 2015. A hybrid algorithm for the multi-stage flow shop group scheduling with sequence-dependent setup and transportation times. International Journal of Production Economics, 170: 258-267.

Low, C., Chang, C., Li, R., & Huang, C. 2014. Coordination of production scheduling and delivery problems with heterogeneous fleet. International Journal of Production Economics, 153: 139-148.

M, R. 2012. Warehousing in the Global Supply Chain. London: Spinger-Verlag.

Melacini, M. & Perotti, S. & Tumino, A. 2011. Development of a framework for pick-and-pass order picking system design. The International Journal of Advanced Manufacturing Technology, 53(9-12): 841-854.

Missfresh. 2019.

Moons, S., Ramaekers, K., Caris, A., & Arda, Y. 2017. Integrating production scheduling and vehicle routing decisions at the operational decision level: A review and discussion. Computers & Industrial Engineering, 104: 224-245.

Moons, S., Ramaekers, K., Caris, A., & Arda, Y. 2018. Integration of order picking and vehicle routing in a B2C e-commerce context. Flexible Services and Manufacturing Journal, 30(4): 813-843.

Moons, S., Braekers, K., Ramaekers, K., Caris, A., & Arda, Y. 2019. The value of integrating order picking and vehicle routing decisions in a B2C e-commerce environment. International Journal of Production Research, 57(20): 6405-6423.

Pan, J. C. & Shih, P. & Wu, M. 2015. Order batching in a pick-and-pass warehousing system with group genetic algorithm. Omega, 57: 238-248.

Papadimitriou, C. H. 1977. The Euclidean travelling salesman problem is NP-complete. Theoretical Computer Science, 4(3): 237-244.

Parikh, P. J. & Meller, R. D. 2008. Selecting between batch and zone order picking strategies in a distribution center. Transportation Research Part E: Logistics and Transportation Review, 44(5): 696-719.

Petersen, C. G. 2002. Considerations in order picking zone configuration. International Journal of Operations

& Production Management, 22(7): 793-805.

Pinedo, M. & Hadavi, K. 2016. Scheduling: theory, algorithms and systems development. A I I E Transactions, 28(8): 3.

Rudolph, G. 1994. Convergence analysis of canonical genetic algorithms. IEEE Transactions on Neural Networks, 5(1): 96-101.

Rudolph, G. 1996. Convergence of evolutionary algorithms in general search spaces.

Sheikh, S., Komaki, G. M., Kayvanfar, V., & Teymourian, E. 2019. Multi-Stage assembly flow shop with setup time and release time. Operations Research Perspectives, 6: 100-111.

Soukhal, A. & Oulamara, A. & Martineau, P. 2005. Complexity of flow shop scheduling problems with transportation constraints. European Journal of Operational Research, 161(1): 32-41.

van Gils, T., Ramaekers, K., Braekers, K., Depaire, B., & Caris, A. 2018. Increasing order picking efficiency by integrating storage, batching, zone picking, and routing policy decisions. International Journal of Production Economics, 197: 243-261.

van Gils, T., Caris, A., Ramaekers, K., & Braekers, K. 2019. Formulating and solving the integrated batching, routing, and picker scheduling problem in a real-life spare parts warehouse. European Journal of Operational Research, 277(3): 814-830.

Yu, M. & De Koster, R. 2008. Performance approximation and design of pick-and-pass order picking systems. IIE Transactions, 40(11): 1054-1069.

Yu, M. & de Koster, R. B. M. 2009. The impact of order batching and picking area zoning on order picking system performance. European Journal of Operational Research, 198(2): 480-490.

Zhang, J. & Wang, X. & Huang, K. 2016a. Integrated on-line scheduling of order batching and delivery under B2C e-commerce. Computers & Industrial Engineering, 94: 280-289.

Zhang, J. & Wang, X. & Huang, K. 2016b. Integrated on-line scheduling of order batching and delivery under B2C e-commerce. Computers & Industrial Engineering, 94: 280-289.

Zhang, J., Wang, X., Chan, F. T. S., & Ruan, J. 2017. On-line order batching and sequencing problem with multiple pickers: A hybrid rule-based algorithm. Applied Mathematical Modelling, 45: 271-284.

Zhang, J. & Wang, X. & Huang, K. 2018. On-line scheduling of order picking and delivery with multiple zones and limited vehicle capacity. Omega, 79: 104-115.

Zhang, J., Liu, F., Tang, J., & Li, Y. 2019a. The online integrated order picking and delivery considering Pickers' learning effects for an O2O community supermarket. Transportation Research Part E: Logistics and Transportation Review, 123: 180-199.

Zhang, J., Liu, F., Tang, J., & Li, Y. 2019b. The online integrated order picking and delivery considering Pickers' learning effects for an O2O community supermarket. Transportation Research Part E: Logistics and Transportation Review, 123: 180-199.

Žulj, I. & Kramer, S. & Schneider, M. 2018. A hybrid of adaptive large neighborhood search and tabu search for the order-batching problem. European Journal of Operational Research, 264(2): 653-664.

前置仓模式下订单分区拣选与配送路径优化集成调度研究

陈伟，张玉林

（东南大学 经济管理学院，江苏 南京 211189）

摘　要：近年来，订单拣选与配送的集成调度在商业领域受到很大关注。越来越多商家采用前置仓与即时配送相结合的订单履约模式来满足即时消费需求。但受限于即时消费订单“多品种、小批量、高频次、时间敏感”的碎片化特征，如何在承诺的送达时间内以最低的成本完成订单履约是商家面临的核心挑战。为了更高效、低成本完成订单履约，本文研究前置仓模式下订单拣选与配送集成调度问题，对分区串行拣选系统下订单分批、拣货排序以及配送路径规划问题进行统一决策以最小化订单履行成本。本文结合实践中订单具有履约期限，拣货分区之间具有运输时间，配送速度随配送状态可变等特点，构建混合整数规划模型，并基于逆序调度（RS）思路设计了两阶段迭代搜索的算法(TIS)进行求解。最后通过数值实验对比了TIS算法与商业求解器以及两阶段算法（H-2），结果验证了算法的高效性。相较于传统顺序调度（TSS）方法，基于RS设计的TIS算法能够有效提高订单履行效率并优化成本。

关键词：前置仓；集成调度；订单分区拣选；路径优化；两阶段迭代搜索算法