Endüstri Mühendisliği / Journal of Industrial Engineering

https://dergipark.org.tr/tr/pub/endustrimuhendisligi

A GREY WOLF OPTIMIZER ALGORITHM FOR THE VEHICLE ROUTING PROBLEM WITH TIME WINDOWS AND SIMULTANEOUS PICK-UPS AND DELIVERIES

Milad FARAMARZZADEH[1] , Şener AKPINAR[2*]

[1]Dokuz Eylül University, Industrial Engineering Department, İzmir, Turkey
ORCID No: https://orcid.org/ 0000-0002-3092-0073
[2]Dokuz Eylül University, Industrial Engineering Department, İzmir, Turkey
ORCID No : https://orcid.org/0000-0001-8115-7330

Keywords

Vehicle routing problem with pick-ups and deliveries and time windows, Grey Wolf Optimizer algorithm, K-means algorithm, Variable Neighbourhood Search algorithm

Abstract

The vehicle routing problem with time windows and simultaneous pick-ups and deliveries (VRPTWSPD) is one of the main distribution planning problems. VRPTWSPD aims to find the best distribution plan that minimizes the number of vehicles used and the total travelled distance. Due to the NP-Hard nature of the VRPTWSPD, large-scale real world instances cannot be solved to optimality within acceptable computational times. Therefore, it is necessary to develop approximation algorithms to tackle the VRPTWSPD as effectively as possible. The main goal of this paper is also to develop an approximation algorithm for the VRPTWSPD and a Grey Wolf Optimizer (GWO) algorithm is designed accordingly. The designed algorithm starts its search with a group of solutions constructed through the K-means algorithm. Moreover, it is aimed the proposed GWO algorithm to intensify its search through a local search algorithm based on the Variable Neighbourhood Search (VNS) algorithm. The performance evaluation tests of the developed GWO algorithm were done on

*Sorumlu yazar; e-posta: sener.akpinar@deu.edu.tr

doi : https://doi.org/10.46465/endustrimuhendisligi.1232511

the standard benchmark sets which is taken from the related literature. Computational results indicate that the proposed GWO algorithm has a satisfactory performance in solving VRPTWSPD instances.

ZAMAN PENCERELI VE TOPLAMALI VE DAĞITIMLI ARAÇ ROTALAMA PROBLEMI IÇIN BIR GRI KURT OPTIMIZASYON ALGORITMASI

Anahtar Kelimeler

Zaman pencereli ve toplamalı ve dağıtımlı araç rotalama problemi, Gri kurt optimizasyon algoritması, K-ortalamalar algoritması, Değişken komşuluk arama algoritması.

Öz

Zaman pencereli ve toplamalı ve dağıtımlı araç rotalama problemi (ZPTDARP) temel dağıtım planlama problemlerinden biridir. ZPTDARP, kullanılan araç sayısını ve toplam seyahat mesafesini en aza indiren en iyi dağıtım planını bulmayı amaçlar. ZPTDARP'nin NP-Zor yapısı nedeniyle, büyük ölçekli gerçek dünya örnekleri, kabul edilebilir hesaplama süreleri içinde optimal olarak çözülemezler. Bu nedenle, ZPTDARP'yi mümkün olduğunca etkin bir şekilde çözmek için yaklaşım algoritmaları geliştirmek gerekmektedir. Bu çalışmanın temel amacı da ZPTDARP için bir yaklaşım algoritması geliştirmektir ve bu doğrultuda bir Gri Kurt Optimizasyon (GKO) algoritması tasarlanmıştır. Tasarlanan algoritma, aramaya K-ortalamalar algoritması aracılığıyla oluşturulan bir grup çözümle başlar. Ayrıca, önerilen GWO algoritmasının, Değişken Komşuluk Arama (VNS) algoritmasına dayalı bir yerel arama algoritması aracılığıyla aramasını yoğunlaştırması amaçlanmaktadır. Geliştirilen Gri Kurt Optimizasyon algoritmasının performans değerlendirme testleri, ilgili literatürden alınan standart kıyaslama setleri üzerinde yapılmıştır. Hesaplamalı sonuçlar, önerilen GKO algoritmasının ZPTDARP örneklerini çözmede tatmin edici bir performansa sahip olduğunu göstermektedir.

Araştırma Makalesi		Research Article	
Başvuru Tarihi	: 11.01.2023	Submission Date	: 11.01.2023
Kabul Tarihi	: 01.05.2023	Accepted Date	: 01.05.2023

1. Introduction

The classical vehicle routing problem (VRP) (Prins, 2004) seeks for the minimum cost routes such that each route starts from the depot and ends at the same depot. Besides the cost minimization logistic companies have to keep satisfy the customers with their known demands. Accordingly, vehicles must consider customers' restrictions while serving them. This problem can be studied in further to more complicated constraints which can be solved with different methods.

Recently, logistics companies have been challenged by economic globalization. To have better performance, companies have to minimize their cost such as repairing vehicles, transportation, and so on. Usually, customers have demands, including delivery and pick up. Considering the location is fundamental in logistics companies also at the same time, logistics companies are willing to have fewer vehicles and at the same time fewer workers. Moreover, the logistics companies to have proper performance need to plan a route for each assigned vehicle, vehicles start from the depot to deliver or pick up the demands and end again at the same depot. Usually, this problem is called the pickup and delivery problem, which is applied in many sectors, such as food delivery, health care services, etc. If the orders are with pick-up requests and delivery requests simultaneously, this problem is defined as a simultaneous pickup–delivery (VRPSPD) (Subramanian, Uchoa and Ochi, 2010 May).

In real case studies another constraint of VRPSPD might be serving time and the availability of customers, for example, if a customer is available from 8:15 AM to 11:45 AM, it is called a time window. The VRPSPD can be extended by adding time windows and service time which is called vehicle routing problem with simultaneous pickup delivery with time window (VRPTWSPD) (Angelelli and Mansini, 2003). VRPTWSPD is defined as several customers that need to be serviced with known delivery and pick up time and how to send the vehicles with specific capacity and minimum number from the distribution centre (DC) and also deliver the goods with traveling cost. Considering the simultaneous delivery and pick-up of a vehicle, the less number of vehicles and the minimum travel cost are needed to be concerned. This current paper aims to design a GWO algorithm to be able to solve the VRPTWSPD since the original algorithms were designed to solve continuous optimization problems (Mirjalili, Mirjalili and Lewis, 2014). The designed algorithm starts its search with a group of solutions constructed through the K-means algorithm and realizes the leadership hierarchy of grey wolves through the crossover. Additionally, the algorithm has been enhanced by incorporating the Variable Neighbourhood Search (VNS) algorithm as a local search algorithm.

Developing effective solution procedure for the VRP variants is still a challenging research field, since the capacitated VRP which is the basic problem in the literature on vehicle routing is still far from being satisfactorily solved (Semet, Toth, and Vigo, 2014). Therefore, we have concerned ourselves to develop a search algorithm for the VRPTWSPD on the basis of the GWO algorithm, which is a recent and popular metaheuristic (Faris, Aljarah, Al-Betar and Mirjalili, 2018. Furthermore, we have the motivation to present an algorithmic proposal as a combinatorial variant of the GWO algorithm within the context of this paper. Moreover, the rationale of combining the GWO algorithm with the VNS algorithm comes from the satisfactory performance of the VNS algorithm in solving the VRP variants (Hansen, Mladenović, and Moreno Pérez, 2010).

The remainder of this paper is organized as fallows. The relevant literature on VRPTWSPD is given in Section 2. The developed algorithm is described in Section 3. Computational studies are given in Section 4. Finally, the paper concludes with some remarks in Section 5.

2. Literature Review

The VRPTWSPD is one of the current challenges in transportation systems. The first algorithm (Branch and Price) has been proposed by (Angelelli and Mansini, 2002) to solve VRPTWSPD, although there were some limitations in their method, it was very time-consuming and not able to get an acceptable solution, especially for NP-hard problems. An improved genetic algorithm has been developed by (Cao and Lai, 2007 August) and the obtained results indicated that the algorithm can obtain an almost optimal solution for VRPSPD. Another algorithm called multi-agent colonies has been presented by (Boubahri, Addouche and El Mhamedi, 2011, March) to solve the VRPTWSPD, even though the author did not show any numerical results. A set of VRPTWSPD instances- extended from well-known Solomon's benchmark instances- was solved with a co-evolution genetic algorithm by (Wang and Chen, 2012). The proposed co-evolution genetic algorithm gave better results than the genetic algorithm. Also (Wang and Chen, 2012) showed that the Cplex solver is poor to solve even small instances compared with the co-evolution genetic algorithm. The same benchmark set was also solved via a parallel simulated annealing algorithm proposed by (Wang, Mu, Zhao and Sutherland, 2015), which had the same results as (Wang and Chen, 2012) obtained for the majority of them, and for some of them, the parallel simulated annealing algorithm had better performance compared against the co-evolution genetic algorithm (Wang and Chen, 2012). Afterward, another algorithm having a tabu search base framework has been proposed by (Shi, Boudouh and Grunder, 2018) to solve VRPTWSPD and the authors obtained better solutions than (Wang et al., 2015) for just some of the instances, most of them are worse than (Wang et al., 2015) however they didn't show all experiments. Recently, an adaptive large neighbourhood search with path relinking (ALNS-PR) algorithm has been proposed by (Hof and Schneider, 2019) to solve the class of VRPSPD, including the VRPTWSPD. Their algorithm had much better performance than previous algorithms of (Wang and Chen, 2012), (Wang et al., 2015) and (Shi et al., 2018) on the benchmark instances. Their numerical experimental results showed that ALNS-PR could provide much better solutions compared with other algorithms. Furthermore (Shi, Zhou, Boudouh and Grunder, 2020) designed a learning-based two-stage algorithm, also called as variable neighbourhood search and bi-structure based tabu search algorithm (BSTS). Also their algorithm obtained same or better results than the previous results for VRPTWSPD. However, significant solutions in the literature have been published for the VRPTWSPD problem. We proposed a GWO algorithm. In this algorithm we used K-means algorithm to cluster the constructed of grey wolves and we improved with applying VNS which can be better and different method in this study.

Since the current paper presents a GWO algorithm for a VRP variant, an advanced search is done on the web of science (WoS) website by using the keywords ‘vehicle routing’ and ‘grey wolf’ it is possible to reach only two papers (Korayem, Khorsid and Kassem, 2015, May; Li and Wang, 2020). To the best of our knowledge, this paper is the first attempt to implement the GWO algorithm for a VRP variant with the combination of the VNS algorithm. Moreover, to the best of our knowledge, this current paper is the first one that starts the GWO algorithm with a clustering algorithm for a VRP variant.

3. The Grey Wolf Optimizer Algorithm

The grey wolf optimizer algorithm is a new bio-inspired algorithm of meta-heuristic which is proposed by (Mirjalili et al., 2014). This algorithm like other algorithms of swarm intelligence uses social behaviour to find the optimal solution. GWO algorithm is based on the hunting behaviour of a population of grey wolves. Grey wolves are considered apex predators, meaning that they live in a pack with a strict leadership hierarchy. As Figure 1 shows the leaders are called alpha (α) and are responsible to make the decision and the best to manage. The next level of this pack is called beta (β). Beta helps in making the decision or other pack activities they are usually considered as an advisor in the pack. The next group is the delta (δ). They play the role of scapegoats, and always have to submit to alpha and beta but they are dominant wolves the omega. The last group is omega, they have different tasks such as taking care of the pack in case of any threat, and watching boundaries of the territory.

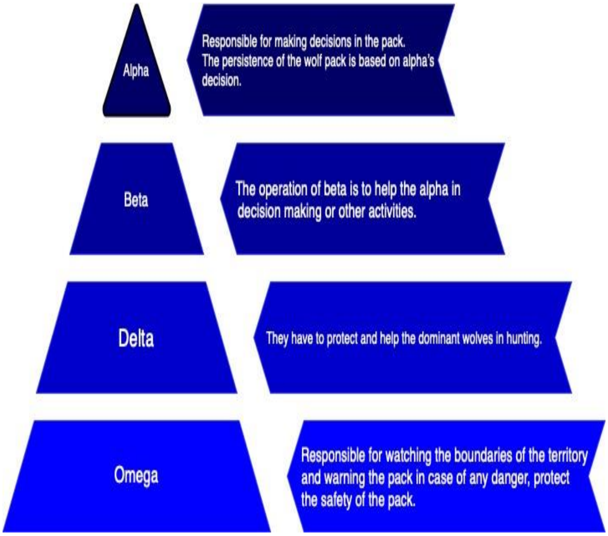

Figure 1.Responsibilities of the Wolves in the Pack

3.1. Grey Wolf Optimizer Algorithm to Solve VRPTWSP

This section presents the proposed Grey Wolf Optimizer (GWO) to solve the VRPTWSPD in combination with the K-Means and Variable Neighbourhood Search (VNS) algorithms. As shown in Figure 2, the proposed GWO algorithm used K-means to build the initial wolves' positions instead of random initialization, since the K-means (MacQueen, 1967 June) can cluster properly and quickly. Then the proposed GWO algorithm realizes the leadership hierarchy of grey wolves through the crossover. This mechanism provides the proposed GWO algorithm to diversify its search. Moreover, through the VNS algorithm, the proposed GWO algorithm would be able to intensify its search. The prosperity VNS algorithm motivates the researchers to this search mechanism since the algorithm provides a satisfactory level of intensification (Hansen and Mladenović, 2003). This study complied with research and publication ethics.

```
Input: Population size (Ps); Crossover rate (R_c); Mutation Rate of exchange two customers (m_1); Mutation Rate of Exchange
adjacent customers (m_2); Mutation Rate of Migration a customer (m_3); maximum iteration (MaxIter)
// Initialization
Initialize BestWolf with its fitness value as BestFitness
Generate initial population (Ps number of solutions) via K-means
// Fitness Evaluation
Evaluate the fitness of wolves in the population and update BestWolf and BestFitness
Determine the best wolf (α), the second best wolf (β), the third best wolf (δ) and the remaining wolves (w) of the population
// Main Loop
for i=1 to MaxIter do
    // Crossover
    for j=1 to Ps do
        Lead wolf j via crossover to generate wolf j'
        Replace wolf j by wolf j' to generate new population
    endfor
    // Fitness Evaluation
    Evaluate the fitness of wolves in the new population and update BestWolf and BestFitness
    Determine α, β, δ and w wolves of the new population
    // Apply local search to every wolf
    for j=1 to Ps do
        // Exchange two customers
        r_1 = rand()
        if r_1 ≥ m_1
            randomly exchange two customers of wolf j to generate wolf j'
            if wolf j' is feasible
                Evaluate the fitness of wolf j' and update BestWolf and BestFitness
                Replace wolf j by wolf j'
            endif
        endif
        // Exchange adjacent customers
        r_2 = rand()
        if r_2 ≥ m_2
            randomly select a customer of wolf j and exchange with its adjacent to generate wolf j'
            if wolf j' is feasible
                Evaluate the fitness of wolf j' and update BestWolf and BestFitness
                Replace wolf j by wolf j'
            endif

        endif
        // Migrate a customer
        r_3 = rand()
        if r_3 ≥ m_3
            randomly select a customer of wolf j and insert it into a randomly determined position to generate wolf j'
            if wolf j' is feasible
                Evaluate the fitness of wolf j' and update BestWolf and BestFitness
                Replace wolf j by wolf j'
            endif
        endif
    endfor
    Determine α, β, δ and w wolves of the new population
endfor
return BestWolf and BestFitness
```

Figure 2. Pseudocode of the Proposed GWO Algorithm

3.1.1. Solution Representation and Initial Solutions

A solution for a VRPTWSPD with 12 customers, one depot and 2 vehicles is illustrated in Figure 3. The zeros in the solution representation correspond the depot, and the other digits between two zeros correspond the route of the customers that are required to be served by a vehicle.

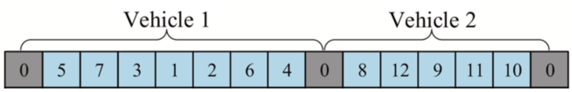

Figure 3. Coded Representation of a Solution

The solution given in Figure 3 contains two routes because of the number of vehicles. The first (second) route therefore the first (second) vehicle starts from the depot and serves the customers in the order of 5-7-3-1-2-6-4 (8-12-9-11-10) and ends in the depot again. The proposed GWO algorithm assigns the customers to the routes through the guidance of the K-Means algorithm (Shi et al., 2018; Kapil, Chawla and Ansari, 2016 December; Barbosa, Christo and Costa, 2015). The mechanism of K-Means algorithm (see Figure 4) first, K points or centroids define the number of clusters, they are chosen randomly. After that each data point assigns its closest distance based on the value of Euclidean distance and the K clusters will be predefined then recalculate the new cluster centre.

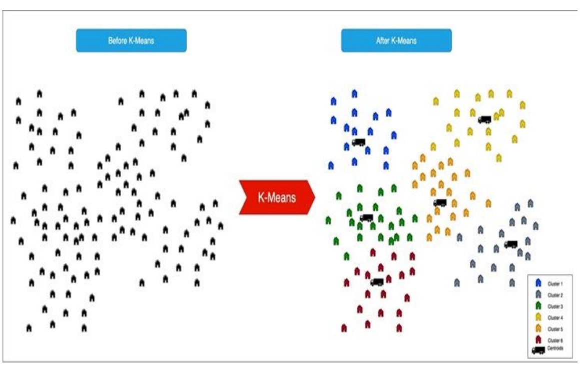

Figure 4. Visualization of the K-Means Algorithm

After assigning the customers to the routes, the proposed GWO algorithm randomly determines the serving order of the customers belonging to a route by considering the time window and capacity constraints of the problem. In essence, the proposed GWO algorithm generates a predefined number of solutions through the guidance of the K-Means algorithm at its first step to manipulate and improve through its succeeding steps.

3.1.2. Crossover

The main purpose of the crossover within the context of the proposed GWO algorithm is to generate offspring through the information of parents. There are many ways to do crossovers such as two-point, uniform, single-point, reduced surrogate and cycle, k-point, shuffle, partially matched, and exchanging two segments of different routes which was proposed by (Taillard, Badeau, Gendreau, Guertin and Potvin, 1997). The crossover of the proposed GWO algorithm (see Figure 5) uses three random cut-off points to make four segments which are α, β, δ, and ω. The largest segment belongs to α which is the leader and strongest wolf in the hierarchy pack and the next biggest segment is β which is the second level of the pack and the next level is δ which is the third best solution, finally, the last segment belongs to ω which is subjected to α, β and δ wolves.

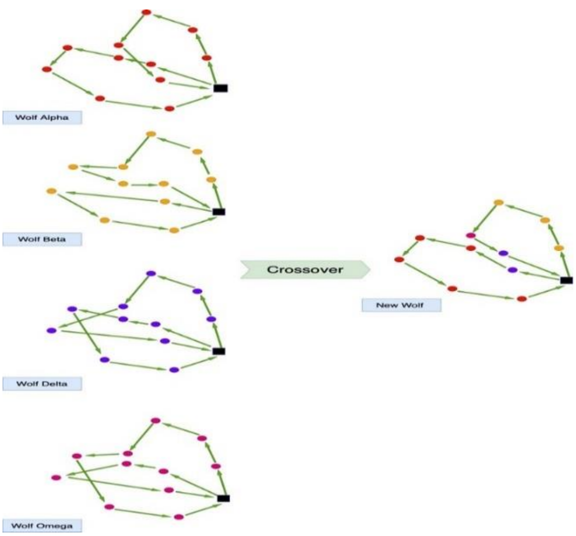

Figure 5.Crossover

3.1.3. Local Search

Neighbourhood search algorithms also called local search algorithms are based on exploring the neighbourhoods of a solution in the search space. A neighbourhood search algorithm tries iteratively to improve the solution on hand by searching its neighbours. Each solution S has a neighbourhood associated with $N(S)$, which consists of the solutions that can be reached from S by a move. At each iteration, the local search procedure seeks an improved solution S' in the neighbourhood $N(S)$ of the current solution S, until no further improvement is found. Generally, meta-heuristics try to escape the trap of local optimal, and Variable Neighbourhood Search (Hansen and Mladenović, 2003) is an improved technique that prevents stuck in the local optimum by changing the structure of the neighbourhood.

Variable Neighbourhood Search (VNS) starts from the current solution and moves to the next solution if it is better than the previous solution, therefore the structure of the neighbourhood can be changed with any movement. Commonly the process of local search stops with maximum computational time (number of iterations, real-time) or when there is no improvement in search space. The neighbourhoods considered are nested and they are based on a set of standard or base moves. Given a base neighbourhood structure N, the series of nested neighbourhood structures are defined as follows. The first neighbourhood structure is the base one $N1(S) = N(S)$.

Local search is a significant method that is used for solving the computational optimization problem. This method starts to form a solution and iteratively moves to a neighbour solution, additionally any candidate solution move to its neighbour solution so the optimal solution can be recognized by moving each candidate solution. The proposed GWO algorithm's search application starts after the initial solution which is created by K-means and GWO. The process in search space continues to find the best solution, and also change the composition of neighbourhoods which is essential to obtain the optimal objective. To restructure our neighbourhood through local search we used a variety of operates but we show just three effective operators: Exchange two customers, Exchange adjacent customers, and Migration of a customer. Exchange two customers (see Figure 6) randomly chooses two customers and tries to swap to find a better solution.

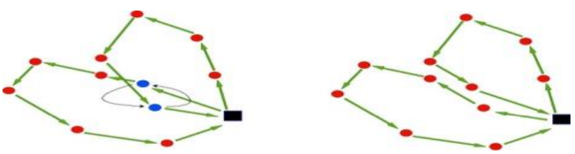

Figure 6. Exchange Two Customers

Exchange adjacent customers (see Figure 7) randomly chooses a customer and swaps it with side neighbours to find an improved feasible solution. If an improvement could not be achieved, the operator tries to swap the randomly chosen customer with the next neighbour. The operator can swap until the second adjacent from each side without violating the feasibility.

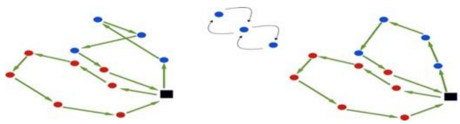

Figure 7.Exchange Adjacent Customers

Migrate of a customer (see Figure 8) randomly chooses a customer and tries to replace it into another better position by taking the time window and capacity constraints into account.

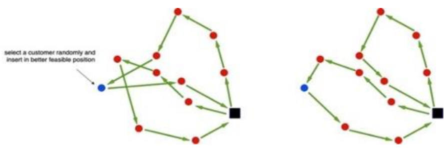

Figure 8.Migrate of a Customer

4. Computational Study

We tested our proposed algorithm on a VRPTWSPD benchmark data set which was generated by (Shiet al., 2020). This benchmark set involves nine small-scale instances (three 10-customer instances, three 25-customer instances and three 50-customer instances) and fifty-six medium-scale instances (fifty-six 100-customer instances). The proposed GWO algorithm is compared against the best solutions of five methods given as a co-evolution genetic algorithm (CoGA) (Wang and Chen, 2012) a parallelized simulated annealing (p-SA) (Wang et al., 2015), the efficient tabu search based procedure (ETSP) (Shi et al., 2018), an adaptive large neighbourhood search with path relinking (LANS-PR) (Hof and Schneider, 2019) and a bi-structure based tabu search (BSTS) (Shiet al., 2020). The proposed GWO algorithm was coded in PYTHON and the results were obtained by running the coded GWO on an Intel core i5, 2.5 GHz processor and 8 GB RAM personal computer. The parameters' values obtained as a result of preliminary experiments and used in the proposed GWO algorithm are given in Table 1.

Table1.
Values of the control parameters

Parameter	Definition	Value
Ps	Population size	{20,50}
R_c	Crossover rate	0.5
m_1	Rate of exchange two customers	0.7
m_2	Rate of exchange adjacent customers	0.9
m_3	Rate of migration a customer	0.9

4.1. Experimental Results

This section compares the performance of the proposed GWO algorithm against the aforementioned algorithms in Tables 2, 3, 4, 5 and 6. The number of vehicles (NV) used and the total travelled distance (TD) are the two performance measures used within the context of this comparison as the related literature did.

Table 2 firstly compares the performance of the proposed GWO algorithm against the CoGA. As can be clearly seen from Table 2, the proposed GWO algorithm can do the distributions related to the problems of rdp110, rcdp101, rcdp102, rcdp106, rcdp107, rdp209 with decreased NVs, however the TDs are higher than the CoGA did. Additionally, GWO can identify better TDs for the problems of rcdp201, rcdp202 and rcdp208 than CoGA.

Table 2.
Comparison of GWO Against CoGa

Instance ID	CoGA		GWO		Gap*	
	NV	TD	NV	TD	NV	TD (%)
rdp101	19	1653.5	20	1731.5	1	4.72
rdp109	12	1160	13	1239.2	1	6.83
rdp110	12	1116.9	11	1175.1	-1	5.21
cdp101	11	1001.9	11	1068.2	0	6.62
cdp102	10	961.3	11	1063.4	1	10.62
cdp103	10	891.6	11	990.3	1	11.07
cdp104	10	878.9	10	994	0	13.10
cdp109	10	940.4	11	1098.4	1	16.80
rcdp101	15	1652.9	14	1716.4	-1	3.84
rcdp102	14	1494	13	1622.5	-1	8.60
rcdp103	12	1338.7	12	1343.2	0	0.34
rcdp105	14	1581.2	14	1690.1	0	6.89
rcdp106	13	1422.8	12	1513.5	-1	6.37
rcdp107	12	1282.1	11	1323.2	-1	3.21
rdp201	4	1280.4	4	1324.2	0	3.42
rdp202	4	1100.9	4	1136.4	0	3.22
rdp203	3	950.7	3	1010.9	0	6.33
rdp204	3	775.2	3	813	0	4.88
rdp205	3	1064.4	3	1114.7	0	4.73
rdp206	3	961.3	3	991.9	0	3.18
rdp207	3	835	3	900.9	0	7.89
rdp208	3	718.5	3	765.3	0	6.51
rdp209	4	930.2	3	1033	-1	11.05
rdp210	3	983.7	3	1028.8	0	4.58
rdp211	3	839.6	3	863	0	2.79
cdp201	3	591.5	3	591.5	0	0.00
cdp202	3	591.5	3	591.5	0	0.00
cdp203	3	591.1	3	591.1	0	0.00
cdp204	3	590.6	3	599.3	0	1.47
cdp205	3	588.8	3	588.8	0	0.00
cdp206	3	588.4	3	613.3	0	4.23
cdp207	3	588.2	3	600.6	0	2.11
cdp208	3	588.3	3	588.3	0	0.00
rcdp201	4	1587.9	4	1406	0	-11.46
rcdp202	4	1211.1	5	1176.2	1	-2.88
rcdp203	4	964.6	4	1073.3	0	11.27
rcdp204	3	822	3	866.3	0	5.39
rcdp205	4	1410.1	4	1424.1	0	0.99
rcdp206	3	1176.8	4	1188.2	1	0.97
rcdp207	4	1036.5	4	1107.3	0	6.83
rcdp208	3	878.5	4	837.8	1	-4.63

$$Gap = \frac{TD_{GWO} - TD_{CoGa}}{TD_{CoGa}} * 100\ \%$$

Average Gap=4.32

The next comparison is done against the p-SA in Table3. With observing the results in Table 3, the proposed GWO can identify better results for two instances of rcdp101 and rcdp106 with regard to NV, for six instances of cdp208, rcdp201, rcdp202, rcdp203, rcdp204 and rcdp208 with regard to TD than p-SA.

Table 3.
Comparison of GWO Against p-SA

Instance ID	p-SA		GWO		Gap*	
	NV	TD	NV	TD	NV	TD (%)
rdp101	19	1660.9	20	1731.5	1	4.25
rdp109	12	1181.9	13	1239.2	1	4.85
rdp110	11	1106.5	11	1175.1	0	6.20
cdp101	11	992.8	11	1068.2	0	7.59
cdp102	10	955.3	11	1063.4	1	11.32
cdp103	10	958.6	11	990.3	1	3.31
cdp104	10	944.7	10	994	0	5.22
cdp109	10	947.9	11	1098.4	1	15.88
rcdp101	15	1659.5	14	1716.4	-1	3.43
rcdp102	13	1522.7	13	1622.5	0	6.55
rcdp103	11	1344.6	12	1343.2	1	-0.10
rcdp105	14	1581.5	14	1690.1	0	6.87
rcdp106	13	1418.1	12	1513.5	-1	6.73
rcdp107	11	1360.1	11	1323.2	0	-2.71
rdp201	4	1286.5	4	1324.2	0	2.93
rdp202	4	1150.3	4	1136.4	0	-1.21
rdp203	3	997.8	3	1010.9	0	1.31
rdp204	2	848	3	813	1	-4.13
rdp205	3	1046	3	1114.7	0	6.57
rdp206	3	959.9	3	991.9	0	3.33
rdp207	2	899.8	3	900.9	1	0.12
rdp208	2	739	3	765.3	1	3.56
rdp209	3	947	3	1033	0	9.08
rdp210	3	1005.1	3	1028.8	0	2.36
rdp211	3	812.4	3	863	0	6.23
cdp201	3	591.5	3	591.5	0	0.00
cdp202	3	591.5	3	591.5	0	0.00
cdp203	3	591.1	3	591.1	0	0.00
cdp204	3	594	3	599.3	0	0.89
cdp205	3	588.8	3	588.8	0	0.00
cdp206	3	588.4	3	613.3	0	4.23
cdp207	3	588.2	3	600.6	0	2.11
cdp208	3	599.3	3	588.3	0	-1.84
rcdp201	4	1513.7	4	1406	0	-7.12
rcdp202	4	1273.2	5	1176.2	1	-7.62
rcdp203	3	1123.5	4	1073.3	1	-4.47
rcdp204	3	897.1	3	866.3	0	-3.43
rcdp205	4	1371	4	1424.1	0	3.87
rcdp206	3	1166.8	4	1188.2	1	1.83
rcdp207	3	1089.8	4	1107.3	1	1.61
rcdp208	3	862.8	4	837.8	1	-2.90

$$Gap = \frac{TD_{GWO} - TD_{p-SA}}{TD_{p-SA}} * 100\ \%$$

Average Gap=2.36

Another comparison is done against the ETSP in Table 4. The proposed GWO algorithm can identify better results for only cdp205 with regard to NV than ETSP. Moreover, the proposed GWO algorithm provides better results for instances cdp204, rcdp201, rcdp202, rcdp203, rcdp204, rcdp207 and rcdp208 with regard to TD than ETSP.

Table 4.
Comparison of GWO against ETSP

Instance ID	ETSP		GWO		Gap*	
	NV	TD	NV	TD	NV	TD (%)
rdp201	4	1268.5	4	1324.2	0	4.39
rdp202	4	1099.6	4	1136.4	0	3.35
rdp203	3	981.4	3	1010.9	0	3.01
rdp204	3	775.9	3	813	0	4.78
rdp205	3	1045.1	3	1114.7	0	6.66
rdp206	3	973.4	3	991.9	0	1.90
rdp207	3	841.2	3	900.9	0	7.10
rdp208	2	740.8	3	765.3	1	3.31
rdp209	3	999	3	1033	0	3.40
rdp210	3	964.5	3	1028.8	0	6.67
rdp211	3	805.5	3	863	0	7.14
cdp201	3	591.5	3	591.5	0	0.00
cdp202	3	591.5	3	591.5	0	0.00
cdp203	3	591.1	3	591.1	0	0.00
cdp204	3	599.8	3	599.3	0	-0.08
cdp205	3	588.8	3	588.8	0	0.00
cdp206	3	588.4	3	613.3	0	4.23
cdp207	3	588.2	3	600.6	0	2.11
cdp208	3	588.3	3	588.3	0	0.00
rcdp201	4	1441.5	4	1406	0	-2.46
rcdp202	4	1216.5	5	1176.2	1	-3.31
rcdp203	3	1106	4	1073.3	1	-2.96
rcdp204	3	900.6	3	866.3	0	-3.81
rcdp205	5	1253.4	4	1424.1	-1	13.62
rcdp206	4	1161.8	4	1188.2	0	2.27
rcdp207	3	1125.5	4	1107.3	1	-1.62
rcdp208	3	873.2	4	837.8	1	-4.05
	$Gap = \frac{TD_{GWO} - TD_{ETSP}}{TD_{ETSP}} * 100\ \%$				Average Gap=2.06	

The fourth comparison is done against the ALNS-PR in Table 5. Table 5 shows the proposed GWO algorithm can achieve remarkable solution for the instance of rcdp101 compared against the ALNS-PR, since the GWO can the distribution with the same NV and a reduced TD. Although the proposed GWO algorithm provides better results for TD but with more number vehicles in the instances of rcdp202, rcdp201, rdp211, rdp204, rdp202, rcdp101.

Table 5.
Comparison of GWO against ALNS-PR

Instance ID	ALNS-PR		GWO		Gap*	
	NV	TD	NV	TD	NV	TD (%)
rdp101	19	1650.8	20	1731.5	1	4.89
rdp109	11	1194.04	13	1239.2	2	3.78
rdp110	10	1148.2	11	1175.1	1	2.34
cdp101	11	976	11	1068.2	0	9.45
cdp102	10	941.4	11	1063.4	1	12.96
cdp103	10	892.9	11	990.3	1	10.91
cdp104	10	871.4	10	994	0	14.07
cdp109	10	910.9	11	1098.4	1	20.58
rcdp101	14	1776.5	14	1716.4	0	-3.38
rcdp102	12	1583.6	13	1622.5	1	2.46
rcdp103	11	1283.5	12	1343.2	1	4.65
rcdp105	14	1548.9	14	1690.1	0	9.12
rcdp106	12	1392.4	12	1513.5	0	8.70
rcdp107	11	1255	11	1323.2	0	5.43
rdp201	4	1253.2	4	1324.2	0	5.67
rdp202	3	1191.7	4	1136.4	1	-4.64
rdp203	3	946.2	3	1010.9	0	6.84
rdp204	2	833	3	813	1	-2.40
rdp205	3	994.4	3	1114.7	0	12.10
rdp206	3	913.6	3	991.9	0	8.57
rdp207	2	890.6	3	900.9	1	1.16
rdp208	2	726.8	3	765.3	1	5.30
rdp209	3	909.1	3	1033	0	13.63
rdp210	3	939.3	3	1028.8	0	9.53
rdp211	2	904.4	3	863	1	-4.58
cdp201	3	591.5	3	591.5	0	0.00
cdp202	3	591.5	3	591.5	0	0.00
cdp203	3	591.1	3	591.1	0	0.00
cdp204	3	590.6	3	599.3	0	1.47
cdp205	3	588.8	3	588.8	0	0.00
cdp206	3	588.4	3	613.3	0	4.23
cdp207	3	588.4	3	600.6	0	2.07
cdp208	3	588.3	3	588.3	0	0.00
rcdp201	4	1406.9	4	1406	0	-0.06
rcdp202	3	1414.5	5	1176.2	2	-16.85
rcdp203	3	1050.6	4	1073.3	1	2.16
rcdp204	3	798.4	3	866.3	0	8.50
rcdp205	4	1297.6	4	1424.1	0	9.75
rcdp206	3	1146.3	4	1188.2	1	3.66
rcdp207	3	1061.8	4	1107.3	1	4.29
rcdp208	3	828.1	4	837.8	1	1.17
$* Gap = \frac{TD_{GWO} - TD_{ALNS-PR}}{TD_{ALNS-PR}} * 100\ \%$					Average Gap =4.33	

Finally, Table 6 compares the proposed GWO algorithm against the BSTS. Table 6 shows that the proposed GWO algorithm can achieve better results in TD for the instance of rcdp201 with the same NV compared against BSTS. Moreover, the proposed GWO algorithm can provide better TD for the instance of rcdp202, rdp204, rdp207, rdp202 and rcdp208 with more vehicles in comparison to BSTS.

Table 6.
Comparison of GWO against BSTS

Instance ID	BSTS		GWO		Gap*	
	NV	TD	NV	TD	NV	TD (%)
rdp101	19	1650.8	20	1731.5	1	4.89
rdp109	11	1224.8	13	1239.2	2	1.18
rdp110	11	1101.3	11	1175.1	0	6.70
cdp101	11	976	11	1068.2	0	9.45
cdp102	10	942.4	11	1063.4	1	12.84
cdp103	10	896.2	11	990.3	1	10.50
cdp104	10	872.3	10	994	0	13.95
cdp109	10	909.2	11	1098.4	1	20.81
rcdp101	14	1708.2	14	1716.4	0	0.48
rcdp102	13	1526.3	13	1622.5	0	6.30
rcdp103	11	1336	12	1343.2	1	0.54
rcdp105	14	1548.3	14	1690.1	0	9.16
rcdp106	12	1408.1	12	1513.5	0	7.49
rcdp107	11	1295.4	11	1323.2	0	2.15
rdp201	4	1254.5	4	1324.2	0	5.56
rdp202	3	1202.2	4	1136.4	1	-5.47
rdp203	3	949.4	3	1010.9	0	6.48
rdp204	2	837.1	3	813	1	-2.88
rdp205	3	1027.4	3	1114.7	0	8.50
rdp206	3	938.6	3	991.9	0	5.68
rdp207	2	912.2	3	900.9	1	-1.24
rdp208	2	737.2	3	765.3	1	3.81
rdp209	3	940.2	3	1033	0	9.87
rdp210	3	945.9	3	1028.8	0	8.76
rdp211	3	805.2	3	863	0	7.18
cdp201	3	591.5	3	591.5	0	0.00
cdp202	3	591.5	3	591.5	0	0.00
cdp203	3	591.1	3	591.1	0	0.00
cdp204	3	599.3	3	599.3	0	0.00
cdp205	3	588.8	3	588.8	0	0.00
cdp206	3	588.4	3	613.3	0	4.23
cdp207	3	588.2	3	600.6	0	2.11
cdp208	3	588.3	3	588.3	0	0.00
rcdp201	4	1437.4	4	1406	0	-2.18
rcdp202	3	1412.5	5	1176.2	2	-16.73
rcdp203	3	1064.9	4	1073.3	1	0.79
rcdp204	3	813.7	3	866.3	0	6.46
rcdp205	4	1316	4	1424.1	0	8.21
rcdp206	3	1154.3	4	1188.2	1	2.94
rcdp207	3	1098.6	4	1107.3	1	0.79
rcdp208	3	843.3	4	837.8	1	-0.65
$* Gap = \frac{TD_{GWO} - TD_{BSTS}}{TD_{BSTS}} * 100\ \%$					Average Gap=3.87	

4.2. Discussions

Computational results indicate that the proposed GWO algorithm has a promising performance in solving the VRPTWSPD however it is not at the desired level. This situation might be a result of an imbalance between the intensification and diversification strategies employed by the proposed GWO algorithm. This imbalance could also be induced a low-level synergy between the intensification and diversification strategies. To overcome this shortage, as a further improvement on the proposed GWO algorithm, different neighbourhood mechanisms may be embedded in the presented algorithmic proposal to provide an improved balance and therefore obtain a high-level synergy between the intensification and diversification strategies. Another possible reason could be starting the algorithm by only one strategy as K-Means algorithm. This potential shortage may be overcome by generating initial solutions via different strategies to start the algorithm's search as diversely as possible in the further versions of the presented algorithmic proposal. As a final note, we can claim that the presented algorithmic proposal is a result of a revered research process although it performed not at the desired level in solving the VRPTWSPD. Furthermore, the proposed GWO algorithm could be evaluated as a beginning of a research direction, since it established the potential to being further improved by means of the provided results.

5. Conclusions

This paper aimed at developing a GWO algorithm to solve the VRPTWSPD. The original GWO was designed to solve the continuous optimization problems and was required to be modified to tackle a combinatorial optimization problem of VRPTWSPD. When designing the GWO algorithm for VRPTWSPD, the first decision point has been arisen as algorithm initialization which involves determining the initial positions of wolves. To response this challenge, we adapted the K-means algorithm to locate the wolves on the search space rather than a random initialization. Another and most important challenging issue was to adapt the leadership hierarchy of grey wolves to a combinatorial optimization problem. To overcome this challenge, we decided to use crossover. Besides realizing the leadership hierarchy of grey wolves, the crossover provided the algorithm to diversify its search during exploring the search space of the problem. Furthermore, we enhanced the proposed GWO algorithm via a VNS based local search to make the algorithm to be able to intensify its search. To evaluate the performance of the proposed GWO algorithm we used a set of standard benchmark set taken from the related literature, and the algorithm compared against five different algorithms designed to solve VRPTWSPD instances. The obtained results indicated that the proposed GWO algorithm is a competitive algorithm in solving VRPTWSPD instances and could be further improved.

Future researches will focus on performance improvement studies of the proposed GWO algorithm and adapting the algorithm to different combinatorial optimization problems.

Conflict of Interest

There is no conflict of interest to declare.

Contribution of Researchers

Milad FARAMARZZADEH and Şener AKPINAR were responsible for conceptualization, methodology, software, data curation, writing- original draft preparation, validation, writing- reviewing and editing. Furthermore, Şener AKPINAR supervised the whole study and edited the article.

References

Angelelli, E., & Mansini, R. (2002). The vehicle routing problem with time windows and simultaneous pick-up and delivery. *In Quantitative approaches to distribution logistics and supply chain management* (pp. 249-267). Springer, Berlin, Heidelberg. Doi: https://doi.org/10.1007/978-3-642-56183-2_15

Angelelli, E., & Mansini, R. (2003). A branch-and-price algorithm for a simultaneous pick-up and delivery problem. *In Quantitative approaches to distribution logistics and supply chain management* (pp. 249-267). Springer, Berlin, Heidelberg. Retrieved from: https://www.academia.edu/es/29065016/A_Branch_and_Price_Algorithm_for_a_Simultaneous_Pick_up_and_Delivery_Problem.

Barbosa, N. D. P., Christo, E. D. S., & Costa, K. A. (2015). Demand forecasting for production planning in a food company. *ARPN Journal of Engineering and Applied Sciences*, *10*(16), 7137-7141. Retrieved from: http://www.arpnjournals.com/jeas/research_papers/rp_2015/jeas_0915_2531.pdf.

Boubahri, L., Addouche, S. A., & El Mhamedi, A. (2011, March). Multi-ant colonies algorithms for the VRPSPDTW. *In 2011 International Conference on Communications, Computing and Control Applications (CCCA)* (pp. 1-6). IEEE. Doi: https://doi.org/10.1109/CCCA.2011.6031488.

Cao, E., & Lai, M. (2007, August). An improved differential evolution algorithm for the vehicle routing problem with simultaneous delivery and pick-up service. *In Third international conference on natural computation (ICNC 2007)* (Vol. 3, pp. 436-440). IEEE. Doi: https://doi.org/10.1109/ICNC.2007.209

Faris, H., Aljarah, I., Al-Betar, M. A., & Mirjalili, S. (2018). Grey wolf optimizer: a review of recent variants and applications. *Neural Computing and Applications*, 30, 413-435. Doi: https://doi.org/10.1007/s00521-017-3272-5.

Hansen, P., & Mladenović, N. (2003). Variable neighborhood search. *In Handbook of metaheuristics* (pp. 145-184). Springer, Boston, MA. Doi: https://doi.org/10.1007/978-3-319-91086-4_3.

Hansen, P., Mladenović, N., & Moreno Pérez, J. A. (2010). Variable neighbourhood search: methods and applications. *Annals of Operations Research*, 175, 367-407. Doi: https://doi.org/10.1007/s10479-009-0657-6.

Hof, J., & Schneider, M. (2019). An adaptive large neighborhood search with path relinking for a class of vehicle-routing problems with simultaneous pickup and delivery. *Networks*, *74*(3), 207-250. Doi: https://doi.org/10.1002/net.21879.

Kapil, S., Chawla, M., & Ansari, M. D. (2016, December). On K-means data clustering algorithm with genetic algorithm. *In 2016 Fourth International Conference on Parallel, Distributed and Grid Computing (PDGC)* (pp. 202-206). IEEE. Doi: https://doi.org/10.1109/PDGC.2016.7913145.

Korayem, L., Khorsid, M., & Kassem, S. S. (2015, May). Using grey wolf algorithm to solve the capacitated vehicle routing problem. *In IOP conference series: materials science and engineering* (Vol. 83, No. 1, p. 012014). IOP Publishing. Doi: https://doi.org/10.1088/1757-899X/83/1/012014.

Li, S., & Wang, F. (2020). Research on Optimization of Improved Gray Wolf Optimization-Extreme Learning Machine Algorithm in Vehicle Route Planning. *Discrete Dynamics in Nature and Society*, 2020, Article ID 8647820, 7 pages,. Doi: https://doi.org/10.1155/2020/8647820.

MacQueen, J. (1967, June). Some methods for classification and analysis of multivariate observations. *In Proceedings of the fifth Berkeley symposium on mathematical statistics and probability* (Vol. 1, No. 14, pp. 281-297). Retrieved from: https://digitalassets.lib.berkeley.edu/math/ucb/text/math_s5_v1_article-17.pdf.

Mirjalili, S., Mirjalili, S. M., & Lewis, A. (2014). Grey wolf optimizer. *Advances in Engineering Software*, *69*, 46-61. Doi: https://doi.org/10.1016/j.advengsoft.2013.12.007.

Prins, C. (2004). A simple and effective evolutionary algorithm for the vehicle routing problem. *Computers & Operations Research*, *31*(12), 1985-2002. Doi: https://doi.org/10.1016/S0305-0548(03)00158-8 .

Semet, F., Toth, P., & Vigo, D. (2014). Classical exact algorithms for the capacitated vehicle routing problem. Vehicle routing: problems, methods, and applications, 37–57. Doi: https://doi.org/10.1137/1.9781611973594.ch2.

Shi, Y., Boudouh, T., & Grunder, O. (2018). An efficient tabu search based procedure for simultaneous delivery and pick-up problem with time window. *IFAC-PapersOnLine*, *51*(11), 241-246. Doi: https://doi.org/10.1016/j.ifacol.2018.08.278.

Shi, Y., Zhou, Y., Boudouh, T., & Grunder, O. (2020). A lexicographic-based two-stage algorithm for vehicle routing problem with simultaneous pickup–delivery and time window. *Engineering Applications of Artificial Intelligence*, *95*, 103901. Doi: https://doi.org/10.1016/j.engappai.2020.103901.

Subramanian, A., Uchoa, E., & Ochi, L. S. (2010, May). New lower bounds for the vehicle routing problem with simultaneous pickup and delivery. *In*

International Symposium on Experimental Algorithms (pp. 276-287). Springer, Berlin, Heidelberg. Doi: https://doi.org/10.1007/978-3-642-13193-6_24.

Taillard, É., 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. Doi: https://doi.org/10.1287/trsc.31.2.170.

Wang, C., Mu, D., Zhao, F., & Sutherland, J. W. (2015). A parallel simulated annealing method for the vehicle routing problem with simultaneous pickup–delivery and time windows. *Computers & Industrial Engineering*, *83*, 111-122. Doi: https://doi.org/10.1016/j.cie.2015.02.005.

Wang, H. F., & Chen, Y. Y. (2012). A genetic algorithm for the simultaneous delivery and pickup problems with time window. *Computers & Industrial Engineering*, *62*(1), 84-95. Doi: https://doi.org/10.1016/j.cie.2011.08.018.