Priority Based Vehicle Routing for Agile Blood Transportation between Donor/Client Sites

Mehmet Karakoc[1], Melih Gunay[2]
[1]Department of Software Engineering, Antalya AKEV University, Antalya, Turkey
pamukkaleli@gmail.com
[2]Department of Computer Engineering, Akdeniz University, Antalya, Turkey
mgunay@akdeniz.edu.tr

***Abstract*— In this paper, we study *Vehicle Routing Problem* (VRP) for *Blood Transporters* (BTs) and propose an efficient vehicle routing scheme for blood transportation between hospitals or *Donor/Client Sites* (DCSs) within a region that is based on *Artificial Intelligence*. It is assumed that each BT in a fleet of vehicles starts and completes its route at a blood-bank while visiting a subset of DCSs using the shortest path. However, unlike traditional logistic planning, blood transportation may be time critical. Therefore, in our approach, the vehicle routing is formulated to take into account the urgency of the requests and responses. Consequently, the objective of this study is to minimize the number of BTs while maintaining their minimum traveling lengths considering priority. In this regards, we extended the classical *Capacitated VRP* (CVRP) and reformulated requests to take into account the priority by assigning weight to each request. A hybrid meta-heuristic algorithm including *Genetic Algorithms* and *Local Search* is used to simulate transporting blood requests of DCSs. We challenged our approach with symmetrical CVRP instances taken from literature. In this case study, we observed that both the cost and response time are reduced dramatically for emergency.**

Keywords — priority; vehicle routing; blood transportation; donor/client sites; genetic algorithms; local search.

I. INTRODUCTION

In medicine, blood is used during births, bypasses, general surgeries, organ or even blood transplants. Despite the current advances in medicine and technology, there is no alternative to providing blood except from humans. As blood is precious asset and has a limited shelf time, excessive storage on site is not desired. Therefore, transportation of blood and its components between donors and clients is a constant and diligent operation. In order to better manage and improve logistics, blood-banks are often established that serve a number of *Donor/Client Sites* (DCSs) using specialized vehicles. Planning of collection and distribution of blood units through blood-banks between DCSs is done by utilizing the approaches in solving the classical *Shortest Path Problem* (SPP), *Traveling Salesman Problem* (TSP) or *Vehicle Routing Problem* (VRP - Dantzig and Ramser, 1959) of *Computer Science*.

In SPP, the shortest path between the start s and the target t locations is found so that this path includes a number of sites to be followed while aiming at minimizing the number(s) of turns and/or sites to be visited (Figure 1(a)). In TSP and VRP, unlike the SPP, the start and the target are the same. In TSP, single traveler seller starts and completes his/her route at the same location after visiting a number of sites (cities) in which each one is visited only once using the shortest path (Figure 1(b)). In VRP, on the other hand, a fleet of vehicles (travelers) simultaneously start at a base station such as a depot or a loading point and return back to this originating location after visiting a number of sites (customers/places to be served) using an optimal set of routes with the minimum cost (Figure 1(c)). In VRP, if each vehicle is assigned a certain load capacity, then this problem is known as *Capacitated VRP* (CVRP - Toth and Vigo, 2002; Ralphs et al., 2003) where several approaches have been proposed to solve this problem. Also, in VRP, sites are assigned to vehicles (scheduling) and the routing problem is solved as in TSP (special case of the VRP) for each vehicle concurrently. However, during the planning, assignments of sites may be exchanged between routes for better solutions. Figure 1 demonstrates the cases for SPP, TSP and VRP.

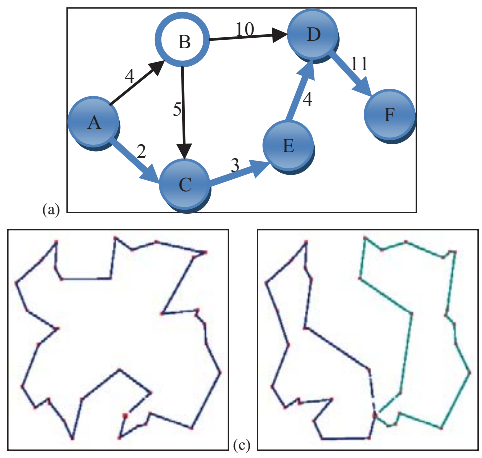

Figure 1. Illustration for the SPP (a), TSP (b), and VRP (c)

978-1-5386-0930-9/17/$31.00 ©2017 IEEE

In the simplest form, when a path is planned from single start to single end, then the solution is relatively straightforward (the shortest path is "A-C-E-D-F" between vertices *A* and *F* in Figure 1(a)). However, in VRP, when multiple routes are to be planned through several intermediates, then the search space increases exponentially and hence this problem becomes NP-hard. In *Computer Science*, the solution to NP-hard problems requires the use of *Artificial Intelligence* methods with heuristics. Among these heuristics, *Genetic Algorithms* (GA), *Ant Colony Optimization* and/or *A* Algorithm* are often used to find the optimal set of routes simultaneously. Even though numerous approaches have been proposed for the solution, some are more efficient than others depending on the case. In the following paragraph, some of these approaches are explained.

The SPP was studied for the path planning of mobile robots in (Castillo and Trujillo, 2005; Achour and Chaalal, 2011; Parvez and Dhar, 2013). Parvez and Dhar (2013) applied GA for finding the optimal path effectively while reducing the path cost in a static environment with the established obstacles. Castillo and Trujillo (2005) and Achour and Chaalal (2011) also used GA for solving the same problem. In (Umitsu and Fushimi, 2006; Amadini et al., 2013), the emergency response for post-disaster was studied. While in (Umitsu and Fushimi, 2006) the alternative paths are suggested for blocked roads due to collapsed buildings, in (Amadini et al., 2013), the objective is to transport as many victims as possible by using decision support algorithms that utilize mixed integer programming techniques with constraint satisfaction. As a variant of scheduling problem, Meinzer and Storandt (2014) studied ambulance allocation. Since a number of ambulances serve a certain city, emergency calls should be responded as soon as possible with the limited number of vehicles. In their study, the optimization of ambulance distribution throughout a city and the reassignment of ambulances for the requests are considered. The management of emergency medical services is examined and new strategies are proposed for the allocation. For ambulance routing problem, Javidaneh et al. (2010) adapted a VRP based solution enhanced with *Ant Colony Optimization* for the accommodation of injured people at hospitals as soon as possible. While routing, the traveling length of the routes and the capacity of the hospitals are considered. Karakoc et al. (2015) adapted classical VRP approach with GA to blood transportation between medical facilities. However, in their approach they did not consider the urgency requirements of the blood transportation.

As explained above, the SPP, TSP and VRP include a large number of variations in which some of them may be transformed to each other by applying special constraints. In this regards, we study the classical CVRP that is well-studied in *Supply-Chain Management* and extend it based on priority by assigning weight to each request for the sites to be visited by *Blood Transporters* (BTs). To solve this multi-objective and complex problem, priority based CVRP, we developed a hybrid meta-heuristic *Memetic Algorithm* (Moscato and Cotta, 2003 - GALS) that improves GA with *Local Search* (LS). Consequently, we propose a novel blood transportation framework that minimizes the number of routes and the total traveling length while satisfying urgency and storage requirements.

II. SYSTEM MODEL

In this section, we describe the priority based CVRP and explain our approach for the solution.

In this study, we consider three main components as depicted in Figure 2: (1) the blood-bank, (2) *Donor/Client Sites* (DCSs), and (3) *Blood Transporters* (BTs) to use the routes. The blood-bank is both the source and the destination for each route where all requests are met through this point, and all the DCSs on a route are the targets for the BT to use this route. In our model, we decide on the number of routes, and the order of intermediates to be visited (*assignments to the BT*) and the blood packages to be transported per each BT while minimizing the total traveling length.

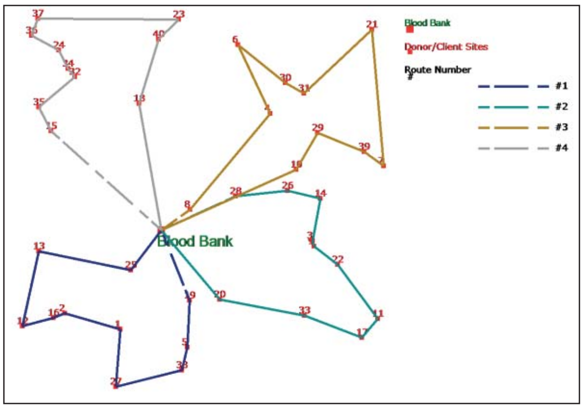

Figure 2. *Vehicle routing through the blood-bank with four BTs/routes for 40 DCSs*

The inputs, hard constraints and output for the model are given in Table 1.

Inputs
(1) The blood-bank and DCSs within a region, and their positions. (2) The priority weighted amount of blood request for each DCS. (3) The vehicle capacity allowed for each identical BT.
Hard constraints
(1) Each DCS should be visited only once by exactly one BT. (2) Each BT should start and complete its route at the blood-bank. (3) Total weighted load on any route cannot be more than the vehicle capacity.
Output
(1) A set of routes

TABLE I. THE INPUTS, HARD CONSTRAINTS AND OUTPUT FOR THE PRIORITY BASED CVRP

To calculate the priority weighted amount (pw_i) for a site *i* with given load (l_i), the Equation (1) is used as:

$$pw_i = l_i \times p_i \tag{1}$$

where p_i is the priority of the load. In this schema, it is possible to avoid visiting of certain sites by setting the priority to zero for that load.

Since the complexity (time) of the VRP increases exponentially with the increase in the number of sites, the problem requires heuristics for the solution. GA is a global search heuristic inspired from Darwin's theory about natural evolution. It can be used for solving complex and large combinatorial optimization problems that cannot be solved with conventional methods. GA is a robust optimization technique which searches a set of candidate solutions in a population (chromosome group) simultaneously. The optimization based on GA has numerous successful applications to various real-world problems including scheduling and routing (TSP and VRP). It contains basic adaptive techniques to simulate the evolution process based on natural selection and genetics in nature. Therefore, over generations, GA inherently explores regions of best performance that in turn likely to lead optimal offspring (solutions). This process continues until some termination criteria is met. Finally, from the last generation, best solution is chosen.

III. GENETIC ALGORITHMS WITH LOCAL SEARCH (GALS) APPROACH TO THE PROBLEM

LS is an effective approach which improves the solution and convergence in GA by exploring its neighborhoods iteratively. In this section, we explain how we adapted GA to the priority based CVRP and hybridized it with LS to increase the quality of solution and performance.

A. Chromosome Representation

Each chromosome includes a set of routes given by a chain of integers. In this schema, the blood-bank is identified as "0" and the positive integers correspond to the DCSs within the region. Example 1 demonstrates the encoding of a chromosome with three routes including eight DCSs in the sequence assigned.

Example 1:

0 - 4 - 5 - 2 - 0 - 8 - 6 - 3 - 1 - 0 - 7

Route#1: blood-bank 4 5 2

Route#2: blood-bank 8 6 3 1

Route#3: blood-bank 7

In Table 2, the loads with priorities and weighted loads are given per each route for the encoding in Example 1.

TABLE II. THE LOADS, PRIORITIES AND THE WEIGHTED LOADS PER EACH ROUTE FOR THE ENCODING IN EXAMPLE 1

Total load is 360 for three BTs											
Route	**#1**				**#2**					**#3**	
site *i*	**0**	**4**	**5**	**2**	**0**	**8**	**6**	**3**	**1**	**0**	**7**
l_i	**120**	40	40	40	**120**	30	30	30	30	**120**	120
p_i	-	0.75	1.5	1.5	-	0.5	1.0	1.0	1.5	-	0.75

Total load is 360 for three BTs											
Route	**#1**				**#2**					**#3**	
pw_i	**150**	30	60	60	**120**	15	30	30	45	**90**	90

In Table 2, when the vehicle capacity is 120, the encoding with the weighted loads is not feasible. Therefore, the new and feasible solution may be as demonstrated in Example 2.

Example 2:

0 - 5 - 2 - 0 - 8 - 6 - 3 - 1 - 0 - 7 - 4

Route#1: blood-bank 5 2

Route#2: blood-bank 8 6 3 1

Route#3: blood-bank 7 4

B. Initial Population Creation

Good starting point improves the solution quality significantly. For each site, we either choose a random site (10 percent of the time) or a close site (among up to the five closest randomly) to the previous one. For demonstration purposes, assume that the sites "3, 2, 5" are the closest ones to the blood-bank (0). Then the initial population may include "0-5".

C. Fitness Function

Our multi-objective cost function is given in Equation (2) as:

$$\mu \times R + \lambda \times L \tag{2}$$

where μ and λ are the coefficients for the number of routes R and the total traveling length L, respectively, in which each one is used for one objective. L is calculated using Equation (3) as:

$$L = \sum_{r=1}^{R} RouteLength(r) \tag{3}$$

where *RouteLength* is the sum of the *Euclidean distances* between all consecutive site pairs on path.

D. Genetic Algorithms Implementation

Creation of a new population involves four steps:

1) Selection is done for choosing a chromosome pair in applying genetic operators.

2) Crossing-over is applied between the selected parents to be mated for producing offspring.

3) Mutation is applied for exploring search space further by modifying some genes of a chromosome. A chromosome is broken from two points and recreated so that the range is to be in reverse order using 2-opt LS algorithm until local minimum is achieved.

4) Reproduction is done for the construction of the next generation by using both the parents and offspring of the current generation. To preserve the best chromosomes, a number of elite ones (e) in the current generation are directly copied to the next. This is called as elitism. Then, for maintaining the population size (n), "n - e" offspring are produced. Among the offspring the worst ones are eliminated.

The algorithm to implement the GALS for solving priority based CVRP and the improved LS algorithm applied are given in Table 3 and Table 4, respectively.

TABLE III. PSEUDO-CODE OF THE GALS

Algorithm: solvePriorityBasedCVRPWithGALS()

```
Create initial population
Calculate fitness by using (2)
begin
 Elitism
 {parent1, parent2}          = Selection
 {offspring1, offspring2}    = Crossing-over(parent1, parent2)
 applyImprovedLS(offspring1) // mutation on the first child
 applyImprovedLS(offspring2) // mutation on the second child
 Calculate fitness by using (2)
repeat "until max_num_of_generations (m) is reached"
return best_solution in the current_generation
```

The time complexity of the GALS is $O(m \times n^2 \times \log n)$ where *m* is the number of iterations.

TABLE IV. PSEUDO-CODE OF THE IMPROVED LS ALGORITHM APPLIED

Algorithm: applyImprovedLS(Chromosome individual)

```
while "local minimum is not achieved for individual" do
 determine the best "i, i+1" and "j, j+1" edge pair for the individual
 if dMat(i, i+1) + dMat(j, j+1) > dMat(i, j) + dMat(i+1, j+1) then
  exchange the edges based on 2-opt algorithm and update fitness
 end if
end while
```

where *i* and *j* are the numbers corresponding to the genes in the chromosome and the *dMat* is the matrix that keeps the *Euclidean distances* between all site pairs. As shown in Table 4, LS is only applied if there is going to be improvement in the offspring. After the application of LS, the existence of consecutive zeros allows us to drop a route hence BT as demonstrated in Example 3.

Example 3:

Old solution : 0 1 3 5 0 4 2 0 6 → (3 routes)

New solution : 0 1 3 5 0 0 2 4 6 → 0 1 3 5 0 2 4 6 (2 routes)

IV. PERFORMANCE EVALUATION

In this section, we evaluate the performance of the GALS by comparing the total traveling lengths for varying numbers of DCSs and BTs. For data set, we used a number of symmetrical CVRP instances taken from literature (Networking and Emerging Optimization, 2013).

A. Performance Variables and Metrics

In our experiments, we ran each test 100 times and took the mean and standard deviation of (a) the total traveling length, (b) the number of generations, and (c) the corresponding execution time.

The following lists the metrics of interest:

Load: is the priority weighted amount of blood request for each DCS. For instance, a very urgent blood request may weight more than several low priority blood requests. Therefore, such a request may take precedence over other requests.

Capacity: is the maximum load for each BT.

Tightness: is the vehicle utilization in the solution and calculated using Equation (4) as:

$$Q \div (m \times q) \qquad (4)$$

where Q is the total amount of all demands (total load on all vehicles), m is the number of vehicles/routes and q is the vehicle capacity while "$m \times q$" corresponds to the total capacity of all vehicles.

GA parameters and the values used in our experiments are given in Table 5.

TABLE V. GA PARAMETERS AND THE VALUES USED İN THE EXPERİMENTS

Parameter	Value
Population size	100
Maximum number of generations	1000
Selection method	Tournament selection
Crossing-over and mutation operators	Permutation – 2-opt algorithm
Probabilities for crossing-over and mutation	80% – 5%
μ and λ in (2)	0.7 – 0.5 (Chand and Mohanty, 2013)
e	2

To implement the GALS and develop the related GUI, we preferred C# on *Microsoft Visual Studio 2012*, and for the experiments conducted, we used a Windows 7 Ultimate OS installed desktop PC with Intel(R) Core(TM) i5-3470 CPU at 3.20 GHz processor and 8GB RAM.

B. Simulation Results

The results obtained with CVRP instances are given in Table 6.

TABLE VI. RESULTS OBTAINED FOR THE EXPERIMENT WITH CVRP INSTANCES

Instance	Best-Known Value (*)	n (**)	m	q	Maximum Length (***)	Tightness
att-n48-k4	40002	47	4	15	40002	0,73
A-n34-k5	778	33	5	100	778	0,92
A-n80-k10	1763	79	10	100	1763	0,94

Instance	Best-Known Value (*)	n (**)	m	q	Maximum Length (***)	Tightness
B-n39-k5	549	38	5	100	549	0,88
E-n22-k4	375	21	4	6000	375	0,94
E-n23-k3	569	22	3	4500	569	0,75
E-n30-k3	534	29	3	4500	534	0,94
E-n51-k5	521	50	5	160	521	0,97
E-n101-k8	815	100	8	200	815	0,91
F-n45-k4	724	44	4	2010	724	0,90
F-n72-k4	237	71	4	30000	237	0,96

*(Computational Infrastructure for Operations Research, 2003) **(# of customers) ***(obtained with the GALS)

As shown in Table 6, the minimum number of routes and the minimum total traveling length were obtained with complete convergence in all runs for each instance. The results are in consistent with the ones in (Stanojević et al., 2013; Uchoa et al., 2014).

V. CONCLUSIONS

Efficient and urgent blood distribution is needed for daily operations. However, in cases of disasters such as earthquakes, hurricanes or others where mass injuries are reported, it becomes even more critical to better manage resources for effective emergency response. In this study, we proposed an efficient vehicle routing scheme that is based on *Artificial Intelligence* for a region. As an application, each BT in a fleet of vehicles starts and completes its route at a blood-bank while visiting a subset of hospitals/DCSs using the shortest path. The minimum cost is achieved within a reasonable time complexity using the hybrid meta-heuristic memetic algorithm *GALS* that we developed. In our case study with symmetrical CVRP instances, we observed that minimizing the total traveling length of vehicles decreases the response time for emergency with our rather effective approach.

Even though our approach assumes static blood-bank and DCSs, it may be extended to include dynamic blood-bank and DCSs with heterogeneous vehicles at different capacities. However, modeling and formulation of such cases and conducting additional experiments with different priorities are left for future work.

REFERENCES

[1] Achour, N., and M. Chaalal. 2011. Mobile Robots Path Planning using Genetic Algorithms. ICAS 2011 : The Seventh International Conference on Autonomic and Autonomous Systems.

[2] Amadini, R., I. Sefrioui, J. Mauro, and M. Gabbrielli. 2013. A Constraint-Based Model for Fast Post-Disaster Emergency Vehicle Routing. International Journal of Artificial Intelligence and Interactive Multimedia 2(4): 67-75.

[3] Castillo, O., and L. Trujillo. 2005. Multiple Objective Optimization Genetic Algorithms for Path Planning in Autonomous Mobile Robots. International Journal of Computers, Systems and Signals 6(1).

[4] Chand, P., and J.R. Mohanty. 2013. Solving Vehicle Routing Problem with Proposed Non-Dominated Sorting Genetic Algorithm and Comparison with Classical Evolutionary Algorithms. International Journal of Computer Applications (IJCA) 69(26): 34- 41.

[5] Computational Infrastructure for Operations Research. 2003. Vehicle Routing Data Sets [accessed on November 22, 2016]. Available at: http://www.coin-or.org/SYMPHONY/branchandcut/VRP/data/index.htm.old

[6] Dantzig, G.B., and J.H. Ramser. 1959. The Truck Dispatching Problem. Management Science 6(1): 80-91.

[7] Javidaneh, A., M. Ataee, and A.A. Alesheikh. 2010. Ambulance Routing with Ant Colony Optimization. Geospatial Information Systems Congress, Volume 6, National Cartographic Center, Tehran University.

[8] Karakoc, M., F. Al-Turjman, and M. Gunay. 2015. Routing Approach for Urgent Blood Transportation between Medical Facilities. 6th Hospital and Health Services Management Congress, 16-19 December, Spice Hotel & Spa, Antalya/Belek, Turkey.

[9] Meinzer, N., and S. Storandt. 2014. Decision Support in Emergency Medical Systems: New Strategies for Dynamic Ambulance Allocation. The First AAAI Workshop on World Wide Web and Public Health Intelligence (W3PHI-2014).

[10] Moscato, P., and C. Cotta. 2003. A Gentle Introduction to Memetic Algorithms. In Handbook of Metaheuristics, edited by Glover, F., and G.A. Kochenberger, pp. 105-144. Springer US, 57, Boston MA, 560p.

[11] Networking and Emerging Optimization. 2013. Capacitated VRP Instances | Vehicle Routing Problem [accessed on November 22, 2016]. Available at: http://neo.lcc.uma.es/vrp/vrp-instances/capacitated-vrp-instances/

[12] Parvez, W., and S. Dhar. 2013. Path Planning of Robot in Static Environment using Genetic Algorithm (GA) Technique. International Journal of Advances in Engineering & Technology, July.

[13] Ralphs, T.K., L. Kopman, W.R. Pulleyblank, and L.E. Trotter. 2003. On the Capacitated Vehicle Routing Problem. Mathematical Programming 94(2): 343-359.

[14] Stanojević, M., B. Stanojević, and M. Vujošević. 2013. Enhanced Savings Calculation and Its Applications for Solving Capacitated Vehicle Routing Problem. Applied Mathematics and Computation 219(20): 10302-10312.

[15] Toth, P., and D. Vigo. 2002. Models, Relaxations and Exact Approaches for the Capacitated Vehicle Routing Problem. Discrete Applied Mathematics 123(1-3): 487-512.

[16] Uchoa, E., D. Pecin, A. Pessoa, M. Poggi, A. Subramanian, and T. Vidal. 2014. New Benchmark Instances for the Capacitated Vehicle Routing Problem. Technical Report -- UFF, Rio de Janeiro, Brazil, Optimization Online : 2014-10-4597.

[17] Umitsu, R., and M. Fushimi. 2006. Shortest Path Problems for Ambulances in Case of Severe Earthquakes. The Sixth International Symposium on Operations Research and Its Applications (ISORA'06), pp. 283-291.