# Optimal vehicle route schedules in picking up and delivering cargo containers considering time windows in logistics distribution networks: A case study

**Thi Diem Chau Le[1,2]** iD **, Duy Duc Nguyen[3]** iD **, Judit Oláh[1, 4*]** iD **, Miklós Pakurár[1]** iD

[1] Faculty of Economics and Business, Institute of Applied Informatics and Logistics, University of Debrecen, 4028 Debrecen, Hungary,
[2] Ho Chi Minh City University of Technology-VNU, Ho Chi Minh City, Vietnam,
[3] School of Management Technology, Sirindhorn International Institute of Technology, Thammasat University, Pathum Thani 12120, Thailand,
[4] TRADE Research Entity, North-West University, Vanderbijlpark 1900, South Africa,
*Corresponding author e-mail: olah.judit@econ.unideb.hu

**Abstract**

This study describes a pickup and delivery vehicle routing problem, considering time windows in reality. The problem of tractor truck routes is formulated by a mixed integer programming model. Besides this, three algorithms - a guided local search, a tabu search, and simulated annealing - are proposed as solutions. The aims of our study are to optimize the number of internal tractor trucks used, and create optimal routes in order to minimize total logistics costs, including the fixed and variable costs of an internal vehicle group and the renting cost of external vehicles. Besides, our study also evaluates both the quality of solutions and the time to find optimal solutions to select the best suitable algorithm for the real problem mentioned above. A novel mathematical model is formulated by OR tools for Python. Compared to the current solution, our results reduced total costs by 18%, increased the proportion of orders completed by internal vehicles (84%), and the proportion of orders delivered on time (100%). Our study provides a mathematical model with time constraints and large job volumes for a complex distribution network in reality. The proposed mathematical model provides effective solutions for making decisions at logistics companies. Furthermore, our study emphasizes that simulated annealing is a more suitable algorithm than the two others for this vehicle routing problem.

## 1. Introduction

In recent years, transportation costs are becoming a main concern of logistics companies, especially companies in the developing logistics industry, such as in Vietnam. The fact remains that transportation costs take up over 50% of total logistics costs. Besides, it is one of the vital factors in determining an effective logistics industry (Jean-Paul Rodrigue, 2020). Hence, it is necessary for logistics companies to find effective solutions for improving transportation cost.

The Multiple Travelling Salesman Problem (MTSP) is the classical problem for optimizing transportation costs (Xu et al., 2018). An MTSP network consists of a salesman and cities (Masmoudi & Mellouli, 2014). A salesman may travel to many different cities in the network. Therefore, the MTSP requires that each city must be visited by a salesman. The MTSP

aims to find optimal routes for a salesman and an optimal total cost (An & Li, 2011). The development of the MTSP is a Vehicle Routing Problem (VRP) when considering the addition of capacity constraints (Toth & Vigo, 2014). The capacity constraints are described by the capacity limitation of vehicles in transferring products to customers (Shuai et al., 2019). In particular, a variation of the VRP is the Pickup and Delivery Vehicle Routing Problem (PDVRP). The PDVRP consists of the activities of vehicles in loading cargo at one location and delivering cargo to another location. The PDVRP includes two types of problems: paired and unpaired. In paired PDVRP, the pickup and delivery locations and the demand for them are predefined. By contrast, the unpaired PDVRP indicates that the number of locations for receiving cargo and picking up cargo is unknown beforehand (Dongyang et al., 2020; Chen & Liu, 2014; Bent & Van Hentenryck, 2006). On the other hand,

when a specific time window is required by customers at each destination, the vehicle routing problem becomes a Vehicle Routing Problem with Time Window (VRPTW). A VRPTW is defined by determining the optimal transportation routes to minimize total costs. These routes are formulated based on the movement of vehicles from the depot or starting point to customer groups, and from customers back to the initial depot, and the end of the vehicles' journeys (Gan et al., 2012). The time window constraint is considered to take two forms: soft and hard time windows. A soft time window describes the violation in relation to the time of a vehicle, and this violation will be a charged cost. For example, the waiting time for picking up a cargo container at ports is limited to 30 minutes. If the vehicle arrives 30 minutes early, they must pay penalty costs. A hard time window is a limitation of service time at each location. This means that each location only provides a service for a specified amount of time. For instance, customer A always receives cargo from 9:00 a.m. to 11:00 a.m. every day. Hence, vehicles can arrive before 9:00 a.m. or after 11:00 a.m. but they have to wait until the time window starts (Aggarwal and Kumar, 2019)

In this study, a paired pickup and delivery vehicle routing problem considering hard time windows is presented. This network is two echelons of a distribution network including three members: a logistics company, a group of 35 depots/ports, and a group of 106 customers. Cargo is picked up at depots/ports or received at depots/ports and moved to customers. In some cases, cargoes are picked up at customers and delivered to other customers. All the above activities are implemented by 38 tractor trucks from a logistics company in Vietnam. All tractor trucks must arrive at destinations in the time windows that are required by customers. Finally, all tractor trucks must return to their company when finishing their trip. This pickup and delivery cargo problem is described by a mixed integer programming (MIP) model. The novel mathematical model is solved by three algorithms: Guided Local Search (GLS), Tabu Search (TS), and Simulated Annealing (SA) which are implemented in OR- Tools for Python. Our study's objective is to minimize total logistics costs by determining optimal routes and the optimal number of vehicles used. Besides, another aim is to find the best suitable algorithm among the three algorithms under consideration. A vast set of real input data is collected on three different days and used for computing the results of the mathematical model. They include data regarding costs (fixed costs, operational costs, and hiring costs), time, and moving distance between nodes. The result of the model is an optimal solution of costs and routes. Besides, the result shows that the solution of the simulated annealing algorithm is better than the others when considering both computed results and running time.

The structure of this paper is organized as follows. A literature review in section 2 describes the main gaps between our study and previous studies. In section 3, a mathematical model is developed, including assumptions, an objective, and constraints. Besides, an algorithm flowchart to solve the mathematical model is described. Results and conclusion are presented in sections 4 and 5.

## 2. Literature review

The vehicle routing problem is a classic problem and a topic of main concern in logistics and transportation fields. The concept of the VRP focuses on planning routes for vehicles from a central depot to a group of customers (Dantzig and Ramser 1959). The aims of the VRP are mainly to optimize routes and the number of vehicles used, as well as the transportation time. Its objective is to minimize total logistics costs.

The VRP is interesting and important in both practical and academic research. Thus, research in the VRP tends to become more complicated in the fields of distribution and computation. According to Cao, W., and Yang, W. (2017), the VRP basically consists of two variations which are a pure vehicle routing problem and a pickup and delivery vehicle routing problem (Table 1 summarizes some previous studies in finding solutions for the two variations of the VRP). First of all, the pure vehicle routing problem is described as a pure cargo delivery problem. This means that cargo is distributed by vehicles from a central depot to customers. Almost all previous studies focus on developing mathematical models and algorithms for the pure vehicle routing problem. Mixed-integer programming is used popularly in formulating a model of the VRP. For example, Aggarwal, D., and Kumar, V. (2019) and Tasar, B., et al. (2019) used an MIP to model their problems with a large number of nodes. For more complexity in the constraints and variables of the VRP, MIP also develops into Mixed-Integer Linear Programming (MILP). For instance, Londoño, J.C., et al. (2020) considered variables and constraints related to the limitation of subtours in a network configuration from a depot to 51 customers. Thus, they proposed mixed-integer linear programming for describing their problem. In addition to forming mathematical models, previous studies have applied or developed an efficient algorithm to search for solutions quickly. To find optimal solutions, algorithms include the following: an estimation of distribution, heuristic, artificial bee colony, local search, response surface method, biased random-key genetic, exact branch-price-and-cut algorithm, tabu search, and simulated annealing (Qi and Hu, 2020; Kantawong and Pravesjit, 2020; Londoño et al., 2020; Aggarwal and Kumar, 2019; Pérez-Rodríguez and Hernández-Aguirre, 2019; Tasar et al., 2019; Setamanit, 2019Zhu and Hu, 2019; Ruiz et al., 2019; Zhang et al., 2017; Huang et al., 2017; Afifi et al., 2016; Spliet and Desaulniers, 2015).

The pickup and delivery vehicle routing problem (PDVRP) concerns activities of loading cargo at one point and transferring it to other points. The PDVRP consists of two variants: paired and unpaired. The unpaired PDVRP indicates that the locations for receiving and picking up cargo are unknown beforehand. By contrast, the paired PDVRP shows that the pickup and delivery locations and demand between pickup point and delivery point are predefined (Dongyang et al., 2020). Similarly, a pure vehicle routing problem, simulated annealing, and genetic algorithm are used to find optimal solutions (Gunawan et al., 2020; Straka et al., 2018; Mohammed et al., 2017; Birim, 2016; Vincent, et al., 2016).

This study develops a mixed-integer programming model for a paired pickup and delivery vehicle routing problem based

on the mathematical model of previous studies. The significant contribution of our research can be summarised as follows: firstly, the characteristics of the paired pickup and delivery vehicle routing problem are different when compared to studies by Gunawan, A., et al. (2020), Birim, Ş. (2016), and Vincent, F. Y., et al. (2016) and our study. All three of the previous studies considered the activity of picking up cargo and the activity of delivering cargo separately. This indicates that they divided their network into two groups: 1- vehicles move from a depot to suppliers to pick up cargo and back to the depot, and 2- vehicles deliver cargo from a depot to customers and back to the depot when finishing their journey. Besides, the movement of vehicles is one-way, which indicates that vehicles move from location 1 to locations 2, 3, 4, or the next locations. In our study, by contrast, vehicles may have 2-way movement and pick up and delivery of cargo are combined. This means that vehicles depart from the logistics company to the first locations (depots/ports or customers) to pick cargo containers and deliver cargo containers to second locations. Then our vehicles may receive a new container at the

same location and transfer it to the initial location, or arrive at third locations from second locations to pick up or deliver the next cargo containers. This process is repeated until customer demand is satisfied, and vehicles return to the logistics company. Secondly, in comparison with Gunawan, A., et al. (2020), Mohammed, M. A., et al. (2017), Birim, Ş. (2016), and Vincent, F. Y., et al. (2016) this study considers a factor of hard time windows. This means that each activity of picking up and delivering cargo containers must meet the requirement of the earliest and latest service time at each location. Finally, our research considers a real data and complicated network which is collected from a logistics company in Vietnam as a case study. The network consists of a logistics company as an original node and 243 locations, while other previous studies take into account consider only 10, 30, and 50, and up to 200 locations. Furthermore, this work adopts three algorithms simultaneously - guided local search, tabu search, and simulated annealing - to find the solution and conduct a comparison among algorithms based on the objective function and solving time.

**Table 1.** Characteristics of vehicle routing problem

| Authors | Objective Function: (Minimize) | | | | Pure VRP | PDVRP | | Network conf. (P+D) | Hard time windows | Mathematical model & Algorithm | Data | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | Total costs | Time | # of vehicle | Optimal route | | Paired | Un-paired | | | | Exp. | Reality |
| Spliet, R., and Desaulniers, G. (2015) | ✓ | | | | ✓ | | | 60L | ✓ | MIP, Exact branch-price-and-cut algorithm | ✓ | |
| Birim, Ş. (2016) | ✓ | | | | | ✓ | | 10L | | MILP, SA | ✓ | |
| Vincent, F. Y., et al. (2016) | ✓ | | | | | ✓ | | 50L | | MILP, SA | ✓ | |
| Afifi, S., et al. (2016) | | ✓ | | | ✓ | | | 30L | ✓ | MIP, SA | ✓ | |
| Huang, M., et al. (2017) | ✓ | | | | ✓ | | | 15L | ✓ | MIP, C-W algorithm | | ✓ |
| Mohammed, M. A., et al. (2017) | | | | ✓ | ✓ | | | 2 routes | | Genetic algorithm | | ✓ |
| Zhang, D., et al. (2017) | ✓ | | | | ✓ | | | 27L | ✓ | TS & the artificial bee colony | ✓ | |
| Aggarwal, D., and Kumar, V. (2019) | ✓ | | ✓ | | ✓ | | | 60L | ✓ | MIP | ✓ | |
| Pérez-Rodríguez, R., and Hernández-Aguirre, A. (2019) | | | | ✓ | ✓ | | | 6L | ✓ | An estimation of distribution | | ✓ |
| Tasar, B., et al. (2019) | ✓ | | ✓ | | ✓ | | | 100L | | MIP, Heuristic | ✓ | |
| Zhu, L., and Hu, D. (2019) | ✓ | | | | ✓ | | | 200L | | MILP, Response Surface Method | ✓ | |
| Ruiz, E., et al. (2019) | ✓ | | | | ✓ | | | 14L | | MIP, Biased random-key genetic | | ✓ |
| Dongyang, X., et al. (2020) | ✓ | | | ✓ | | | ✓ | 10L | | MIP, variable neighborhood search | ✓ | |
| Gunawan, A., et al. (2020) | ✓ | | | | | ✓ | | 30L | | MIP | ✓ | |
| Qi, C., and Hu, L. (2020) | ✓ | | ✓ | | ✓ | | | 13L | | MIP, Heuristic | ✓ | |
| Kantawong, K., and Pravesjit, S. (2020) | ✓ | | | ✓ | ✓ | | | 100L | ✓ | MIP, Artificial bee colony | ✓ | |
| Londoño, J. C., et al. (2020) | ✓ | | | ✓ | ✓ | | | 51L | | MILP, local search | ✓ | |
| This study | ✓ | | ✓ | ✓ | | ✓ | | 243L | ✓ | MIP, TS, SA, GLS | | ✓ |

Note: P+D-total nodes of both picking up and delivery cargo, L-Location, VRP-Vehicle routing problem, PDVRP-Pickup and delivery vehicle routing problem, Exp.-Experience

# 3. Mathematical Model and Algorithm Flowchart

## 3.1. Problem description

A pickup and delivery vehicle routing problem in logistics management is considered in this study. The transportation network consists of two echelons that involve a logistics company, 35 ports/depots, and 106 customers (Figure 1). The logistics company has 38 tractor trucks, and each tractor truck may carry a container of 40 feet (ft) or 20 feet (ft). The vehicles' process allows tractor trucks to depart from the original company to depots/ports or customers to pick up cargo containers, and deliver these cargo containers to other customers or depots/ports. Finally, the tractor trucks return to the logistics company when completing their trip. This operational process is quite complex, as is illustrated in an example showing by the shipping of tractor truck A. Tractor truck A starts at the logistics company and comes to pick up a container at customer KNQ2. Then, tractor truck A transfers the container from customer KNQ2 to customer Phuc Long 2. After that, it receives a new container at customer Phuc Long 2 and delivers it back to customer KNQ2. Finally, tractor truck A picks up another container at customer KNQ2, ships to it to Cat Lai port, and back to the logistics company to finish its journey on the same day.
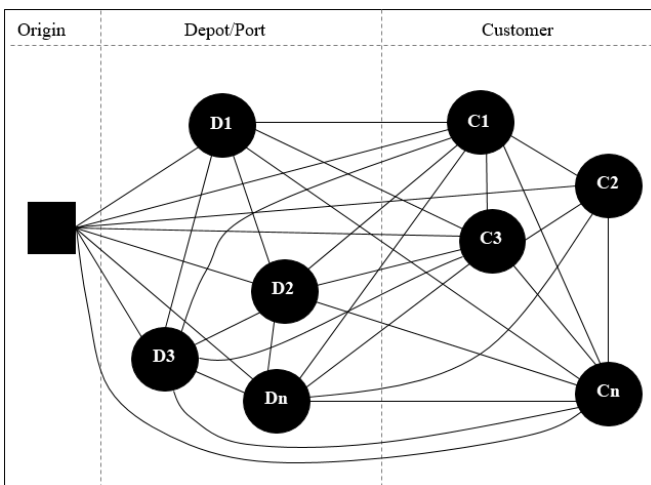


**Fig. 1.** Distribution network

Due to the complex operational process, the 2-way movement of vehicles, and the large number of orders, tractor truck's pick up location and delivery containers are assigned by the start point and end point of an order instead of a journey of a vehicle or the location of customer or depot/port. For instance, orders 1, 2, and 3 are delivered from BOX-PAX company to customer Song Than, and each order is transferred by a vehicle. The order of the locations will be assigned as follows: order 1 is moved from location 1 to location 2, order 2 is moved from location 3 to location 4, and order 3 is moved from location 5 to location 6. Thus, from the logistics company, these tractor trucks may pick up cargo containers at, and transfer them to, 234 locations in the network per day.

All input data for the mathematical model are collected from a logistics company in Vietnam. They include information about customers and depots/ports, cost, orders, and time. First of all, customer and depot/port locations are collected to determine the moving distance of vehicles between two locations. The latitude and longitude of each location are determined by Google Maps. Then, the distance between two locations is calculated by the haversine formula, proposed by James Andrew (Van Brummelen, 2012). The information about costs, represented by two types of cost (fixed costs and variable costs), are collected. Fixed costs include the salary of the workers, periodical maintenance cost, and road tax. Variable costs consist of fuel costs and variable maintenance costs which depend on the vehicle traveling distance. Next, the data related to orders and time are two important factors which are collected. The order information shows the number of orders and customer demand at each node. Besides this, the information about orders also shows the number of orders implemented by internal and external vehicles. Finally, the starting time and the end point, and the transportation time between two nodes also are shown in input data sets. In particular, the time information describes the time window for picking up and delivering containers at each node.

## 3.2. Assumption

To solve our pickup and delivery vehicle routing problem, a mathematical model is developed. The development of such as a mathematical model is based on certain assumptions, as follows:

- Only one type of tractor truck is used in this problem
- Only one type of container (40 feet) is used
- The number of customers, customer demand, distance, the time window, and cost units are deterministic.
- The costs include the salary of workers, periodical maintenance costs, road tax, fuel costs, variable maintenance costs, and the costs of hiring tractor trucks.
- Customer demand changes day-by-day
- Working time is from 8:00 a.m. to 20:00 p.m. every day, and 26 business days per month
- Each customer may be serviced by at least one tractor truck
- Each tractor truck always starts and ends at the logistics company

## 3.3. Mathematical model

A mathematical model consists of input data, variables, an objective function, and constraints. Input data are described by sets and parameters. These provide the information about vehicle fleet such as their jobs, time, and costs. Variables indicate that jobs are performed by internal vehicles and/or external vehicles. The objective function and constraints are presented by functions (1) – (11).

*Sets:*

$k$ : Tractor truck

$i, j$ : Job of tractor truck $k$

$m$ : Work order of tractor truck $k$

$K = \{1, 2, 3, ..35\}$ : Set of tractor trucks

$N = \{1, 2, 3, ..., N\}$ : Set of job $i$

$M = \{1, 2, 3, .., M\}$ : The maximum number of jobs for tractor truck $k$ to complete jobs on one working day.

*Parameters:*

$F$: The fixed costs

$C_i$ : The variable costs of a tractor truck when implementing job $i$

$S_i$ : The costs of hiring a tractor truck when completing job $i$

$T_{m,k,0}$ : The starting time of the $m^{th}$ job of tractor truck $k$

$T_{m,k,1}$ : The arrival time at the 1st point of the $m^{th}$ job of tractor truck $k$

$T_{m,k,2}$ : The arrival time at the 2nd point of $m^{th}$ job of tractor truck $k$

$D_{i,j}$ : The moving time between the ends of job $i$ and the starting point of job $j$

$D_i$ : The moving time of job $i$ from the 1st point to the 2nd point

$H_{i,1}$ : The waiting time of job $i$ at the 1st point

$H_{i,2}$ : The waiting time of job $i$ at the 2nd point

$R_{i,0}$ : The lower bound of the time window for job $i$ at the 1st point

$R_{i,1}$ : The upper bound of the time window for job $i$ at the 1st point

$R_{i,2}$ : The lower bound of the time window for job $i$ at the 2nd point

$R_{i,3}$ : The upper bound of the time window for job $i$ at the 2nd point

$A_{k,0}$ : The time to start work by tractor truck $k$

$A_{k,1}$ : The time to end work by tractor truck $k$

*Variables:*

$x_{i,k,m} = 1$, if job $i$ is allocated as the $m^{th}$ work order of tractor truck $k$. Otherwise, $x_{i,k,m} = 0$

$x_{i,0} = 1$, if job $i$ is implemented by an external company. Otherwise, $x_{i,0} = 0$

*Objective function and constraints:*

$$\text{Minimize } Z = F + \sum_{i \in N} \sum_{k \in K} \sum_{m \in M} C_i x_{i,k,m} + \sum_{i \in N} S_i x_{i,0} \quad (1)$$

$$\sum_{k \in K} \sum_{m \in M} x_{i,k,m} + x_{i,0} = 1; \forall i \in N \quad (2)$$

$$\sum_{i \in N} x_{i,k,m+1} \le \sum_{i \in N} x_{i,k,m} \le 1; \forall k \in K, m \in \{1, 2, 3, ..., M-1\} \quad (3)$$

$$T_{m+1,k,0} \ge T_{m,k,2}; \forall k \in K, m \in \{1, 2, 3, ..., M-1\} \quad (4)$$

$$T_{m,k,1} = T_{m,k,0} + (D_{j,i} H_{i,1}) \sum_{i \in N} \sum_{j \in N} x_{i,k,m} x_{j,k,m-1}; i \ne j, \forall k \in K, m \in M \quad (5)$$

$$T_{m,k,2} = T_{m,k,1} + D_i H_{i,2} \sum_{i \in N} x_{i,k,m}; \forall k \in K, \forall m \in M \quad (6)$$

$$A_{k,0} \le T_{m,k,0} \le A_{k,1} - (T_{m,k,2} - T_{m,k,0}); \forall k \in K, \forall m \in M \quad (7)$$

$$R_{i,0} \le (T_{m,k,1} - H_{i,1}) \sum_{k \in K} \sum_{m \in M} x_{i,k,m} + R_{i,0} x_{i,0} \le R_{i,1}; \forall i \in N \quad (8)$$

$$R_{i,2} \le (T_{m,k,2} - H_{i,2}) \sum_{k \in K} \sum_{m \in M} x_{i,k,m} + R_{i,2} x_{i,0} \le R_{i,3}; \forall i \in N \quad (9)$$

$$0 \le x_{i,k,m} \le 1; x_{i,k,m} \in \{0,1\}, \forall i \in N, \forall k \in K, \forall m \in M \quad (10)$$

$$0 \le x_{i,0} \le 1; x_{i,0} \in \{0,1\}, \forall i \in N \quad (11)$$

Objective function (1): minimize total costs including the fixed and operational costs of internal vehicles and the hiring costs of external vehicles on one day. Constraint (2) describes that each job is only implemented by one tractor truck. The work order of tractor truck $k$ is shown by constraint (3). Jobs have the order $m+1$, which must be followed by the $m^{th}$ job of vehicle $k$. Constraint (4) provides that the starting time of job $m+1$ of vehicle $k$ must be right after or after the time of finishing job $m$ of vehicle $k$. Constraints (5) and (6) show that the ending time of vehicle $k$ for job $i$ is the total of the starting time of job $i$, the moving time, and the waiting time. Constraint (7) requires that all jobs of truck $k$ must be completed on time. Constraints (8) and (9) are limitations of the time window at each destination. The arrival time of each tractor truck at the 1st point must belong to time window of the 1st point. Similarly, at the 2nd point, the arrival time of the vehicle completely belongs to the time window of the 2nd point. Finally, a description of variables appears in constraints (10) and (11). Both $x_{i,k,m}$ and $x_{i,0}$ are binary, which means that they have only two values (0 or 1).

### 3.4. Algorithm flowchart

The tabu search, simulated annealing, and guided local search algorithms are meta-heuristic, and help the search process get rid of local minima and find better solutions. Besides, all three algorithms are evaluated as suitable methods for the pickup and delivery vehicle routing problem (Voudouris and Tsang, 2003).

*Tabu search algorithm*

The tabu search is a heuristic procedure used to find the best solution in its neighbourhood. It is a fundamental foundation for guiding other methods to avoid local optimality (Archetti

et al., 2006). The process starts by generating an initial solution, and then neighbourhoods are identified to begin the activities required for a neighbourhood search. Finally, stopping criteria are checked to finish the searching process (Barbarosoglu and Ozgur, 1999). Although tabu is a simple algorithm and is easy to implement, a tabu search is affected significantly by an initial solution (Zhang et al., 2017).

Step 1: initial solution
- Set $k = 1$ and select $t_1$
- Select $S_1$ based on the experience algorithm
- Set $S_0 = S_1$

Step 2: general neighbourhood
- Select $S_c$ from the neighbourhood of $S_k$
- If moving $S_k$ to $S_c$ is forbidden and is kept as a mutation in the Tabu-list, $S_{k+1} = S_k$
- If $G(S_c) < G(S_0)$, $S_0 = S_c$ and go to step 3

Step 3:
- Set $k = k + 1$
- If $k = N$, STOP; otherwise, back to step 2

*Simulated annealing*

Unlike tabu search, simulated annealing applies a probabilistic approach to obtain the same result (Connor and Shea, 2000). SA is also a meta-heuristics procedures, based on the annealing process of metal and the search for a minimum in a more general system (Birim 2016; Busetti, 2003). In an SA algorithm, a stochastic approach is used to search for neighbourhood solutions. Neighbourhood solution X′ is replaced by current solution X when its result generates a better objective than the current solution (Vincent et al., 2016; Kuo, 2010). SA is flexible and is easily "tuned" so it is suitable for providing good quality solutions to difficult problems (Wei et al., 2018; Afifi et al., 2016; Wang et al., 2015). The three steps of simulated annealing are implemented as follows:

Step 1:
- Set $k = 1$ and select $t_1$
- Select $S_1$ based on the experience algorithm
- Set $S_0 = S_1$

Step 2:
- Select $S_c$ from the neighbourhood of $S_k$, $S_c \in N(S_k)$
- If $G(S_0) < G(S_c) < G(S_k)$, $S_{k+1} = S_c$ and go to step 3
- If $G(S_c) < G(S_0)$, $S_0 = S_{k+1} = S_c$ and go to step 3
- If $G(S_c) > G(S_k)$, generate $U_k$ randomly and $U_k$ followed by Uniform distribution $(0,1)$
  - If $U_k \leq P(S_k, S_c) = \exp(\frac{G(S_k) - G(S_c)}{t_k})$,
    $S_{k+1} = S_c$
  - Else: $S_{k+1} = S_k$ and go to step 3

Step 3:
- Select $t_{k+1} \leq t_k$
- Set $k = k + 1$

- If the stopping criterion is satisfied, finish the process. Otherwise, back to step 2

*Guided local search*

Guided local search is a relatively new meta-heuristic algorithm. GLS works by building up penalties during a search. Penalty factors will be added into the objective function if the search tends to stray too close to previously visited local minima (Kilby et al., 1999). By utilizing this idea, the search escapes local searches and migrates to neighbourhood solutions. In some research studies- GLS has performed as the best meta-heuristic method in solving routing problems (Londoño et al., 2020; Alsheddy, 2011). The process of the guided local search includes:

Step 1: An initial solution is generated by a local search
Step 2: The objective function is augmented by adding penalty factors
Step 3: A apply guided local search is applied with an updated objective function
Step 4: Local minimums for a penalty are found
- Features involving high costs take a priority in terms of being penalized. These features are called $i$. The utility expression of feature $i$ ($util_i$) is defined as follows:

$$util_i(s^*) = I_i(s^*) \times \frac{c_i}{1 + p_i}$$

where $s^*$: a local minimum, $c_i$: cost, $p_i$: penalty value of feature $i$

- If feature $i$ is not in a local minimum solution $I_i(s^*)$, $util_i(s^*) = 0$; otherwise, features with the highest utility in local minimum $s^*$ will be penalized. When a feature is penalized, its penalty value will go up to 1. The penalty rate is adjusted by parameter $\lambda$ which is a single parameter in a guided local search.

$$\lambda = \alpha * g(s^*) / F_{s^*}$$

where $\alpha$: coefficient of $\lambda$, $g$: objective of mathematical model, $s^*$: local minimum, $F_{s^*}$: number of features represented in $s^*$

- If the best solution is found, go to step 5. Otherwise, back to step 3

Step 5: Finish

*General flowchart of algorithms*

Each algorithm is applied in a practical pickup and delivery vehicle routing problem through a formulated mathematical model. A general flowchart of algorithms is formulated by combining the flowchart of a guided local search, a tabu search, and simulated annealing (Figure 2). This flowchart provides a general guideline for finding processes for an optimal solution of each algorithm in a real-life vehicle routing problem.
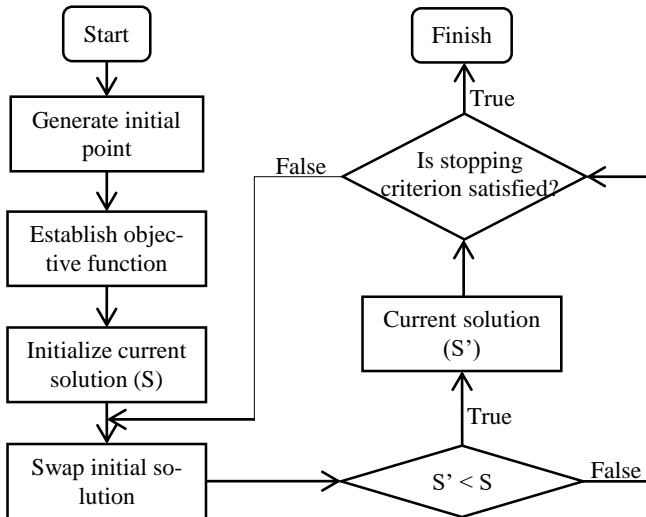
**Fig. 2.** Algorithm flowchart

- Step 1: Generate an initial point
  Enter input data such as order information, distance, time window at each location, and coefficient of costs
- Step 2: Establish objective function
- Step 3: Initialize current solution (S)
  Scheduling a vehicle's routes is implemented by random selecting orders and assigning orders to the routes of each tractor truck. Besides, there is only one order that is assigned to the route of each vehicle. When an order is not assigned to the current route of vehicle, it will be assigned to next route. This process is repeated until there is no route receiving any orders. The rest of the orders are shipped by the external vehicle.
  There are certain conditions for selecting orders:
  - The starting time for the servicing of each selected order is the earliest
  - The starting time for the servicing of each selected order must belong to the time window
  - The time for each vehicle's return to a logistics company from the selected job must belong to a logistics company's operation time
  Stopping criteria:
  - Cannot find a job that satisfies all the above conditions
  - All jobs are served
- Step 4: Swap initial solution based on the steps of each algorithm
- Step 5: Comparison between solutions of each algorithm.
  If swap cost (S') is lower than initial cost (S), the current solution is the neighbour solution (S'). Otherwise, go to step 6
- Step 6: Check criterion to stop solution.
  If a solution satisfies all the criteria, the process will be finished. Otherwise, back to step 4.

## 4. Results

The results of a mathematical model are found by applying three algorithms of a guided local search, a tabu search, and simulated annealing, respectively. The results of solving mathematical model include total transportation costs, the number of tractor trucks used, the proportion of orders delivered on time, the operational capacity of vehicles, and the time for determining the solution. Following this, the results of each algorithm are compared with the Recent Solution/Real result (RS) which is defined by the operators' experience.

### 4.1. Total cost

The total cost result is shown in Figure 3. The result of the experiment shows that the total cost is 324,961,410 (VND-Vietnam currency). The total cost is calculated by adding the fixed and operating costs of internal vehicles and the cost of hiring external vehicles as in objective function (1). When this is solved by three optimal algorithms, the total cost reduces significantly. The guided local search algorithm provides the lowest total cost (250,490,035 VND) among the three algorithms, and a total cost which is 23% lower than the real result.
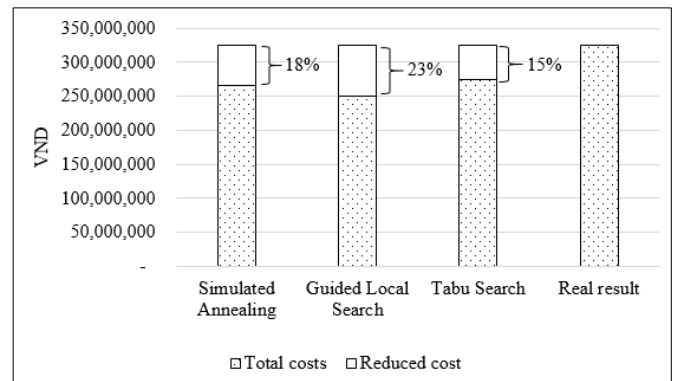


**Fig. 3.** Total transportation cost

### 4.2. Numbers, trips, moving distances, and utilization rates of the vehicle fleet

Figure 4 shows the solution calculated from algorithms related to the necessary number of internal vehicles, the number of trips, the moving distances, and the utilization of the tractor truck fleet to pick up and deliver cargo following customer requirements in three workdays, as follows:

- The necessary number of internal vehicles (Figure 4a): The highest number of internal vehicles is found by a guided local search of 35 internal vehicles. The simulated annealing algorithm uses 34 internal tractor trucks to complete customer orders. Although the result of simulated annealing is lower than that of the guided local search, it is higher than the result of other algorithms. As a result, only 32 internal vehicles are used in the solution of the tabu search algorithm.
- The number of vehicle trips (Figure 4b): The total number of vehicle trips used in practice is 106 trips, including both internal and external trips. All three algorithms pro-

vide detailed trip numbers for internal and external vehicles. However, the analysis is focused on the number of trips made by internal vehicles. The results show that the guided local search uses the highest number of internal trip (90 trips) while the tabu search algorithm finds the lowest number of internal trips (84 trips). Finally, the simulated annealing solution gives a result of 89 internal trips.

- The total distance of trips (Figure 4c): The total distance of trips is calculated by summing the distance of each trip in three workdays of the vehicle fleet. Figure 4c shows that the lowest (14,910 km) total moving distance for the vehicle fleet is that found by the guided local search. By contrast, simulated annealing finds the greatest distance for vehicle fleet movement (15,260 km). Finally, the tabu search solution produces a total moving distance which is

higher than that of simulated annealing, but lower than the result of the guided local search.

- The utilization rate of the vehicle fleet (Figure 4d): Based on the total moving distance and the number of vehicles, the utilization rate of vehicles is computed. In general, all three algorithms give better solutions than the solution occurring in reality for all three business days. Among the three algorithms, the simulated annealing algorithm provides the best optimal solution (45.03% for day 1, 58.12% for day 2, and 54.44% for day 3) and this solution is closest to the logistics company's objective (60%) in using their vehicles. The solution ranking of the guided local search and the tabu search have changed slightly. For days 1 and 3, the guided local search produces a better utilization rate than the tabu search, but on day 2 its rate is worse.
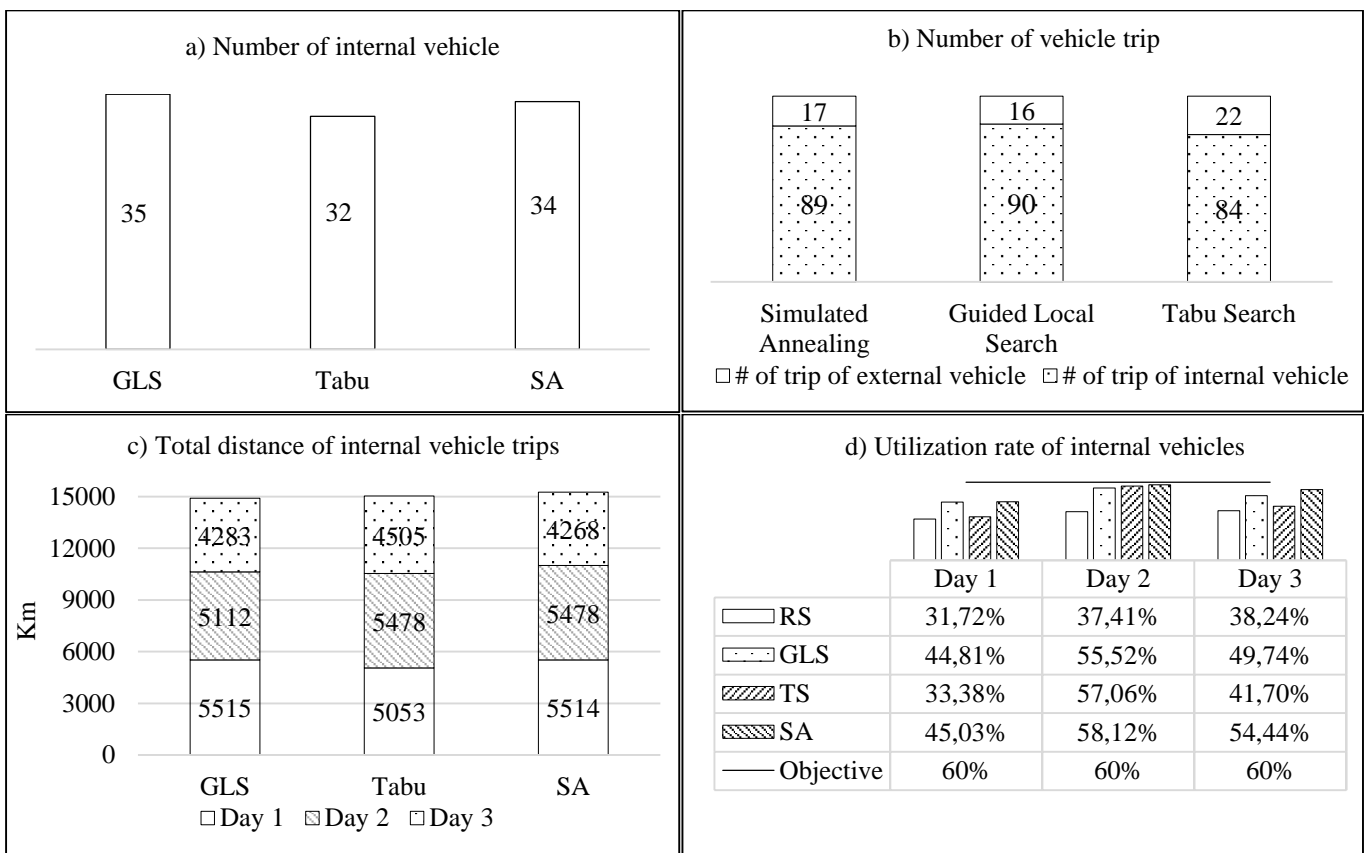


**Fig. 4.** Number, trip, moving distance, and utilization of vehicles

## 4.3 Amount and rate of orders delivered on time

Figure 5 describes the amount and rate of orders delivered on time. The results of all three algorithms show that 100% of orders are picked up and delivered to customers on time. These results are an improvement on the approximately 20% of late deliveries which occur in reality. Similarly, the numbers of orders which are transferred by internal vehicles using three algorithms are higher than occurs in reality. In the three results, the highest number of orders (318 orders) completed by internal tractor trucks was achieved by the guided local

search algorithm. By contrast, the lowest number of orders (292 orders) transferred by internal vehicles, occurred with the tabu search algorithm.
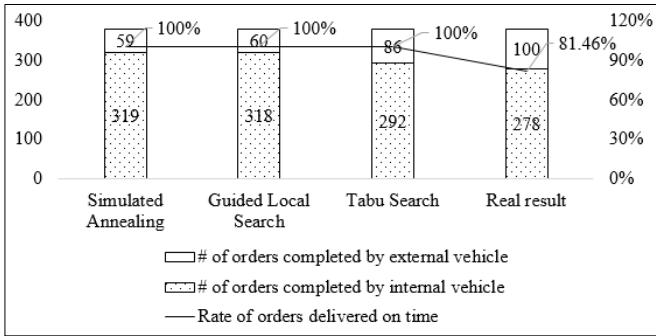
**Fig. 5** The number of orders completed by tractor trucks

## 4.4 Comparison between the solutions of the algorithms

Comparing the results achieved by the algorithms means finding the best suitable solution that should be used in the logistics industry. This comparison is based on six results, i.e. the proportion of costs saved, the proportion of internal vehicles used, the proportion of orders completed by internal vehicles, the proportion of orders delivered on time, the vehicle fleet utilization rate, and the computation speed of finding an optimal solution (Table 2).

Table 2 shows the differences in the results among three of the algorithms, except for the proportion of orders delivered on time. First, the proportion of orders delivered on time is 100% for all three algorithms. Other information is the proportion of saved costs, the proportion of internal vehicles used, and the vehicle fleet utilization rate. The tabu search algorithm cannot generate the optimal solution as others algorithms can. By contrast, both simulated annealing and guided local search calculate the proportion of internal vehicles used at approximately (84%), and their vehicle fleet utilization rates are the closest to the target. However, there is a difference in the proportion of costs saved between the simulated annealing and the guided local search. The simulated annealing algorithm saves 18% of costs compared to the costs saved in reality, but this result is lower than the solution provided by the guided local search (23%). Last but not least, is the computation speed of finding an optimal solution. The guided local search takes 3,960 minutes to find optimal solution for this problem while the simulated annealing and the tabu search both only need 3 minutes to arrive at an optimal solution.

In summary, for a complex vehicle routing problem including 142 nodes and 38 tractor trucks, the guided local search provides the most optimal solution. The second best is simulated annealing, and in last place is the tabu search. These optimal solutions are considered on the basis of the proportion of costs saved, the proportion of internal vehicles used, the proportion of orders completed by internal vehicles, the proportion of orders delivered on time, and the vehicle fleet utilization rate. However, to find an optimal solution, the guided local search takes 3,960 minutes while the other algorithms take only 3 minutes to arrive at a feasible solution. Thus, the results of simulated annealing algorithm are recommended as the most suitable solution for this vehicle routing problem, considering the time window.

**Table 2.** Algorithms comparison

| Results | Simulated an-nealing | Guided local search | Tabu search |
|---|---|---|---|
| Running time | 3 minutes | 3,960 minutes | 3 minutes |
| Proportion of costs saved | 18% | 23% | 15% |
| Proportion of orders completed by internal vehicles | 84% | 84% | 77% |
| Proportion of orders delivered on time | 100% | 100% | 100% |
| Vehicle fleet utilization rate: gap between the rate of each algorithm and target (60%) | Day 1: 15% Day 2: 2% Day 3: 6% | Day 1: 15% Day 2: 4% Day 3: 10% | Day 1: 27% Day 2: 3% Day 3: 8% |

## 5. Conclusion

In this study, a mixed integer programming model is proposed for a pickup and delivery vehicle routing problem, considering hard time windows in the logistics and transportation field. Unlike previous studies, such as Gunawan, A., et al. (2020), Birim, Ş. (2016), and Vincent, F. Y., et al. (2016), our research does not divide the distribution network into two separate groups of routes: 1) a route group to get the product and 2) a route group for product distribution. In addition, the routes in the previous studies were one-way, meaning that vehicles arrived to pick up or deliver the products at locations 1, 2, and 3 until reaching the final destination, and they did not turn around. By contrast, our study looks at the problem of more complex vehicle routing. Each vehicle performs both pickup and delivery jobs. Their routes are bidirectional, meaning that the vehicles can make a U-turn from site 2 to site 1 after the completion of the journey from site 1 to site 2. Furthermore, our mathematical model is created to schedule the jobs of 38 tractor trucks to pick up containers from, and deliver containers to, 243 locations in the network, including 35 depots/ports and 106 customers. All our input data are real data and collected from a logistics company in Vietnam. Another difference is that the constraint of a hard time window is not considered in Gunawan, A., et al. (2020), Mohammed, M. A., et al. (2017), Birim, Ş. (2016), or Vincent, F. Y., et al. (2016) but it is an important constraint in our study. This indicates that our study considers the limitations of the earliest and latest service time at each location. The final difference is the method used to solve the problem. Our study applies three methods: a tabu search, simulated annealing, and a guided local search to find the optimal solutions. Besides, the solutions were compared together to find the most suitable one for this real problem.

The result of the experiment shows that all algorithms can achieve a lower total cost than the original solution. A guided local search, simulated annealing, and a tabu search can save 23%, 18%, and 15% compared to the total cost of the recent

solution, respectively. The solutions obtained by the three algorithms can achieve a 100% success rate for picking up and delivering goods to customers on time, while the recent solution achieves only 80%. Although the guided local search can give a very good solution, it takes 3,960 minutes to find an optimal solution. In practice, decision-makers need to make decisions every day due to updates in the system such as customer demand, the number of available trucks, etc. Meanwhile, simulated annealing and the tabu search take only 3 minutes to arrive at an optimal solution. In comparing the running time and the quality of solutions, simulated annealing is the most suitable for a large-scale network.

The results of the simulated annealing algorithm indicate that there are 34 internal tractor trucks used to complete customer orders. These internal tractor trucks move 15,260 km/three days. The utilization of vehicles is approximately 60%, which satisfies the expectations of the logistics company. Besides, 34 internal tractor trucks implement 89 trips so the logistics company needs to rent external vehicles to complete another 17 trips. The total costs are 266,039,825 VND (Vietnam currency). Based on the simulated annealing algorithm's optimal solution, managers can have exact knowledge of the necessary numbers of vehicles, the vehicles' utilization, the total costs, and the routes. This leads to better policymaking and a better job schedule in the short-term.

We made two important contributions to research into the optimal vehicle scheduling. Firstly, a useful mathematical model is created to solve a complex and common problem in today's logistics and transportation industry. Considering the constraints of logistics problems in reality, our mathematical model helps users find optimal solution for scheduling vehicles' routes. Secondly, our results demonstrate that the simulated annealing algorithm is suitable for solving a complex problem of real-world pickup and delivery vehicle routing, both in terms of the quality of the solution and the time needed to find it. However, a limitation of our study creates research opportunities. In our study, the delivery time and the pickup and delivery locations are deterministic. Hence, some ideas could be proposed for future research directions. Firstly, researchers might develop an unpaired pickup and delivery vehicle routing problem with one type of vehicle and a single-product from our study. In that problem, a fleet of vehicles could pick up and deliver product used by customers and suppliers. Its aims would be to determine the optimal routing solutions and minimize transportation costs. Secondly, researchers could expand our study from a problem related to determined delivery times to a problem related to uncertain delivery times by considering the traffic jam factor. Its aims would to be minimize transportation costs and minimize the number of customers who are affected by products delivered late because of traffic jams.

# References

An, H., Li, W. 2011. *Synthetically improved genetic algorithm on the traveling salesman problem in material transportation*, In Proceedings of 2011 International Conference on Electronic & Mechanical Engineering and Information Technology, 7, IEEE, 3368-3371.

Afifi, S., Dang, D.C., Moukrim, A., 2016. *Heuristic solutions for the vehicle routing problem with time windows and synchronized visits*, Optimization Letters, 10(3), 511-525.

Aggarwal, D., Kumar, V., 2019. *Mixed integer programming for vehicle routing problem with time windows*, International Journal of Intelligent Systems Technologies and Applications, *18*(1-2), 4-19.

Alsheddy, A., 2011. *Empowerment scheduling: a multi-objective optimization approach using guided local search,* Doctoral dissertation, University of Essex.

Archetti, C., Speranza, M.G., Hertz, A., 2006. *A tabu search algorithm for the split delivery vehicle routing problem*, Transportation science, *40*(1), 64-73.

Barbarosoglu, G., Ozgur, D., 1999. *A tabu search algorithm for the vehicle routing problem*, Computers Operations Research, *26*(3), 255-270.

Bent, R., & Van Hentenryck, P. (2006). A two-stage hybrid algorithm for pickup and delivery vehicle routing problems with time windows. *Computers & Operations Research*, *33*(4), 875-893.

Birim, Ş., 2016. *Vehicle routing problem with cross docking: A simulated annealing approach*, Procedia-Social and Behavioral Sciences, *235*(Supplement C), 149-158.

Busetti, F., 2003. *Simulated annealing overview*, World Wide Web URL, www. geocities. com/francorbusetti/saweb. pdf, 4.

Cao, W., Yang, W., 2017. *A survey of vehicle routing problem*, In MATEC Web of Conferences, 100, EDP Sciences,01006.

Connor, A. M., Shea, K., 2000. *A comparison of semi-deterministic and stochastic search techniques*, In Evolutionary Design and Manufacture, Springer, London, 287-298.

Chen, Q., Li, K., & Liu, Z. (2014). Model and algorithm for an unpaired pickup and delivery vehicle routing problem with split loads. *Transportation Research Part E: Logistics and Transportation Review*, *69*, 218-235.

Dantzig, G. B., Ramser, J. H., 1959. *The truck dispatching problem*, Management science, *6*(1), 80-91.

Dongyang, X., Kunpeng, L., Jiehui, Y., Ligang, C., 2020. *A multicommodity unpaired pickup and delivery vehicle routing problem with split loads and unloads*, Industrial Management & Data Systems.

Gan, X., Wang, Y., Li, S., Niu, B., 2012. *Vehicle routing problem with time windows and simultaneous delivery and pick-up service based on MCPSO*, Mathematical Problems in Engineering, *2012*.

Gunawan, A., Widjaja, A. T., Gan, B., Yu, V. F., Jodiawan, P., 2020. *Vehicle routing problem for multi-product cross-docking*.

Huang, M., Yang, J., Ma, T., Li, X., Wang, T., 2017. *The modeling of milk-run vehicle routing problem based on improved CW algorithm that joined time window*, Transp. Res. Procedia, *25*, 716-728.

Rodrigue J.P., 2020, *New York: Routledge*, ISBN 978-0-367-36463-2, 456.

Kantawong, K., Pravesjit, S., 2020. *An Enhanced ABC algorithm to Solve the Vehicle Routing Problem with Time Windows*, ECTI Transactions on Computer and Information Technology (ECTI-CIT), 14(1), 46-52.

Kilby, P., Prosser, P., Shaw, P., 1999. *Guided local search for the vehicle routing problem with time windows*, In Meta-heuristics, Springer, Boston, MA, 473-486.

Kuo, Y., 2010. *Using simulated annealing to minimize fuel consumption for the time-dependent vehicle routing problem*, Computers Industrial Engineering, 59(1), 157-165.

Masmoudi, M., Mellouli, R. 2014. *MILP for synchronized-mTSPTW: application to home healthCare scheduling*, In 2014 International Conference on Control, Decision and Information Technologies (CoDIT), IEEE, 297-302.

Mohammed, M.A., Abd Ghani, M.K., Hamed, R.I., Mostafa, S.A., Ahmad, M.S., Ibrahim, D.A., 2017. *Solving vehicle routing problem by using improved genetic algorithm for optimal solution*, Journal of computational science, 21, 255-262.

Londoño, J.C., Tordecilla, R.D., Martins, L.D.C., Juan, A.A., 2020. *A biased-randomized iterated local search for the vehicle routing problem with optional backhauls*, TOP: An Official Journal of the Spanish Society of Statistics and Operations Research, 1-30.

Pérez-Rodríguez, R., Hernández-Aguirre, A., 2019. *A hybrid estimation of distribution algorithm for the vehicle routing problem with time windows*, Computers & Industrial Engineering, 130, 75-96.

Qi, C., Hu, L., 2020. *Optimization of vehicle routing problem for emergency cold chain logistics based on minimum loss*, Physical Communication, 101085.

Ruiz, E., Soto-Mendoza, V., Barbosa, A.E.R., Reyes, R., 2019. *Solving the open vehicle routing problem with capacity and distance constraints with a biased random key genetic algorithm*, Computers & Industrial Engineering, 133, 207-219.

Setamanit, S.-O. 2019. *Improving transportation contract management using simulation*, Polish Journal of Management Studies, 20 (2), 466-477.

Shuai, Y., Yunfeng, S., Kai, Z., 2019. *An effective method for solving multiple travelling salesman problem based on NSGA-II*, Systems Science & Control Engineering, 7(2), 108-116.

Straka, M., Rosová, A., Lenort, R., Besta, P., & Šaderová, J. 2018. *Principles of computer simulation design for the needs of improvement of the raw materials combined transport system*. Acta Montanistica Slovaca, 23(2), 163-174.

Tasar, B., Türsel Eliiyi, D., Kandiller, L., 2019. *Vehicle Routing with Compartments Under Product Incompatibility Constraints*, Promet-Traffic&Transportation, 31(1), 25-36.

Toth, P., & Vigo, D. (Eds.). (2014). *Vehicle routing: problems, methods, and applications*. Society for Industrial and Applied Mathematics.

Van Brummelen, G., 2012. *Heavenly mathematics: The forgotten art of spherical trigonometry*, Princeton University Press.

Vincent, F.Y., Jewpanya, P., Redi, A.P., 2016. *Open vehicle routing problem with cross-docking*, Computers & Industrial Engineering, 94, 6-17.

Voudouris, C., Tsang, E. P., 2003. *Guided local search*, In Handbook of metaheuristics, Springer, Boston, MA, 185-218.

Spliet, R., Desaulniers, G., 2015. *The discrete time window assignment vehicle routing problem*, European Journal of Operational Research, 244(2), 379-391.

Wang, C., Mu, D., Zhao, F., Sutherland, J. W., 2015. *A parallel simulated annealing method for the vehicle routing problem with simultaneous pickup–delivery and time windows*, Computers & Industrial Engineering, 83, 111-122.

Wei, L., Zhang, Z., Zhang, D., Leung, S.C., 2018. *A simulated annealing algorithm for the capacitated vehicle routing problem with two-dimensional loading constraints*, European Journal of Operational Research, 265(3), 843-859.

Xu, X., Yuan, H., Liptrott, M., & Trovati, M. (2018). Two phase heuristic algorithm for the multiple-travelling salesman problem. *Soft Computing*, *22*(19), 6567-6581.

Zhang, D., Cai, S., Ye, F., Si, Y. W., Nguyen, T.T., 2017. *A hybrid algorithm for a vehicle routing problem with realistic constraints*, Information Sciences, 394, 167-182.

Zhu, L., Hu, D., 2019. *Study on the vehicle routing problem considering congestion and emission factors*, International Journal of Production Research, 57(19), 6115-6129.

.

## 考虑物流配送网络中时间窗口的最佳拾起和运送集装箱的车辆路线计划：案例研究

**關鍵詞**

车辆路线问题 物流配送网络 时间窗 引导性本地搜索 禁忌搜索 模拟退火

**摘要**

这项研究描述了实际中考虑时间窗的皮卡和送货车路线问题。牵引车路线的问题由混合整数规划模型表示。除此之外，还提出了三种算法–引导局部搜索，禁忌搜索和模拟退火–作为解决方案。我们研究的目的是优化内部牵引车的数量，并创建最佳路线，以最大程度地降低总物流成本，包括内部车辆组的固定和可变成本以及外部车辆的租赁成本。此外，我们的研究还评估了解决方案的质量和寻找最佳解决方案的时间，以针对上述实际问题选择最佳的算法。 Python 的 OR 工具制定了一个新颖的数学模型。与当前解决方案相比，我们的结果使总成本降低了18%，内部车辆完成的订单比例提高了 84%，按时交付的订单比例提高了 100%。我们的研究为实际的复杂分销网络提供了一个具有时间限制和大量工作量的数学模型。所提出的数学模型为物流公司的决策提供了有效的解决方案。此外，我们的研究强调，对于该车辆路径问题，模拟退火算法比其他两种算法更合适。