Production Systems and Information Engineering
Volume 10 (3), pp. 51–64.
https://doi.org/10.32968/psaie.2022.3.5

SOFTWARE FOR THE GENERALIZATION OF THE VEHICLE ROUTING PROBLEM

ANITA AGÁRDI
University of Miskolc
Hungary Institute of Information Technology
agardianita@iit.uni-miskolc.hu

Abstract. This article presents a software solution to the generalization of the Vehicle Routing Problem. The Vehicle Routing Problem is a logistics task, the purpose of which is to transport goods and services to a given place at a given time in a cost-effective manner. Many versions of the task have developed over the years, which model constantly changing, complicated systems. The article details the software solution of a model that simulates a general system. The software representation contains the following components: vector describing the order of nodes, the matrix describing the assignment of vehicles-nodes-products, vector describing the assignment of vehicle-recharger stations, vector describing the vehicle-start-end-node assignment, vector describing level, vector describing period, solution description vector. The following optimization algorithms were used in the article: Ant Colony System, Genetic Algorithm, Simulated Annealing and Tabu Search. The article presents the literature of the Vehicle Routing Problem, the software solution of the general system, the applied algorithms and their running results through a case study.

Keywords: Vehicle Routing Problem, generalized model, heuristics

1. Introduction

During the basic VRP [1], the vehicles start from a depot and visit the customers and deliver the products to them. They then return to the depot at the end of their journey. The goal of the task is to minimize the traveled distance. VRP has evolved over the years in many versions. Capacitated VRP [2] is a case, where the vehicles have a capacity limit for the goods to be transported. The goods can be of one type (VRP with single types of products [3]) or of several types (VRP with multiple types of products [3]). Vehicles can be homogeneous [4] or heterogeneous [5]. If the system contains a single depot, then the system is called a Single Depot VRP [6], if more, then Multiple-Depot VRP [7]. Two-Echelon VRP [8] means that there are not only depots and customers in the system, but also satellites. In this case, the goods will be first delivered to one of the satellites, and then from the satellites to the customers. Different depot-satellite and satellite-customer vehicles transport the goods on each level. Customers can also have a time window, in which case they must be visited within a time interval. The time window can be soft [9], hard [10] or multiple [11]. The soft time window means that it can be ignored: the customer can be visited outside the time interval, but the solution receives a penalty point. During the hard time window, the customer can only be visited within the time interval. During the multiple time window, several time windows are assigned to a customer, and the customer can be visited in one of the time windows. Open VRP [12] means that vehicles do not have to return to the depot after visiting customers. Vehicles can be environmentally friendly (Environmentally Friendly VRP [13]). The VRP can also contain stochastic [14] or fuzzy data [15], in which case we call the task Stochastic or Fuzzy VRP.

2. Software for the generalization of the Vehicle Routing Problem

In this section, the software for the generalized VRP model is presented. The mathematical formulation of the generalized model is presented in paper [16]. The generalized model contains the following component: number of positions, levels, vehicles, periods, products, services. The attributes of the system is divided into groups. The node group consists of the travel time, travel distance, reliability, route status between the nodes and the type of the node. The vehicles group contains the following: capacity constraint, fuel consumption, recharger time, own or borrowed vehicle, rental fee and the maximum distance with full tank. The time group contains the followings: service handling, packing, unpacking, loading, unloading, fixed capital, administration, quality control and time window. The product group contains the following attributes: capacity constraint, product demand of the node, the prices of the product, given order of product, products handling together, storage level of the nodes. The cost group contains component in connection with cost: packing, unpacking, loading, unloading, administrative, quality control cost. The functional parameter group has the following attributes: inter-depot route, delivery, pickup, soft time window, open route. Also two test cases are presented in paper [16], and this two test cases (treatment of waste and treatment of newspaper) are solved with Simulated Annealing algorithm, but the paper [16] does not contain the software representation. The paper describes the representation mode of the generalized VRP model. The paper also describes case studies association of vehicles-nodes-products at level.

2.1. Representation mode and operators of Vehicle Routing Problem algorithms

In this chapter, the representation and evaluation strategy is presented. The representation mode contains the following main components: vector describing the order of nodes (depot, satellite, customer), matrix describing the assignment of vehicle-node-products, vector describing the assignment of the vehicle-recharger station and the vector describing the assignment of vehicle-start-end-node, vector describing level and period and the solution description vector.

The representation is the following: the representation vector of the whole solution is the vector describing the period, where the elements of the vector are the vectors describing the levels. The vectors describing the levels include a vector describing the order of the nodes (depot, satellite, customer), a matrix describing the vehicle-node-products assignment, a vector describing the vehicle-recharger station assignment, and a vehicle-start-end-node assignment.

The components of the general model can be divided into two groups in terms of optimization. One of the groups includes the components, the order and assignment of which give a solution. By changing the order and assignment, we get another (possible) solution. This group includes nodes, vehicles, products, recharger stations, levels, periods. The other group includes components that we cannot associate with each other, we cannot define an order. These system constraints (eg time window) are either parameters describing the functional operation of the system (delivery of products, collection) or can even serve as metrics (distance between nodes, reliability factor, cost of loading and unloading products, etc.)

2.1.1. Representation mode

<u>Vector describing the order of nodes</u>

A permutation is used to describe the nodes, each node is denoted by a number. The serial number of the node will be included in this section. The length of the part is equal to the number of nodes. The vector of the nodes of level i. is the following:

$$\bar{P}_i = [p_1^i, p_2^i, \dots, p_{nol_i}^i]$$

where nol_i denotes the number of nodes in the level i. and

$$p_1^i, p_2^i, \dots, p_{nol_i}^i \in POS_i, \qquad p_1^i, p_2^i, \dots, p_{nol_i}^i \neq RECHARGERSTATION$$

An example of a vector describing the order of nodes is shown in Figure 1.

1	**3**	**2**	**4**	**5**

Figure 1. Example of a vector representation describing the order of nodes (satellite, customer)

In this example, node 1 must be visited, then node 3, then node 2, then node 4, and finally node 5.

<u>The matrix describing the assignment of vehicles-nodes-products</u>

Not a permutation, but an assignment is used here. This means that the individual numbers here indicate the individual vehicles in the line, only here the numbers appear more than once. The length of the assignment is equal to the product of the number of nodes and the number of types of products in the system. This specifies which vehicle will deliver each item at each node. Let V_i the matrix describing the association of vehicles-nodes-products at level i, then

$$V_i = \begin{bmatrix} v^i_{1,1} & \cdots & v^i_{1,n_{producttype}} \\ \vdots & \cdots & \vdots \\ v^i_{nol_i,1} & \cdots & v^i_{nol_i,n_{producttype}} \end{bmatrix}$$

where $v^i_{j_m}$ denotes the service in level i. the node j. product m. and $v^i_{j_m} \in \{1, \dots, cntvehicle_i\}$,$cntvehicle_i = \sum_k cntvehicle^k_i$. An example of the matrix describing the assignment of vehicles-nodes-product is illustrated in Figure 2.

1	**2**	**2**	**1**	**2**	**2**	**2**	**2**	**1**	**1**

Figure 2. An example of a matrix describing the assignment of vehicles-nodes-products

In the example, there are 5 nodes, 2 vehicles and 2 types of products. Then the first part (1,2) will belong to the first node, so the first type of product demand of the first node is served by the vehicle 1, and the second type of product demand of the second node. The first and second types of product demand of the third node are served by vehicle 2, as well as node 4. The first and second types of product demand of node 5 are vehicle 1.

<u>Vector describing the assignment of vehicle-recharger stations</u>

Here it is necessary to determine to which recharger station each vehicle will belong, so a recharger station-vehicle assignment must be made. Its length is equal to the number of vehicles. Let $\bar{R}_i$ be a vector, describing the association of vehicle-recharger stations in level i., then

$$\bar{R}_i = [r^i_1, r^i_2, \dots, r^i_{cntvehicle_i}]$$

where $r^i_1, r^i_2, \dots, r^i_{cntvehicle_i} \in POS_i,\ r^i_1 = RECHARGERSTATION, \dots, r^i_1 = RECHARGERSTATION.$

An example of the vector describing the assignment of vehicle – recharger stations is presented in Figure 3.

5	**2**

Figure 3. An example of a vector describing the assignment of vehicle - recharger stations

The example includes 2 vehicles and 5 recharger stations. The first vehicle will then belong to recharger station 5 and the second vehicle will belong to recharger station 2.

Vector describing the vehicle-start-end-node assignment

The length of the assignment is equal to the number of vehicles at the current level since a start node and end node is assigned to the vehicles. Mark the vector $\overline{UL}_i$, describing the vehicle start-end-node assignment at the level i. then

$$\overline{UL}_i = [ul_1^i, ul_2^i, \dots, ul_{cntvehicle_i}^i]$$

where $ul_1^i, ul_2^i, \dots, ul_{cntvehicle_i}^i \in POS_{i-1}$. An example of the vector describing the vehicle-start-end-node assignment is illustrated in Figure 4.

5	**2**

Figure 4. An example of a vector describing the vehicle-portal node assignment

The example contains 2 vehicles, 5 possible portal nodes. Then the first vehicle will belong to node 5 above the “current” level, while the second vehicle will belong to such node 2.

Vector describing level

The Vehicle Routing Problem has a level structure. Vehicles start from one node in each of the above levels (and deliver/collect the products) at the lower levels and then return to the higher level (if no open route component is used). Let L be the level descriptive vector, then

$$\overline{L_i} = [\bar{P}_i, \bar{V}_i, \bar{R}_i, \overline{UL}_{i-1}]$$

The vector L contains vectors describing the order of the nodes, matrices describing the assignment of the vehicle-nodes-products, vectors describing the assignment of the vehicle-recharger stations, and vectors describing the assignment of the vehicle-start-end-node.

Vector describing period

I designed the representation mode so that the descriptive vectors can handle the period as well. Let $\bar{T}$ be the vector describing the period, then

$$\bar{T} = [\bar{L}_1, \bar{L}_2, \dots, \bar{L}_{n_{level}}]$$

The vector describing the period thus contains matrices describing the levels.
An example of a matrix describing a level is shown in Figure 5.

Level 1 (depot)										
Vector describing the order of nodes	1	3	2	4	5					
Matrix describing the assignment of vehicles-nodes-products	1	2	2	1	2	2	2	2	1	1
Vector describing the assignment of vehicle-recharger stations	5	2								
Vector describing the vehicle-start-end-node assignment	NO									

Level 2 (satellite level 1.)										
Vector describing the order of nodes	1	2	3	4	5					
Matrix describing the assignment of vehicles-nodes-products	2	2	2	1	2	1	1	2	2	1
Vector describing the assignment of vehicle-recharger stations	2	4								
Vector describing the vehicle-start-end-node assignment	3	2								

Level 3 (satellite level 2.)										
Vector describing the order of nodes	5	3	2	4	1					
Matrix describing the assignment of vehicles-nodes-products	2	2	2	1	2	1	2	1	1	2
Vector describing the assignment of vehicle-recharger stations	3	2								
Vector describing the vehicle-start-end-node assignment	4	2								

Level 4 (customers)										
Vector describing the order of nodes	1	3	2	4	5					
Matrix describing the assignment of vehicles-nodes-products	1	2	2	1	2	2	2	2	1	1
Vector describing the assignment of vehicle-recharger stations	5	2								
Vector describing the vehicle-start-end-node assignment	5	2								

Figure 5. Example of a vector describing a level

Level 1 (depot) contains 5 nodes, 2 types of products and 2 vehicles. The system contains 5 recharger stations (globally). The path of the first vehicle is (1, 3, 5, 1) the path of the second vehicle is (1, 3, 2, 4, 1). Recharger stations for each vehicle: recharger station 5 for the first vehicle and recharger station 2 for the second vehicle. At this point, when the vehicle is running out of fuel, it will visit the recharger station in the meantime.

In the example, there are also 5 nodes at *level 2 (satellite level 1)*. At that time, two vehicles also deliver two products. The first vehicle departs from node 3 above and then returns to node 3 after nodes 2, 3, 4, 5. Product 2 is transported to node 2, product 2 is also transported to node 3, product 1 is also transported to node 4 and product 2 is also transported to node 5. Vehicle 2 departs from node 2 one above the level and then returns to node 2 one level above after nodes 1, 2, 3, 4, 5. It also transports product 1 and product 2 to node 1, product 1 to nodes 2 and 3, product 2 to node 4, and product 1 to node 5. The first vehicle has recharger station 2, the second vehicle has recharger station 4.

At *level 3 (satellite level 2)* we also find 5 nodes. At that time, two vehicles also deliver the two types of products. The first vehicle departs from node 4 one above level and then returns to node 4 one above level after visiting node 3, 2, 4, 1. The second vehicle departs from node 2 above and then returns to node 2 after visiting nodes 5, 3, 2, 4, 1. The first vehicle transports product 2 to node 3, product 2 also to nodes 2 and 4, and product 1 to node 1. The second vehicle enters node 5 for product 1 and product 2, product 1 to node 3, node 2 and node 4 and then product 2 to node 1. The first vehicle has a recharger station 3 and the second vehicle has a recharger station 2.

At *level 4 (customer level)*, the demands of 5 customers must be served. Two vehicles deliver the two types of products. The first vehicle first departs from node 5 from a higher level, then visits nodes 1,3,5, and then returns to node 5 in the higher level. The second vehicle departs from node 2 in the higher level, then visits nodes 1, 3, 2, 4, and then returns to node 2 in the higher level. The first vehicle delivers product 1 to node 1, product 2 to node 2, and product 1 and 2 to node 5. The second vehicle delivers product 2 to node 1, product 1 to node 3, and product 1 and 2 to nodes 2 and 4.

Solution description vector

The solution description vector contains the period description vectors. Let $\bar{S}$ be the vector describing the solution, then

$$\bar{S} = [\bar{T}_1, \bar{T}_2, \dots, \bar{T}_{n_{period}}]$$

The vector describing the solution thus contains the vectors describing the period. The solution description vector is illustrated in Figure 6.

Level 1 (depot)										
Vector describing the order of nodes	1	3	2	4	5					
Matrix describing the assignment of vehicles-nodes-products	1	2	2	1	2	2	2	2	1	1
Vector describing the assignment of vehicle-recharger stations	5	2								
Vector describing the vehicle- portal node assignment	NO									

Level 2 (satellite level 1)										
Vector describing the order of nodes	1	2	3	4	5					
Matrix describing the assignment of vehicles-nodes-products	2	2	2	1	2	1	1	2	2	1
Vector describing the assignment of vehicle-recharger stations	2	4								
Vector describing the vehicle-portal node assignment	3	2								

Level 3 (satellite level 2)										
Vector describing the order of nodes	5	3	2	4	1					
Matrix describing the assignment of vehicles-nodes-products	2	2	2	1	2	1	2	1	1	2
Vector describing the assignment of vehicle-recharger stations	3	2								
Vector describing the vehicle-portal node assignment	4	2								

Level 4 (customers)										
Vector describing the order of nodes	1	3	2	4	5					
Matrix describing the assignment of vehicles-nodes-products	1	2	2	1	2	2	2	2	1	1
Vector describing the assignment of vehicle-recharger stations	5	2								
Vector describing the vehicle-portal node assignment	5	2								

The representation mode for day 1 of the period

Level 1 (depot)										
Vector describing the order of nodes	5	3	2	4	1					
Matrix describing the assignment of vehicles-nodes-products	2	2	1	1	2	2	1	2	1	1
Vector describing the assignment of vehicle-recharger stations	5	2								
Vector describing the vehicle-portal nodeassignment	NO									

Level 2 (satellite level 1)										
Vector describing the order of nodes	1	2	5	3	4					
Matrix describing the assignment of vehicles-nodes-products	2	1	2	1	2	1	1	2	1	1
Vector describing the assignment of vehicle-recharger stations	5	4								
Vector describing the vehicle-portal node assignment	1	2								

Level 3 (satellite level 2)										
Vector describing the order of nodes	5	3	2	4	1					
Matrix describing the assignment of vehicles-nodes-products	1	1	2	1	2	1	2	2	1	2
Vector describing the assignment of vehicle-recharger stations	3	5								
Vector describing the vehicle-portal node assignment	4	1								

Level 4 (customers)										
Vector describing the order of nodes	1	3	2	4	5					
Matrix describing the assignment of vehicles-nodes-products	1	2	2	2	2	2	1	2	1	2
Vector describing the assignment of vehicle-recharger stations	1	2								
Vector describing the vehicle-portal node assignment	1	2								

The representation mode for day 2 of the period

Level 1 (depot)										
Vector describing the order of nodes	1	2	3	4	5					
Matrix describing the assignment of vehicles-nodes-products	2	2	2	1	1	2	2	2	1	1
Vector describing the assignment of vehicle-recharger stations	3	2								
Vector describing the vehicle-portal node assignment	NO									

Level 2 (satellite level 1)										
Vector describing the order of nodes	1	2	3	4	5					
Matrix describing the assignment of vehicles-nodes-products	1	2	1	1	2	1	2	2	2	1
Vector describing the assignment of vehicle-recharger stations	1	4								
Vector describing the vehicle-portal node assignment	1	2								

Level 3 (satellite level 2)										
Vector describing the order of nodes	1	3	2	4	5					
Matrix describing the assignment of vehicles-nodes-products	1	1	2	1	2	1	2	1	2	2
Vector describing the assignment of vehicle-recharger stations	1	1								
Vector describing the vehicle-portal node assignment	4	4								

Level 4 (customers)										
Vector describing the order of nodes	1	3	2	4	5					
Matrix describing the assignment of vehicles-nodes-products	1	2	2	1	2	2	2	2	1	1
Vector describing the assignment of vehicle-recharger stations	5	5								
Vector describing the vehicle-portal node assignment	2	2								

The representation mode for day 3 of the period

Figure 6. Example of a solution description vector

2.1.2. Applied operators

The applied operators can be divided into two groups. The first group consists of the vector describing the order of nodes representation part. In this case a permutation representation mode is used, where the sequence numbers of the nodes are included in the representation mode. The following neighbourhood and crossover operators are used in the vector describing the order of nodes part: 2-opt [17], 3-opt [18], Order Crossover [19], Partially Matched Crossover [20], Cycle Crossover [21].

The second group consists of the following representation parts: the matrix describing the assignment of vehicles-nodes-products, vector describing the assignment of vehicle-recharger stations, vector describing the vehicle-start-end-node assignment. In this case a "regeneration" is used, where the neighbour of a current solution will be completely new solution.

3. Applied optimization algorithms

In this section, the applied optimization algorithms are presented: the Ant Colony System (ACS) [22], Genetic Algorithm (GA) [23], Simulated Annealing (SA) [24] and Tabu Search (TS) [25].

The Ant Colony System [22] algorithm is a population algorithm. It was inspired by ants, which deposit hormones during their journey. Ants are more likely to go to areas that contain more hormones (pheromones). The first step of the algorithm is the initialization of the population, which means the creation of random solutions (ants). Next, the calculation of the pheromone level at each edge is performed. Then the evaporation of the pheromone is taken into account, and then the determination of the path of the ants. Updating the pheromone levels and calculating the path of the ants is an iterative process, which continues until the termination condition.

The Genetic Algorithm [23] is a population algorithm, that maintains a series of solutions through each iteration. The first step is the initialization and then evaluation of the initial population. Then creating new solutions using crossover and mutation are performed. A crossover is a major change in two individuals. A mutation step is a small change in a single individual. The old solutions are replaced by new solutions. The creation of new solutions is an iterative process, which continues until the termination condition.

Simulated Annealing [24] is based on only one solution. The first step is the creation of the current solution and setting the temperature. The next step is the generation of a neighbor solution. If the neighboring solution is better than the current solution, it is accepted. If it is not better, it will only be accepted with a certain probability. This probability decreases with decreasing temperature. And the temperature decreases by a certain amount (alpha) during each iteration.

Tabu Search [25] maintains a tabu list, which contains already discovered solutions. The first step is to initialize the tabu list, which is initially empty. The current solution is initialized, which is initially a randomly chosen solution. Then a neighbor of the current solution must be generated. If it is better than the current solution, then this will be the current solution. The worst solution is added to the tabu list (if the tabu list does not already contain the element). The tabu list can be full, in which case the oldest entered element is deleted. The creation of neighboring solutions, the evaluation, and the updating of the tabu list are repeated until the termination condition.

4. Case study and test result of the generalized Vehicle Routing software

In this section a case study and test result are presented. The patient transport is chosen as case study. During patient transport, we considered the case when people are transported from the hospital. In this case, it is possible to transport several patients at the same time and there is no time window (so it does not matter how fast we transport the patient from the hospital). Here, too, there are only two levels, the first level is the hospital and the second level is the home of the patients. The number of vehicles is low, as is the number of patients. We deliver a single patient to each node. The other parameters are general, the parameters that appear during most transports (such as minimizing the distance traveled, minimizing travel time, etc.)

Table 1. The test results in case of patient transport

Parameter	Value
Base parameters	
Number of levels	2
Number of nodes belonging to the first level	1
Location of first level nodes	[0,100]
Number of second-level nodes	10, 20
Location of second level nodes	[200,300]
Number of periods	1
Number of product types (product means here a person)	1
Number of vehicles	5
Node parameters	
Route Status Between Nodes	static, [100,500]
Travel Distance Between Nodes	static, coordinate based
Travel Time Between Nodes	static, [10,100]
Product parameters	
Product Demand of The Node (it means the number of persons to be transported per house here)	static, [1]
Vehicle parameters	
Capacity Constraint of The Vehicle (for person)	static, [5]
Fuel Consumption of The Vehicle	static, [10, 100]
Metrics	
Length of the route	
Fuel consumption	

Route status Route time Unvisited customers (undelivered person)

Table 2. The test results in case of patient transport

Instance + Algorithm	Average fitness	Average running time (sec)
Number of nodes: 1st level: 1, 2nd level: 10		
I-1-10 + ACS	4140.48	8.19
I-1-10 + GA	3731.43	7.41
I-1-10 + SA	3700.97	7.19
I-1-10 + TS	3050.45	7.88
Number of nodes: 1st level: 1, 2nd level: 20		
I-1-20 + ACS	4220.07	46.90
I-1-20 + GA	4074.67	41.34
I-1-20 + SA	3955.29	43.90
I-1-20 + TS	4555.99	46.01

Both of the data sets contain 2 levels, with the first level containing 1 and the second level containing 10 and 20 nodes. For this type of transport, the running time was also low due to the ease of the data set. For the first data set (I-1-10), TS proved to be the best, while for the second data set (I-1-20), GA. It can be seen that the second data set is much more complex than the first, and the runtime is much higher, but even this runtime can be said to be low.

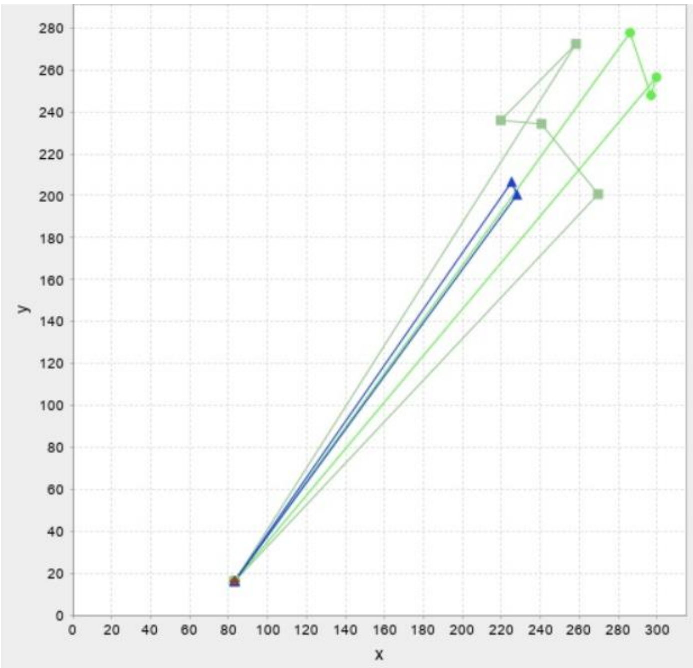

Figure 7. The running result of the patient transportation (I-1-10 + GA)

Figure 7. shows a result of data series I-1-10. The figure shows that the vehicles start from the point with the coordinate (80,15). A total of three carriers were used (blue, gray, green routes). The vehicles visit some of the coordinates in the [200,300] interval and then return to the coordinate (80,15).

5. Conclusions and future work

In this article, the software for the generalization of the Vehicle Routing Problem was presented. First, it was presented how many types of Vehicle Routing Problems inspired the general model. Then the general model and its software representation were presented. The representation method consists of the following elements: vector describing the order of nodes, the matrix describing the assignment of vehicles-nodes-products, vector describing the assignment of vehicle-recharger stations, vector describing the vehicle-start-end-node assignment, vector describing level, vector describing period and the solution description vector. Then the applied heuristic algorithms were presented: the Ant Colony System (ACS), Genetic Algorithm (GA), Simulated Annealing (SA) and Tabu Search (TS). Then a case study, the patient transport, was presented. Based on the running results, the developed software model can be well applied to solve the generalized Vehicle Routing Problem. Another area of research is the integration of the designed software system with new heuristic methods and the expansion of the generalized model with new Vehicle Routing Problem trends.

References

[1] Konstantakopoulos, G. D., Gayialis, S. P., Kechagias, E. P.: Vehicle routing problem and related algorithms for logistics distribution: A literature review and classification. *Operational research*, 22 (3), 2022, pp. 2033–2062. https://doi.org/10.1007/s12351-020-00600-7

[2] Li, J., Xu, M., Sun, P.: Two-echelon capacitated vehicle routing problem with grouping constraints and simultaneous pickup and delivery. *Transportation Research Part B: Methodological*, 162, 2022, pp. 261–291. https://doi.org/10.1016/j.trb.2022.06.003

[3] Fahmy, S. A., Gaafar, M. L.: Modelling and solving the split-delivery vehicle routing problem, considering loading constraints and spoilage of commodities. *International Journal of Systems Science: Operations & Logistics*, 2022, pp. 1–19. https://doi.org/10.1080/23302674.2022.2074566

[4] Li, J., Xu, M., Sun, P.: Two-echelon capacitated vehicle routing problem with grouping constraints and simultaneous pickup and delivery. *Transportation Research Part B: Methodological*, 162, 2022, pp. 261–291. https://doi.org/10.1016/j.trb.2022.06.003

[5] Xu, L., Wang, Z., Chen, X., Lin, Z.: Multi-Parking Lot and Shelter Heterogeneous Vehicle Routing Problem With Split Pickup Under Emergencies. *IEEE Access*, 10, pp. 36073–36090, https://doi.org/10.1109/ACCESS.2022.3163715.

[6] Gu, Z., Zhu, Y., Wang, Y., Du, X., Guizani, M., Tian, Z.: Applying artificial bee colony algorithm to the multidepot vehicle routing problem. *Software: Practice and Experience*, 52 (3), 2022, pp. 756–771, https://doi.org/10.1002/spe.2838.

[7] Radzki, G., Bocewicz, G., Wikarek, J., Nielsen, P., Banaszak, Z.: Multi Depot UAVs Routing Subject to Changing Weather and Time Windows Variation. *In Conference on Automation*, 2022, pp. 64–74, https://doi.org/10.1007/978-3-031-03502-9_7.

[8] Sluijk, N., Florio, A. M., Kinable, J., Dellaert, N., Van Woensel, T.: Two-echelon vehicle routing problems: A literature review. *European Journal of Operational Research*, 2022, https://doi.org/10.1016/j.ejor.2022.02.022.

[9] Jie, K. W., Liu, S. Y., Sun, X. J.: A hybrid algorithm for time-dependent vehicle routing problem with soft time windows and stochastic factors. *Engineering Applications of Artificial Intelligence*, 109, 2022, p. 104606. https://doi.org/10.1016/j.engappai.2021.104606

[10] Bahalke, U., Hamta, N., Shojaeifard, A. R., Alimoradi, M., Rabiee, S.: A New Heuristic Algorithm for Multi Vehicle Routing Problem with AND/OR-Type Precedence Constraints and Hard Time Windows. *Operational Research in Engineering Sciences: Theory and Applications*, 2022. https://doi.org/10.31181/oresta300622015b

[11] Ruiz-Meza, J., Montes, I., Pérez, A., Ramos-Márquez, M.: *VRP model with time window, multiproduct and multidepot*. 2020. http://dx.doi.org/10.6180/jase.202006_23(2).0008

[12] Torres, F., Gendreau, M., Rei, W.: Crowdshipping: An open VRP variant with stochastic destinations. *Transportation Research Part C: Emerging Technologies*, 140, 2022, p. 103677, https://doi.org/10.1016/j.trc.2022.103677.

[13] Ghorbani, E., Alinaghian, M., Gharehpetian, G. B., Mohammadi, S., Perboli, G.: A survey on environmentally friendly vehicle routing problem and a proposal of its classification. *Sustainability*, 12 (21), 2020, p. 9079. https://doi.org/10.3390/su12219079

[14] Torres, F., Gendreau, M., Rei, W.: Crowdshipping: An open VRP variant with stochastic destinations. *Transportation Research Part C: Emerging Technologies*, 140, 2020, p. 103677, https://doi.org/10.1016/j.trc.2022.103677.

[15] Yalcin, G. D., Erginel, N.: An Adapted Fuzzy Multi-Objective Programming Algorithm for Vehicle Routing. *Universal Journal of Operations and Management*, 2022, pp. 56–74, https://doi.org/10.37256/ujom.1120221144.

[16] Agárdi, A., Kovács, L., Bányai, T.: Mathematical Model for the Generalized VRP Model. *Sustainability*, 14, 2022, p. 11639. https://doi.org/10.3390/su141811639.

[17] Costa, P. R d O., Rhuggenaath, J., Zhang, Y., Akcay, A.: Learning 2-opt heuristics for the traveling salesman problem via deep reinforcement learning. In *Asian Conference on Machine Learning*, 2020, pp. 465–480.

[18] Chou, X., Gambardella, L. M., Luangpaiboon, P., Aungkulanon, P., Montemanni, R.: 3-opt Metaheuristics for the Probabilistic Orienteering Problem. In 2021 *The 8th*

International Conference on Industrial Engineering and Applications (Europe), 2021, pp. 210–215, https://doi.org/10.1145/3463858.3463867.

[19] Arram, A., Ayob, M.: A novel multi-parent order crossover in genetic algorithm for combinatorial optimization problems. *Computers & Industrial Engineering*, 133, 2019, pp. 267–274, https://doi.org/10.1016/j.cie.2019.05.012.

[20] Hossain, M. S., Choudhury, S. S., Hayat, S. A. I., Tanim, A. S., Kabir, M. N., Islam, M. M.: An Efficient Solution to Travelling Salesman Problem using Genetic Algorithm with Modified Crossover Operator. *EMITTER International Journal of Engineering Technology*, 7 (2), 2019, pp. 480–493. https://doi.org/10.24003/emitter.v7i2.380

[21] Visutarrom, T., Chiang, T. C.: An evolutionary algorithm with heuristic longest cycle crossover for solving the capacitated vehicle routing problem. In *2019 IEEE Congress on Evolutionary Computation (CEC)*, 2019, pp. 673–680. https://doi.org/10.1109/CEC.2019.8789946

[22] Mutar, M., Burhanuddin, M., Hameed, A., Yusof, N., Mutashar, H.: An efficient improvement of ant colony system algorithm for handling capacity vehicle routing problem. *International Journal of Industrial Engineering Computations*, 11 (4), 2020, pp. 549–564, http://dx.doi.org/10.5267/j.ijiec.2020.4.006.

[23] Abbasi, M., Rafiee, M., Khosravi, M. R., Jolfaei, A., Menon, V. G., Koushyar, J. M.: An efficient parallel genetic algorithm solution for vehicle routing problem in cloud implementation of the intelligent transportation systems. *Journal of cloud Computing*, 9 (1), 2020, pp. 1–14, https://doi.org/10.1186/s13677-020-0157-4.

[24] İlhan, İ. L. H. A. N.: An improved simulated annealing algorithm with crossover operator for capacitated vehicle routing problem. *Swarm and Evolutionary Computation*, 64, 2021, p. 100911, https://doi.org/10.1016/j.swevo.2021.100911.

[25] Gmira, M., Gendreau, M., Lodi, A., Potvin, J. Y.: Tabu search for the time-dependent vehicle routing problem with time windows on a road network. *European Journal of Operational Research*, 288 (1), 2021, pp. 129–140. https://doi.org/10.1016/j.ejor.2020.05.041