

Article

The Volume-Based Pollution-Routing Problem with Time Windows: A Case Study

Bilal Bencharif ^{1,2,*} , Mohamed Amine Beghoura ^{1,2}  and Emrah Demir ^{3,*} 

¹ Department of Computer Science, University Mohamed El Bachir El Ibrahim, Bordj Bou Arreridj 34030, Algeria; mohamedamine.beghoura@univ-bba.dz

² Laboratoire Matériaux et Systèmes Electroniques LMSE, University Mohamed El Bachir El Ibrahim, Bordj Bou Arreridj 34030, Algeria

³ Logistics and Operations Management Section, Cardiff Business School, Cardiff University, Cardiff CF10 3EU, UK

* Correspondence: bilal.bencharif@univ-bba.dz (B.B.); demire@cardiff.ac.uk (E.D.)

Abstract: Green logistics has gained significant attention in recent years due to increasing pollution levels and their negative effects. This area of research is crucial as governments and countries worldwide recognize the severity of pollution and its detrimental effects. Despite progress, significant gaps remain due to the lack of advanced models that consider additional factors and the influence of speed on their outcomes. This paper presents a case study on the Volume-based Pollution-Routing Problem with Time Windows (VPRPTW). The objective is to minimize CO₂ emissions and improve customer satisfaction using a fleet of delivery vehicles. We propose a mathematical model and a probabilistic Tabu Search (TS) algorithm to solve the studied VPRPTW. The study revealed a decrease in daily fleet size from 16 to 12, indicating improved operational efficiency. In our study, we evaluate the impact of vehicle speed on fuel consumption and compare the results with a constant route speed to those obtained at varying speeds. Computational experiments reveal a significant difference of over 20% between fixed and variable speed assumptions. Additionally, we confirm that distance alone does not always correlate with energy consumption and CO₂ emissions. This highlights the importance of considering variable speeds in routing problems to assist logistics companies, urban planners, and policymakers achieve more accurate and environmentally friendly transportation solutions.

Keywords: green logistic; sustainability; Metaheuristics; Tabu Search algorithm; crowdsourced travel data



Received: 21 November 2024

Revised: 24 December 2024

Accepted: 13 January 2025

Published: 16 January 2025

Citation: Bencharif, B.; Beghoura, M.A.; Demir, E. The Volume-Based Pollution-Routing Problem with Time Windows: A Case Study. *Modelling* **2025**, *6*, 6. <https://doi.org/10.3390/modelling6010006>

Copyright: © 2025 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Each industrial revolution has introduced technological advancements that improved human comfort and prosperity. Energy, a fundamental element of these revolutions, has powered various sectors, including agriculture, commerce, construction, and transportation [1]. Historically, energy production has relied heavily on burning fossil fuels, a process that generates greenhouse gas (GHG) emissions such as methane (CH₄), carbon dioxide (CO₂), nitrous oxide (N₂O), and ozone (O₃). According to the United States Environmental Protection Agency, CO₂ emissions in 2021 amounted to 6.340 million metric tons, with transportation contributing 28% of total emissions [1]. In response to these environmental challenges, recent years have witnessed significant scientific efforts to develop strategies to mitigate the environmental impact of transportation, particularly through advances in operations research.

Over the past two decades, green logistics has evolved to include a subfield focused on green vehicle routing problems (GVRPs), with the aim of developing environmentally friendly routing strategies [2,3]. A key objective within this domain is to minimize fuel consumption, which is critical to reducing GHG emissions [4]. Bektas and Laporte [5] introduced the Pollution-Routing Problem (PRP), a variant of the Vehicle Routing Problem (VRP), to integrate fuel consumption into route planning. Other studies have also explored similar environmental objectives in routing, focusing on minimizing emissions, fuel consumption, and other ecological impacts in the context of logistics and transportation optimization.

Pollution from road vehicles is influenced by various factors, including fuel type and environmental conditions [6]. Accurate emission estimation requires accounting for variables such as distance traveled, vehicle weight, speed, and external conditions [7]. Numerous mathematical models have been developed to estimate fuel consumption, each differing in the factors considered and the experimental settings under which they were validated. Turkensteen [8] demonstrated that assuming a constant vehicle speed can result in deviations of up to 80% compared to real-world data.

The contributions of this paper are three-fold: First, it introduces a practical routing problem, termed the Volume-Based Pollutant-Routing Problem with Time Windows (VPRPTW), which incorporates multiple factors affecting emissions, with a specific focus on the impact of speed on fuel consumption using average speed values for each route. Second, the VPRPTW is formulated as a mixed-integer non-linear programming model aimed at minimizing total emissions, providing a comprehensive framework for environmentally conscious routing. Third, a probabilistic TS algorithm is developed to efficiently solve the problem, utilizing the Google Distance Matrix API to construct asymmetric distance and speed matrices.

The study found that daily fleet size decreased from 16 to 12, indicating improved operational efficiency. However, shorter distances did not always lead to lower energy consumption, as greater distances were associated with reduced energy use. The study also highlighted the impact of vehicle speed on fuel consumption, with variable speeds leading to deviations exceeding 20%. The findings underscore the importance of fine-tuning speed profiles for reducing energy consumption and improving environmental sustainability.

The paper is organized as follows: Section 2 provides a review of the literature on various GVRP variants. Section 3 outlines the problem definition, case study, and its formulation. Section 4 details the solution methodology, while Section 5 describes the numerical experiments and analyzes the results. Lastly, Section 6 offers concluding remarks, key insights, and limitations.

2. Literature Review

The GVRP is a crucial field of research in transportation and logistics that aims to minimize environmental effects. The purpose of this review of the literature is to summarize the body of knowledge regarding GVRP and the use of the Tabu Search algorithm as a well-known methodological technique. Determining the scope of the review among the many works in the literature was to rely on two bases, the most recent, and give an overview of GVRP by highlighting different variants of GVRP, single or multi-objectives, and different addressing methods.

The VRP, first introduced by Dantzig and Ramser in 1959 [9], is widely recognized as one of the most extensively studied operational-level transportation problems. Over the years, numerous variants of the VRP have been developed, with one notable extension being the GVRP. The GVRP has gained increasing attention in recent years due to growing concerns about the environmental impact of transportation, particularly last-mile logistics.

A significant body of literature has emerged that addresses various aspects of GVRP. For example, Fan et al. [10] proposed an integer programming model for the multi-depot VRP with a time-varying road network aimed at minimizing total costs. The authors used a hybrid genetic algorithm (GA) combined with a variable neighborhood search (VNS) and the Methodologies for Estimating Air Pollutant Emissions from Transport (MEET) model to estimate fuel consumption. Similarly, Gutiérrez-Padilla et al. [11] introduced the discrete speed VRPTW and proposed a mixed-integer linear programming model. Their model incorporated factors such as road slope, discrete speeds, pollution, and transport costs in the Colombian context. Dutta et al. [12] introduced a bi-objective model for GVRP, which was solved using the Non-Dominated Sorting Genetic Algorithm II (NSGA-II), considering both distance and emissions.

To further reduce CO₂ emissions, several recent studies have explored variable vehicle speeds in the context of the GVRP. Yao et al. [13] proposed a green vehicle routing optimization model that accounts for the impact of varying vehicle speeds on emissions, reflecting actual road conditions in emission calculations. Their model was solved using GA on the standard Solomon dataset. In another notable study, Shi and Lin [14] developed a bi-objective optimization model for the multi-depot GVRP with time windows, focusing on operating costs and emissions. They solved the problem using the NSGA-II algorithm. Wang et al. [15] investigated the effects of traffic restrictions on distribution logistics, proposing a multi-objective GVRP model with soft time windows. This model minimizes both delivery time and emissions, and the authors applied an improved ant colony optimization algorithm to solve it.

Liu et al. [16] introduced an adaptive large neighborhood search (ALNS) algorithm for the time-dependent GVRP with time windows (GVRPTW). The ALNS algorithm uses time discretization and feasibility checks to determine departure times, showing effective performance in solving small-sized instances and providing high-quality solutions for larger instances with up to 1000 customers. Zhang et al. [17] investigated the impact of driver behavior on fuel consumption and CO₂ emissions in road freight transportation. Using a dataset of over 4000 driving records, the study identified key behaviors, such as harsh acceleration and cornering, that significantly impact fuel consumption. The authors introduced an advanced fuel consumption model that incorporates these driver behaviors. For electric vehicles, Cataldo-Díaz et al. [18] explored the electric VRPTW (E-VRPTW), considering the battery state of charge. Their study compared linear and non-linear charging methods and found that non-linear charging reduced overall route time by minimizing unnecessary charging stops.

In a recent study, Lou et al. [19] addressed the low-carbon VRP with time-dependent speeds, speed fluctuations, road conditions, and time windows. They proposed a hybrid GA combined with adaptive-VNS, which was validated using a case study from Jingzhou, China. Ferreira et al. [20] proposed a GVRP model with two-dimensional loading constraints and split deliveries to reduce emissions, solving it using a variable neighborhood search method. Finally, Gkyrtis [21] examined the impact of road design on fuel consumption, particularly for heavy vehicles in urban freight transport. The study found that road design, particularly the longitudinal slope of highways, significantly influences fuel economy, with fuel consumption increasing by up to 2.5 times when the slope increases from 2% to 7%. This body of research highlights the growing recognition of environmental concerns in vehicle routing and the diverse approaches explored to reduce emissions and fuel consumption in logistics and transportation.

The TS algorithm is a neighborhood-based metaheuristic based on local search (LS) methods created by [22] in 1989; although it is relatively ancient, it has proven its effectiveness and is still widely used today, as it has several advantages, including the ability to

escape the local optimum. Therefore, we will briefly review the literature: in 2021, Gmira et al. [23] proposed a TS algorithm for solving a vehicle routing problem with time-dependent travel times, with the computational results showing that this strategy provides near-optimal solutions. Another recent study used a hybrid TS called simulated annealing TS for solving the Time-Dependent Vehicle Routing Problem with Soft Time windows presented by Liao and Shao [24]. Tlili et al. [25] addressed the ambulance routing problem with two algorithms derived from TS: hybrid TS (HTS) and TS-based hyper-heuristic (TSHH).

A closer look at the previous literature and Table 1, in particular, as well as many articles that are too numerous to mention, makes it clear that there is a scientific gap that needs to be covered. In PRP, the models that are used are considered rather old; the most famous and widely used example is the MEET model, which first appeared in 1998. Despite its simplicity, it lacks the study of many factors; usually, only the distance and speed factor are studied. On the other hand, previous studies have not addressed the complex interplay of route optimization and energy dynamics, as it is common knowledge that longer distances inevitably lead to more energy consumption.

Table 1. Comparison of features across various studies.

Reference	Varying Speed	Route-Energy Dynamics	Load Factor	Time Windows	New Model	Meta-Heuristic
[10]	✓		✓	✓		✓
[11]	✓			✓	✓	
[12]			✓			
[13]	✓		✓			✓
[14]				✓		
[15]				✓	✓	✓
[16]				✓		✓
[17]	✓					
[18]				✓	✓	
[19]	✓			✓		✓
[20]			✓			✓
[21]					✓	
This paper	✓	✓	✓	✓	✓	✓

3. Problem Description and Formulation

This section outlines the problem and its features. First, we introduce the case study and formally define the problem. Then, we present a mixed-integer non-linear programming model for the problem. In addition, the energy consumption method is introduced within the objective function.

This study focuses on a real-world case problem concerning a pharmaceutical company that owns a limited number of homogeneous vehicles with limited capacity (load and volume). The company's goal is to distribute pharmaceutical products to customers scattered in several cities in Algeria. The company's objective is to shift to environmentally friendly routing solutions that reduce energy consumption while satisfying customers according to the company's available resources.

The current distribution strategy involves aggregating customer requests each night for service on the following day. In addition, each vehicle is assigned a specific number of customers within a particular city. Basically, vehicles travel to their designated area daily, regardless of the actual number of customers to be served, even if it is just one customer. This approach encounters challenges related to energy consumption and the inability to ensure customer satisfaction, particularly when serving them later in the day. In addition, when the volume of customers is substantial, there is often a delay in servicing

some customers. The company management also classified customers according to their loyalty to the company. A higher number of transactions results in a more favorable classification, ensuring priority satisfaction and early service compared to other customers. To implement this, time windows are established that correspond to each customer's classification. The management determines these time windows based on the loyalty criteria mentioned earlier. Typically, customers are available from morning to evening, except for exceptional cases specified within the time window. Consequently, the determination of customers' time windows is structured primarily on loyalty, with consideration given to exceptional availability.

On the other hand, to estimate the energy consumed, fuel consumption models can be used, which are influenced by several factors that have been studied in [7,26,27]. These factors were collected by Demir et al. [28] and organized into five categories: vehicle, environment, traffic, driver, and operations. Usually, the focus is on vehicle, traffic, and environmental influences when estimating energy consumption. In particular, factors related to distance, payload, and vehicle speed are widely used. Using these factors as constants along the route may lead to inaccurate results.

3.1. Energy Consumption Model

Equation (1) is the objective function by which we want to reduce the energy consumed, and thus we can conclude the amount of CO₂ emissions (or fuel consumption) corresponding to each. We use an instantaneous energy consumption model, developed by Chaim and Shmerling [29], as an extension of the Guzzella and Sciarretta model [30] with more factors. This feature is very important compared to other models, as more factors can be included. Most models proposed in the literature require a map of engine consumption, assuming that fuel consumption is constant regardless of all operating modes. However, there is no doubt that this is inaccurate and does not apply in practice [29]. The proposed model addresses this drawback by incorporating calculations for energy requirement and fuel consumption based on the instantaneous specific fuel consumption. The most important feature of this chosen model is the consideration of changes in the movement pattern.

Equations (1)–(13) belong to Ben-Chaim and Shmerling [29] and Ben-Chaim et al. [31]. We have used these models to facilitate understanding of the model in an integrated, clear, and logical sequence.

The energy consumption formula between customer i and customer j is expressed as:

$$E_{s_{ij}} = \frac{\left[(E_{1_{ij}} + E_{2_{ij}}) * d_{ij} \right]}{100} \quad (1)$$

Given the complexity of the equation, it has been decomposed into several simpler equations for clarity. For the full logical sequence and detailed explanation, please refer to Appendix A.

This model in Equation (1) is based on the assumption that the engine is operating in two main modes. The first mode in which the movement is at average speed, $E_{1_{ij}}$ is the energy required to overcome the resistance forces in this mode. The second mode is based on episodic accelerations, where E_2 is the kinetic energy required, given by the joule (J). The models of $E_{1_{ij}}$ and $E_{2_{ij}}$ are given in Equations (2) and (3).

$$E_{1_{ij}} = \frac{1}{\eta_T \eta_P \eta_{P,n_{ij}}} \left(m_{a_{ij}} \cdot g \cdot c_{r_{ij}} + \frac{\rho}{2} \cdot C_D \cdot A_{f_{ij}} \cdot V_{a_{ij}}^2 \right) \cdot S \quad (2)$$

$$E_{2_{ij}} = \frac{q m_{a_{ij}}}{\eta_T \eta_e} \sum_1^k \frac{\alpha_l \gamma_{ml} S_l}{\mu_{p_{ij}} \mu_{n_{ij}}} \quad (3)$$

so:

$$E_{s_{ij}} = \left[\left(\frac{1}{\eta_T \eta_{P,n_{ij}}} \left(m_{a_{ij}} \cdot g \cdot c_{r_{ij}} + \frac{\rho}{2} \cdot C_D \cdot A_{f_{ij}} \cdot V_{a_{ij}}^2 \right) \cdot S + \frac{q m_{a_{ij}}}{\eta_T \eta_e} \sum_l^k \frac{\alpha_l \gamma_{ml} S_l}{\mu_{p_{l_{ij}}} \mu_{n_{l_{ij}}}} \right) * d_{ij} \right] / 100 \quad (4)$$

All of these parameters are described in Table 2.

Table 2. A list of parameters related to vehicle energy consumption model.

Parameter	Description
η_T	Efficiency of the transmission.
$\eta_{P,n}$	Efficiency of the engine.
m_a	Car mass.
g	Acceleration of gravity.
c_r	Rolling resistance coefficient.
ρ	Air density.
C_D	Coefficient of aerodynamic resistance of the car.
A_f	Characteristic area of the car.
V_a	Average speed of the vehicle.
S	Car mileage.
q	Number of accelerations in each acceleration interval.
η_e	Engine's peak efficiency.
α_l	Acceleration of the vehicle.
γ_{ml}	Mass factor of the vehicle.
S_l	Acceleration distance of the vehicle.
k	The number of acceleration intervals.
μ_{p_l}	Influence coefficient of the degree of power utilization on the peak efficiency of the engine.
μ_{n_l}	Influence coefficient of engine speed mode on the peak efficiency of the engine.
d	Traveled distance.

The factor $\eta_{P,n}$ represents the efficiency of the engine, which depends on the degree of power utilization and the speed of the engine. This is calculated by the following formula:

$$\eta_{P,n} = \eta_e \mu_p \mu_n \quad (5)$$

η_e , μ_p , and μ_n are calculated as follows:

$$\eta_e = 1 / (0.0119531 * g_e) \quad (6)$$

$$\mu_p = 0.5968 - 0.1666(P_i/P_e) + 2.4968(P_i/P_e)^2 - 2.1128(P_i/P_e)^3 \quad (7)$$

$$\mu_n = 0.7107 + 0.9963(n_i/n_p) - 1.0582(n_i/n_p)^2 + 0.3124(n_i/n_p)^3 \quad (8)$$

where:

$$P_i = (m_a g c_r + 0.5 c_r A_f V_a^2 + m_a a \gamma_m) V_a \quad (9)$$

$$P_e = 10^3 P_{max} \left[(n/n_p) + 0.5(n/n_p)^2 - 0.5(n/n_p)^3 \right] \quad (10)$$

$$n = \frac{9.55 V_a \delta_{ax} \delta_n}{r_d} \quad (11)$$

The remaining parameters are explained in Table 3.

Table 3. A list of parameters related to vehicle energy consumption.

Parameter	Description
g_e	Specific fuel consumption.
P_i	Engine power required for the given mode of motion.
P_e	Engine power by the performance characteristics of the engines.
n_i	Engine speed at average speed of the vehicle.
n_p	Engine speed at the maximum power of the engine.
P_{max}	Engine maximum power.
δ_{ax}	Finale drive gear ratio.
δ_n	Gear ratio in the gearbox.
r_d	Rolling radius of the tire.

The formulas for calculating both c_r and A_f are as follows.

$$c_r = 0.0136 + 0.40 * 10^{-7} V_a^2 \tag{12}$$

$$A_f = 1.6 + 0.00056(m_a - 765). \tag{13}$$

3.2. A Mixed-Integer Non-Linear Programming Formulation

We consider the formulation of VPRPTW in which we have a set K of homogeneous vehicles with load capacity Q and volume capacity V . Each customer i has an associated pair of values $[e_i, l_i]$. It is a time window that represents the earliest and the latest time when unloading can start. However, any vehicle is allowed to start serving customer i even if the vehicle arrives before it is ready, without waiting for the start of the time window e_i . This is due to the synchronized time windows e_i for all customers in the time matrix. The time constraints are modelled as two constraints (26) and (27), which makes this a soft time windows problem.

The VPRPTW model is defined on a complete graph $G = (N, A)$ with a set of nodes $N = \{0, 1, 2, \dots, n\}$ representing customers and the depot and a set of arcs A representing the roads between customers. We consider $n = |N|$ customers numbered from 1 to n , and node 0 is the depot. There are also two auxiliary vertices with numbers 0 and $n + 1$ representing the depot node for the route start and finish, respectively.

We also have the energy consumption E , where E_{ij} indicates the energy consumption of traveling from customer i to customer j , calculated using Equation (1). The matrix of travel times T specifies the time units t_{ij} required to get from customer i to customer j , and the distance from i to j is denoted by d_{ij} . The set $N_0 = N \setminus \{0\}$ is a customer set without the depot. Parameters e_0 and l_0 determine the earliest time when a vehicle can leave the depot and the latest time when it can return. The decision variables are specified as follows.

$$x_{ijk} = \begin{cases} 1 & \text{if arc } (i, j) \text{ is used by vehicle } k \\ 0 & \text{otherwise} \end{cases} \tag{14}$$

$$w_{ik} = \begin{cases} \text{service start time} & \text{if customer } i \text{ appears in route of vehicle } k \\ 0 & \text{otherwise} \end{cases} \tag{15}$$

The MINLP formulation for the VPRPTW is given as follows.

$$\min \sum_{i \in N} \sum_{j \in N} \sum_{k \in K} E_{ij} x_{ijk} \tag{16}$$

Objective function (16) is subject to the following constraints:

$$\sum_{k \in K} \sum_{j \in N} x_{ijk} = 1 \quad \forall i \in N_0 \quad (17)$$

$$\sum_{k \in K} \sum_{i \in N} x_{ijk} = 1 \quad \forall j \in N_0 \quad (18)$$

$$\sum_{i \in N_0} x_{i0k} \leq 1 \quad \forall k \in K \quad (19)$$

$$\sum_{j \in N_0} x_{0jk} \leq 1 \quad \forall k \in K \quad (20)$$

$$\sum_{i \in N} x_{ipk} = \sum_{j \in N} x_{pjk} \quad \forall p \in N_0, k \in K \quad (21)$$

$$\sum_{i \in N} \sum_{k \in K} q_i x_{ijk} \leq Q_k \quad \forall k \in K \quad (22)$$

$$\sum_{i \in N} \sum_{k \in K} v_i x_{ijk} \leq V_k \quad \forall k \in K \quad (23)$$

$$x_{ijk}(w_{ik} + s_i + t_{ij} - w_{jk}) \leq 0 \quad \forall k \in K, (i, j) \in N_0 \quad (24)$$

$$e_i = e_0 \quad \forall i \in N_0 \quad (25)$$

$$e_i \sum_{j \in N} x_{ijk} \leq w_{ik} \quad \forall k \in K, i \in N_0 \quad (26)$$

$$l_i \sum_{j \in N} x_{ijk} \geq w_{ik} \quad \forall k \in K, i \in N_0 \quad (27)$$

$$w_{0k} \geq e_0 \quad \forall k \in K \quad (28)$$

$$w_{n+1,k} \leq l_0 \quad \forall k \in K \quad (29)$$

$$u_{ik} - u_{jk} + (n - 1)x_{ijk} \leq n - 2 \quad \forall k \in K, (i, j) \in N_0 \quad (30)$$

$$x_{ijk} = \{0, 1\} \quad \forall k \in K, (i, j) \in N. \quad (31)$$

The objective function (16) minimizes the total energy consumption across all routes in the solution. Constraints (17) and (18) ensure that each customer is served by exactly one vehicle. Constraints (19) and (20) ensure that a vehicle departs from a depot and returns to the same depot only once. Constraints (21) enforce the continuity of the route by requiring that a vehicle proceeds from one customer to the next. Constraints (22) and (23) prevent overloads on the route by limiting total demand not to exceed vehicle capacity, considering both weight (Q) and volume (V). Constraints (24) define the relationship between the departure time of a vehicle from a customer and its immediate successor. Constraints (25)–(29) ensure feasibility in relation to the specified time windows. Constraints (30) eliminate sub-cycles within the solution. Finally, constraints (31) restrict the decision variable x_{ijk} to a binary value.

4. Solution Methodology

This section provides the details of the proposed algorithm. To solve VRP and its variants, the TS metaheuristic has been frequently used in the literature, as it has a number of advantages, iteratively exploring the solution space, employing memory structures to prevent the re-examination of previously investigated solutions, and integrating methods to evade local optima, thus facilitating a more extensive search for the global optimum. The problem at hand is an NP-hard problem, and only small-sized instances can be solved to optimality. Given the novelty of the problem, we opted to create a new algorithm for its solution based on the TS algorithm.

Algorithm 1 shows the steps of the proposed algorithm. In the initial stage, the algorithm generates data matrices on distance, time, and speed using the Distance Matrix API. Subsequently, it defines variables such as *maxIteration*, *iteration*, *tabuTenure*, *bestSolution*, and *currentSolution*. These variables represent, respectively, the maximum number of iterations in the main loop (a stop condition), the current iteration, the time or number of iterations an arc stays in the Tabu list, the best solution across all iterations, and the most recently obtained solution.

Algorithm 1 Probabilistic Tabu search for the VPRPTW

```

1: Initialize:
2: Create data matrixes (distance, time, and speed) using Distance Matrix API
3: Generate a feasible initial solution  $x$  using greedy insertion
4:  $currentSolution \leftarrow x$ 
5:  $bestSolution \leftarrow currentSolution$ 
6: while  $iteration < maxIteration$  do
7:    $bestNeighborCost \leftarrow \infty$ 
8:    $bestMove \leftarrow \emptyset$ 
9:   Generate a random number  $p$   $\triangleright p \in [0, 1]$ 
10:  if  $p > 0.8$  then
11:    Use 0-1 exchange strategy
12:  else
13:    Use i-cross exchange strategy
14:  end if
15:  for  $move \in allPossibleMoves$  do
16:     $neighborCost \leftarrow getCost(currentSolution, move)$ 
17:    if  $neighborCost < bestNeighborCost$  then
18:       $bestNeighborCost \leftarrow neighborCost$ 
19:       $bestMove \leftarrow move$ 
20:      if  $bestNeighborCost = \infty$  then
21:        break
22:      end if
23:    end if
24:  end for
25:  Update tabu list
26:  Generate a random number  $d$   $\triangleright d \in [0, 5]$ 
27:   $TabuMatrix \leftarrow tabuTenure + d$   $\triangleright$  for new arcs in besMove
28:   $TabuMatrix \leftarrow TabuMatrix - 1$   $\triangleright$  for each old arc in besMove
29:   $applyMove()$ 
30:   $iteration \leftarrow iteration + 1$ 
31:   $currentSolution \leftarrow bestNeighborCost$ 
32:  if  $currentSolution \leq bestSolution$  then
33:     $bestSolution \leftarrow currentSolution$ 
34:  end if
35: end while
36: return  $bestSolution$ 

```

The main loop stop condition occurs when we reach a certain number of predetermined iterations. The *bestNeighborCost* is declared, which is the best difference between the cost of the current situation and all possible moves. It is followed by the variable *bestMove*, which returns the best move among the moves in the current situation. Variable *p* is the probability of moving between the two strategies, i-cross exchange and 1-0 exchange. The candidate transition is qualified by the *allPossibleMoves* function, which considers all the conditions of weight, volume, time windows, and if the move is not Tabu.

After choosing one of the two strategies, the *getCost* function calculates the difference between the cost of the current situation and all possible moves and stores the value in the *neighborCost* variable. If it is better than *bestNeighborCost*, then *bestNeighborCost* takes its value, and the *bestMove* is the current move. After completing all possible moves, the Tabu list is updated as previously explained, and the move is applied to the *currentSolution*. If the *currentSolution* is better than the *bestSolution*, the *bestSolution* is updated. When the condition for stopping is met, the *bestSolution* is returned.

First, an initial solution is obtained using greedy insertion. It is the current solution and the base to start the algorithm. It is obtained most often after creating distance and time matrices. The algorithm remains operational until the stopping criterion is met. The best solution is adopted during the search, while at each iteration, the current solution is the best in the neighborhood, without comparing it with the previous solution regardless of whether it is better or worse. The iCROSS exchange and 1-0 exchange strategies were adopted to generate the current solution in the neighborhood structure. In the following section, we explain the important elements of the proposed algorithm.

4.1. Creation of Data Matrixes

In VRP with time windows, we need to know the distance and the time required to travel the distance between every two points (customers) back and forth. The way back may differ from the outbound journey, resulting in asymmetric matrices for distance, time, and speed. In practice, measuring these distances and times, especially with a considerable number of customers, is impractical due to the associated high costs and time constraints.

For this, we use The Distance Matrix Application Programming Interface (API), which is one of the Google Maps services that provide us with the distance and time of the matrix of origins and destinations between the start and endpoints on the recommended route [32]. Requests for Distance Matrix API are restricted to a maximum of 100 elements per (server-side and client-side) request. The total number of elements is determined by multiplying the number of origins by the number of destinations; 100 elements are calculated as (10 origins multiplied by 10 destinations). Through our review of the literature, we did not notice that a significant number of elements were used as in our case, which used 39,204 elements in each matrix (distance, time, and speed).

Following the asymmetric creation of distance and time matrices for all arcs, we generate the speed matrix using the established formula: $\text{speed}(v) = \text{distance}(d)/\text{time}(t)$.

$$v_{ij} = \frac{d_{ij}}{t_{ij}} \quad \forall i, j \in N_0.$$

The algorithm was implemented using the Java programming language and various libraries, the most notable being Google Maps Services 0.2.9 and GMapsFX-2.10.0. The Google Distance Matrix API, which provides trip time estimates based on crowdsourced data from Google users' mobile phones, was used in the implementation [33]. This API was first used by [34] to address routing problems. The study by [35] demonstrated that the Distance Matrix API delivers geographical data consistent with real-world conditions.

4.2. An Initial Solution

In the initial solution, routes are constructed sequentially using the greedy insertion heuristic. In each iteration, the algorithm attempts to sequentially assign customers to the current route until the constraints are satisfied. During the creation of each route, the next customer is selected from the unselected customers who will achieve the best solution for that route. The algorithm stops when an acceptable current solution is reached after visiting all customers, with the key advantage of the greedy heuristic being its ability to quickly obtain a feasible local solution.

4.3. The Neighborhood Structure

Choosing a neighborhood structure to exploit space solutions is the most important step in designing the TS algorithm [36]. The CROSS exchange [37] is the most convenient and widely used method for problems with time windows. The basic idea of the CROSS exchange is clear through its name, as two segments of two different routes are exchanged (swapped) in the form of a cross, as illustrated in Figure 1. Bräysy et al. [38] introduced an extension of the CROSS exchange where the segments to be exchanged were inverted. On this basis, it is called the inverted CROSS exchange (iCROSS exchange). The exchange depends on deleting and creating some arcs, so $(i - 1, i)$, $(k, k + 1)$ from the first route and $(j - 1, j)$, $(l, l + 1)$ from the second route are deleted. To achieve an inverse exchange of the two segments (i, k) and (j, l) , that is, (k, i) and (l, j) , between the two routes, $(i - 1, l)$, $(j, k + 1)$, and $(j - 1, k)$, $(i, l + 1)$ arcs are created. Details of how these exchanges work are illustrated in Algorithm 2. It is evident in this proposed solution that the number of customers in the segments is not restricted but is within the range of 1 to (length route - 2).

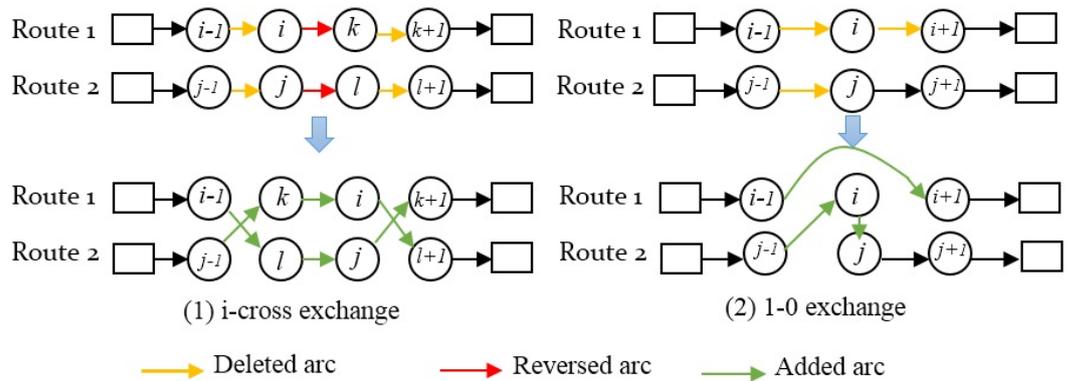


Figure 1. Neighborhood structure (1) i-cross exchange (2) 1-0 exchange.

When using the iCROSS exchange strategy alone, we observed that the results were generally good. However, when we replaced it with the 1-0 exchange strategy, we found that while some results were better, we often encountered the issue of the search becoming stagnant in a specific region. Based on these observations, we experimented with incorporating both strategies and proposed using one of them in each iteration based on probability. Specifically, in each iteration, the probability of selecting the iCROSS exchange strategy was set at 80%, while the probability for the 1-0 exchange strategy was 20%. This approach helps strike a balance between exploration and exploitation during the search process, which is a key challenge in LS algorithms.

Algorithm 2 iCROSS exchange function

```

1: best cost = Max value
2:  $item_1 = -1, item_2 = -1, item_3 = -1, item_4 = -1$ 
3: for  $i = 1 \rightarrow route_1.length - 2$  do
4:   for  $k = i \rightarrow route_1.length - 1$  do
5:     for  $forj = 1 \rightarrow route_2.length - 2$  do
6:       for  $forl = j \rightarrow route_2.length - 1$  do
7:         calculate new demand in  $route_1, route_2$ 
8:         calculate new volume in  $route_1, route_2$ 
9:         if  $capacity = true$  and  $volume = true$  then
10:          update  $route_1, route_2$ 
11:          calculate total time in  $route_1, route_2$ 
12:          if  $timeCheck(timeRoute_1) = true$  and  $timeCheck(timeRoute_2) = true$ 
then
13:            calculate  $old_{cost}$ 
14:            calculate  $new_{cost}$ 
15:            if  $crosscost < bestcost$  then
16:               $item_1 = i, item_2 = k, item_3 = j, item_4 = l$ 
17:            end if
18:          end if
19:        end if
20:      end if
21:    end for
22:  end for
23: end for
24: end for return  $item_1, item_2, item_3, item_4, bestcost$ 

```

4.4. Tabu List

The Tabu list, organized as a matrix with each cell representing an arc, is initialized with all values set to 0 in the first iteration. After each iteration, which concludes with the selection of a strategy (iCROSS exchange or 0-1 exchange) based on probability and results in obtaining a current solution, we update the Tabu list. This involves subtracting 1 from the value of each arc where the current value is greater than 0. Additionally, a specific number of iterations, referred to as the Tabu tenure, is added to the arcs that have been modified. These arcs include $(i - 1, l)$, $(j, k + 1)$, $(j - 1, k)$, and $(i, l + 1)$ according to the defined neighborhood structure. In addition to the Tabu tenure, a random number between 0 and 4 is added. Assigning a random number to each arc ensures that all arcs are not simultaneously removed from the Tabu list and that they are added together in a specific iteration.

5. Computational Experiments and Analysis

The probabilistic TS algorithm is applied to address real-life case study instances involving a pharmaceutical distribution company operating in Algeria, as illustrated in Figure 2. All datasets feature 16 identical vehicles with a specific capacity load and volume of 1200 kg and 5 cubic meters, respectively. The total number of customers is 198, distributed across four instances representing consecutive days of the week, chosen randomly to emulate diverse scenarios throughout the year. Key customer data include time windows, service time (estimated at 10 min), demand in kilograms, and order volume in cubic meters. The stopping condition for all tests is set at 500 iterations, with a Tabu tenure length of 15. The algorithm is executed on an Intel Core i3-4030U processor with 1.9 GHz and 4GB RAM using Java NetBeans IDE 8.2, running on Windows 7.



Figure 2. Customer locations on the map.

We address the problem by introducing time windows based on customer classifications. Each category, determined by the company’s management, is associated with a specific time window according to the loyalty status. Typically, customers are available from morning to evening, except for exceptional cases, and are served between 8 am and 2 pm. To eliminate waiting time, the customer readiness time is set at 8 am, ensuring all time windows open simultaneously at 8 am. Time window closing times are distributed as follows: 9 am, 11 am, 12 pm, and 2 pm, with priority given to loyalty in determining the closing time. The depot is available from 5 am to 8 pm. Table 4 provides the list of parameters with their values.

Table 4. Typical values assigned for the parameters.

Parameter	Value
η_T	0.95
g	9.8
ρ	1.165
C_D	0.30
g_e	107
S	10^5
q	14.37
γ_{mi}	1.1
α_i	2.277
S_i	169.45
n_i	62
n_p	$5 \cdot 10^3$
P_{max}	94
δ_{ax}	3.5
δ_n	1.2
r_d	0.5
k	4

To understand the impact of speed on the prediction of energy consumption during trips, we investigate scenarios involving a fixed speed for each trip, fixed speeds for individual vehicles within the same trip (referred to as varying speeds), and the average speed between each arc.

Table 5 presents the results obtained to solve the problem using the speed of Google Maps services, which means that the speed varies at each arc. The results of the energy consumption of fixed speeds along the path were segmented into a range from 40 to 100 km/h at intervals of 20 km/h (40, 60, 80, 100). More details on the results obtained are

provided in Tables 8–11. For example, Table 8 illustrates the results obtained for solving the problem at a constant speed of 40 km/h throughout the trip.

Table 5. Energy consumption, distance, and CO₂ emissions using speed from Google Maps services.

Dataset	Number of Customers	Number of Vehicles Used	Distance (km)	Energy Consumed (MJ)	CO ₂ Emission (Kg)	Fuel Consumption (L)	Time CPU (s)
w_1d_1	081	12	5620.62	13,375.43	984.43	349.23	45
w_1d_2	065	09	2857.61	4630.76	340.82	120.90	29
w_1d_3	081	14	5721.38	13,655.35	1005.03	356.54	40
w_1d_4	064	11	3221.00	6747.51	496.62	176.18	28
w_1d_5	077	13	4410.81	10,474.14	770.90	273.48	35
w_2d_1	079	15	6366.49	15,248.50	1122.29	398.13	34
w_2d_2	071	10	3333.11	7721.31	568.29	201.60	41
w_2d_3	082	13	4623.28	11,083.74	815.76	289.39	61
w_2d_4	075	11	3881.50	9117.64	671.06	238.06	42
w_2d_5	105	16	6373.45	15,014.20	1105.06	392.02	81
w_3d_1	069	13	5413.74	12,760.86	939.20	333.18	32
w_3d_2	082	12	4160.96	9910.85	729.44	258.77	50
w_3d_3	063	12	5593.45	13,330.59	981.13	348.06	26
w_3d_4	050	09	3377.93	7914.78	582.53	206.65	17
w_3d_5	058	10	3901.90	9149.96	673.44	238.90	23
w_4d_1	074	13	5109.84	12,218.53	899.28	319.02	31
w_4d_2	061	10	3447.32	8038.35	591.62	209.88	25
w_4d_3	049	08	3630.01	8634.20	635.48	225.44	13
w_4d_4	069	10	4011.79	9562.82	703.82	249.68	93
w_4d_5	080	13	4910.54	11,627.58	855.79	303.59	118
average	71.75	11.7	4498.34	10,510.86	773.60	274.44	43.2

As shown in Table 5, the first column represents the dataset divided into four sections: the weeks represented by the letter w , followed by the letter d , which represents the day in that week. Knowing the amount of energy consumed, we can deduce the emissions as well as the amount of fuel. Regarding the column of CO₂ emissions, its results were obtained depending on the amount of energy consumed by reducing the amount of emissions emitted through Equation (32), including diesel engine compatibility [39].

$$CO_2Emission(kg) = \frac{Energyconsumed(MJ) * 73.6}{1000} \quad (32)$$

The first note that draws attention is the number of vehicles used during trips every day, which ranged from a minimum of 8 vehicles to a maximum of 16 vehicles that were used every day. An average of 12 cars were used per day, with fewer than 4 vehicles operating on some days. This is a good improvement, especially in economic terms. A distance of 2857.61 km was obtained as the lowest distance traveled in the dataset, corresponding to a minimum of emissions estimated at 340.82 kg in w_1d_2 . On the other hand, a maximum distance of 6373.45 km was recorded in w_2d_5 , which corresponded to the amount of emissions of 1105.06 kg. However, this was not the maximum emission, which was recorded in w_2d_1 at 1122.29 kg over a distance of 6366.49 km. The difference between w_2d_5 and w_2d_1 was 6.96 km in distance and 17.23 kg in emissions. This shows that a long distance does not necessarily mean more fuel consumption. In the previous example, despite the decrease in distance by about 7 km, the emissions increased by 17 kg. This is explained by factors that control energy consumption, such as speed and weight.

To compare the performance of our proposed algorithm, TS, we evaluated it against an initial solution obtained using the greedy insertion heuristic in Table 6. The greedy

insertion heuristic provides a feasible, yet suboptimal, solution by iteratively inserting the best available option into the current solution. In contrast, the Tabu Search algorithm, with its memory-based approach and flexibility in exploring the solution space, refines the initial solution to achieve better optimization results. The comparison highlights the efficiency and effectiveness of TS in improving solution quality beyond the capabilities of the greedy insertion heuristic.

Table 6. The results of initial solution using the greedy insertion heuristic.

Dataset	Number of Customers	Number of Vehicles Used	Distance (km)	Energy Consumed (MJ)	CO ₂ Emission (Kg)	Fuel Consumption (L)	Time CPU (s)
w_1d_1	81	24	13,348.95	32,379.21	2383.11	845.41	1
w_1d_2	65	19	8486.14	20,574.13	1514.26	537.18	1
w_1d_3	81	29	14,616.02	35,603.85	2620.44	929.60	1
w_1d_4	64	20	9659.96	23,598.09	1736.82	616.14	1
w_1d_5	77	27	13,592.63	33,045.57	2432.15	862.81	1
w_2d_1	79	27	15,748.61	38,397.90	2826.09	1002.56	1
w_2d_2	71	21	10,356.94	25,041.73	1843.07	653.83	1
w_2d_3	82	25	14,313.21	34,790.21	2560.56	908.36	1
w_2d_4	75	23	11,625.92	28,364.57	2087.63	740.59	1
w_2d_5	105	31	17,795.65	43,269.05	3184.60	1129.74	1
w_3d_1	69	25	12,765.76	30,896.53	2273.98	806.70	1
w_3d_2	82	23	11,528.64	30,003.80	2208.28	783.39	1
w_3d_3	63	23	12,286.37	30,003.80	2208.28	783.39	1
w_3d_4	50	19	9364.14	22,759.15	1675.07	594.23	1
w_3d_5	58	18	10,143.91	24,720.02	1819.39	645.43	1
w_4d_1	74	25	12,765.76	30,896.53	2273.98	806.70	1
w_4d_2	61	22	10,042.51	24,451.66	1799.64	638.42	1
w_4d_3	49	16	8770.96	21,157.53	1557.19	552.42	1
w_4d_4	69	21	11,436.48	27,915.39	2054.57	728.86	1
w_4d_5	80	24	13,868.12	33,742.63	2483.46	881.01	1
average	71.75	23.15	12,091.18	29,395.23	2163.49	767.50	1

When trying to solve the problem, we identified some interesting results, which are shown in Table 7. In particular, for solutions employing a smaller number of vehicles, the company favors the adoption of new routes because of their increased profitability. Table 7 presents a comparison of two different solutions for the same case, where one involves a greater distance but results in fewer emissions.

The company prioritizes solutions that involve fewer vehicle usages when the emissions are relatively similar. For example, in the case of w_1d_5 , the alternative solution presented a distance of 120 km with one less vehicle, despite an increase of 7.79 kg in emissions. Similarly, in w_3d_2 , an alternative solution suggested using one less vehicle but resulted in a 10.66 kg increase in emissions. In the remaining cases, the number of vehicles remained the same.

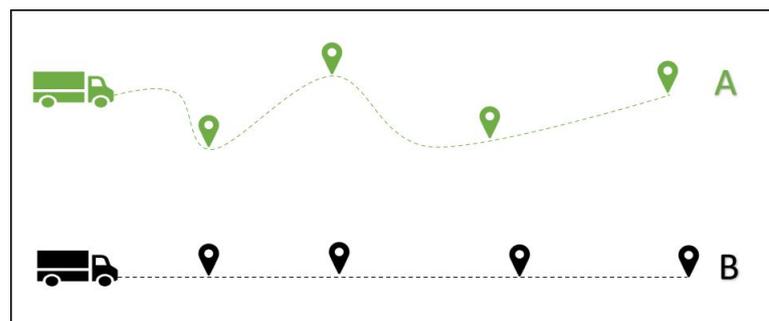
Concerning the discrepancy between distance and emissions, we note the lowest distance difference with an increase of 5.77 km in w_1d_1 in contrast to 7.23 kg less emissions, while an increase in distance of 197 km with 13.7 kg of emission reduction was recorded in w_2d_1 . The bottom line is that the distance length does not always indicate the amount of CO₂ emissions. There are exceptional cases where the distance is longer, but the amount of emissions is lower, as shown in Figure 3, where path A is the longest compared to path B, but the emissions in path A are lower than in path B.

Table 7. Cases with longer distance and lower emissions.

Dataset	Distance (km)	Number of Vehicles Used	CO ₂ Emission (Kg)	Fuel Consumption (L)	Difference Distance (km)
w_1d_1	5620.62	12	984.43	349.23	+5.77
	5614.85	12	991.66	351.79	
w_1d_2	2857.61	09	340.82	120.90	+87.41
	2770.20	09	472.84	167.74	
w_1d_5	4410.81	13	770.90	273.48	+120.47
	4290.34	12	778.69	276.24	
w_2d_1	6563.49	15	1108.59	393.27	+197
	6366.49	15	1122.29	398.13	
w_3d_2	4243.08	14	718.78	254.99	+82.12
	4160.96	12	729.44	258.76	
w_4d_2	3447.32	10	591.62	209.87	+6.73
	3440.59	10	595.94	211.41	

This outcome highlights the critical role of route optimization in reducing fuel consumption and emissions. Optimized routes often involve smoother driving patterns with fewer stops, less idling, and reduced speed fluctuations, all of which contribute to improved fuel efficiency. Furthermore, optimized routes tend to bypass high-traffic or congested areas, where vehicles are prone to burning excess fuel due to frequent braking and acceleration. This suggests that route characteristics, such as traffic flow, terrain, and driving conditions, can have a significant impact on emissions, sometimes outweighing the effect of distance alone.

Moreover, fleet management efficiency plays a role in cases like w_1d_5 and w_3d_2 , where fewer vehicles are used despite a longer distance, leading to a lower overall carbon footprint. This emphasizes the need for logistics systems to focus not only on minimizing distance but also on optimizing other variables like traffic avoidance, load balancing, and fuel economy to achieve sustainable outcomes. Therefore, the results provide strong evidence that route optimization strategies can deliver environmental and economic benefits by decoupling emissions from the distance traveled.

**Figure 3.** Long distance with lower CO₂ emissions.

To assess the impact of speed on the prediction of energy consumption or emissions, we analyze the variations in results at different speeds using both Google Maps services and fixed speeds in the study. The speed obtained from Google Maps services is called *the real speed* in our investigation.

The results of constant speeds along the route are shown in Tables 8–11 and the corresponding Figure 4a–d. In this experiment, each proposed solution is generated based on a constant speed along the path, simulating the optimization scenario with a constant speed. The algorithm's results are correlated to the objective function, where speed is one

of its components. This approach contrasts with proposing a single solution and deducing the corresponding emission results at constant speeds.

To determine the difference between the emissions, the constant speed and the real speed are calculated as the percentage deviation. We define it as the difference between the corresponding real emissions CO₂ and the obtained CO₂ emissions divided by the obtained CO₂ emissions and all multiplied by 100.

Table 8. Energy consumption at constant speed 40 km/h.

Dataset	Distance (km)	Number of Vehicles Used	Energy Consumed (MJ)	CO ₂ Emission (Kg)	Fuel Consumption (L)	Time CPU (s)	Real Energy Consumed (MJ)	Real CO ₂ Emission (Kg)	Real Fuel Consumption	Percentage Deviation
<i>w₁d₁</i>	5642.71	12	11,838.95	871.35	309.11	46	13,473.69	991.66	351.79	−13.81
<i>w₁d₂</i>	2783.96	9	5855.80	430.99	152.89	32	2783.96	481.03	170.65	−11.61
<i>w₁d₃</i>	5731.49	14	11,973.54	881.25	312.63	42	13,703.32	1008.56	357.79	−14.45
<i>w₁d₄</i>	3107.87	10	6531.20	480.70	170.53	28	6610.35	486.52	172.59	−1.21
<i>w₁d₅</i>	4418.20	13	9264.00	681.83	241.88	37	10,433.15	767.88	272.41	−12.62
<i>w₂d₁</i>	6525.39	14	13,663.22	1005.61	356.74	34	15,531.40	1143.11	405.51	−13.67
<i>w₂d₂</i>	3573.22	10	7477.35	550.33	195.23	33	8299.67	610.85	216.70	−11.00
<i>w₂d₃</i>	4627.46	11	9664.85	711.33	252.35	51	10,964.87	807.01	286.29	−13.45
<i>w₂d₄</i>	3838.31	11	8074.71	594.30	210.83	39	8926.19	656.97	233.06	−10.55
<i>w₂d₅</i>	6381.20	15	13,345.55	982.23	348.45	87	15,049.28	1107.63	392.93	−12.77
<i>w₃d₁</i>	5449.92	13	11,399.69	839.02	297.64	25	12,842.20	945.19	335.31	−12.65
<i>w₃d₂</i>	4168.20	13	8733.31	642.77	228.02	49	9778.21	719.68	255.31	−11.97
<i>w₃d₃</i>	5587.20	12	11,650.14	857.45	304.18	26	13,222.87	973.20	345.24	−13.50
<i>w₃d₄</i>	3382.73	9	7076.37	520.82	184.76	16	7943.63	584.65	207.41	−12.26
<i>w₃d₅</i>	3872.83	9	8092.68	595.62	211.30	24	9149.52	673.40	238.89	−13.06
<i>w₄d₁</i>	4952.59	12	10,392.00	764.85	271.33	36	11,752.32	864.97	306.85	−13.09
<i>w₄d₂</i>	3443.30	10	7208.84	530.57	188.22	25	8072.29	594.12	210.76	−11.98
<i>w₄d₃</i>	3615.53	9	7566.71	556.91	197.56	15	8509.64	626.31	222.18	−12.46
<i>w₄d₄</i>	4079.69	11	8536.02	628.25	222.87	27	9664.53	711.31	252.34	−13.22
<i>w₄d₅</i>	4984.64	12	10,429.24	767.59	272.30	43	11,807.56	869.04	308.29	−13.22
average	4508.32	11.45	9438.71	694.69	246.44	35.75	10,425.93	781.15	292.63	−12.13

Table 9. Energy consumption at constant speed 60 km/h.

Dataset	Distance (km)	Number of Vehicles Used	Energy Consumed (MJ)	CO ₂ Emission (Kg)	Fuel Consumption (L)	Time CPU (s)	Real Energy Consumed (MJ)	Real CO ₂ Emission (Kg)	Real Fuel Consumption	Percentage Deviation
<i>w₁d₁</i>	5636.35	12	13,237.69	974.29	345.63	46	13,476.19	991.85	351.86	−1.77
<i>w₁d₂</i>	2826.25	9	6648.30	489.31	173.58	31	6648.95	489.36	173.60	−0.01
<i>w₁d₃</i>	5788.16	14	13,550.65	997.33	353.80	40	13,937.85	1025.83	363.91	−2.78
<i>w₁d₄</i>	3163.70	10	7438.79	547.49	194.22	27	7143.73	525.78	186.52	4.13
<i>w₁d₅</i>	4290.34	12	10,075.37	741.55	263.06	39	10,580.08	778.69	276.24	−4.77
<i>w₂d₁</i>	6466.96	14	15,166.28	1116.24	395.99	33	15,400.13	1133.45	402.09	−1.52
<i>w₂d₂</i>	3402.77	10	7983.10	587.56	208.44	34	7969.26	586.54	208.07	0.17
<i>w₂d₃</i>	4661.77	12	10,910.65	803.02	284.87	54	11,128.57	819.06	290.56	−1.96
<i>w₂d₄</i>	3832.75	11	9021.43	663.98	235.55	38	9126.58	671.72	238.29	−1.15
<i>w₂d₅</i>	6318.48	15	14,803.72	1089.55	386.52	81	14,847.04	1092.74	387.65	−0.29
<i>w₃d₁</i>	5422.55	13	12,705.61	935.13	331.74	23	12,782.41	940.79	333.74	−0.60
<i>w₃d₂</i>	4223.01	13	9905.50	729.04	258.63	50	9678.40	712.33	252.70	2.35
<i>w₃d₃</i>	5593.45	12	13,074.27	962.27	341.36	27	13,330.59	981.13	348.06	−1.92
<i>w₃d₄</i>	3359.40	9	7875.72	579.65	205.63	16	7897.77	581.28	206.20	−0.28
<i>w₃d₅</i>	3858.41	10	9034.79	664.96	235.90	25	8997.78	662.24	234.93	0.41
<i>w₄d₁</i>	5027.43	13	11,808.95	869.14	308.33	35	12,095.83	890.25	315.82	−2.37
<i>w₄d₂</i>	3444.04	10	8074.00	594.25	210.81	23	8092.21	595.59	211.28	−0.22
<i>w₄d₃</i>	3622.06	9	8489.77	624.85	221.66	14	8586.17	631.94	224.18	−1.12
<i>w₄d₄</i>	4092.52	12	9594.97	706.19	250.52	33	9521.35	700.77	248.60	0.77
<i>w₄d₅</i>	5037.8	13	11,802.06	868.63	308.15	53	11,949.69	879.50	312.00	−1.24
average	4503.41	11.65	10,560.08	777.22	275.72	36.10	10,659.53	784.54	278.32	−0.71

The results revealed significant variations for each speed setting. At 40 km/h, the average deviation reached 12.13%, with the highest deviation at 14.45% and the lowest

at 1.21%, as detailed in Table 8. This variance in deviation is evident on all days, as shown in Figure 4a. For the fixed speed of 60 km/h (Table 9), the highest and lowest deviation rates were 4.77% and 0.01%, respectively, averaging at 0.71%—a relatively weak but non-negligible deviation. In particular, in w_1d_3 , the emission difference at the highest deviation rate was 37.14 kg, yet Figure 4b indicates that the constant speed and the real speed are nearly identical. There is a direct relationship between constant speed and deviation percentage, with deviations increasing at speeds of 80 km/h and 100 km/h (Tables 10 and 11). The highest deviation rates were recorded at 12.86% and 20.34%, respectively, with the lowest deviation rates at 7.29% and 12.32%, averaging at 8.71% and 17.26%, respectively. In particular, in w_3d_2 at 100 km/h, there was a significant emission difference of 142.21 kg, surpassing the deviation of 20%. Figure 4c,d illustrates the substantial differences between the real speed results for speeds of 80 km/h and 100 km/h.

Table 10. Energy consumption at constant speed 80 km/h.

Dataset	Distance (km)	Number of Vehicles Used	Energy Consumed (MJ)	CO ₂ Emission (Kg)	Fuel Consumption (L)	Time CPU (s)	Real Energy Consumed (MJ)	Real CO ₂ Emission (Kg)	Real Fuel Consumption	Percentage Deviation
w_1d_1	5638.27	12	14,494.39	1066.79	378.44	53	13,500.26	993.62	352.49	7.36
w_1d_2	2783.87	9	7169.01	527.64	187.18	32	6537.91	481.19	170.70	9.65
w_1d_3	5755.86	14	14,746.59	1085.35	385.03	37	13,744.99	1011.63	358.88	7.29
w_1d_4	3217.73	11	8278.39	609.29	216.15	26	7668.76	564.42	200.28	7.95
w_1d_5	4401.29	12	11,309.85	832.41	295.30	36	10,473.98	770.88	273.47	7.98
w_2d_1	6666.64	15	17,106.44	1259.03	446.64	34	15,913.02	1171.20	415.48	7.50
w_2d_2	3461.20	11	8884.82	653.92	231.98	42	8144.48	599.43	212.65	9.09
w_2d_3	4694.28	12	12,022.79	884.87	313.91	52	11,131.76	819.30	290.65	8.00
w_2d_4	3994.40	12	10,289.88	757.34	268.67	36	9343.91	687.71	243.97	10.12
w_2d_5	6359.29	16	16,305.50	1200.08	425.73	80	14,447.91	1063.37	377.23	12.86
w_3d_1	5479.71	13	14,048.39	1033.96	366.80	33	12,862.23	946.66	335.83	9.22
w_3d_2	4216.18	12	10,827.00	796.87	282.69	47	9957.02	732.84	259.97	8.74
w_3d_3	5539.75	11	14,174.27	1043.23	370.09	29	13,211.42	972.36	344.95	7.29
w_3d_4	3302.28	9	8471.18	623.48	221.18	15	7794.55	573.68	203.51	8.68
w_3d_5	3861.92	10	9895.98	728.34	258.38	22	9115.11	670.87	237.99	8.57
w_4d_1	5108.06	13	13,126.64	966.12	342.73	30	12,111.83	891.43	316.24	8.38
w_4d_2	3443.30	10	8836.64	650.38	230.72	22	8016.82	590.04	209.32	10.23
w_4d_3	3615.51	9	9275.60	682.68	242.18	14	8508.41	626.22	222.15	9.02
w_4d_4	4029.88	11	10,339.62	761.00	269.96	30	9570.52	704.39	249.88	8.04
w_4d_5	5061.95	14	12,979.81	955.31	338.90	41	11,998.52	883.09	313.28	8.18
average	4531.57	11.80	11,629.14	855.90	303.63	35.55	10,702.67	787.72	279.45	8.71

Table 11. Energy consumption at constant speed 100 km/h.

Dataset	Distance (km)	Number of Vehicles Used	Energy Consumed (MJ)	CO ₂ Emission (Kg)	Fuel Consumption (L)	Time CPU (s)	Real Energy Consumed (MJ)	Real CO ₂ Emission (Kg)	Real Fuel Consumption	Percentage Deviation
w_1d_1	5646.87	12	15,640.35	1151.13	408.36	52	13,434.97	988.81	350.78	16.42
w_1d_2	2821.28	9	7823.56	575.81	204.27	32	6651.44	489.55	173.67	17.62
w_1d_3	5772.04	14	15,932.02	1172.60	415.98	36	13,765.96	1013.17	359.42	15.74
w_1d_4	3203.10	10	8882.53	653.75	231.92	23	7576.33	557.62	97.82	17.24
w_1d_5	4290.09	12	11,881.63	874.49	310.23	34	10,578.36	778.57	276.20	12.32
w_2d_1	6247.80	14	17,286.02	1272.25	451.33	33	14,863.36	1093.94	388.08	16.30
w_2d_2	3443.10	11	9521.98	700.82	248.62	36	7952.05	585.27	207.63	19.74
w_2d_3	4644.49	12	12,820.13	943.56	334.73	49	11,049.71	813.26	288.50	16.02
w_2d_4	3904.82	11	10,841.24	797.92	283.06	35	9032.30	664.78	235.83	20.03
w_2d_5	6334.36	16	17,501.08	1288.08	456.95	88	14,887.27	1095.70	388.70	17.56
w_3d_1	5426.04	13	14,992.27	1103.43	391.44	26	12,788.83	941.26	333.91	17.23
w_3d_2	4131.23	13	11,430.76	841.30	298.45	49	9498.48	699.09	248.00	20.34
w_3d_3	5698.75	13	5709.08	1156.19	410.16	34	13,486.00	992.57	352.11	16.48
w_3d_4	3345.07	9	9247.55	680.62	241.45	15	7872.77	579.44	205.56	17.46
w_3d_5	3875.85	10	10,701.97	787.67	279.42	23	9130.76	672.02	238.40	17.21

Table 11. Cont.

Dataset	Distance (km)	Number of Vehicles Used	Energy Consumed (MJ)	CO ₂ Emission (Kg)	Fuel Consumption (L)	Time CPU (s)	Real Energy Consumed (MJ)	Real CO ₂ Emission (Kg)	Real Fuel Consumption	Percentage Deviation
w_4d_1	5043.38	13	13,967.21	1027.99	364.68	31	11,932.35	878.22	311.55	17.05
w_4d_2	3448.26	10	9535.22	701.79	248.96	24	7982.79	587.53	208.43	19.45
w_4d_3	3615.53	9	9995.48	735.67	260.98	14	8509.64	626.31	222.18	17.46
w_4d_4	4136.49	12	11,437.47	841.80	298.63	30	9891.91	728.04	258.27	15.63
w_4d_5	5087.65	13	14,055.24	1034.47	366.98	64	11,918.60	877.21	311.19	17.93
average	4505.81	11.80	11,960.14	917.07	325.33	36.40	10,640.19	783.12	272.81	17.26

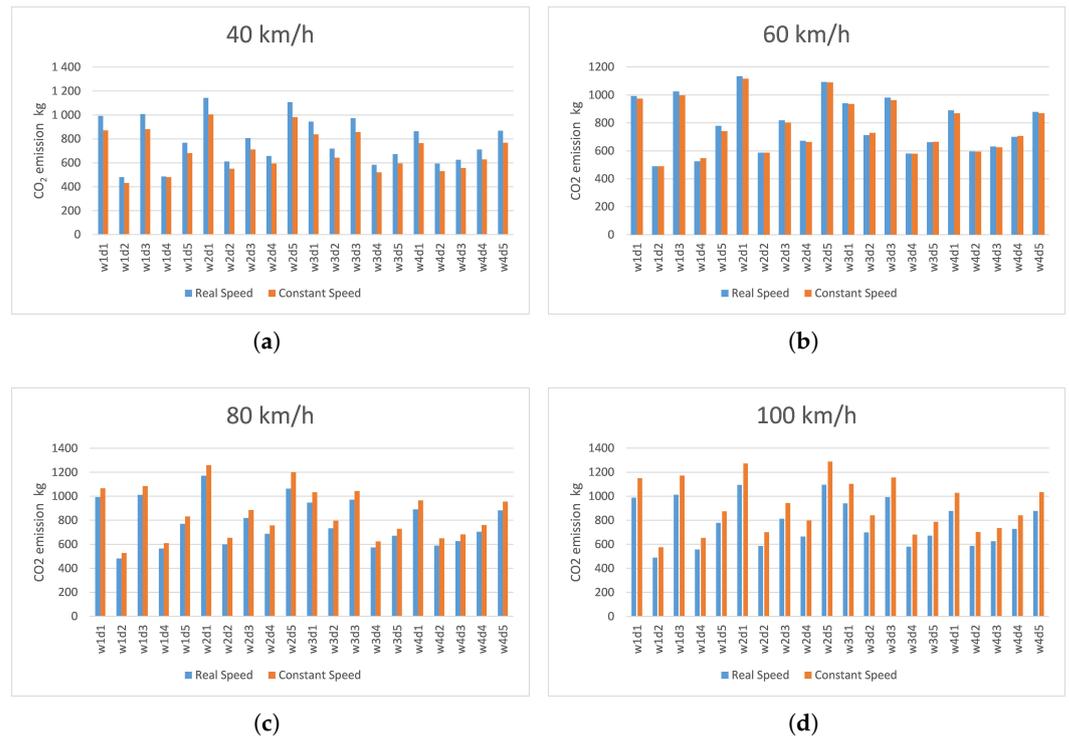


Figure 4. CO₂ emissions for actual speed vs. fixed speed: (a) fixed speed at 40 km/h; (b) fixed speed at 60 km/h; (c) fixed speed at 80 km/h; (d) fixed speed at 100 km/h.

Using the results of Tables 5 and 8–11, we have created the graph represented in Figure 5, which represents a comparison between emissions for the real speed in Table 5 with the rest of the fixed speeds in Tables 8–11. We note that the emissions are almost identical at a speed of 60 km/h. Compared with the rest of the results, the difference in value can reach proportions that cannot be discounted, as the farther the speed is from the value within 60 km/h, the greater the variance. Therefore, the increase or decrease in speed from the previous value leads to a variance with positive or negative values, respectively.

Statistical Analysis and Validation

An analysis of variance (one-way ANOVA) test was conducted to validate the observed differences in emissions and energy consumption across various speed profiles. The one-way ANOVA method is used to determine if there are statistically significant differences between the means of multiple groups. The speed profiles here represent the different groups being compared. Using one-way ANOVA, we can determine whether the observed variations in emissions and energy consumption are due to changes in speed profiles or if they are simply random. One-way ANOVA test results are considered significant if the

p -value is less than 0.05 and the F-statistic values are high. By using this rigorous statistical approach, the observed differences are not only noticeable but also statistically significant.

To check if the mean emissions and energy consumption significantly differ across the four speed profiles (40 km/h, 60 km/h, 80 km/h, and 100 km/h), we perform a one-way ANOVA.

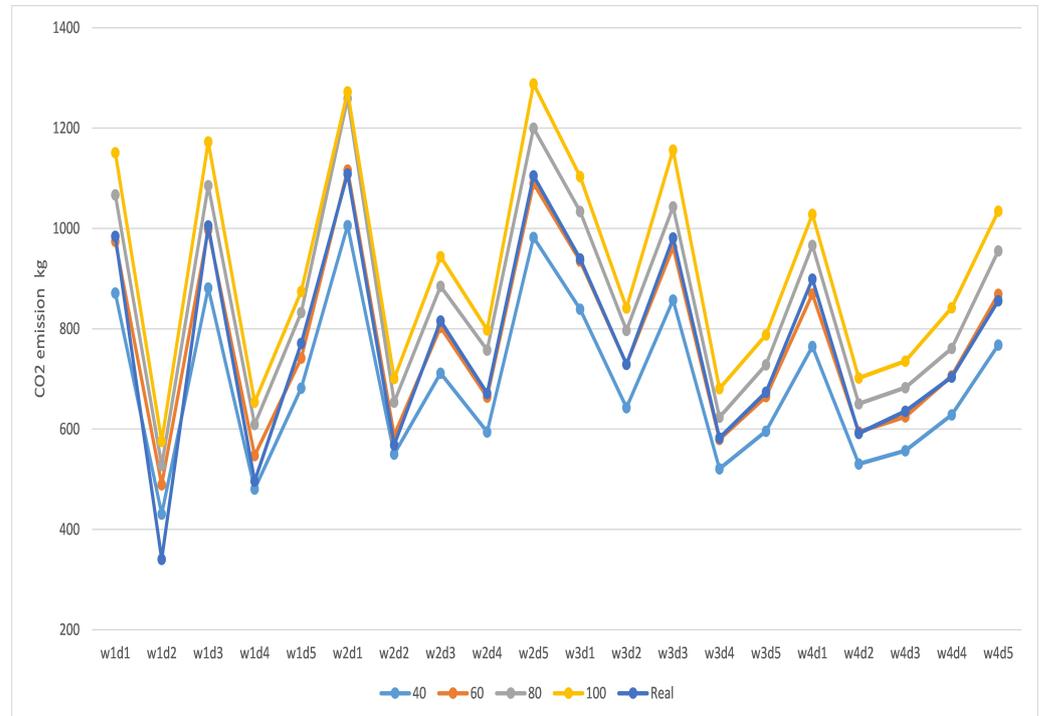


Figure 5. CO₂ emissions for real speed vs constant speeds conditions.

- Null hypothesis (H_0): there is no significant difference in emissions or energy consumption between speed profiles.
- Alternative hypothesis (H_1): there is a significant difference in emissions or energy consumption between at least two speed profiles.

The probabilities of p -values :

- If p -value < 0.05, we reject the null hypothesis and conclude that there are significant differences in emissions across the speed profiles.
- If p -value \geq 0.05, we fail to reject the null hypothesis, indicating no significant differences.

The results of the one-way ANOVA tests are as follows in Table 12.

Table 12. Statistical comparison of CO₂ emissions and energy consumption across speed profiles.

Parameter	F-Statistic	p -Value
CO ₂ emissions (kg)	4.75	0.0043
Energy consumption (MJ)	3.41	0.0216

The results indicate significant variations in both CO₂ emissions and energy consumption between the speed profiles, highlighting the impact of vehicle speed on environmental and energy-related factors. The findings suggest that vehicle speed plays a critical role in determining emissions and energy usage, with higher speeds generally corresponding to increased values.

To evaluate the proposed algorithm, we performed 10 independent runs using different random number generator seeds at real speeds to assess the robustness of our

non-deterministic approach. The results were analyzed by computing the average and standard deviation “S D” of the objective function, which is the energy consumed, across these runs, as shown in the Tables 13–16. The percentage of the standard deviation “% S D”, ranging from 1.06% to 4.04%, highlights the stability of the method in optimizing energy consumption. These relatively small variations demonstrate consistent performance across different runs, underscoring the reliability of the proposed approach under varying conditions.

Table 13. Performance statistics for the first week.

Dataset	Statistics	Distance	Number of Vehicles Used (km)	Energy Consumed (MJ)	CO ₂ Emission (Kg)	Fuel Consumption (L)	Time CPU (s)
w_1d_1	Average	5761.03	12.50	13,427.78	988.28	350.59	69.70
	S D	119.16	0.50	277.74	20.44	7.25	3.63
	% S D	2.07	4.00	2.07	2.07	2.07	5.21
w_1d_2	Average	2898.26	9.20	6755.25	497.18	176.37	42.80
	S D	99.81	0.42	232.64	17.12	6.07	1.75
	% S D	3.44	4.58	3.44	3.44	3.44	4.09
w_1d_3	Average	5858.84	14.60	13,655.77	1005.06	356.54	58.90
	S D	82.36	0.92	191.97	14.13	5.01	2.66
	% S D	1.41	6.28	1.41	1.41	1.41	4.52
w_1d_4	Average	3400.53	11.10	7925.94	583.35	206.94	38.60
	S D	125.09	0.70	291.56	21.46	7.61	2.46
	% S D	3.68	6.31	3.68	3.68	3.68	6.37
w_1d_5	Average	4673.25	14.20	10,892.38	333.51	284.40	54.10
	S D	183.96	0.75	428.77	142.90	11.19	4.09
	% S D	3.94	5.27	3.94	3.94	3.94	7.55

Table 14. Performance statistics for the second week.

Dataset	Statistics	Distance	Number of Vehicles Used (km)	Energy Consumed (MJ)	CO ₂ Emission (Kg)	Fuel Consumption (L)	Time CPU (s)
w_2d_1	Average	6823.90	15.30	15,905.12	1170.62	415.28	50.70
	S D	267.10	1.35	622.56	45.82	16.25	4.82
	% S D	3.91	8.79	3.91	3.91	3.91	9.50
w_2d_2	Average	3571.91	11.10	8325.39	612.74	217.37	53.70
	S D	75.61	0.70	176.25	12.97	4.60	3.16
	% S D	2.12	6.31	2.12	2.12	2.12	5.89
w_2d_3	Average	4772.45	12.30	11,123.61	818.69	290.43	74.60
	S D	50.36	0.46	117.37	8.64	3.07	4.92
	% S D	1.06	3.73	1.06	1.06	1.06	6.60
w_2d_4	Average	3864.91	10.90	9008.33	663.01	235.20	57.00
	S D	109.94	0.54	256.24	18.86	6.69	1.90
	% S D	2.84	4.94	2.84	2.84	2.84	3.33
w_2d_5	Average	6629.84	17.00	15,452.80	1137.32	403.46	129.00
	S D	138.77	0.89	323.44	23.81	8.45	9.82
	% S D	2.09	5.26	2.09	2.09	2.09	7.61

Table 15. Performance statistics for the third week.

Dataset	Statistics	Distance	Number of Vehicles Used (km)	Energy Consumed (MJ)	CO ₂ Emission (Kg)	Fuel Consumption (L)	Time CPU (s)
w_3d_1	Average	5536.61	13.30	12,904.70	949.78	336.93	35.00
	S D	100.16	0.64	233.44	17.18	6.09	1.41
	% S D	1.81	4.81	4.04	1.81	1.81	4.04
w_3d_2	Average	4251.63	13.20	9909.69	729.35	305.21	72.30
	S D	102.76	0.98	239.52	17.63	138.50	4.65
	% S D	2.42	7.42	2.42	2.42	2.42	6.43
w_3d_3	Average	5695.89	12.40	13,275.97	977.11	346.63	36.30
	S D	108.42	0.49	252.70	18.60	6.60	2.41
	% S D	1.90	3.95	1.90	1.90	1.90	6.64
w_3d_4	Average	5536.61	13.30	12,904.70	949.78	336.93	35.00
	Average	3450.15	9.90	8041.59	591.86	209.96	21.20
	S D	89.10	0.70	207.67	15.28	5.42	1.40
	% S D	2.58	7.07	2.58	2.58	2.58	6.60
w_3d_5	Average	3976.47	9.90	9268.35	682.15	241.99	31.60
	S D	68.72	0.54	160.18	11.79	4.18	1.91
	% S D	1.73	5.44	1.73	1.73	1.73	6.04

Table 16. Performance statistics for the fourth week.

Dataset	Statistics	Distance	Number of Vehicles Used (km)	Energy Consumed (MJ)	CO ₂ Emission (Kg)	Fuel Consumption (L)	Time CPU (s)
w_4d_1	Average	5166.98	13.40	12,043.18	886.38	314.44	49.90
	S D	205.11	0.80	478.07	35.19	12.48	1.58
	% S D	3.97	5.97	3.16	3.97	3.97	3.16
w_4d_2	Average	3506.75	10.40	8173.53	601.57	213.40	31.00
	S D	35.05	0.49	81.69	6.01	2.13	1.84
	% S D	1.00	4.71	1.00	1.00	1.00	5.95
w_4d_3	Average	3666.00	9.50	8544.70	628.89	223.10	20.70
	S D	48.59	0.50	113.26	8.34	2.96	1.10
	% S D	1.33	5.26	1.33	1.33	1.33	5.31
w_4d_4	Average	4123.14	11.80	9610.20	707.31	250.91	44.20
	S D	98.42	0.75	229.39	16.88	5.99	3.74
	% S D	2.39	6.34	2.39	2.39	2.39	8.45
w_4d_5	Average	5186.20	15.20	12087.97	889.67	315.61	63.70
	S D	77.98	1.17	181.75	13.38	4.74	2.83
	% S D	1.50	7.67	1.50	1.50	1.50	4.44

6. Conclusions

Reducing the environmental impact of transport operations through sustainable practices has been an urgent priority. To address this challenge, our research has introduced a variant of VRP that considers different types of customer groups. Specifically, we examined a fleet of homogeneous vehicles, constrained by both volume and weight capacities, tasked with distributing pharmaceuticals to pharmacies located across Algeria. Aiming to minimize GHG emissions, we developed an MINLP model tailored to this problem. In this research, we defined a so-called VPRPTW and proposed a probabilistic TS algorithm utilizing the Google Distance Matrix API. The algorithm incorporates a probabilistic neighborhood structure to alternate between the iCROSS exchange and 0-1 exchange methods and integrates customer loyalty-based time windows. We also formulated an MINLP model

to consider both vehicle volume and weight constraints, integrating an energy consumption equation to account for various operational factors.

Our results revealed significant insights. The daily fleet size was reduced from an average of 16 vehicles to 12, demonstrating improved operational efficiency. Interestingly, while distance is a critical factor in energy consumption, shorter distances do not always result in lower energy use. In some scenarios, greater distances were associated with reduced energy consumption, underscoring the complex interplay of route optimization and energy dynamics. In addition, the findings highlighted the influence of vehicle speed on fuel consumption. Specifically, maintaining variable speeds, as opposed to constant speeds, led to deviations in energy consumption exceeding 20%, as observed through API data. These results emphasize the critical role of fine-tuning speed profiles in reducing energy consumption and improving environmental sustainability. The results of this study were consistent with some results in the literature regarding the effect of the speed factor on energy consumption. However, the results on the distance factor were interesting; while it was prevalent in the logistics field that cost is related to distance, the studied model showed that the opposite can happen.

This study provides managers and policymakers with actionable insights to improve transport fleet operations, reduce emissions, and achieve sustainability goals. By reducing the number of vehicles in use, it enables managers to reduce both fixed costs (vehicle maintenance) and variable costs (fuel and employment). By addressing the trade-off between distance and energy use, the companies' goal is to minimize costs; therefore, these models can be adopted to reduce energy consumption, and they can verify that each route consumes less energy by conducting several experiments on alternative routes since the proposed model integrates real data from Google services. Moreover, the results of the study confirm the necessity of speed regulation to reduce fuel consumption, so decision-makers can guide and train drivers to travel at speeds that achieve environmental goals.

While this study provides valuable insights into green logistics, several limitations should be acknowledged. First, we did not study the problem with several algorithms to compare and find the most appropriate algorithm for the problem. Second, although the most important feature of the study is the multi-factor model, it does not include an important factor, which is the road gradient. Finally, the results of this study are within the framework of the proposed model, which limits the generalizability of the results.

Despite these limitations, the findings contribute to the growing body of evidence on green logistics and offer a foundation for future investigation. Future research should focus on refining vehicle speed management strategies to further optimize operational efficiency and minimize the environmental footprint of transport operations. To ensure that the proposed approach produces a stable and ideal solution, it will be essential to evaluate its convergence in further research. This evaluation will assist in verifying whether the selected parameter values are suitable and whether they improve the efficacy and efficiency of the algorithm. The no free lunch theorem (NFL), which states that no single algorithm performs well on all problem types, should also be considered. This emphasizes the necessity of researching and contrasting different metaheuristic strategies to determine which is most suited for the given problem. These assessments will provide a more thorough understanding of the effectiveness of the approach and areas for improvement. It is interesting to compare the results of the proposed model with other models when it relates to energy consumption amounts.

Author Contributions: Conceptualization, B.B., M.A.B., and E.D.; methodology, B.B.; software, B.B.; validation, B.B. and E.D.; formal analysis, E.D.; investigation, E.D.; resources, B.B. and E.D.; data curation, B.B.; writing—original draft preparation, B.B. and M.A.B.; writing—review and editing,

E.D.; visualization, B.B.; supervision, M.A.B. and E.D. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: Data used in this study can be found online: <https://github.com/Bilaldf/ResearchData> (accessed on 15 January 2025).

Acknowledgments: The authors thank Lino Guzzella, Department of Mechanical and Process Eng, ETH Zürich, for scientific assistance. The authors also thank Darmawan Satyananda, Department of Mathematics, State University of Malang, for his support. We thank the Editor-In-Chief and Associate Editor for their professional work, especially the speedy processing of the manuscript. We also do not forget the reviewers who provided a lush addition through their valuable comments.

Conflicts of Interest: The authors Bilal Bencharif, Mohamed Amine Beghoura, and Emrah Demir declare no conflicts of interest and that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Appendix A

To simplify the mathematical model to make it clearer, we include the logical sequence. The general formula for calculating energy $E_{s_{ij}}$ is shown in (1) and is in terms of $E_{1_{ij}}$, $E_{2_{ij}}$, and distance d_{ij} . Equations $E_{1_{ij}}$ and $E_{2_{ij}}$ were then incorporated into (2) and (3). The general equation $E_{s_{ij}}$ is formed after equating equations $E_{1_{ij}}$ and $E_{2_{ij}}$ in (4). All of the above factors and variables are explained in Table 2.

In Equation (5), the coefficient $\eta_{p,n}$, consisting of η_e , μ_p and μ_n , was discussed, and how to calculate them is shown in Equations (6)–(8). When calculating μ_p , we need the equation for P_i and P_e , which is shown in (9) and (10), and when calculating μ_n , we also need the equation for calculating n , which is shown in (11). All these factors and variables are explained in Table 3.

Finally, the mathematical formulas for calculating all the factors from c_r and A_f in (12) and (13) were provided.

References

1. Agency, E.P. Sources of Greenhouse Gas Emissions. 2022. Available online: <https://www.epa.gov/ghgemissions/sources-greenhouse-gas-emissions> (accessed on 13 November 2024).
2. Marrekchi, E.; Besbes, W.; Dhoub, D.; Demir, E. A review of recent advances in the operations research literature on the green routing problem and its variants. *Ann. Oper. Res.* **2021**, *304*, 529–574. <https://link.springer.com/article/10.1007/s10479-021-04046-8>. [CrossRef]
3. Moghdani, R.; Salimifard, K.; Demir, E.; Benyettou, A. The green vehicle routing problem: A systematic literature review. *J. Clean. Prod.* **2021**, *279*, 123691. [CrossRef]
4. Poonthalir, G.; Nadarajan, R. A Fuel Efficient Green Vehicle Routing Problem with varying speed constraint (F-GVRP). *Expert Syst. Appl.* **2018**, *100*, 131–144. [CrossRef]
5. Bektaş, T.; Laporte, G. The Pollution-Routing Problem. *Transp. Res. B Methodol.* **2011**, *45*, 1232–1250. [CrossRef]
6. Franco, V.; Kousoulidou, M.; Muntean, M.; Ntziachristos, L.; Hausberger, S.; Dilara, P. Road vehicle emission factors development: A review. *Atmos. Environ.* **2013**, *70*, 84–97. [CrossRef]
7. Demir, E.; Bektaş, T.; Laporte, G. A comparative analysis of several vehicle emission models for road freight transportation. *Transp. Res. D Transp. Environ.* **2011**, *16*, 347–357. [CrossRef]
8. Turkensteen, M. The accuracy of carbon emission and fuel consumption computations in green vehicle routing. *Eur. J. Oper. Res.* **2017**, *262*, 647–659. [CrossRef]
9. Dantzig, G.B.; Ramser, J.H. The Truck Dispatching Problem. *J. Manag. Sci.* **1959**, *6*, 80–91. [CrossRef]
10. Fan, H.; Zhang, Y.; Tian, P.; Lv, Y.; Fan, H. Time-dependent multi-depot green vehicle routing problem with time windows considering temporal-spatial distance. *Comput. Oper. Res.* **2021**, *129*, 105211. [CrossRef]
11. Gutiérrez-Padilla, M.V.; Morillo-Torres, D.; Gatica, G. A Novel Mathematical Model for a Discrete Speed Pollution Routing Problem with Time Windows in a Colombian Context. *IFAC-PapersOnLine* **2021**, *54*, 229–235. [CrossRef]

12. Dutta, J.; Barma, P.S.; Mukherjee, A.; Kar, S.; De, T.; Pamučar, D.; Šukevičius, Š.; Garbinčius, G. Multi-objective green mixed vehicle routing problem under rough environment. *Transport* **2022**, *37*, 51–63. [CrossRef]
13. Yao, Z.; Cao, H.; Cui, Z.; Wang, Y.; Huang, N. Research on Urban Distribution Routes Considering the Impact of Vehicle Speed on Carbon Emissions. *Sustainability* **2022**, *14*, 15827. [CrossRef]
14. Shi, Y.; Lin, Y. Bi-objective Optimization for Multi-depot Green Vehicle Routing problem with Time Windows Considering Carbon Emission. *J. Phys. Conf. Ser.* **2023**, *2583*, 012008. [CrossRef]
15. Wang, H.; Li, M.; Wang, Z.; Li, W.; Hou, T.; Yang, X.; Zhao, Z.; Wang, Z.; Sun, T. Heterogeneous Fleets for Green Vehicle Routing Problem With Traffic Restrictions. *IEEE Trans. Intell. Transp. Syst.* **2023**, *24*, 8667–8676. [CrossRef]
16. Liu, Y.; Roberto, B.; Zhou, J.; Yu, Y.; Zhang, Y.; Sun, W. Efficient feasibility checks and an adaptive large neighborhood search algorithm for the time-dependent green vehicle routing problem with time windows. *Eur. J. Oper. Res.* **2023**, *310*, 133–155. : 10.1016/j.ejor.2023.02.028 [CrossRef]
17. Zhang, Z.; Demir, E.; Mason, R.; Cairano-Gilfedder, C.D. Understanding freight drivers' behavior and the impact on vehicles' fuel consumption and CO₂e emissions. *Oper. Res.* **2023**, *23*, 59. [CrossRef]
18. Cataldo-Díaz, C.; Linfati, R.; Escobar, J.W. Mathematical models for the electric vehicle routing problem with time windows considering different aspects of the charging process. *Oper. Res.* **2023**, *24*, 1. [CrossRef]
19. Lou, P.; Zhou, Z.; Zeng, Y.; Fan, C. Vehicle routing problem with time windows and carbon emissions: A case study in logistics distribution. *Environ. Sci. Pollut. Res.* **2024**, *31*, 41600–41620. [CrossRef]
20. Ferreira, K.M.; de Queiroz, T.A.; Munari, P.; Toledo, F.M.B. A variable neighborhood search for the green vehicle routing problem with two-dimensional loading constraints and split delivery. *Eur. J. Oper. Res.* **2024**, *316*, 597–616. [CrossRef]
21. Gkyrtis, K. Theoretical Considerations from the Modelling of the Interaction between Road Design and Fuel Consumption on Urban and Suburban Roadways. *Modelling* **2024**, *5*, 737–751. [CrossRef]
22. Glover, F. Tabu search—part I. *INFORMS J. Comput.* **1989**, *1*, 190–206. [CrossRef]
23. Gmira, M.; Gendreau, M.; Lodi, A.; Potvin, J.Y. Tabu search for the time-dependent vehicle routing problem with time windows on a road network. *Eur. J. Oper. Res.* **2021**, *288*, 129–140. [CrossRef]
24. Liao, X.; Shao, Y. Optimization algorithm for the time-dependent vehicle routing problem with softtime windows: A research based on mixed integer model and mixed tabu search. *Math. Found. Comput.* **2024**. [CrossRef]
25. Tlili, T.; Nasser, S.B.; Chicano, F.; Krichen, S. Tabu Search-based hyper-heuristic for Solving the Heterogeneous Ambulance Routing Problem with Time Windows. *J. Intell. Transp. Syst.* **2024**, *22*, 446–461. [CrossRef]
26. Ardekani, S.; Hauer, E. *Traffic Impact Models*, 1st ed.; Oak Bridge National Laboratory: Oak Ridge, TN, USA, 1992; pp. 1–24. Available online: <https://web.engr.oregonstate.edu/~bertinir/TFT/docs/chap7.pdf> (accessed on 24 December 2024)
27. Eglese, R.; Black, D. *Optimizing the Routing of Vehicles*, 1st ed.; Kogan Page Limited: London, UK, 2010; pp. 215–228. Available online: https://gc.scalahed.com/recursos/files/r161r/w25826w/McKinno_S1.pdf#page=228 (accessed on 24 December 2024).
28. Demir, E.; Bektaş, T.; Laporte, G. A review of recent research on green road freight transportation. *Eur. J. Oper. Res.* **2014**, *237*, 775–793. [CrossRef]
29. Ben Chaim, M.; Shmerling, E. A model of vehicle fuel consumption at conditions of the EUDC. *Int. J. Mech.* **2013**, *7*, 10–17. Available online: <https://www.naun.org/main/NAUN/mechanics/16-659.pdf> (accessed on 24 December 2024).
30. Guzzella, L.; Sciarretta, A. In *Vehicle Propulsion Systems*, 3rd ed.; Springer: Heidelberg, Germany, 2013; pp. 1–403. [CrossRef]
31. Ben Chaim, M.; Shmerling, E.; Kuperman, A. Analytic modeling of vehicle fuel consumption. *Energies* **2013**, *6*, 117–127. [CrossRef]
32. developers, G. Get Started | Distance Matrix API | Google Developers. 2024. Available online: <https://developers.google.com/maps/documentation/distance-matrix/start?hl=id> (accessed on 13 November 2024).
33. Kumara, S. Use of Crowdsourced Travel Time Data in Traffic Engineering Applications. Ph.D. Thesis, University Of Moratuwa, Moratuwa, Sri Lanka, 2018. Available online: <http://dl.lib.uom.lk/handle/123/14615> (accessed on 24 December 2024).
34. Király, A.; Abonyi, J. A Google Maps based novel approach to the optimization of multiple Traveling Salesman problem for limited distribution systems. *Acta Agrar. Kaposvariensis* **2010**, *14*, 1–14. Available online: <https://journal.uni-mate.hu/index.php/aak/article/view/1952> (accessed on 24 December 2024).
35. Satyananda, D.; Artikel, R. Google Map API service for VRP Solving Application. *Semin. Nas. Integr. Mat. Dan Nilai Islam. Malang, Indones.* **2017**, *1*, 240–245. Available online: <http://conferences.uin-malang.ac.id/index.php/SIMANIS/article/view/76> (accessed on 24 December 2024).
36. Gendreau, M. An Introduction to Tabu Search. In *Handbook of Metaheuristics*; Glover, F., Kochenberger, G.A., Eds.; Springer: Boston, MA, USA, 2003; pp. 37–54. [CrossRef]
37. Taillard, É.; Badeau, P.; Gendreau, M.; Guertin, F.; Potvin, J.Y. A tabu search heuristic for the vehicle routing problem with soft time windows. *Transp. Sci.* **1997**, *31*, 170–186. [CrossRef]

38. Bräysy, O. A reactive variable neighborhood search for the vehicle-routing problem with time windows. *INFORMS J. Comput.* **2003**, *15*, 347–368. [[CrossRef](#)]
39. Jokiniemi, T. *Greenhouse Gas Emissions from Direct Combustion of Various Fuels*; Technical Report; University of Helsinki: Helsinki, Finland, 2011. Available online: https://enpos.weebly.com/uploads/3/6/7/2/3672459/co2_direct_combustion_jokiniemi.pdf (accessed on 24 December 2024).

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.