Article

A Novel Multistrategy-Based Differential Evolution Algorithm and Its Application

Jinyin Wang [1], Shifan Shang [2,3], Huanyu Jing [2], Jiahui Zhu [2], Yingjie Song [4], Yuangang Li [5,*] and Wu Deng [2,6,*]

1. UNI-FI Credit Solutions Co., Ltd., Beijing 100083, China
2. School of Electronic Information and Automation, Civil Aviation University of China, Tianjin 300300, China
3. HAI ROBOTICS Co., Ltd., Shenzhen 518000, China
4. College of Computer Science and Technology, Shandong Technology and Business University, Yantai 264005, China
5. Faculty of Business Information, Shanghai Business School, Shanghai 200235, China
6. Traction Power State Key Laboratory, Southwest Jiaotong University, Chengdu 610031, China
* Correspondence: yuangang.li@sbs.edu.cn (Y.L.); wdeng@cauc.edu.cn (W.D.)

Abstract: To address the poor searchability, population diversity, and slow convergence speed of the differential evolution (DE) algorithm in solving capacitated vehicle routing problems (CVRP), a new multistrategy-based differential evolution algorithm with the saving mileage algorithm, sequential encoding, and gravitational search algorithm, namely SEGDE, is proposed to solve CVRP in this paper. Firstly, an optimization model of CVRP with the shortest total vehicle routing is established. Then, the saving mileage algorithm is employed to initialize the population of the DE to improve the initial solution quality and the search efficiency. The sequential encoding approach is used to adjust the differential mutation strategy to legalize the current solution and ensure its effectiveness. Finally, the gravitational search algorithm is applied to calculate the gravitational relationship between points to effectively adjust the evolutionary search direction and further improve the search efficiency. Four CVRPs are selected to verify the effectiveness of the proposed SEGDE algorithm. The experimental results show that the proposed SEGDE algorithm can effectively solve the CVRPs and obtain the ideal vehicle routing. It adopts better search speed, global optimization ability, routing length, and stability.

Keywords: differential evolution; capacitated vehicle routing planning; saving mileage; gravity search

1. Introduction

The vehicle routing problem (VRP) was formally presented in 1959 by Dantzig [1]. The problem is defined as finding the optimal route of a vehicle under certain constraint conditions (such as vehicle capacity, customer demand, transportation process, etc.), so as to minimize the transportation cost or find the shortest transportation distance [2–4]. VRP is a NP-hard problem and is one of the hotspots in operations research and combinatorial optimization. In recent years, heuristic algorithms have been widely explored in solving large-scale VRPs [5–8]. Therefore, a new algorithm for VRP has a certain theoretical significance and practical value.

The algorithms for solving VRP can be broadly divided into exact algorithms and heuristic algorithms (including metaheuristics). The exact algorithm can obtain the optimal solution, but its high computational complexity makes it unsuitable for solving large-scale VRPs [9–11]. Heuristic algorithms can be further divided into neighborhood-based algorithms and population-based algorithms [12–14]. The neighborhood-based algorithms maintain a single solution during the search process and seek a more optimal solution by iterating between neighborhood solutions according to the strategy. The algorithms include iterative local search, Tabu search, and so on.

The differential evolution (DE) algorithm is a heuristic search algorithm based on population, and each individual in the population corresponds to a solution vector [15]. The evolution process of DE favors that of GA, which includes mutation, crossover, and selection, but its specific definition is different from that of GA. Since the DE has a simple structure, fast convergence, and so on, it is applied in data mining, pattern recognition, electromagnetics, and so on. However, the DE algorithm also has some defects in solving large-scale VRPs, such as poor searchability and population diversity, slow convergence speed, and so on. Therefore, some variants of DE algorithms are proposed from the different aspects of algorithm, such as parameter adaption, new mutation strategies, crossover strategy strategies, population initialization, hybrid DE with the other algorithms, and so on.

To some extent, these improved DE algorithms have improved the searchability, accelerated the convergence, strengthened avoidance of falling into local optimum, and so on, which can help better obtain optimization results in solving the complex optimization problems and the different VRPs. However, there still exists some defects in solving the complex optimization problems, such as poor population diversity, low search accuracy, easily falling into local optimum, and so on. To solve these problems, a new multistrategy-based differential evolution algorithm with the saving mileage algorithm, sequential encoding, and gravitational search algorithm, namely SEGDE, is proposed to solve the CVRP. A planning method of the CVRP based on SEGDE is implemented to solve the actual CVRP for obtaining the ideal results of the vehicle routing problems.

The main contributions of this study are described as follows:

(1)   A new multistrategy DE algorithm, namely SEGDE, is developed to improve the solution quality and the search efficiency in solving the CVRPs.
(2)   The saving mileage algorithm is used to initialize the population of the DE to ensure the initial solution quality and improve the search efficiency.
(3)   The sorting and coding strategy is used to adjust the differential mutation strategy, and the vectors are added and subtracted.

The structure of this paper is as follows: In Section 3, the related works are reviewed, and the basic DE is introduced. In Section 4, the capacitated vehicle routing model is constructed. Section 5 develops a new multistrategy DE algorithm, and the idea, model, and steps are described in detail. The experimental calculation and analysis are executed in the Section 6. Finally, the conclusions are summarized in Section 7.

2. Related Works

Since the VRP was proposed, many researchers have made in-depth explorations and solved VRPs. When the traditional methods, the exact algorithm, heuristic algorithms, and so on are used to solve the VRPs, a slow solving speed and excessive calculation will occur. In recent years, the focus for solving VRPs has been on combining heuristic algorithms with artificial intelligence technology, such as simulated annealing (SA), tabu search (TS), genetic algorithm (GA), ant colony optimization (ACO), different improvements, and so on. Yusuf et al. [16] studied the GA to solve a combinatorial problem of VRP. Akpinar [17] presented a hybrid algorithm with a large neighborhood search and ACO for CVRP. Zhang et al. [18] presented a hybrid approach with Tabu search and ABC to solve VRP. Dechampai et al. [19] presented a MESOMDE_G-Q-DVRP-FD for solving GQDVRP. Gutierrez et al. [20] presented a new memetic algorithm with multipopulation to solve VRP. Fallah et al. [21] presented a robust algorithm to solve the competitive VRP. Altabeeb et al. [22] presented a new CVRP-firefly algorithm. Altabeeb et al. [23] presented a cooperative hybrid FA with multipopulation to solve VRP. Xiao et al. [24] presented a heuristic EMRG-HA to solve CVRP with a large scale. Jia et al. [25] presented a novel bilevel ACO to solve the CEVRP. Jiang et al. [26] presented a fast evolutionary algorithm called RMEA to accelerate convergence for CVRP. Deng et al. [27] presented an ACDE/F for the gate allocation problem. Zhang et al. [28] presented a branch-and-cut algorithm to solve the two-dimensional loading constraint VRP. Song et al. [29] presented a dynamic hybrid

mechanism CDE to solve the complex optimization problem. Niu et al. [30] presented a multiobjective EA to tackle the MO-VRPSD. Deng et al. [31] presented a new MPSACO with CWBPSO and ACO for solving the taxiway planning problem. Gu et al. [32] presented a hierarchical solution evaluation approach for a general VRPD. Azad et al. [33] presented a QAOA to solve VRP. Lai et al. [34] presented a data-driven flexible transit method with the origin-destination insertion and mixed-integer linear programming for scheduling vehicles. Voigt et al. [35] presented a hybrid adaptive large neighborhood search method to solve three variants of VRP. Seyfi et al. [36] presented a matheuristic method with a variable neighborhood search with mathematical programming to solve multimode HEVRP. Cai et al. [37] presented a hybrid evolutionary multitask algorithm to solve multiobjective VRPTWs. Wen et al. [38] presented an improved adaptive large neighborhood search algorithm to efficiently solve large-scale instances of the multidepot green VRP with time windows. Ma et al. [39] presented an adaptive large neighborhood search algorithm to find near-optimal solutions for larger-size time-dependent VRPs. In addition, some other algorithms are also presented for solving VRPs and the other optimization problems [40–51].

The DE algorithm is widely applied in solving different VRPs. For solving large-scale VRPs, there exist poor searchability, worsened population diversity, a slow convergence speed, and so on. Many researchers have deeply studied and proposed some improvements to the DE algorithm. Zhang et al. [52] presented a new constrained DE to obtain an optimal feasible routing. Teoh et al. [53] presented a local search-based DE to solve CVRP. Pitakaso et al. [54] presented five modified DEs for solving three subproblems. Xing et al. [55] presented a hybrid discrete DE for solving the split delivery VRP in the logistic distribution. Sethanan et al. [56] presented a novel hybrid DE with a genetic operator to solve the multitrip VRP with backhauls. Hameed et al. [57] presented a hybrid algorithm based on discrete DE and TS for solving many instances of QAP. Liu et al. [58] presented a mixed-variable DE for solving the hierarchical mixed-variable optimization problem. Moonsri et al. [59] presented a hybrid and self-adaptive DE for solving an EGG distribution problem. Chai et al. [60] presented a multi-strategy fusion DE with multipopulation, self-adaption and interactive mutation to solve the path planning of UAV. Wu et al. [61] presented a fast and effective improved DE to solve the integer linear programming model. Hou et al. [62] presented a multistate-constrained MODE with a variable neighborhood to solve the real-world-constrained multiobjective problem. Chen et al. [63] presented a fast-neighborhood algorithm based on crowding DE. In addition, some other DE algorithms are also improved for solving the complex optimization problems [64–66]. A summary of the main works is shown Table 1.

Table 1. A summary of the main works.

Name	Key Points	Advantages	Disadvantages
Zhang et al. [52]	Constrained DE	Improve optimization performance	Lack of population diversity
Teoh et al. [53]	Local search-based DE	Explore new search areas	Lack of global searchability
Pitakaso et al. [54]	Five modified DE	Improve population diversity	Fall into local optimal value
Xing et al. [55]	Hybrid discrete DE	Avoid the prematurity and ensure the solution quality	Slow convergence to some extent
Sethanan et al. [56]	Hybrid DE with a genetic operator	Balance the exploration ability	Fall into local optimal value
Hameed et al. [57]	Hybrid algorithm	Enhance solutions, to reduce the distances between the locations	Increase the time complexity
Liu et al. [58]	Mixed-variable DE	Hierarchical mixed-variable mutation operator	Lack of population diversity

Table 1. *Cont.*

Name	Key Points	Advantages	Disadvantages
Moonsri et al. [59]	Hybrid and self-adaptive DE	Self-adaptive mutation strategy	Fall into local optimal value
Chai et al. [60]	Multistrategy fusion DE	Enhance population diversity	Slow convergence to some extent
Hou et al. [62]	Multistate-constrained MODE	Enhance the optimization effectiveness	Increase the time complexity
Chen et al. [63]	Fast-neighborhood DE	Faster convergence	Lack of population diversity

Through these variants of DE, algorithms from various aspects have improved its performance by parameter adaption, designing new mutation/crossover strategy, and hybridity with the other algorithms, and so on. However, some defects, such as poor population diversity and low search accuracy, still exist in solving the complex optimization. Therefore, the DE algorithm needs to be further and more deeply studied in order to solve the large-scale complex optimization problem.

3. Differential Evolution Algorithm

DE is an efficient evolutionary algorithm with a simple and clear structure and idea. It combines parent individuals with other individuals in a population to produce new offspring, which will continue to evolve in place of the parent if they possess better fitness values. In brief, DE consists of the following parts:

3.1. Initialization

The parameters of DE are initialized and generally include: population (Np), dimension (D), mutation factor (F), crossover factor (CR), and the maximum number of iteration (Gm). In addition, the individuals are initialized randomly within the specified range:

$$\left\{ x_{i,1}^{(G)}, x_{i,2}^{(G)}, \ldots, x_{i,D}^{(G)} \right\}, x_{i,D} \in R^D, i = 1, 2, \ldots, NP.$$

3.2. Mutation

In each iteration of evolution, the parent generation generates Np mutation vectors through certain mutation strategies. The mutation strategy is usually expressed as DE/x/y, where x represents the vector to be mutated and Y represents the number of vectors to be mutated during the mutation process. There are five variation strategies that are commonly used in DE:

(1) DE/rand/1
(2) DE/Rand/1

$$V_i^g = X_{r_1}^g + F \times (X_{r_2}^g - X_{r_3}^g) \tag{1}$$

(3) DE/best/1
(4) DE/Best/1

$$V_i^g = X_{best}^g + F \times (X_{r_1}^g - X_{r_2}^g) \tag{2}$$

(5) DE/rand-to-best/1
(6) DE/Rand-to-best/1

$$V_i^g = X_i^g + F \times (X_{best}^g - X_i^g) + F \times (X_{r_1}^g - X_{r_2}^g) \tag{3}$$

(7) DE/current-to-rand/1
(8) DE/Current-to-rand/1

$$V_i^g = X_i^g + K \times (X_{r_1}^g - X_i^g) + F \times (X_{r_2}^g - X_{r_3}^g) \tag{4}$$

(9) DE/current-to-best/1
(10) DE/Current-to-best/1

$$V_i^g = X_i^g + F_1 \times (X_{best}^g - X_i^g) + F_2 \times (X_{r_1}^g - X_{r_2}^g) \tag{5}$$

where r_1, r_2 and r_3 are individuals selected randomly from 1 to *Np* individuals, and *X* is the individual with the best adaptation in the gth iteration.

3.3. Crossover

After the mutation is executed, a crossover operation is performed to generate the final experimental vector *U* by crossing the parent vector *X* with the mutation vector *V* with a certain probability:

$$U_{i,j}^g = \begin{cases} V_{i,j}^g, if \ \text{rand}(0,1) \leq \text{CR or j} = \text{j}_{\text{rand}} \\ X_{i,j}^g, otherwise \end{cases} \tag{6}$$

where $j \in [1, D]$.

3.4. Selection

If the experimental vector *U* performs better in fitness than the parent individual *X*, then the parent individual is replaced with it:

$$X_i^{g+1} = \begin{cases} U_i^g, if \ f\left(U_i^g\right) \leq f(X_i^g) \\ X_i^g, otherwise \end{cases} \tag{7}$$

where *X* will be the parent individual of the next generation evolution, and $f(U)$ and $f(X)$ represent the adaptation values of the current generation experiment vector and the parent individual, respectively.

4. Modeling Capacitated Vehicle Routing

VRP generally refers to organizing and calling a certain number of vehicles to a series of shipping and receiving points, arranging appropriate travel routes so that the vehicles pass through them in an orderly manner [67]. Under specified constraints (e.g., demand and delivery of goods, delivery time, vehicle capacity limits, mileage limits, travel time limits, etc.), we strive to achieve certain goals (e.g., shortest total vehicle miles driven, lowest total transportation costs, vehicles arriving at a certain time, minimum number of vehicles used, and so on.) [68–71].

4.1. Model Assumptions

The following assumptions are made for the model based on the actual problem:

(1) The distribution center is assigned to complete a series of demand point distribution services.
(2) The relative geographical location and the corresponding demand quantity of the distribution center and each demand point are given clearly.
(3) Vehicle distribution is completed and returned to the designated distribution center.
(4) The vehicles have the same specifications, and there are no errors.
(5) There is no consideration of urban traffic congestion.
(6) The distribution vehicles always travel at a constant speed, and the distribution cost is equal within the unit distance, so the travel distance can represent the distribution cost.
(7) Each demand point shall be served by only one delivery vehicle, and the sum of the requirements of all the demand points of the vehicle service shall be less than or equal to the rated load limit of the vehicle.

4.2. Symbolic Description

The relevant symbols are described in Table 2.

Table 2. List of symbols involved in the CVRP model.

Symbols	Meaning
m	Number of vehicles in distribution center
n	Number of customer points
Q	Vehicle capacity
di	The requirement for customer points I, di > 0 (i > 0), and D0 = 0
cij	The distance from point i to point j
Xijk	The degree of delivery requirements from the k vehicle distribution Point i to point j
V	A collection of distribution centers and customer points

4.3. Objective Optimization Function

The CVRP model can be constructed based on the mentioned distribution objectives and distribution requirements as follows:

Distribution objective:

$$Min \ Z = \sum_{i=0}^{n} \sum_{j=0}^{n} \sum_{k=1}^{m} c_{ij} x_{ijk} \tag{8}$$

Constraints:

$$\sum_{i(j)=0}^{n} \sum_{k=1}^{m} x_{ijk} = 1 \ , i,j = 0,1,2,\ldots,n \tag{9}$$

$$\sum_{i=0}^{n} x_{ipk} - \sum_{j=0}^{n} x_{ijk} = 0 \ , \ k = 1,2,\ldots,m \ , \ p = 0,1,\ldots,n \tag{10}$$

$$\sum_{i=0}^{n} \sum_{j=0}^{n} d_{i} x_{ijk} \leq Q, k = 1,2,\ldots,m \tag{11}$$

$$\sum_{i=1}^{n} \sum_{j=1}^{n} x_{ijk} \leq |V| - 1 \ , k = 1,2,\ldots,m \tag{12}$$

$$x_{ijk} \in \{0,1\} \ , \ i,j = 0,1,2,\ldots,n \ , \ k = 1,2,\ldots,m \tag{13}$$

The optimization goal is represented by an Equation (8) to minimize the total distance traveled. The constraint (9) represents the availability of one and only one vehicle per customer point to provide service. The constraint (10) ensures that a customer point is visited the same number of times as it is left. The constraint (11) ensures that the vehicle works within its maximum load. The constraint (12) means that the subtour is eliminated. The constraint (13) provides a mutable limit.

5. A Multistrategy-Based Differential Evolution Algorithm

The DE is a population-based adaptive global optimization algorithm with a simple structure and high robustness. However, there are some problems in solving optimization problems, such as poor searchability, slow convergence, and a tendency to fall into local optimality. Therefore, a multistrategy DE algorithm, namely SEGDE, is proposed by introducing the population initialization strategy, the differential mutation strategy, and the gravity search algorithm. The mileage saving method is used to initialize the population of the DE to improve the initial solution quality and the search efficiency. The differential mutation strategy is adjusted by using a sequential encoding approach to perform a legalization operation on the current solution to ensure that the solution is valid. Finally,

the gravity search algorithm (GSA) is introduced to calculate the gravitational relationship between points, which can be used to legitimize the solution, reinsert the points, effectively adjust the search direction of evolution, optimize the search efficiency, and prevent the algorithm from falling into local optimum, to obtain better optimization ability of complex optimization problems.

These strategies in the SEGDE are described in detail as follows.

5.1. Population Initialization Strategy

Traditional DE algorithms usually use population random initialization to randomly distribute the initial population in the feasible domain. In this way, the algorithm does not depend on the initial population solution, but the quality of the initial population often affects the efficiency and accuracy of the global search algorithm. The saving mileage method is a heuristic algorithm for solving transportation problems [72]. The key idea of the heuristic method is to combine the two circuits of the transportation problem according to the distance table, which can reduce the total transportation distance and make the distribution more efficient. Therefore, the initial population is a combination of the solution of the mileage-saving method and the random individuals, which ensures the initial population solution quality and allows the algorithm to carry out the follow-up search around the individuals with better quality, to improve search efficiency.

5.2. Differential Mutation Strategy

Since the CVRP is discrete, a ranking encoding approach is used to adjust the operation of the differential variation strategy DE/neighbor-to-neighbor/1 by using ranking numbers instead of vectors for addition and subtraction. In addition, the solution after mutation operation is not necessarily the legal solution to meet the requirements; after the mutation operation, the current solution should be legal operation to ensure the effectiveness of the solution. The solutions are searched from right to left, the repeated points are set to zero, and the zero positions are re-inserted by using contemporary evolutionary individuals. The individual variation was calculated using Equation (14), and the adjusted variation process is shown in Table 3.

$$V^g_{i,j} = \begin{cases} \mathrm{mod}\left(X^g_{r_3,j} + \left(X^g_{best,j} - X^g_{r_3,j}\right) + \left(X^g_{r_1,j} - X^g_{r_2,j}\right) + j - 1, j\right), if\ rand < F \\ X^g_{best,j}, if\ rand \geq F \end{cases} \tag{14}$$

Table 3. Examples of variant operations (F = 0.5).

$X^g_{r_1} - X^g_{r_2}$						
$X^g_{r_1}$　　7	4	3	5	2	1	6
$X^g_{r_2}$　　5	2	1	3	7	4	6
$X^g_{r_1} - X^g_{r_2}$　　2	2	2	2	−5 −5	−3 −3	0
$X^g_{best} - X^g_{r_3}$						
X^g_{best}　　5	1	3	4	2	7	6
$X^g_{r_3}$　　2	3	5	1	7	6	4
$X^g_{best} - X^g_{r_3}$　　3	−2 −2	−2 −2	3	−5 −5	1	2
V^g_i						
rand Rand　　0.18	0.22	0.53	0.78	0.61	0.39	0.42
U^g_i　　1	1	3	1	1	5	1
V^g_i　　5	1	3	1	1	7	6

5.3. Variable Correlation Using GSA

VRP is an optimization problem with point-line network topology. The key to solving this problem is discovering the correlation between the points and connecting them. The gravitational search algorithm (GSA) is used to calculate the gravitational relationship between points, and the point-point relationship table is used for the legitimization of the solution and the reinsertion link of points, which can effectively adjust the evolutionary search direction and optimize the search efficiency. GSA is a bionic algorithm based on the laws of Newton's law of gravity and the laws of kinematics [73]. The core idea of the algorithm is to calculate the value of the gravitational force between points according to Newton's universal gravity formula, update the gravitational table, adjust the mass of the points according to the gravitational table, and use the mass table updated in the current generation to guide the next generation solution.

Define the attraction between individual i and individual j as follows:

$$F_{ij}^d(t) = G(t) \frac{M_{pi}(t) \times M_{aj}(t)}{R_{ij}(t) + \varepsilon} \left(x_j^d(t) - x_i^d(t) \right) \tag{15}$$

where M_{aj} is the related active gravitational mass of individual j, and M_{pj} is the related passive gravitational mass of individual i. ε is a variable to prevent variables with denominators. $R_{ij}(t)$ is the Euclidean distance between individuals i and j.

$$R_{ij}(t) = \|X_i(t) \cdot X_j(t)\|_2 \tag{16}$$

In the d-dimension space, the exerted force on any particle is the exerted resultant force on it by other particles, and the random weighted sum of the gravitational forces of each particle is expressed as follows:

$$F_i^d(t) = \sum_{j=1, j \neq i}^{N} \text{rand}_j F_{ij}^d(t) \tag{17}$$

where rand_j is a random value in [0,1].

Therefore, the acceleration of an individual i in the d-dimension is described as follows:

$$a_i^d(t) = \frac{F_i^d(t)}{M_{ii}(t)} \tag{18}$$

where M_{ii} is the inertial gravity of individual i at iteration t.

Based on the above model, the position update of individuals can be obtained as follows:

$$v_i^d(t+1) = \text{rand}_i \times v_i^d(t) + a_i^d(t) \tag{19}$$

$$x_i^d(t+1) = x_i^d(t) + v_i^d(t+1) \tag{20}$$

where rand_i is a random value in [0,1].

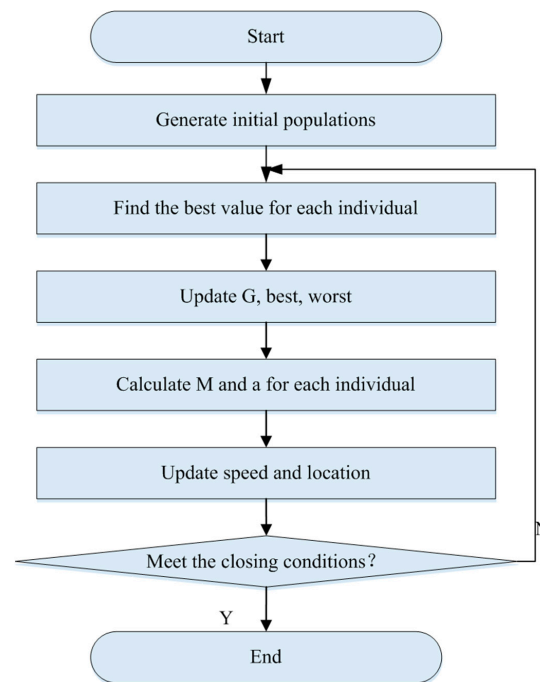The GSA algorithm framework is shown in Figure 1.

Figure 1. The framework of the GSA.

5.4. Model of the SEGDE

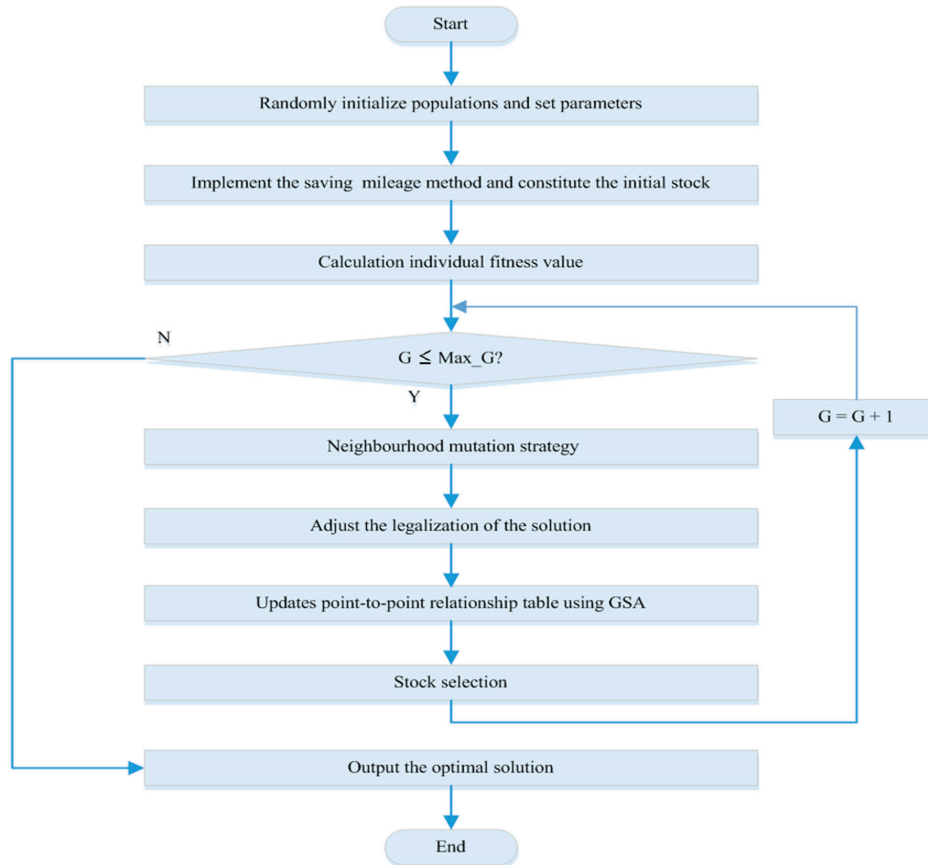The flow of the SEGDE algorithm is shown in Figure 2.

Figure 2. The flow of the SEGDE algorithm.

The implementation steps of the SEGDE are described as follows:

Step 1. The initial population is randomly generated by sequence coding, and the size of the initial population is NP, the dimension D, the maximum evolutionary iteration number Max, and the iteration number G = 1.

Step 2. The initial population is composed of the solution of the mileage saving method and the random solution of the mileage saving method.

Step 3. Calculate the initial fitness values of the individuals.

Step 4. If the number of iterations G is less than the maximum number of evolutionary iterations Max, enter Step 5; otherwise, proceed to Step 10.

Step 5. The strategy of neighborhood mutation is implemented to legalize the solution of the mutated population.

Step 6. The neighborhood search is carried out for the individual population, and the optimal solution in the local search is preserved.

Step 7. The gravity search algorithm is used to explore the relationship between variables and update the table of point-point relations, preserving the optimal solution.

Step 8. A population selection operation is performed.

Step 9. If the number of iterations G = G + 1, return to Step 4.

Step 10. The output evolutionary optimal solution is obtained.

6. Experimental Calculation and Analysis

6.1. Experimental Data

In order to verify the effectiveness of the SEGDE algorithm in solving the CVRP, data sets were selected from the operational research database OR-LIBRARY and the VRP database | NEO Research Group (uma.es). A total of 41 data instances with fewer than 50 dimensions were selected from among four test data sets.

6.2. Experimental Environment and Parameter Settings

The experimental environment included CPU-intel Core I5-4200H, Windows-Win8, RAM-4GB, and MATLAB R2018B. In the experiment, many alternative values are tested, and some classical values were selected from the literature; these parameter values were experimentally modified until the most reasonable parameter values were determined. These selected parameter values obtained the optimal solution, so that they could accurately and efficiently verify the effectiveness of the proposed SEGDE algorithm. Each experiment was carried out 25 times independently, and the optimal solution of 25 experiments was selected to compare with the other five algorithms. The five comparison algorithms were standard DE, GA, SA, the mileage-saving method (MS), and the improved MS(IMS) method. The settings of the parameters are shown in Table 4.

Table 4. The initial parameters of all algorithms.

Algorithms	Parameter Settings
SA	delta = 0.85, T = 150, Np = 100
GA	CR = 0.7, F = 0.5, Np = 100
DE	CR = 0.9, F = 0.5, Np = 100
SEGDE	Fmin = 0.5, Fmax = 0.9, CR = 0.9, Np = 100

6.3. Experimental Results and Analysis

The obtained experimental results are shown in Tables 5–8.

Table 5. The experimental results of six algorithms in solving set A.

Test Data	Opt.	SA	GA	MS	IMS	DE	SEGDE
A32_5	784	**739**	850	842	827	1426	813
A33_5	661	740	700	713	700	1194	**680**
A33_6	742	924	798	775	**743**	1233	746
A34_5	778	895	856	810	793	1347	**789**
A36_5	799	814	897	826	806	1367	**805**
A37_5	669	806	752	705	708	1366	**685**
A37_6	949	**949**	1047	975	974	1595	954
A38_5	730	908	789	765	751	1497	**734**
A39_5	822	1009	954	898	894	1575	**871**
A39_6	831	1011	940	861	**848**	1618	852
A44_6	937	1021	974	985	1785	1534	**943**
A45_6	944	1231	1111	1005	**955**	2093	963
A45_7	1146	1431	1282	1200	**1178**	1968	1203
A46_7	914	1431	1068	940	**934**	1862	935
A48_7	1073	1343	1280	1110	**1102**	2180	1129

Table 6. The experimental results of six algorithms in solving set E.

Test Data	Opt.	SA	GA	MS	IMS	DE	SEGDE
E22_K4	375	394	**375**	388	**375**	441	**375**
E23_K3	569	575	575	621	574	888	**569**
E30_K3	508	564	557	532	-	976	**508**
E33_K4	835	929	904	**841**	**841**	1180	**841**
E51_K5	521	697	685	582	-	1315	**575**

Table 7. The experimental results of six algorithms in solving set P.

Test Data	Opt.	SA	GA	MS	IMS	DE	SEGDE
P16_K8	450	889	**451**	478	472	452	**451**
P19_K2	212	**213**	**213**	237	219	276	**213**
P20_K2	216	**217**	218	234	247	452	**217**
P21_K2	211	**213**	**213**	236	233	318	**213**
P22_K2	216	222	219	240	234	317	**218**
P22_K8	589	**589**	**589**	591	590	624	**589**
P23_K8	529	541	532	537	537	633	**531**
P40_K5	458	561	526	516	**484**	629	508
P45_K5	510	616	614	569	**519**	1142	563

Table 8. The experimental results of six algorithms in solving set B.

| Test Data	Opt.	SA	GA	DE	SEGDE
B31_K5	672	697	706	886	**679**
B34_K5	788	839	799	1186	**790**
B35_K5	955	1021	991	1665	**970**
B38_K6	805	887	845	1343	**825**
B39_K5	549	649	577	1314	**563**
B41_K6	829	989	880	1565	**838**
B43_K6	742	907	833	1387	**775**
B44_K7	909	1139	1058	1725	**931**
B45_K5	751	918	880	1631	**755**
B45_K6	678	888	791	1317	**698**
B50_K7	741	1006	879	1875	**766**
B50_K8	1312	1462	1401	2132	**1352**
B31_K5	672	697	706	886	**679**
B34_K5	788	839	799	1186	**790**
B35_K5	955	1021	991	1665	**970**

As can be observed from Tables 5–8, for set A, the proposed SEGDE algorithm has the best solutions of A33_5, A34_5, A36_5, A37_5, A38_5, and A39_5, and the IMS has the best solutions of A33_6, A39_6, A45_6, A45_7, A46_7, and A48_7. SA has the best solutions of A32_5 and A37_6. The IMS and SEGDE algorithm have obtained the best solutions of six cases. The obtained best solutions of A33_6, A34_5, A37_6, A38_5, and A44_6 are close to the optimal values by using the proposed SEGDE algorithm. For set E, the proposed SEGDE algorithm has obtained the best solutions of all cases. In particular, the optimal solutions of E22_K4, E23_K3, and E30_K3 are obtained using the proposed SEGDE algorithm. The best solutions of the other cases are also close to the optimal values using the proposed SEGDE algorithm. For set P, the proposed SEGDE algorithm has obtained the best solutions, except those of P40_K5 and P45_K5. The optimal solution of P22_K8 is obtained, and the obtained other solutions are also infinitely close to the optimal values using the proposed SEGDE algorithm. The IMS has obtained the best solutions of P40_K5 and P45_K5. For set B, the proposed SEGDE algorithm has obtained all best solutions of all cases. The obtained best solutions of B31_K5, B34_K5, B45_K5, and B34_K5 are infinitely close to the optimal values using the proposed SEGDE algorithm. The experimental results demonstrate that the proposed SEGDE algorithm can better solve these CVRPs from the operational research database OR-LIBRARY and the VRP database, and the optimized solutions are the optimal values, or are (infinitely) close to the optimal values. Therefore, the proposed SEGDE algorithm takes on a better global optimization ability in solving these different CVRPs. The reason for this is that the proposed SEGDE algorithm optimizes the abilities of the saving mileage algorithm, the sequential encoding approach, and the differential mutation strategy.

The routing comparison curves for generations 1 and 200 in the A33-K6 and B34-K5 optimization iterations are shown in Figures 3 and 4.

As can be observed from the optimization curves of the A33-K6 and B34-K5 cases in Figures 3 and 4, the obtained optimization paths by using the proposed SEGDE algorithm overlap to lessen, eliminate the path knot phenomenon, and effectively connect the adjacent points. In addition, the paths gradually become localized, which achieves the total path reduction. Through the experimental results of the test data, it can be observed that the proposed SEGDE algorithm possesses an advantage in addressing the vehicle path planning problem, and can approach the optimal solution to a great extent when the problem of fewer than 30 dimensions are processed. It also performs well on most of the problems with fewer than 50 dimensions, which proves the effectiveness of the proposed SEGDE algorithm in solving the different CVRPs. Therefore, the proposed SEGDE algorithm can effectively solve the CVRPs and obtain the optimized vehicle routing, as well as eliminate the path knotting, thus avoiding overlap. It is an effective algorithm for solving the CVRPs and the complex optimization problems.
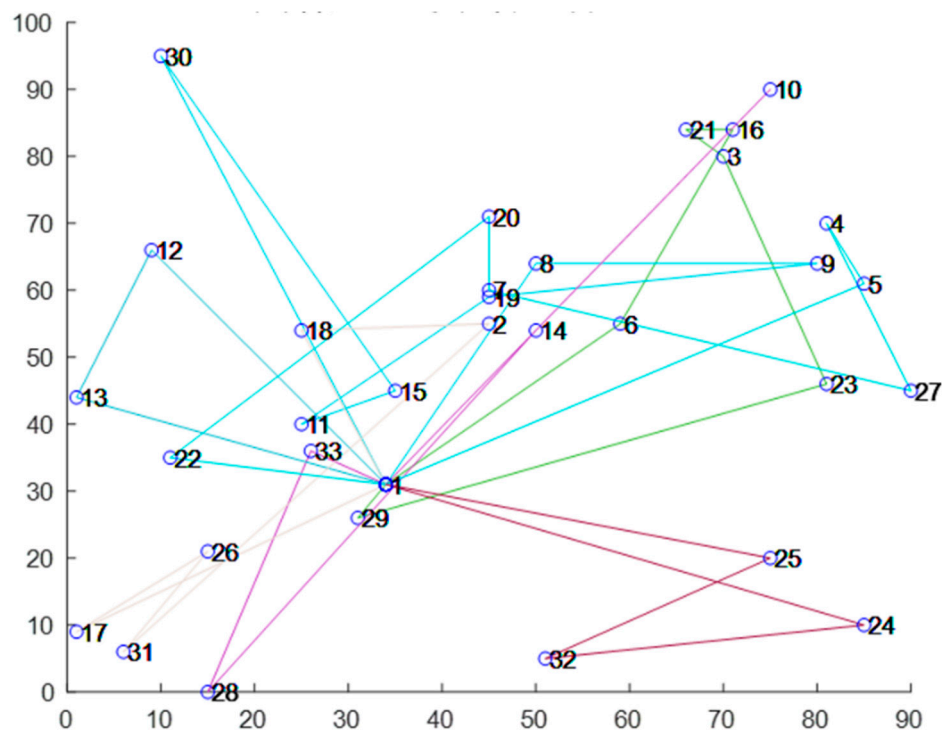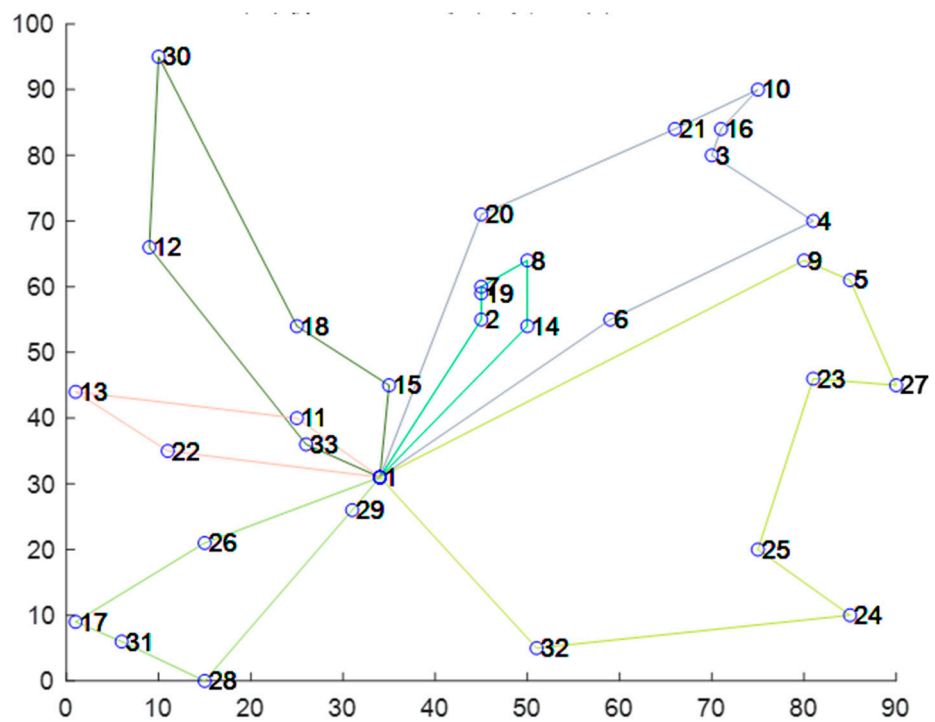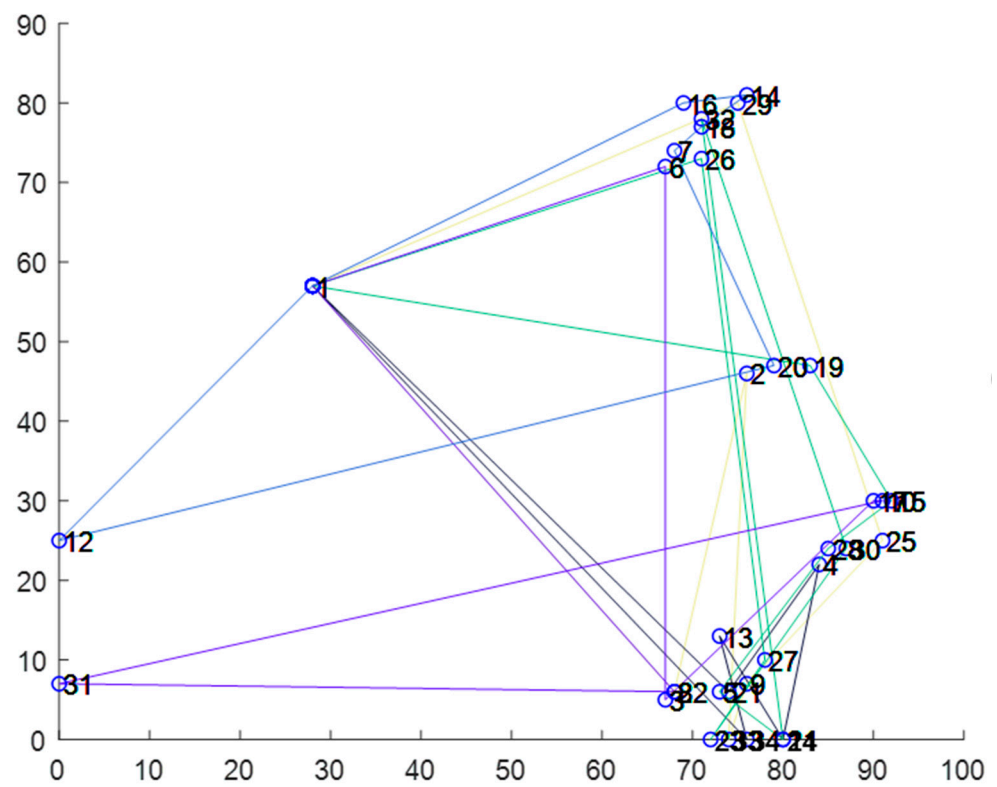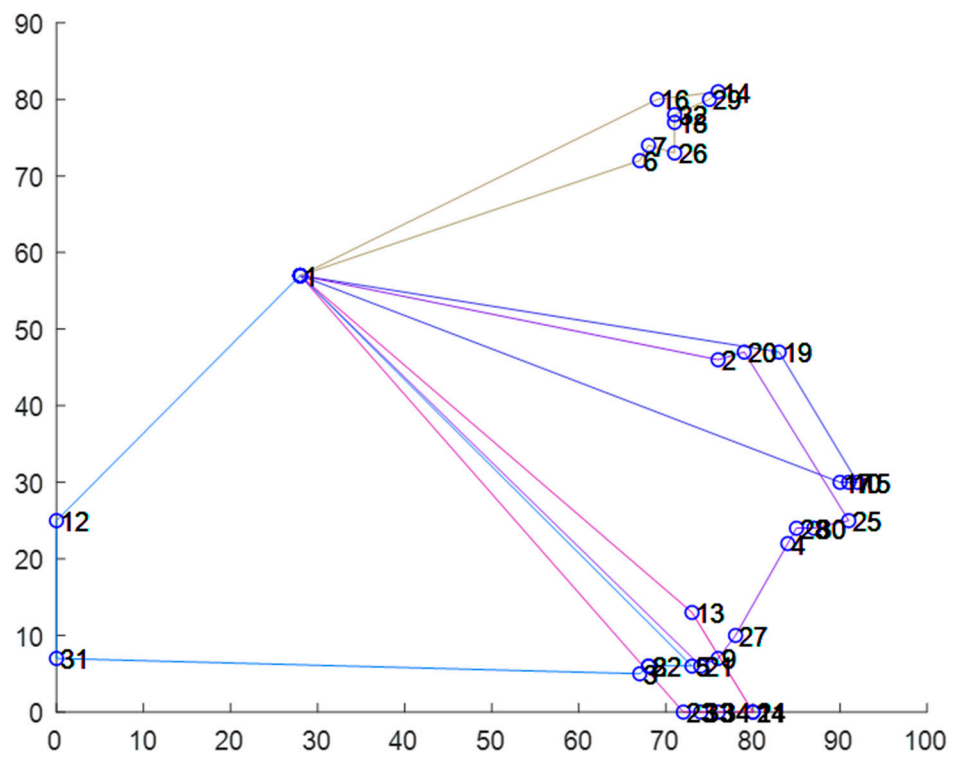
Figure 3. The optimization effect of A33-K6. (**a**) Optimization curve at Generation 1(1336.2577). (**b**) Optimization curve at Generation 200(745.6772).

Figure 4. The optimization effect of B34-K5. (**a**) Optimization curve at Generation 1(1492.6296). (**b**) Optimization curve at Generation 200(790.3643).

6.4. Discussion

As can be observed from Tables 5–8 and Figures 3 and 4, the proposed SEGDE algorithm is used to solve CVRPs of set A, set B, set E, and set P; the obtained best solutions of E22_K4, E23_K3, E30_K3, and P22_K8 are the optimal values, and the obtained best solutions of A36_5, A38_5, E33_K4, P16_K8, P19_K2, P20_K2, P21_K2, P22_K2, and P23_K8 are (infinitely) close to the optimal values. Compared with the SA, GA, MS, IMS, and DE, the proposed SEGDE algorithm can effectively solve these various CVRPs and obtain the ideal vehicle routing, as well as eliminate the path knotting, avoiding overlap. Therefore, the proposed SEGDE algorithm adopts a better global optimization ability. The reason is that the proposed SEGDE algorithm is based on the saving mileage algorithm, the sequential encoding approach, and the differential mutation strategy. It optimizes the abilities of the saving mileage algorithm, the sequential encoding approach, and the differential mutation strategy. The saving mileage algorithm can improve the initial solution quality and the search efficiency by initializing the population of the DE. The sequential encoding approach can legalize the current solution and ensure its effectiveness by adjusting the differential mutation strategy. The gravitational search algorithm can effectively adjust the evolutionary search direction and further improve the search efficiency by calculating the gravitational relationship between points.

7. Conclusions

In this paper, a new multistrategy DE, namely SEGDE, is proposed to solve various CVRPs. In order to improve the search efficiency, the saving mileage algorithm is employed to initialize the population of DE. The sequential encoding method is used to adjust the differential mutation strategy to legalize the current solution and ensure its effectiveness. The GSA is applied to calculate the gravitational relationship between points for solution legalization and point reinsertion, which can effectively adjust the evolutionary search direction and optimize the search efficiency. Finally, the CVRP example from the operational research database is selected to verify the effectiveness of the proposed SEGDE algorithm. The obtained best solutions of E22_K4, E23_K3, E30_K3, and P22_K8 are the optimal values, and the obtained best solutions of A36_5, A38_5, E33_K4, P16_K8, P19_K2, P20_K2, P21_K2, P22_K2, and P23_K8 are (infinitely) close to the optimal values. Compared with the SA, GA, MS, IMS, and DE, the proposed SEGDE algorithm can effectively solve these different CVRPs and obtain the ideal vehicle routing, as well as eliminate the path knotting, avoiding overlap. Therefore, the experimental results demonstrate that the proposed SEGDE algorithm has a good optimization ability, search speed, and routing length. In addition, the stability of the SEGDE also possesses a good advantage.

Author Contributions: Conceptualization, J.W. and S.S.; methodology, S.S.; software, H.J.; validation, J.Z., H.J. and Y.S.; formal analysis, H.J.; resources, Y.L.; data curation, Y.L.; writing—original draft preparation, J.W. and S.S.; writing—review and editing, Y.L. and W.D.; visualization, J.Z.; supervision, H.J.; project administration, J.W.; funding acquisition, W.D. All authors have read and agreed to the published version of the manuscript.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

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

References

1. Hulagu, S.; Celikoglu, H.B. An electric vehicle routing problem with intermediate nodes for shuttle fleets. *IEEE Trans. Intell. Transp. Syst.* **2022**, *23*, 1223–1235. [CrossRef]
2. Yang, Y.; Chen, H.; Heidari, A.A.; Gandomi, A.H. Hunger games search: Visions, conception, implementation, deep analysis, perspectives, and towards performance shifts. *Expert Syst. Appl.* **2021**, *177*, 114864. [CrossRef]
3. Felipe, A.; Ortuno, M.T.; Righini, G.; Tirado, G. A heuristic approach for the green vehicle routing problem with multiple technologies and partial recharges. *Transp. Res. Part E-Logist. Transp. Rev.* **2014**, *71*, 111–128. [CrossRef]
4. Ahmadianfar, I.; Heidari, A.A.; Noshadian, S.; Chen, H.; Gandomi, A.H. INFO: An efficient optimization algorithm based on weighted mean of vectors. *Expert Syst. Appl.* **2022**, *195*, 116516. [CrossRef]
5. Wang, Z.; Sheu, J.B. Vehicle routing problem with drones. *Transp. Res. Part B-Methodol.* **2019**, *122*, 350–364. [CrossRef]
6. Dorling, K.; Heinrichs, J.; Messier, G.G.; Magierowski, S. Vehicle routing problems for drone delivery. *IEEE Trans. Syst. Man Cybern.-Syst.* **2017**, *47*, 70–85. [CrossRef]
7. Wang, X.Y.; Shao, S.; Tang, J.F. Iterative local-search heuristic for weighted vehicle routing problem. *IEEE Trans. Intell. Transp. Syst.* **2021**, *22*, 3444–3454. [CrossRef]
8. Wang, H.; Li, M.H.; Wang, Z.Y.; Li, W.; Hou, T.J.; Yang, X.Y.; Zhao, Z.Z.; Wang, Z.F.; Sun, T. Heterogeneous fleets for green vehicle routing problem with traffic restrictions. In *IEEE Transactions on Intelligent Transportation Systems*; IEEE: Piscataway, NJ, USA, 2022. [CrossRef]
9. Khaitan, A.; Mehlawat, M.K.; Gupta, P.; Pedrycz, W. Socially aware fuzzy vehicle routing problem: A topic modeling based approach for driver well-being. *Expert Syst. Appl.* **2022**, *205*, 117655. [CrossRef]
10. Ahmadianfar, I.; Heidari, A.A.; Gandomi, A.H.; Chu, X.; Chen, H. Run beyond the metaphor: An efficient optimization algorithm based on Runge Kutta method. *Expert Syst. Appl.* **2021**, *181*, 115079. [CrossRef]
11. Oztas, T.; Tus, A. A hybrid metaheuristic algorithm based on iterated local search for vehicle routing problem with simultaneous pickup and delivery. *Expert Syst. Appl.* **2022**, *202*, 117401. [CrossRef]
12. Feng, B.; Wei, L.X. An improved multi-directional local search algorithm for vehicle routing problem with time windows and route balance. *Appl. Intell.* **2022**, 1–13. [CrossRef]
13. Thiebaut, K.; Pessoa, A. Approximating the chance-constrained capacitated vehicle routing problem with robust optimization. *4OR-A Q. J. Oper. Res.* **2022**, 1–19. [CrossRef]
14. Li, S.; Chen, H.; Wang, M.; Heidari, A.A.; Mirjalili, S. Slime mould algorithm: A new method for stochastic optimization. *Futur. Gener. Comput. Syst.* **2020**, *111*, 300–323. [CrossRef]
15. Storn, R.; Price, K. *Differential Evolution: A Simple and Efficient Adaptive Scheme for Global Optimization over Continuous Spaces*; Technical Report; TR-95-012; International Computer Science Institute: California, CA, USA, 1995.
16. Yusuf, I.; Baba, M.S.; Iksan, N. Applied genetic algorithm for solving rich VRP. *Appl. Artif. Intell.* **2014**, *28*, 957–991. [CrossRef]
17. Akpinar, S. Hybrid large neighbourhood search algorithm for capacitated vehicle routing problem. *Expert Syst. Appl.* **2016**, *61*, 28–38. [CrossRef]
18. Zhang, D.F.; Cai, S.F.; Ye, F.R.; Si, Y.W.; Nguyen, T.T. A hybrid algorithm for a vehicle routing problem with realistic constraints. *Inf. Sci.* **2017**, *394*, 167–182. [CrossRef]
19. Dechampai, D.; Tanwanichkul, L.; Sethanan, K.; Pitakaso, R. A differential evolution algorithm for the capacitated VRP with flexibility of mixing pickup and delivery services and the maximum duration of a route in poultry industry. *J. Intell. Manuf.* **2017**, *28*, 1357–1376. [CrossRef]
20. Gutierrez, A.; Dieulle, L.; Labadie, N.; Velasco, N. A multi-population algorithm to solve the VRP with stochastic service and travel times. *Comput. Ind. Eng.* **2018**, *125*, 144–156. [CrossRef]
21. Fallah, M.; Tavakkoli-Moghaddam, R.; Alinaghian, M.; Salamatbakhsh-Varjovi, A. A robust approach for a green periodic competitive VRP under uncertainty: DE and PSO algorithms. *J. Intell. Fuzzy Syst.* **2019**, *36*, 5213–5225. [CrossRef]
22. Altabeeb, A.M.; Mohsen, A.M.; Ghallab, A. An improved hybrid firefly algorithm for capacitated vehicle routing problem. *Appl. Soft Comput.* **2019**, *84*, 105728. [CrossRef]
23. Altabeeb, A.M.; Mohsen, A.M.; Abualigah, L.; Ghallab, A. Solving capacitated vehicle routing problem using cooperative firefly algorithm. *Appl. Soft Comput.* **2021**, *108*, 107403. [CrossRef]
24. Xiao, J.H.; Zhang, T.; Du, J.G.; Zhang, X.Y. An evolutionary multiobjective route grouping-based heuristic algorithm for large-scale capacitated vehicle routing problems. *IEEE Trans. Cybern.* **2021**, *51*, 4173–4186. [CrossRef] [PubMed]
25. Jia, Y.H.; Mei, Y.; Zhang, M.J. A bilevel ant colony optimization algorithm for capacitated electric vehicle routing problem. *IEEE Trans. Cybern.* **2022**, *52*, 10855–10868. [CrossRef] [PubMed]
26. Jiang, H.; Lu, M.X.; Tian, Y.; Qiu, J.F.; Zhang, X.Y. An evolutionary algorithm for solving Capacitated Vehicle Routing Problems by using local information. *Appl. Soft Comput.* **2022**, *117*, 108431. [CrossRef]
27. Deng, W.; Ni, H.C.; Liu, Y.; Chen, H.L.; Zhao, H.M. An adaptive differential evolution algorithm based on belief space and generalized opposition-based learning for resource allocation. *Appl. Soft Comput.* **2022**, *127*, 109419. [CrossRef]
28. Zhang, X.Y.; Chen, L.; Gendreau, M.; Langevin, A. A branch-and-cut algorithm for the vehicle routing problem with two-dimensional loading constraints. *Eur. J. Oper. Res.* **2022**, *302*, 259–269. [CrossRef]
29. Song, Y.J.; Cai, X.; Zhou, X.; Zhang, B.; Chen, H.; Li, Y.G.; Deng, W.; Deng, W. Dynamic hybrid mechanism-based differential evolution algorithm and its application. *Expert Syst. Appl.* **2023**, *213*, 118834. [CrossRef]

30. Niu, Y.Y.; Shao, J.; Xiao, J.H.; Song, W.; Cao, Z.G. Multi-objective evolutionary algorithm based on RBF network for solving the stochastic vehicle routing problem. *Inf. Sci.* **2022**, *609*, 387–410. [CrossRef]
31. Deng, W.; Zhang, L.R.; Zhou, X.B.; Zhou, Y.Q.; Sun, Y.Z.; Zhu, W.H.; Chen, H.Y.; Deng, W.Q.; Chen, H.L.; Zhao, H.M. Multi-strategy particle swarm and ant colony hybrid optimization for airport taxiway planning problem. *Inf. Sci.* **2022**, *612*, 576–593. [CrossRef]
32. Gu, R.X.; Poon, M.; Luo, Z.H.; Liu, Y.; Liu, Z. A hierarchical solution evaluation method and a hybrid algorithm for the vehicle routing problem with drones and multiple visits. *Transp. Res. Part C Emerg. Technol.* **2022**, *141*, 103733. [CrossRef]
33. Azad, U.; Behera, B.K.; Ahmed, E.A.; Panigrahi, P.K.; Farouk, A. Solving vehicle routing problem using quantum approximate optimization algorithm. In *IEEE Transactions on Intelligent Transportation Systems*; IEEE: Piscataway, NJ, USA, 2022. [CrossRef]
34. Lai, Y.X.; Yang, F.; Meng, G.; Lu, W. Data-driven flexible vehicle scheduling and route optimization. In *IEEE Transactions on Intelligent Transportation Systems*; IEEE: Piscataway, NJ, USA, 2022. [CrossRef]
35. Voigt, S.; Frank, M.; Fontaine, P.; Kuhn, H. Hybrid adaptive large neighborhood search for vehicle routing problems with depot location decisions. *Comput. Oper. Res.* **2022**, *146*, 105856. [CrossRef]
36. Seyfi, M.; Alinaghian, M.; Ghorbani, E.; Catay, B.; Sabbagh, M.S. Multi-mode hybrid electric vehicle routing problem. *Transp. Res. Part E-Logist. Transp. Rev.* **2022**, *166*, 102882. [CrossRef]
37. Cai, Y.Q.; Cheng, M.Q.; Zhou, Y.; Liu, P.Z.; Guo, J.M. A hybrid evolutionary multitask algorithm for the multiobjective vehicle routing problem with time windows. *Inf. Sci.* **2022**, *612*, 168–187. [CrossRef]
38. Wen, M.Y.; Sun, W.; Yu, Y.; Tang, J.F.; Ikou, K. An adaptive large neighborhood search for the larger-scale multi depot green vehicle routing problem with time windows. *J. Clean. Prod.* **2022**, *374*, 133916. [CrossRef]
39. Ma, B.S.; Hu, D.W.; Wang, Y.; Sun, Q.; He, L.W.; Chen, X.Q. Time-dependent vehicle routing problem with departure time and speed optimization for shared autonomous electric vehicle service. *Appl. Math. Model.* **2023**, *113*, 333–357. [CrossRef]
40. Zhou, X.B.; Ma, H.J.; Gu, J.G.; Chen, H.L.; Deng, W. Parameter adaptation-based ant colony optimization with dynamic hybrid mechanism. *Eng. Appl. Artif. Intell.* **2022**, *114*, 105139. [CrossRef]
41. Ren, Z.; Han, X.; Yu, X.; Skjetne, R.; Leira, B.J.; Sævik, S.; Zhu, M. Data-driven simultaneous identification of the 6DOF dynamic model and wave load for a ship in waves. *Mech. Syst. Signal Process.* **2023**, *184*, 109422. [CrossRef]
42. Chen, H.Y.; Miao, F.; Chen, Y.J.; Xiong, Y.J.; Chen, T. A hyperspectral image classification method using multifeature vectors and optimized KELM. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **2021**, *14*, 2781–2795. [CrossRef]
43. Xu, G.; Dong, W.; Xing, J.; Lei, W.; Liu, J. Delay-CJ: A novel cryptojacking covert attack method based on delayed strategy and its detection. *Digit. Commun. Netw.* 2022, in press. [CrossRef]
44. Yao, R.; Guo, C.; Deng, W.; Zhao, H.M. A novel mathematical morphology spectrum entropy based on scale-adaptive techniques. *ISA Trans.* **2022**, *126*, 691–702. [CrossRef]
45. Li, T.Y.; Shi, J.Y.; Deng, W.; Hu, Z.D. Pyramid particle swarm optimization with novel strategies of competition and cooperation. *Appl. Soft Comput.* **2022**, *121*, 108731. [CrossRef]
46. Zhao, H.M.; Liu, J.; Chen, H.Y.; Chen, J.; Li, Y.; Xu, J.J.; Deng, W. Intelligent diagnosis using continuous wavelet transform and gauss convolutional deep belief network. In *IEEE Transactions on Reliability*; IEEE: Piscataway, NJ, USA, 2022. [CrossRef]
47. Wu, D.; Wu, C. Research on the time-dependent split delivery green vehicle routing problem for fresh agricultural products with multiple time windows. *Agriculture* **2022**, *12*, 793. [CrossRef]
48. Xu, G.; Bai, H.; Xing, J.; Luo, T.; Xiong, N.N. SG-PBFT: A secure and highly efficient distributed blockchain PBFT consensus algorithm for intelligent Internet of vehicles. *J. Parallel Distrib. Comput.* **2022**, *164*, 1–11. [CrossRef]
49. Wei, Y.Y.; Zhou, Y.Q.; Luo, Q.F.; Deng, W. Optimal reactive power dispatch using an improved slime Mould algorithm. *Energy Rep.* **2021**, *7*, 8742–8759. [CrossRef]
50. Zhang, Z.; Huang, W.G.; Liao, Y.; Song, Z.; Shi, J.; Jiang, X.; Shen, C.; Zhu, Z. Bearing fault diagnosis via generalized logarithm sparse regularization. *Mech. Syst. Signal Process.* **2022**, *167*, 108576. [CrossRef]
51. Chen, H.Y.; Fang, M.; Xu, S. Hyperspectral remote sensing image classification with CNN based on quantum genetic-optimized sparse representation. *IEEE Access* **2020**, *8*, 99900–99909. [CrossRef]
52. Zhang, X.Y.; Duan, H.B. An improved constrained differential evolution algorithm for unmanned aerial vehicle global route planning. *Appl. Soft Comput.* **2014**, *26*, 270–284. [CrossRef]
53. Teoh, B.E.; Ponnambalam, S.G.; Kanagaraj, G. Differential evolution algorithm with local search for capacitated vehicle routing problem. *Int. J. Bio-Inspired Comput.* **2015**, *7*, 321–342. [CrossRef]
54. Pitakaso, R.; Sethanan, K.; Srijaroon, N. Modified differential evolution algorithms for multi-vehicle allocation and route optimization for employee transportation. *Eng. Optim.* **2019**, *52*, 1225–1243. [CrossRef]
55. Xing, L.N.; Liu, Y.Y.; Li, H.Y.; Wu, C.C.; Lin, W.C.; Song, W. A hybrid discrete differential evolution algorithm to solve the split delivery vehicle routing problem. *IEEE Access* **2020**, *8*, 207962–207972. [CrossRef]
56. Sethanan, K.; Jamrus, T. Hybrid differential evolution algorithm and genetic operator for multi-trip vehicle routing problem with backhauls and heterogeneous fleet in the beverage logistics industry. *Comput. Ind. Eng.* **2020**, *146*, 106571. [CrossRef]
57. Hameed, A.S.; Aboobaider, B.M.; Mutar, M.L.; Choon, N.H. A new hybrid approach based on discrete differential evolution algorithm to enhancement solutions of quadratic assignment problem. *Int. J. Ind. Eng. Comput.* **2020**, *11*, 51–72. [CrossRef]
58. Liu, W.L.; Gong, Y.J.; Chen, W.N.; Liu, Z.Q.; Wang, H.; Zhang, J. Coordinated charging scheduling of electric vehicles: A mixed-variable differential evolution approach. *IEEE Trans. Intell. Transp. Syst.* **2020**, *21*, 5094–5109. [CrossRef]

59. Moonsri, K.; Sethanan, K.; Worasan, K.; Nitisiri, K. A hybrid and self-adaptive differential evolution algorithm for the multi-depot vehicle routing problem in EGG distribution. *Appl. Sci.* **2022**, *12*, 35. [CrossRef]

60. Chai, X.Z.; Zheng, Z.S.; Xiao, J.M.; Yan, L.; Qu, B.Y.; Wen, P.W.; Wang, H.Y.; Zhou, Y.; Sun, H. Multi-strategy fusion differential evolution algorithm for UAV path planning in complex environment. *Aerosp. Sci. Technol.* **2022**, *121*, 107287. [CrossRef]

61. Wu, P.; Xu, L.; D'Ariano, A.; Zhao, Y.X.; Chu, C.B. Novel formulations and improved differential evolution algorithm for optimal lane reservation with task merging. In *IEEE Transactions on Intelligent Transportation Systems*; IEEE: Piscataway, NJ, USA, 2022. [CrossRef]

62. Hou, Y.; Wu, Y.L.; Han, H.G. Multistate-constrained multiobjective differential evolution algorithm with variable neighborhood strategy. In *IEEE Transactions on Cybernetics*; IEEE: Piscataway, NJ, USA, 2022. [CrossRef]

63. Chen, M.C.; Yerasani, S.; Tiwari, M.K. Solving a 3-dimensional vehicle routing problem with delivery options in city logistics using fast-neighborhood based crowding differential evolution algorithm. *J. Ambient. Intell. Humaniz. Comput.* **2022**, 1–14. [CrossRef]

64. Deng, W.; Xu, J.; Song, Y.; Zhao, H.M. Differential evolution algorithm with wavelet basis function and optimal mutation strategy for complex optimization problem. *Appl. Soft Comput.* **2021**, *100*, 106724. [CrossRef]

65. Li, X.; Zhao, H.; Yu, L.; Chen, H.; Deng, W.Q.; Deng, W. Feature extraction using parameterized multisynchrosqueezing transform. *IEEE Sens. J.* **2022**, *22*, 14263–14272. [CrossRef]

66. Wu, X.; Wang, Z.C.; Wu, T.H.; Bao, X.G. Solving the family traveling salesperson problem in the adleman–lipton model based on DNA computing. *IEEE Trans. NanoBioscience* **2021**, *21*, 75–85. [CrossRef]

67. Abu-Monshar, A.; Al-Bazi, A. A multi-objective centralised agent-based optimisation approach for vehicle routing problem with unique vehicles. *Appl. Soft Comput.* **2022**, *125*, 109187. [CrossRef]

68. Torres, F.; Gendreau, M.; Rei, W. Vehicle routing with stochastic supply of crowd vehicles and time windows. *Transp. Sci.* **2021**, *56*, 631–653. [CrossRef]

69. Kuo, R.J.; Lu, S.H.; Mara, S.T.W. Vehicle routing problem with drones considering time windows. *Expert Syst. Appl.* **2022**, *191*, 116264. [CrossRef]

70. Ochelska-Mierzejewska, J.; Poniszewska-Maranda, A.; Maranda, W. Selected genetic algorithms for vehicle routing problem solving. *Electronics* **2022**, *10*, 3147. [CrossRef]

71. Lei, D.M.; Cui, Z.Z.; Li, M. A dynamical artificial bee colony for vehicle routing problem with drones. *Eng. Appl. Artif. Intell.* **2022**, *107*, 104510. [CrossRef]

72. Sheng, Y.K.; Lan, W.L. *Application of Clarke-Wright Saving Mileage Heuristic Algorithm in Logistics Distribution Route Optimization*; Trans Tech Publications Ltd.: Baech, Switzerland, 2011.

73. Hosseinabadi, A.A.R.; Vahidi, J.; Balas, V.E.; Mirkamali, S.S. OVRP_GELS: Solving open vehicle routing problem using the gravitational emulation local search algorithm. *Neural Comput. Appl.* **2017**, *29*, 955–968. [CrossRef]