

Solving the order batching and sequencing problem with multiple pickers: A grouped genetic algorithm

Jose Alejandro Cano¹, Pablo Cortés², Emiro Antonio Campo³, Alexander Alberto Correa-Espinal⁴

¹Faculty of Economics and Administrative Sciences, Universidad de Medellín, Colombia

²Escuela Técnica Superior de Ingeniería, Seville University, Spain

³ESACS-Escuela Superior en Administración de Cadena de Suministro, Colombia

⁴Departamento de Ingeniería de la Organización, Universidad Nacional de Colombia, Colombia

Article Info

Article history:

Received Aug 21, 2020

Revised Oct 7, 2020

Accepted Oct 27, 2020

Keywords:

Grouped genetic algorithms
Heterogeneous load capacity
Multiple pickers
Order batching
Sequencing

ABSTRACT

This paper introduces a grouped genetic algorithm (GGA) to solve the order batching and sequencing problem with multiple pickers (OBSPMP) with the objective of minimizing total completion time. To the best of our knowledge, for the first time, an OBSPMP is solved by means of GGA considering picking devices with heterogeneous load capacity. For this, an encoding scheme is proposed to represent in a chromosome the orders assigned to batches, and batches assigned to picking devices. Likewise, the operators of the proposed algorithm are adapted to the specific requirements of the OBSPMP. Computational experiments show that the GGA performs much better than six order batching and sequencing heuristics, leading to function objective savings of 18.3% on average. As a conclusion, the proposed algorithm provides feasible solutions for the operations planning in warehouses and distribution centers, improving margins by reducing operating time for order pickers, and improving customer service by reducing picking service times.

This is an open access article under the [CC BY-SA](https://creativecommons.org/licenses/by-sa/4.0/) license.



Corresponding Author:

Jose Alejandro Cano
Faculty of Economics and Administrative Sciences
Universidad de Medellin
Carrera 87 # 30-65, Medellin, Colombia
Email: jacano@udem.edu.co

1. INTRODUCTION

The order picking problem is in charge of retrieving the set of items from storage locations to fulfill customer orders and deliver them on time, while pickers walk or operate a picking device through the warehouse [1, 2]. Manual order picking systems are prevalent in practice involving human operators at large scale due to the flexibility and autonomy offered by them [3] and the low labor costs in territories where automated systems are not viable [4, 5].

Concerning to achieve picking efficiency, the order batching groups customer orders into batches with a maximum fixed capacity [6], then the batches are assigned to a picking device and batch sequencing determines the picking scheduling and the completion time batches and customer orders [3, 7]. Therefore, the joint order batching and sequencing problem with multiple pickers (OBSPMP) is pivotal to enhance the efficiency and customer service [8-10]. One of the most important objectives in order picking systems is minimizing the maximal completion time (makespan), which allows reducing working time for order pickers, improving profit margins for warehouse operations, reducing delivery lead times and improving customer services [11]. There are only a few studies considering makespan as their objective to minimize the service

time for all possible batches [12], thus, minimize the makespan supports other objectives like minimizing total tardiness [13, 14].

The order batching problem is considered NP-Hard when the number of customer orders per batch is greater than two [15], which means it is impossible to obtain a polynomial-time solution for it [16], therefore, this type of problem requires to be solved using approximate methods such as metaheuristics [17, 18], among which the particle swarm optimization [19], ant colony optimization [20], genetic algorithms (GA) [21], among others, can be mentioned. Specifically, group-oriented genetic algorithms (GGA) support the successful application to grouping problems because critical information from the chromosome is preserved and is correctly transferred in the crossover operators [22]. Thus, making use of a group-oriented encoding scheme is sensitive to the group features of the OBSPMP where a gene represents a batch instead of a customer order. Moreover, the studies of [22-24] have implemented GGAs to solve the order batching problem in warehouses and order picking systems, for which they are used as a reference to jointly solve the OBSPMP in this study, which, unlike the models proposed in the literature, includes the assignment of batches to multiple picking devices with heterogeneous loading capacity.

To solve the OBSPMP, Henn [8], Scholz *et al.* [25], and Van Gils *et al.* [10] have proposed variable neighborhood descent and variable neighborhood search approaches to minimize tardiness and total order pick time. Matusiak *et al.* [26] proposed an adaptive large neighborhood search algorithm considering pickers with diverse skills to minimize the total order processing time. Zhang *et al.* [9] proposed a rule-based approach to minimize the maximum completion time of all batches for online environments. However, population-based metaheuristics like genetic algorithms have not been found in the literature to solve the OBSPMP, as well as no OBSPMP models considering picking devices with heterogeneous load capacity, which is a characteristic of modern warehouses and distribution centers.

Therefore, this paper aims to present for the first time the application of a GGA for the OBSPMP, considering picking devices with a heterogeneous load capacity to minimize the maximum completion time (makespan). The remainder of this paper is organized as follows. Section 2 introduces the features and assumptions for the OBSPMP. In section 3, a GGA to solve the OBSPMP is presented. Section 4 introduces the experiments to determine the performance of the proposed model. Section 5 compares the performance of the GGA with six benchmarks, showing savings for the makespan. Conclusions are discussed in section 6.

2. OBSPMP FEATURES AND ASSUMPTIONS

The main features and assumptions of the order picking system for the OBSPMP are the following; In i) The order picking problem is based on low-level picker-to-parts systems, ii) the warehouse configuration is based on a parallel-aisle single-block warehouse, iii) each customer order is composed of several items, iv) multiple picking devices with heterogeneous load capacity and constant horizontal speed are allowed, and batch size does not exceed the capacity of picking devices (v) each batch is assigned to a picking device and it follows the S-shape routing heuristic to retrieve all the items of the batch, vi) the completion time of a batch is equal to the completion time of the orders assigned to it, vii) the service time of a batch is equal to the travel time, measured as the traveled distance divided the speed of the picking device and viii) a picking device can handle the next batch only when a previous batch is finished. The warehouse configuration is based on a parallel-aisle single-block warehouse as described in Figure 1, where storage locations, aisles width and cross-aisles dimensions are illustrated, as well as an example of the s-shape routing strategy used to solve the picker routing problem.

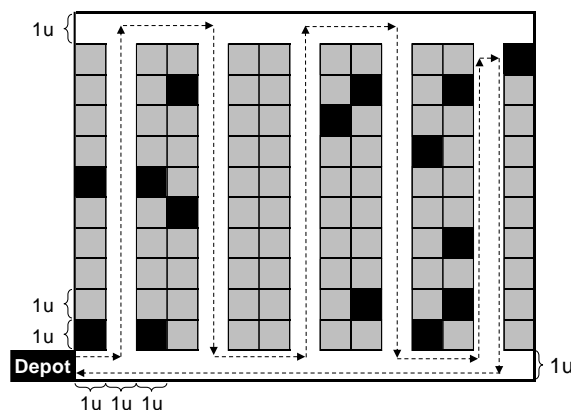


Figure 1. Warehouse dimensions and S-shape routing strategy

Given a set of batches $b \in B$, a set of customer orders $o \in O$, a set of storage locations $i, j \in L$, a subset of storage locations $S \subset L$, a set of positions to schedule a batch in a picking device, and a set of picking devices $e \in E$, the mathematical formulation of the OBSPMP is described as follows:

Parameters

$$\begin{aligned} w_o &= \text{Capacity required for order } o \\ t_{ij}^e &= \text{Travel time from position } i \text{ to } j \text{ for picking device } e \\ C_e &= \text{Maximum capacity of picking device } e \\ s_{io} &= \begin{cases} 1 & \text{if an item of the order } o \text{ is retrieved from position } i \\ 0 & \text{otherwise} \end{cases} \end{aligned}$$

Decision variables

$$\begin{aligned} X_o^b &= \begin{cases} 1 & \text{if order } o \text{ is assigned to batch } b \\ 0 & \text{otherwise} \end{cases} \\ V_e^b &= \begin{cases} 1 & \text{if batch } b \text{ is assigned to the picking device } e \\ 0 & \text{otherwise} \end{cases} \\ Y_{ij}^b &= \begin{cases} 1 & \text{if batch } b \text{ visits } i \text{ just after visiting } j \\ 0 & \text{otherwise} \end{cases} \\ Z_i^b &= \begin{cases} 1 & \text{if batch } b \text{ visits } i \text{ to retrieve an item} \\ 0 & \text{otherwise} \end{cases} \\ U_k^b &= \begin{cases} 1 & \text{if batch } b \text{ is scheduled in position } k \\ 0 & \text{otherwise} \end{cases} \\ R_{ke}^b &= \begin{cases} 1 & \text{if batch } b \text{ is assigned to position } k \text{ in picking device } e \\ 0 & \text{otherwise} \end{cases} \\ CT_k^e &= \text{Completion time for a batch scheduled in position } k \text{ in picking device } e \\ c_o &= \text{Completion time for order } o \end{aligned}$$

$$\text{MinMax}_{o \in O} \{C_o\} \tag{1}$$

$$\sum_{b \in B} R_{ke}^b \leq 1 \quad \forall k \in K, e \in E \tag{2}$$

$$\sum_{e \in E} \sum_{k \in K} \sum_{b \in B} X_o^b * R_{ke}^b = 1 \quad \forall o \in O \tag{3}$$

$$\sum_{o \in O} w_o * X_o^b * R_{ke}^b \leq C_e \quad \forall b \in B, e \in E, k \in K \tag{4}$$

$$Z_i^b \geq s_{io} * X_o^b \quad \forall b \in B, o \in O, i \in L \tag{5}$$

$$\sum_{j \in L, j \neq i} Y_{ij}^b = Z_i^b \quad \forall b \in B, i \in L \tag{6}$$

$$\sum_{i \in L, i \neq j} Y_{ij}^b = Z_j^b \quad \forall b \in B, j \in L \tag{7}$$

$$\sum_{i \in S, j \in L \setminus S} Y_{ij}^b \geq Z_i^b \quad \forall b \in B, S \subset L \tag{8}$$

$$\sum_{b \in B} (R_{1e}^b * \sum_{i \neq j \in L} \sum_{j \neq i \in L} t_{ij}^e * Y_{ij}^b) \leq CT_1^e \quad \forall e \in E \tag{9}$$

$$CT_{k-1}^e + \sum_{b \in B} (R_{ke}^b * \sum_{i \neq j \in L} \sum_{j \neq i \in L} t_{ij}^e * Y_{ij}^b) \leq CT_k^e \quad \forall e \in E, k \in K \setminus \{1\} \tag{10}$$

$$c_o = \sum_{e \in E} \sum_{k \in K} \sum_{b \in B} X_o^b * R_{ke}^b * CT_k^e \quad \forall o \in O \tag{11}$$

$$CT_k^e, c_o \geq 0 \quad \forall k \in K, e \in E, o \in O \tag{12}$$

$$X_o^b, Y_{ij}^b, Z_i^b, R_{ke}^b \in \{0,1\} \quad \forall k \in K, e \in E, b \in B, o \in O, i, j \in L \tag{13}$$

The objective function in (1) minimizes the maximum completion time (makespan). Constraints in (2) ensures that at most a batch is scheduled in a specific position in a picking device. Constraints in (3) guarantees the assignment of each customer order to a batch. In (4) limits the capacity of the batches assigned to each picking device according to the loading capacity of the picking devices. Constraints in (5-8) embody the TSP formulation. In (9) and (10) measures the completion time for the batch scheduled in positions 1 and

k in each picking device respectively. In (11) calculates the completion time of an order as the completion time of the batch where it is assigned. Constraints in (12) and (13) determine the non-negativity and domain of the variables. The order batching problem is considered NP-Hard when the number of customer orders per batch is greater than two [15], therefore any extension of this problem is considered NP-Hard as well [13, 27, 28]. Then, the OBSMP is considered an NP-hard problem, so it cannot be solved using exact solution methods at least for large instances [25], thus metaheuristics like genetic algorithms can provide satisfactory solutions in short computing times for combinatorial problems related to logistics and operations management [29, 30]. In this study, we introduce a GGA to provide high-quality solutions in short computing times for the OBSMP, satisfying the operating requirements of real warehouses.

3. GROUPED GENETIC ALGORITHMS

The proposed group-oriented encoding scheme represents the assignment of orders to batches and the sequencing of batches in picking devices. Due to each gene represents a batch in a picking device; the chromosomes are of variable length. In the encoding scheme shown in Figure 2, each gene is composed of the number of the picking device, the batch assigned to the picking device, and the customer orders grouped in the batch.

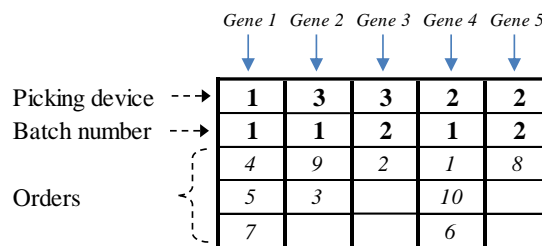


Figure 2. GGA encoding scheme

To produce the initial population of size P , we follow an order group procedure that uses an order pool to place orders that have not yet been assigned to a batch. For each gene, a picking device is randomly chosen, and a batch is opened for this picking device, then, an order is randomly chosen from the order pool and is assigned to the open batch. Another order is randomly chosen from the order pool and is assigned to the opened batch if the capacity requirement of the order is less than or equal to the available capacity of the batch, otherwise, another order is randomly chosen from the order pool. If none of the orders in the order pool fit on the open batch, then the batch is closed. The order group procedure ends when all customer orders are grouped into batches. The fitness function represents the objective function of the OBSMP, which is minimizing total completion time, so the maximum completion time or total completion time (makespan) is identical to the time required to collect a given set of customer orders. Given the set of orders, $o = 1, \dots, O$, let C_o be the completion time of the order o , so the fitness function to minimize with the GGA is $Max_{oco}\{C_o\}$.

The selection of parental chromosomes for the crossover operator is based on the linear selection ranking method; this method assigns the highest selecting probability to chromosomes with better performance, promoting the crossing between parents with high-quality genetic information. Then, the number of pairs of parents is determined according to the crossover rate (Cr) and parents are chosen using the roulette wheel selection. Figure 3 illustrates the crossover operator that begins with the selection of two crossing points delimiting the crossing section (step 1). The exchange of the crossing section between each pair of parents may lead to infeasible solutions when an order appears twice on a chromosome (step 2). A correction mechanism is applied to fix infeasible offspring, removing old genes containing orders appearing on the new genes, and then updates the sequencing of batches in each picking device (step 3). Orders that have not yet been assigned to the chromosome become part of the order pool, and the order group procedure is applied to complete each chromosome (step 4).

The survival mechanism ensures that elite individuals prevail in each generation according to the survival rate (Sr). The immigration rate (Ir) defines the number of new individuals to create using the order group procedure to provide diversity to the population and prevent premature convergence. The mutation operator is implemented in a number of individuals defined by the mutation rate (Mr). The mutation procedure starts selecting two genes randomly, then the selected genes are removed and the orders of these genes become available into the order pool. The sequence of batches in each picking device is updated and the remaining orders are assigned to new genes using the order group procedure. Lastly, when G generations

are satisfied the genetic algorithm will stop. Figure 4 shows the flowchart of the GGA summarizing the steps of the evolutionary process.

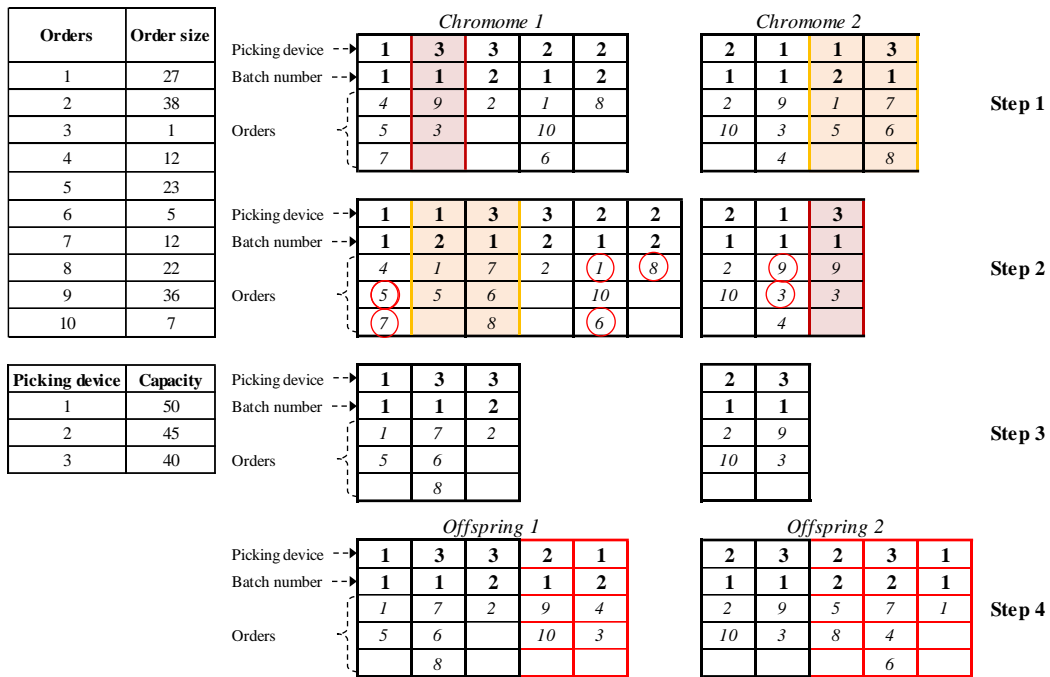


Figure 3. Crossover operator for the GGA

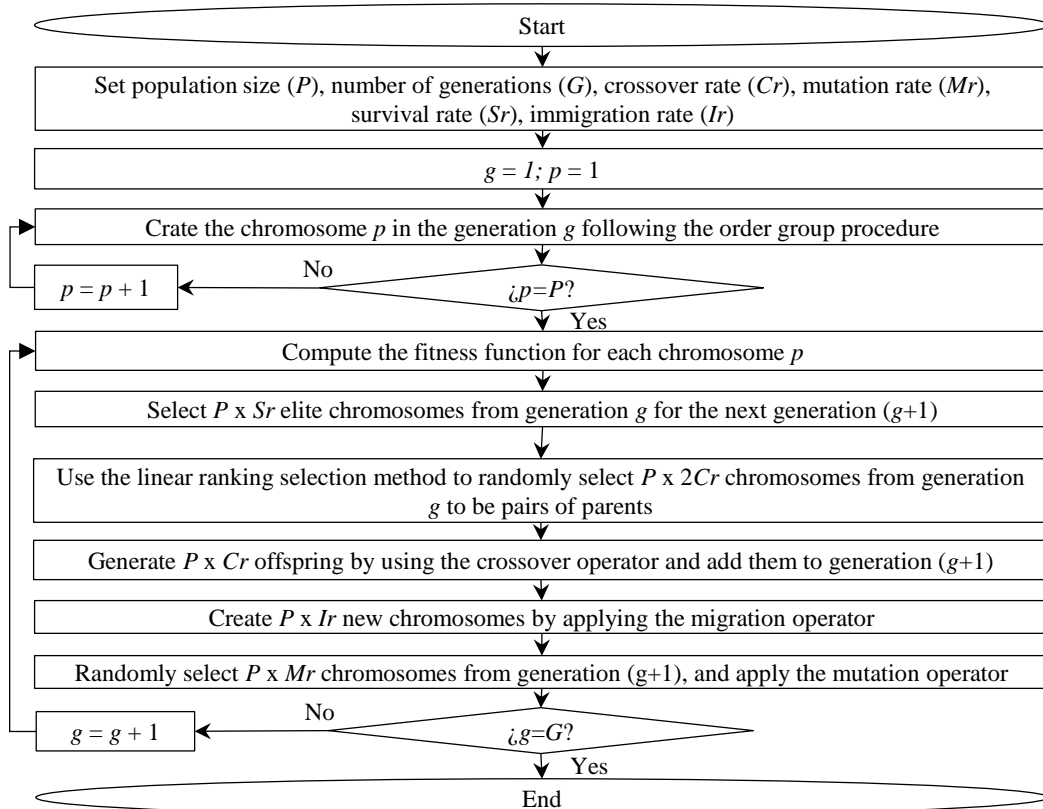


Figure 4. The flowchart of the proposed GGA

4. EXPERIMENTS

In order to determine the performance of the GGA, several experiments are carried out. The results of the GGA are compared with six benchmark rules called FCFS-LH, FCFS-HL, SLOS-LH, SLOS-HL, LSOS-LH, and LSOS-HL. The abbreviation LH means that batches are first assigned to the picking device 1, then to the picking device 2 and so on until the last picking device, after which, the next batches are assigned to the picking device 2 and so on until assigning all batches to picking devices. The abbreviation HL means that batches are assigned first to the last picking device, then to the penultimate picking device and so on until the first picking device, after which, the next batches are assigned to the penultimate picking device and so on until assigning all batches to picking devices. Moreover, FCFS means that the assignment of orders to batches is based on the first-come-first-served rule. SLOS means that the assignment of orders to batches is performed from the smallest-sized order to the largest-sized order. Likewise, LSOS means that the assignment of orders to batches is performed from the largest-sized order to the smallest-sized order. Figure 5 illustrates the solutions provided by the proposed benchmark heuristics for the OBSPMP.

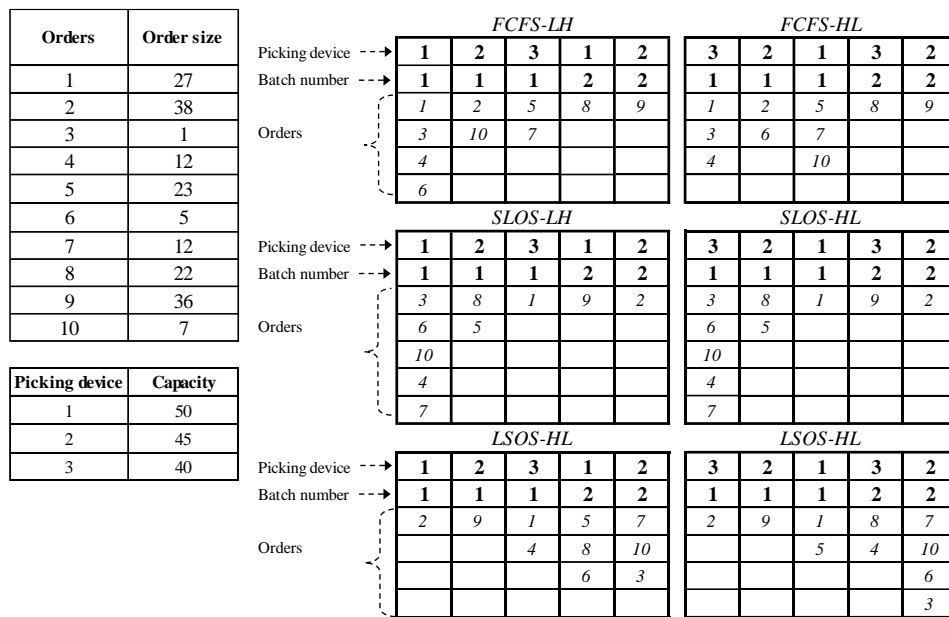


Figure 5. Benchmark heuristics for the OBSPMP

The experiments are configured with the parameters described in Table 1. By combining different values for O , m , and w , 24 problem classes are generated, and 10 instances are calculated for each problem. Therefore, we provide 240 instances in total. The parameters of the GGA are $P=20$, $G=40$, $Cr=0.85$, $Mv=0.05$, $St=0.1$, and $It=0.05$. The experiments are carried out on an Intel Core i5-2300 CPU at 2.8 GHz and 8 GB RAM. The algorithm is implemented with Visual Basic.

Table 1. Order picking simulation environments

| Parameters | Values |
|--|----------------------|
| Total number of customer orders (O) | 10, 30, 50 |
| Number of items per order (m) | U[1, 5], U[5, 15] |
| Warehouse layout (storage locations) (w) | 400, 900, 1250, 2000 |
| Routing strategy | S-shape |
| Storage policy | Random-based |
| Capacity of the picking device 1, 2, 3 | 30, 40, 50 |
| Speed of the picking devices | 45 m/min |

5. RESULTS AND DISCUSSION

The results of the GGA are compared with the six benchmarks, calculating savings for the total completion time. As shown in Table 2, GGA saves on average between 14.3% and 23.5% of total completion time when compared to the benchmark heuristics. The proposed GGA indeed offers savings of up to 40.7%

in total completion time when compared to SLOS-HL when considering 10 orders, items per order between 1 and 5, and a layout with 2,000 storage positions.

For all the evaluated instances, the GGA saves on average 18.3% the total completion time when compared to the six benchmarks; in this manner, the proposed algorithm improves the efficiency of order picking significantly. From the particular results of Table 3, GGA generates average savings of 26.6% when $O=10$. For instances when $O=30$ and $O=50$, GGA provides average savings of 15.5% and 12.7% respectively. Although minimizing the total picking time is not the objective function in the proposed algorithm, for the evaluated instances the GGA offers average savings of 4.6% on picking time when compared to the benchmark heuristics, so the GGA in addition to improving customer service through the completion time also improves the operating costs of the order picking. Furthermore, the average computing times of the algorithm when $O=10$, $O=30$, and $O=50$ are 0.04, 1.60, and 9.89 minutes respectively, which are viable for the daily operations planning of warehouses and distribution centers.

Table 2. Total completion time savings for GGA

| | FCFS-LH | FCFS-HL | SLOS-LH | SLOS-HL | LSOS-LH | LSOS-HL |
|---------|---------|---------|---------|---------|---------|---------|
| Average | 14.3% | 22.4% | 16.3% | 23.5% | 14.4% | 18.8% |
| Maximum | 27.9% | 38.7% | 35.5% | 40.7% | 27.9% | 33.4% |
| Minimum | 5.3% | 9.6% | 5.7% | 12.3% | 2.6% | 5.4% |

Table 3. Average saving on total completion time for GGA

| O | FCFS-LH | FCFS-HL | SLOS-LH | SLOS-HL | LSOS-LH | LSOS-HL |
|-----|---------|---------|---------|---------|---------|---------|
| 10 | 20,94% | 30,21% | 25,66% | 30,62% | 23,75% | 28,68% |
| 30 | 11,15% | 20,64% | 12,54% | 21,45% | 11,32% | 15,65% |
| 50 | 10,82% | 16,48% | 10,63% | 18,33% | 8,15% | 12,02% |

6. CONCLUSION

This study addresses the order batching and sequencing problem with multiple pickers (OBSPMP) to minimize the makespan. We dealt with the assignment of orders to batches and with the assignment of batches to picking devices with heterogeneous load capacity. By means of numerical experiments, it was proved that the GGA offers solutions superior to those provided by rule-based heuristics. Consequently, the GGA provides average savings of 18.3% on makespan compared to six widely used heuristics in real warehouse environments, and the solutions are obtained in feasible computational times that allow their application in daily warehouse operations. Therefore, implementing these solutions can improve warehouse performance significantly by improving profit margins, reducing operating time for the order pickers, improving customer service and reducing picking service times. Future research could address the joint order batching, sequencing, assignment and picker routing problem considering 3D warehouses, online customer orders and picking devices with heterogeneous velocity.

REFERENCES

- [1] J. A. Cano, Alexander Alberto Correa-Espinal, and Rodrigo Andrés Gómez-Montoya, "An evaluation of picking routing policies to improve warehouse efficiency," *International Journal of Industrial Engineering and Management*, vol. 8, no. 4, pp. 229-238, 2017.
- [2] E. Ardjmand, Omid Sanei Bajgiran, and Eyad Youssef, "Using list-based simulated annealing and genetic algorithm for order batching and picker routing in put wall based picking systems," *Applied Soft Computing Journal*, vol. 75, pp. 106-119, 2019.
- [3] S. Henn and V. Schmid, "Metaheuristics for order batching and sequencing in manual order picking systems," *Computers and Industrial Engineering*, vol. 66, no. 2, pp. 338-351, 2013.
- [4] J. A. Cano, Alexander Correa-Espinal, and Rodrigo Gómez-Montoya., "Using genetic algorithms for order batching in multi-parallel-aisle picker-to-parts systems," *International Journal of Applied Decision Sciences*, vol. 13, no. 4, pp. 417-434, 2020.
- [5] J. A. Cano, "Order Picking Optimization Based on a Picker Routing Heuristic: Minimizing Total Traveled Distance in Warehouses," *Handbook of Research on the Applications of International Transportation and Logistics for World Trade*, G. Ç. Ceyhan, Ed. PA, USA: IGI Global, pp. 74-96, 2020.
- [6] B. Menéndez, et al., "General Variable Neighborhood Search for the Order Batching and Sequencing Problem," *European Journal of Operational Research*, vol. 263, no. 1, pp. 82-93, 2017.
- [7] M. Bustillo, et al., "An algorithm for batching, sequencing and picking operations in a warehouse," *International Conference on Industrial Engineering and Systems Management, IEEE IESM 2015*, 2015, pp. 842-849.

- [8] S. Henn, "Order batching and sequencing for the minimization of the total tardiness in picker-to-part warehouses," *Flexible Services and Manufacturing Journal*, vol. 27, no. 1, pp. 86-114, 2015.
- [9] J. Zhang, et al., "On-line order batching and sequencing problem with multiple pickers: A hybrid rule-based algorithm," *Applied Mathematical Modelling*, vol. 45, no. 1, pp. 271-284, 2017.
- [10] T. Van Gils, et al., "Formulating and Solving the Integrated Batching, Routing, and Picker Scheduling Problem in a Real-life Spare Parts Warehouse," *European Journal of Operational Research*, vol. 277, no. 3, pp. 814-830, 2019.
- [11] S. Henn, "Algorithms for on-line order batching in an order picking warehouse," *Computers and Operations Research*, vol. 39, no. 11, pp. 2549-2563, 2012.
- [12] E. Ardjmand, et al., "Minimizing order picking makespan with multiple pickers in a wave picking warehouse," *International Journal of Production Economics*, vol. 206, no. 1, pp. 169-183, 2018.
- [13] T. L. Chen, et al., "An efficient hybrid algorithm for integrated order batching, sequencing and routing problem," *International Journal of Production Economics*, vol. 159, pp. 158-167, 2015.
- [14] B. Menéndez, et al., "Parallel variable neighborhood search for the min-max order batching problem," *International Transactions in Operational Research*, vol. 24, no. 3, pp. 635-662, 2017.
- [15] N. Gademann and S. van de Velde, "Order batching to minimize total travel time in a parallel-aisle warehouse," *IIE Transactions*, vol. 37, no. 1, pp. 63-75, 2005.
- [16] M. M. Tavakol and A. Sami, "Particle swarm optimization in solving Vehicle Routing Problem," *Bulletin of Electrical Engineering and Informatics (BEEI)*, vol. 2, no. 4, pp. 252-257, 2013.
- [17] S. Kalantari and M. S. Abadeh, "GASA-JOSH: A Hybrid Evolutionary-Annealing Approach for Job-Shop Scheduling Problem," *Bulletin of Electrical Engineering and Informatics (BEEI)*, vol. 2, no. 2, pp. 132-140, 2013.
- [18] T. L. Duong, et al., "Application of a new constraint handling method for economic dispatch considering electric market," *Bulletin of Electrical Engineering and Informatics (BEEI)*, vol. 9, no. 4, pp. 1542-1549, 2020, doi: 10.11591/eei.v9i4.2351.
- [19] S. Sabikan, S. W. Nawawi, and Naa Aziz., "Modelling of time-to collision for unmanned aerial vehicle using particles swarm optimization," *IAES International Journal of Artificial Intelligence (IJ-AI)*, vol. 9, no. 3, pp. 488-496, 2020.
- [20] A. S. Girsang, et al., "Fast Ant Colony Optimization for Clustering," *Indonesian Journal of Electrical Engineering and Computer Science (IJECS)*, vol. 12, no. 1, pp. 78-86, 2018.
- [21] Z. A. Ali, et al., "An enhanced hybrid genetic algorithm for solving traveling salesman problem," *Indonesian Journal of Electrical Engineering and Computer Science (IJECS)*, vol. 18, no. 2, pp. 1035-1039, 2020.
- [22] S. Koch and G. Wäscher, "A Grouping Genetic Algorithm for the Order Batching Problem in Distribution Warehouses," *Journal of Business Economics*, vol. 86, no. 1, pp. 131-153, 2016.
- [23] M. Mutingi and C. Mbohwa, "Optimizing Order Batching in Order Picking Systems: Hybrid Grouping Genetic Algorithm," *Grouping Genetic Algorithms*, pp. 121-140, 2017.
- [24] J. C.-H. Pan, et al., "Order batching in a pick-and-pass warehousing system with group genetic algorithm," *Omega*, vol. 57, pp. 238-248, 2015.
- [25] A. Scholz, et al., "Order picking with multiple pickers and due dates-Simultaneous solution of order batching, batch assignment and sequencing, and picker routing problems," *European Journal of Operational Research*, vol. 263, no. 2, pp. 461-478, 2017.
- [26] M. Matusiak, et al., "Utilizing individual picker skills to improve order batching in a warehouse," *European Journal of Operational Research*, vol. 263, no. 3, pp. 888-899, 2017.
- [27] J. A. Cano, "Formulations for joint order picking problems in low-level picker-to-part systems," *Bulletin of Electrical Engineering and Informatics (BEEI)*, vol. 9, no. 2, pp. 836-844, 2020.
- [28] J. A. Cano, Alexander A. Correa-Espinal, and Rodrigo Andrés Gómez-Montoya., "Mathematical programming modeling for joint order batching, sequencing and picker routing problems in manual order picking systems," *Journal of King Saud University-Engineering Sciences*, vol. 32, no. 3, pp. 219-228, 2019.
- [29] S. S. Sadiq, et al., "Solving multi-objective master production schedule problem using memetic algorithm," *Indonesian Journal of Electrical Engineering and Computer Science (IJECS)*, vol. 18, no. 2, pp. 938-945, 2020, doi: 10.11591/ijeecs.v18.i2.pp938-945.
- [30] A. K. Ariyani, Wayan Firdaus Mahmudy, and Yusuf Priyo Anggodo, "Hybrid Genetic Algorithms and Simulated Annealing for Multi-trip Vehicle Routing Problem with Time Windows," *International Journal of Electrical and Computer Engineering (IJECE)*, vol. 8, no. 6, pp. 4713-4723, 2018.

BIOGRAPHIES OF AUTHORS



Jose Alejandro Cano obtained his PhD in Engineering, Industry and Organizations from the National University of Colombia, and obtained his Master in Administrative Engineering and Bachelor in Industrial Engineering from the same university. He is working as an Associate Professor at the University of Medellin, Colombia. He obtained his PhD in Engineering, Industry and Organizations. His research interest includes order picking systems, logistics, optimisation, and metaheuristics.



Pablo Cortés is a Cathedric Professor at the Seville University (Spain) where he belongs to the Department of Industrial Organization and Business Management II. His research interest includes computational intelligence, operations research, transport & logistics, vertical transportation, and utility networks.



Emiro Antonio Campo obtained his Master in Industrial Engineering from the National University of Colombia, Colombia and obtained his Bachelor in Industrial Engineering from the same university. He is working as an operations and logistics consultant for different manufacturing and service companies. His research interest includes operations management, logistics, optimisation, and metaheuristics.



Alexander Alberto Correa-Espinal is a Titular Professor at the National University of Colombia, Colombia. He obtained his PhD from the Universitat Politècnica de Catalunya, Spain. He obtained his MSc in Industrial Engineering from Universidad de los Andes, Colombia. His research interest includes optimisation, operations research, design of experiments, and supply chain management.