

## Vehicle routing and scheduling problem with order acceptance for pharmaceutical refrigerated logistics

Seung Jae Lee, Byung Soo Kim\*

*Department of Industrial and Management Engineering, Incheon National University, 119, Academy-ro, Yeonsu-gu, Incheon 22012, Republic of Korea*



### HIGHLIGHTS

- The addressed problem is formulated as a mixed integer linear programming model.
- An effective genetic algorithm is proposed to efficiently solve the optimization model.
- The managerial insights of the order acceptance strategy are suggested.

### ARTICLE INFO

**Keywords:**  
 Vehicle routing  
 Scheduling  
 Pharmaceutical cold chain  
 Mixed integer linear programming  
 Genetic algorithm

### ABSTRACT

This paper studies a delivery problem in a pharmaceutical cold chain. In this problem, each customer orders multiple pharmaceutical products and requires a due date for each order. Each product has sensitive characteristics of storage temperature range and shelf life. A refrigerated vehicle routing and scheduling problem with order acceptance is formulated as a mixed-integer-linear-programming model. In the model, the acceptance of each pharmaceutical order and the route and schedule of a limited number of refrigerated vehicles is simultaneously determined to maximize the total profit. A hybrid genetic algorithm embedding an insertion heuristic with modification operation is proposed. The performance of the proposed algorithm is evaluated and verified by comparing other metaheuristics. Managerial insights about various experimental environments for the order acceptance are suggested by conducting a sensitivity analysis.

### 1. Introduction

A cold chain offers continuous refrigeration processes from production to transportation, storage, and delivery to maintain the storage temperature of products. It is required to maintain the usability of items such as perishable foods, chemicals, and pharmaceutical products [1–5]. The efficient management of the cold chain has been especially emphasized in the pharmaceutical logistics chain [6].

To effectively manage a pharmaceutical cold chain logistics (PCL) company, it is required to improve both delivery performance and customer satisfaction. PCL companies should implement batch delivery for various pharmaceutical products to enhance delivery performance. Batches should be organized by grouping products requiring similar storage temperatures within the maximum load capacity of vehicles because PCL companies must maintain the storage temperature of each pharmaceutical product during delivery. This temperature constraint makes batch delivery in the PCL company more complex than the batch

delivery of industrial products. Additionally, to increase customer satisfaction, the PCL company should reflect the earliness and tardiness penalty and disposal penalty depending upon the due date and shelf life, respectively. Unlike general supply chains, the delivery of pharmaceutical products is completed when they are delivered to the ordered customers and stored in the customers' refrigerated facility. The refrigerated facility requires additional storage costs to maintain the storage temperature of the products in order to preserve the best quality. Thus, PCL company should serve 'just in time' delivery of pharmaceutical products to meet the due date. In addition, the PCL company must incur the disposal cost of pharmaceuticals when a PCL company delivers the pharmaceuticals exceeding their shelf lives.

PCL companies simultaneously consider the storage temperature, due date, and shelf life to determine the delivery routes for pharmaceutical orders. PCL companies are not able to avoid incurring earliness, tardiness, and disposal costs if they deliver lots of pharmaceutical orders over their delivery capacity due to a limited number of vehicles. These

\* Corresponding author.

E-mail address: [bskim@inu.ac.kr](mailto:bskim@inu.ac.kr) (B.S. Kim).

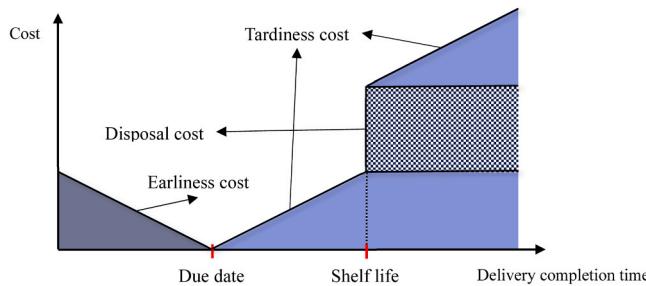


Fig. 1. Cost function for delivery penalty for each pharmaceutical order.

penalty costs may even exceed its revenue. To reduce unnecessary costs and enhance the delivery operation with a limited number of vehicles, the PCL company should adopt an effective management strategy that appropriately rejects the orders that incur unnecessary costs. Order acceptance (OA) is the management strategy in which a firm determines the order to accept or reject based on its business direction, the current capacity of the resource, the trade-off between the company cost policy and order revenue, and other pertinent considerations. The importance of the OA strategy has been emphasized in practices, and it has been actively studied in various industrial fields [7–13].

In this paper, a vehicle routing and scheduling problem with order acceptance for pharmaceutical refrigerated logistics (VRSP-OAPRL) is addressed to maximize the total profit of the PCL company. Batch organizing, batch routing, and batch scheduling processes are the main decision processes to determine five sub-decisions from (i) to (v) in VRSP-OAPRL. In the batch organizing process, (i) the acceptance of pharmaceutical orders, and (ii) the grouping of the accepted pharmaceutical orders to construct delivery batches are decided. Temperature and capacity constraints must be addressed when grouping the accepted pharmaceutical orders as the delivery batches. To satisfy the temperature constraint, the storage temperature ranges of pharmaceutical products grouped in the same delivery batch must be overlapped. The

total batch weight must not exceed the capacity of the homogeneous vehicle to satisfy the capacity constraint. In the batch routing process, (iii) the delivery route of pharmaceutical orders grouped into the same delivery batch is determined. In the batch scheduling process, (iv) the assignment of each delivery batch to one vehicle, and (v) the sequence of delivery batches assigned to the same vehicle are decided. The total profit consists of the total delivery revenue of accepted orders and the total cost following the cost function. The cost function defined in Eq. (1) is as follows:

$$f_i(c_i) = \begin{cases} \alpha \times R_i \times (D_i - c_i) & , 0 \leq c_i < D_i \\ \alpha \times R_i \times (c_i - D_i) & , D_i \leq c_i \leq S_i \\ \alpha \times R_i \times (c_i - D_i) + R_i & , S_i < c_i \end{cases} \quad (1)$$

where  $\alpha$ ,  $R_i$ ,  $D_i$ , and  $S_i$  denote the cost weight, delivery revenue, due date, and shelf life of the pharmaceutical product  $i$ , respectively. Fig. 1 represents a cost function  $f_i$  of pharmaceutical products  $i$  with delivery completion time  $c_i$  in the proposed VRSP-OAPRL. If the pharmaceutical product  $i$  is delivered before the due date, it linearly incurs only the earliness cost that is multiplied by the cost weight, earliness amount and revenue of product  $i$ . Similarly, if the pharmaceutical product  $i$  is delivered between the due date and shelf life, it linearly incurs only the tardiness cost that is multiplied by the cost weight, tardiness amount and revenue of product  $i$ . However, if the pharmaceutical product  $i$  is delivered exceeding its shelf life, it simultaneously incurs the tardiness cost and the fixed disposal cost that equals to its delivery revenue of pharmaceutical product  $i$ .

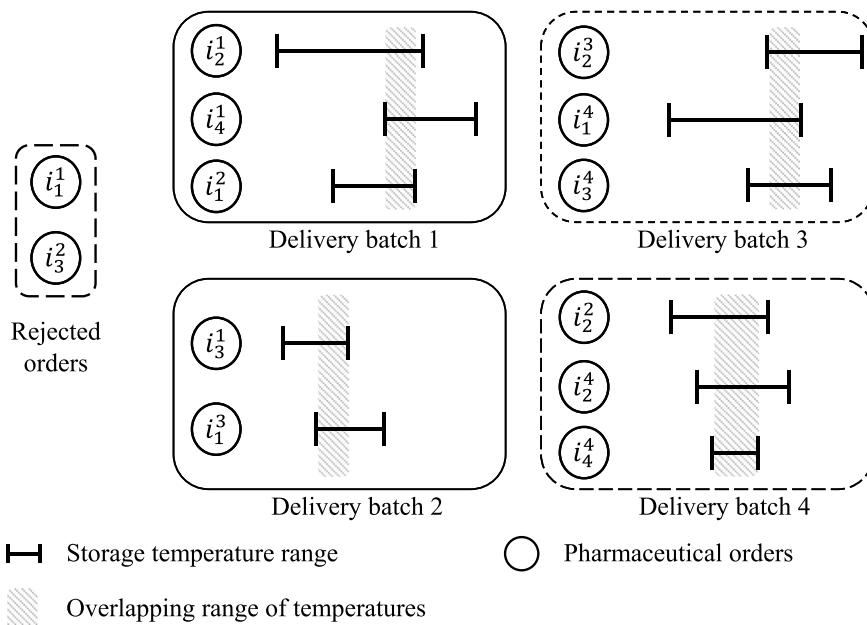
VRSP-OAPRL is formulated as a mixed-integer-linear-programming (MILP) model to maximize the total profit. A hybrid genetic algorithm embedding an insertion heuristic with modification operation (HGAM) is proposed to effectively solve the MILP in large-sized instances. By comparing it with other metaheuristic algorithms, the effectiveness of HGAM is verified.

The literature survey for previous studies related to the VRSP-OAPRL is addressed in Section 2. In Sections 3 and 4, a problem description and MILP model are described, respectively. In Section 5, HGAM is

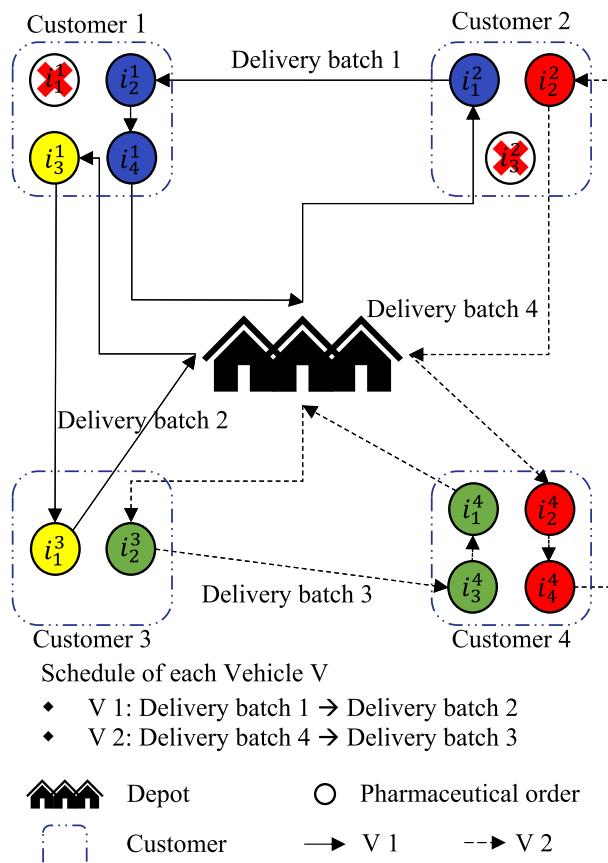
Table 1  
The summary of literature review.

Reference	Vehicle			OA	Delivery Products			Time Components			Methodology	Objective Value
	Homo	Hetero	RV		Perishable	Pharmaceutical	Etc.	TW	Due date	Deadline		
[16]	✓				✓			✓			MILP, IMOLS	Min transportation time, Min delay time
[17]	✓				✓			✓			MIP	Min. Traveling cost
[18]	✓				✓			✓			MS-ILS	Min. traveling and supplying the auxiliary depots costs
[19]	✓							✓	✓		TS	Min. maximum routing cost
[20]	✓	✓	✓	✓	✓			✓	✓	✓	Exact algorithm	Min. the demand served within the deadline
[21]	✓				✓			✓			SA	Min. travel and cargo theft costs
[22]	✓	✓	✓		✓						HACROA	Min. total traveling distance
[23]	✓	✓	✓					✓			GTS	Min. the costs of maintaining stable thermal conditions
[24]	✓	✓	✓	✓				✓		✓	MIP	Min. traveling, holding, earliness, and tardiness costs
[25]	✓	✓	✓					✓			ALNS	Min. fixed, travel, total fuel-consumption costs
[26]	✓	✓	✓	✓				✓			ACO	Min. transportation economic and safety loss costs
[27]		✓	✓								Greedy Search	Min. transportation cost
[28]	✓	✓	✓					✓			ACS with PLS	Min. fuel-consumption, cooling, damage cost
[29]	✓			✓	✓			✓	✓	✓	MIP, HACO	Max. total expected profit
[30]	✓			✓	✓			✓	✓	✓	WWO	Max. revenue per unit time
[31]	✓			✓	✓			✓	✓	✓	B&P	Max. total profit
This	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	MILP, HGAM	Max. total profit

\*RV: Refrigerated vehicle; TW: Time window;



(a) Batching process of VRSP-OAPRL



(b) Routing and scheduling processes of VRSP-OAPRL

**Fig. 2.** Illustrative example for VRSP-OAPRL.

developed. In [Section 6](#), numerical experiments and sensitivity analysis are conducted. Finally, the conclusion is summarized in [Section 7](#).

## 2. Literature review

The vehicle routing problem (VRP) and vehicle routing and scheduling problem (VRSP) are well-known NP-hard problems and were first proposed in [\[14\]](#) and [\[15\]](#), respectively. Since then, various VRPs and VRSPs have been studied according to the conditions and assumptions of specific problems. In this section, the literature survey of VRP and VRSP regarding the pharmaceutical, cold chain, and order acceptance strategy is addressed. [Table 1](#) indicates the summary of the related papers.

In the literature survey regarding the pharmaceutical VRP and VRSP, Bouziyane et al. [\[16\]](#) introduced the disrupted VRP with soft-time windows (VRPSTW) for pharmaceutical suppliers. To react to the increasing pharmaceutical demands, the proposed VRPTW consisted of a homogeneous capacity constraint and multi-objective to minimize the total travel and delay times. To solve the VRPSTW, an improved multi-objective local search (IMOLS) was used as a neighborhood search heuristic. Lubis and Mawengkang [\[17\]](#) studied the capacitated heterogeneous VRP with time window (CHVRPTW) to deliver the pharmaceutical orders and conducted the case study for a medicine supply firm in Medan City, Indonesia. The heterogeneous capacity and hard time window constraints were addressed, and the CHVRPTW was solved by a mixed-integer-programming (MIP). Kramer et al. [\[18\]](#) presented the rich VRP (RVRP) encountered by the third-party pharmaceutical logistics firm and conducted the case study with the pharmaceutical distributor in Tuscany. The proposed RVRP minimized the cost consisting of the travel cost and auxiliary depots costs and considered various constraints such as multi-depots, heterogeneous vehicles, time window considering flexibility, and multi-periodic demands. To solve the presented RVRP, the multi-start iterated local search (MILS) algorithm was proposed. Liu et al. [\[19\]](#) addressed the periodic home healthcare pick-up and delivery problem (PHPDP) and extended the periodic vehicle routing problem with the time windows (PVRPTW). A case study was conducted by using the field data of a company providing home healthcare services in this study. The hybrid tabu search (TS) was proposed as the methodology in this paper to minimize the longest length among routes. Ceselli et al. [\[20\]](#) proposed the location and routing problem (LRP) to optimize the coordination of the distribution center and delivery process of drugs or vaccines in the emergency healthcare logistics system. In [\[20\]](#), the number of demands served within the deadlines for drugs or vaccines was maximized by the proposed branch and bound with column generation algorithm. Repolho et al. [\[21\]](#) studied the cargo theft-weighted VRP (CTWVRP). For the CTWVRP, the heterogeneous capacity constraint was considered. The simulated annealing (SA) was proposed to solve the CTWVRP and minimize the total travel and cargo theft costs. In this study, a case study was conducted by applying the proposed CTWVRP and SA to the pharmaceutical distributor in Brazil. The optimization of the vaccine distribution in the cold chain was studied in [\[22\]](#) and [\[32\]](#). Sujaree and Samattapapong [\[22\]](#) addressed the refrigerated vehicle routing problem for the COVID-19 vaccine in Thailand to minimize the traveling distances by the hybrid artificial chemical reaction optimization algorithm (HACROA). To consider the temperature-sensitivity character of the vaccine, capacitated refrigerated vehicles were used. To evaluate the effectiveness of the COVID-19 vaccine distribution in Norway, a simulation study was conducted in [\[32\]](#).

In the literature survey regarding the cold chain, Awad et al. [\[33\]](#) proposed a literature review about the cold chain for perishable and temperature sensitive foods. The refrigerated capacitated VRP (RCVRP) was studied to minimize the energy consumption during the routing in [\[23\]](#). The proposed RCVRP considered the homogeneous capacity and

temperature-control of the refrigerated vehicle, and the granular TS was developed to solve this problem. Komijan and Delavari [\[24\]](#) presented VRSP considering multi-period and multiple perishable products to minimize the total cost by the mixed-integer-programming model. In the proposed VRSP, there was an assumption, which heterogeneous refrigerated vehicles were used to deliver the perishable products, and the available time constraint for product delivery was considered. Chen et al. [\[25\]](#) studied the multi-compartment VRP with time windows (MC-VRPTW) to minimize the total cost. A mathematical model and adaptive large neighborhood search (ALNS) were proposed, and the real-case problem in China was solved. For MC-VRPTW, the compartment capacity constraint and the assumption of temperature were considered. In [\[26\]](#), VRP considering the time-varying road and temperature-controlled vehicles was studied, and the objective value was defined as the total cost. The optimal refrigerated temperature of the vehicles was determined, and the homogeneous capacity constraint was considered. Qiang et al. [\[27\]](#) presented a refrigerated VRP to minimize the transportation cost. The homogeneous capacity constraint was considered, and the greedy search algorithm was proposed. Qi and Hu [\[28\]](#) studied a VRP in the emergency cold chain, and the ant colony system (ACS) with pareto local search was used to solve the proposed VRP. The papers reviewed in this paragraph considered the temperature of the vehicles to minimize the energy consumption costs or used the refrigerated vehicles to deliver.

In the literature survey regarding the order acceptance, Ma et al. [\[29\]](#) addressed a combined order selection and time-dependent vehicle routing problem with time window for perishable product delivery (COSTDVRPTW). To advise providers who might face insufficient resources for perishable product delivery, maximizing total profit not minimizing delivery cost was defined as the objective function of the COSTDVRPTW. Zhang et al. [\[30\]](#) proposed an integration problem of order selection and delivery path planning in takeaway industry. A hybrid evolutionary algorithm was addressed to efficiently maximize the revenue per time unit. Yu et al. [\[31\]](#) proposed a combined order selection and periodic vehicle routing problem with time windows for perishable products (COS-PVRPTW-P) to maximize the total profit composed of the travel cost and collected revenue. To solve the proposed COS-PVRPTW-P, the branch-and-price algorithm was addressed. The integrated production and distribution problem considering order acceptance was handled in various studies with extension to maximize total profit [\[8,11,34\]](#).

In this paper, VRSP-OAPRL including the just-in-time delivery, the storage temperature, and the shelf life is dealt with to effectively manage the PCL company and apply it to the real-world. Also, the pharmaceutical orders incurring unnecessary delivery penalty costs are effectively rejected in VRSP-OAPRL. To the best of our knowledge, there is no study that simultaneously considers the just-in-time delivery, the strict storage temperature, and the shelf life of the pharmaceutical products to maximize the total profit by determining the order acceptance. Thus, the main contributions are as follows:

- 1) The MILP model simultaneously considers the total cost function reflecting the due date and the shelf life, determining the acceptance of pharmaceutical orders, and the temperature constraint to construct delivery batches.
- 2) HGAM is proposed to solve the optimization model and shows a statistically significant result in terms of solution performance compared to other metaheuristic algorithms in numerical experiments.
- 3) The managerial insights of the OA strategy are suggested through the sensitivity analysis in various experimental environments configured with the number of pharmaceutical orders, the condition of delivery resources, and the weight of earliness and tardiness costs.

### 3. Problem description

**Fig. 2** represents an illustrative example of VRSP-OAPRL.  $C = \{1, 2, \dots, n\}$  and  $n$  represent a set of all customers and the total number of customers, respectively.  $I_c = \{i_1^c, i_2^c, \dots, i_j^c\}$  and  $i_j^c, \forall c \in C$  represent a set of all pharmaceutical products and the number of pharmaceutical products ordered by customer  $c$ , respectively.  $I = I_1 \cup I_2 \dots \cup I_n$  represents a set of whole pharmaceutical products ordered by every customer.  $DE = \{0\}$  represents a set of a depot. VRSP-OAPRL is able to be defined on a directed graph  $G = (N, A)$  in which  $N = I \cup DE$  is a set of nodes, and  $A = \{(i, j) : i, j \in N\}$  is a set of edges. The assumptions in the VRSP-OAPRL are as follows:

- 1) In VRSP-OAPRL, the location, order list, and due date of the customers are predetermined information. Each customer orders various pharmaceutical products with related due dates. The location of each pharmaceutical product (node) equals the customer, who orders the pharmaceutical product. Thus, VRSP-OAPRL is able to be represented as the directed graph  $G = (N, A)$  using the pharmaceutical orders.
- 2) If the pharmaceutical products  $i$  and  $j$  are ordered by the same customer, the due dates  $D_i$  and  $D_j$  are the same, and the delivery time nodes  $i$  to  $j, DT_{ij}$  equals 0.
- 3) Regardless of the customers, each pharmaceutical order  $i$  has its own storage temperature range  $[ST_i^{LB}, ST_i^{UB}]$  and the shelf life  $S_i$ .

The batch organizing, batch routing, and batch scheduling processes of VRSP-OAPRL, as above mentioned, are described in **Fig. 2**. Four customers, thirteen pharmaceutical orders with weight 1, and two vehicles with a maximum load capacity 3 are presented in **Fig. 2**. Customers 1–4 order corresponding pharmaceutical product groups,  $(i_1^1, i_2^1, i_3^1, i_4^1)$ ,  $(i_1^2, i_2^2, i_3^2)$ ,  $(i_1^3, i_2^3)$ , and  $(i_1^4, i_2^4, i_3^4, i_4^4)$ , respectively. The order list of each customer is already known information. The process of batching pharmaceutical products is shown in **Fig. 2(a)**. In the batch organizing process, the acceptance procedure of each pharmaceutical order and the grouping procedure of the accepted orders as the delivery batch are sequentially determined. Since pharmaceutical orders  $i_1^1$  and  $i_3^2$  are rejected by the PCL company, they are not able to be used to construct the delivery batch. Delivery batches 1–4 are constructed as corresponding pharmaceutical order groups,  $(i_2^1, i_4^1, i_1^2)$ ,  $(i_3^1, i_1^3)$ ,  $(i_2^3, i_1^4, i_3^4)$ , and  $(i_2^4, i_3^4, i_4^4)$ , respectively. In each delivery batch, the storage temperature ranges of pharmaceutical orders must be overlapped, and the total weight of the batch must not exceed the vehicle capacity. The processes of routing each delivery batch and scheduling each vehicle are described in **Fig. 2(b)**. The delivery route within each constructed batch is determined by the routing process. The delivery sequences within the delivery batches 1–4 correspond to

$[Depot \rightarrow Cus. 2 (i_1^2) \rightarrow Cus. 1 (i_2^1, i_4^1) \rightarrow Depot]$ ,

$[Depot \rightarrow Cus. 1 (i_3^1) \rightarrow Cus. 3 (i_1^3) \rightarrow Depot]$ ,

$[Depot \rightarrow Cus. 3 (i_2^3) \rightarrow Cus. 4 (i_3^4, i_4^4) \rightarrow Depot]$ , and

$[Depot \rightarrow Cus. 4 (i_2^4, i_3^4) \rightarrow Cus. 2 (i_2^2) \rightarrow Depot]$ , respectively. Note that  $i_3^1$  and  $i_4^1$ , ordered by Customer 1. Since  $i_3^1$  and  $i_4^1$  do not overlap their storage temperature ranges from **Fig. 2(a)**, they are separately delivered through different delivery batches 2 and 1, respectively. In the scheduling process, the assignment and sequencing of each delivery batch are determined. In vehicle 1, delivery batches 1 and 2 are assigned, and delivery batch 1 is followed by delivery batch 2. In vehicle 2, delivery batches 3 and 4 are assigned, and delivery batch 3 is followed by delivery batch 4. The route, assignment, and sequencing are determined to

minimize the total cost.

### 4. Mixed integer linear programming

In this section, a MILP is described to solve the VRSP-OAPRL. The sets, parameters, and decision variables of the MILP model are as follows:

#### Decision variables

$a_i$	Equal to 1, if pharmaceutical order $i \in I$ is accepted; Otherwise 0
$x_{ib}$	Equal to 1, if pharmaceutical order $i \in I$ is assigned to delivery batch $b \in B$ ; Otherwise, 0
$y_{bv}$	Equal to 1, if delivery batch $b \in B$ is assigned to vehicle $v \in V$ ; Otherwise, 0
$z_i$	Equal to 1, if pharmaceutical order $i \in I$ is delivered after $S_i$ ; Otherwise, 0
$t_b$	Temperature of delivery batch $b \in B$
$r_{ij}^b$	Equal to 1, if node $i \in N$ immediately precedes node $j \in N$ in delivery batch $b \in B$ ; Otherwise, 0
$ui_{ib}$	Delivery sequence of node $i \in N$ in delivery batch $b \in B$
$c_i$	Delivery completion time of pharmaceutical order $i \in I$
$s_{ij}^v$	Equal to 1, if delivery batch $i \in B \cup DE$ immediately precedes delivery batch $j \in B \cup DE$ in vehicle $v \in V$ ; Otherwise, 0
$sb_b$	Starting time of delivery batch $b \in B$
$cb_b$	Completion time of delivery batch $b \in B$
$ub_{bv}$	Scheduling sequence of delivery batch $b \in B \cup DE$ in vehicle $v \in V$
$E_i$	Earliness of pharmaceutical order $i \in I$
$T_i$	Tardiness of pharmaceutical order $i \in I$

#### 4.1. Objective function

The objective function of VRSP-OAPRL is to maximize the total profit  $TP$ . The  $TP$  is composed of the delivery revenue, delivery failure cost, and earliness and tardiness cost. The formulations of the revenue and costs are proposed as follows.

$$DR = \sum_{i \in I} R_i \times a_i \quad (2)$$

**Eq. (2)** represents the *delivery revenue*,  $DR$ . In VRSP-OAPRL, the  $DR$  accrues when the products are delivered. Hence, the total delivery revenue equals to the summation of the delivery revenue of accepted pharmaceutical orders.

$$E_i \geq D_i - c_i - M \times (1 - a_i) \quad \forall i \in I \quad (3)$$

$$T_i \geq c_i - D_i - M \times (1 - a_i) \quad \forall i \in I \quad (4)$$

$$ETC = \sum_{i \in I} \alpha \times R_i \times (E_i + T_i) \quad (5)$$

**Eqs. (3) and (4)** calculate the earliness and tardiness of the pharmaceutical order  $i$ , respectively. **Eq. (5)** represents the *earliness and tardiness cost*,  $ETC$ . In VRSP-OAPRL, there is the earliness cost if the pharmaceutical order is delivered before the due date and the tardiness cost if the pharmaceutical order is delivered after the due date. The  $ETC$  is represented as the linear function of the pharmaceutical revenues.

$$c_i - S_i \leq M \times z_i \quad \forall i \in I \quad (6)$$

$$DFC = \sum_{i \in I} R_i \times z_i \quad (7)$$

**Eq. (6)** calculates whether the pharmaceutical order  $i$  is delivered after the shelf life or not. **Eq. (7)** represents the *disposal cost*,  $DC$ . In VRSP-OAPRL, delivering a pharmaceutical order after the shelf life is considered to be disposal. Then, as described in **Section 1**, the disposal cost that equals the amount of the delivery revenue is incurred. Hence, the total disposal cost equals the summation of the delivery revenue of pharmaceutical orders delivered beyond shelf lives.

#### 4.2. Mathematical model formulation

The MILP model is formulated as follows:

$$\text{Maximize } TP = DR - (DFC + ETC)$$

$$\sum_{b \in B} x_{ib} = a_i \quad \forall i \in I$$

$$\sum_{i \in I} x_{ib} \times W_i \leq CA \forall b \in B$$

$$ST_i^{LB} - M \times (1 - x_{ib}) \leq t_b \leq ST_i^{UB} + M \times (1 - x_{ib}) \quad \forall b \in B, i \in I$$

$$\sum_{v \in V} y_{bv} \leq \sum_{i \in I} x_{ib} \quad \forall b \in B$$

$$x_{ib} \leq \sum_{v \in V} y_{bv} \leq 1 \quad \forall b \in B, i \in I$$

$$r_{0i}^b + \sum_{j \in I} r_{ji}^b = x_{ib} \quad \forall b \in B, i \in I : i \neq j$$

$$r_{i0}^b + \sum_{j \in I} r_{ij}^b = x_{ib} \quad \forall b \in B, i \in I : i \neq j$$

$$\sum_{i \in I} x_{ib} \leq M \times \sum_{j \in I} r_{0j}^b \quad \forall b \in B$$

$$\sum_{i \in I} x_{ib} \leq M \times \sum_{j \in I} r_{j0}^b \quad \forall b \in B$$

$$r_{ii}^b = 0 \quad \forall b \in B, i \in N$$

$$u_{i0b} = 0 \quad \forall b \in B$$

$$u_{ib} + r_{ij}^b \leq u_{jb} + M \times (1 - r_{ij}^b) \quad \forall b \in B, i \in N, j \in I : i \neq j$$

$$u_{ib} \leq M \times \sum_{j \in N : i \neq j} r_{ij}^b \quad \forall b \in B, i \in I$$

$$u_{ib} \leq \sum_{j \in I} x_{jb} \quad \forall b \in B, i \in N$$

$$s_{0i}^v + \sum_{j \in B} s_{ji}^v = y_{iv} \quad \forall v \in V, i \in B : i \neq j$$

$$s_{i0}^v + \sum_{j \in B} s_{ij}^v = y_{iv} \quad \forall v \in V, i \in B : i \neq j$$

$$\sum_{i \in B} y_{iv} \leq M \times \sum_{j \in B} s_{0j}^v \quad \forall v \in V$$

$$\sum_{i \in B} y_{iv} \leq M \times \sum_{j \in B} s_{j0}^v \quad \forall v \in V$$

$$s_{bb}^v = 0 \quad \forall v \in V, b \in B \cup DE$$

$$u_{b0v} = 0 \quad \forall v \in V$$

$$u_{bv} + s_{ij}^v \leq u_{bv} + M \times (1 - s_{ij}^v) \quad \forall v \in V, i \in B \cup DE, j \in B : i \neq j$$

$$u_{bv} \leq M \times \sum_{j \in B \cup DE : i \neq j} s_{ij}^v \quad \forall v \in V, i \in B$$

$$ub_{iv} \leq \sum_{j \in B} y_{jv} \quad \forall v \in V, i \in B \cup DE \quad (31)$$

$$(8) \quad cb_i \leq sb_j + M \times (1 - s_{ij}^v) \quad \forall v \in V, i \in B, j \in B : i \neq j \quad (32)$$

$$(9) \quad sb_j \leq cb_i + M \times (1 - s_{ij}^v) \quad \forall v \in V, i \in B, j \in B : i \neq j \quad (33)$$

$$(10) \quad sb_i \leq 0 + M \times (1 - s_{0i}^v) \quad \forall v \in V, i \in B \quad (34)$$

$$(11) \quad 0 \leq sb_i + M \times (1 - s_{0i}^v) \quad \forall v \in V, i \in B \quad (35)$$

$$(12) \quad c_i + DT_{ij} \leq c_j + M \times (1 - r_{ij}^b) \quad \forall b \in B, i \in I, j \in I : i \neq j \quad (36)$$

$$(13) \quad c_j \leq c_i + DT_{ij} + M \times (1 - r_{ij}^b) \quad \forall b \in B, i \in I, j \in I : i \neq j \quad (37)$$

$$(14) \quad sb_b + DT_{0i} \leq c_i + M \times (1 - r_{0i}^b) \quad \forall b \in B, i \in I \quad (38)$$

$$(15) \quad c_i \leq sb_b + DT_{0i} + M \times (1 - r_{0i}^b) \quad \forall b \in B, i \in I \quad (39)$$

$$(16) \quad cb_b \leq c_i + DT_{i0} + M \times (1 - r_{i0}^b) \quad \forall b \in B, i \in I \quad (40)$$

$$(17) \quad c_i + DT_{i0} \leq cb_b + M \times (1 - r_{i0}^b) \quad \forall b \in B, i \in I \quad (41)$$

$$(18) \quad a_i, \quad x_{ib}, \quad y_{bv}, \quad z_i \in \{0, 1\} \quad \forall i \in I, \quad b \in B, \quad v \in V \quad (42)$$

$$(19) \quad r_{ij}^b \in \{0, 1\} \quad \forall i, j \in N, \quad b \in B \quad (43)$$

$$(20) \quad ui_{ib} \geq 0 \quad \forall i \in N, \quad b \in B \quad (44)$$

$$(21) \quad c_i, \quad E_i, \quad T_i, \quad sb_b, \quad cb_b \geq 0 \quad \forall i \in I, \quad b \in B \quad (45)$$

$$(22) \quad s_{ij}^v \in \{0, 1\} \quad \forall i, j \in B \cup DE, \quad v \in V \quad (46)$$

$$(23) \quad ub_{bv} \geq 0 \quad \forall b \in B \cup DE, \quad v \in V \quad (47)$$

The objective function of VRSP-OAPRL is to maximize the  $TP$  composed of  $DR$ ,  $DFC$ , and  $ETC$ . Eq. (9) ensures that one accepted pharmaceutical product must be assigned to one delivery batch. Eq. (10) ensures the capacity constraint described in Section 1. Eq. (11) ensures the temperature constraint described in Section 1 and determines the temperature of the delivery batch during the routing. Eqs. (12) and (13) determine the assigning of the delivery batches to the vehicles. Eq. (13) prevents the assignment of an empty delivery batch to any vehicle. Eqs. (14) to (22) determine the routing process described in Section 1. Eqs. (14) and (15) determine the routing of the accepted pharmaceutical products in the assigned delivery batch. If  $x_{ib}$  equals 1, it means that, in the delivery batch  $b$ , there must determine a route from node  $i$  to any node  $j$  or the depot and a route from any node  $j$  or the depot to node  $i$ . Eqs. (16) and (17) ensure that for a non-empty delivery batch, there must determine a route from the depot and a route to the depot. For delivery batch  $b$ , if  $\sum_{i \in I} x_{ib}$  does not equal 0, delivery batch  $b$  is not a non-empty delivery batch, and there must determine a route from the depot to any node  $j$  and a route from any node  $j$  to the depot. Otherwise, Eqs. (16) and (17) are relaxed. Eqs. (18) to (22) prevent sub-tours in the route of each delivery batch. Eqs. (23) to (31) determine the scheduling process described in Section 1. In the scheduling process, each delivery batch is considered a job, and vehicles are considered as parallel resources. For Eqs. (23) to (31), the logic similar to the constraints for the routing process is applied, and 0 represents a dummy job. Eqs. (23) and (24) determine the scheduling of the delivery batches in the assigned vehicle. If  $y_{iv}$  equals 1, it means that there must determine the delivery

sequence of the delivery batch  $i$  in vehicle  $v$ . Eqs. (25) and (26) ensure that if at least more than one delivery batch is assigned to the vehicle, the first and last delivery sequence among the assigned delivery batches must be determined in the vehicle. Eqs. (27) to (31) prevent sub-tours in the scheduling of each vehicle. Eqs. (32) to (35) calculate the delivery start time and completion time of the delivery batch in the vehicle. Eqs. (32) and (33) mean that if the delivery batch  $i$  immediately precedes the delivery batch  $j$  in the vehicle  $v$ ,  $s_{ij}^v = 1$ , the delivery starts time of the delivery batch  $j$  equals the delivery completion time of the delivery batch  $i$ ,  $cb_i = sb_j$ . Eqs. (34) and (35) indicate that the delivery start times of the first delivered batch are equal to 0. Eqs. (36) and (41) calculate the delivery completion times of the accepted pharmaceutical orders. Eqs. (36) and (37) ensure that if the pharmaceutical order  $i$  immediately precedes the pharmaceutical order  $j$  in the delivery batch  $b$ ,  $r_{ij}^b = 1$ , the delivery completion time of the pharmaceutical order  $j$  increases from the delivery completion time of the pharmaceutical order  $i$  to the delivery time from the node  $i$  to the node  $j$ ,  $c_j = c_i + DT_{ij}$ . Eqs. (38) and (39) represent the relationship between  $sb_b$  and  $c_i$ . If the pharmaceutical order  $i$  is delivered for the first time in the delivery batch  $b$ , the delivery completion time of the pharmaceutical order  $i$  increases from the delivery start time of the delivery batch  $b$  to the delivery time from the depot to node  $i$ ,  $c_i = sb_b + DT_{0i}$ . Eqs. (40) and (41) represent the relationship between  $cb_b$  and  $c_i$ . If the pharmaceutical order  $i$  is delivered for the last time in the delivery batch  $b$ , the delivery completion time of the delivery batch  $b$  increases from the delivery completion time of the pharmaceutical order  $i$  to the delivery time from node  $i$  to the depot,  $cb_b = c_i + DT_{i0}$ .

## 5. Algorithms

Holland [35] initially proposed the genetic algorithm (GA) inspired by natural selection and the theory of evaluation. GA has been used in variant studies of VRP and VRSP and has shown a good performance [36–38]. Since the proposed VRSP-OAPRL is NP-hard, the MILP described in Section 4 is not able to solve within a limited time. Thus, a hybrid genetic algorithm embedding an insertion heuristic with modification operation (HGAM) is proposed to solve the VRSP-OAPRL within a limited time.

### 5.1. Chromosome representation and decoding process

For HGAM, a chromosome consists of two one-dimensional arrays indicating the acceptance and insertion sequence of pharmaceutical orders with the length of  $|I|$ . The acceptance array in the chromosome is expressed as a binary  $[0, 1]$  representation. If the  $i$ th gene equals 1, the pharmaceutical order  $i$  is accepted, otherwise, it is not accepted. The insertion sequence array in the chromosome is expressed as a random-key  $U(0, 1)$  representation, and it means the insertion sequence of each order by sorting in ascending order. The chromosome is transformed into a pharmaceutical order list by combining the acceptance array with a sorted insertion sequence array. The list is composed of the acceptance and insertion sequences of pharmaceutical orders.

The insertion heuristic constructs the routes and schedules of vehicles by constantly inserting an un-routed node into the best position among insertable candidates. The insertion heuristic has shown good performance in the various studies of VRP and VRSP when it is combined with GA [39–42]. In this study, the insertion heuristic is used to construct the routes and schedules of each vehicle, aiming to minimize the earliness and tardiness costs. The insertion heuristic uses the pharmaceutical order list as input data. Given the sequence of pharmaceutical orders, each accepted pharmaceutical order is inserted into the best position with the lowest cost among candidates. The candidates consist

of every position of a partially built route of vehicles and the first sequence in an additional route of every vehicle. After the insertion of the pharmaceutical order, it is removed from the list. The modification operation is applied to the decoding process to reduce unnecessary disposal costs. When constructing the routes and schedules of the vehicles by using the insertion heuristic, each accepted order must be inserted into the candidate, ignoring some candidates with pharmaceutical disposal during the insertion process. Even if the current-inserting pharmaceutical order is delivered beyond its shelf life in every insertion candidate due to the inappropriate insertion sequence, the order must be inserted and delivered. The modification operation is to remove some insertion candidates or modify the acceptance gene of each accepted pharmaceutical order. An insertion candidate, in which the accepted pharmaceutical order of the current-insertion sequence is delivered beyond its shelf life, is removed from the candidate list by the modification operation. Additionally, if the current-insertion order is delivered beyond its shelf life in every insertion candidate, the order is considered as an order occurring the unnecessary delivery penalty cost, and the acceptance gene of the order is modified as 0 not to be inserted any routes of vehicles. The general pseudo-code of the insertion heuristic with modification operation is as follows:

#### Pseudo-code 1. : Insertion heuristic with modification operation

**Fig. 3** illustrates the example of the chromosome and the decoded solution. In the example, the data of customers, pharmaceutical orders, and vehicles from **Fig. 2** are utilized. As shown in **Fig. 3(a)**, the insertion sequence and acceptance arrays correspond to  $[0.04, 0.15, 0.64, 0.88, 0.18, 0.87, 0.36, 0.70, 0.44, 0.37, 0.89, 0.54, 0.84]$  and  $[0, 1, 1, 1, 1, 0, 1, 1, 1, 1, 1, 1]$ , respectively. The pharmaceutical orders  $i_1^1$  and  $i_3^2$  are rejected since acceptance genes are equal to 0. After incorporating acceptance and sorted insertion sequence arrays, the pharmaceutical order list is described in **Fig. 3(b)**. The list is used as the input data of the insertion heuristic, and the processes of constructing the routing and scheduling of vehicles using the insertion heuristic are described in **Fig. 3(c)**. For the first insertion sequence  $i_1^1$ , since it is rejected, no insertions occur. Order  $i_2^1$ , the second insertion sequence, is realistically the first insertion sequence. Thus, it is inserted into the first delivery sequence of the first delivery batch of vehicle 1. The storage temperature range of order  $i_1^2$  is overlapped with the storage temperature range of order  $i_2^1$ , as shown in **Fig. 2(a)**, and the capacity of vehicle 1 is not exceeded when order  $i_1^2$  is assigned with order  $i_2^1$ . Thus, order  $i_1^2$  is able to be inserted into one position among four candidates, as illustrated in Step 2 of **Fig. 3(c)**. Among the four candidates, if it is assumed that candidate 2 has the lowest cost, order  $i_1^2$  is inserted by following candidate 2. This insertion process of constructing the routing and scheduling is repeated until inserting every accepted order. **Fig. 3(d)** illustrates the decoded solution structure, where each string and \* represent the pharmaceutical order and depot, respectively. For vehicle1, two delivery batches are assigned, and the route and schedule correspond to the route and schedule of vehicle 1 in **Fig. 2(b)**. Similarly, two delivery batches are assigned to vehicle 2, and the route and schedule correspond to the route and schedule of vehicle 2 in **Fig. 2(b)**.

**Fig. 4** describes the modification operation using Step 2 in **Fig. 3(c)**. **Fig. 4(a)** represents a case in which delivery failures occur in some candidates. In **Fig. 4(a)**, it is assumed that order  $i_1^2$ , the current-insertion sequence, is delivered after its shelf life in candidates 1 and 3. Thus, candidates 1 and 3 are removed from the insertion candidate list. **Fig. 4(b)** represents the other case in which delivery failures occur in every candidate. If the delivery failure of order  $i_1^2$  occurs in every candidate, the acceptance gene of order  $i_1^2$  is modified to 0. Order  $i_1^2$  is not inserted in any route of vehicles because it is regarded as the order incurring unnecessary disposal cost and decreasing TP.

---

**Input:** Pharmaceutical order list  $L$

**Output:** Routing and scheduling of each vehicle

---

```

1: Begin
2:    $i \leftarrow 1$ 
3:   Let  $O_i, \forall i \in I$  as the pharmaceutical order of  $i$  sequence in the list  $L$ 
4:   Let  $A_i, \forall i \in I$  as the acceptance gene of pharmaceutical order of  $i$  sequence in the list  $L$ 
5:   While ( $i \leq |I|$ )
6:     If ( $A_{[i]} = 0$ )
7:       Remove pharmaceutical order  $i$  in the pharmaceutical order list  $L$ 
8:        $i \leftarrow i + 1$ 
9:     Else
10:    For ( $v \in V$ )
11:      Let  $L_v^1, \forall i \in I$  as the insertion candidate list of current tour of vehicle  $v$ 
12:      If ( $O_i$  satisfies temperature and capacity constraints in  $v$ )
13:        For ( $l \in L_v^1$ )
14:          Let  $c_i^{lv}$  as the delivery completion time of order  $i$  at candidate  $l$  in vehicle  $v$ 
15:          Calculate the delivery completion time  $c_i^{lv}$  of order  $i$  at candidate  $l$ 
16:          If ( $c_i^{lv} > D_i$ )
17:            Remove insertion candidate  $l$  in the list  $L_v^1$ 
18:          Else
19:            Calculate  $f_i(c_i^{lv})$  by following Eq. (1)
20:          End If
21:        End For
22:      Else
23:        Clear  $L_v^1$ 
24:      End If
25:      Virtually terminate the current tour and append  $l$ , the first position of new tour, into
26:       $L_v^1$  as a new candidate
27:      Calculate the delivery completion time  $c_i^{lv}$  of order  $i$  at candidate  $l$ 
28:      If ( $c_i^{lv} > D_i$ )
29:        Remove insertion candidate  $l$  in the list  $L_v^1$ 
30:      Else
31:        Calculate  $f_i(c_i^{lv})$  by following Eq. (1)
32:      End If
33:    End For
34:    If ( $L_v^1 = Null, \forall v \in V$ )
35:       $A_{[i]} \leftarrow 0$ 
36:      Remove pharmaceutical order  $i$  in the pharmaceutical order list  $L$ 
37:       $i \leftarrow i + 1$ 
38:    Else
39:      Insert the pharmaceutical order  $i$  into position  $l$  of vehicle  $v$  with the
40:      lowest  $f_i(c_i^{lv})$ 
41:      Update the temperature and capacity of the inserted vehicle  $v$ 
42:      Remove pharmaceutical order  $i$  in the pharmaceutical order list  $L$ 
43:       $i \leftarrow i + 1$ 
44:    End If
45:  End If
46: End While
47: End

```

---

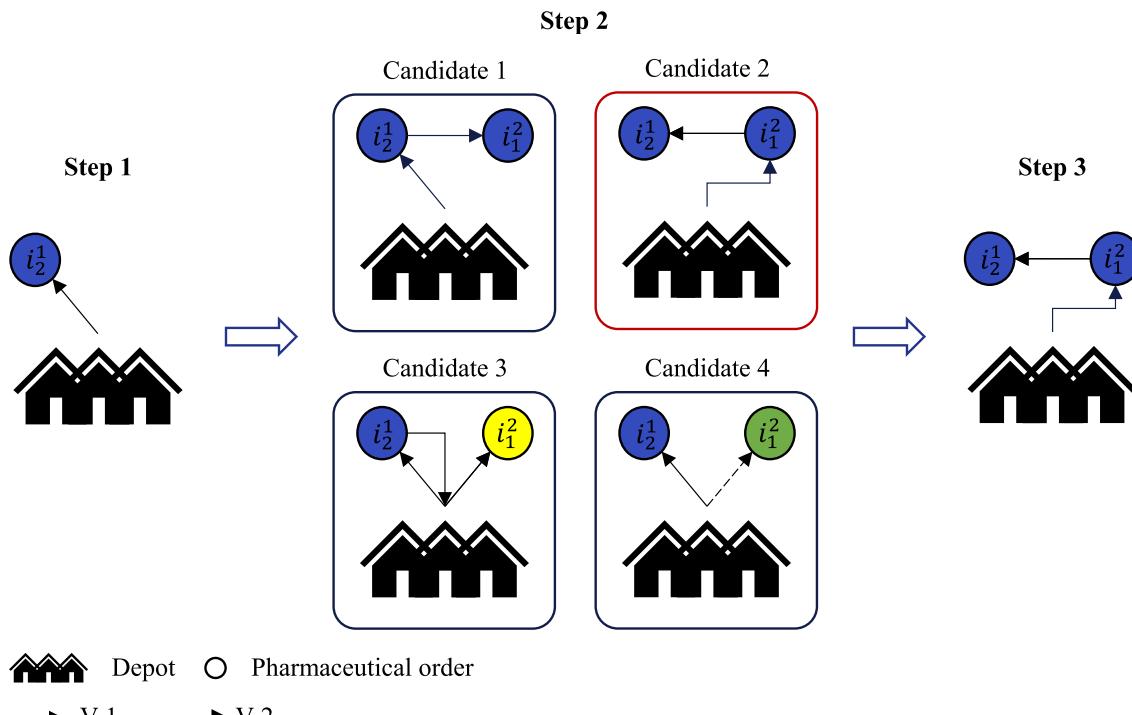
0	1	1	1	1	1	0	1	1	1	1	1	1
Insertion sequence array												

0.04	0.15	0.64	0.88	0.18	0.87	0.36	0.70	0.44	0.37	0.89	0.54	0.84
Acceptance array												

(a) Chromosome representation

Insertion Sequence	1	2	3	4	5	6	7	8	9	10	11	12	13
Acceptance	0	1	1	0	1	1	1	1	1	1	1	1	1
Order	$i_1^1$	$i_2^1$	$i_1^2$	$i_3^2$	$i_1^4$	$i_2^3$	$i_3^4$	$i_3^1$	$i_1^3$	$i_4^4$	$i_2^2$	$i_4^1$	$i_2^4$

(b) Pharmaceutical order list



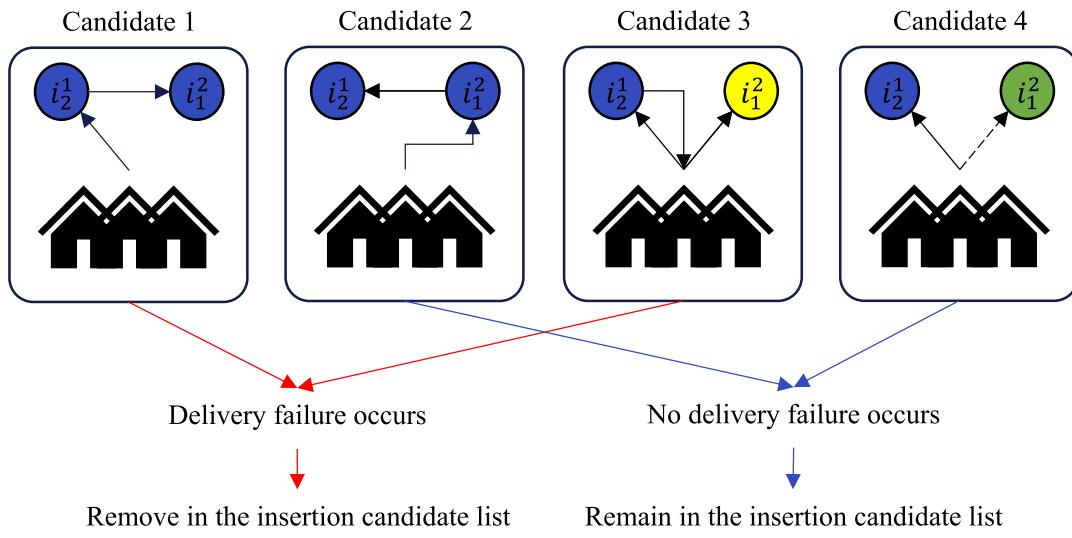
(c) Example of insertion heuristic

Reject	$i_1^1$	$i_3^2$							
V 1	*	$i_1^2$	$i_2^1$	$i_4^1$	*	$i_3^1$	$i_1^3$	*	
V 2	*	$i_2^3$	$i_3^4$	$i_1^4$	*	$i_2^4$	$i_4^4$	$i_2^2$	*

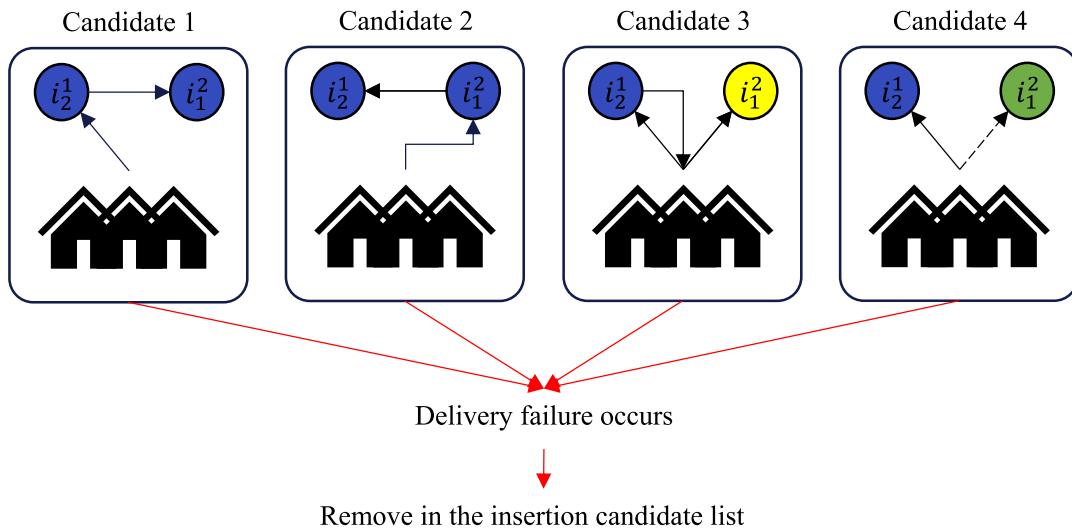
\*: Depot

(d) Decoded solution for VRSP-OAPRL

**Fig. 3.** Solution representation.



(a) Case1: Occurring the delivery failure in some candidates



Insertion Sequence	1	2	3	4	5	6	7	8	9	10	11	12	13
Acceptance	0	1	<b>0</b>	0	1	1	1	1	1	1	1	1	1
Order	$i_1^1$	$i_2^1$	$i_1^2$	$i_3^2$	$i_1^4$	$i_2^3$	$i_3^4$	$i_3^1$	$i_1^3$	$i_4^4$	$i_2^2$	$i_4^1$	$i_2^4$

(b) Case2: Occurring the delivery failure in every candidate

Fig. 4. Modification operation.

### 5.2. Genetic algorithm

The procedure of GA consists of initialization, decoding, evaluation, evolution operation, and genetic operation. In the initialization, the initial population is generated by following the distribution of each one-dimensional array randomly. Each chromosome is decoded to obtain  $TP$  in the decoding. In the evaluation, the fitness  $F(c)$  of each

chromosome  $c$  is calculated by following Eq. (48):

$$F(c) = \max(TP_c, \epsilon) \quad (48)$$

where  $TP_c$  and  $\epsilon$  are the total profit of each chromosome  $c$  and the small positive value to penalize the negative  $TP_c$ , respectively. A roulette wheel selection is used in the evolution operation. One-cut crossover, and bit-flip and uniform mutations are used in the genetic operations.

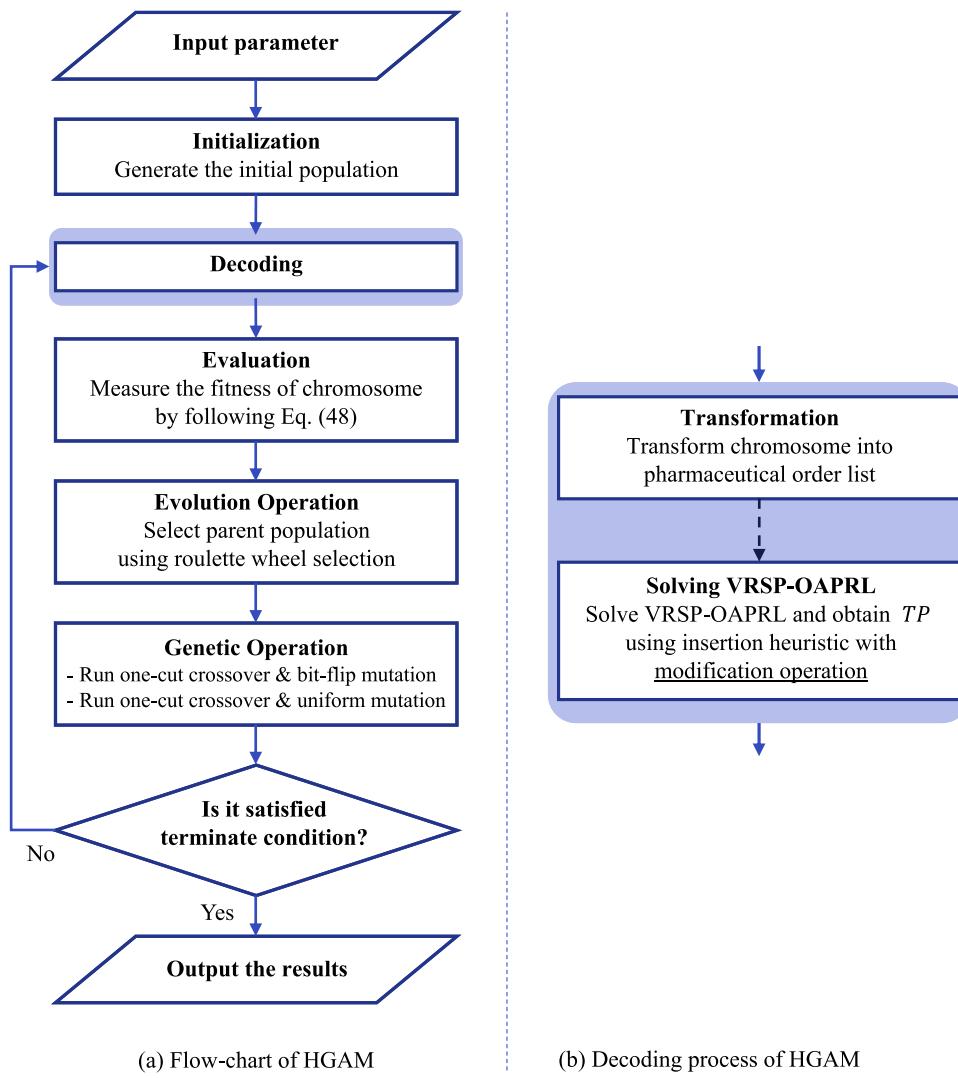


Fig. 5. Flow-chart of HGAM.

The current population is probabilistically selected by the roulette wheel selection, and a parent population is generated in the evolution operation. An offspring population is generated by tweaking the parent population in the genetic operation. For the one-cut crossover, two different parent chromosomes are swapped based on selected one cutting-point and generate two offspring chromosomes. The genes from the acceptance and the insertion sequence arrays in the offspring population are probabilistically changed by bit-flip and uniform mutations, respectively. For the acceptance array, probabilistically selected gene  $x_i$  is flipped as  $|x_i - 1|$ . For the insertion sequence array, probabilistically selected gene  $x_i$  is re-produced by using the uniform distribution  $U(0, 1)$ . The offspring population converts as the current population in the next generation, and GA is repeated until satisfying the stopping criteria, maximum generation. The flow-chart of HGAM is summarized in Fig. 5. Fig. 5(a) represents the general flow-chart of HGAM, except for the decoding process. The decoding process of HGAM is described in Fig. 5(b).

### 5.3. Particle swarm optimization and simulated annealing

Particle swarm optimization (PSO) and simulated annealing (SA) are representative metaheuristics, and various papers that studied the variants of the VRP and VRSP showed a good performance by adopting PSO and SA [43–46]. To compare the performance of the proposed HGAM,

hybrid PSO embedding an insertion heuristic with modification operation (HPSOM) and hybrid SA embedding an insertion heuristic with modification operation (HSAM) are addressed. The flow-charts of HPSOM and HSAM are summarized in Fig. 6. For HPSOM and HSAM, the identical encoded solution structure, decoding process, and evaluation operation with HGAM are used. The procedure of HPSOM consists of initialization, decoding, evaluation, and update operation [47]. In initialization, the swarm is initially generated by following the distribution of each one-dimensional array, randomly. Each particle is decoded to obtain  $TP$  in the decoding by using the insertion heuristic and modification operation proposed in Sections 5.1 and 5.2. In the evaluation, the fitness  $F(p)$  of each particle  $p$  is calculated by following Eq. (48). In the update operation, personal best position, global best position, swarm velocity, and swarm position are updated, and the insertion sequence array and the acceptance array are independently updated. The velocity and position vectors of the insertion sequence array is updated as follows:

$$V_p \leftarrow w \times V_p + c_1 \times U(0, 1) \times (P_p - X_p) + c_2 \times U(0, 1) \times (G - X_i) \quad (49)$$

$$X_p \leftarrow X_p + V_p \quad (50)$$

where  $w$ ,  $c_1$ , and  $c_2$  mean weight, and cognitive and social coefficients, respectively.  $V_p$ ,  $X_p$ , mean velocity and position of particle  $p$ , respectively.  $P_p$  and  $G$  means personal best of particle  $p$  and global best,

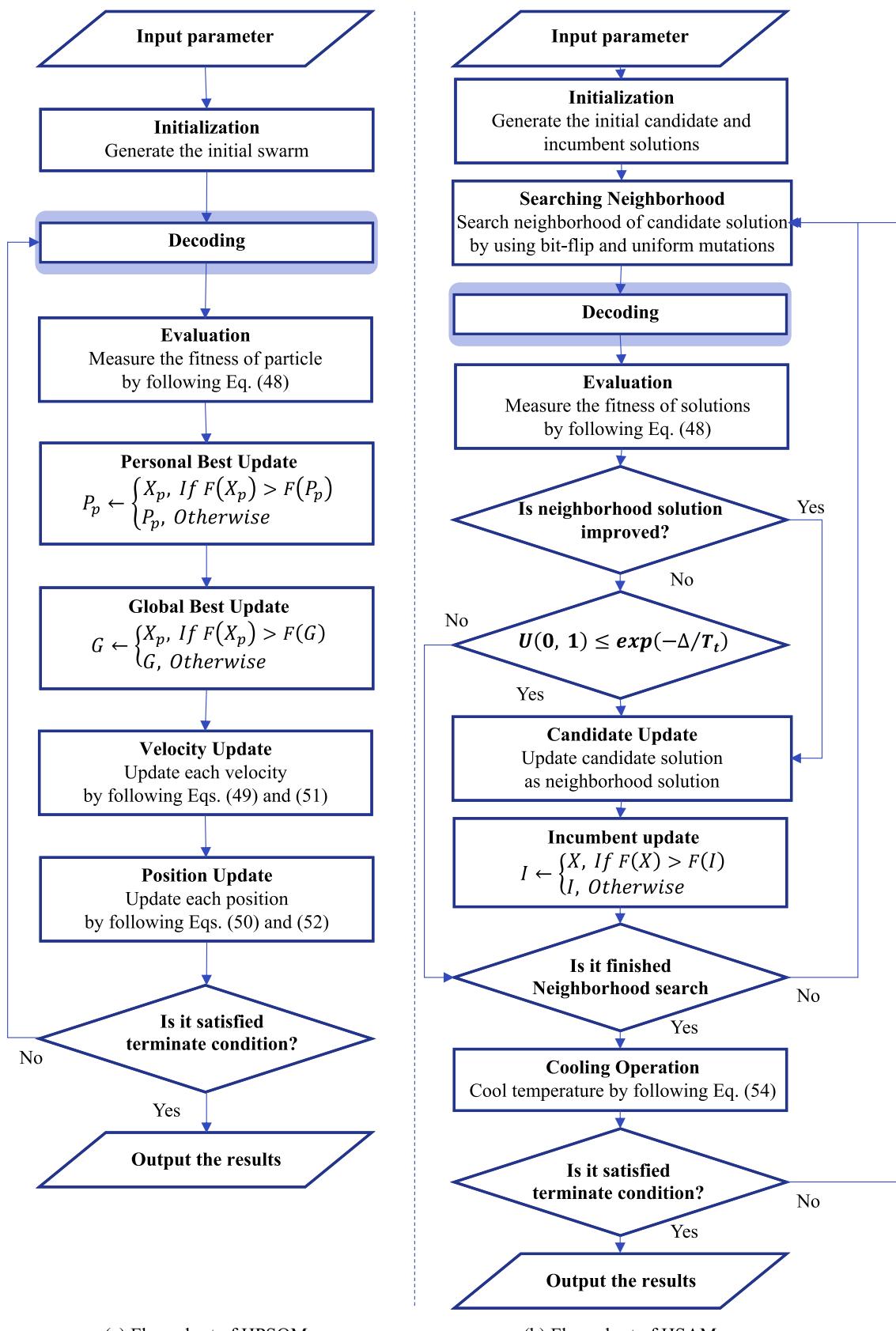


Fig. 6. Flow-chart of HPSOM and HSAM.

**Table 2**  
Design of experiments.

Experimental parameter	Small-sized experiment (Level)	Large-sized experiment (Level)
Number of customers $n_{cus}$	4 (1)	20 (1)
Number of orders $n_t$	{8, 10, 12} (3)	{100, 150, 200} (3)
Number of vehicles $n_v$	{2, 3} (2)	{4, 5, 6} (3)
Capacity CA	{2, 3} (2)	{40, 50, 60} (3)
Number of orders of customer $j O_j$	$U(1, 2), U(1, 3), U(1, 4)$ (-)	$U(3, 7), U(5, 10), U(7, 13)$ (-)
Coordinate of depot	(0, 0)	
Coordinate of customer	$(U(-4, 4), U(-4, 4))$	$(U(-20, 20), U(-20, 20))$
Delivery time from nodes $i$ to $j$ $DT_{ij}$	Manhattan distance	
Storage temperature window of order $i$ $[ST_i^{LB}, ST_i^{UB}]$	$U(-100, 0)$ (1)	$U(-100, 0)$ (1)
Weight of order $i W_i$	1 (1)	$U(5, 15)$ (1)
Revenue of order $i R_i$	$U(5, 20)$ (1)	$U(50, 200)$ (1)
Weight of ETC $\alpha$	0.001 (1)	0.003 (1)
Maximum due date $T_{max}$	20 (1)	480 (1)
Due date of order $i D_i$	$U(\min(DT_{ij}), T_{max})$ (1)	$U(\min(DT_{ij}), T_{max})$ (1)
Shelf life of order $i S_i$	$D_i + \overline{DT} \times (C/W_i) \times U(0, 1)$ (1)	$D_i + \overline{DT} \times (C/W_i) \times U(0, 1)$ (1)

respectively. For the acceptance array, the velocity vector is updated by following Eq. (49). However, since the acceptance array uses binary representation, the position vector of the acceptance array is updated by following distinct position update equations addressed in Marini and Walczak [47] to avoid infeasible solution that result in real number. The position update equations are as followed:

$$V'_{pi} \leftarrow \frac{1}{1 + e^{v_{pi}}} \quad (51)$$

$$X'_{pi} \leftarrow \begin{cases} |X_{pi} - 1|, & \text{if } V'_{pi} > s \\ X_{pi}, & \text{if } V'_{pi} \leq s \end{cases} \quad (52)$$

where  $V_{pi}$ ,  $X_{pi}$ , and  $s$  are velocity, position of  $i$ th dimension in particle  $p$ , and random value following  $U(0, 1)$ . The procedure of HSAM consists of initialization, searching neighborhood, decoding, evaluation, update operation, and cooling temperature. In the initialization, candidate solution  $X$  and incumbent solution  $I$  are randomly generated by following the distribution of each one-dimensional array. In the searching neighborhood, bit-flip mutation, and uniform mutation in HGAM are used to generate the neighborhood solution  $X'$  of the candidate solution by swapping the acceptance and insertion sequence arrays, respectively. In the update operation, the candidate solution and incumbent solution are updated. To probabilistic allow non-improving move in the candidate update, the gap  $\Delta$  between  $F(X)$  and  $F(X')$  is calculated as follows::

$$\Delta \leftarrow F(X) - F(X') \quad (53)$$

In the incumbent update, only improving moving is allowed between  $X$  and  $I$ . In the cooling temperature, a temperature is cooled as follows:

$$T_{t+1} \leftarrow T_t \times \alpha \quad (54)$$

where  $T_t$  and  $\alpha$  are temperature at cooling iteration  $t$  and cooling rate, respectively.

## 6. Numerical experiments

To evaluate the performance of the proposed algorithms and suggest

Ins.	I	V	CA	BTP	TP	CPU Time	MLP		HSA		HSAM		HPSO		HGAM	
							MRPD	CPU								
1	8	2	2	94.74*	94.74	158.36	0.00	14.66	0.04	19.25	0.00	18.06	0.00	22.32	0.00	20.97
2	2	3	3	59.73*	59.73	35.45	0.00	11.48	0.00	11.46	0.00	14.82	0.00	13.33	0.00	12.30
3	3	2	2	59.63*	59.63	185.98	0.00	9.99	0.00	9.94	0.10	12.90	0.22	10.64	0.00	9.79
4	4	3	3	94.68*	94.68	598.02	0.00	7.57	0.00	7.60	0.17	9.46	0.17	8.53	0.00	12.55
5	5	10	2	2	106.77	3600++	0.00	21.43	0.00	21.44	0.00	28.61	0.00	26.70	0.00	26.10
6	6	3	3	105.65	105.65	3600++	0.00	16.14	0.00	16.12	0.00	21.05	0.00	20.11	0.00	24.38
7	7	3	2	116.72	116.72	3600++	0.00	16.06	0.00	15.95	0.00	21.08	0.00	19.59	0.00	24.30
8	8	3	3	98.79	98.79	3600++	0.00	13.04	0.00	12.99	0.03	16.35	0.05	14.13	0.00	17.19
9	9	12	2	2	106.40	3600++	0.95	27.75	0.65	27.80	1.68	37.64	1.76	16.85	0.00	41.73
10	10	3	3	136.87	136.87	3600++	0.00	22.88	0.00	23.13	4.36	30.37	4.48	28.01	0.00	32.82
11	11	3	2	128.66	128.66	3600++	3.05	22.00	3.95	22.07	3.05	28.72	3.95	28.14	3.05	31.21
12	12	3	3	140.61	140.61	3600++	0.00	18.90	0.01	18.83	0.10	11.26	0.10	22.02	0.00	35.04
			Avg.	104.10	103.60	244.45	0.33	16.83	0.39	16.83	0.79	20.96	0.90	18.69	0.26	24.59

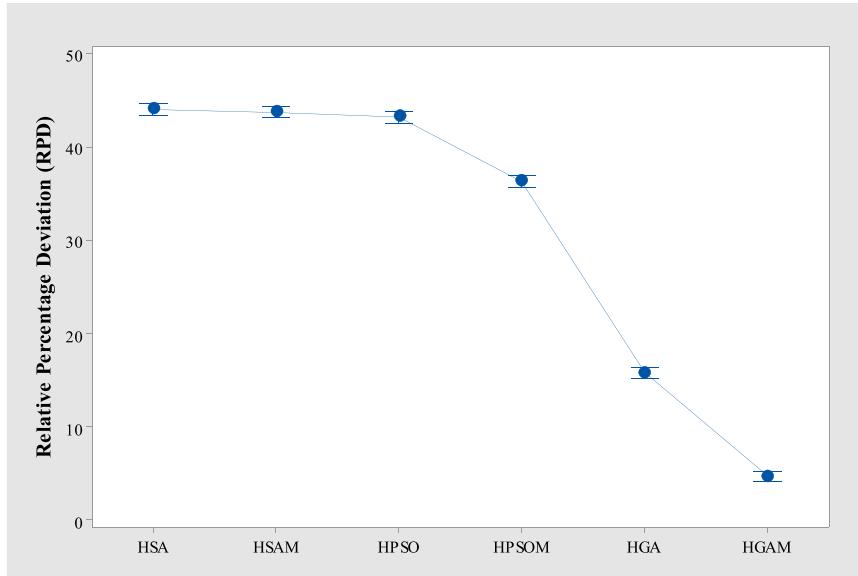
**Table 3**  
Results of small-sized experiments.

- BTP marked with bold font and \* indicates the optimal solution

**Table 4**

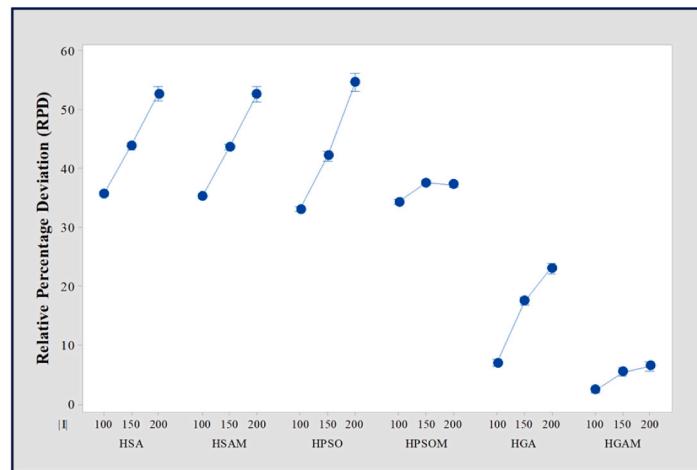
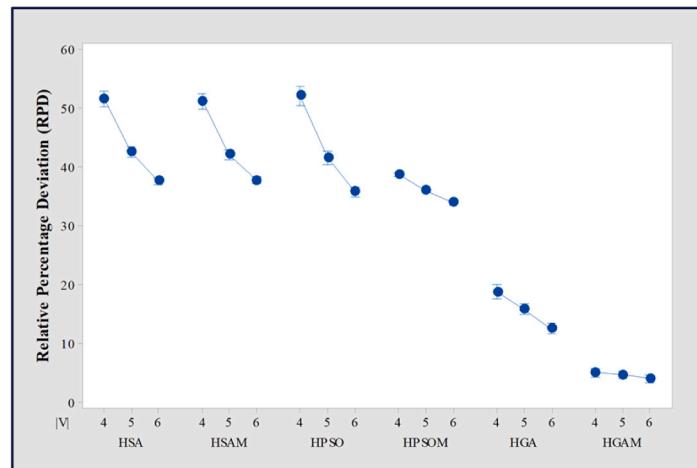
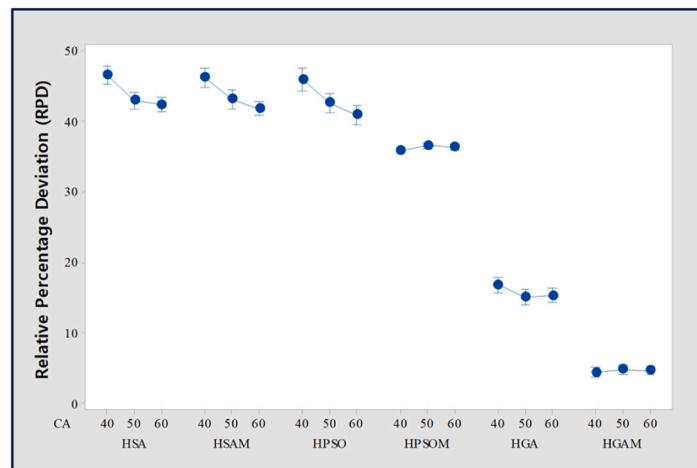
Results of large-sized experiments.

Ins.	I	V	CA	BTP	HSA		HSAM		HPSO		HPSOM		HGA		HGAM		
					MRPD	CPU											
1	100	4	40	9811.56	39.02	46.74	39.07	46.59	38.63	59.00	32.55	46.81	4.43	71.08	0.19	60.36	
2		50		10400.76	41.20	43.75	39.51	43.62	37.32	53.19	39.10	43.09	6.22	71.55	3.62	63.73	
3		60		12080.08	38.33	33.08	37.36	32.96	37.54	40.33	39.25	35.92	7.15	69.14	0.89	64.18	
4		5	40	10097.61	35.18	43.30	34.96	43.42	30.30	54.86	32.69	46.08	6.66	77.98	1.37	73.45	
5		50		11348.01	34.38	34.91	34.52		35.00	33.79	43.56	35.43	37.15	7.21	64.92	1.18	62.81
6		60		11720.09	34.64	31.06	33.36	30.92	33.13	37.83	32.30	32.39	7.84	67.55	1.59	62.51	
7		6	40	11619.47	34.48	37.30	32.78	37.33	29.25	47.02	34.24	39.51	2.26	77.85	0.51	67.79	
8		50		11258.14	29.02	28.20	30.52	27.91	29.14	32.79	31.37	30.55	4.28	55.19	0.12	54.11	
9		60		10716.64	31.15	21.38	31.91	21.41	26.83	25.07	28.04	23.32	0.65	47.04	3.30	45.60	
10	150	4	40	13290.68	53.00	108.89	50.81	107.82	53.10	131.60	41.67	101.02	23.65	118.51	5.18	111.94	
11		50		13281.18	49.21	86.56	48.77	86.07	47.92	106.80	38.04	78.53	17.31	106.88	5.48	97.14	
12		60		14058.82	47.57	82.32	46.56	82.35	43.97	99.69	37.02	76.47	11.67	128.17	2.86	111.35	
13		5	40	14828.49	42.43	93.46	42.90	93.64	43.41	113.08	38.51	93.46	19.00	124.22	1.94	126.65	
14		50		14512.06	40.55	89.73	38.95	89.80	36.50	109.19	33.94	86.66	10.65	131.05	3.10	122.38	
15		60		15222.25	44.26	80.53	42.92	80.67	42.08	96.92	38.10	75.50	17.40	114.85	3.61	118.87	
16		6	40	15512.45	39.02	91.08	39.45	173.09	37.06	104.03	32.17	93.40	12.42	139.46	2.46	127.56	
17		50		16103.08	37.18	81.47	37.42	152.45	36.67	97.53	37.46	151.36	16.06	141.35	2.68	218.91	
18		60		16452.96	38.37	66.50	38.90	124.51	35.99	79.76	38.04	126.12	14.17	117.47	4.16	218.12	
19	200	4	40	13913.79	68.81	199.02	67.53	373.89	70.63	243.55	35.12	353.08	26.22	178.56	3.67	294.41	
20		50		14785.00	63.43	344.92	69.30	344.98	66.06	226.85	43.67	305.10	31.68	184.26	3.09	296.88	
21		60		15735.99	60.93	294.61	57.56	294.60	69.73	356.89	41.16	272.84	29.35	328.06	5.62	292.24	
22		5	40	15626.48	57.61	333.97	58.95	334.12	61.24	393.66	37.80	323.82	23.57	357.93	5.15	290.23	
23		50		16793.25	46.21	321.62	45.58	321.87	51.47	382.17	35.54	296.35	16.48	389.26	4.51	333.69	
24		60		17095.52	45.24	246.60	43.92	246.71	41.66	295.65	37.55	225.70	22.00	323.22	6.28	273.78	
25		6	40	18034.51	47.95	309.89	47.19	310.18	48.60	376.62	36.01	301.06	23.53	362.79	4.41	320.73	
26		50		18799.50	38.56	271.70	38.87	272.26	40.74	324.62	33.62	252.91	10.33	396.55	3.47	341.64	
27		60		18511.91	38.23	238.90	39.42	238.95	36.48	283.24	35.03	228.10	16.68	341.31	2.52	298.65	
		Avg.		14133.71	43.55	135.61	43.30	149.89	42.94	156.13	36.13	139.86	14.40	169.86	3.07	168.51	

**Fig. 7.** Interval plot with 0.95 significance level.

the appropriate algorithm for VRSP-OAPRL, numerical experiments are conducted with randomly generated experimental instances. Since the size of  $|I|$ ,  $|V|$ , and CA affect the complexity of VRSP-OAPRL, the experimental instances are categorized as small and large-sized experimental instances depending upon the size of  $|I|$ ,  $|V|$ , and CA. For small-sized experimental instances,  $|I|$ ,  $|V|$ , and CA are defined as set {8, 10, 12}, {2, 3}, and {2, 3}, respectively. For large-sized experimental instances,  $|I|$ ,  $|V|$ , and CA are defined as set {100, 150, 200}, {4, 5, 6}, and {40, 50, 60}, respectively. The numbers of customers  $n_{cus}$  are fixed as 4 and 20 in small and large-sized experimental instances, respectively. Since  $|I|$  equals to the product of  $|C|$  and the average of the number of orders by the customer,  $\bar{O}_j$ , the uniform

distribution generating  $O_j$  differs following the size of  $|I|$ . The coordinate of the depot is fixed as (0, 0) in every experimental instance. The coordinates of the customers are randomly generated within the problem space of each experimental instance, and all coordinates of the customers are different. Each problem space of experimental instances is  $[-|C|, |C|] \times [-|C|, |C|]$ , which represents the range of the X-axis and Y-axis. The  $DT_{ij}$  equals the distance between nodes  $i$  to  $j$ , and the distance is calculated by using the Manhattan distance.  $D_i$  and  $S_i$  are able to be used to express the time window of node  $i$ ,  $[D_i, S_i]$ , and ETC and DC are incurred by following  $D_i$  and  $S_i$ .  $S_i$  is generated by considering  $D_i$  and the expected average total delivery time of a tour,  $\bar{DT} \times (CA/\bar{W}_i)$ . Additionally, to prevent an explosive increase in  $S_i$ , a value following

(a) Interval plot with 0.95 significance level for  $|I|$ (b) Interval plot with 0.95 significance level for  $|V|$ (c) Interval plot with 0.95 significance level for  $CA$ **Fig. 8.** Interval plot with 0.95 significance level for each experimental parameter.

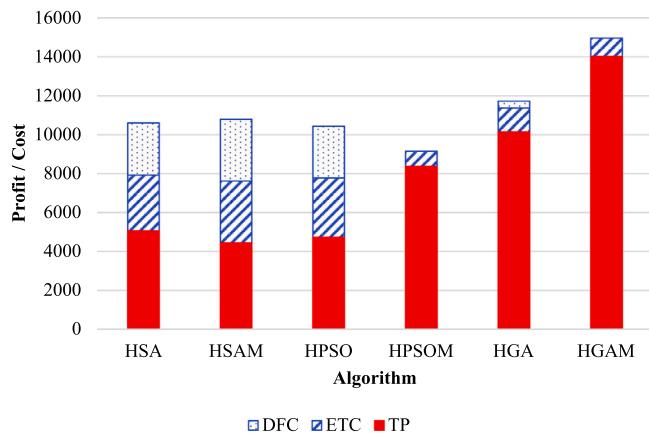


Fig. 9. Bar chart for results of instance  $(|I|, |V|, CA)$  as  $(200, 4, 50)$ .

$U(0, 1)$  was multiplied as the weight. Table 2 indicates the design of experiments.

In this section, the performances of HGAM are compared to five hybrid metaheuristics, which are HPSOM, HSAM, hybrid genetic algorithm (HGA) hybrid particle swarm optimization (HPSO), and hybrid simulated annealing (HSA). HGA, HPSO, and HSA are hybrid metaheuristics that only include the insertion heuristic and not the modification operation. For HGA, HPSO, and HSA, the identical solution structure and decoding process presented in Appendix A are used. Also, except for the decoding process, HGA, HPSO, and HSA procedures are identical to the flow-charts illustrated in Figs. 5 and 6, respectively.

The two types of the numerical experiments with randomly generated experimental instances are conducted. The relative percentage deviation ( $RPD$ ) is used as the performance measure in the numerical experiments.  $RPD$  is able to be calculated as follows:

$$RPD(\%) = \frac{BEST - ALG_{sol}}{BEST} \times 100, \quad (55)$$

where, for each experimental instance,  $ALG_{sol}$  is the solution obtained by the proposed algorithm.  $BEST$  is the best solution among the solutions obtained by the proposed algorithms or MILP model implemented in the solver. In small-sized experiments, if the MILP model finds an optimal solution within a limited time,  $BEST$  means the optimal solution. The size of the population ( $SP$ ), rate of crossover ( $RC$ ), rate of mutation ( $RM$ ), and the maximum number of generations ( $MG$ ) are used as the algorithm parameters for HGA and HGAM. The size of the swarm ( $SS$ ), weight ( $W$ ), rate of cognitive coefficient ( $C1$ ), rate of social coefficient ( $C2$ ), and maximum number of iterations ( $MI$ ) are used as the algorithm parameters for HPSO and HPSOM. The number of the neighborhood search ( $NN$ ), cooling rate ( $CR$ ) and maximum number of cooling iterations ( $MC$ ) are used as the algorithm parameters for HSA and HSAM. The expected number of tours of each vehicle ( $\widehat{ET}$ ) are used to define the  $SP$ ,  $SS$ , and  $NN$ , and  $\widehat{EN}$  is calculated as follows:

$$\widehat{ET} = \frac{|I| \times \overline{W}_i}{|V| \times CA}, \quad (56)$$

where  $\overline{W}_i$  is the average of weight of pharmaceutical order  $i$ . ( $SP, RC, RN, MG$ ) for HGA and HGAM are  $(\widehat{ET} \times 2, 0.35, 0.06, 1000)$  and  $(\widehat{ET} \times 2, 0.30, 0.04, 1000)$ , respectively. ( $SS, W, C1, C2, MI$ ) for HPSO and HPSOM are  $(\widehat{ET} \times 2, 0.50, 0.30, 0.30, 1000)$  and  $(\widehat{ET} \times 2, 0.30, 0.10, 0.30, 1000)$ , respectively. ( $NN, CR, MC$ ) for HSA and HSAM are  $(\widehat{ET} \times 2, 0.97, 1000)$  and  $(\widehat{ET} \times 2, 0.98, 1000)$ , respectively. For every experimental instance, the proposed six metaheuristic algorithms conduct the 30 replication tests. The MILP model described in Section 4 is implemented in IBM ILOG CPLEX 22.10.00. All proposed six metaheuristics are implemented in Python IDE. The proposed MILP

model metaheuristic algorithms are executed on an Intel Core i7-13700KF 3.40 GHz CPU.

### 6.1. Results of numerical experiments

The results of the small-sized experimental instances are summarized in Table 3. For each instance,  $BTP$  indicates the best total profit among the obtained solutions. In addition,  $TP$  indicates the objective value obtained by the MILP model implemented in the solver. If the MILP model finds an optimal solution,  $BTP$  is marked with bold font and \*. The median of  $RPD(MRPD)$  and the median of  $CPU$  time ( $CPU$ ) obtained by the proposed six metaheuristic algorithms are presented in Table 3. This is because the median presents not only a clear interpretation of the result but a more robust measure for stochastic search algorithms, such as PSO, GA, etc., than an arithmetic mean [48–50].  $CPU$  time uses the unit of second (Sec.). If MILP does not solve an optimal solution for an experimental instance within 3600 Secs., the best feasible solution and 3600++ are marked in  $TP$  and  $CPU$  time, respectively. Also,  $RPD$  is calculated by following above mentioned for each metaheuristic algorithm. In Table 3, all metaheuristic algorithms show the average of  $MRPD$  within 1.00 %. MILP model implemented in CPLEX is not able to optimally solve the instances with 10 or more orders within 3600 Secs..

The results of large-sized experimental instances are summarized in Table 4. For each large-sized experimental instance, the median of  $RPD(MRPD)$  and the median of  $CPU$  time ( $CPU$ ) obtained by proposed metaheuristic algorithms are presented in Table 4. This is also to use the advantages of the median as the performance measure for stochastic search algorithms, such as PSO, GA, etc. [48–50] in large-sized experiments. In addition,  $BTP$  is the best total profit among the obtained solutions for each instance. The metaheuristics HSA and HSAM, implemented as SA, show the average of  $MRPD$  as 44.00 and 43.73, respectively. The metaheuristics HPSO and HPSOM, implemented as PSO, show the average of  $MRPD$  as 43.20 and 36.29, respectively. Meanwhile, the average of  $MRPD$  HGA and HGAM are 15.77 and 4.67, respectively. From the results of  $MRPDs$ , the metaheuristics implemented as GA show a relatively better average of  $RPD$  than other metaheuristics implemented as SA and PSO. Especially, HGAM shows the best average of  $MRPD$  less than 5 %. Fig. 7 indicates the interval plot with a 0.95 significance level of the large-sized experiment results. As shown in Table 4 and Fig. 7, HGAM has the lowest average of  $MRPD$  among the proposed metaheuristic algorithms, and there is no overlap in the interval plot of the HGAM with other interval plots. This result indicates that HGAM is statistically significant.

Graphs (a) - (c) in Fig. 8 represent the interval plot with a 0.95 significance level of the large-sized experiment results for experimental parameters  $|I|$ ,  $|V|$ , and  $CA$ , respectively. In Graph (a) – (c), HGAM exhibits the lowest mean and tightest confidence interval for each parameter level. It implies that HGAM is the most robust algorithm among the proposed algorithms for parameters  $|I|$ ,  $|V|$ , and  $CA$ . HGAM shows more variability for changing  $|I|$  than changing  $|V|$  and  $CA$ . It means that  $|I|$  more affects the performance of HGAM than  $|V|$  and  $CA$ . Fig. 9 represents the bar chart for the average value of  $DR$  consisting of  $TP$ ,  $ETC$ , and  $DFC$  for the six proposed algorithms in instance  $(|I|, |V|, CA)$  as  $(200, 4, 50)$ . As shown in Fig. 9, HGAM shows the largest value for  $TP$  and the lowest values for the summation of  $ETC$  and  $DFC$  among the six metaheuristic algorithms. This means that HGAM effectively decides the acceptance of pharmaceutical orders and the routing and scheduling of each RV. HGAM performs well within a reasonable  $CPU$  time with the proposed algorithms. Therefore, HGAM is the most appropriate algorithm for VRSP-OAPRL among the proposed algorithms.

### 6.2. Sensitivity analysis for OA strategy

The sensitivity analysis with HGAM is conducted to address the impacts of the OA strategy on the PCL company according to the weight

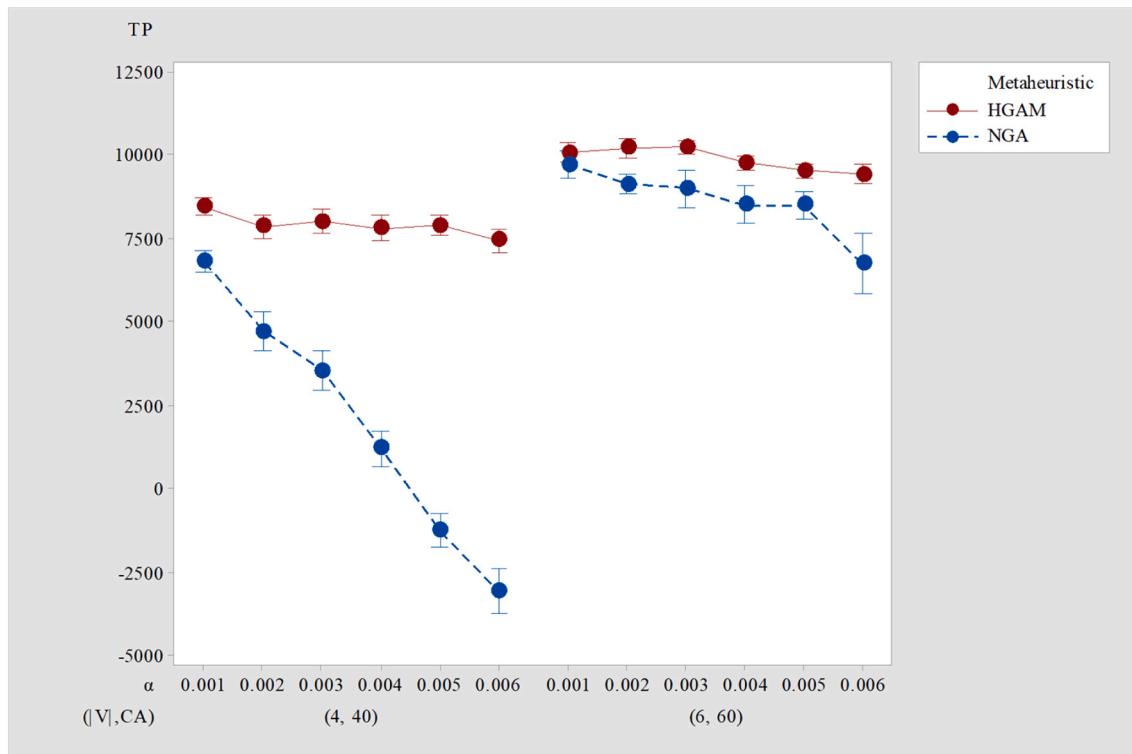
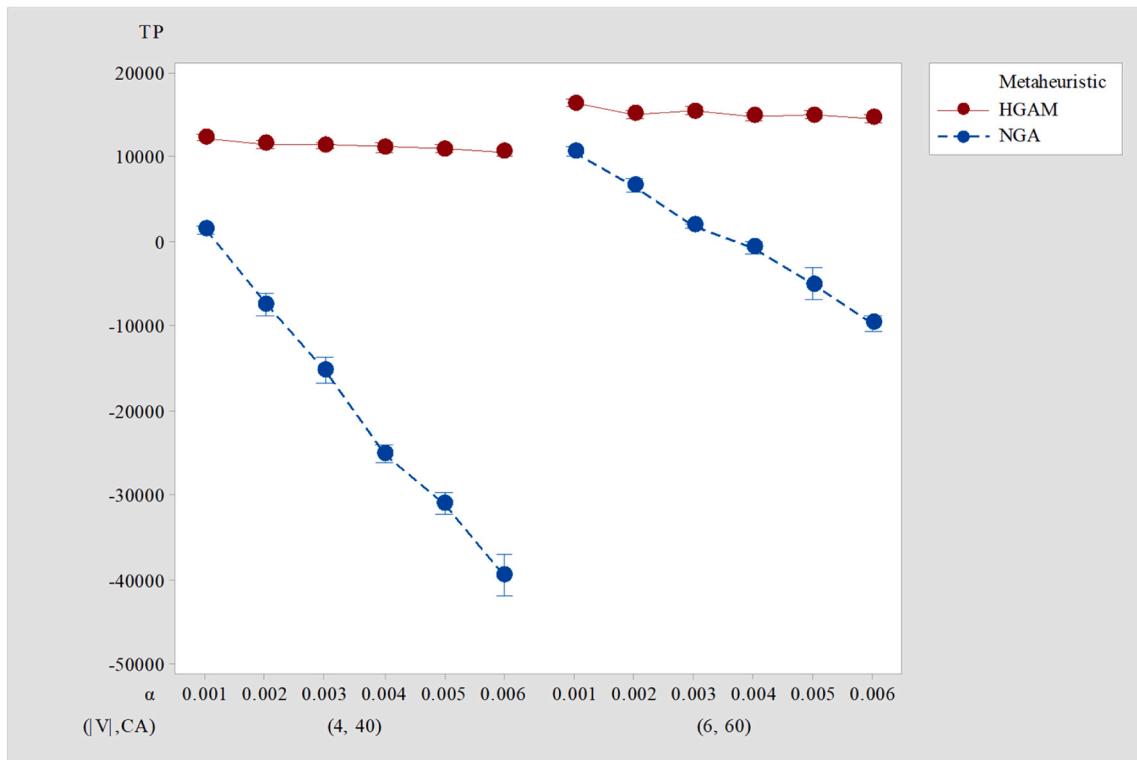
(a) Interval plot with 0.95 significance level for  $|I|$  as 100(b) Interval plot with 0.95 significance level for  $|I|$  as 200

Fig. 10. Interval plot with 0.95 significance level for sensitivty analysis.

of ETC,  $\alpha$ , in various experimental environments. HGAM and NGA (HGAM without the OA strategy) are compared in the sensitivity analysis. Since the changing of  $|I|$  affects the mean plot of HGAM compared to other experimental parameters  $|V|$  and CA as shown in Fig. 8, the experimental instances of the sensitivity analysis are configured with  $|I|$  and  $(|V|, CA)$  as {100, 200} and {(4, 40), (6, 60)}, respectively. The experimental environment  $(|V|, CA)$  configured with (4, 40) represents an environment with insufficient delivery resources, and the other one represents an environment with relatively sufficient delivery resources. The values of  $\alpha$  are defined as {0.001, 0.002, 0.003, 0.004, 0.005, 0.006}. Each experimental instance is repeated 10 times for each  $\alpha$ .

Fig. 10 (a) represents the interval plot of TP with 0.95 significance level for the parameter  $|I|$  as 100. Within the identical  $(|V|, CA)$  environment, HGAM shows the robust TP as  $\alpha$  changes. However, NGA shows decreased TP as  $\alpha$  changes. Especially, NGA shows rapidly decreased TP in the environment configured  $(|V|, CA)$  as (4, 40). Also, NGA shows the negative TP in which  $\alpha$  is more than 0.005 in the environment configured  $(|V|, CA)$  as (4, 40). Additionally, the mean plots of HGAM and NGA are overlapped in the environment configured  $(|I|, |V|, CA, \alpha)$  as (100, 4, 40, 0.001), and it indicates that HGAM and NGA are not a statistically significant in this environment. Fig. 10 (b) represents the mean plot of TP with 0.95 significance level for the parameter  $|I|$  as 200. HGAM shows the less variation of TP compared with NGA as  $\alpha$  change in the identical experimental environment. NGA shows the negative TP in which  $\alpha$  is more than 0.001 and 0.004 in both environments configured  $(|V|, CA)$  as (4, 40) and (6, 60), respectively. Since HGAM, unlike NGA, is able to accept or reject the pharmaceutical orders by the revenue, costs, and delivery capacity considered together, HGAM shows more robust TP than NGA as  $\alpha$  changes in the same experimental environment  $(|I|, |V|, CA)$ .

The result of the sensitivity analysis implies that the OA strategy is more effective than the strategy without determining the acceptance of orders to maximize the profit. The OA strategy is especially effective for the PCL companies with high service costs for earliness and tardiness delivery and insufficient delivery resources relative to the order quantity. However, the OA strategy is not significantly effective for the PCL companies with the low service costs and sufficient delivery resources relative to the order quantity.

## 7. Conclusion

In this paper, a delivery problem called VRSP-OAPRL is studied in a pharmaceutical cold chain. The problem simultaneously considers the

storage temperature, just-in-time delivery, the strict shelf life of pharmaceuticals, and order acceptance. To optimally solve the problem, MILP is derived. HGAM is proposed to effectively and efficiently solve VRSP-OAPRL in the large-sized instances. In the numerical experiments, HGAM shows a good RPD performance compared with HSA, HSAM, HPSO, HPSOM, and HGA within a reasonable computing time. To offer managerial insights about the order acceptance strategy for the PCL company, the sensitivity analyses are conducted by comparing HGAM to NGA. The order acceptance strategy is effective when the weight of earliness and tardiness costs is high, and delivery resources and load capacities are relatively insufficient than the number of orders. From the results, the paper shows the managerial insight that the OA strategy is especially effective to the PCL companies with high service costs for earliness and tardiness delivery and insufficient delivery resources relative to the order quantity.

In future works, VRSP-OAPRL will be extended to three problems: the delivery problem with a heterogenous capacity of each vehicle, the integrated problem with manufacturing and delivery, and the delivery problem with controlling the inventory.

## CRediT authorship contribution statement

**Byung Soo Kim:** Writing – review & editing, Validation, Supervision, Methodology, Formal analysis, Conceptualization. **Seung Jae Lee:** Writing – original draft, Visualization, Software, Methodology, Formal analysis, Conceptualization.

## Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

## Data availability

I have shared the link to my data

## Acknowledgements

This research was supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Science, ICT and Future Planning (Grant number: NRF-2022R1F1A1068304).

## Appendix A. . Chromosome representation and decoding process of HGA

For HGA, a chromosome consists of two one-dimensional arrays indicating the acceptance and insertion sequence of pharmaceutical orders with the length of  $|I|$ . The acceptance array in the chromosome is expressed as a binary [0, 1] representation. If the  $i$ th gene equals 1, the pharmaceutical order  $i$  is accepted, otherwise, it is not accepted. The insertion sequence array in the chromosome is expressed as a random-key-

$U(0, 1)$  representation, and it means the insertion sequence of each order by sorting in ascending order. The chromosome is transformed into a pharmaceutical order list by combining the acceptance array with a sorted insertion sequence array. The list is composed of the acceptance and insertion sequences of pharmaceutical orders.

The insertion heuristic constructs the routes and schedules of vehicles by constantly inserting an un-routed node into the best position among insertable candidates. The insertion heuristic uses the pharmaceutical order list as input data. Given the sequence of pharmaceutical orders, each accepted pharmaceutical order is inserted into the best position with the lowest cost among candidates. The candidates consist of every position of a partially built route of vehicles and the first sequence in an additional route of every vehicle. After the insertion of the pharmaceutical order, it is removed from the list. The insertion heuristic for HGA is as follows:

**Pseudo-code A1.** : Insertion heuristic

---

**Input:** Pharmaceutical order list  $L$

**Output:** Routing and scheduling of each vehicle

---

```

1: Begin
2:    $i \leftarrow 1$ 
3:   Let  $O_i, \forall i \in I$  as the pharmaceutical order of  $i$  sequence in the list  $L$ 
4:   Let  $A_i, \forall i \in I$  as the acceptance gene of pharmaceutical order of  $i$  sequence in the list  $L$ 
5:   While ( $i \leq |I|$ )
6:     If ( $A_{[i]} = 0$ )
7:       Remove pharmaceutical order  $i$  in the pharmaceutical order list  $L$ 
8:        $i \leftarrow i + 1$ 
9:     Else
10:      For ( $v \in V$ )
11:        Let  $L_v^1, \forall i \in I$  as the insertion candidate list of current tour of vehicle  $v$ 
12:        If ( $O_i$  satisfies temperature and capacity constraints in  $v$ )
13:          For ( $l \in L_v^1$ )
14:            Let  $c_i^{lv}$  as the delivery completion time of order  $i$  at candidate  $l$  in vehicle  $v$ 
15:            Calculate the delivery completion time  $c_i^{lv}$  of order  $i$  at candidate  $l$ 
16:            Calculate  $f_i(c_i^{lv})$  by following Eq. (1)
17:          End For
18:          Else
19:            Clear  $L_v^1$ 
20:          End If
21:          Virtually terminate the current tour and append  $l$ , the first position of new tour, into
22:           $L_v^1$  as a new candidate
23:          Calculate the delivery completion time  $c_i^{lv}$  of order  $i$  at candidate  $l$ 
24:          Calculate  $f_i(c_i^{lv})$  by following Eq. (1)
25:        End For
26:        Insert the pharmaceutical order  $i$  into position  $l$  of vehicle  $v$  with the lowest  $f_i(c_i^{lv})$ 
27:        Update the temperature and capacity of the inserted vehicle  $v$ 
28:        Remove pharmaceutical order  $i$  in the pharmaceutical order list  $L$ 
29:         $i \leftarrow i + 1$ 
30:      End If
31:    End While
32:  End

```

---

## References

- [1] A. Ashok, M. Brison, Y. LeTallec, Improving cold chain systems: challenges and solutions, *Vaccine* 35 (2017) 2217–2223.
- [2] A.K. Sinha, A.R. Verma, A. Chandrakar, S.P. Khes, P.S. Panda, S. Dixit, Evaluation of cold chain and logistics management practice in Durg district of Chhattisgarh: pointer from Central India, *Int. J. Community Med. Public Heal.* 4 (2017) 390.
- [3] V. Salin, R.M. Nayga, A cold chain network for food exports to developing countries, *Int. J. Phys. Distrib. Logist. Manag.* 33 (2003) 918–933.
- [4] R. Montanari, Cold chain tracking: a managerial perspective, *Trends Food Sci. Technol.* 19 (2008) 425–431.
- [5] J.-P. Rodrigue, T. Notteboom, The cold chain and its logistics, *Geogr. Transp. Syst.* (2014) 416.
- [6] R.H. Bishara, Cold chain management - an essential component of the global pharmaceutical supply chain, *Am. Pharm. Rev.* 9 (2006) 105–109.
- [7] M.S. Kapadia, R. Uzsoy, B. Starly, D.P. Warsing Jr, A genetic algorithm for order acceptance and scheduling in additive manufacturing, *Int. J. Prod. Res.* (2021) 1–18.
- [8] İ. Tarhan, C. Oğuz, Generalized order acceptance and scheduling problem with batch delivery: Models and metaheuristics, *Comput. Oper. Res.* 134 (2021).
- [9] A. Noroozi, M. Mahdavi Mazdeh, K. Noghondarian, M. Rasti-Barzoki, M. Heydari, Evolutionary computation algorithms to coordinating order acceptance and batch delivery for an integrated supply chain scheduling, Springer International Publishing, 2018.
- [10] S.A. Slotnick, Order acceptance and scheduling: a taxonomy and review, *Eur. J. Oper. Res.* 212 (2011) 1–11.
- [11] M. Khalilizadeh, M. Esmailpour, B. Naderi, The production-distribution problem with order acceptance and package delivery: models and algorithm, *Manuf. Rev.* 3 (2016).
- [12] L. Wang, Z. Zhang, Y. Yin, Order acceptance and scheduling problem with outsourcing in seru production system considering lot-splitting, *Eur. J. Ind. Eng.* 16 (2022) 91–116.
- [13] M. Reisi-Nafchi, G. Moslehi, Two-agent order acceptance and scheduling to maximise total revenue, *Eur. J. Ind. Eng.* 9 (2015) 664–691.
- [14] G.B. Dantzig, J.H. Ramser, The truck dispatching problem, *Manag. Sci.* 6 (1959) 80–91.
- [15] L. Bodin, B. Golden, Classification in vehicle routing and scheduling, *Networks* 11 (1981) 97–108.
- [16] B. Bouziyane, B. Dkhissi, M. Cherkaoui, Multiobjective optimization in delivering pharmaceutical products with disrupted vehicle routing problem, *Int. J. Ind. Eng. Comput.* 11 (2020) 299–316.
- [17] A. Lubis, H. Mawengkang, A capacitated heterogeneous vehicle routing problem for pharmaceutical products delivery, *Syst. Rev. Pharm.* 11 (2020) 738–741.
- [18] R. Kramer, J.F. Cordeau, M. Iori, Rich vehicle routing with auxiliary depots and anticipated deliveries: an application to pharmaceutical distribution, *Transp. Res. Part E Logist. Transp. Rev.* 129 (2019) 162–174.
- [19] R. Liu, X. Xie, T. Garaix, Hybridization of tabu search with feasible and infeasible local searches for periodic home health care logistics, *Omega (U. Kingd.)* 47 (2014) 17–32.
- [20] A. Ceselli, G. Righini, E. Tresoldi, Combined location and routing problems for drug distribution, *Discret. Appl. Math.* 165 (2014) 130–145.
- [21] H.M. Repollo, J.F. Marchesi, O.S.S. Júnior, R.R.R. Bezerra, Cargo theft weighted vehicle routing problem: modeling and application to the pharmaceutical distribution sector, *Soft Comput.* 23 (2019) 5865–5882.
- [22] K. Sujaree, N. Samattappong, A hybrid chemical based metaheuristic approach for a vaccine cold chain network, *Oper. Supply Chain Manag.* 14 (2021) 351–359.
- [23] J.W. Escobar, J.L.R. Duque, R. García-Cáceres, A granular tabu search for the refrigerated vehicle routing problem with homogeneous fleet, *Int. J. Ind. Eng. Comput.* 13 (2022) 135–150.
- [24] A.R. Komijan, D. Delavari, Vehicle routing and scheduling problem for a multi-period, multi-perishable product system with time window: a case study, *Int. J. Prod. Manag. Eng.* 5 (2017) 45–53.
- [25] L. Chen, Y. Liu, A. Langevin, A multi-compartment vehicle routing problem in cold-chain distribution, *Comput. Oper. Res.* 111 (2019) 58–66.
- [26] Z. Zhao, X. Li, X. Zhou, Optimization of transportation routing problem for fresh food in time-varying road network: considering both food safety reliability and temperature control, *PLoS One* 15 (2020) 1–19.
- [27] X. Qiang, M.Y. Appiah, K. Boateng, F.V.W. Appiah, Route optimization cold chain logistic distribution using greedy search method, *Opsearch* 57 (2020) 1115–1130.
- [28] C. Qi, L. Hu, Optimization of vehicle routing problem for emergency cold chain logistics based on minimum loss, *Phys. Commun.* 40 (2020) 101085.
- [29] Z.J. Ma, Y. Wu, Y. Dai, A combined order selection and time-dependent vehicle routing problem with time widows for perishable product delivery, *Comput. Ind. Eng.* 114 (2017) 101–113.
- [30] M.X. Zhang, J.Y. Wu, X. Wu, Y.J. Zheng, Hybrid evolutionary optimization for takeaway order selection and delivery path planning utilizing habit data, *Complex Intell. Syst.* 8 (2022) 4425–4440.
- [31] B. Yu, W. Shan, J.B. Sheu, A. Diabat, Branch-and-price for a combined order selection and distribution problem in online community group-buying of perishable products, *Transp. Res. Part B Methodol.* 158 (2022) 341–373.
- [32] X. Sun, E.A. Andoh, H. Yu, A simulation-based analysis for effective distribution of COVID-19 vaccines: a case study in Norway, *Transp. Res. Interdiscip. Perspect.* 11 (2021) 100453.
- [33] M. Awad, M. Ndiaye, A. Osman, Vehicle routing in cold food supply chain logistics: a literature review, *Int. J. Logist. Manag.* 32 (2020) 592–617.
- [34] A. Noroozi, M.M. Mazdeh, M. Heydari, M. Rasti-Barzoki, Coordinating order acceptance and integrated production-distribution scheduling with batch delivery considering Third Party Logistics distribution, *J. Manuf. Syst.* 46 (2018) 29–45.
- [35] J.H. Holland, Adaptation in natural and artificial systems: an introductory analysis with applications to biology, control, and artificial intelligence, MIT press, 1992.
- [36] S. Karakatić, V. Podgorelec, A survey of genetic algorithms for solving multi depot vehicle routing problem, *Appl. Soft Comput.* 1. 27 (2015) 519–532.
- [37] T. Vidal, T.G. Crainic, M. Gendreau, N. Lahrichi, W. Rei, A hybrid genetic algorithm for multidepot and periodic vehicle routing problems, *Oper. Res.* 60 (2012) 611–624.
- [38] A. Baniamerian, M. Bashiri, F. Zabihi, Two phase genetic algorithm for vehicle routing and scheduling problem with cross-docking and time windows considering customer satisfaction, *J. Ind. Eng. Int.* 14 (2018) 15–30.
- [39] J. Berger, M. Barkaoui, A hybrid genetic algorithm for the capacitated vehicle routing problem. in: *Genet. Evol. Comput. Conf.*, Springer, 2003, pp. 646–656.
- [40] K. Ghoseiri, S.F. Ghannadpour, Multi-objective vehicle routing problem with time windows using goal programming and genetic algorithm, *Appl. Soft Comput.* 10 (2010) 1096–1107.
- [41] K. Ghoseiri, S.F. Ghannadpour, A hybrid genetic algorithm for multi-depot homogenous locomotive assignment with time windows, *Appl. Soft Comput.* 10 (2010) 53–65.
- [42] J. Berger, M. Barkaoui, A parallel hybrid genetic algorithm for the vehicle routing problem with time windows, *Comput. Oper. Res.* 31 (2004) 2037–2053.
- [43] M.A. Islam, Y. Gajpal, T.Y. ElMekkawy, Hybrid particle swarm optimization algorithm for solving the clustered vehicle routing problem, *Appl. Soft Comput.* 110 (2021) 107655.
- [44] R.J. Kuo, M. Fernanda Luthfiansyah, N. Aini Masruroh, F. Eva Zulvia, Application of improved multi-objective particle swarm optimization algorithm to solve disruption for the two-stage vehicle routing problem with time windows, *Expert Syst. Appl.* 225 (2023) 120009.
- [45] İ. İLHAN, An improved simulated annealing algorithm with crossover operator for capacitated vehicle routing problem, *Swarm Evol. Comput.* 64 (2021) 100911.
- [46] L. Wei, Z. Zhang, D. Zhang, S.C.H. Leung, A simulated annealing algorithm for the capacitated vehicle routing problem with two-dimensional loading constraints, *Eur. J. Oper. Res.* 265 (2018) 843–859.
- [47] F. Marini, B. Walczak, Particle swarm optimization (PSO). A tutorial, *Chemom. Intell. Lab. Syst.* 149 (2015) 153–165.
- [48] M. Birattari, M. Dorigo, How to assess and report the performance of a stochastic algorithm on a benchmark problem: mean or best result on a number of runs? *Optim. Lett.* 1 (2007) 309–311.
- [49] N. Ivković, R. Kudelić, M. Črepinsk, Probability and certainty in the performance of evolutionary and swarm optimization algorithms, *Mathematics* 10 (2022).
- [50] Y. Che, K. Hu, Z. Zhang, A. Lim, Machine scheduling with orientation selection and two-dimensional packing for additive manufacturing, *Comput. Oper. Res.* 130 (2021) 105245.