International Journal of Innovative
Computing, Information and Control
Volume **19**, Number **1**, February **2023**

ICIC International ©2023 ISSN 1349-4198
pp. **269–278**

A PULL ORDER FULFILLMENT METHOD COMBINING JIT WITH GROUP TECHNOLOGY FOR ONLINE SUPERMARKET

Yanxin Wang[1,*], Tingli Wang[2], Yueyue Zhao[2] and Minfang Huang[2]

[1]Business College
Beijing Union University
A3, Yanjingdongli, Chaoyang District, Beijing 100025, P. R. China
*Corresponding author: yanxin.wang@buu.edu.cn

[2]School of Economics and Management
North China Electric Power University
No. 2, Beinong Road, Changping District, Beijing 102206, P. R. China
993454271@qq.com; zhaoyueyue0228@163.com; huangmf@ncepu.edu.cn

Received April 2022; revised July 2022

Abstract. *The order fulfillment problem in online supermarket is extremely complicated due to the unique features of orders. In the paper, we attempt to combine JIT (just in time) philosophy with group technology and propose a method of pull-type order fulfillment for online supermarket to improve its flexibility and efficiency. First, referring to the JIT production line with free beat, we build the pull order processing of order fulfillment including order picking, checking, packing, and delivering for order batches. Then we address the implementation strategies of key issues, order grouping, flow balance, sequencing of the optimization activities and the Kanban setting, as well as the construction of the similarity matrix for orders and a two-stage workload balancing strategy with dynamic hybrid beats. Finally, a numerical experiment is constructed to verify the effectiveness of the method. The proposed method in this paper offers a basis for developing and rethinking order fulfillment strategy in the warehouse for large online supermarkets.*
Keywords: Pull order fulfillment, JIT, Group technology, Online supermarket

1. Introduction.

In recent years, the large-scale online supermarket is gradually becoming one of the new online retailing modes that mainly sells the daily fast moving consumer goods. The large number of orders have randomness in arriving time and delivery locations and consist of a variety of goods that is 8 to 10 times the average number of normal online retailing. The customer delivery time requirements are strict and diverse, e.g., half-day delivery, regular delivery and three deliveries per day. Therefore, the organization and scheduling of the large number of personalized orders' fulfillment process (including order picking, checking, packing and delivering) in large-scale online supermarkets are complicated.

Recently, many academic results have been achieved on the order batching for order picking problem and order delivery problems of online supermarkets. Among them, the main achievements include batching orders by time window [1,2] and similarity of items in the orders [3]. For order picking problem, the batching models consider the similarity of order arrival time or the similarity of items between orders and the goals of the shortest total order picking time or total picking distance. Heuristic algorithms are usually developed to solve the models [4]. For order delivery problem, the results mainly focus on the delivery route planning and utilization rate optimization of vehicles [5,6]. However,

DOI: 10.24507/ijicic.19.01.269

these results still mainly address the optimization strategy or solution procedure for one single specific step, and less consider its impact on upstream and downstream steps and the efficiency of the entire order fulfillment process. Although several scholars have studied joint optimization of multi-stage order processing [7], their results are still essentially sequential forwarding optimizations along the process of order fulfillment, which is hard to decrease the stacking or buffering time of goods in each step. Thus, there still lacks a scientific theory to guide the multi-step operational process to be implemented efficiently and timely. Furthermore, there exist high order splitting rate and logistics cost in online supermarket, that is, most of orders are split and fulfilled by multiple packages and multiple deliveries. To solve these problems and enhance the service flexibility and improve the core competitiveness of online supermarkets, it is necessary to introduce the group flow mode in mass manufacturing and the just in time (JIT) theory in assembly lines to the order fulfillment process in online supermarkets. JIT was firstly applied in manufacturing industry. It aims to improve production efficiency and reduce wastes through pulling materials and parts and implementing mass production. Lean production incorporating JIT has been widely applied in the fields of mass processing and manufacturing, and steadily extended to small-size production with specific customization [8]. Lean logistics [9], lean services [10], lean knowledge management [11], etc., have emerged helping enterprises improve their core competitiveness by less cost and higher benefits.

Compared with those related works, we are aiming to combine the whole order fulfillment process as an integrated optimization problem, which focuses on not one specific step but at least three process steps: item picking, sorting and packaging. Therefore, referring to the JIT philosophy of delivering the right product to the right place in the right quantity at the right time [12], this paper aims to innovatively build a method to organize and schedule the whole process of order fulfillment with high efficiency by reducing the preparing and waiting time in each step and the cost on goods stacking, and to provide strategies of pull order processing fulfillment based on group technology and Kanban.

The key issues in achieving pull order processing include the following. 1) How to comprehensively establish order similarity rules to group orders from the perspective of the entire order fulfillment process? 2) For the grouped orders, how to achieve the workload balance between steps and realize JIT order processing in the operational flow? In this paper, we will discuss the strategies to solve the above issues by the implementation of pull order processing.

The remainder of the paper is structured as follows. The design principle of pull order fulfillment is presented in Section 2. Section 3 presents the implementation of key strategies for JIT order pull processing. In Section 4, a numerical experiment is conducted to demonstrate the performance of the proposed method. These are followed by a conclusion and a discussion of possible future research directions in Section 5.

2. **Design Principle of Pull Order Fulfillment.**

JIT pull production is a demand-driven, bottom-up model which avoids redundant inventory and delayed delivery time. The basic theory of the pull production mode is that the materials are only delivered when and where they are really needed, or the component is produced only when the demand is received from a customer. Accordingly, to apply JIT to order fulfillment process, we need to transform the traditional push order processing into pull type to realize delivering the right product to the right place in the right quantity at the right time, and thus reduce the amount of stacking goods and waiting time on each step in order fulfillment.

Considering the high randomness of orders and the strictness of delivery deadline, the biggest difference between online supermarket logistics and traditional logistics is that the former should try to form a fast-in and fast-out channel to save costs and improve efficiency. Currently, the push-type system for the group flow mode is widely used in order fulfillment in online retailing. It usually takes four steps. In the first step, orders are batched by the principle of optimal picking efficiency for received orders over a period of time. The second step is to pick the required goods and send them to the collection position for checking. The third step is to pack the items by different orders. Finally, the packages will be loaded on the truck and delivered to the distribution centers or customers. In this case, the results of previous step will directly push the next step, and each step is planned and organized by current resources. Therefore, order batching, splitting, picking, checking, packing, and delivering in the push-type order fulfillment are a type of sequential advancing relationship. The advantages of this processing are that the workflow flows naturally, and each operation can schedule its workflow according to their needs and resources, and the process can be optimized by various optimal methods. However, since the impact of subsequent job is seldom considered while implementing the current job, and each job might need to be rearranged according to the previous job's output, it is hard to realize collaborative optimization between jobs.

The pull order processing is established from the perspective of the overall process of order fulfillment, including the steps of picking, checking, packing, and delivering. Pull order processing first applies the similarity principle of group technology to summarizing grouping rules for orders. Next, orders are grouped and a processing sequence is established for each group, which is equivalent to an order assembly line. Finally, the beat is set for the order assembly line according to the features of the grouped orders and the designing of beat in JIT assembly line. It ensures the items in the orders go through each step of the order processing in accordance with the beat and only the right quantity of items or packages are pulled to the right location at the right time during the order processing. The diagram of JIT pull order processing established in this paper is shown in Figure 1.

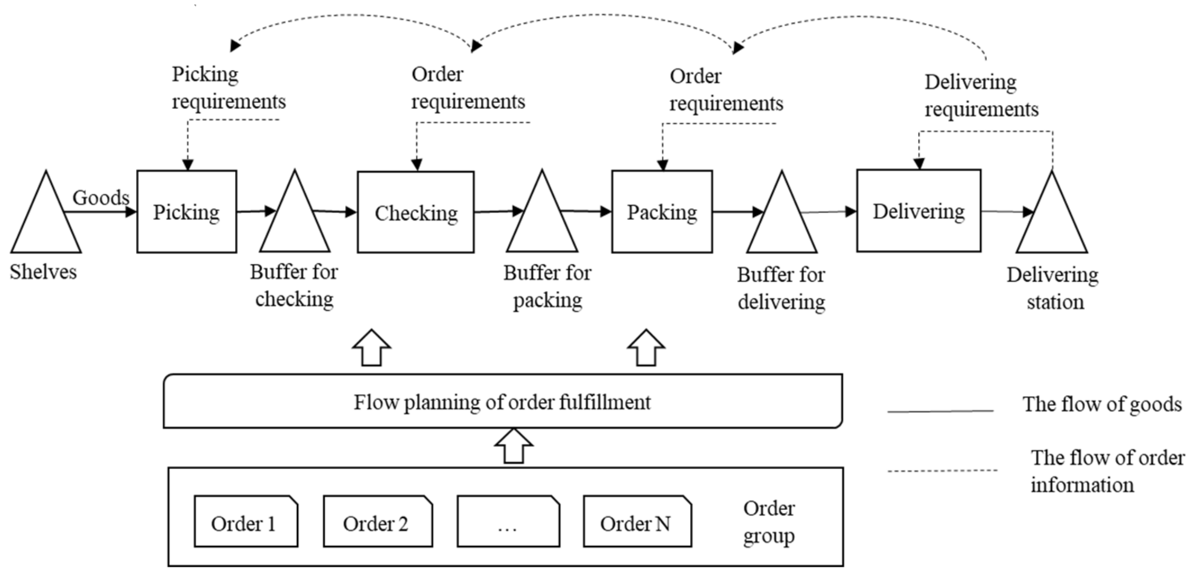


FIGURE 1. The diagram of the pull order processing

3. Implementation of Key Strategies for JIT Order Pull Processing.

3.1. **Order grouping.** Group technology identifies and explores the similarities of the relevant elements or objects in an activity and uses them to categorize similar objects into groups. In recent years, the application areas of group technology have been extended from manufacturing industry to software engineering, knowledge management, etc.

The prerequisite of building a pull order assembly line is possessing the capability of batch processing. Considering the highly personalized characteristics of online supermarket orders, group technology is appropriately adopted to batch the orders having similar attributes before going into the process of picking, checking, packing, and delivering. For orders within a group, not only the similarity between orders, but also the requirements of customers for orders, e.g., delivery time, delivery destination should be as similar as possible. Meanwhile, beats of the pull order assembly line require balanced pace for each order group.

The implementation of order grouping includes standardization of key order features and evaluation of order similarity. For orders in large-scale online supermarkets, the key features include order arrival time, item type, item quantity, delivery destination, delivery time, etc. Since they have different units of measurement, the similarity between orders cannot be quantified completely. To establish order similarity evaluation, we need to standardize and quantify the above features by using fuzzy clustering method. The order similarity matrix can be constructed as follows.

The set of orders arrived in the same period is represented as $X = \{x_1, x_2, \ldots, x_n\}$, in which n represents the number of orders. X_{ij} represents the value of feature j of order i, and $j \in \{1 = \text{Order arrival time}, 2 = \text{Delivery time}, 3 = \text{Delivery destination}, 4 = \text{Product items}\}$. The key features of the orders can be represented in a matrix shown in Table 1.

TABLE 1. The values of parameters describing order features

Order	Order arrival time X_{i1}	Delivery time X_{i2}	Delivery destination X_{i3}	Product items X_{i4}
1	X_{11}	X_{12}	X_{13}	X_{14}
2	X_{21}	X_{22}	X_{23}	X_{24}
...	...	...	...	...
n	X_{n1}	X_{n2}	X_{n3}	X_{n4}

We normalize them by the following steps.

1) For X_{i1}, set the earliest order arrival time as 1. For orders arriving within one hour later to the earliest order arrival time, all their X_{i1} will be 1. And for orders arriving above one hour delay to the earliest order arrival time, their X_{i1} will be added by 1 for each hour of delay.

2) For X_{i2}, set the earliest required delivery time as 1. Taking half a day as a time slot, we set X_{i2} of the orders within the same time slot 1 and every delay of a time slot will add 1 to X_{i2}.

3) For X_{i3}, calculate the distance between the order delivery destination and the delivery station and take the shortest distance as 1. Set the standard of the delivery destination distance as S (e.g., 5 km). If the distance difference between the order delivery distance and the shortest distance is less than S, X_{i3} will be 1; otherwise, X_{i3} will be added by 1 for each increase of S in the distance.

4) For X_{i4}, calculate the similarity of the items in the orders. The X_{i4} of each order takes the value of the vector containing the order items and the similarity of the order items can be evaluated by the following three steps. First, calculate the ratio of the frequency of an item appearing in an order to the total number of orders, represented as $\alpha \in [0, 1]$. Secondly, the items with the ratio higher than a specific value (e.g., 0.6) are defined as core items to form the core item vector. Finally, the Manhattan distance between the item vector is represented by X_{i4} and the core item vector is calculated as a measure of order similarity.

After standardizing the matrix, systematic clustering and stepwise clustering can be used to decide the grouping results. The JIT pull order processing is then performed in order groups. Each group is equivalent to an order assembly line. Orders within one group can be divided into several batches handled successively, so that the production line can run with a more balanced loading.

3.2. **Balance of workload.** For orders in each group, we need to batch them considering balancing the picking workload of each batch in different picking zones and the workload of each step in order processing. Due to the huge number of categories, the goods need to be classified and stored according to their relevance, creating multiple independent storage areas with fixed pickers in charge of fixed areas of goods. This strategy can reduce the picking zones one picker covers, as well as reduce the crowded aisles and increase the efficiency. The balance of the workload in a JIT assembly line means all the steps of order processing should proceed smoothly in a rhythm. It can eliminate underload at the beginning and overload at the end in order processing, and vice versa. And it can avoid the extra processing costs caused by unstable order processing tempo.

The beat is a key parameter in JIT balanced optimization which refers to the time interval between consecutive processes of two identical products on an assembly line. It is necessary to perform balanced adjustment within the order processes. The order assembly line balancing implementation is to optimize the pull order fulfillment process under a continuous and stable condition, balancing the load of each step, minimizing idle time, and maintaining a high efficiency of the whole process. The balance of the workload in a JIT order assembly line can be handled in two stages. In the first stage, considering picking is the most complex, labor-intensive, and cost-intensive process of the warehousing. And customers require orders should be delivered within a strict time window, order batching will take several factors (e.g., picking strategy, and workload balancing between picking zones) into account to balance the workloads between batches. In the second stage, the balance of operational flow will be optimized and scheduled. Due to the high uncertainty of online supermarket orders, the load on the whole order fulfilment process constantly changes and needs to be adjusted according to the reality. In the spare period, some orders containing one single item or those urgent can be added to and filled the shortage to balance the line.

We apply dynamic hybrid beats as a balancing strategy to implementing the two-stage balancing strategy.

1) Balancing the picking workload of each batch in different picking zones

We build an order picking optimization model with an objective of the minimum sum of standard deviation between the completion time in different picking zone of each batch and its average order picking completion time. Meanwhile, depending on the different situations of orders in online supermarkets, other objective functions can be built to replace the smoothing index, such as the loading equation.

We use B to represent the set of order batches in one group ($b \in B$), F the set of all picking zones ($f \in F$), and t_{pf} the picking time of order batch p in picking zone f.

The objective function is

$$\min \sqrt{\frac{\sum_{f \in F, p \in B} \left(t_{pf} - \frac{\sum_{f \in F, p \in B} t_{pf}}{F} \right)^2}{F}} \tag{1}$$

2) Balancing the workload of each step in order processing – order batch sequencing

The three operations of picking, checking, and packing of orders can be regarded as three steps carried out sequentially on the assembly line. The idle time of this line is caused by the difference of the operating time of all order batches in each step and the operating time of a single batch order in different steps of the same process [13]. For example, in the step of picking, there is a difference in workload of a batch of orders distributed in different picking zones due to the different demands for the items in each zone. When batching the orders, idle time can be reduced by balancing both the size of each batch of orders and the workload of a single batch order split to picking zones. Thus, the balancing is a permutation flow shop problem with the object as the minimum time of total order fulfilling time in all batches of one order group, which is represented by $T_{\max}$. To simplify the calculation of the picking time for each order batch, we define the following parameters.

h is the step of the order fulfillment process, $h \in \{1 = \text{picking}, 2 = \text{checking}, 3 = \text{packing}\}$;
t^{setup} is the setup time;
t^{pick} is the picking time for each SKU (Stock Keeping Unit);
NF_p is the number of picking zones for order batch p;
NS_p is the total number of SKUs in order batch p;
t_{pf}^h is the operation time on step h of order batch p in picking zone f;
t_p^h is the total operation time on step h of order batch p;
ts_k^h is the start time on position k of step h;
te_k^h is the end time on position k of step h.

Thus, for order batch p, the picking time on picking zone f is $t_{pf}^1 = t^{setup} + NS_p t^{pick}$. The total picking time for batch p is $t_p^1 = \max\left\{t_{pf}^1\right\}$. The checking time is most relative to numbers of picking zones and SKUs in the order batch, and it is represented as $t_p^2 = t^{checking} NF_p NS_p$, where $t^{checking}$ is a constant coefficient of collecting and checking time. The packing time is most relative to numbers of orders SKUs in the order batch, and it is represented as $t_p^3 = t^{pack} N_p NS_p$, where t^{pack} is a constant coefficient of packing time and N_p is the number orders in batch p. The balancing strategy can be described as the following:

$$Min\left(\sum_k \sum_h \left(ts_k^h + \sum_p t_p^h\right)\right) \tag{2}$$

$$s.t.\ \ ts_k^h = \begin{cases} \max\left\{te_{k-1}^h, te_k^{h-1}\right\}, & \forall k \geq 2,\ h \geq 2 \\ te_k^{h-1}, & \forall k = 1,\ h \geq 2 \\ te_{k-1}^h, & \forall k \geq 2,\ h = 1 \\ 0, & k = 1,\ h = 1 \end{cases} \tag{3}$$

3.3. **The Kanban management for order fulfillment.** Kanban refers to a signal board or a visual trigger which contains information that needs to be referenced for each step of the assembly line. Kanban is a relatively simple tool to realize pull production in JIT system and a critical part to implement the JIT pull order processing. The Kanban system is based on flow process. It replaces the traditional material delivery system with a pick-up process, using Kanban as the instruction for pick-up, delivery, and production to control the operations. It is triggered by the demand at the final stage and follows the reverse process route sequence. The process is carried out from the final stage until to the raw material preparation and thus all functions can be coordinately operated with punctuality and synchronization [14].

The information of the Kanban contains work content, quantity, time, destination, storage place and handling tools in each step of order fulfillment. In pull order processing, Kanban is used as a command for controlling order picking, checking, packing, and delivering. By pushing forward from delivering Kanban to picking Kanban to make the synchronization of all aspects of picking, we design the application diagram of Kanban in pull order processing shown in Figure 2.

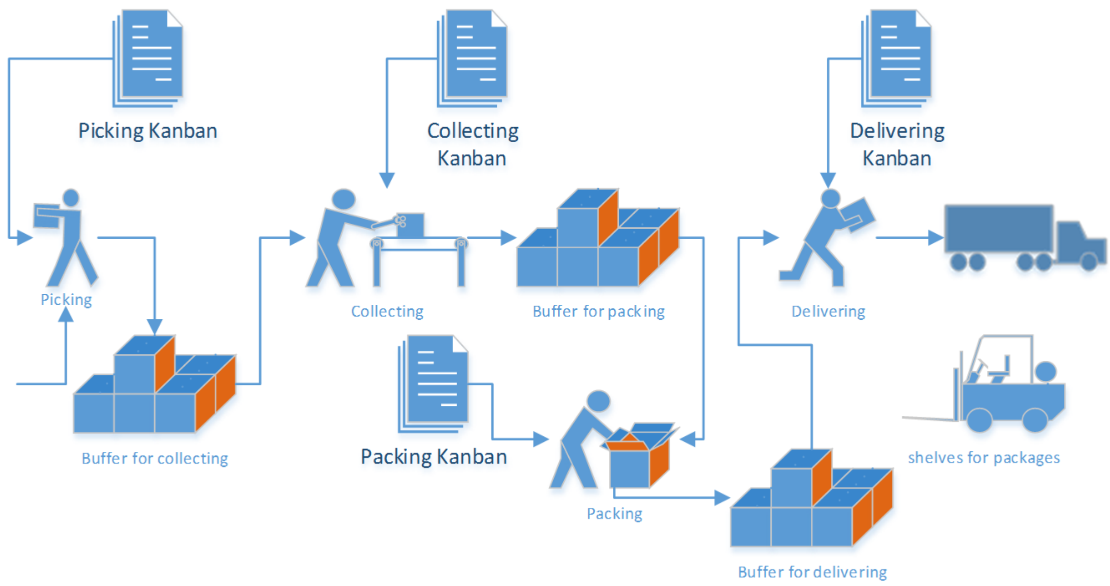


FIGURE 2. The application of Kanban in pull order processing

The Kanban flow forms an invisible assembly line, the two major driving forces of which are the final step, the delivering, and the sequence of production (from picking to checking, packing, and finally to delivering). The flow of items is activated by the final delivering, which is the terminal of the pull production system. With the use of the Kanban flow, the delivering synchronically generates control information to pull upstream lines. The operation performance of a multi-stage, mixed-model assembly line depends on various decision variables. Both reasonable application and parameters setting of a Kanban system, therefore, have significant impact on the whole system.

The capacity of the Kanban decides the level of inventory. That means the work-in-process inventory will correspondingly increase as the capacity of the Kanban increases. Oppositely, the work-in-process inventory will decrease as the capacity of the Kanban decreases; meanwhile, shortage may occur. The number of Kanbans will affect the capacity of order processing buffer and system performance. Therefore, in JIT production system, it is important to determine the optimal capacity of the Kanban to ensure minimum work-in-process inventory and no shortage happens. Through immediate conversion of customer requirements into production, the Kanban system demonstrates the ability to react quickly. Kanban also can visually indicate when production should start and end. Through controlling of a series of Kanban, only necessary items are produced. In turn, the number and assignment of Kanbans determine the beat and workload balance of the assembly line. The beat $T_{\max}$ can be defined as the maximum processing time of one step in the assembly line, i.e., the processing time of the bottleneck job. It determines the beat of the entire quasi-assembly line.

The operational rules of all the Kanban are as follows: in the process of loading, managers release the information of delivering Kanban according to the current truck's planned delivery path and vehicle loading status; the workers for delivering calculate and generate the information of packing Kanban based on delivering Kanban information and buffer information; in the packing process, we calculate and generate the collecting Kanban information from the information of packing Kanban and the packing buffer; workers in charge of items collecting calculate and generate the picking Kanban based on the information of the collecting Kanban and the collecting buffer, guiding the picking workers to complete the picking operation. At the beginning of each order fulfillment cycle, the Kanban at each step is eliminated to form an optimal processing sequence for one group of orders within its time window. The unique feature of a pull Kanban strategy is to determine when to start the first step and what sequence to operate in.

4. **A Numerical Experiment.** According to the actual situation of the warehouse layout and order fulfillment process of a large online supermarket in China, an example is constructed to verify the strategies proposed in this paper. The initial 10 parameters with its setting value in this experiment are shown in Table 2. The data set generator and algorithms are developed using the java language, running on Windows 10 (Xeon 2.40 GHz CPU, 32 GB memory). The numerical experiments results include 7 order batches with picking time, checking time, and packing time in each batch, as well as the total order completion time compared with wave picking, which is shown in Table 3. The experiment showed that the method in this paper can save the total order completion time and improve the efficiency of online supermarket order processing operations.

TABLE 2. The setting of the parameters

Parameters	Value	Parameters	Value	Parameters	Value
Number of orders	1000	SKUs for orders	Poisson distribution $\lambda = 1.5$	t^{pack}	1.1
Number of picking zones	3	t^{setup}	50 s	Delivery destination	$U([0,0],[60,120])$
$t^{checking}$	0.8	t^{pick}	7 s	Algorithms for order grouping	Systematic clustering
Delivery time	Random distribution $[6,72]$				

TABLE 3. Numerical calculation and comparative analysis

Batch	Picking time	Checking time	Packing time	Total order completion time		
				Pull order fulfillment	Wave picking	Improvement
1	410	363	376	9317	9984	6.68%
2	479	416	426			
3	517	492	503			
4	500	422	436			
5	447	433	445			
6	476	417	431			
7	454	431	443			

5. **Conclusions.** In this paper, we propose a pull order processing method combining JIT and grouping technology, which is built from the perspective of the overall process of order fulfillment. First, the rules for determining the similarity of orders are defined and orders are grouped. Then a processing flow for each group of orders has been regarded as an assembly line. Finally, the workload balance of the order processes is achieved through the optimization of order batching and sequencing. Following the philosophy of JIT assembly line with free beat, we confirm the beat and pull the right quantity of items to arrive in the right position at the right time. It ensures the items can flow through all the steps of order processing according to the established beat. The method achieves the goal of minimizing order processing time by reducing the idle time in each step of order processing, as well as the waiting time of goods in each buffer zone. We provide a new idea for order fulfillment and contribute to the exploration of new order management method.

Future research directions include designing a Kanban for pull order assembly line with dynamic beats and establishing a two-stage solution procedure for it.

Acknowledgments. This work is supported by the National Natural Science Foundation of China (72071078, 72101230).

REFERENCES

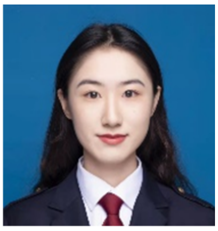
[1] G. P. Sharp, K. Il-Choe and S. Y. Chang, Small parts order picking: Analysis framework and selected results, *Material Handling'90. Progress in Material Handling and Logistics*, Springer, Berlin, Heidelberg, 1991.
[2] O. Briant, H. Cambazard, D. Cattaruzza, N. Catusse, A. L. Ladier and M. Ogier, An efficient and general approach for the joint order batching and picker routing problem, *European Journal of Operational Research*, vol.285, no.2, pp.497-512, 2020.
[3] J. Li, R. Huang and J. B. Dai, Joint optimisation of order batching and picker routing in the online retailer's warehouse in China, *International Journal of Production Research*, vol.55, no.2, pp.447-461, 2017.
[4] C. A. Valle and J. E. Beasley, Order batching using an approximation for the distance travelled by pickers, *European Journal of Operational Research*, vol.284, no.2, pp.460-484, 2020.
[5] C. Miao, G. Chen, C. Yan and Y. Y. Wu, Path planning optimization of indoor mobile robot based on adaptive ant colony algorithm, *Computers & Industrial Engineering*, vol.156, 2021.
[6] Y. Zhuang, Y. Sun, M. Yu and X. Hu, A path selection model for multimodal transportation problem of fresh fruits in online supermarket, *ICIC Express Letters*, vol.13, no.3, pp.201-208, 2019.
[7] R. Zahedi and R. Yusriski, Stepwise optimization for model of integrated batch production and maintenance scheduling for single item processed on flow shop with two machines in JIT environment, *Procedia Computer Science*, vol.116, pp.408-420, 2017.
[8] K. P. A. Nazar and V. M. Pillai, Mixed-model sequencing problem under capacity and machine idle time constraints in JIT production systems, *Computers & Industrial Engineering*, vol.118, pp.226-236, 2018.
[9] M. U. Garte, J. S. Golden and K. J. Dooley, Lean versus green: The impact of lean logistics on greenhouse gas emissions in consumer goods supply chains, *Journal of Purchasing and Supply Management*, vol.22, no.2, pp.98-109, 2016.
[10] U. Dombrowski and C. Malorny, Methodological approach for a process-orientated Lean Service implementation, *Procedia CIRP*, vol.73, pp.235-240, 2018.
[11] R. Chavez, W. Yu, M. Jacobs, B. Fynes, F. Wiengarten and A. Lecuna, Internal lean practices and performance: The role of technological turbulence, *International Journal of Production Economics*, vol.160, pp.157-171, 2015.
[12] W. Chung, S. Talluri and G. Kovács, Investigating the effects of lead-time uncertainties and safety stocks on logistical performance in a border-crossing JIT supply chain, *Computers & Industrial Engineering*, vol.118, pp.440-450, 2018.
[13] D. A. Saad, M. Masoud and H. Osman, Multi-objective optimization of lean-based repetitive scheduling using batch and pull production, *Automation in Construction*, vol.127, 2021.

[14] L. Yang, X. Zhang and M. Jiang, An optimal Kanban system in a multi-stage, mixed-model assembly line, *Journal of Systems Science and Systems Engineering*, vol.19, no.1, pp.36-49, 2010.

Author Biography

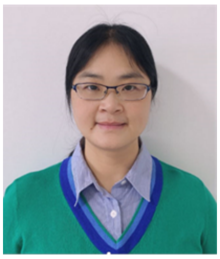
Yanxin Wang received the B.Sc. degree in technological economics, the M.Sc. degree in technological economics and management and the Ph.D. degree in management science and engineering from Harbin Institute of Technology, China in 1999, 2001 and 2011, separately.

Dr. Wang is currently a full-time associate professor at the Business College of Beijing Union University, China. His research interests include data analysis and intelligent decision. He has published over 30 papers in journals and conferences.



Tingli Wang received the B.Sc. degree in traffic management from Dalian Maritime University, China, in 2021. She is currently a graduate student at North China Electric Power University. Her main research interests include logistics management and scheduling optimization.

Yueyue Zhao received the B.Sc. degree in logistics management from Shenyang Institute of Engineering, China in 2020. She is a graduate student in North China Electric Power University, China and her interests include logistics scheduling optimization and supply chain management.



Minfang Huang received the B.Sc. degree in technological economics and the M.Sc. degree in system engineering from Harbin Institute of Technology, China in 1999 and 2001, separately; the Ph.D. degree in E-commerce and Logistics Management from Dalian University of Technology, China, in 2009.

Dr. Huang is currently a full-time associate professor at the School of Economics and Management, North China Electric Power University, China. Her research interests include logistics management and combination optimization. She has published over 40 papers in journals and conferences.