Universal Journal of Operations and Management
https://ojs.wiserpub.com/index.php/UJOM/

UNIVERSAL WISER PUBLISHER

Research Article

An Adapted Fuzzy Multi-Objective Programming Algorithm for Vehicle Routing

Gulcin Dinc Yalcin *, **Nihal Erginel**

Department of Industrial Engineering, Faculty of Engineering, Eskisehir Technical University 2 Eylul Campus, Turkey
E-mail: gdinc@eskisehir.edu.tr

Received: 8 September 2021; **Revised:** 2 December 2021; **Accepted:** 16 December 2021

Abstract:
The vehicle routing problem (VRP) is a well-known problem in the logistics sector. In this study, two objectives, minimizing the total distance and maximizing the saving value, were considered in VRP with a fuzzy environment. The game theory approach is proposed for determining the weights of objectives when decision-makers have insufficient knowledge of assigning the weights. Thus, a fuzzy pay-off matrix is proposed for determining the weights of objectives by combining the fuzzy two-person zero-sum game with mixed strategies (FTZG with MS) and membership functions. Therefore, the fuzzy multi-objective programming (FMOP) model is adapted to the VRP model, which is named Adapted FMOP algorithm for VRP. Proposed algorithm clusters customers according to two objectives and by using four fuzzy operators, and routes customers with the traveling salesman problem (TSP) model in order to avoid the non-deterministic polynomial-time hardness (NP-hard) structure of VRP. In the end, the results are improved using local search methods. The main contribution of the Adapted FMOP algorithm for VRP is that it provides a solution that considers more than one objective without the need for decision makers' view on the weights of objectives in all decision models in the fuzzy environment. Also, the proposed algorithm can find the solution with the help of a mathematical model without requiring any heuristics or metaheuristics, since it primarily performs clustering. Firstly, the efficiency of this algorithm was tested on problems in the literature. The Adapted FMOP algorithm for VRP achieved the best-known solutions by some small margins and exceeded the best-known solution for one problem in the literature. After seeing that the performance of the algorithm was sufficient, a data set of a firm in the construction sector was implemented to see how the algorithm works in real life and the obtained results were discussed. The solutions demonstrate that the Adapted FMOP algorithm for VRP also works well for real-world problems.

Keywords: vehicle routing problem (VRP), fuzzy multi-objective programming (FMOP) model, game theory under fuzziness, fuzzy pay-off matrix

1. Introduction and motivation

1.1 *Vehicle routing problem*

The vehicle routing problem (VRP), first proposed by Dantzig and Ramser (1959), involves the transportation process with an impact on the total cost ranging between 10% and 20% (Toth & Vigo, 2002). Thus, it is significant to reduce the cost of transportation of each vehicle involved in delivering the demands to the customers. In this regard,

DOI: https://doi.org/10.37256/ujom.1120221144

VRP aims to set a route for each vehicle that departs from and arrives at a depot and to minimize the total distance of these routes. The main constraints are visiting each customer at once, not exceeding each vehicle capacity, and providing all customer demands. In addition to the constraints, time windows of each customer, number of different depots, additional features of customers such as backhaul customers, pick up customers, etc. may be considered. Furthermore, for some types of VRP, the route may end with one of the customers instead of a depot.

1.2 *Motivation*

VRP has many variants with additional constraints such as VRP with time windows, VRP with pickup and deliveries, VRP with backhauls, open VRP, VRP with multiple depots, and VRP with a heterogeneous fleet, etc. Since VRP is non-deterministic polynomial-time hardness (NP-hard), all variants are also NP-hard. This precludes obtaining a feasible solution by solving mathematical models. Thus, various heuristic and metaheuristic algorithms have been proposed in the literature to procure the results within a reasonable time. However, to solve VRP with backhauls, Yalcin and Erginel (2015) developed an algorithm named as the fuzzy multi-objective programming-vehicle routing problem with backhauls (FMOP-VRPB) algorithm, which is based on mathematical models. The FMOP-VRPB algorithm is a cluster-first route-second algorithm and it uses mathematical models at each phase. Since the size of the mathematical models for each phase is relatively small, the solutions for each phase are obtained within an appropriate time manner. Furthermore, the computational results of the FMOP-VRPB algorithm are competitive.

Although there are remarkable studies about heuristic and metaheuristic algorithms to solve VRP, there is a gap in developing algorithms based on simple mathematical models. Thus, the main aim of this paper is to solve the NP-hard VRP using mathematical models which are easy to implement, can be solved by optimization software, and do not required parameter setting as in heuristics and metaheuristics. Additionally, a fuzzy environment is significant since the value of membership functions varies between 0 and 1 regardless of the objective function being maximization and minimization. Furthermore, the membership function consists of ideal and anti-ideal values of the objective function, which are the best possible and the worst possible solutions under the problem constraints. Thus, fuzzy methods try to find a solution close to the ideal value and far away from the anti-ideal value. Therefore, the FMOP-VRPB algorithm is adopted to solve the VRP although VRP with backhaul is a more difficult problem than VRP due to additional constraints and decision variables for backhaul customers. Moreover, the implementation of FMOP-VRPB algorithm fits our purpose without the need for any heuristics or metaheuristics. The adopted method is called the adopted FMOP algorithm for VRP, which consists of clustering, routing, and local search phases. Since VRP does not have backhaul customers, the mathematical models for each phase are remodeled according to the structure of VRP and the implementation of local search is redesigned under the constraints of VRP. The clustering phase uses (FMOP) and determines the weights with fuzzy structure and game theory. Then the routes are generated by using the mathematical model formulation of traveling salesman problem (TSP). Finally, the solution improves by local searches. Thus, the decision-maker does not need to define any parameter for the algorithm. It is emphasized that the main advantage of the algorithm is to find a solution by standard optimization software and not to require parameter setting, unlike heuristic and metaheuristic algorithms.

In Section 2, the literature review is given. Some preliminaries are explained in Section 3. The proposed Adapted FMOP algorithm for VRP is described in detail in Section 4. Then, in Section 5, the computational results in terms of benchmark problems are given and discussed. A real case of a company in the construction sector is explained in Section 6. Concluding remarks are made in Section 7.

2. Literature review

Numerous variants of VRP have been introduced for many years in the literature. The variants of classical VRP problems can be classified as (see also Figure 1) capacitated VRP (Letchforda & Salazar-González, 2015; Wei et al., 2015; Cardoso et al., 2015), VRP with time windows (Hong & Park, 1999; Qi et al., 2015), VRP with pickup and deliveries (Dimitrakos & Kyriakidis, 2015; Avcı & Topaloglu, 2015), VRP with backhauls (Goetschalckx & Jacobs-Blecha, 1989; Ropke & Pisinger, 2006; Wang & Wang, 2009; Yalcin & Erginel, 2015; Koc & Laporte, 2018), open VRP (Zachariadis & Kiranoudis, 2010; Aksen et al., 2007; Brito et al., 2015; Erbao et al., 2014), VRP with multiple depots (Montoya-Torres, 2015; Chan & Baker, 2005; Ghafurian & Javadian, 2011), and VRP with the heterogeneous fleet (Koc et al., 2015; Salhi et al., 2014). Furthermore, VRP with time-dependent travel times has been studied recently because travel times are also dependent on the traffic congestion of the roads, which is based on queuing theory (van Woensel et al., 2007; van Woensel & Cruz, 2009; Oyola et al., 2018 for review). Besides, with the increase in the use of drone and electric vehicles, studies have been started on the routing of these vehicles (Pelletier et al, 2019; Wang & Sheu, 2019; Schermer et al, 2019, Keskin et al, 2019).

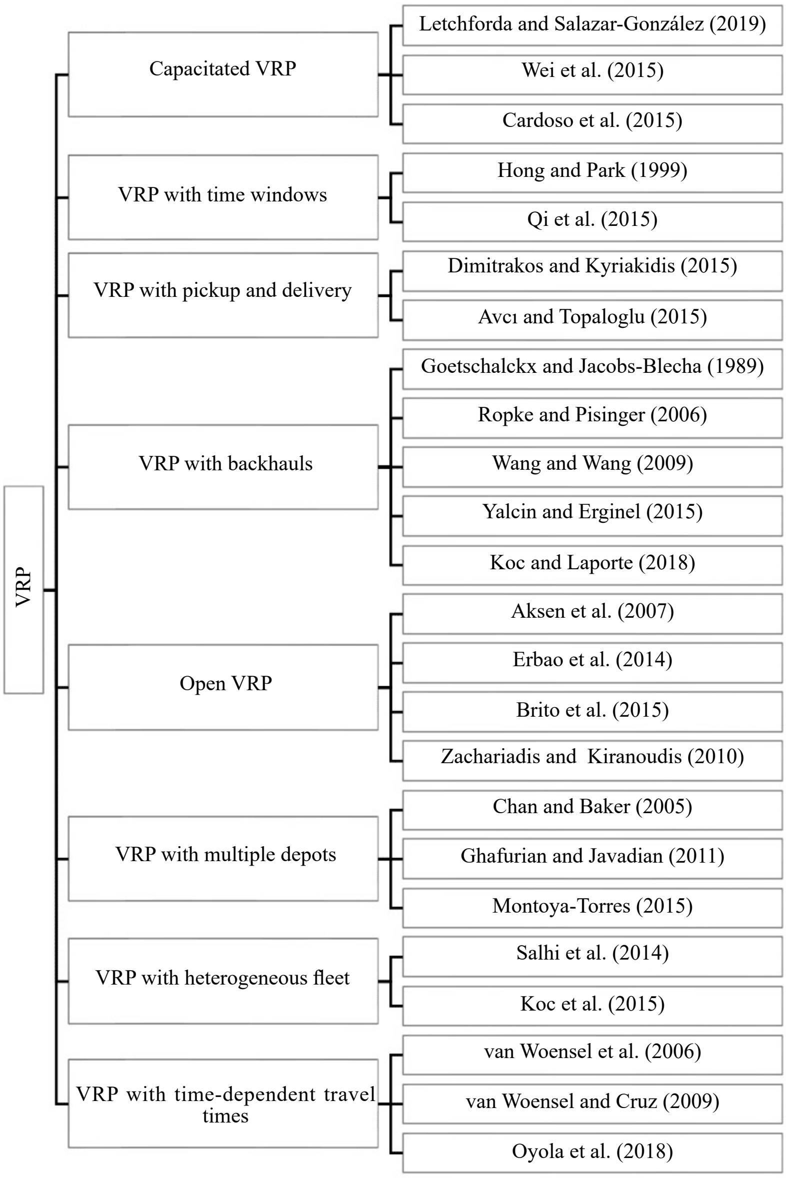

Figure 1. The variants of VRP

2.1 *Exact algorithms for vehicle routing problem*

Exact algorithms (see Laporte & Nobert, 1987; Valle et al., 2009), several heuristics categorized as constructive heuristics, improvement heuristics and two-phase heuristics (see Laporte & Semet, 2001), and meta-heuristics have been proposed to solve the VRP. Constructive heuristics generate a solution under the problem constraints and no improvement is applied, while improvement heuristics involve an improvement of the initial solution. The Saving algorithm (Clarke & Wright, 1964) and Lin's λ-opt approach (Lin, 1965) are examples of constructive heuristics and improvement heuristics, respectively.

2.2 *Heuristic algorithms for vehicle routing problem*

Cluster-first route-second methods and route-first cluster-second methods are classified as two-phase heuristics. Sweep algorithm (Gillett & Miller, 1974), Fisher and Jaikumar (1981) algorithm, Bramel and Simchi-Levi (1995) algorithm are cluster-first route-second methods. In these methods, customers are clustered to vehicles first by using different clustering methods, and then the customers for each cluster are routed by using a traveling salesman algorithm. In the sweep algorithm, clusters are formed using the polar-coordinate angle. In the Fisher and Jaikumar (1981) algorithm, clusters are generated by a generalized assignment model. In the Bramel and Simchi-Levi (1995) algorithm, clusters are formed by capacitated location problems. The seed customers are defined at the beginning in both the Fisher and Jaikumar (1981) and the Bramel and Smichi-Levi (1995) algorithms. Afterward, Koskosidis and Powell (1992) and Baker and Sheasby (1999) extended the Fisher and Jaikumar (1981) algorithm with different seed customers selection strategies. Furthermore, in these studies, one objective, that is, the minimization of the cost, was taken into consideration. Besides, Dijkstra's (1959) algorithm is an example of route-first cluster-second methods. This method routes all customers first and then divides the route into clusters. Beyond these heuristics, Juan et al. (2010) proposed a simulation for the routing of the vehicles using the generalized Clark and Wright (1964) saving heuristic hybrid algorithm and Ball (2011) developed heuristics based on mathematical programming.

2.3 *Metaheuristic algorithms for vehicle routing problem*

Metaheuristics developed for solving VRP are summarized as simulated annealing (Alfa et al., 1991; Breedam, 1995), tabu search (Taillard, 1993; Osman, 1993; Gendreau et al., 1994; Xu & Kelly, 1996; Augerat et al., 1998; Barbarosoglu & Ozgur, 1999; Toth & Vigo, 2003; Cordeau & Maischberger, 2012), ant colony optimization (Bullnheimer et al., 1999; Reimann et al., 2004; Maezzeo & Loiseau, 2004; Bell & McMullen, 2004; Li et al., 2019; Zhang et al., 2009), genetic algorithm (Baker & Ayechew, 2003; Nazif & Lee, 2012), and particle swarm optimization (Ai & Kachitvichyanukul, 2009). In addition to these, hybrid of several metaheuristics such as simulated annealing and tabu search (Osman, 1993; Lin et al., 2009), ant colony optimization and scatter search (Zhang & Tang, 2009), particle swarm, multiphase neighborhood, and greedy randomized adaptive searches procedures (Marinakis et al., 2010), and genetic and particle swarm optimization algorithms (Marinakis & Marinaki, 2010) are proposed in the literature to solve VRP. Additionally, Yurtkuran and Emel (2010) used an electromagnetism-like algorithm for continuous problems with bounded variables. Chen et al. (2010) proposed an iterated variable neighborhood descent algorithm. Szeto et al. (2011) used an artificial bee colony algorithm to solve the VRP. Drexl (2012) provided the state of the art of scientific research on VRP. He defined the characteristics of VRP in five dimensions: requests, fleet, route structure, objectives, and scope of planning. He presented the rich VRP term that incorporates more complex constraints and objectives of real-life routing problems. Derigs and Vogel (2013) proposed a heuristic framework for solving rich VRPs and implemented a flexible software framework (also see Toth & Vigo, 2002, and Cordeau et al., 2002 for surveys).

Moreover, fuzzy set theory is used to model VRP with fuzzy demand (Teodorović & Pavković, 1996; Erbao & Mingyong, 2009, 2010; Kuo et al., 2012; Mehrjerdi & Nadizadeh, 2013) and fuzzy travel time (Zheng & Liu, 2006; Tang et al., 2009; Zarandi et al., 2011; Ghannadpour et al., 2013), and the fuzzy models are solved with a heuristic or a metaheuristic algorithm such as heuristic-based on sweep algorithm, hybrid of genetic algorithm and fuzzy simulation, hybrid of evolutionary algorithm and simulated annealing, and hybrid of genetic algorithm and particle swarm optimization, etc.

3. Preliminaries

3.1 *Multi-objective optimization and membership functions*

The multi-objective model has more than one objective function such as z_1, z_2, ... z_n to optimize under the problem constraints $x \in X$. In this paper, two objective functions are considered z_1 and z_2. Every objective has an ideal value and an anti-ideal value which can be obtained by solving each objective in optimal and suboptimal manners individually under the problem constraints, respectively. The ideal value represents the optimum value, and the anti-ideal value indicates the furthest value to the ideal value when there is only one objective in the model under the constraints. It is difficult to find the optimum solutions for all objectives simultaneously. For this reason, there is a set of solutions called the Pareto Optimal Solution. There is a trade-off between objectives. While constructing the fuzzy pay-off matrix, the ideal (utopian) and anti-ideal (nadir) values of each objective function are used to define the membership functions $\mu_k(x)\ \forall\ k = 1,2$, so no decision is made. The value of the membership function may take 1 at the best case, 0 at the worst case, and alternate between 0 and 1, regardless of whether the objective is a maximization or a minimization problem. If the membership function of an objective is 1, then the objective reaches its ideal value, and similarly, if the membership function of an objective is 0, then the objective reaches its anti-ideal value.

3.2 *Fuzzy operators*

Fuzzy operators which are used in the clustering phase are explained below. The original constraints of the multi-objective programming model are shown as $x \in X$.

Max-min operator (MO): The following model in equation (1) describes the min operator, where λ is the overall satisfaction level (Zimmermann, 1978)

$$\begin{aligned} &\max \lambda \\ &\text{subject to} \quad \mu_k(x) \geq \lambda \quad \forall k = 1,2 \\ &\qquad \lambda \in [0,1] \\ &\qquad x \in X \end{aligned} \tag{1}$$

Two-phase approach (TPA): The first phase is the same as the MO. The second phase is modeled in equation (2) (Li et al., 2006)

$$\begin{aligned} &\max \sum_{k=1}^{2} w_k \lambda_k \\ &\text{subject to} \quad \mu_k(x) \geq \lambda_k \geq \lambda^*_k \quad \forall k = 1,2 \\ &\qquad \lambda_k \in [0,1] \quad \forall\ k = 1,2 \\ &\qquad x \in X \end{aligned} \tag{2}$$

where w_k is the weight of the kth objective, λ_k is the satisfaction level of the kth objective, and λ_k^* is the membership degree of the kth objective that is obtained from the first phase.

Weighted additive model (WAM): The following model of equation (3) describes the structure of the WAM (Tiwari et al., 1987)

$$\begin{aligned} &\max \sum_{k=1}^{2} w_k \mu_k(x) \\ &\text{subject to} \quad \lambda_k \in [0,1] \quad \forall k = 1,2 \\ &\qquad x \in X \end{aligned} \tag{3}$$

Weighted max-min model (WMM): The structure of the model is described in equation (4) (Lin, 2004)

$$\begin{aligned} &\max \lambda \\ &\text{subject to} \quad \mu_k(x) \geq \lambda w_k \quad \forall k = 1,2 \\ &\qquad \lambda \in [0,1] \\ &\qquad x \in X \end{aligned} \tag{4}$$

4. Problem formulation and resolution methodology

The Adapted FMOP algorithm for VRP is based on the paper by Yalcin and Erginel (2015). The proposed Adapted

FMOP algorithm for VRP has three sequential phases. The steps of the proposed algorithm are given as follows:

Phase 1: Clustering Phase

Step 1: Set the multi-objective model in the crisp case for clustering customers.

Step 2: Calculate the membership functions.

Step 3: Determine the weights of the objectives by the fuzzy two-person zero-sum game with mixed strategies (FTZG with MS).

Step 4: Set the FMOP model and solve it with fuzzy operators to cluster customers and assign them to the vehicles.

Phase 2: Routing Phase

Step 5: Set and solve the TSP integer programming model for routing.

Phase 3: Local search

Step 6: Use local search operations to improve the route.

4.1 *Phase 1: Clustering phase*

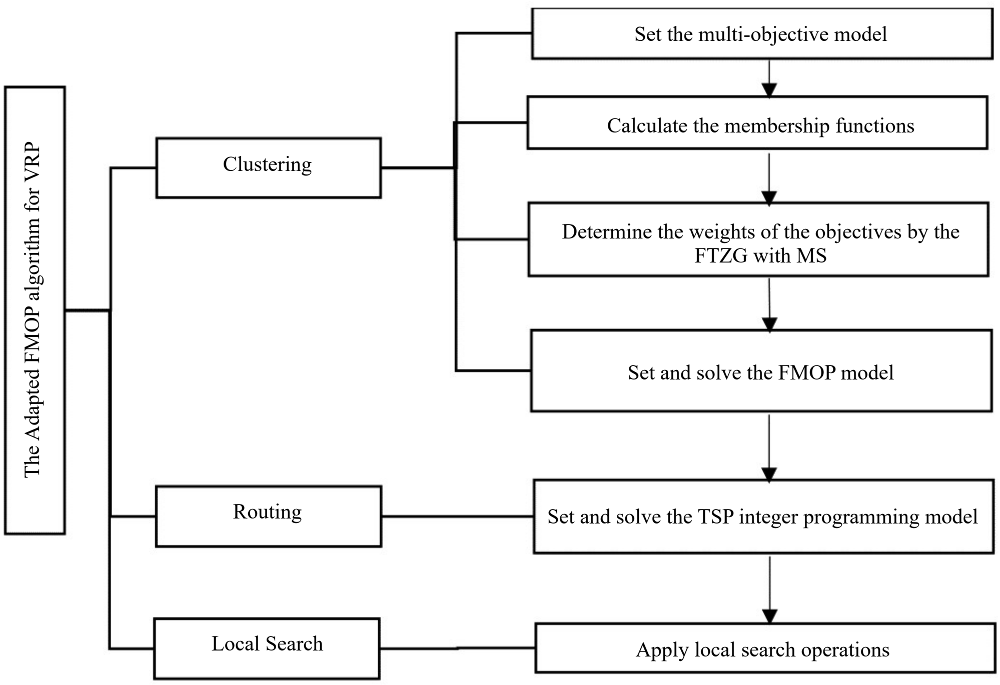

Figure 2. Flow chart of the Adapted FMOP algorithm for VRP

Clusters of customers are formed by assigning the customers to each vehicle under the vehicle capacity in the clustering phase. Two objectives (the first objective is minimizing the distance and the second objective is maximizing the saving value) are considered while the other studies (Fisher and Jaikumar, 1981; Bramel and Simichi-Levi, 1995; Koskosidis & Powell, 1992; Baker & Sheasby, 1999) consider one objective. Saving value is the rate that considers the benefit of going from a customer to another, taking into account the distance of these customers from the depot. Furthermore, the mathematical model defines the seed customers (equal to the number of vehicles) unlike similar studies (Fisher & Jaikumar, 1981; Bramel & Simichi-Levi, 1995, Koskosidis & Powell, 1992; Baker & Sheasby, 1999) and assigns other customers to the seed customers. Therefore, there is no need to define seed customers at the beginning of the algorithm by a decision-maker or by any method. In a sense, the proposed clustering phase has two new novelties unlike the existing literature: considering two objectives and defining the seed customers by the multi-objective model. Moreover, using fuzzy approaches for both defining the weights of objectives and solving the multi-objective model are other contributions.

4.1.1 *Step 1: Set the multi-objective model in the crisp case for clustering customer*

Sets of the cluster phase:

i, j customer nodes ($i = j = 1, \dots, N$)

Parameters of the cluster phase:

N number of customers

d_{ij} distance between nodes i and j

s_{ij} saving value between nodes i and j ($d_{0i} + d_{0j} - d_{ij}$)

a_i the demand of customer i

C the capacity of each of the vehicles

K number of vehicles

Decision variable:

x_{ij} = {1 if node i is assigned to node j_0 otherwise}

If $x_{ij} = 1 \ni i = j$, it means that the node i/j is a seed customer.

$$\text{Model:} \quad \text{minimize } z_1 = \sum_i \sum_j d_{ij} x_{ij} \tag{5}$$

$$\text{maximize } z_2 = \sum_i \sum_j s_{ij} x_{ij} \tag{6}$$

$$\text{Subject to:} \quad \sum_{j=1}^{N} x_{ij} = 1 \qquad \forall i = 1, \dots, N \tag{7}$$

$$\sum_{i=1}^{N} a_i x_{ij} \le C x_{jj} \qquad \forall j = 1, \dots, N \tag{8}$$

$$\sum_{i=1}^{N} \sum_{j=1}^{N} x_{jj} = K \tag{9}$$

$$x_{ij} \in \{0,1\} \tag{10}$$

While equation (5) is minimizing the distance within a cluster, equation (6) is maximizing the saving value within a cluster. Equation (7) ensures the assignment of each customer to each cluster at once. The capacity and demand constraints are indicated in equation (8). Equation (9) ensures that the number of clusters is equal to the number of vehicles. Finally, equation (10) represents the decision variable constraint.

4.1.2 *Step 2: Calculate the membership functions*

For the clustering phase, the membership functions of objectives are defined as in the following, and equation (11) and equation (12) refer to the first z_1 and the second z_2 objective functions, respectively.

$$\mu_1(x) = \begin{Bmatrix} 1 & \text{if } z_1 < z^*_1 \\ \dfrac{[z'_1 - z_1]}{[z'_1 - z^*_1]} & \text{if } z^*_1 \le z_1 \le z'_1 \\ 0 & \text{if } z_1 < z'_1 \end{Bmatrix} \tag{11}$$

$$\mu_2(x) = \begin{Bmatrix} 1 & \text{if } z_2 < z^*_2 \\ \dfrac{[z_2 - z'_2]}{[z^*_2 - z'_2]} & \text{if } z'_2 \le z_2 \le z^*_2 \\ 0 & \text{if } z_2 < z'_2 \end{Bmatrix} \tag{12}$$

where z^*_1 and z'_1 are respectively the ideal and anti-ideal values of the first objective function z_1 and the same formulation is valid for the second objective function z_2. These values are calculated from (13) and (14) by solving them individually under the problem constraints.

$$z^*_1 = \min(z_1) \qquad z'_1 = \max(z'_1) \tag{13}$$

$$z^*_2 = \max(z_2) \qquad z'_2 = \min(z_2) \tag{14}$$

The degree of membership function changes between 0 and 1, which infers that the ideal or the anti-ideal solution is reached. Thus, the membership function degree shows the achievement level of the related objective and it means identical whatever the objective (maximization or minimization) is.

4.1.3 *Step 3: Calculate the weights of the objectives*

The weights of the objectives are needed to use the fuzzy operators as solution approaches for the multi-objective model. For this purpose, the FTZG with MS model, proposed by Yalcin and Erginel (2011) is used. There are two types of strategies: objectives z_1,z_2, and the ideal solutions of the objectives x^1, x^2. Then the membership functions $\mu_i(x^j)$ are calculated related to the ideal solutions to form the pay-off matrix shown in Table 1. Finally, the linear program given in Equation (15) is solved to obtain the weights of the objectives.

Table 1. Fuzzy pay-off matrix of the FTZG with MS model

Person I strategies	**Person II strategies**	
	x^1	x^2
z_1	$\mu_1(x^1)$	$\mu_1(x^2)$
z_2	$\mu_2(x^1)$	$\mu_2(x^2)$

$$\begin{aligned} &\max v \\ &\text{subject to:} \quad \sum_{i=1}^{2} w_i \mu_i\left(x^j\right) \geq v \qquad \forall j = 1,2 \\ &\qquad\qquad \sum_{i=1}^{2} w_i = 1 \\ &\qquad\qquad w_i \geq 0 \qquad \forall i = 1,2 \end{aligned} \tag{15}$$

Where z_1 is the objective function, x^j is the ideal value of the objective function, $\mu_i(x^j)$ is the membership function value, and w_i is the weight of objectives.

After determining the weights of the objectives, a multi-objective model can be solved in the fuzzy case using several fuzzy operators, such as the max-min operator, the two-phase approach, the weighted additive model approach, and the weighted max-min model as follows:

4.1.4 *Step 4: Set the FMOP model and solve it with fuzzy operators to cluster customers and assign them to the vehicles*

After constructing membership functions for each objective, the fuzzy operators given by equations (1)-(4) are used to solve the FMOP problem.

4.2 *Phase 2: Routing*

4.2.1 *Step 5: Set and solve the TSP integer programming model for routing*

The TSP integer programming model is solved for each vehicle to obtain the initial routes.

4.3 *Phase 3: Local search*

4.3.1 *Step 6: Use local search operations to improve the route*

Insertion and interchange operations are commonly applied in VRP problems since they are easy to implement and successful in improving the solutions obtained so far. Thus, these operations are used to apply the local search to improve the solution obtained after the first two phases. Insertion operation picks a suitable customer from the selected vehicle and adds the customer into an alternative vehicle while interchange operation exchanges two particular customers from two particular vehicles. The operations are applied if the capacity constraint is held and the distance is decreased.

5. Computational experiment

The proposed Adapted FMOP algorithm for VRP is tested on the benchmark problems of Christofides (1979) to test the performance of the algorithm. The mathematical models in clustering routing phases are solved by using (General Algebraic Modeling System) GAMS with the CPLEX 9.0 solver and the local search is applied by Excel Visual Basic Application (VBA). All experiments are run on a 2.20 GHz computer with 1.0 GB of RAM. Table 2 represents the computational results. The gap is calculated by $\left(\frac{\text{by proposed model-best known}}{\text{best known}}\right)\times 100$, which is commonly used in VRP. The gap shows the relative distance from the best-known solution. The FMOP-VRP and the best-known solutions indicate the total distance for the relevant computed routes. Central processing unit (CPU)-Phase-1, CPU-Phase-2, CPU-Phase-3, and total CPU are the times in seconds that are needed to compute each phase and total time, respectively.

Table 2. The results of test problems using the Adapted FMOP algorithm

Problem	Best known	The name of solution phase	FMOP-VRP	% Gap	CPU-Phase-1	CPU-Phase-2	CPU-Phase-3	Total CPU
		TPA	537.34	2.43	7.139	0.873	0.031	8.043
C1	524.61[a]	WAM	537.34	2.43	7.139	0.873	0.031	8.043
		WMM	537.34	2.43	7.139	0.873	0.031	8.043
		MO	869.26	4.07	872.03	0.841	0.04	872.91
C2	835.26[a]	TPA	869.26	4.07	921.48	0.841	0.04	922.36
		WAM	869.26	4.07	814.53	0.841	0.04	815.41
C3	826.14[a]	TPA	854.82	3.47	166.58	3.716	2.25	172.54
C4	1028.42[a]	MO	1073.43	4.38	1000	92.399	4.66	1097.06
C5	1291.29[b]	WAM	1359.62	5.28	5000	3.27	12.41	5015.71
C11	1042.11[a]	MO	1053.83	1.12	1387.19	2002.81	0.12	3390.12
		MO	**815.24**	**-0.53**	31.16	0.87	2.06	34.09
C12	819.56[a]	TPA	**815.24**	**-0.53**	29.56	0.869	2.06	32.08
		WAM	**815.24**	**-0.53**	28.89	0.869	2.06	31.82

[a]Taillard (1993) [b]Mester and Bräysy (2005)

As shown, the proposed Adapted FMOP algorithm for VRP obtained better results for the C12 problem due to the best-known solution. In addition, the improved route is shown in detail in Table 3. If the results are compared according to gaps, it may be said that the average gap is 2.89. Furthermore, in addition to the average gap, the minimum gap except C12 is 1.12 for the C11 and the maximum gap is 4.38 for the C4. The gap of problems except the 12 problems varies between 1.12 and 5.28. Thus, the results indicate that the Adapted FMOP algorithm for VRP is able to find sufficient solutions. The average CPU time is 853.66 seconds (14.23 min) for the clustering phase, 175.75 seconds (2.9 min) for the routing phase, and 1.98 seconds for the local search, and 1031.37 (17.19 min) for the total process. The computational times are reasonable for obtaining a solution for VRP.

Table 3. Detailed route for the problem C12

	Distance of each route	Nodes in each route															
	49.15	0	8	9	6	7	4	3	75	0							
	98.09	0	10	13	17	18	19	15	16	14	12	11	0				
	43.88	0	21	23	26	28	30	27	25	22	20	0					
	97.84	0	24	29	34	36	39	38	37	35	31	33	32	0			
	64.81	0	43	42	41	40	44	45	46	48	51	50	52	49	47	0	
	102.94	0	69	68	55	54	53	56	58	60	59	57	0				
	129.23	0	81	78	76	71	70	73	77	79	80	74	65	0			
	52.28	0	67	66	64	61	72	62	67	0							
	76.08	0	91	89	88	85	84	82	83	86	87	90	0				
	97.93	0	5	1	2	99	100	97	93	92	94	95	96	98	0		
Total distance of each route	812.23																

A comparison of results with other algorithms are is given in Table 4. The better results are shown in bold. The Adapted FMOP algorithm finds better solutions 6, 3, 4 and 3 times than the sweep algorithm, the Fisher and Jaikumar (1981) algorithm, the Bramel and Simchi-Levi (1995) algorithm, Baker and Sheasby's (1999) method 2, respectively. Besides, the average gaps while comparing with the above algorithms are found as -5.42, 0.20, -1.22, 1.04, and 0.46, which show the competence of the Adapted FMOP algorithm.

Table 4. Comparison with other cluster-first route-second methods

Problem	FMOP-VRP	Sweep	Gap	Fisher and Jaikumar (1981)	Gap	Bramel and Smichi-Levi (1995)	Gap	Baker and Sheasby method 1 (1999)	Gap	Baker and Sheasby method 2 (1999)	Gap
C1	537.34	532	1.00	524	2.55	524.6	2.43	524.61	2.43	524.61	2.43
C2	853.11	874	**-2.39**	857	**-0.45**	848.2	0.58	847.50	0.66	847.50	0.66
C3	838.54	851	**-1.46**	833	0.67	832.9	0.68	837.44	0.13	841.32	**-0.33**
C4	1073.43	1079	**-0.52**	1014	5.86	1088.6	**-1.39**	1053.50	1.89	1077.41	**-0.37**
C5	1335.08	1389	**-3.88**	1420	**-5.98**	1461.2	**-8.63**	1333.72	0.10	1336.49	**-0.11**
C11	1046.02	1266	**-17.38**	-	-	1051.5	**-0.52**	-	-	-	-
C12	812.23	937	**-13.32**	824	**-1.43**	826.1	**-1.68**	-	-	-	-
		Average	**-5.42**	Average	**0.20**	Average	**-1.22**	Average	1.04	Average	0.46

6. Real-world application for a firm in the construction sector

A firm from the construction sector in Turkey has customers in different cities. A routing problem for a day is solved with the proposed Adapted FMOP algorithm for VRP. The problem has 57 customers in 10 cities, and the location of the depot is Eskisehir. The cities and demands of these customers are presented in Table 5. The capacity of the vehicles is 28.000 tons and 25 vehicles are required to serve the entire demand of the customers.

The mathematical models are coded and solved by The GAMS CPLEX 9.0 and the local search is applied in Excel Visual Application.

Table 5. City and demand of customers

Customer number	City	Demand (tons)	Customer number	City	Demand (tons)
1	Afyon	3864	30	Istanbul	15025
2	Ankara	209	31	Istanbul	14000
3	Ankara	66	32	Istanbul	14000
4	Ankara	1475	33	Istanbul	14000
5	Ankara	12268	34	Istanbul	14000
6	Ankara	12322	35	Istanbul	911
7	Ankara	6100	36	Istanbul	13089
8	Antalya	21554	37	Istanbul	14000
9	Antalya	450	38	Istanbul	15025
10	Antalya	7556	39	Istanbul	15025
11	Antalya	11211	40	Istanbul	7537
12	Antalya	5522	41	Istanbul	9026
13	Antalya	12478	42	Istanbul	2958
14	Istanbul	19585	43	Istanbul	11042
15	Istanbul	16000	44	Izmir	14000
16	Istanbul	15052	45	Kayseri	11677
17	Istanbul	7748	46	Kayseri	7689
18	Istanbul	14218	47	Kocaeli	17210
19	Istanbul	1061	48	Kocaeli	2178
20	Istanbul	10526	49	Kocaeli	6374
21	Istanbul	5553	50	Kocaeli	19436
22	Istanbul	15090	51	Kocaeli	14660
23	Istanbul	14000	52	Kocaeli	19000
24	Istanbul	14000	53	Kocaeli	24000
25	Istanbul	14000	54	Mugla	15398
26	Istanbul	14000	55	Mugla	16065
27	Istanbul	14000	56	Sakarya	17300
28	Istanbul	14000	57	Tekirdag	18662
29	Istanbul	14000			

6.1 *The clustering phase of the application*

In the first phase of the proposed Adapted FMOP algorithm for VRP, ideal and anti-ideal values of the objectives should be achieved to obtain the membership functions of the objectives. The calculated ideal and anti-ideal values are listed in Table 6. The membership functions are given in equation (16) and equation (17) for the first objective and the second objective, respectively.

Table 6. Ideal and anti-ideal values of objectives

Objective	Ideal value	Anti-ideal value
z_1	3889	17274
z_2	16952	5005

The fuzzy pay-off matrix is set as described in Table 7 by using membership functions, where x^1 and x^2 correspond to the ideal solutions of the first objective and the second objective, respectively.

$$\mu_1(x) = \begin{Bmatrix} 1 & \text{if } z_1 < 450507 \\ \frac{[17274 - z_1(x)]}{[17274 - 3889]} & \text{if } 17274 \leq z_1(x) \leq 3889 \\ 0 & \text{if } z_1(x) > 17274 \end{Bmatrix} \tag{16}$$

$$\mu_1(x) = \begin{Bmatrix} 1 & \text{if } z_2 < 16952 \\ \frac{[z_2(x) - 5005]}{[16952 - 5005]} & \text{if } 5005 \leq z_2(x) \leq 16952 \\ 0 & \text{if } z_2(x) > 5005 \end{Bmatrix} \tag{17}$$

Table 7. Fuzzy pay-off matrix with membership functions

Person I strategies	Person II strategies	
	x^1	x^2
z_1	1	0.968
z_2	0.987	1

The FTZG with MS model is set as equation (18) using the values given in Table 7.

$$\begin{aligned} & \max v \\ & \text{subject to} \quad w_1 + 0.987 w_2 \geq v \\ & \qquad 0.987 w_1 + w_2 \geq v \\ & \qquad \sum_{i=1}^{2} w_i = 1 \\ & \qquad w_i \geq 0 \quad \forall i = 1,2 \end{aligned} \tag{18}$$

Table 8. Results of the fuzzy operators

Objectives	Fuzzy operators			
	MO	TPA	WMM	WAM
z_1,	4029	4029	4066	4029
z_2	16952	16952	16952	16952
CPU-1 (seconds)	57.727	22.743	25.067	22.914

After solving the FTZG with MS model, the weights are obtained as 0.29 and 0.71 for the first objective and the second objective, respectively. The fuzzy multi-objective model given in equations (1)-(4) is solved using the weights. The values of the solutions and the CPU time that is required to solve the multi-objective model are presented in Table 8 for each fuzzy operator.

After all of the fuzzy operators are found, the same solution is achieved, except for the weighted max-min model. While the weighted max-min model finds a great first objective compared to others, it finds the same value as the second objective. The CPU times of the operators are quite equal, except for the min operator that requires 2.5 times more CPU time than the other operators.

Clusters that are found by min operator, two-phase approach, weighted max-min model, and weighted additive model are given a cluster number and the seed customer of each cluster; in addition, the clusters are assigned customers to the seed customers, as described in Table 9 and Table 10.

Table 9. Clusters determined by MO, TPA, and WAM

Customer number	Seed customer	Assigned customers		Cluster number	Seed customer	Assigned customers		Cluster number	Seed customer	Assigned customers	
Cluster 1	Ankara	Istanbul	Kayseri	Cluster 10	Istanbul	Istanbul	Istanbul	Cluster 18	Istanbul	Istanbul	
Cluster 2	Ankara	Ankara	Sakarya	Cluster 11	Istanbul	Istanbul	Istanbul	Cluster 19	Istanbul	Istanbul	Istanbul
Cluster 3	Ankara	Ankara	Kocaeli	Cluster 12	Istanbul	Istanbul		Cluster 20	Istanbul	Istanbul	
Cluster 4	Ankara	Istanbul		Cluster 13	Istanbul	Istanbul		Cluster 21	Kayseri	Kocaeli	
Cluster 5	Antalya	Antalya	İzmir	Cluster 14	Istanbul	Istanbul		Cluster 22	Kocaeli	Kocaeli	
Cluster 6	Antalya	Afyon	Mugla	Cluster 15	Istanbul	Istanbul		Cluster 23	Kocaeli	Kocaeli	
Cluster 7	Antalya	Muğla		Cluster 16	Istanbul	Istanbul		Cluster 24	Kocaeli		
Cluster 8	Antalya	Antalya		Cluster 17	Istanbul	Istanbul		Cluster 25	Tekirdag	Istanbul	
Cluster 9	Istanbul	Istanbul									

Table 10. Clusters that are determined by WMM

Customer number	Seed customer	Assigned customers		Cluster number	Seed customer	Assigned customers		Cluster number	Seed customer	Assigned customers	
Cluster 1	Ankara	Ankara	Istanbul	Cluster 10	Istanbul	Istanbul	Istanbul	Cluster 18	Istanbul	Istanbul	
Cluster 2	Ankara	Ankara	Istanbul	Cluster 11	Istanbul	Istanbul		Cluster 19	Istanbul	Istanbul	Istanbul
Cluster 3	Ankara	Kayseri	Kocaeli	Cluster 12	Istanbul	Istanbul		Cluster 20	Kayseri	Kocaeli	
Cluster 4	Ankara	Sakarya		Cluster 13	Istanbul	Istanbul		Cluster 21	Kocaeli	Kocaeli	
Cluster 5	Antalya	Antalya	Antalya	Cluster 14	Istanbul	Istanbul		Cluster 22	Kocaeli	Kocaeli	
Cluster 6	Antalya	Afyon	İzmir	Cluster 15	Istanbul	Istanbul		Cluster 23	Kocaeli		
Cluster 7	Antalya	Mugla		Cluster 16	Istanbul	Istanbul		Cluster 24	Muğla	Antalya	
Cluster 8	Istanbul	Istanbul		Cluster 17	Istanbul	Istanbul	Istanbul	Cluster 25	Tekirdag	Istanbul	
Cluster 9	Istanbul	Istanbul									

6.2 *The routing and local phases of the application*

Each cluster is solved by the TSP integer programming model based on equations (16)-(20) and then the local search is applied. Table 11 shows the solution and CPU time for each fuzzy operator. All of the fuzzy operators find the same solution in the routing phase. The CPU times are quite equal, except for that of the two-phase approach that requires 1.5 times more CPU time than the other clusters.

Table 11. Total distance and CPU time after routing and local search phases

Fuzzy operator	Solution of routing phase (km)	CPU-Phase-2	Solution (km)	CPU-Phase-3	Total CPU
MO	20374	1.375	20374	0.075	**1432.727**
TPA	20374	2.331	20374	0.064	**2353.743**
WMM	20374	1.566	20374	0.114	**1591.067**
WAM	20374	1.454	20374	0.067	**1476.914**

The local search does not improve the solution in the application although it is effective for the algorithm for the test problems. In that, the application problem has less variety in the customer city. All of the CPU times are quite equal for the local search. The CPU times for the proposed algorithm, min operator, weighted max-min model, and weighted additive model are found to be equal, whereas the two-phase approach requires 1.5 times more CPU time than the weighted max-min model and weighted additive model. The routes of the solution are presented in Table 12.

Table 12. Routes of the solution

Route number	Route					Distance (km)
1	Eskisehir (Depot)	Istanbul (30)	Ankara (2)	Kayseri (45)	Eskisehir (Depot)	1644
2	Eskisehir (Depot)	Ankara (4)	Ankara (7)	Sakarya (56)	Eskisehir (Depot)	720
3	Eskisehir (Depot)	Ankara (3)	Ankara (5)	Kocaeli (51)	Eskisehir (Depot)	794
4	Eskisehir (Depot)	Istanbul (39)	Ankara (6)	Eskisehir (Depot)		1016
5	Eskisehir (Depot)	Antalya (13)	Antalya (9)	İzmir (44)	Eskisehir (Depot)	1282
6	Eskisehir (Depot)	Afyon (1)	Antalya (10)	Mugla (54)	Eskisehir (Depot)	1251
7	Eskisehir (Depot)	Mugla (55)	Antalya (11)	Eskisehir (Depot)		1239
8	Eskisehir (Depot)	Antalya (8)	Antalya (12)	Eskisehir (Depot)		848
9	Eskisehir (Depot)	Istanbul (21)	Istanbul (14)	Eskisehir (Depot)		660
10	Eskisehir (Depot)	Istanbul (41)	Istanbul (42)	Istanbul (15)	Eskisehir (Depot)	660
11	Eskisehir (Depot)	Istanbul (22)	Istanbul (40)	Istanbul (19)	Eskisehir (Depot)	660
12	Eskisehir (Depot)	Istanbul (33)	Istanbul (24)	Eskisehir (Depot)		660
13	Eskisehir (Depot)	Istanbul (37)	Istanbul (25)	Eskisehir (Depot)		660
14	Eskisehir (Depot)	Istanbul (26)	Istanbul (23)	Eskisehir (Depot)		660
15	Eskisehir (Depot)	Istanbul (31)	Istanbul (28)	Eskisehir (Depot)		660
16	Eskisehir (Depot)	Istanbul (32)	Istanbul (29)	Eskisehir (Depot)		660
17	Eskisehir (Depot)	Istanbul (34)	Istanbul (27)	Eskisehir (Depot)		660
18	Eskisehir (Depot)	Istanbul (36)	Istanbul (18)	Eskisehir (Depot)		660
19	Eskisehir (Depot)	Istanbul (35)	Istanbul (38)	Istanbul (20)	Eskisehir (Depot)	660
20	Eskisehir (Depot)	Istanbul (43)	Istanbul (16)	Eskisehir (Depot)		660
21	Eskisehir (Depot)	Istanbul (46)	Kocaeli (52)	Eskisehir (Depot)		1422
22	Eskisehir (Depot)	Kocaeli (48)	Kocaeli (47)	Eskisehir (Depot)		438
23	Eskisehir (Depot)	Kocaeli (50)	Kocaeli (49)	Eskisehir (Depot)		438
25	Eskisehir (Depot)	Tekirdag (57)	Istanbul (17)	Eskisehir (Depot)		924
Total Distance (km)						**20374**

As can be seen from the results, all customer demands are delivered with a traveling distance of 20374 km and by using 25 vehicles. The total CPU time is 39 min. at most. The time needed to find a solution is reasonable for a logistic planning manager compared to creating manually constructed routes. Besides CPU time, since the algorithm does not require additional parameters except the basic parameters of VRP, the logistic planning manager can apply the algorithm straightforwardly. Furthermore, in the clustering phase, the customers which are in sight of each other are grouped by solving FMOP models. Then, these clusters make it easy to generate routes for each vehicle since the number of customers is reduced for each vehicle. Thus, these clusters may be utilized in future planning such as delivery, marketing, new collaborations, etc. by the logistic planning manager of the company since the customers in each cluster are relatively nearby.

7. Conclusion

In this study, the Adapted FMOP algorithm for VRP was introduced for solving NP-hard structure VRP. The algorithm was tested on problems available in the literature, and applied to solve a real-world problem. The Adapted FMOP algorithm for VRP has three phases: the clustering phase with two objective values, the routing phase solved by integer programming, and the local search phase for improving the solution. In addition, the clusters were obtained by solving the fuzzy multi-objective model with the weights obtained to solve the FTZG with MS model. After the clustering phase, the VRP was converted into the TSP for each cluster.

The proposed algorithm was tested on problems presented by Christofides (1979), which represent benchmark problems (Yalcin and Erginel, 2012). The proposed algorithm generates routes with less transportation distance for one problem, and the average difference of the test problems is 4.26. Additionally, the CPU times required to solve the test problems are below 200 seconds for three of them, below 2000 seconds for two of them, and above 3000 seconds for two of them. Thus, it is concluded that the proposed algorithm is able to provide sufficient solutions within an acceptable amount of time.

A real-world logistic problem of a ceramic factory in Turkey was solved using the Adapted FMOP algorithm for VRP. Twenty-five vehicles were required to set their routes. The total distance of the problem was found to be 20374 km with all of the fuzzy operators. The total CPU times for the fuzzy operators was found to be very close to each other. Furthermore, the average CPU time is 1713.61 seconds, that is, 28.5 minutes, which is acceptable. These computational results represent that the Adapted FMOP algorithm for VRP outperforms for real-world problems.

The main contributions of the Adapted FMOP algorithm for VRP are listed as follows:

- considers more than one objective in the clustering phase,
- defines the weights of objectives using the FTZG with MS model that does not require an expert decision or equal scale,
- constructs a fuzzy multi-objective model that maximizes the achievement level of each objective,
- only uses mathematical programming models while solving VRP in the clustering and routing phases,
- successfully solves a real-world problem.

In further studies, other objectives can be considered in the fuzzy VRP model, for example, maximizing the used capacity ratio of vehicles, minimizing the number of vehicles, minimizing the cost of carbon emission, and so on. Additionally, the other variants of VRP models such as time windows, heterogeneous vehicles can be considered.

Conflict of interest statement

We have no conflict of interest to disclosure.

References

Ai, T. J., & Kachitvichyanukul, V. (2009). Particle swarm optimization and two solution representations for solving the capacitated vehicle routing problem. *Computers & Industrial Engineering*, *56*(1), 380-387. https://doi.org/10.1016/j.cie.2008.06.012

Aksen, D., ÖZyurt, Z., & Aras, N. (2007). Open vehicle routing problem with driver nodes and time deadlines. *Journal of the Operational Research Society*, *58*(9), 1223-1234. https://doi.org/10.1057/palgrave.jors.2602249

Alfa, A. S., Heragu, S. S., & Chen, M. (1991). A 3-OPT based simulated annealing algorithm for vehicle routing problems. *Computers & Industrial Engineering*, *21*(1-4), 635-639. https://doi.org/10.1016/0360-8352(91)90165-3

Augerat, P., Belenguer, J., Benavent, E., Corberán, A., & Naddef, D. (1998). Separating capacity constraints in the CVRP using tabu search. *European Journal of Operational Research*, *106*(2-3), 546-557. https://doi.org/10.1016/s0377-2217(97)00290-7

Avci, M., & Topaloglu, S. (2015). An adaptive local search algorithm for vehicle routing problem with simultaneous and mixed pickups and deliveries. *Computers & Industrial Engineering*, *83*, 15-29. https://doi.org/10.1016/j.cie.2015.02.002

Baker, B. M., & Ayechew, M. (2003). A genetic algorithm for the vehicle routing problem. *Computers & Operations Research*, *30*(5), 787-800. https://doi.org/10.1016/s0305-0548(02)00051-5

Baker, B. M., & Sheasby, J. (1999). Extensions to the generalised assignment heuristic for vehicle routing. *European Journal of Operational Research*, *119*(1), 147-157. https://doi.org/10.1016/s0377-2217(98)00348-8

Ball, M. O. (2011). Heuristics based on mathematical programming. *Surveys in Operations Research and Management Science*, *16*(1), 21-38. https://doi.org/10.1016/j.sorms.2010.07.001

Barbarosoglu, G., & Ozgur, D. (1999). A tabu search algorithm for the vehicle routing problem. *Computers & Operations Research*, *26*(3), 255-270. https://doi.org/10.1016/s0305-0548(98)00047-1

Bell, J. E., & McMullen, P. R. (2004). Ant colony optimization techniques for the vehicle routing problem. *Advanced Engineering Informatics*, *18*(1), 41-48. https://doi.org/10.1016/j.aei.2004.07.001

Bramel, J., & Simchi-Levi, D. (1995). A location based heuristic for general routing problems. *Operations Research*, *43*(4), 649-660. https://doi.org/10.1287/opre.43.4.649

Breedam, A. V. (1995). Improvement heuristics for the vehicle routing problem based on simulated annealing. *European Journal of Operational Research*, *86*(3), 480-490. https://doi.org/10.1016/0377-2217(94)00064-j

Brito, J., Martínez, F., Moreno, J., & Verdegay, J. (2015). An ACO hybrid metaheuristic for close–open vehicle routing problems with time windows and fuzzy constraints. *Applied Soft Computing*, *32*, 154-163. https://doi.org/10.1016/j.asoc.2015.03.026

Bullnheimer, B., Hartl, R., & Strauss, C. (1999). An improved ant system algorithm for theVehicle Routing Problem. *Annals of Operations Research*, *89*, 319-328. https://doi.org/10.1023/a:1018940026670

Cardoso, P. J., Schütz, G., Mazayev, A., Ey, E., & Corrêa, T. (2015). A solution for a real-time stochastic capacitated vehicle routing problem with time windows. *Procedia Computer Science*, *51*(C), 2227-2236. https://doi.org/10.1016/j.procs.2015.05.501

Chan, Y., & Baker, S. (2005). The multiple depot, multiple traveling salesmen facility-location problem: Vehicle range, service frequency, and heuristic implementations. *Mathematical and Computer Modelling*, *41*(8-9), 1035-1053. https://doi.org/10.1016/j.mcm.2003.08.011

Chen, P., Huang, H. K., & Dong, X. Y. (2010). Iterated variable neighborhood descent algorithm for the capacitated vehicle routing problem. *Expert Systems with Applications*, *37*(2), 1620-1627. https://doi.org/10.1016/j.eswa.2009.06.047

Christofides, N. (1979). *Combinatorial Optimization*. Wiley.

Clarke G., & Wright, J. V. (1964). Scheduling of vehicles from a central depot to a number of delivery points. *Operations Research, 12*(4), 568-581. https://www.jstor.org/stable/167703

Cordeau, J. -F., & Maischberger, M. (2012). A parallel iterated tabu search heuristic for vehicle routing problems. *Computers & Operations Research*, *39*(9), 2033-2050. https://doi.org/10.1016/j.cor.2011.09.021

Cordeau, J. -F., Gendreau, M., Laporte, G., Potvin, J. Y., & Semet, F. (2002). A guide to vehicle routing heuristics. *Journal of the Operational Research Society*, *53*(5), 512-522. https://doi.org/10.1057/palgrave.jors.2601319

Dantzig, G. B., & Ramser, J. H. (1959). The truck dispatching problem. *Management Science*, *6*(1), 80-91. https://doi.org/10.1287/mnsc.6.1.80

Derigs, U., & Vogel, U. (2013). Experience with a framework for developing heuristics for solving rich vehicle routing problems. *Journal of Heuristics*, *20*(1), 75-106. https://doi.org/10.1007/s10732-013-9232-z

Dijkstra, E. W. (1959). A note on two problems in connexion with graphs. *Numerische Mathematik*, *1*(1), 269-271. https://

doi.org/10.1007/bf01386390
Dimitrakos, T., & Kyriakidis, E. (2015). A single vehicle routing problem with pickups and deliveries, continuous random demands and predefined customer order. *European Journal of Operational Research*, *244*(3), 990-993. https://doi.org/10.1016/j.ejor.2015.02.038
Drexl, M. (2012). Rich vehicle routing in theory and practice. *Logistics Research*, *5*(1-2), 47-63. https://doi.org/10.1007/s12159-012-0080-2
Erbao, C., & Mingyong, L. (2009). A hybrid differential evolution algorithm to vehicle routing problem with fuzzy demands. *Journal of Computational and Applied Mathematics*, *231*(1), 302-310. https://doi.org/10.1016/j.cam.2009.02.015
Erbao, C., & Mingyong, L. (2010). The open vehicle routing problem with fuzzy demands. *Expert Systems with Applications*, *37*(3), 2405-2411. https://doi.org/10.1016/j.eswa.2009.07.021
Erbao, C., Mingyong, L., & Hongming, Y. (2014). Open vehicle routing problem with demand uncertainty and its robust strategies. *Expert Systems with Applications*, *41*(7), 3569-3575. https://doi.org/10.1016/j.eswa.2013.11.004
Fisher, M. L., & Jaikumar, R. (1981). A generalized assignment heuristic for vehicle routing. *Networks*, *11*(2), 109-124. https://doi.org/10.1002/net.3230110205
Gendreau, M., Hertz, A., & Laporte, G. (1994). A tabu search heuristic for the vehicle routing problem. *Management Science*, *40*(10), 1276-1290. https://doi.org/10.1287/mnsc.40.10.1276
Ghafurian, S., & Javadian, N. (2011). An ant colony algorithm for solving fixed destination multi-depot multiple traveling salesmen problems. *Applied Soft Computing*, *11*(1), 1256-1262. https://doi.org/10.1016/j.asoc.2010.03.002
Ghannadpour, S. F., Noori, S., & Tavakkoli-Moghaddam, R. (2013). Multiobjective dynamic vehicle routing problem with fuzzy travel times and customers' satisfaction in supply chain management. *IEEE Transactions on Engineering Management*, *60*(4), 777-790. https://doi.org/10.1109/tem.2013.2257794
Gillett, B. E., & Miller, L. R. (1974). A heuristic algorithm for the vehicle-dispatch problem. *Operations Research*, *22*(2), 340-349. https://doi.org/10.1287/opre.22.2.340
Goetschalckx, M., & Jacobs-Blecha, C. (1989). The vehicle routing problem with backhauls. *European Journal of Operational Research*, *42*(1), 39-51. https://doi.org/10.1016/0377-2217(89)90057-x
Hong, S. C., & Park, Y. B. (1999). A heuristic for bi-objective vehicle routing with time window constraints. *International Journal of Production Economics*, *62*(3), 249-258. https://doi.org/10.1016/s0925-5273(98)00250-3
Juan, A. A., Faulin, J., Ruiz, R., Barrios, B., & Caballé, S. (2010). The SR-GCWS hybrid algorithm for solving the capacitated vehicle routing problem. *Applied Soft Computing*, *10*(1), 215-224. https://doi.org/10.1016/j.asoc.2009.07.003
Keskin, M., Laporte, G., & ÇAtay, B. (2019). Electric vehicle routing problem with time-dependent waiting times at recharging stations. *Computers & Operations Research*, *107*, 77-94. https://doi.org/10.1016/j.cor.2019.02.014
Koç, A., & Laporte, G. (2018). Vehicle routing with backhauls: Review and research perspectives. *Computers & Operations Research*, *91*, 79-91. https://doi.org/10.1016/j.cor.2017.11.003
Koç, A., Bektaş, T., Jabali, O., & Laporte, G. (2015). A hybrid evolutionary algorithm for heterogeneous fleet vehicle routing problems with time windows. *Computers & Operations Research*, *64*, 11-27. https://doi.org/10.1016/j.cor.2015.05.004
Koskosidis, Y. A., & Powell, W. B. (1992). Clustering algorithms for consolidation of customer orders into vehicle shipments. *Transportation Research Part B: Methodological*, *26*(5), 365-379. https://doi.org/10.1016/0191-2615(92)90032-r
Kuo, R., Zulvia, F. E., & Suryadi, K. (2012). Hybrid particle swarm optimization with genetic algorithm for solving capacitated vehicle routing problem with fuzzy demand – A case study on garbage collection system. *Applied Mathematics and Computation*, *219*(5), 2574-2588. https://doi.org/10.1016/j.amc.2012.08.092
Laporte, G., & Nobert, Y. (1987). Exact algorithms for the vehicle routing problem. *Surveys in Combinatorial Optimization*, 147-184. https://doi.org/10.1016/s0304-0208(08)73235-3
Laporte. G., & Semet., F. (2001). Classical Heuristics for the Capacitated VRP. In P. Toth, & D. Vigo (Eds.), *The Vehicle Routing Problem Discrete Math (Siam Monographs on Discrete Mathematics and Applications)* (pp. 109-128). Society for Industrial and Applied Mathematics. https://dl.acm.org/doi/10.5555/505847.505852
Letchford, A. N., & Salazar-González, J. J. (2015). Stronger multi-commodity flow formulations of the capacitated vehicle routing problem. *European Journal of Operational Research*, *244*(3), 730-738. https://doi.org/10.1016/j.ejor.2015.02.028
Li, X. Q., Zhang, B., & Li, H. (2006). Computing efficient solutions to fuzzy multiple objective linear programming problems. *Fuzzy Sets and Systems*, *157*(10), 1328-1332. https://doi.org/10.1016/j.fss.2005.12.003
Li, Y., Soleimani, H., & Zohal, M. (2019). An improved ant colony optimization algorithm for the multi-depot green vehicle routing problem with multiple objectives. *Journal of Cleaner Production*, *227*, 1161-1172. https://doi.org/10.1016/j.jclepro.2019.03.185

Lin, C. -C. (2004). A weighted max-min model for fuzzy goal programming. *Fuzzy Sets and Systems*, *142*(3), 407-420. https://doi.org/10.1016/s0165-0114(03)00092-7

Lin, S. (1965). Computer solutions of the traveling salesman problem. *Bell System Technical Journal*, *44*(10), 2245-2269. https://doi.org/10.1002/j.1538-7305.1965.tb04146.x

Lin, S. W., Lee, Z. J., Ying, K. C., & Lee, C. Y. (2009). Applying hybrid meta-heuristics for capacitated vehicle routing problem. *Expert Systems with Applications*, *36*(2), 1505-1512. https://doi.org/10.1016/j.eswa.2007.11.060

Marinakis, Y., & Marinaki, M. (2010). A hybrid genetic – Particle swarm optimization algorithm for the vehicle routing problem. *Expert Systems with Applications*, *37*(2), 1446-1455. https://doi.org/10.1016/j.eswa.2009.06.085

Marinakis, Y., Marinaki, M., & Dounias, G. (2010). A hybrid particle swarm optimization algorithm for the vehicle routing problem. *Engineering Applications of Artificial Intelligence*, *23*(4), 463-472. https://doi.org/10.1016/j.engappai.2010.02.002

Mazzeo, S., & Loiseau, I. (2004). An ant colony algorithm for the capacitated vehicle routing. *Electronic Notes in Discrete Mathematics*, *18*, 181-186. https://doi.org/10.1016/j.endm.2004.06.029

Mehrjerdi, Y. Z., & Nadizadeh, A. (2013). Using greedy clustering method to solve capacitated location-routing problem with fuzzy demands. *European Journal of Operational Research*, *229*(1), 75-84. https://doi.org/10.1016/j.ejor.2013.02.013

Mester, D., & Bräysy, O. (2005). Active guided evolution strategies for large-scale vehicle routing problems with time windows. *Computers & Operations Research*, *32*(6), 1593-1614. https://doi.org/10.1016/j.cor.2003.11.017

Montoya-Torres, J. R., López Franco, J., Nieto Isaza, S., Felizzola Jiménez, H., & Herazo-Padilla, N. (2015). A literature review on the vehicle routing problem with multiple depots. *Computers & Industrial Engineering*, *79*, 115-129. https://doi.org/10.1016/j.cie.2014.10.029

Nazif, H., & Lee, L. S. (2012). Optimised crossover genetic algorithm for capacitated vehicle routing problem. *Applied Mathematical Modelling*, *36*(5), 2110-2117. https://doi.org/10.1016/j.apm.2011.08.010

Osman, I. H. (1993). Metastrategy simulated annealing and tabu search algorithms for the vehicle routing problem. *Annals of Operations Research*, *41*(4), 421-451. https://doi.org/10.1007/bf02023004

Oyola, J., Arntzen, H., & Woodruff, D. L. (2018). The stochastic vehicle routing problem, a literature review, part I: models. *EURO Journal on Transportation and Logistics*, *7*(3), 193-221. https://doi.org/10.1007/s13676-016-0100-5

Pelletier, S., Jabali, O., & Laporte, G. (2019). The electric vehicle routing problem with energy consumption uncertainty. *Transportation Research Part B: Methodological*, *126*, 225-255. https://doi.org/10.1016/j.trb.2019.06.006

Qi, Y., Hou, Z., Li, H., Huang, J., & Li, X. (2015). A decomposition based memetic algorithm for multi-objective vehicle routing problem with time windows. *Computers & Operations Research*, *62*, 61-77. https://doi.org/10.1016/j.cor.2015.04.009

Reimann, M., Doerner, K., & Hartl, R. F. (2004). D-Ants: Savings based ants divide and conquer the vehicle routing problem. *Computers & Operations Research*, *31*(4), 563-591. https://doi.org/10.1016/s0305-0548(03)00014-5

Ropke, S., & Pisinger, D. (2006). A unified heuristic for a large class of vehicle routing problems with backhauls. *European Journal of Operational Research*, *171*(3), 750-775. https://doi.org/10.1016/j.ejor.2004.09.004

Salhi, S., Imran, A., & Wassan, N. A. (2014). The multi-depot vehicle routing problem with heterogeneous vehicle fleet: Formulation and a variable neighborhood search implementation. *Computers & Operations Research*, *52*, 315-325. https://doi.org/10.1016/j.cor.2013.05.011

Schermer, D., Moeini, M., & Wendt, O. (2019). A matheuristic for the vehicle routing problem with drones and its variants. *Transportation Research Part C: Emerging Technologies*, *106*, 166-204. https://doi.org/10.1016/j.trc.2019.06.016

Szeto, W., Wu, Y., & Ho, S. C. (2011). An artificial bee colony algorithm for the capacitated vehicle routing problem. *European Journal of Operational Research*, *215*(1), 126-135. https://doi.org/10.1016/j.ejor.2011.06.006

Taillard, É. (1993). Parallel iterative search methods for vehicle routing problems. *Networks*, *23*(8), 661-673. https://doi.org/10.1002/net.3230230804

Tang, J., Pan, Z., Fung, R. Y., & Lau, H. (2009). Vehicle routing problem with fuzzy time windows. *Fuzzy Sets and Systems*, *160*(5), 683-695. https://doi.org/10.1016/j.fss.2008.09.016

Teodorović, D., & Pavković, G. (1996). The fuzzy set theory approach to the vehicle routing problem when demand at nodes is uncertain. *Fuzzy Sets and Systems*, *82*(3), 307-317. https://doi.org/10.1016/0165-0114(95)00276-6

Tiwari, R., Dharmar, S., & Rao, J. (1987). Fuzzy goal programming — An additive model. *Fuzzy Sets and Systems*, *24*(1), 27-34. https://doi.org/10.1016/0165-0114(87)90111-4

Toth, P., & Vigo, D. (2002). *The vehicle routing problem*. Society for Industrial and Applied Mathematics. https://doi.org/10.1137/1.9780898718515

Toth, P., & Vigo, D. (2003). The granular tabu search and its application to the vehicle-routing problem. *INFORMS Journal on Computing*, *15*(4), 333-346. https://doi.org/10.1287/ijoc.15.4.333.24890

Valle, C. A., Salles Da Cunha, A., Mateus, G. R., & Martinez, L. C. (2009). Exact algorithms for a selective vehicle routing problem where the longest route is minimized. *Electronic Notes in Discrete Mathematics*, *35*, 133-138. https://doi.org/10.1016/j.endm.2009.11.023

van Woensel, T., & Cruz, F. (2009). A stochastic approach to traffic congestion costs. *Computers & Operations Research*, *36*(6), 1731-1739. https://doi.org/10.1016/j.cor.2008.04.008

van, Woensel T., Kerbache, L., Peremans, H., & Vandaele, N. (2007). A Queueing Framework for Routing Problems with Time-dependent Travel Times. *Journal of Mathematical Modelling and Algorithms*, *6*(1), 151-173. https://doi.org/10.1007/s10852-006-9054-1

Wang, Z., & Sheu, J. B. (2019). Vehicle routing problem with drones. *Transportation Research Part B: Methodological*, *122*, 350-364. https://doi.org/10.1016/j.trb.2019.03.005

Wang, Z., & Wang, Z. (2009). A novel two-phase heuristic method for vehicle routing problem with backhauls. *Computers & Mathematics with Applications*, *57*(11-12), 1923-1928. https://doi.org/10.1016/j.camwa.2008.10.045

Wei, L., Zhang, Z., Zhang, D., & Lim, A. (2015). A variable neighborhood search for the capacitated vehicle routing problem with two-dimensional loading constraints. *European Journal of Operational Research*, *243*(3), 798-814. https://doi.org/10.1016/j.ejor.2014.12.048

Xu, J., & Kelly, J. P. (1996). A network flow-based tabu search heuristic for the vehicle routing problem. *Transportation Science*, *30*(4), 379-393. https://doi.org/10.1287/trsc.30.4.379

Yalcin, G. D., & Erginel, N. (2011). Determining weights in multi-objective linear programming under fuzziness. In S. I. Ao, L. Gelman, D. W. L. Hukins, A. Hunter, & A. M. Korsunsky (Eds.), *Proceedings of the World Congress on Engineering Vol II* (pp. 1122-1127). Newswood Limited. http://www.iaeng.org/publication/WCE2011/WCE2011_pp1122-1127.pdf

Yalcin, G. D., & Erginel, N. (2012). A heuristic based on multi objective linear programming under fuzziness for the vehicle routing problem. In C. Kahraman, E. E. Kerre, & F. T. Bozbura (Eds.), *Proceedings of the 10th International FLINS Conference* (pp. 368-373). World Scientific. https://doi.org/10.1142/8564

Yalcin, G. D., & Erginel, N. (2015). Fuzzy multi-objective programming algorithm for vehicle routing problems with backhauls. *Expert Systems with Applications*, *42*(13), 5632-5644. https://doi.org/10.1016/j.eswa.2015.02.060

Yurtkuran, A., & Emel, E. (2010). A new hybrid electromagnetism-like algorithm for capacitated vehicle routing problems. *Expert Systems with Applications*, *37*(4), 3427-3433. https://doi.org/10.1016/j.eswa.2009.10.005

Zachariadis, E. E., & Kiranoudis, C. T. (2010). An open vehicle routing problem metaheuristic for examining wide solution neighborhoods. *Computers & Operations Research*, *37*(4), 712-723. https://doi.org/10.1016/j.cor.2009.06.021

Zarandi, M. H. F., Hemmati, A., & Davari, S. (2011). The multi-depot capacitated location-routing problem with fuzzy travel times. *Expert Systems with Applications*, *38*(8), 10075-10084. https://doi.org/10.1016/j.eswa.2011.02.006

Zhang, X., & Tang, L. (2009). A new hybrid ant colony optimization algorithm for the vehicle routing problem. *Pattern Recognition Letters*, *30*(9), 848-855. https://doi.org/10.1016/j.patrec.2008.06.001

Zheng, Y., & Liu, B. (2006). Fuzzy vehicle routing model with credibility measure and its hybrid intelligent algorithm. *Applied Mathematics and Computation*, *176*(2), 673-683. https://doi.org/10.1016/j.amc.2005.10.013

Zimmermann, H. J. (1978). Fuzzy programming and linear programming with several objective functions. *Fuzzy Sets and Systems*, *1*(1), 45-55. https://doi.org/10.1016/0165-0114(78)90031-3