

## Improving cross-docking operations for consumer goods sector using metaheuristics

Rodrigo Andrés Gómez-Montoya<sup>1</sup>, Jose Alejandro Cano<sup>2</sup>, Emiro Antonio Campo<sup>3</sup>, Fernando Salazar<sup>4</sup>

<sup>1</sup>Faculty of Engineering, Politécnico Colombiano Jaime Isaza Cadavid, Colombia

<sup>2</sup>Faculty of Economics and Administrative Sciences, Universidad de Medellín, Colombia

<sup>3</sup>ESACS-Escuela Superior en Administración de Cadena de Suministro, Colombia

<sup>4</sup>Faculty of Economics and Administrative Sciences, Pontificia Universidad Javeriana, Colombia

### Article Info

#### Article history:

Received Apr 5, 2020

Revised Jun 2, 2020

Accepted Jul 13, 2020

#### Keywords:

Consumer goods sector

Cross-docking

Distribution center

Particle swarm optimization

Simulated annealing

### ABSTRACT

This paper aims to model a consumer goods cross-docking problem, which is solved using metaheuristics to minimize makespan and determine the capacity in terms of inbound and outbound docks. The consumer-goods cross-docking problem is represented through inbound and outbound docks, customer orders (products to be delivered to customers), and metaheuristics as a solution method. Simulated annealing (SA) and particle swarm optimization (PSO) are implemented to solve the cross-docking problem. Based on the results of statistical analysis, it was identified that the two-way interaction effect between inbound and outbound docks, outbound docks and items, and items and metaheuristics are the most statistically significant on the response variable. The best solution provides the minimum makespan of 973.42 minutes considering nine inbound docks and twelve outbound docks. However, this study detected that the combination of six inbound docks and nine outbound docks represents the most efficient solution for a cross-docking design since it reduces the requirement of docks by 28.6% and increases the makespan by only 4.2% when compared to the best solution, representing a favorable trade-off for the cross-docking platform design.

*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 Medellín,

Carrera 87 # 30-65, Medellín, Colombia.

Email: jacano@udem.edu.co

## 1. INTRODUCTION

The consumer goods industry is classified as a key sector in the food supply chain since it allows meeting the market demand effectively. The consumer goods industry is characterized by high volume transactions, frequent and split deliveries, and by handling perishable products. Therefore, logistics systems for the consumer goods supply chain must be agile, reliable, and effective [1, 2]. Cross-docking is a logistics process in which products are unloaded from inbound trucks, sorted, consolidated and transferred to outbound docks, and loaded directly into outbound trucks [3, 4]. Therefore, freight distribution operations like cargo transshipment, consolidation, deconsolidation, and other logistics activities are frequently used in the cross-docking process, which eliminates the need for storage processes and prevent congestion within the logistics system [5-7]. Consequently, this process is implemented to adopt an agile supply chain approach, representing a key process for the consumer goods sector due to its logistics characteristics.

Assadi and Bagheri [8] presented a study addressing a differential evolution algorithm and population-based simulated annealing for a truck scheduling problem in multiple-door cross-docking systems. Nikolopoulou *et al.* [9] developed an analysis of transporting products between locations in a

*Journal homepage:* <http://beei.org>

logistic network, including a comparative analysis between cross-docking versus direct-shipping. Wisittipanich and Hengmeechai [10] tackle a truck scheduling problem in a multi-door cross-docking terminal, which is solved using a PSO. Goodarzi and Zegordi [11] addressed a location-routing problem for cross-docking networks, which is solved by implementing a biogeography-based optimization algorithm to increase the logistic process efficiency. Maknoon *et al.* [12] developed a model for optimizing transshipment operations for a cross-docking process. Based on the existing literature, no studies were detected addressing directly cross-docking processes for the consumer goods sector; however, several studies focused on general economic sectors.

Therefore, this paper aims to model and solve a consumer goods cross-docking problem to minimize makespan and determine the capacity in terms of inbound and outbound docks, which is a relevant objective for logistics and cross-docking terminals [13]. Inbound and outbound docks and customer orders are represented for the consumer goods cross-docking problem, and this problem is solved using SA, PSO, and an empirical rule. The remainder of this paper is divided into the following sections. Section 2, introduces the description and formulation of the cross-docking problem. Section 3 presents the metaheuristics (SA and PSO) and empirical rule to solve the cross-docking problem. In section 4, experimental simulations are planned by the full factorial design of experiments (DOE). Section 5 shows the results and discussions. Finally, conclusions and future research are developed in section 6.

## 2. PROBLEM FORMULATION

A minimum makespan cross-docking problem design is addressed in this study, representing the total execution time of cross-docking operations [14], to reduce operational time and logistics costs in the supply chain management [15]. The cross-docking process is composed of operations, resources, and logistics infrastructure. Operations such as truck/trailer loading and unloading as well as transshipment activity are performed in the cross-docking process to meet customer demand effectively. These are key operations for the consumer goods sector in which frequent and small deliveries are processed, handling multiple items within each customer order.

Information and communication technologies (ICT) such as warehouse management system (WMS) [16, 17], enterprise resourcing planning (ERP) [18, 19] and electronic data interchange (EDI) [20] are used to manage information about customer orders (due date, amount of products to be delivered and logistical conditions), origin and destination points, among other information for cross-docking operations. Besides, material handling equipment (MHE) and labor represent the resources that execute those operations, while loading and unloading docks and logistics platforms denote the key infrastructure for the cross-docking process. Figure 1 shows the operations, resources, and infrastructure considered for the cross-docking problem.

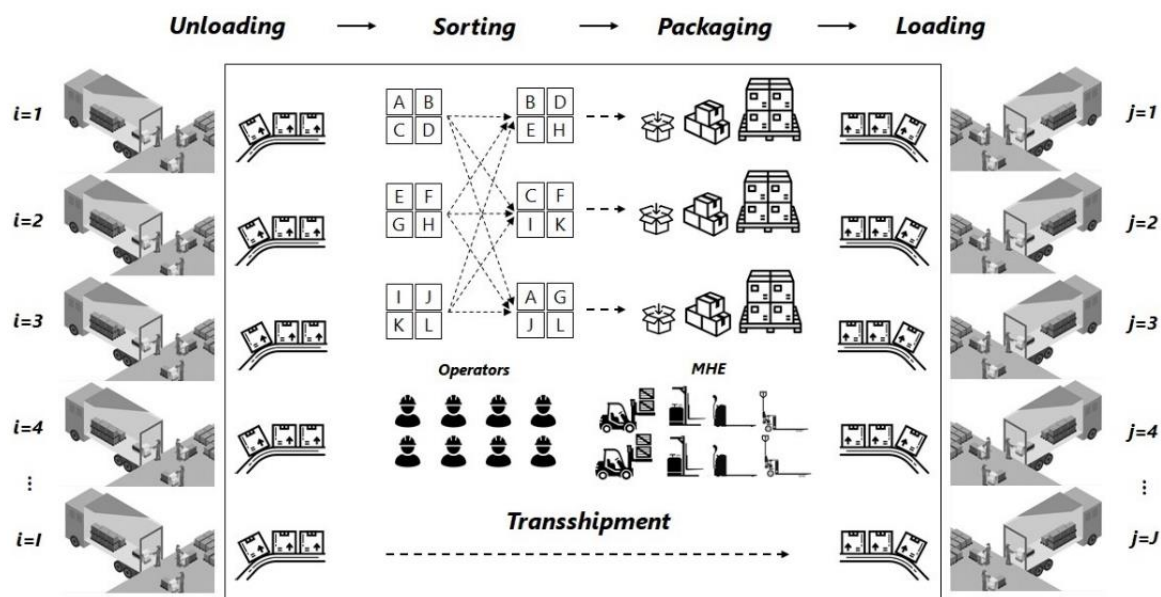


Figure 1. Operations, resources and infrastructure for the cross-docking problem

The cross-docking platform is composed by  $I$  inbound docks and  $J$  outbound docks. Additionally, operation platform is used to perform operations such truck loading and unloading, sorting, packaging, and transshipment, and these operations are supported by resources like operators, MHEs, and inbound and outbound docks (infrastructure). On the other hand, EDI allows receiving customer orders, while WMS classifies such orders based on due dates, number of items, and quantities to be delivered. Additionally, WMS compares available and incoming stock against product quantities requested by customer orders, guaranteeing satisfactory service levels. Therefore, customer orders with available stock are scheduled to the cross-docking process, assigning trucks to inbound and outbound docks, and assigning MHE and operators to perform cross-docking operations to minimize makespan.

In order to formulate the cross-docking problem as a mixed-integer linear programming (MILP) model, this study considers that inbound trucks should be unloaded completely to leave the inbound dock; outbound trucks should be loaded completely to leave the outbound dock; the cross-docking platform considers no capacity limit for temporary storage; operational time for loading and unloading items is assumed constant, as well as the transfer time from inbound docks to outbound docks; information about inbound and outbound trucks and the number of items is known in advance. Based on the studies of [21], indices, sets, parameters, decision variables, objective function, and constraints are represented as follows:

*Sets and indices:*

- $i$  =index for inbound trucks ( $i = 1, \dots, I$ )
- $i'$  =index for inbound trucks ( $i = 1, \dots, I$ )
- $j$  =index for outbound trucks ( $j = 1, \dots, J$ )
- $j'$  =index for outbound trucks ( $j = 1, \dots, J$ )
- $m$  =index for inbound docks ( $m = 1, \dots, M$ )
- $n$  =index for outbound docks ( $n = 1, \dots, N$ )
- $k$  =index for items ( $k = 1, \dots, K$ )

*Parameters:*

- $r_{ik}$  =quantity of item  $k$  to be unloaded from inbound truck  $i$
- $s_{jk}$  =quantity of item  $k$  to be loaded to outbound truck  $j$
- $DT$  =delay in arrival time for inbound and outbound trucks
- $t_{mn}$  =transshipment time from inbound  $m$  dock to outbound dock  $n$  within cross-docking platform.
- $G$  a =very large number
- $A_i$  =arrival time of inbound truck  $i$
- $A_j$  =arrival time of outbound truck  $j$
- $t_{load}$  =loading time for an item
- $t_{unload}$  =unloading time for an item

*Decision variables:*

- $X_{ijk}$  = quantity of item  $k$  transferred from inbound truck  $i$  to outbound truck  $j$
- $V_{ij} = \begin{cases} 1, & \text{if any SKU is transferred from inbound truck } i \text{ to outbound truck } j; \\ 0, & \text{otherwise.} \end{cases}$
- $Y_{im} = \begin{cases} 1, & \text{if inbound truck } i \text{ is assigned to inbound dock } m; \\ 0, & \text{otherwise.} \end{cases}$
- $Z_{jn} = \begin{cases} 1, & \text{if outbound truck } j \text{ is assigned to outbound dock } n; \\ 0, & \text{otherwise.} \end{cases}$
- $U_{ijmn} = \begin{cases} 1, & \text{if any SKU is transferred from truck } i \text{ assigned to} \\ & \text{inbound dock } m \text{ to truck } j \text{ assigned to outbound dock } n; \\ 0, & \text{otherwise.} \end{cases}$
- $P_{ii'} = \begin{cases} 1, & \text{if inbound truck } i \text{ precedes inbound truck } i'; \\ 0, & \text{otherwise.} \end{cases}$
- $Q_{jj'} = \begin{cases} 1, & \text{if outbound truck } j \text{ precedes outbound truck } j'; \\ 0, & \text{otherwise.} \end{cases}$
- $c_i$  =start time of inbound truck  $i$
- $C_i$  =finish time of inbound truck  $i$
- $l_j$  =start time of outbound truck  $j$
- $L_j$  =finish time of outbound truck  $j$

Objective function (1) minimizes total earliness and tardiness. Constraint (2) ensures the total quantity of item  $k$  from inbound truck  $i$  is equivalent to the quantity sent to outbound trucks. Constraint (3) ensures the total quantity of item  $k$  sent to outbound truck  $j$  is equal to the quantity released from inbound

trucks. Constraint (4) ensures the transference of items with a relationship between inbound truck  $i$  and an outbound truck  $j$ . Constraint (5) guarantees that each inbound truck  $i$  is assigned to an inbound dock. Constraint (6) ensures that only one inbound truck is allocated to an inbound dock at a time. In (7) determines the finish time of inbound truck  $i$ . In (8) sets the start time of inbound trucks after the finish time of all its predecessors. Constraint (9) ensures the unloading operation for truck  $i$  starts after it arrives at the inbound dock. Constraint (10) guarantees that each outbound truck  $j$  is assigned to an outbound dock. Constraint (11) ensures that only one outbound truck is assigned to an outbound dock at a time. Constraint (12) ensures the loading operation for truck  $j$  starts after it arrives at the outbound dock. In (13) determines the finish time of outbound truck  $j$ . In (14) sets the start time of outbound trucks after the finish time of all its predecessors. Constraint (15) sets relationships among  $V_{ij}$ ,  $Y_{im}$ ,  $Z_{jn}$ , and  $U_{ijmn}$  variables. In (16) states the finish time of outbound truck  $j$  is equal or more than the start time of inbound trucks, unloading time of items, transshipment time from inbound to outbound docks, and loading time of items to the outbound truck  $j$ . In (17-18) represent the non-negativity and domain constraints.

$$\text{MinMax}\{L_j\} \quad \forall j \quad (1)$$

$$\sum_{k=1}^J X_{ijk} = r_{ik} \quad \forall i, k \quad (2)$$

$$\sum_{i=1}^I X_{ijk} = s_{jk} \quad \forall j, k \quad (3)$$

$$\sum_{k=1}^K X_{ijk} \leq G * V_{ij} \quad \forall i, j, k \quad (4)$$

$$\sum_{m=1}^M Y_{im} = 1 \quad \forall i \quad (5)$$

$$Y_{im} + Y_{i'm} \leq P_{ii'} + P_{i'i} + 1, \quad \forall m, i, i' \quad i \neq i' \quad (6)$$

$$c_i + t_{\text{unload}} \sum_{k=1}^K r_{ik} \leq C_i \quad \forall i \quad (7)$$

$$C_i + DT - G(1 - P_{ii'}) \leq c_{i'}, \quad \forall i, i' \quad i \neq i' \quad (8)$$

$$c_i \geq A_i \quad \forall i \quad (9)$$

$$\sum_{n=1}^N Z_{jn} = 1 \quad \forall j \quad (10)$$

$$Z_{jn} + Z_{j'n} \leq Q_{jj'} + Q_{j'j} + 1, \quad \forall n, j, j' \quad j \neq j' \quad (11)$$

$$l_j \geq A_j \quad \forall j \quad (12)$$

$$l_j + t_{\text{load}} \sum_{k=1}^K s_{jk} \leq L_j \quad \forall j \quad (13)$$

$$L_i + DT - G(1 - Q_{jj'}) \leq l_j, \quad \forall j, j' \quad j \neq j' \quad (14)$$

$$U_{ijmn} \geq V_{ij} + Y_{im} + Z_{jn} - 2 \quad \forall i, j, m, n \quad (15)$$

$$c_i + (t_{\text{unload}} + t_{\text{load}}) \sum_{k=1}^K x_{ijk} + t_{mn} G(1 - U_{ijmn}) \leq L_j \quad \forall i, j, m, n \quad (16)$$

$$V_{ij}, Y_{im}, Z_{jn}, U_{ijmn}, P_{ii'}, Q_{jj'} \in \{0, 1\} \quad \forall i, i', j, j', m, n, k \quad (17)$$

$$X_{ijk}, c_i, C_i, l_j, L_j \geq 0 \quad \forall i, j, k \quad (18)$$

### 3. RESEARCH METHOD

The cross-docking problem is considered NP-Hard since it includes the generalized assignment problem [22], therefore, two metaheuristics called simulated annealing (SA) and particle swarm optimization (PSO) are implemented for solving the cross-docking problem for the consumer goods sector to provide high-quality solutions in short computing times [23, 24]. SA is classified as a trajectory method meanwhile PSO is a population-based method. Therefore, the metaheuristics not only are used for solution cross-docking

problem design but also to compare their performance. This is raised due to the differences in their computational structures and modeling. SA and PCO metaheuristics are represented in the following subsections.

### 3.1. Simulated annealing

SA is a metaheuristic classified as a trajectory method to find the solution of combinatorial problems [25], and is adapted for solving the cross-docking problem [26]. In the first stage, temperature ( $T=T_0$ ) and iterations ( $n=0$ ) are initialized as well as final temperature ( $T_{min}$ ), total iterations ( $N_{max}$ ), and cooling rate ( $\alpha$ ) are set for the SA [27]. In the second stage, the initial solution  $r$  is generated, and the fitness function is calculated following the objective function of the cross-docking operation. In the third stage, a neighbor solution  $r'$  is created by performing swap and insertion moves on current solution  $r$ . In the fourth stage, fitness functions for  $r$  and  $r'$  are compared, and if  $f(r') < f(r)$  then  $r'$  becomes the current solution for the SA. If this condition is not satisfied, then (19) is used to calculate the transition probability.

$$P(r, r', T) = \exp\left(\frac{f(r') - f(r)}{T}\right) \quad (19)$$

If a random number between 0 and 1 is less than the transition probability, then  $r'$  is set as the current solution for the SA ( $r=r'$ ), otherwise,  $r$  remains as the current solution. Subsequently, if the current iteration is equal to the maximum number of iterations ( $n=N_{max}$ ), then the temperature is compared to the final temperature ( $T=T_{min}$ ). If this condition is met then the SA stops and solution  $r$  remains as the current solution for the cross-docking problem, otherwise, the cooling rate ( $\alpha$ ) is multiplied by initial current temperature, as shown in (20), and temperature  $T$  is used to perform the SA from the third stage.

$$T = \alpha * T \quad (20)$$

---

#### Algorithm 1. Simulated Annealing to solve the cross-docking problem

---

```

Input Data();
Initial parameters  $T = T_0$ ,  $n = 0$ ,  $T_{min}$ ,  $N_{max}$ ,  $\alpha$ , bestr
Generate initial solution  $r$  for cross-docking operation
Generate neighbor solutions  $r'$  for  $r$  using swap and insertion moves
While  $T > T_{min}$  do
  While  $n < N_{max}$  do
    Fitness Function();
    If  $f(r') < f(r)$  then
      Set  $r = r'$ 
      If  $f(r) < f(bestr)$  then
        Set  $bestr = r$ 
      End if
    Else
      If  $\text{random}(0,1) < \exp(f(r) - f(r')/T)$  then
        Set  $r = r'$ 
      End if
    End if
     $n = n + 1$ 
  Loop
  End while
  Set  $T = \alpha * T$ 
Loop
End while
Output Data();
Print bestr,  $f(bestr)$ 

```

---

### 3.2. Particle swarm optimization

PSO is a population-based metaheuristic based on the social feeding of some animals [28, 29], implemented in this case for solving the cross-docking problem. In the first stage, initial velocity  $V_r^0$  and initial position  $X_r^0$  parameters are generated for each particle  $r$ . The second stage measures the makespan for each particle belonging to the solution set. The third stage updates velocity  $V_r^t$  and position  $X_r^t$  using (21) and (22) respectively, generating a new solution set for the cross-docking problem. The velocity updating uses some parameters like  $\omega$  (inertia factor),  $rnd_1$  and  $rnd_2$  (random numbers  $\in [0,1]$ ),  $c_1$  and  $c_2$  (positive constant linked to cognitive and social components for PSO), and some variables like  $X_{best_r}^t$  (the best position of particle  $r$  for  $t$ ) and  $G_{best}$  (the best position from all particles for  $t$ ). In the fourth stage, if fitness value of the particle  $r$  in  $t$   $f(X_r^t)$  is less than  $f(X_{best_r}^t)$ , then  $X_r^t$  is set as  $X_{best_r}^t$ , and if  $f(X_{best_r}^t)$  is less than

$f(G_{best})$ , then  $X_{best_r}^t$  is set as  $G_{best}$ . Subsequently, if the PSO is executed until reaching the maximum number of iterations ( $t=T_{max}$ ).

$$V_r^{t+1} = \omega V_r^t + c_1 * rnd_1 * (P_{best_r}^t - X_r^t) + c_2 * rnd_1 * (X_{best_r}^t - X_r^t) \tag{21}$$

$$X_r^{t+1} = X_r^t + V_r^{t+1} \tag{22}$$

---

**Algorithm 2. Particle Swarm Optimization to solve the cross-docking problem**

---

```

Input Data();
Initial parameters  $T_{max}$ , R
Generate initial velocity  $V_r^0$  and initial position  $X_r^0$  for each particle r randomly
Do while  $t \leq T_{max}$  iteraciones o ciclos
  Do while  $r \leq R$ 
    Fitness Function();
    If  $f(X_r^t) < f(X_{best_r}^t)$  then
      Set  $X_{best_r}^t = r$ 
      If  $f(X_{best_r}^t) < f(G_{best})$  then
        Set  $G_{best} = X_{best_r}^t$ 
      End if
    End if
    Update velocity of particle r using Equation (X)
    Update position of particle r using Equation (XX)
     $r=r+1$ 
  Loop
End while
 $t=t+1$ 
Loop
End while
Output Data();
Print  $G_{best}$ ,  $f(G_{best})$ 

```

---

**3.3. Experimental study**

A full factorial design  $3 \times 4 \times 3 \times 3$  is chosen to perform an experimental study for the cross-docking problem. Factor levels are chosen based on results obtained from the literature, distribution centers, and consumer goods companies. On the other hand, the level “empirical rule” for the solution method factor, represents a solution method based on the knowledge and experience of a distribution center analyst. Thus, the computational experiments are based on 108 treatments, and 3 replicates per treatment, obtaining 324 experimental runs. As the experimental factors are independent, it contributes to the viability of the full factorial design. Table 1 shows the factors, levels, and the response variable of the experimental study. Solution methods are coded in Java® programming language and experimental runs were performed using an Intel Core I7 Laptop with 8GB RAM memory.

Table 1. Parameters for the experimental study

Factor	Levels	Number of levels	Response variable
Inbound docks	3; 6; 9	3	Makespan (minutes)
Outbound docks	3; 6; 9; 12	4	
Items to be handled	2,000; 5,000; 8,000	3	
Solution method	SA; PSO; Empirical rule	3	

**4. RESULTS AND DISCUSSION**

Table 2 presents the outputs of the experiments through a full factorial model ANOVA, obtained by using Minitab® 17. Based on the ANOVA from Table 2, it is detected that the main effects, as well as two-way interaction effects, are statistically significant on the variable response ( $p\text{-value} < 0.05$ ) [30]. Additionally, two-way interactions are evaluated to identify the factor levels generating the minimum makespan for cross-docking operations. From the ANOVA, it is identified that inbound docks by outbound docks, outbound docks by the number of items (products) and solution method by the number of items two-way interaction effects are significant on variable response.

Table 2. Full factorial model ANOVA for the cross-docking problem

Source	DG	SS	MS	F-value	p-value
<i>Model</i>	107	84200094	786917	1102460	0.00
<i>Linear</i>	9	82586959	9176329	12855915	0.00
Idocks	2	1788188	894094	1252614	0.00
Odocks	3	1900131	633377	887353	0.00
Products	2	77675843	38837922	54411413	0.00
Metaheuristic	2	1222797	611399	856561	0.00
<i>Two way interaction effects</i>	30	757540	25251	35377	0.00
Idocks*Odocks	6	283955	47326	66303	0.00
Idocks*Products	4	62542	15636	21905	0.00
Idocks*Metaheuristic	4	28377	7094	9939	0.00
Odocks*Products	6	201005	33501	46934	0.00
Odocks*Metaheuristic	6	59898	9983	13986	0.00
Products*Metaheuristic	4	121762	30441	42647	0.00

The inbound docks by outbound docks interaction effects are depicted in Figure 2(a), highlighting that the combination of nine inbound docks by twelve outbound docks generates the shortest makespan of 973.42 minutes. The second-best solution is the combination of six inbound docks by twelve outbound docks, providing a makespan of 1,008.42 minutes, increasing the cross-docking operation time in 35 minutes (3.6%) compared to the best solution. The third most efficient solution comprises the combination of six inbound docks and nine outbound docks, generating a makespan of 1,014.08 minutes, which increases the operation time in 40 minutes (4.2%) compared to the best factor combination. These results suggest selecting nine inbound docks by twelve outbound docks (21 docks in total) to perform cross-docking operations generating the minimum makespan. However, six inbound docks by nine outbound docks represent the most efficient solution for a cross-docking design since it requires less amount of inbound and outbound docks (15 docks in total), reducing the requirement of docks by 28.6% and increasing the makespan by only 4.2%, representing a favorable trade-off for cross-docking platform design.

Likewise, Figure 2(b) shows that each combination of PSO by the number of items (products) generates the shortest average makespan for the cross-docking problem. The difference between PSO by 2,000 items and SA by 2,000 represents 42 minutes (7.7%), while the difference between PSO by 8,000 items and SA by 8,000 represents 70.6 minutes (4.1%). When comparing the empirical rule to PSO, the empirical rule increases the average makespan between 87 minutes (16%) and 181.8 minutes (10.8%) considering 2,000 and 8,000 items respectively. Therefore, PSO and SA provide satisfactory results to solve the cross-docking problem, consequently, these metaheuristics can be used in daily operations of cross-docking platforms, which in this case will have six inbound docks and nine outbound docks to reduce infrastructure investment.

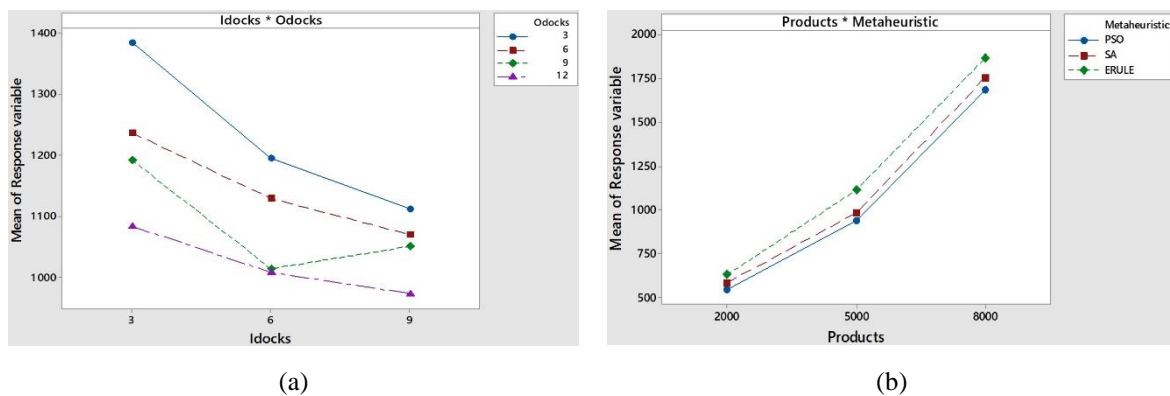


Figure 2. Two-way interaction effects, (a) Inbound docks by outbound docks, (b) Solution methods by the number of items (products)

## 5. CONCLUSION

Cross-docking for customer goods companies is a logistics process adopted to implement an agile supply chain approach. Transshipment, cargo consolidation, and deconsolidation, and other logistics operations are commonly used in the cross-docking process to avoid the storage process and increase the speed of supply chains. In this study, inbound and outbound docks, items (products to be delivered), and solution methods (metaheuristics and empirical rule) are represented for the cross-docking problem.

Based on the ANOVA, the most statistically significant two-way interaction effects on the response variable are inbound docks by outbound docks, outbound docks by items, and items by solution methods. Consequently, the results show that the combination of nine inbound docks by twelve outbound docks generates the shortest makespan (973.4 minutes), while the combinations of PSO by items produced the shortest average makespan for solving the cross-docking problem. However, the combination of six inbound docks and nine outbound docks provides the most efficient solution for a cross-docking design since it reduces the requirement of docks by 28.6% and increases the makespan by only 4.2% when compared to the best solution, representing a favorable trade-off for the cross-docking platform design. Further research could consider congestion and heterogeneous MHE to improve the ability to represent the cross-docking problem. Finally, this study could be improved by considering the capacity limit for temporary storage, which would directly affect the start and finish time of inbound and outbound trucks.

## REFERENCES

- [1] K. K. Chandriah and N. V Raghavendra, "Multi-objective optimization for preemptive and predictive supply chain operation," *Int. J. Electr. Comput. Eng.*, vol. 10, no. 2, pp. 1533-1543, 2020.
- [2] H. Mezouar and A. El Afia, "Proposal for an approach to evaluate continuity in service supply chains : case of the Moroccan electricity supply chain," *Int. J. Electr. Comput. Eng.*, vol. 9, no. 6, pp. 5552-5559, 2019.
- [3] A. Rijal, M. Bijvank, and R. De Koster, "Integrated scheduling and assignment of trucks at unit-load cross-dock terminals with mixed service mode dock doors," *Eur. J. Oper. Res.*, vol. 278, no. 3, pp. 752-771, 2019.
- [4] M. F. Monaco and M. Sammarra, "Managing loading and discharging operations at cross-docking terminals," *Procedia Manuf.*, vol. 42, pp. 475-482, 2020.
- [5] M. Gaudio, M. Flavia, and M. Sammarra, "A Lagrangian heuristics for the truck scheduling problem in multi-door, multi-product Cross-Docking with constant processing time," *Omega*, no. In press, 2020.
- [6] T. H. Nogueira, F. P. Coutinho, R. P. Ribeiro, and M. G. Ravetti, "Parallel-machine scheduling methodology for a multi-dock truck sequencing problem in a cross-docking center," *Comput. Ind. Eng.*, vol. 143, no. March, p. 106391, 2020.
- [7] G. B. Fonseca, T. H. Nogueira, and M. G. Ravetti, "A hybrid Lagrangian metaheuristic for the cross-docking flow shop scheduling problem," *Eur. J. Oper. Res.*, vol. 275, no. 1, pp. 139-154, 2019.
- [8] M. T. Assadi and M. Bagheri, "Engineering differential evolution and population-based simulated annealing for truck scheduling problem in multiple door cross-docking systems," *Comput. Ind. Eng.*, vol. 96, pp. 149-161, 2016.
- [9] A. I. Nikolopoulou, P. P. Repoussis, C. D. Tarantilis, and E. E. Zachariadis, "Moving products between location pairs: Cross-docking versus," *Eur. J. Oper. Res.*, vol. 256, no. 3, pp. 803-819, 2017.
- [10] W. Wisittipanich and P. Hengmeechai, "Truck scheduling in multi-door cross docking terminal by modified particle swarm optimization," *Comput. Ind. Eng.*, vol. 113, pp. 793-802, 2017.
- [11] A. H. Goodarzi and S. H. Zegordi, "A location-routing problem for cross-docking networks: A biogeography-based optimization algorithm," *Comput. Ind. Eng.*, vol. 102, pp. 132-146, 2016.
- [12] M. Y. Maknoon, F. Soumis, and P. Baptiste, "Optimizing transshipment workloads in less-than-truckload," *Intern. J. Prod. Econ.*, vol. 179, pp. 90-100, 2016.
- [13] S. Kusolpuchong, K. Chusap, O. Alhawari, and G. Suer, "A Genetic Algorithm Approach for Multi Objective Cross Dock Scheduling in Supply Chains," *Procedia Manuf.*, vol. 39, pp. 1139-1148, 2019.
- [14] M. A. Aziz and I. H. Ninggal, "Scalable workflow scheduling algorithm for minimizing makespan and failure probability," *Bull. Electr. Eng. Informatics*, vol. 8, no. 1, pp. 283-290, 2019.
- [15] A. Golshahi-roudbaneh, M. Hajiaghahi-keshteli, and M. M. Paydar, "Developing a lower bound and strong heuristics for a truck scheduling problem in a cross-docking center," *Knowl.-Based Syst.*, vol. 129, pp. 17-38, 2017.
- [16] A. A. Correa, R. A. Gómez, and J. A. Cano, "Warehouse management and information and communication technology," *Estud. Gerenciales*, vol. 26, no. 117, pp. 145-171, 2010.
- [17] J. Van Belle, P. Valckenaers, and D. Cattrysse, "Cross-docking: State of the art," *Omega*, vol. 40, no. 6, pp. 827-846, 2012.
- [18] J. A. Cano, "Order picking optimization based on a picker routing heuristic: minimizing total traveled distance in warehouses," in *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.
- [19] J. A. Cano and J. J. Baena, "Trends in the use of information and communication technologies for international negotiation," *Estud. Gerenciales*, vol. 31, no. 136, pp. 335-346, 2015.
- [20] J. A. Cano and J. J. Baena, "Impact of information and communication technologies in international negotiation performance," *Revista Brasileira de Gestão de Negócios*, vol. 17, no. 54, pp. 751-768, 2015.
- [21] A. Amini, R. Tavakkoli-moghaddam, and A. Omidvar, "Cross-docking truck scheduling with the arrival times for inbound trucks and the learning effect for unloading/loading processes," *Prod. Manuf. Res.*, vol. 2, no. 1, pp. 784-804, 2014.
- [22] S. Gelareh, F. Glover, O. Guemri, S. Hanafi, P. Nduwayo, and R. Todosijevic, "A comparative study of formulations for a cross-dock door assignment problem," *Omega*, vol. 91, p. 102015, 2020.
- [23] J. A. Cano, "Formulations for joint order picking problems in low-level picker-to-part systems," *Bull. Electr. Eng. Informatics*, vol. 9, no. 2, pp. 836-844, 2020.
- [24] J. A. Cano, A. A. Correa-Espinal, and R. A. Gómez-Montoya, "Mathematical programming modeling for joint order batching, sequencing and picker routing problems in manual order picking systems," *J. King Saud Univ.-Eng.*



- Sci.*, vol. 32, no. 3, pp. 219-228, 2019.
- [25] Z. K. A. Baizal, K. M. Lhaksana, A. A. Rahmawati, M. Kirom, and Z. Mubarak, "Travel route scheduling based on user's preferences using simulated annealing," *Int. J. Electr. Comput. Eng.*, vol. 9, no. 2, pp. 1275-1287, 2019.
- [26] A. Shahmardan and M. S. Sajadieh, "Truck scheduling in a multi-door cross-docking center with partial unloading – Reinforcement learning-based simulated annealing approaches," *Comput. Ind. Eng.*, vol. 139, p. 106134, 2020.
- [27] S. M. Mousavi, R. Tavakkoli-Moghaddam, and A. Siadat, "Optimal design of the cross-docking in distribution networks: Heuristic solution approach," *Int. J. Eng. Trans. A Basics*, vol. 27, no. 4, pp. 533-544, 2014.
- [28] W. R. Abdul-Adheem, "An enhanced particle swarm optimization algorithm," *Int. J. Electr. Comput. Eng.*, vol. 9, no. 6, pp. 4904-4907, 2019.
- [29] T. M. Shin, A. Adam, and A. F. Z. Abidin, "A comparative study of PSO, GSA and SCA in parameters optimization of surface grinding process," *Bull. Electr. Eng. Informatics*, vol. 8, no. 3, pp. 1117-1127, 2019.
- [30] D. C. Montgomery, "Design and analysis of experiments," 10th ed. Massachusetts: John Wiley & Sons Inc., 2019.

## BIOGRAPHIES OF AUTHORS



**Rodrigo Andrés Gómez-Montoya** obtained his PhD in Engineering, Industry and Organizations from the National University of Colombia, Colombia and obtained his Bachelor in Industrial Engineering from the same university. He is working as an Associate Professor at the Politécnico Colombiano Jaime Isaza Cadavid, and as a researcher in ESACS-Escuela Superior en Administración de Cadena de Suministro, Colombia. His research interest includes design of experiments, warehouse management, logistics, and supply chain management.



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



**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, optimization, and metaheuristics.



**Fernando Salazar** obtained his Post-doctorate in Humanitarian Logistics, Agro-Food Security from the University of Girona, Spain and obtained his PhD in Logistics and Supply Chain Management from UPAEP, México. He is working as an Associate Professor at the Pontificia Universidad Javeriana, Colombia. His research interest includes financial logistics, strategic cost management, logistics planning, and humanitarian logistics.