International Journal of Industrial Engineering & Production Research June 2023 Vol. 34, No. 2: 1-19
DOI: 10.22068/ijiepr.34.2.5

RESEARCH PAPER

Problem Development, Model Formulation and Proposed Algorithm for Capacitated Arc Routing Problem with Priority Edges

Fahimeh Tanhaie[*1] **& Aylin Pakzad**[2]

Received 13 April 2022; Revised 10 April 2023; Accepted 18 April 2023;

ABSTRACT
The capacitated arc routing problem (CARP) is an important vehicle routing problem with numerous real world applications. In this paper, an extended version of CARP, the capacitated arc routing problem with priority edges is presented. The new introduced CARP is more general and closer to reality, and thus is more worthwhile to be solved. In this problem, a set of important priority edges is given and the task is to service of all edges with positive demand in such a way that the higher priority edges are visited as soon as possible. The capacitated arc routing problem with priority edges is an NP-hard problem, so we propose an algorithm that can quickly obtain optimal or near-optimal solution for the defined problem. Another important contribution is that our proposed algorithm is fast and easy to apply. In this paper, through some examples, efficiency of the proposed algorithm has been showed and some guidelines for the future studies have been given.

KEYWORDS: *GRAPH; CARP; Priority edges.*

1. Introduction

The Capacitated Arc Routing Problem (CARP) was introduced by [1] in 1981, that is applicable in many number of real world problems. For example routing of street sweepers, waste collection, winter gritting, the inspection of electric power lines, etc. Golden and Wong showed The CARP is an NP-hard even if the number of vehicles is known, that means reaching to an optimal solution is very hard.

The CARP is a graph that each edge has a non-negative demand and cost. The objective of the problem is to find a collection of routes with minimum total routing cost by considering the following conditions:

1) Vehicles that are located at a central depot have to service all edge demands.
2) Each edge with positive demand is serviced exactly once.
3) The vehicle capacity is limited.
4) All routes start from the depot and terminate at the depot.

During the past years, the CARP and its applications have been studied and many heuristic algorithms and lower bounding procedures have been developed. In this paper, we offer a new algorithm for CARP with priority edges (because the graph is undirected, we say edge instead of arc). In fact, a sequence of priority edges are defined in this problem. A generalization of the CARP is considered, in which a set of important priority edges is given and the task is to service of all edges with positive demand. In the solution the shortest route and the priority of the set of important edge are considered and the higher priority edges are visited as soon as possible.

This paper deals with optimal organization of real word problems such as street snow plowing, solid waste collection, salt gritting and etc., with considering the economic and security effects in CARP. For example, snow plowing problem is a problem where certain streets have to be serviced sooner than others. However, if these streets are

* Corresponding author: ***Fahimeh Tanhaie***
fahimeh.tanhaie@kub.ac.ir

1. *Industrial Engineering Department, Faculty of basic science and Engineering, Kosar university of Bojnord.*
2. *Industrial Engineering Department, Faculty of basic science and Engineering, Kosar university of Bojnord.*

preferred, then we should to service them as soon as possible even if this requires a longer overall route or a higher overall cost. The snow plowing problem is not the only situation where capacity constrained route optimization problems with priority edges appear. There are much more practical situations where a set of important edges with priority restrict the route optimization problem. In fact, the work proposed by this study is a new approach not because of using mathematical programming nor proposed exact algorithm. Rather, this is a new approach because it tries to service of all edges with positive demand in such a way that the higher priority edges are visited as soon as possible. The new introduced CARP is more general and closer to reality, and thus is more worthwhile to be solved. There are several related problems, for example; road gritting, road sweeping, garbage collection and meter reading that the priority edges is necessary. In this paper, we consider a set of important edges with their priority and the task is to find a collection of routes with minimum total routing cost by visiting the higher priority edges as soon as possible. As it is clear in the literature and to the best of our knowledge, the study on CARP with priority edges has been ignored in comparison with other CARP related studies especially in emergency time.

This paper is organized as follows: In Section 2, we give a literature review of the CARP. In Section 3, we explain the problem and in section 4, notation and formulation of the problem is defined. Section 5 covers the proposed algorithm for the CARP with priority edges. In section 6, some examples are illustrated and finally in Section 7, we conclude and offer future research for the area of CARP.

2. Literature Review

In the literature, several different algorithms have been developed for the CARP, while most of them are not exact approaches because the CARP is NP-hard. [2] proved that obtaining optimal solution for the Arc Routing Problem (ARP) is straight forward by modeling the integer linear programming. While, adding the capacity constraints to the integer linear programming model makes it NP-hard. In fact, their result showed that determining the minimal total route length for the CARP is very difficult. One of the exact algorithm for this problem was presented by [3] who transformed CARP examples into capacitated vehicle routing problem (CVRP). They presented a branch-and-cut algorithm to solve the transformed CARP optimally. They added different valid constraints to the basic CVRP and computed the lower bounds for the problem. The result showed the advantages of their approaches in comparison with other branching methods for CVRP. [4] developed a new exact approach based on a branch and bound algorithm. In this paper the subtour elimination method was used and the proposed algorithm applied the node duplication lower bounding method to calculate the lower bounds for the problems.

[5] applied the CARP in the household waste collection in a quarter of Lisbon. They developed two lower-bounding approaches by using of the transportation model and eliminated some of the constraints. They also developed a three-phase heuristic algorithm to obtain a near-optimal solution by using of the lower-bounding approach solution. Their result on the set of test problems showed the advantages of their proposed method. [6] presented a cutting plane algorithm to determine a good lower bound for CARP. They applied aggregated variables to model the problem and added the new constraints for CARP. Finally for the new model a cutting plane algorithm was implemented and the result proved that the proposed lower bound is better than all the lower bounds applied for CARP. [7] defined both profits and costs on arcs in the CARP and considered minimizing the travelling cost and maximizing the total profit in the profitable mixed CARP (PMCARP). [8] developed the classical CARP by considering time-based capacities instead of traditional loading capacities. The increasing application of electric vehicles, indicates the importance of time-capacitated routing problems. They applied random variables on the actual demands on each edge and solved the proposed model by a strategic oscillation heuristic algorithm. [9] presented a memetic algorithm based on Two_Arch2 algorithm to determine a good lower

bound for CARP. They considered vehicle driving routes on urban roads to deliver services by minimizing total cost, makespan, carbon emission and load utilization rate.

[10] proposed an improved evolutionary algorithm for the Extended Capacitated Arc Routing Problem (ECARP) considering the smooth condition. In this paper, the basic specs of CARP, node routing problem was ignored. The result illustrated that the proposed algorithm was very effective and competitive. [11] developed a new problem decomposition factor, named the route cutting off operator that solves the interactions between the tasks in the model. The computations showed that the proposed factor can amend the effectiveness of the decomposition, and lead to significantly further results especially when the mode size is big. [12] presented a work about a support system for the enhancement of the waste collection. They transformed CARP into CVRP, in fact into a node routing equivalent and solved it with different metaheuristics. The result showed the effectiveness of different metaheuristics on real problems and proved that the Variable Neighborhood Search algorithm was the best. The CVRP is an important problem especially in routing and combinatorial optimization problems [13].

In the CARP, it is assumed that the demand of each edge is deterministic, while in real world the demand of each edge is better to describe as a stochastic variable similar to the amount of the household waste. [14] presented the graph that all edge demands were described by random variables and called it Capacitated Arc Routing Problem with Stochastic Demands (CARPSD). They solved their model by a Branch-and-Price algorithm without using of the graph transformation. Also, [15] considered four stochastic factors in the CARP and developed a robust solution based on repair operator that showed the advantages of the proposed approach. [16] presented a branch-and-bound method to solve the instances with 15 to 50 arcs.

The number of research papers on the multiple centers CARP is rather fewer. For example, [17] extended the Existing CARP to the multi depot capacitated arc routing problem (MCARP). They developed some heuristics to solve the MCARP. Their proposed evolutionary approach that was evaluated on several examples with up to 140 nodes and 380 arcs. Also, [18] proposed an improved evolutionary algorithm for the Extended Capacitated Arc Routing Problem (ECARP) considering the multiple centers in CARP. The exploitation of the evolutionary algorithm was applied in this article and three classical heuristics were developed for the ECARP. They used Arc Assignment Priority Information to select an appropriate crossover and mutation operators for their evolutionary algorithm. The result showed the effectiveness of the proposed approach. [19] proposed an extended model of CARP that considered the fixed investment costs and total service time. They developed a hybrid ant colony optimization algorithm (HACOA) to solve the extended CARP. The applied information were priority arcs information and arc cluster information. The algorithm was tested on different sets of problems to evaluate the effectiveness of the proposed approach.

During the past years, the CARP and its applications have developed and several methods were presented to generate a near-optimal solution (see, for example, [20]- [35]). As it is clear in the literature and to the best of our knowledge, the study on CARP with priority edges has been ignored in comparison with other CARP related studies. Considering priority edges is very important especially in emergency time that there is one facility (or postman) with capacity constraint that it (or he) not able to response to edge demands (carry all the mail) at once. That means it (he) has to return several times to its depot (his bike) to can response all demand (to reload mail). Overall, the literature reviews have made notable contributions towards developing models for solving CARP. However, few researches have noted the priority edges, exact algorithm or some real concept such as "higher priority edges are visited as soon as possible " in CARP. The new introduced CARP is more general and closer to reality, and thus is more worthwhile to be solved. Table 1 summarizes some important published papers by taking into account the priority edges, exact algorithm and real concept to prove the novelty of the research.

Tab. 1. An overview of approaches in CARP

References	exact algorithm		priority edges		real concept	
	YES	NO	YES	NO	YES	NO
[3]		*		*		*
[8]		*		*		*
[11]	*		*		*	
[12]		*		*		*
[14]	*			*	*	
[17]		*		*		*
[19]		*		*		*
[25]		*		*	*	
[30]	*		*		*	
[31]		*		*		*
[34]	*			*	*	

There are several related problems, for example; road gritting, road sweeping, garbage collection and meter reading that the priority edges is necessary. In this paper, we consider a set of important edges with their priority and the task is to find a collection of routes with minimum total routing cost by visiting the higher priority edges as soon as possible.

3. Problem Definition

Let G=(V; E) be a non-directed and connected graph where each edge e ∈E has a cost, and a demand. In the CARP with priority edges, a sequence of priority edges are defined, the subset of edge with positive demand is called required edges and the capacity of the vehicle is C. A feasible CARP solution is composed of a set of routes for the considered vehicle that the vehicle starts at the depot and returns to it, because all demands are more than the capacity of the vehicle, it is not possible to response all edge demands at once and vehicle has to pass several cycles. We say edge i has the highest priority when this edge is visited as soon as possible in comparison to other edges in the graph. A vehicle tour is a closed walk starting and ending at the depot where some edges are just traversed by that route while others are also serviced by it. In the definition of this problem, a route deadheads an edge whenever it traverses that edge without servicing it. The CARP with priority edges has the following attributes:

Assumptions:

- The first node in the graph represents the depot.
- The number of circuits in the route is set in advance.
- The graph is a non-directed connected graph G = (V, E), V is the set of vertices and E is the set of undirected edges.

Inputs

- Cost to traverse of each edge
- Demand of service for each edge
- a set of important edges (a sequence of priority edges)
- One vehicle with capacity C is available that has to return several times to its depot to can response all demand.
- d) One depot is available.

Output

- An optimal set of vehicle routes

Objective

- Minimize the cost

Constraints

- A rout must start and return the same depot.
- the total demand in each route it services must not exceed the capacity C
- Demand constraints
- No partial service (All required edges must be serviced exactly once)
- visiting the edges with higher priority as soon as possible

Consider, for example the CARP with priority edges instance denoted by the graph G depicted in Figure 1 to make the graph attributes more clear. Demand and cost are shown by the (C, D) where C is the cost of each edge, positive D is the demand of required edge and a vehicle capacity is C=12. The red edges represent a set of important edges that their priority is clear.

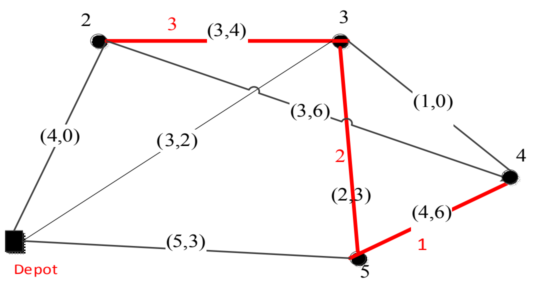


Fig. 1. CARP with priority edges instance

The vehicle has to traverse every required edges at least once to response all edges with positive demands. However, because the demand is more than the capacity of the vehicle, it is not possible to response all edges demand at once and vehicle has to pass several cycles visiting the higher priority edges as soon as possible. A feasible not necessarily optimal CARP solution consisting of two routes is shown in Figures 2 and 3, where solid lines indicate the edges serviced in the route and dotted lines indicate the deadheading edges.

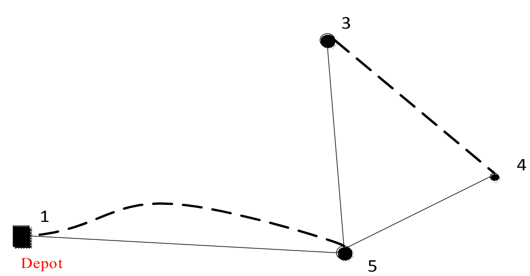


Fig. 2. Feasible solution of CARP (route one)

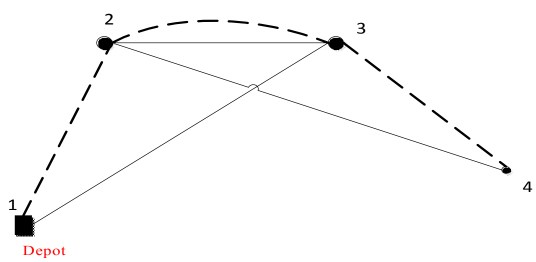
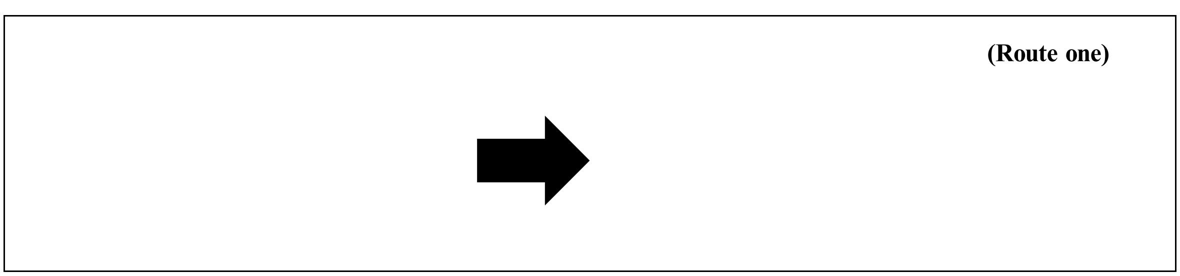


Fig. 3. Feasible solution of CARP (route two)

Because the vehicle cannot response to all demands, at once it has to pass the first route and return to the depot and start the second route (Figure 4).



(Route two)

Fig. 4. An example with feasible solution (red edges: important edges, solid lines: serviced edges, dotted lines: deadheading edges)

4. Problem Notation and Formulation

Different formulation of CARP have been used in the literature. The integer linear programming model is presented Belenguer & Benavent [6] for the undirected CARP. By considering this model and the problem definition, we define the notation and formulation to model the CARP with priority edges.

d_e	Demand of edge e
c_e	Cost of edge e
R	The set of edges with positive demand
P	The set of important edges with their priority numbers
Q	the vehicle capacity
m	The number of predefined circuits
k	Indicator of the circuits, k=1,2,…m
S	The subset of vertices except depot
E(S)	The subset of edges with two vertices like i and j that i is S and j is not
$E^+(S)$	The subset of E(S) with positive demand

To model the non-directed connected graph G = (V, E) some variables need to be introduced (V is the set of vertices and E is the set of undirected edges):

x_{ek} $\quad$ $\textit{The number of times edge e is traversed in round k } (\textit{not served}) \quad e \in E, k = 1,\dots,\mathrm{m}$

$$y_{ek} \quad \begin{cases} 1 & \textit{if edge e is served in round k} \quad e \in E, k = 1,\dots,m \\ 0 & \textit{otherwise} \end{cases}$$

$$y_{sek} \quad \begin{cases} 1 & \textit{if edge e is with both sides in S and e served in round k} \\ 0 & \textit{otherwise} \end{cases}$$

$e \in E, S \subseteq V \setminus \{1\}, k = 1,\dots,m$

z_k^S $\quad$ *an auxiliary variable* $\quad S \subseteq V \setminus \{1\}, k = 1,\dots,m$

With the above definition, the model is:

$$\textit{Minimize} \sum_{e \in E,k} c_e \left(x_{ek} + y_{ek} \right)$$

$$\sum_{k=1}^{m} y_{ek} = 1 \qquad \forall e \in R \tag{1}$$

$$\sum_{e \in E(S)} q_e y_{ek} \le Q_k \qquad k = 1,\dots,m \tag{2}$$

$$\sum_{v=1}^{|p|} y_{vk} \geq k \qquad k=1,2,..|p| \quad ,|p| \textit{ is the number of edges in } p \tag{3}$$

$$\sum_{k=1}^{i} y_{vk} \geq y_{v'k} \qquad i=1,2,..m \quad \forall v,v' \in P, (\text{edge v } \textit{has higher priority in comparison with edge } \text{v}') \tag{4}$$

$$\sum_{e\in E(\mathrm{S})} x_{ek} + \sum_{e\in E^{+}(\mathrm{S})} y_{ek} \geq 2y_{Sfk} \qquad \forall \textit{Edges } f \textit{ that are total in } S, \forall S \neq \varnothing, k=1,\ldots m \tag{5}$$

$$\sum_{e\in E(\mathrm{S})} x_{ek} + \sum_{e\in E^{+}(\mathrm{S})} y_{ek} = 2z_k^S \qquad \forall S \neq \varnothing, k=1,\ldots m \tag{6}$$

$$x_{ek} \geq 0 \textit{ and integer} \qquad \forall e \in E, k=1,\ldots,m \tag{7}$$

$$y_{ek} = 0 \text{ or } 1 \qquad \forall e \in E, k=1,\ldots,m \tag{8}$$

$$z_k^S \geq 0 \textit{ and integer} \qquad \forall S \subseteq V \setminus \{1\}, k=1,\ldots,m \tag{9}$$

The outcome of the model is a set of vehicle routes that serve all edges with a positive demand. The objective function minimize total distance traveled. First equation ensures that each required edge will be serviced exactly once. Vehicle capacity is not violated in second equation .Third and fourth equations are related to the CARP with priority edges state that higher priority edges are visited as soon as possible and if edge v has higher priority in comparison with edge v', the demand of v has to response earlier in comparison to edge v'.

Fifth equation is called connectivity constraint that states if a route service edge e, then it must to go at least one time into S and at least one time out of S. The sixth constraint ensures that the obtained routes are rounds and an even number guarantees this constraint. It states that a route has to go out of a subset as many times as it goes into a subset.

Several different models have been developed for CARP, while most of them are nonlinear programming. There are two integer linear programming models formulated for the undirected CARP that we added some additional constraints to the Belenguer & Benavent [6]] formulation to model the CARP with priority edges. These extra constraints make that solving the problems is NP-hard, because the number of constraints grows exponential. In this model, there are $|R|*|S|*m$ constraints in equation (5) and $|S|*m$ constraints in equation (6) that $|R|$ is the number of edges in R and $|S|$ is the number of subsets. Therefore, this formulation has many variables and constraints and the number of constraints grows exponential if we add one node to a graph. Also, the number of subsets in a graph is $2^{|v-1|}$, that explained the exponential growth.

This is too big to implement and solve in a reasonable time. It is clear that the above model is complex and even impossible to obtain the optimal route length for the CARP with edges priority especially when the graph is large. Due to this complex solving model, in the next section we propose an algorithm to obtain optimal or near optimal solution for CARP with edges priority.

5. The Proposed Algorithm

Given an undirected graph G(V,E), and a sequence of priority edges H_1 , H_2 , …H_h. The objective is to find a collection of routes, which is

(1) Minimum total routing cost, and

(2) Among the solutions which satisfy (1), find the route that visits H_1 as soon as possible,

(3) Among the solutions which satisfy (2), find the route that visits H_2 as soon as possible,

….

(h) Among the solutions which satisfy (h-1), find the route that visits H_h as soon as possible.

As mentioned in before sections, in CARP with priority edges, if some edges are preferred, then we should to service them as soon as possible even if this requires a longer overall route or higher overall cost. We can assume without loss of generality, there is at least one edge with priority in each route or based on the remained demands, the route for the vehicle is the last route. In addition, the service time of any edge with positive demand is zero, it means the time of passing any edge, with or without servicing it, is identical. We propose an algorithm, which is a combination of the Dijkstra's algorithm for computing shortest paths and some heuristic steps. Some parameters are defined as follows:

C	The capacity of vehicle
h	The number of edges with priority
$n-h$	The number of edges without priority
$C_i \quad i=1,2,...,n$	Cost of edge i
$H_j \quad j=1,2,...,h \quad h<n$	Demand of edge j (edge with priority)
$D_i \quad i=h+1,h+2,...,n$	Demand of edge i (edge without priority)

The algorithm is defined in following steps:

Step 1. Denote p, the edge with highest priority that has not been serviced and $p'=p$

Step 2. Find maximum t that satisfies the following formula.

$$\sum_{j=p}^{t} H_j \le C \qquad t \in \{1,2,..,h\}$$

Therefore, in this route the vehicle can satisfy the demand of t edges with higher priority as soon as possible.

Step 3. Dijkstra's algorithm is used for finding the shortest paths between two vertexes. Find the shortest path between the depot and the vertex of the one side of edge p (the side that path from depot to it is shorter) and call it walk p. If p=t go to step 5, otherwise go to step 4.

Step 4. Find the shortest path between another side of edge p and the vertex of the one side of edge p+1 (the side that path from depot to it is shorter) and call it walk p+1. Denote p=p+1. If p=t go to step 5, otherwise repeat this step.

Step 5. Find the shortest path between another side of edge t and depot, call it walk p+1.

Step 6. Find the maximum edges (maximum number of members of set f) in walk p' to walk $p'+t$, which satisfy the following formula.

$$f = \left\{ i \,\middle|\, \sum_{i=h+1}^{n} D_i \le C - \sum_{p=p'}^{p} H_p \qquad i \in edges\ in\ walk\ 1\ to\ walk\ p+1 \right\}$$

Therefore, the vehicle satisfy the demand of edges p' (or $p', p'+1,.., p'+t-1 \quad if\ p' \ne t$) and the edges in the set f.

Step 7. Change the demand of satisfied edges to zero.

Step 8. If p=h go to step 9, otherwise go to step 1.

Step 9. If all edges with positive demand were served go to step 11, otherwise go to step 10.

Step 10. Find the shortest path between depot and each of remained edges and satisfy the demand of them.

Step 11. Finish

This algorithm can reach to the optimum solution, but in some situation generates the near optimum solution because of step 6. In this step algorithm tries to service the maximum edges, while sometimes servicing the edge with maximum demand and maximum cost, causes to obtain optimum solution. In fact, the cost of this edge is high and it is better to service it in this route and vehicle does not pass it again. In this situation, the algorithm can obtain near optimum solution not optimum solution.

5.1. Accuracy of the proposed algorithm

In the CARP with priority edges, some edges should be serviced as soon as possible, so the algorithm considers the edge with highest priority that need service in step 1. Now the vehicle has to go to this edge based on the shortest path, so the Dijkstra's algorithm finds the shortest path between depot and one side the edge with highest priority in step 3. In fact, the vehicle not only services the edge with highest priority as soon as possible, but also passes the shortest path. If the remained capacity of the vehicle based on the $\sum_{j=p}^{t} H_j \leq C$ in step 2 allows to service the next highest priority edge, Dijkstra's algorithm finds the shortest path between the edge with highest priority and the next highest priority edge in step 4. Otherwise, Dijkstra's algorithm finds the shortest path between the edge with highest priority and depot in step 5.

Until this step proposed algorithm finds the best paths and services important edges as soon as possible, but in step 6, when the remained capacity is assigned to other edges that need service in the found path, the algorithm may generate the optimum or near optimum solution because of the following formula in step 6:

$$f = \left\{ i \,\middle|\, \sum_{i=h+1}^{n} D_i \leq C - \sum_{p=p'}^{p} H_p \right\}$$

In this formula algorithm tries to service the maximum edges (it is taken into account that the capacity cannot be exceeded), while sometimes servicing the edges with maximum demand and maximum cost causes to obtain optimum solution. Similarly, the construction of other routes is repeated until there are no possible routes left. So, the proposed algorithm finds the best path from depot to the determined edges (and vice versa) based on the Dijkstra's algorithm and the important edges have been serviced as soon as possible.

5.2. Complexity of the proposed algorithm

Let G=(V; E) be a non-directed and connected graph with following parameters:

$n = |V| =$ Number of vertices in graph G.

$m = |E| =$ Number of edges in graph G.

In the worst case, the considered graph has m important (priority) edges and for reaching and servicing each edge based on the proposed algorithm, we need to apply Dijkstra's algorithm two times(in reaching to the edge and coming back to depot). The complexity of the Dijkstra's algorithm is O(n^2), so we have:

There are at most m priority edges (m).

In each iteration (for each priority edge) for reaching to the priority edge, Dijkstra's algorithm takes O(n^2) time ($f(n^2)$).

In each iteration (for each priority edge), servicing each edge of G takes O(1) time.

In each iteration (for each priority edge) for reaching to the depot, Dijkstra's algorithm takes O(n^2) time ($f(n^2)$).

$$m*\left(f(n^2)+1+f(n^2)\right) = m*\left(2f(n^2)+1\right) \in \mathrm{O}\left(mn^2\right)$$

Thus, total time complexity of the proposed algorithm is O(mn^2).

6. The Proposed Algorithm for Some Problems

6.1. First problem

We illustrate the proposed algorithm with some examples. For the first example, the solution is obtained step by step based on the algorithm but for the others we only shows the final solutions. Consider the CARP instance defined by the graph G depicted in Figure 5.

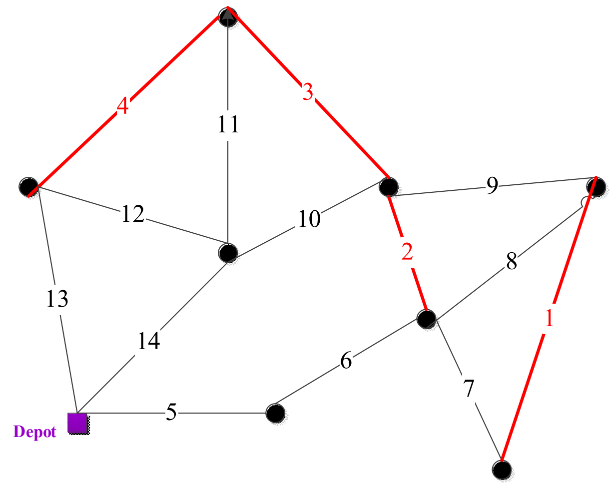


Fig. 5. First problem

The vehicle capacity is C=15. The red edges represent a set of important edges that their priority numbers are specified on them. Demand and cost are shown in table 2.

Tab. 2. Demand and cost of edges for first problem

Edge number	(Cost, Demand)	Edge number	(Cost, Demand)
1	(4, 11)	8	(4, 3)
2	(3, 5)	9	(3, 0)
3	(2, 6)	10	(2, 4)
4	(6, 6)	11	(3, 4)
5	(2, 2)	12	(1, 1)
6	(3, 3)	13	(3,4)
7	(2, 2)	14	(5, 6)

We want to solve this example base on the proposed algorithm in the previous section.
Step 1. p=1, $p'=1$

Step 2. $\sum_{j=p}^{t} H_j \le C \rightarrow H_1 \le C \rightarrow 11 \le 15 \rightarrow t=1$

Step 3. Walk 1: edges 5, 6, 7. Because p=t=1, we go to step 5.
Step 5. Walk 2: edges 8, 6, 5
Step 6.

$$f=\left\{ i \left| \sum_{i=h+1}^{n} D_i \le C - \sum_{p=p'}^{p} H_p \rightarrow D_5 + D_7 \le C - H_1 \rightarrow 2+2 \le 15-11 \right. \right\}$$

So, the vehicle satisfy the demand of edges 1, 5, 7 in the first route as it shown in figure 6, where solid lines indicate the edges serviced in the route and dotted lines indicate the deadheading edges.

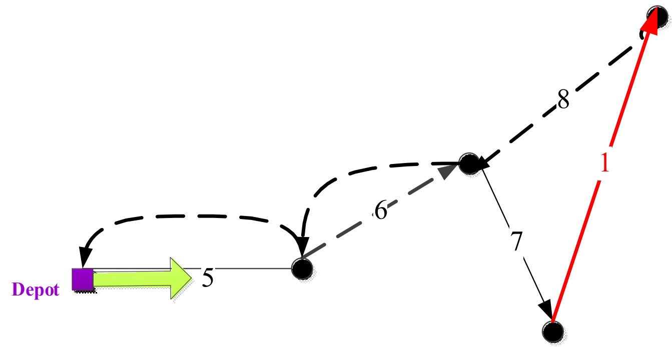


Fig. 6. First route: 5, 6, 7, 1, 8, 6, 5

Step 7. Change the demand of satisfied edges to zero.

$H_1=0$, $D_5=0$, $D_7=0$

Step 8. Because p≠h go to step 1.

Step 1. p=2, $p'=2$

Step 2. $\sum_{j=p}^{t} H_j \le C \rightarrow H_2 + H_3 \le C \rightarrow 5+6 \le 15 \rightarrow t=3$

Step 3. Walk 2: edges 5, 6. Because p≠t, we go to step 4.

Step 4. Walk 3:0.

P=2+1=3

Because p=t go to step 5.

Step 5. Walk 4: edges 11, 12, 13.

Step 6.

$$f = \left\{ i \,\middle|\, \sum_{i=h+1}^{n} D_i \le C - \sum_{p=p'}^{p} H_p \rightarrow D_6 + D_{12} \le C - H_2 - H_3 \rightarrow 3+1 \le 15-5-6 \right\}$$

So, the vehicle satisfy the demand of edges 2, 3,6,12 in the second route as it shown in figure 7, where solid lines indicate the edges serviced in the route and dotted lines indicate the deadheading edges.

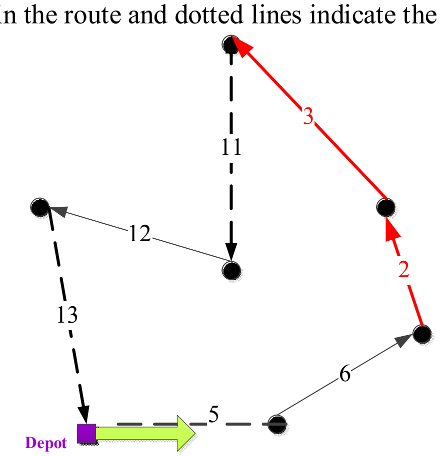


Fig. 7. Second route: 5, 6, 2, 3, 11, 12, 13

Step 7. Change the demand of satisfied edges to zero.

$H_2=0$, $H_3=0$, $D_6=0$, $D_{12}=0$

Step 8. Because p≠h go to step 1.

Step 1. p=4, $p'=4$

Step 2. $\sum_{j=p}^{t} H_j \leq C \rightarrow H_4 \leq C \rightarrow 6 \leq 15 \rightarrow t = 4$

Step 3. Walk 4: edges 13. Because p=t=1, we go to step 5.

Step 5. Walk 5: edges 11, 14.

Step 6.

$$f = \left\{ i \middle| \sum_{i=h+1}^{n} D_i \leq C - \sum_{p=p'}^{p} H_p \rightarrow D_{13} + D_{11} \leq C - H_4 \rightarrow 4 + 4 \leq 15 - 6 \right\}$$

So, the vehicle satisfy the demand of edges 4, 13,11 in the third route as it shown in figure 8, where solid lines indicate the edges serviced in the route and dotted lines indicate the deadheading edges.

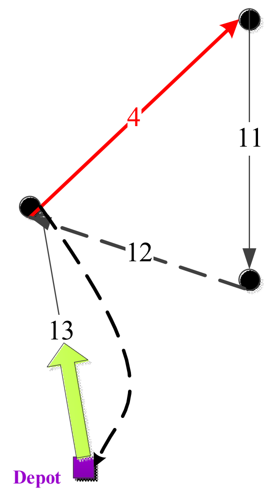


Fig. 8. Third route: 13, 4, 11, 12, 13

Step 7. Change the demand of satisfied edges to zero.

H_4=0, D_{13}=0, D_{11}=0

Step 8. Because p=h go to step 9.

Step 9. Because all edges with positive demand were not served go to step 10.

Step 10. Shortest path between depot and remained edges: edges 14, 10, 9, 8, 6, 5.

So, the last route is produced as it shown in figure 9, where solid lines indicate the edges serviced in the route and dotted lines indicate the deadheading edges.

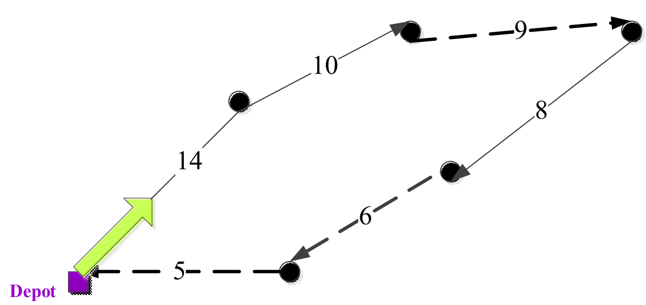


Fig. 9. Last route: 14, 10, 9, 8, 6, 5

Step 11. Finish

Table 3 summarizes the results for the problem. We see that the overall cost of the generated problem with the proposed algorithm is 72. In order to evaluate our algorithm, we considered all states of solutions and found the optimum solution for the problem that was very time consuming. The optimum solution was 72 that is equal to the solution of the proposed algorithm.

Tab. 3. The results of first problem

Route number	Routes(edges)	Total demand of each route	Total cost of each route	Proposed algorithm solution	Optimum solution based on all states
1	5, 6, 7, 1, 8, 6, 5	$15 \leq 15$	20	72(optimum)	72
2	5, 6, 2, 3, 11, 12, 13	$15 \leq 15$	17		
3	13, 4, 11, 12, 13	$14 \leq 15$	16		
4	14, 10, 9, 8, 6, 5	$13 \leq 15$	19		

A vehicle route is a closed walk starting and ending at the depot where some edges are just traversed (black numbers) by that route while others are also serviced by it (red numbers). As it is seen in table 3, the total demand in each route does not exceed the capacity C and the priority edges has been visited as soon as possible in comparison to other edges in the graph. The solution is composed of a set of four routes for the considered vehicle that the vehicle starts at the depot and returns to it. Because the demands of all edges are more than the capacity of the vehicle, it is not possible to response all edge demands at once and vehicle has to pass several cycles.

6.2. Solutions of the generated problems

In order to apply the proposed algorithm, the solution for some problems were determined by the proposed algorithm. The problems were generated and the details of them are in appendix. Table 4 indicates the solutions of the generated problems by the proposed algorithm.

Tab. 4. The problems solution

Problems	Route number	Routes(edges)	Total demand of each route	Total cost of each route	Proposed algorithm solution	Optimum solution based on all states
Problem #2 C=10	1	5, 6, 1, 4	$8 \leq 10$	18	51(optimum)	51
	2	3, 2, 6, 5	$10 \leq 10$	15		
	3	3, 7, 1, 8	$8 \leq 10$	18		
Problem #3 C=12	1	5, 1, 3, 6	$11 \leq 12$	11	50(near optimum)	49
	2	6, 2, 4, 7, 1, 5	$11 \leq 12$	21		
	3	6, 8, 7, 1, 5	$12 \leq 12$	18		
Problem #4 C=15	1	5, 1, 8, 2, 5	$13 \leq 15$	8	23(optimum)	23
	2	4, 3, 8, 7, 3, 6	$9 \leq 15$	15		
Problem #5 C=15	1	4, 1, 2, 7	$13 \leq 15$	11	37(optimum)	37
	2	6, 3, 9, 5	$13 \leq 15$	14		
	3	6, 8, 2, 7	$14 \leq 15$	12		
Problem #6 C=14	1	3, 9, 1, 8, 6	$13 \leq 14$	18	58(near optimum)	56
	2	5, 2, 8, 6	$12 \leq 14$	15		
	3	3, 9, 9, 7, 4, 5	$13 \leq 14$	25		
Problem #7 C=20	1	3, 1, 7, 9	$18 \leq 20$	13	56(optimum)	56
	2	9, 7, 5, 2, 4, 3	$18 \leq 20$	23		
	3	8, 6, 4, 3	$16 \leq 20$	20		

In each route some edges are just traversed (black numbers) by that route while others are also serviced by it (red numbers). The results reported in table 4 clearly shows that the total demand in each route does not exceed the capacity C and the priority edges has been visited as soon as possible in comparison to other edges in the graph, so the proposed algorithm generates optimum or near optimum solution for all sample problems. In problem 3 the routes of optimum solution are route 1: 5, 1, 3, 6, route 2: 6, 2, 4, 7, 1, 5, route 3: 6, 8, 4, 1, 3, 6 and in problem 6 the routes of optimum solution are route 1: 3, 9, 1, 8, 6, route 2: 5, 2, 8, 6 and route 3: 5, 4, 7, 3, 6, 6.

7. Conclusion and Future Research

In this paper, we developed an extended version of CARP, the CARP with priority edges. In this problem a set of important priority edges is given and the task is to service of all edges with positive demand in such a way that the higher priority edges are visited as soon as possible. As it is clear in the literature and to the best of our knowledge, the study on CARP with priority edges has been ignored in comparison with other CARP related studies. Considering priority edges is very important especially in emergency time. The new introduced CARP is more general and closer to reality, and thus is more worthwhile to be solved. The mathematical model is complex and even impossible to obtain the optimal route length. Due to this complex model, we proposed an algorithm to obtain optimum or near optimum solution for CARP with priority edges. Our proposed algorithm was applied to some generated problems and the result showed that the algorithm finds quickly acceptable solutions for all generated problems of CARP which has additional property that it visits the first priority edge as soon as possible. Another important contribution is that our proposed algorithm is fast and easy to apply. Although no exact approach has been proposed for CARP with edges priority in this paper, the formal definition of the problem creates the foundation of further research works, and the definition about the problem gives some guidelines for the future studies. This work can be extended under several directions such as developing different algorithms to identify the new and better contributions with respect to quality and speed.

8. Acknowledgments

The authors would like to thank the anonymous reviewers of this paper for their thought-provoking and insightful comments and corrections. The Study was supported by Kosar University of Bojnord with the grant number 0108271901.

References

[1] B. Golden and R. Wong, "Capacitated arc routing problems," Networks, Vol. 11, No. 3, (1981), pp. 305-315.

[2] J. Karskens and R. Bekker, Mail Delivery Problem, Netherlands: VU University Amsterdam, (2013).

[3] R. Baldacci and V. Maniezzo, "Exact methods based on node-routing formulations for undirected arc-routing problems," Network, Vol. 47, No. 1, (2005), pp. 52-60.

[4] Y. Saruwatari, R. Hirabayashi and N. Nishida, "Subtour Elimination Algorithm for the Capacitated Arc Routing Problem," in Operations Research Proceedings, (1992).

[5] "M. Mourão and M. Almeida, 'Lower-bounding and heuristic methods for a refuse collection vehicle routing problem," European Journal of Operational Research, Vol. 121, No. 2, (2000), pp. 420-434.

[6] J. Belenguer and E. Benavent, "A cutting plane algorithm for the capacitated arc routing problem," omputers & Operations Research, Vol. 30, No. 5, (2003), pp. 705-728.

[7] E. Benaven, A. Corberán, L. Gouveia and M. Mourão, "Profitable mixed capacitated arc routing and related problems," TOP, Vol. 23, No. 1, (2014), pp. 244-274.

[8] P. Keenan, J. Panadero, A. Juan and R. Martí, "A strategic oscillation simheuristic for the Time Capacitated Arc Routing Problem with stochastic demands,"

Computers & Operations Research, Vol. 133, (2021), p. 105377.

[9] B. Cao, W. Zhang, X. Wang, J. Zhao, Y. Gu and Y. Zhang, "A memetic algorithm based on two_Arch2 for multi-depot heterogeneous-vehicle capacitated arc routing problem," Swarm and Evolutionary Computation, Vol. 63, (2022), p. 100864.

[10] L. X. Zhao, X. Zuo, J. Yuan and X. Yao, "Memetic algorithm with non-smooth penalty for capacitated arc routing problem," Knowledge-Based Systems, Vol. 220, (2021), p. 106957.

[11] Y. Zhang, Y. Mei, B. Zhang and K. Jiang, "Divide-and-conquer large scale capacitated arc routing problems with route cutting off decomposition," Information Sciences,, Vol. 553, (2021), pp. 208-224.

[12] V. Maniezzo and M. Roffilli , "Algorithms for Large Directed Capacitated Arc Routing Problem Instances," Recent Advances in Evolutionary Computation for Combinatorial Optimization, (2008), pp. 259-274.

[13] B. Chandran and S. Raghavan, "Modeling and Solving the Capacitated Vehicle Routing Problem on Trees', The Vehicle Routing Problem," Latest Advances and New Challenges, (2008), pp. 239-261.

[14] C. Christiansen, J. Lysgaard and S. Wøhlk, "A Branch-and-Price Algorithm for the Capacitated Arc Routing Problem with Stochastic Demands," Operations Research Letters, Vol. 37, No. 6, (2009), pp. 392-398.

[15] Y. Mei, K. Tang and X. Yao, "Evolutionary Computation for Dynamic Capacitated Arc Routing Problem," Evolutionary Computation for Dynamic Optimization Problems, (2013), pp. 377-401.

[16] R. Y. S. a. N. N. Hirabayashi, "Tour construction algorithm for the capacitated arc routing problem," Asia-Pacific Journal of Operational Research, Vol. 9, (1992), pp. 155-175.

[17] L. Xing, P. Rohlfshagen, Y. Chen and Y. Xin, "An Evolutionary Approach to the Multidepot Capacitated Arc Routing Problem," IEEE Transactions on Evolutionary Computation, Vol. 14, No. 3, (2010), pp. 356-374.

[18] H. Xu, C. Zhang, Y. Tan and J. Lu, "An improved evolutionary approach to the Extended Capacitated Arc Routing Problem," Expert Systems with Applications, Vol. 38, No. 4, (2011), pp. 4637-4641.

[19] L. Xing, P. Rohlfshagen and Y. Chen, "A Hybrid Ant Colony Optimization Algorithm for the Extended Capacitated Arc Routing Problem," IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics), Vol. 41, No. 4, (2011), pp. 1110-1123.

[20] A. Corberán and C. Prins, "Recent results on Arc Routing Problems: An annotated bibliography," Network, (2009).

[21] S. Wøhlk, "A Decade of Capacitated Arc Routing', the Vehicle Routing Problem," Latest Advances and New Challenges, Vol. 43, (2008), pp. 29-48.

[22] R. Arakaki and F. Usberti, "An efficiency-based path-scanning heuristic for the capacitated arc routing problem," Computers & Operations Research, Vol. 103, (2018), pp. 288-295.

[23] E. Tirkolaeea, M. Alinaghian, A. Rahmani and M. Bakhs, "An improved ant colony optimization for the multi-trip Capacitated Arc Routing Problem," Computers & Electrical Engineering, (2018).

[24] Y. Chen and H. Hao, "Two phased hybrid local search for the periodic capacitated arc routing problem," European Journal of Operational Research, Vol. 264, (2018), pp. 55-65.

[25] Y. Zhang, Y. Mei, K. Tang and K. Jiang, "Memetic algorithm with route decomposing for periodic capacitated arc routing problem," Applied Soft Computing, Vol. 52, (2017), pp. 1130-1142.

[26] E. Tirkolaee, L. Mahdavi and M. Esfahani, "A robust periodic capacitated arc routing problem for urban waste collection considering drivers and crew's working time," Waste Management, Vol. 76, (2018), pp. 138-146.

[27] Y. Chen, J. Hao and F. Glover, "A hybrid metaheuristic approach for the capacitated arc routing problem," European Journal of Operational Research, Vol. 253, (2016), pp. 25-39.

[28] E. Willemse and J. Joubert, "Splitting procedures for the Mixed Capacitated Arc Routing Problem under Time restrictions with Intermediate Facilities," Operations Research Letters, Vol. 44, (2016), pp. 569-574.

[29] H. Eiselt, M. Gendreau and G. Laporte, "Arc Routing Problems, Part II: The Rural Postman Problem," Operations Research, Vol. 43, No. 3, (1995), pp. 399-414.

[30] A. Khajepoura, M. Sheikhmohammadyab and E. Nikbakhsh, "Field path planning using capacitated arc routing problem," Computers and Electronics in Agriculture, Vol. 173, (2020.

[31] H. Zbib and G. Laporte, "The commodity-split multi-compartment capacitated arc routing problem," Computers & Operations Research, Vol. 122, (2020).

[32] Y. Mingzhu, J. Xin, Z. Zizhen, Q. Hu and L. Qidong, "The split-delivery mixed capacitated arc-routing problem: Applications and a forest-based tabu search approach," Transportation Research Part E: Logistics and Transportation Review, Vol. 132, (2019), pp. 141-162.

[33] J. Willemse and W. Joubert, "Efficient local search strategies for the Mixed Capacitated Arc Routing Problems under Time Restrictions with Intermediate Facilities," Computers & Operations Research, Vol. 105, (2019), pp. 203-225.

[34] M. Araújo, P. Silva, A. Subramanian and E. Fonseca, "On solving the capacitated routing and spectrum allocation problem for flexgrid optical networks," Computer Networks, Vol. 181, (2020).

[35] G. Marques, R. Sadykov, J. Deschamps and R. Dupas, "An improved branch-cut-and-price algorithm for the two-echelon capacitated vehicle routing problem," Computers & Operations Research, Vol. 114, (2020).

Appendix:

The problems 2, 3, 4, 5, 6 and 7 were generated and the details of them are in following tables. Example 2 defined by the graph G depicted in Figure 10.

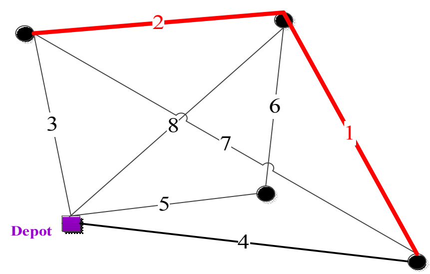


Fig. 10. Example 2

The vehicle capacity is C=10. The red edges represent a set of important edges that their priority numbers are specified on them. Demand and cost are shown in table 4.

Tab. 4. Demand and cost of edges

Edge number	(Cost, Demand)	Edge number	(Cost, Demand)
1	(4, 6)	5	(2, 5)
2	(4, 5)	6	(6, 1)
3	(3, 4)	7	(5, 2)
4	(6, 1)	8	(6, 2)

Example 3 defined by the graph G depicted in Figure 11.

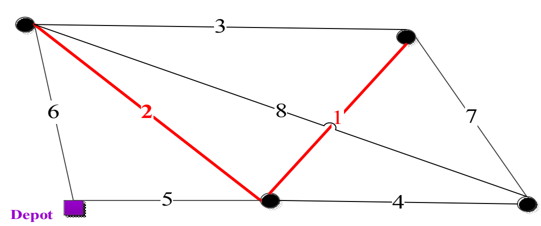


Fig. 11. Example 3

The vehicle capacity is C=12. The red edges represent a set of important edges that their priority numbers are specified on them. Demand and cost are shown in table 5.

Tab. 5. Demand and cost of edges

Edge number	(Cost, Demand)	Edge number	(Cost, Demand)
1	(1, 6)	5	(3, 6)
2	(3, 7)	6	(2, 2)
3	(5, 3)	7	(7, 3)
4	(5, 2)	8	(5, 3)

Example 4 defined by the graph G depicted in Figure 12.

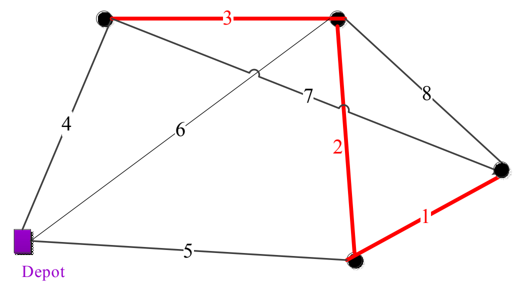


Fig. 12. Example 4

The vehicle capacity is C=15. The red edges represent a set of important edges that their priority numbers are specified on them. Demand and cost are shown in table 6.

Tab. 6. Demand and cost of edges

Edge number	(Cost, Demand)	Edge number	(Cost, Demand)
1	(4, 7)	5	(1, 1)
2	(1, 5)	6	(3, 3)
3	(3, 4)	7	(3, 2)
4	(2, 0)	8	(1, 4)

Example 5 defined by the graph G depicted in Figure 13.

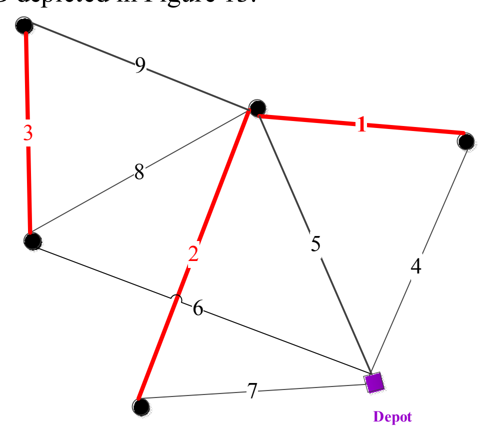


Fig. 13. Example 5

The vehicle capacity is C=15. The red edges represent a set of important edges that their priority numbers are specified on them. Demand and cost are shown in table 7.

Tab. 7. Demand and cost of edges

Edge number	(Cost, Demand)	Edge number	(Cost, Demand)
1	(4, 6)	6	(3, 8)
2	(3, 5)	7	(2, 5)
3	(5, 5)	8	(4, 1)
4	(2, 2)	9	(2, 3)
5	(4, 5)		

Example 6 defined by the graph G depicted in Figure 14.

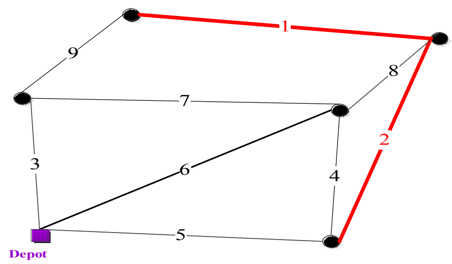


Fig. 14. Example 6

The vehicle capacity is C=14. The red edges represent a set of important edges that their priority numbers are specified on them. Demand and cost are shown in table 8.

Tab. 8. Demand and cost of edges

Edge number	(Cost, Demand)	Edge number	(Cost, Demand)
1	(4, 8)	6	(4, 3)
2	(4, 7)	7	(5, 1)
3	(2, 5)	8	(3, 2)
4	(4, 2)	9	(5, 5)
5	(4, 5)		

Example 7 defined by the graph G depicted in Figure 15.

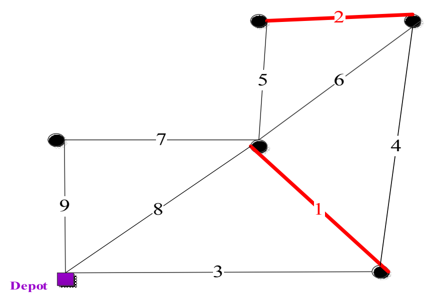


Fig. 15. Example 7

The vehicle capacity is C=20. The red edges represent a set of important edges that their priority numbers are specified on them. Demand and cost are shown in table 9.

Tab. 9. Demand and cost of edges

Edge number	(Cost, Demand)	Edge number	(Cost, Demand)
1	(4, 11)	6	(5, 5)
2	(6, 10)	7	(2, 4)
3	(4, 7)	8	(6, 4)
4	(5, 4)	9	(3, 3)
5	(3, 4)		

Follow this article at the following site:

Fahimeh Tanhaie & Aylin Pakzad: Problem development, model formulation and proposed algorithm for CARP with priority edges. IJIEPR 2023; 34 (2) :1-19
URL: http://ijiepr.iust.ac.ir/article-1-1470-en.html