sustainability

MDPI

Article

Solution of the Capacity-Constrained Vehicle Routing Problem Considering Carbon Footprint Within the Scope of Sustainable Logistics with Genetic Algorithm

Bedrettin Türker Palamutçuoğlu [1], Selin Çavuşoğlu [1], Ahmet Yavuz Çamlı [1], Florina Oana Virlanuta [2,*], Silviu Bacalum [3], Deniz Züngün [4] and Florentina Moisescu [5]

[1] Department of Management and Organization, Kula Vocational School, Manisa Celal Bayar University, Manisa 45140, Turkey; turker.palamutcuoglu@cbu.edu.tr (B.T.P.); selin.cavusoglu@cbu.edu.tr (S.Ç.); ahmetyavuz.camli@cbu.edu.tr (A.Y.Ç.)
[2] Department of Economics, Faculty of Economics and Business Administration, "Dunarea de Jos" University of Galati, 800008 Galaţi, Romania
[3] Department of Sciences, Cross-Border Faculty, "Dunarea de Jos" University of Galati, 800201 Galaţi, Romania; silviu.bacalum@ugal.ro
[4] Department of International Trade and Logistics, Faculty of Economics and Administrative Sciences, İstanbul Yeni Yüzyıl University, İstanbul 35000, Turkey; deniz.zungun@yeniyuzyil.edu.tr
[5] Department of Business Administration, Faculty of Economics and Business Administration, "Dunarea de Jos" University of Galati, 800008 Galaţi, Romania; florentina.moisescu@ugal.ro
* Correspondence: florina.virlanuta@ugal.ro

Abstract: One of the important problems of sustainable logistics is routing vehicles in a sustainable manner, the green vehicle routing problem, or vehicle routing problems which aim to reduce CO_2 emissions. In the literature research, it was seen that these problems were solved with heuristic, metaheuristic, or hyper-heuristic methods and hybrid approaches since they are in the NP-hard class. This work presents a parallel multi-process genetic algorithm that incorporates problem-specific genetic operators to minimize CO_2 emissions in the capacity-constrained vehicle routing problem. Unlike previous research, the algorithm combines parallel computing with tailored genetic operators in order to enhance the diversity of solutions and speed up convergence. Genetic algorithm models were developed to minimize total distance, CO_2 emissions, and both objectives simultaneously. Two genetic algorithm models were developed to minimize total distance and CO_2 emissions. Experimental results using the reference CVRP examples such as A-n32-k5 and B-n44-k7 show that the proposed approach reduces CO_2 emissions by 1.2% more than hybrid artificial bee colony optimization, 1.3% more than ant colony optimization, and 4% more than the traditional genetic algorithm. Experimental results using benchmark CVRP instances demonstrate that the proposed approach outperforms hybrid artificial bee colony optimization, ant colony optimization, and traditional genetic algorithms for most of the test cases. This is done by exploiting multi-core processors, and the parallel architecture has improved computational efficiency; the modules compare and update solutions against the global optimum. Results obtained show that prioritizing CO_2 emissions as the only objective yields better results compared to multi-objective models. This study makes two significant contributions to the literature: (1) it introduces a novel parallel genetic algorithm framework optimized for CO_2 emission reduction, and (2) it provides empirical evidence underscoring the advantages of emission-focused optimization in CVRP.

Keywords: carbon footprint; green vehicle routing; sustainable vehicle routing; genetic algorithm

check for updates

Academic Editor: Manuel Pedro Rodríguez Bolívar

Received: 3 December 2024
Revised: 26 December 2024
Accepted: 14 January 2025
Published: 17 January 2025

Citation: Palamutçuoğlu, B.T.; Çavuşoğlu, S.; Çamlı, A.Y.; Virlanuta, F.O.; Bacalum, S.; Züngün, D.; Moisescu, F. Solution of the Capacity-Constrained Vehicle Routing Problem Considering Carbon Footprint Within the Scope of Sustainable Logistics with Genetic Algorithm. *Sustainability* **2025**, *17*, 727. https://doi.org/10.3390/su17020727

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

Sustainability has become a large focus of public discourse as well as scientific literature due to the alarming rise in global warming and its associated climate change. Most of the initial research focused on the conceptual aspects and cause-and-effect analysis, while many recent efforts have focused more on practical and measurable approaches to cutting emissions of greenhouse gases—primarily CO_2, CO, and NO_2—which make major contributions to climate change. Transportation, which is one of the main sectors in emissions, has captured particular attention, driving the need to optimize logistics operations to achieve sustainability goals.

The logistics sector is crucial to supply chains, but an increasing reliance on outsourcing and global trade has greatly heightened its environmental impact. Of the many challenges in this field, one that has attracted much attention is the capacitated vehicle routing problem (CVRP). The CVRP basically aims to find the most efficient routes for a fleet of vehicles with capacity constraints so that customers can be served effectively at minimal cost, distance, or even emissions. The integration of the CVRP with CO_2 emission reduction strategies, called green vehicle routing, is a key area of research within sustainable logistics.

One of the most significant challenges in logistics is the capacitated vehicle routing problem (CVRP). Vehicle routing problems (VRPs) are a class of combinatorial optimization problems where the objective is to determine optimal routes for a fleet of vehicles to serve customers while minimizing cost, distance, or CO_2 emissions. In the context of sustainable logistics, green vehicle routing problems (G-VRPs), which focus specifically on reducing CO_2 emissions, have gained significant attention. Many factors influence CO_2 emissions in transportation, including vehicle characteristics (engine power, friction coefficients, capacity, fuel type, engine operating mode, etc.) and operational factors like speed, acceleration, and road conditions (weather conditions, the slope of the road, curvy or straight, altitude, etc.). Additionally, the amount of cargo carried directly impacts the emission levels during transportation.

In this study, we utilize the CO_2 emission formula proposed by Adiba et al. [1] to calculate emissions for capacity-constrained vehicle routing problem instances (A-n32-k5, B-n44-k7, E-n76-k10, and M-n200-k17, etc.). This formula has been taken from Adiba et al. because it includes the load carried by the vehicle, which many methods do not take into account. It explicitly considers the load carried by vehicles and thus allows a more realistic estimate of CO_2 emissions, compared to methods that do not consider this relation. Moreover, the usage of this formula allows comparison to other works in the literature based on this approach.

To achieve sustainable transportation goals, models aiming to minimize distance or CO_2 emissions were tested to identify the most effective approach for reducing environmental impact. Previously, various studies such as [1–11] have explored related problems, with ant colony optimization and bee colony optimization along with particle swarm optimization being the most preferred when developing optimization metaheuristic algorithms for many optimization algorithms. Of these, the GAs have found a wider acceptance due to their efficiency in solving optimization problems; GAs have been widely adopted because of their effectiveness in solving combinatorial optimization problems [8,10,12–16]. The algorithms that are based on population or swarm intelligence–ant colony optimization [17], bee colony optimization [18], and particle swarm optimization [19] are inspired by nature and the self-organizing capabilities observed in groups of animals. All of these algorithms mimic collective behavior and decision-making processes of living creatures in solving difficult optimization problems. Ant colony optimization, for example, relies on pheromone trails to explore the optimal path, while particle swarm optimization models each individual as an agent that flows through the solution space, driven by personal

and collective experiences. Genetic algorithms, which are population-based algorithms, simulate the process of natural evolution [20,21]. They operate on a population of candidate solutions that evolve over generations using genetic operators such as crossover, mutation, and selection. The crossover operator combines features from parent solutions to create offspring, while the mutation operator introduces random changes to maintain diversity and avoid premature convergence to local optima. Recent advancements include hybrid approaches integrating local search heuristics [22] or swarm intelligence methods [23] and parallel computing techniques to improve computational efficiency and scalability.

The main aim of this paper is to show that enhanced genetic algorithms, with well-performing genetic operators, combined with parallel programming, can solve the capacity-constrained vehicle routing problem while effectively reducing CO_2 emissions. The proposed approach has been tested in three studies [1,2,15] using the same CO_2 emission formulation. The performance of the genetic algorithm developed within this study has outperformed the methods in a lot of instances. Moreover, the research has pointed out that prioritizing CO_2 emissions as a single objective of optimization is very important and yields better results compared to distance-based or multi-objective models.

2. Literature Review

Logistics are vital to the success of supply chains and are linked to overall organizational performance. Logistics aims to implement a series of decisions that include purchasing raw materials, parts, and components, moving and storing stocks, and moving goods from one place to another. Additionally, logistics plays an important role in the entire supply chain and contributes significantly to air pollution [24].

Logistics includes vital activities for the economic growth of an organization because it manages the flow of materials and information within and outside the organization, as well as reverse flow. Like many other industrial processes, logistics has been affected by the rise of Industry 4.0 technologies. However, Logistic 4.0. focuses mainly on economic benefits while ignoring environmental and social concerns [25]. The adoption of modern technologies is an important factor in the sustainability of many businesses today. Modern technologies also allow businesses to compete and improve their products and services [26]. In this context, the modern logistics industry has become an increasingly important part of the modern economy, which is booming on a global scale with the rapid development of the world economy and the advancement of science and technology [27].

Logistics has the potential to have a tremendous impact on both local and global trade by aiming to supply the right products, at the right place and at the right time. The efficiency of the logistics system largely determines the performance of a business in terms of cost, customer satisfaction, and profitability. Over the past two decades, there has been increasing interest in the environmental and social impacts of logistics, as well as its economic dimensions. Recent research on logistics has focused on the adoption of new technologies such as big data analytics, blockchain, artificial intelligence (AI), the Internet of Things (IoT), and additive manufacturing. In this context, the terms “green logistics” and “sustainable logistics” are used by organizations and academic researchers [28–31]. Other recent works have looked into the various aspects of the issues in sustainable logistics and vehicle routing problems. For example, Dişkaya and Dinçer came up with a mathematical model for the green vehicle routing problem with capacity constraints and heterogeneous fleets, with an aim to minimize transportation cost and carbon emissions [32]. Jamal et al. proposed the metaheuristic-based methods for traffic signal timing optimization with the aim of improving the performance of intersections and reducing traffic delays [33]. Fang et al. proposed a Voronoi neighborhood-based search heuristic to solve distance and capacity-constrained very large-scale vehicle routing problems [34]. Last but not least, Hao

et al. (2024) [35] conducted a review regarding the literature on vehicle routing problems and outlined the advances as well as challenges of the area of study.

Several studies in the literature have explored sustainable logistics and vehicle routing problems aimed at reducing CO_2 emissions. For instance, Napoli et al. [36] assessed the use of electric vehicles for freight distribution and their impact on CO_2 emissions reduction in Sicily. Nguyen et al. [37] examined the performance of heterogeneous vehicle fleets under traffic restrictions in the context of last-mile delivery and CO_2 emissions. Santos et al. [38] analyzed green reverse logistics and its impact on CO_2 emissions in vehicle routing problems with deliveries and pickups. Wang et al. [13] examined the performance of heterogeneous vehicle fleets under traffic restrictions. Liu et al. [9] analyzed the impact of CO_2 emissions in spatiotemporal-dependent vehicle routing problems. Dinu et al. [7] assessed the economic and cultural effects of EU energy policies on sustainable logistics. Lastly, Ghosh et al. [39] explored multi-objective optimization for waste management logistics, considering CO_2 emissions in a neutrosophic hesitant fuzzy environment. These studies address critical gaps in the sustainability literature and provide valuable insights into the environmental impact of logistics activities.

According to Chang and Qin [40], sustainable logistics, also known as green logistics, means planning, controlling, managing, and implementing the logistics system through advanced logistics technology and environmental management, aiming to reduce pollutant emissions. In other words, it is defined as "supply chain management practices and strategies that reduce the ecological and energy footprints of the distribution of goods focusing on material handling, waste management, packaging and transportation" [41]. It is also a set of actions aimed at minimizing the environmental consequences of logistics activities. These are the integrated management of logistics processes (production, storage, and waste transportation). It can also be expressed as a subsystem used to manage product flows from supplier to consumer with minimal impact on the environment [42].

Nowadays, market competition is becoming increasingly fierce due to diversified customer needs, stringent environmental requirements, and global competitors. One of the most important factors for companies to survive as well as be successful in today's competitive market is their logistics performance [29]. In global markets, businesses and supply chains should encourage actions that increase efficiency and sustainability, especially within logistics and transportation. The adequate management of goods transport has become a strategic issue on the agenda of businesses and administrations not only because it contributes to reducing costs and improving services, but also because it is a way to reduce resource consumption, CO_2 emissions, and traffic congestion [43]. Table 1 summarizes the comparison of methods used in green intermediate routing problems.

Table 1. Literature summary including the aim of reducing CO_2 emissions.

Method	Study	Application	Advantages	Limitations
Hybrid Ant Colony System	[1]	Green capacitated vehicle routing problem	Reduces CO_2 emissions, effective for large datasets	Computationally intensive, may require fine-tuning
Multi-objective Evolutionary Algorithm	[2]	Green vehicle routing problem	Balances travel distance and carbon emissions	Tradeoff may vary significantly with different parameters
Two-stage Ant Colony System	[3]	Multi-depot green vehicle routing problem	Minimizes carbon emissions, handles multiple depots	Complex implementation, may require extensive computational resources
Chance-Constrained Programming	[4]	Wet waste collection and transportation	Considers carbon emissions, handles uncertainty	Requires accurate estimation of waste generation rates

Table 1. *Cont.*

Method	Study	Application	Advantages	Limitations
Column Generation Approach	[5]	Emission-oriented vehicle routing on a multigraph	Detailed emission estimation, effective for heterogeneous vehicles	May be complex to implement, computationally demanding
Hybrid Whale Optimization Algorithm	[6]	Green vehicle routing problem	Minimizes distribution cost, competitive performance	May require parameter tuning, computationally intensive
Voronoi Neighborhood-Based Search Heuristic	[34]	Distance/capacity constrained very large vehicle routing problems	Better route planning, efficient for large datasets	May not handle dynamic changes in real-time effectively
Review of Vehicle Routing Problems	[35]	Overview of advancements and challenges in vehicle routing	Comprehensive review, identifies research gaps	Does not provide new empirical data or solutions

Table 1 presents some of the methods adopted to address green vehicle routing and vehicle routing problems, highlighting their advantages and disadvantages. For instance, the hybrid ant colony system is very effective in reducing CO_2 emissions and handling large datasets; however, it is computationally demanding and might require modifications [1]. Furthermore, the multi-objective evolutionary algorithm of Wu et al. provides a balanced solution for travel distance and carbon emissions; however, it is subject to significant variation in performance due to its variable parameters [2]. The two-stage ant colony system ensures a very efficient reduction in carbon emissions with multiple depots, but it requires intricate implementation and high computational resources [3]. The column generation methodology can give a comprehensive estimate of the emissions for different types of vehicles, but it is typified by its complexity and high computational demands [5]. Therefore, the method used will depend on the specific application and the available resources.

Increasing pressure from governments and customers for environmentally sustainable operations is encouraging organizations to collaborate to improve their environmental efforts. To do this, logistics providers as third parties need to translate buyers' requirements into offers related to environmentally sustainable logistics and follow them in interactions [44].

Sustainable Logistic 4.0 is the deployment of digitalization initiatives to accelerate the development of a connected, traceable, and flexible logistics system that can meet individualized customer requirements by adopting environmentally friendly practices. The application of digitalization technologies in logistics ensures that products are available at the right place, time, and quantity, thus reducing waste [45]. Thus, it also captures the ability to save resources, minimize waste, and increase operational efficiency by eliminating wasteful processes in the execution of operations [46].

Adopting sustainable logistics management practices initially limits an organization's financial capacity. However, it improves environmental reputation, which eventually translates into market share and increased sales. When the logistics operation is more efficient, the associated costs are reduced. Thus, the logistics industry can promote the improvement of industrial structure and increase economic quality and efficiency [47]. The increasing demand for sustainable business practices that lead to a protected environment has led to the widespread adoption of environmental or green practices in both developed and developing countries [48].

Studies on sustainability in a company generally focus on the corporate sustainability approach that covers the environmental, social, and economic needs of all stakeholders [49]. In sustainable logistics, not only economic and environmental dimensions but also social sustainability indicators, namely employment creation and working environments, are addressed holistically. Therefore, sustainable logistics aims to balance the socio-economic

performance of a logistics system with its eco-environmental soundness in managing system activities. This balance is embodied in decision-making, considering the interaction of different logistics functions such as network configuration, transportation, purchasing, demand allocation, and resource management. The optimization of a sustainable logistics system largely depends on the ability to balance the relationship between these three dimensions of sustainability [29].

Trends currently driving developments in urban logistics include, but are not limited to, urbanization and population growth, the growth of e-commerce, the desire for fast delivery, and climate change. More than 50 percent of the global population currently lives in cities, and this proportion is expected to rise to 66 percent by 2050 as more people move to cities every year. This makes cities even more dense in terms of logistics and difficult to serve. The increasing urban population has increased e-commerce. This has led to more package deliveries contributing to climate change due to greenhouse gas emissions. The conflicting expectations of today's consumers make the situation even more complex. On the one hand, online shopping consumers' expectations for fast, cost-effective, and personalized delivery are increasing. This requires more delivery vehicles on the roads. On the other hand, consumers expect logistics practices to be sustainable at less cost. This situation actually brings about a carbon footprint and less congested roads [50]. Based on all this, new methods are being investigated by both academia and industrial practitioners to improve the economic, environmental, and social sustainability of sustainable logistics activities [29].

3. Research Methodology

The flow chart followed when creating the research methodology is shown in the Figure 1.

The vehicle routing problem describes a class of problems in which routes, consisting of one or more depots, are established for a fleet of vehicles between geographically distributed cities or customers. In fact, it can be considered as a problem in which the traveling salesman problem is expanded for more than one vehicle [51]. Figure 2 shows the traveling salesman problem and the vehicle routing problem.

The circle that appears empty in Figure 2 shows the warehouse or the location to exit and return. Circles with blue inside indicate customers whose demands need to be met. In the traveling salesman problem, a vehicle must complete its tour without ever returning to a customer. For the problem in Figure 1, 9! = 362,880 different tours can be created. In the traveling salesman problem, the cost is attempted to be reduced by determining the shortest of these tours. In the vehicle routing problem, there are n number of vehicles to meet the needs of the customers closest to each other. In Figure 2, 3 vehicles meet the demands of 15 customers. In this problem, for each vehicle n! different tours can be created. In Figure 2, the number of different tours is 5! + 4! + 6! = 120 + 24 + 720 = 864. It is necessary to find the shortest tour for different tours. As the problem size increases, the number of these rounds increases exponentially and the problem becomes a non-polynomial hard (np-hard) problem.

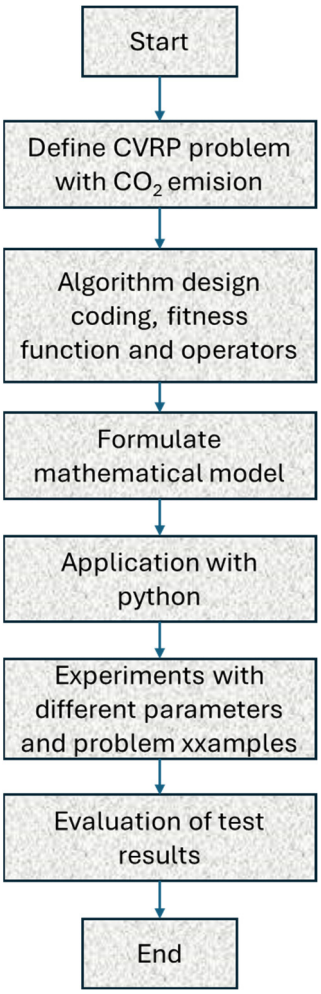


Figure 1. Flowchrat for Our Research Methodology.

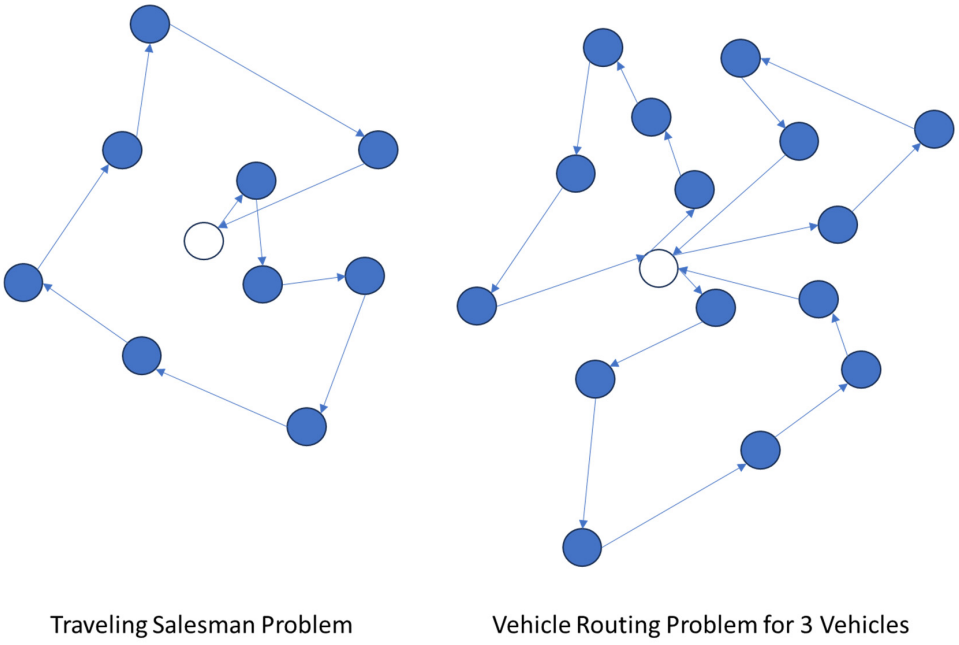


Figure 2. Traveling salesman problem and vehicle routing problem.

The capacity constrained vehicle routing problem (CVRP) is a variant of the classical vehicle routing problem (VRP), where the goal is to optimize the routes of a fleet of vehicles that depart from a depot, serve a set of customers, and return to the depot while adhering to

specific constraints. These constraints include the capacity of each vehicle, which limits the total demand it can serve, and the requirement that each customer is visited exactly once by a single vehicle. The CVRP was first defined by Dantzig and Ramser [52] as finding a set of minimum-cost vehicle routes starting and ending at the depot such that each customer is visited only by one vehicle and the total demand on any route does not exceed the capacity of the vehicle.

In their article published in 1959, Dantzig and Ramser defined the capacity-constrained vehicle routing problem as finding a set of minimum cost vehicle routes starting and ending at the depot such that each customer is visited only by one vehicle and the capacities of vehicles are not exceeded [52]. By changing this basic model, new models with multiple warehouses have been created [53]. A time and capacity-limited model was created [54]. New models have been created by changing the distance to minimize fuel cost [55] or CO_2 emissions. Even in some models, time and capacity are limited, reducing routing costs [56] or CO_2 emissions [57].

There are various models in the literature that try to minimize CO_2 emissions [16,46,53]. In most of these studies, CO_2 emissions are calculated and minimized by multiplying fuel consumption or distance by a fixed number (emission factor) [52,53]. While some studies focused on the effect of fuel consumption on speed and traffic congestion [53,57], some studies also took into account the effect of the amount of cargo carried [55,56].

3.1. Problem Definition

In this study, the formula suggested by Adiba et al. was used when routing. This formula allows emissions to be calculated based on both distance and vehicle load. For example, different emission factors are applied when vehicles are fully loaded versus empty, leading to more accurate estimations of emissions. Other models used in the literature primarily focus on distance, whereas Adiba et al.'s formula incorporates the weight of transported goods, providing a more holistic view of the environmental impact of logistics operations. CO_2 emissions are calculated for each vehicle based on the total distance and the total amount of load to be carried to the customer. When calculating CO_2 emissions, the effect of traffic congestion was neglected and the speed was assumed to be constant 80 km/h. The objective equation of the problem includes total CO_2 emissions. For this purpose, the capacity-constrained vehicle routing problem has been modified as follows.

Notation:

$G = (V, A)$ Directed graph where V represents nodes (warehouse and customers) and A represents arcs between nodes.

$V = \{v_0, v_1, v_2, v_3, v_4, \ldots, \}$. The set of nodes, where v_0 represents the warehouse and $v_1, v_2, \ldots, v_n$ represent the customers.

$q^k_{i,j}$: The amount of load (in tons) that vehicle k will carry from node i to node j, i,j ϵ V and k ϵ K.

$d_{i,}$: Euclidean distance between nodes (clients) *vi ve vj*.

$e^k_{i,j}$: The CO_2 emissions of vehicle k while traveling from customer i to customer j.

$K = \{k_1, k_2, \ldots, k_m\}$. The set of vehicles in the fleet.

Q: The maximum capacity of each vehicle in a homogeneous structure, $k \epsilon$ K

Decision variables:

$$x^k_{i,j} = \begin{cases} 1 \;\; If\ vehicle\ k\ moves\ from\ customer\ i\ to\ customer\ j \\ 0 \;\; otherwise \end{cases}$$

$$y^k_i = \begin{cases} 1 \;\; \text{If agent k serves customer i} \\ 0 \;\; otherwise \end{cases}$$

e_{el} = The CO_2 emissions of homogeneous vehicles when empty.

e_{fl} = The CO_2 emissions of homogeneous vehicles when fully loaded.

Purpose equation:

By the methodology and results of [1], the emissions to make a delivery with a given distance and flow can be calculated by the generic formula below:

$$\sum_{i\in V}\sum_{j\in V}\sum_{k\in K} d_{i,j}\cdot\left[\left(\frac{e_{fl}-e_{el}}{Q}\right)\cdot q_{i,j}^{k}+e_{el}\right]\cdot x_{i,j}^{k} \quad (1)$$

Restrictions:

$$\sum_{k\in K}\sum_{j\in V} x_{i,j}^{k}=1,\forall i\in V \quad (2)$$

$$\sum_{j\in V} x_{i,j}^{k}+\sum_{j\in V} x_{j,i}^{k}=1,\forall i\in V,k\in K \quad (3)$$

$$\sum_{k\in K}\sum_{j\in V} x_{0,j}^{k}=K \quad (4)$$

$$\sum_{j\in V} x_{0,j}^{k}=1,\forall k\in K \quad (5)$$

$$\sum_{j\in V} x_{j,n+1}^{k}=1,\forall k\in K \quad (6)$$

$$x_{i,j}^{k}=1\Rightarrow y_i-q_j=y_j,\forall i,j\in V,k\in K \quad (7)$$

$$y_0=Q,0\le y_i,\forall i\in V \quad (8)$$

$$\sum_{i\in V} q_i\sum_{j=0,j\neq i} x_{i,j}^{k}\le Q,\forall k\in\{1,\ldots,m\} \quad (9)$$

$$x_{i,j}^{k}\in\{0,1\},\forall i,j\in V,k\in K \quad (10)$$

Equation (11) shows the objective function that wants to minimize the distance emitted by vehicles. Constraint (12) ensures that 1 vehicle serves each route containing i,j pairs, constraint (13) ensures that the vehicle passes through a node 1 time along the route, constraint (14) ensures that each vehicle is in the warehouse at the beginning, constraint (15) ensures that each vehicle has only one, constraint (16) ensures that only one vehicle serves each customer, and constraint (17) shows the decrease in vehicle capacity due to the transition between nodes i and j. Constraint (18) ensures that the initial capacity of the vehicles is Q, constraint (19) ensures that the total customer demands on the route do not exceed the vehicle capacity, and constraint (20) ensures a 0 or 1.

The assumptions in this study are derived from the standardized scenarios used in CVRP Challenge competitions and CVRPLIB instances. These frameworks, widely recognized in the VRP research community, include fixed vehicle speeds, known customer demands, and static traffic conditions. While these assumptions do not fully replicate the complexity of real-world logistics, they are critical for establishing a controlled testing environment. This standardization allows for direct comparisons of algorithm performance across studies, providing a robust baseline for further development. There are simplifying assumptions. The first of these is the use of Euclidean distances as the crow flies between customer locations. In reality, these locations include mountains, streams, seas, etc. Distances may be longer due to crossing geographical structures. In addition, the slopes of the routes are the same, so the effect of road slopes on fuel consumption and CO_2 emissions is not taken into account. In addition, the speed factor that may have an impact on fuel consumption and CO_2 emissions is assumed to be fixed at 80 km/h.

In this study, the capacity constrained version (CVRP) of the vehicle routing problem (VRP) is discussed. The proposed method calculates CO_2 emissions based on both distance and vehicle load using the formula of Adiba et al. This approach provides more accurate emissions estimates than other models that focus only on distance. However, constant speed assumptions and the use of Euclidean distances, which ignore real-world complexities such as traffic congestion and variable speeds, can limit the method's accuracy. Al-Turki et al.'s work uses an NSGA-II-based optimization for intelligent traffic control [58]. It aims to reduce delays, fuel consumption, and emissions. This study presents a multi-objective optimization that takes into account various traffic performance metrics. However, its focus on urban intersections may result in it not directly addressing wider logistics networks. Neumann et al.'s [59] study examines the application of artificial intelligence (AI) to automate the simulation modeling of logistics systems. This study highlights the effects of AI-based predictions and optimization techniques on operational efficiency and sustainability. The proposed method similarly aims to minimize CO_2 emissions for sustainable logistics operations. However, the realism of the model can be increased by integrating dynamic traffic data and geographic information systems (GISs), as in the study of [59].

This method provides more accurate emissions estimates by taking distance and vehicle load into account. It contributes to sustainable logistics operations by aiming to minimize CO_2 emissions. It increases the efficiency of finding optimal solutions using a multi-process parallel genetic algorithm. It includes comprehensive constraints such as vehicle capacity, which ensures the practical applicability of the method. Assumptions such as constant speed and ignoring traffic congestion may prevent the method from fully reflecting real-world complexities. Using linear distances can cause a loss of accuracy in route planning, especially in areas with geographical obstacles. Assuming static traffic conditions that do not account for dynamic traffic changes can affect fuel consumption and emissions. A multi-process parallel genetic algorithm approach was used to solve the problem whose mathematical formulation was given above, which aims to minimize CO_2 emissions within the scope of sustainable logistics. The flow chart of our genetic algorithm is shown in Figure 3.

3.2. Coding

The first and most important stage of genetic algorithms is to code the problem correctly. There are different approaches to coding such as binary coding, real number coding, and permutation coding. In this study, permutation coding was used, consisting of an integer array representing the customers and an integer array representing the vehicles assigned to the customers. The genotype and phenotype of an individual are shown in Figure 4.

While genotype refers to the gene sequences in genetic coding in living things, phenotype determines the distinguishing features of the living thing, such as eye color and skin color. In our study, similarly, the chromosome structure shows gene alignments at two levels, customers and vehicles. The genotype shown on the left in Figure 4 is the customer array, [17, 20, 18, 15, 12, 16, 19, 21, 14, 13, 11, 4, 3, 8, 10, 9, 7, 5, 2, 1, 6], which is assigned to the vehicle array, [0, 0, 0, 0, 0, 1, 1, 1, 1, 2, 2, 2, 2, 2, 2, 3, 3, 3, 3, 3, 3]. When this coding is decoded, it represents Route 1 [17, 20, 18, 15, 12], Route 2 [16, 19, 21, 14], Route 3 [13, 11, 4, 3, 8, 10], and Route 4 [9, 7, 5, 2, 1, 6]. The phenotype, as shown on the right side of Figure 4, is the graphical representation of these routes. Each route begins and ends at the depot (indicated by the central node) and connects the customers assigned to the respective vehicle. The phenotype is critical for evaluating the quality of solutions. While genetic algorithms perform crossover and mutation operations on the genotype,

the phenotype determines the actual cost or feasibility of the solution. In this study, the phenotype is used to calculate the total distance traveled by all vehicles and evaluate the CO_2 emissions generated by these routes. The graphical representation in Figure 2 allows researchers to visually validate the efficiency of the routes, ensuring no unnecessary overlaps or excessive distances.

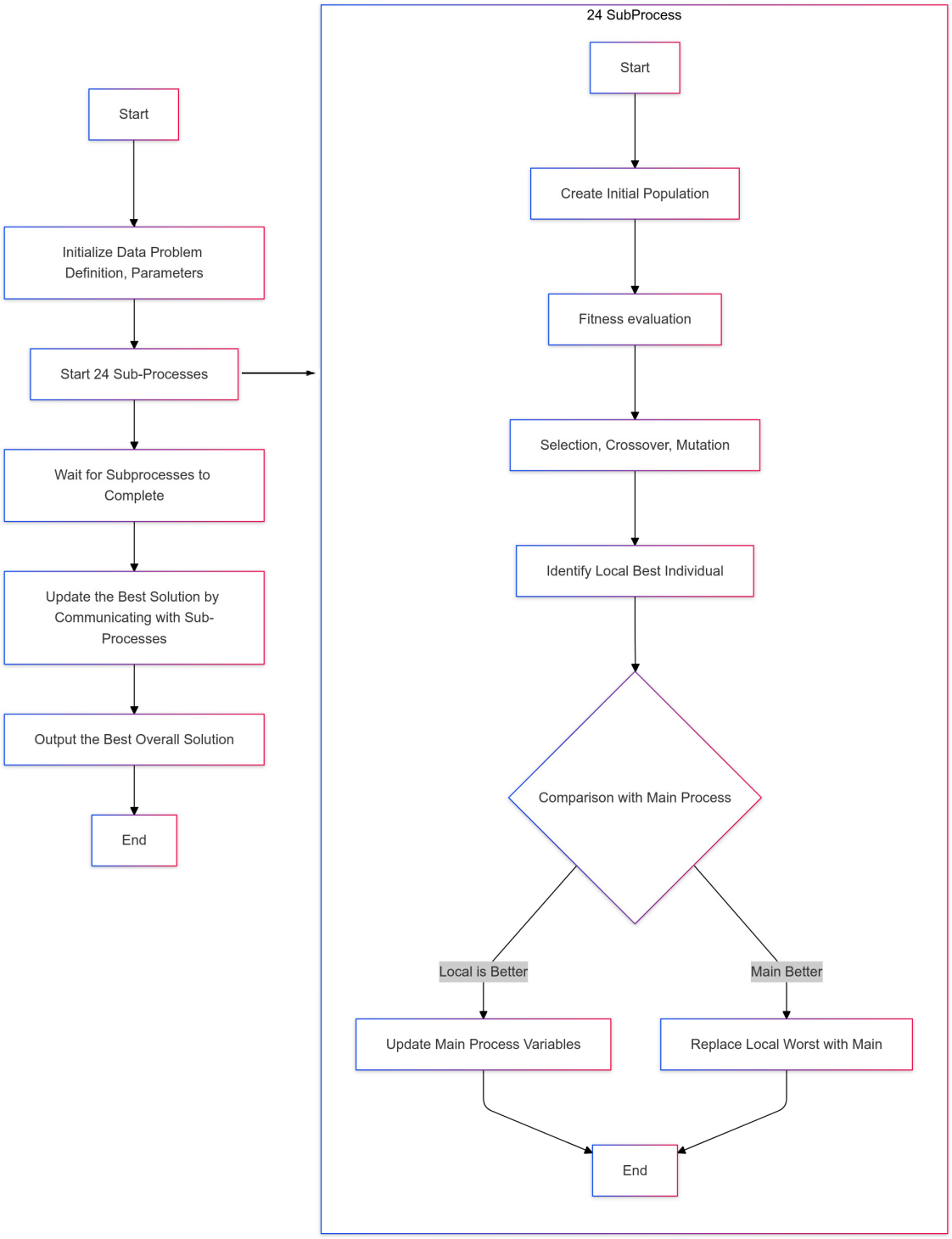


Figure 3. Flowchart for Our Genetic Algorithm.

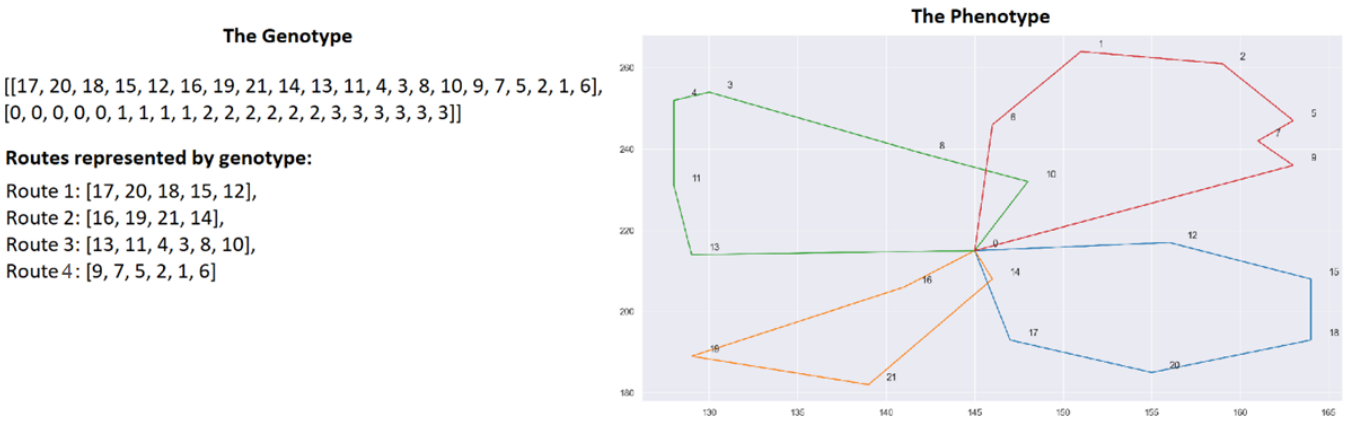


Figure 4. Genotype and phenotype of the optimal solution of sample E-n22-k4.

By integrating genotypes and phenotypes, the algorithm ensures that the solutions are both computationally feasible and practically implementable. This connection between the encoded (genotype) and decoded (phenotype) representations enables effective optimization tailored to the problem constraints.

3.3. Fitness Functions of Two Genetic Algorithm Models

For genetic algorithms to be effective, the fitness function must align with the specific objectives of the problem. In this study, three distinct genetic algorithm models were developed, each with a different fitness function aimed at solving the capacitated vehicle routing problem (CVRP) with a focus on sustainability. These fitness functions were designed to minimize distance, CO_2 emissions, or both.

1. Distance minimization model

The first genetic algorithm model focuses on minimizing the total distance traveled by all vehicles, which indirectly reduces fuel consumption and operational costs. The fitness function for this model is defined as follows:

$$Z_1 = \sum_{i \in V} \sum_{j \in V} \sum_{k \in K} d_{i,j} \cdot x_{i,j}^k \tag{11}$$

where

- $d_{i,j}$ represents the distance between nodes i and j;
- $x_{i,j}^k$ is a binary variable indicating whether vehicle k travels from i to j.

2. CO_2 emissions minimization model

The second genetic algorithm model targets minimizing CO_2 emissions generated during transportation. The fitness function for this model is given by the following:

$$Z_2 = \sum_{i \in V} \sum_{j \in V} \sum_{k \in K} d_{i,j} \cdot \left[\left(\frac{e_{fl} - e_{el}}{Q} \right) \cdot q_{i,j}^k + e_{el} \right] \cdot x_{i,j}^k \tag{12}$$

where e_{fl} = CO_2 emissions of homogeneous vehicles when fully loaded; e_{el} = CO_2 emissions of homogeneous vehicles when empty (gr/km); $q_{i,j}^k$ represents the load that vehicle k will carry from node i to node j; Q represents the capacity of each vehicle in a homogeneous structure; $d_{i,j}$ = Euclidean distance between nodes (clients) *vi ve vj*; and $x_{i,j}^k$ is a binary decision variable: 1 *if vehicle k travels from customer i to customer j*, and 0 otherwise.

3.4. Initial Population

While creating the initial population, a series of customer numbers was first created and randomly mixed. The array containing the vehicles was randomly generated in the range [0, number of vehicles − 1]. Apart from this, no special heuristic approach was used. In other words, it can be said that the initial population was created randomly. An experimental study was conducted to determine the optimal size of the population, and the two population sizes that gave the best results (130, 140) were selected for final evaluation. Details of this experimental study are explained in Section 4.1. Parameter Optimization.

3.5. Selection

During the evolution of the genetic algorithm, individuals that will be matched to undergo crossover and mutation are selected by various methods and added to the mating pool. Crossover and/or mutation operators are applied to the peers in this pool with probability determined by the crossover and mutation rates. If the selection method gives too high a frequency of selection to the best individuals, then the genetic algorithm runs the risk of premature convergence where the population is too homogeneous and the exploration of the solution space is stifled. Selection therefore needs to strike a balance between emphasizing high-quality individuals and maintaining diversity in the population in order to avoid local optima and encourage robust exploration. In this study, various selection algorithms in the literature were tested. These operators are uniform selection, rank selection, steady state selection, roulette wheel selection, lexicase selection, epsilon lexicase selection, automatic epsilon lexicase selection, tournament selection, random selection, double tournament selection, tournament DCD selection, and universal stochastic sampling. All selection operators were evaluated over ten independent runs, while keeping all other parameters of the algorithm constant. The test results are shown in Figure 5.

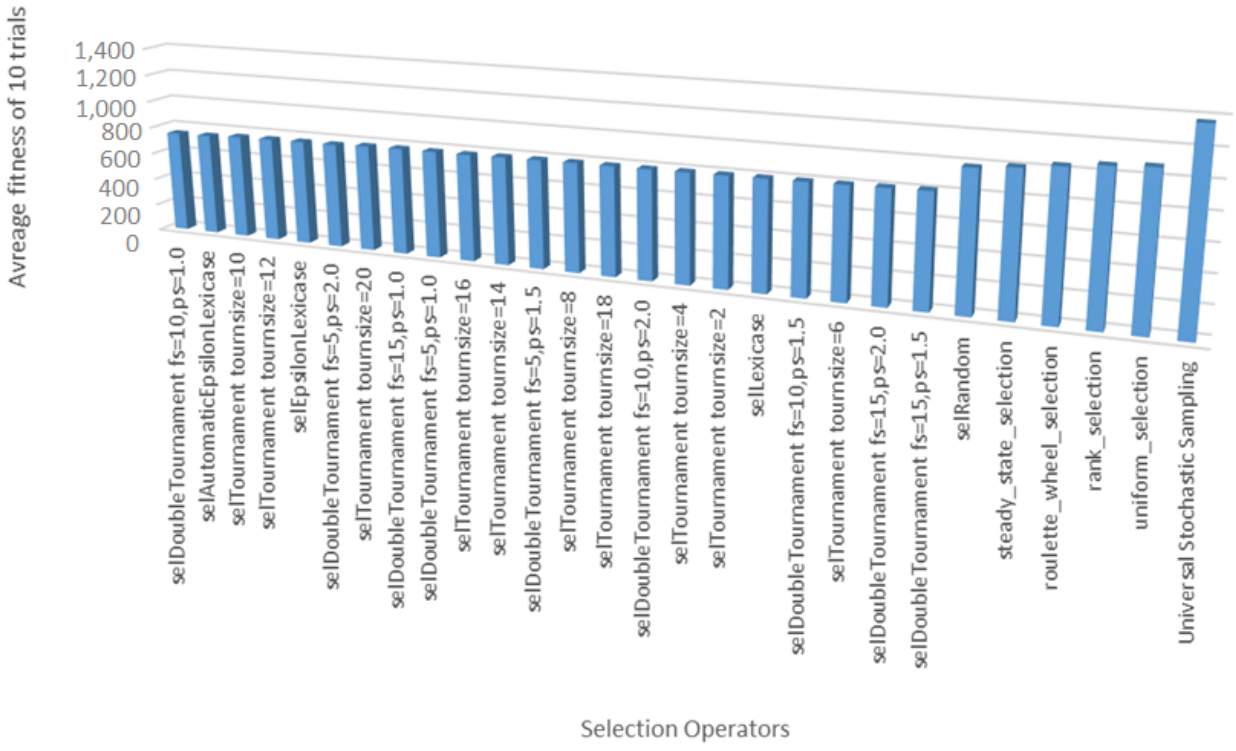


Figure 5. Performances of selection algorithms.

Of the numerous techniques discussed, double tournament selection and automatic epsilon lexicase selection were chosen for the final evaluation. The double tournament

selection was chosen, since it genuinely combines the exploration of diverse solutions with a focus on high-quality individuals—thanks to the alternation between fitness-based and diversity-based criteria—thereby keeping the diversity of the population while reaching better and better solution quality. The use of automatic epsilon lexicase selection has shown higher efficiency in discriminating niche solutions, which is very useful in the case of multi-objective optimization problems such as simultaneous CO_2 emissions and route distance reduction. Moreover, its ability to adaptively adjust thresholds helped to maintain a diverse set of promising solutions along the evolutionary path. The aforementioned selection techniques were chosen primarily because of their capacity to reconcile exploitation with exploration, thereby preventing the genetic algorithm from succumbing to premature convergence while simultaneously attaining steady enhancements in both CO_2 emissions and routing efficiency.

3.6. Crossover

We conducted a consideration of one-point crossover (1X), two-point crossover (2X), uniform crossover (UX), cycle crossover (CX), order-based crossover (OBX), order crossover 1 (OX1), order crossover 2 (OX2), order crossover 3 (OX3), position-based crossover (PBX), partially mapped crossover (PMX), linear order crossover (LOX), edge recombination crossover (ERX), enhanced edge recombination crossover (EERX), subtour chunks crossover (SCX), maximal preservation crossover (MPX), two-points center crossover (2PCX), two-points end crossover (2PEX), and two-points center and end crossover (2PECX). All crossover operators were tested in 10 independent runs, keeping constant the other parameters of the algorithm. Their ability to preserve diversity in the solutions, preserve important genetic material, and converge to better solutions was evaluated. As a result of this evaluation, PBX and 2PECX performed the best at exploration–exploitation balancing and hence, were kept for final evaluation. The working principle of the PBX crossover operator is shown in Figure 6.

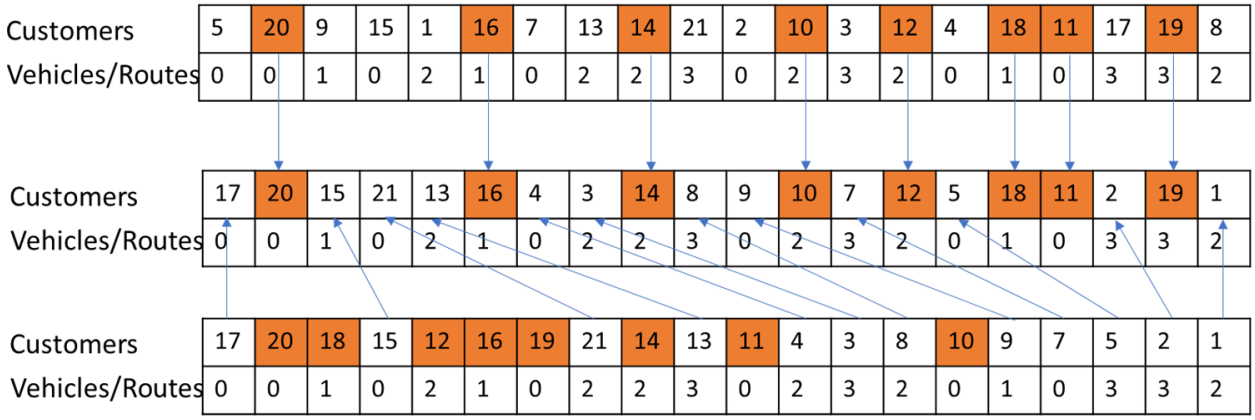


Figure 6. Working principle of position based crossover—PBX operator.

As can be seen in Figure 6, PBX involves transferring genes from the random loci of one parent to the child. The remaining genetic material is inherited from the other parent in the order in which it appears. This was selected because it can retain important features of the solution while, at the same time, introducing diversity among the offspring. The random choice of loci ensures that good features from both parents are preserved without compromising the integrity of the solution as a whole. The working principle of the 2PECX crossover operator is shown in Figure 7.

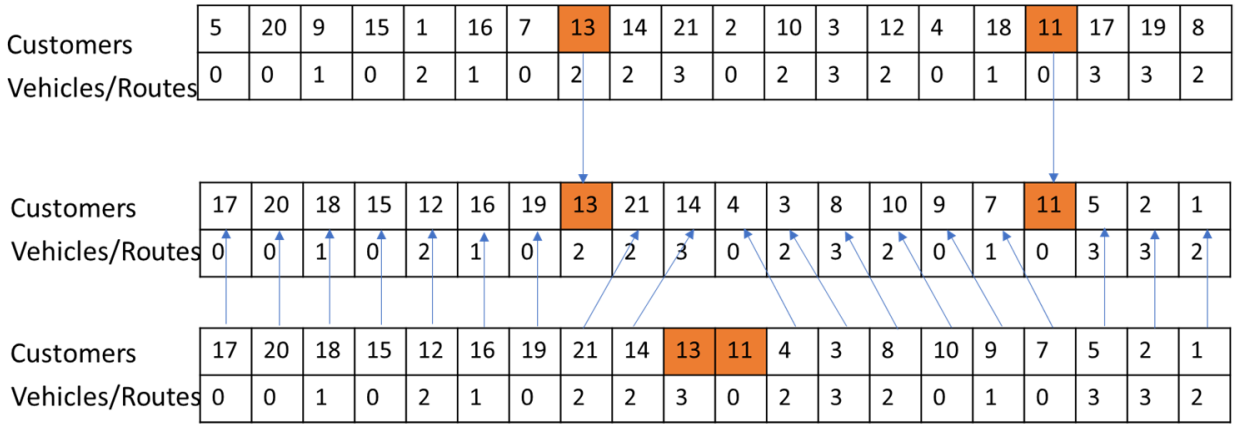


Figure 7. Working principle of two-points end/center crossover—2PECX operator.

As shown in Figure 7, 2PECX directly copies two randomly chosen genes from the center or end segments of the first parent to the child. The remaining genes are inherited from the second parent in sequence. This operator is selected because it can efficiently preserve both localized and global solution features, thus keeping a balance between intensifying the search around the high-quality solutions and exploring new areas of the solution space.

Those crossover operators under debate showed the best average performances either in solution quality or in computational efficiency, thus fitting best with the objectives concerning CO_2 reduction and the improvement of logistics routing. Their presence in the final analysis is thus of relevance to obtaining the right balance between exploration and convergence with the genetic algorithm.

3.7. Mutation

While determining the mutation operator, various mutation strategies commonly used in combinatorial coding were evaluated. These operators were modified to take into account the coding structure of the problem, composed of two strings, one representing customers and the other representing vehicles. In order to perform an exploration of the solution space that is well balanced, each string was assigned a 50% chance of being modified during mutation. The studied mutation operators are the two operations adjacent swap, 2OAS; the three operations adjacent swap, 3OAS; the two operations random swap, 2ORS; the three operations random swap, 3ORS; the inverse mutation, IM; the shift operation mutation, SOM; the center inverse mutation, CIM; the enhanced two operations random swap, E2ORS; and the shuffle mutation, SM. All operators were compared on 10 independent runs with all of the parameters set to the same in order to test their ability to maintain diversity and improve the solution quality.

Based on these analyses, the two operations random swap (2ORS) and the three operations random swap (3ORS) methods were chosen for final implementation. 2ORS: This operator exchanges two randomly selected genes in the sequence. Such operators introduce small changes that are important in the process of exploring the local neighborhood of a solution. 3ORS: This operator extends the previous one by swapping three randomly chosen genes. This introduces larger variations, thus helping to avoid local optima and encouraging a wider exploration of the solution space. These operators are chosen such that the diversification of the population becomes effective and viable.

Furthermore, the 2ORS and 3ORS operators were modified in order to solve an issue that was observed during testing. Most mutation operators are capable of modifying the initial number of vehicles assigned to a solution, and therefore potentially limiting the algorithm's capability of finding optimal solutions having different vehicle distributions.

To mitigate this drawback, the behavior of the 2ORS and 3ORS operators was modified as follows: With 50% probability, the 2ORS mutation is applied to the customer sequence, changing the order of customers or reassigning them to different routes. With the remaining 50% probability, a randomly chosen gene in the vehicle string is given a new value from the range.

This approach allows for the exploration of both route configurations and vehicle distributions. To maintain the feasibility of the solution, capacity constraints are checked and enforced during the mutation process. As shown in Figure 8, this method generates a variety of constraint-compliant solutions that improve the algorithm's ability to achieve high-quality results.

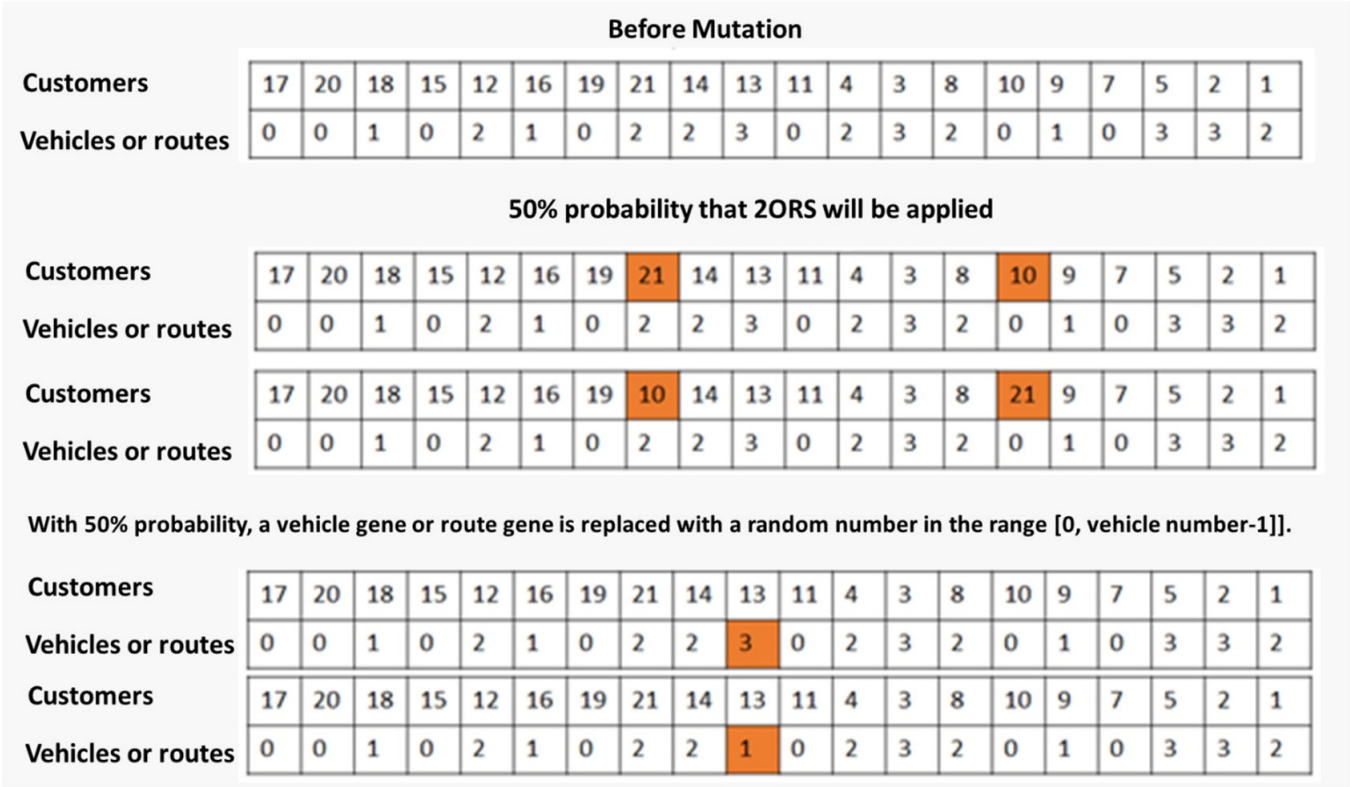


Figure 8. How the mutation operator works.

The use of modified 2ORS and 3ORS mutation operators significantly increases the ability of the genetic algorithm to efficiently explore the solution space while ensuring that all solutions remain feasible under the constraints of the problem. Their contributions are critical to minimizing CO_2 emissions and achieving robust optimization performance.

The mutation operator randomly changes two customers in the customer sequence with a 50% probability, changing the order of these customers on the route or causing the route to which the customers are assigned to change. This process creates a solution that satisfies the basic constraints. Additionally, the mutation operator provides a check to ensure that the capacity constraint is not exceeded. With a 50% probability, the value in the tool string of a randomly selected gene is replaced with a value in the range [0, vehicle number $-$ 1]. In this case, the customer represented by that gene is included in a new route. While this process is performed, the capacity limit is checked. As a result, the mutation operator always produces feasible solutions to the constraints.

3.8. Elitism

The main problem with genetic algorithms is the possibility of losing the best solution during the crossover and mutation stages. In order to avoid this, an elitism strategy is followed, whereby the best individual in each generation is preserved for subsequent iterations. Before the application of crossover and mutation, the best individual is determined and stored in a different memory structure. After the creation of the new population, this stored individual replaces the worst performing solution in the updated population. In this way, the algorithm ensures that at least the optimal solution from the previous generation is preserved, and there will be no degradation in solution quality due to genetic operations. In the process, this approach protects high-quality solutions from destruction, and hastens convergence since there is a guarantee that at each generation, the best candidate is carried forward from generation to generation. The elitism mechanism therefore acts to balance space exploration with the retention of valuable genetic material.

4. Findings

4.1. Parameter Optimization

In genetic algorithms, parameter settings such as population size, crossover rate, mutation rate, and the number of generations significantly affect the quality of solutions and computational efficiency. Population size directly influences the diversity of the solutions obtained from evolutionary algorithms. A larger population generally ensures more diversity in solutions, which enhances the algorithm's ability to explore the solution space. However, it is usually at the expense of increased computational time. A smaller population may converge faster but risks premature convergence to suboptimal solutions because of a lack of diversity. Within genetic algorithms, diversity in populations is crucial for preventing convergence to local optima. There is a widely recognized trade-off between the exploration of new areas within the solution space and an exploitation of the already retrieved solutions. Larger populations strengthen exploration but lead to increased computational cost. The mutation rate determines how often individuals randomly change. Higher values of mutation rates increase exploration by introducing further variations into the population, therefore helping to escape from a possible local optimum. However, excessive mutation can also disrupt the convergence process and lead to instability. A low mutation rate favors exploitation but risks stagnation. The mutation rate controls the possibility of the algorithm maintaining diversity in later stages. Works in evolutionary dynamics show that there is an optimal mutation rate for a particular problem, which is often adaptive in nature. For example, high mutation is beneficial in early stages to escape local optima, but in later stages, it can destabilize convergence toward a global optimum. The crossover rate sets the probability of two parent solutions crossing over to generate offspring. A high crossover rate promotes exploitation by recombining high-quality solutions; a low rate may reduce the algorithm's ability to refine solutions effectively. If it is too aggressive, however, crossover may prematurely eliminate diversity. The crossover mechanism naturally draws from the principle of building blocks (schema theory) where combining fit substrings from different parents produces better solutions. The success of crossover relies on a balance between the intensification of the search around good solutions and the maintenance of diversity. Selection pressure is the factor which indicates how strongly better solutions are favored when reproduction takes place. High selection pressure ensures that fitter solutions dominate the population quickly, accelerating convergence. However, this can reduce diversity and lead to premature convergence. Low selection pressure maintains diversity but slows down convergence. High selection pressure aligns with exploitation, focusing on refining high-quality solutions. However, theoretical studies in evolutionary computation claim that high selection pressure already violates

the principle of the survival of the fittest: the algorithm may lose valuable genetic material present in less fit solutions. The stopping criterion defines the point at which the algorithm stops looking for an optimal solution. Premature stopping might yield a suboptimal solution, whereas too much leniency may simply waste computational resources. Common stopping criteria are a fixed number of generations, convergence thresholds, and a lack of improvement over a predefined duration. The stopping criterion ensures computational efficiency while there is a balance in solution quality. Research on optimization suggests that adaptive stopping criteria, based on convergence rates or fitness plateaus, are more selective than fixed thresholds, especially for complex, nonlinear problems.

Proper parameter tuning ensures a balance between exploration (diversity in the solution space) and exploitation (the refinement of promising solutions). The goal of this study was to identify optimal parameter settings for the proposed models to achieve superior performance in solving the capacitated vehicle routing problem (CVRP) while minimizing CO_2 emissions. The experimental study was carried out on the A-n32-k5 instance. This CVRP instance consists of five vehicles serving 32 customers. For this reason, the algorithm was run 10 times for different levels of these three parameters, and the two values that gave the best results for each parameter (ÇO = 40% and 50%; MO = 80% and 90%; and population size = 150 and 200) were determined for use in the final evaluation. Additionally, the two best values were determined for crossover, mutation, and selection operators. A total of 64 experiments are required to test two different values of six parameters ($2^6 = 64$). Therefore, the following process was carried out.

The genetic algorithm has been transformed into a multi-process structure. The main process runs 64 different genetic algorithm processes. The global best solution and global best fit variables are defined in the main process. Each process compares the best solution it finds with the global best fit. If the process finds a better solution than the global best fit, it updates the global best fit and global best solution values with its own solution. If the process finds a solution worse than the global best fit, it updates its worst individual with the global best fit and global best solution values. In this way, the software that determines the performances of 64 different genetic algorithms for 1000 iterations and 10 repetitions is designed to run the main algorithm by determining the best parameter combination itself.

During the algorithmic determination of optimum parameters, values for different population sizes, crossover rates, and mutation rates were found and shown graphically in Figure 9a–b.

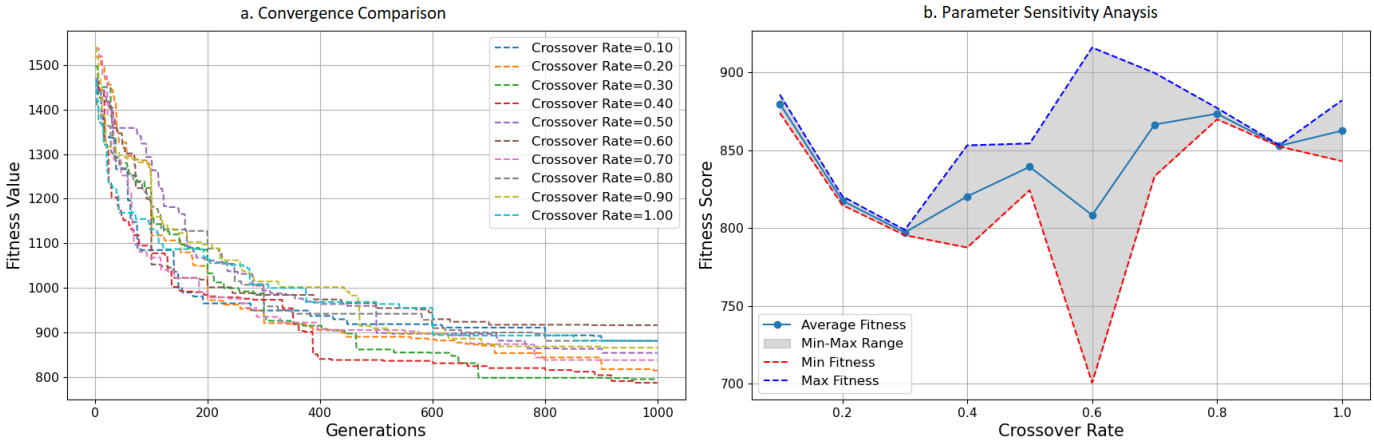


Figure 9. Effect of crossover rate on algorithm performance.

Figure 9a,b are complementary in the sense that they analyze the performance of the genetic algorithm from different viewpoints. Figure 9a shows how fitness values evolve over generations for different crossover rates, while Figure 9b focuses on the overall effect

of these rates on fitness scores. Figure 9a shows the effect of crossover rates on the solution quality of the genetic algorithm. The crossover rates 0.3 and 0.4 converge fast to low fitness values, which implies that the solutions generated with such crossover rates have better solution quality, since this is enhancing the effectiveness of the algorithm. Higher crossover rates, for example, 0.9 and 1.0, converge very slowly, suggesting that solution requires longer periods of evolution. Figure 9b summarizes crossover rates' influence on fitness scores, including the average, minimum, and maximum values of fitness. The best fitness scores are found at crossover rates of 0.3 and 0.4, which also demonstrate narrower min–max ranges, indicating higher stability. In contrast, the marked deterioration in performance for the crossover rate 0.6 shows that badly chosen parameter values may be quite harmful to the performance of an algorithm. Taken together, these figures highlight the critical role that crossover rates play in determining the performance of the genetic algorithm. Not only does a rate of 0.3 give a better result, but it also guarantees greater stability, which implies the importance of the sensitivity analysis of parameters in optimizing the performance of the algorithm.

The following figures represent the performance dynamics of the genetic algorithm when the mutation and crossover rates are varied. Figure 10a gives the analysis of the trend of convergence at different mutation rates. The results here show that higher mutation rates (for example, 0.7 and 0.9) basically result in slower convergence, evidenced by the delayed stabilization of values of fitness. In contrast, moderate mutation rates—say, 0.1 and 0.2—achieve a good trade-off between exploration and exploitation, thus converging faster. Very low mutation rates may cause the algorithm to converge prematurely into local optima.

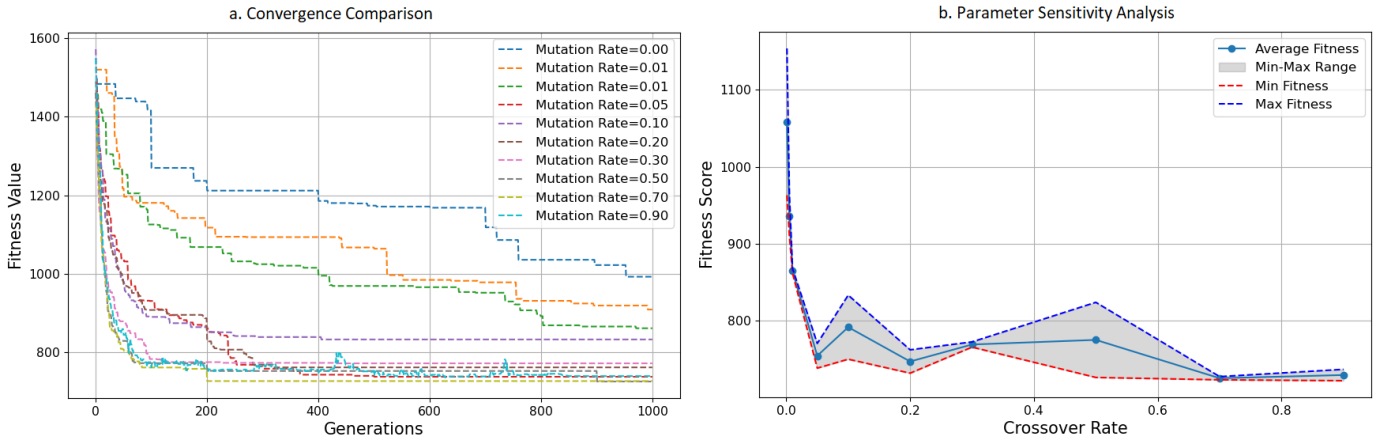


Figure 10. Effect of mutation rate on algorithm performance.

Figure 10b illustrates the sensitivity of the algorithm with regard to variations in crossover rate. The average fitness score peaks at a low crossover rate, approximately 0.1, and becomes less consistent, as seen from the increasing min–max range with growing rates. That would suggest that lower crossover rates support stable performance, while higher rates might introduce too much randomness, reducing solution quality. These results show how important it is to find good mutation and crossover parameters for the best performance of the algorithm.

The size of the initial population is a critical parameter in genetic algorithms (GAs) as it directly influences both the solution quality and computational efficiency. To determine an appropriate population size for the proposed models, an experimental study was conducted. The objective of the experimental study was to identify the population size that provides a balance between solution quality (ensuring the algorithm explores a diverse solution space) and computational efficiency (minimizing runtime while maintaining per-

formance). The tested population sizes ranged from small (e.g., 10 individuals) to large (e.g., 200 individuals) in increments of 10.

The figures illustrate the influence of population size on the performance of the genetic algorithm. In the convergence trends shown in Figure 11a, it is evident that smaller population sizes, such as 10 or 30, demonstrate faster convergence during the initial generations. However, these smaller populations are more prone to premature convergence, as they explore a narrower solution space. On the other hand, larger population sizes, such as 150 or 200, exhibit slower initial convergence but maintain diversity for a longer duration, potentially leading to an improved exploration of the solution space.

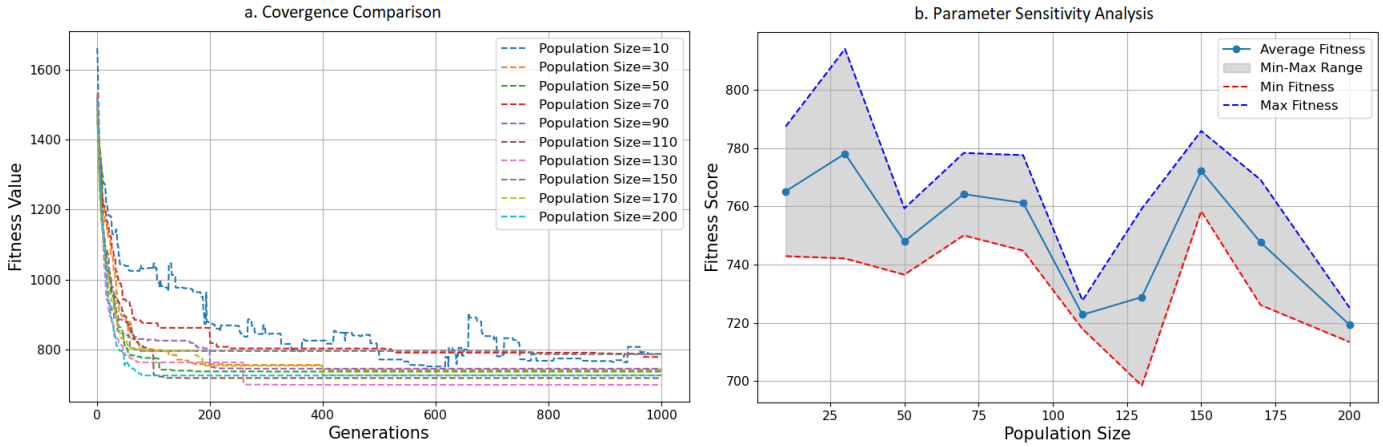


Figure 11. Effect of population size on algorithm performance.

Figure 11b provides a sensitivity analysis of population size with respect to fitness score. The average fitness score shows variability across different population sizes. Moderate population sizes, such as 50 and 150, achieve relatively higher average fitness, while both very small and very large population sizes exhibit lower average performance. The min–max fitness range highlights the variability in solution quality, which is wider for smaller population sizes, indicating inconsistent performance. In contrast, larger populations tend to stabilize the fitness range but may incur higher computational costs. These results emphasize the need for balancing population size to optimize both computational efficiency and solution quality.

4.2. Parallelization and Multi-Process Programming

The system architecture illustrated in Figure 9 showcases a parallel elitist genetic algorithm framework designed for CO_2 emission optimization. The workflow is coordinated by the main process, which manages task distribution, global solution evaluation, and the iterative refinement of the global best solution. Tasks are distributed via the task queue to multiple child processes, each running an independent genetic algorithm to evolve its population across generations. These child processes perform localized optimization and continuously compare their local best solutions to the global best solution stored in the main process.

In each generation, a child process evaluates whether its local best solution outperforms the global best solution in terms of fitness. If the local best is superior, the child process updates the global best solution in the main process. Conversely, if the local best does not surpass the global best, the child process incorporates the global best solution into its population by replacing its worst-performing individual. This mechanism ensures that all processes are influenced by the globally optimized solution, fostering convergence while maintaining population diversity.

The results queue serves as a centralized repository for exchanging solutions, facilitating seamless communication between the child processes and the main process. Once all child processes complete their tasks, the results writer consolidates the final optimized solutions for reporting and analysis.

This approach leverages Python 3.9.18's multiprocessing module to achieve efficient parallel execution, significantly reducing computation time compared to sequential algorithms. By allowing each child process to evolve independently yet remain influenced by the global optimization objective, the system ensures a balance between exploration and exploitation. The integration of the global-best-based update mechanism, combined with elitism, accelerates convergence to high-quality solutions while preventing stagnation in suboptimal regions of the solution space.

This workflow, as depicted in Figure 12, highlights the robustness and scalability of the proposed framework, making it particularly suitable for large-scale optimization problems in sustainability-focused domains, such as reducing CO_2 emissions in manufacturing and logistics operations.

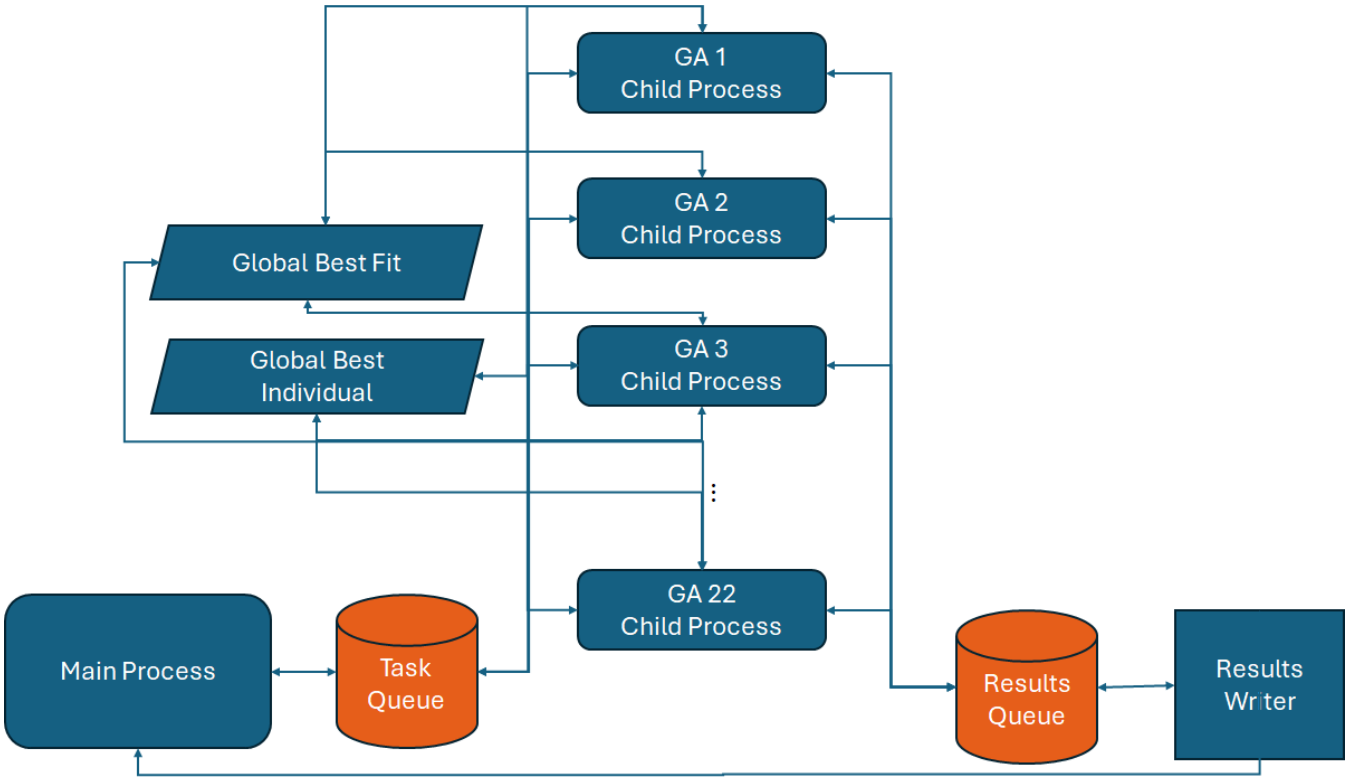


Figure 12. Multi-process parallel genetic algorithms.

It should be taken into account that each process created in multi-process programming will compete in using the computer's resources, and therefore, the completion times of the processes will increase. To determine the appropriate number of processes, processes with a population size of 130, crossover rate of 40%, mutation rate of 80%, selection operator double tournament, crossover operator PBX, and mutation operator 2ORS were used. The results obtained according to different process numbers are shown in Figure 13.

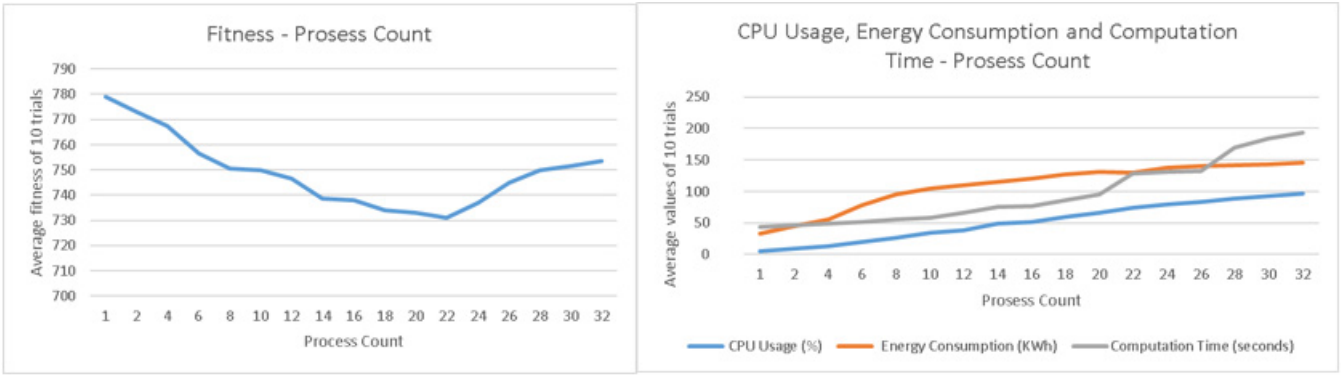


Figure 13. The effect of process number on fitness, CPU usage, energy consumption, and computation time.

Figure 13 shows that as the number of processes increases, better results are obtained. However, when the number of processes is more than 22, it is seen that the obtained fitness values begin to increase, although CPU usage, processing time, and energy consumption continue to increase. Therefore, it was decided that the most appropriate number of processes was 22. Since the CPU used is the 13th Gen Intel(R) Core(TM) i9-13900K 3.00 GHz, it has 24 physical and 32 logical cores.

5. Performance Comparison of Our Algorithm with Other Algorithms

To test the performance of the developed genetic algorithm, examples from three studies using the CO_2 emission formula suggested by Adiba et al., shown in Equation (11), were used. In order to make comparisons, capacity-constrained vehicle routing problem examples in these studies were used. The comparison results of the algorithms' performances on small samples are shown in Table 2. Table 2 also includes the best known solution (BKS) values.

Table 2. Performance comparison of our algorithm with others using the same CO_2 emissions formulation on small-scale problem instances.

Instance	Shortest Distance (km)				Minimum CO_2 (gr)				
	Hybrid Artificial Bee Colony Optimization (efl = 0.390; eel = 0.296) [19]	Genetic Algorithm (efl = 0.390; eel = 0.296) [2]	Our Parallel Genetic Algorithm (efl = 0.390; eel = 0.296)	Hybrid Ant Colony Optimization. (efl = 1.096; eel = 0.772) [1]	Hybrid Artificial Bee Colony Optimization. (efl = 0.390; eel = 0.296) [19]	Genetic Algorithm. (efl = 0.390; eel = 0.296) [2]	Our Parallel Genetic Algorithm. (efl = 1,096; eel = 0.772)	Our Parallel Genetic Algorithm. (efl = 0.390; eel = 0.296)	Best Known Solution (efl = 0.390, eel = 0.296
A-n32-k5	685.38	687.39	683.09	-	682.8	685.05	-	681.85	678.65
A-n33-k5	579.98	-	570.44	-	579.20	-	-	573.74	564.05
A-n34-k5	685.79	-	684.28	-	683.11	-	-	671.27	662.79
A-n36-k5	704.01	771.00	698.27	-	702.36	768.23	-	695.26	679.84
A-n37-k5	583.00	619.79	579.14	-	580.69	611.98	-	608.06	563.75
A-n38-k5	642.89	649.04	639.01	-	642.65	648.54	-	642.22	635.01
A-n39-k5	730.20	-	710.92	-	727.33	-	-	720.04	703.53
A-n46-k7	810.73	-	803.11	-	803.73	-	-	841.00	779.55
A-n48-k7	944.29	1.007.83	943.41	-	944.01	1005.91	-	940.13	924.73
B-n31-k5	585.19	-	571.09	-	585.19	-	-	582.59	559.43
B-n35-k5	839.60	858.90	830.87	-	838.91	858.90	-	835.80	808.02
B-n38-k6	698.50	-	682.68	-	698.50	-	-	695.28	679.84
B-n39-k5	484.67	-	477.67	-	480.88	-	-	503.72	453.52

Table 2. *Cont.*

Instance	Shortest Distance (km)				Minimum CO_2 (gr)				
	Hybrid Artificial Bee Colony Optimization (efl = 0.390; eel = 0.296) [19]	Genetic Algorithm (efl = 0.390; eel = 0.296) [2]	Our Parallel Genetic Algorithm (efl = 0.390; eel = 0.296)	Hybrid Ant Colony Optimization. (efl = 1.096; eel = 0.772) [1]	Hybrid Artificial Bee Colony Optimization. (efl = 0.390; eel = 0.296) [19]	Genetic Algorithm. (efl = 0.390; eel = 0.296) [2]	Our Parallel Genetic Algorithm. (efl = 1,096; eel = 0.772)	Our Parallel Genetic Algorithm. (efl = 0.390; eel = 0.296)	Best Known Solution (efl = 0.390, eel = 0.296
B-n43-k6	652.50	-	643.47	-	651.52	-	-	664.17	619.15
B-n44-k7	805.66	842.00	802.05	-	804.94	839.13	-	805.02	754.78
E-n22-k4	325.63	-	327.03	-	325.63	-	-	321.01	307.98
E-n23-k3	481.50	-	481.50	-	479.02	-	-	488.84	465.29
E-n33-k4	735.69	-	734.26	-	734.88	-	-	733.15	689.46

In the manuscript, we have clarified that the emissions presented in Table 1 represent the total CO_2 emissions over the entire travel duration, not emissions per kilometer. This distinction has been made clear to avoid any confusion regarding the units.

Regarding the comment about the units of CO_2 emissions in the mathematical formulation, we have provided an explanation of the units involved. Specifically, the CO_2 emissions, denoted by efl and eel, are expressed in grams per kilometer (g/km). Additionally, the variables representing the distance (d_{ij}) are in kilometers (km), and the vehicle capacity (Q) and the load carried by the vehicle (q_{ij}^{k}) are in tons. Therefore, the resulting emissions from the formulation are calculated in grams (g), reflecting the total emissions for the journey.

Table 2 gives a comparison between Shuzhu's algorithm, the genetic algorithm by Wu et al. [2], our parallel genetic algorithm, and the best known solution for each test instance. As a whole, our parallel genetic algorithm is competitive in effectiveness and often outperforms the other algorithms concerning both the reduction of CO_2 emissions and smaller distances.

For A-n32-k5, our proposed algorithm yields 681.85 g of CO_2 emissions, which is better than Shuzhu's algorithm with 682.58 g and the algorithm of Wu et al. with 685.05 g, although a little bit higher than the best known solution with 678.65 g. In the same way, for A-n33-k5, our algorithm returns 573.74 g of CO_2 emissions, which is better than Shuzhu's algorithm with 579.20 g, while still a little bit higher than the best known solution with 564.05 g. For instance, in A-n36-k5, our proposed algorithm improves significantly with CO_2 emissions of 695.26 g over the performance of Shuzhu's algorithm with 702.36 g and the algorithm proposed by Wu et al. with 768.23 g, but still worse than the best known solution at 679.84 g. These results show that our algorithm is efficient in many instances compared to some reported in the literature. For problems whose objective is to reduce the distance, there exist problems, including E-n22-k4, for which our considered algorithm yields higher CO_2 emissions (327.03 g) when compared against Shuzhu's results (325.63 g), and the BKS (307.98 g) solution. For instance, in the case of E-n23-k3, our algorithm finds a set of routes which gives emissions of 488.84 g, considerably higher than Shuzhu's record (479.02 g) and even higher than the BKS (465.29 g). However, in the case of minimizing CO_2 emissions, our algorithm shows better results: in the case of E-n22-k4 with a minimum CO_2 goal, our algorithm outperforms Shuzhu's and coincides with the BKS. It evidences the flexibility and efficiency of our algorithm to satisfy the given optimization goal.

In summary, while our algorithm demonstrates good performance in minimizing both distances and CO_2 emissions, it shows particularly impressive performance in the minimization of CO_2 emissions, often significantly outperforming the algorithms designed by Shuzhu and Wu et al. in many cases. These outcomes demonstrate its effectiveness in

dealing with vehicle routing problems with capacity constraints where sustainability is a major concern. The comparison results of the algorithms' performances on large samples are shown in Table 3.

Table 3. Performance comparison of our algorithm with others using the same CO_2 emission formulation on large-scale problem instances.

Instance	Shortest Distance(km)			Minimum CO_2 (gr)						
	[19] (efl = 0.390; eel = 0.296)	[2] (efl = 0.390; eel = 0.296)	Our Parallel Genetic Algorithm (efl = 0.390; eel = 0.296)	[1] (efl = 1.096; eel = 0.772)	[19] (efl = 0.390; eel = 0.296)	[2] (efl = 0.390; eel = 0.296)	Our Parallel Genetic Algorithm. (efl = 1.096; eel = 0.772)	Our Parallel Genetic Algorithm. (efl = 0.390; eel = 0.296)	Best Known Solution (efl = 0.390; eel = 0.296)	Best Known Solution (efl = 1.096; eel = 0.772)
E-n51-k5	464.17	483.0	481.09	524.61	463.19	483.80	510.27	469.12	437.63	-
E-n76-k10	-	-	-	843.54	-	-	826.58	-	-	772.12
E-n101-k8	-	784.84	795.23	835.28	-	784.84	812.54	788.01	678.06	749.67
M-n101-k10	-	-	-	819.56	-	-	816.18	-	-	755.12
M-n121-k7	-	-	-	1090.01	-	-	1080.02	-	-	969.00
M-n151-k12	-	-	-	1054.92	-	-	1055.62	-	-	945.16
M-n200-k17	-	-	-	1377.71	-	-	1372.89	-		1181.96

Table 3 shows that our parallel genetic algorithm performs competitively in large-scale problem instances compared to Shuzhu's algorithm, Wu et al.'s genetic algorithm, and Adiba et al.'s hybrid ant colony optimization, while also producing results close to the best known solutions. For example, in the E-n51-k5 instance, our algorithm (469.12 g) performs similarly to Shuzhu's algorithm (463.19 g) and better than Wu et al.'s genetic algorithm (483.80 g) and Adiba et al.'s algorithm (524.61 g), while being close to the best known solution (437.63 g). In the E-n76-k10 instance, our algorithm (826.58 g) performs better than Adiba et al.'s algorithm (843.54 g), though comparisons with Shuzhu's and Wu et al.'s algorithms are unavailable, and it is reasonably close to the best known solution (772.12 g).

In the E-n101-k8 instance, our algorithm (793.01 g) outperforms Adiba et al.'s algorithm (835.28 g) and produces results comparable to Wu et al.'s genetic algorithm (784.84 g) and Shuzhu's algorithm (784.84 g). It also approaches the best known solution (749.67 g), showcasing its effectiveness. Similarly, in the M-n101-k10 and M-n121-k7 instances, our algorithm (816.18 g and 1080.02 g, respectively) produces results close to the best known solutions (755.12 g and 969.00 g, respectively). In the E-n101-k8 instance, our algorithm (793.01 g) outperforms Adiba et al.'s algorithm (835.28 g) and produces results comparable to Wu et al.'s genetic algorithm (784.84 g) and Shuzhu's algorithm (784.84 g). It also approaches the best known solution (749.67 g), showcasing its effectiveness. Similarly, in the M-n101-k10 and M-n121-k7 instances, our algorithm (816.18 g and 1080.02 g, respectively) produces results close to the best known solutions (755.12 g and 969.00 g, respectively).

In larger problem instances, such as M-n151-k12 and M-n200-k17, our algorithm still generates competitive results. For instance, in M-n151-k12, the solution of Adiba et al. outperforms ours, with 1055.62 g against 1054.92 g, while ours is closer to the best known solution, which is 945.16 g. For M-n200-k17, the performance achieved by our algorithm was 1372.89 g, higher than the best known solution of 1181.96 g but still a very good solution for such a challenging instance. To conclude, our algorithm is well-performing in most large-scale instances and mostly keeps results close to the best known solutions, aside from a few exceptions like in M-n151-k12, where the results for Adiba et al. are marginally better; our approach has a really great promise in the case of larger problem sets. Also, it is clear that by focusing on minimizing the shortest distance, the ability to reduce CO_2 emissions may be slightly reduced with our algorithm, which offers several future development potentials at specific points.

In general, Table 1 illustrates the high efficiency of our algorithm for small-scale problem instances and its ability to optimize CO_2 emissions better in most cases than other heuristics. The findings in Table 1 show that our algorithm produces a solution close to the best known solutions in many instances; thus, it is rather accurate and reliable. In this case, our algorithm performs well in big sets, especially with the CO_2 emission minimization goal, while staying near the best known solutions. The results confirm the validity of our proposed parallel genetic algorithm for reducing the level of CO_2 emissions when tested on instances of varying difficulty: scalability and effectiveness can be seen from Tables 1 and 2, respectively. For example, in the case of instance E-n51-k5, our algorithm has found results close to the best known solution while outperforming many of the previously known algorithms. In larger-scale instances such as M-n121-k7 and M-n151-k12, our algorithm performs competitively by returning results within a reasonable distance from the best known solutions on each problem instance. Our findings underpin the efficacy of our algorithm to yield sound results in challenging and big scenarios.

Overall, Tables 1 and 2 represent a trade-off between minimum CO_2 emissions and shortest distance. Although the shortest distance is sacrificed sometimes, our algorithm performs very well in minimizing CO_2 emissions on small- and large-scale problem instances. Furthermore, its proximity to the best known solutions further validates its effectiveness and reliability. This significant reduction in CO_2 emissions is what places our algorithm in a better position, most especially for applications in the real world with a core focus on reducing environmental impact. These advantages make it a helpful tool in modern challenges related to vehicle routing and sustainable logistics.

6. Discussion

As the distances between consumers and businesses increase with the influence of global marketing activities, logistic activities have become more important. Optimization models that minimize efficiency and costs have been used in logistic activities. However, as a result of the decrease in resources, the increase in environmental problems and scientists' warnings about global warming, concepts and optimization models such as sustainable logistics and green logistics have come to the fore in recent years. Vehicle routing problems are also an optimization problem that has an important place among logistics activities. As a result of sustainability studies, it is seen that green vehicle routing or vehicle routing problems and solution methods aimed at reducing CO_2 emissions continue to increase day by day in the literature. In this study, taking these trends into account, a multi-process genetic algorithm was created that minimizes CO_2 emissions of capacity-constrained vehicle routing problems.

The genetic algorithm model, whose optimum parameters were determined according to the experimental studies, was designed to include a multi-process programming technique as a parallel genetic algorithm. Inter-process communication is achieved according to an approach in which each genetic algorithm process updates the best result and adds the best solution found to its population. Studies using Adiba et al.'s formulation to calculate CO_2 emissions in algorithms and problem examples used in these studies were identified. The created algorithm was first tested for the minimization of the total distance. Although it is seen that the model that minimizes the distance also decreases CO_2 emissions, it has been observed that the model that minimizes CO_2 emissions reduces CO_2 emissions more. This is most likely due to the amount of cargo carried on the routes. In the shortest distance model, while 30 tons of material is transported for two customers on one route, 100 tons of material may be transported for six customers on the other route. When the same problem is solved in a model that minimizes CO_2 emissions, 60 tons of material can be transported to three customers on one route, while 70 tons of material is transported to

five customers on the other route. Depending on the amount of load carried on the routes, the fuel consumption of the vehicles, and therefore their CO_2 emissions, will vary. The model that minimizes CO_2 emissions can further reduce emissions because it balances the loads carried on the routes.

The performance of the algorithm we developed was compared with the studies of [1,2], which use the same CO_2 emission formula. It can be said that our algorithm performs worse than Adiba et al.'s algorithm only in the M-n151-k12 example when CO_2 emissions are used as the fitness function. When distances are used as the fitness function, our algorithm gives worse results than Wu et al.'s algorithm only in the E-n101-k8 problem example, and worse than Shuzu's algorithm in the E-n22-k4 and E-n51-k5 examples. It has been observed that if the fitness function is CO_2 emissions, our algorithm gives worse results than Shuzu's algorithm only in the E-n51-k5 example and Adiba et al.'s algorithm in the M-n151-k12 example. For all other problem examples, our algorithm gave better results than the hybrid ant colony optimization stated in [1], the artificial bee colony optimization stated in [2], and the genetic algorithm stated in [2]. As a result of these comparisons, it can be said that our algorithm is quite successful in minimizing CO_2 emissions.

The findings of this study provide significant insights for traffic planners and policymakers aiming to reduce the environmental impact of logistics operations. The use of genetic algorithms demonstrates that optimizing logistics routes can effectively reduce CO_2 emissions. This has critical implications for businesses operating under carbon tax or environmental regulations, as it offers cost-saving advantages. Moreover, adopting Adiba et al.'s formula encourages route planning that accounts for load factors, which can lead to more efficient and environmentally friendly logistics operations. This approach is particularly relevant for managing urban traffic and optimizing resource utilization.

We are aware of the difficulty in validating our model with real-world data. As this is a rather complex task to collect, we acknowledge limitations in this paper and suggest avenues for future work using real-world data through other means, such as simulated datasets or historical case studies that mimic real-world conditions. While we are aware that real-world data would provide more precise validation, even with synthetic or benchmark datasets, our model provides valuable insights into the optimization of CO_2 emissions in vehicle routing problems. Furthermore, sensitivity analyses should be performed to study the behavior of the model under variations in traffic, load, and speed factors that characterize the most realistic scenarios. We further assume Euclidean distances and fixed speeds, which is not realistic under real-world conditions. These assumptions are made initially in order to keep the model simple and focus on the salient aspects of optimization. It may further be improved in the future by taking into consideration realistic speed from real-time traffic flow data and non-Euclidean metrics of distances that provide a good representation of the actual topological structure of the road networks and geographical constraints. It will definitely improve accuracy in modeling realistic scenarios related to logistics.

The most important problem in Turkey, and perhaps in the world, is that researchers cannot obtain sufficient data on real-life problems. It would be much more effective if all the technical information needed about a distribution company's real vehicles and real customers could be obtained and if this application could be carried out with real road conditions using geographical information systems. So, it can be said that the most important limitation is the lack of real-life problem data. We believe that it can be developed further into an algorithm yielding more realistic results, having taken into consideration real world conditions such as road speed, terrain, and fluctuating traffic. For this reason, we had to develop and test our algorithm using problem examples used in vehicle routing competitions.

Although the study uses controlled scenarios based on the CVRP Challenge and CVRPLIB standards, the methodology can be extended to account for real-world variables. One of the reasons why we chose these CVRP examples is that their optimal results are known. Future studies could incorporate dynamic elements, such as fluctuating traffic conditions, stochastic customer demand, and heterogeneous vehicle capacities. By leveraging advancements in real-time data acquisition, such as IoT and traffic monitoring, the proposed algorithm can adapt to complex and dynamic logistics environments.

With the help of real-life data and geographic information systems, the research can determine real road conditions such as speed, slope, etc. The incorporation of these real-life data variables can significantly enhance algorithm precision and predict CO_2 emission values more accurately. A genetic algorithm was used in this study. Other effective algorithms such as particle swarm optimization, ant colony optimization, bee colony optimization, and artificial immune system can also be tried. The operators used in these algorithms, neighbor generation mechanisms, and strategies, and the correct selection of parameters, significantly affect the performance. Hybrid approaches that use different approaches together can be tried. Moreover, it may be advisable to try hyper-heuristic approaches that compete with these algorithms.

In practical applications, the challenges of routing algorithms arise from dynamic traffic conditions and random customer needs. They can adapt to changes dynamically by incorporating real-time traffic data through IoT devices and advanced monitoring systems. The major challenge is how to further improve computational efficiency for large-scale logistics networks by adopting distributed computing and using GPU acceleration. Further work could investigate the incorporation of stochastic optimization techniques to address uncertainties in customer demand and traffic dynamics, further increasing the applicability of the algorithm to real-world settings.

In terms of sustainability, logistic businesses must continue all of their activities by taking measures to reduce harmful gas emissions such as CO_2, CO, and NO_2. While taxes such as carbon taxes are currently being discussed in many countries, such taxes have started to be implemented in some countries. In other words, in order to prepare for the days when polluting nature will be reflected as a cost to the business, applying sustainable planning techniques, using sustainable technologies, and continuing their activities by protecting nature will reduce their costs. For this reason, it would be very useful for the algorithms mentioned in this study to be carried out by companies, to support these studies and to provide the necessary data for these studies.

7. Conclusions

In this study, our genetic algorithm was compared with three studies that used the same CO_2 emission formula as the objective equation. Capacity-limited vehicle routing problem examples are the sample problems used in these studies. Experimental results have shown that the genetic algorithm we developed is competitive with the hybrid ant colony optimization, artificial bee colony optimization, and genetic algorithm model used in these three studies, and can be even more successful in most of the problem examples. Experiments have shown that using the CO_2 formula instead of distance as the fitness function is more effective in reducing the carbon footprint. These results showed that the main purpose of the research (the success of the genetic algorithm in solving CVRP problems aimed at minimizing CO_2 emissions) was achieved.

This study emphasizes the need for sustainable planning techniques in the logistics sector. The results indicate that optimizing logistics routes not only reduces environmental impact but also lowers costs. Traffic planners can adopt such algorithms to develop more efficient and eco-friendly logistics operations. This has practical implications for businesses

aiming to comply with increasing carbon taxes and regulatory pressures, making their operations more sustainable.

Considering that businesses may face pollution costs such as carbon taxes in the near future, the importance of algorithms to reduce CO_2 emissions will be more understood. Reducing the contribution of logistics activities to global warming is also extremely important for reducing disasters caused by climate change.

The study adopts assumptions from CVRP Challenge competitions and CVRPLIB instances to ensure consistency and comparability in algorithm testing. While these benchmarks simplify certain aspects of logistics, they serve as an essential foundation for evaluating algorithm performance. Future research could build on this framework by integrating dynamic factors, enabling a more comprehensive analysis of real-world traffic conditions and logistics operations. Additionally, combining real-life data with information derived from geographic information systems (GISs) offers an opportunity to enhance the algorithm's adaptability to complex environments. Further advancements could also include hybrid and hyper-heuristic approaches to leverage the strengths of multiple optimization techniques, thereby improving solution quality and scalability.

Author Contributions: Conceptualization, B.T.P., A.Y.Ç. and D.Z.; Formal analysis, S.Ç., F.O.V., S.B. and D.Z.; Investigation, B.T.P., A.Y.Ç. and F.M.; Methodology, B.T.P., S.Ç. and A.Y.Ç.; Resources, B.T.P., S.Ç., F.O.V., S.B. and D.Z.; Validation, A.Y.Ç. and F.M.; Writing—original draft, S.Ç. and F.O.V. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Data will be available on request.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Adiba, E.B.E.I.; Elhassania, M.; Ahemd, E.H.A. A hybrid ant colony system for green capacitated vehicle routing problem in sustainable transport. *J. Theor. Appl. Inf. Technol.* **2013**, *54*, 198–208.
2. Wu, C.Y.; Visutarrom, T.; Chiang, T.C. Green Vehicle Routing Problem: The Tradeoff between Travel Distance and Carbon Emissions. In Proceedings of the 15th International Conference on Control, Automation, Robotics and Vision (ICARCV), Singapore, 18–21 November 2018.
3. Zhang, W.; Gajpal, Y.; Appadoo, S.S.; Wei, Q. Multi-depot green vehicle routing problem to minimize carbon emissions. *Sustainability* **2020**, *12*, 3500. [CrossRef]
4. Wu, H.; Tao, F.; Qiao, Q.; Zhang, M. A chance-constrained vehicle routing problem for wetwaste collection and transportation considering carbon emissions. *Int. J. Environ. Res. Public Health* **2020**, *17*, 458. [CrossRef]
5. Behnke, M.; Kirschstein, T.; Bierwirth, C. A column generation approach for an emission-oriented vehicle routing problem on a multigraph. *Eur. J. Oper. Res.* **2021**, *288*, 794–809. [CrossRef]
6. Dewi, S.K.; Utama, D.M. A new hybrid whale optimization algorithm for green vehicle routing problem. *Syst. Sci. Control Eng.* **2021**, *9*, 61–72. [CrossRef]
7. Dinu, V.; Baciu, L.E.; Mortan, M.; Vereș, V.A. Effect of Economic, Institutional and Cultural Factors on the Implementation of EU Energy Policies. *Amfiteatru Econ.* **2023**, *25*, 306–325. [CrossRef]
8. Dumitrescu, D.G.; Horobeț, A.; Tudor, C.D.; Belaşcu, L. Renewables and Decarbonization: Implications for Energy Policy in the European Union. *Amfiteatru Econ.* **2023**, *25*, 345–361. [CrossRef]
9. Liu, Z.; Chen, Y.; Li, J.; Zhang, D. Spatiotemporal-dependent vehicle routing problem considering carbon emissions. *Discret. Dyn. Nat. Soc.* **2021**, *2021*, 9729784. [CrossRef]
10. Lo, S.C.A. Particle swarm optimization approach to solve the vehicle routing problem with cross-docking and carbon emissions reduction in logistics management. *Logistics* **2022**, *6*, 62. [CrossRef]
11. Yao, Q.; Zhu, S.; Li, Y. Green vehicle-routing problem of fresh agricultural products considering carbon emission. *Int. J. Environ. Res. Public Health* **2022**, *19*, 8675. [CrossRef]

12. You, G. Sustainable vehicle routing problem on real-time roads: The restrictive inheritance-based heuristic algorithm. *Sustain. Cities Soc.* **2022**, *19*, 103682. [CrossRef]
13. 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]
14. Liu, W.Y.; Lin, C.C.; Chiu, C.R.; Tsao, Y.S.; Wang, Q. Minimizing the carbon footprint for the time-dependent heterogeneous-fleet vehicle routing problem with alternative paths. *Sustainability* **2014**, *6*, 4658–4684. [CrossRef]
15. Cooray, P.L.N.U.; Rupasinghe, T.D. Machine learning-based parameter tuned genetic algorithm for energy minimizing vehicle routing problem. *J. Ind. Eng.* **2017**, *2017*, 3019523. [CrossRef]
16. Rathnakumara, G.W.H.H.P.; Rupasinghe, T.D. Clustering-based Augmented Tabu Search for Energy Minimizing Vehicle Routing Problem (EMVRP). In Proceedings of the SLAAI—International Conference on Artificial Intelligence, University of Moratuwa, Moratuwa, Sri Lanka, 31 October 2017.
17. Dorigo, M.; Stützle, T. *Ant Colony Optimization*; MIT Press: Cambridge, MA, USA, 2004.
18. Karaboga, D. *An Idea Based on Honey Bee Swarm for Numerical Optimization*; Technical Report-TR06; Erciyes University: Kayseri, Turkey, 2005.
19. Kennedy, J.; Eberhart, R. Particle swarm optimization. In Proceedings of the ICNN'95—International Conference on Neural Networks, Perth, WA, Australia, 27 November–1 December 1995.
20. Holland, J.H. *Adaptation in Natural and Artificial Systems*, 1st ed.; University of Michigan Press: Ann Arbor, MI, USA, 1975.
21. Goldberg, D.E. *Genetic Algorithms in Search, Optimization, and Machine Learning*; Addison-Wesley: Boston, MA, USA, 1989.
22. Laporte, G.; Potvin, J.Y.; Vigo, D. Vehicle routing: Historical perspective and recent contributions. *EURO J. Transp. Logist.* **2016**, *2*, 1–2. [CrossRef]
23. Nallaperuma, S.; Wagner, M.; Neumann, F. Feature-based diversity optimization for vehicle routing problems. *Eur. J. Oper. Res.* **2019**, *272*, 971–983.
24. Li, J.; Wang, F.; He, Y. Electric vehicle routing problem with battery swapping considering energy consumption and carbon emissions. *Sustainability* **2020**, *12*, 10537. [CrossRef]
25. Shuzhu, Z. Modeling Green Logistics Activities for Sustainable Development Using Swarm Intelligence. PhD Thesis, The Hong Kong Polytechnic University, Department of Industrial and Systems Engineering, Hong Kong, China, 2016.
26. de Souza, E.D.; Kerber, J.C.; Bouzon, M.; Rodriguez, C.M.T. Performance evaluation of green logistics: Paving the way towards circular economy. *Clean. Logist. Supply Chain* **2022**, *3*, 100019. [CrossRef]
27. Ferraro, S.; Cantini, A.; Leoni, L.; De Carlo, F. Sustainable logistics 4.0: A study on selecting the best technology for internal material handling. *Sustainability* **2023**, *15*, 7067. [CrossRef]
28. Ceynowa, W.; Przybylowski, A.; Wojtasik, P.; Ciskowski, Ł. ICT Adoption for sustainable logistics development in the horeca and wholesale sectors. *Sustainability* **2023**, *15*, 3746. [CrossRef]
29. Lan, S.; Tseng, M.-L.; Yang, C.; Huisingh, D. Trends in sustainable logistics in major cities in China. *Sci. Total Environ.* **2020**, *712*, 136381. [CrossRef] [PubMed]
30. Qaiser, F.H.; Ahmed, K.; Sykora, M.; Choudhary, A.; Simpson, M. Decision support systems for sustainable logistics: A review and bibliometric analysis. *Ind. Manag. Data Syst.* **2017**, *117*, 1376–1388. [CrossRef]
31. Sun, X.; Yu1, H.; Solvang, W.D.; Wang, Y.; Wang, K. The application of Industry 4.0 technologies in sustainable logistics: A systematic literature review (2012–2020) to explore future research opportunities. *Environ. Sci. Pollut. Res.* **2022**, *29*, 9560–9591. [CrossRef] [PubMed]
32. Dişkaya, F.; Dinçer, S.E. A sectoral application for green vehicle routing problem optimization with capacity constrained and heterogeneous fleet. *EKOIST J. Econom. Stat.* **2024**, *40*, 183–198. [CrossRef]
33. Jamal, A.; Tauhidur Rahman, M.; Al-Ahmadi, H.M.; Ullah, I.; Zahid, M. Intelligent Intersection Control for Delay Optimization: Using Meta-Heuristic Search Algorithms. *Sustainability* **2020**, *12*, 1896. [CrossRef]
34. Fang, Z.; Tu, W.; Li, Q.; Shaw, S.; Chen, S.; Chen, B.Y. A Voronoi neighborhood-based search heuristic for distance/capacity constrained very large vehicle routing problems. *Int. J. Geogr. Inf. Sci.* **2012**, *27*, 741–764. [CrossRef]
35. Hao, D.; Mao, X.; Wei, Y.; Sun, L.; Yang, Y. Review of research on vehicle routing problems. In Proceedings of the SPIE 13018, International Conference on Smart Transportation and City Engineering (STCE 2023), Chongqing, China, 16–18 December 2023. [CrossRef]
36. Napoli, G.; Micari, S.; Dispenza, G.; Andaloro, L.; Antonucci, V.; Polimeni, A. Freight distribution with electric vehicles: A case study in Sicily. RES, infrastructures and vehicle routing. *Transp. Eng.* **2021**, *3*, 100047. [CrossRef]
37. Nguyen, Q.T.; Patrix, J.; Oulmakki, O.; Ramirez-Rios, D.G.; Verny, J. Estimating costs and CO_2 emissions for last mile delivery of groceries in Paris using alternative modes and routing strategies. *Int. J. Logist. Res. Appl.* **2023**, *27*, 2325–2350. [CrossRef]
38. Santos, M.J.; Jorge, D.; Ramos, T.; Barbosa-Póvoa, A. Green reverse logistics: Exploring the vehicle routing problem with deliveries and pickups. *Omega* **2023**, *118*, 102864. [CrossRef]

39. Ghosh, S.; Roy, S.K. Closed-loop multi-objective waste management through vehicle routing problem in neutrosophic hesitant fuzzy environment. *Appl. Soft Comput.* **2023**, *148*, 110854. [CrossRef]
40. Chang, Q.; Qin, R. Analysis on development path of Tianjin green logistics. *Int. J. Bus. Manag.* **2008**, *3*, 96–98. [CrossRef]
41. Parhi, S.; Joshi, K.; Gunasekaran, A.; Sethuraman, K. Reflecting on an empirical study of the digitalization initiatives for sustainability on logistics: The concept of sustainable logistics 4.0. *Clean. Logist. Supply Chain* **2022**, *4*, 100058. [CrossRef]
42. Ali, A.H.; Gruchmann, T.; Melkonyan, A. Assessing the impact of sustainable logistics service quality on relationship quality: Survey-based evidence in Egypt. *Clean. Logist. Supply Chain* **2022**, *4*, 1–12. [CrossRef]
43. Jayarathna, C.P.; Agdas, D.; Dawes, L. Exploring sustainable logistics practices toward a circular economy: A value creation perspective. *Bus Strat Env. Wiley* **2022**, *32*, 704–720. [CrossRef]
44. Dzwigol, H.; Trushkina, N.; Kwilinski, A. Green Logistics as a Sustainable Development Concept of Logistics Systems in a Circular Economy. In Proceedings of the 37th International Business Information Management Association (IBIMA), Cordoba, Spain, 30–31 May 2021.
45. García-Arca, J.; Comesaña-Benavides, J.A.; González-Portela Garrido, A.T.; Prado-Prado, J.C. Rethinking the Box for Sustainable Logistics. *Sustainability* **2020**, *12*, 1870. [CrossRef]
46. Persdotter Isaksson, M.; Hulthén, H.; Forslund, H. Environmentally Sustainable Logistics Performance Management Process Integration between Buyers and 3PLs. *Sustainability* **2019**, *11*, 3061. [CrossRef]
47. Raji, I.O.; Shevtshenko, E.; Rossi, T.; Strozzi, F. Modelling the relationship of digital technologies with lean and agile strategies. *Supply Chain Forum Int. J.* **2021**, *22*, 323–346. [CrossRef]
48. Chhabra, D.; Garg, S.K.; Singh, R.K. Analyzing alternatives for green logistics in an Indian automotive organization: A case study. *J. Clean. Prod.* **2017**, *167*, 962–969. [CrossRef]
49. Baah, C.; Amponsah, K.T.; Issau, K.; Ofori, D.; Acquah, I.S.K.; Agyeman, D.O. Examining the Interconnections Between Sustainable Logistics Practices, Environmental Reputation and Financial Performance: A Mediation Approach. *Vision J. Bus. Perspect.* **2021**, *25*, 47–64. [CrossRef]
50. Werner-Lewandowska, K.; Golinska-Dawson, P. Sustainable logistics management maturity—The theoretical assessment framework and empirical results from Poland. *Sustainability* **2021**, *13*, 5102. [CrossRef]
51. Samad, T.A.; Ganguly, K.K.; Das, D. Towards a framework for development of crowd logistics: Paving the way for sustainable logistics. *Comput. Ind. Eng.* **2023**, *176*, 109008. [CrossRef]
52. Dantzig, G.; Ramser, J. The truck dispatching problem. *Manag. Sci.* **1959**, *6*, 80–91. [CrossRef]
53. Nguyen, V.; Pham, Q.; Nguyen, T.; Bui, Q. Modeling and solving a multi-trip multi-distribution center vehicle routing problem with lower-bound capacity constraints. *Comput. Ind. Eng.* **2022**, *172*, 108597. [CrossRef]
54. Ma, H.; Cheang, B.; Lim, A.; Zhang, L.; Zhu, Y. An investigation into the vehicle routing problem with time windows and link capacity constraints. *Omega* **2012**, *40*, 336–347. [CrossRef]
55. Jozefowiez, N.; Semet, F.; Talbi, E. Multi-objective vehicle routing problems. *Eur. J. Oper. Res.* **2008**, *189*, 293–309. [CrossRef]
56. Xiao, Y.; Zhao, Q.; Kaku, I.; Xu, Y. Development of a fuel consumption optimization model for the capacitated vehicle routing problem. *Comput. Oper. Res.* **2012**, *39*, 1419–1431. [CrossRef]
57. Cai, L.; Lv, W.; Xiao, L.; Xu, Z. Total carbon emissions minimization in connected and automated vehicle routing problem with speed variables. *Expert Syst. Appl.* **2021**, *165*, 113910. [CrossRef]
58. Al-Turki, Y.A.; Alturki, R.; Alturki, T. Smart traffic control system using NSGA-II optimization to reduce delays, fuel consumption, and emissions at urban intersections. *Sustainability* **2020**, *12*, 7394. [CrossRef]
59. Neumann, W.P.; Winkelhaus, S.; Grosse, E.H.; Glock, C.H. Artificial intelligence in logistics: A framework and research agenda. *Int. J. Prod. Res.* **2023**, *62*, 2888–2917. [CrossRef]

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.