Decision Science Letters 13 (2024) 525–534

Contents lists available at GrowingScience

Decision Science Letters

homepage: www.GrowingScience.com/dsl

Stas crossover with K-mean clustering for vehicle routing problem with time window

Ratchadakorn Poohoi[a*], Kanate Puntusavase[a] and Shunichi Ohmori[b]

[a]*Department of Industrial Engineering and Management, Faculty of Engineering and Industrial Technology, Silpakorn University, Nakhon Pathom, 73000, Thailand*
[b]*Graduate School of Creative Science and Engineering, Waseda University, Tokyo 169-8050, Japan*

C H R O N I C L E

Article history:
Received: February 20, 2024
Received in the revised format: May 8, 2024
Accepted: May 31, 2024
Available online: May 31, 2024

Keywords:
Vehicle Routing Problem with Time Window
Genetic Algorithm
K-mean Clustering
Crossover Operator

A B S T R A C T

Vehicle Routing Problem (VRP) is important in the transportation and logistics industries. Vehicle Routing Problem with Time Window (VRPTW) is a kind of VRP with the additional time windows constraint in the model and is classified as an NP-hard problem. In this study, we proposed Stas crossover in Genetic Algorithm (GA) to solve VRPTW by developing the problem with K-mean clustering. The experiments use the standard Solomon's benchmark problem instances for VRPTW. The results with K-mean clustering are shown to perform better for minimum distance and average distance than without K-mean clustering. In the case of location and dispersion characteristics of the customer, the paths with K-mean clustering are arranged into groups and are orderly, but the paths without K-mean clustering are disordered. After that, this paper shows the comparison of the crossover operator performance on instances of Solomon benchmark, and appropriate crossover operators are recommended for each type of problem. The results of the proposed algorithm are better than the best-known solutions from the previous studies for some instances. Moreover, our proposed research will serve as a guideline for a real-world case study.

1. Introduction

Vehicle Routing Problem (VRP) is important in the transportation and logistics industries. Vehicle Routing Problem with Time Window (VRPTW) is a kind of VRP with an additional time windows constraint in the model. The objective is to minimize the total distance traveled or the number of vehicles used and specify the routes for the vehicles. The problem can be described as finding routes with a limited number of vehicles, where each vehicle has a limited capacity. It starts from a central depot to serve exactly one customer in the time window and terminates at the depot (Kallehauge et al., 2005; Aruyani et al., 2018; Thangiah, 1995). The VRPTW is classified as an NP-hard problem (Ahmed et al., 2023), which means the computational difficulty required to solve this problem increases exponentially with the size of the problem.

Many researchers pay attention to finding the different methods for solving VRPTW, such as the exact method heuristics method, and meta heuristics method. For example, Ant Colony Optimization: Shi & Weise (2013) proposed the primary goal of decreasing the number of vehicles to serve customers and reducing the distance traveled. Kosolsombat and Ratanavilisagul (2022) present a novel ACO-based optimization method for VRPTW and the re-initialization technique to reduce or solve trapping in the local optimum. Gambardella (2000) designed to continuously optimize multiple objectives: the first will minimize the number of vehicles, and the second will minimize the distance traveled. Particle Swarm Optimization: Amini (2011) proposed PSO is used for VRPTW in a real-case study of a Chlorine Capsule distribution company to the water reservoir in Tehran. Relevant results indicate that the algorithm can significantly reduce costs and time. Simulated Annealing: Mohammadi and Mahmoodian (2022) focused on minimizing the total distance traveled by

* Corresponding author.
E-mail address: poohoi_r@silpakorn.edu (R. Poohoi)

doi: 10.5267/dsl.2024.5.008

vehicles in the distribution chain and determining the desired schedule in which vehicles should serve specific customers in the network.

Genetic Algorithm (GA) has been a popular algorithm to solve VRPTW problems. May et al. (2021) study propose a new improved GA to solve variants of VRPTW with the hard time windows by developing the problem-specific crossover and seven different mutation operators. Ghani et al. (2016) studied assigning several vehicles to the customers and depot to minimize the overall distance traveled and complete delivery operations within the time windows required by customers. Kinoshita and Uchiya (2021) propose a method to verify optimization accuracy while maintaining dynamic switching with multiple crossovers according to the diversity of the gene population.

Solomon benchmarks are the most popular problem to solve in VRPTW. The problem is composed of six problem sets of different problem types, which are described as C1, R1, RC1, C2, R2, and RC2 (Solomon, 1987; Solomon & Desrosiers, 1998). Each set contains between eight to twelve 100-node problems. These are the six sets of problems that Set C has generated in the cluster of customers. Set R has generated uniformly random locations, while Set RC has a combination of Set C and Set R. Type 1 has narrow time windows and small vehicle capacity, while Type 2 has large time windows and large vehicle capacity (Solomon, 2005; Gambardella, 2000).

In this paper, we proposed Stas crossover (Poohoi et al., 2023) in GA to solve VRPTW by developing the problem with K-mean clustering. For this study, we specifically examine Stas crossover compared with four crossover operators and adjust the size of the area probability for Stas crossover. After that, improved GA has been tested on Solomon benchmarks with six problem sets of different problem types, which are described as R1, C1, RC1, R2, C2, and RC2, then recommend appropriate crossover operators for each type of problem. Moreover, our proposed research will be a guideline for a real-world case study.

2. Mathematical model of VPRTW

The proposed VRPTW is to serve products to several customers within time windows, and each vehicle has limited capacity (Ghani et al., 2016). VRPTW is formulated into the mathematical model (May et al., 2021) as follows:

The following are the symbols to describe the model:

N = number of customers

K = number of vehicles

D(ij) = distance that can be traveled from customer *i* to customer *j*

d(i) = delivery demand of customer *i*

C(k) = capacity of vehicle

T(i) = arrival time at customer *i*

e(i) = earliest arrival time at customer *i*

l(i) = latest arrival time at customer *i*

s(i) = service time at customer *i*

Decision Variable:

$$x(ijk) = \begin{cases} 1 & \text{if the variable } k \text{ travels from customer } i \text{ to customer } j \\ 0 & \text{otherwise} \end{cases}$$

$i \neq j$; $i, j \in \{0, 1, 2, \ldots, N\}$; 0 refers to depot.

Objective function:

$$\text{minimize}\left(\sum_{i=0}^{N}\sum_{j=0}^{N}\sum_{j\neq 1,k=1}^{K} D(ij).x(ijk)\right) \tag{1}$$

subject to

$$\sum_{j=0}^{N} x(ijk) = 1, \mathrm{i} = 0 \text{ and } \forall\, k \in K \tag{2}$$

$$\sum_{j=0}^{N} x(ijk) \leq K, \text{ i } = 0 \tag{3}$$

$$\sum_{k=1}^{N} \sum_{j=0, j \neq i}^{N} x(ijk) = 1, \forall i \in N$$
$$\sum_{k=1}^{N} \sum_{i=0, i \neq j}^{N} x(ijk) = 1, \forall j \in N \tag{4}$$

$$\sum_{i=1}^{N} x(ijk) - \sum_{i=1}^{N} x(ijk) = 0, \forall i \in N, \forall k \in K \tag{5}$$

$$\sum_{i=1}^{N} \sum_{j=0, j \neq 1}^{N} d(i). x(ijk) \leq C(k), \forall k \in K \tag{6}$$

$$e(i) \leq T(i) + s(i) \leq l(i) \tag{7}$$

Eq. (1) is the objective function to minimize the total distance traveled by all vehicles, and each vehicle has a limited capacity within the time windows the customer requires. Eq. (2) represents each vehicle starting from a central depot and terminating at the depot. Eq. (3) defines the number of vehicles at the depot, which means the number of routes. Eq. (4) determines that each customer can be visited only once by one of the vehicles from the depot. Eq. (5) constraint that the same vehicle must arrive and depart from that customer. Eq. (6) specifies that the demand of each customer on each vehicle route must be less than or equal to the vehicle capacity. Eq. (7) requires that vehicles cannot arrive earlier than the earliest arrival time and cannot be later than the latest arrival time.

3. Methodology

John Holland developed GA based on Darwin's evolutionary theory in 1988 (Goldberg & Holland, 1988; Goldberg, 1989) and expanded GA in 1992 (Holland, 1992). GA has been a popular algorithm to solve VRPTW problems. The first stage, where GA starts with a randomly generated initial population. The operations of GA include chromosome representation, selection, crossover, mutation, and fitness function computation. In Fig. 1, a chromosome is represented by the number sequence of vehicles. Chromosomes with higher fitness values have a higher chance of being selected than chromosomes with lower fitness values. The condition will terminate after a maximum number of generations or close the optimal solutions at the end of the run.

Many studies on solving VRPTW using GA have been done. However, this study used Stas crossover in GA to solve VRPTW by applying K-mean clustering for better performance. The proposed algorithm imports the databases and specifies the number of vehicles required for the model based on capacity and the total amount of cargo. Identify the number of vehicles required for the model according to its capacity and the total amount of cargo to be transported (Villalba et al., 2022; Alfiyatin et al., 2018). Then K-mean clustering was applied, and the GA operations were performed.

3.1 K-mean clustering

K-mean clustering is the process of dividing all data into groups (called clusters) based on patterns in the data. The first thing that is done in the clustering process is initializing k, which the number of clusters. The center of each cluster was then randomly determined, and the objects were grouped based on the minimum distance. The proposed clustering problem is considered a customer to vehicle assignment problem.

3.2 Crossover Operators

The crossover operator is the recombination of two individuals to exchange and produce a completely new offspring. For this work, we will explain the most used crossover operators, including Single point crossover, Two points crossover, Arithmetic crossover, and Scattered crossover. In addition to that, there is also a Stas crossover.

Single point crossover operator that selected two parent chromosomes and designed the cut-off point. The values of the bits to the right of the cut-off point are swapped with the two parent chromosomes. The offspring are created by crossing over the genes of the parents. Two points crossovers are chosen randomly from the parent chromosomes. The genetic information between the two points is swapped to create the offspring. Arithmetic crossover is an operator that linearly combines the two parent chromosomes to create the offspring as follows.

Offspring1=a×Parent1 + (1-a) ×Parent2

$$Offspring2=(1-a)\times Parent1 + a\times Parent2$$

where *a* is the random weighting factor (selected before each crossover operation).

Scattered crossover is the selection of parent bits from those randomly generated bits at the time of crossover. Some bit positions are randomly selected; one is from the first parent, and the other is from the second parent. Another individual is to select the converse bit. Two new individuals are made for the next generation.

Stas crossover is a combination of four crossover operators. This includes Single point crossover, Two points crossover, Arithmetic crossover, and Scattered crossover. Stas crossover operator chooses two parents for the crossover process to place all crossover operators into a roulette wheel so that it can adjust the size of the area probability. It allows more diversity in selecting the way to create offspring and increases the opportunity for offspring to directly obtain good genetic information. The roulette is turned to select the crossover operator to create the new offspring. The following Fig. 2. illustrates Stas crossover process, which shows that the new offspring would have an equal probability of occurrence at 25%.

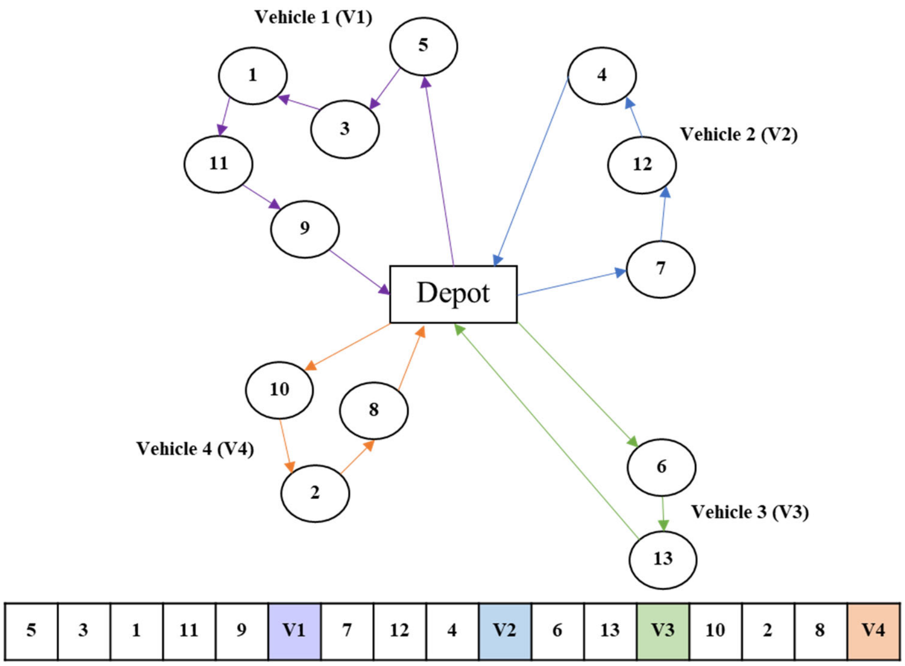

Fig. 1. Illustrates Chromosome representation

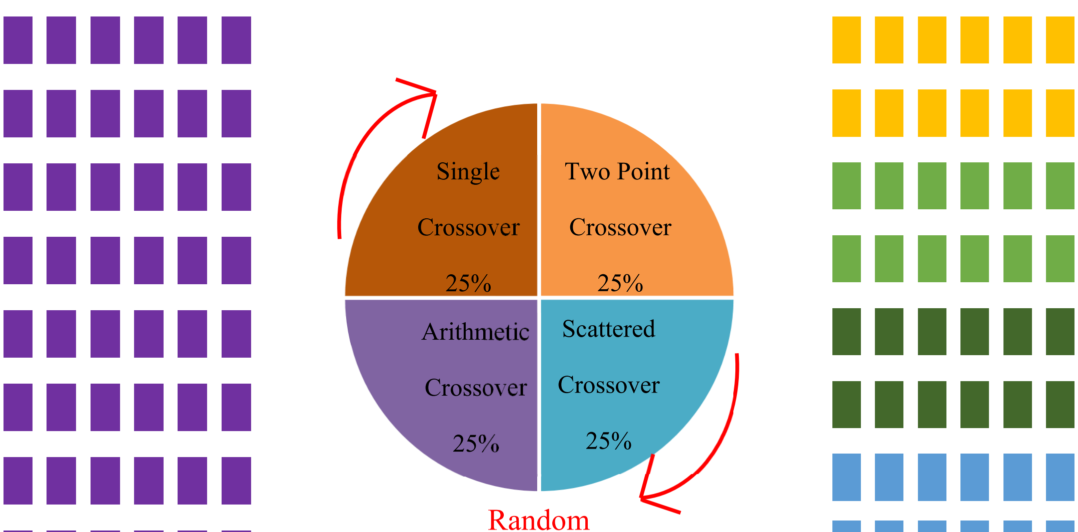

Fig. 2. Illustrates Stas crossover process.

4. Results and Discussion

Results obtained from the aim Stas crossover in GA with K-mean clustering approach can solve VRPTW. In this research, the algorithm was coded in Microsoft Visual Studio 2010. The experiments use the standard Solomon's benchmark problem instances for VRPTW. The population size is 2,000, the maximum generation is 350, and the number of clusters is 10 groups. The code has been run for up to 2,000 iterations. Results are obtained for 10 run iterations and the average value is to be reported for each combination. Fig. 3 shows six scenarios with the location and dispersion characteristics of the customer. K-mean clustering is better performance in routing than without K-mean clustering. For example, it can be seen obviously that in Set C in both type 1 and type 2, the paths without K-mean clustering are disordered, but when using K-mean clustering, the paths are arranged into groups and are orderly. Furthermore, in Set R and Set RC, they are the same in both types too.

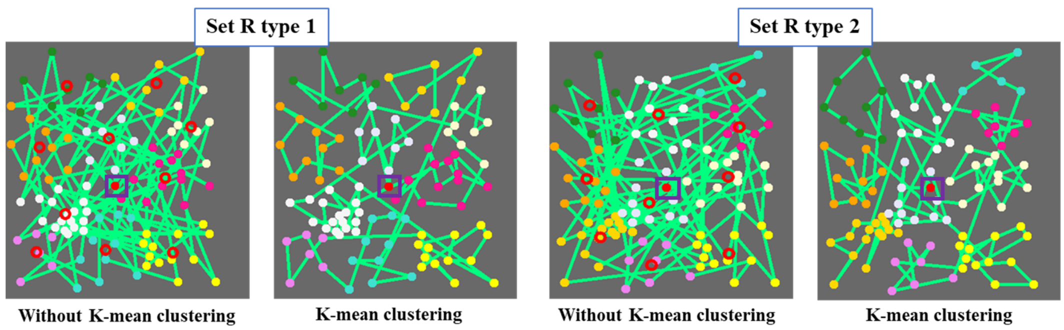

Fig. 3. (a) Scenario of Set R type 1 and type 2

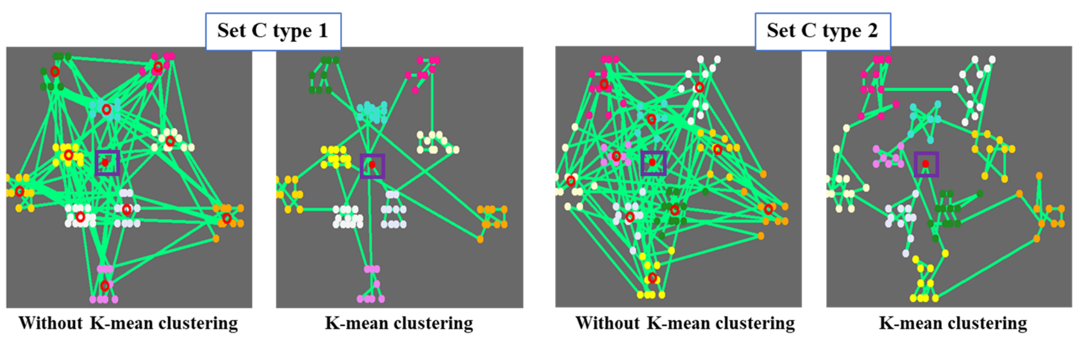

Fig. 3. (b) Scenario of Set C type 1 and type 2

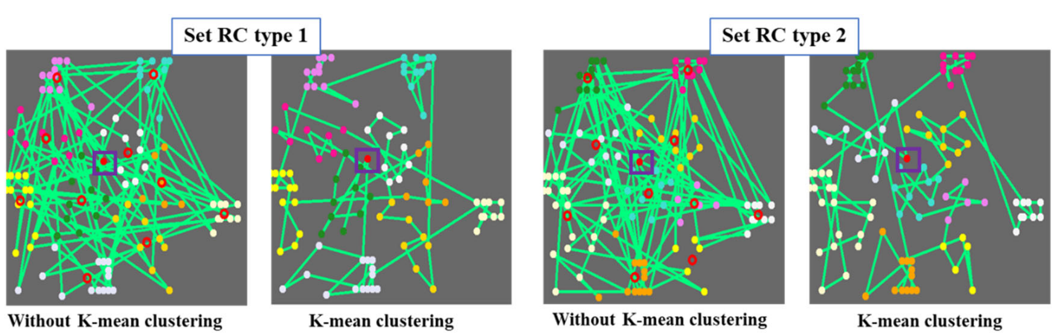

Fig. 3. (c) Scenario of Set RC type 1 and type 2

Table 1 shows the comparison of the crossover operator performance on instances of the Solomon benchmark with Set R type 1 and type 2. Set R is generated randomly from the customer locations. From the results, in minimum distance without K-mean clustering, Set R type 1 is appropriate for Single point crossover, Set R type 2 is appropriate for Single point crossover or Scatter crossover. In minimum distance with K-mean clustering, Set R type 1 is appropriate for Stas1117 crossover, Set R type 2 is appropriate for Stas5005 crossover. In average distance without K-mean clustering, Set R type 1 is appropriate for Scatter crossover, Set R type 2 is appropriate for Scatter crossover. In average distance with K-mean clustering, Set R type 1 is appropriate for Stas1117 crossover, Set R type 2 is appropriate for Stas1117 crossover.

Table 2: Comparison of the crossover operator performance on instances of the Solomon benchmark with Set C type 1 and type 2. Set C is clustered all of the customer location coordinates. From the results, it can be seen that in minimum distance without K-mean clustering, Set C type 1 is appropriate for Single point crossover, Set C type 2 is appropriate for Single point crossover or Scatter crossover. In minimum distance with K-mean clustering, Set C type 1 is appropriate for Stas7111 crossover, Set C type 2 is appropriate for Single point crossover. In average distance without K-mean clustering, Set C type 1 is appropriate for Scatter crossover, Set C type 2 is appropriate for Single point crossover, Scatter crossover or Stas1117 crossover. In average distance with K-mean clustering, Set C type 1 is appropriate for Stas7111 crossover, Set C type 2 is appropriate for Scatter crossover.

Table 3: Comparison of the crossover operator performance on instances of the Solomon benchmark with Set RC type 1 and type 2. Set RC is mixed of Set R and Set C. From the results, it can be seen that in minimum distance without K-mean clustering, Set RC type 1 is appropriate for Single point crossover, Set RC type 2 is appropriate for Scatter crossover, Stas crossover or Stas7111 crossover. In minimum distance with K-mean clustering, Set RC type 1 is appropriate for Stas7111 crossover, Set RC type 2 is appropriate for Stas1117 crossover. In average distance without K-mean clustering, Set RC type 1 is appropriate for Stas1117 crossover, Set RC type 2 is appropriate for Scatter crossover. In average distance with K-mean clustering, Set RC type 1 is appropriate for Stas7111 crossover, Set RC type 2 is appropriate for Stas1117 crossover.

Table 4: Comparison of the algorithm performance on instances of the Solomon Benchmark for type 1 and type 2. From the results, Stas crossover with K-mean Clustering is significantly improved as it allows more diversity to select how to create offspring and arrange orderly paths. It increased the opportunity to create offspring with good genetic information directly. Some experimental results are performing better as compared with the previous best-published studies. However, it is shown that the proposed algorithm has better performance on Set R and Set RC in some instances, which means K-mean Clustering has an impact on Set R and Set RC due to the paths that are arranged into orderly groups. Moreover, K-mean Clustering has not affected Set C due to all the customer location coordinates being clustered. It has been shown that adding K-mean Clustering and providing Stas crossover efficiently contributes to the performance. The bolder results show the best performance in minimizing the number of vehicles and the total distance traveled.

Table 1
Comparison of the crossover operator performance on instances of the Solomon benchmark with Set R type 1 and type 2

Instance	Minimum Distance				Average Distance			
	Without K-mean clustering		K-mean clustering		Without K-mean clustering		K-mean clustering	
R101	Stas1117	2661.54	Stas7111	1176.85	Scatter	2741.76	Stas7111	1203.86
R102	Stas5005	2624.53	Scatter	1207.91	Scatter	2737.38	Stas1117	1236.01
R103	Scatter	2631.42	Stas1117	1272.30	Single	2742.82	Stas7111	1297.40
R104	Single	2687.98	Stas5005	1160.19	Scatter	2740.94	Stas5005	1201.04
R105	Single	2626.82	Single	1255.75	Scatter	2712.62	Stas1117	1274.15
R106	Single	2638.98	Stas1117	1177.49	Stas5005	2721.67	Scatter	1194.37
R107	Single	2655.12	Scatter	1269.40	Single	2753.53	Single	1280.79
R108	Stas5005	2658.86	Stas1117	1149.87	Scatter	2736.72	Scatter	1186.05
R109	Scatter	2597.00	Stas1117	1160.15	stas1117	2711.99	Stas1117	1186.65
R110	Scatter	2636.11	Stas5005	1275.23	Stas7111	2716.47	Stas1117	1314.48
R111	Scatter	2695.78	Stas7111	1176.79	Single	2742.46	Scatter	1197.66
R112	Single	2690.56	Single	1204.94	Stas7111	2739.33	Single	1221.31
R201	Single	2569.49	Scatter	1203.32	Stas1117	2727.72	Stas5005	1225.15
R202	Scatter	2661.34	Single	1252.15	Single	2720.70	Scatter	1287.60
R203	Stas1117	2633.12	Stas5005	1289.26	Scatter	2716.15	Stas5005	1308.46
R204	Single	2678.35	Stas1117	1214.08	Scatter	2739.97	Single	1234.28
R205	Stas1117	2666.82	Stas5005	1138.55	Stas1117	2719.96	Stas1117	1161.81
R206	Scatter	2527.57	Scatter	1168.98	Stas1117	2721.75	Single	1202.86
R207	Single	2657.99	Stas	1221.14	Stas7111	2712.93	Stas	1241.62
R208	Scatter	2652.14	Single	1160.64	Scatter	2704.79	Scatter	1184.85
R209	Scatter	2670.66	Stas5005	1191.89	Scatter	2728.46	Stas1117	1226.74
R210	Stas1117	2666.41	Stas7111	1234.09	Single	2727.43	Stas7111	1258.13
R211	Single	2669.43	Stas1117	1129.32	Scatter	2727.98	Stas1117	1146.94

Table 2
Comparison of the crossover operator performance on instances of the Solomon benchmark with Set C type 1 and type 2

Instance	Minimum Distance				Average Distance			
	Without K-mean clustering		K-mean clustering		Without K-mean clustering		K-mean clustering	
C101	Stas7111	3085.22	Stas7111	931.92	Scatter	3162.29	Stas7111	948.93
C102	Single	3009.74	Stas	946.25	Single	3143.85	Stas	957.96
C103	Single	3068.65	Stas7111	931.45	Scatter	3151.30	Stas7111	942.93
C104	Stas5005	3020.30	Single	903.38	Stas7111	3133.30	Scatter	919.49
C105	Single	2916.59	Stas1117	834.57	Stas7111	3126.38	Stas5005	854.71
C106	Single	3013.41	Stas7111	887.89	Scatter	3123.65	Stas7111	908.07
C107	Scatter	3049.58	Single	880.56	Single	3125.00	Scatter	894.64
C108	Single	3005.27	Stas	856.68	Scatter	3125.57	Stas	872.13
C109	Single	3036.13	Stas5005	917.99	Stas	3126.49	Stas5005	932.37
C201	Single	3030.71	Stas1117	1131.23	Scatter	3183.54	Stas1117	1151.99
C202	Stas1117	3039.12	Stas5005	1146.95	Stas1117	3166.01	Stas5005	1190.10
C203	Stas	3083.54	Single	1169.85	Stas7111	3194.94	Scatter	1202.78
C204	Single	3053.46	Stas7111	1149.25	Stas1117	3178.60	Scatter	1181.14
C205	Scatter	3118.02	Single	1117.09	Stas	3196.28	Scatter	1144.59
C206	Stas7111	3083.63	Scatter	1189.92	Scatter	3191.40	Stas5005	1213.57
C207	Stas5005	3098.08	Single	1109.08	Single	3207.74	Stas5005	1136.67
C208	Scatter	3124.39	Stas	1172.29	Single	3201.71	Scatter	1208.24

Table 3
Comparison of the crossover operator performance on instances of the Solomon benchmark with Set RC type 1 and type 2

Instance	Minimum Distance				Average Distance			
	Without K-mean clustering		K-mean clustering		Without K-mean clustering		K-mean clustering	
RC101	Stas1117	3388.78	Single	1317.54	Scatter	3563.52	Stas7111	1347.91
RC102	Single	3404.87	Scatter	1301.06	Stas7111	3561.81	Scatter	1326.59
RC103	Scatter	3417.54	Stas7111	1231.68	Stas1117	3543.58	Stas7111	1257.56
RC104	Stas5005	3422.21	Stas7111	1312.75	Stas5005	3569.62	Stas7111	1353.25
RC105	Single	3364.39	Stas1117	1233.67	Stas1117	3543.06	Single	1281.75
RC106	Stas7111	3428.35	Stas5005	1290.10	Scatter	3576.80	Stas5005	1323.63
RC107	Single	3467.97	Single	1344.91	Stas1117	3544.65	Scatter	1374.80
RC108	Stas1117	3468.41	Stas7111	1204.56	Stas1117	3559.11	Single	1264.83
RC201	Stas1117	3464.40	Stas1117	1249.62	Single	3574.83	Stas1117	1272.89
RC202	Stas7111	3497.97	Stas	1195.54	Scatter	3590.15	Scatter	1234.26
RC203	Stas7111	3470.91	Single	1240.28	Stas7111	3582.81	Stas1117	1266.12
RC204	Single	3493.42	Stas1117	1196.61	Scatter	3548.85	Stas1117	1231.88
RC205	Stas	3423.66	Stas7111	1287.49	Scatter	3578.89	Stas1117	1339.55
RC206	Scatter	3364.38	Stas1117	1263.11	Stas1117	3566.50	Stas1117	1279.64
RC207	Scatter	3414.01	Scatter	1309.64	Scatter	3570.89	Single	1341.80
RC208	Stas	3436.73	Single	1234.96	Stas	3540.55	Single	1269.57

Table 4
Comparison of the algorithm performance on instances of the Solomon Benchmark for type 1 and type 2

Instance	Best-know solution		Ref.	Proposed Stas crossover with K-mean Clustering	
	vehicles	distance		vehicles	distance
R101	**11**	**1125.00**	May et al. (2021)	10	1176.85
R102	**11**	**1128.00**	May et al. (2021)	10	1207.91
R103	**11**	**1212.00**	May et al. (2021)	10	1272.30
R104	**9**	**1007.31**	Mester et al. (2007)	10	1160.19
R105	11	1260.00	May et al. (2021)	**10**	**1255.75**
R106	12	1251.00	May et al. (2021)	**10**	**1177.49**
R107	**10**	**1104.66**	Shaw (1997)	10	1269.40
R108	**9**	**960.88**	Berger et al. (2001)	10	1149.87
R109	11	1194.73	Homberger & Gehring (1999)	**10**	**1160.15**
R110	**10**	**1104.00**	May et al. (2021)	10	1275.23
R111	**10**	**1096.72**	Rousseau et al. (2002)	10	1176.79
R112	**9**	**982.14**	Gambardella et al. (1999)	10	1204.94
C101	**10**	**828.94**	Rochat & Taillard (1995)	10	931.92
C102	**10**	**828.94**	Rochat & Taillard (1995)	10	946.25
C103	**10**	**828.06**	Rochat & Taillard (1995)	10	931.45
C104	**10**	**824.78**	Rochat & Taillard (1995)	10	903.38
C105	**10**	**828.94**	Rochat & Taillard (1995)	10	834.57
C106	**10**	**828.94**	Rochat & Taillard (1995)	10	887.89
C107	**10**	**828.94**	Rochat & Taillard (1995)	10	880.56
C108	**10**	**828.94**	Rochat & Taillard (1995)	10	856.68
C109	**10**	**828.94**	Rochat & Taillard (1995)	10	917.99
RC101	12	1474.00	May et al. (2021)	**10**	**1317.54**
RC102	11	1338.00	May et al. (2021)	**10**	**1301.06**

Table 4
Comparison of the algorithm performance on instances of the Solomon Benchmark for type 1 and type 2. (Continue)

Instance	Best-know solution		Ref.	Proposed Stas crossover with K-mean Clustering	
	vehicles	distance		vehicles	distance
RC103	11	1250.00	May et al. (2021)	**10**	**1231.68**
RC104	**10**	**1135.48**	Cordeau et al. (2000)	10	1312.75
RC105	11	1274.00	May et al. (2021)	**10**	**1233.67**
RC106	11	**1270.00**	May et al. (2021)	10	1290.10
RC107	11	**1230.48**	Shaw (1997)	10	1344.91
RC108	**10**	**1139.82**	Taillard et al. (1997)	10	1204.56
R201	**2**	**791.00**	May et al. (2021)	10	1203.32
R202	**2**	**740.00**	May et al. (2021)	10	1252.15
R203	**2**	**738.00**	May et al. (2021)	10	1289.26
R204	**2**	**734.00**	May et al. (2021)	10	1214.08
R205	**2**	**726.00**	May et al. (2021)	10	1138.55
R206	**2**	**728.00**	May et al. (2021)	10	1168.98
R207	**2**	**742.00**	May et al. (2021)	10	1221.14
R208	**2**	**732.00**	May et al. (2021)	10	1160.64
R209	**2**	**733.00**	May et al. (2021)	10	1191.89
R210	**2**	**732.00**	May et al. (2021)	10	1234.09
R211	**2**	**751.00**	May et al. (2021)	10	1129.32
C201	**3**	**591.56**	Rochat & Taillard (1995)	10	1131.23
C202	**3**	**591.56**	Rochat & Taillard (1995)	10	1146.95
C203	**3**	**591.17**	Rochat & Taillard (1995)	10	1169.85
C204	**3**	**590.60**	Rochat & Taillard (1995)	10	1149.25
C205	**3**	**588.88**	Rochat & Taillard (1995)	10	1117.09
C206	**3**	**588.49**	Rochat & Taillard (1995)	10	1189.92
C207	**3**	**588.29**	Rochat & Taillard (1995)	10	1109.08
C208	**3**	**588.32**	Rochat & Taillard (1995)	10	1172.29
RC201	**2**	**708.00**	May et al. (2021)	10	1249.62
RC202	**2**	**717.00**	May et al. (2021)	10	1195.54
RC203	**2**	**722.00**	May et al. (2021)	10	1240.28
RC204	**2**	**711.00**	May et al. (2021)	10	1196.61
RC205	**2**	**713.00**	May et al. (2021)	10	1287.49
RC206	**2**	**718.00**	May et al. (2021)	10	1263.11
RC207	**2**	**718.00**	May et al. (2021)	10	1309.64
RC208	**2**	**717.00**	May et al. (2021)	10	1234.96

5. Conclusion

In this study, we used Stas crossover in GA to solve VRPTW by developing the problem with K-mean clustering. Stas crossover performs better than Single point crossover, Two points crossover, Arithmetic crossover, and Scatter crossover. However, the probability area size adjustment of Stas crossover affects the results. It is shown in better performance, too. In VRPTW problems, we divided the results into four parts, including minimum distance without K-mean clustering, minimum distance with K-mean clustering, average distance without K-mean clustering, and average distance with K-mean clustering. The experiments use the standard Solomon's benchmark problem instances for VRPTW with six problem sets of different problem types. The results of Set R, Set C, and Set RC with type 1 and type 2, K-mean clustering is better than without K-mean clustering for minimum distance and average distance. In term of location and dispersion characteristics of the customer, the paths with K-mean clustering are arranged into groups and are orderly, but the paths without K-mean clustering are disordered After that, this paper shows the comparison of the crossover operator performance on instances of Solomon benchmark, and appropriate crossover operators are recommended for each type of problem.

References

Ahmed, Z.H., Maleki, F., Yousefikhoshbakht, M., & Haron, H. (2023). Solving the vehicle routing problem with time windows using modified football game algorithm. *Egyptian Informatics Journal, 24*, 1 – 13.

Alfiyatin, A.N., Mahmudy, W.F., & Anggodo, Y.P. (2018). K-Means Clustering and Genetic Algorithm to Solve Vehicle Routing Problem with Time Windows Problem. *Indonesian Journal of Electrical Engineering and Computer Science, 11*(2), 462 – 468, ISSN: 2502-4752, doi: 10.11591/ijeecs.v11.i2.

Amini, Sh. (2011). A Novel PSO For Solving The VRPTW With Real Case Study. *Proceedings of the 2011 International Conference on Industrial Engineering and Operations Management*, 562 – 567, ISBN: 978-0-9808251-0-7.

Ariyani, A.K., Mahmudy, W.F., & Anggodo, Y.P. (2018). Hybrid Genetic Algorithms and Simulated Annealing for Multi-trip Vehicle Routing Problem with Time Windows. *International Journal of Electrical and Computer Engineering (IJECE), 8*(6), 4713 – 4723, ISSN: 2088-8708, doi: 10.11591/ijece.v8i6.

Berger, J., Barkaoui, M., & Bräysy,O. (2001). A Parallel Hybrid Genetic Algorithm for the Vehicle Routing Problem with Time Windows. *Working paper, Defense Research Establishment Valcartier*, Canada.

Cordeau, J.-F., Laporte, G., & Mercier, A. (2000). A Unified Tabu Search Heuristic for Vehicle Routing Problems with Time Windows. *Working Paper CRT-00-03, Centre for Research on Transportation, Montreal*, Canada.

Gambardella, L.M. (2000). MACS-VRPTW: A Multiple Ant Colony Optimization System for Vehicle Routing Problems with Time Windows (VRPTW). Retrieved from https://people.idsia.ch/~luca/macs-vrptw/solutions/welcome.htm

Gambardella, L. M., Taillard, E., & Agazzi, G. (1999). MACS-VRPTW: A Multiple Ant Colony System for Vehicle Routing Problems with Time Windows. in *New Ideas in Optimization*, Corne, D., Dorigo, M., & Glover, F. (eds), 63-76, McGraw-Hill, London.

Ghani, N.E.A., Shariff, S. S.R., & Zahari, S.M. (2016). An Alternative Algorithm for Vehicle Routing Problem with Time Windows for Daily Deliveries. *Advances in Pure Mathematics, 6*, 342-350. http://dx.doi.org/10.4236/apm.2016.65025

Goldberg, D.E., & Holland, J.H. (1988). Genetic Algorithms and Machine Learning. *Machine Learning, 3*, 95 – 99.

Goldberg, D. E. (1989). *Genetic Algorithms in Search, Optimization & Machine Learning*. Pearson Education Pvt. Ltd., Singapore

Holland, J.H. (1992). Genetic Algorithm. *Scientific American, 267*(1), 66 – 73.

Homberger, J., & Gehring, H. (1999). Two Evolutionary Metaheuristics For The Vehicle Routing Problem With Time Windows. I*nfor*, 37, 297-318.

Kallehauge, B., Larsen, J., Madsen O.B.G., & Solomon, M.M. (2005). The Vehicle Routing Problem with Time Windows. *Column Generation*, 67 – 98, Springer, New York. ISBN 978-0-387-25485-2.

Kinoshita, T., & Uchiya, T. (2021). Diversity Maintenance Method Using Multiple Crossover in Genetic Algorithm for VRPTW. *2021 IEEE 10th Global Conference on Consumer Electronics (GCCE)*, 563 – 565, doi: 10.1109/GCCE53005.2021.9621864.

Kosolsombat, S., Ratanavilisagul, Ch. (2022). Modified ant colony optimization with selecting and elimination customer and re-initialization for VRPTW. *Bulletin of Electrical Engineering and Informatics, 11*(6), 3471 – 3482, ISSN: 2302-9285, doi: 10.11591/eei.v11i6.3943.

May, A.T., Jariyavajee, Ch., & Polvichai, J. (2021). An Improved Genetic Algorithm for Vehicle Routing Problem with Hard Time Windows. *Proc. of the International Conference on Electrical, Computer and Energy Technologies (ICECET)*, 1 – 6, DOI: 10.1109/ICECET52533.2021.9698698.

Mester, D., Braysy, O., & Dullaert, W. (2007). A multi-parametric evolution strategies algorithm for vehicle routing problems. *Expert Systems with Applications, 32*(2), 508-517. Advance online publication. https://doi.org/10.1016/j.eswa.2005.12.014

Mohammadi, M., & Mahmoodian, N. (2022). A Simulated Annealing Approach (SA) to Vehicle Routing Problem with Time Windows (VRPTW). *2022 8th International Conference on Control, Instrumentation and Automation (ICCIA)*, 1 – 6, doi: 10.1109/ICCIA54998.2022.9737187.

Rochat, Y., & Taillard, É.D. (1995) Probabilistic diversification and intensification in local search for vehicle routing. *J Heuristics 1*, 147–167. https://doi.org/10.1007/BF02430370

Rousseau, LM., Gendreau, M. & Pesant, G. (2002). Using Constraint-Based Operators to Solve the Vehicle Routing Problem with Time Windows. *Journal of Heuristics 8*, 43–58, https://doi.org/10.1023/A:1013661617536

Shaw, P. (1997). A new local search algorithm providing high quality solutions to vehicle routing problems. *APES Group, Dept of Computer Science*, University of Strathclyde, Glasgow, Scotland, UK, 46.

Shi, W., & Weise, T. (2013). An Initialized ACO for the VRPTW. *Intelligent Data Engineering and Automated Learning – IDEAL 2013*, LNCS 8206, 93–100, Springer-Verlag Berlin Heidelberg 2013.

Solomon, M.M. (1987). Algorithms for the vehicle routing and scheduling problems with time window constraints. *Operations Research, 35*, 254 – 265.

Solomon, M.M., & Desrosiers, J. (1998). Time Window Constrained Routing and Scheduling Problems. *Transportation Science,22*(1), 1 – 11.

Solomon, M.M. (2005). Best Known Solutions Identified by Heuristics, Northeastern University, Massachusetts, Boston. Retrieved from http://web.cba.neu.edu/~solomon/ heuristic.htm.

Taillard, E., Badeau, P., Gendreau, M., Guertin, F., & Potvin, J.-Y. (1997. A Tabu Search Heuristic for the Vehicle Routing Problem with Soft Time Windows. *Transportation Science, 31*(2), 170-186.https://doi.org/10.1287/trsc.31.2.170

Thangiah, S.R. (1995). Vehicle Routing with Time Windows using Genetic Algorithms. *Applications Handbook of Genetic Algorithms: New Frontiers*, 253 – 278, doi:10.1201/9781420050073.ch11

Villalba, A.F.L., & Rotta, E.C.G.L. (2022). Clustering and heuristics algorithm for the vehicle routing problem with time windows. *International Journal of Industrial Engineering Computations, 13,* 165 – 184, doi: 10.5267/j.ijiec.2021.12.002.

Yousefi, H., Tavakkoli-Moghaddam, R., Oliaei, M.T.B., Mohammadi, M., & Mozaffari, A. (2017). Solving a bi-objective vehicle routing problem under uncertainty by a revised multi choice goal programming approach. *International Journal of Industrial Engineering Computations, 8*, 283 – 302, doi: 10.5267/j.ijiec.2017.1.003.